# Task 1: Analysis

## 1.1 Introduction

Analysis can be act as a static and dynamic model. Static model which mean the view of the system that does not change with the time whereas dynamic model it can be also known as view of the system that does change with time

Sometimes the final delivery of the project will not meet clients the expectations. So, it is hard to understand the requirement of clients adequately. To understand the expectation, we cannot do while doing testing the solution. That is why expectation must be understood-first that should be in high level project charter, but at firstly we must understand the business requirement.

So, analysis in the first step of waterfall model where the project begins. In this phase we break deliverable in the high-level agreement into the detailed requirement. This is the part of the project where you identify all way that this project while making project strategy.

In this analysis we have to identify this of this step

* Feasibility study
* Requirement analysis
* Functional
* Non functional
* Moscow prioritization
* Use case
* Class diagram

To analyze this project is use SWOT analysis which involved

* Strength: in this analysis phase, we can identify the strength of the system.
* Weakness: in this analysis phase, we can identify the weakness of the system so that we can convert those weakness into opportunity and then strength.
* Opportunity: in this analysis phase, we look what kind of opportunity we have I this project. So, we can change this our opportunity in strength.
* Threat: this is the phase where we identify the threats and upcoming threats for the system. And make a plan we overcome all of those threat.

## 1.2 feasibility study

Feasibility study can be defined as the initial stage of any project, which element the information that indicate whether the project can be implement or not.

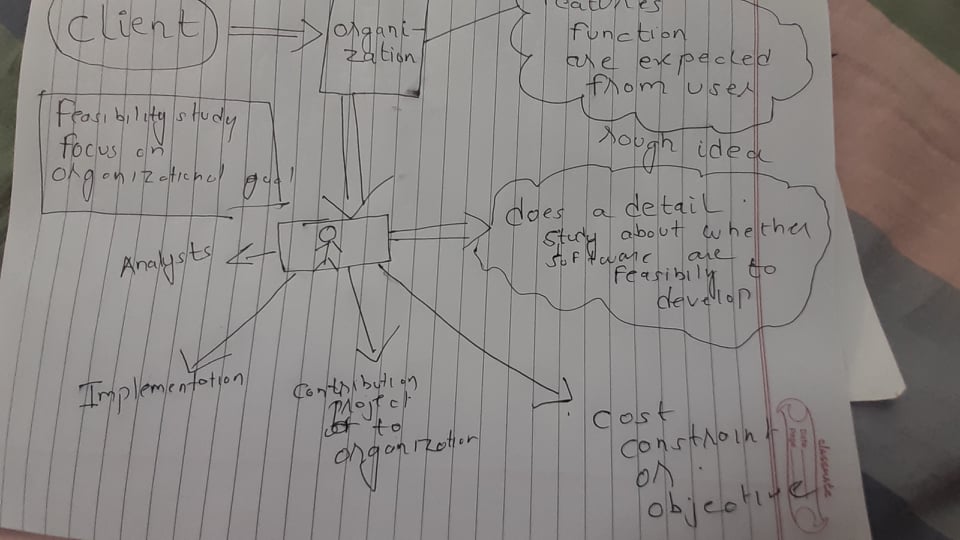


Figure :rough idea of feasibility study

Feasibility study focus on organizational goal. In this figure, client visit to can organization taking the system with the features, function are expected from user. This features in submitted to analyst who will check whether the system are feasible to develop. Who check implementation, contribution project to organization and cost constraint and objective.

**The types of feasibility study are**

* **Economic feasibility:**

Analyzing economic status and condition, project is economically fit being on costs from every expects. So, this project is fully supportive in term of economic feasibility.

* **Technical feasibility:**

For this project we need better pc and software where we can developed all php, java, html, css can be developed that mean we can run this application without any software or hardware problem. We need fast system with better processor and better ram and graphics. So their enough faster system as I need. So this project is technically feasible.

* **Schedule feasibility**

We have already divided the time in Gantt chart we we have schedule perfect time to implement this project. For this schedule we have use hard approach so we do not have xtra time for additional features. So, I will subject this overall project in given times.

