IT314

Lab6 - Domain Analysis Modelling
Software Engineering
Winter Semester 2022-23

Boundary, Entity and Control objects

1. Boundary Objects

- **UI Screen formats :**
  The format in which screens and objects are displayed to the users.

- **Registration /Login interface:**
  This interface helps users to enter the information required for the registration and login purpose.

- **Profile updation interface:**
  This interface is used by students when they want to update their profile and resume.

- **Job Posting interface:**
  This interface is used by companies to post jobs.

- **Company display interface:**
  This interface is used to display the company list to the students and admin.

- **Job registering interface:**
  This interface is used by students to register for particular jobs.

- **Admin interface:**
  This interface gives admins their special functions and rights.It helps admins to do changes in the website when required.

- **Company Interface:**
  This interface is shown to companies for posting jobs and other company specific functions.

2. Entity Objects

- **Job:**
  It contains the data of all the job postings done by different companies.

- **Company:**
  It contains all the data about different companies that have registered for the placement drive.

- **Schedule:**
  It contains the data about the schedule for the placement process /Interviews for different companies.

- **Students:**
  It contains the data about the students that have registered for the current placement drive.

- **Admin:**
  It contains the data about the admins.

- **Datasheet:**
  It contains the data about the students that are placed and their placement related data.

- **Placement Manager:**
  It contains the data about the placement manager.

3. Control Objects

- **Profile authentication system**
  It is a controller object which will authenticate user profiles.This function is done manually by the admin or the placement manager.

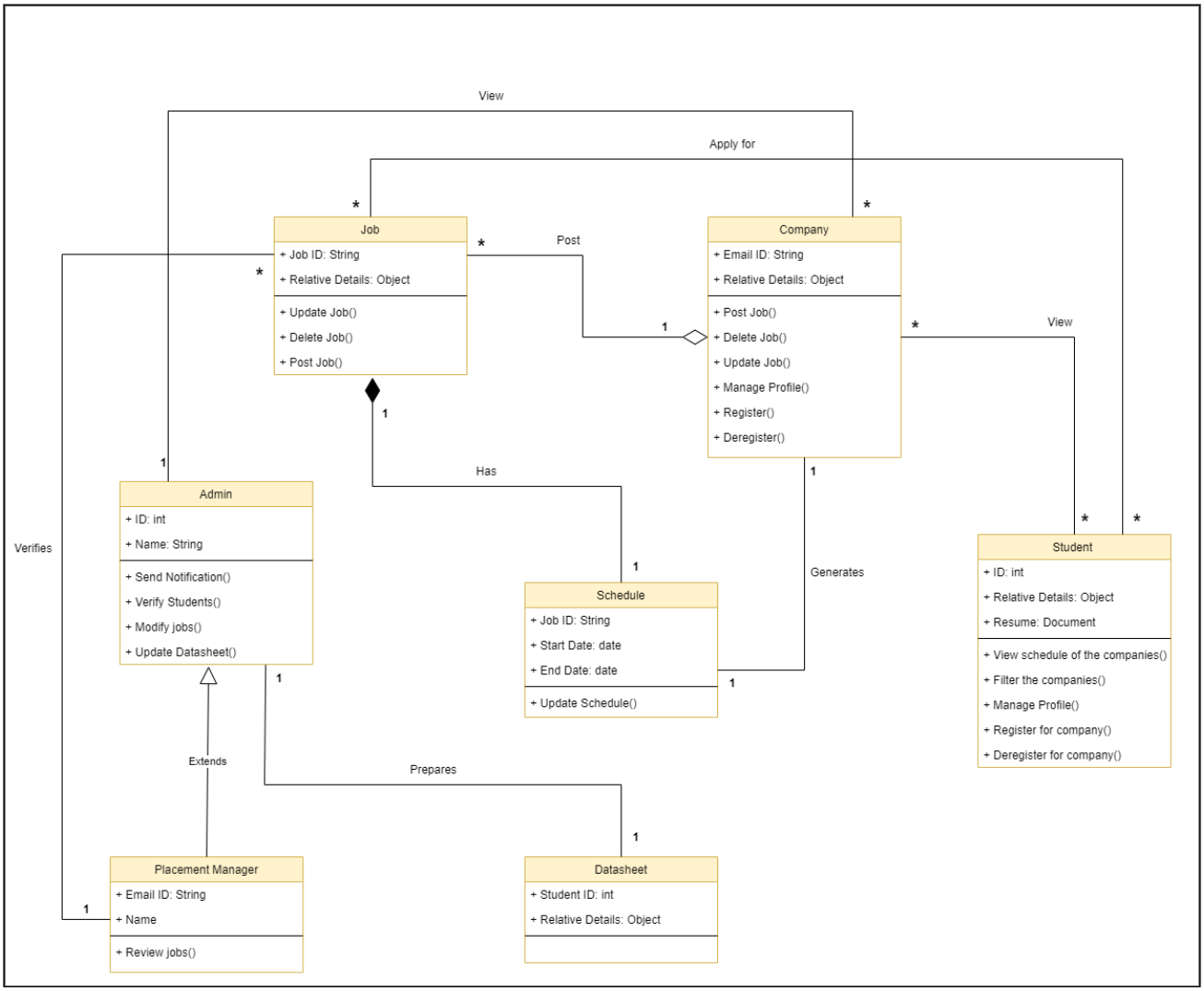
- **Profile data manager**
  This system will update and manage the data of companies and students according to the changes requested .

