

CS 521 – FINAL PROJECT

PROFESSIONAL DIRECTORY AND NETWORKING

CAREERNET

-Sudharshana, Vidya(A20472468)

vsudharshana@hawk.iit.edu

1. PROJECT OVERVIEW

CareerNet is a professional networking platform to connect with people professionally and is designed for business community. The primary goal of this application is to allow registered users to create professional networks of people they know and trust. The main aim of this platform is to apply for jobs, learn courses, get career consultations. Basic services like applying for jobs are offered for free and services like video courses, career consultations are offered for paid members. The secondary aim is to connect people from the companies they work within, seniority level, alumni of same institution and skill sets. All the system information is maintained on a database.

2. PROJECT REQUIREMENTS

- **Registration and Profile Management:**

The user must create an account using registration and update the date of birth, Resume, skills, summary and so on.

The user can purchase paid membership to access all the services offered in the career net.

- **Job Management:**

Job Management handles the lifecycle of user's job search.

- **Learning Management:**

Learning management handles the courses which are uploaded by organization, recruiters etc.

- **Career Counselling Management:**

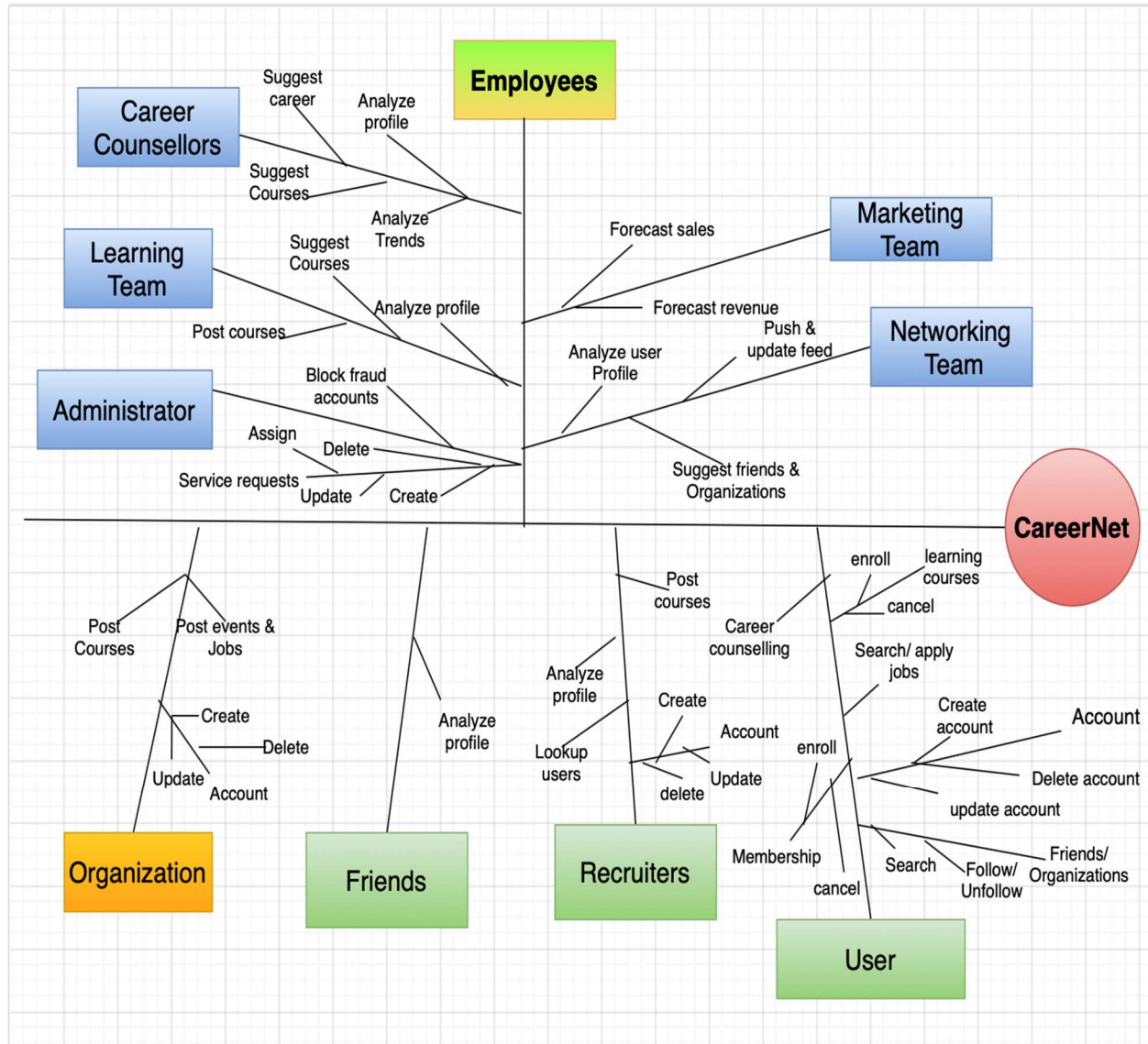
This management manages the services related to one's career, suggest them courses to gain potential and offers suggestions to users based on current trends. They can schedule one to one meeting with the candidates, recruiters and so on. Team also offer specialized career consultations for entry level, experienced and freelancing jobs.

- **Networking Management:**

The management which handles the user's network of friends and organizations and provide the updates through live feed and provide suggestions to follow the organizations and friends.

- The users can search using keywords relating to a person, job or a company.
- Recruiters and Organizations can post a new job and courses to the portal.
- The users also have access to online messaging which allows them to read or send online messages.
- The users will get the notifications for friend request, messages, job alerts, suggestions to follow/unfollow, make payments.

3. FEATURE TREE DIAGRAM



4. DOMAIN DICTIONARY

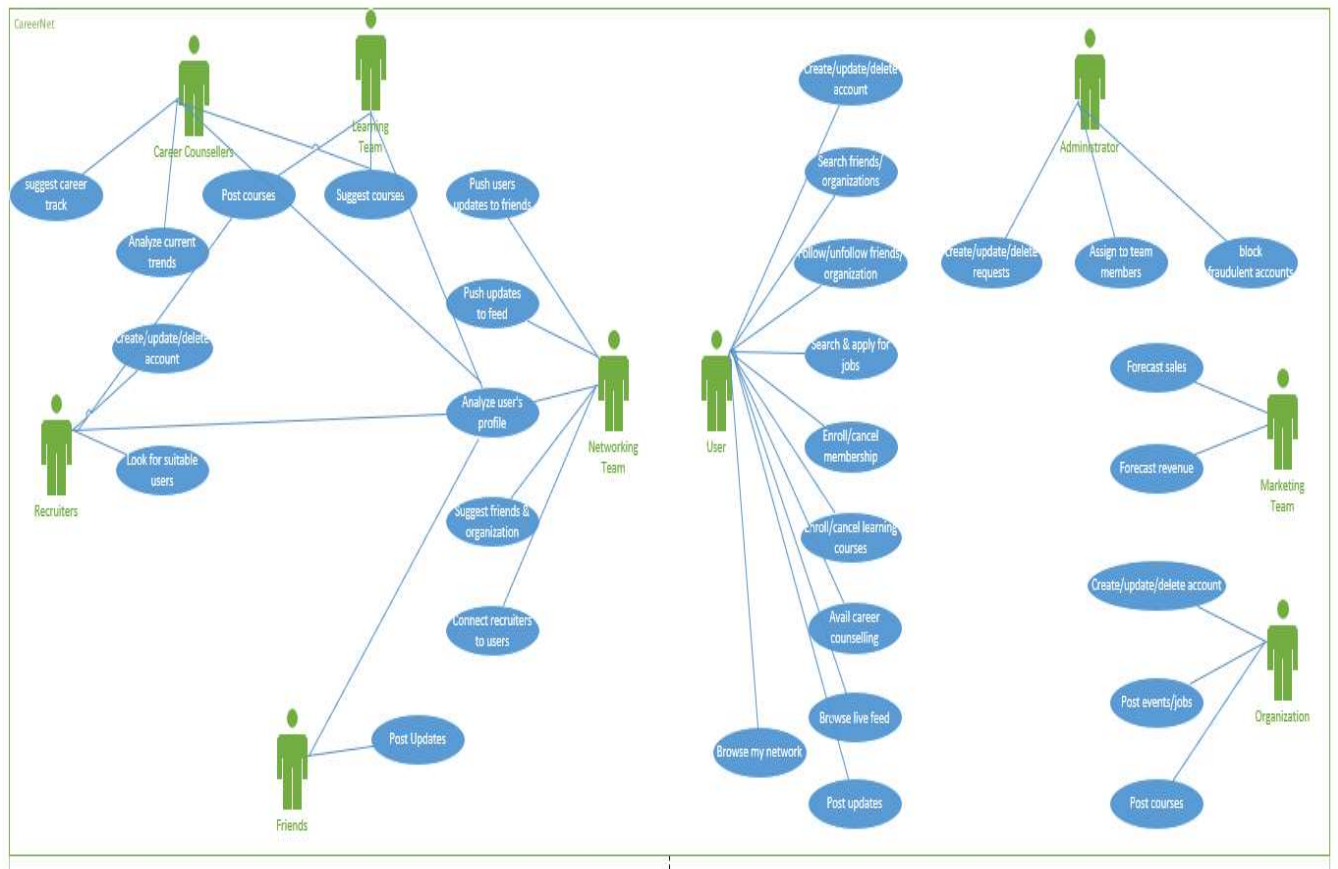
Name	Type	Description
User	Role	Create/Update/Delete account Search friends/organizations Follow/Unfollow friends/Organizations Search/Apply Jobs Enroll/Cancel membership Enroll/Cancel for learning courses Avail career counselling services See updates from friends & organizations Post updates Send/receive message to friends get job alerts
Account	Object	Information about user
Create account	function	create an account
update account	function	update an account
delete account	function	delete an account
My network	object	list of followed friends and organizations
Friends	object	list of friends
Organization	object	list of organizations
Search friends	function	search friends
search organizations	function	search organizations
Follow Friends	function	follow friends
Follow organizations	function	follow organization
Job	object	job details
search job	function	search for a job
apply for job	function	apply for a job
Enroll membership	function	enroll membership
cancel membership	function	cancel membership
Enroll learning courses	function	enroll learning courses
cancel learning courses	function	cancel learning course
course	object	Information about courses
Avail career counselling	process	avail career counselling
Feed	object	live feed of the user
browse feed	process	browse live feed
post update	function	post update to feed
message	object	users messages
send message	function	send message
receive message	function	receive message from friends

Administrator	Role	Create/Update/Delete service requests Assign to respective team members Block fraudulent accounts
Request	object	information about requests
create request	function	create a new service request
update request	function	update a request
delete request	function	delete a request
assign request	function	assign a service request to team members
block fraudulent accounts	process	block fraudulent accounts
Networking Team	Role	Analyze user activity Suggest friends & organizations to users Push contents from friends & organizations to live feed Push user updates to friends
analyze profile	function	analyze user profile
suggestuser	function	suggest friends & organization to users
pushUpdate	function	push users updates to friends
Marketing Team	Role	Forecast sales Forecast revenue
forecast sales	function	forecast sales
forecast revenue	function	forecast revenue
Learning Team	Role	Analyze user's profile Suggest users to enroll for courses Post online courses
suggestuser	function	suggest courses to users
post courses	function	post new courses to library
Career Counsellors	Role	Analyze user's profile Analyze current trends Suggest users to enroll for courses Cousills users with respect to career
analyze trends	function	analyze current trends
counselUser	function	give career suggestions to users
Organization	Role	Create/Update/Delete account Post organization events/jobs Post online courses
post events	function	post organization events
post jobs	function	post openings in organizations
Recruiters	Role	Create/Delete account Look for suitable users Analyze users profile Post online courses
lookup users	function	lookup for suitable users
Friends	Role	Analyze users profile
Notifications	object	Alerts for users
Payment	object	Various payment methods for users
Search	Object	class uses for all tyoe of searches