* **Operational feasibility**

this project is based on browser-based application. It can be work in minimum environment. It is operational feasible with current technology.

* **Legal feasibility**

This system involves parking booking where user can book the parking space for their vehicle. As I have identified the problem in the proposal. This project is legally feasible and ethically valuable.

## 1.2 Requirement Analysis

It in the main system we have to fulfill in this system which I have describe below.

### 1. functional Requirement

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | Functional requirement | Description | Rational | Dependency |
| 1. | Login | End user have to enter username and password for login into web | Need authentication to get access  For regular user | 2 |
| 2. | Register | End user need to valid its information | Validating user account and create account in parking booking system | 1 |
| 3. | Select parking space | User need to search free parking space | To get free space to park their vehicle | 4 |
| 4. | Enter number plate | Have to enter user number plate | To get the security of the vehicle | 4 |
| 5. | Enter user data | User should give their information | In order to keep record of customer. | 4 |

### 2.Non functional

|  |  |  |  |
| --- | --- | --- | --- |
| Id | Non functional | Description | Dependency |
| 1. | Vehicle service | User can give their vehicle for service or if he/she do not want he may ignore |  |
| 2. | Login | User may or may not login to register in our account |  |
| 3. | Register to login | User may or may not login to register their personal account |  |

### 3. Moscow prioritization



Figure 2:Moscow prioritization

Moscow prioritization can be known as Moscow method is famous way for management of the requirement. In order to maintain priority of the project. The way to prioritize requirement is to:

* It helps to reduce development cost for this project.
* It makes sure the most efficient requirement has been cared.
* It helps to manage the project better.

Basically, there are four type of this method

* Must have: number plate detail, Supervision panel, Contact detail
* Should have: login, registration, feedback
* Could have: CCTV under surveillance
* Won’t have: donation, discount, online payment system.

### 4. System Requirement Specification

System requirement. This I have done to analyze the system requirement of my laptop whether the system is supportive or not for my project parking booking system. A requirements specification for a software system is a complete description of the behavior of the system to be developed. This system specification:

* It includes set of use cases that describes all interactions the users will have with the software.
* It is consisting of functional which describe functionality or system service which depend on type of software expected user
* It also consists of non-functional requirements that defines system properties and constraint. For example: reliability, response and storage requirement.

**System requirement**

* **Operating system:** Windows 7,8,10
* **System type:** 64 bit or 32 bits.
* **Processor:** Intel(R) Core (TM) or duo.
* **Ram:** 4 gigabyte or 8 gigabytes.
* **Programming language:** PHP, HTML, JAVASCRIPT, CSS.
* **Database:** MS SQL Management Studio.

### 5. NLA

NLA is a process selecting the requirement that has been provided in this project. I have use this natural language analysis to select the class from the given verb and noun.

**Steps that will help in identifying the candidate class are:**

1. List out the noun from project background.

2. Reducing the repeating noun.

3. Only some of the noun are candidate classes.

4. list out the verb from project background.

5. reducing the repeating verb.

6. only some of the verb are candidate classes.

**NOUN**

Technology, people, problem, parking, vehicle, workstation, parking, webpage, vehicle, parking, client, online, parking, booking, price, system, security, user, interface, customer, registration, booking, php, model, individual, time, traffic jam, country, user, valid user, parking, space, workstation, vehicle, application, storage, google, webpage, task, story, case, statement, project, register, record, system, engineer, reserve, spot, transaction, price, bill, manager, owner, parking, slot, car, system user, manager, member, user, utilities, card

|  |  |
| --- | --- |
| **Id** | **Candidate Class** |
| 1 | manager |
| 2. | Parking |
| 3. | Car park |
| 4. | Member |
| 5 | Parking Slot |

**VERB**

Advanced, trouble, book, online parking, booking, register, wait, affordable, login, discount, designed, tutorial, other, identify, login, book, service, park, download, manage, direct, find, short, analyze, edit, check, configuration.

|  |  |
| --- | --- |
| **Id** | **Candidate Class** |
| 1 | Online parking |
| 2. | Booking |
| 3. | Registration |
| 4. | Login |
| 5 | edit |

### 6. use case diagram

* A use case diagram shows a set of use cases and actors or special kind of class and their relationship
* Address the static view of the system which cannot be changed
* Especially it is important in organizing and modeling the behavior of system.

