

|  |  |
| --- | --- |
| **ASSIGNMENT- 1** | |
| **Course Code** | CSC402A |
| **Course Name** | Web Architecture and Application development |
| **Programme** | B. Tech |
| **Department** | Computer Science and Engineering |
| **Faculty** | FET |

#### 

|  |  |
| --- | --- |
| **Name of the Student** | K Srikanth |
| **Reg. No** | 17ETS002124 |
| **Semester/Year** | 3rd / 6th Semester |
| **Course Leader/s** | Mr. Kishore S. M. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Declaration Sheet | | | | | | | | |
| Student Name | K Srikanth | | | | | | | |
| Reg. No | 17ETCS002124 | | | | | | | |
| Programme | B. Tech | | | | | Semester/Year | 3rd / 6th Semester | |
| Course Code | 19CSC215A | | | | | | | |
| Course Title | Web Architecture and Application development | | | | | | | |
| Course Date | 18/03/2021 | | to | | 29/06/2021 | | | |
| Course Leader | Mr. Kishore S. M. | | | | | | | |
| **Declaration**  The assignment submitted herewith is a result of my own investigations and that I have conformed to the guidelines against plagiarism as laid out in the Student Handbook. All sections of the text and results, which have been obtained from other sources, are fully referenced. I understand that cheating and plagiarism constitute a breach of University regulations and will be dealt with accordingly. | | | | | | | | |
| Signature of the Student | | K Srikanth | | | | | Date | 22/04/2021 |
| Submission date stamp  (by Examination & Assessment Section) | |  | | | | | | |
| Signature of the Course Leader and date | | | | Signature of the Reviewer and date | | | | |
|  | | | |  | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Faculty of Engineering and Technology | | | |
| Ramaiah University of Applied Sciences | | | |
| Department | Computer Science and Engineering | Programme | B. Tech |
| Semester/Batch | 06/2018 | | |
| Course Code | CSC402A | Course Title | Web Architecture and Application development |
| Course Leader | Mr. Kishore S. M. | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Assignment | | | | | | | | | | | | |
| Register No. | | | K Srikanth | | | Name of the Student | | | | 17ETCS002124 | | |
| Sections |  | **Marking Scheme** | | | | | | Marks | | | | |
| Max  Marks | | | First Examiner Marks | Moderator |
| Part 1 |  | | | | | | | | | | | |
| 1.1 | Functional and non-functional requirements | | | | | | 5 | | |  |  |
| 1.2 | Identification and design of the entity classes using E-R diagrams | | | | | | 5 | | |  |  |
| 1.3 | Design of UML interaction sequence diagrams | | | | | | 8 | | |  |  |
| 1.4 | Design of Algorithm/ Flowchart | | | | | | 7 | | |  |  |
|  | **Part- 1 Max Marks** | | | | | | **25** | | |  |  |
|  | | | | | | | | | | | | |
| **Course Marks Tabulation** | | | | | | | | | | | | | |
| **Component- CET B Assignment** | | | | **First Examiner** | **Remarks** | | **Second Examiner** | | **Remarks** | | | | |
| 1 | | | |  |  | |  | |  | | | | |
| 2 | | | |  |  | |  | |  | | | | |
| **Total Marks** | | | |  |  | |  | |  | | | | |
| **Signature of First Examiner Signature of Second Examiner** | | | | | | | | | | | | | |

**Scenario:** **(25 Marks)**

In an online Smartphone shopping Web application, users can register and login to the web application. The online smartphone application maintains account details for each user (user ID, user name, phone number, shipping address and items purchased etc.). The user may select any item from the list of available smartphones or can search for all the available smartphones. It is assumed that an item purchased is reserved and made available to the user offline.

