**CHAPTER 1**

**PROJECT OVERVIEW**

**INTRODUCTION**

In the modern world, where parking-space has become a very big problem and in the era of miniaturization, it is become a very crucial necessity to avoid the wastage of space in modern, big companies and apartments etc. In places where more than 100 cars need to be parked, this system proves to be useful in reducing wastage of space. This Automatic Multi-Level Car Parking System enables the parking of vehicles, floor after floor and thus reducing the space used. Here any number of cars can be parked according to the requirement. This makes the system modernized and even a space-saving one.

Automatic Multi-Level Car Parking System provides an automatic parking mechanism which works as follow -

When any vehicle enters the parking facility it gets detected using some Infra-red sensors and gate is opened for vehicle entry and Vehicle Counter is incremented.

One Bi-Color Led which blinks Red if lift is busy or Green if lift is free and on ground floor ready to serve the next vehicle. Lift is equiped with multiple sensors for automatic operations and one button for returning condition.

On Green Signal vehicle must moved into the lift. When Lift detects the vechile placed in proper orientation, it starts moving and stops on the floor where slots for parking is available. Space Availablity on a particular floor is maintained on sensor basis. After the parking is done user has to  press a button on lift to come back down on ground floor.

Signal for a particular event is indicated timely by some Leds and a beeper. Parking System also include a LCD to display Important messages to user.

**Features**

* Car Counting System using Infra-Red Sensors.
* Automatic Door Opening using IR Sensors.
* Sensor based Lift System for moving vehicle to the right floor.
* Easy to use System.
* Multiple Floor to reduce space.

**CD Content**

* Project Report in Word and PDF format
* Circuit Diagram
* PCB Layout
* Micrcontroller Program in C-language
* Hex File(Binary Code)
* Software Tools and IDE
* Datasheet of all components and ICs used in project
* Video Introduction of Project
* Details of Assembling Guide and Working of Project
* **DISADVANTAGES OF CAR PARKING:-**
* Drivers who use multi parking facilities, sometimes known as parking garages, often enjoy a number of benefits the structures provide. Despite the ability of the garages to house a large number of cars, multi-story parking facilities also carry a number of distinct disadvantages that arise from their tall, enclosed and often dimly lit nature
* **1.Deterioration and Maintenance**
* Multi-story parking facilities support hundreds of thousands of pounds of vehicles, people and equipment every day. Because the garages support very large amounts of weight and loads that constantly change, the structures quickly deteriorate in the absence of constant maintenance activity. In addition, according to Canada’s National Resource Council, changing weather and environmental conditions can deteriorate a garage’s steel support structure, creating an unsafe environment for garage users. A number of corrosion inhibitors can help delay processes that eat away at the structure’s integrity, according to the National Resource Council, but constant maintenance and upkeep must include anti-corrosion measures to keep multi-story parking facilities structurally sound.

1. **2.Parking Angle Considerations**

Because many drivers of varying levels of skills and experience drive in, around and out of parking garages every day, designers must pay special attention to the configuration of parking spaces within the structures. In a municipal parking garage presentation prepared by architects Sakri and Khairuddin, the designers noted that two-way traffic flow in a multi-story garage presents a number of parking challenges for drivers and designers. Parallel parking, for example, creates an inefficient use of limited space, while straight parking spaces make parking difficult for some drivers. Other options, like angled parking, do not work well with a two-way traffic flow and can only work well in garages with separate entrance and exit openings.

1. **3.Lighting**

While most parking lots open at night, multi-story or otherwise, require some form of lighting, the multi-story nature of parking garages creates a need for numerous lights throughout the structure. In addition, because the inside of the structure may remain dark even during the day, many of these lights must run at all times. This arrangement can create high energy bills for garage owners and may require frequent lighting maintenance to replace broken or burned-out bulbs

**CHAPTER NO.2**

**SYSTEM ANALYSIS**

**2.1 Software requirement analysis:**

The requirements gathering process was intensified and focused specially on software. To understand the nature of the problem to be built, one must understand the information domain for the software, as well as required function. Behavior, performance and interface requirements of both the system and the software were documented and received with the company.

**Design:**

Software design is actually a multi – step process that mainly focuses on four distinct attributes of a program. Data structures, software architecture, interface representation and procedural detail. The design process translates requirements into a design document and becomes a part of the software configuration.

**Code generation:**

In this process design is translated into machine – readable form. As the design process was performed in a detailed manner code generation was accomplished easily.