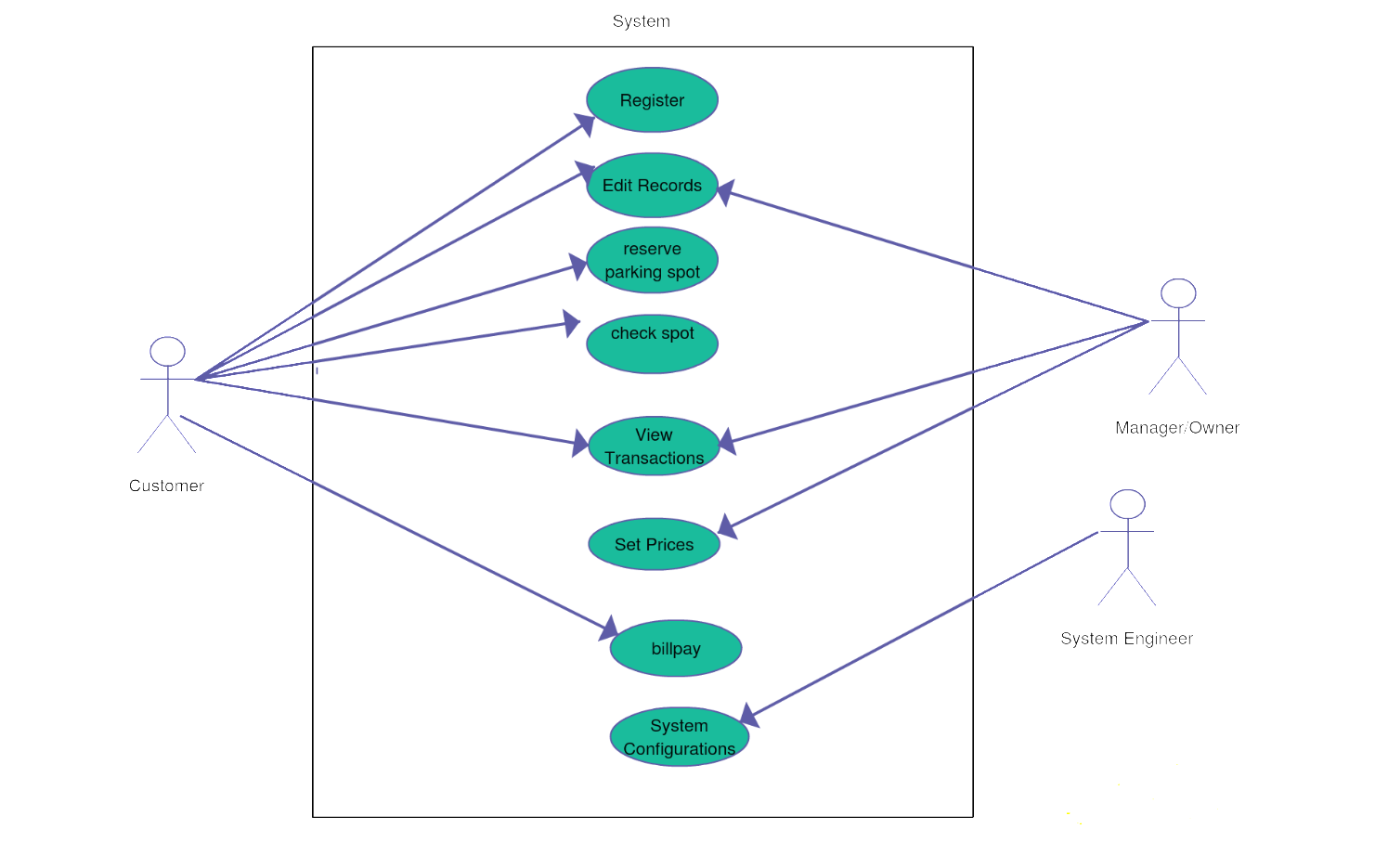


Figure :use case diagram for parking booking system

|  |  |  |
| --- | --- | --- |
| **ID** | **TITLE** | **DESCRIPTION** |
| 1. | REGISTER | To access login for customer. Customer have to register in this application |
| 2. | EDIT RECORD | To change the record set by the customer. |
| 3. | RESERVE PARKING SPOT | For s |
| 4. | CHECK SPOT |  |
| 5. | VIEW TRANSACTION |  |
| 6. | SET PRICE |  |
| 7. | BILL PAY |  |
| 8. | SYSTEM CONFIGURATION |  |

