**Table of Contents**

1. Introduction

1.1 Purpose

1.2 Scope

1.3 Definitions, Acronyms, and Abbreviations 6

1.4 References 6

1.5 Overview

2. The Overall Description 7

2.1 Product Perspective 7

2.1.1 System Interfaces 8

2.1.2 Interfaces 8

2.1.3 Hardware Interfaces 8

2.1.4 Software Interfaces 8

2.1.5 Communications Interfaces 8

2.1.6 Memory Constraints 9

2.1.7 Operations 9

2.1.8 Site Adaptation Requirements 9

2.2 Product Functions 9

2.3 User Characteristics 9

2.4 Constraints 9

2.5 Assumptions and Dependencies

2.6 Apportioning of Requirements

3. Specific Requirements

3.1 External interfaces

3.2 Functions 11

3.3 Performance Requirements

3.4 Logical Database Requirements 12

3.5 Design Constraints

3.5.1 Standards Compliance

3.6 Software System Attributes

3.6.1 Reliability

3.6.2 Availability

3.6.3 Security

3.6.4 Maintainability

3.6.5 Portability

3.7 Organizing the Specific Requirements 13

3.7.1 System Mode 13

3.7.2 User Class 13

3.7.3 Objects 13

3.7.4 Feature 13

3.7.5 Stimulus 13

3.7.6 Response 13

3.7.7 Functional Hierarchy 13

4. Change Management Process 14

5. Document Approvals 14

**1. INTRODUCTION**

Time is Everything. As we know people waste a lot of time in finding parking place. So there is a need of a system which will immediately people if there is free space to park.

**1.1 Purpose**

The purpose of this software requirements specification is to verify that all the specifications are correct and are verified. This document also serves to ensure that the software is traceable throughout its software development life cycle.

Developing **Parking Allocation Software** helps people in saving their precious time by showing them free space if available to park their vehicle.

**1.2 Scope**

The scope of this project is everywhere since there is need of Parking everywhere. Our project helps every society , apartment, mall, complex, stadium, hospitals etc.

Some common Features are:

* Facility to show available space available.
* People can know how many time you have come at that place..
* People can book by calling at Parking office(for emergency purpose).
* Facility to maintain records vehicle came in particular day.
* Facility to keep information about car and its owner.
* Facility to know for how much time a particular vehicle stayed there.
* Facility to give feedback.

**1.3 DEFINITIONS, ACRONYMS AND ABBREVIATIONS**

 **Administrator:** Administrator is the person who operates the software and enter the info about the car its owner.

 **User:** General public uses it as parking is needed by all.

 **Actor:** An actor is someone or something outside the system that interacts

with the system (e.g., an end-user or system administrator might be actors or

another system with which the system interacts might also be an actor).

 **Attribute:** In data modeling, specific items of data that can be collected for a

class. Any software performing either the final stage in a process, or a task not

apparent to the user. A common usage is a verification of permissions based

on user-class and security. The end-user accesses the application and pages

are retrieved based on user-class permissions.

 **Client-server:** A common form of distributed system in which software is

split between server tasks and client tasks. A client sends requests to a server

asking for information or action and the server responds.

 **End User:** The ultimate consumer of a product, especially the one for whom

the product has been designed. End-users for a patient management

application include physicians, nurses, epidemiologists, outreach workers and

health care providers.

 **Functional Requirement:** A description of what a system should be able to

do–a function it should perform.

 **Non-functional Requirement:** Software design requirements related to the

efficiency, reliability, portability, and usability of the system. Also known as

supplementary requirement.

 **Requirement:** A requirement describes a condition or capability to which a

system must conform; either derived directly from user needs, or stated in a

contract, standard, specification, or other formally imposed document. A

requirement is a desired feature, property, or function to be met by the

application.

 **Software Requirement:** A software requirement is a specification of an

externally observable behavior of the system; for example, inputs to the

system, outputs from the system, functions of the system, attributes of the

system, or attributes of the system environment.

29

 **Software Requirement Specification:** A project artifact that defines the

complete system requirements through use cases and supplementary

specifications.

