Aim :

Write a System Requirement specification for the selected topic / project.

Theory :

Purpose :

The main objective of this document is to illustrate the requirements of the project E-learning system. The document gives the detailed description of the both functional and non-functional requirements proposed by the client. The purpose of this project is to provide a friendly environment to learning process. this project describes the hardware and software interface requirements using UML diagrams.

Scope of Development Project :

The e-Learning System is basically updating the traditional learning system into an internet based application so that the users / students can know the details of their course, give the test online and see their progress report.

The project is specifically designed for the use of students and teachers. The product will work as a complete user interface for the students and the teachers.

Scanned by CamScanner

The project can be easily implemented under various situations. We can add new features as and when we require, making reusability possible as there is flexibility in all the modules. The language used for developing the project is HTML as it is quite advantageous than other languages in terms of performance, tools available, cross platform compatibility, libraries, cost (freely available) and development process.

User Classes and characteristics :

The system provides different types of services based on the type of users. [student / teacher]. The teacher / Admin will be acting as the controller and he will have all the privileges of an administrator. The users can be either a student or teaching staff who will be accessing the ~~Library~~ e-learning portal online.

The features that are available to the student are :

- Search for the course
- Take a course
- Give the tests
- Ask Queries
- See the course status
- Analysis the result.
- ~~a~~

Scanned by CamScanner

The features that are available to the Teacher are :-

- Upload video lectures
- Take tests
- Answer to the student queries
- Display the notice / Announcement
- Prepare result of tests
- Update course content.

Operating Environment :

The product will be operating over internet in browsers. The e-Learning system is a website and shall operate in all famous browsers, for a model we are taking Google chrome and Mozilla Firefox. Most of the features will be compatible with Google chrome. The only requirement to use this online product would be the internet connection. The hardware configuration include Hard Disk : 40 GB, Monitor : 15" color monitor, sound systems, etc. The basic input devices required are keyboard, mouse and output devices are monitor, printer etc.

Assumptions and Dependencies :

The assumptions are :-

- The coding should be error free.
- The system should be user-friendly so that it is easy to use for the users.

Scanned by CamScanner

- The information of all students, courses and teachers must be stored in a database that is accessible by the website.
- The system should have more storage capacity and provide fast access to the DB.
- The system should provide search facility and support quick transactions.
- The e-Learning system is running 24 hours a day.
- Users may access from any computer that has Internet browsing capabilities and an Internet connection.
- Users must have their correct usernames and passwords to enter into their online accounts and do actions.

The dependencies are :-

- The specific hardware and software due to which the product will run
- On the basis of listing requirements and specification the project will be developed and run.
- The end users should have proper understanding of the product.
- The information of all users must be stored in a database that is accessible by the system.
- Any update regarding the course content is to be recorded to the database and the data entered should be correct.

Scanned by CamScanner

Data Requirement :

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be the queries as fired by the users like create an account, selecting courses and putting into account. Now the output will be visible when the user requests the server to get details of their account in the form of time, date and which courses are currently in the account.

~~System~~ Security Requirement :

- System will use secured database
- Normal users can just read information ~~but~~ about courses but they cannot edit or take that course.
- System will have different types of users and every user has access constraints.
- Proper user authentication should be provided
- No one should be able to hack user's password
- There should be separate accounts for admin and members such that no member can access the database and only admin has the rights to update the database.

Scanned by CamScanner

Performance Requirement :-

The proposed system that we are going to develop will be used as the chief performance system within the different campuses of the university which interacts with the university staff and students. Therefore, it is expected that the database would perform functionally all the requirements that are specified by the university.

- The performance of the system should be fast and accurate
- The e-Learning system shall handle expected and non-expected errors in ways that prevent loss in information and long down-time period. Thus it should have inbuilt error testing to identify invalid username/ password.
- The system should be able to handle large amount of data. Thus it should accommodate high number of courses and users without any fault.

Conclusion :-

Thus we have studied ~~the~~ and identified the system requirement specification for the e-Learning system.