# 7. Class Diagram

it is the graphical representation of a set of elements, extracted as a connected graph of things abd relationship.

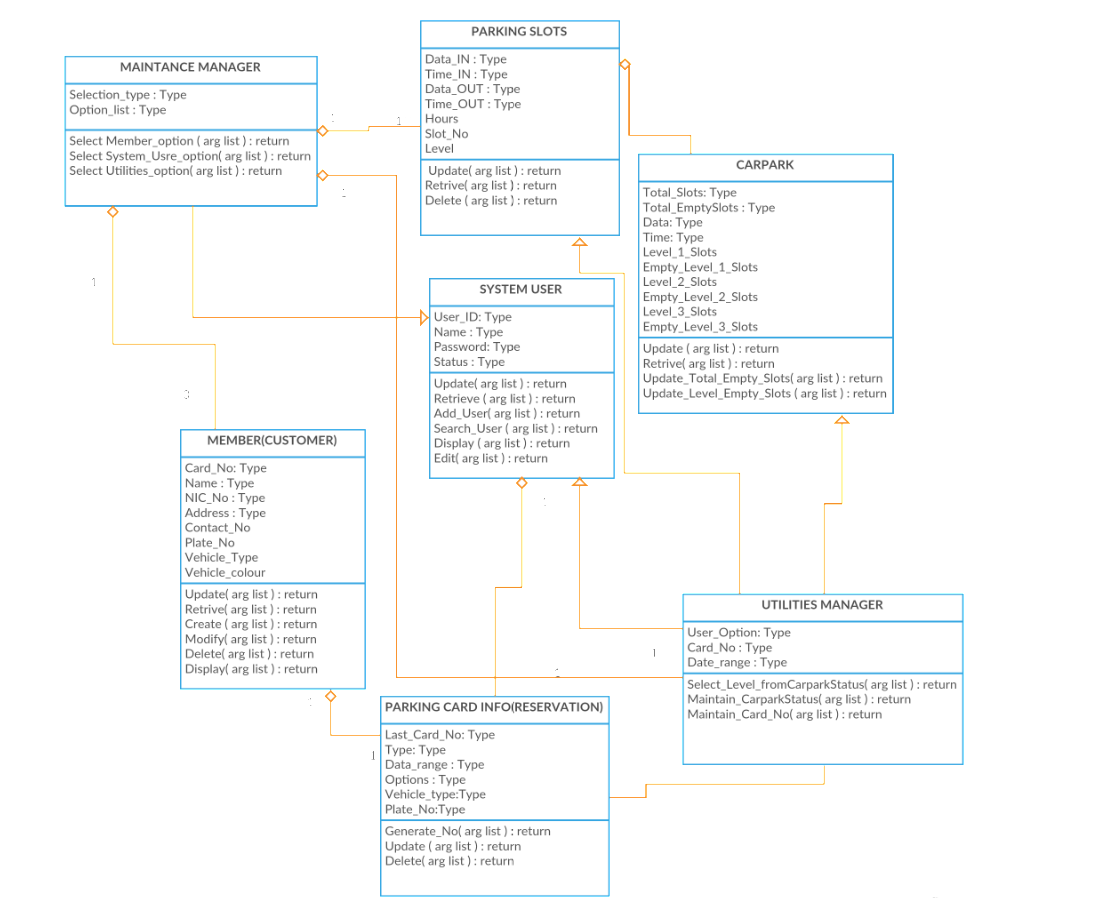
* Class diagram shows the set of classes, interface & collaboration and their relationship
* Most common diagram found in modeling object-oriented system
* Address the static design view of the system.

Figure :class diagram