**VOT NUMBER: 79107**

# Development of Appointment Scheduling Agent Using Distributed Constraint Satisfaction (DisCS)

PEMBANGUNAN AGEN PENJADUALAN TEMUJANJI MENGGUNAKAN KEPUASAN KEKANGAN TERAGIH (DisCS)

**NURULHUDA FIRDAUS MOHD AZMI**

**ZETI DARLEENA ERI**

**NUZULHA KHILWANI IBRAHIM**

**ROZANA DIANA AHMAD RUSLI**

**RESEARCH VOTES NUMBER:**

**79107**

**Pusat Pengurusan Penyelidikan**

**Universiti Teknologi Malaysia**

**2008**

This document was created using SOLID CONVERTER PDF

To remove this message, purchase the product at www.SolidPDF.com

# DEDICATION

In the name of ALLAH the most merciful and compassionate. Salawat and salaam to our great prophet Muhammad SAW and his family and friends. May ALLAH bestow His blessing upon their souls. All the praises and gratefulness to ALLAH of because for His blessing and willing may this paper be presented today.

Our highest appreciation dedicated to the Ministry of Scien ce, Technology and Innovation (MOSTI) for the grant that was made available for us to carry out this study.

Our big thank you to Centre of Research Management, UTM for the support and guidance in managing this research grant.

To all the researchers, Pn Zeti Darleena bt Eri, Cik Nuzulha Khilwani Bt Ibrahim and Cik Rozana Diana Bt Ahmad Rusli, thank you very much for all the fruitfull discussions, your hard work, perseverance and the support in completing this study.

Last but not least, our gratitude goe s to all the staffs of Centre for Advanced Software Engineering (CASE) at UTM City Campus for providing us with all the facilities in ensuring the smoothness of our work.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# ABSTRACT

Development of Appointment Scheduling Agent Using Distributed Const raint Satisfaction (DisCS)

*(Keywords: Distributed Artificial Intelligence, Distributed Computing System (DCS), Distributed Constraint Satisfaction (DisCS), Service Operation Scheduling, Multiple Agent)*

Scheduling problem (that is assigning resources and t ime points to given tasks) arise in many real-world domains, for example reservation environment. A reservation environment typically refers to a parallel -machine with n jobs available for processing. The processing time of each job has to fit within a giv en time window and with n jobs available for processing. The goal of this research is to develop appointment scheduling agent in reservation environment with implementation of Distributed Constraint Satisfaction (DisCS). The aim of this development is to optimize the reservation schedule towards efficiency schedule. In the other hand, to improve a better operation system level in organization. In this study, several algorithms were produced for different types of appointment policies and constraints. This study follows the Rational Unified Process (RUP) methodology as software development life cycle (SDLC) in the software development process. The design of the system architecture is illustrated using Unified Modeling Language (UML) technique.

**Key Researchers:**

Nurulhuda Firdaus Mohd Azmi ( Project Leader)
Zeti Darleena Eri
Nuzulha Khilwani Ibrahim
Rozana Diana Ahmad Rusli

Email: huda@utm.my
Phone Num.: 03-26154901
Vote Num: 79107

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# ABSTRAK

Pembangunan Agen Penjadualan Temujanji Menggunakan Kepuasan Kekangan Teragih (DisCS)

*(Keywords: Distributed Artificial Intelligence, Distributed Computing System (DCS), Distributed Constraint Satisfaction (DisCS), Service Operation Scheduling, Multiple Agent)*

Masalah penjadualan didefinasikan sebagai kesulitan untuk menguruskan sumber dan masa untuk satu-satu tugasan. Ianya wujud pada berbagai situasi di tempat kerja, sebagai contoh dalam tugas penempahan. Suasana tugas penempahan biasanya merujuk kepada suatu alat pemprosesan yang selari dengan bilangan (n) tugasan yang perlu diproses. Masa pemprosesan untuk satu-satu tugasan haruslah menepati satu masa yang ditentukan supaya kesemua bilangan (n) tugasan yang ada dapat diproses dalam satu jumlah masa keseluruhan yang diperuntukan. Matlamat kajian ini adalah untuk membangunkan agen penjadualan temujanji untuk diguna pakai dalam suasana tugas penempahan dengan menggunakan kaedah Kepuasan Kekangan Teragih. Kajian ini bertujuan untuk mengoptimakan jadual penempahan kepada satu jadual yang lebih efisien, atau dalam erti kata lain, untuk memperbaiki dan meningkatkan tahap sistem operasi dalam organisasi. Dalam kajian ini beberapa algoritma telah dihasilkan untuk berbagai jenis polisi penempahan dan kekangannya. Kajian ini adalah menuruti metodologi RUP sebagai kitar hayat pembangunan perisian dalam proses pembangunan perisian. Rekabentuk sistem dibangunkan menggunakan teknik Unified Modeling Language (UML).

**Penyelidik Utama:**

Nurulhuda Firdaus Mohd Azmi (Ketua Projek)
Zeti Darleena Eri
Nuzulha Khilwani Ibrahim
Rozana Diana Ahmad Rusli

Emel: huda@utm.my
No. Tel.: 03-26154901
No. Vot: 79107

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# TABLE OF CONTENT

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## LIST OF TABLE

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## LIST OF FIGURES

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# ABBREVIATIONS

| | |
|---|---|
| ASRS | Appointment Scheduling for Reservation System |
| RUP | Rational Unified Process |
| SDLC | Software Development Life Cycle |
| UML | Unified Modeling Language |
| VIC | Vehicle Inspection Center |

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# LIST OF APPENDICES

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# CHAPTER 1

# INTRODUCTION

## 1.1 Introduction

Scheduling problem (that is assigning resources and time points to given tasks) arise in many real-world domains, for example reservation environment. A reservation environment typically refers to a parallel -machine with n jobs available for processing. The processing time of each job has to fit within a given time window and with n jobs available for processing. The processing time of each job has to fit within a given time window and there may or may not be any slack. The time window of job is specified by a release date rj and a due date dj.

It may be the case that is not possible to process all n jobs, and the scheduler has to decide which jobs to process. Several objectives may be considered, for example, maximizing the number of job processed or maximizing the total amount of processing. Recently, several researchers have applied agent technology to such problems especially in manufacturing and business process control. Because of the widely spread lack of theoretically foundation for this kind of collaborative problem solving, the constraint satisfaction/optimization community has recently extended the theoretically well -founded constraint satisfaction problem (CSP) model to a distributed constraint satisfaction problems (DCSP).

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

A distributed CSP is a constraint satisfaction problem in with variables and constraints are distributed among multiple agents. The variables are connected by constraints that define the constraints network among the agents.
As a result, the search algorithm for solving these problems is distributed algorithm. The algorithm is run by agents that communicate by sending and receiving messages. In general, messages contain information about assignments of values to variables and refutation of assignments, by agents that have no compatible assignments, to their own variables.

## 1.2 Research Objectives

The goal of this research is to develop appointment scheduling agent in reservation environment with implementation of Distributed Constraint Satisfaction (DisCS). In order to arrive this goal there are other objectives that need to be achieved, which are:

1. To develop applicable mathematical model based on Distributed Constraint Satisfaction (DisCS) in appointment schedule.
2. To design and develop system architecture of appointment scheduling agent.

## 1.3 Background of Case Study – Vehicle Inspection Centre

The number of vehicles undergoing inspection appears to be gradually increasing every year. The current workload based on 2004 statistics indicates that the VIC performs 3.2 million commercial tests per year with 750,000 vehicles on the road. Within this statistic, VIC outstations in the rural areas service some 5%, a mobile station which covers three locations, services about 2% (i.e. around 900 – 1000 vehicles per month for each location) and the rest are carried by VIC fixed centers. All the VIC's branches have their similar computerized inspection system. There are statistics of inspections for three highest inspection volumes by August 2004.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

| | | |
|---|---|---|
| i. | Wangsa Maju | : 17, 091 vehicles inspected |
| ii. | Padang Jawa | : 15, 613 vehicles inspected |
| iii. | Johor Bahru | : 15, 306 vehicles inspected |

The other VIC branches are Cheras, B.Maluri, Banting, Ipoh, Taiping, Teluk Intan, Mak Mandin, Teluk Kumbar, Sungai Petani, Alor Setar , Arau, Kuantan, Kota Bharu, Kuala Terengganu, Raub, Muar, Kluang, Alor Gajah, Seremban, Kota Kinabalu, Tawau, Sandakan, Kuching, Sibu, Miri, Labuan, Pelabuhan Klang, Kuala Krai and Kijal. Total inspections from all the centers for August 2004 is 177, 950 vehicles.

Generally, the inspection services offered at the VIC are as follows:

i. Initial Inspection : Inspection to determine vehicle status before registration with the Road Transport Department, or before transfer of ownership for commercial vehicles.

ii. Routine Inspection : Routine half-yearly checks to gauge roadworthiness of commercial vehicles and ensure compliance with Malaysia Construction and Use Rules 1959.

iii. Re-Inspection : To be conducted after a vehicle fails initial or routine inspection.

iv. Special Inspection : To determine roadworthiness of modified vehicles, as well as verification of imported vehicles.

v. Accident Inspection : In aid of police investigations of fatal accidents involving vehicles.

There are 10 points of vehicle inspection in the centre and it is defined as follow:

i. Identification check – verifies the engine and chassis number of the vehicle to be tested.

ii. Above carriage check – the vehicle is then scrutinized for surface defects and non-compliance with road transport act 1987.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

iii. Emission test – density of particles in diesel engines and volume of gases eg HC, CO etc in petrol engines are gauged through free acceleration tests.

iv. Side-slip test – lateral movements of the vehicle are tested for optimum road-handling criteria.

v. Suspension test – checks the effectiveness of the suspension system (springs, shock absorber, and joints) on each axle.

vi. Brake test – the various brake efficiencies, such as service brake efficiency, dynamic unbalance, run-out, residual force and parking brake are evaluated for maximum performance.

vii. Speedometer – checks the road speed at the wheels against the value indicated on the vehicle's speedometer.

viii. Headlight test – measures the intensity and projection of the vehicle's headlights

ix. Undercarriage check – a vehicle's undercarriage is thoroughly scrutinized for defects while it is tested on an axle play detector.

x. Computerized analysis – a state-of-the-art central computer controls and monitors each test phase. All relevant test data is then analyzed and the final result printed out.

### 1.3.1 Johor Bahru Vehicle Inspection Centre

A VIC in the southern part of peninsular Malaysia, named VIC Johor Bahru (JB) has been selected as a case study for this research. It was identified by the VIC top management as one of the busiest centres and is known to have the highest level of operational problems. VIC JB is situated in an industrial area with a high volume of vehicles coming for inspection. On average, the number of inspected vehicles in VIC JB is 400 per day and is increase by an average of 5% every year. VIC JB is led by a Centre Manager who is assisted by two supervisors and a quality control officer. There are currently a total of 23 vehicle examiners at this site.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

The size of the VIC premises is about 7.2 $km^2$ (i.e. 120 metres X 60 metres) and is an open area. The layout of VIC JB is as in **Figure 1.1**. Queuing areas for light and heavy vehicles are located separately, except for initial inspections. The separation is to ease movement and for safety purposes. As an addition, separa te queuing areas for light and heavy vehicles are the reason of why the VIC has two entrances and one exit. Two entrances, each for light and heavy vehicle are designed at two different locations as shown in **Figure 1.1**, each of which located near its queue lane. This is to enable smooth flow for both types of vehicles.

For initial inspections, these two types of vehicles are combined in a single queue because initially they share the same equipment (i.e. weighing machine). Queuing areas for light and heavy vehicles at other VIC's are usually adjacent to each other. There are also two separate traffic wardens at each light and heavy queuing area. This shows that two numbering systems, exist for a different type of vehicle.

There are six inspection lanes wi th three light and three heavy lanes. Currently lanes 1 and 2 have been allocated for finance vehicles, lane 3 for light vehicles and lanes 4 -6 are for heavy vehicles.

Generally, the layout of a VIC affects its inspection flow. At VIC JB, as shown in **Figure 1.1**, we can see that there are four different flows, indicated by four different arrows. Each arrow represents a different type of inspection which flows through the system, throughout the inspection process. Private and accident inspections are loc ated separately from other inspections to speed up the inspection process.

Generally, the sequence of machines in an inspection lane in VIC JB is similar to other VIC's, with a slight difference. **Figure 1.2** shows the sequence of inspection machines located on one of the light inspection lanes and on one of the heavy inspection lanes. Both lanes are preceded by a smoke meter and end with a printer to generate results. In other centers like Wangsa Maju, a printer is shared among other lanes and is located at

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

one place. When a vehicle has completed its inspection, every driver will go to the printer to obtain the corresponding result.

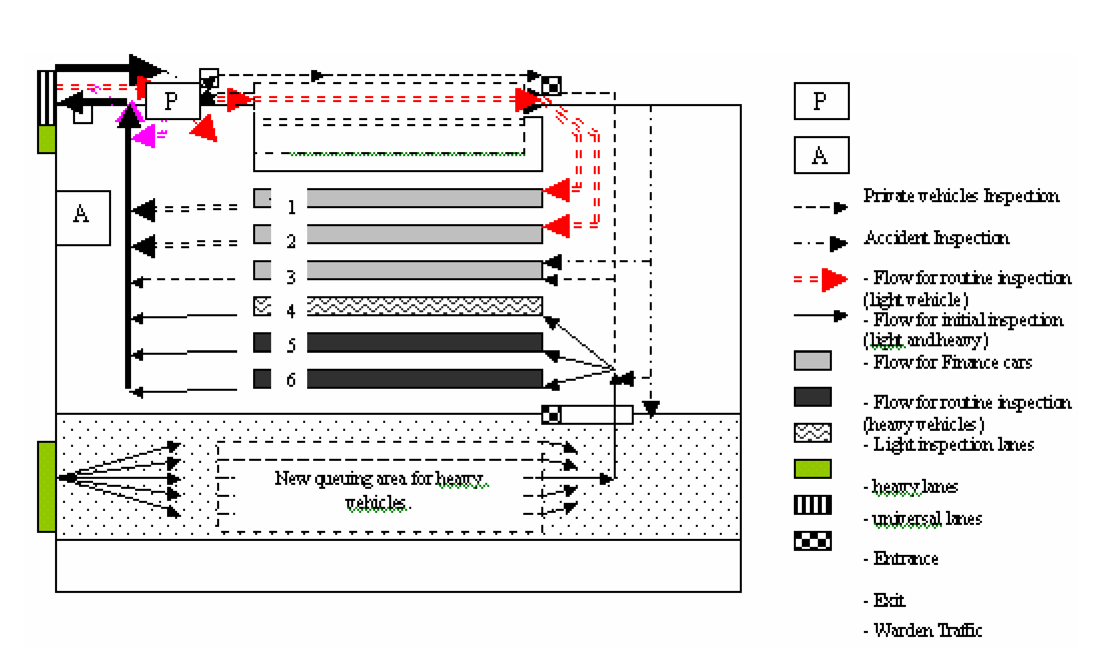

P
A
1
2
3
4
5
6
New queuing area for heavy vehicles.
P
A
Private vehicles Inspection
Accident Inspection
- Flow for routine inspection (light vehicle)
- Flow for initial inspection (light and heavy)
- Flow for Finance cars
- Flow for routine inspection (heavy vehicles)
- Light inspection lanes
- heavy lanes
- universal lanes
- Entrance
- Exit
- Warden Traffic

**Figure 1.1** : Layout of VIC JB

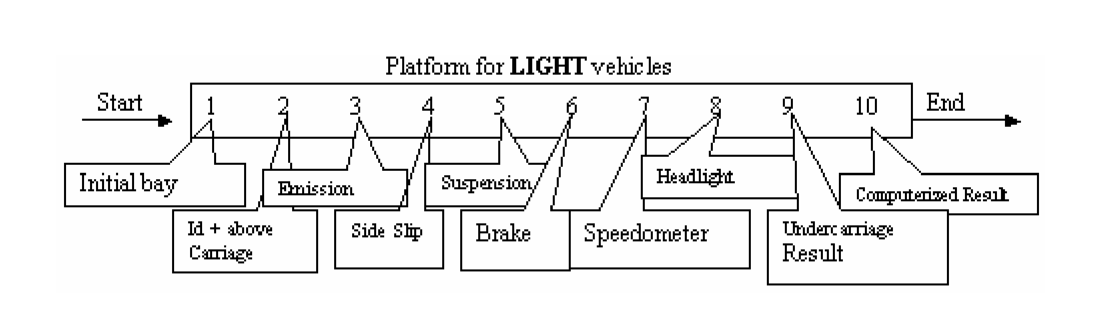


**Figure 1.2** : Sequence of Inspection Machines Located on Li ght Inspection Lane

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

### 1.3.2 Domain Problem – Finance Car Inspection Lane

An inspection flow is different from an inspection procedure as it describes the activity that takes place on the day of inspection. The flow of inspection is slightly different from one VIC to another due to their unique physical layouts. Generally, upon arrival at a VIC, the vehicle owner or driver first shows the security guard related documents as proof of identification of his vehicle and himself. This allows him to enter the VIC premises. Only a vehicle with the correct appointment date and time is permitted to enter the VIC. This is to prevent the premises from becoming over crowded.

The vehicle owner then has to drive the vehicle to a specific queuing area, depending on the required type of inspection. He then has to obtain a queue number from a traffic warden. At the VIC, a numbering system is implemented as there are more than one inspection lane and a sequence of customers keep on coming.

While queuing and waiting to be inspected in the lane, vehicle owner makes a payment as his vehicle approaches the payment counter. When his number is called, his vehicle is passed on to a Vehicle Examiner who then takes the vehicle to the inspection lane and the inspection process starts. After inspection is completed, the result is printed and given to the vehicle owner. Actions regarding the result are then taken by the vehicle owner as described earlier. See **Figure 1.3** for the general procedure in Finance Car Inspection Lane.

The constraints for finance company vehicles are:

i. Number of lane considered is one.
ii. No. of inspection considered is ten.
iii. Rule of inspection is that if all three major items passed, the result is pass.
iv. Number of major item is three.
v. The three major items named chassis number, engine number and original body.
vi. Maximum number of vehicle examiner is four.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

vii. Time stagger is one hour (but still not been used in this analysis).

viii. Time slot is assumed as complete time with waiting time at initial bay.

ix. Complete time is assumed as processing time with movement period and reserve time.

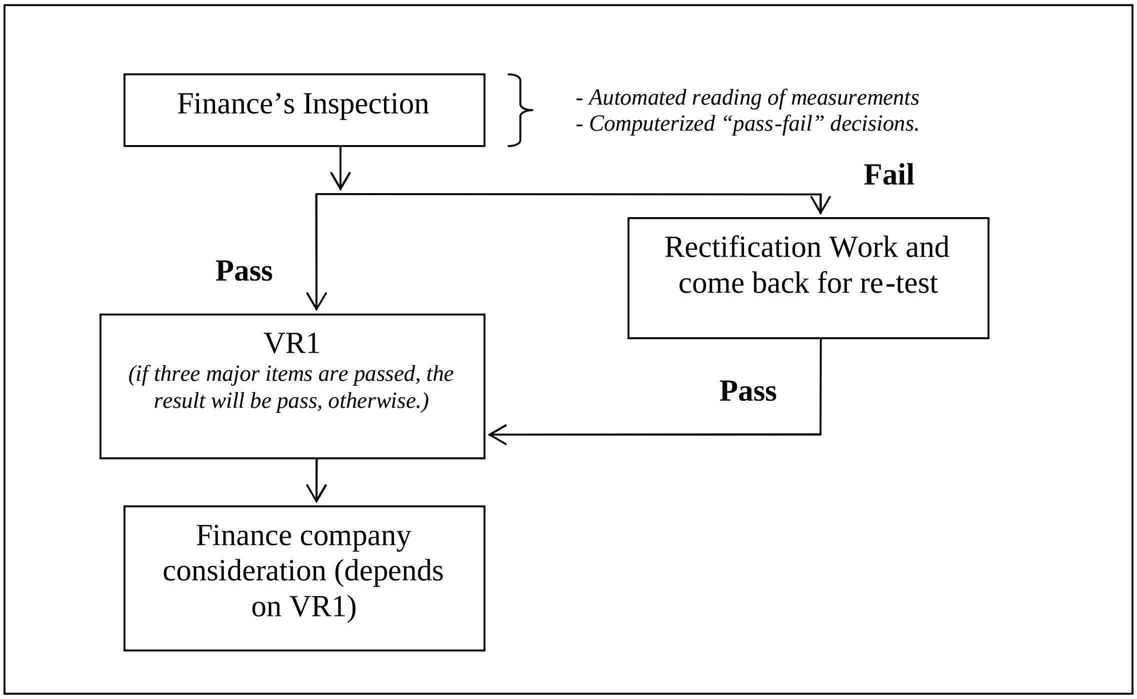

Finance's Inspection
- Automated reading of measurements
- Computerized "pass-fail" decisions.
Fail
Pass
Rectification Work and come back for re-test
VR1
(if three major items are passed, the result will be pass, otherwise.)
Pass
Finance company consideration (depends on VR1)

**Figure 1.3**: General procedure for a finance car inspection

x. Operation hour in this study is defined from 8 a.m. until 5 p.m.

xi. There is no tea break defined in this study. It is based on the real operation at the time of data been collected.

xii. There is no lunch hour defined in this study. It is also based on the real operation at the time of data been collected.

xiii. Each vehicle must have a valid time slot of appointment to be inspected. But, if there was any vehicle that came at the wrong date and time, it was also been inspected.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

## 1.4 Vehicle Inspection Appointment Schedule

The date and time for a routine inspection is fixed by an appointment system. It is set during the previous routine inspection by a computerized vehicle appointment scheduling system. The system generates a specific inspection date and time for a particular vehicle. A system known as “staggered system” has been applied at the VIC and has significantly reduced congestion at the premises. The staggered system schedules the vehicles’ arrival times on an hourly basis. Vehicles should arrive at the allotted times, otherwise they will not be allowed to enter the premises. However the system has been identified as possessing several weaknesses, such as being very rigid, and caters for routine inspections only.

The system does not take into account constraints like holidays. **Figure 1.4** shows the number of vehicles, scheduled for inspection at VIC JB, each day in De cember 2004. The x axis represents the date and note that 1 st is on Monday, 2nd is on Tuesday and so on. Sundays fell on 7th, 14th, 21st and 28th December. As noted in figure 6 there are vehicles which were scheduled for inspection on weekends.

To overcome this problem, manual adjustments are made by the Center Manager. **Figure 1.5** shows the numbers of vehicles which have been re -scheduled via “manual adjustment”. From the figure, we can see that vehicles previously scheduled for a Sunday have mostly been shifted to the next working day.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

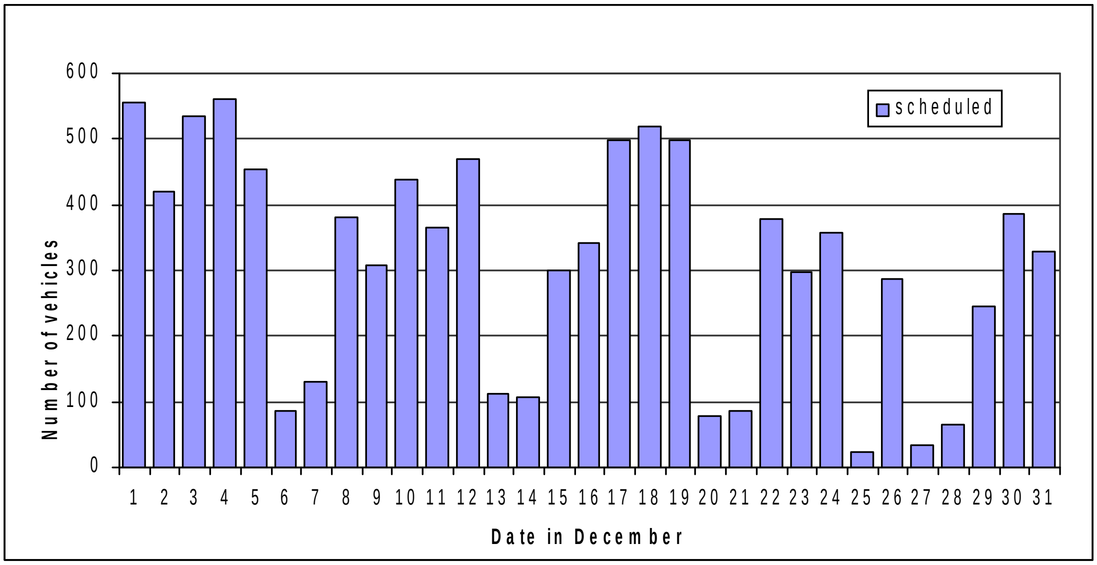


**Figure 1.4** : Number of Vehicles Scheduled in December

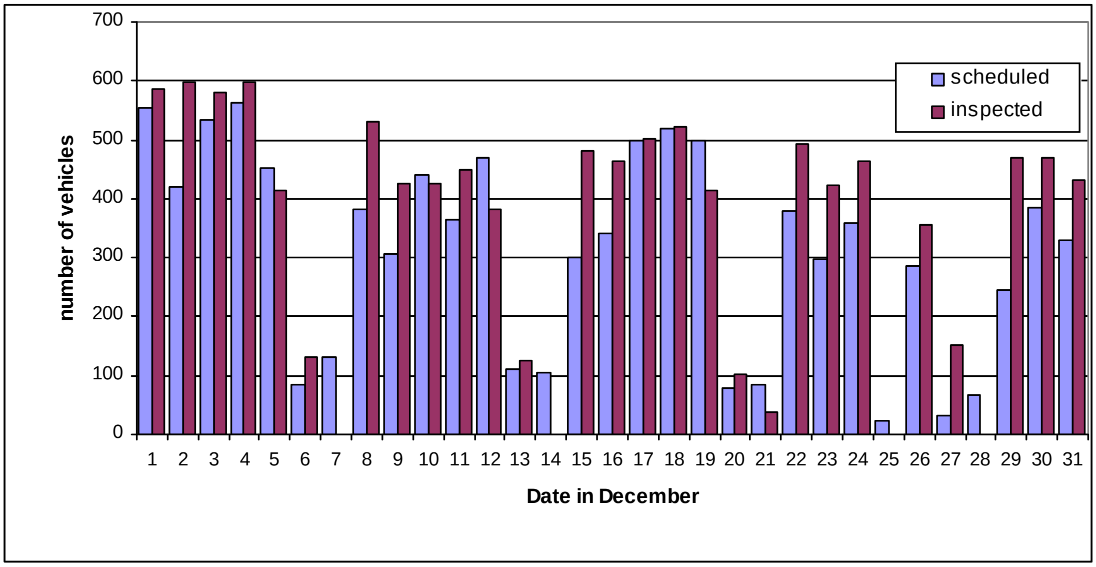


**Figure 1.5 :** Number of Vehicles Scheduled in December (after manual adjustment)

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

Another factor which makes the system rigid is that it does not consider the maximum capacity of vehicles that can be inspected per hour. The present staggered system sets the next routine inspection time (which is scheduled for exactly six months after the issue of a pass certificate) for a particular vehicle, according to the vehicle's current completion time of inspection. The weaknesses of the vehicle scheduling system at the VIC have some implications on the variability of daily demand, accuracy of vehicles' inspection schedule and workload planning.

Variability in daily demand is created by "manual adjustment" process and "drive-in" vehicles. "Manual adjustment" activity has largely caused the number of vehicles scheduled on specific days to be shifted to the next working days. Hence, the number of vehicles to be inspected on that working day increases. This situation is worsened by "drive-in" vehicles.

When a manual adjustment is done, the altered inspection schedule becomes packed. The situation is worsened by the arrival of drive-in vehicles which is quite high. A schedule which continues to get packed will have smaller inter-arrival times. A smaller inter arrival compared to a constant service time would result in longer waiting times.

Apart from increasing waiting times, the weaknesses of the vehicle scheduling system at the VIC have resulted in difficulties of determine the actual workload for a specific vehicle examiner, to be done beforehand. This has disabled the planning of an effective staff workload which finally creates a situation of "a fixed workforce handling variable demand".

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

# CHAPTER 2

# LITERATURE REVIEW

## 2.1 Distributed Artificial Intelligence (DAI)

Distributed artificial intelligence (DAI) was a subfield of Artificial intelligence research dedicated to the development of distributed solutions for complex problems regarded as requiring intelligence. These days DAI has been largely supplanted by the field of Multi - Agent Systems. See the paper by Inman and Hewitt on some of the limitations of classic DAI.

### ***Background***

Main streams in DAI research i ncluded the following:

- Parallel problem solving: mainly deals with how classic AI concepts can be modified, so that multiprocessor systems and clusters of computers can be used to speed up calculation.
- Distributed problem solving (DPS): the concept of agen t, autonomous entities that can communicate with each other, was developed to serve as an abstraction for developing DPS systems. See below for further details.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

- Multi-Agent Based Simulation (MABS): a branch of DAI that builds the foundation for simulations that need to analyze not only phenomena at macro level but also at micro level, as it is in many social simulation scenarios.

## 2.2 Complexity of Constraints Satisfaction

The computational complexity of constraint satisfaction has mainly been studied for discriminating between tractable and intractable classes of constraint satisfaction problems on finite domains.

Solving a constraint satisfaction problem on a finite domain is an NP -complete problem in general. Research has shown a number of polynomial -time sub cases, mostly obtained by restricting either the allowed domains or constraints or the way constraints can be placed over the variables. Research has also established relationship of the constraint satisfaction problem with problems in other areas such as finite model theory and databases.

### *Overview*

Establishing whether a constraint satisfaction problem on a finite domain has solutions is an NP complete problem in general. This is an easy consequence of a number of other NP complete problems being expressi ble as constraint satisfaction problems. Such other problems include propositional satisfiability and three -colorability.

Tractability can be obtained by considering specific classes of constraint satisfaction problems. As an example, if the domain is bina ry and all constraints are binary, establishing satisfiability is a polynomial -time problem because this problem is equivalent to 2-SAT, which is tractable. Research has shown a number of tractable sub cases. Some of these classes are based on restricting the allowed domains or relations, some on restricting the way constraints are placed over variables, and some on both kinds of restrictions.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

One line of research used a correspondence between constraint satisfaction problem and the problem of establishing the existence of a homomorphism between two relational structures. This correspondence has been used to link constraint satisfaction with topics traditionally related to database theory.

A considered research problem is about the existence of dichotomies a mong sets of restrictions. This is the question of whether a set of restrictions contains only polynomial - time restrictions and NP-complete restrictions. This question is settled for some sets of restrictions, but still open for the set of all restrictions based on a fixed domain and set of relations, as of 2005. This is considered by some authors the most important open question about the complexity of constraint satisfaction.

***Restrictions***

Tractable subcases of the general constraint satisfaction problems can be obtained by placing suitable restrictions on the problems. Various kinds of restrictions have been considered.

### 2.2.1 Structural and relational restrictions

Tractabiliy can be obtained by restricting the possible domains or constraints. In particular, two kinds of restrictions have been considered:

- *relational restrictions* bounds the domain and the values satisfying the constraints;
- *structural restrictions* bounds the way constraints are distributed over the variables.

More precisely, a relational restriction specifies a *constraint language*, which is a domain and a set of relations over this domain. A constraint satisfaction problem meets this restriction if it has exactly this domain and the relation of each constraint is in the given set of relations. In other words, a relational restriction bounds the domain and the set of satisfying values of each constraint, but not how the constraints are placed over variables.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

This is instead done by structural restrictions. Structural restriction can be checked by looking only at the scopes of constraints (their variables), ignoring their relations (their set of satisfying values).

A constraint language is tractable if there exists a polynomial algorithm solving all problems based on the language, that is, using th e domain and relations specified in the domain. An example of a tractable constraint language is that of binary domains and binary constraints. Formally, this restriction corresponds to allowing only domains of size 2 and only constraints whose relation is a binary relation. While the second fact implies that the scopes of the constraints are binary, this is not a structural restriction because it does not forbid any constraint to be placed on an arbitrary pair of variables. Incidentally, the problem becomes NP complete if either restriction is lifted: binary constraints and ternary domains can express the graph coloring problem, while ternary constraints and binary domains can express 3-SAT; these two problems are both NP-complete.

An example of a tractable class defined in terms of a structural restriction is that of binary acyclic problems. Given a constraint satisfaction problem with only binary constraints, its associated graph has a vertex for every variable and an edge for every constraint; two vertices are joined if they are in a constraint. If the graph of a problem is acyclic, the problem is called acyclic as well. The problem of satisfiability on the class of binary ayclic problem is tractable. This is a structural restriction because it does not pl ace any limit to the domain or to the specific values that satisfy a constraints; rather, it restricts the way constraints are placed over variables.

While relational and structural restrictions are the ones mostly used to derive tracable classes of constraint sastisfaction, there are some tractable classes that cannot be defined by relational restrictions only or structural restrictions only. The tractable class defined in terms of row convexity cannot be defined only in terms of the relations or only in t erms of the structure, as row convexity depends both on the relations and on the order of variables (and therefore cannot be checked by looking only at each constraint in turn).

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

### 2.2.2 Uniform and non-uniform restrictions

The subcase obtained by restricting to a finite constraint language is called a *non-uniform problem*. These problems are mostly considered when expressing constraint satisfaction in terms of the homomorphism problem, as explained below. *Uniform problems* were also defined in the settings of homomorphism problems; a uniform problem can be defined as the union of a (possibly infinite) collection of non-uniform problems. A uniform problem made of an infinite set of non-uniform problems can be intractable even if all these non-uniform problems are tractable.

### 2.2.3 Tree-based restrictions

Some considered restrictions are based on the tractability of the constraint satisfaction problem where the constraints are all binary and form a tree over the variables. This is a structural restriction, as it can be checked by looking only at the scopes of the constraints, ignoring domains and relations.

This restriction is based on primal graph of the problem, which is a graph whose vertices are the variables of the problem and the edges represent the presence of a constraint between two variables. Tractability can however also be obtained by placing the condition of being a tree to the primal graph of problems that are reformulations of the original one.

***Equivalence conditions***

Constraint satisfaction problems can be reformulated in terms of other problems, leading to equivalent conditions to tractability. The most used reformulation is that in terms of the homomorphism problem.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 2.3 Constraints satisfaction and the homomorphism problem

A link between constraint satisfaction and database theory has been provided in the form of a correspondence between the problems of constraint satisfiability to the problem of checking whether there exists a homomorphism between two relational structures. A relational structure is a mathematical representation of a database: it is a set of values and a set of relations over these values. Formally, $A = \left(V, R_1^A, \ldots\ldots\ldots, R_n^A\right)$, where each $R_i^A$ is a relation over *V*, that is, a set of tuples of values of *V*.

A relational structure is different from a constraint satisfaction problem because a constraint is a relation *and* a tuple of variables. Also different is the way in which they are used: for a constraint satisfaction problem, finding a satisfying assignment is the main problem; for a relation structure, the main problem is finding the answer to a query.

The constraint satisfaction problem is however related to the problem of establishing the existence of a homomorphism between two relational structures. A homomorphism is a function from the values of the first relation to the values of the second that, when applied to all values of a relation of the first structure, turns it into a subset of the corresponding relation of the second structure. Formally, *h* is a homomorphism from $A = \left(V, R_1^A, \ldots\ldots\ldots, R_n^A\right)$ to $B = \left(D, R_1^B, \ldots\ldots\ldots, R_n^B\right)$ if it is a function from *V* to *D* such that, if $\left(x_1, \ldots\ldots, x_k\right) \in R_i^A$ then $\left(h\left(x_1\right), \ldots\ldots\ldots, h\left(x_k\right)\right) \in R_i^B$.

A direct correspondence between the constraint satisfaction problem and the homomorphism problem can be established. For a given constraint satisfaction problem, one can build a pair of relational structures, the first encoding the variables and the signatures of constraints, the second encoding the domains and the relations of the constraints. Satisfiability of the constraint satisfaction problem corresponds to finding a value for every variable such that replacing a value in a signature makes it a tuple in the relation of the constraint. This is possible exactly if this evaluation is a homomorphism between the two relational structures.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

The inverse correspondence is the opposite one: given two relational structures, one encodes the values of the first in the variables of a constraint satisfaction problem, and the values of the second in the domain of the same p roblem. For every tuple of every relation of the first structure, there is a constraint having as values the correspondent relation of the first structure. This way, a homomorphism corresponds to mapping every scope of every constraint (every tuple of ever y relation of the first structure) into a tuple in the relation of the constraint (a tuple in the corresponding relation of the second structure).

A non-uniform constraint satisfaction problem is a restriction where the second structure of the homomorphism problem is fixed. In other words, every relational structure defines a non-uniform problem, that of telling whether a relation structure is homomorphic to it. A similar restriction can be placed on the first structure; for any fixed first structure, the homomorphism problem is tractable. A uniform constraint satisfaction problem is an arbitrary restriction to the sets of structures for the first and second structure of the homomorphism problem.

### 2.3.1 Conjunctive query evaluation and containment

Since the homomorphism problem is equivalent to conjunctive query evaluation and conjunctive query containment, these problems below are equivalent to constraint satisfaction as well.

#### i) Join evaluation

Every constraint can be viewed as a table in a database, where the variables are interpreted as attributes names and the relation is the set of records in the table. The solutions of a constraint satisfaction problem are the result of an inner join of the tables representing its constraints; therefore, the problem of existence of solutions can be reformulated as the problem of checking whether the result of an inner join of a number of tables is empty.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## ***Dichotomy theorems***

Some constraint languages (or non-uniform problems) are known to be polynomial-time, and some others are known to be NP-complete. However, it is possible that some constraint languages are neither. It is indeed known by Ladner's theorem that, if P is not equal to NP, there exists problems in NP that are neither polynomial-time nor NP-hard. As of 2006, it is not known if such problems include a constraint language. If this is not the case, the set of all constraint languages could be divided exactly in the set of polynomial-time and NP-complete problems, that is, this set has a dichotomy.

Some partial results are however known for some sets of constraint languages. The most known such result is Schaefer's dichotomy theorem, which proves the existence of a dichotomy in the set of constraint languages on a binary domain. More precisely, it proves that a relation restriction on a binary domain is tractable if all its relations belong to one of six classes, and is NP-complete otherwise. Bulatov proved a dichotomy theorem for domains of three elements.

Another dichotomy theorem for a subset of the constraint languages is the Hell-Nesetril theorem, which shows a dichotomy for problems on binary constraints having all the same fixed symmetric relation. In terms of the homomorphism problem, every such problem is equivalent to the existence of an homomorphism from a relational structure to a given fixed undirected graph (an undirected graph is a relational structure with a single binary symmetric relation). The Hell-Nesetril theorem proves that every such problem is either polynomial-time or NP-complete. More precisely, the problem is polynomial-time if the graph is 2-colorable, that is, it is bipartite, and is NP-complete otherwise.

## ***Sufficient conditions for tractability***

Some complexity results prove that some restrictions are polynomial without giving proving that all other possible restrictions of the same kind are NP-hard. A sufficient condition for tractability is related to expressible in Datalog. A Boolean Datalog query

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

gives a truth value to a set of literals over a given alphabet, each literal being an expression of the form $L(a_1,.......,a_n)$; as a result, a Boolean Datalog query expresses a set of sets of literals, as it can be considered semantically equivalent to the set of all sets of literals that it evaluates to true.

On the other hand, a non-uniform problem can be seen as a way for expressing a similar set. For a given non-uniform problem, the set of relations that can be used in constraints is fixed; as a result, one can give unique names $R_1,.....,R_n$ to them. An instance of this non-uniform problem can be then written as a set of literals of the form $R_i(x_1,....,x_n)$. Among these instances/sets of literals, some are satisfiable and some are not; whether a set of literals is satisfiable depends on the relations, which are specified by the non-uniform problem. In the other way around, a non-uniform problem tells which sets of literals represent satisfiable instances and which ones represent unsatisfiable instances. Once relations are named, a non-uniform problem expresses a set of sets of literals: those associated to satisfiable (or unsatisfiable) instances.

A sufficient condition of tractability is that a non-uniform problem is tractable if the set of its unsatisfiable instances can be expressed by a Boolean Datalog query. In other words, if the set of sets of literals that represent unsatisfiable instances of the non-uniform problem is also the set of sets of literals that satisfy a Boolean Datalog query, then the non-uniform problem is tractable.

### ii) Local consistency

Satisfiability can sometimes be established by enforcing a form of local consistency and then checking the existence of an empty domain or constraint relation. This is in general a correct but incomplete unsatisfiability algorithm: a problem may be unsatisfiable even if no empty domain or constraint relation is produced. For some forms of local consistency, this algorithm may also require exponential time. However, for some problems and for some kinds of local consistency, it is correct and polynomial-time.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

The following conditions exploit the primal graph of the problem, which has a vertex for each variable and an edge between two nodes if the corresponding variables are in a constraint. The following are conditions on binary constraint satisfaction problems where enforcing local consistency is tractable and allows establishing satisfiability:

1. enforcing arc consistency, if the primal graph is acyclic
2. enforcing directional arc consistency for an ordering of the variables that makes the ordered graph of constraint having w idth 1 (such an ordering exists if and only if the primal graph is a tree, but not all orderi ngs of a tree generate width 1)
3. enforcing strong directional path consistency for an ordering of the variables that makes the primal graph having induced width 2.

A condition that extends the last one holds for non -binary constraint satisfaction problems. Namely, for all problems for which there exists an ordering that makes the primal graph having induced width bounded by a constant i, enforcing strong directional i-consistency is tractable and allows establishing satisfiability.

### iii) Tree-based conditions

Constraint satisfaction problems composed of binary constraints only can be viewed as graphs, where the vertices are variables and the edges represent the presen ce of a constraint between two variables. This graph is called the Gaifman graph or primal graph of the problem.

If the primal graph of a problem is acyclic, establishing satisfiability of the problem is a tractable problem. This is a structural restrictio n, as it can be checked by looking only at the scopes of the constraints, disregarding their relations and the domain. An acyclic graph is a forest, but connectedness is usually assumed; as a result, the condition that is usually considered is that primal graphs are trees.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

This property of tree-like constraint satisfaction problems is exploited by decomposition methods, which convert problems into equivalent ones that only contain binary constraints arranged as a tree. The variables of these problems correspond to sets of variables of the original problem; the domain of such a variable is obtained by considering some of the constraints of the original problem whose scope is contained in the corresponding original set of variables; constraints of these new problems represent equality of variables that are contained in two sets.

If the graph of one such equivalent problem is a tree, the problem can be solved efficiently. On the other hand, producing one such equivalent problem may be not be efficient because of two factors: the need to determine the combined effects of a group of constraints over a set of variables, and the need to store all tuples of values that satisfy a given group of constraints.

### ***Necessary condition for tractability***

A necessary condition for the tractability of a constraint language based on the universal gadget has been proved. The universal gadget is a particular constraint satisfaction problem that was initially defined for the sake of expressing new relations by projection.

## 2.4 Distributed Constraints Satisfaction Problem (DisCSP)

DisCSP is a framework for describing a problem in terms of constraints that are known and enforced by distinct participants (agents). The framework was used under different names in the 1980's. The first known usage with the current name is in 1990. The constraints are described on some variables with predefined domains, and have to be assigned to the same values by the different agents. Problems defined with this framework can be solved by any of the algorithms that are proposed for it.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

The currently most well known paper is the one which introduced in 1992 as asynchronous complete algorithm for solving such problems is Asynchronous backtracking (ABT). A large number of such techniques are now available, See **Table 2.1** below.

| Year | Technique | Abbreviation | Characteristics |
|---|---|---|---|
| 1992 | Asynchronous Backtracking | ABT | Static ordering, complete search |
| 1994 | Asynchronous Weak-Commitment | AWC | Reordering, fast, complete search |
| 1995 | Distributed Breakout Algorithm | DBA | Incomplete but fast |
| 2000 | Distributed Forward Chaining | DFC | Slow, comparable to ABT |
| 2000 | Asynchronous Aggregation Search | AAS | Aggregation of values in ABT |
| 2001 | Maintaining Asynchronously Consistencies | DMAC | The fastest algorithm |
| 2001 | Asynchronous Backtracking with Reordering | ABTR | Reordering in ABT with bounded nogoods |
| 2002 | Secure Computation with Semi-Trusted Servers | - | Security increases with the number of trustworthy servers |
| 2003 | Secure Multiparty Computation for Solving DisCSPs | MPC-DisCSP1-MPC-DisCSP4 | Secure if half (½) of the participants are trustworthy. |

**Table 2.1:** Distributed Constraints Satisfaction (DisCS) Techniques

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 2.5 Agent Technology

***What is Agent***

Agent is anyone who acts on behalf or in the interest of somebody else. So most agent are designed to act as or for the user to help execute some task or operation. Agents come in various form for various purpose. Here, an agent can be a human being, a robot, or a computer program. Consequently, a more precise and very general definition can be provided for agents is it can do routine work for users or assist them with complicated and finished their tasks.

In the internet definition, it defines that an agent (also called an intelligent agent and sometime called a bot) is a program that gathers information or performs some other service without your immediate presence and on some standard schedule. Typically, an agent is an entity that is able to carry out some task usually to help human user. Agent can be biologic, robotic or computational. In general we can describe the agent as referring to a component of software or hardware which is able of acting in order to accomplish tasks on behalf of its user.

***Software Agent***

A software agent is a computer program (more to software) that can accept tasks from its human user can figure out which actions to perform in order to solve these tasks and can actually perform these task actions without user aided.

### 2.5.1 Agent Properties

The behavior of the agent must have at least the following properties: -

> ***Autonomy*****:** Agent can operate on their own without the need for human guidance and it also called as proactive because these agents can take the initiative and make decision independently related to control; although an agent may interact

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

with its environment, the processes performed by an agent are in full control of the agent itself.

***Reactivity*****:** leading the agent to rapidly respond to new information perceived from its environment. Reactivit y indicates that the agent will respond to perceived changes in the environment and able to handle changing environment.

***Proactively*****:** agents should not simply act in response to their environment; they should be able to exhibit opportunistic, goal -directed behavior and take their initiative where appropriate. In situation where the environment changes or the goal achievement fails, the agent will still to find way to reach the goal

***Learning:*** React and interact with their external environment, from observ ation or by user's instructions.

***Flexible*****:** Agents have flexible behavior, because they can achieve in several ways. To reach a goal an agent chooses a plan to follow. If the plan fails, the agent can choose between other possibilities to obtain the desire d result.

***Social ability / Cooperation:*** is the central property in *multi*-agent systems. Agents must be able to communicate and cooperate with other agents and humans to complete their own problem solving and help others with their activities. It also called as cooperation, which is the ability to interact with other agent, cooperate with each others and interaction with possible human via some communication language.

Other non-mandatory properties of agents include ***adaptability*****,** ***mobility*****, and** ***intelligence***. Although the term software agent is sometimes used as a synonym for intelligent software systems, software agents do not have to be intelligent. In software engineering, for instance, the ability of an agent to communicate and cooperate with other systems in a flexible manner, and the ability of a mobile agent to migrate to another

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

computer providing resources via suitable network links are considered more fundamental than any form of intelligence.

### 2.5.2 How they applied this technology in a system

Agents propose new possibilities and solution to answer the problem suitable to their properties and characteristics. Some of the characteristic of agent is know the preferences of their user. The efficiency of the scheduling process used and the quality of the schedule generated, then affects the working of an organization to a large extends. The used of agent can developed more efficient schedule because it can save time and effort on the part of human. There is also a communication between agent and user. The age nts manage the scheduling process on behalf the individual user, without human interaction. Each agent can access to calendar and preference information of its user.

Generally, agent has initial information about schedule the meeting. That means, the agents have to learn and get to know the habit and preference of user by observing the action of the user. Some of the actions are marking suggestion, collecting or receive feedback either positive or negative and the last is suggestion from the other user. So from this way, the user comes to trust the agent and decide to delegate their meeting scheduling decision. In this approach, agent also provide autonomous scheduling capabilities between intelligent agent, where these agent use knowledge about the priorities of people to perform their task such as automatically screening, directing and even responding to information.

The meeting to be schedule must have a day, a start time and place, and the problem is solve when all agent reach an agreement on values fo r these attributes. Agents negotiate by having one agent propose a meeting. The agent who initiates the new meeting called as host agent. While the meeting is start, the other attendee or invitee agent have ability to make decision on behalf of their user to respond the invitation by accepting or rejecting them, based on weather or not it fit their own user's schedules. User will be supervising a

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

group of agent such as in their activity and monitoring for any change in the schedule. Then agent predicts a response and negotiates with other agent to suggest an optimal time for the meeting.

## 2.6 Intelligent Agent

The terms "agent" and "intelligent agent" are ambiguous and have been used in two different, but related senses, which are often confused.

In computer science, an intelligent agent (IA) is a software agent that assists users and will act on their behalf, in performing non-repetitive computer-related tasks. An agent in the sense of the word is like an insurance agent or travel agent. While the working of software agents used for operator assistance or data mining (sometimes referred to as bots) is often based on fixed pre-programmed rules, "intelligent" in this context is often taken to imply the ability to adapt and learn.

In artificial intelligence, the same term is used for intelligent actors, which observe and act upon an environment, to distinguish them from intelligent thinkers isolated from the world. An agent in this sense of the word is an entity that is capable of perception and action. Such an agent might be a robot or an embedded real time software system - and is intelligent if it interacts with its environment in a manner that would normally be regarded as intelligent if that interaction were carried out by a human being.

There is no reason why these two notions of intelligent agent need to be related. An intelligent agent in the first sense might be implemented using conventional software techniques and display no more intelligence than a conventional computer program. On the other hand, an intelligent agent in the second sense might be wholly autonomous, carrying out its own agenda, and acting as an agent for no one.

In the artificial intelligence sense of the term, there are multiple types of agents and sub-agents. For example:

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

1. Physical Agents - A physical agent is an entity which *percepts* through sensors and *acts* through actuators.
2. Temporal Agents - A temporal agent may use time based stored information to offer instructions or data *acts* to a computer program or human being and takes program inputs *percepts* to adjust its next behaviors.

A simple agent program can be defined mathematically as an agent function which maps every possible percepts sequence to a possible action the agent can perform or to a coefficient, feedback element, function or c onstant that affects eventual actions:

$$f : P* -> A$$

The program agent, instead, maps every possible percept to an action.

It is possible to group agents into five classes based on their degree of perceived intelligence and capability:

1. simple reflex agents;
2. model-based reflex agents;
3. goal-based agents;
4. utility-based agents;
5. learning agents.