**Testing:**

Once code has been generated, program testing begins. The testing process focuses on the logical internals of the software, ensuring that all functional externals are tested to uncover errors and ensure that defined input will produce actual result that agree with required results.

## 2.2 Software and Hardware Requirement Specifications

The softwares which are used to develop this system are quite familiar and not new and complex to the user and are also easily available, and the software has been planned in such a way that it can be implemented on any operating system and can easily interact with any of the databases which have been provided in the software.

**2.2.1 HARDWARE:**

* INTEL CORE i3 2.40 GHz



**Figure no 1**

* 2GB RAM



**Figure no 2**

* 500 Gb disk storage at the server end



**Figure no 3**

**2.2.2 SOFTWARE:**

* Language: Java 1.8.0
* Database: Ms-SQL Server 2005 Express Edition, My-SQL Server 5.5
* O.S. : Platform-Independent (Windows, Linux, Solaris)
* JDK 1.8 - NetBeans 8.0.2
* Database - MySQL Database Server 5.0

**CHAPTER 3**

**TECHNOLOGY USED**

**1.1 JAVA**

* Java is a **programming language** and a **platform**.
* Java is a high level, robust, secured and object-oriented programming language.
* PHP supports many databases (MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL)
* Java is open source Software.

**USE JAVA :**

* Desktop Applications such as acrobat reader, media player, antivirus etc.
* Web Applications such as irctc.co.in, javatpoint.com etc.
* Enterprise Applications such as banking applications.
* Mobile
* Embedded System
* Smart Card
* Robotics
* Games etc.

**FEATURE OF JAVA:**

* Simple
* Object-Oriented
* Portable
* Platform independent
* Secured
* Robust
* Architecture neutral
* Dynamic
* Interpreted
* High Performance
* Multithreaded
* Distributed

Java Features

**Figure 1: Features of Java**

## TYPES OF JAVA APPLICATIONS

#### Standalone Application :It is also known as desktop application or window-based application.

#### An application that we need to install on every machine such as media player, antivirus etc. AWT and Swing are used in java for creating standalone applications.

#### Web Application :An application that runs on the server side and creates dynamic page, is called web application. Currently, servlet, jsp, struts, jsf etc. technologies are used for creating web applications in java.

#### Enterprise Application : An application that is distributed in nature, such as banking applications etc.

#### It has the advantage of high level security, load balancing and clustering. In java, EJB is used for creating enterprise applications.

#### Mobile Application :An application that is created for mobile devices. Currently Android and Java ME are used for creating mobile applications.

## JAVA PLATFORMS / EDITIONS

#### Java SE (Java Standard Edition) :It is a java programming platform. It includes Java programming APIs such as java.lang, java.io, java.net, java.util, java.sql, java.math etc. It includes core topics like OOPs, String, Regex, Exception, Inner classes, Multithreading, I/O Stream, Networking, AWT, Swing, Reflection, Collection etc.

#### Java EE (Java Enterprise Edition) : It is an enterprise platform which is mainly used to develop web and enterprise applications. It is built on the top of Java SE platform. It includes topics like Servlet, JSP, Web Services, EJB, JPA etc.

#### 3) Java ME (Java Micro Edition) :It is a micro platform which is mainly used to develop mobile applications.

#### 4) JavaFx :It is used to develop rich internet applications. It uses light-weight user interface API.

**2.1 DATABASE USED:**

* **Mysql**

MYSQL is comprehensive, integrated data management and analysis software that enables organizations to reliably manage mission-critical information and confidently run today’s increasingly complex business applications. MYSQL allows companies to gain greater insight from their business information and achieve faster results for a competitive advantage.

The driving force behind MYSQL is the process of integration. With MSSQL 2005 the database developer experiences a paradigm shift. He can now locate his code with reference to its functionality; he can access data in its native formats or build complex systems that are server driven.

**ABOUT MYSQL**

Among the many application development and end user products available with in the MYSQL family there is a common ability to access the database. Whether directly or in directly this is achieved through the Structured Query Language (SQL). These query languages have been useful in developing the software in an efficient way. The Query language has Data Definition Language (DDL), Data Manipulation Language (DML) and Data Control Language (DCL).

**Data definition language** helps in defining various database objects such as Tables and views. This has been useful in defining all the relational tables of the project. It has also been used in creating many views providing for information hiding and data security. Data Definition Language (DDL) also supports alteration of the existing tables.