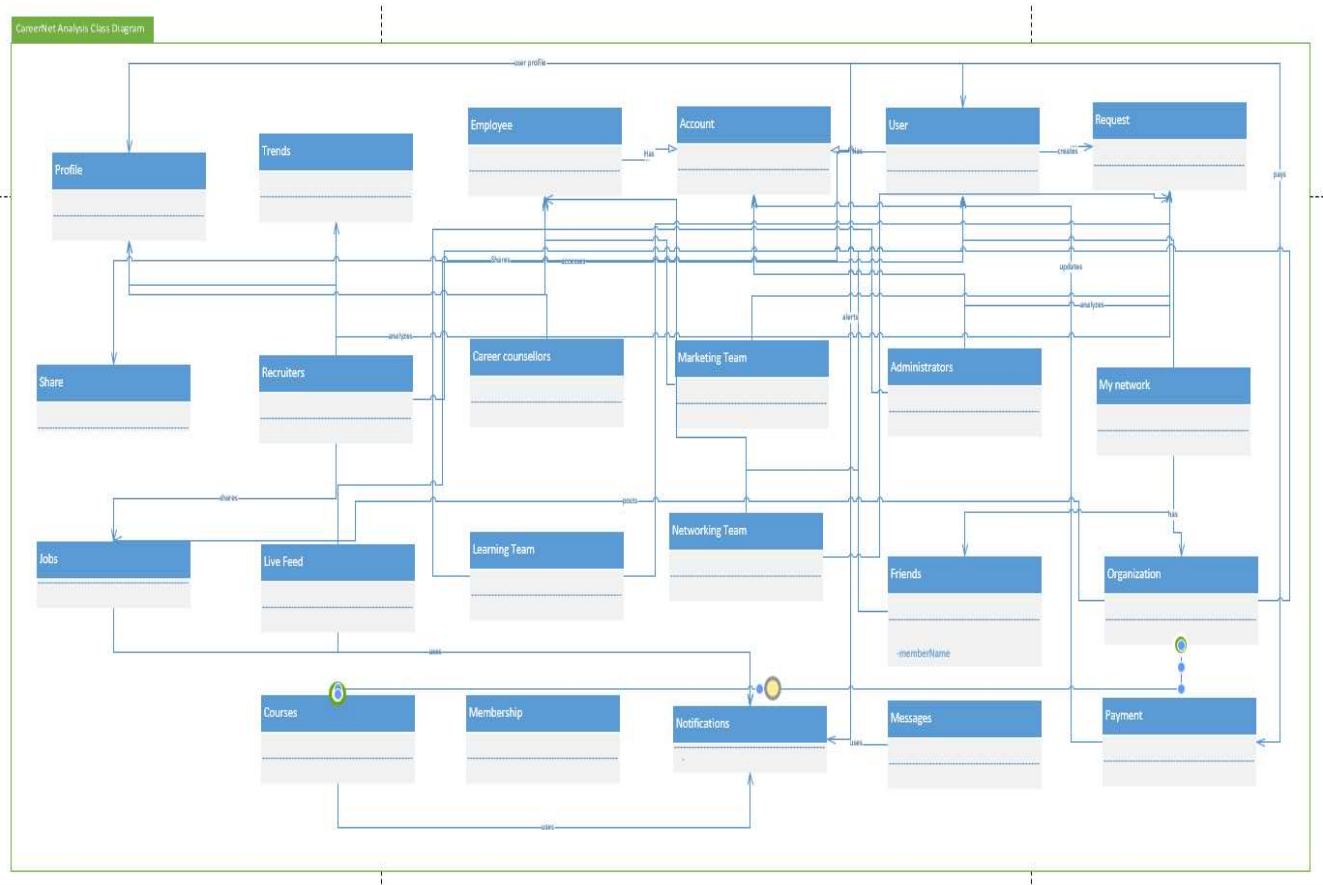
5. ACTOR DICTIONARY

Actor	Description	Abstract	Use Case
User	Create/Update/Delete account View/update profile Search friends/organizations Follow/Unfollow friends/Organizations Search/Apply Jobs Enroll/Cancel membership Enroll/Cancel for learning courses Avail career counselling services See updates from friends & organizations Post updates Send/receive message to friends Browse friends list		Create/Update/Delete account View/update profile Search friends/organizations Follow/Unfollow friends/Organizations Search/Apply Jobs Enroll/Cancel membership Enroll/Cancel for learning courses Avail career counselling services See updates from friends & organizations Post updates Send/receive message to friends Browse friends list
Administrator	Create/Update/Delete service requests Assign to respective team members Block fraudulent accounts		Create/Update/Delete service requests Assign to respective team members Block fraudulent accounts
Networking Team	Analyze user activity Suggest friends & organizations to users Push contents from friends & organizations to live feed Push user updates to friends		Analyze user activity Suggest friends & organizations to users Push contents from friends & organizations to live feed Push user updates to friends
Marketing Team	Forecast sales Forecast revenue		Forecast sales Forecast revenue
Learning Team	Analyze user's profile Suggest users to enroll for courses Post online courses		Analyze user's profile Suggest users to enroll for courses Post online courses
Career Counselors	Analyze user's profile Analyze current trends Suggest users to enroll for courses Counsels users with respect to career		Analyze user's profile Analyze current trends Suggest users to enroll for courses Counsels users with respect to career
Organization	Create/update/Delete account Post organization events/jobs Post online courses		Create/update/Delete account Post organization events/jobs Post online courses
Recruiters	Create/update/Delete account Look for suitable users Analyze users profile Post online courses		Create/update/Delete account Look for suitable users Analyze users profile Post online courses
Friends	Analyze users profile		Analyze users profile