Scanned by CamScanner

Aim :

To develop activity diagram for selected project.

Theory :

Activity Diagram :

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagram also describe the steps in a use case diagram. Activities modeled can be sequential and concurrent. In both cases an activity diagram will have a beginning and an end. In between there are ways to depict activities, flows, decisions, guards, merge and time events and more.

Activity Diagram Notations :

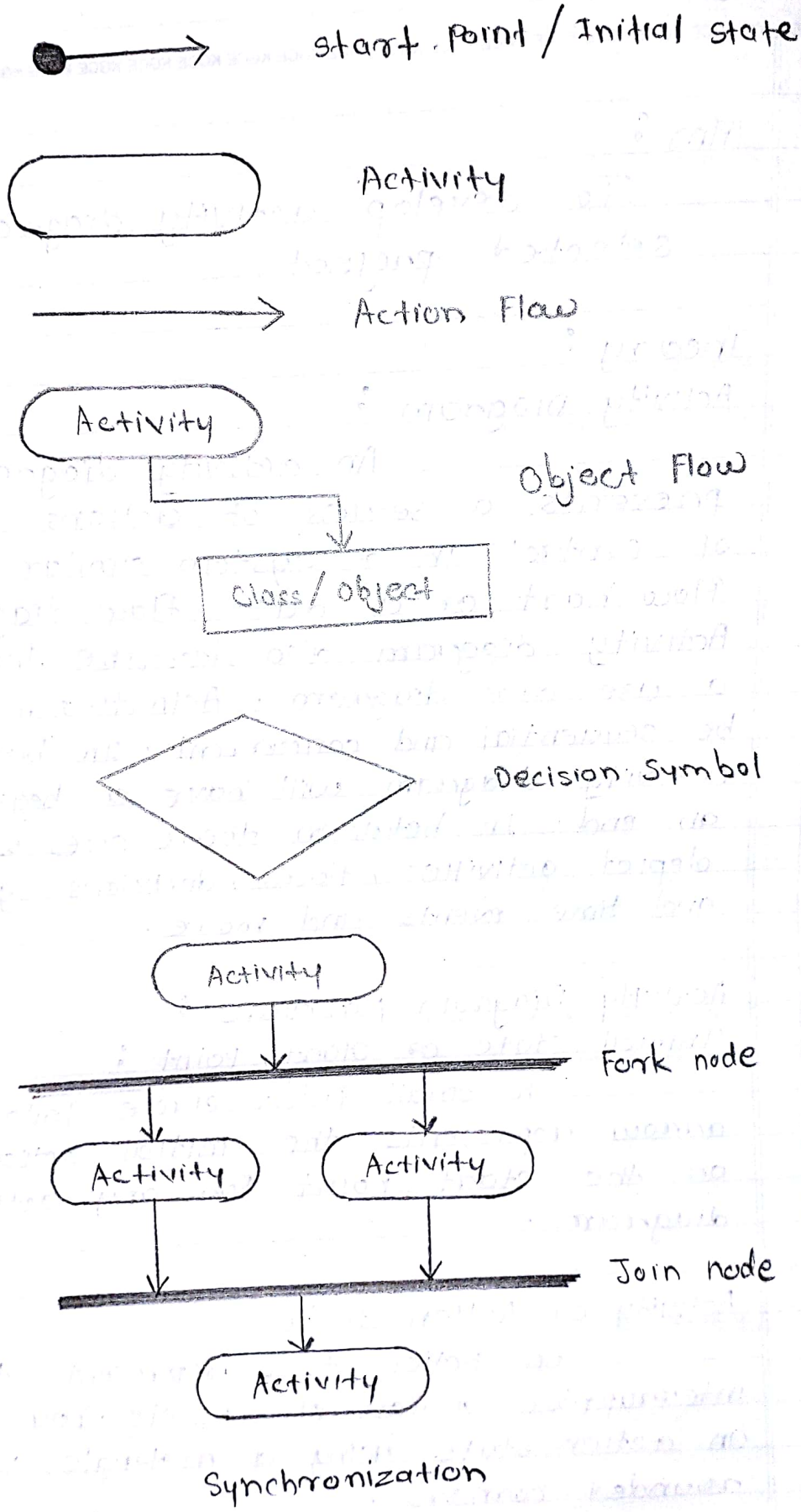
Initial State or Start Point :

A small filled circle followed by an arrow represents the initial action state or the start point for any activity diagram.

