

American International University-Bangladesh (AIUB)  
**Department of Computer Science  
Faculty of Science &Technology (FST)  
Summer 19\_20**

**Section: H  
Group No: 04**

**AIUB Parking Management System**

A software Engineering project submitted

By

|  |  |  |
| --- | --- | --- |
| SN | Student Name | Student ID |
| **19** | **Pappu Saha** | **18-36339-1** |
| **27** | **Chowdhury, MD. Saif** | **18-36828-1** |
| **33** | **Sobhan, MD. Abdus** | **18-37855-2** |
| **34** | **Ami, Nooren Zabran** | **18-37915-2** |
|  |  |  |

The project will be Evaluated for the following Course Outcomes

|  |  |
| --- | --- |
| CO3: Choose appropriate software engineering model in a software development   environment | Total Marks |
|  |
| Project Background Analysis (problem, needs, goal, benefits, etc.) [5Marks] |  |
| Appropriate Process Model Selection and Argumentation with Evidence [5Marks] |  |
| Completeness, Spelling, Grammar and Organization of the Answer [5Marks] |  |
|  | |
| CO4: Explain the roles and their responsibilities in the software project   management activities | Total Marks |
|  |
| Content Knowledge (e.g. System Requirements, System Design) [5Marks] |  |
| Project Role identification and Responsibilities descriptions [5Marks] |  |
| Presentation Delivery and Defense [5Marks] |  |

**Problem Background**

**Introduction**

AIUB is a reputed university in Bangladesh. It has been need a unique and smart parking space management system. We think, if we maintain the vehicles in a proper way, we can reduce traffic jam in the Kuratoli road and also in front of our campus. So, we need a proper management system to maintain the vehicles.

**Problem Background**

There are fifteen thousand students and around five hundred faculty members and employees in our university. There are around seven hundred vehicles passing in the campus. And AIUB has some own vehicles also. So, every day we are facing traffic jam problem in front of our main gate and also Kuratoli road.

**Project Objectives**

The objective of this project is to build a 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 slot and determine if the parking slot is full or not.

**Benefits**

Every day we are facing traffic jam problem to go to our campus in Kuratoli road. We think, if we maintain the vehicles in a proper way, we can reduce traffic jam in the Kuratoli road and also in front of our campus so that, students, faculties and employees can get free parking and it can also reduce traffic jam in Kuratoli road. Besides students, faculties, employees and university authority can be benefited in this such way:

* Maintain records in short time of period
* Determines the parking area is full or not.

**Basic Functionalities**

Our proposed solution will be based on some functionalities. There have been database system where stores all information of the students, faculties, employees. There have admin panel. We all the group members are discussing about it for better solution also better option for this system. We will work for better functionalities, modern technologies, also useful for our university.

**Requirement Analysis**

1. **Software Login**

**Functional Requirements**

* 1. The user login into the software with their username and password.
  2. If the login successful the main home page in this software will be displayed.
  3. If the user forget the username and password so they can be reset password using their valid email address and phone number.
  4. If the user input 3 times wrong password, the system will block the user account login for half one hour

**Priority level**: High

**Precondition:** User have valid username and password.

**Cross-references: None**

1. **User Information Management**

**Functional Requirements**

* 1. We will create users account according to the users vehicle information.

2.2 We will give each user a user parking ID card that includes all user information.

2.3 When users pass main gate with the vehicle, they will scan their parking ID card into the machine.

2.4 In this time the software will store all the information about the users in the database.

**Priority level**: High

**Precondition**: User need to bring their parking ID card.

**Cross-references: 5.1, 5.2, 5.3**

1. **Vehicle Parking Details (Secondary Gate)**

**Functional Requirements**

3.1 There will be separate parking slot for each type of vehicles.

3.2 It will ensure that how many vehicles available at a time.

3.3 There will be different parking slot for students, teachers, authorities and guests. If user valid for specific slot then “Park” otherwise “Try another slot”.

**Priority level**: Medium

**Precondition**: The vehicles must be registered according to the vehicles and user’s information.

**Cross-references: None**

1. **Admin**

**Functional Requirements**

4.1 Monitoring user information by maintaining accurate database of all parking and access card for tracking and reporting.

4.2 If the user is valid based on his/her information then “activate” card otherwise “reject” and report to the user.

4.3 If the card is activated by admin then user can park.

4.4 Admin will able to see the payment status. If paid then allow to park otherwise report to the user for payment. After reporting 2 times if user doesn’t pay then admin can deactivate the user card.

**Priority level**: High

**Precondition**: Admin must have the rights to access the whole system.

**Cross-references: 1.1, 1.2, 2.1, 6.1**

1. **Security Management (Primary Gate)**

**Functional Requirements**

5.1 The security guard also checks the user’s parking ID Card.

5.2 After checking user’s parking ID card, the security guard will check the free available space.

5.3 If the space is free for parking, the guard will scan the car and give access for the parking.

**Priority level**: High

**Precondition**: The security will must verified the registered vehicles.

**Cross-references: 2.1, 3.2, 3.3**

1. **Payment System**

**Functional Requirements**

6.1 The payment transaction will be done using user’s parking ID card.

6.2 The payment slip can be get from the account department using user’s parking ID card.

6.3 Then the payment will be received in the bank in our campus.

**Priority level**: Medium

**Precondition**: Payment system must be pay using user Parking ID card.

**Cross-references: 2.1**

**Non-Functional Requirements:**

1. **Interoperability**

Our desktop application system can exchange data or service with other system to provide one-stop services to the customers. Our System will exchange user information with the register office payment software system for user's payment purpose. Payment software of the register office will exchange valid user information with our system.