**Answer the following questions: (25 Marks)**

List all the functional and non-functional requirements for the given scenario. Identify and design entity classes using E-R diagrams and UML sequence diagrams. Document the following in the report:

* 1. **Functional and non-functional requirements**

**Functional requirements**

**Table 1.1: Functional requirement 1**

|  |  |
| --- | --- |
| Requirement Tag | FR1 |
| Requirement Description | The system should have an interface where user can be able to register |
| Dependent on Requirements | - |
| User/System interacting with the requirement | User |

**Table 1.2: Functional requirement 2**

|  |  |
| --- | --- |
| Requirement Tag | FR2 |
| Requirement Description | The system should have an interface where user can be able to login |
| Dependent on Requirements | FR1 |
| User/System interacting with the requirement | User |

**Table 1.3: Functional requirement 3**

|  |  |
| --- | --- |
| Requirement Tag | FR3 |
| Requirement Description | The system should be able to store user data. |
| Dependent on Requirements | FR1 |
| User/System interacting with the requirement | System |

**Table 1.4: Functional requirement 4**

|  |  |
| --- | --- |
| Requirement Tag | FR4 |
| Requirement Description | The system should have an interface where user can be able to see all the products |
| Dependent on Requirements | - |
| User/System interacting with the requirement | User |

**Table 1.5: Functional requirement 5**

|  |  |
| --- | --- |
| Requirement Tag | FR5 |
| Requirement Description | The system should be able to show if the product is available or unavailable |
| Dependent on Requirements | FR4 |
| User/System interacting with the requirement | System |

**Table 1.6: Functional requirement 6**

|  |  |
| --- | --- |
| Requirement Tag | FR6 |
| Requirement Description | The system should have an interface where the user can reserve a product. |
| Dependent on Requirements | FR4 |
| User/System interacting with the requirement | User |

**Table 1.7: Functional requirement 7**

|  |  |
| --- | --- |
| Requirement Tag | FR7 |
| Requirement Description | The system should have an interface where the user can see his/her order summary. |
| Dependent on Requirements | FR4, FR5, FR6 |
| User/System interacting with the requirement | User |

**Table 1.8: Functional requirement 8**

|  |  |
| --- | --- |
| Requirement Tag | FR8 |
| Requirement Description | The system should have an interface where admin can add or update or delete the product from the website |
| Dependent on Requirements | FR4 |
| User/System interacting with the requirement | Admin |

**Non-Functional Requirements**

**Table 2.1: Non-Functional requirement 1**

|  |  |
| --- | --- |
| Requirement Tag | NFR2 |
| Requirement Description | The system should be able to manage traffic of incoming requests |
| Dependent on Requirements | - |
| User/System interacting with the requirement | System |

**Table 2.2: Non-Functional requirement 2**

|  |  |
| --- | --- |
| Requirement Tag | NFR2 |
| Requirement Description | The system should have easy interface where customer can understand the UI/UX elements. |
| Dependent on Requirements | FR1, FR2, FR4, FR6 |
| User/System interacting with the requirement | User |

**Table 2.3: Non-Functional requirement 3**

|  |  |
| --- | --- |
| Requirement Tag | NFR2 |
| Requirement Description | The system should have an interface where product details are detailly described |
| Dependent on Requirements | FR4 |
| User/System interacting with the requirement | System |

**Table 2.4: Non-Functional requirement 4**

|  |  |
| --- | --- |
| Requirement Tag | NFR4 |
| Requirement Description | The system should be able to store user details in a secure way |
| Dependent on Requirements | FR3 |
| User/System interacting with the requirement | System |

**Table 2.5: Non-Functional requirement 5**

|  |  |
| --- | --- |
| Requirement Tag | NFR5 |
| Requirement Description | The system should be able to load the pages under 7 seconds |
| Dependent on Requirements | FR1, FR2, FR4, FR6 |
| User/System interacting with the requirement | System |

**Table 2.6: Non-Functional requirement 6**

|  |  |
| --- | --- |
| Requirement Tag | NFR6 |
| Requirement Description | The system should be online 24/7. |
| Dependent on Requirements | FR1, FR2, FR3, FR4, FR5, FR6 |
| User/System interacting with the requirement | System |

**Table 2.7: Non-Functional requirement 7**

|  |  |
| --- | --- |
| Requirement Tag | NFR7 |
| Requirement Description | The system should undergo maintenance if there are any issues |
| Dependent on Requirements | FR1, FR2, FR3, FR4, FR5, FR6 |
| User/System interacting with the requirement | System |

