CAREER INFORMATION AND RECRUITMENT PORTAL

PROJECT REPORT

submitted by

**JINCY P JANARDHANAN** (IEAREIT017)

**ALEENA SUNNY** (IEAREIT006)

**ALKA BHAGAVALDAS K** (IEAREIT007)

**AMEENA SHIRIN** (IEAREIT009)

to

the University of Calicut

in partial fulfilment of the requirements for the award of the Degree

of

**Bachelor of Technology**

in

Information Technology



Department of Information Technology

Institute of Engineering and Technology, University of Calicut, Thenjipalam

Kerala

July 18, 2020

DECLARATION

We undersigned hereby declare that the project report **Career Information and Recruitment Portal**, submitted for partial fulfilment of the requirements for the award of degree of Bachelor of Technology of the University of Calicut, Kerala is a bonafide work done by us under supervision of **Ms. Sruthimol M P**. This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources. We also declare that we have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

**Place:** Thenjipalam

**Date:** July 18, 2020

Jincy P Janardhanan

Aleena Sunny

Alka Bhagavaldas

Ameena Shirin

DEPARTMENT OF INFORMATION TECHNOLOGY  
INSTITUTE OF ENGINEERING AND TECHNOLOGY UNIVERSITY OF CALICUT, THENJIPALAM



CERTIFICATE

This is to certify that the report entitled **"CAREER INFORMATION AND RECRUITMENT PORTAL",** submittedby **Jincy P Janardhanan, Aleena Sunny, Alka Bhagavaldas, Ameena Shirin**to the **UNIVERSITY OF CALICUT** in partial fulfilment of the requirements for the award ofthe degree of Bachelor of Technology in Information Technology is a bonafiderecord of the project presented by them under our guidance and supervision. This report inany formhas not been submitted to any other University or Institute for any purpose.

|  |  |
| --- | --- |
| **Ms. Anu Manohar**  Assistant Professor  Department of IT  (Coordinator) | **Ms. Sruthimol M P**  Lecturer  Department of IT  (Coordinator & Guide) |

ACKNOWLEDGEMENT

We would like to express our warmest gratitude to all the people who had been a great help, support andmotivation throughout this project. Prima facie, we would like to thank the Almighty for givingus the wisdom and grace for the successful completion ofour project. Forever we owe our deepest gratitude to our guide andcoordinator, **Ms. Sruthimol M P**, Lecturer, Departmentof Information Technology, for her expert guidance, co-operation and immenseencouragement in pursuing this project. With a profound sense of gratitude, we would like to express our heartfelt thanks to our coordinator, **Ms. Anu Manohar**,Assistant Professor, Department of Information Technology for her earnest co-operation, support and constant inspiration.We extend oursincere gratitude to all the teachers of the Departmentof IT and to all our friends for their help and support.

Jincy P Janardhanan

Aleena Sunny

Alka Bhagavaldas

Ameena Shirin

ABSTRACT

This scheme aims to develop a web application to connect colleges, recruiters, students, and alumni on a single platform. It helps colleges for efficient management of their placement cell. Recruiters can use this application for their HR hiring activities related to campus recruitment. Students and alumni can further stay informed about various career choices available for them using the information portal. Finding jobs after graduation that best suits their interests and skill set is quite a challenging task for students or alumni. The difficulties in job finding arise from not having proper knowledge of the organization’s objective, their work culture, and current job openings. Discerning the best candidate amongst the list of candidates is a primary responsibility for the recruiters. The web application provides an easy and convenient search feature for students or alumni to find their desired jobs and for recruiters to find the right candidate. Students and alumni can conveniently use this web portal for job seeking. Colleges verify student and alumni information before registering them on the platform. A server admin verifies college and recruiter registration on the platform. Only alumni can update their details on their own, whereas students cannot. The recommendation system available in the web application is very advantageous for students and alumni. The web application automatically generates a CV for students and alumni using their details and information, which can be attached to their job applications. A web application to find and apply for a job is easy to use as a substitute for a manual and paper-based method.

Contents

[List of Figures viii](#_Toc46003693)

[List of Tables ix](#_Toc46003694)

[Abbreviations x](#_Toc46003695)

[Chapter 1 Introduction 1](#_Toc46003696)

[1.1 Problem Statement 1](#_Toc46003697)

[1.2 Motivation and Objective 1](#_Toc46003698)

[Chapter 2 Literature Review 4](#_Toc46003699)

[2.1 Existing Methodologies 4](#_Toc46003700)

[Chapter 3 Proposed System and Feasibility Study 7](#_Toc46003701)

[3.1 Proposed Solution 7](#_Toc46003702)

[3.2 Technical Feasibility 7](#_Toc46003703)

[3.3 Operational Feasibility 7](#_Toc46003704)

[3.4 Economic Feasibility 7](#_Toc46003705)

[3.5 Schedule Feasibility 8](#_Toc46003706)

[Chapter 4 Requirements Gathering and Analysis 10](#_Toc46003707)

[4.1 End User Specification 10](#_Toc46003708)

[4.2 Software Specification 10](#_Toc46003709)

[4.3 Hardware Specification 10](#_Toc46003710)

[4.4 SRS Document 10](#_Toc46003711)

[Chapter 5 System Design 13](#_Toc46003712)

[5.1 System Architecture 13](#_Toc46003713)

[5.2 Interface Design 13](#_Toc46003714)

[5.3 Data Flow Diagrams 13](#_Toc46003715)

[5.4 Data Dictionary 13](#_Toc46003716)

[Chapter 6 Implementation 16](#_Toc46003717)

[6.1 Module Description 16](#_Toc46003718)

List of Figures

List of Tables

Abbreviations

# Introduction

When the first digital computers appeared in the early 1940s,[5] the instructions to make them operate were wired into the machine. Practitioners quickly realized that this design was not flexible and came up with the "stored program architecture" or von Neumann architecture. Thus the division between "hardware" and "software" began with abstraction being used to deal with the complexity of computing.

Programming languages started to appear in the early 1950s[6] and this was also another major step in abstraction. Major languages such as Fortran, ALGOL, PL/I, and COBOL were released in the late 1950 and 1960s to deal with scientific, algorithmic, and business problems respectively. David Parnas introduced the key concept of modularity and information hiding in 1972[7] to help programmers deal with the ever-increasing complexity of software systems.

## Problem Statement

## Motivation and Objective

# Literature Review

## Existing Methodologies

After completion of a graduate or undergraduate course students have to apply for a job. Colleges will be also looking forward to recruiters who provide good opportunities for their students. Moreover, recruiters will be looking for students with good skills. But all these works are done using manpower. Students have to apply for a job and keep track of the recruiter updates.

In the existing system, all data processing is done manually. There are a lot of issues such as retrieval and storage of the information and keeping track of them becomes a tedious task. By implementing a computerized system, the limitations in the present system will be reduced. Manpower can be reduced to a great extent and efficiency and accuracy can be increased. Moreover, consumption of time can be reduced to a far great extend by the implementation of the proposed system.

# Proposed System and Feasibility Study

## Proposed Solution

We propose a web application career information and recruitment portal which consists of college, student, recruiter and alumni. It connects students/alumni, recruiters and college. It provides a great opportunity for students to find a job which is suitable for them. The college admin adds all the student details. Students can personalize these details. A CV is generated automatically using this information.

When students, alumni apply for a job the recruiters receives applications in sorted order according to their score calculated based on work experiences, educational qualifications, skills and recommendations. It’s also easy for applicants and recruiters to communicate with each other using a chat feature. The proposed system provides fast operation and low-cost expense than the old system. It also provides information about local and international higher studies and job opportunities.

## Technical Feasibility

The technical feasibility assessment focuses on technical resources available to the developers. The assessment helps us to determine whether the technical resources meet the capacity to convert the ideas into a working system. Technical feasibility involves the evaluation of hardware, software, and other technical requirements of the proposed system.[1]

The end-user requirements, software and hardware requirements listed in the requirements analysis easily available and configurable by the user or developer. Hence it is technically feasible.

## Operational Feasibility

Operational feasibility studies how a project satisfies the requirements identified in the requirement analysis phase. It involves undertaking a study to analyze and determine whether-and how-well the requirements are met by completing the project.[1]

All the functional and non-functional requirements are listed out in section 4.4. These functional and non-functional requirements can be easily implemented using the technologies available. Hence it is operationally feasible

## Economic Feasibility

Economic feasibility assessment involves a cost/benefits associated with a project, helping us to determine the viability, cost, and benefits associated with a project before financial resources are allocated. [1]

The cost-benefit analysis of the system is carried out accordingly. The system can be built and used using basic hardware requirements.Moreover, it is easy to maintain the system. Hence it is economically feasible

## Schedule Feasibility

Schedule feasibility is the most important for project success; after all, a project will fail if not completed on time. In scheduling feasibility, we estimate how much time the project will take to complete.[1]

Our project is developed using an iterative and incremental model which helps in the successful completion of the project. Moreover, the technologies used in developments is time-saving. We are using GitHub for project management so that we can manage the schedule. Hence it is schedule feasible.