**Data Manipulation Language** (DML) is the most important of the query Language. This has features for inserting records into the tables, Deleting the existing records from the tables, Updating the existing records in the tables and retrieving the data from one or more tables as and when required.

**Data Control Language** (DCL) is used for controlling data and having access to the databases. This part of the language provides extensive security features in order to safe guard the users information from both unauthorized access and intentional damage. Granting and revoking privileges provide security. Oracle uses have names and passwords and own tables, views and other database objects.

These facilities have eased the job of developing the software to meet the organizational requirements. Many similar programming logic have been coded as functions and procedures and have been used in many areas. For Example retrieving the records, Calculation of various parameters of Pay etc. This has been a big advantage in developing the package.

The facility of exceptions also helped in developing a user-friendly package. Exceptions are raised and a suitable actions are taken when errors occur such as ‘when no data found’, ‘Too many rows’ etc.

The facility of Cursors where used in retrieving records one bye one in an orderly way. This helps in management in viewing the records one after other to get the required information about each employee. This usage of Cursors eliminates many errors such as ‘Too many rows’ etc. Each record can be retrieved and studied for any amount OD time before retrieving the next record. Cursors retrieve the records in ascending order of the values of the primary Key.

**Physical data independence**

You'll now play a little with MySQL's command-line tool, mysql, located in the C:\MySQL\bin\ folder. Open a command prompt window, change to the appropriate directory, and type mysql. The MySQL prompt should then appear:

mysql>

Note that when MySQL is installed on Windows, all local users have full privileges to all databases by default. Although you can skip it for the purposes of the samples in this book, production scenarios require MySQL to be more secure, and you should set an individual password for all MySQL users and remove the default record in the mysql.user table that has Host='localhost' and User='' using the following command:

mysql> USE mysql

mysql> DELETE FROM user WHERE Host='localhost' AND User='root';

mysql> QUIT

The first command tells SQL to use the mysql database, and the second removes the appropriate record from the user table of that database (don't forget the final semicolon!). Lastly you exit the mysql tool using the QUIT command (alternatively, you can use EXIT if you want).

|  |  |  |
| --- | --- | --- |
|  |  | Note that SQL commands, such as DELETE, WHERE, and QUIT, aren't case-sensitive; however, according to convention, we'll capitalize them in this book. Be aware, though, that the names of SQL tables and columns, as well as values themselves, are case-sensitive, and they must use the casing applied at definition. |

Now, you should also add a password for the root user. The root user, also called the admin user, has full privileges to the database, so anyone logged into MySQL as root has the power to change the database however they please (or to make costly mistakes). To change the password, make sure you're in the C:\MySQL\bin\ folder and use another command-line tool, mysqladmin:

> mysqladmin reload

> mysqladmin -u root password your\_password

where your\_password is the password you want to use. Make sure you type this correctly because you won't be asked to confirm it. Once the password is set, you then shut down the mysql server with the following command:

> mysqladmin --user=root --password=your\_password shutdown

Now that you can get the MySQL server up and running securely, you can experiment with some basic database administration issues.

After you've connected successfully, you can disconnect at any time by typing QUIT at the prompt or by pressing Ctrl+D:

mysql> QUIT

Bye

Issuing SQL Commands

Run the following simple command that requests the current date from the server:

mysql> SELECT CURRENT\_DATE;

Note the semicolon that ends the SQL statement. It should produce output something like the following:

+--------------+

| CURRENT\_DATE |

+--------------+

| 20017-6-21 |

+--------------+

A command normally consists of a SQL statement followed by a semi-colon; however, there are cases where the semicolon isn't needed. QUIT, mentioned earlier, is one of them.

When you issue a command, it's sent to the server for execution. Any results are displayed in the form of a grid. The first row typically shows the names of the columns, and results from the query appear below.mysql shows how many rows were returned and how long the query took to execute as a rough idea of server performance.

You've just entered is a single-line command consisting of a single SQL statement, but more complex commands can comprise multiple statements. You can either add all the statements on a single line one after the other or enter them on multiple lines by simply pressing Enter before the typing the final semicolon. mysql indicates what's expected next by changing the prompt to one shown here.

| **MySQL Prompts** | |
| --- | --- |
| **Prompt** | **Meaning** |
| mysql> | Ready for new command |
| -> | Waiting for next line of multiple-line command |
| '> | Waiting for next line, collecting a string that begins with a single quote |
| "> | Waiting for next line, collecting a string that begins with a double quote |