### 1. Simple reflex agents

Simple reflex agents act only on the basis of the current percept. The agent function is based on the condition-action rule:

*if condition then acti on rule*

This agent function only succeeds when the environment is fully observable. Some reflex agents can also contain information on their current state which allows them to disregard conditions whose actuators are already triggered.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

### 2. Model-based reflex agents

Model-based agents can handle partially observable environments. Its current state is stored inside the agent maintaining some kind of structure which describes the part of the world which cannot be seen. This behavior requires information on how the world behaves and works. This additional information completes the “World View” model.

### 3. Goal-based agents

Goal-based agents are model-based agents which store information regarding situations those are desirable. This allows the agent a way to choo se among multiple possibilities, selecting the one which reaches a goal state.

### 4. Utility-based agents

Goal-based agents only distinguish between goal states and non -goal states. It is possible to define a measure of how desirable a particular state is. Th is measure can be obtained through the use of a *utility function* which maps a state to a measure of the utility of the state.

### 5. Learning agents

In some literature IAs are also referred to as *autonomous intelligent agents*, which mean they act independently, and will learn and adapt to changing circumstances. IA systems should exhibit the following characteristics:

- learn and improve through interaction with the environment (embodiment)
- adapt online and in real time
- learn quickly from large amounts of data
- accommodate new problem solving rules incrementally
- have memory based exemplar storage and retrieval capacities
- have parameters to represent short and long term memory, age, forgetting, etc.
- be able to analyze itself in terms of behavior, error and success.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

To actively perform their functions, Intelligent Agents today are normally gathered in a hierarchical structure containing many “sub-agents”. Intelligent sub-agents process and perform lower level functions. Taken together, the intelligent agent and sub-agents create a compete system that can accomplish difficult tasks or goals with behaviors and responses that display a form of intelligence.

Some of the sub-agents (not already mentioned in this treatment) that may be a part of an Intelligent Agent or a complete Intelligent Agent in them are:

1. Temporal Agents (for time-based decisions);
2. Spatial Agents (that relate to the physical real-world);
3. Input Agents (that process and make sense of sensor inputs - example neural network based agents neural network);
4. Processing Agents (that solve a problem like speech recognition);
5. Decision Agents (that are geared to decision making);
6. Learning Agents (for building up the data structures and database of other Intelligent agents);
7. World Agents (that incorporate a combination of all the other classes of agents to allow autonomous behaviors).

A very limited set of agents, that might be classified as semi-intelligent due to their lack of complexity, decision making, extremely limited world view and learning capacity can be found in the reference: Third Canadian Edition of "Management Information Systems for the Information Age". This document suggests that there are only four essential types of Intelligent Agents:

1. Buyer agents or shopping bots
2. User or personal agents
3. Monitoring-and-surveillance agents
4. Data Mining agents

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## Buyer Agent

Buyer agents travel around a network (i.e. the internet) retrieving information about goods and services. These agents, also known as 'shopping bots', work very efficiently for commodity products such as CDs, books, electronic components, and other one -size-fits-all products. *Amazon.com* is a good example of a shopping bot. The website will offer you a list of books that you might like to buy on the basis of what you're buying now and what you have bought in the past.

## User or Personal Agents

User agents, or personal agents, are intelligent agents that take action on your behalf. In this category belong those intelligent agents that already perform, or will shortly perform, the following tasks:

- Check your e-mail, sort it according to priority (your priority), and alert you when good stuff comes through - like college acceptance letters
- Play computer games as your opponent or patrol game areas for you
- Assemble customized news reports for you. There are several versions of these, CNN being a prime example
- Find information for you on the subject of your choice
- Fill out forms on the Web automatically for you, storing your information for future reference
- Scan Web pages looking for and highlighting text that constitutes the "important" part of the information there
- "Discuss" topics with you ranging from your deepest fears to sports

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

## Monitoring and Surveillance Agents

These agents, also known as "predictive agents", are intelligent agents that observe and report on equipment. For example, NASA's Jet Propulsion Laboratory has an agent that monitors inventory, planning, and scheduling equipment ordering to keep costs down, as well as food storage facilities. These agents usually monitor complex computer networks that can keep track of the configuration of each computer connected to the network.

## Data Mining Agents

A data mining agent operates in a data warehouse discovering information. A 'data warehouse' brings together information from lots of different sources. 'Data mining' is the process of looking through the data warehouse to find information that you can use to take action, such as ways to increase sales or keep customers who are considering defecting. 'Classification' is one of the most common types of data mining, which finds patterns in information and categorizes them into different classes. Data mining agents can also detect major shifts in trends or a key indicator and can detect the presence of new information and alert you to it.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# CHAPTER 3

# METHODOLOGY

## 3.1 Introduction

Some organization especially in safety-critical area requires that certain methodologies be used as it increases the quality of their software and also offer a certain degree of legal protection to all parties involve. Methodology helps to manage with the complexity of the software development process. Methodology may provide a framework for application of technique and resources at appropriate times during the development process. According to Bernard a good software methodology is a methodology which:-

> *"can be describes qualitatively, as well as qualitatively , can be used repeatedly, each time achieving similar results, can be taught to others within a reasonable timeframe, can be applied by others with a reasonable level of success"*

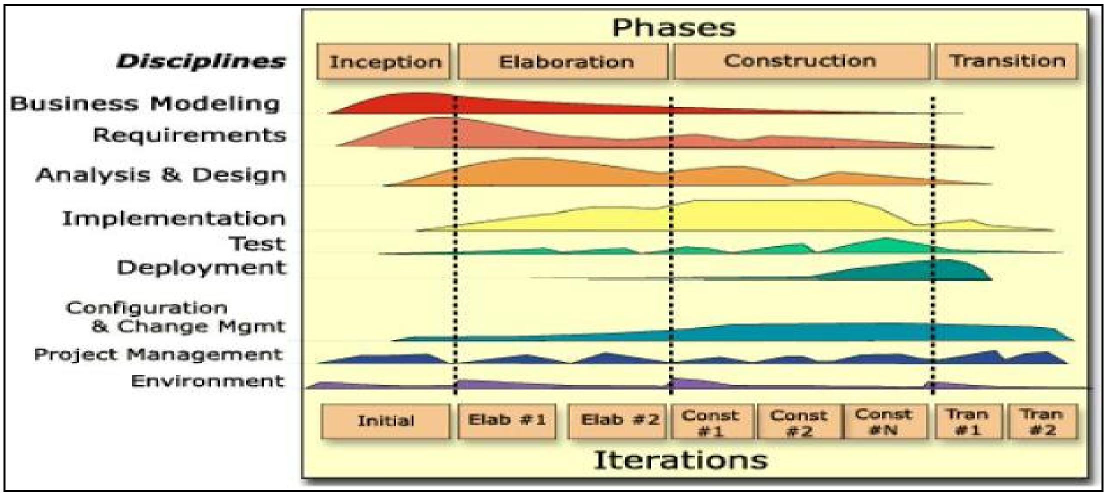
This chapter emphasizes on system development process which focuses on project methodology, a brief explanation on Unified Modeling Language as a technique for design the functionality of system and the end of this section this chapter will be concluded by the summary. In this area the RUP methodology has been used as a technique and guidelines for software development process. There are including the workflow on RUP and also elaborate in detail on their phase such as inception, elaboration, construction and testing phase .

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 3.2 Software Development Process

A software development process is describing software life cycle and software process. It begins with architecture, moves through analysis, design, and implementation, undergoes testing and finally is delivered to user. It also covers how to develop the software in the best way, a guideline and manner to develop the best quality of product by variety of tasks or activities for each phase in project methodology.

There are various difference software development methodologies available for modeling software system. The Rational Unified Process (RUP) has been selected in this project methodology. The choice of RUP methodology for a software development process because this technique is the best practice that can be tailored with large and suitable for all range of project and it aimed at meeting the needs of end users. RUP is an Object Oriented framework which provides guidelines and templates and tools to support software d evelopment process.



**Figures 3.1:** RUP methodology

Based on the figure above, that is show the RUP software development process which consists four chronological phases. There are Inception, Elaboration, Construction and Transition. The vertical line represents the workflow in RUP and the detail description is shown as below:

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

**1. Business modeling**

Understanding the structure and operation of the business

How the product fits into their organization

**2. Requirements**

Understand what user need from a product and to define these needs is a way that can be the basis for designing and planning product's content.

**3. Analysis and design**

Translate requirement into a design for the system

**4. Implementation**

Construct a working version of the system.

**5. Test**

Verify that the system meets requirement and minimize errors

**6. Deployment**

Distribute the product to users.

## ***The Benefits of RUP***

RUP provides the system development team with the advantages of RUP best practice s while providing a framework for addressing overall system issues. Some of the benefits of RUP include:

- Provides for ongoing collaboration among business analysts, architects, system engineers, software developers, hardware developers, and testers.
- Provides views that enable teams to address system quality issues in an architecture - driven process.
- Provides UML support for systems architecture.
- Scales upward to very large systems.
- Provides workflows for determining hardware and software components.
- Supports concurrent design, and iterative development of hardware and software components.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

- Easy to understand and facilitate resource planning and work structuring
- RUP delivers a well-structured framework, divided into phase and workflow, allowing easy navigation within the framework

## ***Phases in RUP Methodology***

According to RUP methodology, the phase in RUP had been applied as a guideline to accomplish the development of new reservation scheduling system. Following of that, the detail of description about each phase in RUP methodology and some related works that have been done show as below:-

### *Inception Phase*

Inception phase aimed to establish a business case. In general, the problem is first defined and analysis of the requirement of the system will be identified by discussion with the user. Included in this phase, the title was proposed, the background of problem in case study has been studied. The scope and objectives for development of this Appointment Scheduling for Reservation System (ASRS) system had been determined and clearly stated. This phase is very important to identify the business needs from the system and also to get the business value from the proposed system.

Following this, the real problem and process had been conducted by observation and interview session. The real information of their operations has been collected by visited at some of vehicle inspection center. From this study some of the problem was define and the requirement from the client was generated. In order to have better performance on inspection appointment, they required for modeling and designing new model system to solve all problems occurred in the manual system especially to reduce the amount of time processing. The literature review was also conducted in this phase. It is covers on technique and a new technology that would be applied in this new system such as Agent Technology, Constraint Logic Programming and others.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

### *Elaboration Phase*

Elaboration is the initial series of iteration which the team does serious investigation, implements the program, build the system architectures, analysis the user requirement specification and find out some high -risk issues.

During elaboration phase, the architecture of new system will be design. In this phase the real data for processing time in each process of vehicle inspection was collected. Data collection had been carried in vehicle inspection center. The processing time was recorded in seconds due to precision. From data collection, the data analysis is initiated by extracting data into the algorithm. Data analyses is some of steps for transforming data with the aim to generate some algorithm that will serve as an engine to this new system to solve time constraint problem.

Beside of that, the functional requirement was defined requested by user for new reservation scheduling system. The user requirements have been described in Software Requirement Specification (SRS) document. In this document, that is not only cover on user requirement but also the explanation about the performance, reliability, security, maintainability and other critical issues. Finally, the initial design and architecture modeling will be generated using UML design for the proposed system development. The preliminary designs have been developed including use case diagram, class diagram, sequence diagram and activity diagram.

### *Construction Phase*

A coding design may be the most obvious and difficult part on development of system. The system will de coded by programming language. In this process, Java programming language has been selected as a programming language to build ASRS system. During this phase, the unit testing must be conducted together to make sure the functionality of module process was successfully worked and meets the user's requirement.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

***Transition Phase***

After the code is appropriately tested and approved, it is moved into transition phase to ensure the software is available to end user. This phase includes final testing of product and making some minor adjustment based on error feedback. As a part o f this phase, the application will be tested to end user and they will evaluate the quality of system. It is very important to have training classes for most users to build excitement into adopting the new system. At the end of this phase is documenting th e internal design of software for the purpose of future maintenance and enhancement. Documentation is important as a user guideline to use the new system. The several project documents were produced such as user manual, a plan to support the user and a plan for upgrading the information system in the future. Then all requirement of the system will be installed and the related document will be delivered to end user.

## 3.3 Software Development Tools

The well-known Unified Modeling Language (UML) has been c hosen as tools for software development. The UML is a very important part of developing objects oriented software and the software development process. The UML represent a collection of best engineering practices that have proven successful in the modeling of large and complex system. The UML can help even non-programmer to understanding the overall functionality of the system in a way that meets the entire requirement in the system. The modeling design patterns with UML have several advantages include:-

- It quite naturally to object oriented modeling
- Process models can be communicated more easily in a large number of people
- UML provide a large set of diagram which can be used to define both structured and behavior of dynamic software process
- Object oriented modeling support the earlier phase of process model development
- By UML process model it easy to map them to programmed systems.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

| **Phase** | **Aim** | **Activity** | **Previous Works** |
|---|---|---|---|
| **Inception** | To establish a business case | -define a case study<br>-investigate the real problem<br>-Make some literature review | -analyze the real specification in case study<br>-doing some literature review in DisCSP, Agent Technolology and CLP that would be applied in this system |
| **Elaboration** | Analysis System design and system architecture | -data collection<br>-data analysis<br>-Initial design | -Collect real data in vehicle inspection center<br>-Analyze the data<br>-Doing some data analysis and extract the data to mathematical model<br>-Designing the mathematical model<br>-Designing system architecture and initial design for new system |
| **Construction** | Code writing and unit testing | -Full design<br>-code writing<br>-unit testing | -Designing full design using UML diagram for new system<br>-Full coding for ASRS system using Java programming using Eclipse<br>-Unit testing has been conducted in each module. |
| **Transition** | Documentation report and deliver to end user | -final testing<br>-make some minor adjustment<br>-user manual<br>-documentation for user<br>-installation new system | -some adjustment has been conducted<br>The full documentation has been done<br>User manual has been done |

**Table 3.1:** Summary of RUP phases

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

### 3.3.1 UML diagrams

UML defines various types of diagrams: use case diagram, class diagram, sequence diagram, activity diagram, deployment diagram, component diagram, collaboration diagram and object diagra m. The UML diagrams were recommended used as input to test cases. Use case were input functional test specification, sequence diagram were input to integration testing and class diagram were input to unit testing [4]. For further detail, here some description about UML diagram and their function.

***Use case Diagram***

Use case describes how external actors interact with the software system. Use case diagram consist of name pieces of functionality, the person invoking the functionality which called as actor and possibly the element responsible for implementing the use case. During this interaction an actor generates events to a system. That is a way to capture the functionality and requirement of system. Generally, use cases express the behavior of a system, model the functionality requirement of system using actors and use cases without the excessive detail which often confuses people with a less technical background.

***Class Diagram***

This diagram shows the structure and behavior of program system. In class diagr am contents using design elements such as classes, packages and objects, name, signature, and properties. There are several types of relationship in class diagrams such as association, dependency, generalization, aggregation, composition and realization. E ach relationship is represented in the diagram by a difference type of arrow.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## ***Activity Diagram***

That is used to model workflow or business processes and internal operation. It illustrates the dynamic nature of a system by modeling the flow of control f rom activity to activity. An activity represents an operation on some class in the system that results in a change in the state of the system.

## ***Sequence diagram***

The sequence diagram will be used to illustrate the design by represent how object interact with each other via message in the operation to fulfill work. This diagram used in elaborating and detailing the behavior of a system. This diagram show the interaction between the object involved in the use case, the actors and flows of events. On horizont al axis it shows of the object that it represented, while on vertical axis it shows the flow or sequence of the event.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# CHAPTER 4

# EXPERIMENTAL DATA AND ANALYSIS

## 4.1 Introduction

Data Analysis is some sort of steps of transforming data with th e aim of extracting the data that we get through particular data collection. In our study, we use both quantitative and qualitative data as our input to be used in the next steps of analysis. Based on analysis of the history of science, Kuhn concludes that :

> *“Large amounts of qualitative work have usually been prerequisite to fruitful quantification in the physical sciences.”*

The result of these steps is a number, or a series of numbers that would be used to the next steps of data analysis, or it can be concluded as a result of the study. In our case, we got the results in a form of table, and then we use the figure from the table to be analyzed into the next experimental steps. Generally, results of quantitative methods often presented in tables, charts, graphs or other forms of statistics. Statistics is a mathematical science relating to the collection of data, analysis of the data, and also interpretation or explanation of the data, as well as presentation of data as a final stage.

Marshall et. al in his study outlined the four broad questions that should be considered in the data collection.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

i. In what ways does it help our study?

ii. What are the advantages or disadvantages of the data?

iii. What are the contributions will we get from the data?

iv. How it could be changed or improved?

In general, there are two statistical methods those can be used to collect and analyze, interpret, explain, and present the data. The one which can be used to summarize or describe a collection of data is known as descriptive statisti cs, while inferential statistics is a method to model the data in a way that accounts for randomness and uncertainty in the observation, and then used to draw inferences about the process or population being studied. Statistical inference or also called as statistical induction comprises the use of statistics to make inferences concerning some unknown aspect of a population. It is distinguished from descriptive statistics.

We use Ms Excel 2003 as an analytical tool to calculate the mean and average values of the data. Then, we use this software to analyze the rest of steps involve in this analyses.

## 4.2 Data Collection

Before collecting real data from the case study, we first planed and defined the field and information which relevant for this study. Fro m the observation, we made our strategy to collect only the relevant data which is includ e the processing time for every point of the vehicle inspection. There are 10 points of vehicle inspection. Every point of inspection is defined by a prefix P1, P2, .., P10. The details for point of inspection described as follow:

1) P1 : Identification check – verifies the engine and chassis number of the vehicle to be tested.
2) P2 : Above carriage check – the vehicle is then scrutinized for surface defects and non-compliance with road transport act 1987.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

3) P3 : Emission test – density of particles in diesel engines and volume of gases e.g. HC, CO etc in petrol engines are gauged through free acceleration tests.
4) P4 : Side-slip test – lateral movements of the vehicle are tested for optimum road-handling criteria.
5) P5 : Suspension test – checks the effectiveness of the suspension system (springs, shock absorber, and joints) on each axle.
6) P6 : Brake test – the various brake efficiencies, such as service brake efficiency, dynamic unbalance, run-out, residual force and parking brake are evaluated for maximum performance.
7) P7 : Speedometer – checks the road speed at the wheels against the value indicated on the vehicle's speedometer.
8) P8 : Headlight test – measures the intensity and projection of the vehicle's headlights
9) P9 : Undercarriage check – a vehicle's undercarriage is thoroughly scrutinized for defects while it is tested on an axle play detector.
10) P10 : Computerized analysis – a state-of-the-art central computer controls and monitors each test phase. All relevant test data is then analyzed and the final result printed out.

Data are collected from various sources as follows:

i. Observation on environment, management and services.
ii. Observation on the process of inspections.
iii. Recording time taken on every point of inspections' process.
iv. Recording any significant or weird action or event happened during observation.
v. Request certain appropriate form and information from the organization.

Here are the descriptions of data that we have collected for this study.

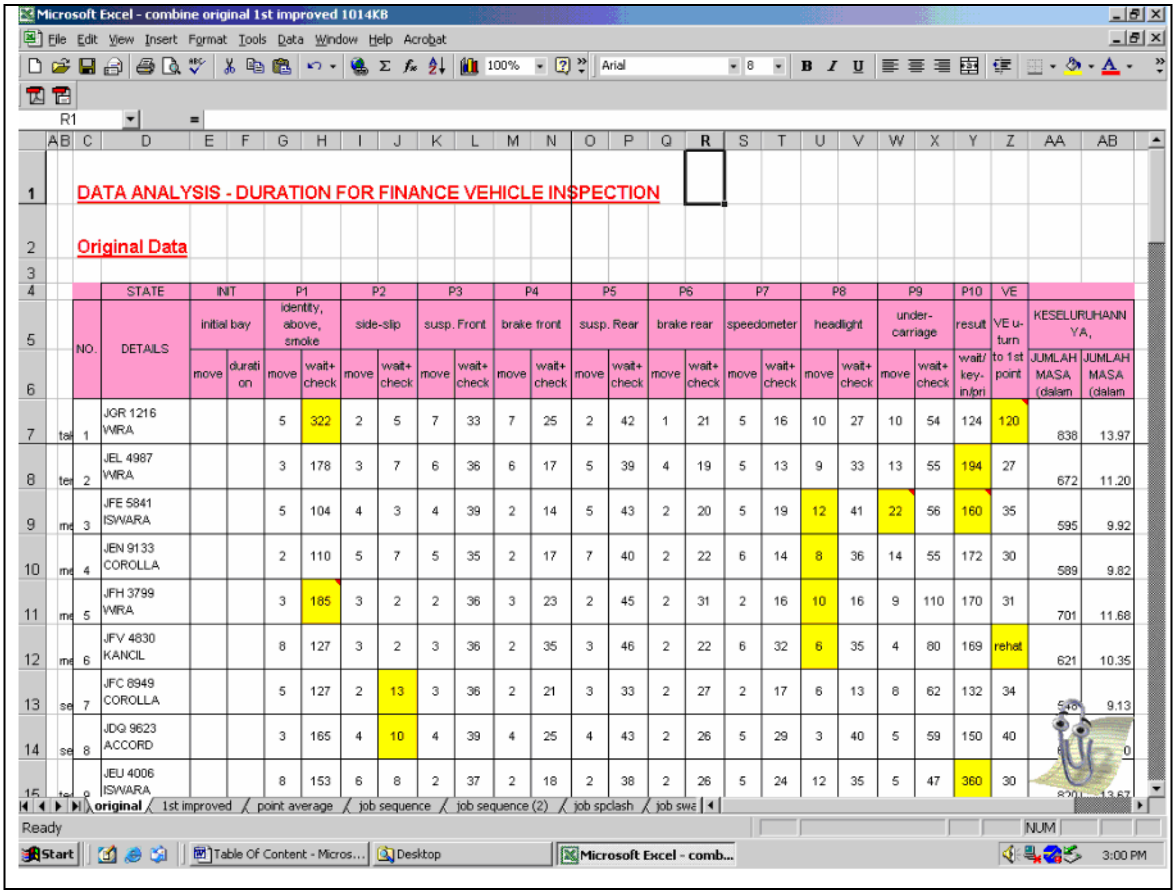
- Plat number of the vehicle.
- Brand and model of the vehicle.
- Traveling time to the initial bay.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

- Length of time in the initial bay.
- Traveling time to the first point, P1.
- Length of time in the first point, P1.
- Traveling time to the second point, P2.
- Length of time in the second point, P2.
- Traveling time to the third point, P3.
- Length of time in the third point, P3.
- Traveling time to the forth point, P4.
- Length of time in the forth point, P4.
- Traveling time to the fifth point, P5.
- Length of time in the fifth point, P5.
- Traveling time to the sixth point, P6.
- Length of time in the sixth point, P6.
- Traveling time to the seventh point, P7.
- Length of time in the seventh point, P7.
- Traveling time to the eighth point, P8.
- Length of time in the eighth point, P8.
- Traveling time to the ninth point, P9.
- Length of time in the ninth point, P9.
- Traveling time to the tenth point, P10.
- Length of time in the tenth point, P10.
- Traveling time for the Vehicle Examiner (VE) to make u -turn; which is VE has to turn back from the P10 of the completed inspection to the P1 of the new inspection.

Based from the meeting with Quality Assurance Manager, the average processing time for every vehicle being inspected is around five minutes. This figure is based on the production volume of the day which recording the start and end time for every vehicle inspection. **Figure 4.1** below shows the sample of the data that are collected at the vehicle inspection centre.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com


Microsoft Excel - combine original 1st improved 1014KB
File Edit View Insert Format Tools Data Window Help Acrobat
100%
Arial
8
R1
=
original
1st improved
point average
job sequence
job sequence (2)
job spclash
job swa
Ready
NUM
Start
Table Of Content - Micros...
Desktop
Microsoft Excel - comb...
3:00 PM

DATA ANALYSIS - DURATION FOR FINANCE VEHICLE INSPECTION

Original Data

| | | STATE | INIT | | P1 | | P2 | | P3 | | P4 | | P5 | | P6 | | P7 | | P8 | | P9 | | P10 | VE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | NO. | DETAILS | initial bay | | identity, above, smoke | | side-slip | | susp. Front | | brake front | | susp. Rear | | brake rear | | speedometer | | headlight | | under-carriage | | result | VE u-turn to 1st point | KESELURUHANN YA, | |
| | | | move | duration | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | wait/ key-in/pri | | JUMLAH MASA (dalam | JUMLAH MASA (dalam |
| tal | 1 | JGR 1216 WIRA | | | 5 | 322 | 2 | 5 | 7 | 33 | 7 | 25 | 2 | 42 | 1 | 21 | 5 | 16 | 10 | 27 | 10 | 54 | 124 | 120 | 838 | 13.97 |
| ter | 2 | JEL 4987 WIRA | | | 3 | 178 | 3 | 7 | 6 | 36 | 6 | 17 | 5 | 39 | 4 | 19 | 5 | 13 | 9 | 33 | 13 | 55 | 194 | 27 | 672 | 11.20 |
| me | 3 | JFE 5841 ISWARA | | | 5 | 104 | 4 | 3 | 4 | 39 | 2 | 14 | 5 | 43 | 2 | 20 | 5 | 19 | 12 | 41 | 22 | 56 | 160 | 35 | 595 | 9.92 |
| me | 4 | JEN 9133 COROLLA | | | 2 | 110 | 5 | 7 | 5 | 35 | 2 | 17 | 7 | 40 | 2 | 22 | 6 | 14 | 8 | 36 | 14 | 55 | 172 | 30 | 589 | 9.82 |
| me | 5 | JFH 3799 WIRA | | | 3 | 185 | 3 | 2 | 2 | 36 | 3 | 23 | 2 | 45 | 2 | 31 | 2 | 16 | 10 | 16 | 9 | 110 | 170 | 31 | 701 | 11.68 |
| me | 6 | JFV 4830 KANCIL | | | 8 | 127 | 3 | 2 | 3 | 36 | 2 | 35 | 3 | 46 | 2 | 22 | 6 | 32 | 6 | 35 | 4 | 80 | 169 | rehat | 621 | 10.35 |
| se | 7 | JFC 8949 COROLLA | | | 5 | 127 | 2 | 13 | 3 | 36 | 2 | 21 | 3 | 33 | 2 | 27 | 2 | 17 | 6 | 13 | 8 | 62 | 132 | 34 | 548 | 9.13 |
| se | 8 | JDQ 9623 ACCORD | | | 3 | 165 | 4 | 10 | 4 | 39 | 4 | 25 | 4 | 43 | 2 | 26 | 5 | 29 | 3 | 40 | 5 | 59 | 150 | 40 | [illegible] | [illegible] |
| tal | 9 | JEU 4006 ISWARA | | | 8 | 153 | 6 | 8 | 2 | 37 | 2 | 18 | 2 | 38 | 2 | 26 | 5 | 24 | 12 | 35 | 5 | 47 | 360 | 30 | 820 | 13.67 |

**Figure 4.1**: Sample of Data Collection

## 4.3 Data Filtering

From the original raw data, data were improved by a little adjustment. Data with major difference will be deleted from the record and the rest of the data will be use for the next step of analysis. There are several ways the data are filtered. In this study the data filtering is done as follows:

i. Sorting the value for each column of point of inspection, one by one, to be in ascending order.

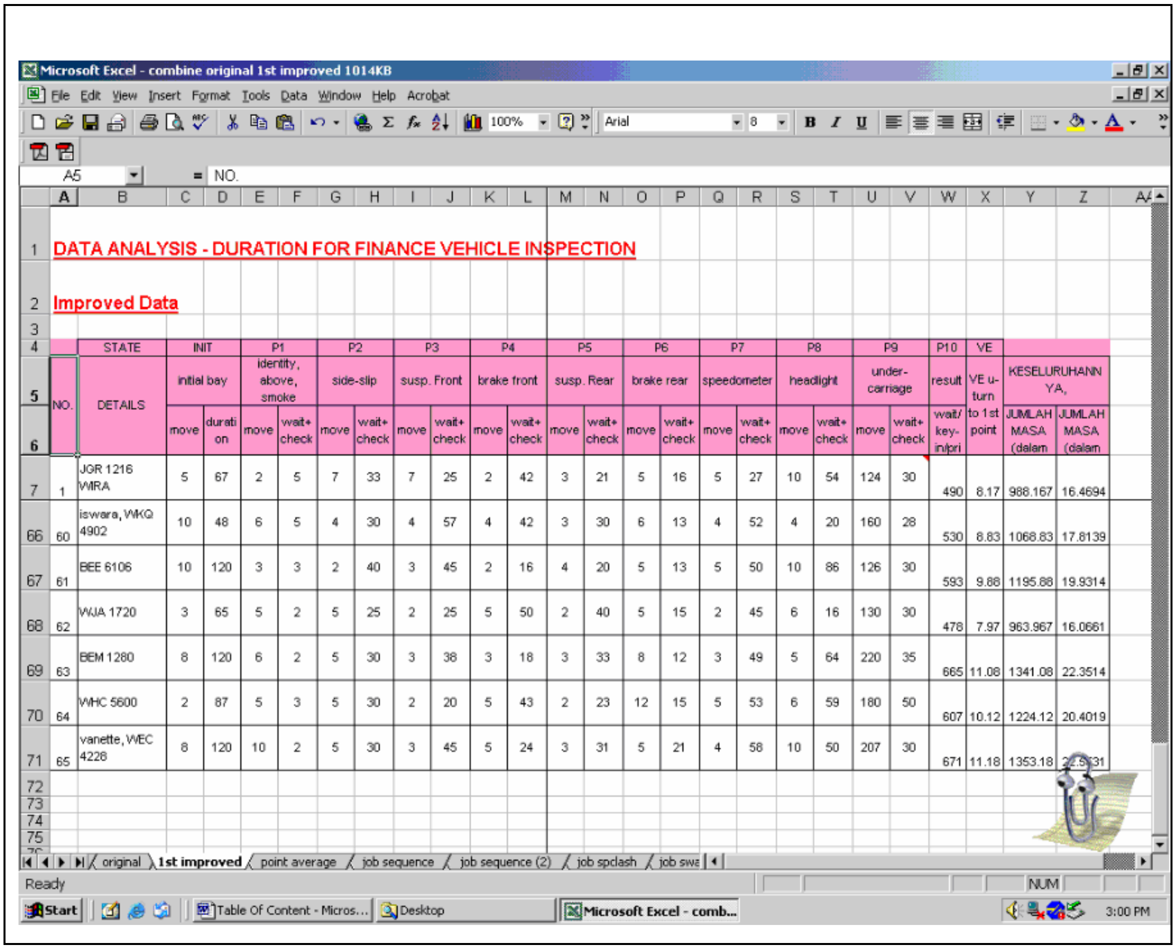
This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

ii. Then, recognizing if there is any weird value from the first and last part of the overall value in each column. If the value is out of normal range, then the value is marked as a weird value.

iii. If there is a weird value, it is tried to be rectified. If cannot, deleting a record at all.

iv. Repeat the steps above to each column in database.

**Figure 4.2** below shows the result after the filtering technique is done.


Microsoft Excel - combine original 1st improved 1014KB
File Edit View Insert Format Tools Data Window Help Acrobat
100%
Arial
8
A5
= NO.
original
1st improved
point average
job sequence
job sequence (2)
job spclash
job swa
Ready
NUM
Start
Table Of Content - Micros...
Desktop
Microsoft Excel - comb...
3:00 PM

DATA ANALYSIS - DURATION FOR FINANCE VEHICLE INSPECTION

Improved Data

| | | STATE | INIT | | P1 | | P2 | | P3 | | P4 | | P5 | | P6 | | P7 | | P8 | | P9 | | P10 | VE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 | NO. | DETAILS | initial bay | | identity, above, smoke | | side-slip | | susp. Front | | brake front | | susp. Rear | | brake rear | | speedometer | | headlight | | under-carriage | | result | VE u-turn to 1st point | KESELURUHANNYA, | |
| 6 | | | move | duration | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | wait/key-in/pri | | JUMLAH MASA (dalam | JUMLAH MASA (dalam |
| 7 | 1 | JGR 1216 WIRA | 5 | 67 | 2 | 5 | 7 | 33 | 7 | 25 | 2 | 42 | 3 | 21 | 5 | 16 | 5 | 27 | 10 | 54 | 124 | 30 | 490 | 8.17 | 988.167 | 16.4694 |
| 66 | 60 | iswara, WKQ 4902 | 10 | 48 | 6 | 5 | 4 | 30 | 4 | 57 | 4 | 42 | 3 | 30 | 6 | 13 | 4 | 52 | 4 | 20 | 160 | 28 | 530 | 8.83 | 1068.83 | 17.8139 |
| 67 | 61 | BEE 6106 | 10 | 120 | 3 | 3 | 2 | 40 | 3 | 45 | 2 | 16 | 4 | 20 | 5 | 13 | 5 | 50 | 10 | 86 | 126 | 30 | 593 | 9.88 | 1195.88 | 19.9314 |
| 68 | 62 | WJA 1720 | 3 | 65 | 5 | 2 | 5 | 25 | 2 | 25 | 5 | 50 | 2 | 40 | 5 | 15 | 2 | 45 | 6 | 16 | 130 | 30 | 478 | 7.97 | 963.967 | 16.0661 |
| 69 | 63 | BEM 1280 | 8 | 120 | 6 | 2 | 5 | 30 | 3 | 38 | 3 | 18 | 3 | 33 | 8 | 12 | 3 | 49 | 5 | 64 | 220 | 35 | 665 | 11.08 | 1341.08 | 22.3514 |
| 70 | 64 | WHC 5600 | 2 | 87 | 5 | 3 | 5 | 30 | 2 | 20 | 5 | 43 | 2 | 23 | 12 | 15 | 5 | 53 | 6 | 59 | 180 | 50 | 607 | 10.12 | 1224.12 | 20.4019 |
| 71 | 65 | vanette, WEC 4228 | 8 | 120 | 10 | 2 | 5 | 30 | 3 | 45 | 5 | 24 | 3 | 31 | 5 | 21 | 4 | 58 | 10 | 50 | 207 | 30 | 671 | 11.18 | 1353.18 | 22.5531 |

**Figure 4.2** : Sample of datasets after filtering

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 4.4 Data Experimental and Analysis

After data filtering, we then compute some descriptive value to get the pattern of the data. The main goal in this step is to obtain the average value from every point of inspection. The values computed are listed below, and shown in **Figure 4.3** and **Figure 4.4.**

i. Minimum value

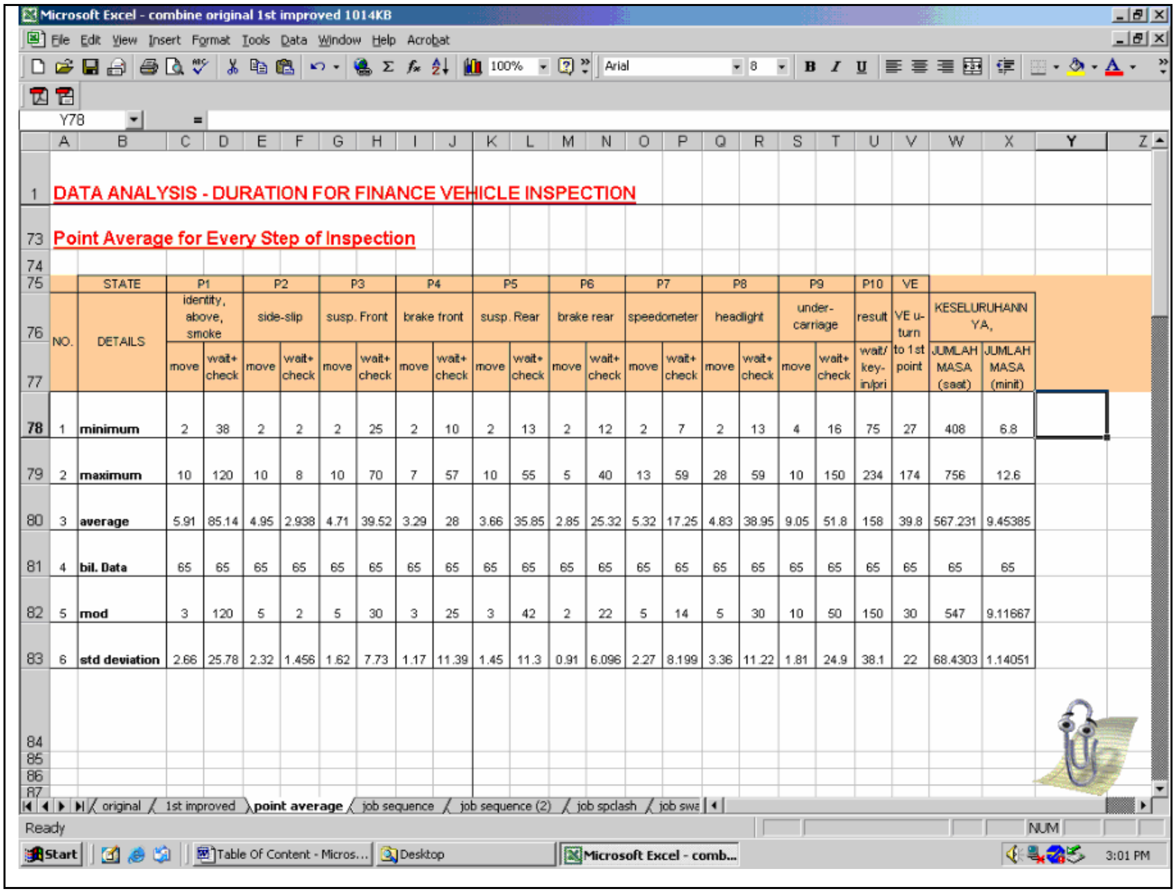
ii. Maximum value

iii. Number of data

iv. Average value

v. Mod value

vi. Standard deviation


Microsoft Excel - combine original 1st improved 1014KB
File Edit View Insert Format Tools Data Window Help Acrobat
100%
Arial
8
Y78
original
1st improved
point average
job sequence
job sequence (2)
job spclash
job swa
Ready
NUM
Start
Table Of Content - Micros...
Desktop
Microsoft Excel - comb...
3:01 PM

DATA ANALYSIS - DURATION FOR FINANCE VEHICLE INSPECTION

Point Average for Every Step of Inspection

| | STATE | P1 | | P2 | | P3 | | P4 | | P5 | | P6 | | P7 | | P8 | | P9 | | P10 | VE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NO. | DETAILS | identity, above, smoke | | side-slip | | susp. Front | | brake front | | susp. Rear | | brake rear | | speedometer | | headlight | | under-carriage | | result | VE u-turn to 1st point | KESELURUHANN YA, | |
| | | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | wait/ key-in/pri | | JUMLAH MASA (saat) | JUMLAH MASA (minit) |
| 1 | **minimum** | 2 | 38 | 2 | 2 | 2 | 25 | 2 | 10 | 2 | 13 | 2 | 12 | 2 | 7 | 2 | 13 | 4 | 16 | 75 | 27 | 408 | 6.8 |
| 2 | **maximum** | 10 | 120 | 10 | 8 | 10 | 70 | 7 | 57 | 10 | 55 | 5 | 40 | 13 | 59 | 28 | 59 | 10 | 150 | 234 | 174 | 756 | 12.6 |
| 3 | **average** | 5.91 | 85.14 | 4.95 | 2.938 | 4.71 | 39.52 | 3.29 | 28 | 3.66 | 35.85 | 2.85 | 25.32 | 5.32 | 17.25 | 4.83 | 38.95 | 9.05 | 51.8 | 158 | 39.8 | 567.231 | 9.45385 |
| 4 | **bil. Data** | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 | 65 |
| 5 | **mod** | 3 | 120 | 5 | 2 | 5 | 30 | 3 | 25 | 3 | 42 | 2 | 22 | 5 | 14 | 5 | 30 | 10 | 50 | 150 | 30 | 547 | 9.11667 |
| 6 | **std deviation** | 2.66 | 25.78 | 2.32 | 1.456 | 1.62 | 7.73 | 1.17 | 11.39 | 1.45 | 11.3 | 0.91 | 6.096 | 2.27 | 8.199 | 3.36 | 11.22 | 1.81 | 24.9 | 38.1 | 22 | 68.4303 | 1.14051 |

**Figure 4.3**: Data analysis for each point of inspection.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

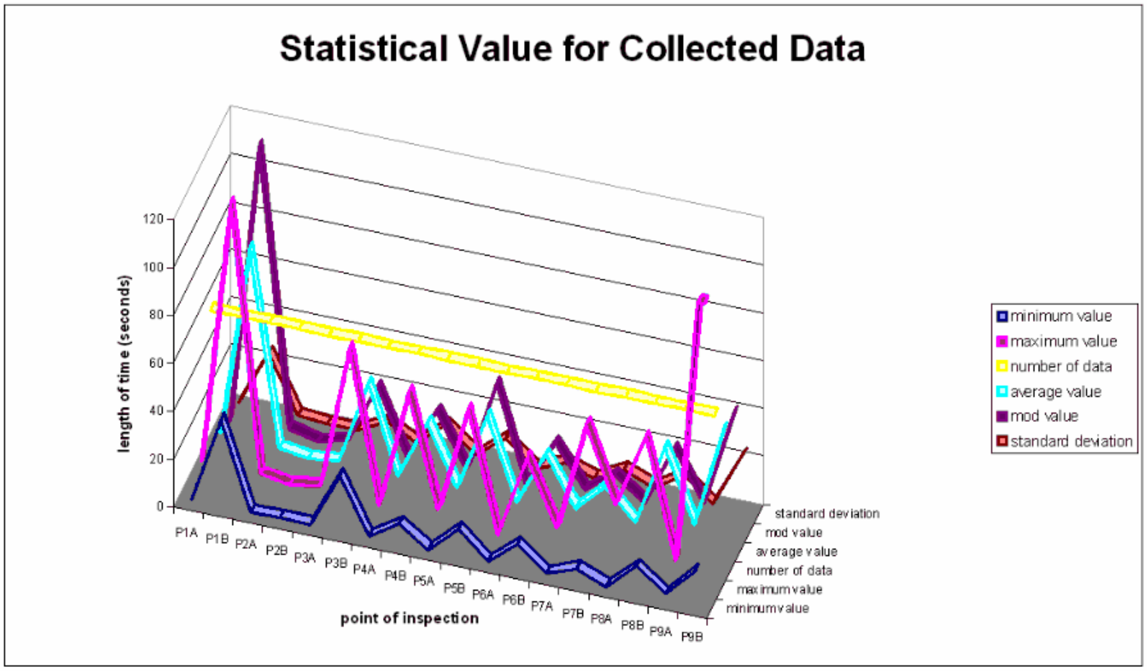


**Figure 4.4**: Data analysis for each point of inspection using 3-D Line Chart.

### ***P1 : Identification check***

The analysis for this point of inspection is described in **Table 4.1** below, while the illustration of the figures is shown in **Figure 4.5.**

| | **P1A (traveling time for P1)** | **P1B (service time for P1)** |
|---|---|---|
| **Minimum value** | 2 seconds | 38 seconds |
| **Maximum value** | 10 seconds | 120 seconds |
| **Number of data** | 65 vehicles | 65 vehicles |
| **Average value** | 5.91 seconds | 85.14 seconds |
| **Mod value** | 3 seconds | 120 seconds |
| **Standard deviation** | 2.66 seconds | 25.78 seconds |

**Table 4.1:** Analysis for point inspection PI – Identification Check

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

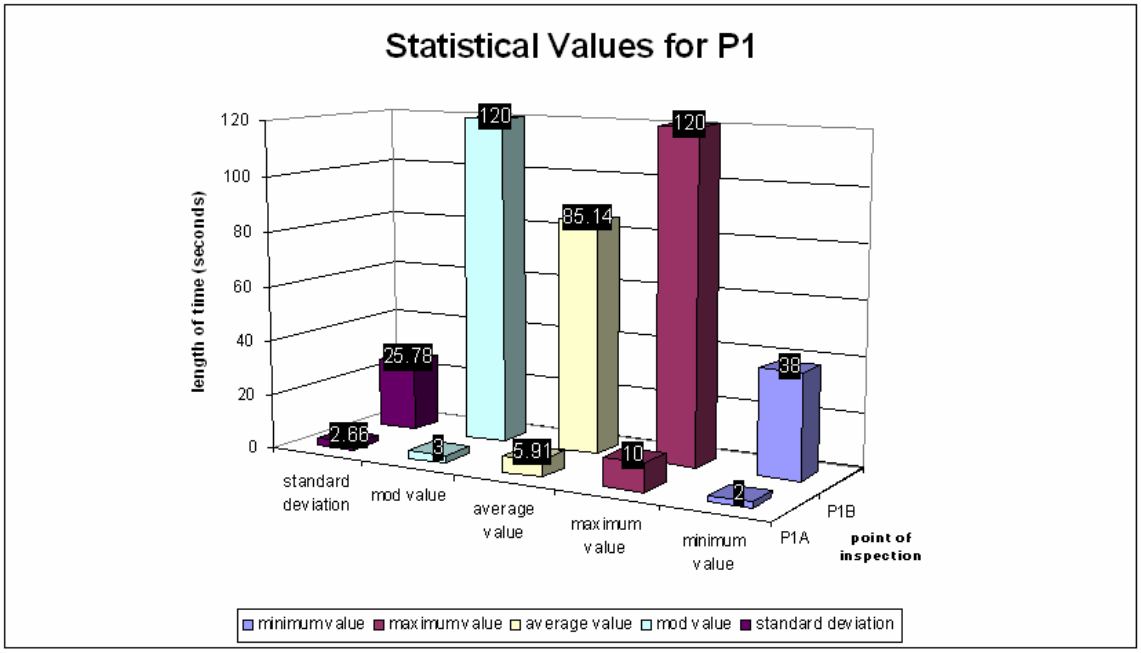


**Figure 4.5**: Illustration of analysis for point inspection PI – Identification Check using 3-D Column Chart.

## ***P2 : Above carriage check***

The analysis for this point of inspection is described in **Table 4.2** below, while the illustration of the figures is shown in **Figure 4.6.**

| | **P1A (traveling time for P1)** | **P1B (service time for P1)** |
|---|---|---|
| **Minimum value** | 2 seconds | 2 seconds |
| **Maximum value** | 10 seconds | 8 seconds |
| **Number of data** | 65 vehicles | 65 vehicles |
| **Average value** | 4.95 seconds | 2.938 seconds |
| **Mod value** | 5 seconds | 2 seconds |
| **Standard deviation** | 2.32 seconds | 1.456 seconds |

**Table 4.2:** Analysis for point inspection P2 – Above Carriage Check

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

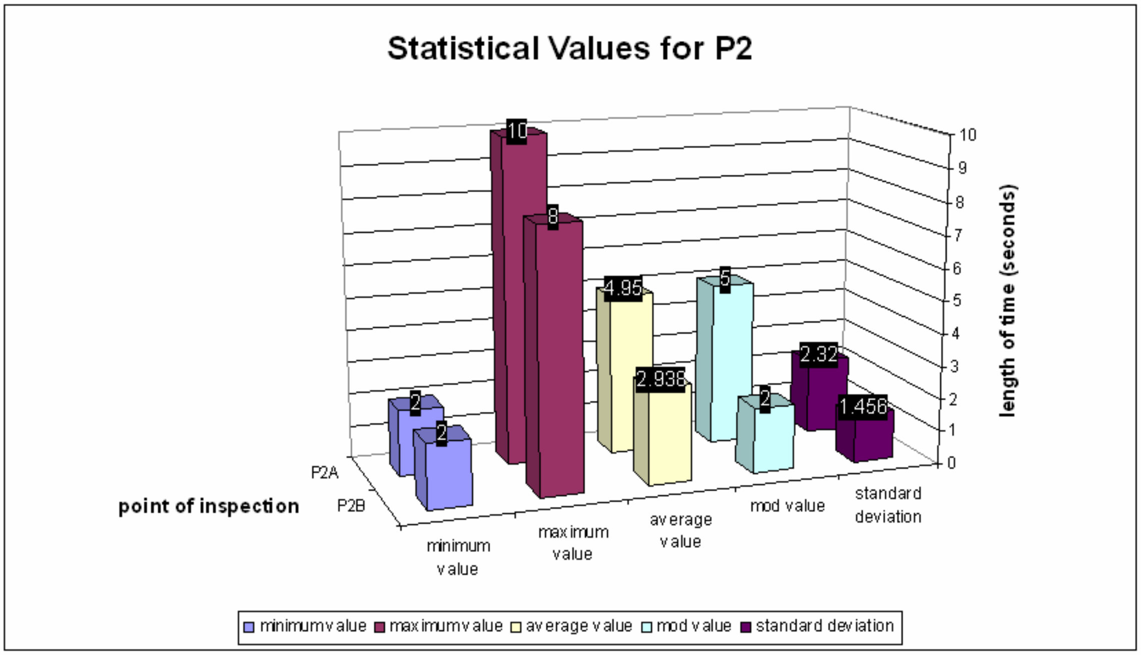


**Figure 4.6**: Illustration of analysis for point inspection P2 – Above Carriage Check using 3-D Column Chart.

***P3 : Emission Test***

The analysis for this point of inspection is described in **Table 4.3** below, while the illustration of the figures is shown in **Figure 4.7.**

| | **P1A (traveling time for P1)** | **P1B (service time for P1)** |
|---|---|---|
| **Minimum value** | 2 seconds | 25 seconds |
| **Maximum value** | 10 seconds | 70 seconds |
| **Number of data** | 65 vehicles | 65 vehicles |
| **Average value** | 4.71 seconds | 39.52 seconds |
| **Mod value** | 5 seconds | 30 seconds |
| **Standard deviation** | 1.62 seconds | 7.73 seconds |

