



American International University-Bangladesh (AIUB)

Department of Computer Science and Engineering
Faculty of Science & Technology (FST)
Fall-2021

Software Engineering
Section: E

Project Title: AIUB Parking Management System

Submitted by: Group #05

SL No	Student Name	Student ID
1	SHAHRIAR, S. M.	19-39596-1
2	CHOWDHURY, IBRAHIM ADNAN	19-40653-1
3	SHAHRIAR MUSHFIQUL ISLAM	19-39497-1
4	UDDIN, MD RAIHAN	19-39648-1

Submitted to:

FARZANA BENTE ALAM

PROJECT TITLE:

AIUB Parking Management System

PROJECT OVERVIEW:

The parking management system is very useful for managing the records of the incoming and outgoing vehicles in a parking space. So that, it becomes much easier for the manager to retrieve the data and update the database.

The parking control system has been generated in such a way that it is filled with many security devices such as parking control gates, sensors, token generating machines, etc. A car/vehicle arrives at the parking booth, at first it has to confirm its identity. If the parking booth gives access, it will arrive at the parking control gate. Then the sensor will give a signal to the system. And the system will take access control for the car/vehicle entry. The system will check the space available for the car/vehicles. If the space is available, it will generate a token using the token generating machine. And the driver has to collect the token from the token/entry manager. However, if the driver of the car/vehicle has an AIUB ID or car/vehicles which are AIUB owned, then those will get direct entry without the token if the space is available. The parking space manager keeps updating the space database and generates a report from time to time.

JUSTIFICATION:

The project provides a platform with technologically advanced security features and parking management systems that can provide you with enhanced security, safety, and privacy.

- It prevents unauthorized access to their parking lots. As a result, car owners will have increased confidence that their cars are well protected.

- It provides a real-time omnichannel capability to ensure that customer demands for convenience and speed are fulfilled.
- It also increases reservations and boosts loyalty.
- This protects the customer's commitment to payment and reduces the labor of workers in the parking lot.
- It provides the necessary return on investment.
- It provides a parking environment that both the management team and their customers can handle comfortably.
- These parking control systems offer a token machine system and provide solutions tailored to individual needs.

All these points lead to positive changes, less parking fraud, and desirable results in the end.

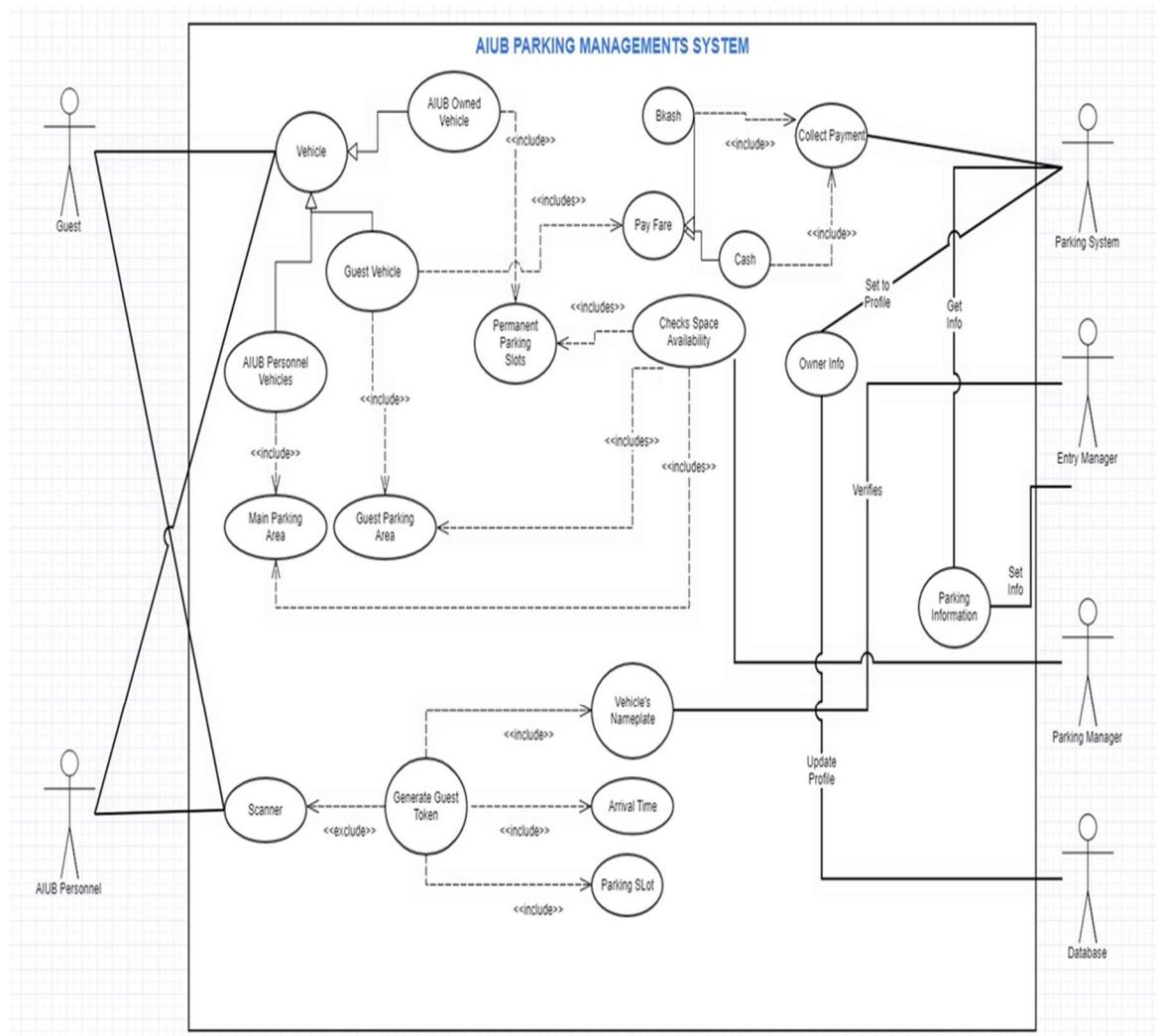
Functional requirement:

Functional requirement define the fundamental actions that system must perform. The functional requirements for the system are divided into one main category.

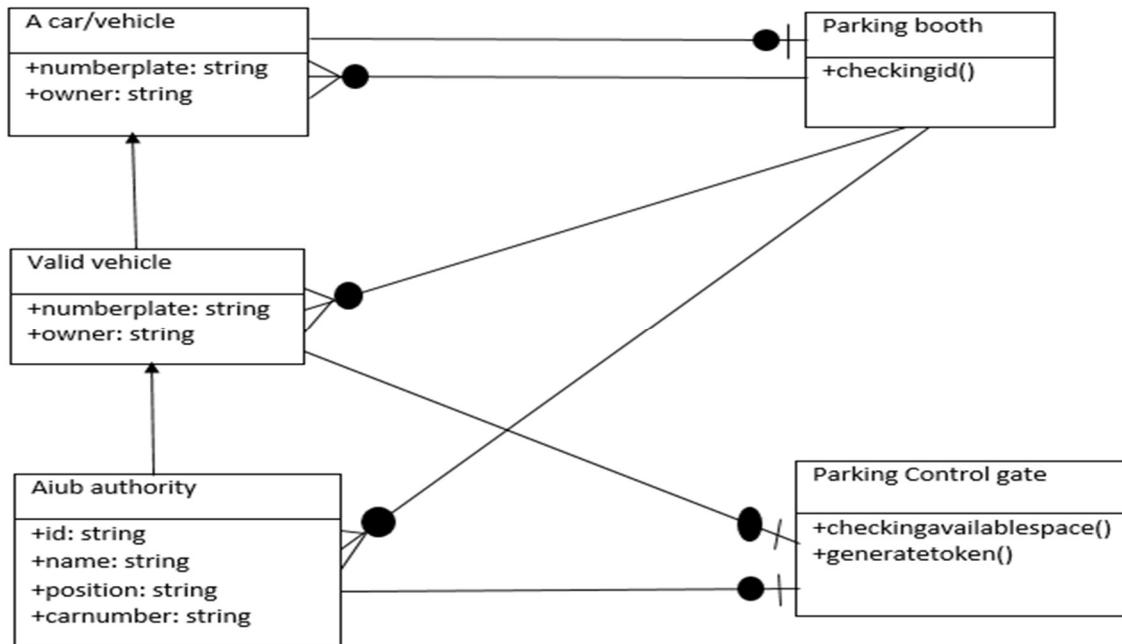
- **Data Entry & searching for space:**
 1. System shall record all the data entry in the database.
 2. System shall record the owner's first name.
 3. System shall record the owner's last name.
 4. System shall record the owner's Id.
 5. System shall record owners position in AIUB (VC/Faculties/Students)
 6. System will provide emergency parking space facilities to the AIUB owners & Faculties.
 7. The system shall record the entry date & time.
 8. The system shall record the owner's vehicle number plate.
 9. After checking the ID the system will check whether there is any space available or not.
 10. If there is space available then system will provide the owner a token.
 11. While leaving the owner have to return the token to the system & the system will match the token number with database row & check whether

the owner is the right person or not & also check vehicles number plate which taken by the owner with database entered vehicles number plate. If it matches then the system will record the exit time & date of that owner. If it do not match then the system will lock the entry gate.