1. **Usability**

In this system, a valid registered user shall be allowed to park his/her vehicle after confirming the slot is available. The parking information of this vehicle will take a place in database of this system. And this activity will take the duration not more than 1 minutes.

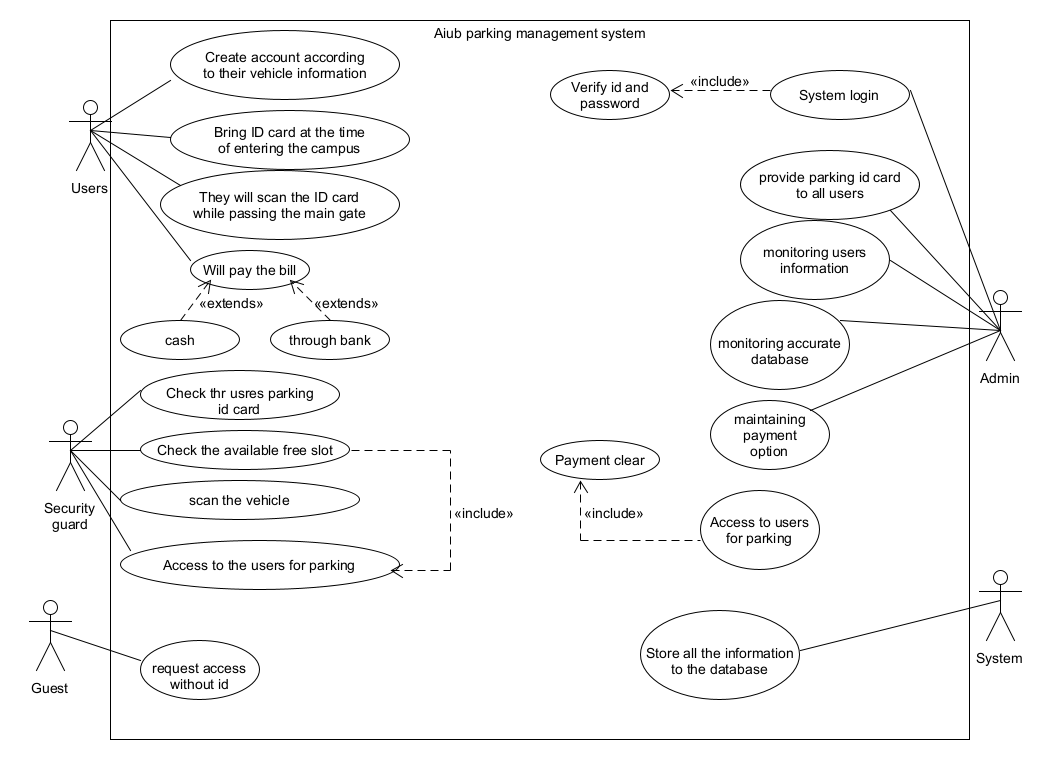
**Project Requirements:**

1. The application will be written in JAVA programming language, because Java is object-oriented. Java is platform-independent. Java was designed to be easy to use and is therefore easy to write, compile, debug, and learn than other programming languages.
2. The total size of our final application should not be more than 150-200 MB.
3. There will be only one application for the University Parking System. So, it is only for the University Authority.
4. Around 3 months (90 days) can be needed to complete this application.
5. Approximately 1 lakh (BDT) can be needed to build this application.

**Use case diagram**

In AIUB parking management system, admin login into the software with the valid username and password. They provide each users a user parking ID card that includes all user information. They monitoring user information by maintaining accurate database of all parking and access card for tracking and reporting. Admin will able to see the payment status. If paid then allow to park otherwise report to the user for payment. Users create users account according to their vehicle information. They Bring ID card at the time of entering the campus. When users pass main gate with the vehicle, they will scan their parking ID card into the machine. They can payment cash or through bank. In this time all information store in the system. There will be security guard. They also check the user’s parking ID card. After checking, they will check the available free slot. Then they scan the vehicle and given access for the parking. The guest also parking their vehicle without ID access.

