

|  |  |
| --- | --- |
| **ASSIGNMENT** | |
| **Course Code** | 19CSC314A |
| **Course Name** | Web Architecture and Application development |
| **Programme** | B.Tech. |
| **Department** | CSE |
| **Faculty** | FET |

#### 

|  |  |
| --- | --- |
| **Name of the Student** | Subhendu Maji |
| **Reg. No** | 18ETCS002121 |
| **Semester/Year** | 6TH / 2018 |
| **Course Leader/s** | Mr. Kishore S. M. /Deepak V/ Hari Krishna S M |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Declaration Sheet** | | | | | | | | |
| Student Name | Subhendu Maji | | | | | | | |
| Reg. No | 18ETCS002121 | | | | | | | |
| Programme | B.Tech. | | | | | Semester/Year | 6th / 2018 | |
| Course Code | 19CSC314A | | | | | | | |
| Course Title | Web Architecture and Application development | | | | | | | |
| Course Date |  | | to | |  | | | |
| Course Leader | Mr. Kishore S. M. /Deepak V/ Hari Krishna 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 | |  | | | | | Date |  |
| Submission date stamp  (by Examination & Assessment Section) | |  | | | | | | |
| Signature of the Course Leader and date | | | | Signature of the Reviewer and date | | | | |
|  | | | |  | | | | |

# **Contents**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

[**Declaration Sheet** ii](#_Toc70111400)

[**Contents** iii](#_Toc70111401)

[**Question No. 1** 5](#_Toc70111402)

[1.1 Functional and non-functional requirements 5](#_Toc70111403)

[1.2 Identification and design of the entity classes using E-R diagrams 7](#_Toc70111404)

[1.3 Design of UML interaction sequence diagrams 10](#_Toc70111405)

[1.4 Design of Algorithm/ Flowchart 12](#_Toc70111406)

|  |  |  |  |
| --- | --- | --- | --- |
| Faculty of Engineering and Technology | | | |
| Ramaiah University of Applied Sciences | | | |
| Department | Computer Science and Engineering | Programme | B. Tech |
| Semester/Batch | 06/2018 | | |
| Course Code | 19CSC314A | Course Title | Web Architecture and Application development |
| Course Leader | Mr. Kishore S. M. /Deepak V/ Hari Krishna S M | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Assignment-1 **Marking Scheme** | | | | | | | | | | | | |
| Register No. | | | 18ETCS002121 | | | Name of the Student | | | | SUBHENDU MAJI | | |
| 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 | | | |  |  | |  | |  | | | | |
| **Total Marks** | | | |  |  | |  | |  | | | | |
| **Signature of First Examiner Signature of Second Examiner** | | | | | | | | | | | | | |

# 

# **Question No. 1**

**Solution to Question No. 1:**

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.

## 1.1 Functional and non-functional requirements

**Functional Requirements:**

|  |  |
| --- | --- |
| Requirement Tag | FR1 |
| Requirement Description | the system should allow new users to register and existing users to login |
| User/System interacting with the requirement | user, admin |

|  |  |
| --- | --- |
| Requirement Tag | FR2 |
| Requirement Description | the system should allow users to search smartphone by brand. |
| User/System interacting with the requirement | User |

|  |  |
| --- | --- |
| Requirement Tag | FR3 |
| Requirement Description | the system should allow users to filter smartphone with phone specification like RAM, Storage, etc. |
| User/System interacting with the requirement | User, admin |

|  |  |
| --- | --- |
| Requirement Tag | FR4 |
| Requirement Description | the software should display smartphone description |
| User/System interacting with the requirement | user |

|  |  |
| --- | --- |
| Requirement Tag | FR5 |
| Requirement Description | the system should allow user to see the inventory available for the smartphone. |
| User/System interacting with the requirement | User, admin |

|  |  |
| --- | --- |
| Requirement Tag | FR6 |
| Requirement Description | the system should allow users to buy multiple smartphones at a time by adding to cart. |
| User/System interacting with the requirement | user |

|  |  |
| --- | --- |
| Requirement Tag | FR7 |
| Requirement Description | the system should allow users to buy smartphone directly without adding to cart |
| User/System interacting with the requirement | user |

|  |  |
| --- | --- |
| Requirement Tag | FR8 |
| Requirement Description | the system should allow admin to add or update smartphone details. |
| User/System interacting with the requirement | admin |

Non-functional Requirements:

**NFR1:** the software should have good maintainability.

​ The maintenance team should update the software periodically. The bugs should be fixed as soon as discovered. the team should update the software with trending UI. New features should be added periodically.

**NFR2:** the system should have a secure payment gateway.

​ The system should be secure and clean. The gateway should be taken from a reputative bank server. The system should not be easy to hack.