* 1. **Identification and design of the entity classes using E-R diagrams**

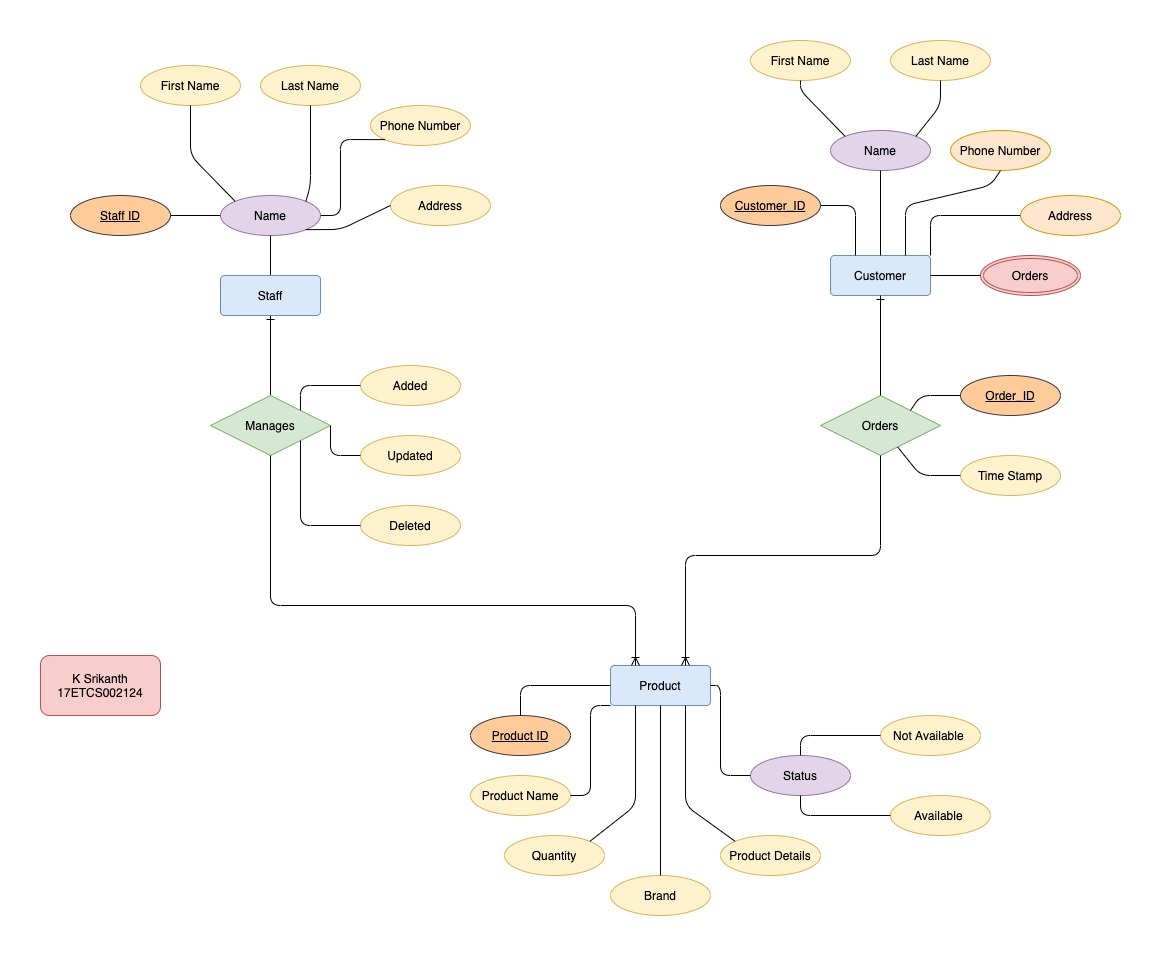
****

Figure 1 ER Diagram of Given Problem Statement

Given problem statement was to build an online **Mobile Shopping Application**, Figure 1 represents ER Diagram which helps us to build a database schema for our web application

Let’s list all the entities, attributes and relations,  
**Entities:** Entities are a real-world object with an existence, we have 3 entities from figure 1

1) **Customer Entity**

Here in our application Customer is an object and has a certain attribute

**a. Customer ID**

This Attribute Stores a unique value of every customer who has registered and this attribute is also a primary key

**b. Name**

This is a composite attribute which stores the first name and last name of the customer and this attribute has a constraint not null.

**c. Orders**

This is a multivalued attribute because there might be **“n”** number of orders placed or already has been placed by a customer so this attribute will have a separate table

**d. Phone Number**

This attribute stores the phone number from the given customer.

**e. Address**

This attribute stores the address from the given customer.