6. USE CASE DIAGRAM

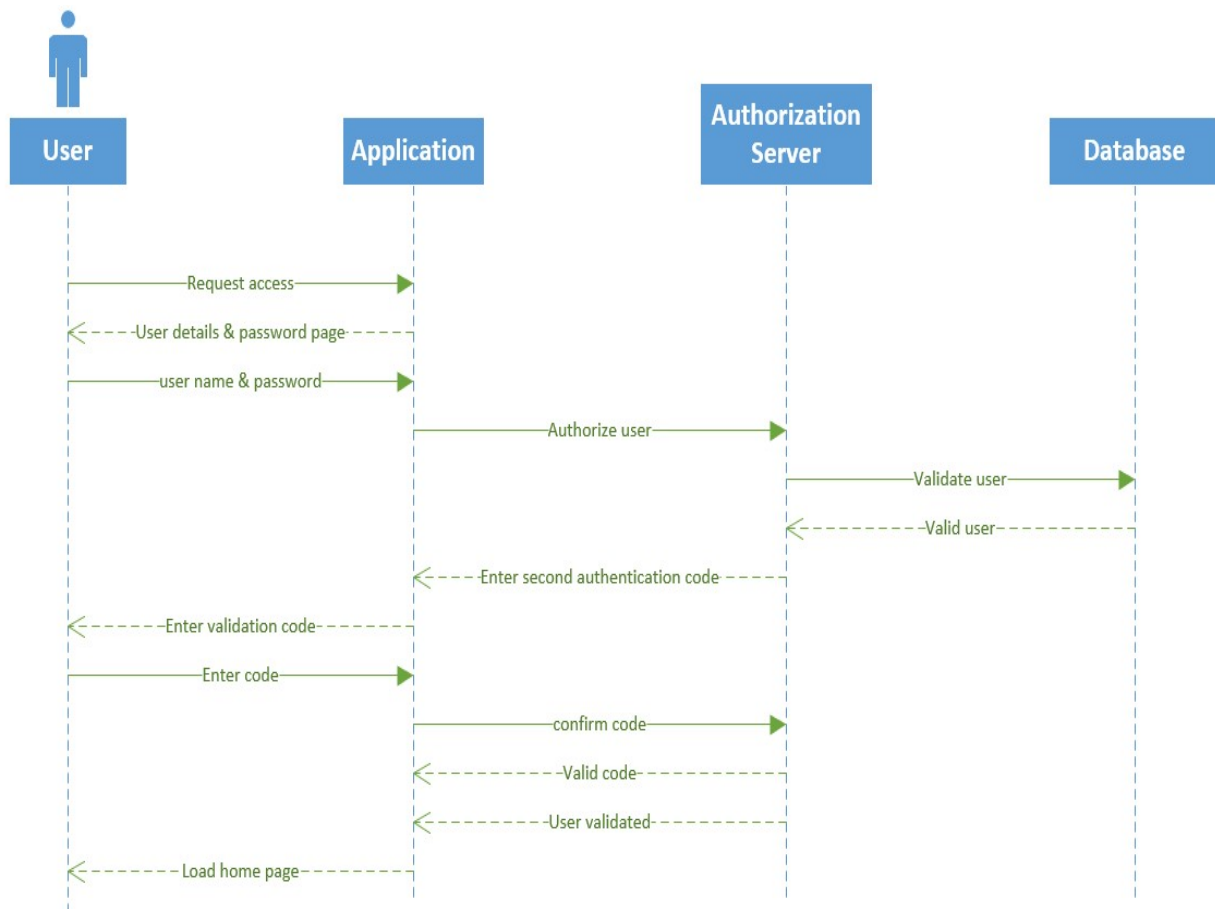


7. ANALYSIS MODEL

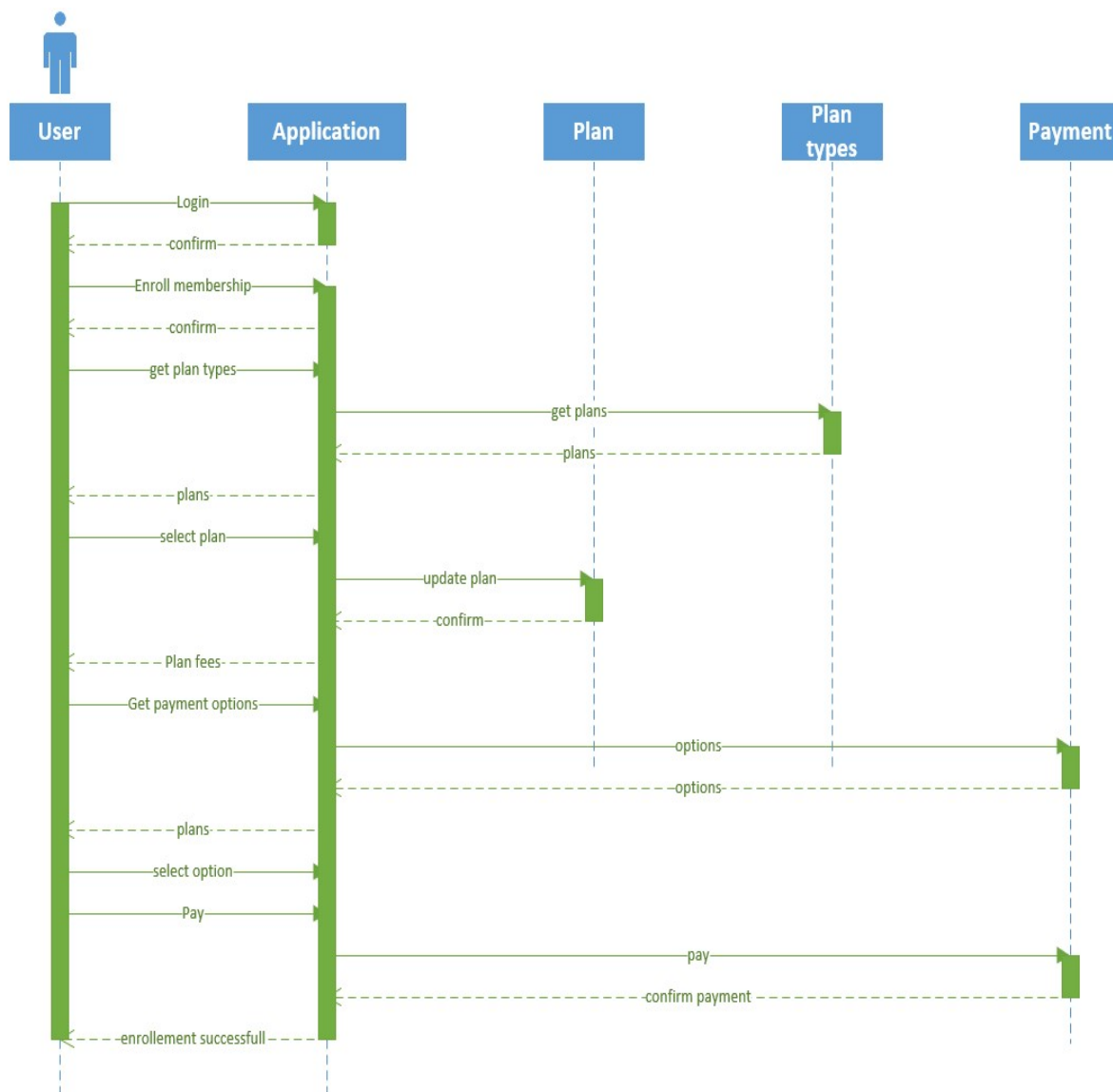


8. INTERACTION SEQUENCE DIAGRAMS

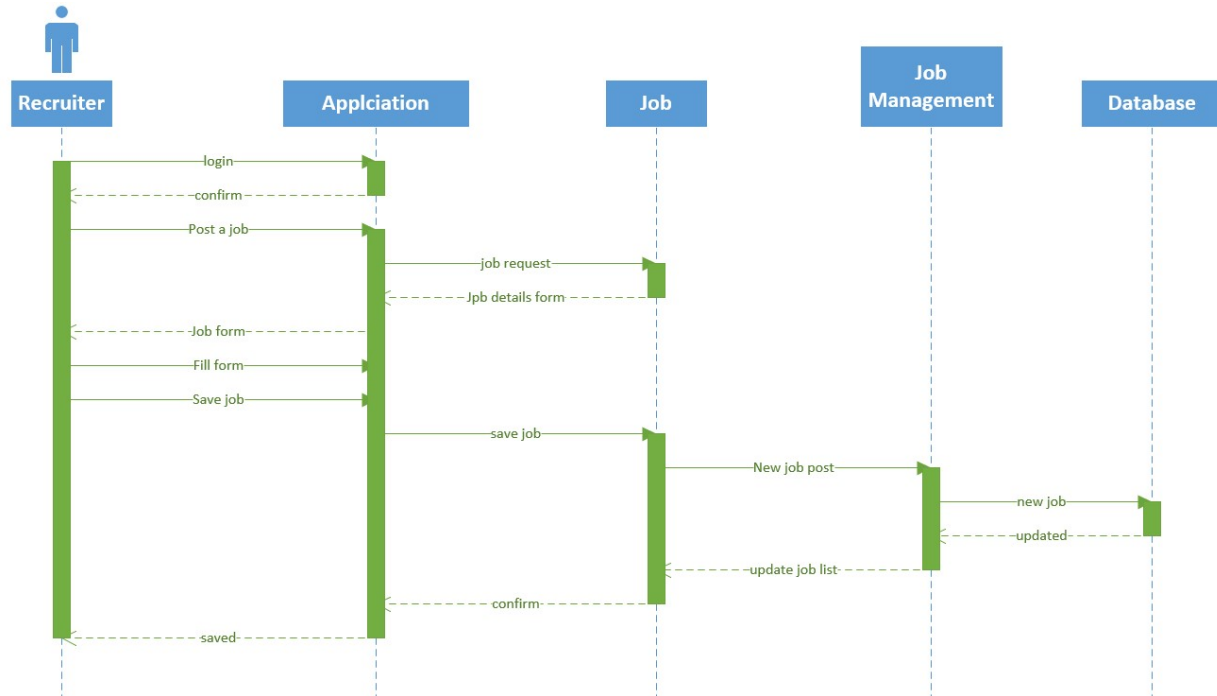
- INTERACTION SEQUENCE DIAGRAM FOR USER AUTHENTICATION



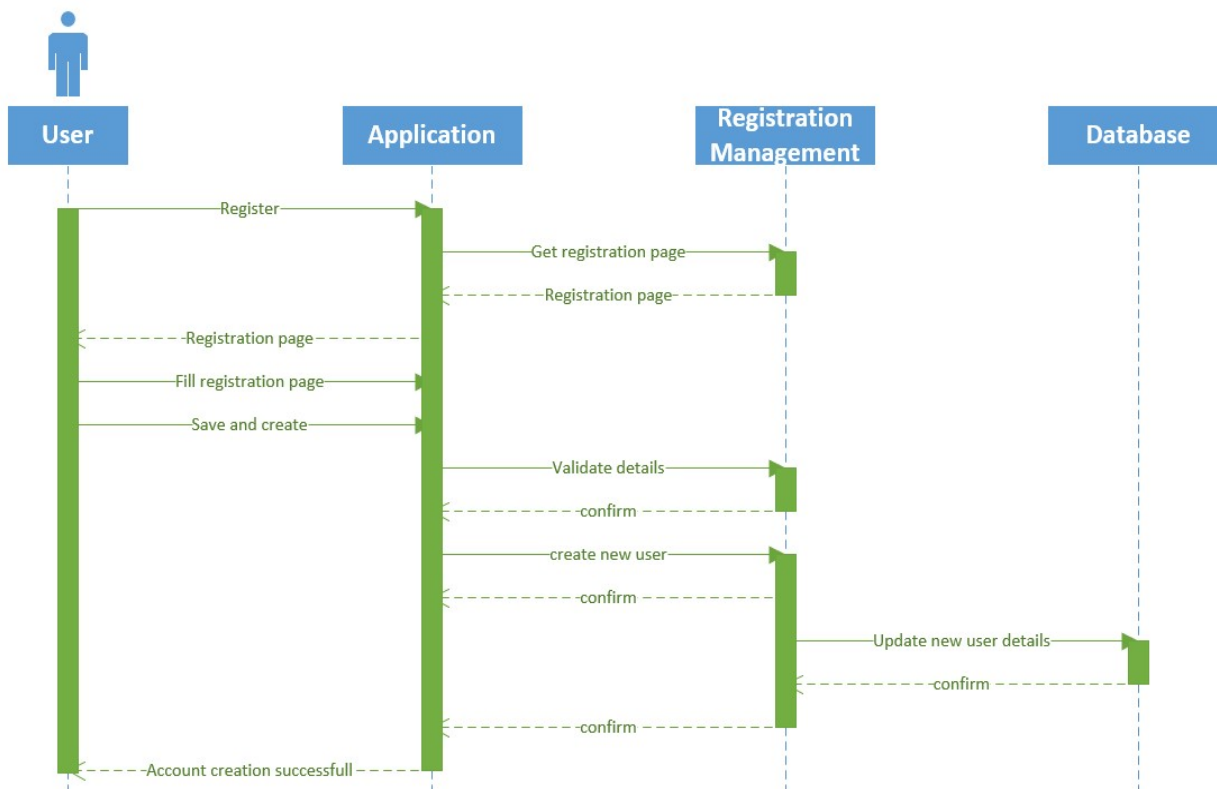
- INTERACTION SEQUENCE DIAGRAM FOR ENROLLING IN MEMBERSHIP



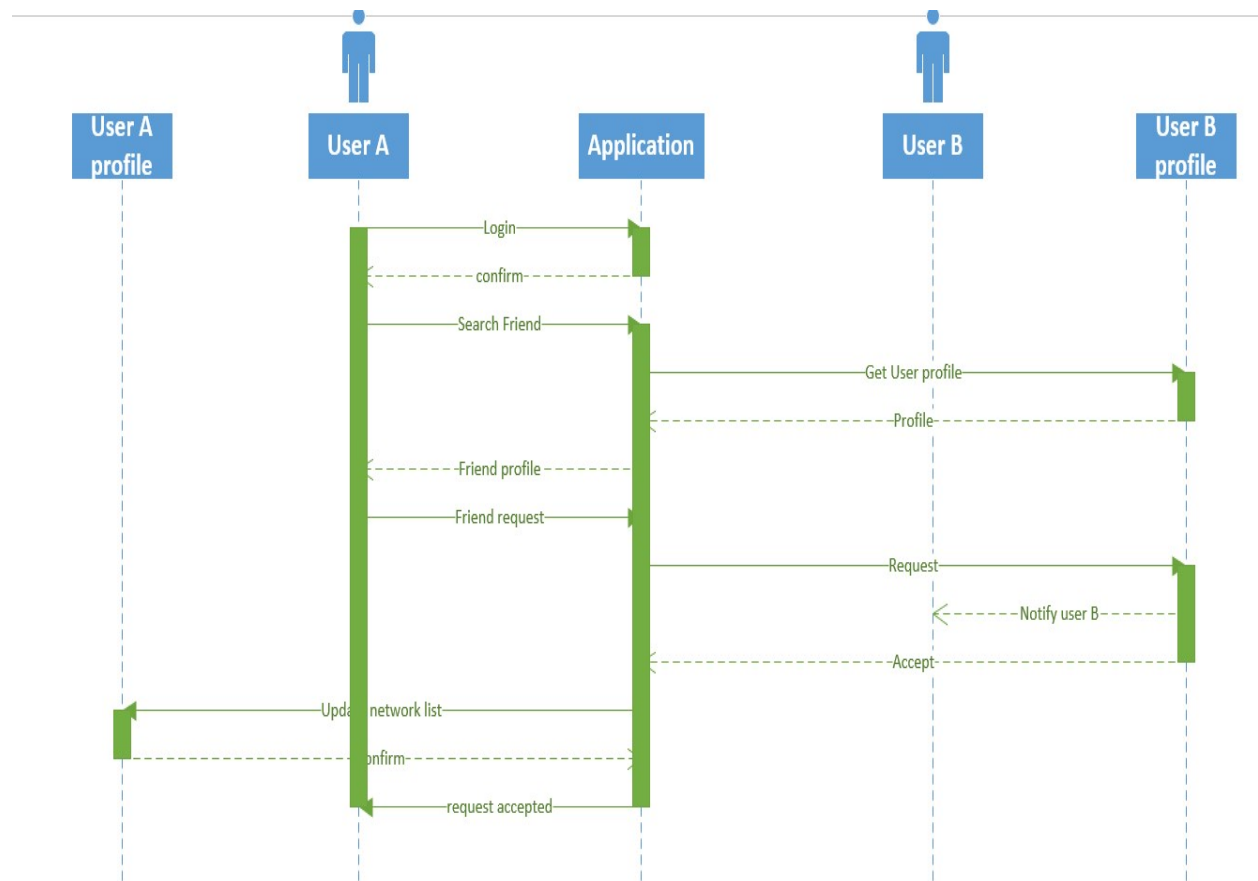
- INTERACTION SEQUENCE DIAGRAM FOR POSTING A JOB