**Table 4.3:** Analysis for point inspection P3 – Emission Test

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

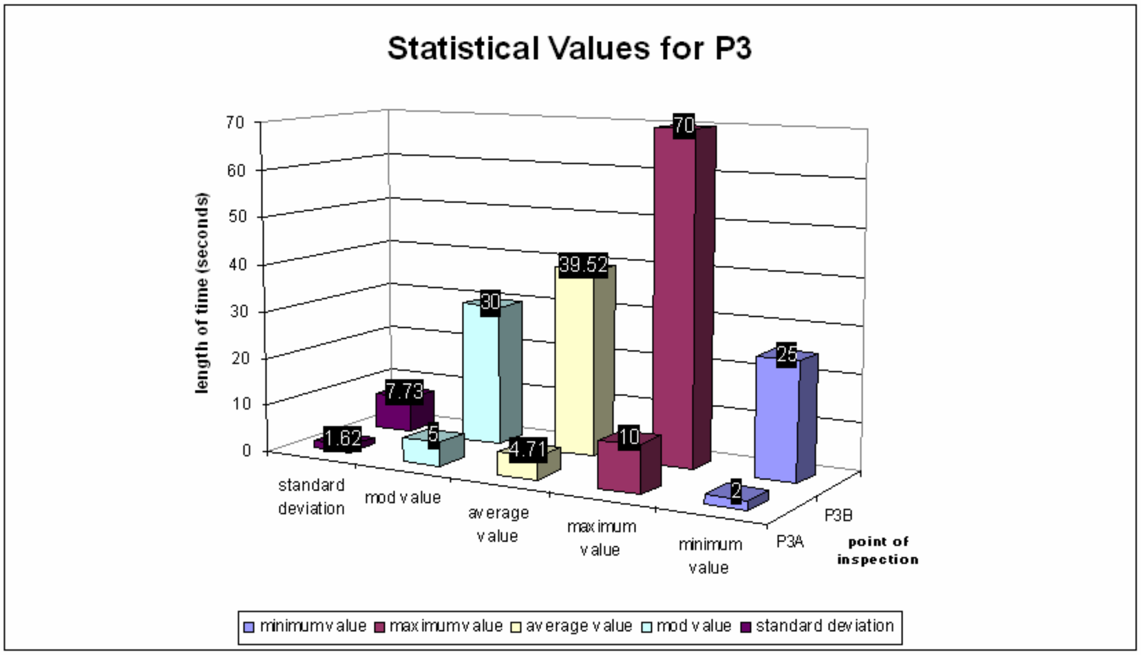


**Figure 4.7**: Illustration of analysis for point inspection P3 – Emission Test using 3-D Column Chart.

***P4 : Side-Slip Test***

The analysis for this point of inspection is described in **Table 4.4** below, while the illustration of the figures is shown in **Figure 4.8.**

| | **P1A (traveling time for P1)** | **P1B (service time for P1)** |
|---|---|---|
| **Minimum value** | 2 seconds | 10 seconds |
| **Maximum value** | 7 seconds | 57 seconds |
| **Number of data** | 65 vehicles | 65 vehicles |
| **Average value** | 3.29 seconds | 28 seconds |
| **Mod value** | 3 seconds | 25 seconds |
| **Standard deviation** | 1.17 seconds | 11.39 seconds |

**Table 4.4:** Analysis for point inspection P4 – Side-Slip Test

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

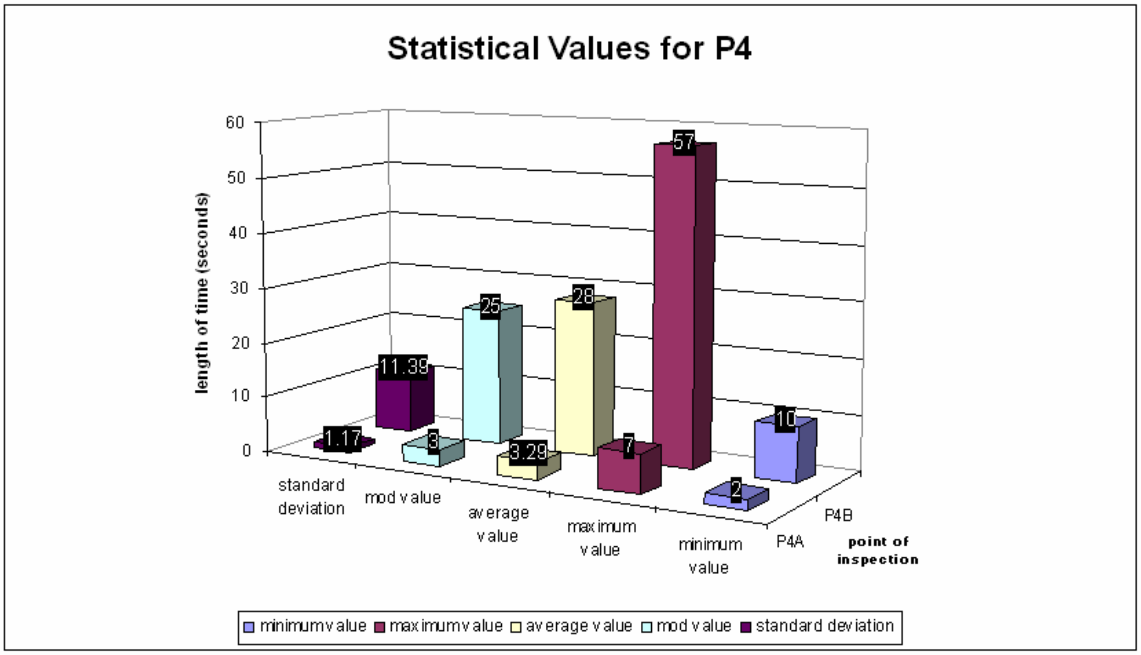


**Figure 4.8**: Illustration of analysis for point inspection P4 – Side-Slip Test using 3-D Column Chart.

### *P5 : Suspension Test*

The analysis for this point of inspection is described in **Table 4.5** below, while the illustration of the figures is shown in **Figure 4.9.**

| | **P1A (traveling time for P1)** | **P1B (service time for P1)** |
|---|---|---|
| **Minimum value** | 2 seconds | 13 seconds |
| **Maximum value** | 10 seconds | 55 seconds |
| **Number of data** | 65 vehicles | 65 vehicles |
| **Average value** | 3.66 seconds | 35.85 seconds |
| **Mod value** | 3 seconds | 42 seconds |
| **Standard deviation** | 1.45 seconds | 11.3 seconds |

**Table 4.5:** Analysis for point inspection P5 – Suspension Test

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

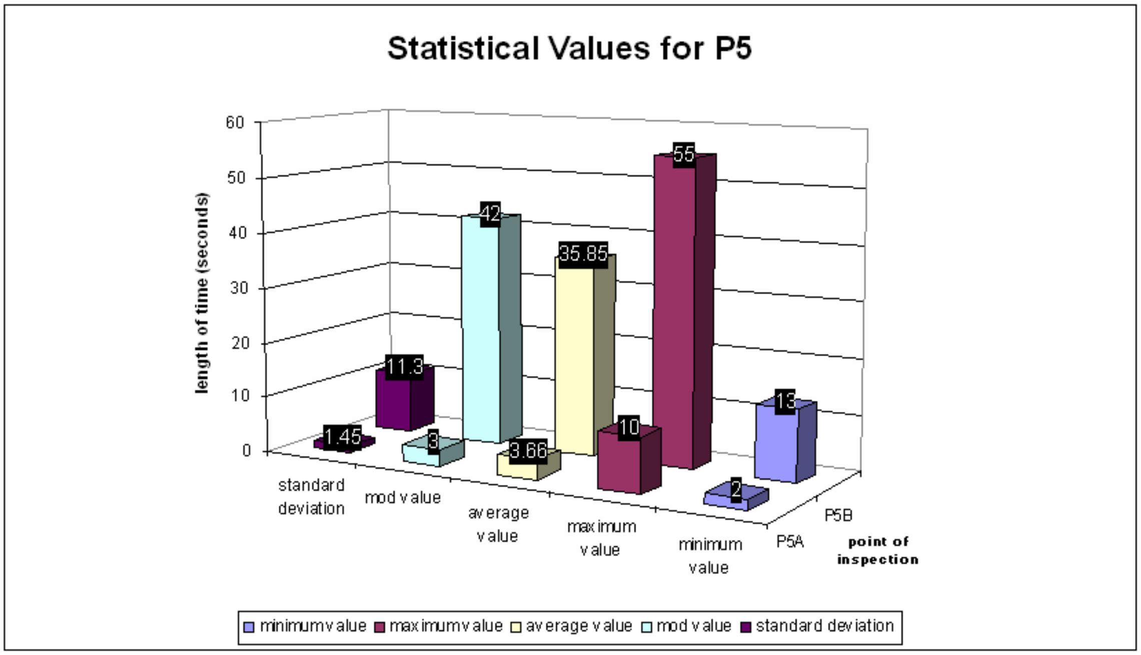


**Figure 4.9**: Illustration of analysis for point inspection P5 – Suspension Test using 3-D Column Chart.

***P6 : Brake Test***

The analysis for this point of inspection is described in **Table 4.6** below, while the illustration of the figures is shown in **Figure 4.10.**

| | **P1A (traveling time for P1)** | **P1B (service time for P1)** |
|---|---|---|
| **Minimum value** | 2 seconds | 12 seconds |
| **Maximum value** | 5 seconds | 40 seconds |
| **Number of data** | 65 vehicles | 65 vehicles |
| **Average value** | 2.85 seconds | 25.32 seconds |
| **Mod value** | 2 seconds | 22 seconds |
| **Standard deviation** | 0.91 seconds | 6.096 seconds |

**Table 4.6:** Analysis for point inspection P6 – Brake Test

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

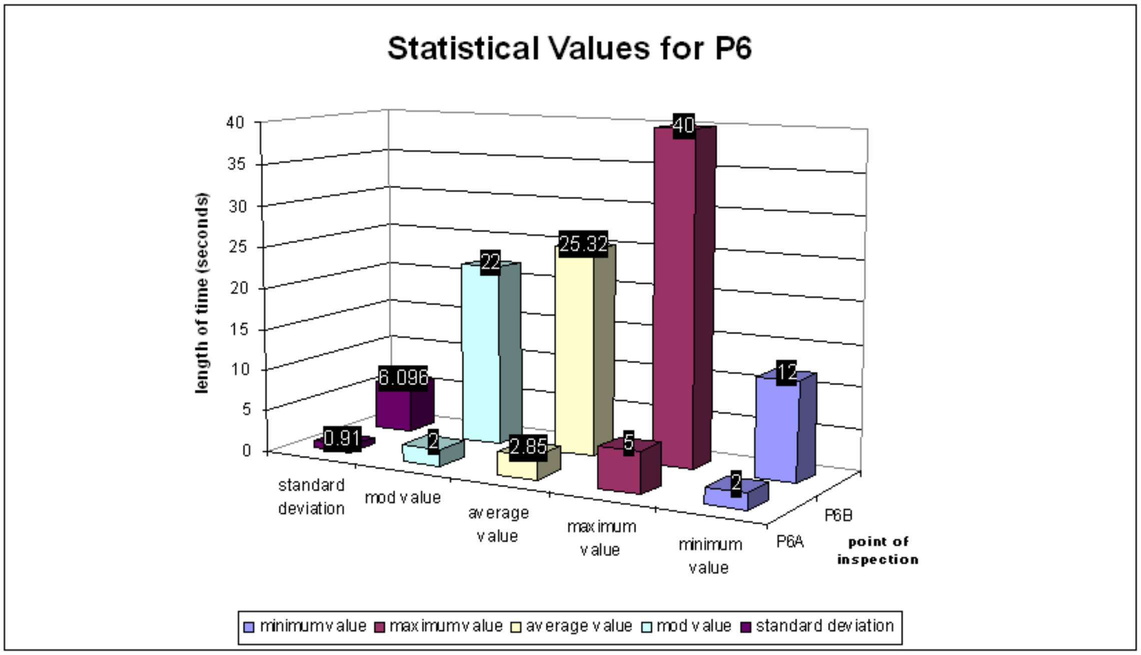


**Figure 4.10**: Illustration of analysis for point inspection P6 – Brake Test using 3-D Column Chart.

***P7 : Speedometer***

The analysis for this point of inspection is described in **Table 4.7** below, while the illustration of the figures is shown in **Figure 4.11.**

| | **P1A (traveling time for P1)** | **P1B (service time for P1)** |
|---|---|---|
| **Minimum value** | 2 seconds | 7 seconds |
| **Maximum value** | 13 seconds | 59 seconds |
| **Number of data** | 65 vehicles | 65 vehicles |
| **Average value** | 5.32 seconds | 17.25 seconds |
| **Mod value** | 5 seconds | 14 seconds |
| **Standard deviation** | 2.27 seconds | 8.199 seconds |

**Table 4.7:** Analysis for point inspection P7 – Speedometer

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

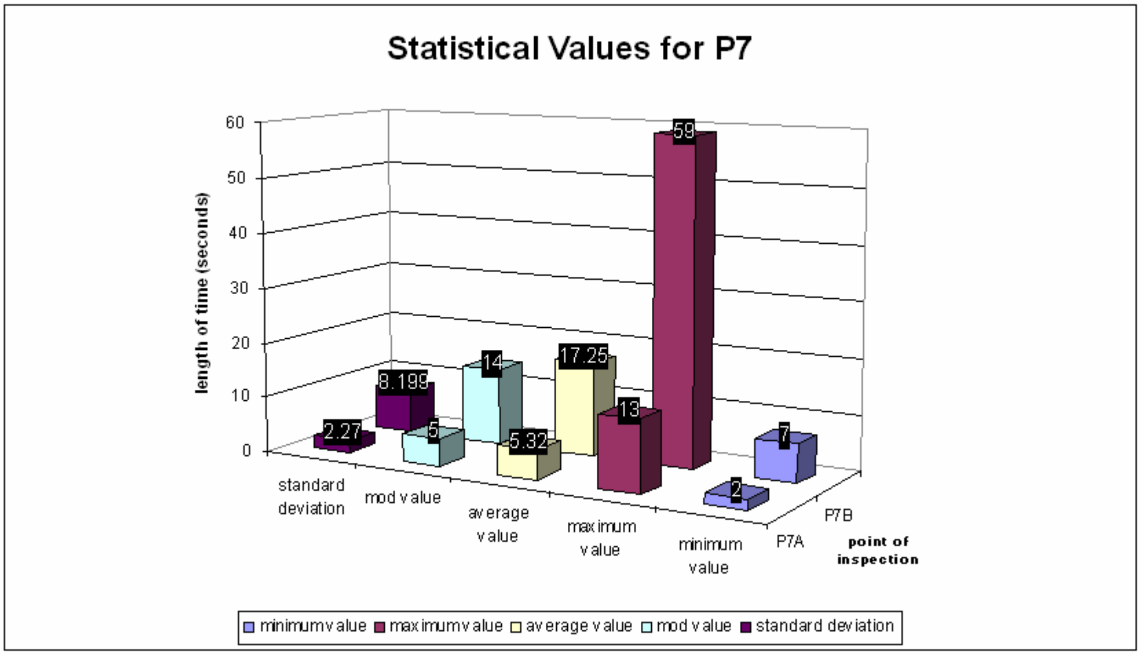


**Figure 4.11**: Illustration of analysis for point inspection P7 – Speedometer using 3-D Column Chart.

***P8 : Headlight Test***

The analysis for this point of inspection is described in **Table 4.8** below, while the illustration of the figures is shown in **Figure 4.12.**

| | **P1A (traveling time for P1)** | **P1B (service time for P1)** |
|---|---|---|
| **Minimum value** | 2 seconds | 13 seconds |
| **Maximum value** | 28 seconds | 59 seconds |
| **Number of data** | 65 vehicles | 65 vehicles |
| **Average value** | 4.83 seconds | 38.95 seconds |
| **Mod value** | 5 seconds | 30 seconds |
| **Standard deviation** | 3.36 seconds | 11.22 seconds |

**Table 4.8:** Analysis for point inspection P8 – Headlight Test

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

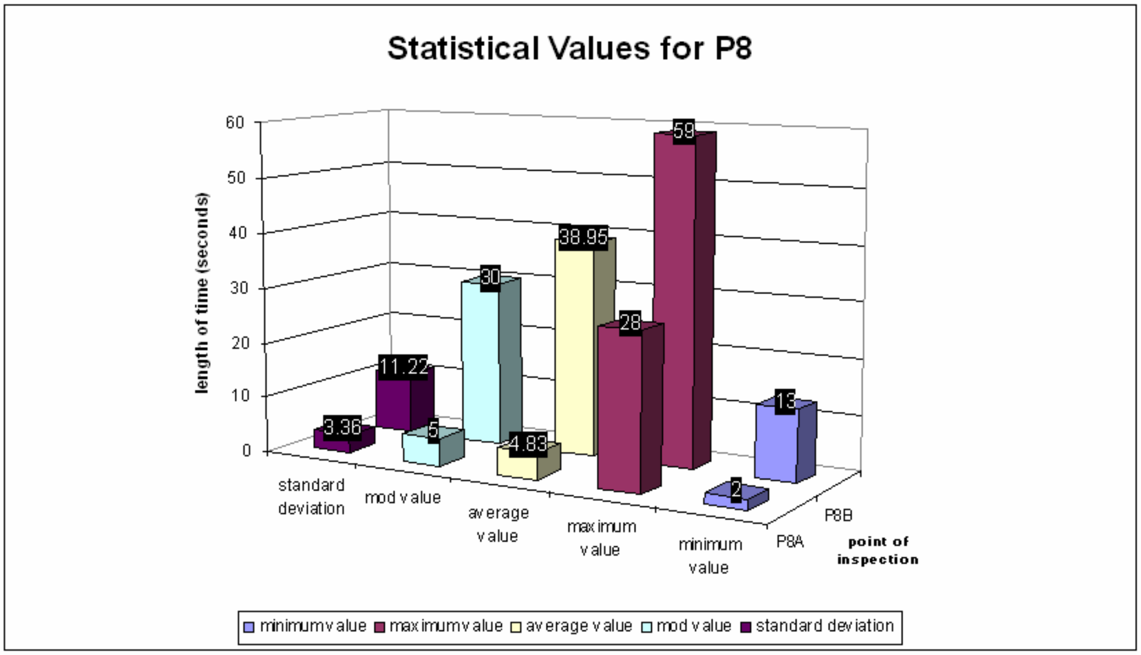


**Figure 4.12**: Illustration of analysis for point inspection P8 – Headlight Test using 3-D Column Chart.

### ***<u>P9 : Undercarriage Check</u>***

The analysis for this point of inspection is described in **Table 4.9** below, while the illustration of the figures is shown in **Figure 4.13.**

| | **P1A (traveling time for P1)** | **P1B (service time for P1)** |
|---|---|---|
| **Minimum value** | 4 seconds | 16 seconds |
| **Maximum value** | 10 seconds | 150 seconds |
| **Number of data** | 65 vehicles | 65 vehicles |
| **Average value** | 9.05 seconds | 51.8 seconds |
| **Mod value** | 10 seconds | 50 seconds |
| **Standard deviation** | 1.81 seconds | 24.9 seconds |

**Table 4.9:** Analysis for point inspection P9 – Under Carriage Check

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

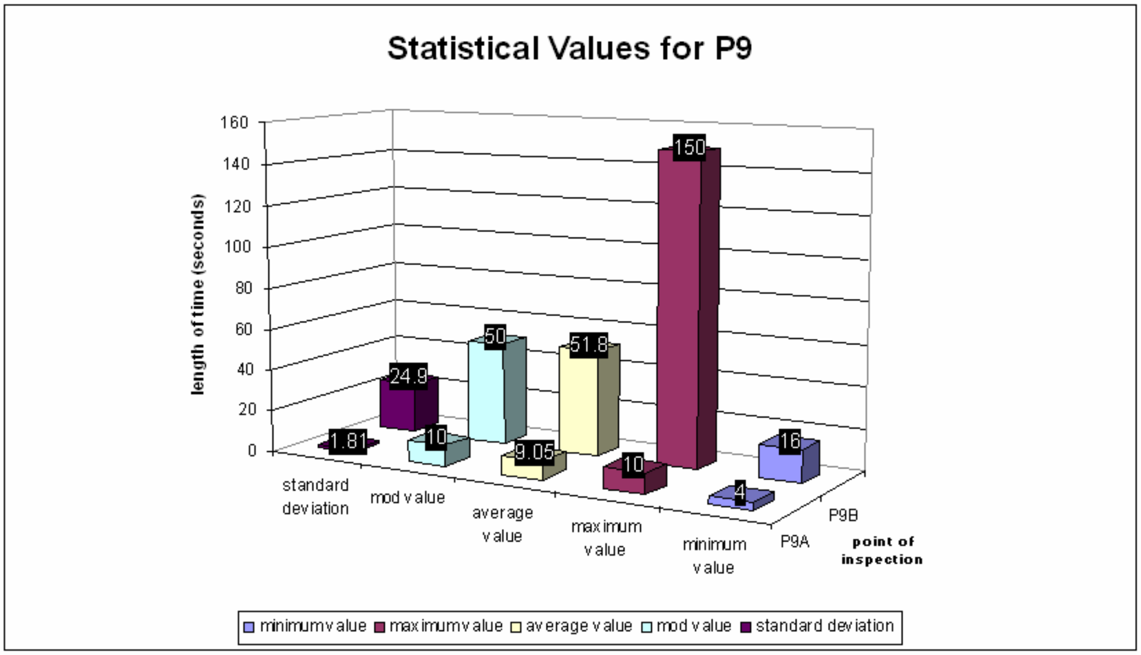


**Figure 4.13**: Illustration of analysis for point inspection P9 – Under Carriage Test using 3-D Column Chart.

***<u>P10 : Computerized Analysis</u>***

The analysis for this point of inspection is described in **Table 4.10** below, while the illustration of the figures is shown in **Figure 4.14.**

| | **P1B (service time for P1)** |
|---|---|
| **Minimum value** | 75 seconds |
| **Maximum value** | 234 seconds |
| **Number of data** | 65 vehicles |
| **Average value** | 158 seconds |
| **Mod value** | 150 seconds |
| **Standard deviation** | 38.1 seconds |

**Table 4.10:** Analysis for point inspection P10 – Computerized Analysis

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

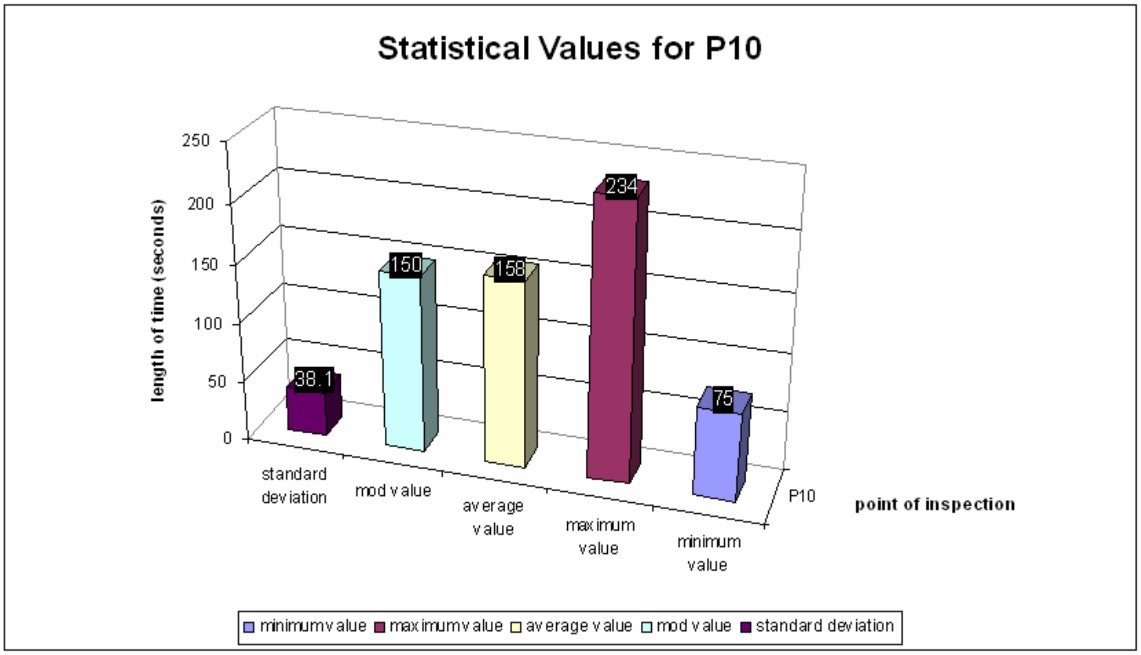


**Figure 4.14**: Illustration of analysis for point inspection P10 – Computerized Analysis using 3-D Column Chart.

***Vehicle Examiner Turning Back***

The analysis for situation of vehicle turning back is described in **Table 4.11** below, while the illustration of the figures is shown in **Figure 4.15.**

| | **P1B (service time for P1)** |
|---|---|
| **Minimum value** | 27 seconds |
| **Maximum value** | 174 seconds |
| **Number of data** | 65 vehicles |
| **Average value** | 39.8 seconds |
| **Mod value** | 30 seconds |
| **Standard deviation** | 22 seconds |

**Table 4.11:** Analysis for Vehicle Turning Back

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

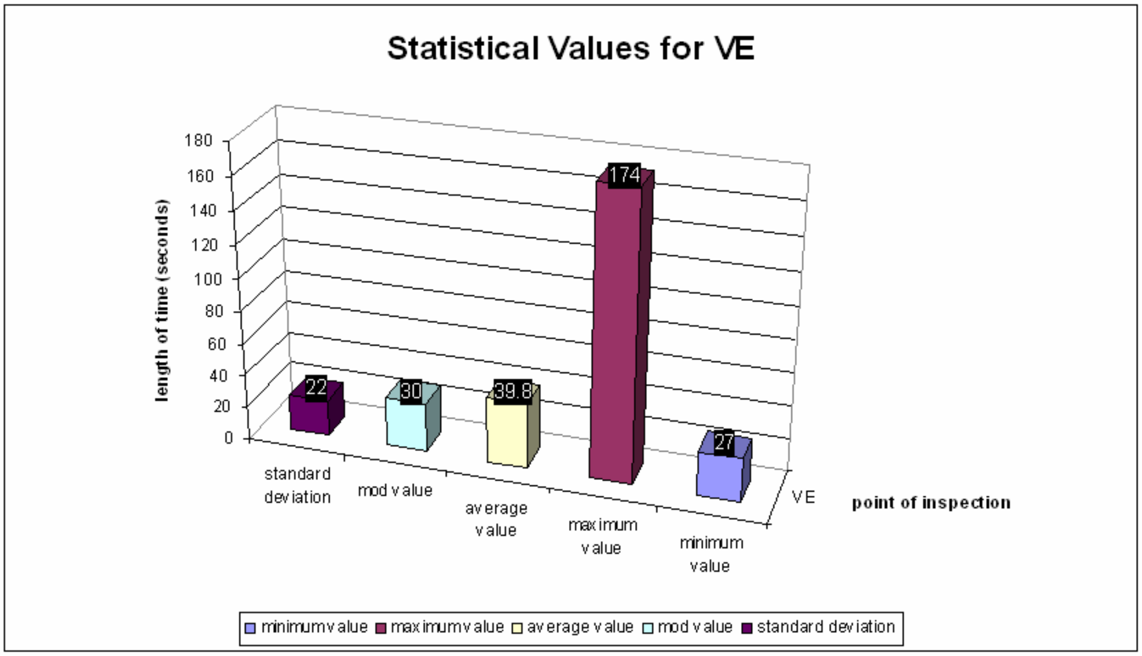


**Figure 4.15**: Illustration of analysis for Vehicle Turning Back using 3-D Column Chart.

***Cumulative Time in Seconds***

The analysis for cumulative time in seconds is described in **Table 4.12** below, while the illustration of the figures is shown in **Figure 4.16.**

| | **P1B (service time for P1)** |
|---|---|
| **Minimum value** | 408 seconds |
| **Maximum value** | 756 seconds |
| **Number of data** | 65 vehicles |
| **Average value** | 567.231 seconds |
| **Mod value** | 547 seconds |
| **Standard deviation** | 68.4303 seconds |

**Table 4.12:** Analysis for Cumulative Time in Seconds

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

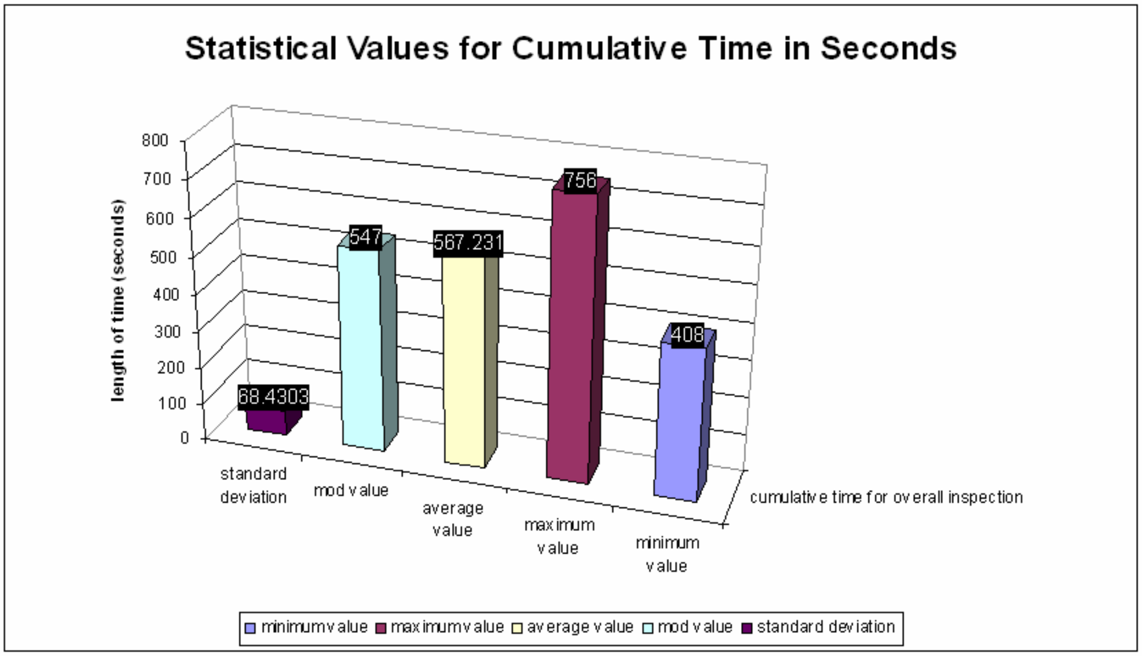


**Figure 4.16**: Illustration of analysis for cumulative time in seconds using 3-D Column Chart.

***Cumulative Time in Minutes***

The analysis for cumulative time in minutes is described in **Table 4.13** below, while the illustration of the figures is shown in **Figure 4.17.**

| | **P1B (service time for P1)** |
|---|---|
| **Minimum value** | 6.8 seconds |
| **Maximum value** | 12.6 seconds |
| **Number of data** | 65 vehicles |
| **Average value** | 9.45385 seconds |
| **Mod value** | 9.1167 seconds |
| **Standard deviation** | 1.140505 seconds |

**Table 4.13:** Analysis for Cumulative Time in Minutes

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

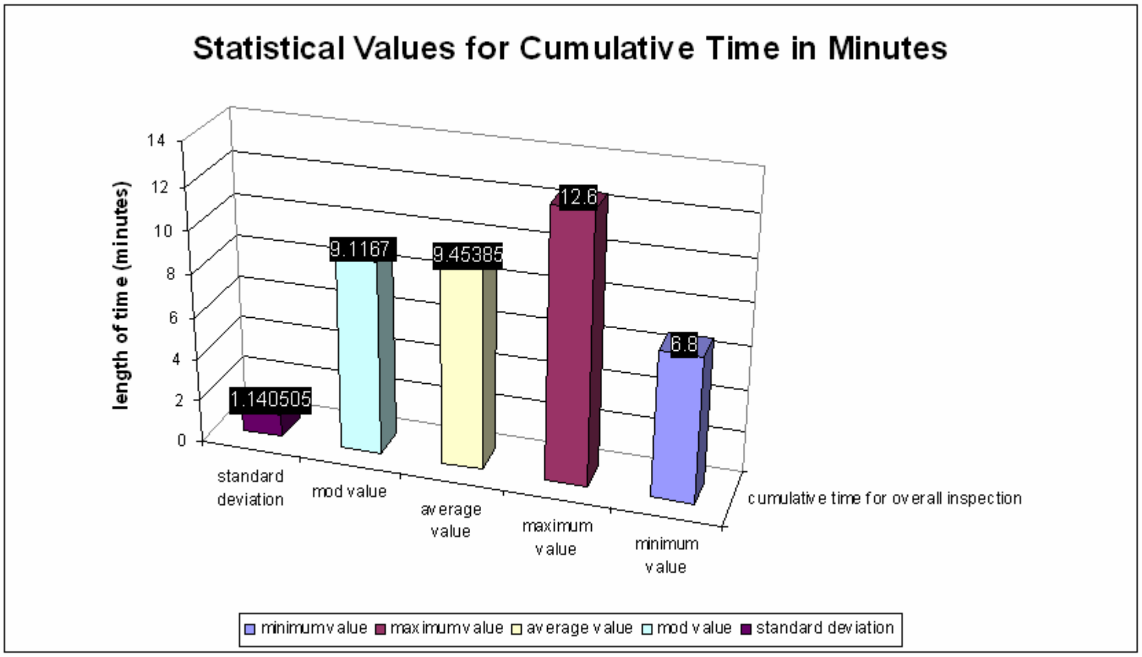


**Figure 4.17**: Illustration of analysis for cumulative time in minutes using 3-D Column Chart.

### 4.4.1 Point Average Summary

The average value for each step of inspection is calculate to get the exact figure on how much time is taken by every inspection for one finance vehicle. The details of the figures are described below, and are shown in **Figure 4.18**.

| | | |
|---|---|---|
| i. | Average value for traveling time of P1 | : 5.91 seconds |
| ii. | Average value for service time for P1 | : 85.14 seconds |
| iii. | Average value for traveling time of P2 | : 4.95 seconds |
| iv. | Average value for service time for P2 | : 2.938 seconds |
| v. | Average value for traveling time of P3 | : 4.71 seconds |
| vi. | Average value for service time for P3 | : 39.52 seconds |
| vii. | Average value for traveling time of P4 | : 3.29 seconds |
| viii. | Average value for service time for P4 | : 28 seconds |
| ix. | Average value for traveling time of P5 | :3.66 seconds |

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

x. Average value for service time for P5 : 35.85 seconds

xi. Average value for traveling time of P6 : 2.85 seconds

xii. Average value for service time for P6 :25.32 seconds

xiii. Average value for traveling time of P7 : 5.32 seconds

xiv. Average value for service time for P7 : 17.25 seconds

xv. Average value for traveling time of P8 : 4.83 seconds

xvi. Average value for service time for P8 : 38.95 seconds

xvii. Average value for traveling time of P9 : 9.05 seconds

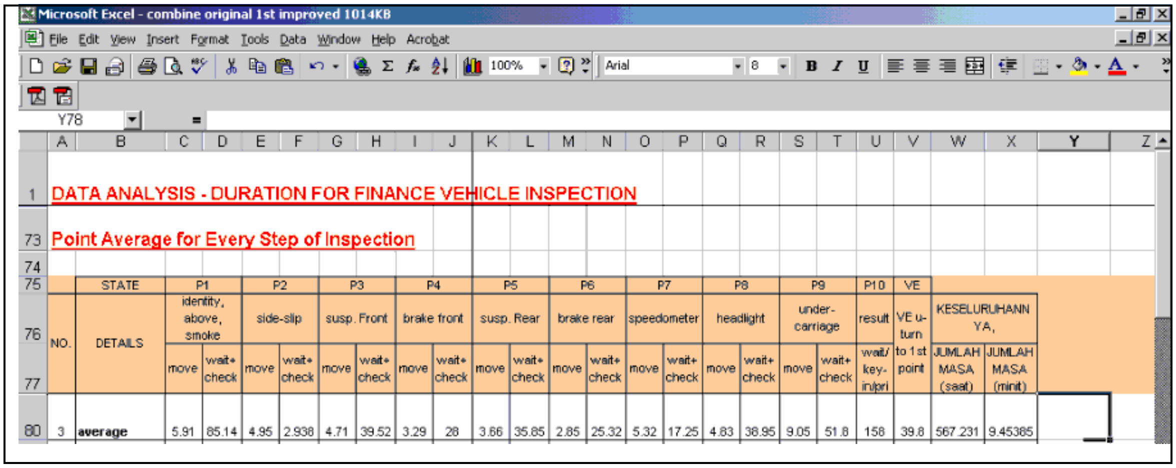
xviii. Average value for service time for P9 : 51.8 seconds

xix. Average value for service time for P10 : 158 seconds

xx. Average value for service time for VE : 39.8 seconds

xxi. Average value for service time for cumulative in seconds : 567.231

xxii. Average value for service time for cumulative in minutes : 9.45385


Microsoft Excel - combine original 1st improved 1014KB
File Edit View Insert Format Tools Data Window Help Acrobat
100%
Arial
8
Y78
=

DATA ANALYSIS - DURATION FOR FINANCE VEHICLE INSPECTION

Point Average for Every Step of Inspection

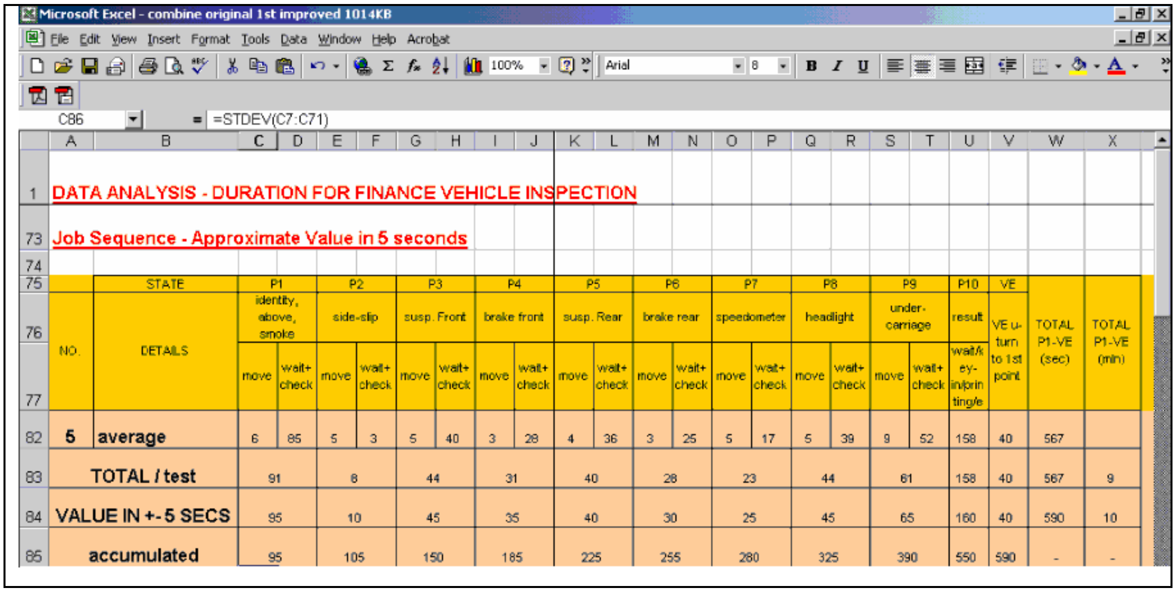
| | STATE | P1 | | P2 | | P3 | | P4 | | P5 | | P6 | | P7 | | P8 | | P9 | | P10 | VE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NO. | DETAILS | identity, above, smoke | | side-slip | | susp. Front | | brake front | | susp. Rear | | brake rear | | speedometer | | headlight | | under-carriage | | result | VE u-turn to 1st point | KESELURUHANNYA, | |
| | | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | move | wait+check | wait/key-in/pri | | JUMLAH MASA (saat) | JUMLAH MASA (minit) |
| 3 | average | 5.91 | 85.14 | 4.95 | 2.938 | 4.71 | 39.52 | 3.29 | 28 | 3.66 | 35.85 | 2.85 | 25.32 | 5.32 | 17.25 | 4.83 | 38.95 | 9.05 | 51.8 | 158 | 39.8 | 567.231 | 9.45385 |

**Figure 4.18**: Point Average Summary

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

## 4.5 Data Simplifying - Approximates Value

Based from the point average summary, the value are rounded and approximate into a range of maximum five seconds plus, as depicted in **Figure 4.19**. This step is done in order to ensure that the process of data analysis being in accurate way and the result will be produced exactly as desired from this research.



Microsoft Excel - combine original 1st improved 1014KB
File Edit View Insert Format Tools Data Window Help Acrobat
100%
Arial
8

C86 = =STDEV(C7:C71)

1 DATA ANALYSIS - DURATION FOR FINANCE VEHICLE INSPECTION

73 Job Sequence - Approximate Value in 5 seconds

|  | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 75 |  | STATE | P1 |  | P2 |  | P3 |  | P4 |  | P5 |  | P6 |  | P7 |  | P8 |  | P9 |  | P10 | VE |  |  |
| 76 |  |  | identity, above, smoke |  | side-slip |  | susp. Front |  | brake front |  | susp. Rear |  | brake rear |  | speedometer |  | headlight |  | under-carriage |  | result | VE u-turn to 1st point | TOTAL P1-VE (sec) | TOTAL P1-VE (min) |
| 77 | NO. | DETAILS | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | move | wait+ check | wait/key-in/printing/e |  |  |  |
| 82 | 5 | average | 6 | 85 | 5 | 3 | 5 | 40 | 3 | 28 | 4 | 36 | 3 | 25 | 5 | 17 | 5 | 39 | 9 | 52 | 158 | 40 | 567 |  |
| 83 | TOTAL / test |  | 91 |  | 8 |  | 44 |  | 31 |  | 40 |  | 28 |  | 23 |  | 44 |  | 61 |  | 158 | 40 | 567 | 9 |
| 84 | VALUE IN +- 5 SECS |  | 95 |  | 10 |  | 45 |  | 35 |  | 40 |  | 30 |  | 25 |  | 45 |  | 65 |  | 160 | 40 | 590 | 10 |
| 85 | accumulated |  | 95 |  | 105 |  | 150 |  | 185 |  | 225 |  | 255 |  | 280 |  | 325 |  | 390 |  | 550 | 590 | - | - |

**Figure 4.19**: A Sample of Data with Approximate Value.

As a result from the above step, the value for each point of inspection detail is listed below and **Figure 4.20** illustrate the result of the analysis.

| | | |
|---|---|---|
| i. | Approximate value for P1 | : 95 seconds |
| ii. | Approximate value for P2 | : 10 seconds |
| iii. | Approximate value for P3 | : 45 seconds |
| iv. | Approximate value for P4 | : 35 seconds |
| v. | Approximate value for P5 | : 40 seconds |
| vi. | Approximate value for P6 | : 30 seconds |
| vii. | Approximate value for P7 | : 25 seconds |
| viii. | Approximate value for P8 | : 45 seconds |

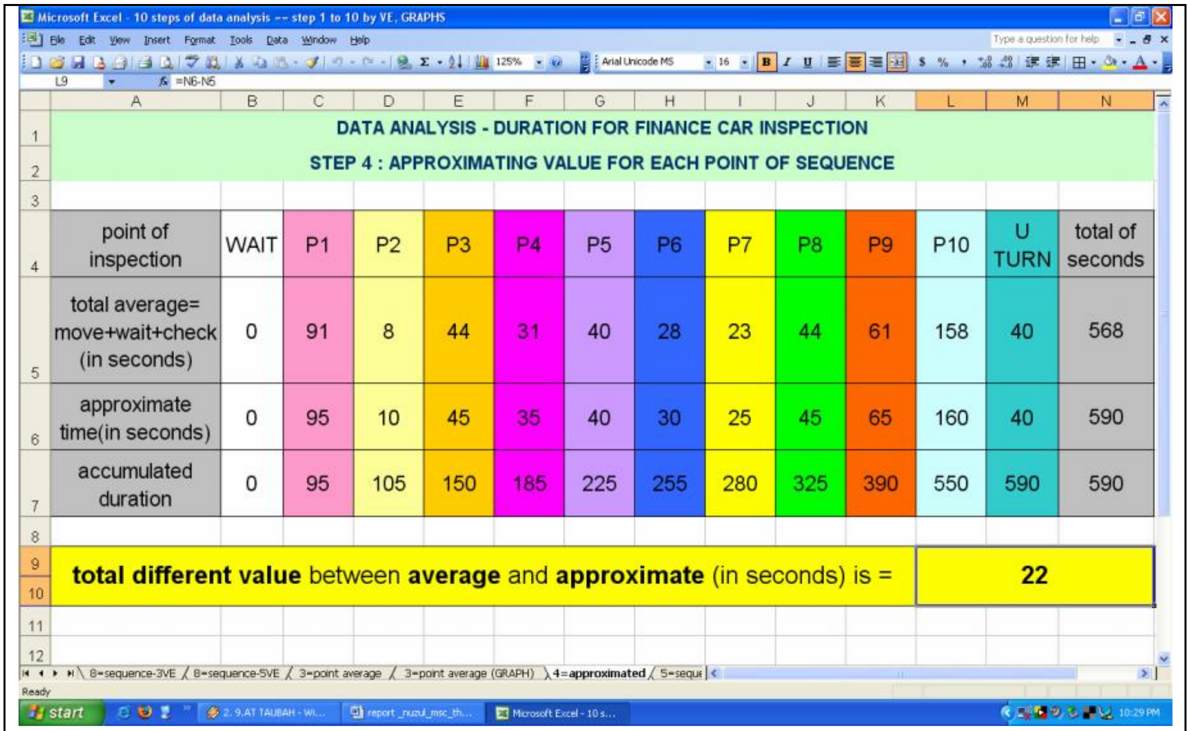
This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

ix. Approximate value for P9 : 65 seconds

x. Approximate value for P10 : 160 seconds

xi. Approximate value for VE : 40 seconds

xii. Approximate value for total inspection time : 590 seconds



DATA ANALYSIS - DURATION FOR FINANCE CAR INSPECTION

STEP 4 : APPROXIMATING VALUE FOR EACH POINT OF SEQUENCE

| point of inspection | WAIT | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | U TURN | total of seconds |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| total average= move+wait+check (in seconds) | 0 | 91 | 8 | 44 | 31 | 40 | 28 | 23 | 44 | 61 | 158 | 40 | 568 |
| approximate time(in seconds) | 0 | 95 | 10 | 45 | 35 | 40 | 30 | 25 | 45 | 65 | 160 | 40 | 590 |
| accumulated duration | 0 | 95 | 105 | 150 | 185 | 225 | 255 | 280 | 325 | 390 | 550 | 590 | 590 |

**total different value** between **average** and **approximate** (in seconds) is = **22**

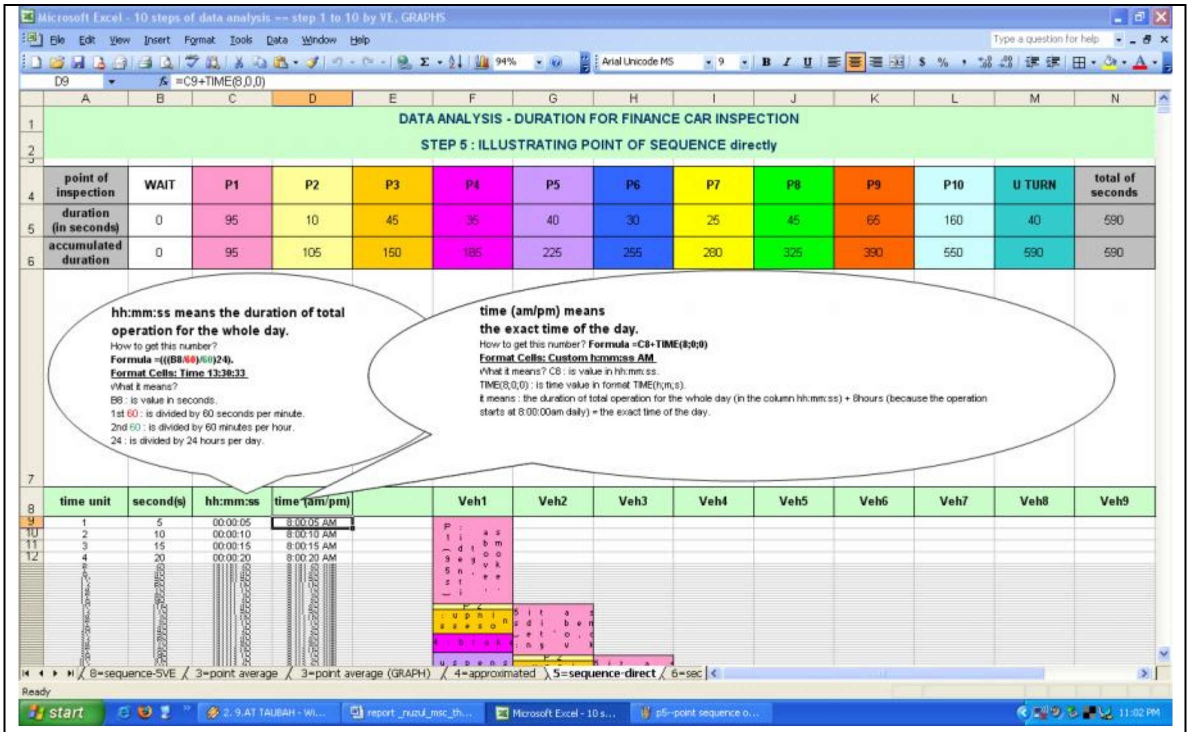
**Figure 4.20:** Approximate Values for each of point inspection

Based form the illustration, it is knew that there is a little difference with the length of cumulative time between the averages for points of inspection which is equal to 568 seconds, while the approximate value for points of inspection is 590 seconds. The total difference between the two values is 22 seconds; this is less than half a minute. So, we assumed the slightly difference in this step as a nil value in this stage of analysis for this study.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 4.6 Data Sequence with Direct Point

The idea for this step of analysis is depicted in the following illustration as shown in **Figure 4.21**.



DATA ANALYSIS - DURATION FOR FINANCE CAR INSPECTION

STEP 5 : ILLUSTRATING POINT OF SEQUENCE directly

| point of inspection | WAIT | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | U TURN | total of seconds |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| duration (in seconds) | 0 | 95 | 10 | 45 | 35 | 40 | 30 | 25 | 45 | 65 | 160 | 40 | 590 |
| accumulated duration | 0 | 95 | 105 | 150 | 185 | 225 | 255 | 280 | 325 | 390 | 550 | 590 | 590 |

**hh:mm:ss means the duration of total operation for the whole day.**
How to get this number?
**Formula =(((B8/60)/60)24).**
**Format Cells: Time 13:30:33**
What it means?
B8 : is value in seconds.
1st 60 : is divided by 60 seconds per minute.
2nd 60 : is divided by 60 minutes per hour.
24 : is divided by 24 hours per day.