2. **Staff Entity**

Staff is an object and has some certain attributes which are,

**a. Staff ID**

This Attribute Stores a unique value of every Staff Member who has registered and this attribute is also a primary key

**b. Name**

This is a composite attribute which stores the first name and last name of the customer also this attribute can’t be null.

**c. Phone Number**

This Attribute stores a unique value of the staff phone number so that only one staff user can have one phone number

**d. Address**

This Attribute stores a unique value of the staff Address so that only one staff user can have one address

3. **Product Entity**

Finally, Product is also an object and attributes which are,

**a. Product ID**

This Attribute Stores a unique value of every product which has be entered and this attribute is also a primary key

**b. Product Name**

This Attribute stores the Product Name and this attribute has a constraint not null.

**c. Quantity**

This Attribute stores the quantity of the product and this attribute has a constraint not null.

**d. Brand**

This Attribute stores the brand of the product and this attribute has a constraint not null

**e. Product Details**

This Attribute stores the details of the product type char array.

**f. Status**

This Attribute keeps a track of if the all the products if they are available or not and this is of type Boolean.

**4. Order Relationship**

This Relationship defines that if a customer places an order and how it’s going to keep track of it

**a. Order ID**

This Attribute Stores a unique value when every time a product is ordered by a customer and this is a primary key for this table.

**b. Time Stamp**

This Attribute keeps a time/date record of when the order was placed. As this relationship is connected between two different entities which are product and customer this table will be having both the keys from the entities.

**2. Manages Relationship**

This Relationship defines that if an admin tries to perform operations on already listed product such as

**a. Added**

This Attribute is type Boolean where we can keep a track if admin adds a new product to the product table.

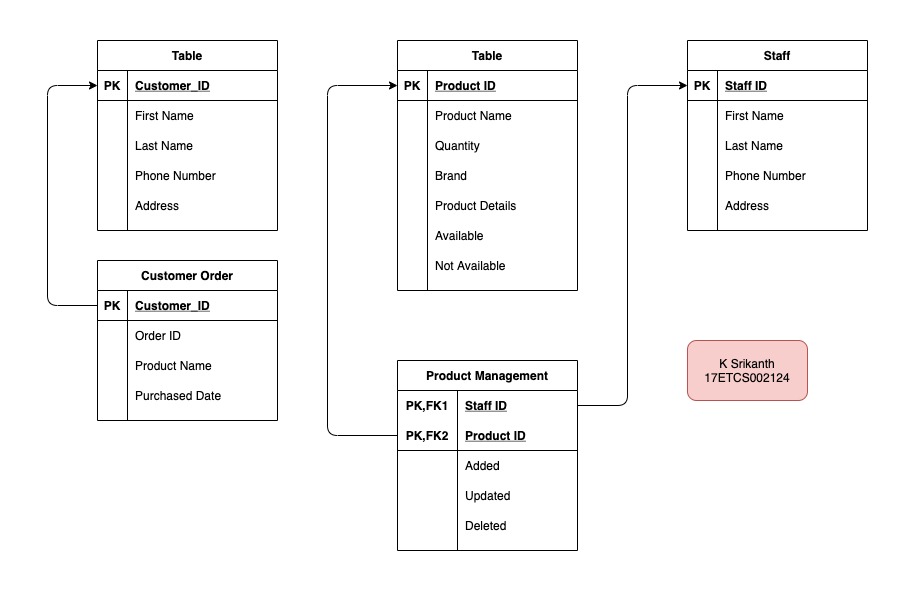
**b. Updated**

This Attribute is type Boolean where we can keep a track if admin wants to update the already listed product to the product table.

**c. Deleted**

This Attribute is type Boolean where we can keep a track if admin wants to delete a product from product table.

After Sketching up the ER Diagram now we have to figure out how do you convert a simple ER diagram to a database schema also known as relational schema (Figure 2) which helps us to build the MySQL Relational Database in an ease manner.