 **Stakeholder Request:** A stakeholder request is a request of any type from a

stakeholder, including a change request, enhancement request, request for a

requirement change, or defect.

 **Use-case Specification:** The text explanation or description of a use case.

 **User Interface:** The programming that controls a display for the user (usually

on a computer monitor) and that allows the end user to interact with the

system with commands and mechanisms to control system operation and input

data.

 **JAR file:** Deployed version of the project.

 **System DSN:** Data source stores information about how to connect to the

indicated data provider. A user data source can be used on the current

machine.

**1.4 References**

**Other Documents and Resource:-**

**1.5 OVERVIEW**

The remaining SRS will include 5 sections: -

1).**Overall Description** will describe major components of the system, interconnection and externalinterfaces.

2).**Specific Requirements** will describe the functions of actors, their role in the system andconstraints.

3).**Change Management** **Process** will identify the change management process to be used to identify, log, evaluate, and update the SRS to reflect changes in project scope and requirements*.*

4).**Document Approvals** will identify the approvers of the SRS document. Approver name, signature, and date should be used.

**5).Supporting Information -**The supporting information makes the SRS easier to use. It includes:

1. Table of Contents
2. Index

* Appendices

**2) OVERALL DESCRIPTION**

**2.1 Product Perspective**

Developing **Parking Allocation Software** helps people in saving their precious time by showing them free space if available to park their vehicle. People can know outside the parking that space is available or not for parking. People can set priority if they have emergency work by calling at particular parking office of the particular building.

**2.1.1 SYSTEM INTERFACES**

**2.1.2 INTERFACES**

* Processor above Intel Pentium IV machines.
* Machines with at least 1 GB RAM should be used.
* All new operating system.
* Windows XP or above.
* Front End: Java, NetBeans.
* Back End : MySql

**2.1.3 HARDWARE INTERFACES**

**Monitor screen** –the software shall display information to the user via the monitor screen◦

**Mouse**– the software shall interact with the movement of the mouse and the mouse buttons. The mouse shall activate areas for data input, command buttons and select options from menus.

**Keyboard** –the software shall interact with the keystrokes of the keyboard. The keyboard

will input data into the active area of the database

**2.1.4 Software INTERFACES**

The GUI and other parts of the E­teaching software are to be done in JAVA and J2EE.The output of this software will need a web browserfor viewing it.

### 2.1.5 Communications Interfaces

Our system will deal with some secure data so for that we will use HTTPS as protocol because it is the secure and data will be transmitted in the form of packets and in case of https there are very less chance for packet loss.

**2.1.6 Memory Constraints**

The application is expected to have a rich client that will require significant computer resources.

The minimum client configuration will be a P4, 2.0 GHz will 512MB of RAM and 40GB storage.

**2.1.7 OPERATIONS**

Since there are offline system where people give their information and administrator note it down but that take too much time. After going inside there is not guarantee that you will going to get a space. Hence it is needed to built a system which will going to provide the free space available for parking..

### 2.1.8 Site Adaptation Requirements

Our system will store the data of the car owners so that they don’t have to give it again and again. Their will be going to be two operating systems in case one get failed.

**2.2PRODUCT FUNCTIONS**

* Facility to show available space available.
* People can know how many time you have come at that place..
* People can book by calling at Parking office(for emergency purpose).
* Facility to maintain records vehicle came in particular day.
* Facility to keep information about car and its owner.
* Facility to know for how much time a particular vehicle stayed there.
* Facility to give feedback.

**2.3 USER CHARACTERISTICS**

Each user must have valid id proof to enter in parking.

**2.4 CONSTRAINTS**

**Hardware Constraints:**

The system requires a database in order to store persistent data. The database should have backup capabilities.

**Software Constraints:**

The development of the system will be constrained by the availability of required software such as web servers, database and development tools. The availability of these tools will be governed by the TechnoSoft Informatics, Indore.

## 2.5 Assumptions and Dependencies

In our system we are assuming that system is available 24\*7 for consistent record of data. High speed computer is required to fetch data quickly. The standard algorithms should be followed.