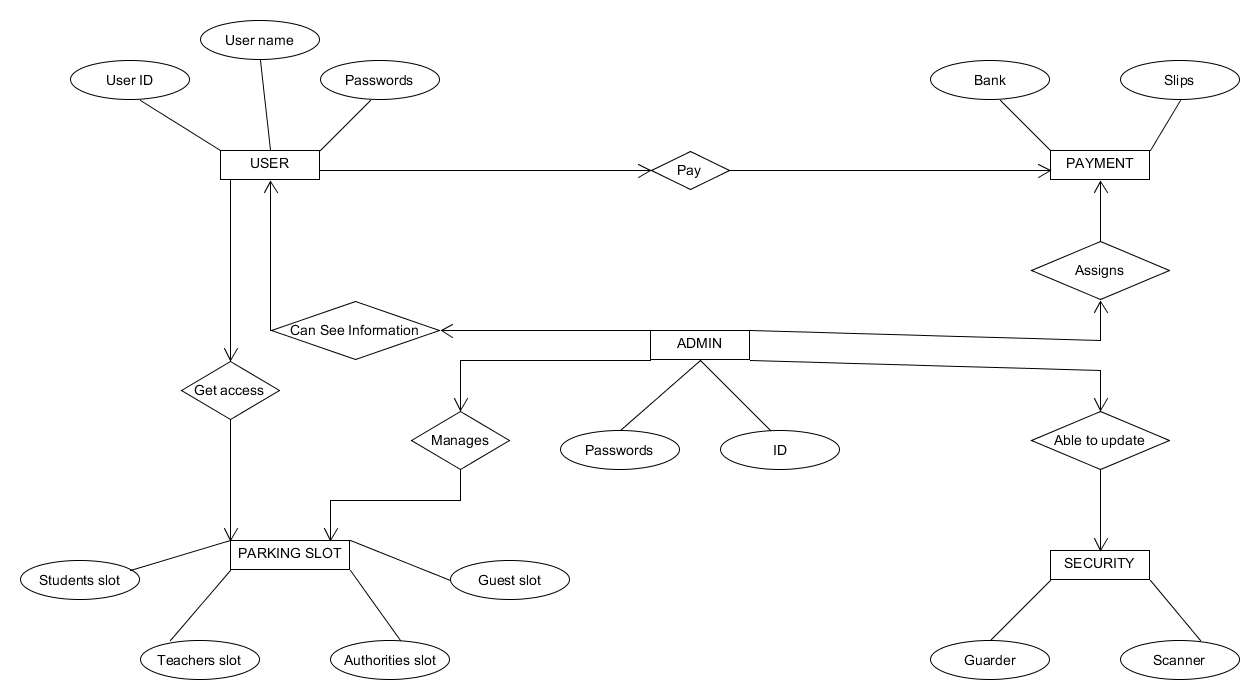
**Use Case Diagram:**



**ER Diagram**

In AIUB parking management, user can get access to the parking slot for parking. Users have their Id, name and unique passwords which are used to get slot for their vehicles. An admin can see all the information of users whoever has own Id and passwords. Parking space has separate slot for each kind of user. Slots are divided for students, teachers, authorities and Guests. All the slots are managed by admin. Users are supposed to pay for getting access to park. Payment should be paid by users as there are some specific bank under payment section and it provides payment slips to the user. Admin must assign the payment details. The system ensures the security for users. Security section has guarder and scanner. The security can be updated by Admin.

**ER Diagram**:



**Class Diagram**

In this system, we developed some class which can be used for our project in class diagram. Class diagrams can also be used for data modeling. In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations and the relationships among objects.

Here we have been fixed six class variables for this project. Such as- Customer/Student, Vehicle, Security, Parking Space, Admin, Payment. Each classes have some attributes. The attributes have been also data types.

A class notation consists of three parts:

1. **Class Name**
2. **Class Attributes**
3. **Class Operations**

**1**. **Class Name**: Customer/Student

**Attribute**: Name: (String), ID: (String), Phone: (int), Parking\_Slot: (String), Start\_Date: (String), End\_Date: (String), Address: (address), Vehicle\_Type: (String)

**Function:** getRole(), setRole(String), getAllotedParkingSpace(), getBill(), setDetails()

**2.** **Class Name**: Vehicle

**Attribute**: Vehicle\_Type: (String), Vehicle\_Name: (String), Vehicle\_Plate\_No: (String), Vehicle\_Colour: (String), Entry\_Time: (String), Exit\_Time: (String)

**Function:**

**3.** **Class Name**: Security

**Attribute**: Name: (String), ID: (String), Phone: (int), Start\_Time: (String), End\_Time: (String)

**Function:** getSummary(), setSecurity()

**4.** **Class Name**: Parking Space

**Attribute**: Parking\_Slot: (String), Is\_Occupied: (Boolean), Is\_Student/Customer: (Boolean), Is\_Guest: (Boolean), Is\_AIUB's\_Vehicle: (Boolean)