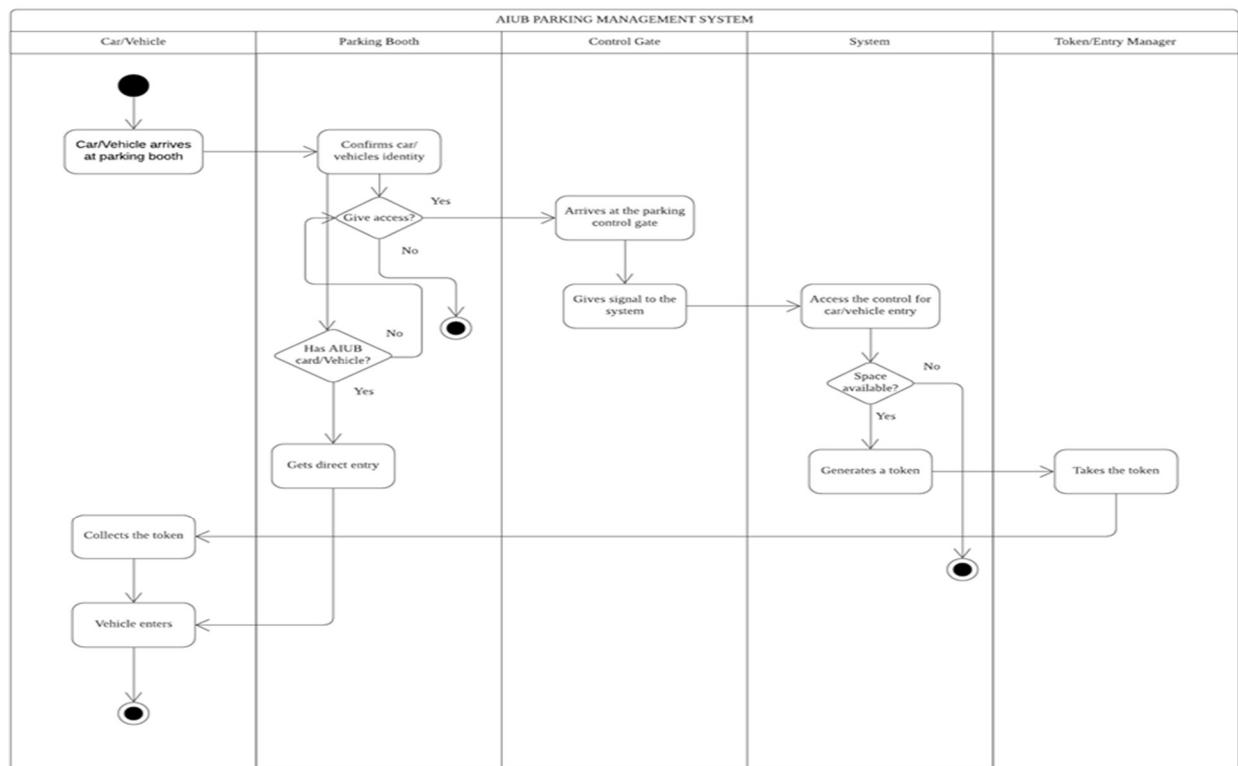
USE CASE DIAGRAM:



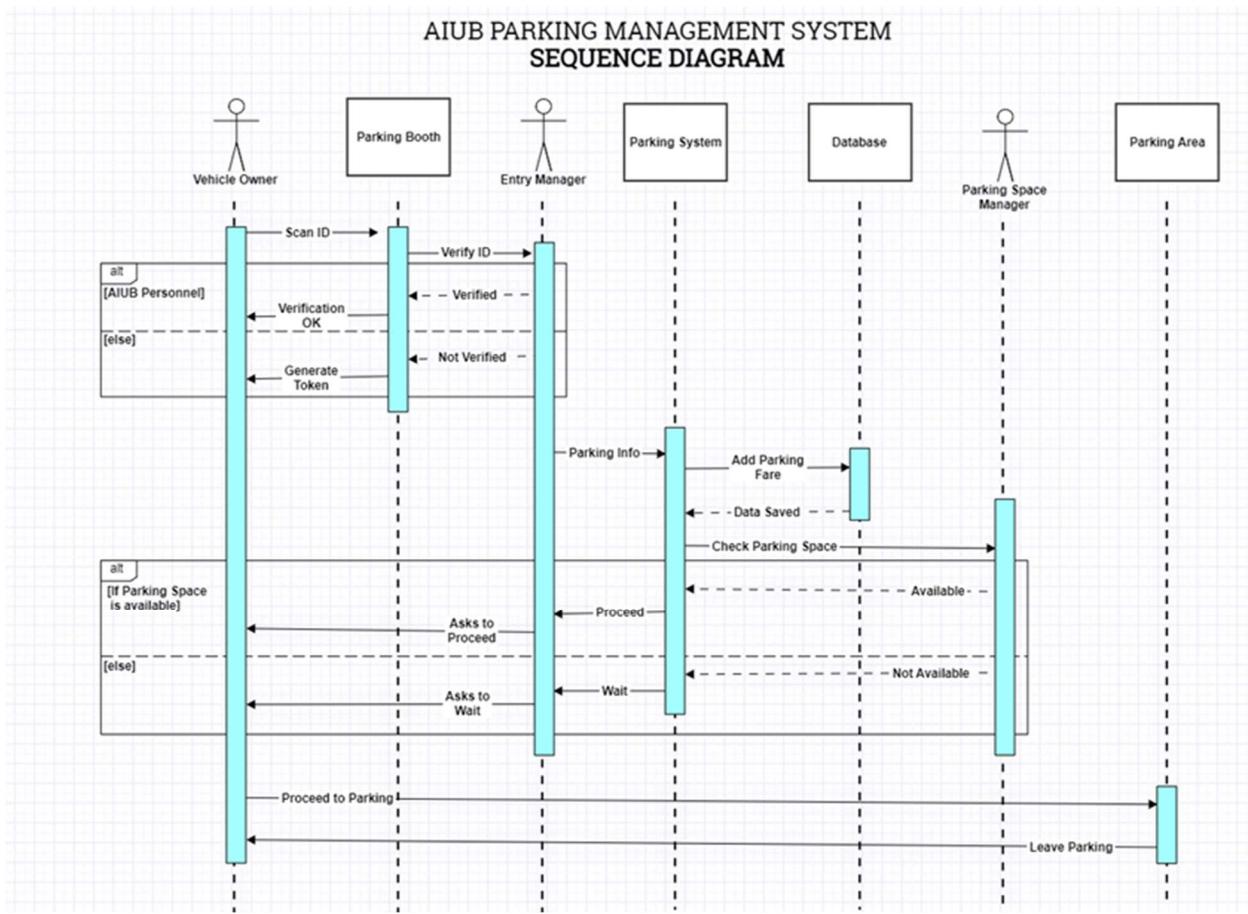
CLASS DIAGRAM:



ACTIVITY DIAGRAM:



SEQUENCE DIAGRAM:



PROCESS MODEL:

We need to develop the software in a user-friendly environment. Because, many new requirements may come when users start to use this software. So, it's better to use "Agile" Method for developing this software. If we use plan driven model to build up this software, then it would not be possible to change the feature with user's requirements. Also, we cannot update the software which would not be user friendly at all. On the other hand, Agile is People-based rather than Plan-based and lightweight also. So, to make the software user friendly as well up to date with the upcoming other future software, we need to use the agile method. There will be a feedback option for various type users and can be change the software based on

those feedback which will make it more user friendly. There are many Agile methods to develop a software. But we will use Scrum model. Because,

- Scrum can help teams complete project deliverables quickly and efficiently
- Scrum ensures effective use of time and money
- Large projects are divided into easily manageable sprints
- Developments are coded and tested during the sprint review
- Works well for fast-moving development projects
- The team gets clear visibility through scrum meetings
- Scrum, being agile, adopts feedback from customers and stakeholders
- Short sprints enable changes based on feedback a lot more easily
- The individual effort of each team member is visible during daily scrum meetings
- In scrum time boxed iterations are prescribed.
- Scrum team commits to a specific amount of work in an iteration.
- Scrum uses Velocity as default metric for the planning and the process improvement.
- Scrum team is Cross-functional.
- In scrum items are broken down so that they can be completed within the one sprint
- Scrum has more transparency and visibility than any other methodology.
- It increases the team accountability.
- Scrum is easy with changes. It accommodates changes.
- Scrum is a cost saving development methodology.

PROJECT ROLL IDENTIFICATION:

Using the SCRUM model, we have to create some roles and teams for software development. Mainly Scrum Master, Product Owner, Scrum Team, Benefitors, Management.

- o **Scrum Master:** Scrum Master is responsible for ensuring that the project is carried through according to the practices, values & rules. Scrum Master also selected items for ‘Sprint Backlog’ with Product Owner & set the goals for the sprint.
- o **Product Owner:** Product Owner is officially responsible for the project, managing, controlling & making visible the product backlog list who is selected by the Scrum Master, The benefitors & The Management. He takes the final decision of the product backlog.
- o **Scrum Team:** Scrum Team is the project team that has the authority to decide on the necessary action & to organize itself in order to achieve the goals of each sprint. The scrum team involved in effort estimation, creating sprint backlog & review the backlog list.
- o **The Benefitors:** They participates in the task related to product backlog items for the system being developed. Mainly it represented all the benefitors who send the feedback of the software.
- o **Management:** Management is in charge of final decision making. It also setting the goal & requirements of the software development.

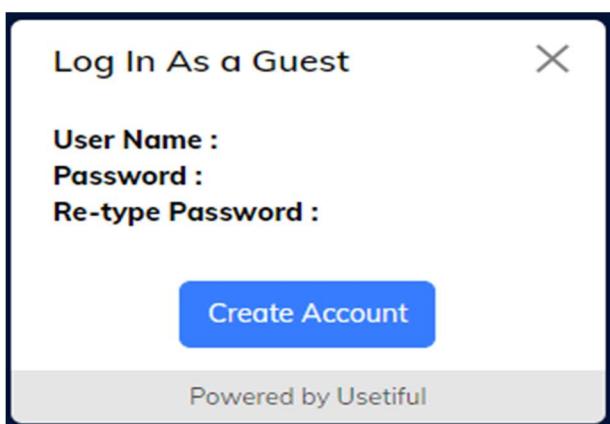
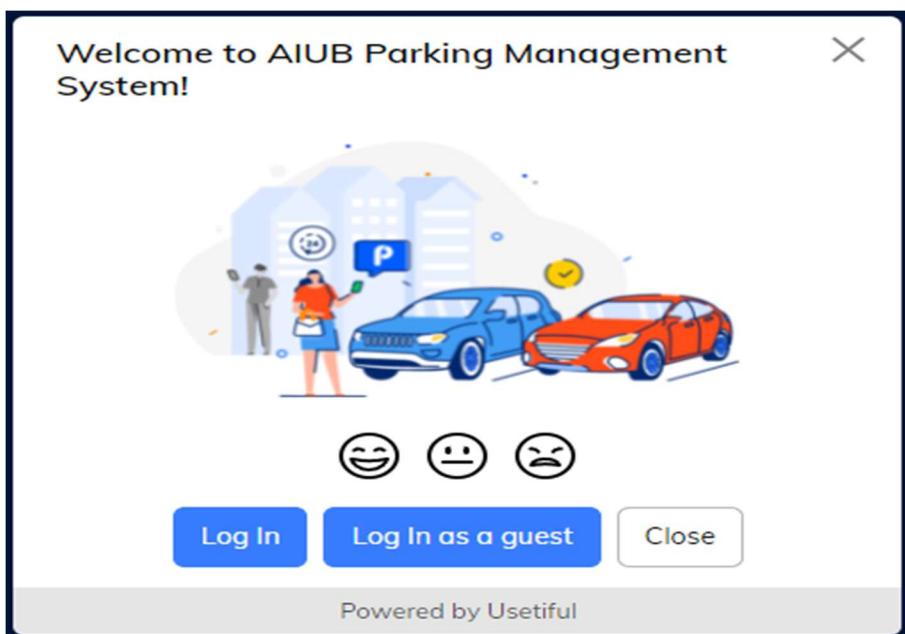
Meetings of SCRUM