**NFR3:** the system should have a good server support.

​ The system should not get crashed frequently. the system should have a backup server support when doing maintenance. the system should not be down for too long.

**NFR4:** The system should be user-friendly.

​ The system should have clear navigation.

**NFR5:** the system should be responsive in multiple platforms.

​ The system should be supportive in mobile, tablet and desktop. The software should be supportive in multiple operating system - Linux, Mac and Windows. The system should be working on Android as well as iOS.

## 1.2 Identification and design of the entity classes using E-R diagrams

**ER Diagram** stands for Entity Relationship Diagram, is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

The purpose of ER Diagram is to represent the entity framework infrastructure.

* **Entity**: a thing with distinct and independent existence.
* **Attributes**: these are properties of entity.
* **Relationship**: the number of occurrences in one entity that is associated with the number of occurrences in another entity.

|  |  |
| --- | --- |
| Entities of Online Smartphone shopping system | |
| USER | It is a strong entity. It is a person who deals with the system and who select the products and buy the products and interact with entire online Furniture system |
| PRODUCT | It is a strong entity. It is the item or thing that the customer buys |
| PAYMENTS | It is a strong entity; it associates a user to the product or cart he has purchased |
| CART | It is a weak entity. A cart only exists if a user exists, because each user has his own cart |
| BRAND | It is a strong entity. The products are classified into different categories(brand) for the user to search |
| ADMIN | It is a strong entity. Admin manages the products and categories. |
| VARIANTS | It is a strong entity. Different variants of a smartphone like RAM, storage etc. |

|  |  |
| --- | --- |
| Relationship of Online Smartphone shopping system | |
| Category\_contains | Binary relationship between category and product entity. It is n:n relationship, because n products can have n categories |
| Searches | It is a binary relationship between customer and categories, it is a 1:n relationship, because a customer can search for n categories |
| Buys | It is ternary relationship between customer, product and payment. It is n:n:1, relationship, as n customers can buy n products and make 1 payment for it |
| Makes\_payment | It’s a binary relationship between, a customer and payment. It’s a 1:n relationship, as 1 customer can make n payments |
| Owns | It’s a weak relationship between a user and a cart. This means if the user exists, then he owns a cart, it’s a 1:1 relationship |
| Cart\_contains | It’s a weak relationship between a cart and a product. If a cart exists, then it can contain multiple products, 1: n relation |
| Payment\_for\_cart | It’s a weak relationship between a car and a payment. If a cart exists, then a payment can be made for it, 1:1 relation |
| Manages\_product | Binary relationship between admin and product entity. It is n:n relationship, because n admins can manage n products |
| Manages\_admin | Binary relationship between admin and category entity. It is n: n relationship, because n admins can manage n categories |

|  |  |
| --- | --- |
| Attributes of Online Smartphone shopping system | |
| USER | **name** – name of the customer  **email** – email id of the customer, it has to be unique  **password** – password of the user, required to login or signup  **address** – the delivery address of the user  **user\_id** – the user\_id of the user, also used as the login id. It is used to uniquely identify each user (primary key)  **phone\_no** – the users contact number |
| SMARTPHONE | **smartphone\_id** – the primary key used to uniquely id the product  **model\_name** – name of the product |
| VARIANTS | **variant\_id** – the primary key used to uniquely id the product  **color** – the color of the product  **price** – the cost of a single unit of the product  **RAM** – the RAM of the smartphone  **Storage** – the storage (HDD) of the smartphone.  **inventory** – the current stock or the number of units to be sold |
| PAYMENTS | **payment\_id** – the primary key used to uniquely identify each transaction  **user\_id** – the foreign key used to id the user associated to the payment |
| CART | **quantity** – the units of product ordered  **payment\_id** – the payment associated with the cart  **user\_id** – the user that owns the cart  **product\_id** – the product that is added to the cart |
| BRAND | **brand\_title** – the title of the brand  **brand \_id** – the primary key of each brand |
| ADMIN | **email** – the admins email address  **password** – the password used by the admin to login or sign-up  **admin \_id** – the primary key or the login id of the admin |