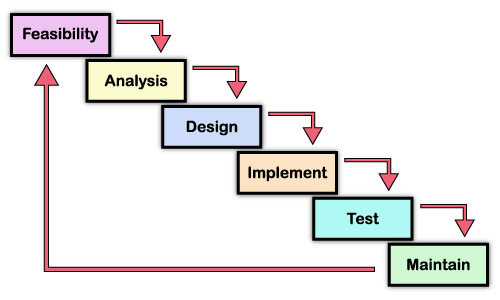
| **Character Data Types** | | |
| --- | --- | --- |
| **Data Types** | **Description** | **Range/Format** |
| CHAR | Fixed-length string | 0–255 characters. |
| VARCHAR | Variable-length string | 0–255 characters. |
| BLOB | Binary Large Object (BLOB) | Binary data 0–65535 bytes long. |
| TINYBLOB | Small BLOB value | Binary data 0–255 bytes long. |
| MEDIUMBLOB | Medium-sized BLOB | Binary data 0–16777215 bytes long. |
| LONGBLOB | Large BLOB value | Binary data 0–4294967295 bytes long. |
| TEXT | Normal-sized text field | 0–65535 bytes. |
| TINYTEXT | Small text field | 0–255 bytes. |
| MEDIUMTEXT | Medium-sized text | 0–16777215 bytes. |
| LONGTEXT | Large text field | 0–4294967295 bytes. |
| ENUM | Enumeration | Column values are assigned one value from a set list. |
| SET | Set value(s) | Column values are assigned zero or more values from a set list. |

| SQL Modifiers | |
| --- | --- |
| **Modifier** | **Description** |
| AUTO\_INCREMENT | Allows a numeric column to be automatically updated when records are added. Useful for creating a unique identification number for each row. |
| DEFAULT value | Specifies the default value for a column. |
| NULL | Specifies that a column may contain undefined, or NULL, values. |
| NOT NULL | Requires that the column must contain a non-NULL value. |
| PRIMARY KEY | Makes the column the primary key. It must also have a NOT NULL modifier. |

##### System Requirement Analysis

Prior to the software development efforts in any type of system it is very essential to understand the requirements of the system and users. A complete specification of the software is the 1st step in the analysis of system. Requirements analysis provides the designer with the representation of function and procedures that can be translated into data, architecture and procedural design.

The goal of requirement analysis is to find out how current system is working and of there are any areas where improvement is necessary and possible.



# Requirement Analysis

Identification of Need

The present system, which is being implemented, is very complex. It is based on Command Line Interface that is very compound and hence it becomes very difficult for a normal user to interact with that software. Only those persons who have complete knowledge about the Java API and DOS can work on this. It becomes very difficult to deal with the same in absence of an expert. It can work with Oracle or mysql. The present system is not User-friendly. The manual system is very expensive and time consuming. The company was facing problems in keeping records of incoming and outgoing messages.

**Conducting interviews**

We interviewed in some of the branches of the company with center head and the other employees to find out the facts behind the problem. As we interviewed in branches with different persons and we found some facts related to their problems. We put almost same questions. So it is difficult to give details of their interview. The question, which we put, was like as:

**The question, which we put with, the head of companies are as follows**

* What database are they using?
* How they manage the records?
* How you they manage in case of any crashes?
* How you they manage different cases of students?
* How much feedback do they get from their students and there records?
* Are you satisfied with the existing system?
* Is the existing system economical to you company?
* What is monthly expenditure of your company?

**The question, which we put with, the employees are as follows:**

* How you manage in the absence of an expert?
* What difficulties you face in managing the system if it hangs?
* Are you satisfied with the existing system?
* What they do when the number of incoming and outgoing messages is high?

##### **FEASIBILITY STUDY**

##### **Technical Feasibility:**

Fundamentally, we are trying to answer the question “Can it actually be built?” To do this we investigated the technologies to be used on the project. For each technology alternative that we accessed we identified the advantages and disadvantages of it. By studying available resources and requirements, we concluded that minimum the application should be made user-friendlier.

**Table 1 some issues to consider when determining technical feasibility**

|  |  |
| --- | --- |
| **Technology Issues** | **Market Issues** |
| * Performance * Ease of learning * Ease of deployment * Ease of support * Operational characteristics * Interoperability with other technologies | * Vendor viability * Alternate sources of the technology, if any |

The similar case exists with hardware. Most of the company is not ready to invest much more in purchasing new hardware peripherals. Therefore keeping their minimum available resources in mind we develop software, which do not require new software other than available software and the hardware requirement. The software, which we develop require following hardware and software.