- o **Sprint Planning Meeting:** There is 2 part of this meeting what is organize by Scrum master. In 1st part, they set the goals & the functionality of the next Sprint. The Scrum master, Management, Product owner & Scrum team participate in this 1st part of the meeting. In 2nd part, only Scrum master & Scrum team discuss about product increment is implemented on the Sprint.

o Daily Scrum Meeting: Everyday approximately 15 minutes the Scrum master & Scrum team discuss about what has been done since the last meeting & what is to be done before the next one.

o Sprint Review Meeting: On the last day of the Sprint, the Scrum team & the Scrum master present the result of the Sprint to the management, benefitters, users & the product owner in this meeting. This meeting also brings out new backlog items for the system built.

Project Prototype:



Log In X

ID/User Name:
PASSWORD:

Next

Powered by Usetiful

Home X



Check Parking Space Situation
Data Entry for Parking

Powered by Usetiful

Parking Space Situation X

Total Parked Vehicle :
Total Space Left :

Close

Powered by Usetiful

Data Entry

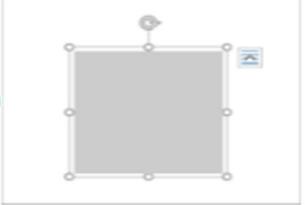
First Name	Last Name	Institute ID	Position In AIUB	Vehicle Number Plate	Parking Date	Parking Time

Generate Token

Powered by Usetiful

Get Vehicle

Scan Token



Go to Payment Method

Powered by Usetiful

Payment Method

Cash
 Bkash

Generate a slip

Powered by Usetiful

Test Planning:

1. Our application is Web base Software.
2. We will use Black Box testing method for our AIUB Parking Management Software so that it can be tested stand alone.

There are many types of Black box testing but the following are the prominent ones:

- **Functional testing:** This black box testing type is related to the functional requirements of a system; it is done by software testers.
- **Non-Functional testing:** This type of black box testing is not related to testing of specific functionality, but non-functional requirements such as performance, scalability, usability.
- **Regression testing:** Regression testing is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.

We choose Black-box testing-In functional testing, one does not have access to the internal details of a program and the program is treated as a black box. Black-box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing.

3. There are some test cases for the software. Test Cases are given below:

Project Name: AIUB Parking Management System		Test Designed by: Shahriar					
Test Case ID: TestRun_1		Test Designed date: 11/8/2021					
Test Priority (Low, Medium, High): Medium		Test Executed by: Group - 5					
Module Name: login as a guest		Test Execution date: 11/8/2021					
Test Title: Create account for login as a guest							
Description: From welcome page to login as a guest							
Precondition (If any): User must have username and strong password							
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)			
1. Go to the website 2. Go to login as a guest 3. Enter username 4. Enter password 5. Enter re-confirm password 6. Click create account	Username: abcde Password: ABcde12345	User should create an account	As expected,	Pass			
Post Condition: User is validated with database and successfully create an account. The account session details are logged in the database.							

Project Name: AIUB Parking Management System		Test Designed by: Shahriar					
Test Case ID: TestRun_2		Test Designed date: 11/8/2021					
Test Priority (Low, Medium, High): Medium		Test Executed by: Group - 5					
Module Name: Login Session		Test Execution date: 11/8/2021					
Test Title: verify login with valid username and password							
Description: Test website login page							
Precondition (If any): User must have valid username and password							
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)			
1. Go to the website 2. Enter username 3. Enter password 4. Click Next	Username: abcde Password: Abcde12345	User should login into the application	As expected,	Pass			
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.							

Project Name: AIUB Parking Management System		Test Designed by: Shahriar					
Test Case ID: TestRun_3		Test Designed date: 11/8/2021					
Test Priority (Low, Medium, High): Low		Test Executed by: Group - 5					
Module Name: Home page Session		Test Execution date: 11/8/2021					
Test Title: Verify home page							
Description: Test home page							
Precondition (If any): User must have valid username and password							
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)			
1. Go to the website 2. Check parking space situation 3. Check data entry for parking	parking space situation, data entry for parking	User should be able to have suggestions through this application.	As expected,	Pass			
Post Condition: None.							

Test Case ID: TestRun_4		Test Designed date: 11/8/2021					
Test Priority (Low, Medium, High): Medium		Test Executed by: Group - 5					
Module Name: Check parking space situation session		Test Execution date: 11/8/2021					
Test Title: Verify parking space situation session							
Description: Test parking space situation working correctly or not							
Precondition (If any): User must have opened the website							
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)			
1. Go to the website 2. Tap parking space situation	void	User should be able to see total parked vehicle at that situation & how much space is left.	As expected,	Pass			
Post Condition: None.							