# Requirements Gathering and Analysis

## End User Specification

## Software Specification

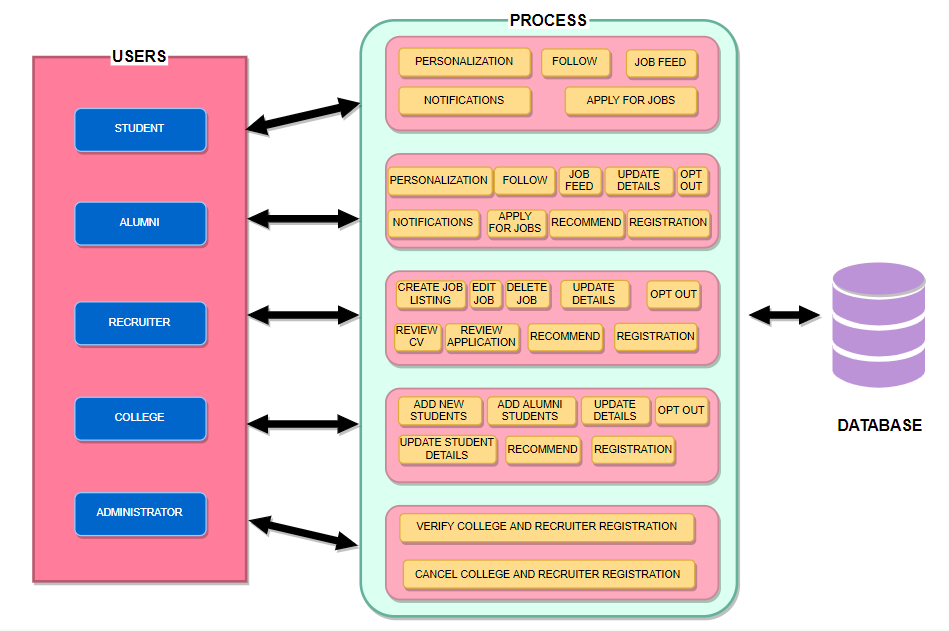
## Hardware Specification

## SRS Document

# System Design

## System Architecture

A System Architecture is the conceptual model that defines the structure, behavior and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structure and behavior of the system. A System Architecture can consist of system components and the sub systems developed that will work together to implement the overall system. This is our system architecture.



## Interface Design

Interface Design requires a good understanding of user needs. It mainly focuses on the requirements of the platform and its user expectations. There are several phases and processes in the user interface design, some of which are more demanded upon than others, depending on the project.

|  |  |
| --- | --- |
| **INTERFACE NAME** | **DESCRIPTIONS** |
| User Registration | UI for Colleges, Recruiters and Alumni can Register. |
| Update Details | UI for College and recruiters can update their details when needed. |
| Add New Student | UI for Colleges can add new students to the portal. |
| User Login | UI for Colleges, Recruiters, and Students can login and go to their dashboard. |
| Create New Job | UI for Recruiters can create a new job listing. |
| View Created Job Listing | UI for Students can view created job listing on college dashboard. |
| Edit job listing | UI for Recruiters can edit job listing when needed. |
| Delete job listing | UI for Recruiters can delete job listing when needed. |
| Update job listing | UI for Recruiters can update job details. |
| Search for job | UI for Student and Alumni can search, open and view jobs. |
| Job Listing Page | UI for Students can apply for the job from their job listing page. |
| Review Job Applications | UI for Recruiters can review received job applications from their job listing page. |
| Forgot Password | UI for Users can request for forgot password. |
| Opt Out Request | UI for Colleges and Recruiters can request for opting out from this portal. |
| Update Student Details | UI for Colleges can update student details. |
| Add Personalization Details | UI for Students can personalize their profile by adding skills, experiences and etc. |
| Follow Topics and Companies | UI for Students can follow interested topics and companies, and can set notification preference for followings. |
| Notification Panel | UI for Students can get notification when a topic/company they follows has a new job posting and can view the job listing. |
| Job feed | UI for Students can get job feed based on the topics and companies they follows. |
| Modify Notifications | UI for Colleges receive alumni registration request in their notification panel. |
| Add Auto Generated CV | UI for Generates automatic CV, if not exists, from student details and personalized details. |
| Request Recommendations | UI for Allow Students and Alumni to request recommendation from privileged user. |
| Recommend | UI for Allow Privileged users to recommend student/fellow Alumni. |
| Chat Box | UI for Recruiters and job applicants can communicate through messages. |

## Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional way to visualize information flows within the system. The purpose of DFD is to show the scope and boundaries of a system as a whole. It can be manual, automated, or a combination of both. The DFD is also called as a data flow graph or bubble chart. This constructed using a set of symbols.

External Entity

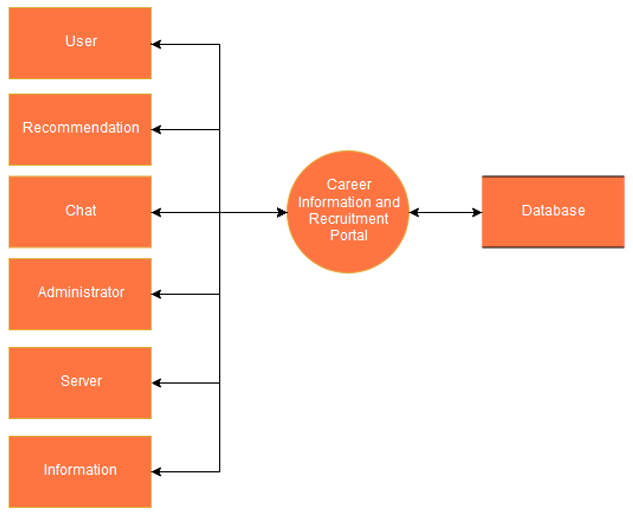
Process

Data Flow

Database

Data Storage

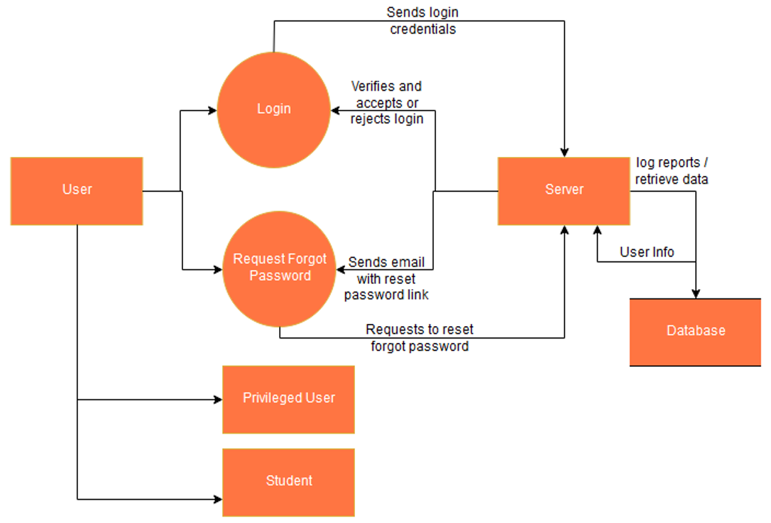
**LEVEL 0 :**



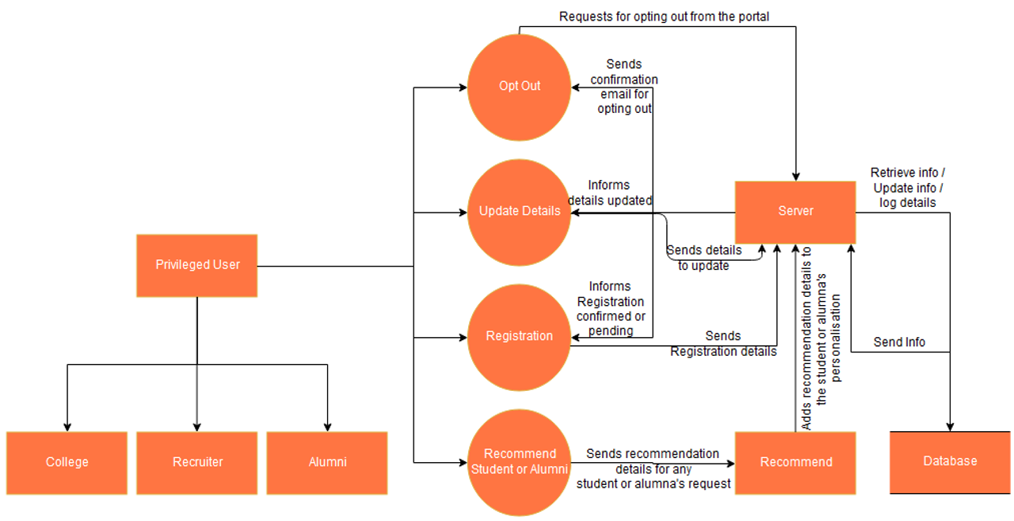
The above diagram is the context diagram for the system, it includes all the modules in the system.

