Lab DA-1 (SE Project), Winter 2020-21, L9+L10

PARKING MANAGEMENT SYSTEM

by Ayushmaan, Sharadindu, & Tanmay
[19BCE2070, 19BCE2105, 19BCE2106]

AIM AND OBJECTIVE: The aim of this project is to build a Vehicle Parking management system that enables the time management and control of vehicles using number plate recognition. The system that will track the entry and exit of cars, maintain a listing of cars within the parking lot, and determine if the parking lot is full or not. It will determine the cost of per vehicle according to their time consumption.

ABSTRACT: It is an automatic system which delivers data processing in very high speed in systematic manner. Parking is a growing need of the time. Development of this system is very useful in this area of field. We can sell this system to any organization. By using our system they can maintain records very easily. Our system covers the every area of parking management. In coming future there will be excessive need of Vehicle parking management system.

MOTIVATION: Parking management system for managing the records of the incoming and outgoing vehicles in a parking house. It's an easy for Admin to retrieve the data if the vehicle has been visited through number, he can get that data.

Through this, we can park our vehicle in our own slot by paying.

- Because of that there will be no towing problems.
- And our vehicle has been parked as a secure condition.
- There is no risk for vehicle owner for parking the car.
- In case of any damages and problem of vehicle that will claim by parking management.
- As the world is facing many threads daily, robberies are done easily with no track to trace, bomb blasts occur with the use of vehicle, so if a proper system is adopted each and every record can be saved and anyone can be track easily therefore mainly is to make a better and fast software, most important user-friendly
- Maintain records in short time of period.
- Determines the parking area is full or not.
- Enhances the visitor's experience.

APPLICABILITY: Since this entire system will be (we're optimistic) implemented on the entire parking slot of a building, it'll be applicable to all those users (customers, placeholders, private residents, owners, and the likes) –whoever parks their vehicles in the said parking system. It'll even become a breeze to the maintenance staff as well us automating the parking ticket process.

SCOPE OF THE PROBLEM IDENTIFIED (PROBLEM DESCIPTION & ISSUES OF EXISTING SYSTEM):

Nowadays in many public places such as malls, multiplex systems, hospitals, offices, market areas there is a crucial problem of vehicle parking. The vehicle parking area has many lanes/slots for car parking. So to park a vehicle one has to look for all the lanes. Moreover, this involves a lot of manual labour and investment. Instead of vehicle caught in towing the vehicle can park on safe and security with low cost.

Parking control system has been generated in such a way that it is filled with many secure devices such as, parking control gates, toll gates, time and attendance machine, car counting system etc. These features are hereby very necessary nowadays to secure your car and also to evaluate the fee structure for every vehicles' entry and exit.

END USERS:

- General Public
- Building owners
- VIPS, VVPIS, Invited guests alike
- Anyone else who drive through the area for parking

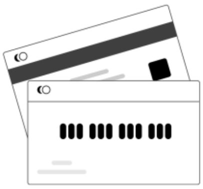
MODULES:

STAFF RECORDS

USER RECORDS

VEHICLE RECORDS

VEHICLE PARKING DETAIL

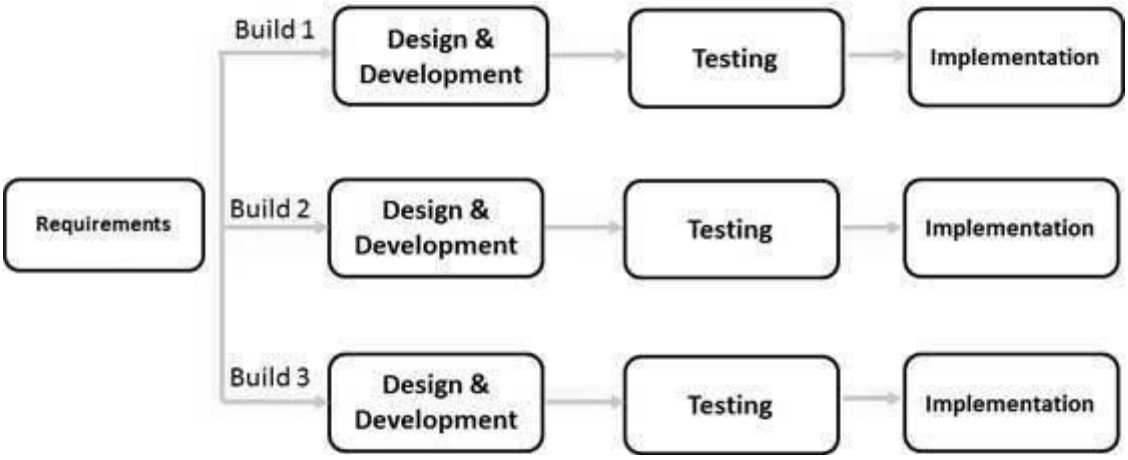


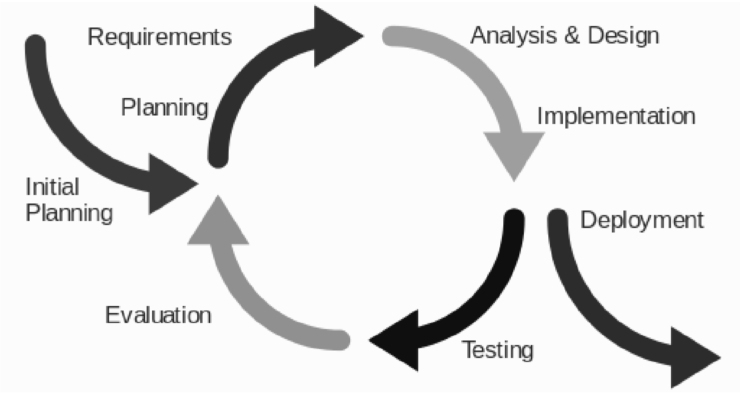
TRANSACTION DETAIL

MODULE DESCRIPTION:

1. STAFF RECORDS: It helps to provide details of staff that uses the Vehicle parking management System.
2. USER RECORDS: This record helps for the authorization for using Vehicle Parking Management System. It Provides the Username and Password for the User (staff). It also includes the level of authority that means it separates the normal users and administrator.
3. VEHICLE RECORDS: This most important record which focuses in our Vehicle Parking Management System.
4. VEHICLE PARKING DETAIL: This report is very essential in this system. This report provides a brief summary of vehicle activities. It shows the overall Entry and Exit time.
5. TRANSACTION DETAIL: Has control over the entire database and regularly updates the list.

PROCESS MODEL: Iterative Model





Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. At each iteration, design modifications are made and new functional capabilities are added.

We’ve chosen iterative life cycle model since it does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements. This process is then repeated, producing a new version of the software for each cycle of the model.

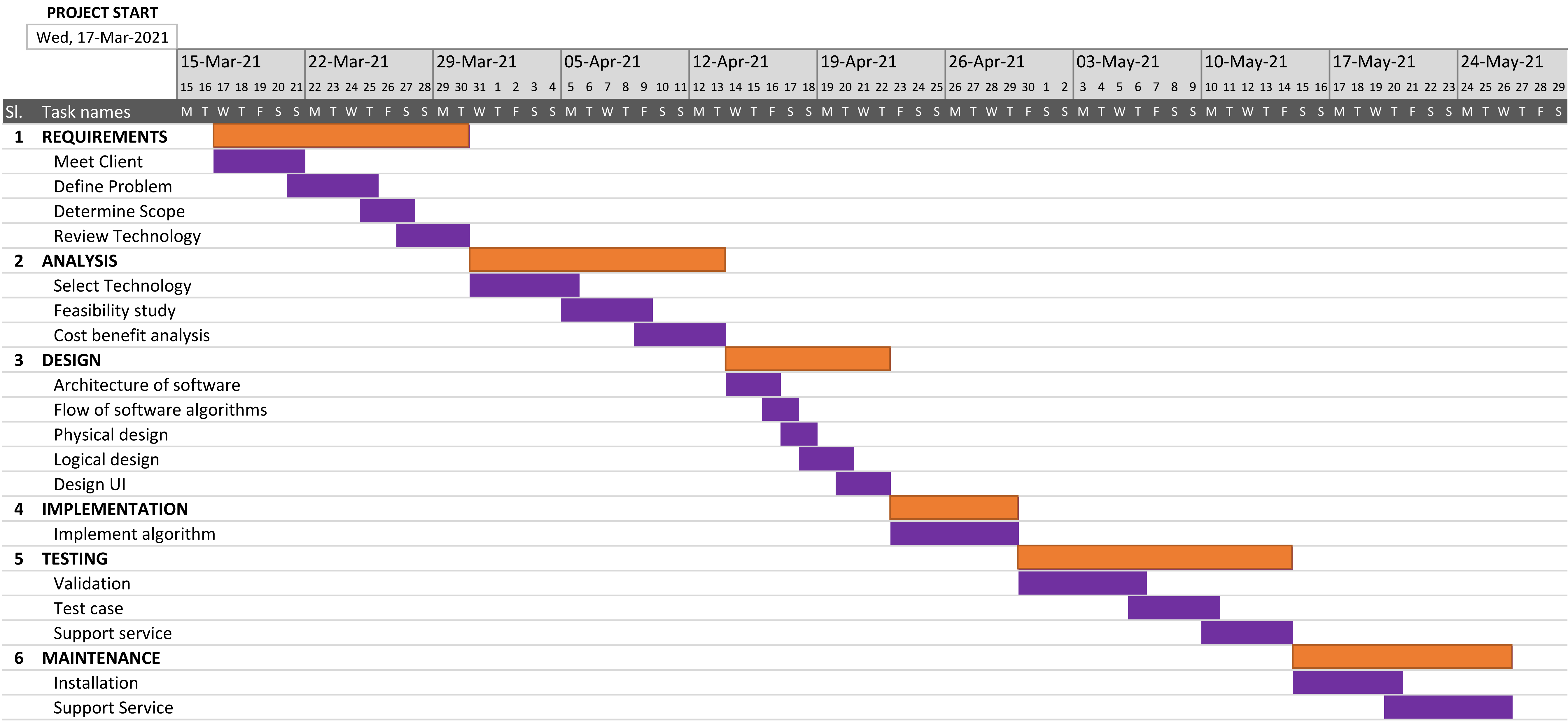
REFERENCES:

1. versionx.in/parking-management-system
2. trume.in/parking-management-system-in-india
3. delopt.co.in/parking-management-system.html
4. omnitecgroup.com/Solutions/parking-controls/parking-management-system

GANTT CHART (probable; subject to changes):

PROJECT START	Wed, 17-Mar-2021

Sl. No.	Task names	Duration	Probable Start	Probable Finish
1	**REQUIREMENTS**	**13 days**	**17-Mar-21**	**30-Mar-21**
	Meet Client	4 days	17-Mar-21	21-Mar-21
	Define Problem	4 days	21-Mar-21	25-Mar-21
	Determine Scope	2 days	25-Mar-21	27-Mar-21
	Review Technology	3 days	27-Mar-21	30-Mar-21
2	**ANALYSIS**	**14 days**	**31-Mar-21**	**13-Apr-21**
	Select Technology	6 days	31-Mar-21	05-Apr-21
	Feasibility study	4 days	05-Apr-21	09-Apr-21
	Cost benefit analysis	4 days	09-Apr-21	13-Apr-21
3	**DESIGN**	**8 days**	**14-Apr-21**	**22-Apr-21**
	Architecture of software	2 days	14-Apr-21	16-Apr-21
	Flow of software algorithms	1 day	16-Apr-21	17-Apr-21
	Physical design	1 day	17-Apr-21	18-Apr-21
	Logical design	2 days	18-Apr-21	20-Apr-21
	Design UI	2 days	20-Apr-21	22-Apr-21
4	**IMPLEMENTATION**	**6 days**	**23-Apr-21**	**29-Apr-21**
	Implement algorithm	6 days	23-Apr-21	29-Apr-21
5	**TESTING**	**14 days**	**30-Apr-21**	**14-May-21**
	Validation	6 days	30-Apr-21	06-May-21
	Test case	4 days	06-May-21	10-May-21
	Support service	4 days	10-May-21	14-May-21
6	**MAINTENANCE**	**11 days**	**15-May-21**	**26-May-21**
	Installation	5 days	15-May-21	20-May-21
	Support Service	6 days	20-May-21	26-May-21


PROJECT START
Wed, 17-Mar-2021
15-Mar-21
22-Mar-21
29-Mar-21
05-Apr-21
12-Apr-21
19-Apr-21
26-Apr-21
03-May-21
10-May-21
17-May-21
24-May-21
15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
Sl.
Task names
M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S
1 REQUIREMENTS
Meet Client
Define Problem
Determine Scope
Review Technology
2 ANALYSIS
Select Technology
Feasibility study
Cost benefit analysis
3 DESIGN
Architecture of software
Flow of software algorithms
Physical design
Logical design
Design UI
4 IMPLEMENTATION
Implement algorithm
5 TESTING
Validation
Test case
Support service
6 MAINTENANCE
Installation
Support Service