- INTERACTION SEQUENCE DIAGRAM FOR CREATING A NEW ACCOUNT

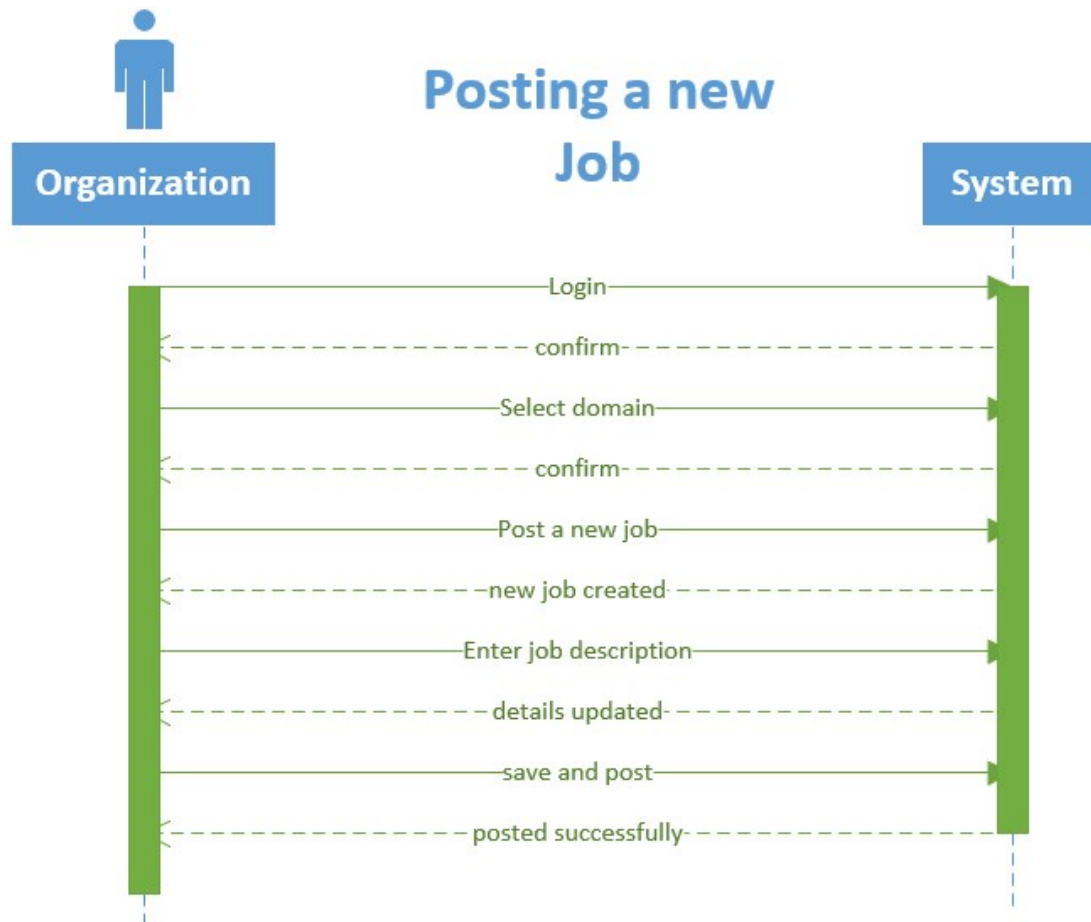


- INTERACTION SEQUENCE DIAGRAM FOR SENDING A FRIEND REQUEST

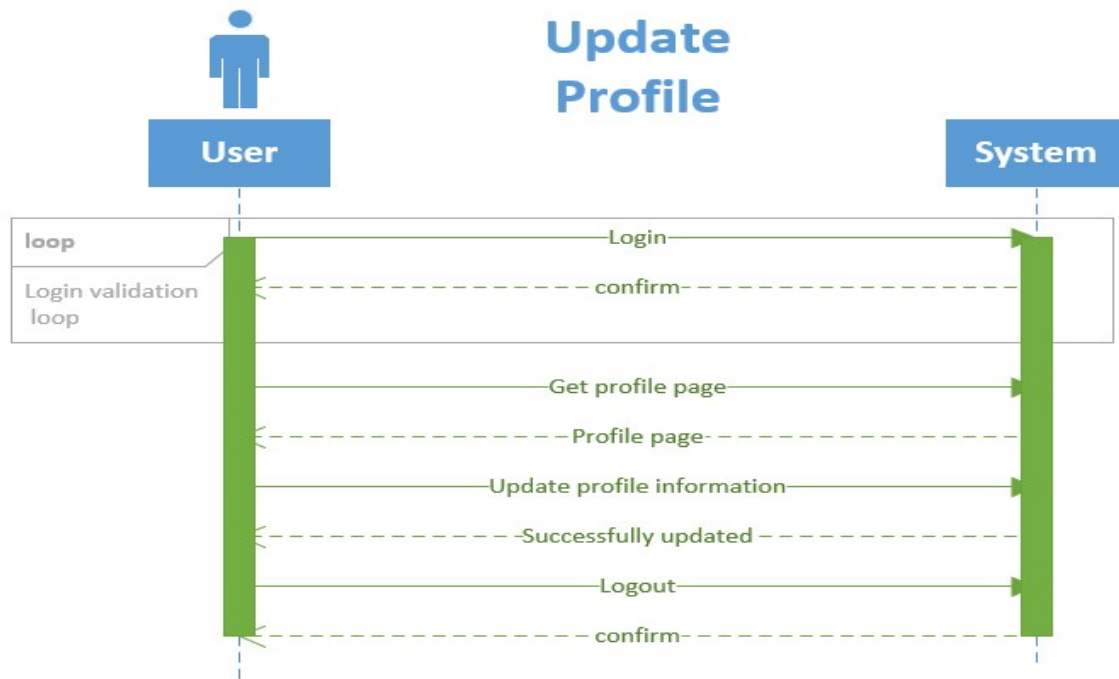


9. SYSTEM SEQUENCE DIAGRAMS

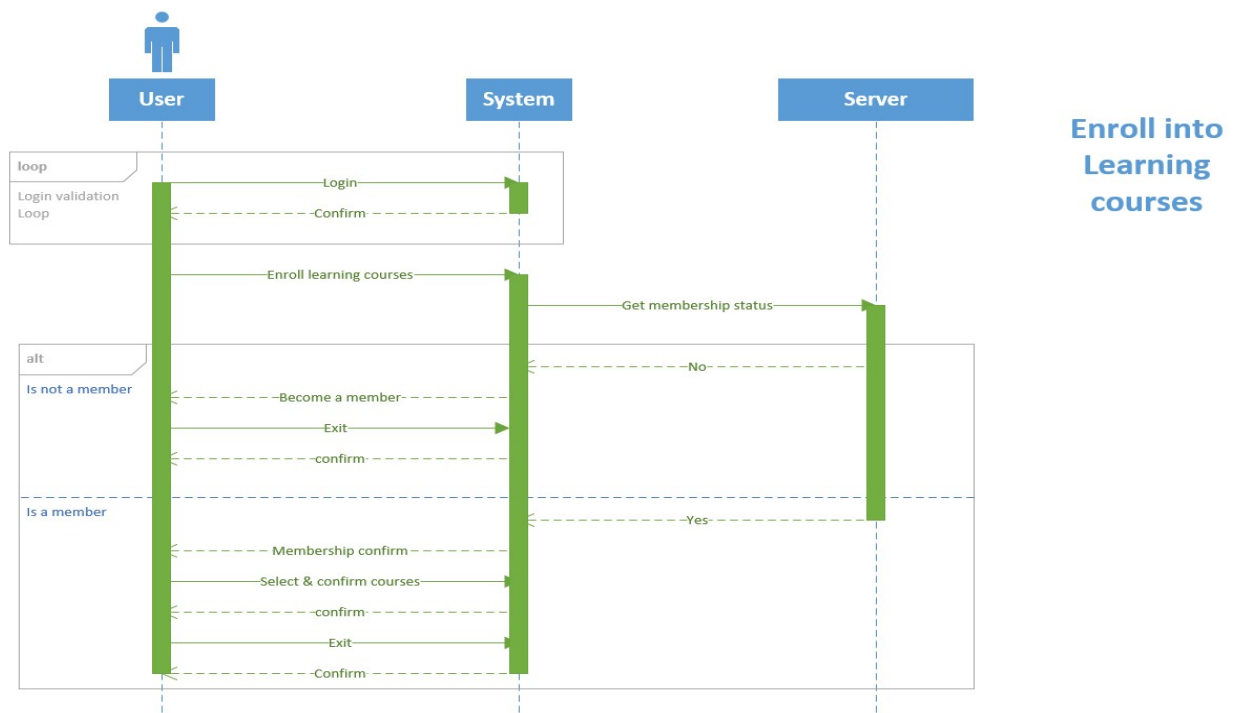
- SYSTEM SEQUENCE DIAGRAM FOR POSTING A NEW JOB



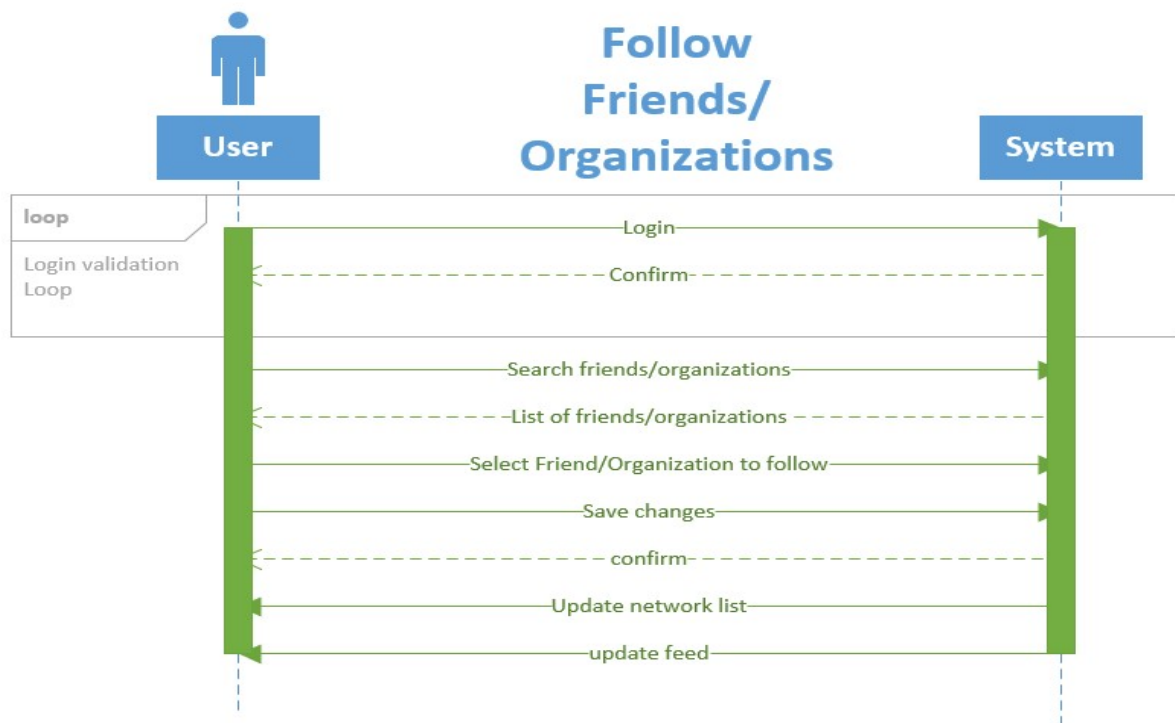
- SYSTEM SEQUENCE DIAGRAM FOR UPDATE A PROFILE



