12.ACTIVITY

12. Identify use cases on files concept and develop algorithm to solve the same.

➢ Designing Use Cases for a Project

Before we start working on any project, it is very important that we are very clear on what we want to do and how do we want to do. In my last Blog, I discussed on how to write a good SRS for your project and what is the advantage we get out of that. In this Blog, I'll discuss Use Cases and their advantage in our projects.

1. What are Use Cases?

In software and systems engineering, a use case is a list of actions or event steps, typically defining the interactions between a role (known in the Unified Modeling Language as an *actor*) and a system, to achieve a goal. The actor can be a human, an external system, or time. In systems engineering, use cases are used at a higher level than within software engineering, often representing missions or stakeholder goals. Another way to look at it is a use case describes a way in which a real-world actor interacts with the system. In a system use case you include high-level implementation decisions. System use cases can be written in both an informal manner and a formal manner.

2. What is the importance of Use Cases?

Use cases have been used extensively over the past few decades. The advantages of Use cases includes:

- The list of goal names provides the shortest summary of what the system will offer
- It gives an overview of the roles of each and every component in the system. It will help us in defining the role of users, administrators etc.
- It helps us in extensively defining the user's need and exploring it as to how it will work.
- It provides solutions and answers to many questions that might pop up if we start a project unplanned.

3. How to plan use case?

Following example will illustrate on how to plan use cases:

- ❖ **Use Case:** What is the main objective of this use case. For eg. Adding a software component, adding certain functionality etc.

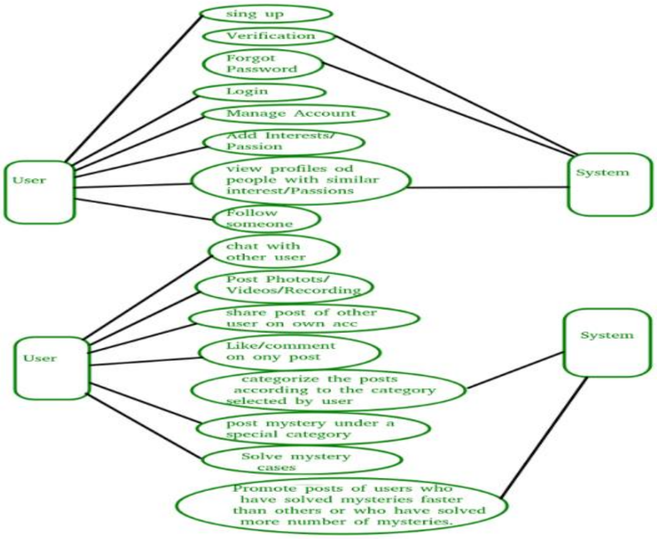
- **Primary Actor:** Who will have the access to this use case. In the above examples, administrators will have the access.
- **Scope:** Scope of the use case
- **Level:** At what level the implementation of the use case be.
- **Flow:** What will be the flow of the functionality that needs to be there. More precisely, the work flow of the use case.

Some other things that can be included in the use cases are:

- **Preconditions**
- **Postconditions**
- **Brief course of action**
- **Time Period**

Use Case Diagram:

- Below is a sample use case diagram which I have prepared for reference purpose for a sample project (much like Facebook). It would help us to understand the role of various actors in our project. Various actors in the below use case diagram are: **User and System.**
- The main use cases are in the system and the diagram illustrates on how the actors interact with the use cases. For eg. During Sign Up, only users need to interact with the use case and not the system whereas when it comes to categorizing posts, only system would be required.


sing up
Verification
Forgot Password
Login
Manage Account
Add Interests/ Passion
User
view profiles od people with similar interest/Passions
System
Follow someone
chat with other user
Post Photots/ Videos/Recording
share post of other user on own acc
System
User
Like/comment on ony post
categorize the posts according to the category selected by user
post mystery under a special category
Solve mystery cases
Promote posts of users who have solved mysteries faster than others or who have solved more number of mysteries.

Tool for drawing use case and other UML diagrams:

Creately is a wonderful tool that helps us to easily create Use case diagrams. *I again emphasize the importance of SRS and use cases for your project as it helps us to organize and plan out things. It also helps us to mitigate the risks that are there and early risk identification.*