##### **Economical Feasibility**

When accessing economical feasibility of an implementation alternative the basic question is “does the project make fianacial sense? ”We did this by performing a cost/benefit analysis, which as its name suggests compares, the full/real costs of the application to its full/real financial benefits. The alternatives are evaluated on the basis of net cash flow, the amount by which the benefits exceed the cost, because the primary objective of all investments is to improve overall organizational performance. It was found that the company is unnecessary spending money and in spite spending much money it is not getting satisfaction and truthful results. We also found that the company just wants to upgrade the existing system because they do not want to invest extra money to accept new system.

Keeping all these needs and demands of system in minimum budget we developed new software which will not only lower their budget but also not require much cost to accept it. It also does not need much maintenance. The new system will save the money, which they are investing on getting short codes, thus keeping all these good qualities is an economical feasibility.

**Table 2 Potential costs and benefits of a software project**

|  |  |  |
| --- | --- | --- |
| **Type** | **Potential Costs** | **Potential Benefits** |
| Quantitive | * Hardware/Software upgrades. * Training costs to train developers. * Expected operational costs. | * Reduced operating costs. * Increased revenue from additional sales of the organization services. |
| Qualitative | * Increased employee dissatisfaction from aear of change. | * Better response from customers * Positive public reception |

##### 

##### **Operational Feasibility**

Not only must an application make economical and technical sense, it must also make operational sense. The basic question that you are trying to answer is “Is it possible to maintain and support this application once it is in production?” Building an application is decidedly different than operating it, therefore one need to determine whether or not one can effectively operate and support it.

By carrying various types of interviews and questioning with center head of almost every center we found out their basic technical skills, their requirements, or whether they are ready to adopt new skills. We concluded that most of the companies wanted to adopt system which require not much more complex skills and are not ready to adopt an absolutely new environment, they wanted a system which is quite familiar with them and do not require extra skill to learn and operate it.

Keeping their demand and expectations in mind we developed a system which do not require any extra technical skill and training. It is developed using such environment, which are quite familiar to most of the people concerned with the system. The new system will prove easy to operate because it is developed in such a way so that it will prove user friendly. User will find it quite familiar and easy to operate because it is carrying necessary help guide and comment when required so that user does not have to face any difficulties, therefore it will prove feasible in respect to operate.

**Table 3 Issues to consider when determining the operational feasibility of a project.**

|  |  |
| --- | --- |
| Operations Issues | Support Issues |
| * What tools are needed to support operations? * What skills will operators need to be trained in? | * What documentation will users be given? * How will change be managed? |