**LEVEL 1 DIAGRAMS** :

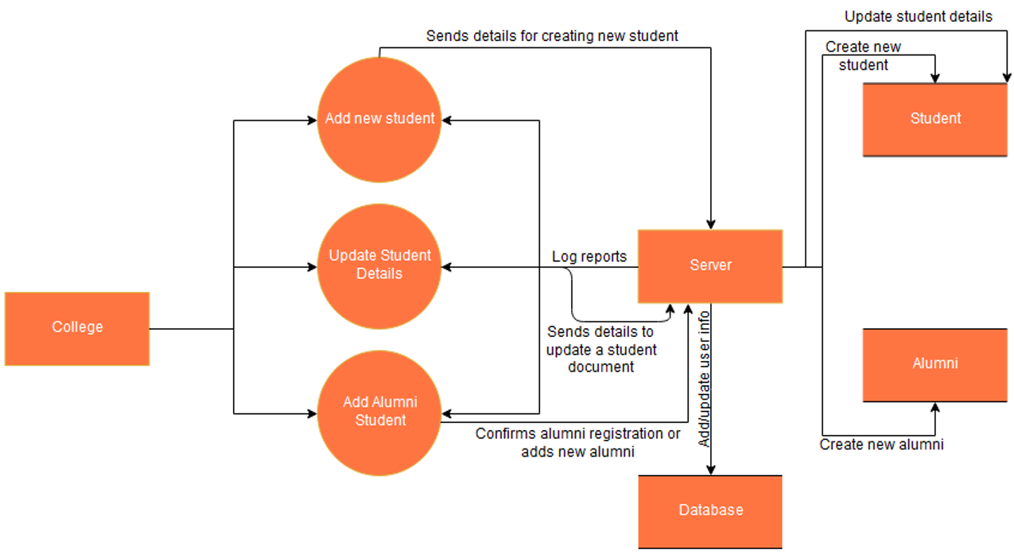
LEVEL 1.1 - User Module



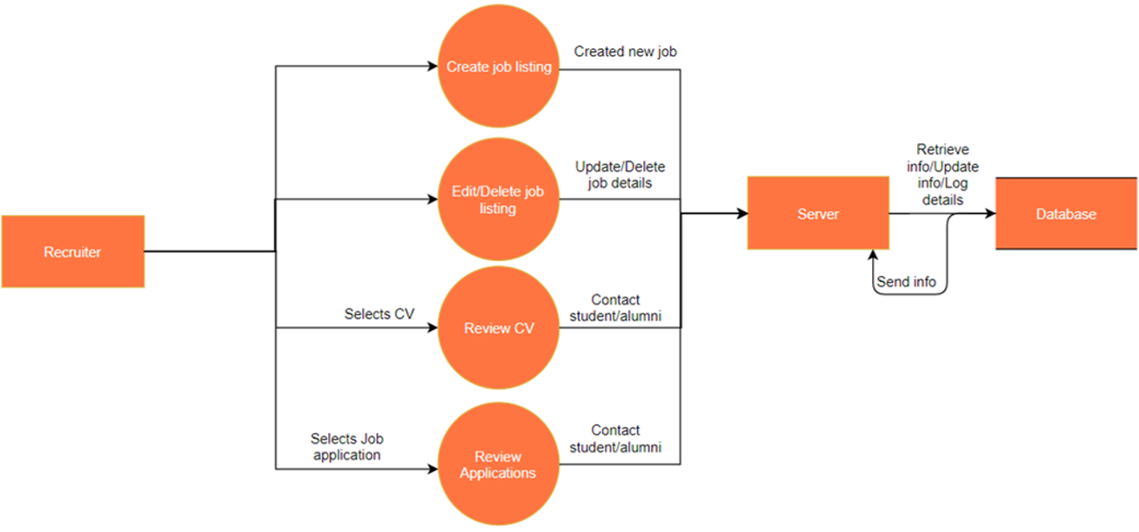
LEVEL 1.1.1 - Privileged User



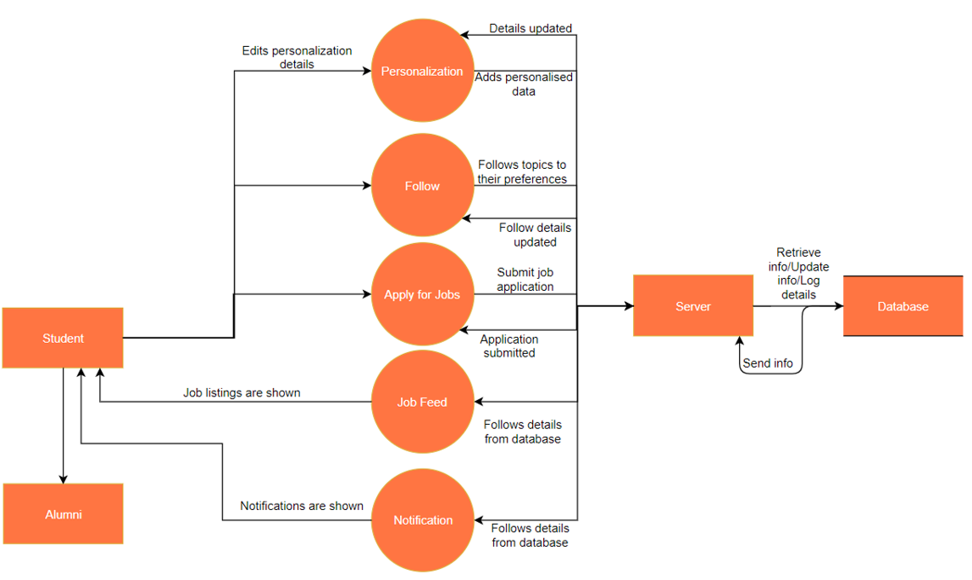
LEVEL 1.1.1.1 - College Module



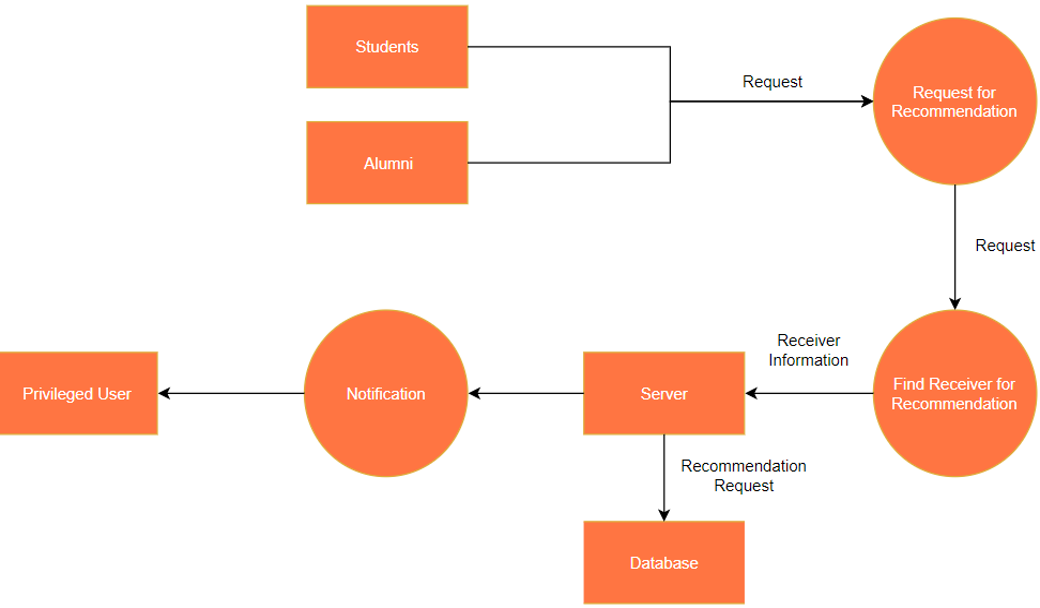
LEVEL 1.1.1.2 - Recruiter Module



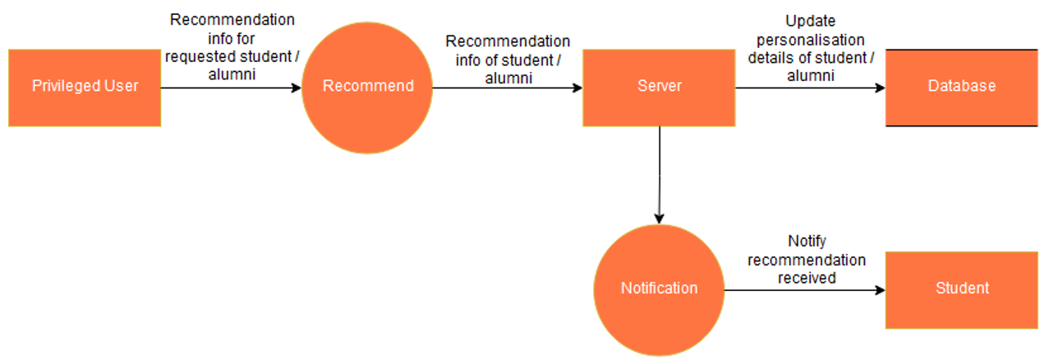
LEVEL 1.1.2 - Student Module



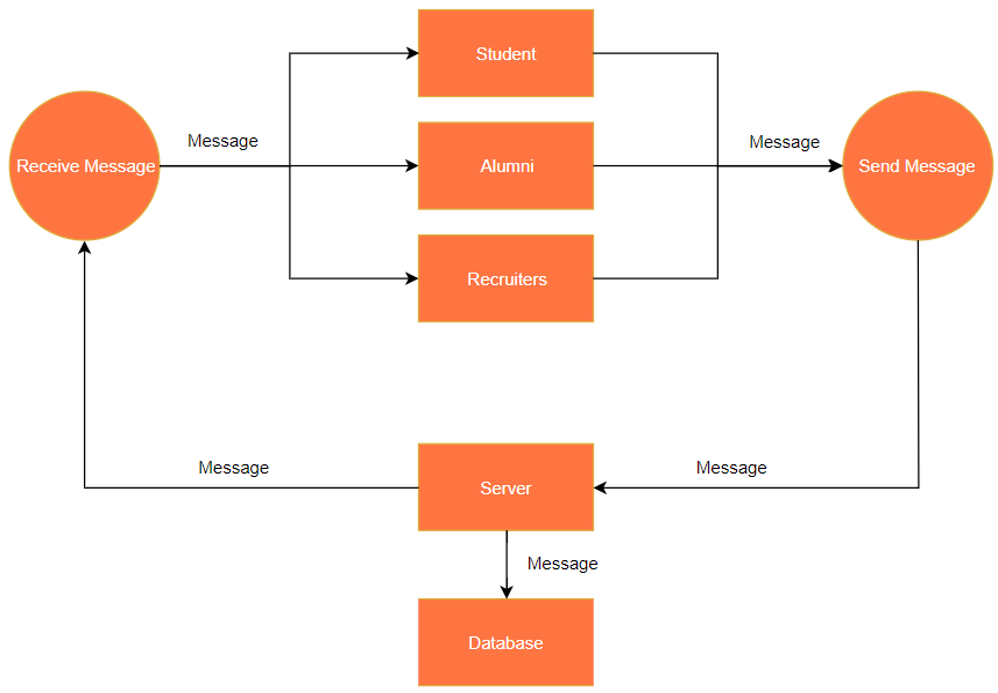
LEVEL 1.2.1 - Request Recommendation Module