# 3. Specific Requirements

## 3.1 External Interfaces

The specific Requirements for the project is to provide users the platform for easy and secure facility. Some of these Requirements are as follows:-

The software first of all authenticates the user (Administrator, Student)

and then provides list of task that can be performed and after proper selection

corresponding information is displayed. If the selection is that of manipulation than

appropriate fields will be displayed and after filling the updated information, changes are

reflected.

 To take input from the administrator about the access rules.

 To capture the information and data.

 To provide User Authentication, User Profile, Data Filter, Command filter,

Session Time, Session id.

 To filter the information and data on the basis of rule set by administrator.

 To provide the logbook for logging information regarding security relevant

events.

 To add and delete the already existing rules.

 To build simple to understand and attractive interface.

## 3.2 Functions

With this software we are aiming to digitalize the Parking Allocation System .This helps people to save their time.

* Register
* GetOwnerInfo
* GetCarInfo
* GetTimeDuration
* Set Priority

**Usability:** It is useful in following situations:

Directly show the available space in parking lot.

 Set priority in emergency cases by calling at parking office.

 In case of crime police can know for how much time that vehicle stayed and info of the owner.

 Only administrator can login.

## 3.2 Performance Requirements

The system can perform in heavy traffic also multiple faculties and students can simultaneously interact. The students will not have to wait even if the teacher is interacting to any other students. The system is designed so as to utilize the CPU resources in an optimum manner.

## 3.5 Design Constraints

* Java will be used for developing the project as all team members are trained to use it.
* Net beans IDE will be used as the development tool due to its familiarity with all the team members.
* Since all the team members are trained to use MySqlServer 5.0 as the database server, so it will be implemented in the back-end development.

### 3.5.1 Standards Compliance

There shall be consistency in variable names within the system. The graphical user interface shall have a consistent look and feel. Since we will access the centralized database so any change in that database will also affect the data transaction in our project also.

## 3.6 Software System Attributes

### 3.6.1 Reliability

Our system is reliable since we this software works for the good processor only. In case of failure data will get stored automatically in the hard drive.

**3.6.2 Availability**

The system shall be available 24 hours a day, 7 days a week. In case of failure the system shall allow users to restart the application after failure with the loss of at most 12 characters of input.

### 3.6.3 Security

Nobody can access private data of the vehicle owner instead of the administrator.

### 3.6.4 Maintainability

The system shall utilize interchangeable plugins. The system shall be easily updatable for fixes and patches. The system shall create logs of all changes, updates, or fixes that are done to the system shall be easy to upgrade.

### 3.6.5 Portability

The system shall be extremely portable via the usb drive.The system shall be easy to migrate or backed up via another usb drive.

* + 1. **Supportability**

This software is supportable on any platform with java compatibility.

## 

## 3.7 Organizing the Specific Requirements

For anything but trivial systems the detailed requirements tend to be extensive. For this reason, it is recommended that careful consideration be given to organizing these in a manner optimal for understanding. There is no one optimal organization for all systems. Different classes of systems lend themselves to different organizations of requirements in section 3. Some of these organizations are described in the following subclasses.

**3.7.1 SYSTEM MODE**

One mode is admin mode in which he will have access to all the data. He can update the database also. Depending upon the requirements the admin can give privileges to other admins and the users.

### 3.7.2 User Class AND OBJECTS

Our system is made on various use cases. These use cases are registor, set priority, entry exit time of a vehicle, create backup, manage database, check authenticity.

# Change Management Process

Changes to this document may be made after approval from the project manager and the client approval officer. This software will undoubtedly undergo change. Change will occur because errors have been encountered because the software must be adapted to accommodate change in its external environment or because the customer requires functional or performance enhancements.

The system can be updated as per requirement of the user. For any queries regarding any

part of the system to changed the user can contact at any of the above mentioned emails.

# Document Approvals

This is to certify that the documents related to project “**Parking Allocation Sytem**” are good & sufficient for a reader to understand what actually this project deals with. I have read the whole documents & found no error. Language used is also simple so that a beginner may also understand.