**

*Figure 2 Relational Schema for ER Diagram (Figure 1)*

* 1. **Design of UML interaction sequence diagrams**

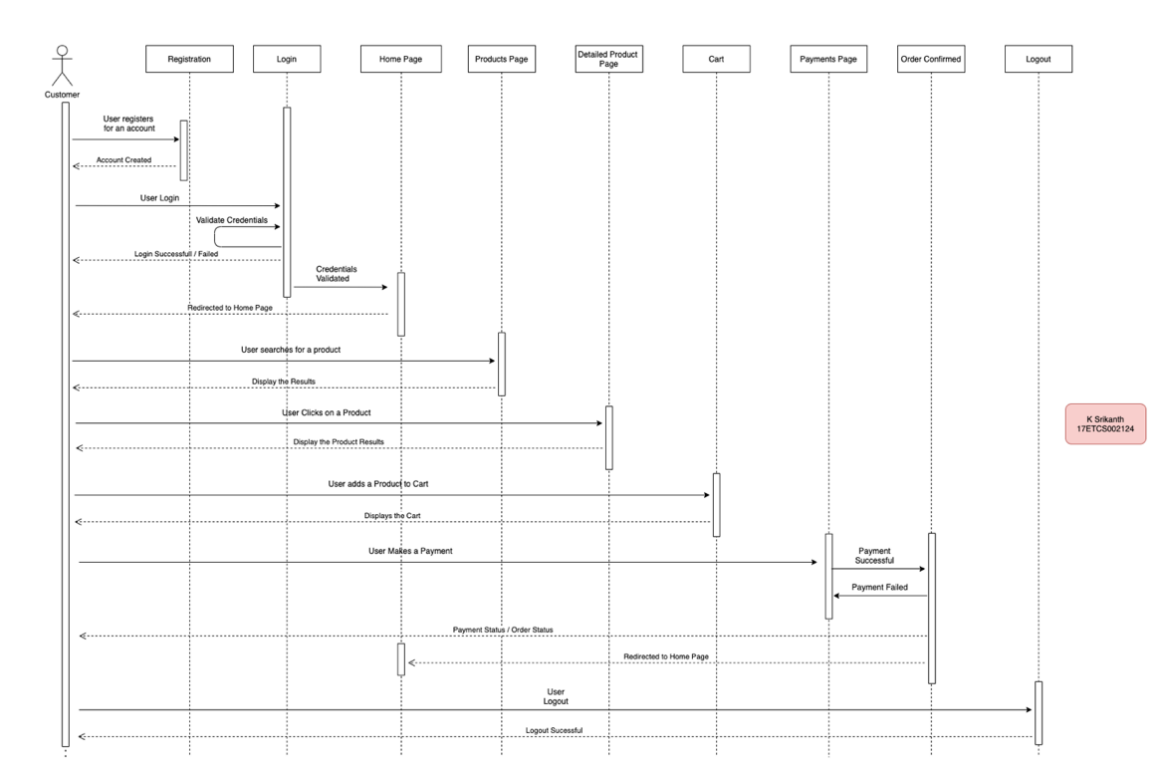
**User interaction**

Figure 3 Sequential Diagram for the given problem statement (User)

Sequential Diagrams are used to represent time sequence of the objects like what order is the user is going to interact with the system,

There are two types of users in our case here

**1. Customer / User**

**2. Admin / Staff**

Let’s see how a customer interacts with our system (Website),

**Scenario**

Firstly, the user has to register using the interface provide by the system which includes filling up all the necessary details to get started. After registration now user can be able to sign in using the interface provide by our system and system is going to verify if the user’s credentials are valid or not and it will display a prompt to the user with a dialog box stating that the user has logged in or failed to login. If the user has failed to login then the user can again enter their credentials on login page. If my dialog box says that successfully logged in then the user will be redirected to the homepage of **Mobile Shopping**

**Website** where the user can search for their preferred Mobile and system will display the results. Now when user clicks on a product it will be able to displays fully detailed information about the product and user can know about the product much better if they are not aware of the product that they are looking at and user can add the product to the cart section of the website after knowing the details about the product now user have an option where they can be able to buy or reserve the product using a payment gateway if the payment is successful then user will receive an order ID followed by the list of products they have purchased after this they would be getting redirected to the home page of our web application and if the payment is failed then user will have to try the payment again. Finally, if the user is done with their shopping, they can logout of the system.