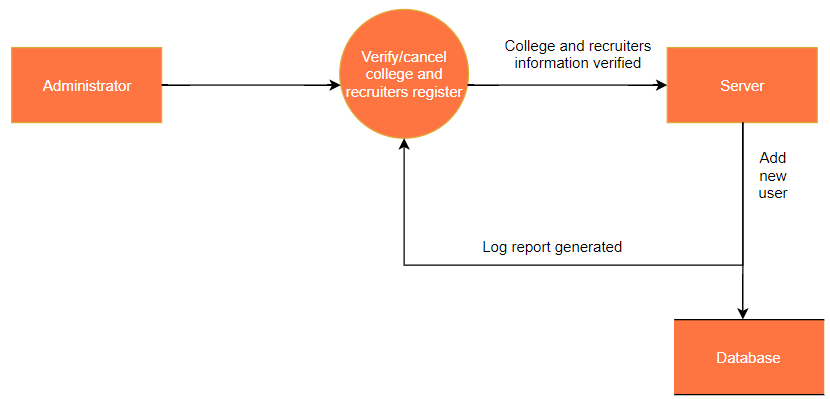
LEVEL 1.2.2 - Recommend Module



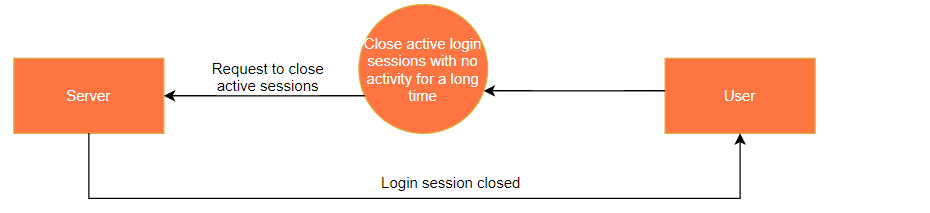
LEVEL 1.3 - Chat Module



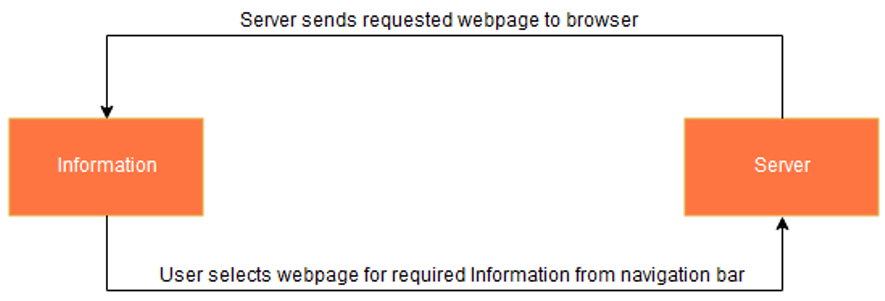
LEVEL 1.4 - Administrator Module



LEVEL 1.5 - Server Module

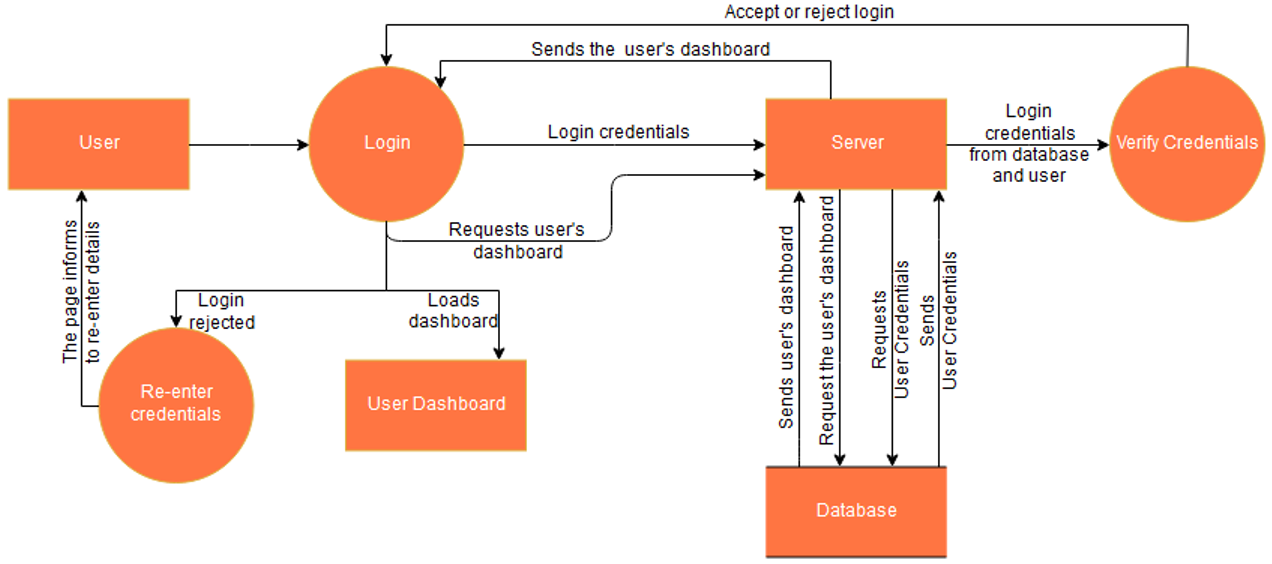


LEVEL 1.6 - Information Module