**time (am/pm) means the exact time of the day.**
How to get this number? **Formula =C8+TIME(8;0;0)**
**Format Cells: Custom h:mm:ss AM**
What it means? C8 : is value in hh:mm:ss.
TIME(8;0;0) : is time value in format TIME(h;m;s).
it means : the duration of total operation for the whole day (in the column hh:mm:ss) + 8hours (because the operation starts at 8:00:00am daily) = the exact time of the day.

| time unit | second(s) | hh:mm:ss | time (am/pm) | | Veh1 | Veh2 | Veh3 | Veh4 | Veh5 | Veh6 | Veh7 | Veh8 | Veh9 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 5 | 00:00:05 | 8:00:05 AM | | | | | | | | | | |
| 2 | 10 | 00:00:10 | 8:00:10 AM | | | | | | | | | | |
| 3 | 15 | 00:00:15 | 8:00:15 AM | | | | | | | | | | |
| 4 | 20 | 00:00:20 | 8:00:20 AM | | | | | | | | | | |

**Figure 4.21:** Data Sequence with Direct Point

The two formulas in this step are included in the two columns which are included in these two headers **hh:mm:ss** and **time (am/pm).** The first formula is noted for considering the duration of total operation for the whole day while the second formula is noted to give the exact time of the day.

### *Header* hh:mm:ss

How do we get this number? There is a toolbar to insert certain form ula, and this column is filled with **Formula = (((B8/60)/60)24).**

**hh:mm:ss** means the duration of total operation for the whole day.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

How to get this number?

**Formula =(((B8/60)/60)24)**.

**Format Cells: Time 13:30:33**.

Where;

**B8** : is value in seconds.

1st **60** : is divided by 60 seconds per minute.

2nd **60** : is divided by 60 minutes per hour.

**24** : is divided by 24 hours per day.

Every cell in this column must be set its format as **Format Cells: Time 13:30:33** as the format itself is provided in Ms Excel 2007.

### ***Header*** **time (am/pm)**

The header **time (am/pm)** means the exact time of the day.

How to get this number?

**Formula =C8+TIME(8;0;0)** and **Format Cells: Custom h:mm:ss AM**.

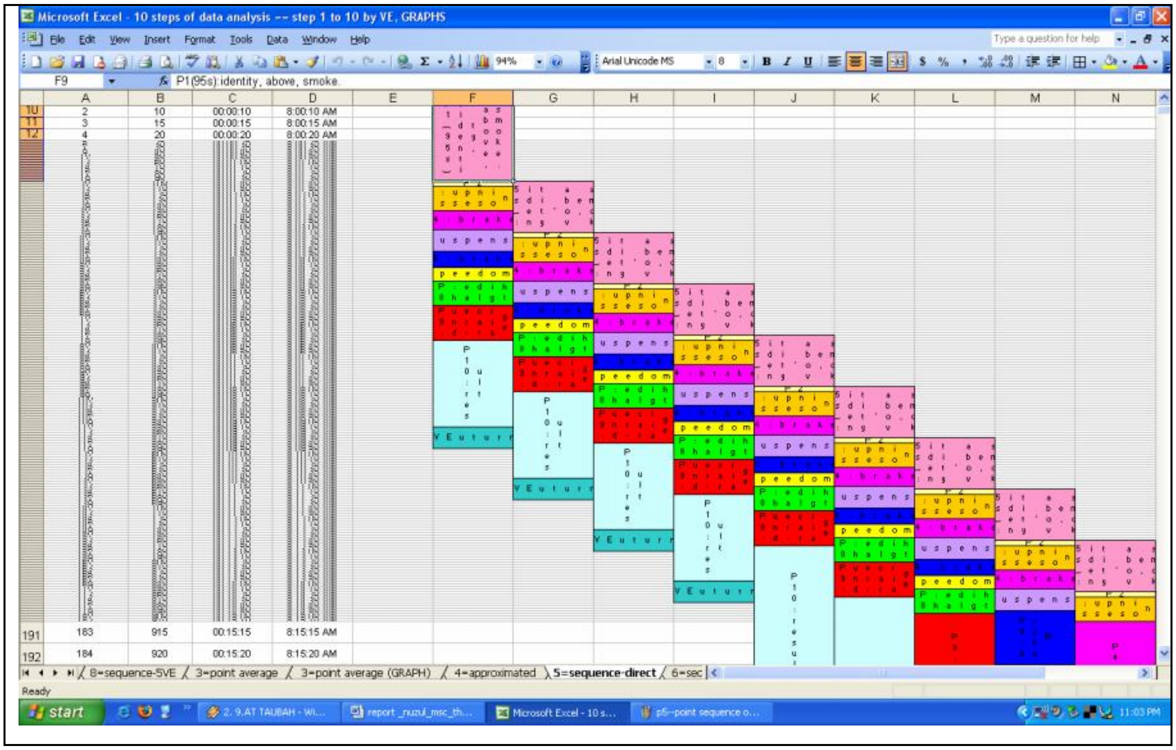
Where;

**C8** : is value in hh:mm:ss.

**TIME(8;0;0)** : is time value in format **TIME(h;m;s)**.

The header of the column means the total operation duration for the whole day (in the column hh:mm:ss) added with 8 hours (because the operation starts at 8:00:00am daily) is equal to the exact time of the day. **Figure 4.22** illustrate the content of this step of analysis.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



Microsoft Excel - 10 steps of data analysis -- step 1 to 10 by VE, GRAPHS
File Edit View Insert Format Tools Data Window Help
F9
P1(95s):identity, above, smoke
8=sequence-5VE
3=point average
3=point average (GRAPH)
4=approximated
5=sequence-direct
6=sec
Ready
start

|  | A | B | C | D |
|---|---|---|---|---|
| 10 | 2 | 10 | 00:00:10 | 8:00:10 AM |
| 11 | 3 | 15 | 00:00:15 | 8:00:15 AM |
| 12 | 4 | 20 | 00:00:20 | 8:00:20 AM |
| 191 | 183 | 915 | 00:15:15 | 8:15:15 AM |
| 192 | 184 | 920 | 00:15:20 | 8:15:20 AM |

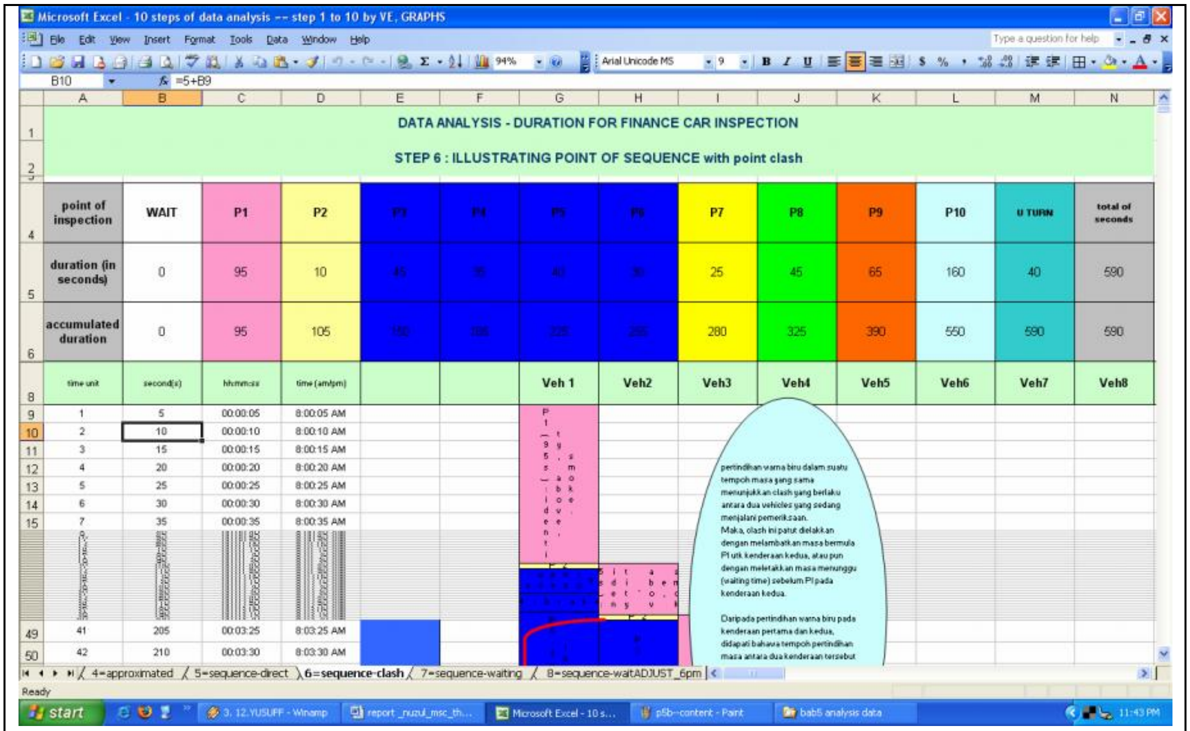
**Figure 4.22**: Point Sequence Analysis.

## 4.7 Data Sequence with Point Clash

In this stage, we gave same color to a few points which being inspected in a same area in the lane of inspection; which is inspected in close proximity between a f ew points of inspection. From the analysis, we colored the third point, P3 to sixth point, P6 as blue cells as they are being inspected in just about the same area in the lane of inspection. The description of this process in depicted in **Figure 4.23**.

Then, we identified the point clash between the two vehicles with same colored points. The blue colored cells show a points clash in a 55 seconds time between two vehicles of Vehicle1 and Vehicle2. See **Figure 4.24**.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com



Microsoft Excel - 10 steps of data analysis -- step 1 to 10 by VE, GRAPHS

B10 =5+B9

DATA ANALYSIS - DURATION FOR FINANCE CAR INSPECTION

STEP 6 : ILLUSTRATING POINT OF SEQUENCE with point clash

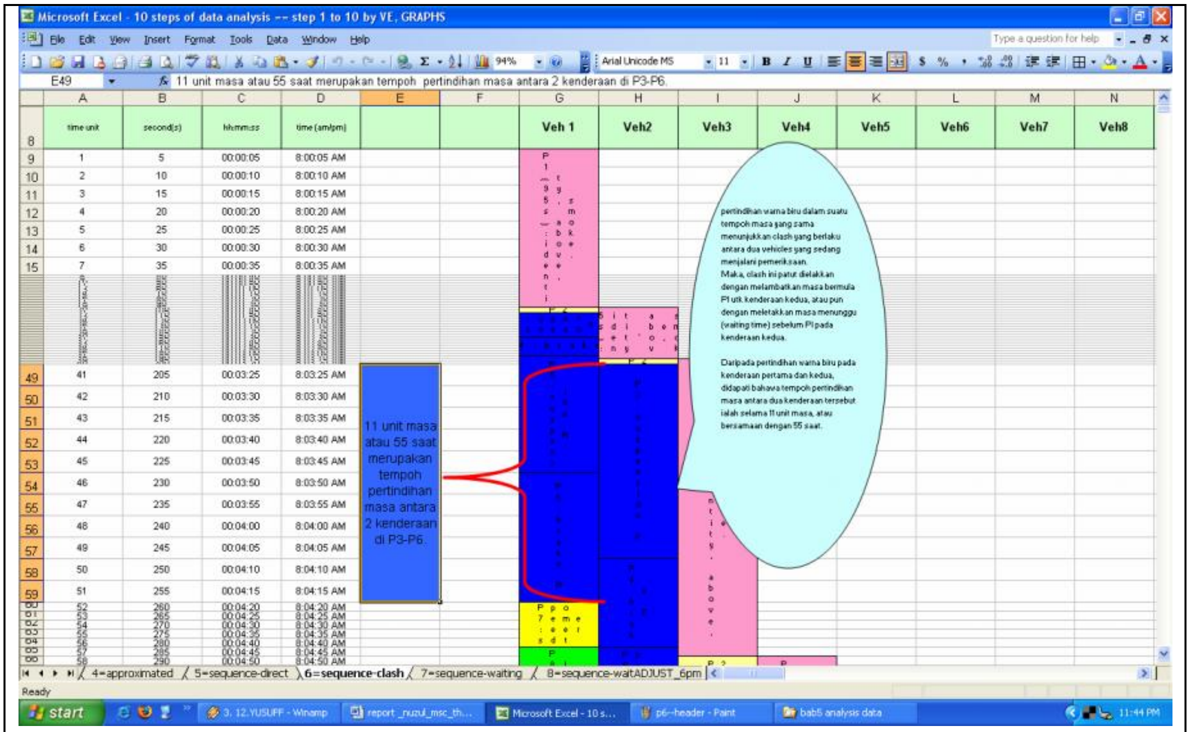
| point of inspection | WAIT | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | U TURN | total of seconds |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| duration (in seconds) | 0 | 95 | 10 | 45 | 35 | 40 | 30 | 25 | 45 | 65 | 160 | 40 | 590 |
| accumulated duration | 0 | 95 | 105 | 150 | 185 | 225 | 255 | 280 | 325 | 390 | 550 | 590 | 590 |

| time unit | second(s) | hh:mm:ss | time (am/pm) | | | Veh 1 | Veh2 | Veh3 | Veh4 | Veh5 | Veh6 | Veh7 | Veh8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 5 | 00:00:05 | 8:00:05 AM | | | | | | | | | | |
| 2 | 10 | 00:00:10 | 8:00:10 AM | | | | | | | | | | |
| 3 | 15 | 00:00:15 | 8:00:15 AM | | | | | | | | | | |
| 4 | 20 | 00:00:20 | 8:00:20 AM | | | | | | | | | | |
| 5 | 25 | 00:00:25 | 8:00:25 AM | | | | | | | | | | |
| 6 | 30 | 00:00:30 | 8:00:30 AM | | | | | | | | | | |
| 7 | 35 | 00:00:35 | 8:00:35 AM | | | | | | | | | | |
| 41 | 205 | 00:03:25 | 8:03:25 AM | | | | | | | | | | |
| 42 | 210 | 00:03:30 | 8:03:30 AM | | | | | | | | | | |

pertindihan warna biru dalam suatu tempoh masa yang sama menunjukkan clash yang berlaku antara dua vehicles yang sedang menjalani pemeriksaan.
Maka, clash ini patut dielakkan dengan melambatkan masa bermula P1 utk kenderaan kedua, atau pun dengan meletakkan masa menunggu (waiting time) sebelum P1 pada kenderaan kedua.

Daripada pertindihan warna biru pada kenderaan pertama dan kedua, didapati bahawa tempoh pertindihan masa antara dua kenderaan tersebut

**Figure 4.23:** Data sequence with point cla sh



Microsoft Excel - 10 steps of data analysis -- step 1 to 10 by VE, GRAPHS

E49 11 unit masa atau 55 saat merupakan tempoh pertindihan masa antara 2 kenderaan di P3-P6.

| time unit | second(s) | hh:mm:ss | time (am/pm) | | | Veh 1 | Veh2 | Veh3 | Veh4 | Veh5 | Veh6 | Veh7 | Veh8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 5 | 00:00:05 | 8:00:05 AM | | | | | | | | | | |
| 2 | 10 | 00:00:10 | 8:00:10 AM | | | | | | | | | | |
| 3 | 15 | 00:00:15 | 8:00:15 AM | | | | | | | | | | |
| 4 | 20 | 00:00:20 | 8:00:20 AM | | | | | | | | | | |
| 5 | 25 | 00:00:25 | 8:00:25 AM | | | | | | | | | | |
| 6 | 30 | 00:00:30 | 8:00:30 AM | | | | | | | | | | |
| 7 | 35 | 00:00:35 | 8:00:35 AM | | | | | | | | | | |
| 41 | 205 | 00:03:25 | 8:03:25 AM | | | | | | | | | | |
| 42 | 210 | 00:03:30 | 8:03:30 AM | | | | | | | | | | |
| 43 | 215 | 00:03:35 | 8:03:35 AM | | | | | | | | | | |
| 44 | 220 | 00:03:40 | 8:03:40 AM | | | | | | | | | | |
| 45 | 225 | 00:03:45 | 8:03:45 AM | | | | | | | | | | |
| 46 | 230 | 00:03:50 | 8:03:50 AM | | | | | | | | | | |
| 47 | 235 | 00:03:55 | 8:03:55 AM | | | | | | | | | | |
| 48 | 240 | 00:04:00 | 8:04:00 AM | | | | | | | | | | |
| 49 | 245 | 00:04:05 | 8:04:05 AM | | | | | | | | | | |
| 50 | 250 | 00:04:10 | 8:04:10 AM | | | | | | | | | | |
| 51 | 255 | 00:04:15 | 8:04:15 AM | | | | | | | | | | |
| 52 | 260 | 00:04:20 | 8:04:20 AM | | | | | | | | | | |
| 53 | 265 | 00:04:25 | 8:04:25 AM | | | | | | | | | | |
| 54 | 270 | 00:04:30 | 8:04:30 AM | | | | | | | | | | |
| 55 | 275 | 00:04:35 | 8:04:35 AM | | | | | | | | | | |
| 56 | 280 | 00:04:40 | 8:04:40 AM | | | | | | | | | | |
| 57 | 285 | 00:04:45 | 8:04:45 AM | | | | | | | | | | |
| 58 | 290 | 00:04:50 | 8:04:50 AM | | | | | | | | | | |

11 unit masa atau 55 saat merupakan tempoh pertindihan masa antara 2 kenderaan di P3-P6.

pertindihan warna biru dalam suatu tempoh masa yang sama menunjukkan clash yang berlaku antara dua vehicles yang sedang menjalani pemeriksaan.
Maka, clash ini patut dielakkan dengan melambatkan masa bermula P1 utk kenderaan kedua, atau pun dengan meletakkan masa menunggu (waiting time) sebelum P1 pada kenderaan kedua.

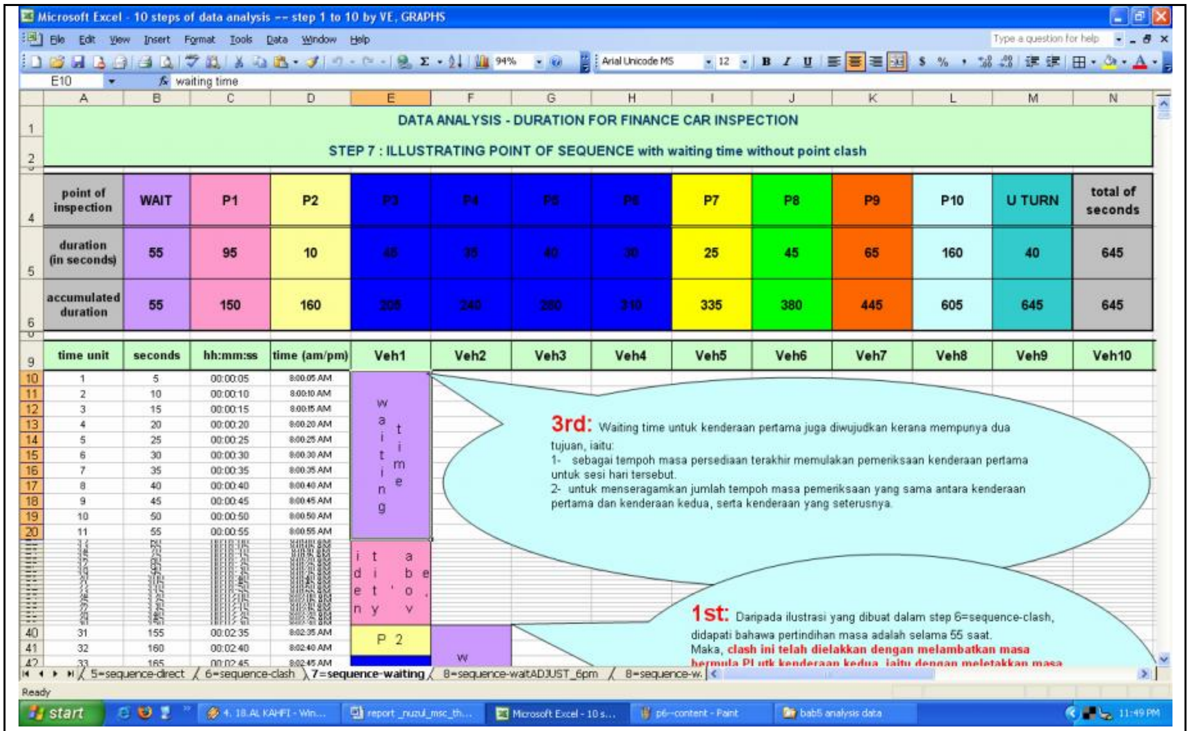
Daripada pertindihan warna biru pada kenderaan pertama dan kedua, didapati bahawa tempoh pertindihan masa antara dua kenderaan tersebut ialah selama 11 unit masa, atau bersamaan dengan 55 saat.

**Figure 4.24:** Data Sequence with points clash in a 55 seconds time between two vehicle

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 4.8 Data Sequence with Waiting Time and without Point Clash

Knowing that there was some clash between the two series of same colored points, we tried to resolve the situation by adding a period of time called ‘waiting time’ to make sure that the conflicting points would be separated into different time units. **Figure 4.25** illustrates the heading and the initial content of this analysis. The first purple -colored represents the period of waiting time for each vehicle inspection.



DATA ANALYSIS - DURATION FOR FINANCE CAR INSPECTION

STEP 7 : ILLUSTRATING POINT OF SEQUENCE with waiting time without point clash

| point of inspection | WAIT | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | U TURN | total of seconds |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| duration (in seconds) | 55 | 95 | 10 | 45 | 35 | 40 | 30 | 25 | 45 | 65 | 160 | 40 | 645 |
| accumulated duration | 55 | 150 | 160 | 205 | 240 | 280 | 310 | 335 | 380 | 445 | 605 | 645 | 645 |

| time unit | seconds | hh:mm:ss | time (am/pm) | Veh1 | Veh2 | Veh3 | Veh4 | Veh5 | Veh6 | Veh7 | Veh8 | Veh9 | Veh10 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 5 | 00:00:05 | 8:00:05 AM | waiting time | | | | | | | | | |
| 2 | 10 | 00:00:10 | 8:00:10 AM | | | | | | | | | | |
| 3 | 15 | 00:00:15 | 8:00:15 AM | | | | | | | | | | |
| 4 | 20 | 00:00:20 | 8:00:20 AM | | | | | | | | | | |
| 5 | 25 | 00:00:25 | 8:00:25 AM | | | | | | | | | | |
| 6 | 30 | 00:00:30 | 8:00:30 AM | | | | | | | | | | |
| 7 | 35 | 00:00:35 | 8:00:35 AM | | | | | | | | | | |
| 8 | 40 | 00:00:40 | 8:00:40 AM | | | | | | | | | | |
| 9 | 45 | 00:00:45 | 8:00:45 AM | | | | | | | | | | |
| 10 | 50 | 00:00:50 | 8:00:50 AM | | | | | | | | | | |
| 11 | 55 | 00:00:55 | 8:00:55 AM | | | | | | | | | | |
| [illegible] | [illegible] | [illegible] | [illegible] | identity, above, | | | | | | | | | |
| 31 | 155 | 00:02:35 | 8:02:35 AM | P 2 | w | | | | | | | | |
| 32 | 160 | 00:02:40 | 8:02:40 AM | | | | | | | | | | |
| 33 | 165 | 00:02:45 | 8:02:45 AM | | | | | | | | | | |

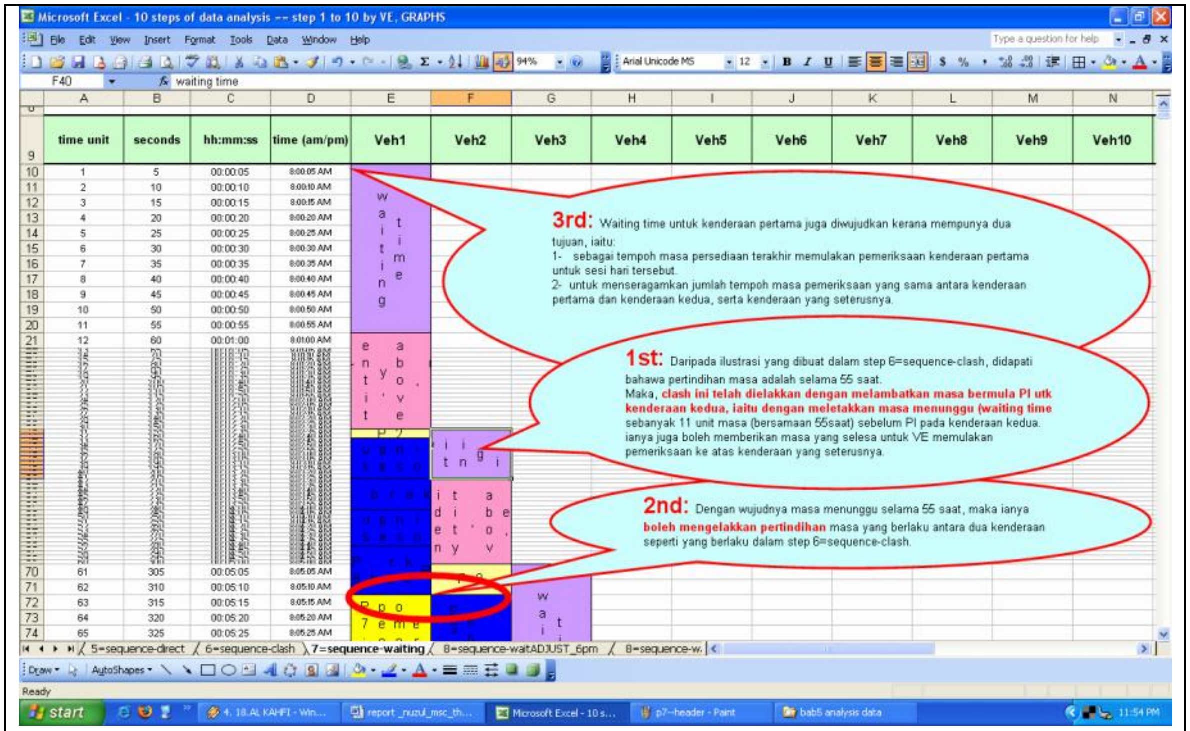
**3rd:** Waiting time untuk kenderaan pertama juga diwujudkan kerana mempunya dua tujuan, iaitu:
1- sebagai tempoh masa persediaan terakhir memulakan pemeriksaan kenderaan pertama untuk sesi hari tersebut.
2- untuk menseragamkan jumlah tempoh masa pemeriksaan yang sama antara kenderaan pertama dan kenderaan kedua, serta kenderaan yang seterusnya.

**1st:** Daripada ilustrasi yang dibuat dalam step 6=sequence-clash, didapati bahawa pertindihan masa adalah selama 55 saat.
Maka, **clash ini telah dielakkan dengan melambatkan masa bermula PI utk kenderaan kedua, iaitu dengan meletakkan masa**

**Figure 4.25**: Captured Screen of Header in Sequence Without Point Clash.

From the previous process, we noticed that there is a point clash in a 55 seconds time. (Refer to **Figure 4.24**). Thus, this clash must be avoided with a waiting time attached to the starting process time (P1) to delay the following vehicle from being conflicting with earlier vehicle. **Figure 4.26** below shows the process of this phase.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com



| | A | B | C | D | E | F | G | H | I | J | K | L | M | N |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 9 | time unit | seconds | hh:mm:ss | time (am/pm) | Veh1 | Veh2 | Veh3 | Veh4 | Veh5 | Veh6 | Veh7 | Veh8 | Veh9 | Veh10 |
| 10 | 1 | 5 | 00:00:05 | 8:00:05 AM | | | | | | | | | | |
| 11 | 2 | 10 | 00:00:10 | 8:00:10 AM | | | | | | | | | | |
| 12 | 3 | 15 | 00:00:15 | 8:00:15 AM | | | | | | | | | | |
| 13 | 4 | 20 | 00:00:20 | 8:00:20 AM | | | | | | | | | | |
| 14 | 5 | 25 | 00:00:25 | 8:00:25 AM | | | | | | | | | | |
| 15 | 6 | 30 | 00:00:30 | 8:00:30 AM | | | | | | | | | | |
| 16 | 7 | 35 | 00:00:35 | 8:00:35 AM | | | | | | | | | | |
| 17 | 8 | 40 | 00:00:40 | 8:00:40 AM | | | | | | | | | | |
| 18 | 9 | 45 | 00:00:45 | 8:00:45 AM | | | | | | | | | | |
| 19 | 10 | 50 | 00:00:50 | 8:00:50 AM | | | | | | | | | | |
| 20 | 11 | 55 | 00:00:55 | 8:00:55 AM | | | | | | | | | | |
| 21 | 12 | 60 | 00:01:00 | 8:01:00 AM | | | | | | | | | | |
| 70 | 61 | 305 | 00:05:05 | 8:05:05 AM | | | | | | | | | | |
| 71 | 62 | 310 | 00:05:10 | 8:05:10 AM | | | | | | | | | | |
| 72 | 63 | 315 | 00:05:15 | 8:05:15 AM | | | | | | | | | | |
| 73 | 64 | 320 | 00:05:20 | 8:05:20 AM | | | | | | | | | | |
| 74 | 65 | 325 | 00:05:25 | 8:05:25 AM | | | | | | | | | | |

**3rd:** Waiting time untuk kenderaan pertama juga diwujudkan kerana mempunya dua tujuan, iaitu:
1- sebagai tempoh masa persediaan terakhir memulakan pemeriksaan kenderaan pertama untuk sesi hari tersebut.
2- untuk menseragamkan jumlah tempoh masa pemeriksaan yang sama antara kenderaan pertama dan kenderaan kedua, serta kenderaan yang seterusnya.

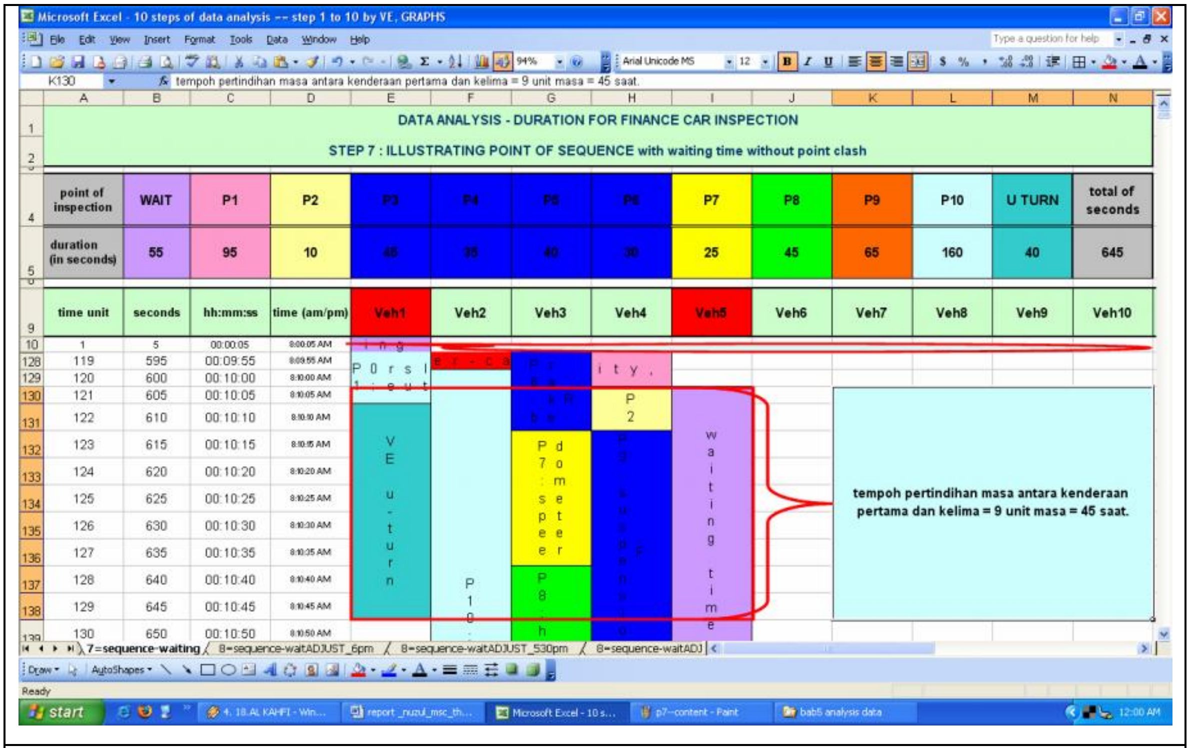
**1st:** Daripada ilustrasi yang dibuat dalam step 6=sequence-clash, didapati bahawa pertindihan masa adalah selama 55 saat.
Maka, **clash ini telah dielakkan dengan melambatkan masa bermula PI utk kenderaan kedua, iaitu dengan meletakkan masa menunggu (waiting time** sebanyak 11 unit masa (bersamaan 55saat) sebelum PI pada kenderaan kedua. ianya juga boleh memberikan masa yang selesa untuk VE memulakan pemeriksaan ke atas kenderaan yang seterusnya.

**2nd:** Dengan wujudnya masa menunggu selama 55 saat, maka ianya **boleh mengelakkan pertindihan** masa yang berlaku antara dua kenderaan seperti yang berlaku dalam step 6=sequence-clash.

**Figure 4.26**: First Part in Sequence With Waiting Time Analysis.

This meaning that a block of 55 seconds waiting time must be allocated to the second vehicles, and also should be given to all vehicles. This waiting time will be able to provide a proper and secure u-turn time for every vehicle examiner to come back from earlier inspected vehicle to the following vehicle that will be inspected. Furthermore, a 55 seconds waiting time (in second callout) will keep away from conflicting the two vehicles as illustrated in **Figure 4.27**.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com



Microsoft Excel - 10 steps of data analysis -- step 1 to 10 by VE, GRAPHS

K130 — tempoh pertindihan masa antara kenderaan pertama dan kelima = 9 unit masa = 45 saat.

DATA ANALYSIS - DURATION FOR FINANCE CAR INSPECTION

STEP 7 : ILLUSTRATING POINT OF SEQUENCE with waiting time without point clash

| point of inspection | WAIT | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | U TURN | total of seconds |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| duration (in seconds) | 55 | 95 | 10 | 45 | 35 | 40 | 30 | 25 | 45 | 65 | 160 | 40 | 645 |

| time unit | seconds | hh:mm:ss | time (am/pm) | Veh1 | Veh2 | Veh3 | Veh4 | Veh5 | Veh6 | Veh7 | Veh8 | Veh9 | Veh10 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 5 | 00:00:05 | 8:00:05 AM | | | | | | | | | | |
| 119 | 595 | 00:09:55 | 8:09:55 AM | | | | | | | | | | |
| 120 | 600 | 00:10:00 | 8:10:00 AM | | | | | | | | | | |
| 121 | 605 | 00:10:05 | 8:10:05 AM | | | | | | | | | | |
| 122 | 610 | 00:10:10 | 8:10:10 AM | | | | | | | | | | |
| 123 | 615 | 00:10:15 | 8:10:15 AM | | | | | | | | | | |
| 124 | 620 | 00:10:20 | 8:10:20 AM | | | | | | | | | | |
| 125 | 625 | 00:10:25 | 8:10:25 AM | | | | | | | | | | |
| 126 | 630 | 00:10:30 | 8:10:30 AM | | | | | | | | | | |
| 127 | 635 | 00:10:35 | 8:10:35 AM | | | | | | | | | | |
| 128 | 640 | 00:10:40 | 8:10:40 AM | | | | | | | | | | |
| 129 | 645 | 00:10:45 | 8:10:45 AM | | | | | | | | | | |
| 130 | 650 | 00:10:50 | 8:10:50 AM | | | | | | | | | | |

tempoh pertindihan masa antara kenderaan pertama dan kelima = 9 unit masa = 45 saat.

**Figure 4.27**: Second Part in Sequence With Waiting Time Analysis

As summary, the purposes of waiting time are as follows:

i. The main point of waiting time is to avoid conflicting between two vehicles in the same time and lane, and of course to make the process of inspection become smoother and comprehensive.

ii. Waiting time gives a proper time to vehicle examiner come back from the earlier inspected vehicle to the following vehicle that will be inspected.

iii. Waiting time as a final warm up chance to the first vehicle before starting the inspection.

iv. Waiting time should be assigned to all vehicles to guarantee that all vehicles in this series obtain a same time period of inspection.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 4.9 Data Sequence with Waiting Time and Adjustment

In this section, we present an adjustment made for the analysis. From the first step illustration in previous process, we also noticed that there is a clash in 9 time units, equally to 45 seconds between the processes of the first and fifth vehicle.

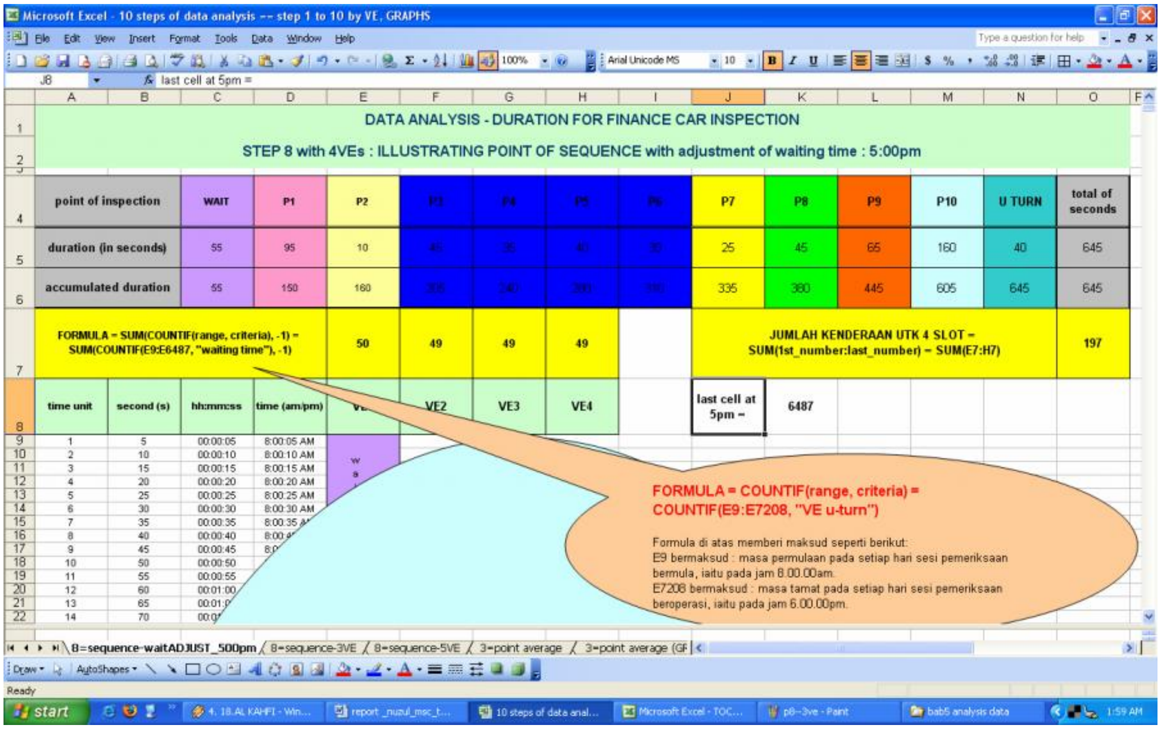
Therefore, this clash must be avoided by putting the fifth vehicle as soon as the first vehicle has completed its last point inspection which means the fifth vehicle will be located after the first VE completed his u-turn from inspecting the first vehicle.

Once the first vehicle has completed its inspection, in next to no time, the fifth vehicle will be carried on with its all points' inspection. This way, there will be no single time unit is unobserved and put in idle situation.

Assigning the fifth vehicle subsequent to the first vehicle, then it will make no problem to allocate the sixth vehicle in no time after the second vehicle without any point clash in any unit time. This second allocation is just keeping up the pattern illustrated in step with waiting time previously.

Why do we need to put the fifth vehicle subsequent to the first vehicle, instead of assigning the sixth or forth vehicle to the first VE? Based from the illustration made in next process as shown in **Figure 4.28**, we have changed the time slot, from vehicles' slot (Veh1, Veh2,..) into vehicle examiners' slot (VE1, VE2, ..). We assume that the step with 4 Vehicle Examiner (VE) is assigning tasks to vehicle examiners in order to complete the inspection process for the day. In slot with waiting time, we noticed that the first vehicle, Veh1 has completed its inspection by the 129th time unit. However, the fifth vehicle is still undergoing its inspection starting from 121st time unit. From the illustration that we have made, we decided to put some time units to wait, before it will be inspected by the first VE as soon as the first vehicle has completed its inspection.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



last cell at 5pm =

DATA ANALYSIS - DURATION FOR FINANCE CAR INSPECTION

STEP 8 with 4VEs : ILLUSTRATING POINT OF SEQUENCE with adjustment of waiting time : 5:00pm

| point of inspection | WAIT | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | U TURN | total of seconds |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| duration (in seconds) | 55 | 95 | 10 | 45 | 35 | 40 | 30 | 25 | 45 | 65 | 160 | 40 | 645 |
| accumulated duration | 55 | 150 | 160 | 205 | 240 | 280 | 310 | 335 | 380 | 445 | 605 | 645 | 645 |

| FORMULA = SUM(COUNTIF(range, criteria), -1) = SUM(COUNTIF(E9:E6487, "waiting time"), -1) | 50 | 49 | 49 | 49 | | JUMLAH KENDERAAN UTK 4 SLOT = SUM(1st_number:last_number) = SUM(E7:H7) | 197 |
|---|---|---|---|---|---|---|---|

| time unit | second (s) | hh:mm:ss | time (am/pm) | VE | VE2 | VE3 | VE4 | | last cell at 5pm = | 6487 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 5 | 00:00:05 | 8:00:05 AM | | | | | | | |
| 2 | 10 | 00:00:10 | 8:00:10 AM | w | | | | | | |
| 3 | 15 | 00:00:15 | 8:00:15 AM | a | | | | | | |
| 4 | 20 | 00:00:20 | 8:00:20 AM | | | | | | | |
| 5 | 25 | 00:00:25 | 8:00:25 AM | | | | | | | |
| 6 | 30 | 00:00:30 | 8:00:30 AM | | | | | | | |
| 7 | 35 | 00:00:35 | 8:00:35 A | | | | | | | |
| 8 | 40 | 00:00:40 | 8:00:4 | | | | | | | |
| 9 | 45 | 00:00:45 | 8:0 | | | | | | | |
| 10 | 50 | 00:00:50 | | | | | | | | |
| 11 | 55 | 00:00:55 | | | | | | | | |
| 12 | 60 | 00:01:00 | | | | | | | | |
| 13 | 65 | 00:01:0 | | | | | | | | |
| 14 | 70 | 00:0 | | | | | | | | |

FORMULA = COUNTIF(range, criteria) = COUNTIF(E9:E7208, "VE u-turn")

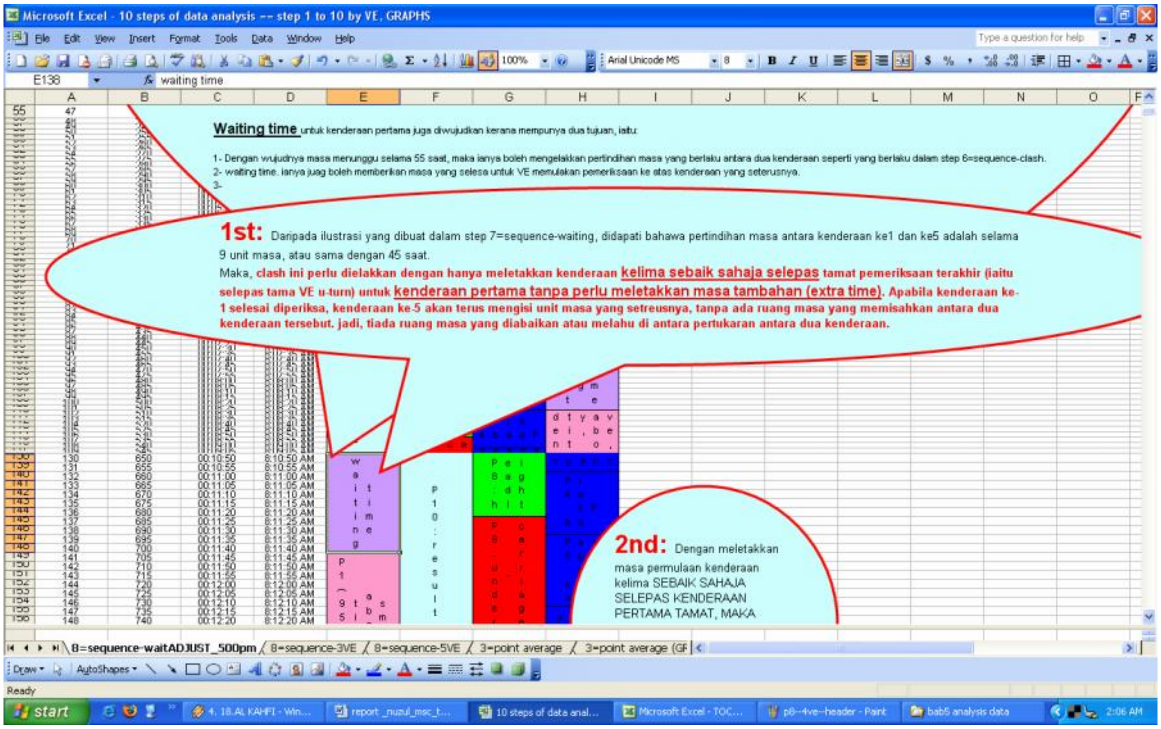
Formula di atas memberi maksud seperti berikut:
E9 bermaksud : masa permulaan pada setiap hari sesi pemeriksaan bermula, iaitu pada jam 8.00.00am.
E7208 bermaksud : masa tamat pada setiap hari sesi pemeriksaan beroperasi, iaitu pada jam 6.00.00pm.

**Figure 4.28**: Changing the Header

Therefore, the fifth vehicle is given a period of waiting time as 11 time units as well starting from 121st time unit to 131st time unit. It means the fifth vehicle has to wait the first vehicle completing its inspection entirely before initiating of its inspection can be done. This adjustment is illustrated in **Figure 4.29**.

As a result, we have determined to assigni ng the fifth vehicle time slot subsequent to the first vehicle's completing time in VE1 (first vehicle examiner) group.  Then, we just continue the sixth vehicle being inspected by the VE2 (second vehicle examiner) after completed inspecting second vehicle. This situation is depicted in **Figure 4.30**.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com


Microsoft Excel - 10 steps of data analysis -- step 1 to 10 by VE, GRAPHS
E138
waiting time
8=sequence-waitADJUST_500pm
8=sequence-3VE
8=sequence-5VE
3=point average
3=point average (GF
Ready
start

**Waiting time** untuk kenderaan pertama juga diwujudkan kerana mempunya dua tujuan, iaitu:

1- Dengan wujudnya masa menunggu selama 55 saat, maka ianya boleh mengelakkan pertindihan masa yang berlaku antara dua kenderaan seperti yang berlaku dalam step 6=sequence-clash.
2- waiting time. ianya juag boleh memberikan masa yang selesa untuk VE memulakan pemeriksaan ke atas kenderaan yang seterusnya.
3-

**1st:** Daripada ilustrasi yang dibuat dalam step 7=sequence-waiting, didapati bahawa pertindihan masa antara kenderaan ke1 dan ke5 adalah selama 9 unit masa, atau sama dengan 45 saat.
Maka, **clash ini perlu dielakkan dengan hanya meletakkan kenderaan kelima sebaik sahaja selepas tamat pemeriksaan terakhir (iaitu selepas tama VE u-turn) untuk kenderaan pertama tanpa perlu meletakkan masa tambahan (extra time). Apabila kenderaan ke-1 selesai diperiksa, kenderaan ke-5 akan terus mengisi unit masa yang setreusnya, tanpa ada ruang masa yang memisahkan antara dua kenderaan tersebut. jadi, tiada ruang masa yang diabaikan atau melahu di antara pertukaran antara dua kenderaan.**

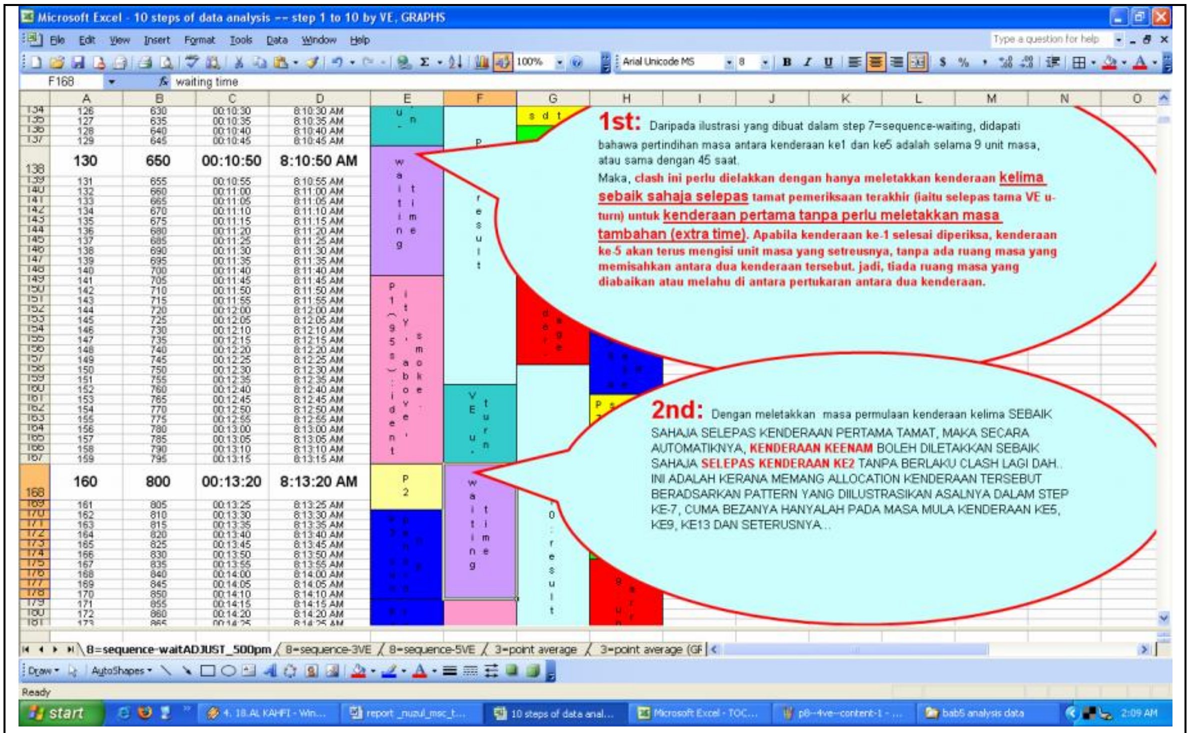
**2nd:** Dengan meletakkan masa permulaan kenderaan kelima SEBAIK SAHAJA SELEPAS KENDERAAN PERTAMA TAMAT, MAKA

**Figure 4.29**: First Part step with 4VE.

Why do we set up the fifth vehicle to be inspected subsequent to first vehicle, instead of forth or sixth vehicle? Based on detail justification in the earlier questions, we noticed that the most optimized solution is to allocate the fifth vehicle subsequent to the first vehicle for the first vehicle examiner (and of course not for the fifth vehicle examiner).