**Function:** getParkingSlot(), setSlot()

**5.** **Class Name**: Admin

**Attribute**: Admin\_Name: (String), Password: (String)

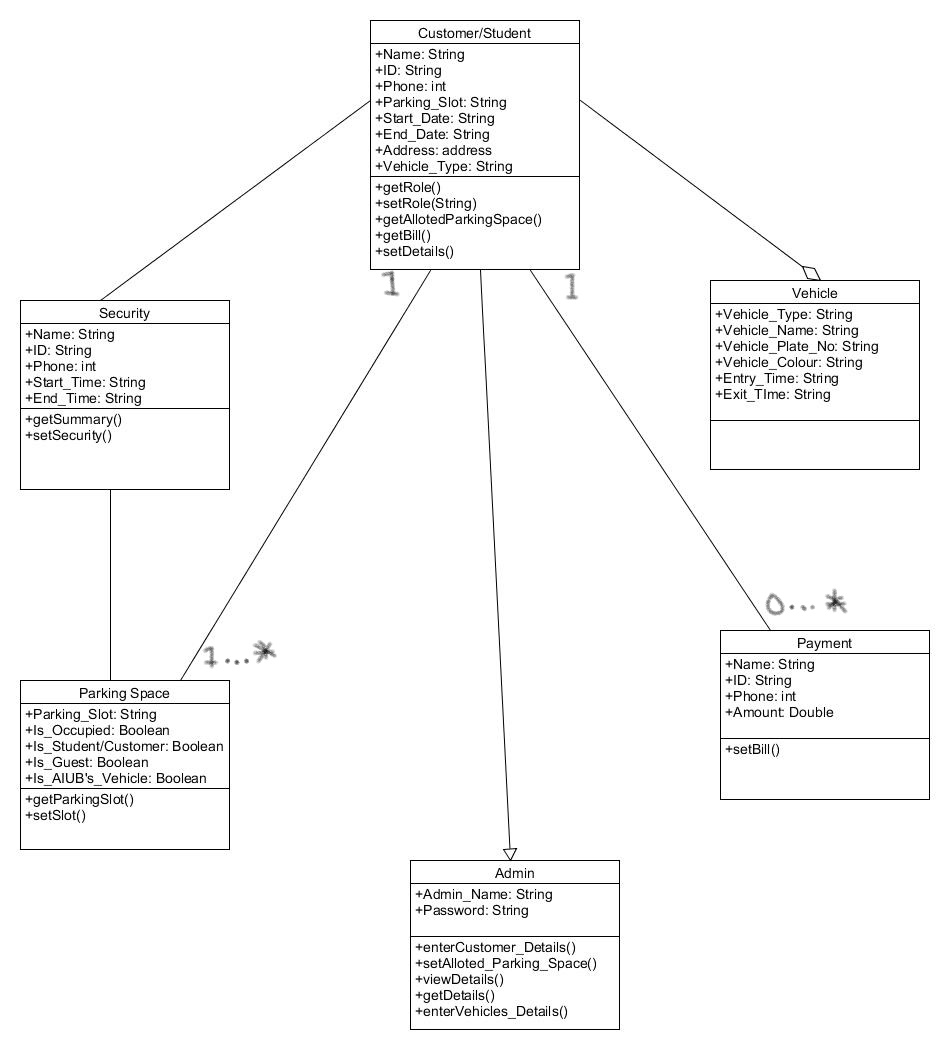
**Function:** enterCustomer\_Details(), setAlloted\_Parking\_Space(), viewDetails(), getDetails(), enterVehicles\_Details()