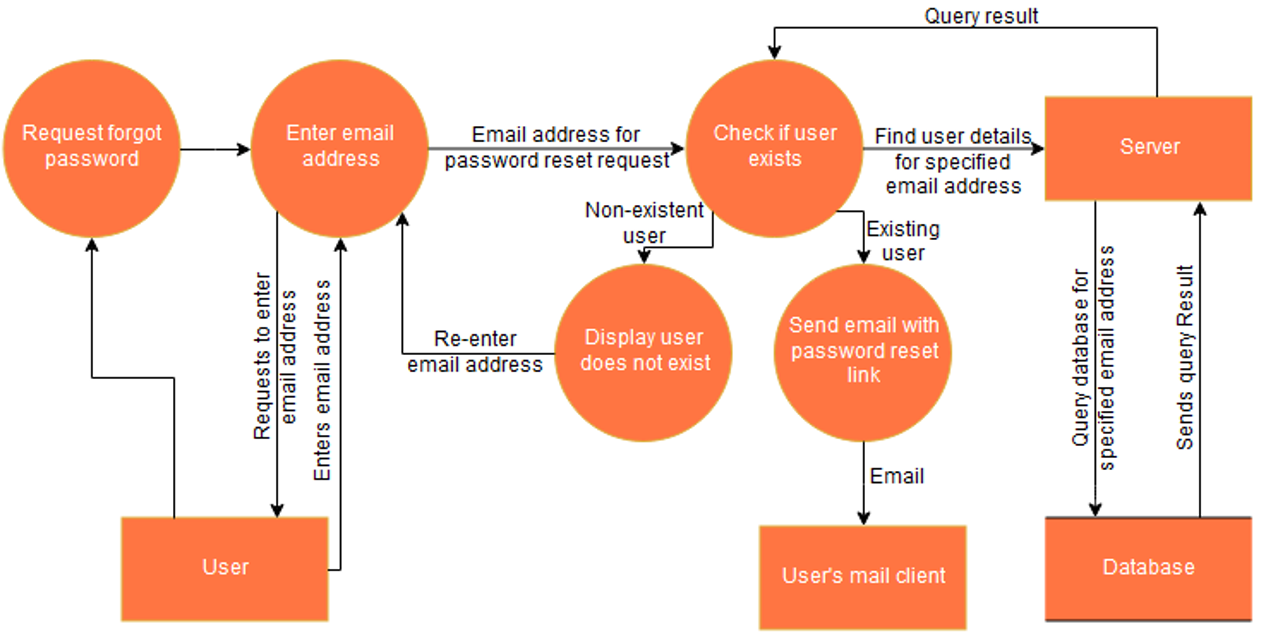


**LEVEL 2 DIAGRAMS:**

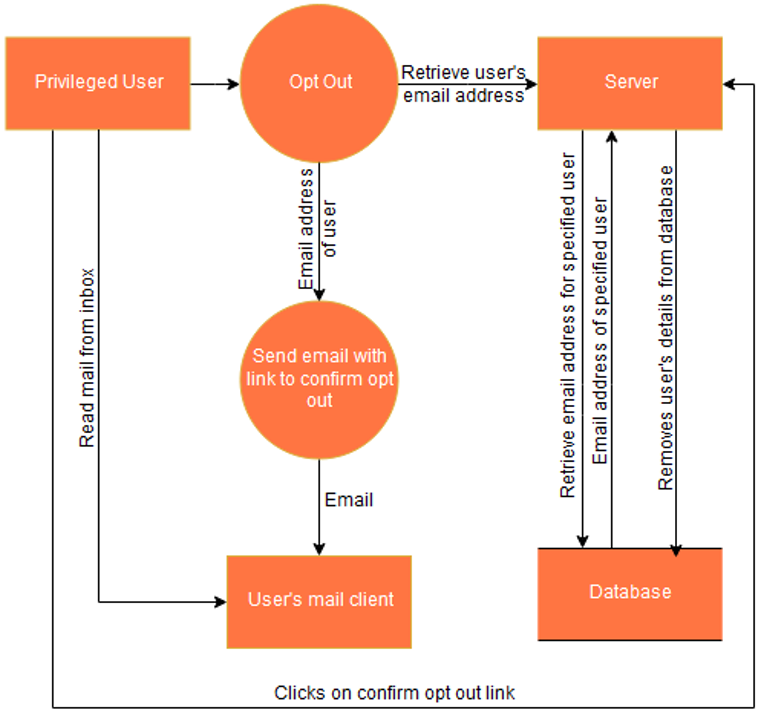
LEVEL 2.1 - User Login



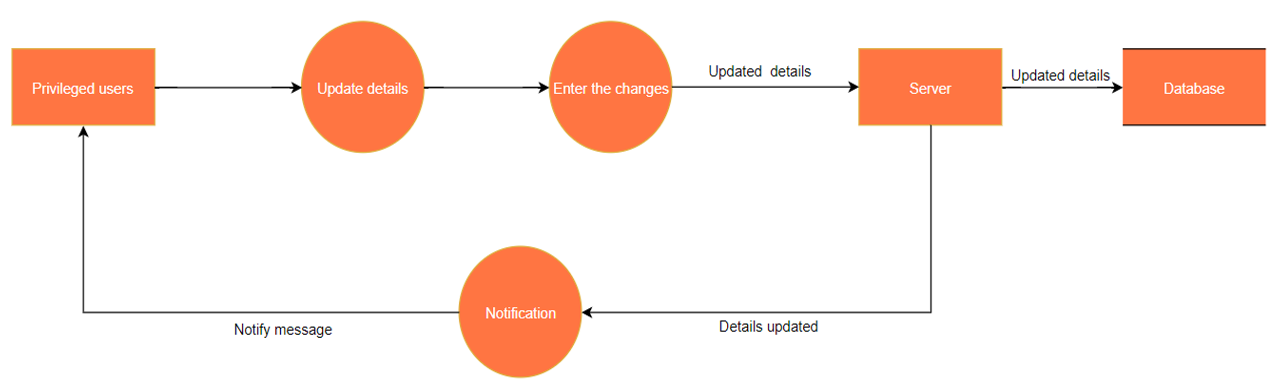
LEVEL 2.2 - Request For Forgot Password



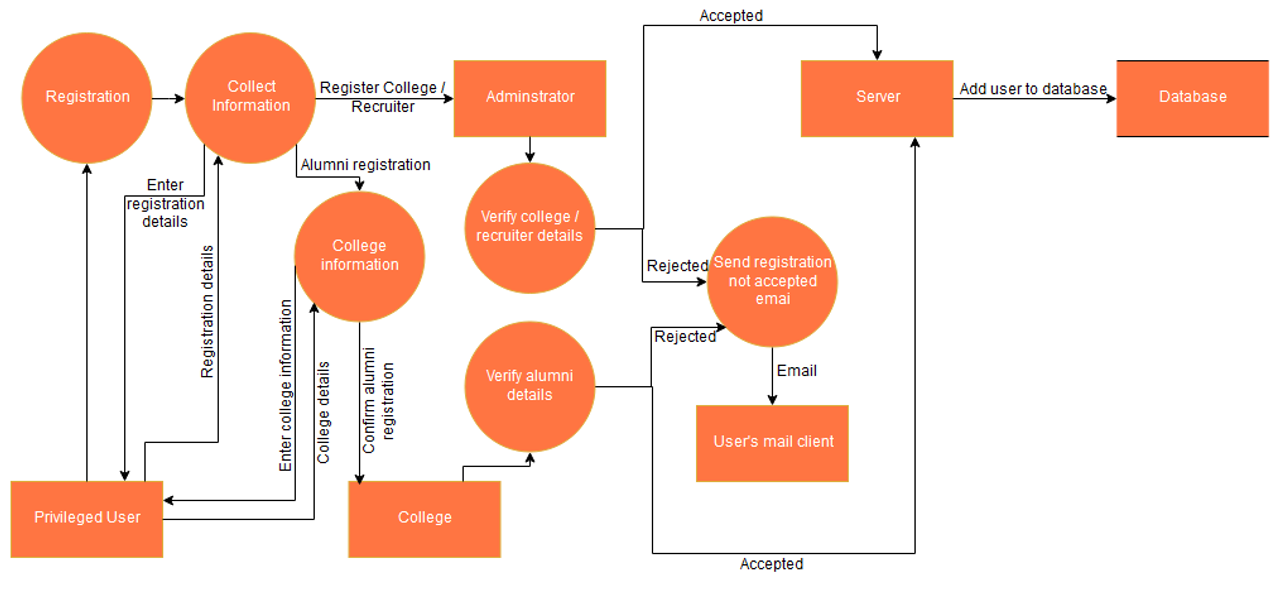
LEVEL 2.3 - Opt out Request by Privileged User