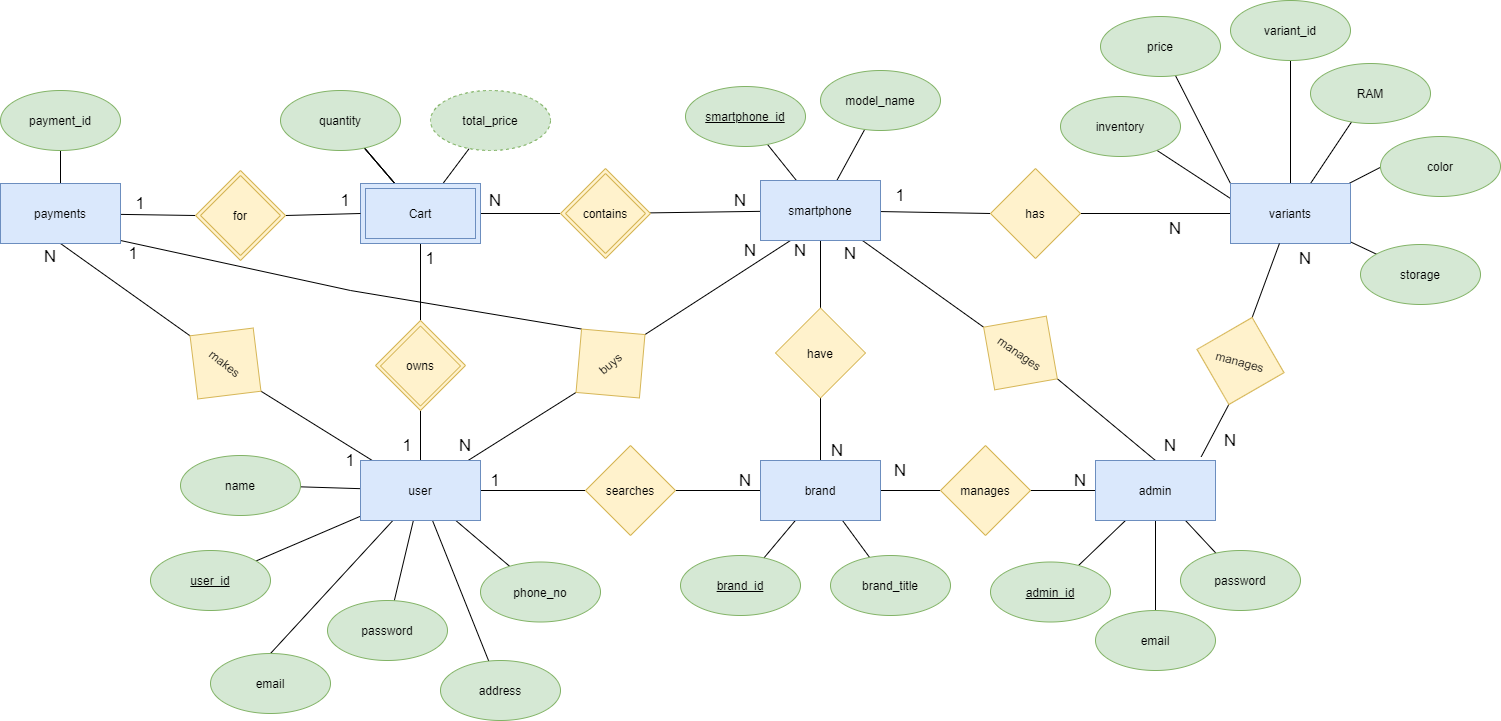


Figure 1 ER Diagram

## 1.3 Design of UML interaction sequence diagrams

**Sequence Diagrams** are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

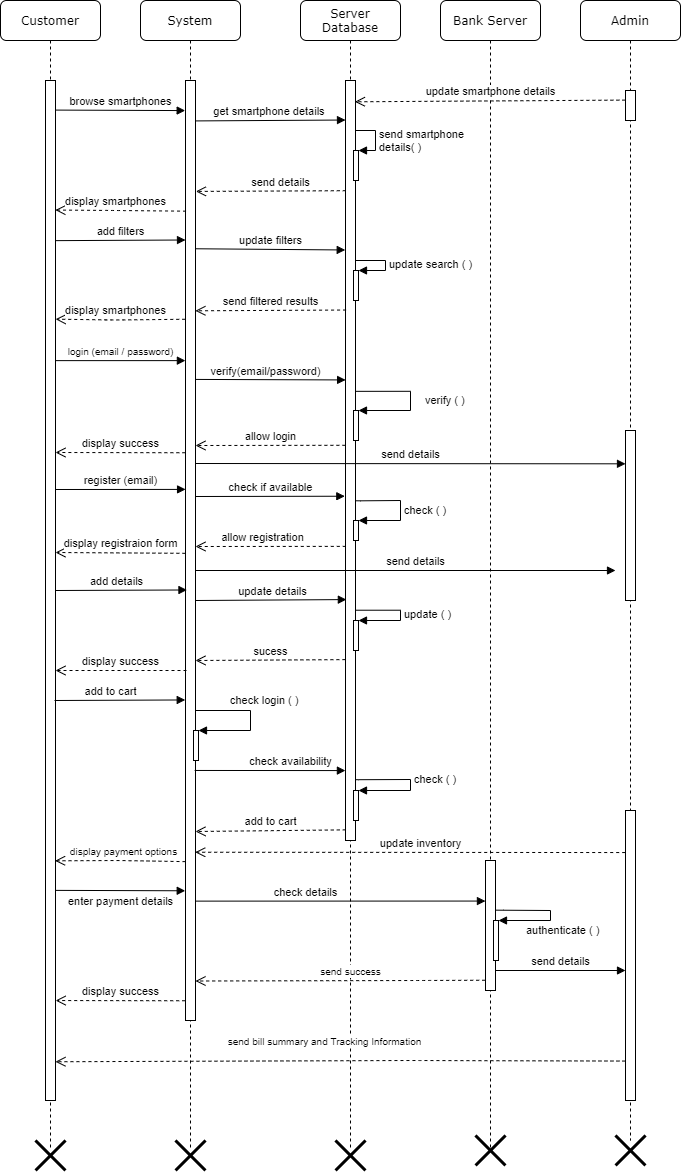


Figure 2 Sequence Diagram

## 1.4 Design of Algorithm/ Flowchart

A new user can register and the old user can login into system to browse smartphones. User can search or filter from brand name