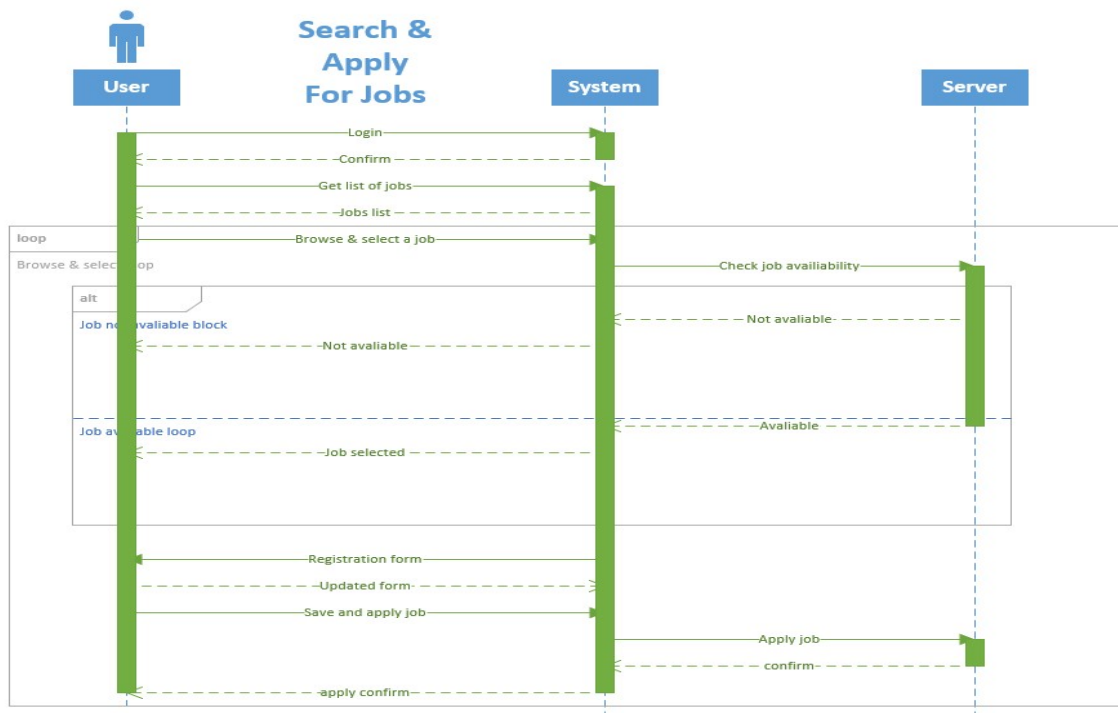
- SYSTEM SEQUENCE DIAGRAM FOR ENROLL INTO LEARNING COURSES



- SYSTEM SEQUENCE DIAGRAM TO FOLLOW FRIENDS/ORGANIZATIONS

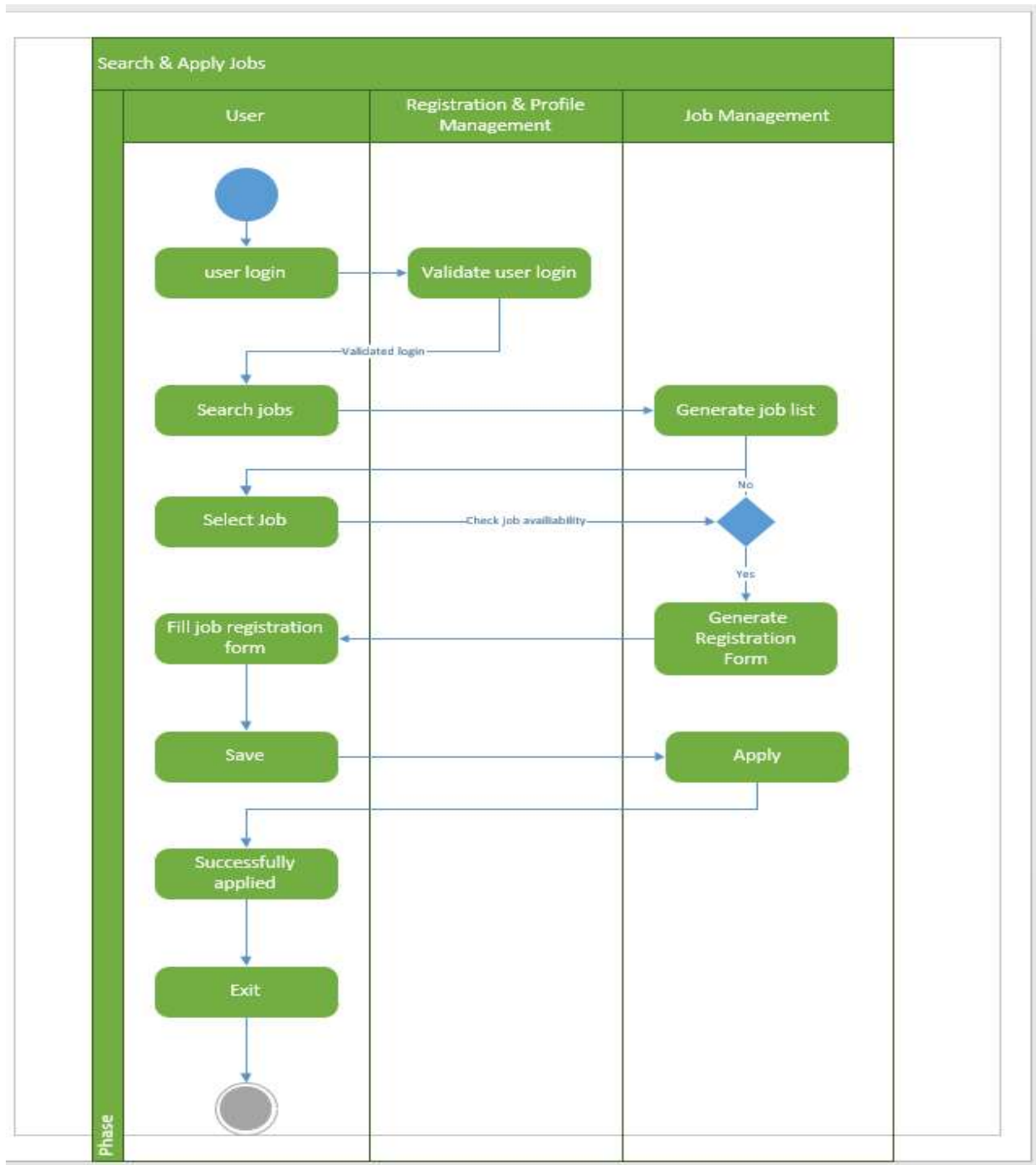


- SYSTEM SEQUENCE DIAGRAM FOR SEARCH & APPLY JOBS

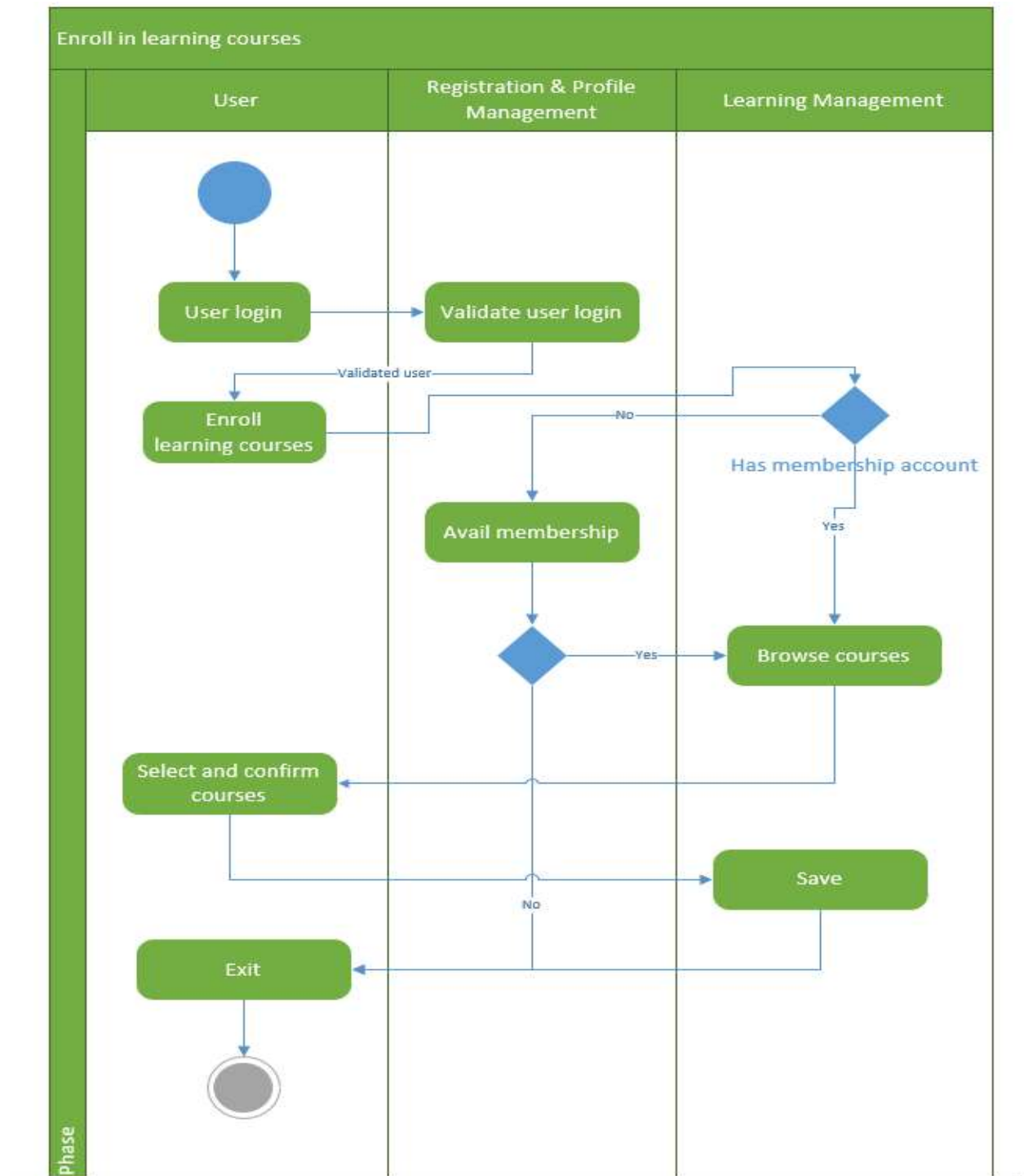


10.ACTIVITY DIAGRAMS

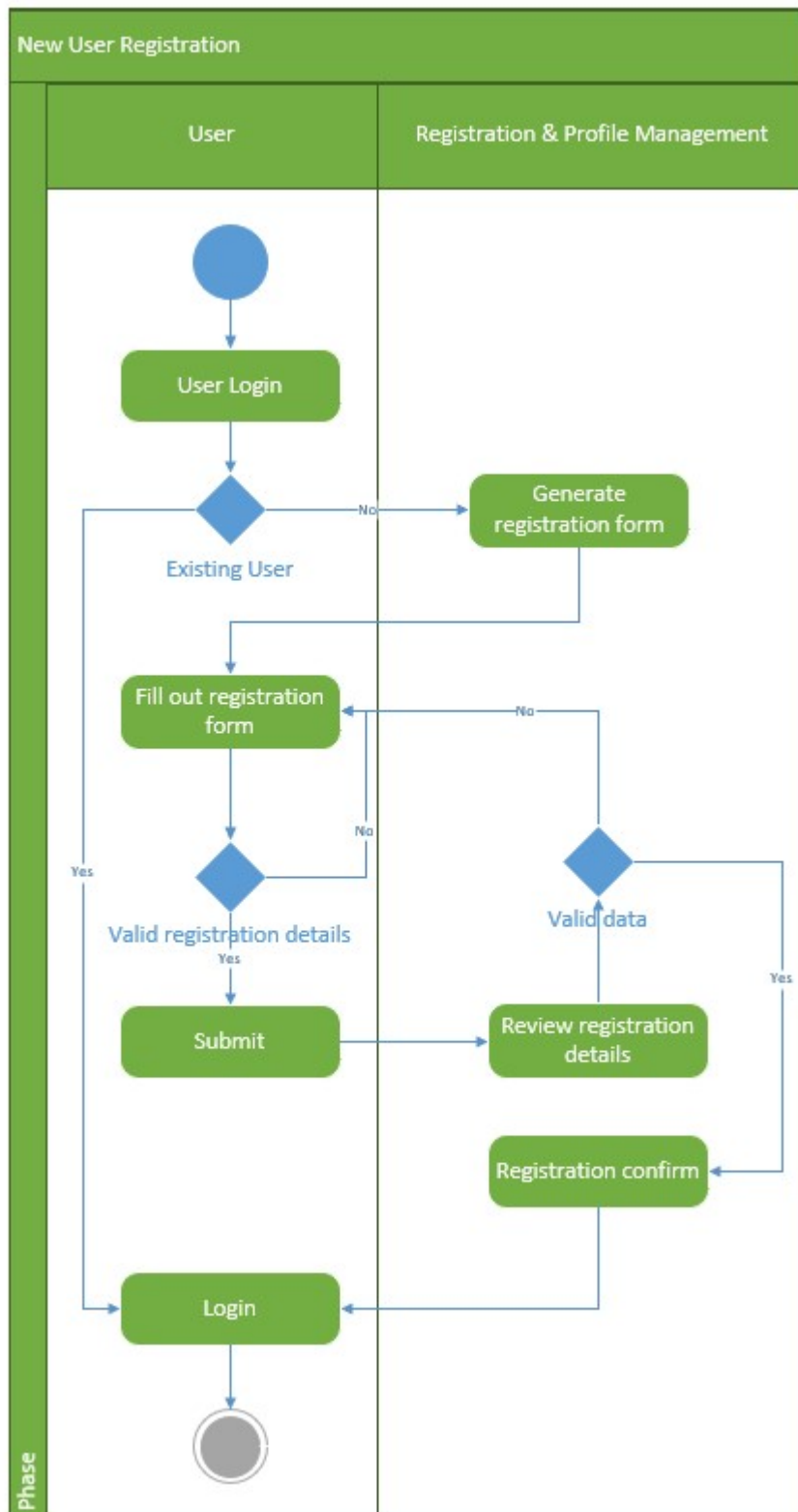
- ACTIVITY DIAGRAM FOR SEARCH & APPLY JOBS



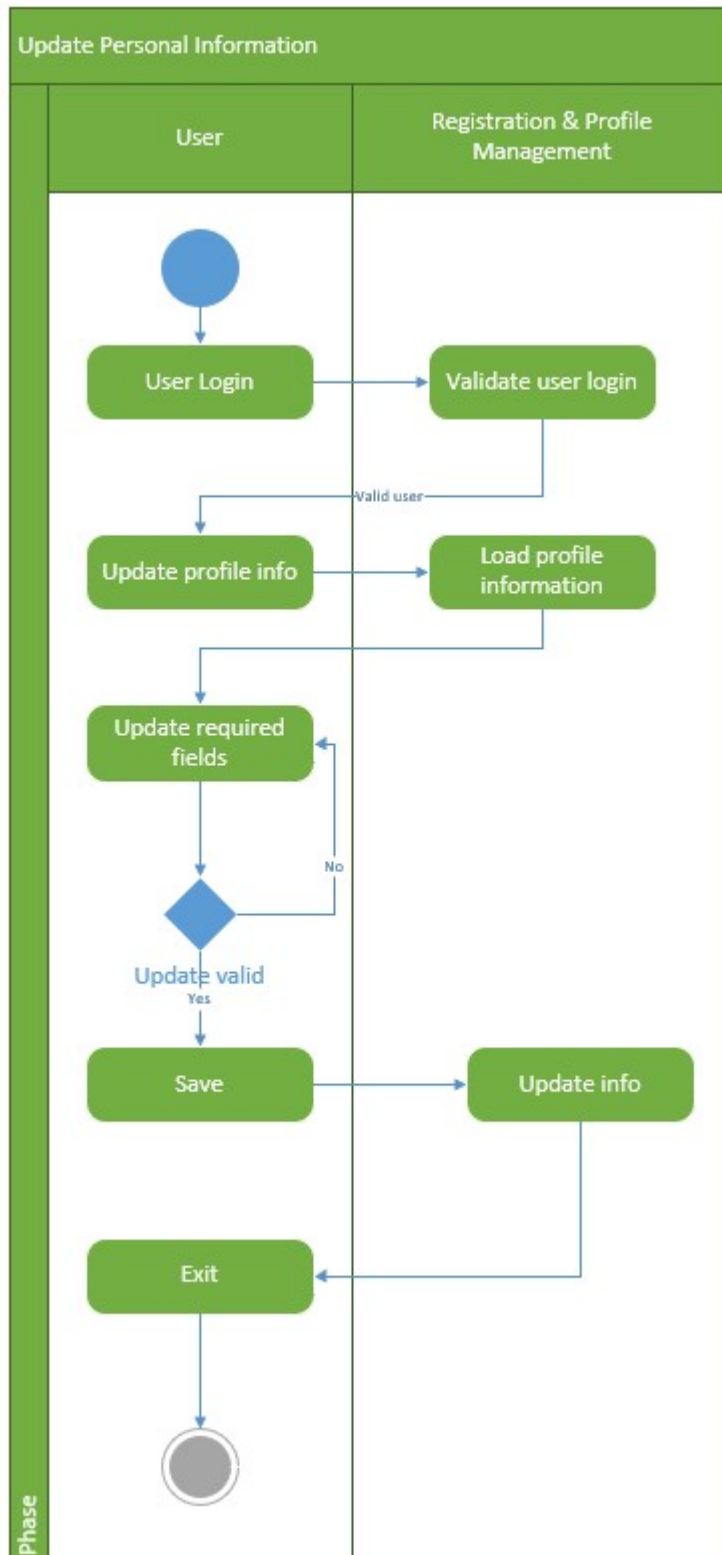
- ACTIVITY DIAGRAM FOR ENROLL IN LEARNING COURSES