Activity or Action State :

An Action state represents the non-interruptible action of objects. You can draw an action state using a rectangle with rounded corners.

Scanned by CamScanner



Scanned by CamScanner

KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE KGCE

Action Flow :

Action flows, also called edges and paths, illustrate the transitions from one action state to another. They are usually drawn with an arrowed line.

Object Flow :

Object flow refers to the creation and modification of objects by activities. An object flow arrow from an action to an object means that the action creates or influences the object. An object flow arrow from an object to an action indicates that the action state uses the object.

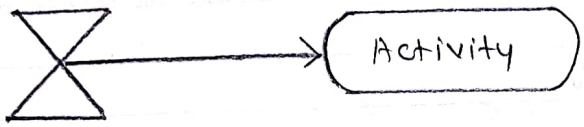
Decisions and Branching :

A diamond represents a decision with alternate paths. When an activity requires a decision prior to moving on to the next activity, add a diamond between the two activities. The outgoing alternates should be labeled with a condition or guard expression.

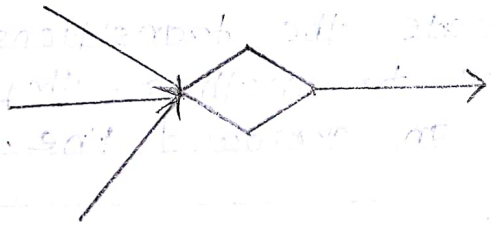
Synchronization :

A fork node is used to split a single incoming flow into multiple concurrent flows. It is represented as a straight, slightly thicker line in an activity diagram.
A join node joins multiple concurrent flows back into a single outgoing flow.

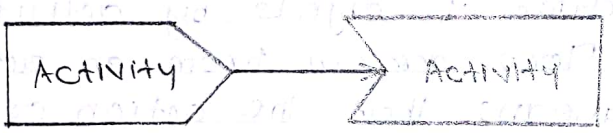
Scanned by CamScanner



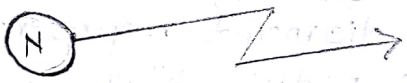
Time Event



Merge



Signal sent and received



Interrupting Edge Symbols

End point symbol

Scanned by CamScanner

A Fork and join node used together are often referred to as synchronization.

Time Event :
This refers to an event that stops the flow for a time; an hourglass depicts it.

Merge Event :
A merge event brings together multiple flows that are not concurrent.

Sent and Received Signals :
Signals represent how activities can be modified from outside the system. They usually appear in pairs of sent and received signals, because the state can't change until a response is received.

Interrupting Edge :
An event, such as a cancellation, that interrupts the flow denoted with a lightning bolt.

Final state or End Point :
An arrow pointing to a filled circle nested inside another circle represents final state.

Conclusion : Thus, we have studied and developed the activity diagram for selected project.

Scanned by CamScanner

Aim :-

To develop use case diagram for selected project.

Theory :

Use Case Diagram :

A use case describes how a user uses a system to accomplish a particular goal. A use case diagram consists of the system, the related use cases and actors and relates these to each other to visualize: what is being described?, who is using the system? and what do the actors want to achieve?, thus, use cases help ensure that the correct system is developed by capturing the requirements from the user's point of view.

A use case is a list of actions or event steps typically defining the interactions between a role of an actor and a system to achieve a goal. A use case is a useful technique for identifying, clarifying, and organizing system requirements. A use case is made up of set of possible sequences of interactions between systems and users that defines the features to be implemented and the resolution of any errors that may be encountered.