**Now,** Let’s see how a admin interacts with our system (Website),

**Admin interaction**

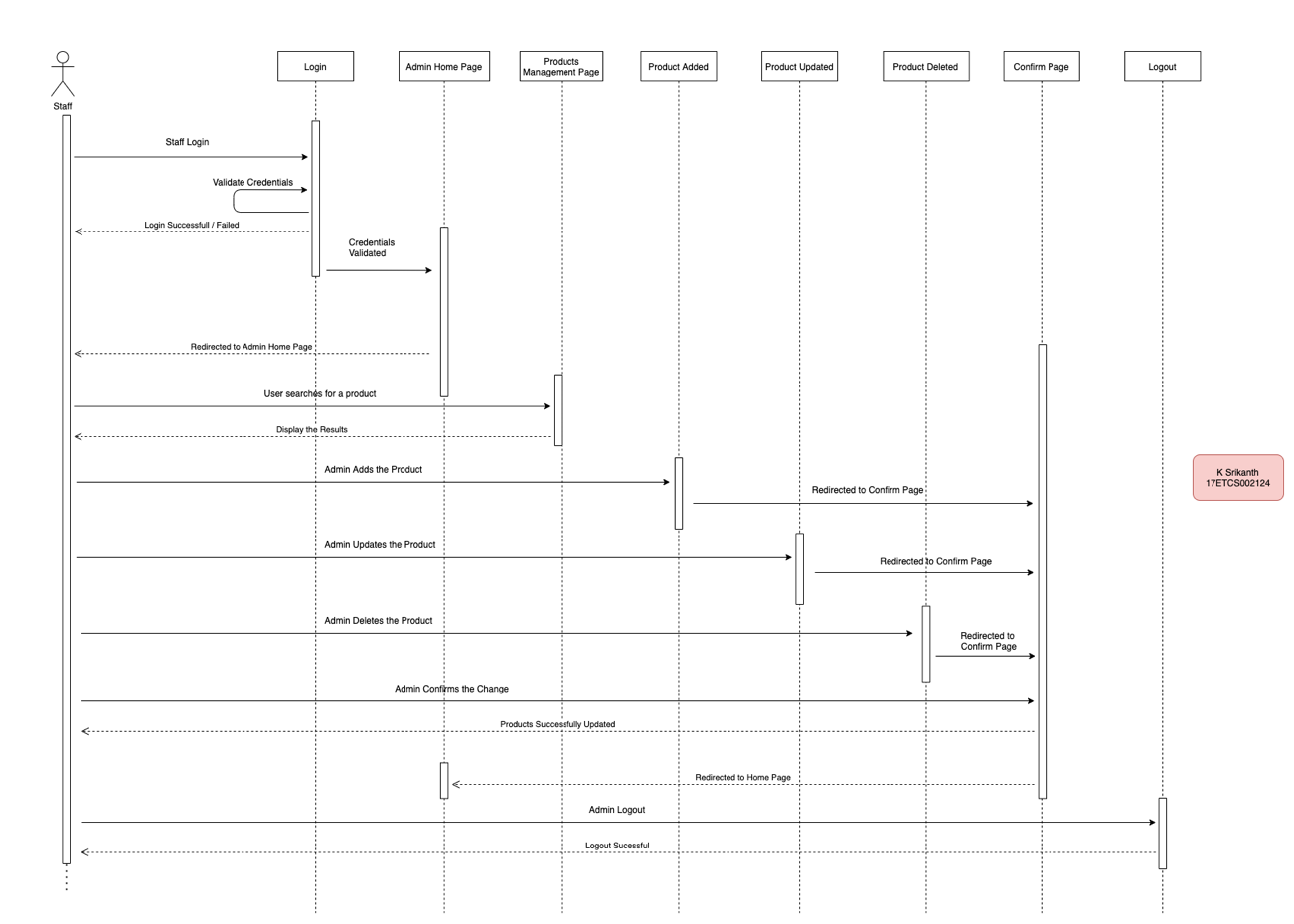
****

Figure 4 Sequential Diagram for the given problem statement (Admin)

**Scenario**

The staff administration are going to provide the credentials for admin account where they can make changes to the websites products stock now the admin has to sign in using the interface provide by the system and the system is going to verify if the admin’s credentials are valid or not and it will display a prompt to a user with a dialog box stating that the user has logged in or failed to login if the user has failed to login then the user can again enter their credentials on login page. If my dialog box says that successfully logged in then It will be redirected to the homepage of our admin where the admin where admin can search for their Mobile and system will display the results. The four major operations that admin can perform are

**1. Add**

If admin wants to add a new product for their website, they can do it via creating a new product and uploading the particular details about the product.