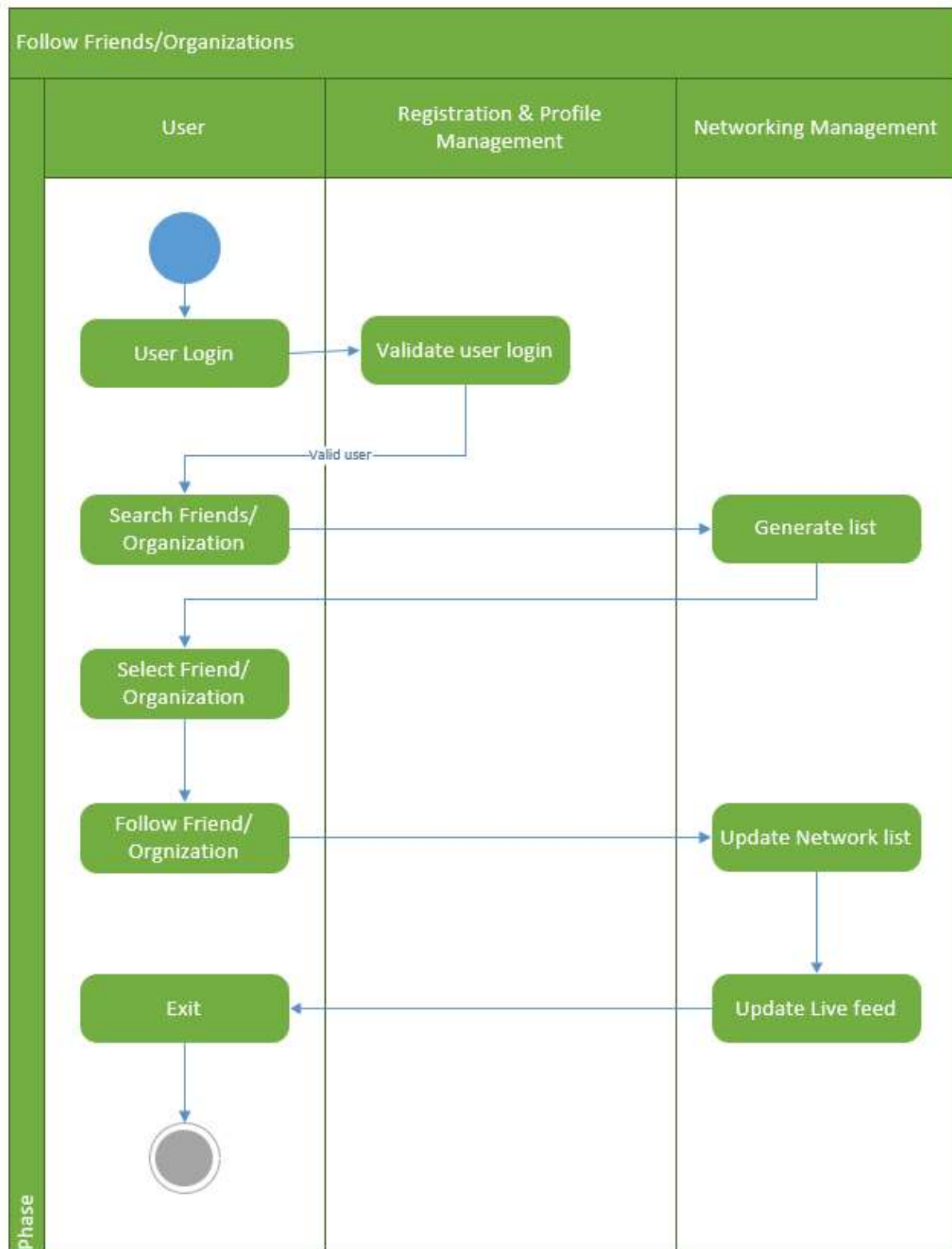
- ACTIVITY DIAGRAM FOR USER REGISTRATION



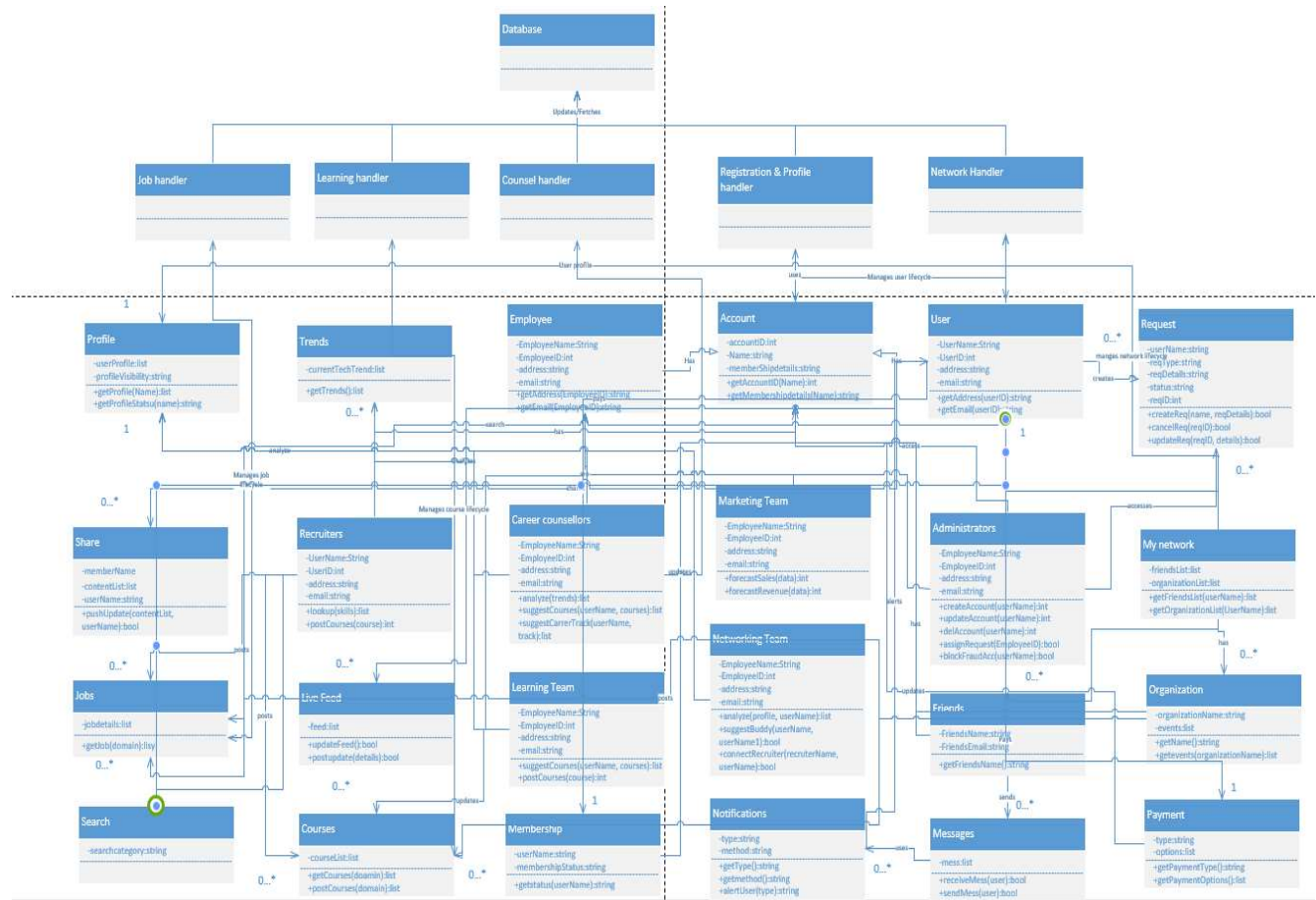
- ACTIVITY DIAGRAM FOR UPDATE PERSONAL INFORMATION