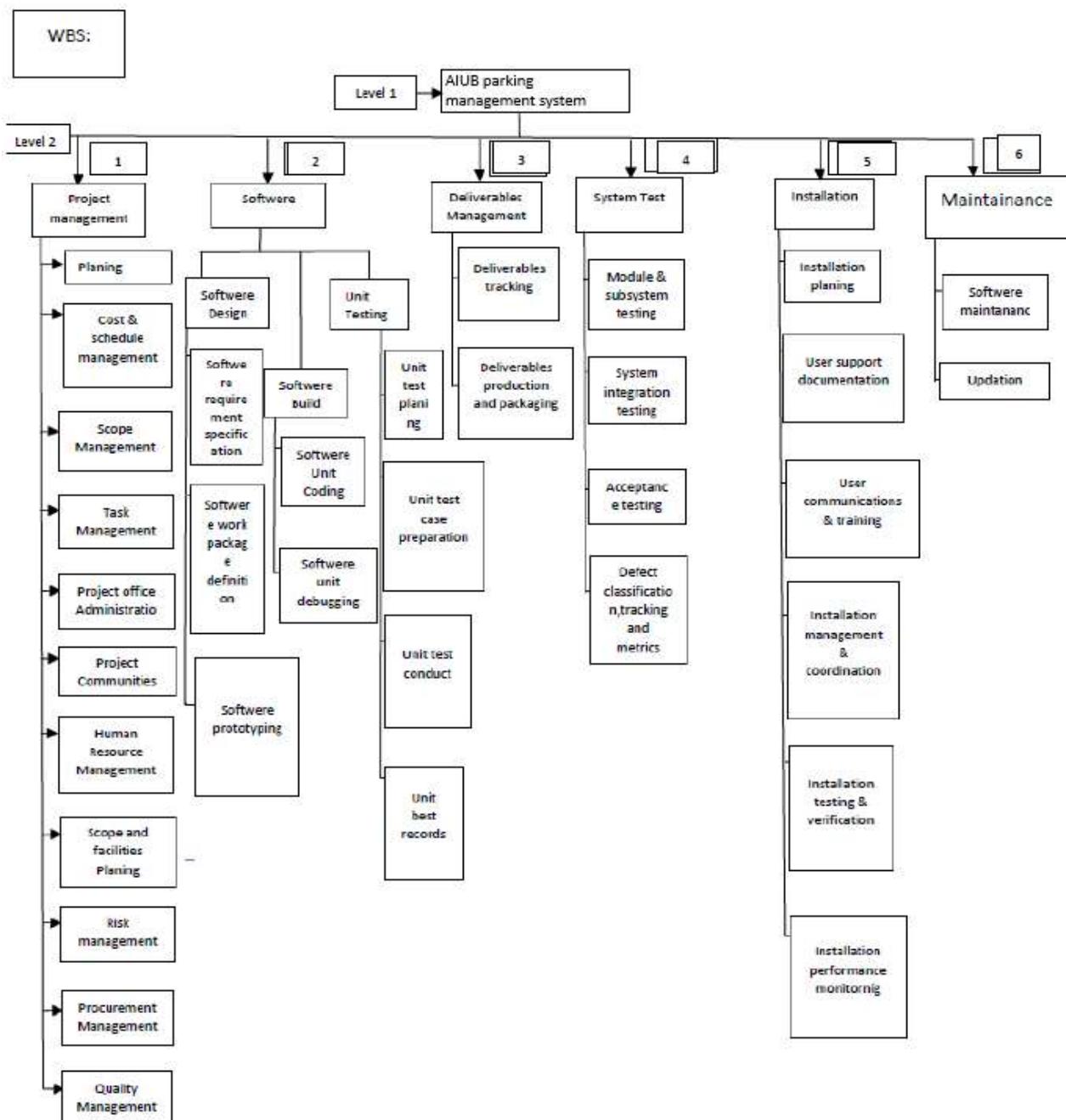
Test Case ID: TestRun_5	Test Designed date: 11/8/2021			
Test Priority (Low, Medium, High): High	Test Executed by: Group - 5			
Module Name: Check data entry session	Test Execution date: 11/8/2021			
Test Title: Verify data entry session				
Description: Test data entry working correctly or not				
Precondition (If any): User must have valid username and password				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Login 3. Tap data entry for parking	First Name: <u>Abc</u> Last Name: Xyz Institute Id: 19-11111-2(for every institute individuals) & null (for guest) Position in AIUB: Student & null (for guest) Vehicle Number Plate: Dhaka Metro <u>Ko</u> 22-2222 Parking Date: 11/8/2021 Parking Time: 4:28 pm	User should be able to put all the information for data entry.	As expected,	Pass
Post Condition: User should successfully put all the information in database.				

Project Name: AIUB Parking Management System	Test Designed by: <u>Shahriar</u>			
Test Case ID: TestRun_6	Test Designed date: 11/8/2021			
Test Priority (Low, Medium, High): High	Test Executed by: Group - 5			
Module Name: Check Generate Token Button	Test Execution date: 11/8/2021			
Test Title: Verify Generate Token Button				
Description: Test Generate Token Button working properly or not				
Precondition (If any): User must put all the required information in the data entry section				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the data entry 2. Fill all the required information 3. Press Generate Token	void	It should generate a unique token.	As expected,	Pass
Post Condition: The system has to record in its own database what token it generates & user have to collect the token.				