- **Job Register system**
  This controller object will allow students to register and deregister
  for jobs under the given time constraint. It will not allow any registrations once the time limit is exceeded.

- **Company filtering system**
  This controller object will help students to filter companies according to their qualifications.This also helps admin to filter companies as he wants.

- **Notification system**
  This controller object will send the Notifications to the students during the placement drive.

Class Diagram



Student relative details contain student name, gender, Date of birth, Mobile No, 10th percentage, 12th percentage, cpi, Active backlog, Total backlog, Branch.

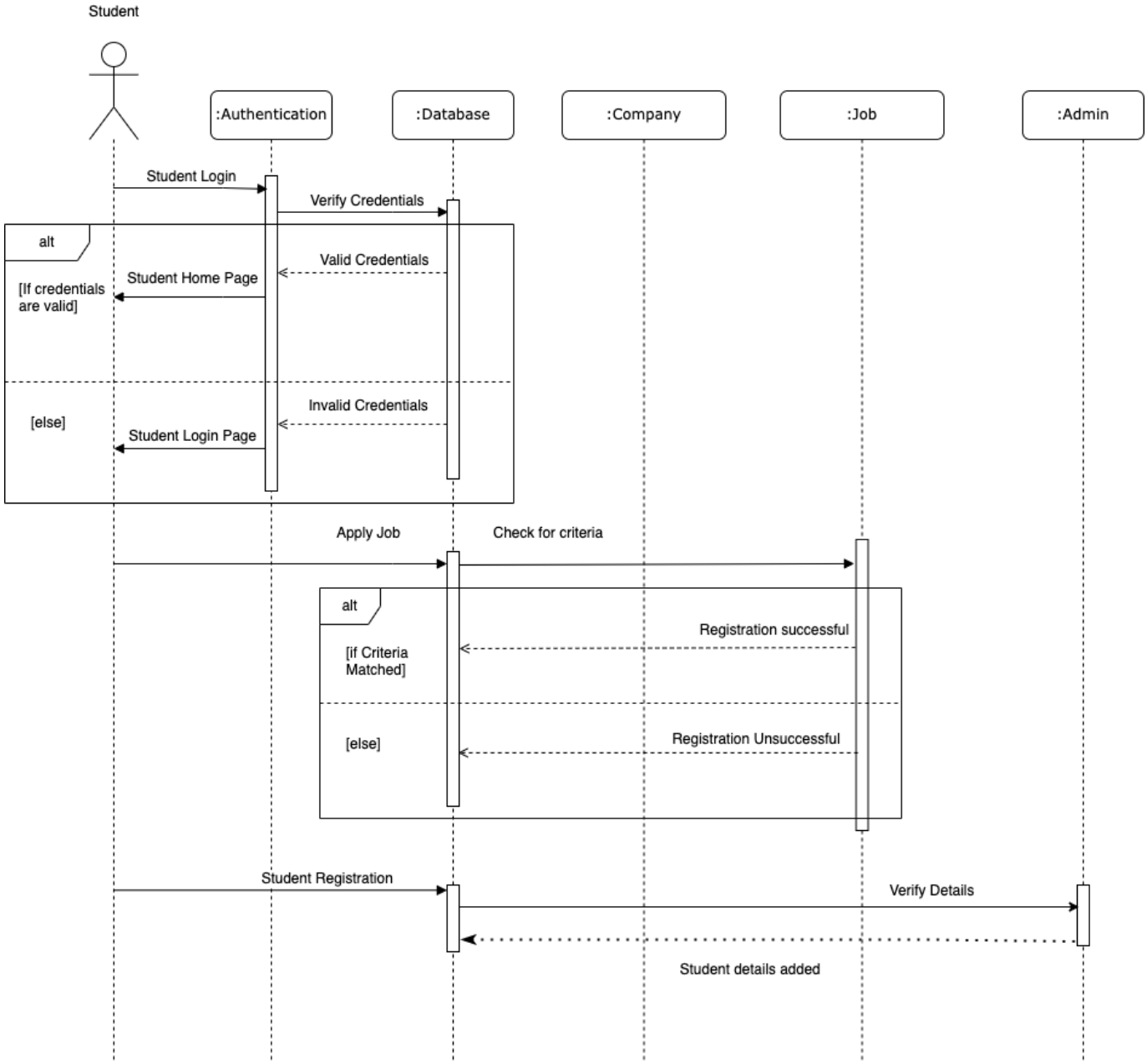
Company relative details contain password, company name, HR Name, Contact number, company brochure.

Job relative details contain posting location,ctc,UG Criteria, CPI cutoff, Job description.