**6.** **Class Name**: Payment

**Attribute**: Name: (String), ID: (String), Phone: (int), Amount: (Double)

**Function:** setBill()

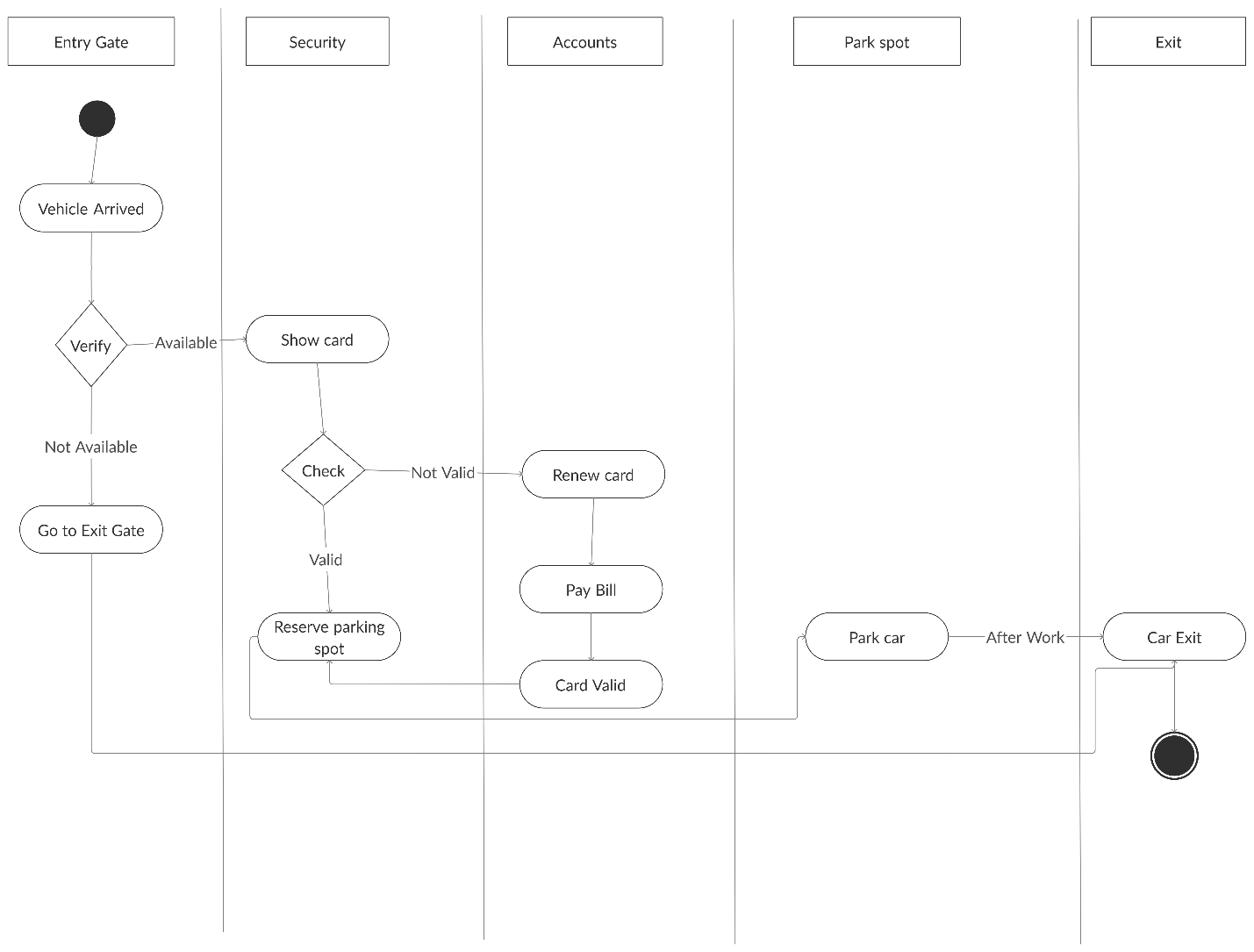
**Class Diagram:**



**Activity Diagram**

At first user vehicle arrived into entry gate. Then if parking slot available user have to show there id card in security guard. Otherwise if parking not available then go to exit gate. Security guard check user id card as it is valid or not valid. If card is not valid then user have to go on accounts for renew card. For renew card user have to pay their monthly bills. After paying bills user card will be valid. On the other hand if the card is valid then security guard will reserve the parking spot. After reserving parking spot user can park their vehicle on that particular spot. After work user exit their vehicle on exit point.

**Activity Diagram:**



**Process Model**

As we know the RAD (Rapid Application Development) model is an incremental model and it maintains the time boxed, we selected this model to develop our system. In RAD model, the functions are developed in parallel as prototypes and in each increments requirements stays frozen. The developments are delivered and then assembled into a working prototype. After delivering a fully functional system within 90 days, the user can give feedback if the system need any update or not. This model is flexible and adaptable to changes as well as it is useful when we have to reduce the overall project risk. In addition, each increment in RAD delivers highest priority functionality to client.

So we are going to develop our system in incremental process so that we can maintain the time boxed and get feedback from the users. Depending on this we will able to make our system useable.

* **Why not Waterfall Model?**

Waterfall model is a linear sequential model where next phase starts only after completion of the previous phase. It is very simple to understand and easy to use because of its systematic well-defined. Waterfall model can be used when requirements are not changing frequently, requirement is clear, environment is stable, resources are available and trained. Besides, high amounts of risk and uncertainty, poor model for long and ongoing projects, cannot accommodate changing requirements. So, waterfall model is not appropriate for our parking management system project.

* **Why not Agile Model?**

**Agile development model** is also a type of Incremental model. Software is developed in incremental, rapid cycles. This results in small incremental releases with each release building on previous functionality. Each release is thoroughly tested to ensure software quality is maintained. It is used for time critical applications. In Agile Model, there is lack of emphasis on necessary designing and documentation. Projects can become ever-lasting because there's no clear end. So, we think it is not appropriate for our project.

* **Why Not V-Model?**

**V-Model** is an SDLC model that has a testing phase corresponding to every development stage in the waterfall model. It is pronounced as the "vee" model. The V-model is an extension of the waterfall model. V model Testing is done in parallel to development. It is also called a Validation and Verification Model. Even though v-model is beneficial, there are few disadvantages attached to it. Such as, Very rigid and least flexible. Software is developed during the implementation phase, so no early prototypes of the software are produced. If any changes happen in midway, then the test documents along with requirement documents has to be updated. For this reason we do not use v-model.

* **Why Not Iterative Model?**

The iterative model is a particular implementation of a software development life cycle (SDLC) that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. Even though, iterative model is extremely beneficial, there are few drawbacks and disadvantages attached to it, such as, each phase of an iteration is rigid with no overlaps. Also, system architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle. It is not suitable for smaller projects. So we do not use iterative model.

* **Why not Saw-Tooth model?**

In the saw-tooth model the prototypes are created and shown to the client for validation. The prototypes are present right after the analysis phase and in between the design and the implementation phase. By making sure that the client is getting an insight into the progress, checkpoints should ensure that development is going in the right direction.

The major benefit comes from the validations in between the critical phases. The client is involved which is supposed to guarantee that the project will become a success.

The huge problem is the time consumption and costs associated with presenting the prototypes to the client. Depending on the complexity of the project, the time and costs might be considerable.

* **Why not Rational Unified process model (RUP)?**

The Rational Unified Process is by far the most complex model. The RUP looks at a project in terms of cycles. One cycle consists of four phases: inception, elaboration, construction and transition. On the other hand process details are expressed in general terms, providing minimal guidance and requiring local customization. Also heavy documentation can be expensive. For this reason we select RAD model.