**CHAPTER 4.**

**SCREEN SHOTS**

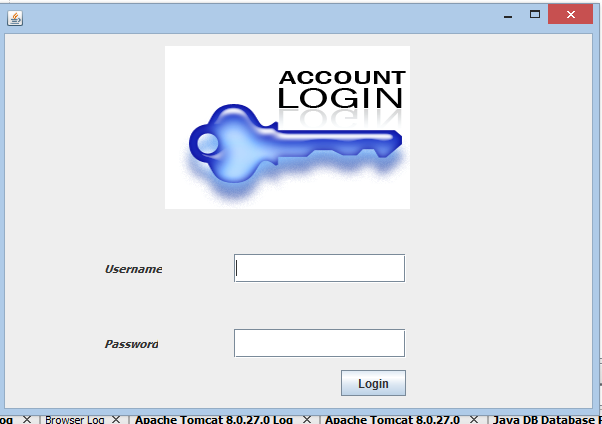


FIGURE 1 LOGIN



FIGURE 2 SEARCH VEHICLE

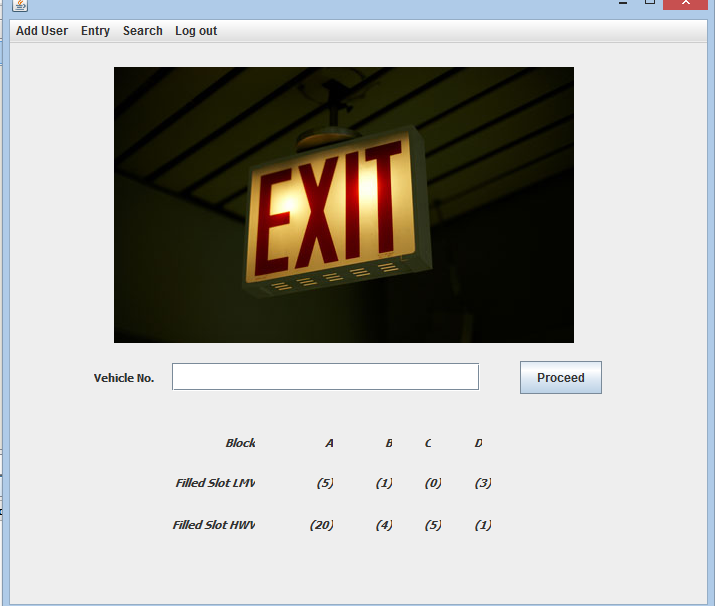


FIGURE 3

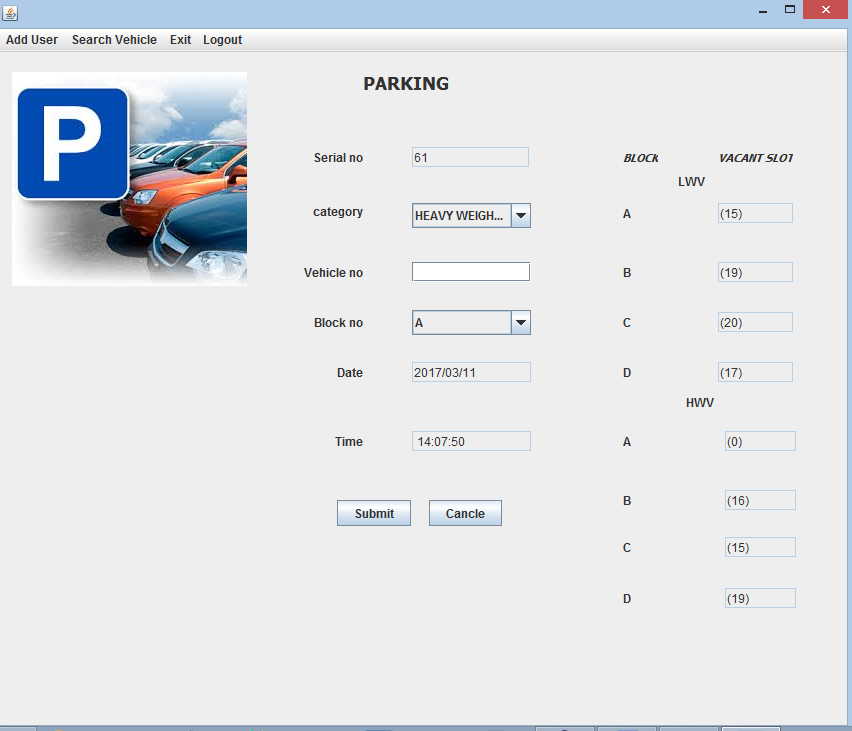


FIGURE 4 ADD VEHICLE

# CHAPTER NO.5

# TESTING PROCEDURES

**5.1 White – Box testing**:

White box testing sometimes called glass – box testing, is a test case design methods that use the control structure the procedural design to derive test case.

1. Guarantee that all independents paths within a module have been excercises at least one.
2. Exercises all logical decisions on their true and false sides.
3. Execute all loops at their boundaries.
4. Exercises internal data sturucture ensures the validity.

The reason behind spending time & energy about logical minutiae.

* + Logic errors and incorrect assumptions and invesely proportion to the probability that a program path will be executed.
  + The logical flow of a program is sometimes contributive meaning that our unconscious assumptions about flow of control and data may lead us to make design errors that are uncoverd.

Typographical errors are random:- when a program is translates into a programming language. Source code is likely that some typing errors will occur.

Using white box testing methods, we drive the following case:

Firstly we checked that all the modules or procedures, which are defined in the program, are used at least once or not. If any procedure or subroutine was not in use, then we removed that procedure from the program.