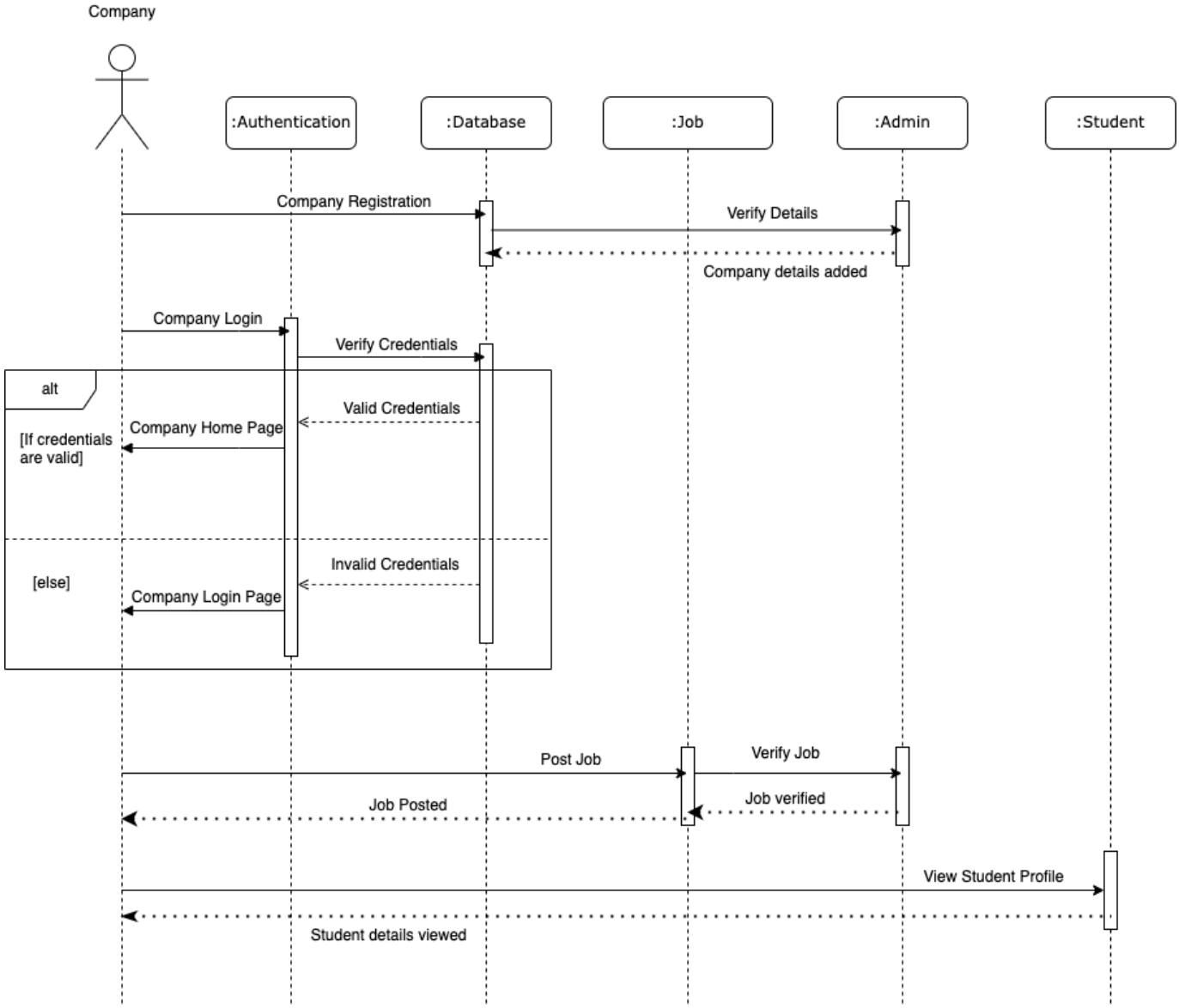
Datasheet relative details contain gender, company name, package, Location, Role.

Sequence Diagram

1. Student



2. Company


Company
:Authentication
:Database
:Job
:Admin
:Student
Company Registration
Verify Details
Company details added
Company Login
Verify Credentials
alt
Valid Credentials
Company Home Page
[If credentials are valid]
[else]
Invalid Credentials
Company Login Page
Post Job
Verify Job
Job Posted
Job verified
View Student Profile
Student details viewed

Design Goals

- **Reliability :** Reliability is the important expectation for student , placement managers and for other points of view. As low reliability leads to higher risk and error , which will lead to erroneous information about students and companies .

- **Maintainability**:The system should be easy to maintain and manage as it should provide services all around the clock to ensure that the web will be easy to use at any time.