In this way, we will be able to control the length of idle time in the first vehicle examiner's slot, which is to achieve our target in order to use all the resources as maximum stage. Specifically, we need to use the lane as well as vehicle examiner at the most optimum point as it can be.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com


Microsoft Excel - 10 steps of data analysis -- step 1 to 10 by VE, GRAPHS
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial Unicode MS
8
100%
F168
waiting time
8=sequence-waitADJUST_500pm
8=sequence-3VE
8=sequence-5VE
3=point average
3=point average (GF
Draw
AutoShapes
Ready
start

| | A | B | C | D |
|---|---|---|---|---|
| 134 | 126 | 630 | 00:10:30 | 8:10:30 AM |
| 135 | 127 | 635 | 00:10:35 | 8:10:35 AM |
| 136 | 128 | 640 | 00:10:40 | 8:10:40 AM |
| 137 | 129 | 645 | 00:10:45 | 8:10:45 AM |
| 138 | **130** | **650** | **00:10:50** | **8:10:50 AM** |
| 139 | 131 | 655 | 00:10:55 | 8:10:55 AM |
| 140 | 132 | 660 | 00:11:00 | 8:11:00 AM |
| 141 | 133 | 665 | 00:11:05 | 8:11:05 AM |
| 142 | 134 | 670 | 00:11:10 | 8:11:10 AM |
| 143 | 135 | 675 | 00:11:15 | 8:11:15 AM |
| 144 | 136 | 680 | 00:11:20 | 8:11:20 AM |
| 145 | 137 | 685 | 00:11:25 | 8:11:25 AM |
| 146 | 138 | 690 | 00:11:30 | 8:11:30 AM |
| 147 | 139 | 695 | 00:11:35 | 8:11:35 AM |
| 148 | 140 | 700 | 00:11:40 | 8:11:40 AM |
| 149 | 141 | 705 | 00:11:45 | 8:11:45 AM |
| 150 | 142 | 710 | 00:11:50 | 8:11:50 AM |
| 151 | 143 | 715 | 00:11:55 | 8:11:55 AM |
| 152 | 144 | 720 | 00:12:00 | 8:12:00 AM |
| 153 | 145 | 725 | 00:12:05 | 8:12:05 AM |
| 154 | 146 | 730 | 00:12:10 | 8:12:10 AM |
| 155 | 147 | 735 | 00:12:15 | 8:12:15 AM |
| 156 | 148 | 740 | 00:12:20 | 8:12:20 AM |
| 157 | 149 | 745 | 00:12:25 | 8:12:25 AM |
| 158 | 150 | 750 | 00:12:30 | 8:12:30 AM |
| 159 | 151 | 755 | 00:12:35 | 8:12:35 AM |
| 160 | 152 | 760 | 00:12:40 | 8:12:40 AM |
| 161 | 153 | 765 | 00:12:45 | 8:12:45 AM |
| 162 | 154 | 770 | 00:12:50 | 8:12:50 AM |
| 163 | 155 | 775 | 00:12:55 | 8:12:55 AM |
| 164 | 156 | 780 | 00:13:00 | 8:13:00 AM |
| 165 | 157 | 785 | 00:13:05 | 8:13:05 AM |
| 166 | 158 | 790 | 00:13:10 | 8:13:10 AM |
| 167 | 159 | 795 | 00:13:15 | 8:13:15 AM |
| 168 | **160** | **800** | **00:13:20** | **8:13:20 AM** |
| 169 | 161 | 805 | 00:13:25 | 8:13:25 AM |
| 170 | 162 | 810 | 00:13:30 | 8:13:30 AM |
| 171 | 163 | 815 | 00:13:35 | 8:13:35 AM |
| 172 | 164 | 820 | 00:13:40 | 8:13:40 AM |
| 173 | 165 | 825 | 00:13:45 | 8:13:45 AM |
| 174 | 166 | 830 | 00:13:50 | 8:13:50 AM |
| 175 | 167 | 835 | 00:13:55 | 8:13:55 AM |
| 176 | 168 | 840 | 00:14:00 | 8:14:00 AM |
| 177 | 169 | 845 | 00:14:05 | 8:14:05 AM |
| 178 | 170 | 850 | 00:14:10 | 8:14:10 AM |
| 179 | 171 | 855 | 00:14:15 | 8:14:15 AM |
| 180 | 172 | 860 | 00:14:20 | 8:14:20 AM |
| 181 | 173 | 865 | 00:14:25 | 8:14:25 AM |

**1st:** Daripada ilustrasi yang dibuat dalam step 7=sequence-waiting, didapati bahawa pertindihan masa antara kenderaan ke1 dan ke5 adalah selama 9 unit masa, atau sama dengan 45 saat.
Maka, **clash ini perlu dielakkan dengan hanya meletakkan kenderaan kelima sebaik sahaja selepas tamat pemeriksaan terakhir (iaitu selepas tama VE u-turn) untuk kenderaan pertama tanpa perlu meletakkan masa tambahan (extra time). Apabila kenderaan ke-1 selesai diperiksa, kenderaan ke-5 akan terus mengisi unit masa yang setreusnya, tanpa ada ruang masa yang memisahkan antara dua kenderaan tersebut. jadi, tiada ruang masa yang diabaikan atau melahu di antara pertukaran antara dua kenderaan.**

**2nd:** Dengan meletakkan  masa permulaan kenderaan kelima SEBAIK SAHAJA SELEPAS KENDERAAN PERTAMA TAMAT, MAKA SECARA AUTOMATIKNYA, **KENDERAAN KEENAM** BOLEH DILETAKKAN SEBAIK SAHAJA **SELEPAS KENDERAAN KE2** TANPA BERLAKU CLASH LAGI DAH.. INI ADALAH KERANA MEMANG ALLOCATION KENDERAAN TERSEBUT BERADSARKAN PATTERN YANG DIILUSTRASIKAN ASALNYA DALAM STEP KE-7, CUMA BEZANYA HANYALAH PADA MASA MULA KENDERAAN KE5, KE9, KE13 DAN SETERUSNYA...

**Figure 4.30**: Second Part step with 4Vehicle Examiner.

## 4.10 Result and Discussion

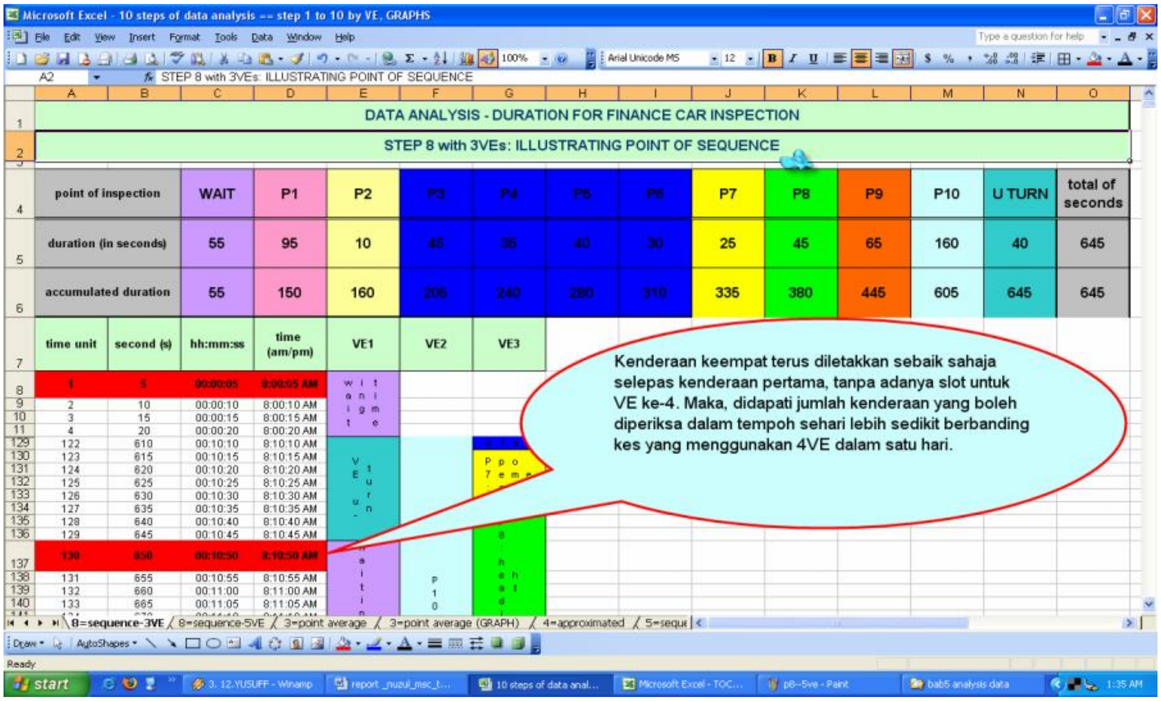
Based from the analysis, the discussion is divided into 2 cases namely analysis with 3 VE and the other is analysis with 5 VE.

***Case 1: Analysis with 3VE***

Why not forth vehicle as a restarting point, instead of fifth vehicle? The answer is here. Based on detail explanation presented earlier, we do not assign the forth vehicle, just because to maximize the volume of vehicles can be inspected per day. If we assigned the forth vehicle subsequent to first vehicle, then it will produce less volume of vehicles' inspected. We decide to use the optimum number of four vehicle examiner instead of using only three vehicle examiner at the same time for the reason that we have to

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

maximize the volume of vehicles' inspected for every day. The captured screen for this explanation is shown in **Figure 4.31.**



DATA ANALYSIS - DURATION FOR FINANCE CAR INSPECTION

STEP 8 with 3VEs: ILLUSTRATING POINT OF SEQUENCE

| point of inspection | WAIT | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | U TURN | total of seconds |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| duration (in seconds) | 55 | 95 | 10 | 45 | 35 | 40 | 30 | 25 | 45 | 65 | 160 | 40 | 645 |
| accumulated duration | 55 | 150 | 160 | 205 | 240 | 280 | 310 | 335 | 380 | 445 | 605 | 645 | 645 |

| time unit | second (s) | hh:mm:ss | time (am/pm) | VE1 | VE2 | VE3 |
|---|---|---|---|---|---|---|
| 1 | 5 | 00:00:05 | 8:00:05 AM | | | |
| 2 | 10 | 00:00:10 | 8:00:10 AM | | | |
| 3 | 15 | 00:00:15 | 8:00:15 AM | | | |
| 4 | 20 | 00:00:20 | 8:00:20 AM | | | |
| 122 | 610 | 00:10:10 | 8:10:10 AM | | | |
| 123 | 615 | 00:10:15 | 8:10:15 AM | | | |
| 124 | 620 | 00:10:20 | 8:10:20 AM | | | |
| 125 | 625 | 00:10:25 | 8:10:25 AM | | | |
| 126 | 630 | 00:10:30 | 8:10:30 AM | | | |
| 127 | 635 | 00:10:35 | 8:10:35 AM | | | |
| 128 | 640 | 00:10:40 | 8:10:40 AM | | | |
| 129 | 645 | 00:10:45 | 8:10:45 AM | | | |
| 130 | 650 | 00:10:50 | 8:10:50 AM | | | |
| 131 | 655 | 00:10:55 | 8:10:55 AM | | | |
| 132 | 660 | 00:11:00 | 8:11:00 AM | | | |
| 133 | 665 | 00:11:05 | 8:11:05 AM | | | |

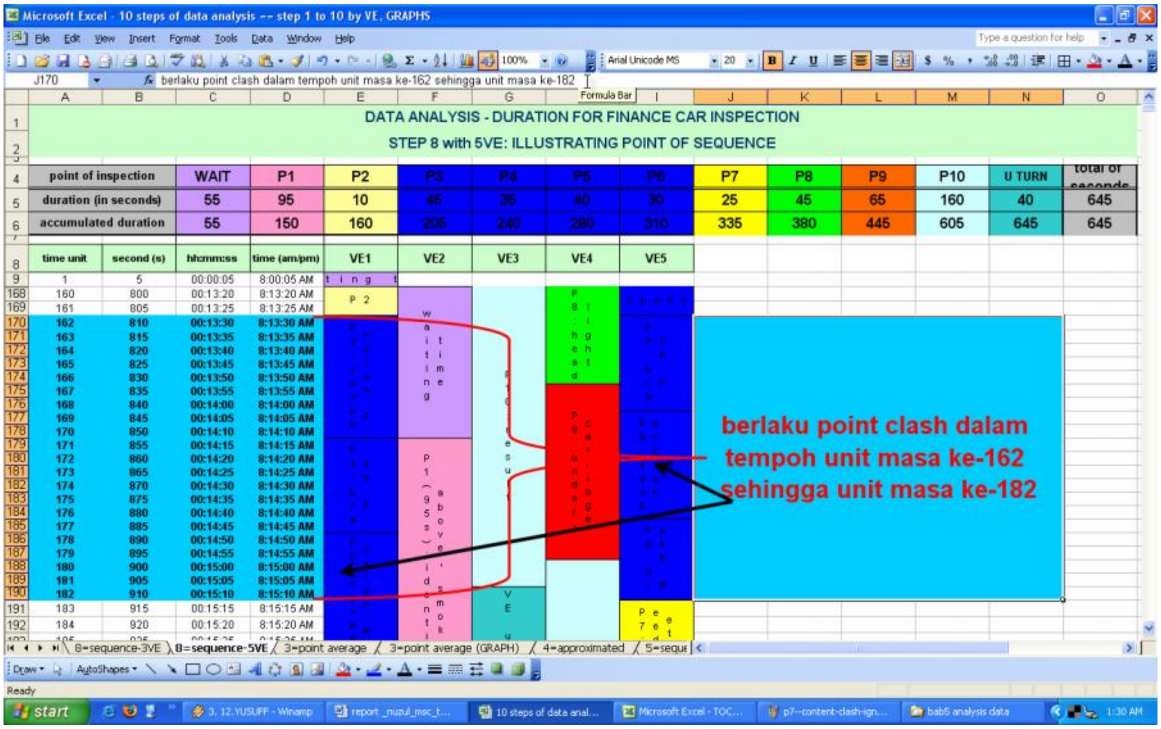
Kenderaan keempat terus diletakkan sebaik sahaja selepas kenderaan pertama, tanpa adanya slot untuk VE ke-4. Maka, didapati jumlah kenderaan yang boleh diperiksa dalam tempoh sehari lebih sedikit berbanding kes yang menggunakan 4VE dalam satu hari.

**Figure 4.31**: The header for the step with waiting time -3VE.

***Case 2: Analysis with 5VE***

Why not sixth vehicle, instead of fifth vehicle assigned subsequent to first vehicle? In this study, our constraint is to maximize the volume of inspection per day. In the same way as the earlier question, we need to obtain the maximum number of vehicles' inspected produced from the illustration we hav e made. Therefore, we do not choose to allocate the sixth vehicle subsequent to the first vehic le because of volume constraint. The description of this result is shown in **Figure 4.32.**

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



berlaku point clash dalam tempoh unit masa ke-162 sehingga unit masa ke-182

DATA ANALYSIS - DURATION FOR FINANCE CAR INSPECTION

STEP 8 with 5VE: ILLUSTRATING POINT OF SEQUENCE

| point of inspection | WAIT | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | U TURN | total of seconds |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| duration (in seconds) | 55 | 95 | 10 | 45 | 35 | 40 | 30 | 25 | 45 | 65 | 160 | 40 | 645 |
| accumulated duration | 55 | 150 | 160 | 205 | 240 | 280 | 310 | 335 | 380 | 445 | 605 | 645 | 645 |

| time unit | second (s) | hh:mm:ss | time (am/pm) | VE1 | VE2 | VE3 | VE4 | VE5 |
|---|---|---|---|---|---|---|---|---|
| 1 | 5 | 00:00:05 | 8:00:05 AM | | | | | |
| 160 | 800 | 00:13:20 | 8:13:20 AM | P 2 | | | | |
| 161 | 805 | 00:13:25 | 8:13:25 AM | | | | | |
| 162 | 810 | 00:13:30 | 8:13:30 AM | | | | | |
| 163 | 815 | 00:13:35 | 8:13:35 AM | | | | | |
| 164 | 820 | 00:13:40 | 8:13:40 AM | | | | | |
| 165 | 825 | 00:13:45 | 8:13:45 AM | | | | | |
| 166 | 830 | 00:13:50 | 8:13:50 AM | | | | | |
| 167 | 835 | 00:13:55 | 8:13:55 AM | | | | | |
| 168 | 840 | 00:14:00 | 8:14:00 AM | | | | | |
| 169 | 845 | 00:14:05 | 8:14:05 AM | | | | | |
| 170 | 850 | 00:14:10 | 8:14:10 AM | | | | | |
| 171 | 855 | 00:14:15 | 8:14:15 AM | | | | | |
| 172 | 860 | 00:14:20 | 8:14:20 AM | | | | | |
| 173 | 865 | 00:14:25 | 8:14:25 AM | | | | | |
| 174 | 870 | 00:14:30 | 8:14:30 AM | | | | | |
| 175 | 875 | 00:14:35 | 8:14:35 AM | | | | | |
| 176 | 880 | 00:14:40 | 8:14:40 AM | | | | | |
| 177 | 885 | 00:14:45 | 8:14:45 AM | | | | | |
| 178 | 890 | 00:14:50 | 8:14:50 AM | | | | | |
| 179 | 895 | 00:14:55 | 8:14:55 AM | | | | | |
| 180 | 900 | 00:15:00 | 8:15:00 AM | | | | | |
| 181 | 905 | 00:15:05 | 8:15:05 AM | | | | | |
| 182 | 910 | 00:15:10 | 8:15:10 AM | | | | | |
| 183 | 915 | 00:15:15 | 8:15:15 AM | | | | | |
| 184 | 920 | 00:15:20 | 8:15:20 AM | | | | | |

**Figure 4.32**: The header for the step with waiting time -5VE.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# CHAPTER 5

# SYSTEM ARCHITECTURE DESIGN

## 5.1 Introduction

Software design is a main part of system engineering. The design elaborated on their behavior process of system with several of diagram. After analyses the user requirement system developer will do the initial design for modeling the new system. It includes the initial plan of design, full design, interface design, database design and so on. One of the important of the design stage is to reduce risk by enabling prediction of system properties. All of this design is covered in this chapter including conceptual design and physical design of ASRS system. The detail process will be shown by use case, activity diagram, sequence diagram and class diagram.

## 5.2 Current Business Process and Data Model

From observation at vehicle inspection center there are have several problem occur in the current process. Most of the problem in time constrained happed in reservation process. Some of the problem in the as-is system only being able to do very limited operations, such as stated below:

- Customers have to take a long time to wait their vehicle to be inspected.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

- To save the appropriate data of registered vehicles. The vehicles will be registered then they come for initial inspection.
- There is no specific time are allocate to finish for every inspection.
- The system can generate the future appointment with manually by increasing 6 month for every inspection date doing by clerk.
- This system are not identify a holiday. So before create the inspection d ate the clerk will refer to calendar to find the available date.

The figure below showed the overall manual process in vehicle inspection center.

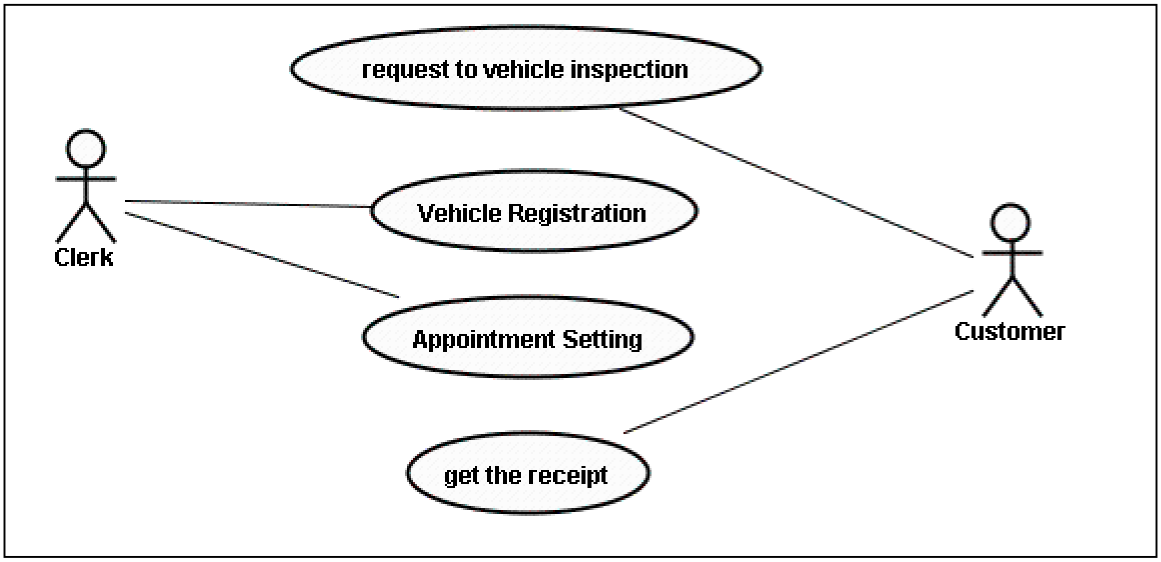


**Figure 5.1:** Use Case Diagram of Manual Process during Inspection Service

## 5.3 User Requirements

From the requirement, we know that why the project was proposed, what the problems are who the stakeholders are, what the problem need and what the proposed solution for the new system that we offer to them.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

A successful project is usually defined as a project that delivers a product which meet user requirement. Similarly, to evaluate the efficiency and success of this process, the result of it need to be comparing against the original requirement.

In order to have better performance on inspection appointment, they required for fully automatic system. As a result, the new system can be more efficient and more effective. In this section describes various functionalities that we propose to be in the new system. The following is a functional requirement requested on new ASRS system.

This user requirement is separated into two sections. First is about Functional Requirement and the second one is Non Functional Requirement. In functional requirement, there are three user involved in this system. There are administrator, clerk and client. The first section is about user requirement behalf on administrator which handle more on the all function in this system. Clerk can initiate a new and how the registration and reservation process are handled. Finally the last section explains how system can provide the services to the customer.

### 5.3.1 Functional Requirement

Functional requirements capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform . In product development, it is useful to distinguish between the baseline functionality necessary for any system to compete in that product domain, and *features* that differentiate the system from competitors' products, and from variants in your company's own product line/family. Features may be additional functionality, or differ from the basic functionality along some quality attribute (such as performance or memory utilization). Use cases have quickly become a widespread practice for capturing functional requirements. This is especially true in the object-oriented community where they originated, but their applicability is not limited to object-oriented systems.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

**Administrator**

1. To have access for all module for ASRS system i.e. the register
2. Have to setting the whole system
3. Can update the data in database
4. Setting new event in calendar

**Clerk and Staff**

1. To register vehicle data
2. To initial a new appointment
3. Have searching function to find available date to set the appointment
4. Automatic searching function to find the vehicle data by enter the vehicle id
5. To view the schedule of appointment
6. To print the document and receipt

**Client and Customer**

1. Be able to view their vehicle information
2. Be able to view their appointment date

### 5.3.2 Non functional requirement

Non-functional requirements are requirements which specify criteria that can be used to judge the operation of a system, rather than specific behaviors. This should be contrasted with functional requirements that specify specific behavior or functions. Non -functional requirements are often called qualities of a system. Other terms for non-functional requirements are constraints, quality attributes, quality goals and quality of service requirements.

1. This system is generated based on online system. So people involve can send or share their information with the centre manager.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

2. Provide the efficient scheduling and has an added capability of processing the reservation with automated system.
3. Number of lane considered is depending on administrator either one or more. The system will set a default value, but then the admin still can make decision on how many lanes will be use for the day. (Admin can change the value lane for a day, week or month)
4. Rule of inspection is that if all three major items passed, the result is pass ed.
5. Number of major item is three.
6. The three major items named chassis number, engine number and original body.
7. Maximum number of vehicle examiner is four.
8. Time stagger is one hour (but still not been used in this analysis).
9. Time slot is assumed as complete time with waiting time at initial bay.
10. Complete time is assumed as processing time with movement period and reserve Time
11. Operation hour in this study is defined from 8 a.m. until 5 p.m.
12. There is no tea break defined in this study. It is based on the real operation at the time of data been collected.
13. There is no lunch hour defined in this study. It is also based on the real operation at the time of data been collected.
14. Each vehicle must have a valid time slot of appointment to be inspected. But, if there was any vehicle that came at the wrong date and time, it was also been inspected.
15. The system will detect the next appointment by plus six month and it can consider the customer's request and based on any constrained and rule.
16. The value of variable can be change for any situation controlled by admin.
17. The system will provide searching function for vehicle information and for available slot time.
18. The searching vehicle information must be flexible. It means the user can define the searching function with any keywords such as by Vehicle Identification Number (VIN), plate number, or class.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

19. The function for searching available slot will be generated using formula, and further detail about this formula will be defined in the next section.

## 5.4 Design Process

Design is a key of the system engineering. In this part, the to -be system will be designed using Unified Modeling Language (UML) as tools to develop a new system.
This case, the figure diagram which includes use case diagram, class diagram, sequence diagram and also activity diagram will be illustrated as shown in the next subtopic.

### 5.4.1 To-Be System Process

***Use case Description***

**Figure 4.2** below represents the whole function of this Appointment Scheduling for Reservation System (ASRS) in vehicle inspection environm ent. This system consist three people invoking the system functionality which are Administrator, clerk and customer.

| | |
|---|---|
| Clerk | Clerk is a main person who indicates a new appointment. So this system the clerk can do function for insert or update form, setting appointment for insert new appointment and update appointment information, view data vehicle and scheduling appointment and print the schedule. |
| Admin | Administrator, people who is responsible for environmental system especially in handling database system and manage the whole system by system setting function. Admin will generate the variable and only admin can change the value of variable then update database. |
| Customer | Customer can view the data only including view for data vehicle and the second is view scheduling appointment. |

**Table 5.1:** Clarification the Actor in Use Case

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

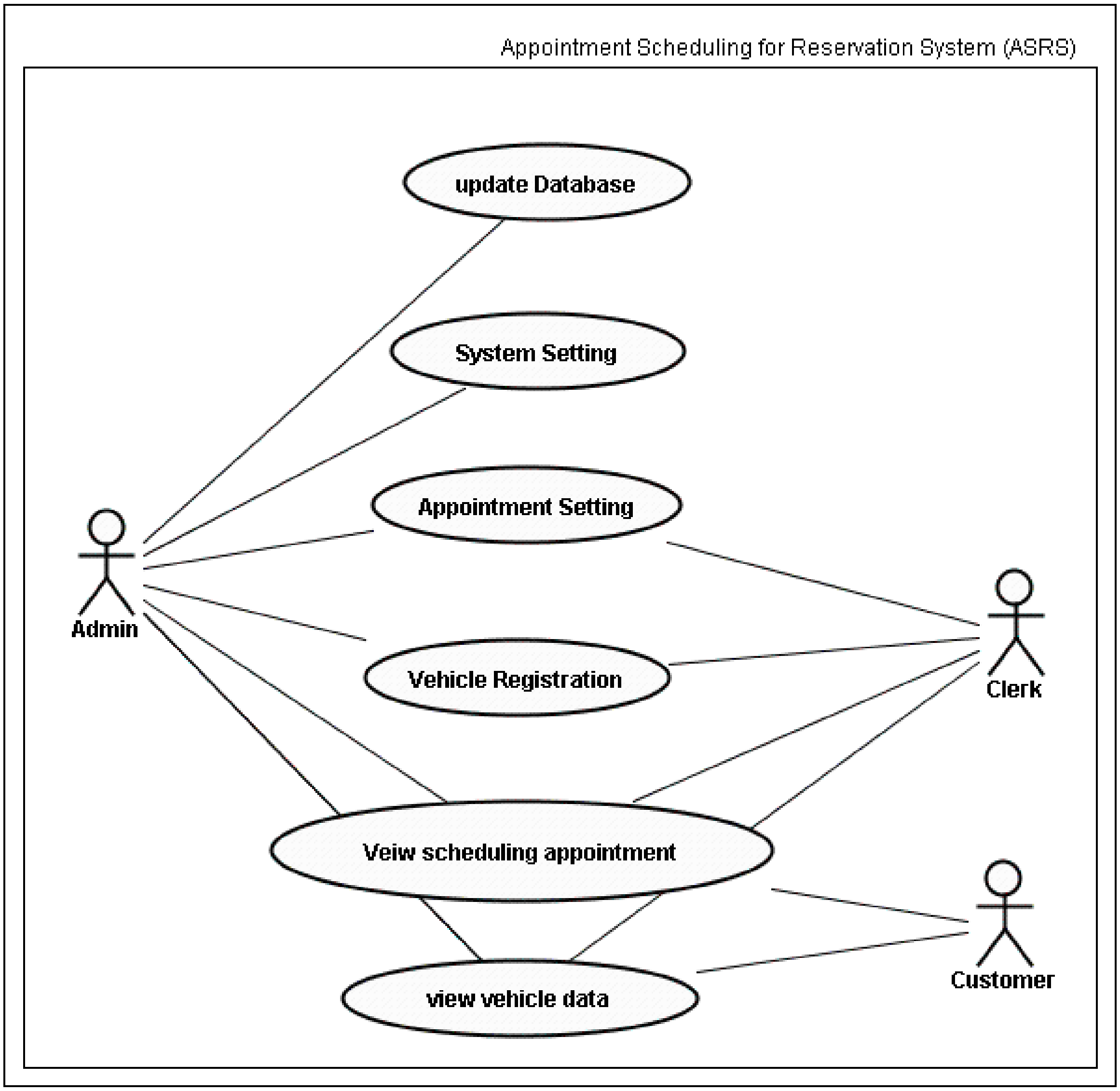


**Figure 5.2:** Use Case in ASRS System

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

| | | |
|---|---|---|
| **Use Case Name :** Update Database | **ID :** 1 | **Importance Level:** High |
| **Primary Actor :** Admin | **Use Case Type :**Details, Essential | |

**Stakeholders and Interest :**
Admin need to update a new data which related during inspection process. The data including the user account, a new registration for vehicle and the others data

**Brief Description :**
This use case describes how the admin can update the data in ASRS system. The processes cover on how to edit, add and delete the data.

**Trigger:** Admin sign up the system first, and achieve the update database fun ction
**Type :** External

**Relationships:**

Association:
Include:
Extend :
Generalization:

**Normal Flow of Events:**

1. Admin input their username and password for login.
2. System verifies the login. If data is valid the update interface will be appear for user in table option.
3. Upon successful login, system will display a list of database table have to be updated (S-1)
4. System verifies the data entry and validates the data and get the confirmation from user before save to database.
5. If user validates, the data will b e saving in database permanently.
6. The user also can view the latest data and also can print as a document report.

**Sub Flows:**

**S1: Update database process**

1. Select table data to update
2. In this point, admin need to choose either to use search data function or not
3. Then admin can choose the edit, delete or add a new data
4. System will be verified the data first before save

**Alternate/Exceptional Flows:**

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

| | | |
|---|---|---|
| **Use Case Name :** System Setting | **ID :**2 | **Importance Level:** High |
| **Primary Actor :** Admin | **Use Case Type :** Details, Essential | |

**Stakeholders and Interest :**
**Admin – Access System Setting**

**Brief Description :**
This use case, admin need to change the system setting. There are two sub functions which are change value for variable and set up new value for appointment schedu le. That is important to set a new value if any modification had to be in a day.

**Trigger:** Admin can access the system setting to manage the whole system
**Type :** External

**Relationships:**
Association :
Include :Setting the system, setting schedule, chan ge variable value
Extend :
Generalization:

**Normal Flow of Events:**

1. The admin will log in into the system by key in the username and password
2. The system will validate whether the username and password valid or not.
3. If the login successful, there are have two selection which are change value the variable or term (S-1) and setting schedule function (S-2)

**Sub Flows:**
**S-1: Change value variable**

1. System views a list of table and a list of variable.
2. Admin can change any variable as example to change the term of variable and the initial value for any variable.

**S-2:Setting schedule system**

1. Admin can change the number of lane, number of VE, operation hour, waiting time, working hour
2. System will verify the input and save into schedule database

**Alternate/Exceptional Flows:**

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

| | | |
|---|---|---|
| **Use Case Name :**view vehicle appointment | **ID :** 3 | **Importance Level:** High |
| **Primary Actor :** Customer | **Use Case Type :**Details, Essential | |
| **Stakeholders and Interest :**<br>Customer – view the information only<br>Clerk – view the vehicle appointment after generate new appointment | | |
| **Brief Description :**<br>This function, customer allow to view their appointment for vehicle inspection | | |
| **Trigger:** User can view the vehicle appointment<br>**Type :**External | | |
| **Relationships:**<br>Association: Clerk<br>Include: view vehicle appointment, print<br>Extend :<br>Generalization: | | |
| **Normal Flow of Events:**<br>1. The customer log in into the system<br>2. System will validate customer login.<br>3. Customer need to key in their vehicle id to check the appointment information<br>4. If valid, the appointment information will be show. | | |
| **Sub Flows:** | | |
| **Alternate/Exceptional Flows:** | | |

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

| | | |
|---|---|---|
| **Use Case Name :** view vehicle information | **ID :** 4 | **Importance Level:** High |
| **Primary Actor :** Customer | **Use Case Type :** Details, Essential | |
| **Stakeholders and Interest :**<br>Customer – view the information only<br>Clerk – view the vehicle information after register the vehicle for inspection | | |
| **Brief Description :**<br>Customer can view their vehicle information | | |
| **Trigger:** All user can view this information<br>**Type :** External | | |
| **Relationships:**<br>Association: Clerk<br>Include: view vehicle information, print<br>Extend :<br>Generalization: | | |
| **Normal Flow of Events:**<br>1. The customer log in into the system<br>2. System will validate customer login.<br>3. Customer need to key in their vehicle id to check the appointment information<br>4. If valid, the appointment information will be show. | | |
| **Sub Flows:** | | |
| **Alternate/Exceptional Flows:** | | |

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

<table>
<tr><td><b>Use Case Name :</b> Vehicle Registration</td><td><b>ID :</b> 5</td><td><b>Importance Level:</b> High</td></tr>
<tr><td><b>Primary Actor :</b>Clerk</td><td colspan="2"><b>Use Case Type :</b>Details, Essential</td></tr>
<tr><td colspan="3"><b>Stakeholders and Interest :</b><br>Clerk – doing vehicle registration behalf on customer</td></tr>
<tr><td colspan="3"><b>Brief Description :</b><br>In initial inspection, customer has to inform for vehicle registration.</td></tr>
<tr><td colspan="3"><b>Trigger:</b> Clerk will insert the registration form.<br><b>Type :</b>External</td></tr>
<tr><td colspan="3"><b>Relationships:</b><br>Association : Customer<br>Include : vehicle registration, view vehicle data</td></tr>
<tr><td colspan="3"><b>Normal Flow of Events:</b><br>1. Login the system by key in username and password.<br>2. Registration form will be show<br>3. Insert the data in form vehicle. (E.g. : vehicle no, ye ar, type)<br>4. System verifies and validates the data. If have any error in some of data, user must to repeat the insert process again until finish.<br>5. System informs to submit and save the</td></tr>
<tr><td colspan="3"><b>Sub Flows:</b><br>The vehicle information will be updated doing by clerk if any new information.<br>The details for this function are given as below:<br>• The user is logged into the system with input their username and password via login form.<br>• System verifies the user login. If data is valid registration form will be show to the user.<br>• User click “<b>Update</b>” button to change any information.<br>• User changes the data in form vehicle.<br>• System verifies and validates the data. If have any error in some of data, user must to repeat the insert process again until finish.<br>• System informs the user to submit and save the data to get confirmation before saving to database.<br>• The system ends when the user takes the printed the document or user can view the data vehicle.</td></tr>
<tr><td colspan="3"><b>Alternate/Exceptional Flows:</b></td></tr>
</table>

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

| | | |
|---|---|---|
| **Use Case Name :** Setting Appointment | **ID :** 6 | **Importance Level:** High |
| **Primary Actor :**Clerk | **Use Case Type :**Details, Essential | |
| **Stakeholders and Interest :**<br>Clerk – Setting new appointment to vehicle inspection | | |
| **Brief Description :**<br>The purpose of this function is to set new appointment to vehicle inspection and also update the appointment. | | |
| **Trigger:** Clerk will initiate a new appointment and also can update the appointment<br>**Type :**External | | |
| **Relationships:**<br>Association : Vehicle data<br>Include: Setting appointment, update appointment, view appointment schedule. | | |
| **Normal Flow of Events:**<br>1. As usual, clerk must to input their username and password in log in function.<br>2. System verifies the user login. If data is valid clerk can choose either to set new appointment (S-1) or update appointment (S-2) | | |
| **Sub Flows:**<br>**S-1 : Set New Appointment**<br>1. First, user will enter the vehicle number to determine data of vehicle.<br>2. System checks vehicle record to see if the vehicle has previously registered. If not registered, go to registration process in add new vehicle.<br>3. If the data is available, insert the record of appointment such as time and date. System validate the data and check either the data entry are clash or not with the holiday. If conflict, user must to key in another day.<br>4. This system also provides the next appointment automatically.<br>5. System gets confirmation from user first before save a new data into database.<br>6. Then fits into the appointment's schedule.<br><br>**S-2 : Update Appointment (Request by customer)**<br>1. Clerk will enter the vehicle number to determine data of vehicle get appointment information.<br>2. Insert a new date of appointment. System validate the data and check either the data entry are clash or. If conflict, user must to key in another day.<br>3. System gets confirmation from user first before save a new data into database.<br>4. Then fits into the appointment's schedule. | | |
| **Alternate/Exceptional Flows:** | | |

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

***Activity Diagram***

This activity diagram focused on how the customers request the appointment date. When customers come to inspection center for initial inspection, they must to register their vehicle as a record data. The vehicle data will be used in others function in particular to set up the appointment schedule. All the registration and reservation process has been done by clerk behalf on the customer. The inspection will be conducted after  all the process finished. At the end of process, the customers get the receipt of the appointment with the inspection number.

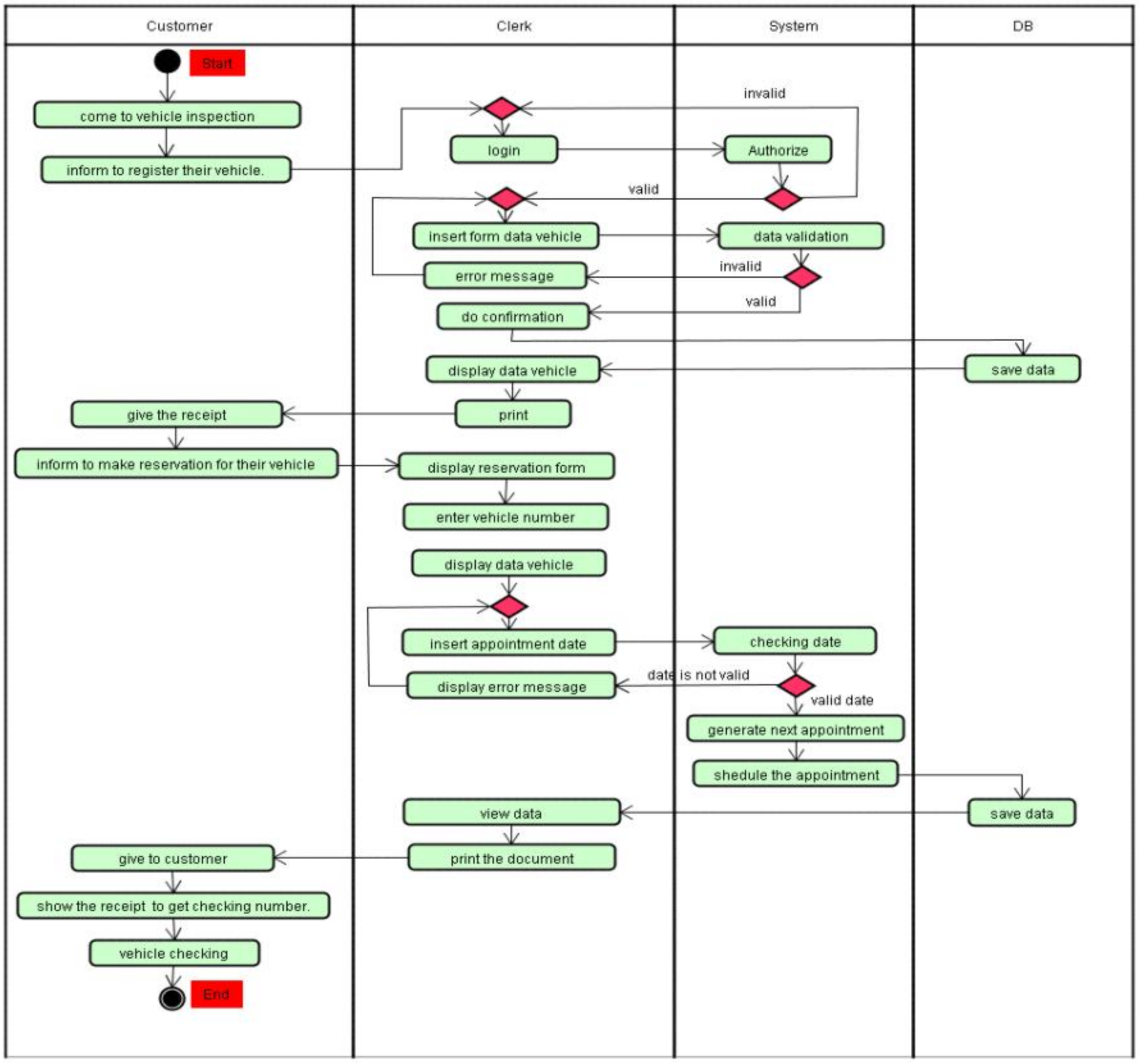


**Figure 5.3:** Activity Diagram for Vehicle Registration Process

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## ***Sequence  Diagram***

The figures below shows the vehicle registrati on process in propose system. The processes started with login function, followed by insert register form, then system validation to check input error and finally save into database and also can view the whole information. There are three associates class  during this process which is Admin class, New Vehicle Class and Vehicle Class.

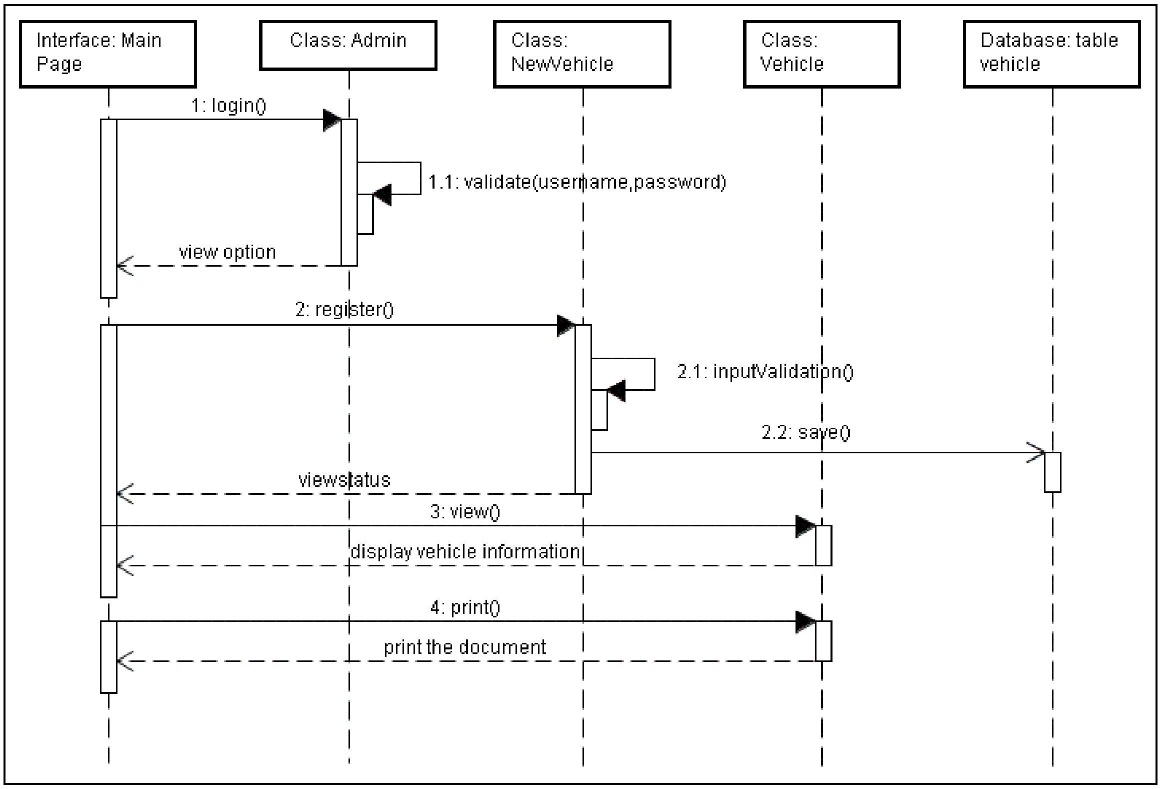


**Figure 5.4:** Sequence Diagram for Vehicle Registration Process

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

This process will be described how clerk setting a new appointment to vehicle inspection behalf on customer. In the end of this function the schedule of the appointment will be generated. This area, the system will generate the propose date automatically based on the available free date from calendar event. The time for inspection will be automatic generated by searching the free slot in reservation schedule.

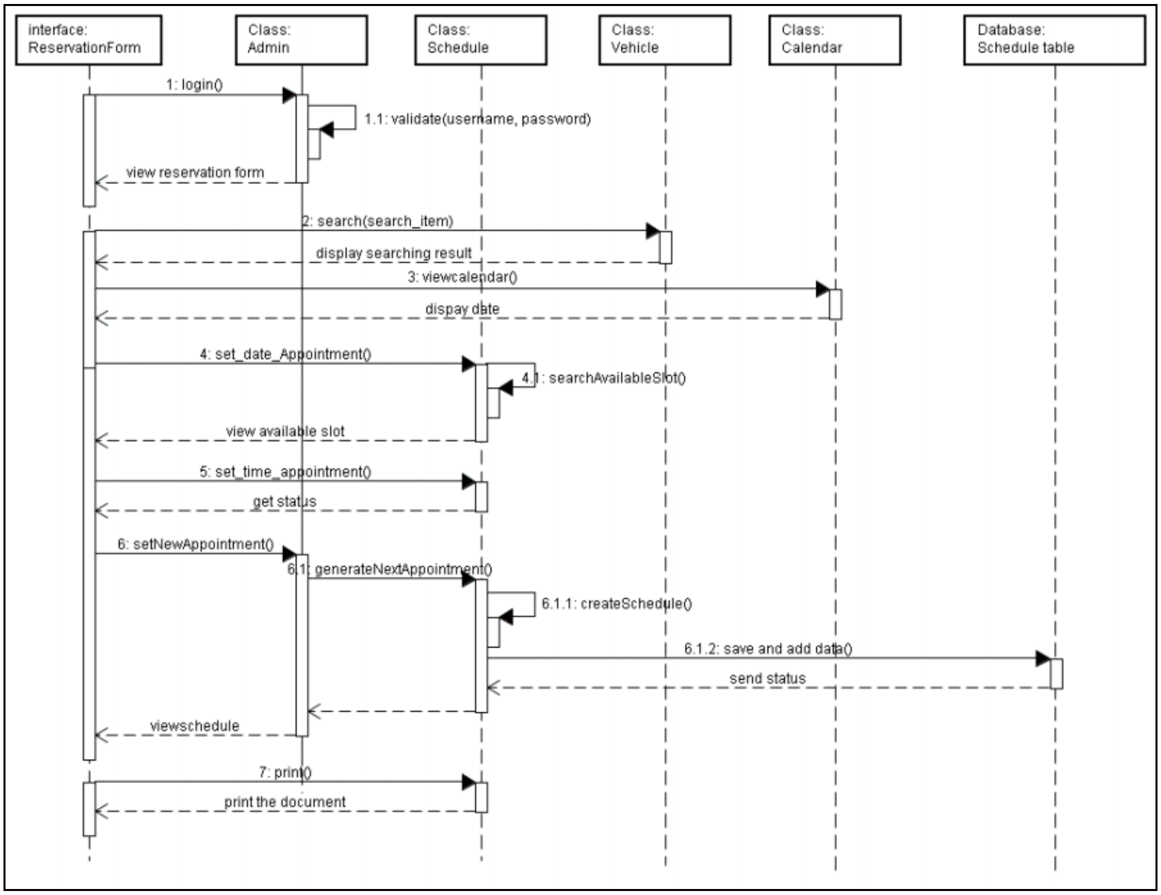


**Figure 5.5:** Sequence Diagram for Reservation of Appointment Process

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

The function admin can insert new holiday or event in calendar activity to generate new calendar. The purpose of this function is to  find available date before setting any reservation base on the new calendar.