**Project Roles**

**Programmer:** keeps the program code as simple and definite as possible.

**Tester:** helps write functional tests, also run functional tests regularly, broadcast test results and maintain testing tools.

**Tracker**: gives feedback and he traces the estimates made by the team and gives feedback on how accurate they are in order to improve future estimations.

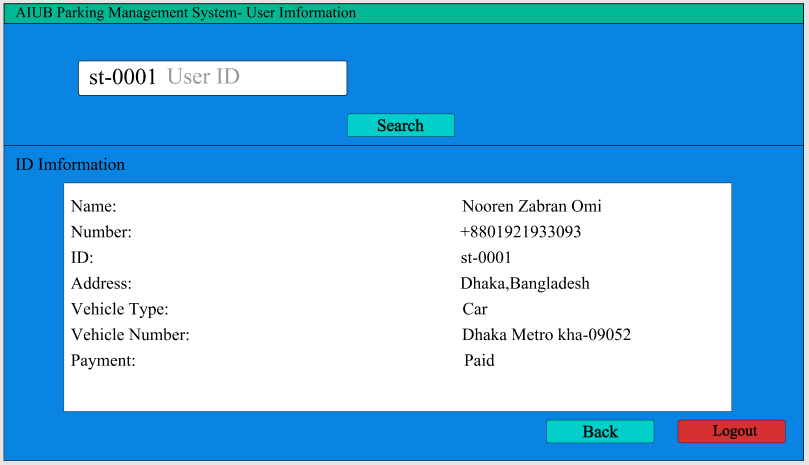
**Consultant:** Consultant is an external member possessing the specific technical knowledge needed**.**

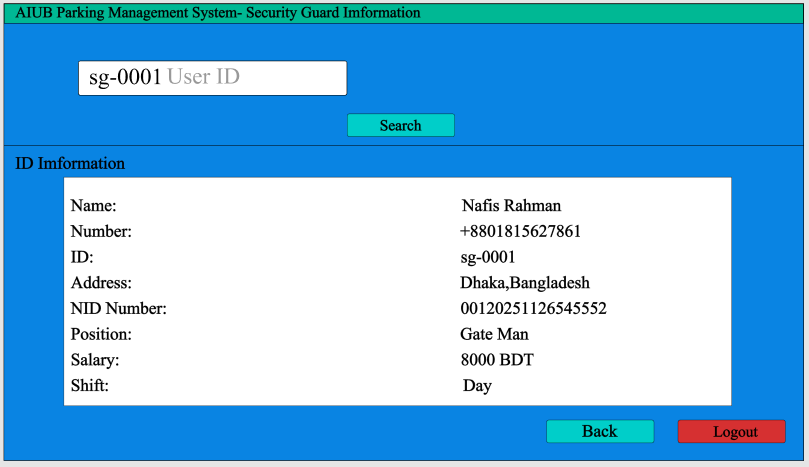
**Manager:** Manager makes the decisions (time, cost, resource, schedule, risk management)