Scanned by CamScanner

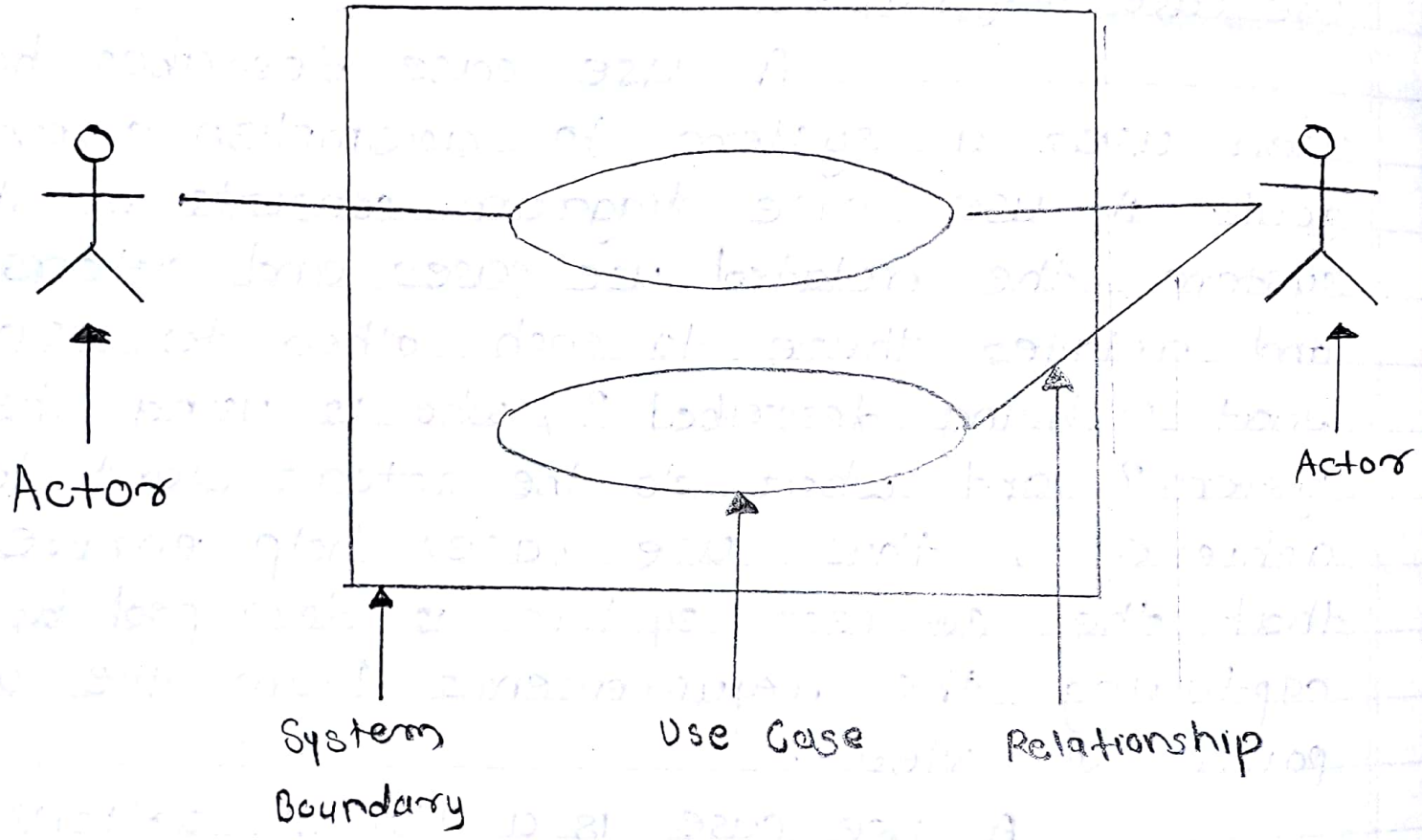


Fig. Use Case Diagram Notations

Scanned by CamScanner

Use Case Diagram Notations :

Use cases define interactions between external actors and the system to attain particular goals. A use case diagram contains four main components

a) Actors :

Actors are usually individuals involved with the system defined according to their roles. The actor can be a human or other external system.

b) Use Case :

A use case describes how actors uses a system to accomplish a particular goal. Use cases are typically initiated by a user to fulfill goals describing the activities and variants involved in attaining the goal.

c) Relationship :

The relationships between and among the actors and the use cases.

d) System Boundary :

The system boundary defines the system of interest in relation to the world around it.