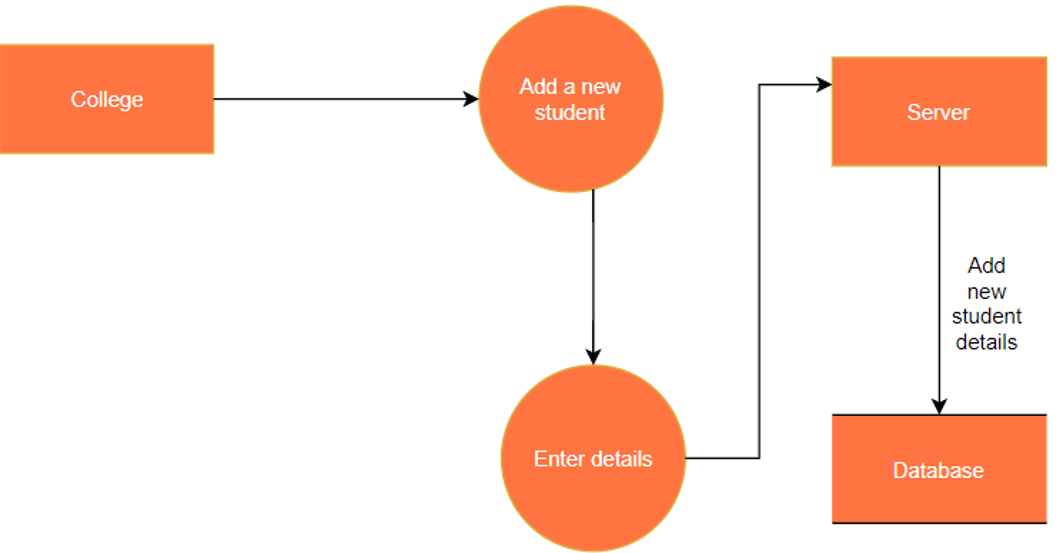
LEVEL 2.4 - Update Details Of Privileged User



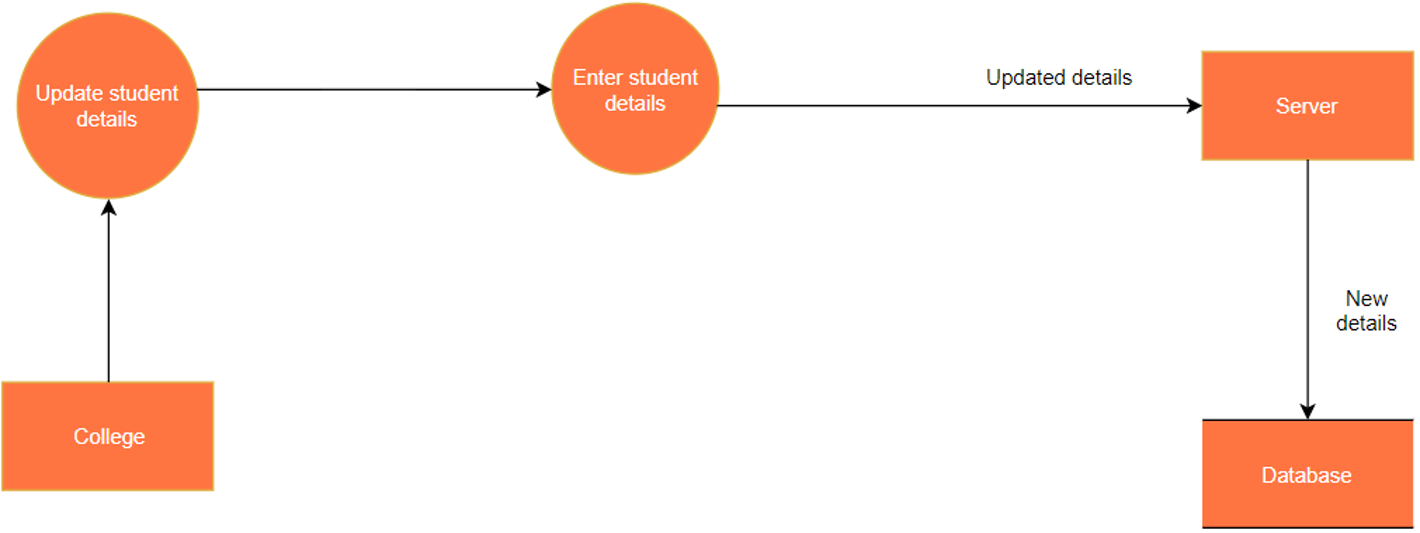
LEVEL 2.5 - Registration of Privileged User