- ACTIVITY DIAGRAM TO FOLLOW FRIENDS/ORGANIZATIONS

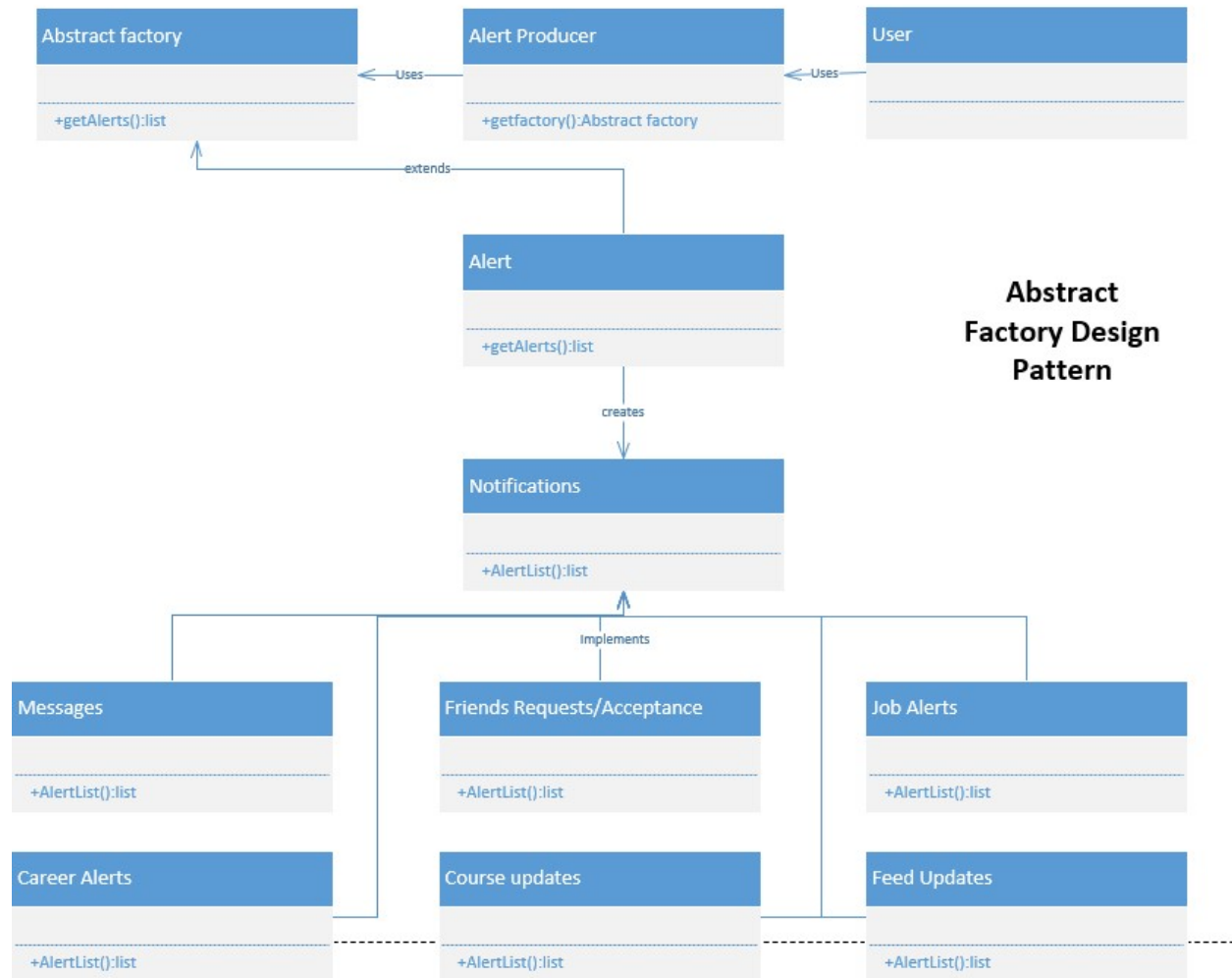


11.DESIGN MODEL



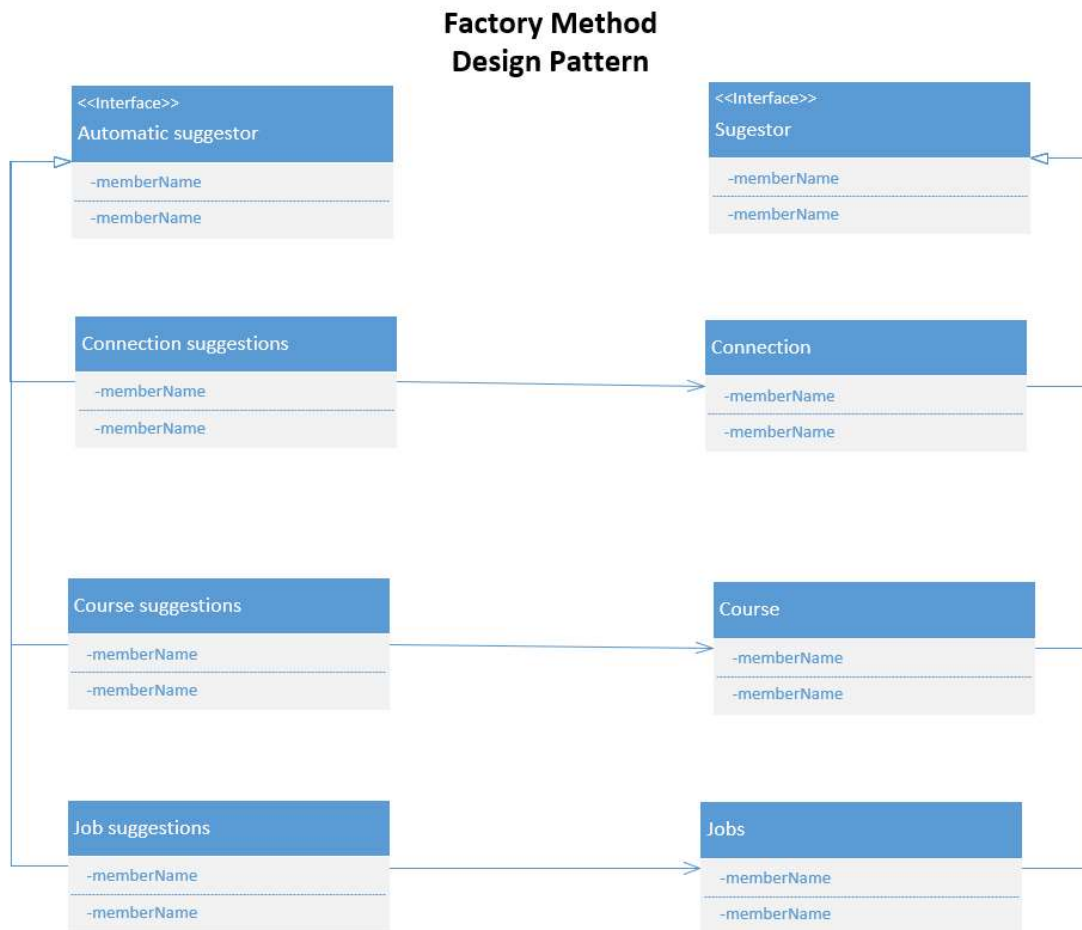
12.DOCUMENTATION AND CLASS DIAGRAMS FOR DESIGN PATTERNS USED

- ABSTRACT FACTORY DESIGN PATTERN:



Abstract factory design refers to a factory of factories, responsible for creating a family of related objects without explicitly specifying their classes. In my design, there are a lot of functionalities which generate alerts and notifications to the user and can easily be represented as a factory of related objects.

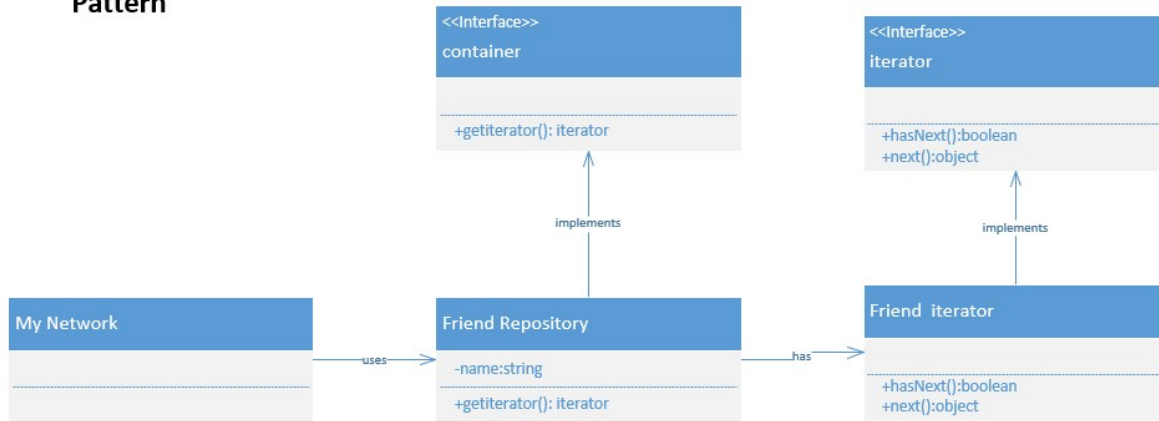
- FACTORY METHOD DESIGN PATTERN:



It is a creational pattern used to deal with creation of objects. In my design, I use factory method design pattern for automatic suggestions for connections or courses or jobs. Based on the type of user we select one from them.

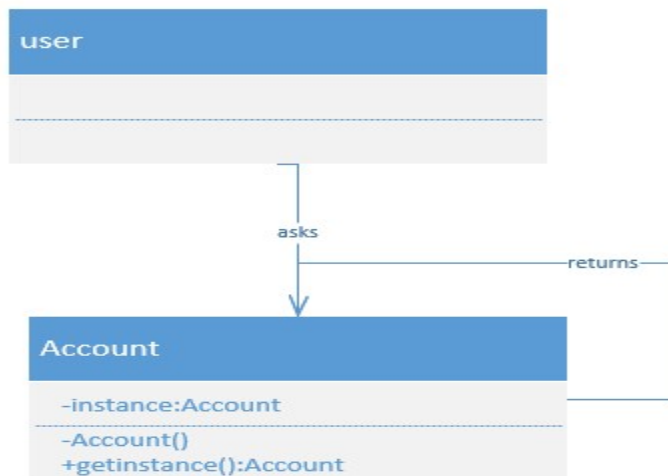
- ITERATOR DESIGN PATTERN:

Iterator Design Pattern



Iterator design pattern is used to get a way to access the elements of a collection object in sequential manner without any need to know its underlying representation. In my design I use this design concept to iterate over the friends list in My Network.

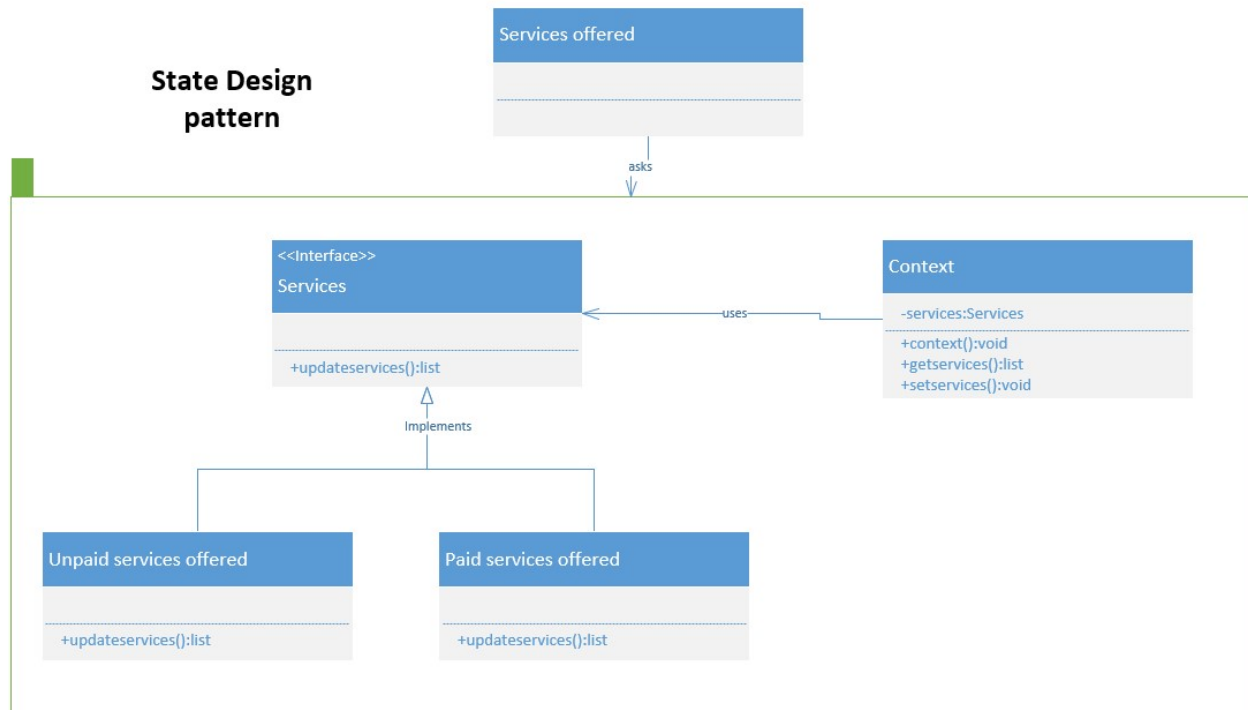
SINGLETON DESIGN PATTERN:



Singleton Design pattern

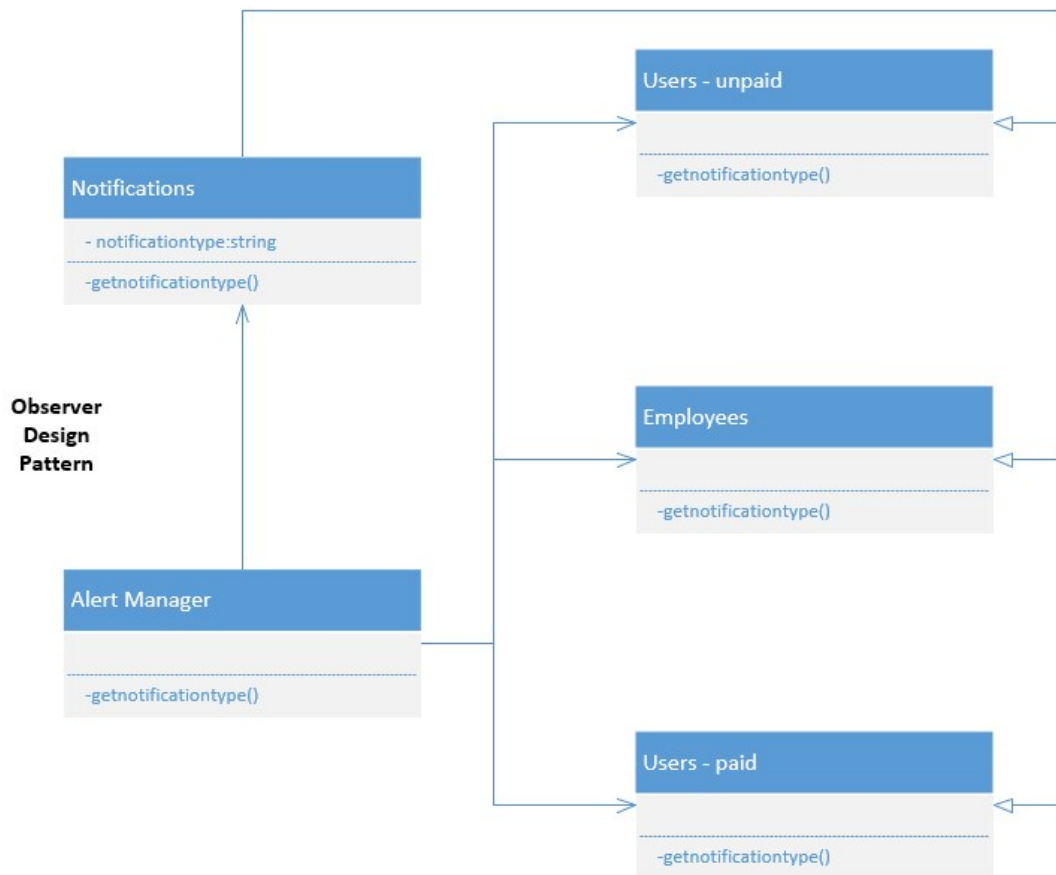
Singleton object involves a single class which is responsible to create an object while making sure that only single object gets created. In my design I used Account class as an singleton object as only one account can be created per user.