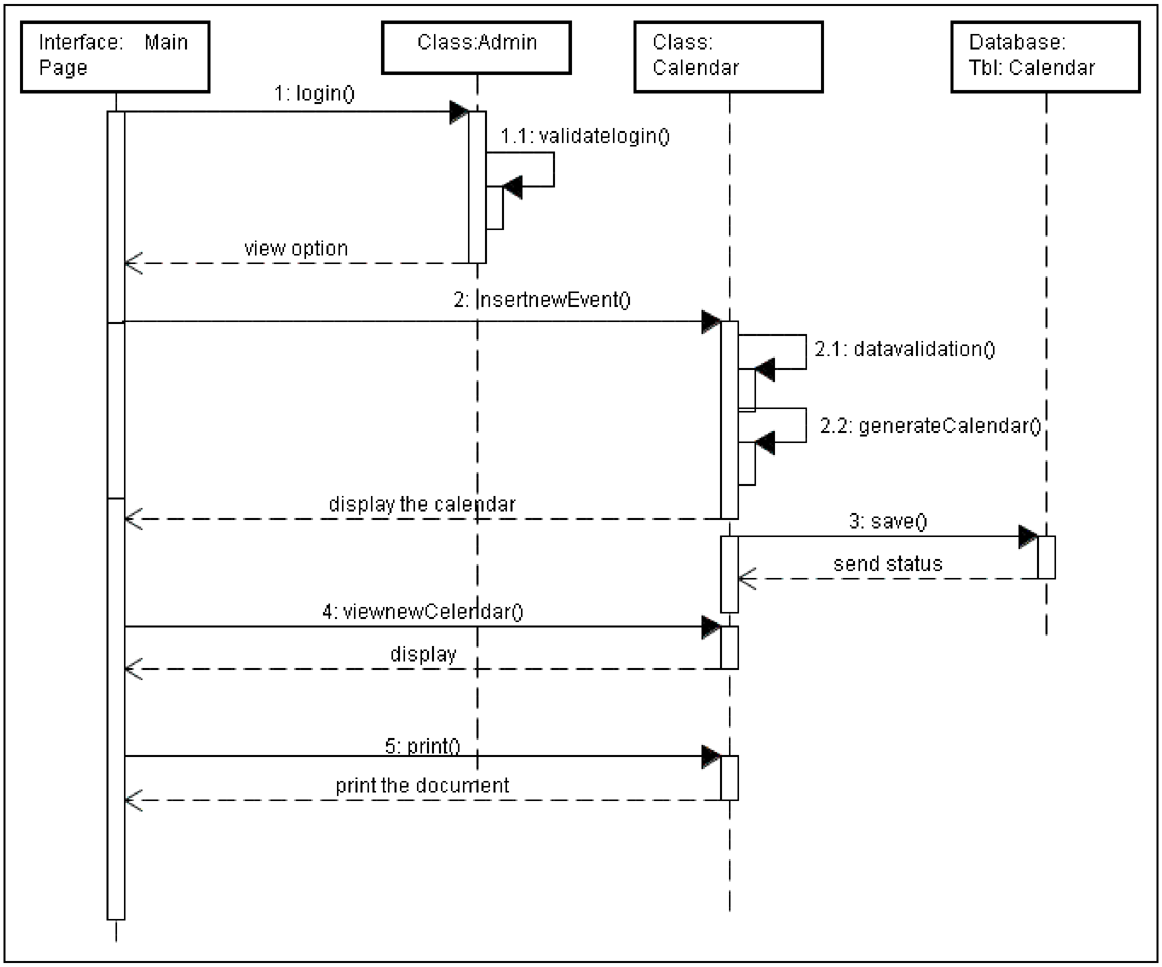


**Figure 5.6:** Sequence Diagram for Insert New Event Process

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

This sequence diagram show the function for update data in database such as change variable, change initial valu e of data, delete or add a new data. This function admin can select the table of database to update the data.

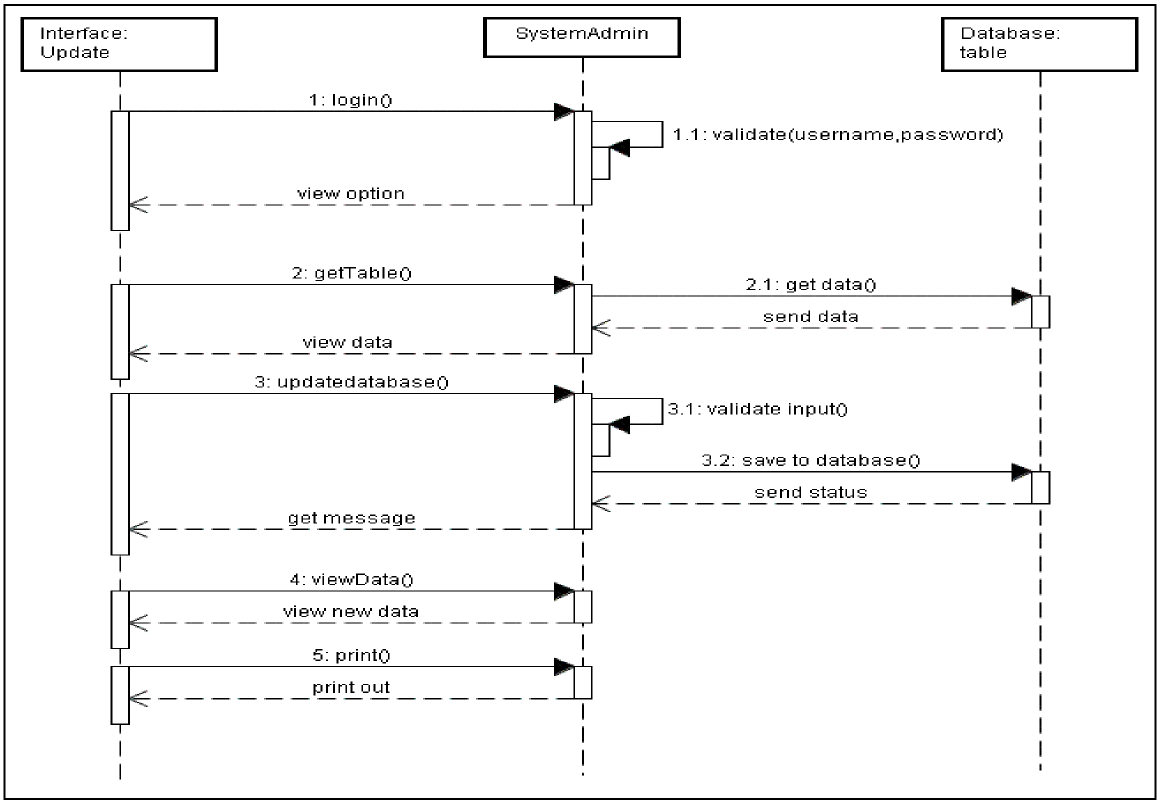


**Figure 5.7:** Sequence Diagram of Update Table in Database

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

### 5.4.2 Conceptual Design

The figure below shows the system architecture for conceptual design of ASRS where all three main user interrelated between them and the ASRS system.

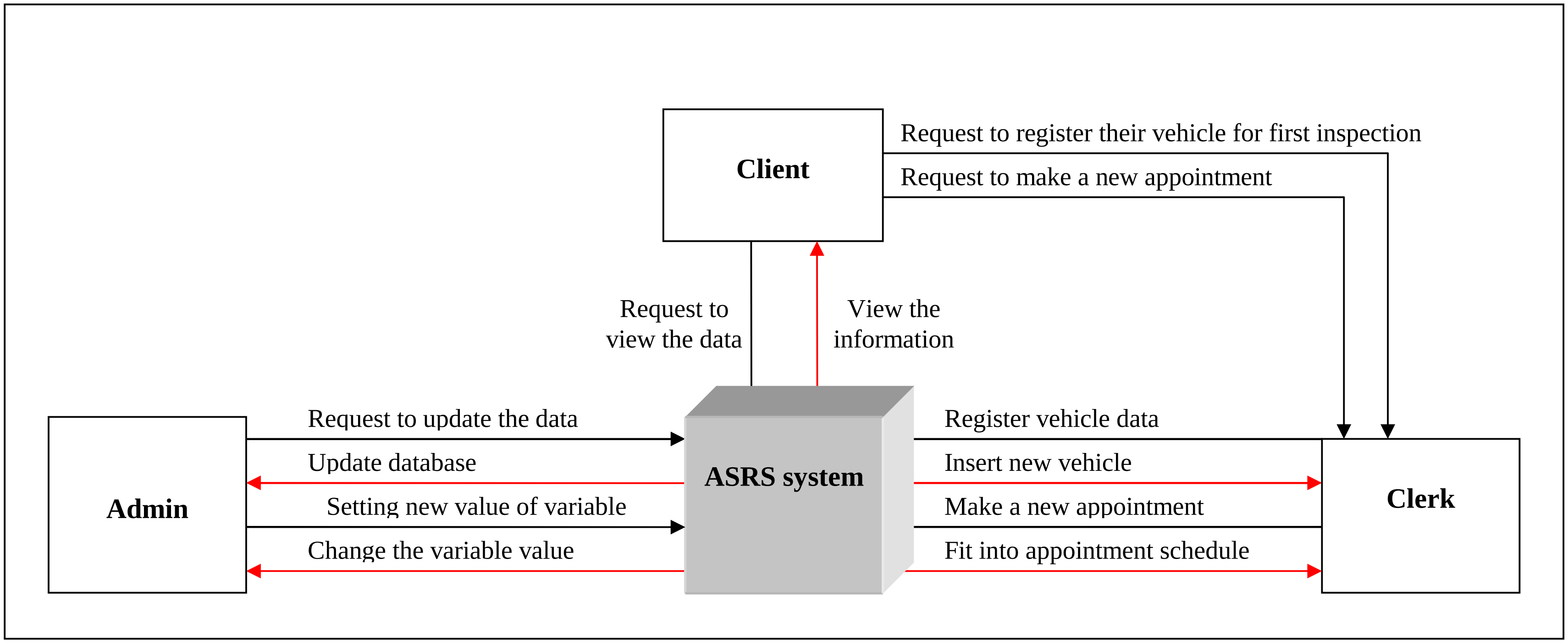

Client
Request to register their vehicle for first inspection
Request to make a new appointment
Request to view the data
View the information
Admin
Request to update the data
Update database
Setting new value of variable
Change the variable value
ASRS system
Register vehicle data
Insert new vehicle
Make a new appointment
Fit into appointment schedule
Clerk

**Figure 5.8:** Conceptual Design for ASRS System

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

## 5.5 Physical Design

Physical design includes database design, program (structure ) chart, interface chart, detailed modules / features and system architecture . Figure 4.9 below shows the whole system architecture for physical design of ASRS system

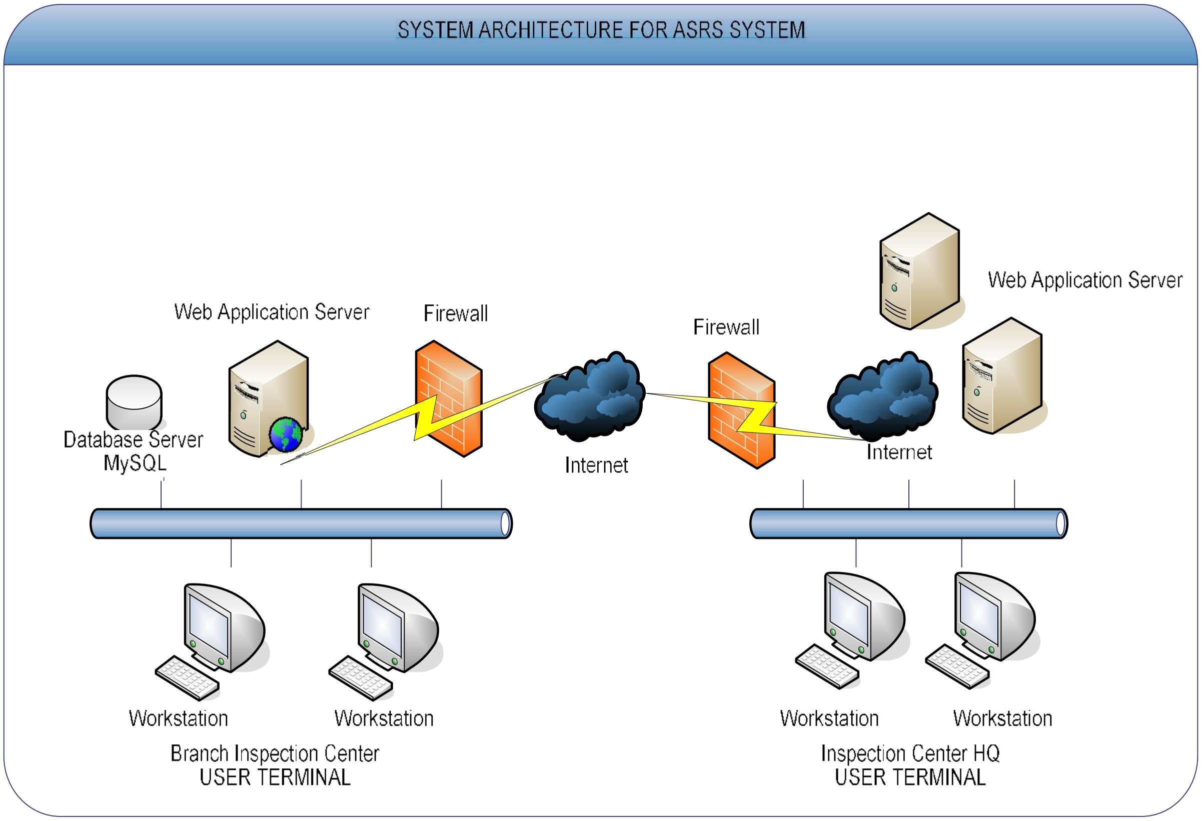


**Figure 5.9:** Physical Design of ASRS system

### 5.5.1 Database Design

Figure 4.10 below shows the class diagram of ARSR system which includes the variable and module for each class. There have eight classes and have some of them had two inherent classes.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

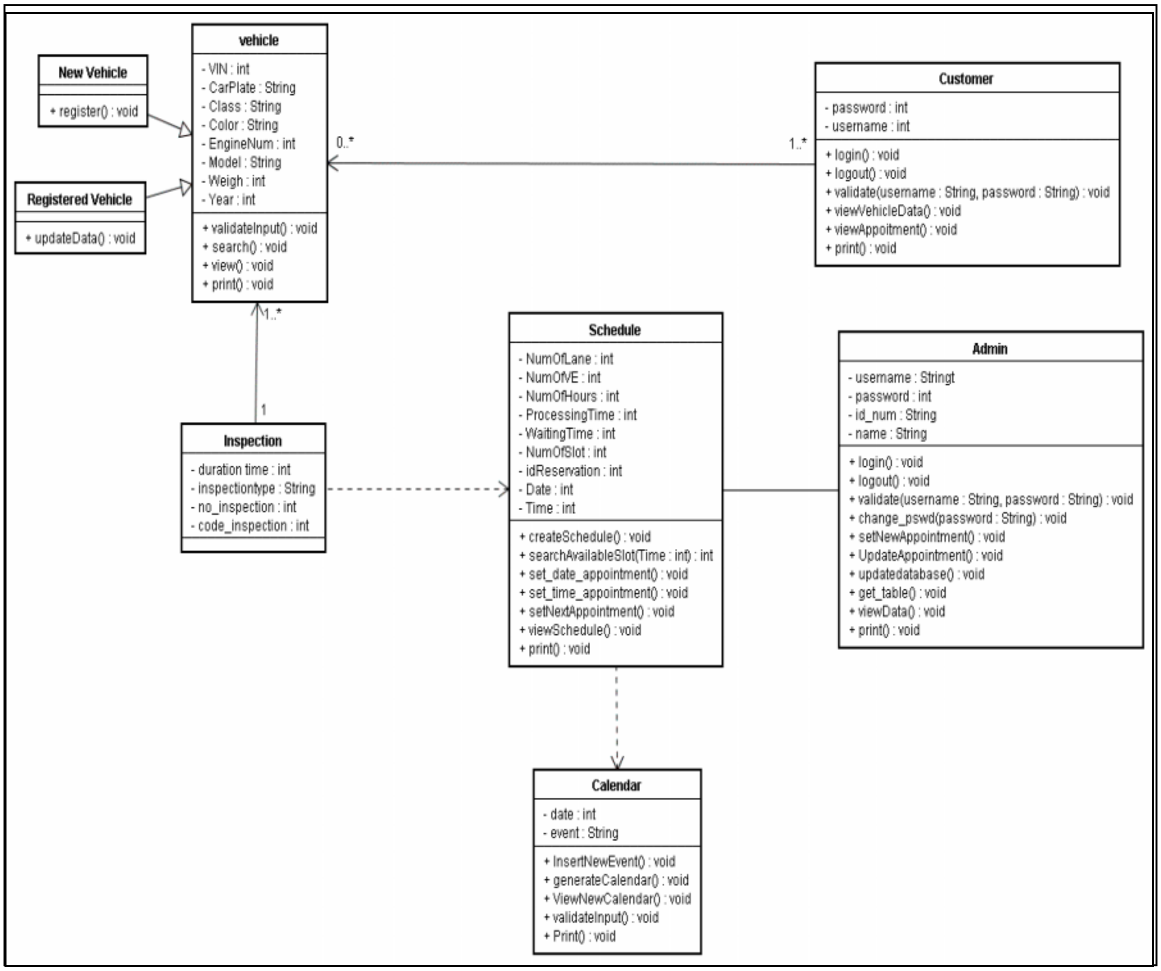


**Figure 5.10:** Class Diagram of ASRS System

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

### 5.5.2 Program Structure

The following Figure 4.11 illustrate the program structure of ASRS system

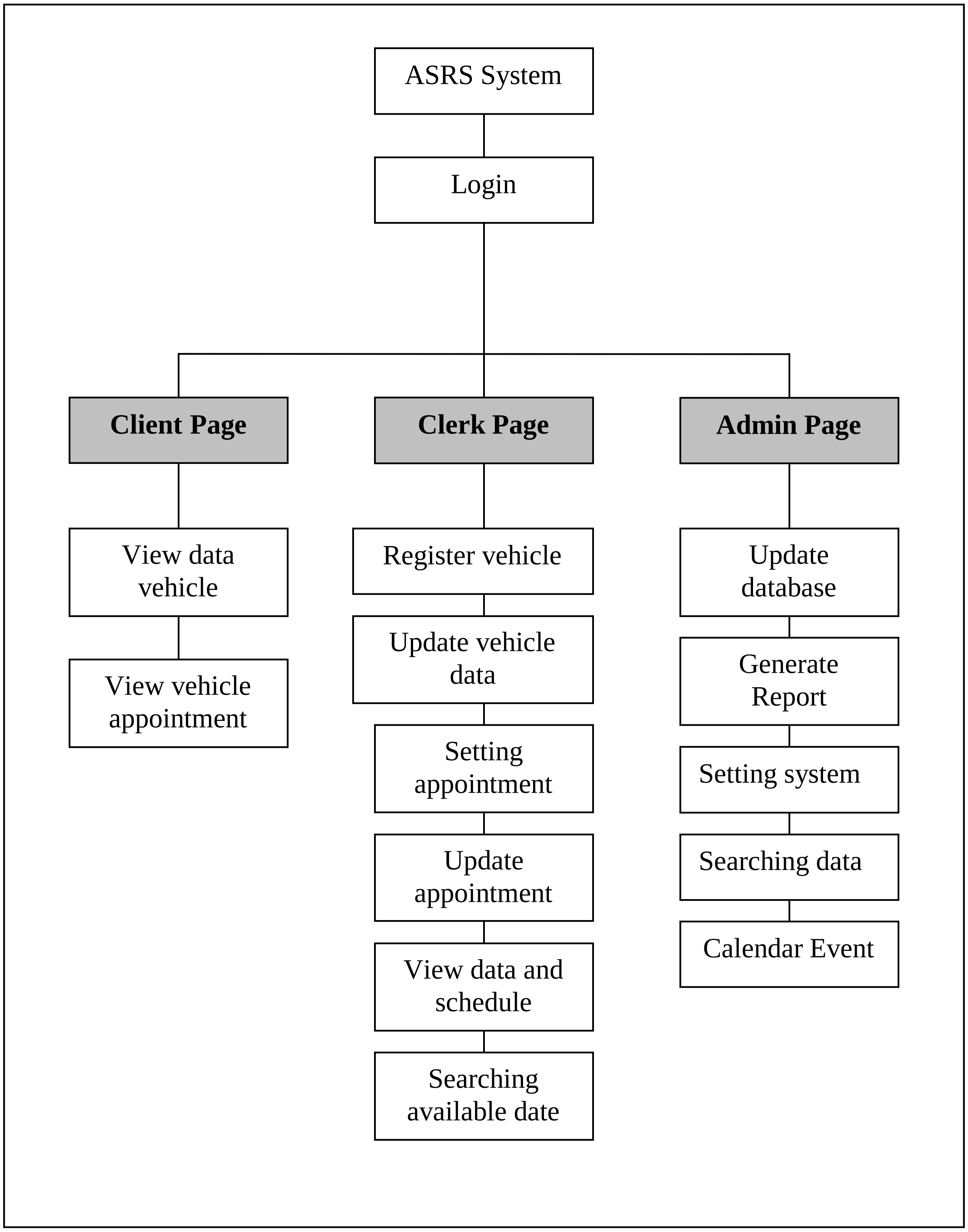


**Figure 5.11:** ASRS Program Structure

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

### 5.5.3 Interface Chart

This section describes the overall interface chart in ASRS system. There are three main categories which are client page, clerk page and admin page. Table below explain the ASRS features which are assessable to user. The symbol signifies that a specific feature is available to correspondence users.

| **Features** | **Admin** | **Clerk** | **Client** |
|---|---|---|---|
| User registration | √ | √ | |
| Login page | √ | √ | |
| **Main Page** | | | |
| About Us | √ | √ | √ |
| Inspection Branch | √ | √ | √ |
| **Administrator main** | | | |
| System Setting | √ | | |
| Update database | √ | | |
| Setting calendar event | √ | | |
| Update user account | √ | | |
| Searching data | √ | | √ |
| View vehicle data | √ | | √ |
| **Clerk main page** | | | |
| View profile | √ | √ | |
| Vehicle Registration | √ | √ | |
| Update Appointment | √ | √ | |
| Setting new appointment | √ | √ | |
| View appointment information | | | √ |
| Searching available date | | √ | |

**Table 5.2:** ASRS Interface Chart

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# CHAPTER 6

# HARDWARE AND SOFTWARE REQUIREMENT

## 6.1 Introduction

This chapter is providing the guide as information resources for hardware and software requirement that required for implement ASRS system in the real environment. This chapter describes the specification of hardware and software requirement to install and run ASRS system. For more detail about how to configure and set up the software, refer to the software installation guides at section 6.4.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 6.2 Hardware requirement

The list for hardware requirement is depicted in **Table 5.1**.

| Hardware | Specification |
|---|---|
| Windows | Window 98,ME,2000,XP, Vista |
| Processor | Intel Pentium 4 and above |
| Memory | 256 MB and above |
| Hard Disk | 10 GB and above |
| Networking | 10/100 Internet |
| Printer | Network printer |

**Table 6.1:** A List of Hardware requirement

## 6.3 Software requirement

**Table 5.2** below shows the list of software requirement that are needed in this project.

| Software Requirement | Purpose |
|---|---|
| 1. **Eclipse Classic 3.3.1.1 Windows:**<br>**eclipse-SDK-3.3.1.1-win32.zip**<br>http://www.eclipse.org/downloads/ | As a development platform to write Java programming |
| 2. **Java Development Kit 6 (JDK) update 3:**<br>**jdk-6u3-windows-1586-p.exe**<br>http://java.sun.com/javase/downloads/index.jsp | To install Java Environment |
| 3. **JBossIDE:**<br>**JBossIDE-2.0.0.Beta2-ALL.zip**<br>http://labs.jboss.com/jbosside/download/index.html | |
| 4. **JBoss Application Server version 4.2.2.GA:**<br>**jboss-4.2.2.GA.zip**<br>http://labs.jboss.com/jbossas/downloads/ | For Server Application |
| 5. **Database MySQL 5.0.45:**<br>**mysql-5.0.45-win32.zip**<br>http://dev.mysql.com/downloads/mysql/5.0.html#win32 | As Server Database |
| 6. Jboss Control Panel 1.0<br>(given by trainer) | For running code |

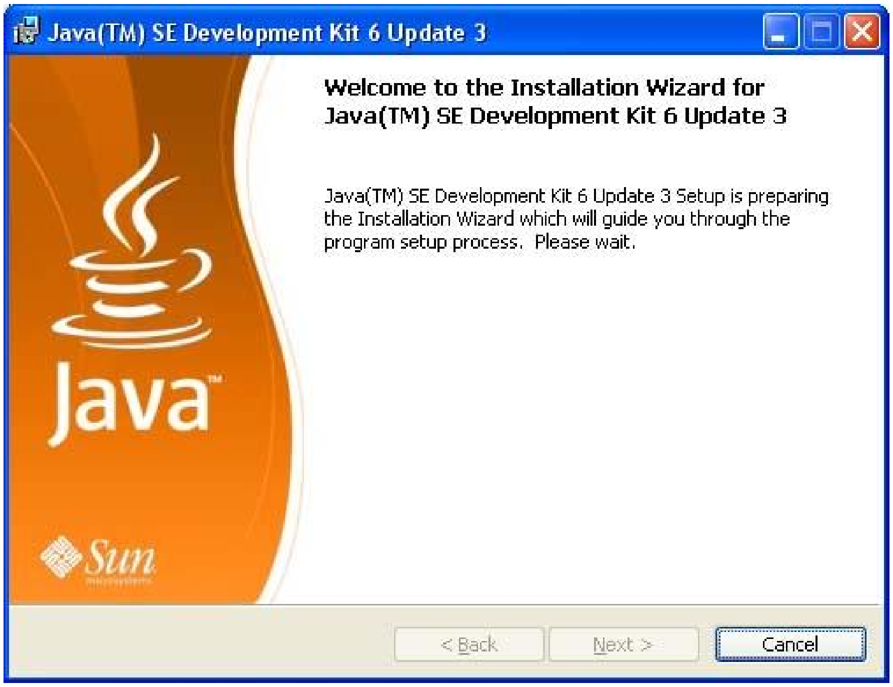
**Table 6.2:** A List of Software requirement

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

## 6.4 Software Installation

### 6.4.1 Installation of Java Development Kit 6 (JDK) update 3

- To install JDK, you need to run jdk-6u3-windows-i586-p.exe. You can see the first screen like Figure below. Then you click **“Next”** button.



Java(TM) SE Development Kit 6 Update 3

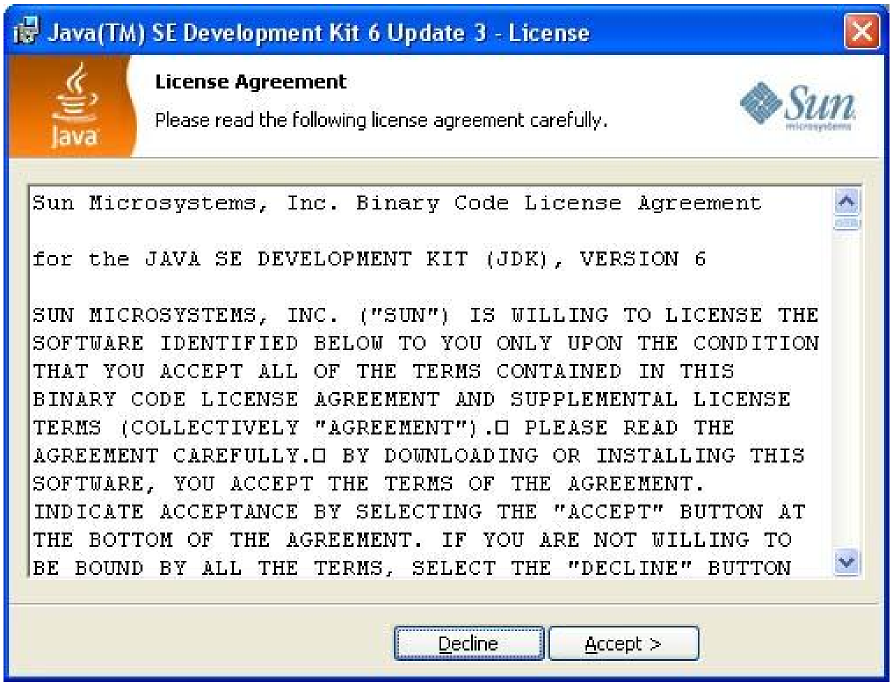
Welcome to the Installation Wizard for Java(TM) SE Development Kit 6 Update 3

Java(TM) SE Development Kit 6 Update 3 Setup is preparing the Installation Wizard which will guide you through the program setup process. Please wait.

**Figure 6.1:** Welcome Screen

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

- The next screen will be License Agreement. You are advised to read the term of the license carefully before proceeding with the installation. If you are decline the license term, the installation cannot proceed. So click “ **Accept**” button to continue installation.



Java(TM) SE Development Kit 6 Update 3 - License

License Agreement

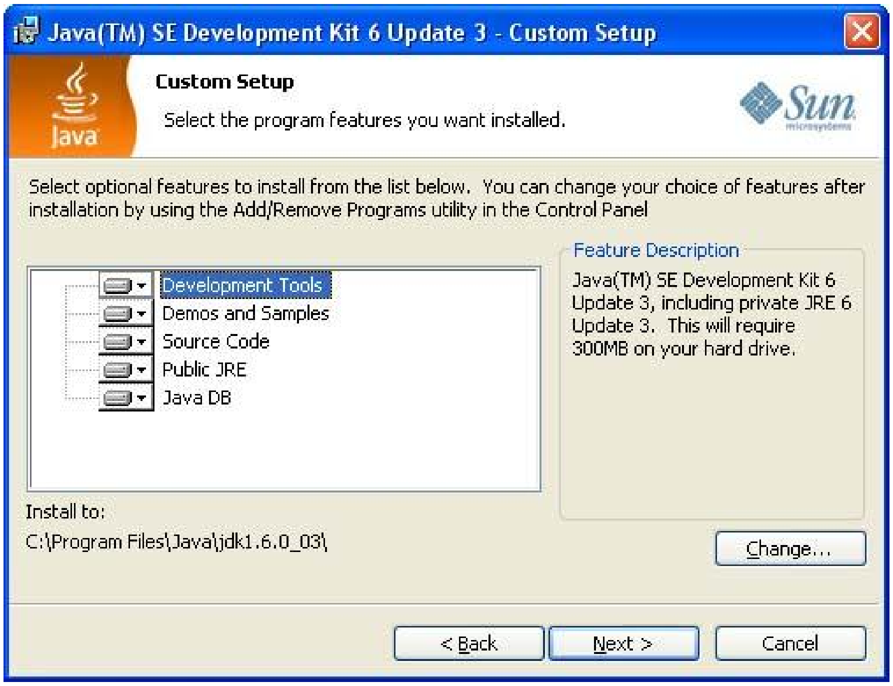
Please read the following license agreement carefully.

```
Sun Microsystems, Inc. Binary Code License Agreement

for the JAVA SE DEVELOPMENT KIT (JDK), VERSION 6

SUN MICROSYSTEMS, INC. ("SUN") IS WILLING TO LICENSE THE
SOFTWARE IDENTIFIED BELOW TO YOU ONLY UPON THE CONDITION
THAT YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS
BINARY CODE LICENSE AGREEMENT AND SUPPLEMENTAL LICENSE
TERMS (COLLECTIVELY "AGREEMENT").□ PLEASE READ THE
AGREEMENT CAREFULLY.□ BY DOWNLOADING OR INSTALLING THIS
SOFTWARE, YOU ACCEPT THE TERMS OF THE AGREEMENT.
INDICATE ACCEPTANCE BY SELECTING THE "ACCEPT" BUTTON AT
THE BOTTOM OF THE AGREEMENT. IF YOU ARE NOT WILLING TO
BE BOUND BY ALL THE TERMS, SELECT THE "DECLINE" BUTTON
```

**Figure 6.2:** License Agreement Screen

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

- At this point you can select any program features for installing Java(TM) SE Development Kit 6 Update 3. But this project needs you to choose “ **Development Tools**”. At bottom area you allow to change the destination fo lder, if you have specific reason for doing so. Just click on **“Change”** button and you can select any folder that you want. Once you have correct path, click **“Next”** to proceed.



Java(TM) SE Development Kit 6 Update 3 - Custom Setup

Custom Setup

Select the program features you want installed.

Select optional features to install from the list below. You can change your choice of features after installation by using the Add/Remove Programs utility in the Control Panel

- Development Tools
- Demos and Samples
- Source Code
- Public JRE
- Java DB

Feature Description

Java(TM) SE Development Kit 6 Update 3, including private JRE 6 Update 3. This will require 300MB on your hard drive.

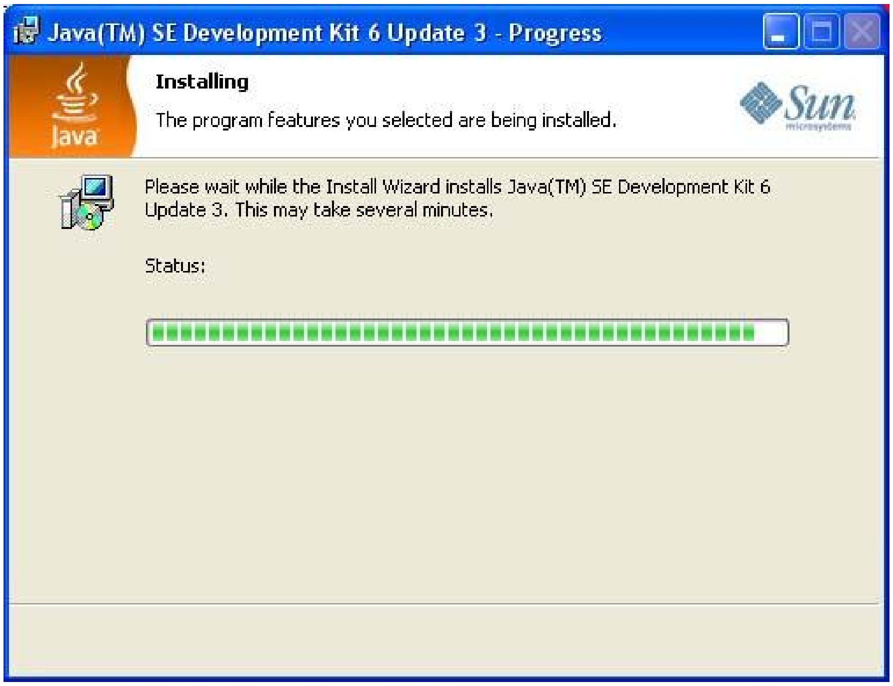
Install to:

C:\Program Files\Java\jdk1.6.0_03\

**Figure 6.3:** Custom Setup Screen

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

- Now the program features are being installed . Please wait until the installation finish.



Java(TM) SE Development Kit 6 Update 3 - Progress

Installing

The program features you selected are being installed.

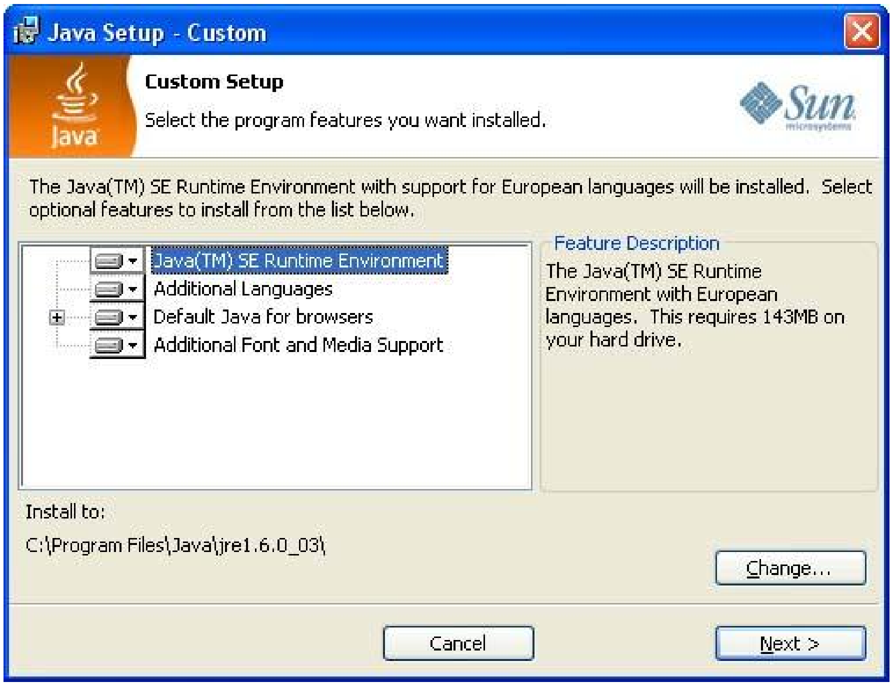
Please wait while the Install Wizard installs Java(TM) SE Development Kit 6 Update 3. This may take several minutes.

Status:

**Figure 6.4:** Installing Screen

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

- That is Java Setup and need you to choose **“Java (TM) SE Runtime Environment**”. As describe above, you allow changing any destination path for install the Java Program. If you need to use the default setting just continues the next process and click the **“Next”** button.



The Java(TM) SE Runtime Environment with support for European languages will be installed. Select optional features to install from the list below.

- Java(TM) SE Runtime Environment
- Additional Languages
- Default Java for browsers
- Additional Font and Media Support

Feature Description

The Java(TM) SE Runtime Environment with European languages. This requires 143MB on your hard drive.

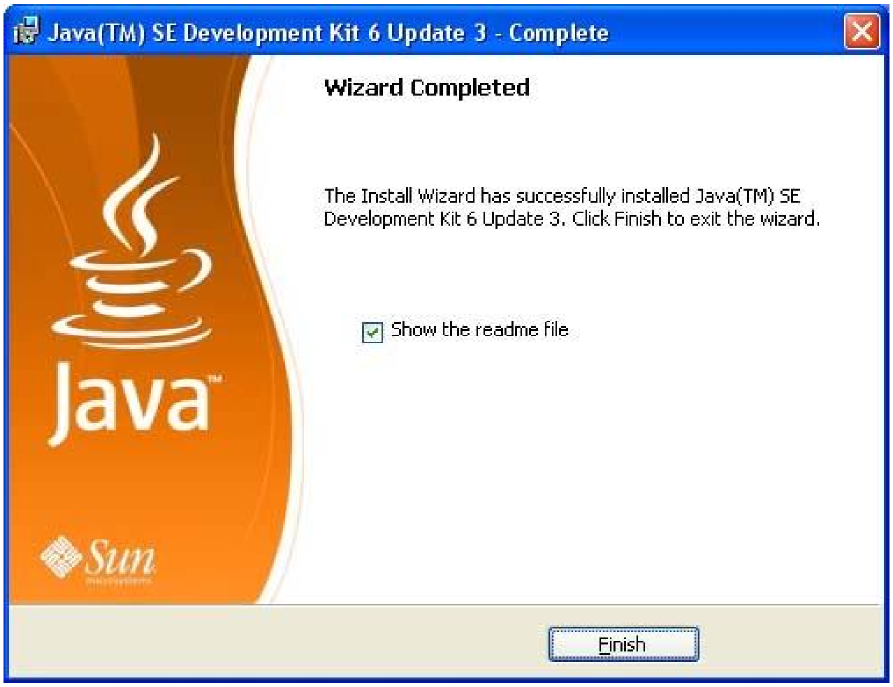
Install to:

C:\Program Files\Java\jre1.6.0_03\

**Figure 6.5:** Java Setup Screen

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

- After all programs are finished, you will see the Wizard completed screen just like Figure below. You can show the readme file to read short information about the installation program. Then click **“Finish”** button to end this process.



Wizard Completed

The Install Wizard has successfully installed Java(TM) SE Development Kit 6 Update 3. Click Finish to exit the wizard.

☑ Show the readme file

**Figure 6.6:** Complete Screen

- After installation has completed, you required to restart your computer.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

### 6.4.2 Eclipse Installation

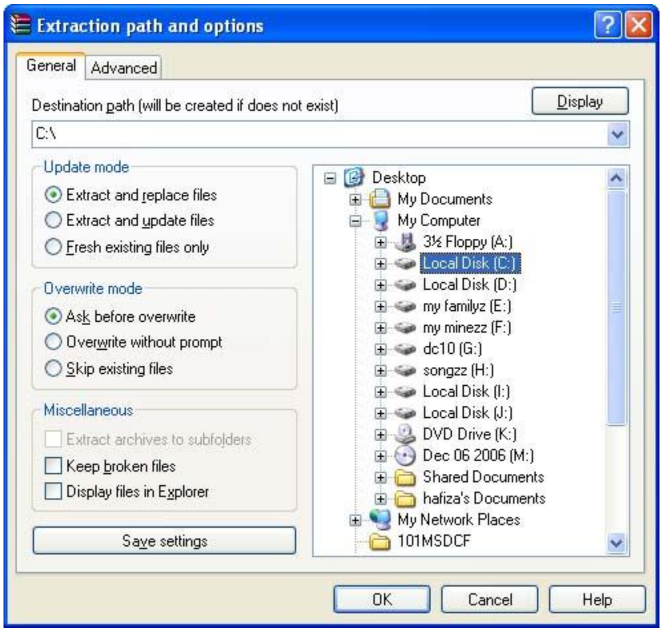
Eclipse is an open source community, whose projects are focused on building an extensible development platform, runtimes and application frameworks for building, deploying and managing software across the entire software lifecycle.

The Eclipse open source community has over 60 open source projects. These projects can be conceptually organized into seven different "pillars" or categories:

1. Enterprise Development
2. Embedded and Device Development
3. Rich Client Platform
4. Rich Internet Applications
5. Application Frameworks
6. Application Lifecycle Management (ALM)
7. Service Oriented Architecture (SOA)

Refer to the Eclipse website [ http://www.eclipse.org ] for further information.

- Extract eclipse-SDK-3.3.1.1-win32.zip to C:\. You can select any folder that you want.



Extraction path and options

General | Advanced

Destination path (will be created if does not exist): C:\

Update mode
● Extract and replace files
○ Extract and update files
○ Fresh existing files only

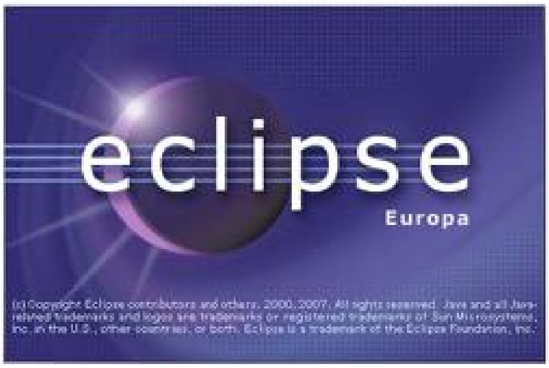
Overwrite mode
● Ask before overwrite
○ Overwrite without prompt
○ Skip existing files

Miscellaneous
□ Extract archives to subfolders
□ Keep broken files
□ Display files in Explorer

**Figure 6.7:** Extract eclipse to location folder

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

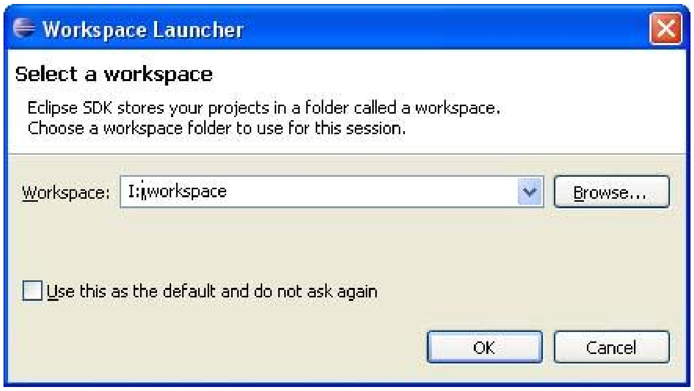
- Launch Eclipse application with double click on eclipse.exe



(c) Copyright Eclipse contributors and others. 2000, 2007. All rights reserved. Java and all Java-related trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S., other countries, or both. Eclipse is a trademark of the Eclipse Foundation, Inc.

**Figure 6.8:** Launch Eclipse

- Select path for workspace



Select a workspace

Eclipse SDK stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

Workspace: I:\workspace

□ Use this as the default and do not ask again

**Figure 6.9:** Select a workspace

- Finish

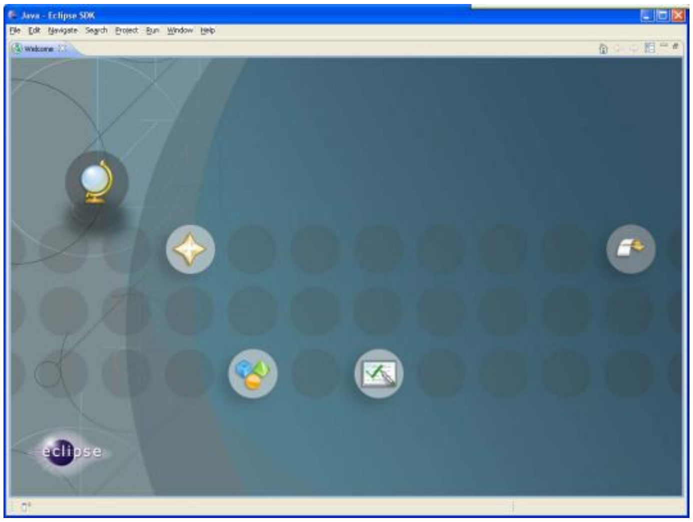


**Figure 6.10:** First screen of Eclipse

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

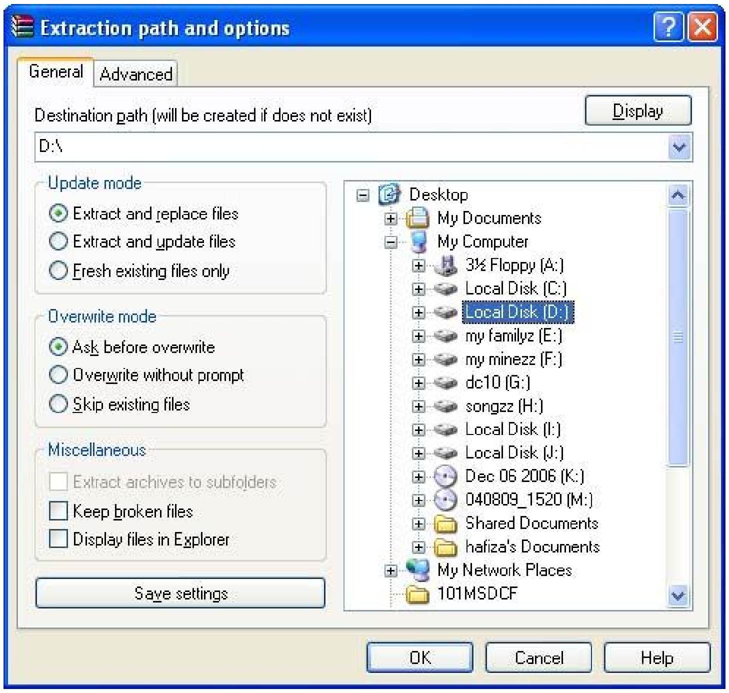
### 6.4.3 JBossIDE Installation

JBoss Eclipse IDE is a set of plug-in designed to be integrated in the Eclipse IDE runtime environment. Some of benefits of JBossIDE are:-

- Extensive and intuitive support for XDoclet.
- The debugging and monitoring of JBoss servers and the controlling of their life cycles.
- An easy way to configure the packaging layout of archives (packed or exploded)
- A simple way to deploy the packaged and/or exploded archive to a JB oss server
- Several J2EE wizards to ease and simplify J2EE development.
- Source code editors for JSP, HTML, and XML

#### 6.4.3.1 Extract JBoss Application Server

- Extract jboss-4.2.2.GA.zip to any location (default is C:\)



Extraction path and options

Destination path (will be created if does not exist)
D:\

Update mode
● Extract and replace files
○ Extract and update files
○ Fresh existing files only

Overwrite mode
● Ask before overwrite
○ Overwrite without prompt
○ Skip existing files

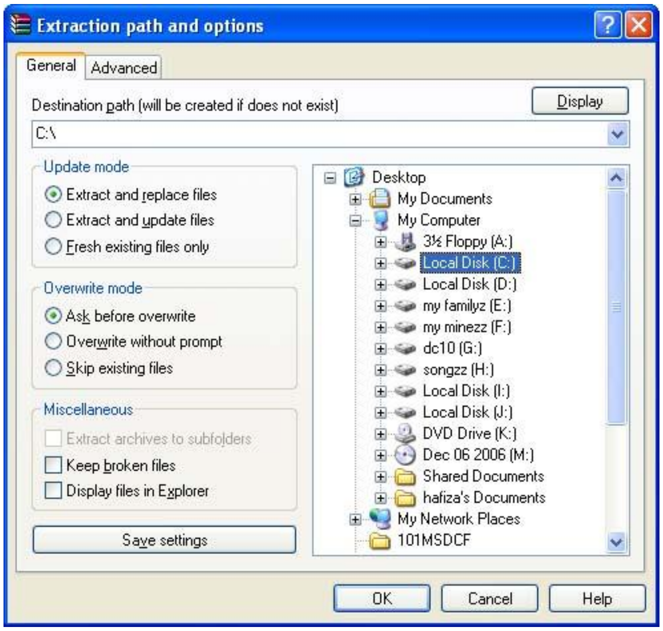
Miscellaneous
□ Extract archives to subfolders
□ Keep broken files
□ Display files in Explorer

**Figure 6.11:** Extract Jboss

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

### 6.4.3.2 Extract JBoss IDE

- Extract JBossIDE-2.0.0.Beta2-ALL.zip to C:\



Extraction path and options

General | Advanced

Destination path (will be created if does not exist): C:\

Update mode
- ● Extract and replace files
- ○ Extract and update files
- ○ Fresh existing files only

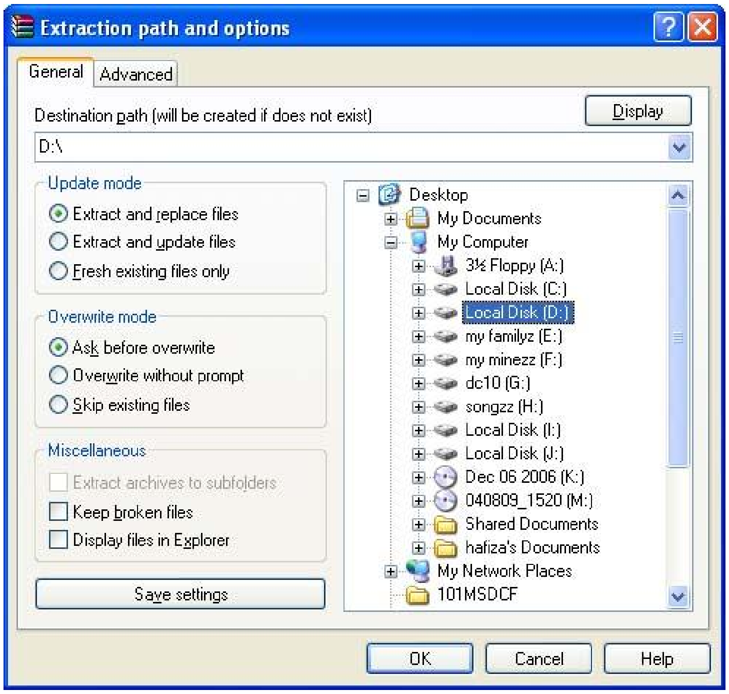
Overwrite mode
- ● Ask before overwrite
- ○ Overwrite without prompt
- ○ Skip existing files

Miscellaneous
- □ Extract archives to subfolders
- □ Keep broken files
- □ Display files in Explorer

**Figure 6.12:** Extract Jboss IDE

### 6.4.3.3 Extract Jboss Control Panel 1.0

- Extract Jboss Control Panel 1.0.zip to any location (default is C: \)



Extraction path and options

General | Advanced

Destination path (will be created if does not exist): D:\

Update mode
- ● Extract and replace files
- ○ Extract and update files
- ○ Fresh existing files only

Overwrite mode
- ● Ask before overwrite
- ○ Overwrite without prompt
- ○ Skip existing files

Miscellaneous
- □ Extract archives to subfolders
- □ Keep broken files
- □ Display files in Explorer

**Figure 6.13:** Extract Jboss Control Panel 1.0

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

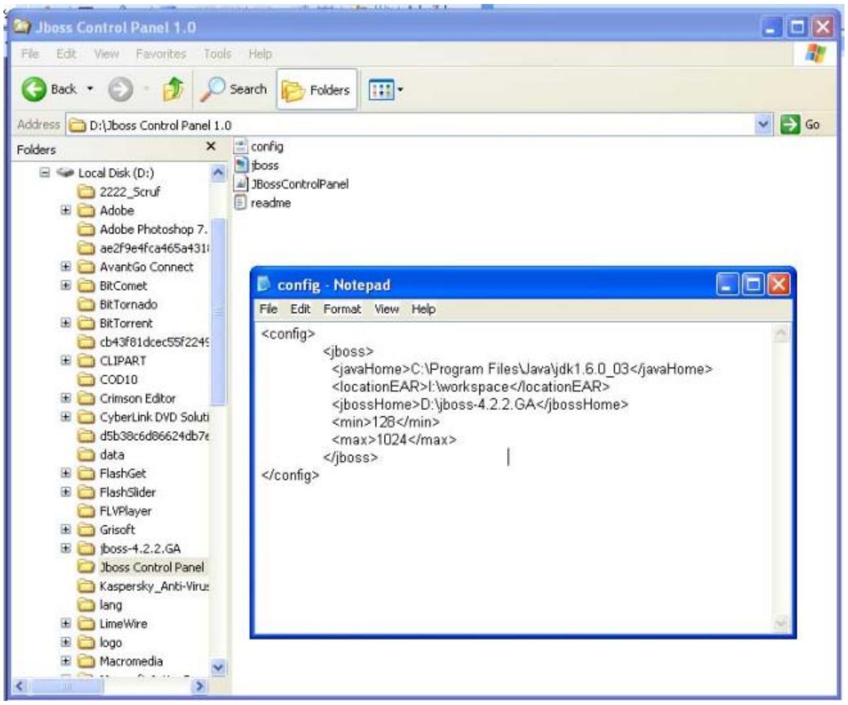
- Edit config.xml file from Jboss Control Panel 1.0 location:

  **1 - <javaHome>$path_java_home</javaHome>**

  **2 - <locationEAR> $path_workspace</locationEAR>**

  **3 - <jbossHome> $path_jboss_home</jbossHome>**

- Example as **Figure 5.14** below:



Address D:\Jboss Control Panel 1.0

```
<config>
        <jboss>
          <javaHome>C:\Program Files\Java\jdk1.6.0_03</javaHome>
          <locationEAR>I:\workspace</locationEAR>
          <jbossHome>D:\jboss-4.2.2.GA</jbossHome>
          <min>128</min>
          <max>1024</max>
        </jboss>
</config>
```

**Figure 6.14:** Edit config.xml

- To open this control panel, run the JBossControlPanel.jar from JBoss Control Panel 1.0 location.
- The JBoss control panel icon will be displayed in system tray.
- Click on the icon to open the panel.