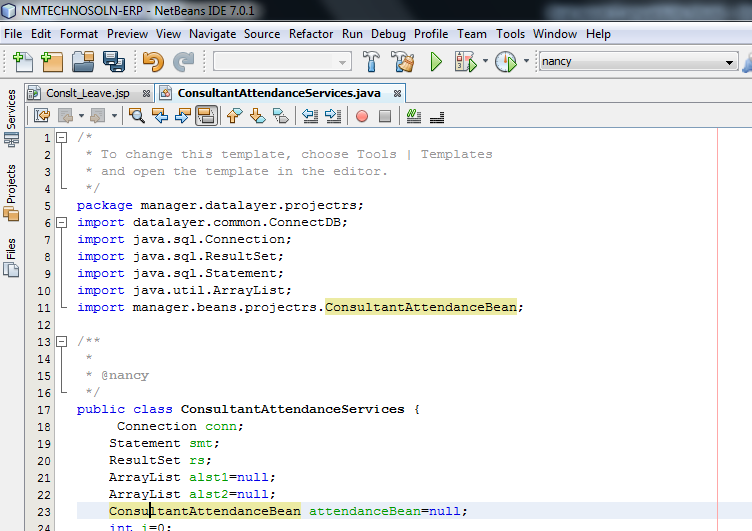
After checking all modules used in the program we checked all the logical decisions used in the conditional statement and loops, if any conditional statement or any looping condition was not true than we modified the condition and then rechecked. These processes were repeated until all the conditions were true.At last all the data structure in terms of public and private variables used in the programs were checked, whether all the variables were used properly, if any variables were not used then we omitted those variables from the program.

**Basis Path Testing**

Basis path testing is a white box testing technique. The basis path methods enable to derive a logical complexity measure of procedureal design and use this measure as a guide for defining a basis set of execution paths.

**Cyclomatic Complexity**

Cyclomatic complexity is software metric that provides a quantitative measure of the logical complexity of a program. When used in the context of the basis path testing method, the value computed for cyclomatic complexity defines the number of independent paths in the basis set of a program and provides us with an upper bound for the number of tests that must be conducted to ensure that all statements have been executed at least once.



**Figure no 1**

**5.2 Black Box Testing:**

Black box testing, also called behavioural testing. Focuses on the functional requirements of the software. Black – box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Black – Box testing is not an alternative to white – Box technique. Rather than it is a complementary approach that is likely to uncover a different of errors than white box method.

Black – Box testing attempts to find errors in the following categories:

1. Incorrect or missing functions
2. Interface errors
3. Errors in date structures
4. Behavior of performance errors
5. Initialization and termination errors.

Unlike white box testing, black box testing tends to be applied during later stages of testing, because black – box testing purposely disregards control structure attention to focused on the information domain. Tests are designed to answer the following questions.

By applying black – box techniques, we derive a list of test cases that satisfy the folowing criteria.

* 1. Test cases that reduce, by a count that is greater than one, the number of additional test cases that must designed to achieve reasonable testing.
  2. Test cases that tell us something about the presence or absence of work of errors.

### Graph -Based Testing Method

The first step in black – box testing is to understand the objects that are modeled in software and the relationship that connect these objects. Once this has been accomplished, the next step is to define a series of tests that verify, “All objects have the expected relationship to another”.

To accomplish these steps, the software engineer begins by creating a graph, a collection of needs that represent object links that represents the relationship between objects.

Directed Link

Link weight

Directed Link

Directed Link

Node weight

(Value)

**(A)**

Is represented as

(Generation time < 1.0 sec)

Menu Select generate Link

Containss

Attributes:

Start dimension: default setting or preferences

Background color: white

Text color : default color or preferences

**(B)**

##### Validation Checks

In the process of validation check, to trap the errors which occurs during the runtime, which results the problem like system hanging and program interruption etc. we provide it own error messages whenever thse occurs to avoid these problem.

Some of the error types with their messages are as follows:

Sl. No. Error type

1. Incomplete details
2. Database linkage error

3. wrong username and password

4. Unsupported Comm Operation

5. Database is not linking

**CHAPTER 6**

**CONCLUSION**

The Development of Smart Parking System met all of my performance specifications. I am able to successfully guide multiple cars to their designated parking spots using the interfaces capabilities. I achieved my goal of providing a fully and interactive proof of concept design to show people the power and usefulness of this system. The system communicates with the Java interfaces and Microsoft SQL Server database through a connection on the system. My idea incorporated all the parts of the software system. When integrated properly, my initial software design is a viable way to implement this type of system. The floor interface guiding multiple cars appropriately. The other interface effectively tracked parked vehicles and moving traffic within the garage. Also, I designed a JAVA Interfaces that is intuitive enough for a first time user to navigate with ease. The program accommodates for physical disabilities, recommends an optimal spot, and also prints return directions on the parking ticket. There were a number of challenges I had overcame in the design and testing of the Development of Smart Parking System. The first was the actual setup connection between JAVA interfaces and Microsoft SQL Server database of the model parking garage. The challenge I faced involved connection between Microsoft SQL Server database and the JAVA program. My original plan was to send the parking spot information in different 57 interfaces from Microsoft SQL Server database to the JAVA interfaces. I wanted to send the status of each parking spot (available or unavailable) as a separate interface. I was able to establish consistent and reliable networking results. The next step for the Development of Smart Parking System is to sell the concept to a leisure mall that would be willing to try it out on a small scale area. A leisure mall like Suria KLCC would be a perfect test leisure mall for the Development of Smart Parking System. My database can be easily modified for Suria KLCC. I would then have to get a navigation system manufacturer to incorporate the software and interface into their existing system. The, the Development of Smart Parking System will be on its way to becoming a solution to the parking schema. 5.2 Recommendation for future project There are a number of improvements and modifications that can be made to my design to increase real world practicality and functionality. The symbols used need to have the ability to differentiate between pedestrians and cars. My Development of Smart Parking System garage is so simple that adding more interfaces was unnecessary. With only twelve parking spots in my system, more than five cars moving through the garage at one time would simply produce congestion. Other improvements would be additional guidance devices such as a return interface with audio or light guidance. Upon the return, either audio playback or a series of LEDs would guide the driver to their vehicle. This was excluded from my original design system due to time constraints. Such an interface could also be integrated with the elevator so that the driver might be taken directly to the appropriate floor. 58 Any types of sensors that could sense the car from about six feet away could be implemented. Six feet is a more realistic distance between a parked car and its parking meter. Also, more powerful transmitters and receivers to increase the range of operation could also be implemented. Ideally, the parking attendance at his or her station should be able to receive data up to several miles away from the transmitter at the parking space. Finally, the concept of the open space locator could be expanded to include multiple parking meters from multiple parking lots. This system can clearly be applied to a number of different parking environments: large parking lots, underground parking facilities, stadium parking, airport parking, etc. The system can be easily networked to monitor and control any number of different parking facilities. Furthermore, this information can be made available online for convenient access. Not only is this system useful and applicable to parking facilities, it can be easily modified to accommodate the needs of hospitals, resorts or conference centers. A similar guidance system could be used to direct patients and doctors to different wings, hotel guests to various attractions, or clients to the proper meeting room. The Development of Smart Parking system is beneficial wherever frequent searching processes are involved. For example, it can be used anywhere from locating specific products within large industrial warehouses to finding a book at your local library. 5.3 Commercialization Potential The Development of Smart Parking System business development programs will be leveraged to support complex business planning activities. In this system application research will involve studies of the market and end-users, as well as business planning for venture capital, business organization, and product design. 59 The Center will provide seed grants where deployment and commercialization are promising. The Center will connect research faculty with private-sector partners and support opportunities to prepare energy-efficiency solutions for the marketplace. This system has been developed to provide parking attendance with parking information on how to get to. The parking attendance is committed to working towards a sustainable parking floors and the range of parking choices. The parking plan brings together existing and new initiatives which offer a better choice in the way car’s owner can parking. This system also easy to use guide providing parking attendance with high quality parking information. This system also is a new service that tracks cars in real time throughout the mall. It means parking attendance can now find out the arrival time of the next car at every parking spot. She/ He can access this system information via database. The ticket parking is available to the entire car that had been parking in that time. Costing just RM1.00 per hour for one car gives unlimited parking spots parking throughout the mall. With the demand for car parking ever increasing in the entire mall has invested, in partnership with the Universiti Malaysia Perlis, to develop the web page car parking scheme. Car parking is when car’s owner parking a car to work, home or shopping. This web page has many users and I can search for a match just from the Universiti Malaysia Perlis.unimapcarsparking.com is a web enabled matching service operated by leading car parking provider. Registration is free and easy to complete. There is a useful section on the website and no personal data is released to other car parking members other than a username and the journey details you’ve added. Proper planning needs to be done carefully and suspiciously in order to make sure the business that has been planned will be a profitable and long lived business. The opportunity to develop and implement a well-defined business strategy is very valuable to the business owner, their customer and as well as the community. Business plan reviews 60 visions and strategic focus as adding value to the target market segments, the small business and also the system users in the local. I had managed to convince administrator and users of websites, I will move one step forward to commercialize my Development of Smart Parking System and try to maintain my product. This application is suitable for parking attendance to manage parking spots in the building.

# CHAPTER NO.7

**BIBLIOGRAPHY**

**Books/Sites Reffered**

1. Complete Reference Herbert Schildt

2. flazx-Flex on Java

3. [www.roseindia.com](http://www.roseindia.com)

4. www.java2s.com

5.www.javatpoint.com