Project Name: AIUB Parking Management System		Test Designed by: Shahriar					
Test Case ID: TestRun_7		Test Designed date: 11/8/2021					
Test Priority (Low, Medium, High): High		Test Executed by: Group - 5					
Module Name: Check Scan Session		Test Execution date: 11/8/2021					
Test Title: Verify Scan Session							
Description: Test Scan working properly or not							
Precondition (If any): User must have to scan his/her token in the system							
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)			
1. Go to Get Vehicle 2. Scan the token	A token picture	It should match with the token the system generated & recorded.	As expected,	Pass			
Post Condition: If It matches the system have to clear the recorded token it generated from the systems database.							

Project Name: AIUB Parking Management System		Test Designed by: Shahriar					
Test Case ID: TestRun_8		Test Designed date: 11/8/2021					
Test Priority (Low, Medium, High): Medium		Test Executed by: Group - 5					
Module Name: Check Payment Method Session		Test Execution date: 11/8/2021					
Test Title: Verify Payment Method							
Description: Test Payment Method							
1. Precondition (If any): If the token matches the payment method option will be visible							
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)			
1. Go to Payment Method option 2. Select preferred payment option	cash	The payment should be clear & It should generate a payment slip	As expected,	Pass			
Post Condition: All the earlier inserted data by the user has to clear after the payment.							

WBS:



Effort Estimation:

Here our software project is small that's why we are using organic constructive cost model.
So, for organic constructive cost model,

Project complexity, $P = 1.05$

SLOC dependent coefficient, $T = 0.38$

Coefficient <Effort Factor> = 2.4

SLOC = 5000

$$\begin{aligned}\text{Effort} &= \text{PM} = \text{coefficient} * (\text{SLOC}/1000)^P \\ &= 2.4 * (5000/1000) ^{1.05} \\ &= 13.06 \text{ (person month)}\end{aligned}$$

$$\begin{aligned}\text{Development Time} &= \text{DM} = 2.50 * (\text{PM})^T \\ &= 2.50 * (13.06) ^{0.38} \\ &= 6.64 \text{ (week days)}\end{aligned}$$

$$\begin{aligned}\text{Required number of people} &= \text{ST} = \text{PM}/\text{DM} \\ &= 13.06 / 6.64 \\ &= 1.97 \\ &= 2 \text{ (peoples)}\end{aligned}$$

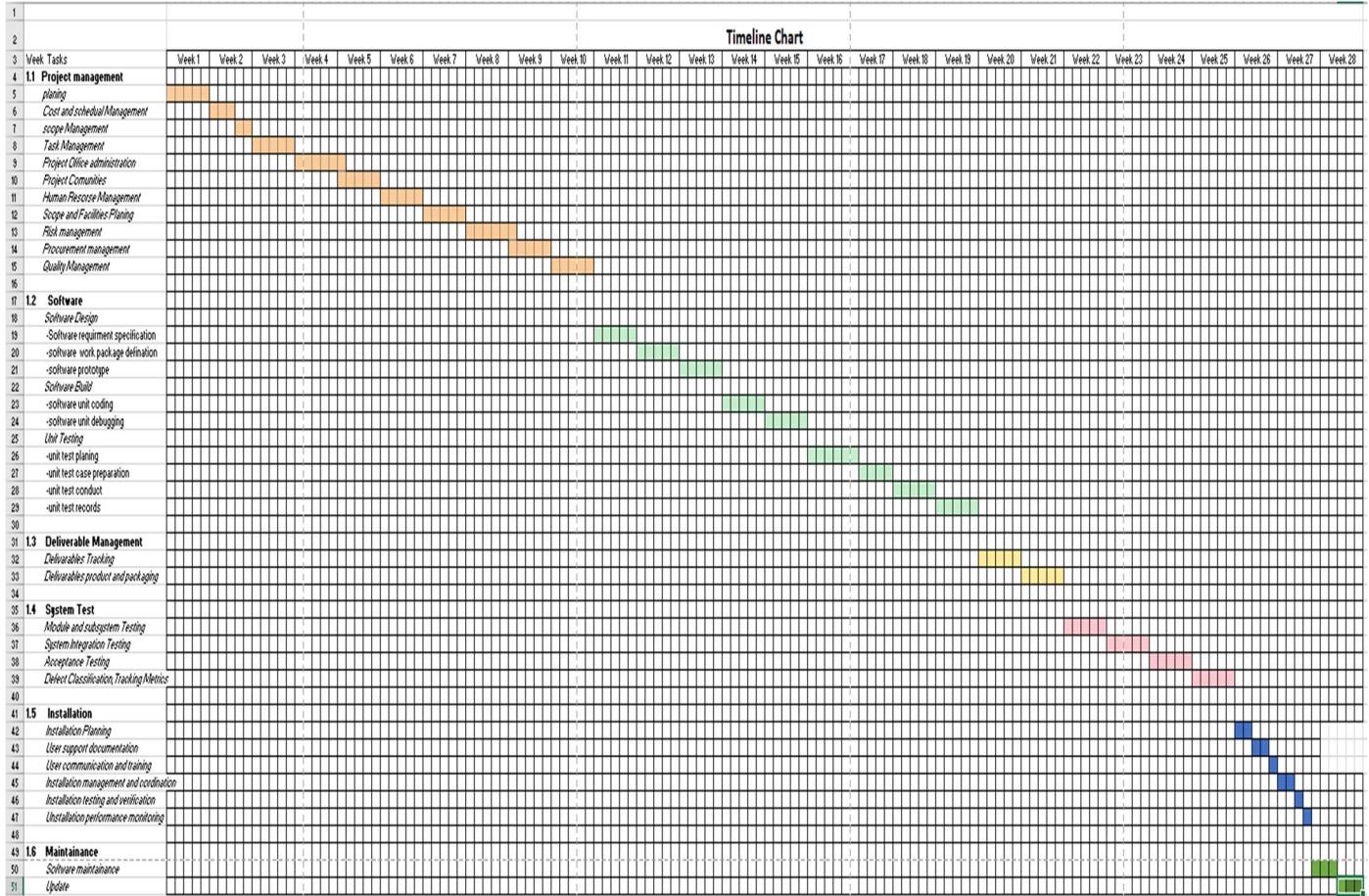
Timeline Chart:

Phases		Pre-Phase			Development phase					
Sprints					Sprint 1			Sprint 2		
Weeks \ Persons		1	2	3	4	5	6	7	8	9
A1 : Shahriar		■				■				
A2 : Ibrahim			■		■			■		
A3 : Raihan				■			■			
A4: Mushficol								■		■

Phases	Development phase						Post-Phase			
Sprints	Sprint 6			Sprint 7						
Weeks \ Persons	19	20	21	22	23	24	25	26	27	28
C1 : Shahriar							■		■	
C2 : Ibrahim		■						■		
C3 : Raihan				■			■	■		
C4: Mushficol	■		■		■					■

Details
SCRUM Team: Shahriar, Ibrahim, Raihan, Mushficul
<u>Pre-Phase:</u>
A1: Planning (W1)
A2: Specification and Analysis (W2)
A3: Design (W3)
<u>Sprint 1:</u>
A2: Specification and Analysis (W4)
A1: Development (W5)
A3: Testing (W6)
<u>Sprint 2:</u>
A4: Specification and Analysis (W7)
A2: Development (W8)
A4: Testing (W9)
<u>Sprint 3:</u>
B3: Specification and Analysis (W10)
B4: Development (W11)
B3: Testing (W12)
<u>Sprint 4:</u>
B1: Specification and Analysis (W13)
B2: Development (W14)
B4: Testing (W15)
<u>Sprint 5:</u>
B2: Specification and Analysis (W16)
B1: Development (W17)
B3: Testing (W18)
<u>Sprint 6:</u>
C4: Specification and Analysis (W19)
C2: Development (W20)
C4: Testing (W21)
<u>Sprint 7:</u>
C3: Specification and Analysis (W22)
C4: Development (W23)
C3: Testing (W24)
<u>Post-Phase:</u>
C1:C3 Integration (W25)
C2: System Testing (W26)
C1: User Acceptance Testing (W27)
C4: Documentation, Install and Delivery (W28)

Timeline Chart 2:



For Better Quality (Main Excel File Link):

<https://docs.google.com/spreadsheets/d/1Xpgw54Nb7NnQ1-FjQzpCRgCpGb61DFqE/edit?usp=sharing&ouid=105555612495707718651&rtpof=true&sd=true>

For Better Quality (Main Image Link):

https://drive.google.com/file/d/1-nuwBrEa0_MtLEiXenP6HUGO_ovhEeGx/view?usp=sharing

EVA:

TASK	PLANNED EFFORT	ACTUAL EFFORT
1		5
2		3
3		3
4		2
5	BCWP = 47	5
6	5	6
7	5	7
8	5	7
9	5	5
10	5	6
11	5	4
12	5	7
13	5	-
14	5	-
15	5	-

Total task = 38

Total effort estimated = $13.06 \times 20 = 261$ person-days

BAC = 261

SPI = BCWP/BCWS = $47/72 = 0.65278$

SV = BCWP-BCWS = $47-72 = -25$ person-day

CPI = BCWP/ACWP = $47/49 = 0.95918$

CV = BCWP-ACWP = $47-49 = -2$ person-day

% Schedule for completion = BCWS/BAC = $72/261$

= 27.56% [% of work scheduled to be done at this

time]

% Complete = BCWP/ BAC = $47/261 = 18.01\%$ [% of work completed at this time]

RISK TABLE:

Risks	Category	Probability	Impact	RMMM
Less use than planned	PS	60%	3	
Customer changing requirements	PS	50%	2	
Technology not meeting expectations	TE	50%	1	
Staff inexperienced	ST	40%	2	
Lack of staff	ST	40%	1	
System efficiency	DE	40%	1	
Security flaws	TE	20%	4	
Larger number of user than planned	PS	40%	4	
Data loss	PS	20%	4	
Cost may be high	BU	30%	2	

Impact values:

- 1- Catastrophic
- 2- Critical
- 3- Marginal
- 4- Negligible

CONCLUSION:

This project focuses on implementing an integrated solution for AIUB parking management system. The planned system has several advantages including detecting parking spaces using the system's database, calculating the time of vehicle entry/exit, and calculating the expected cost. Adopting a parking management system considerably reduces the number of times consumed in seeking the parking zone,

renders valuable information upon the availability of the parking area, and generates an accurate mapping of the parking space.

THE END