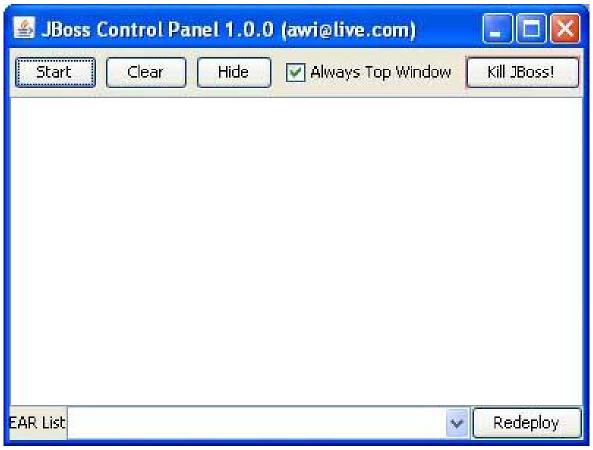


**Figure 6.15:** Screen of JBoss Control Panel 1.0

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

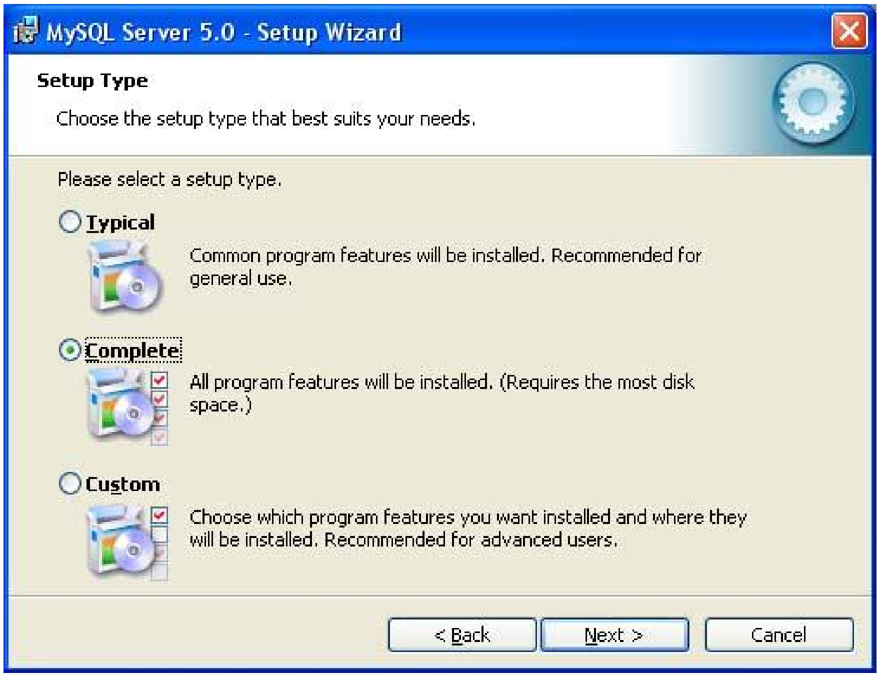
### 6.4.4 Server Database

In this project, MySQL has been chosen as server database. That is because it gives more advantages. Some of the advantages using MySQL are following: -

- Open source database
- Have fast performance, high reliability and easy to use
- Runs more than 20 platforms includi ng Linux, Windows, OS
- Giving the kind of flexibility

#### 6.4.4.1 MySQL Installation

- Extract **mysql-5.0.45-win32.zip** to any location
- To run MySQL Server 5.0, double click on **“setup.exe”** to begin the installation.
- At the welcome screen of setup wizard click **“Next”** button
- Now you get this window like Figure above. You can choose the type of the installation process. Select the **“Complete”** option as the setup type then click **“Next”** button to continue the process



MySQL Server 5.0 - Setup Wizard

Setup Type

Choose the setup type that best suits your needs.

Please select a setup type.

○ Typical

Common program features will be installed. Recommended for general use.

● Complete

All program features will be installed. (Requires the most disk space.)

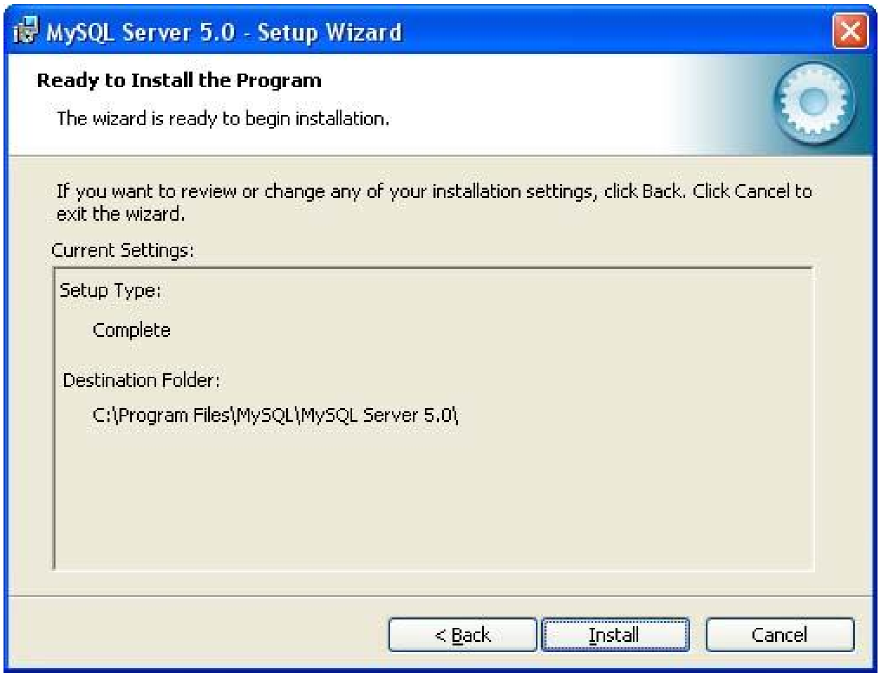
○ Custom

Choose which program features you want installed and where they will be installed. Recommended for advanced users.

**Figure 6.16:** Select the setup Type

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

- Now is ready to install the database into your computer. Click **"Install"** button to begin installation. Please wait until the installation process finish.



**Ready to Install the Program**

The wizard is ready to begin installation.

If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.

Current Settings:

Setup Type:

Complete

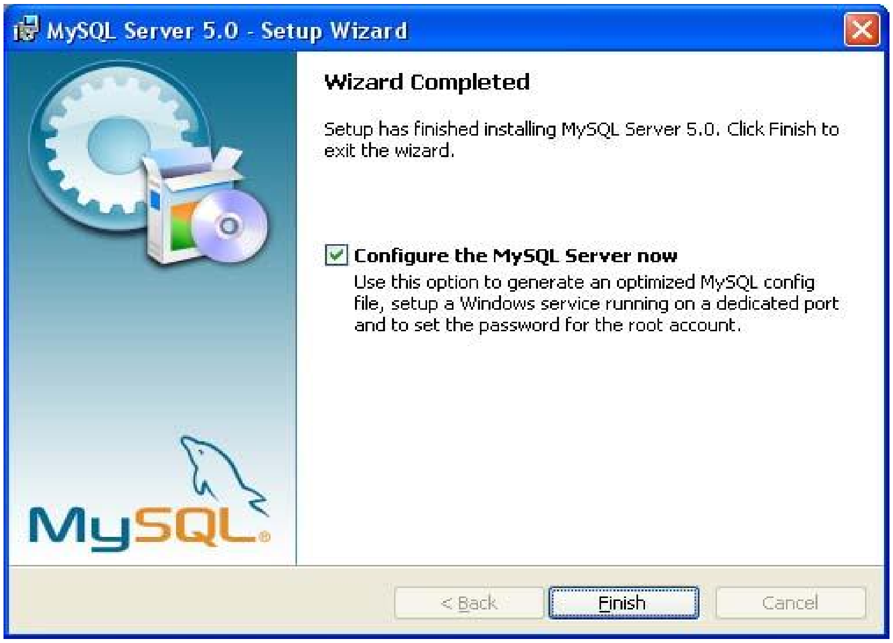
Destination Folder:

C:\Program Files\MySQL\MySQL Server 5.0\

**Figure 6.17:** Ready to Install MySQL Server

- The MySQL Server Database is now installed. At this point the server must be configured, choose "**Configure the MySQLServer now**" option and click "**Finish**" button.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



MySQL Server 5.0 - Setup Wizard

Wizard Completed

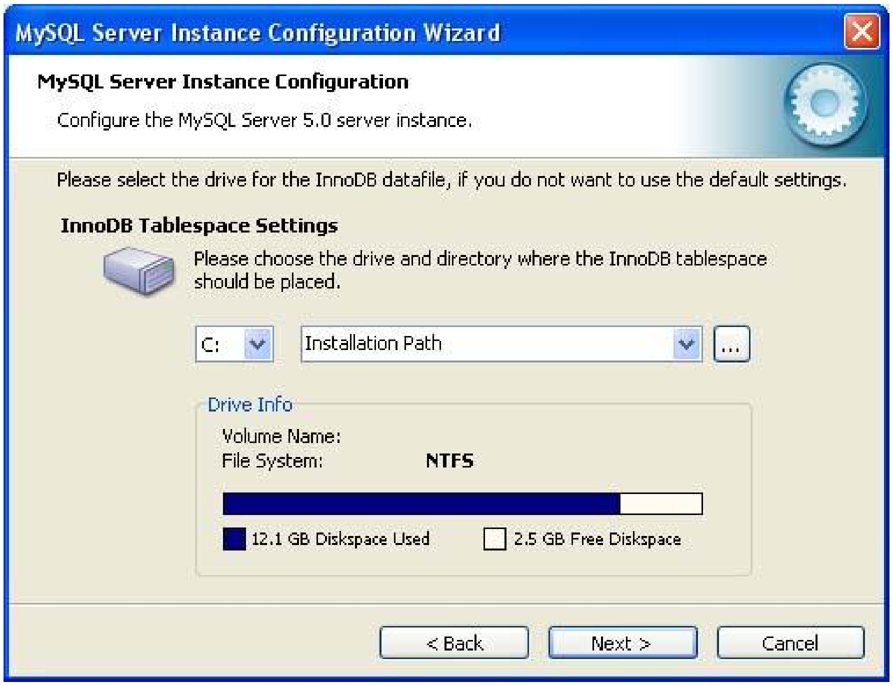
Setup has finished installing MySQL Server 5.0. Click Finish to exit the wizard.

☑ Configure the MySQL Server now

Use this option to generate an optimized MySQL config file, setup a Windows service running on a dedicated port and to set the password for the root account.

**Figure 6.18:** Complete Screen

- Choose a location that will house the database tablespace and click **"Next"**



MySQL Server Instance Configuration Wizard

MySQL Server Instance Configuration

Configure the MySQL Server 5.0 server instance.

Please select the drive for the InnoDB datafile, if you do not want to use the default settings.

InnoDB Tablespace Settings

Please choose the drive and directory where the InnoDB tablespace should be placed.

C: | Installation Path

Drive Info

Volume Name:

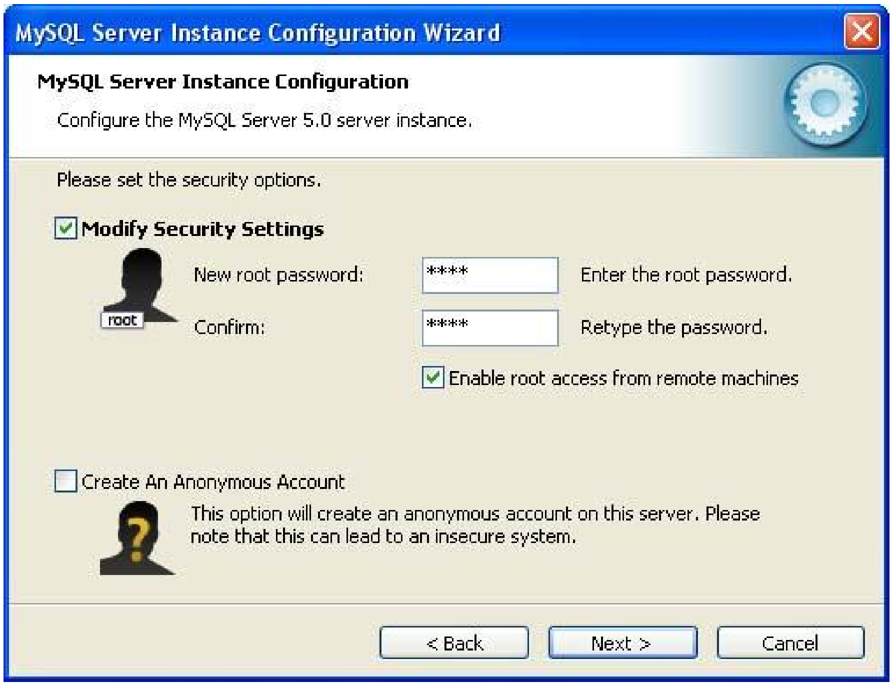
File System: NTFS

■ 12.1 GB Diskspace Used □ 2.5 GB Free Diskspace

**Figure 6.19:** MySQL Server Instance Configuration

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

- In this step, you choose enable the root user and choose a password. Check the **“Modify Security Setting”** option, then enter and confirm the new root user password. When finished, click **“Next”**



MySQL Server Instance Configuration Wizard

**MySQL Server Instance Configuration**

Configure the MySQL Server 5.0 server instance.

Please set the security options.

☑ **Modify Security Settings**

New root password: **** Enter the root password.

Confirm: **** Retype the password.

☑ Enable root access from remote machines

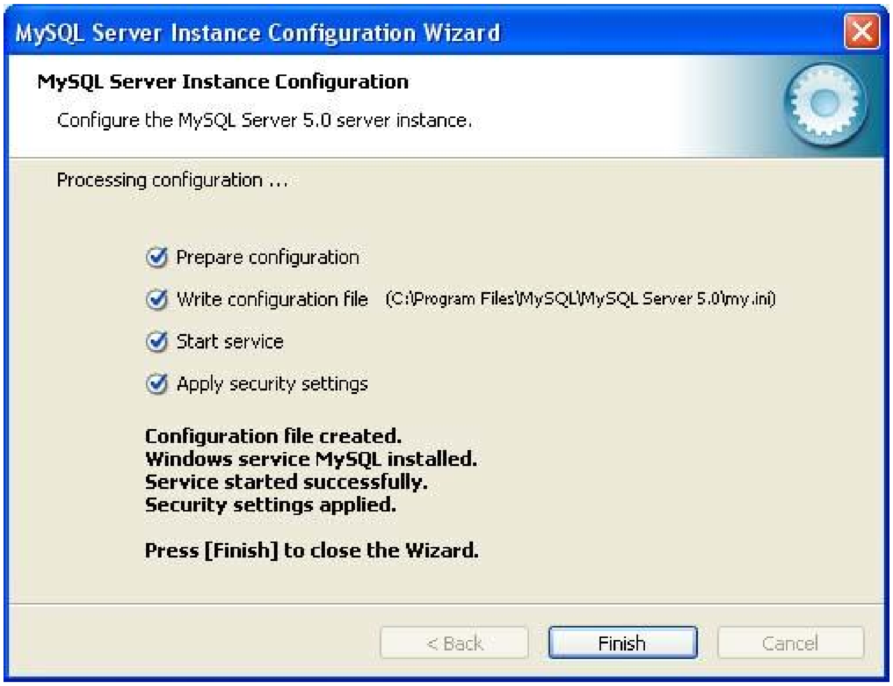
□ Create An Anonymous Account

This option will create an anonymous account on this server. Please note that this can lead to an insecure system.

**Figure 6.20:** Modify Security Setting

- Now the installation finish. Click **“Finish”** to close the wizard

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



MySQL Server Instance Configuration

Configure the MySQL Server 5.0 server instance.

Processing configuration ...

☑ Prepare configuration

☑ Write configuration file (C:\Program Files\MySQL\MySQL Server 5.0\my.ini)

☑ Start service

☑ Apply security settings

Configuration file created.
Windows service MySQL installed.
Service started successfully.
Security settings applied.

Press [Finish] to close the Wizard.

**Figure 6.21:** Finish Screen

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

# CHAPTER 7

# SYSTEM IMPLEMENTATION

## 7.1 Introduction

In this section, we will discuss briefly the system implementation. Basically there are three levels of user namely Hi gh Level Admin, Admin and Clerk.

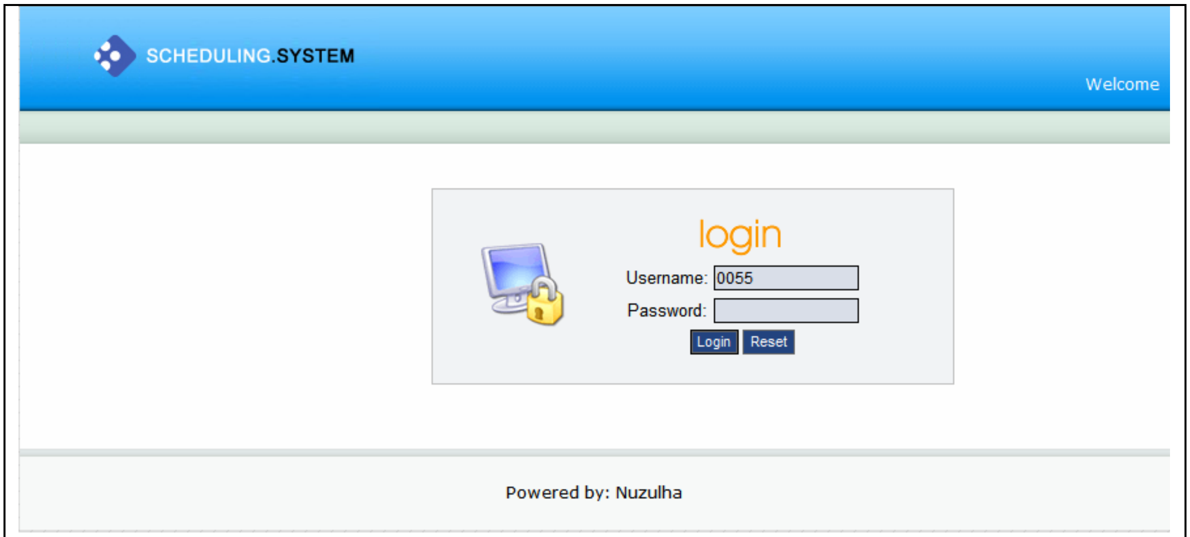
## 7.2 High Level Admin

High level admin is referring to those who have the highest authority in the system. It can be the Headquarters office for example the IT Manager or any other post which equivalent to it.

### 7.2.1 Login

The first user of this system is high level admin. There is only a single position for high level admin in this system. Only one account for high level admin is provided. See **Figure 7.1**. If the user failed to enter the correct login username, the system will prom pt out an error message. These to ensure only the authorized person are allowed to use the system. The source code for this function is shown in **Figure 7.2**.

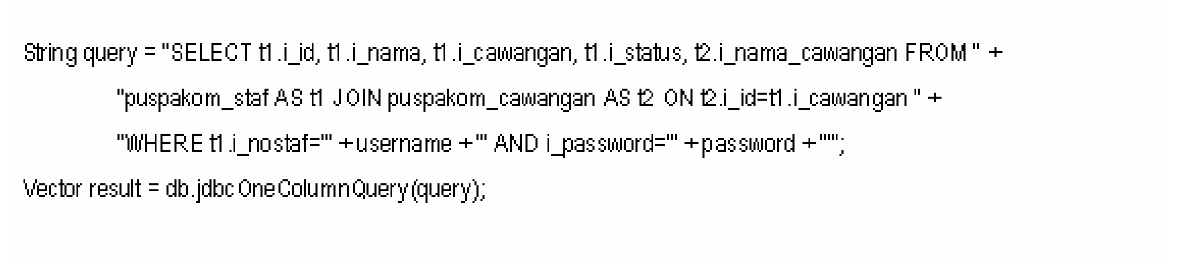
This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



Username: 0055

Password:

**Figure 7.1**: Login screen for high level admin.



```
String query = "SELECT t1.i_id, t1.i_nama, t1.i_cawangan, t1.i_status, t2.i_nama_cawangan FROM " +
            "puspakom_staf AS t1 JOIN puspakom_cawangan AS t2 ON t2.i_id=t1.i_cawangan " +
            "WHERE t1.i_nostaf='" +username +"' AND i_password='" +password +"'";
Vector result = db.jdbcOneColumnQuery(query);
```

**Figure 7.2**: Source code for login function.

### 7.2.2 Menu 1: Centre

There a few menus available when we login to the system. The list of menu as depicted in **Figure 7.3** is recorded below:

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

- Centre
- Centre Admin
- Public Holidays
- Constraint
- Booking
- Register Inspection
- Start Inspection
- Finish Inspection
- Searching
- Booking Schedule
- Inspection Schedule
- Change Password
- Logout

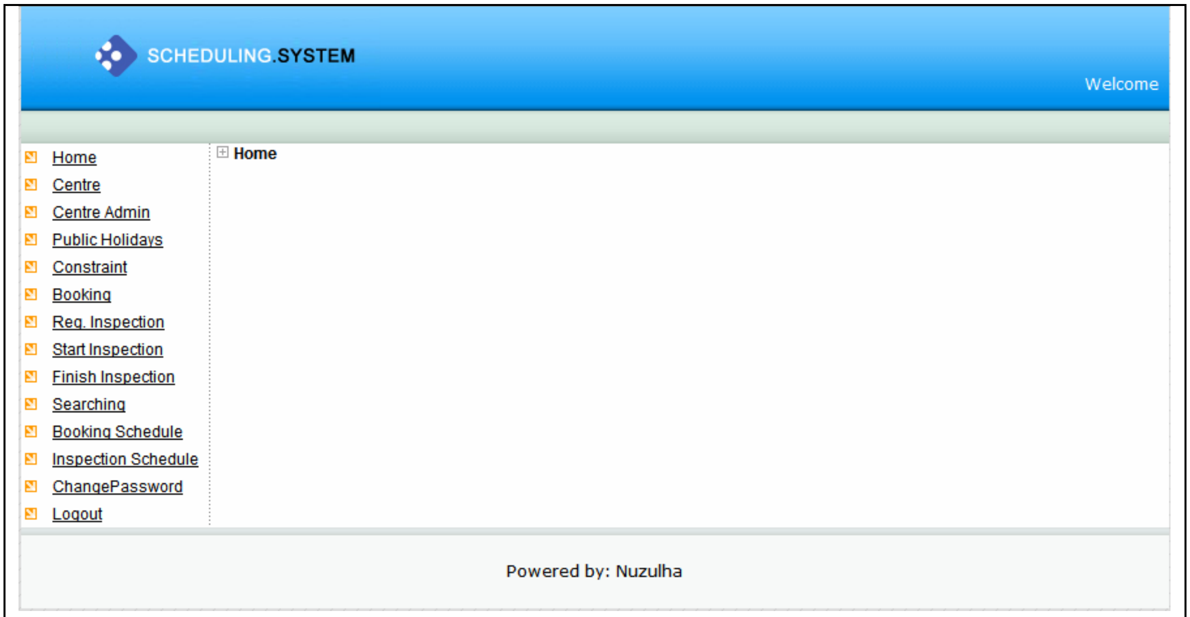


**Figure 7.3**: Menu Screen for super admin.

A menu for centre management is provided as depicted in **Figure 7.4**. In this section it provides in managing all center's information included address, phone number and fax

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

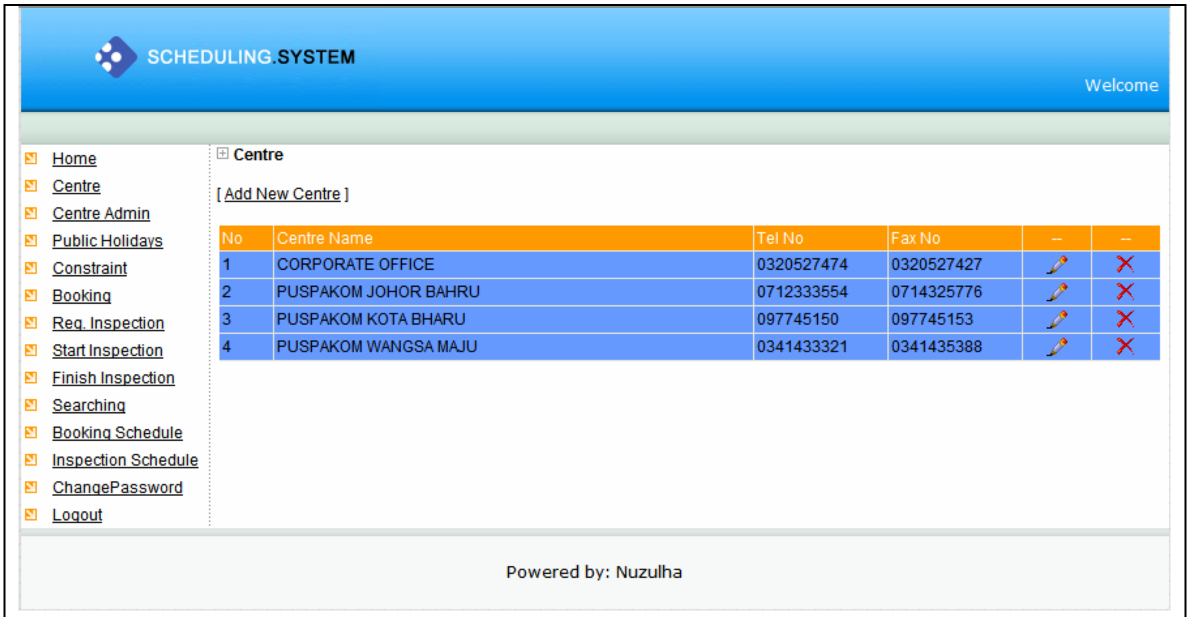
number. The high level admin can update and delete the account for corporate office and all centers.

### 7.2.3 Submenu 1: Add New Centre

**Figure 7.5** shows how the system can add new centre. This feat ure can only be manages by high level admin. The values required to add new centre as listed below:

- Centre Name
- Centre Code
- Address
- Postcode
- Town
- State
- Tel. No
- Fax. No

The source code of this feature is shown in **Figure 7.6**.



| No | Centre Name | Tel No | Fax No | – | – |
|---|---|---|---|---|---|
| 1 | CORPORATE OFFICE | 0320527474 | 0320527427 | ✎ | ✗ |
| 2 | PUSPAKOM JOHOR BAHRU | 0712333554 | 0714325776 | ✎ | ✗ |
| 3 | PUSPAKOM KOTA BHARU | 097745150 | 097745153 | ✎ | ✗ |
| 4 | PUSPAKOM WANGSA MAJU | 0341433321 | 0341435388 | ✎ | ✗ |

**Figure 7.4:** Centre Menu for super admin.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

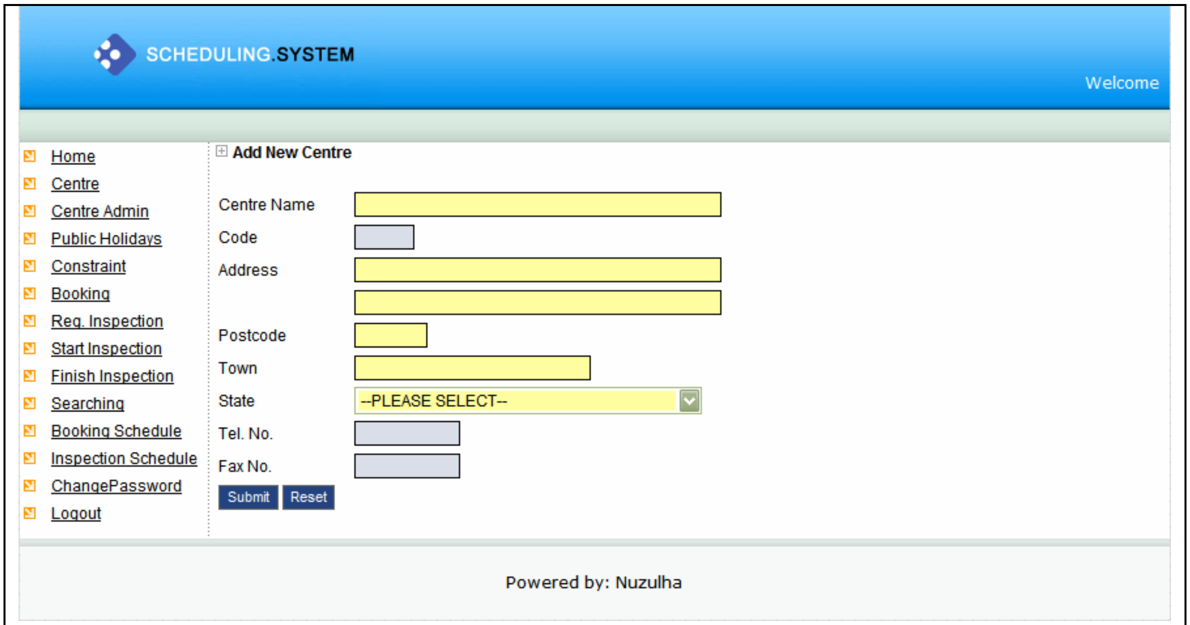
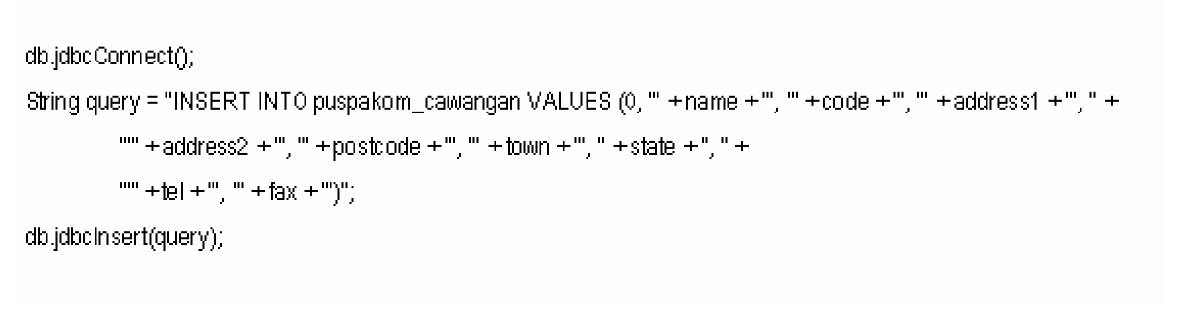


**Figure 7.5**: Menu for super admin to add new Centre.



```
db.jdbcConnect();
String query = "INSERT INTO puspakom_cawangan VALUES (0, '" +name +"', '" +code +"', '" +address1 +"', " +
          "'" +address2 +"', '" +postcode +"', '" +town +"', " +state +", " +
          "'" +tel +"', '" +fax +"')";
db.jdbcInsert(query);
```

**Figure 7.6**: Source code for menu: add new centre.

### 7.2.4 Menu 2: Centre Admin

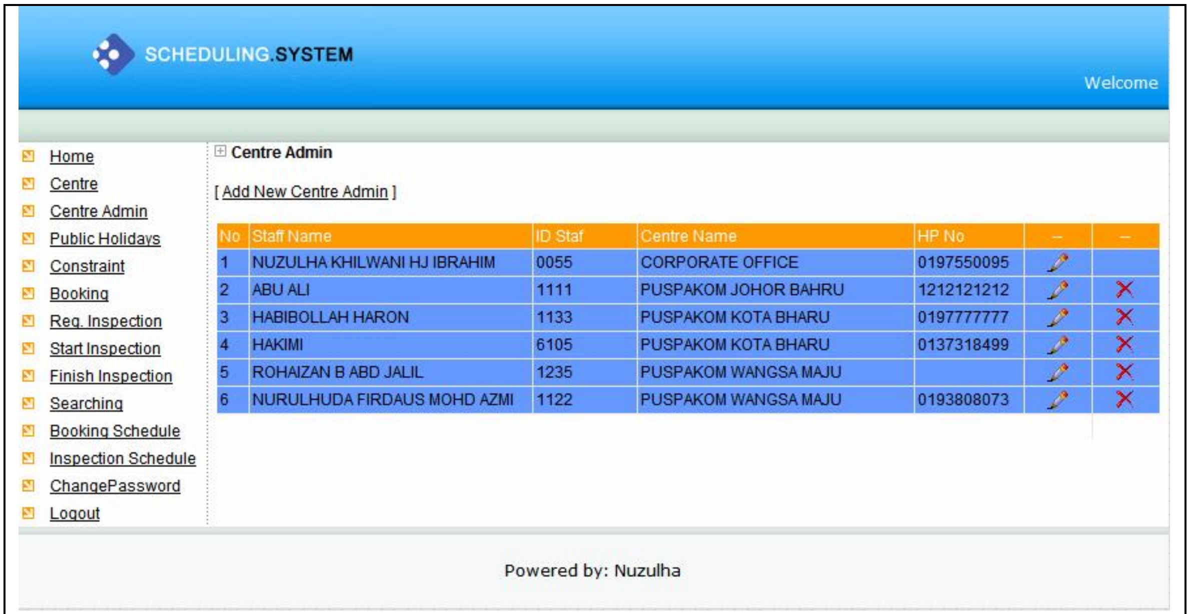
**Figure 7.7** shows the feature on how to manage information on centre admin, or sta ff of the organization. High level admin can also update and delete the record of admin fr om other centers. However, the record of high level admin can only be updated.

The value appeared in this record are listed below:

- Number of staff
- Staff Name

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

- Staff Identity Number
- Centre Name
- Mobile phone number



| No | Staff Name | ID Staf | Centre Name | HP No | – | – |
|---|---|---|---|---|---|---|
| 1 | NUZULHA KHILWANI HJ IBRAHIM | 0055 | CORPORATE OFFICE | 0197550095 | ✎ | |
| 2 | ABU ALI | 1111 | PUSPAKOM JOHOR BAHRU | 1212121212 | ✎ | ✗ |
| 3 | HABIBOLLAH HARON | 1133 | PUSPAKOM KOTA BHARU | 0197777777 | ✎ | ✗ |
| 4 | HAKIMI | 6105 | PUSPAKOM KOTA BHARU | 0137318499 | ✎ | ✗ |
| 5 | ROHAIZAN B ABD JALIL | 1235 | PUSPAKOM WANGSA MAJU | | ✎ | ✗ |
| 6 | NURULHUDA FIRDAUS MOHD AZMI | 1122 | PUSPAKOM WANGSA MAJU | 0193808073 | ✎ | ✗ |

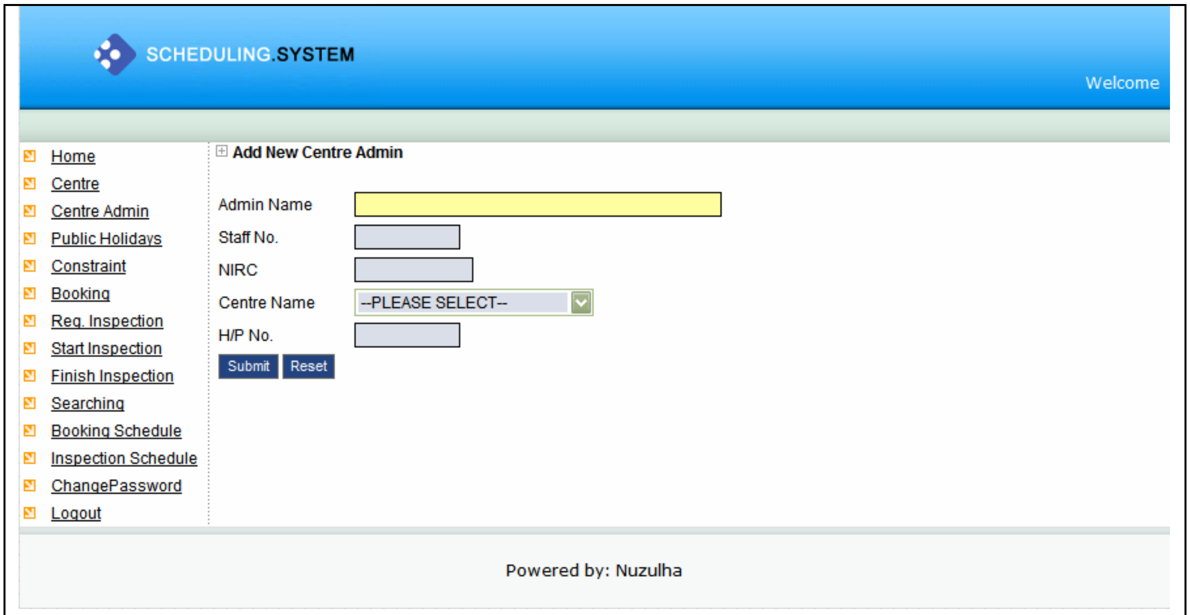
**Figure 7.7**: Screen show for menu: Centre Admin.

### 7.2.5 Submenu1: Add New Centre Admin

High level admin or admin may add new centre admin for their own centre. Please refer to **Figure 7.8**. High level admin can add new record for any new staff in corporate office, while the admin is capable to add new record for any new staff in his centre. The detail information required to add new centre admin is described below:

- Admin Name
- Staff Number
- No. of Identity Card
- Centre Name
- Mobile phone number

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com



Add New Centre Admin

Admin Name

Staff No.

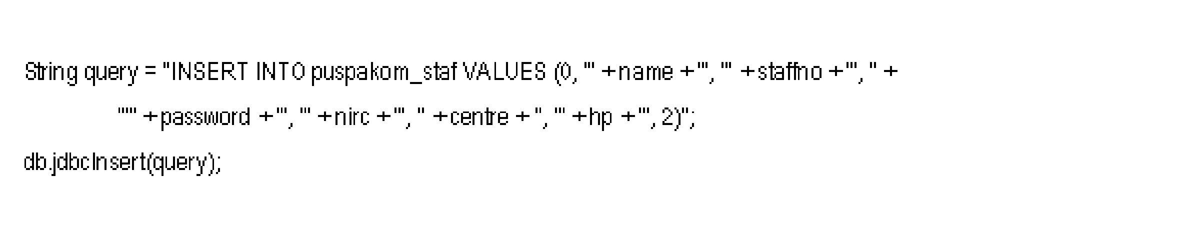
NIRC

Centre Name --PLEASE SELECT--

H/P No.

**Figure 7.8**: Screen show to add new centre admin.

The source code to add new centre admin is shown in **Figure 7.9**.



```
String query = "INSERT INTO puspakom_staf VALUES (0, '" +name +"', '" +staffno +"', '" +
            "'" +password +"', '" +nirc +"', '" +centre +"', '" +hp +"', 2)";
db.jdbcInsert(query);
```

**Figure 7.9**: Source code for menu: add new centre admin.

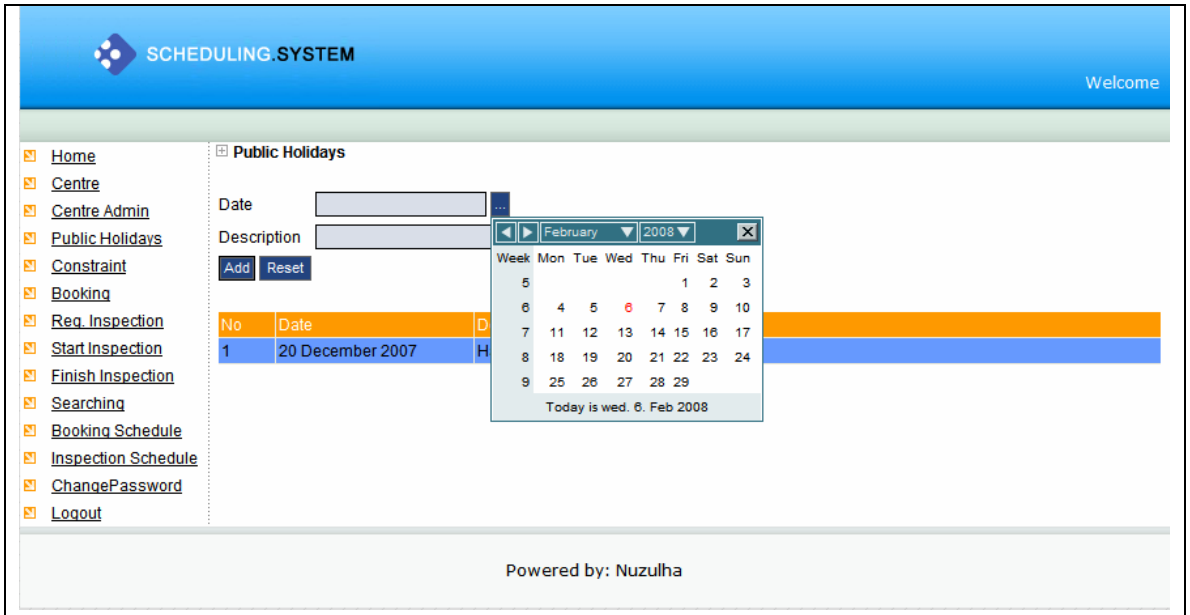
## 7.3 Managing Public Holidays

The modification of public holidays can be done by both high level admin and normal admin. **Figure 7.10** shows how the process being done. High level admin and admin are capable to add new record for any new public holidays declared for the centre. The detail of information required to add new public holiday is described below:

- Date
- Description

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

Then, the list of public holidays will be listed in this screen.



Date

Description

| No | Date | D |
|---|---|---|
| 1 | 20 December 2007 | H |

| Week | Mon | Tue | Wed | Thu | Fri | Sat | Sun |
|---|---|---|---|---|---|---|---|
| 5 | | | | | 1 | 2 | 3 |
| 6 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 7 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| 8 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 9 | 25 | 26 | 27 | 28 | 29 | | |

Today is wed. 6. Feb 2008

**Figure 7.10**: Menu to add new Centre.

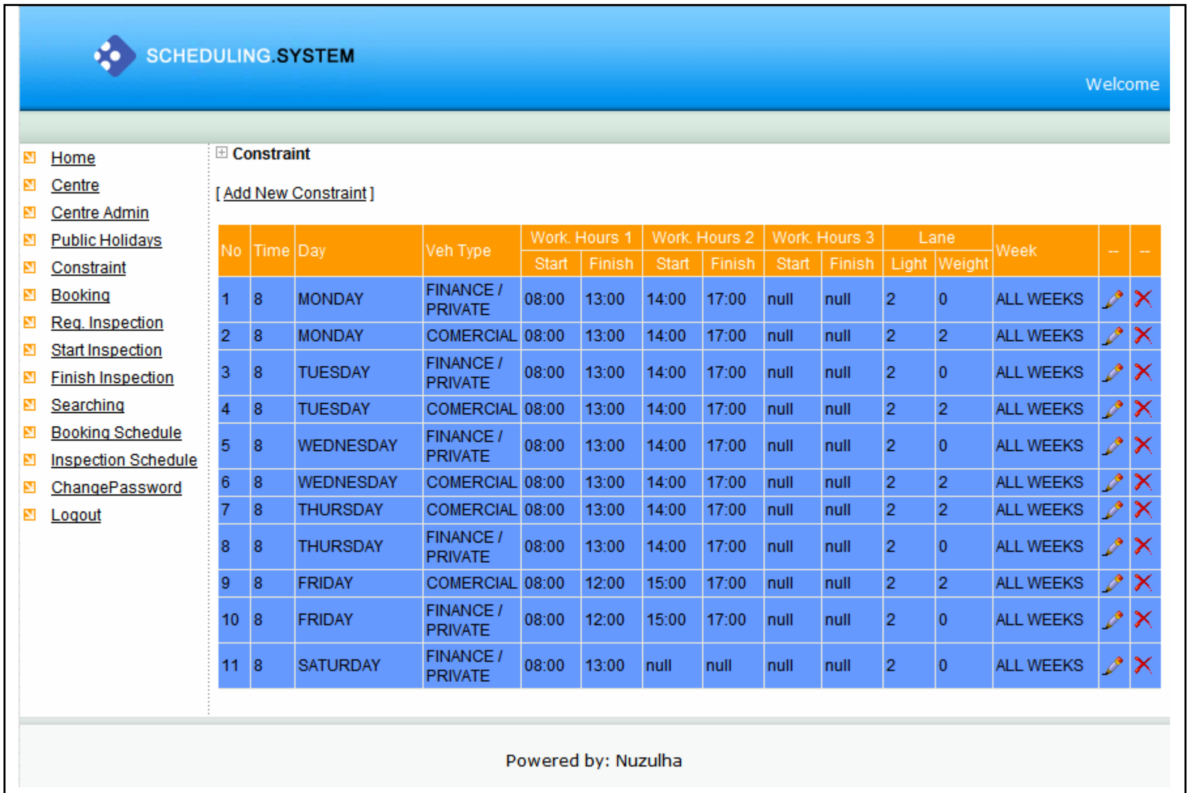
## 7.4 Constraint

This feature provide on how to set up list of working hours for the centre. Every high level admin or normal admin can make the modification for their own centre. Please refer to **Figure 7.11**.