Scanned by CamScanner

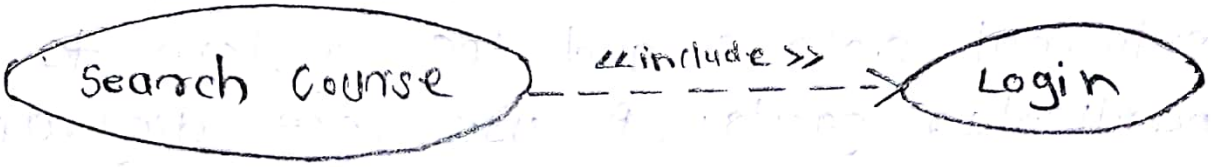


Fig. <<include>> use case

Fig. <<extend>> use case

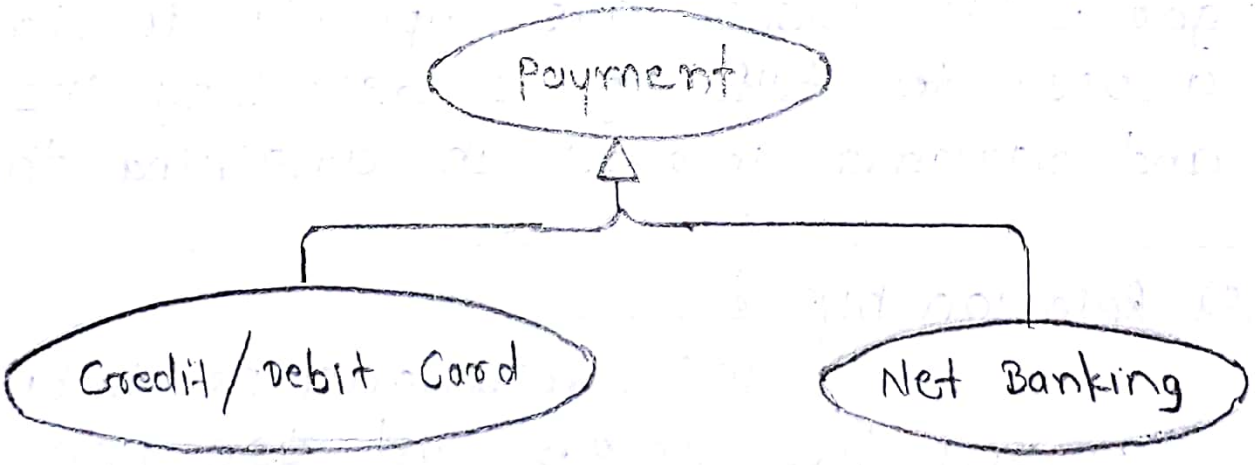


Fig. Generalized Use case

Scanned by CamScanner

Structuring Use Cases :

UML defines three stereotypes of association between use cases :

<< include >> Use case :

The time to use the << include >> relationship is after you have completed the first cut description of all your main use cases. You can now look at the use cases and identify common sequences of user-system interaction.

<< extend >> Use case :

An extending use case is, effectively, an alternate course of the base use case. The << extend >> use case accomplishes this by conceptually inserting additional action sequences into the base use-case sequence.

Abstract and generalized Use case :

The general use case is abstract. It can not be instantiated, as it contains incomplete information. The title of an abstract use case is shown in italics.

Conclusion :

Thus, we have studied and developed the use case diagram for selected project.

Scanned by CamScanner