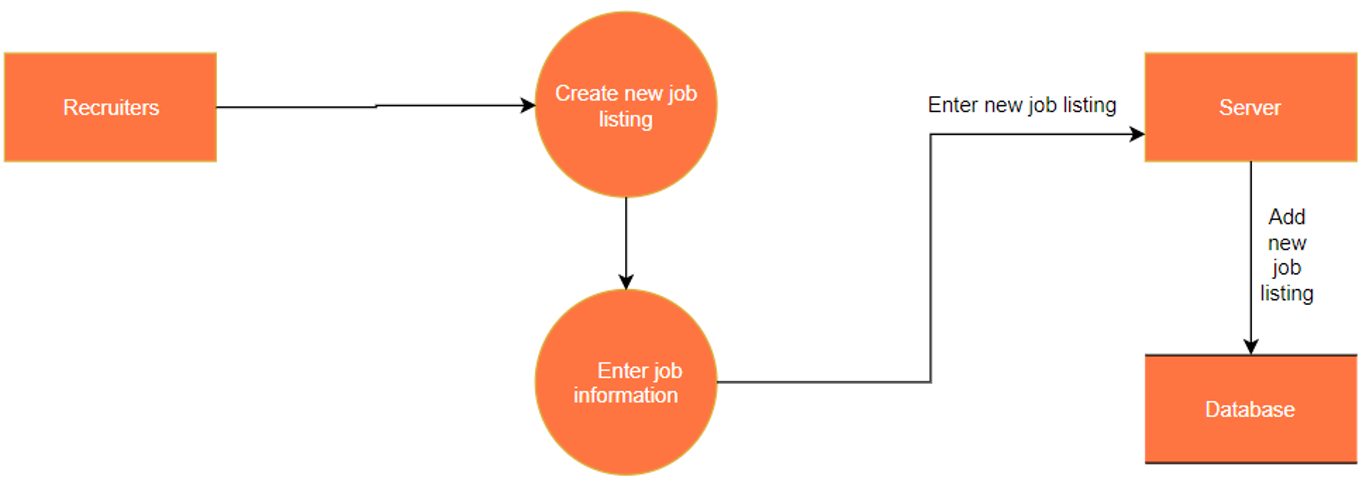
LEVEL 2.6 - Add New Student by College

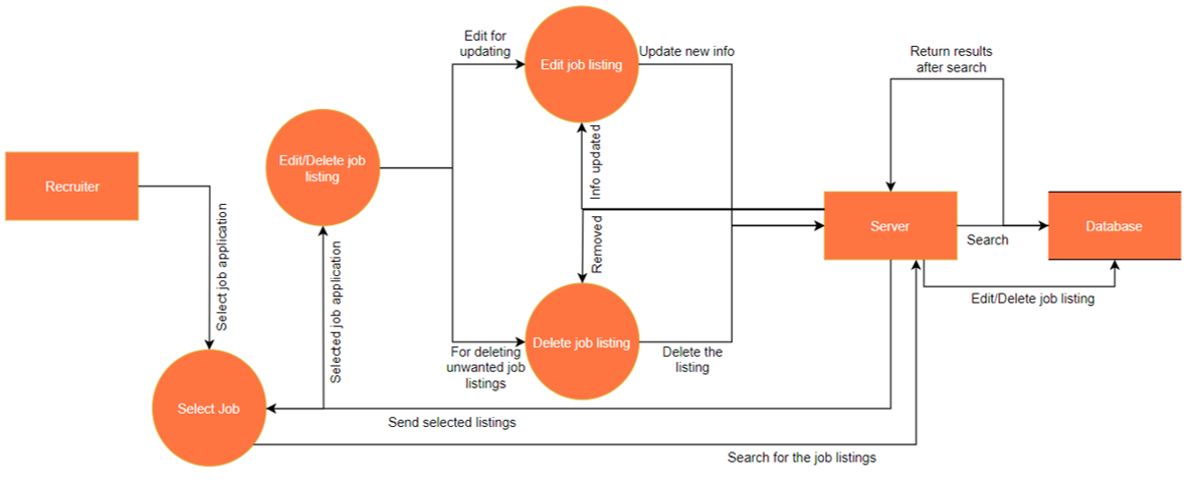


LEVEL 2.7 - Update Student Details

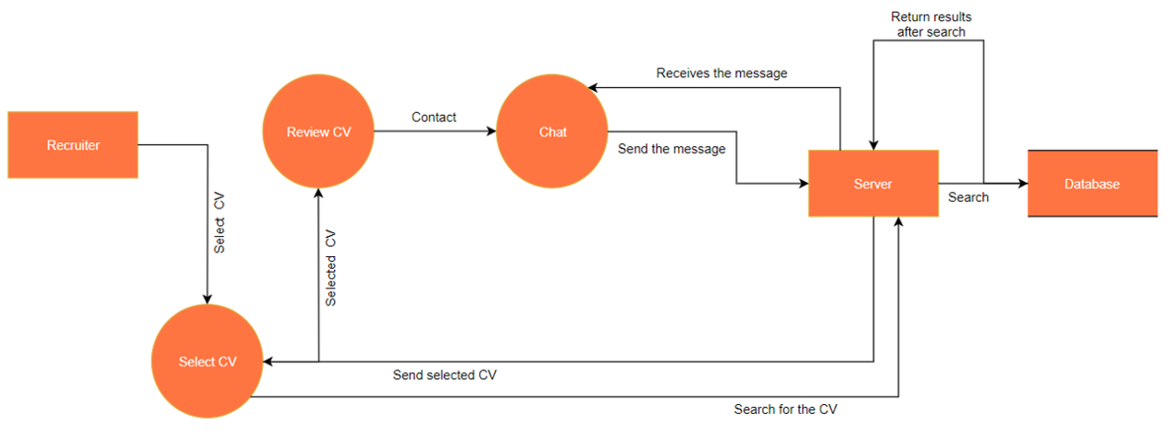


LEVEL 2.8 - Create New Job Listing by Recruiters

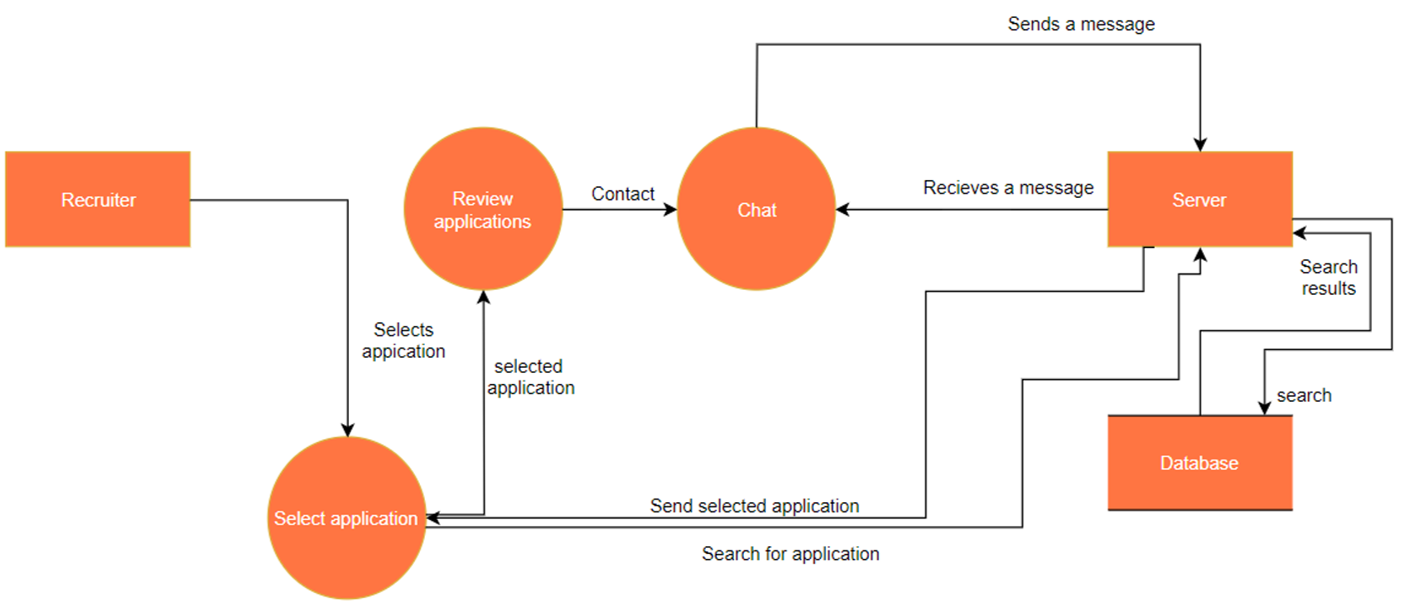
LEVEL 2.9 - Edit or Delete Job listing by Recruiters



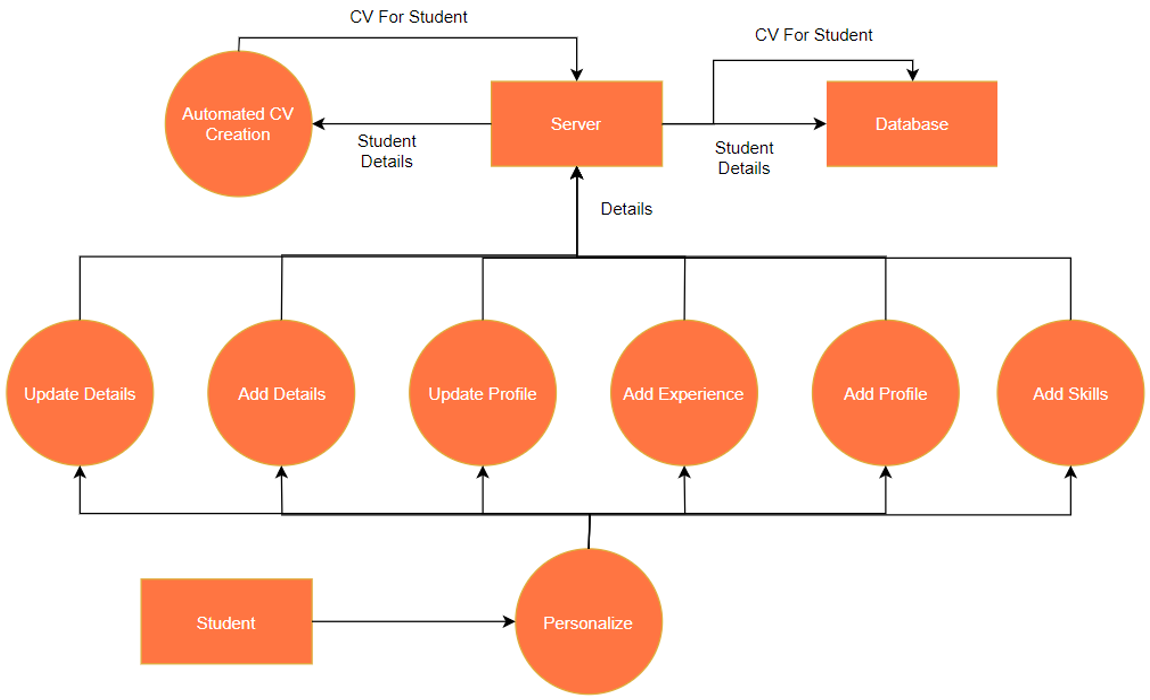
LEVEL 2.10 - Review CV by Recruiter



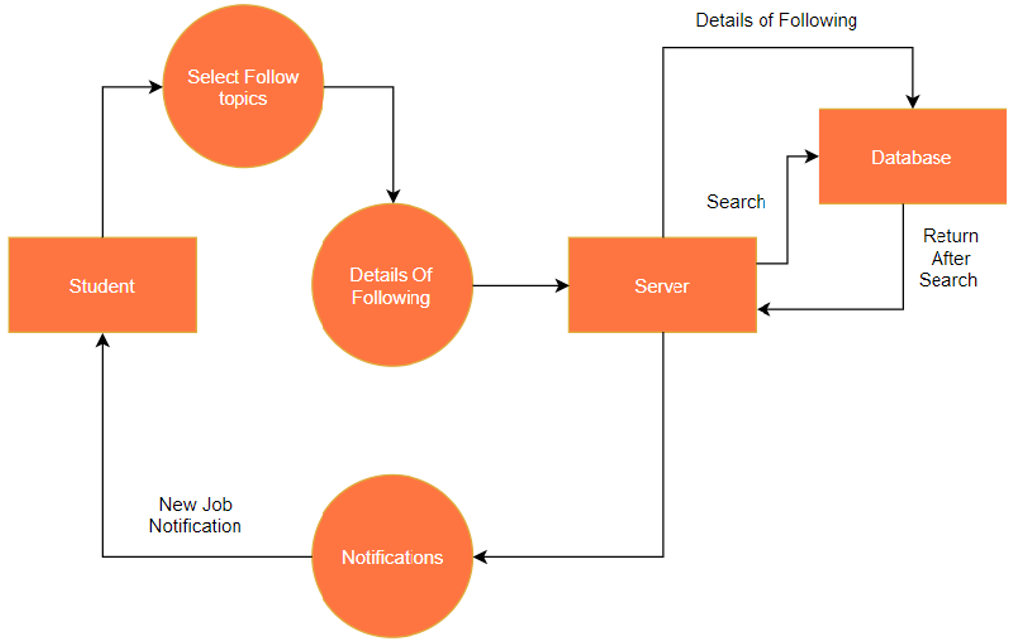
LEVEL 2.11 - Review Application by Recruiter