### 7.4.1 Submenu 1: Add New Constraint

The submenu that shows in **Figure 7.12** is regards with how to add new working hours for the centre. High level admin and normal admin are capable to add new record for any new working hours declared for the centre.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com



⊞ Constraint

[ Add New Constraint ]

| No | Time | Day | Veh Type | Work. Hours 1 | | Work. Hours 2 | | Work. Hours 3 | | Lane | | Week | -- | -- |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Start | Finish | Start | Finish | Start | Finish | Light | Weight | | | |
| 1 | 8 | MONDAY | FINANCE / PRIVATE | 08:00 | 13:00 | 14:00 | 17:00 | null | null | 2 | 0 | ALL WEEKS | ✎ | ✕ |
| 2 | 8 | MONDAY | COMERCIAL | 08:00 | 13:00 | 14:00 | 17:00 | null | null | 2 | 2 | ALL WEEKS | ✎ | ✕ |
| 3 | 8 | TUESDAY | FINANCE / PRIVATE | 08:00 | 13:00 | 14:00 | 17:00 | null | null | 2 | 0 | ALL WEEKS | ✎ | ✕ |
| 4 | 8 | TUESDAY | COMERCIAL | 08:00 | 13:00 | 14:00 | 17:00 | null | null | 2 | 2 | ALL WEEKS | ✎ | ✕ |
| 5 | 8 | WEDNESDAY | FINANCE / PRIVATE | 08:00 | 13:00 | 14:00 | 17:00 | null | null | 2 | 0 | ALL WEEKS | ✎ | ✕ |
| 6 | 8 | WEDNESDAY | COMERCIAL | 08:00 | 13:00 | 14:00 | 17:00 | null | null | 2 | 2 | ALL WEEKS | ✎ | ✕ |
| 7 | 8 | THURSDAY | COMERCIAL | 08:00 | 13:00 | 14:00 | 17:00 | null | null | 2 | 2 | ALL WEEKS | ✎ | ✕ |
| 8 | 8 | THURSDAY | FINANCE / PRIVATE | 08:00 | 13:00 | 14:00 | 17:00 | null | null | 2 | 0 | ALL WEEKS | ✎ | ✕ |
| 9 | 8 | FRIDAY | COMERCIAL | 08:00 | 12:00 | 15:00 | 17:00 | null | null | 2 | 2 | ALL WEEKS | ✎ | ✕ |
| 10 | 8 | FRIDAY | FINANCE / PRIVATE | 08:00 | 12:00 | 15:00 | 17:00 | null | null | 2 | 0 | ALL WEEKS | ✎ | ✕ |
| 11 | 8 | SATURDAY | FINANCE / PRIVATE | 08:00 | 13:00 | null | null | null | null | 2 | 0 | ALL WEEKS | ✎ | ✕ |

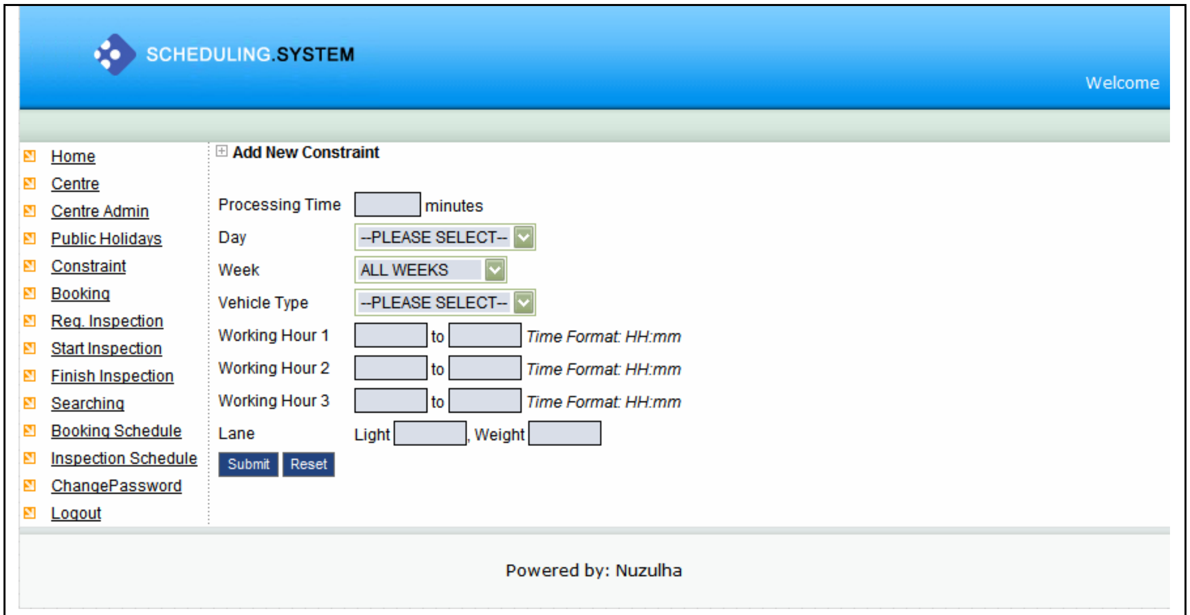
Powered by: Nuzulha

**Figure 7.11**: Menu List of Constraints.

The detail of information required to add new working hours is described below:

- Duration of processing time
- Type of Day (Monday, or others)
- Type of Week (First Week, or others)
- Working Hour 1 (First block of operation) in time format hh:mm
- Working Hour 2 (Second block of operation) in time format hh:mm
- Working Hour 3 (Third block of operation) in time format hh:mm

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



Add New Constraint

Processing Time minutes

Day --PLEASE SELECT--

Week ALL WEEKS

Vehicle Type --PLEASE SELECT--

Working Hour 1 to *Time Format: HH:mm*

Working Hour 2 to *Time Format: HH:mm*

Working Hour 3 to *Time Format: HH:mm*

Lane Light , Weight

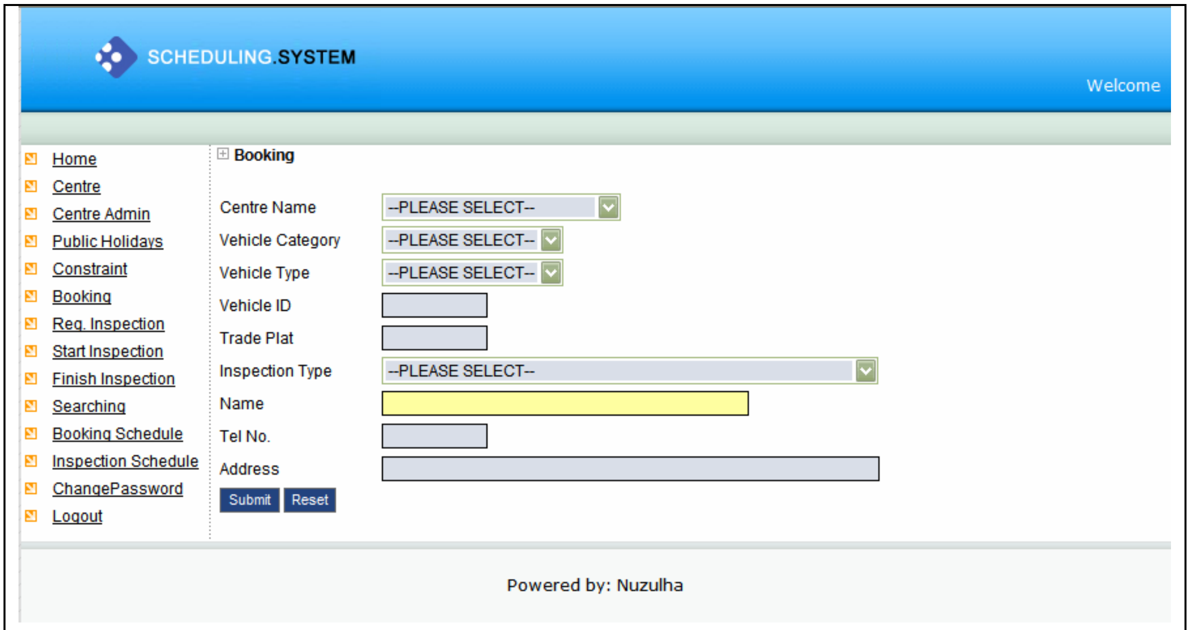
**Figure 7.12**: Menu to add new Centre.

## 7.5 Booking Appointment

This is the main menu in the system. This feature provides adding new appointment booking for the centre. Every high level admin or normal admin may add new booking appointment for their own centre. Refer to **Figure 7.13**. High level admin, normal admin and clerk are capable to add new booking for any customer. The detail of information required to add new booking is described below :

- Centre Name
- Vehicle Category (such as Finance, or others)
- Vehicle Type (Light or Weight)
- Vehicle ID (Plat number)
- Inspection Type (Routine, or others)
- Name of owner
- Phone number of owner
- Address of owner

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



Booking

Centre Name --PLEASE SELECT--
Vehicle Category --PLEASE SELECT--
Vehicle Type --PLEASE SELECT--
Vehicle ID
Trade Plat
Inspection Type --PLEASE SELECT--
Name
Tel No.
Address

**Figure 7.13**: Menu to Make New Booking.

The source code of this process is show in **Figure 7.14** and **Figure 7.15.**

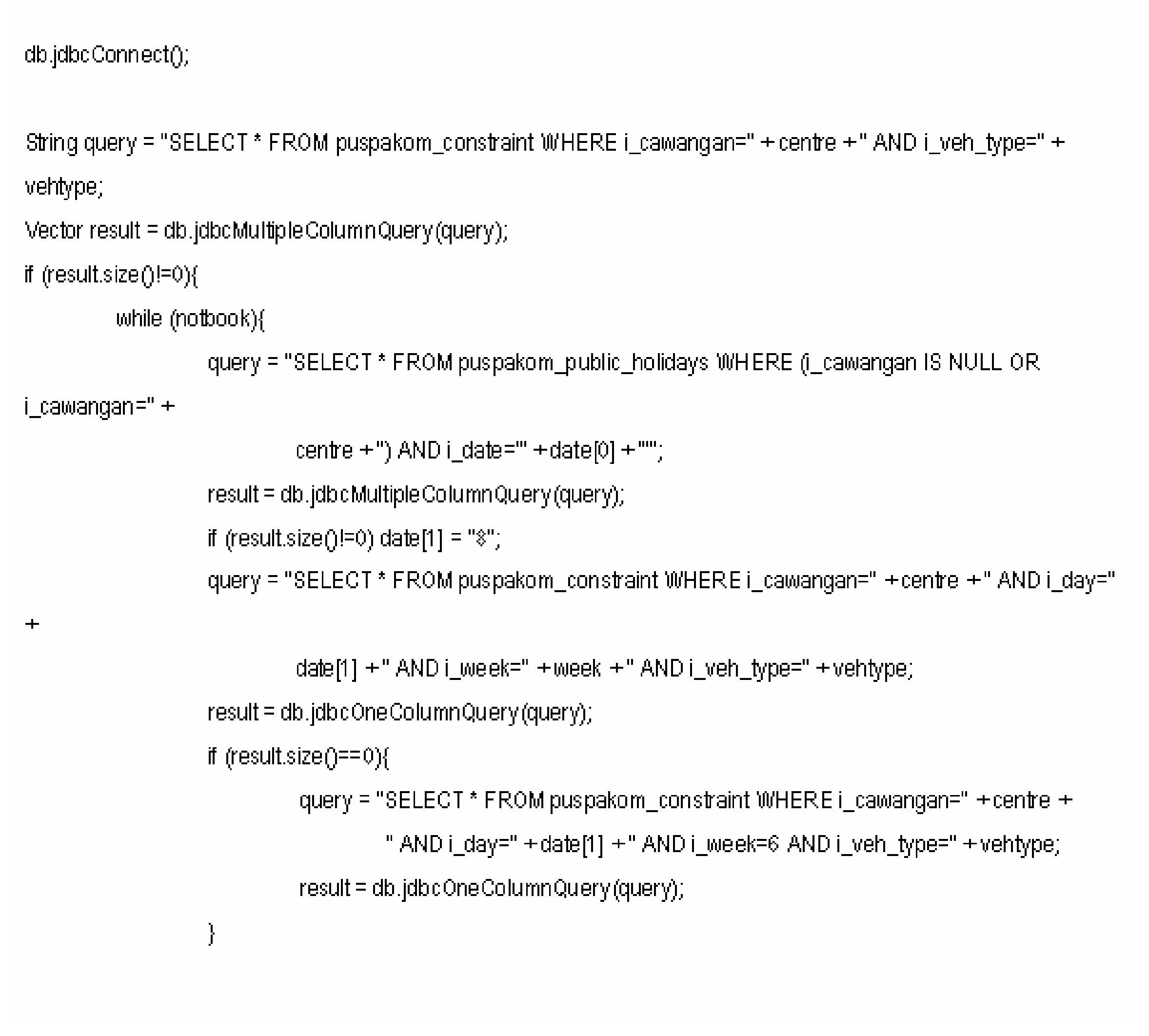
## 7.6 Inspection

This menu provides the feature to add a new inspection for any vehicle that registered at the centre. Every high level admin or normal admin may add new inspection data f or their own centre. Refer to **Figure 7.16**. The detail that required adding new inspection is described below:

- Appointment Number

The appointment number can be obtained from the booking slip. Then, the system will register the data recorded in the databas e and based from it the booking process will be provides.

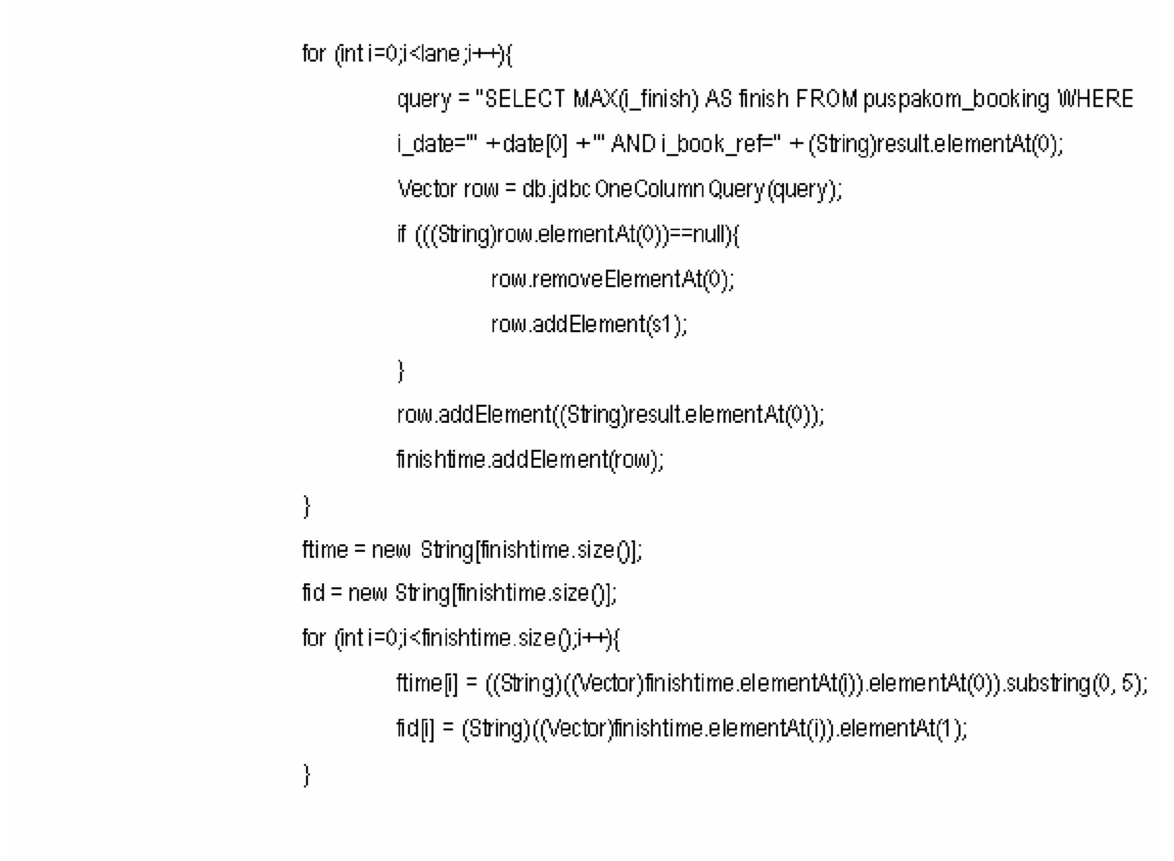
This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



```
db.jdbcConnect();

String query = "SELECT * FROM puspakom_constraint WHERE i_cawangan=" + centre + " AND i_veh_type=" +
vehtype;
Vector result = db.jdbcMultipleColumnQuery(query);
if (result.size()!=0){
        while (notbook){
                query = "SELECT * FROM puspakom_public_holidays WHERE (i_cawangan IS NULL OR
i_cawangan=" +
                        centre + ") AND i_date='" +date[0] + "'";
                result = db.jdbcMultipleColumnQuery(query);
                if (result.size()!=0) date[1] = "8";
                query = "SELECT * FROM puspakom_constraint WHERE i_cawangan=" +centre + " AND i_day="
+
                        date[1] + " AND i_week=" +week + " AND i_veh_type=" +vehtype;
                result = db.jdbcOneColumnQuery(query);
                if (result.size()==0){
                        query = "SELECT * FROM puspakom_constraint WHERE i_cawangan=" +centre +
                                " AND i_day=" +date[1] + " AND i_week=6 AND i_veh_type=" +vehtype;
                        result = db.jdbcOneColumnQuery(query);
                }
```

**Figure 7.14:** First part of the source code to make new appointment booking.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com



```
for (int i=0;i<lane;i++){
        query = "SELECT MAX(i_finish) AS finish FROM puspakom_booking WHERE
        i_date='" +date[0] +"' AND i_book_ref=" + (String)result.elementAt(0);
        Vector row = db.jdbc OneColumn Query(query);
        if (((String)row.elementAt(0))==null){
                row.removeElementAt(0);
                row.addElement(s1);
        }
        row.addElement((String)result.elementAt(0));
        finishtime.addElement(row);
}
ftime = new String[finishtime.size()];
fid = new String[finishtime.size()];
for (int i=0;i<finishtime.size();i++){
        ftime[i] = ((String)((Vector)finishtime.elementAt(i)).elementAt(0)).substring(0, 5);
        fid[i] = (String)((Vector)finishtime.elementAt(i)).elementAt(1);
}
```

**Figure 7.15:** Second part of the source code to make new appointment booking.

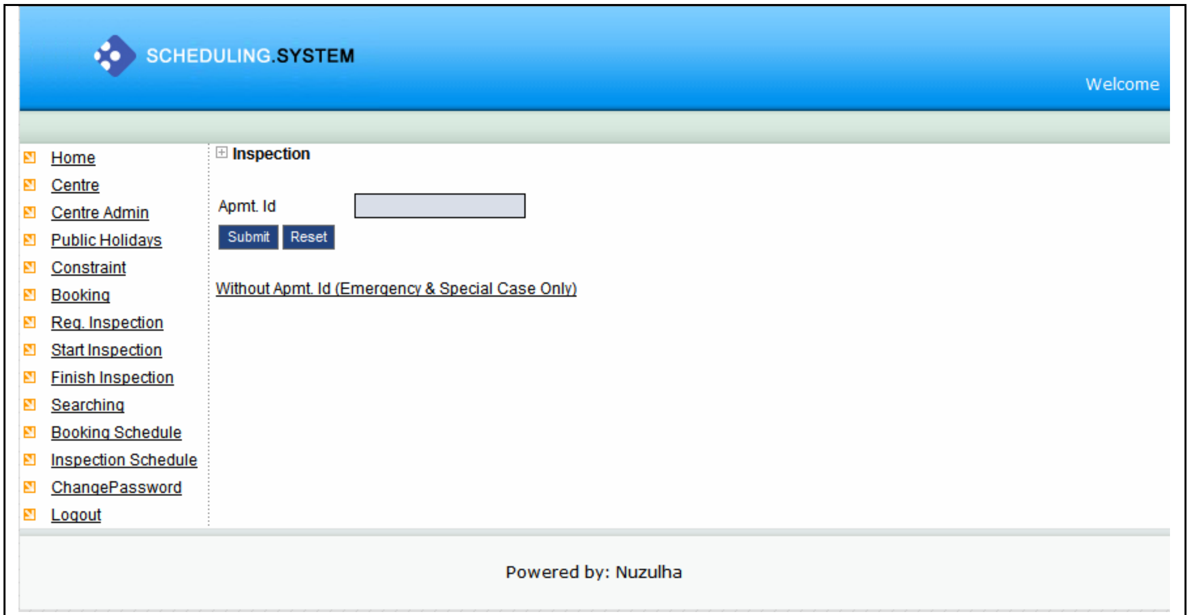


**Figure 7.16**: Menu to add new Inspection.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

### 7.6.1 Submenu 1: Inspection without Appointment ID and Special Case

For this case of inspection, the staff needs to make new record of inspection to get the details of the vehicle. Please refer to **Figure 7.17**. High level admin, normal admin and clerk are capable to add new booking at the centre. The detail of information required to add new appointment for special case inspection is described below:

- Centre Name
- Vehicle Category (such as Finance, or others)
- Vehicle Type (Light or Weight)
- Vehicle ID (Plat number)
- Inspection Type (Routine, or others)
- Name of owner
- Phone number of owner
- Address of owner

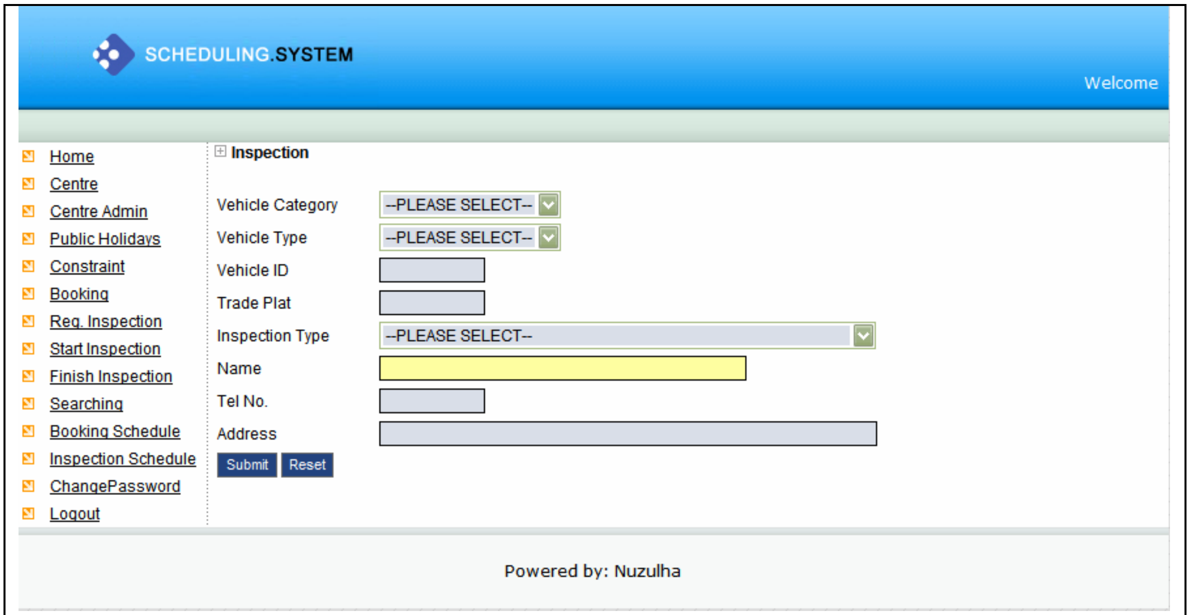


**Figure 7.17**: Submenu to add new Special Inspection

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 7.7 Menu 7: Start and End Inspection

**Figure 7.18** record the start time of inspection for every vehicle and **Figure 7.19** record the end time. The detail of information required to record start inspection is described below:

- Queue Number (can be obtained from the start inspection slip ).

## 7.8 Menu 9: Searching

This menu is to search the record of inspection for any vehicle. Refer to **Figure 7.20**. High level admin, normal admin and clerk are capable to search the record of vehicle inspection at the centre. The detail of information required to search information on vehicle inspection is described below:

- Vehicle Number (Plat Number), or
- Trade Plat

## 7.9 Menu 10: Booking Schedule

This menu is to view the booking schedule for the appropriate date. Refer to **Figure 7.21**. High level admin, normal admin and clerk are capable to view the booking schedule of vehicle inspection at the centre. The detail of information required to view booking schedule is described below:

- Centre Name
- Date of Inspection

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

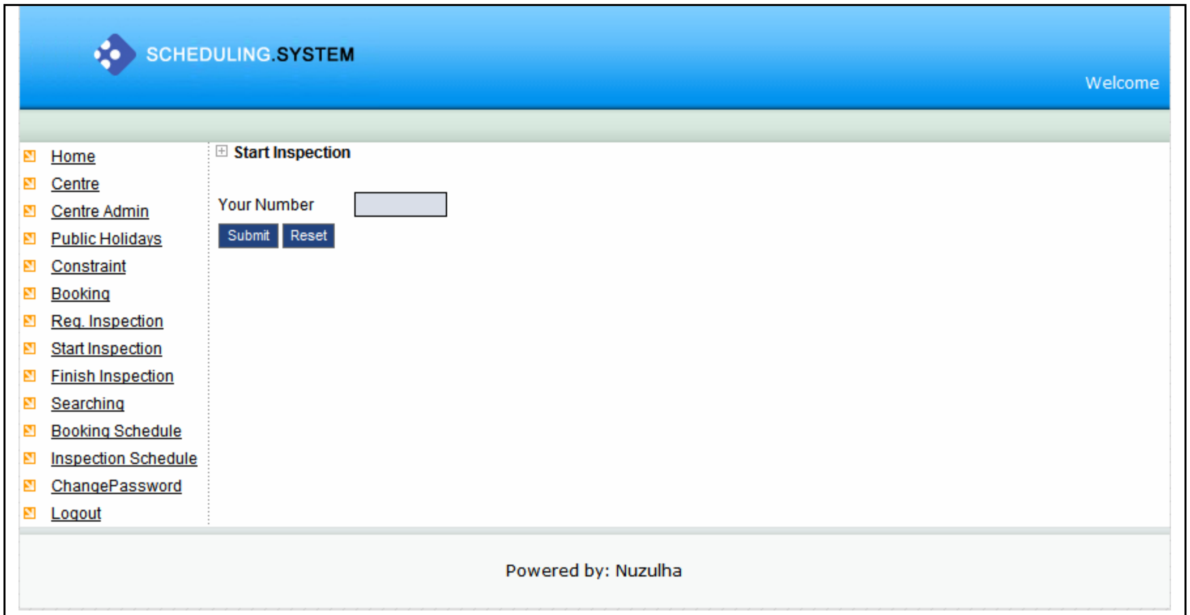


**Figure 7.18**: Menu to Record Start Inspection.

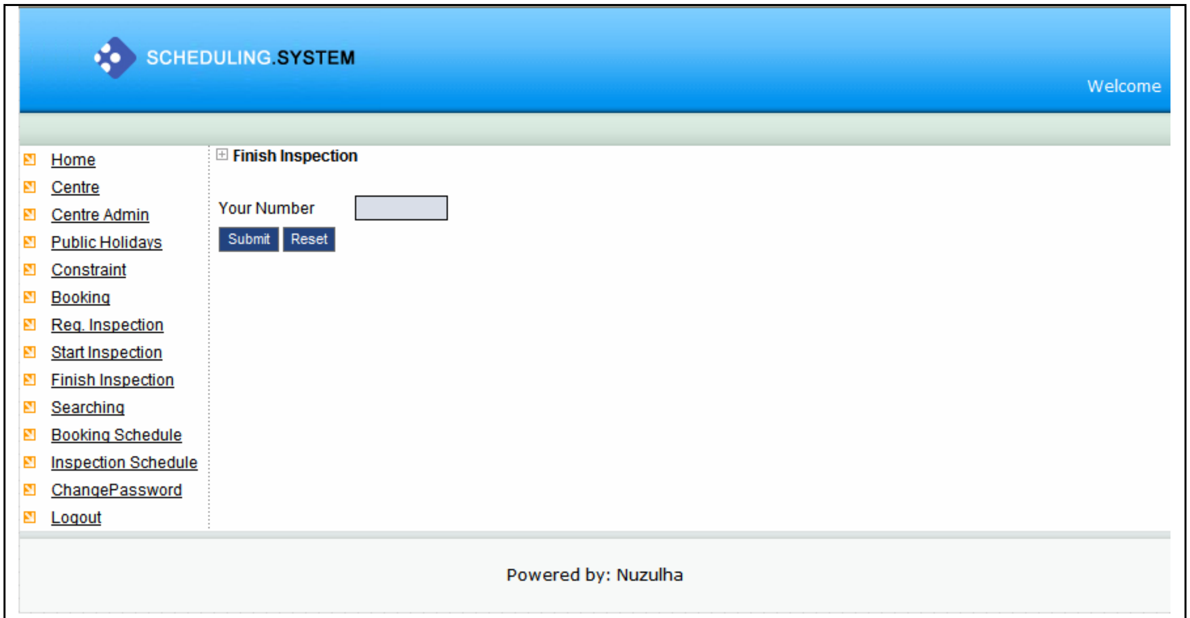


**Figure 7.19**: Menu to Record Finish Inspection.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

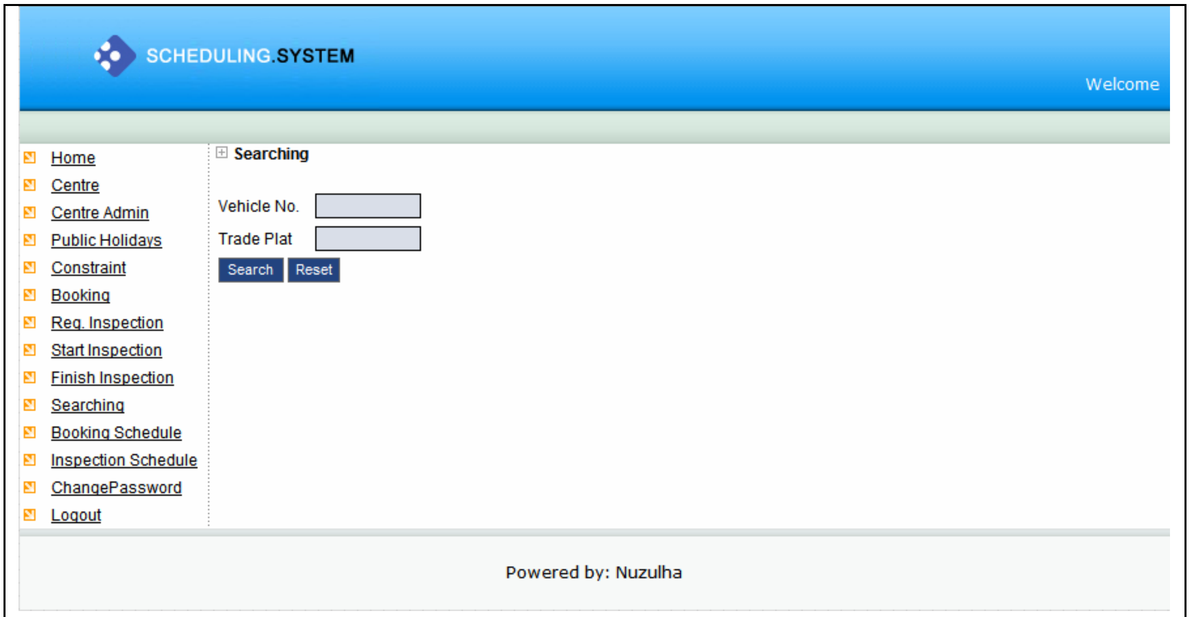


**Figure 7.20**: Menu for Searching.

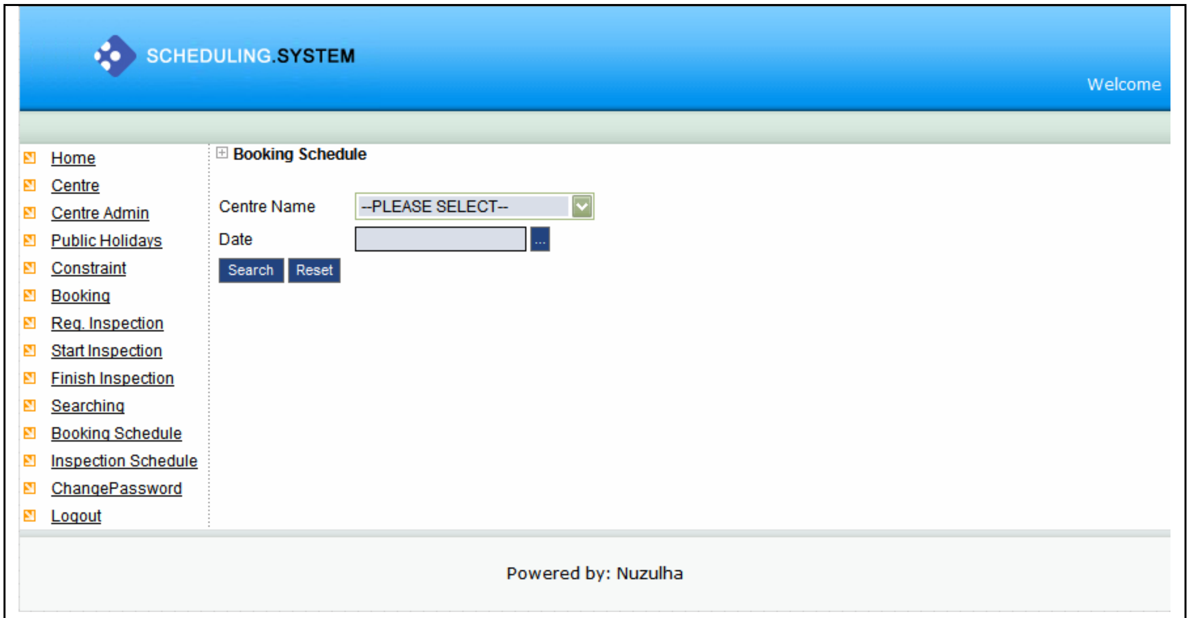


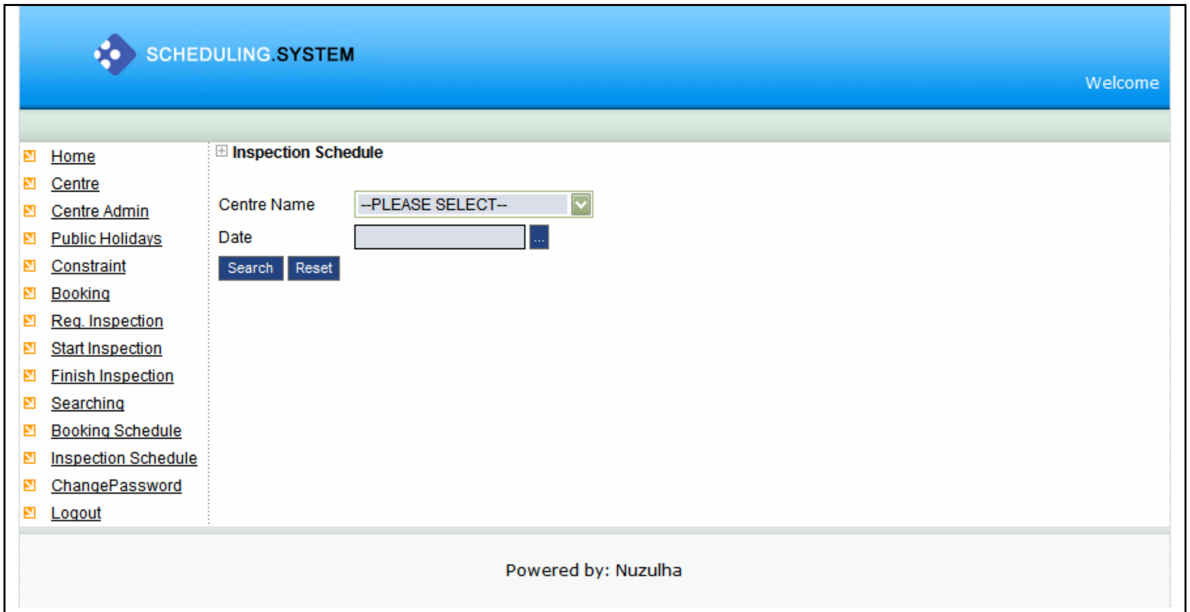
**Figure 7.21**: Menu to View Booking Schedule.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 7.10 Menu 11: Inspection Schedule

This menu is to view the inspection schedule for the appropriate date. Refer to **Figure 7.22**. High level admin, normal admin and clerk are capable to view the inspection schedule of vehicle inspection at the centre. The detail of information required to view inspection schedule is described below:

- Centre Name
- Date of Inspection



Centre Name --PLEASE SELECT--

Date

**Figure 7.22**: Menu to View Inspection Schedule.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 7.11 Menu 12: Change Password

This menu is to change the default password for the made for the super admin, admin and clerk. Please refer to **Figure 7.23**. High level admin, normal admin and clerk are capable to change their password for their own accounts. The detail of information required to change password is described below:

- Old Password
- New Password

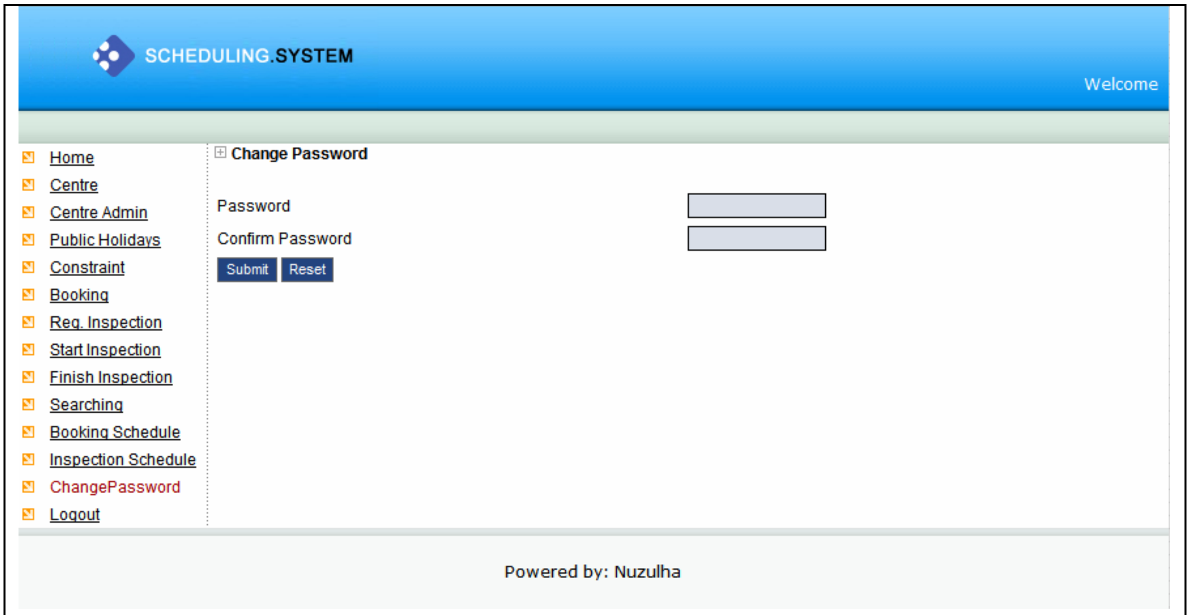


**Figure 7.23**: Menu to Change Password.

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidPDF.com

# CHAPTER 8

# CONCLUSION AND DISCUSSION

## 8.1 Introduction

This chapter states the conclusion from the discussion of the main finding of this study. It summarizes the overall chapter of this report from Chapter 1 toward Chapter 7. Generally, this chapter has discussed the implication and effect from the project implementation. The following topic is about the constraint and challenging during th e project commences and finally the aspiration from the author to the new system proposed.

This report consists of eight main chapters. The eight chapters are Chapter 1 (Introduction), Chapter 2 (Literature Review), Chapter 3 (Methodology), Chapter 4 (Experimental Data and Analysis), Chapter 5 (System Architecture and Design), Chapter 6 (Hardware and Software Requirement), Chapter 7 (System Implementation), and Chapter 8 (Discussion and Conclusion).

Chapter 1 explained about background of the project, statement of project, scope of project, objective of project, the importance of project and chapter summary. In writing a good objective and scope of the project, guidance given from t he project leader is very helpful to ensure the objective have the good quality. A well define scope of project is a key to ensure the success of the project. Following of this is about the case study in vehicle inspection area, with the statement of probl em in the current process.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

The literature review which related to the study is then discussed in the Chapter 2. This chapter is divided into 2 main section which the first part is concerning about distributed constraints satisfaction problem and its techniques that applied in that area. The second part is discussed about software agent and its type in the software environment.

Next, in the Chapter 3 it discussed about the methodology that are been used in the software development. In this chapter, the RUP approach is choose as a methodology for software development. The discussion is not only focusing on the workflow of RUP but also elaborate the detail of its phases such as inception, elaboration, construction and testing phase that are applied in this research. This chapter also discussed about Unified Modeling Language (UML) the software development tools, which includes types of diagrams and notation.

For the purpose of this research, Computerized Vehicle Inspection Centre (PUSPAKOM) has been selected as a real case study. The purpose of selecting PUSPAKOM, is due to the nature of its operation, having many constraints such as capacity of vehicle being inspect for every time slot, processing time, due time, waiting time, release time, availability of working day. The detail explanation of the data collection, the experiment and the result from the analysis are discussed in the Chapter 4.

The system architecture and design are thoroughly discussed in Chapter 5. All the process is illustrated in this chapter which included diagrams in UML notation that are use case diagram, sequence diagram, activity diagram and class diagram. The physical and conceptual design of the to-be system is also discussed in this chapter.

In Chapter 6, the hardware and software requirement are discussed. This includes its specification and installation.

For Chapter 7 it is concerning about the system implementation which includes the code design and the implementation of the system. A few snapshot of the system are showed in this chapter.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

Finally, chapter 8 provides conclusion and discussion related to the project. The detail the project constraints and challenges are also highlight in this chapter.

## 8.2 Achievements

Recently, several researchers have applied agent technolog y to such problems especially in manufacturing and business process control. Because of the widely spread lack of theoretically foundation for this kind of collaborative problem solving, the constraint satisfaction/optimization community has recently exte nded the theoretically well-founded constraint satisfaction problem (CSP) model to a distributed constraint satisfaction problems (DCSP). In order to have more effective in appointment scheduling, we introduce a new approach which incorporated constraint p roblem and agent based theory. In this approach variables are connected by constraints that define the constraints network among the agents. As a result, the search algorithm for solving these problems is a distributed algorithm. The algorithm is run by a gents that communicate by sending and receiving messages. In general, messages contain information about assignments of values to variables and refutations of assignments, by agents that have no compatible assignments, to their own variables.

## 8.3 Constraints

The key constraint is time. Time is very crucial to ensure this project can be achieved. The time given to complete the project is around 1 year and deliverables of this project is conceptual design of the proposed system.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## 8.4 Challenges

Completion of the project milestone was one of the challenges. This task was difficult to achieve mainly due to the requirement from the user that needs to be fulfilled.

The challenge was to get real data from the user on the current v ehicle inspection services and business process. This process takes about 2 month that was to gather the original data during the inspection. After the data collection, one of the challenges was to extract and analyze the data and placed it into mathematic al model. This was quite a difficult process because there was a lot of data to be analyzed.

Another challenge was to understand the technique that would be applied in the new proposed system. Then to model the both AS -IS business process and TO-BE business process using UML diagram. By referencing books on UML and attending workshop, the author managed to overcome the challenges to model the data.

## 8.5 Aspirations

This project aspired to achieved intended objectives agrees earlier in Chapter 1. As a whole, ASRS system has delivered all functionalities stated in the user requirements.

Beside that, other aspiration of this project is to ensure the user is satisfied with the user requirements. By development of this new system, hopefully it can give variou s benefits and make it possible to improve system operation.

Hopefully, this idea of development a new automatic system in vehicle inspection can give various benefits and make it possible to improve system operation, system reliability and to achieve cus tomer and organization satisfactions.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

## REFERENCES

Adrian, P and Boi, F. [2004]. A Distributed Complete Method for Multi -Agent Constraint Optimization. Technical report EPFL/IC/2006/65. 1 -15.

Ahlem, B. H., Xavier, D. and Tu, B. H. [2004]. Agent -Based Approach to Dynamic Meeting Scheduling Problems. International Conference on Autonomous Agents. Proceedings of The Third International Joint Conference on Autonomous Agents and Multiagent Systems. AAMAS' 04, Jul 19 -23, 2004, New York, New York, USA. Vol 3, 1130-1137.

Bruno, J., E.G. Coffman Jr. and R. Sethi [1974]. Scheduling Independent Tasks To Reduce Mean Finishing Time. Communications of The ACM Jul 1974. Vol 17, No 7, 382-384.

Butler W. Lampson [1968]. A Scheduling Philosophy for Multiprocessing Syste ms. Communications of The ACM May 1968. Vol 11, No 5, 347 -360.

Dennis, Y. [1967]. Scheduling A Vehicle Between An Origin And A Destination To Maximize Traveler Satisfaction. Proceedings A.C.M. National Meeting 1967, 233 -245.

Georg, R. [2001]. Distributed Constraint Satisfaction with Cooperating Asynchronous Solvers. Principles and Practice of Constraint Programming – CP 2001, 7th International Conference. CP 2001Paphos, Cyprus. Nov 26 -Dec 1, 2001. Vol 2239/2001, 777.

Georg, R. [2002]. Integrating Search Objects in Asynchronous Constraint Solving. Principles and Practice of Constraint Programming – CP 2002. 8th International Conference, CP 2002, Ithaca, New York, USA, Sept 9 -13, 2002. Vol 2470/2002, 761.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

Georg, R. [2002]. Object-Oriented Constraint Programming with J.CP. MICAI 2002: Advances in Artificial Intelligence: Second Mexican International Conference on Artificial Intelligence Merida, Yucatan, Mexico, Apr 22 -26, 2002. Vol 2313/2002, 194-203.

Hyukchul, J., Milind, T. and Shriniwas K. [2001]. Argumentation as Distributed Constraint Satisfaction: Applications and Results. AGENTS' 01, May 28 -Jun 1, 2001, Montreal, Quebec, Canada. 324-331.

John, K. O. [1982]. Scheduling Techniques for Concurrent Systems. IEEE 1982, 22 -30.

Jun, JB, Jacobson, SH and Swisher, JR [1999]. Application of Discrete-Even Simulation In Health Care Clinics: A Survey. Journal of The Research Society (1999) 50, 109 -123.

Hannebauer, M. [2003]. How to Model and Verify Concurrent Algorithms for Distributed CSPs. Principles and Practice of Constraint Programming – CP 2000. 6th International Conference, CP 2000. Singapore, Sep 2000. 510-514.

Hirayama, K. and Yokoo, M. [1997]. Distributed Partial Constraint Satisfaction Problem. Principles and Practice of Constraint Programming 1997. 22 2-236.

Hsioa-Lan Fang [1994]. Genetic Algorithm in Timetabling and Scheduling. Ph. D. Thesis, Department of Artificial Intelligence, University of Edinburgh.

Markus, H. and Sebastian, M. [2001]. Distributed Constraint Optimization for Medical Appointment Scheduling. AGENTS' 01, May 28-June 1, Montreal, Quebec, Canada. 139-140.

Paul, S., Jay, M., Wei-Min, S. and Milind, T. [2003]. Are Multiagent Algorithms Relevant for Real Hardware? A Case Study of Distributed Constraint Algorithms.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

Proceedings of The 2003 ACM Symposium on Applied Computing. SAC 2003 Melbourne, Florida, USA. 38-44.

Pragnesh, J. M., Wei-Min, S., Milind, T. and Makoto, Y. [2003]. ADOPT: Asynchronous Distributed Constraint Optimization with Quality Guarantees. 1 -33.

Roberto, D. and Teresa, H. [1990]. Case Studies in Fleet Operation Modeling: An Application of AI Scheduling Techniques. International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems. Proceedings of The Third International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Charleston, South Carolina, United States. Vol 1, 290-299.

Roger, M., Victor, L. and Bryan, H. [2003]. Cooperative Negotiation for Soft Rea -Time Distributed Resource Allocation. International Conference on Autonomous Agents. Proceedings of The Second International Joint Conference on Autonomous Agents and Multiagent Systems, Melbourne, Australia. Session: Role and Resource Allocation in MAS, 576-583.

Roger, M. and Victor, L. [2003]. A Mediation Based Protocol for Distributed Constraint Satisfaction. 1-10.

Thomas, L. A., K. M. Chandy and J. R. Dickson [1974]. A Comparison of List Schedules for Parallel Processing Systems. Communications of The ACM Dec 1974. Vo l 17, No 12, 685-690.

Yokoo, M. [1994]. Weak-commitment Search for Solving Constraint Satisfaction Problems. AAAI-94, 313-318.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

Yokoo, M. and Hirayama, K. [1996]. Distributed Breakout Algorithm for Solving Distributed Constraint Satisfaction Problem. In P roceedings of the Second International Conference on Multiagent Systems. 401 -408.

Yokoo, M. and Hirayama, K. [1998]. Distributed Constraint Satisfaction Algorithm for Complex Local Problems. Third International Conference on Multiagent Systems (ICMAS-98), 372-379.

Yokoo, M., Durfee, E. H. and Ishida, T. [1998]. The Distributed Constraint Satisfaction Problem: Formalization and Algotrithms. IEEE Transactions on Knowledge and Data Engineering Sep/Oct 1998. Vol 10, No 5, 673 -685.

Yokoo, M. and Hirayama, K. [2000]. Algorithms for Distributed Constraint Satisfaction: A Review. Autonomous Agents an d Multi-Agent Systems 2000. Vol 3, No 1, 198-212.

Sun, Ron, [2006]. "Cognition and Multi -Agent Interaction". Cambridge University Press.

Attiya, Hagit and Welch, Jennifer (2004). *Distributed Computing: Fundamentals, Simulations, and Advanced Topics*. Wiley-Interscience.

Lynch, Nancy A [1997]. *Distributed Computing*. Morgan Kaufmann.

Tel, Gerard [1994]. *Introduction to Distributed Algorithms*. Cambridge University Press.

Dechter, Rina [2003]. *Constraint processing*. Morgan Kaufmann.

Apt, Krzysztof [2003]. *Principles of constraint programming*. Cambridge University Press.

Marriot, Kim; Peter J. Stuckey [1998]. *Programming with constraints: An introduction*. MIT Press.

Frühwirth, Thom; Slim Abdennadher [2003]. *Essentials of constraint programming*. Springer.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com

Jaffar, Joxan; Michael J. Maher [1994]. "Constraint logic programming: a survey". *Journal of logic programming* **19/20**: 503-581..

Blazewicz, J., Ecker, K.H., Pesch, E., Schmidt, G. und J. Weglarz, [2001].Scheduling Computer and Manufacturing Processes, Berlin (Springer) .

Herrmann, Jeffrey W., editor, [2006]. *Handbook of Production Scheduling,* Springer, New York.

McKay, K.N., and Wiers, V.C.S., [2004]. *Practical Production Control: a Survival Guide for Planners and Schedulers,* J. Ross Publishing, Boca Raton, Florida. Co-published with APICS.

Pinedo, Michael L. [2005]. *Planning and Scheduling in Manufacturing and Services,* Springer, New York.

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidPDF.com