STATE DESIGN PATTERN:



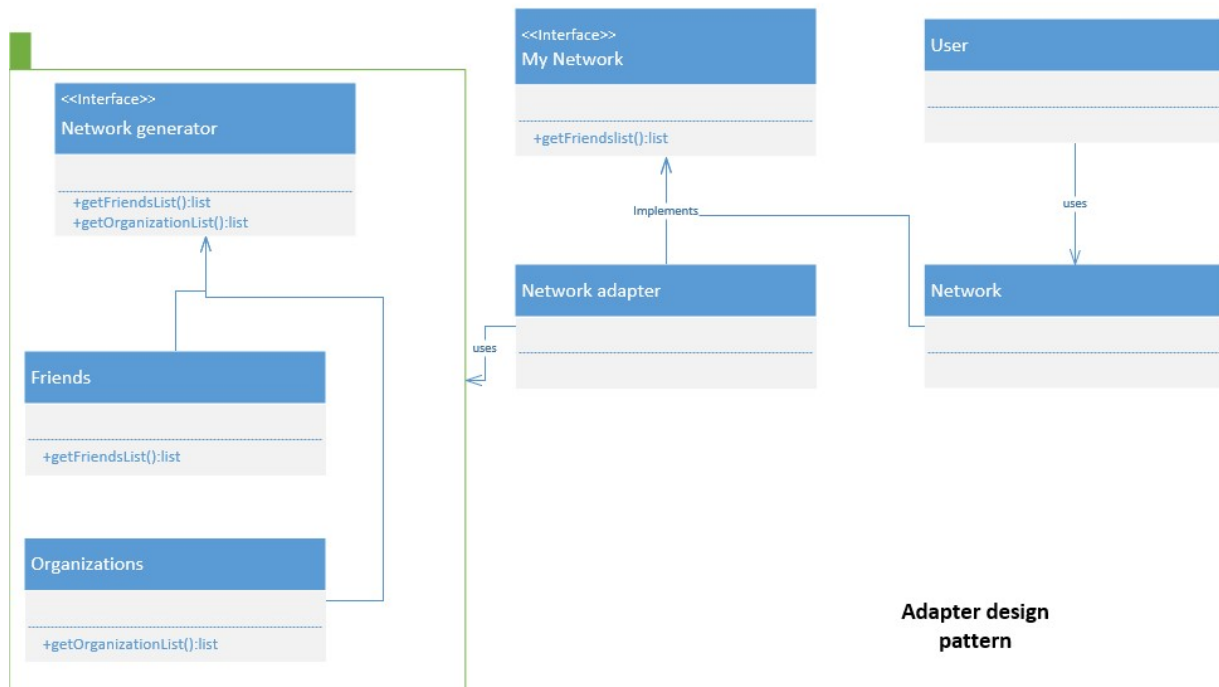
In State design pattern objects are created which represent various states and a context object whose behavior changes as its state object changes. Based on the membership status of the user, services offered to them will vary. Hence, I used a state design pattern to determine the services that can be offered to users.

- OBSERVER DESIGN PATTERN



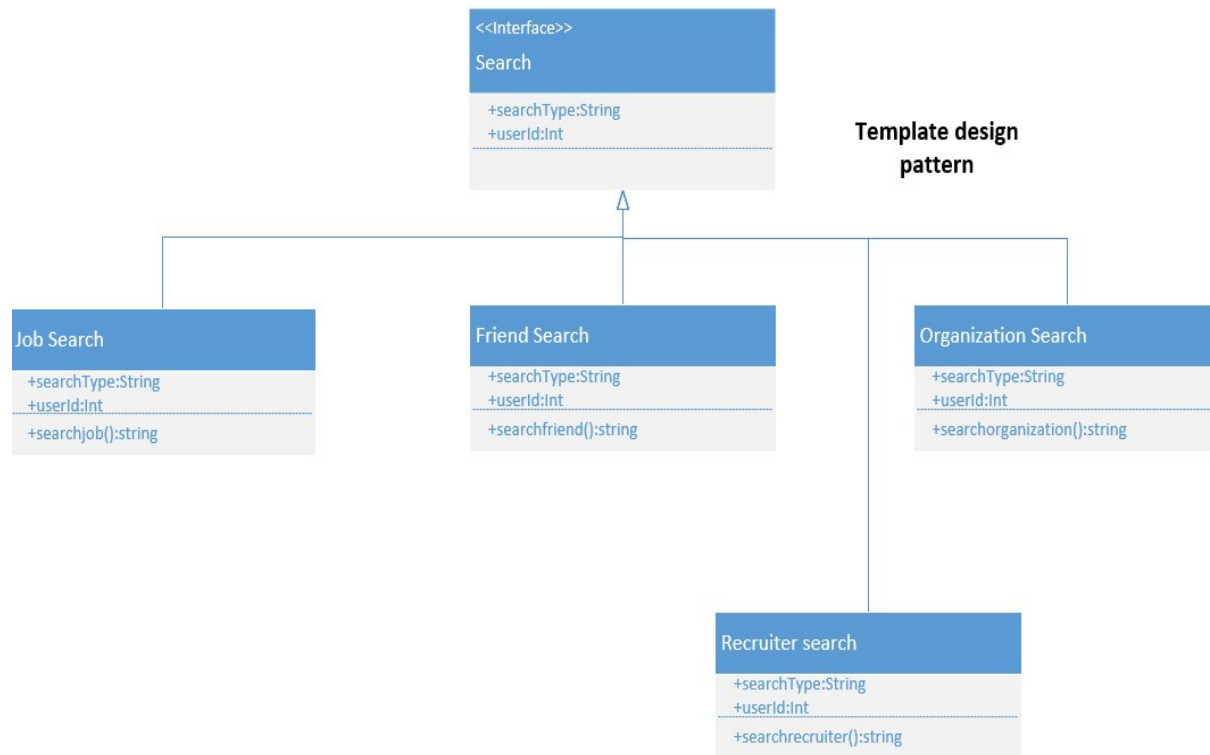
This pattern comes under behavioral pattern category. It describes the coupling between the objects and the observer. It supports for broad-cast type communication. Usage of Observer design pattern, when altering the state of one object necessitates changing the state of other objects, and the exact collection of objects is unknown or changes dynamically. When the change of a state in one object must be reflected in another object without keeping the objects tightly coupled. Helps in developing framework which needs to be enhanced in future with new observers with minimal changes.

- ADAPTER DESIGN PATTERN:



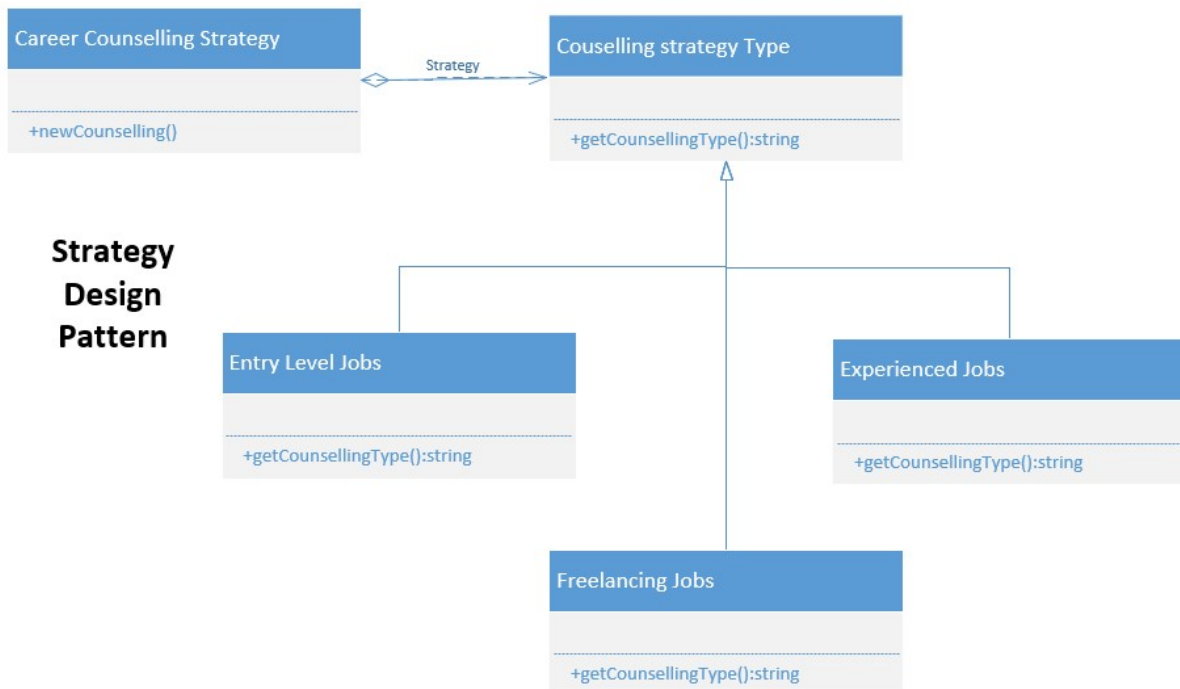
Adapter pattern works as a bridge between two incompatible interfaces. This type of design pattern comes under structural pattern as this pattern combines the capability of two independent interfaces. This pattern involves a single class which is responsible to join functionalities of independent or incompatible interfaces. In my design two different classes Friends and Organizations are bridged to form a list to be used for populating My Network

- TEMPLATE DESIGN PATTERN:



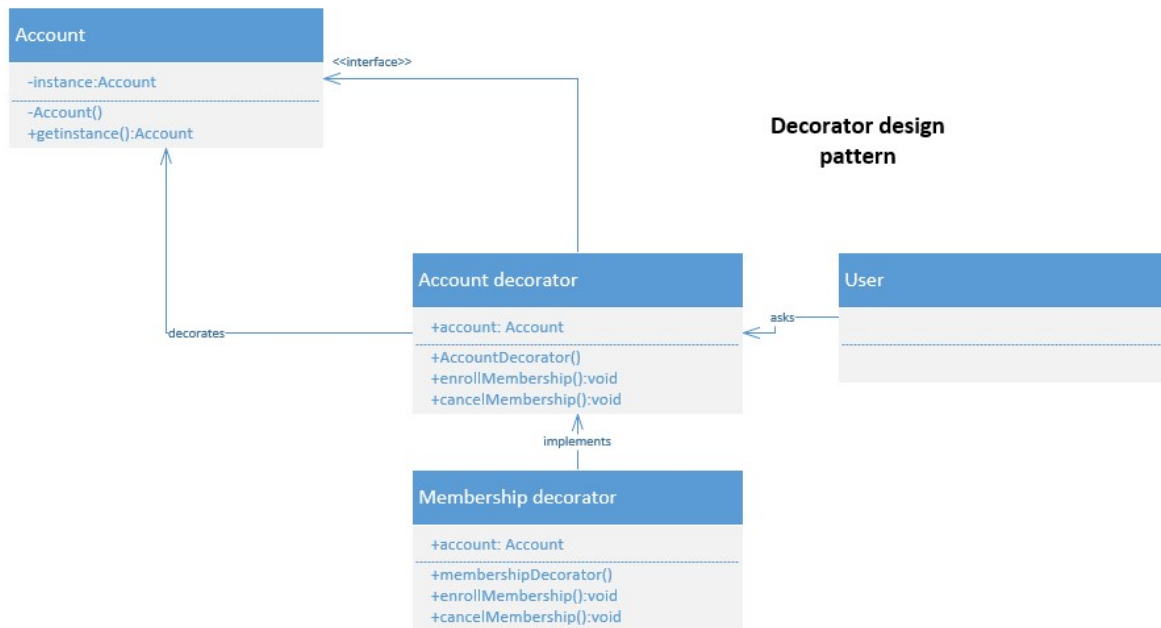
It is used to define a skeleton of an algorithm where some steps are deferred in client's subclass. In our design, we used Template Method Design Pattern for searching jobs. Based on the user's criteria we provide search results and suggestions.

- STRATEGY DESIGN PATTERN



This pattern comes under behavioral pattern category. It provides a substitute to subclassing. Use the Strategy pattern when you want to use different variants of an algorithm within an object and be able to switch from one algorithm to another during runtime, when we have a bunch of similar classes that just differ in how they perform some behavior's original object delegates execution to one of these objects, instead of implementing all variants of the algorithm. The multiple class differs only in the behaviors. It will be used when we need different variations of an algorithm.

- DECORATOR DESIGN PATTERN:



Decorator pattern allows a user to add new functionality to an existing object without altering its structure. This pattern creates a decorator class which wraps the original class and provides additional functionality keeping class methods signature intact. In my design, membership details are added as a wrapper on top of Account class, as membership will not be availed by all, it has been kept as a separate wrapper.

I have not used composite, proxy, Façade design patterns in our design. In my design there was no requirement for proxy classes, hence I did not use proxy design pattern. Since, I am not using any libraries and did not want any interface to them I did not use Façade design pattern. In my design I did not have any structure to satisfy tree structure and hence I did not use composite design pattern.