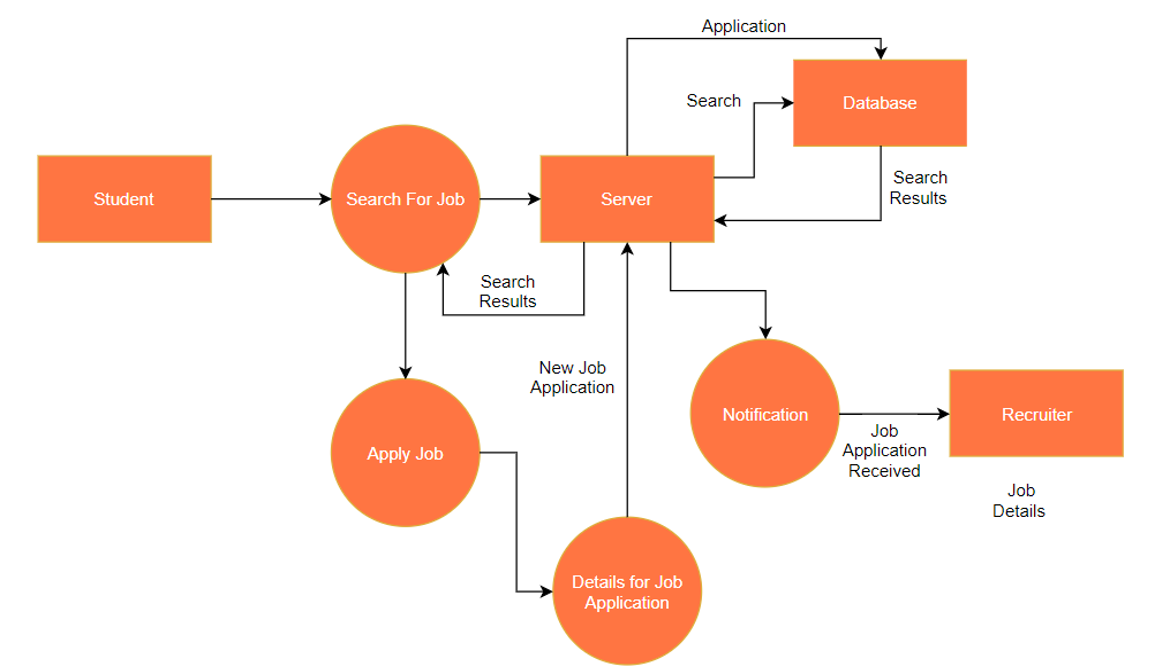
LEVEL 2.12 - Entering Personalisation Details by Student



LEVEL 2.13 - Following Topics or Recruiters by Student



LEVEL 2.14 - Apply For Job by Student



## Data Dictionary

A Data Dictionary is a collection of names , definitions, and attributes about data elements that are being used or captured in a database, information systems, or part of a research project . It describes the meanings and purpose of data elements within the context of a project, and provides guidance on interpretation, accepted meanings and representation. A Data Dictionary also provide metadata about data elements. The metadata included in a Data Dictionary can assist in defining the scope and characteristics of data elements, as well as the rules for their usage and application.

* LIST OF TABLES USED IN PROPOSED SYSTEM

|  |  |
| --- | --- |
| 1. ADDRESS COLLECTION | To store address details of user |
| 2. MAIL COLLECTION | To store mail id of user |
| 3. COLLEGE COLLECTION | To store details about colleges |
| 4. RECRUITER COLLECTION | To store details about recruiters |
| 5. ADMIN COLLECTION | To manage user requests |
| 6. USER COLLECTION | To store details of user |
| 7. STUDENT COLLECTION | To store student details and manage it |

### 5.4.1 ADDRESS COLLECTION

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| Address\_line1 | String | User address |
| Address\_line2 | String | User address |
| City\_or\_ town | String | Place of the user |
| District | String | District of the user |
| State | String | State of the user |
| Country | String | Country of the user |
| Pincode | Long | Pin code of user |

**5.4.2 MAIL COLLECTION**

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| Mail\_id | String | Mail\_id of the user |

**5.4.3 COLLEGE COLLECTION**

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| Affiliated\_univ | String | College information |
| College\_landph | Long | College contact number |
| College\_public\_email | String | College contact email |
| Students | ArrayList | Student details |
| Alumni | ArrayList | Alumni details |

**5.4.4 RECRUITER COLLECTION**

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| Recruiter\_licence\_no | String | Approved licence number of recruiter |
| Recruiter\_landph | Long | Recruiter contact number |
| Recruiter\_public\_email | String | Recruiter contact email |

**5.4.5 ADMIN COLLECTION**

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| College\_request\_pending | ArrayList | Delay College request |
| College\_approved | ArrayList | Approved college details |
| College\_denied | ArrayList | Deny college access |
| Recruiter\_request\_pending | ArrayList | Delay Recruiter request |
| Recruiter\_approved | ArrayList | Approved recruiter details |
| Recruiter\_denied | ArrayList | Deny recruiter access |

**5.4.6 USER COLLECTION**

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| \_id | String | Unique id to identify user |
| Username | String | Confirms username |
| Password | String | Confirm the identity |
| Name | String | Name of user |
| Address | String | Address of the user |
| Mobile | String | Required phone number |
| Email | String | User mail id |

**5.4.7 STUDENT COLLECTION**

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| Registerno | String | Unique student identification number |
| Course | String | The course student attending |
| Branch | String | Branch student been attending |
| Semester | String | Semester student been attending |
| Starting\_year | Numeric | Course started year |
| Ending\_year | Numeric | Course ending year |
| SGPA | ArrayList | Total scores for exams |

# Implementation

## Module Description

* User: A Module that implements all common features of any general users. Functionalities provided by this module are as follows:
* Login
* Request forgot password

Submodules of user

* Privileged user: A module to implement the common features of College, Recruiter, and Alumni (Privileged Users). Functionalities provided by this module are as follows:
* opt out
* Update details
* Registration
* Recommend student/alumni
* College: A module that implements specific features for College, Privileged User Interface, Recommend interface from Recommendation Module. Functionalities provided by this module are as follows:
  + - * + Add a new student
        + Add alumni student
        + Update student details
* Student: A module that implements specific features for student and alumni, Request Recommendation interface from Recommendation module, and chat module. Functionalities provided by this module are as follows:
* Personalization
* Follow
* Job feed
* Notification
* Apply for jobs
* Alumni: A module that implements specific features for alumni and Privileged User Interface. NO ADDITIONAL FUNCTIONS REQUIRED.
* Recruiter: A module that implements specific functionalities for Recruiter, Privileged User Interface, Recommend interface, and Chat module. Functionalities provided by this module are as follows:
* Create job listings
* Edit/ delete job listing
* Review CV
* Review application
* Recommendation: A module for requesting and recommending student/alumni, which provides access only via interfaces. Functionalities provided by this module are as follows:
* Recommendation
* Request recommendation
* Recommend
* Chat: A module that allows recruiters and students/alumni to communicate with each other. Functionalities provided by this module are as follows:
* Send messages
* Receive messages
* Administrator: A module to implement functionalities for Server Admin. Functionalities provided by this module are as follows:
* Verify / Cancel College and Recruiter Registration
* Server: A module to implement functionalities for server-only functionalities (without human interaction). Functionalities provided by this module are as follows:
* Handle Opt-Out Requests of Colleges and Recruiters
* Close Active Login Sessions with No Activity for a Long Duration
* Handle Forgot Password Requests
* Information: A module that implements functional requirements of career information. It is used for serving web pages related to various career opportunities for a student or alumni.