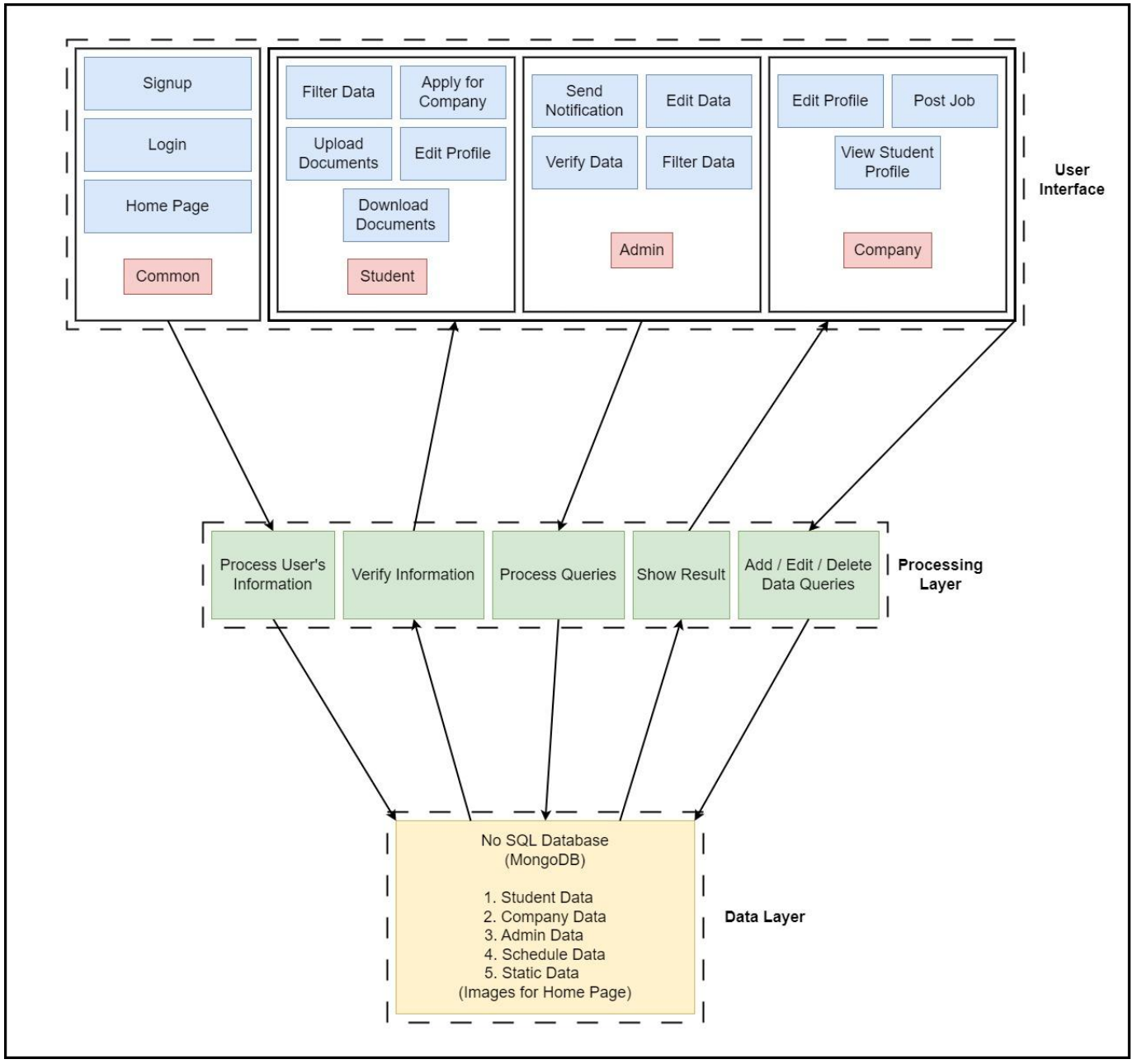
- **Efficiency :** There should not be any delays in providing services to users as it might lead to poor user experience and users may dislike the system.

- **Flexibility:** if any specification change comes in future we can easily adapt to reasonable changes.

- **User-friendliness :** Most students use the system so the system should be user friendly and students can find it difficult if the application is not user friendly.

- **Good Documentation :** The system with good documentation is easy to read and understandable.

High Level System Design



Architecture:

N-tier architecture (Layered Architecture)

Our architecture consists of 3 layers which are Presentation layer, Logic layer and Data layer.

Presentation layer:

The upper layer of the architecture consists of a user interface. The main function of the interface is to translate tasks and results to something the user can understand with the front-end system. The only layer that is exposed to the users.

Logic layer:

The middle layer of the architecture coordinates the application, processes commands, makes logical decisions and evaluation, and performs calculations with the backend system.

Data layer:

The lower layer of the architecture where information is stored and retrieved from the No-SQL database. The information is then passed back to the Logic layer for processing and then eventually back to the user.