Shashwat Shah
60004220126
TYBtech Comps B

Aim : Identify scenarios and develop UML use case and class diagram for the project.
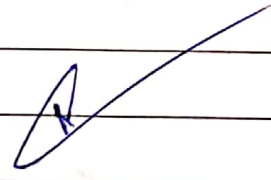
Theory : The UML use case diagram
→ Actor and their roles.

① User/ Customer — will be having access to the deployed software and can enter its location, can also post content and subscribe to a particular news

② Administorator — Keeps a close eye on our platform with all rights to view posts, remove any if needed.

③ Database Manager — At first the authenticaton check which remains as the most important part. Look for the recovery and backup of the entire dB, Managing the users post data. Also the likes bookmarks and likes and dislikes

The UML class diagram

→ The User Class
→ The News class
→ The Media class
→ The advertisment class
→ The Role class
→ The Permission class.

Each class has its own attributes and methods.

Conclusion - We learnt about UML use case diagrams and class diagrams and have created them for our to platform.

Software Engineering

Experiment-3

Div: B
Team Members:
Shashwat Shah : 60004220126
Aagam Shah : 60004210176
Deep Gohil : 60004220122

Batch: C22

Aim: Identify scenarios & develop UML Use case and Class Diagram for the project

Theory:

Use case Diagram

A UML use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behaviour (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation (i.e., use case diagram). A key concept of use case modelling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behaviour in the user's terms by specifying all externally visible system behaviour.Use cases represent only the functional requirements of a system. Other requirements such as business rules, quality of service requirements, and implementation constraints must be represented separately, again, with other UML diagrams.

Use case diagrams are typically developed in the early stage of development and people often apply use case modelling for the following purposes:

- Specify the context of a system.
- Capture the requirements of a system.
- Validate a system architecture.
- Drive implementation and generate test cases.
- Developed by analysts together with domain experts.

Class Diagram

In software engineering, a class diagram in the Unified Modelling 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 (or methods), and the relationships among objects.

Purpose of Class Diagrams

1. Shows static structure of classifiers in a system.
2. Diagram provides a basic notation for other structure diagrams prescribed by UML.
3. Helpful for developers and other team members too
4. Business Analysts can use class diagrams to model systems from a business perspective.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.
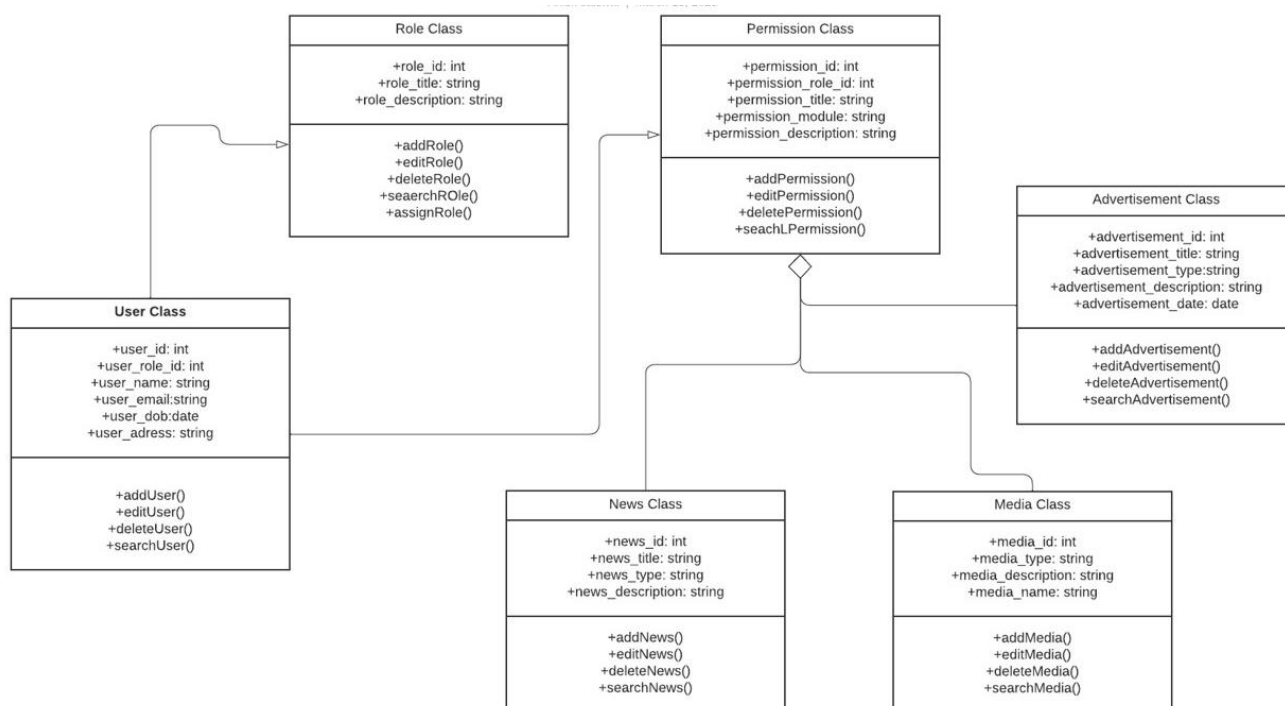
FOR OUR PROJECT:

Use Case Diagram:

Actors and their User scenarios:

- **Users** will have access to the deployed software and can enter their location to get the desired useful news. **Users** will also have the priviledge to add/subscibe to a particular news and to post news.

- **Administrator** will keep a close eye on the type of the news posted and will have the total rights to remove content of certain type or block the user if needed.

- **Database Manager** will have several use cases:
    a. Having authentication check: It becomes useful for the database administrator to provide permittable access to the verified users only.
    b. Managing Users Post data: Saving the data created by the User so as to make it available for all the users. This will also include keeping the count of likes and dislikes on each of the users post.
    c. He will also perform analysis on the users data so as to suggest him similar trends in the future.
    d. He will also be handling the recovery and backup of the entire database.

Class Diagram:

- **Class** - User
  Attributes:
  - User_id
  - User_role_id
  - User_name
  - User_email
  - User_dob
  - User_address

  Operations:
  - addUser()
  - editUser()

- ○ deleteUser()
- ○ searchUser()

- **Class**- Role Class
  Attributes:
    - ○ Role_id
    - ○ Role_title
    - ○ Role_description
  Operations:
    - ○ addRole()
    - ○ editRole()
    - ○ deleteRole()
    - ○ searchRole()
    - ○ assignRole()

- **Class** – Permission Class
  Attributes:
    - ○ permission_id
    - ○ permission_role_id
    - ○ permission_title
    - ○ permission_module
    - ○ permission_description
  Operations:
    - ○ addPermission()
    - ○ editPermission()
    - ○ deletePermission()
    - ○ searchPermission()

- **Class** – Advertisment class
  Attributes:
    - advertisement_id
    - advertisement_title
    - advertisement_type
    - advertisement_description
    - advertisement_date
  Operations:
    - addAdvertisement()
    - editAdvertisement ()
    - deleteAdvertisement ()
    - searchAdvertisement ()

- **Class** – Media Class
  Attributes:
    - media_id
    - media_type
    - media_description
    - media_name
  Operations:
    - addMedia()
    - editMedia()
    - deleteMedia ()
    - searchMedia()

- **Class** – News Class
  Attributes:
    - News_id
    - News_title
    - News_type
    - News_description
  Operations:
    - addNews()
    - editNews()
    - deleteNews ()
    - searchNews()

Conclusion: We learnt about UML Use-Case Diagrams and Class Diagrams and have created them for our tourism analysis project.