**2. Update**

If admin wants to update details of the product which is up online, they can be able to do it with an edit option on the product page.

**3. Delete**

If admin wants to delete the product which is up online, they can be able to do it with a delete option on the product page.

**4. Website for maintenance**

If admin wants to put their website for maintenance, they would be able to do it via settings.

After Performing all the operations, the admin can verify everything like what changes that they made all together and make it online. Finally, if the admin is done making changes with their website, they can logout of the system.

* 1. **Design of Algorithm/ Flowchart**

**User interaction**

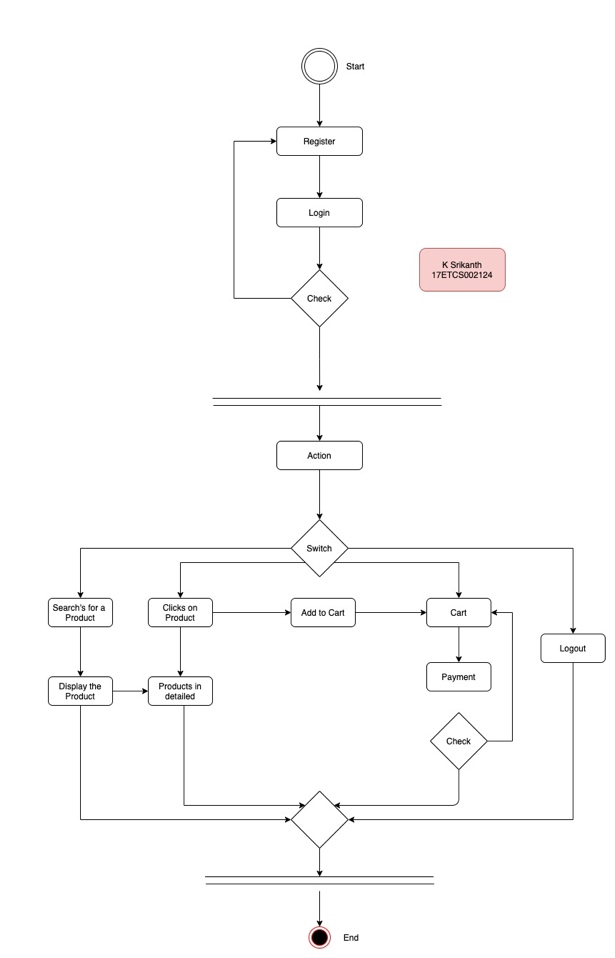
****

Figure 5 Activity Diagram of the given problem statement (User)

The user has to register by filling up all the necessary details to get started. After registration now user can be to login into the system and the system is going to verify if the user’s credentials are valid or not valid then we have a switch case why? cause is no order to use the system

**Example:** User is logged in and now he wants to logout without even looking at the products.

So, there are 4 cases here,

**Case 1:** User can logout right after they login into the system

**Case 2:** User can be able to search for the product and wait for the result to pop up and logout from the system.

**Case 3:** User can be able to search for the product and wait for the result to pop up then look at the product in detail and logout from the system.

**Case 4:** User can be able to add products to their carts and make a payment and then after a successful transaction they can logout from the system**.**

Finally, if the user is done with their shopping, they can logout from the system