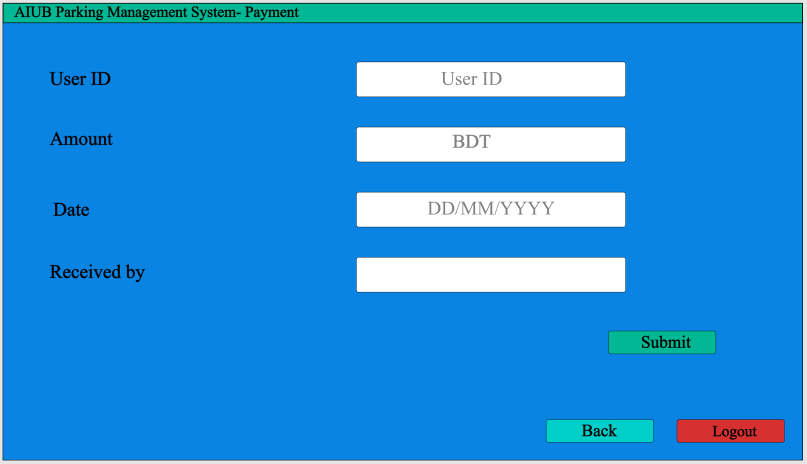
**Interfaces**

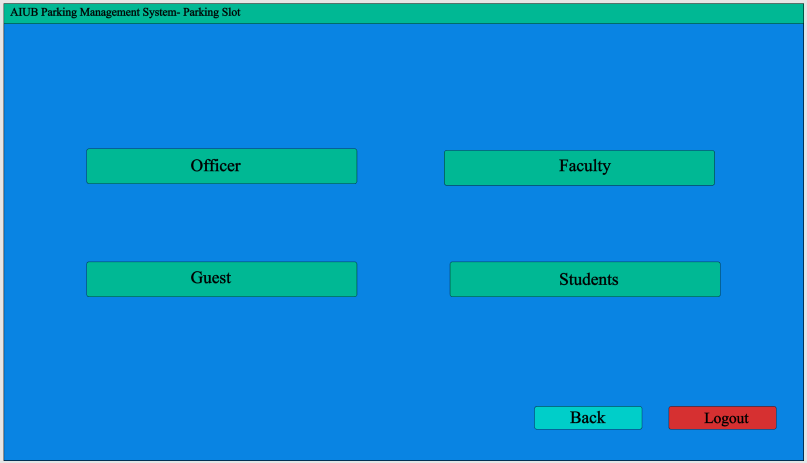
****

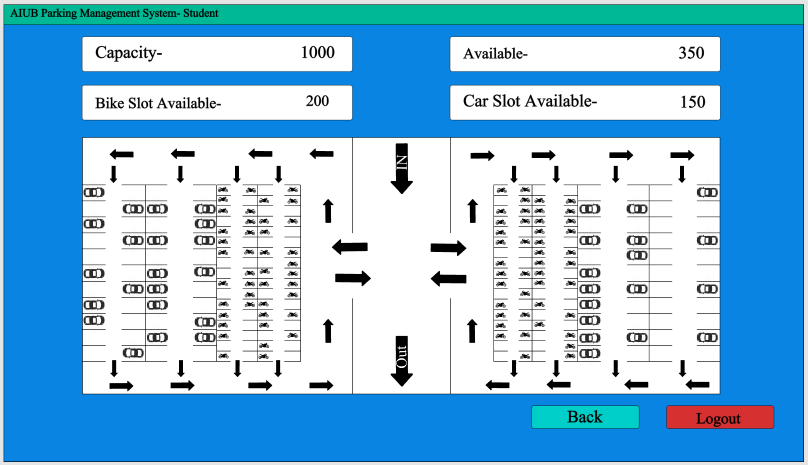
****

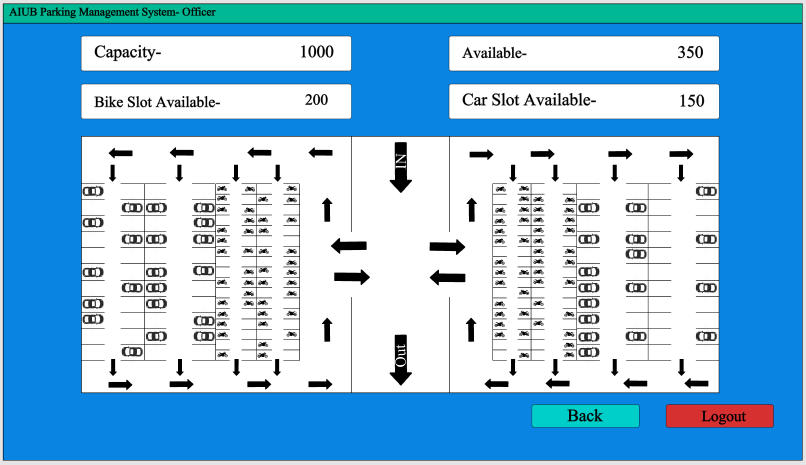
****

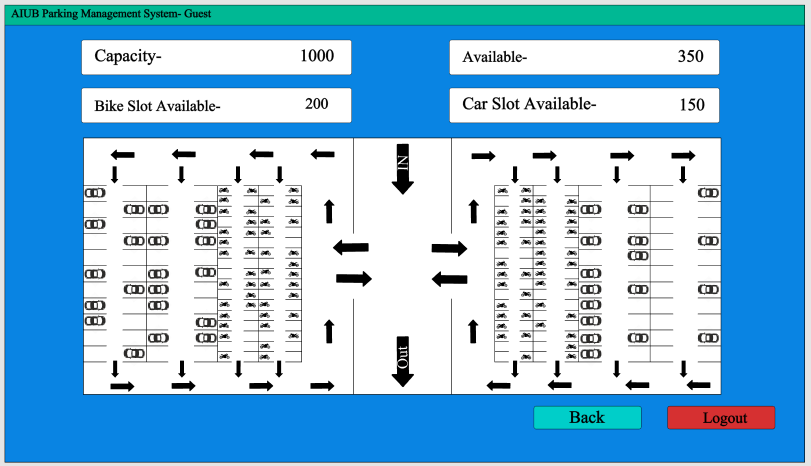
****

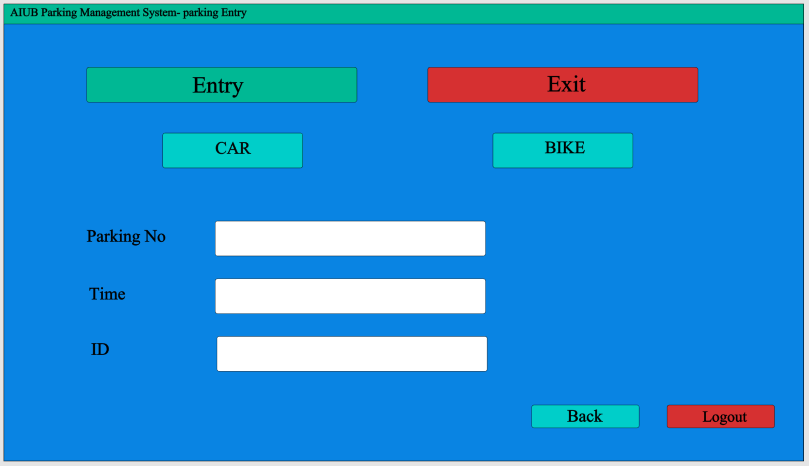
****

****

****

****

****

****

**Test Case**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: AIUB Parking Management System | | Test Designed by: Pappu Saha | | |
| Test Case ID: FR\_1 | | Test Designed date: 01/09/2020 | | |
| Test Priority (Low, Medium, High): High | | Test Executed by: | | |
| Module Name: Software login | | Test Execution date: | | |
| Test Title: Verify login with valid username and password | | | | |
| Description: Test website login page | | | | |
| Precondition (If any): User have valid username and password | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status  (Pass/Fail) |
| 1. Go to website 2. Enter username Input 3. Enter password 4. Click submit | UserName: AiubParkingManagement  Password: AIUB1994 | User should login into the system. |  |  |
| 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: Pappu Saha | | |
| Test Case ID: FR\_2 | | Test Designed date: 01/09/2020 | | |
| Test Priority (Low, Medium, High): High | | Test Executed by: | | |
| Module Name: User Information management | | Test Execution date: | | |
| Test Title: Verify user parking ID card with software database | | | | |
| Description: Test parking ID card | | | | |
| Precondition (If any): User need to bring their parking ID card | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status  (Pass/Fail) |
| 1.Go the software  2.Login with valid username and password  3.Check parking id card when user scan their id card into the machine | UserName: AiubParkingManagement  Password: AIUB1994 | Software user should login into the system and verify users parking ID card. |  |  |
| Post Condition: The user entry details are stored in the database. | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: AIUB Parking Management System | | Test Designed by: Chowdhury, Md. Saif | | |
| Test Case ID: FR\_3 | | Test Designed date: 01/09/2020 | | |
| Test Priority (Low, Medium, High): High | | Test Executed by: | | |
| Module Name: Vehicle Parking Details (Secondary Gate) | | Test Execution date: | | |
| Test Title: Verify vehicle parking system | | | | |
| Description: After login Security guard can see how many vehicles slot available that time. Seeing the available Parking slot security guard ensure the parking slot to user. After ensuring user can park that selected parking place. | | | | |
| Precondition (if any) : The vehicles must be registered according to the vehicles and user’s information | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status  (Pass/Fail) |
| 1.Login to the software  2.Enter parking slot  3.Select available slot  4.Click Confirm | Username: AiubParkingManagement  Password: AIUB1994 | Booking parking slot |  |  |