and specifications like RAM, storage, etc. User can either buy the product now or add the product to cart for later purchase or for multiple product purchase. Then user can view the total amount of the cart and add the shipping address. Then, user can complete the payment process and on successful payment user will receive a order summary and shipping tracking info.

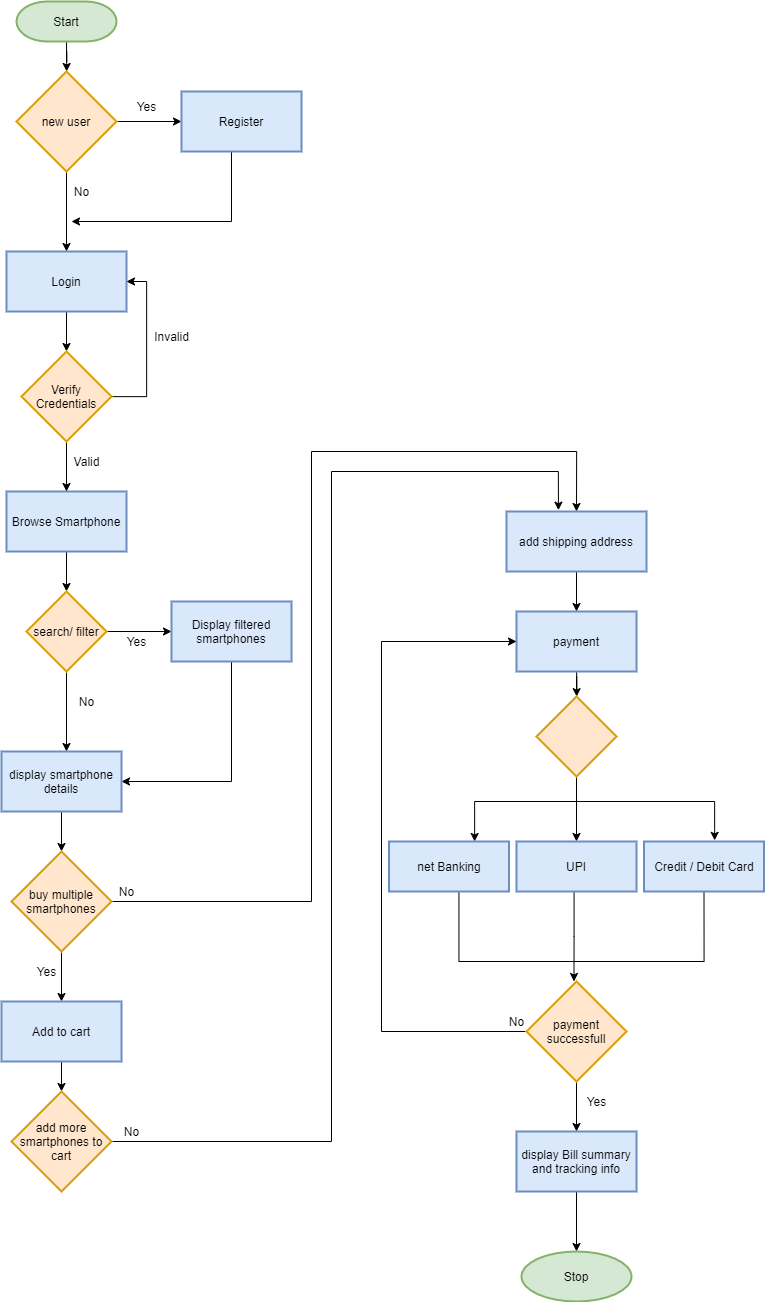


Figure 3 Flowchart