**Admin interaction**

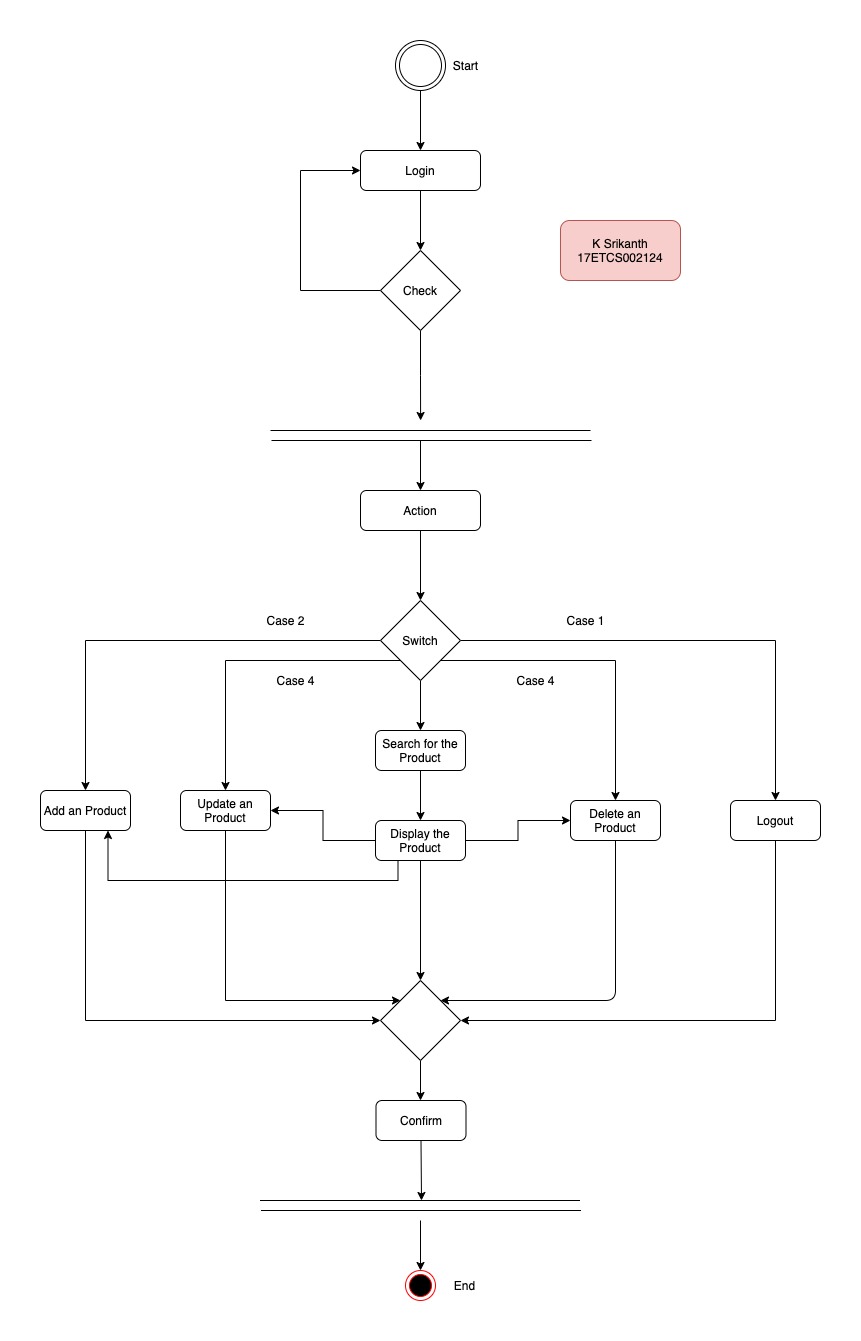


Figure 6 Activity Diagram of the given problem statement (Admin)

The Staff Administration are going to provide the access to staff using admin account which they can login into the system and the system is going to verify if the admin’s credentials are valid or not valid then we have 5 switch cases why? cause is no order to use the system.

**Example:** Admin wants to just add a new product and logout of the system without even performing other actions.

So, there are 5 cases here,  
**Case 1:** Admin can logout right after they login into the system

**Case 2:** Admin can be able to add new product and logout from the system after they are finished posting it.

**Case 3:** Admin can be able to update a product which is online and make changes then logout from system after they are finished.

**Case 4:** Admin can be able to delete a product which is online then logout from system after they are finished

**Case 5:** User can be able to search for the product and wait for the result to pop up then look at the product in detail and logout from the system.

Finally, if the user is done with their shopping, they can logout from the system

**Conclusion**

As a part of our application development, we framed functional and non-functional requirements of the system. Then we identified the general use-cases i.e. operations the system is expected to perform. We drew entity-relationship diagram which hints us about how we should design our database or database schema. We illustrated how the system should interact with the users with the help of sequence diagram. Sequence diagram helps us describe the behaviour of the system. And at last we summarized the whole system into a simple activity diagram which gives a brief overview of how the sequence of actions occur and flow of control.

We used different tools of UML to simplify our application design process. These diagrams give a clear picture of how the system should work. And thus The Unified Modelling Language (UML) is a general- purpose, developmental, modelling language in the field of software engineering that is intended to provide a standard way to conceptualize, visualize and understand the design n of the “to be developed system”.