| Post Condition: Valid User park their vehicle on selected slot if parking available. | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: AIUB Parking Management System | | Test Designed by: Md Abdus Sobhan | | |
| Test Designed date: 01/09/2020 | | |
| Test Case ID: FR\_4 | |
| Test Priority (Low, Medium, High): High | | Test Executed by: | | |
| Module Name: Admin | | Test Execution date: | | |
| Test Title: Verify Admin with valid ID and password. | | | | |
| Description: Test whether Admin can log into the system and the Admin information is valid based on the database. | | | | |
| Precondition (if any): Admin must have the rights to access the whole system. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status  (Pass/Fail) |
| 1. Go to the website  2. Enter ID  3. Enter password  4. Click submit | Username: AiubParkingManagement  Password: AIUB1994 | Admin should login into the application |  |  |
| Post Condition: Admin is validated based on the database information and successfully login to this system. | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: AIUB Parking Management System | | Test Designed by: Ami, Nooren Zabran | | |
| Test Case ID: FR\_5 | | Test Designed date: 01/09/2020 | | |
| Test Priority (Low, Medium, High): High | | Test Executed by: | | |
| Module Name: Security Management | | Test Execution date: | | |
| Test Title: Checks the user parking ID card. | | | | |
| Description: the security guard will check user ID card. | | | | |
| Precondition (if any) : The security will must verified the registered vehicles. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status  (Pass/Fail) |
| 1. Security Login 2. Click User Info 3. Input User ID number 4. Confirm User validation | Security Name: Abul Kashem  Password: abul1992  User ID: 1425 | Security guard will check user ID card and also validation |  |  |
| Post Condition: after verification, the guard will scan the car and give access for the parking. | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Name: AIUB Parking Management System | | Test Designed by: Ami, Nooren Zabran | | |
| Test Case ID: FR\_6 | | Test Designed date: 01/09/2020 | | |
| Test Priority (Low, Medium, High): High | | Test Executed by: | | |
| Module Name: Payment System | | Test Execution date: | | |
| Test Title: Checks the user payment issue. | | | | |
| Description: The payment transaction will be done using user’s parking ID card and payment slip can be get from the account department using user’s parking ID card. Then the payment will be received in the bank in our campus. | | | | |
| Precondition (if any) : Payment system must be pay using user Parking ID card. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status  (Pass/Fail) |
| 1. Login 2. Click User Account Details 3. Input User ID number 4. Confirm payment slip | Name: admin  Password: admin | Payment slip will be generated. |  |  |
| Post Condition: after payment, the vehicle will be able parking for next semester. | |  | | |