

# Project Profile

---

## Project Details

Project Title	Modern Job Portal Platform
Project Category	Website
Front-End	PHP , HTML , CSS , BOOTSTRAP , JS
Back-End	MySql
Database	MySql

## Student Details

Seat No. / Exam No.	164
Name	Tirth Moradiya
Email	tirthmoradiya02@gmail.com

## Project Description

Description	<p>JobNexa is a comprehensive job portal platform designed to connect job seekers with employers efficiently.</p> <p>The platform offers intuitive user interfaces for both job seekers and employers, allowing them to manage their profiles, search for jobs, post job openings, and facilitate the application process seamlessly.</p>
-------------	---

# System Analysis and Modelling

---

## **SRS (Software Requirement Specification)**

### **❖ Functional Requirement :-**

- **Admin Dashboard:**

The admin dashboard provides an overview of key metrics and insights, including the number of job listings, user registrations, and recent activities. This Module is also use for Manage User and Jobseekers by Admin.

- **User Registration/Login:**

New Users can create an account by providing basic information such as name, email, and password. And Existing Users can Login directly using username and password.

- **Job Search:**

This function allows Users to search for job listings based on keywords, location, and job category.

- **Job Listing:**

This function allows Employers to post job vacancies by providing job title, description, location, and requirements. Job listings are visible to all registered users.

- **Application Submission:**

Job seekers able to apply for job listings by submitting their resume and cover letter.

Applications sent directly to the employer's designated contact.

- **Profile Management:**

Users can update their profile information, including contact details, education, and work experience. Employers can manage their company profile and job postings.

- **Job Alerts:**

Users can set up job alerts based on their preferences (e.g., location, job category).  
Users receive notifications when new job listings match their criteria.

- **Application Tracking:**

Job seekers can track the status of their job applications (e.g., pending, under review, rejected). Employers manage received applications and communicate with candidates.

- **User Authentication and Security:**

User accounts protected with secure authentication mechanisms (e.g., password hashing).  
Personal data stored securely and comply with data protection regulations

## ❖ **Non-Functional Requirement :-**

- **Performance:**

The system is responsive and able to handle a large number of concurrent users without significant performance degradation. Page load times for key functionalities (e.g., job search, application submission) within less than 3 seconds.

- **Security:**

User data (e.g., personal information, resumes) stored securely and protected against unauthorized access, modification, or disclosure.

- **Usability:**

The user interface is intuitive and easy to navigate, catering to users with varying levels of technical expertise.

- **Scalability:**

The system is also scalable to accommodate future growth and increasing user demand.

- **User Friendly Interface:**

The system is a user-friendly interface so it can be used by people who are not friendly with **computers**

# Software Design

---

## Database Design

### 1> User

User_id	Int	Primary Key
User_Name	Varchar	
User_Email	Varchar	
User_Password	Varchar	

User_Gender	Varchar	
User_Phonenumner	Int	
User_Address	Varchar	
User_State	Varchar	
User_City	Varchar	

## 2> Employer

Employer_id	Int	Primary Key
Employer_Name	Varchar	
Employer_Email	Varchar	
Employer_Password	Varchar	
Employer_Company	Varchar	
Industry_Name	Varchar	
Website	Varchar	

## 3> Job

Job_id	Int	Primary Key
Job_title	Varchar	
Job_Description	Varchar	
Company_Name	Varchar	
Job_Location	Varchar	
Job_Salary	Int	
Category_id	Int	Foreign Key , CategoryTB
Employer_id	Int	Foreign Key , EmployerTB

## 4> Application

Application_id	Int	Primary Key
Job_id	Int	Foreign Key , JobpostingTB
User_id	Int	Foreign Key , UserTB
Application_Status	Varchar	
Date_applied	Date	

### 5> Resume

Resume_id	Int	Primary Key
User_id	Int	Foreign Key , UserTB
Resume_path	Varchar	

### 6> Category

Category_id	Int	Primary Key
Category_Name	Varchar	

### 7> Admin

Admin_id	Int	Primary Key
Admin_Name	Varchar	
Admin_Email	Varchar	
Admin_Password	Varchar	
Admin_Phonenumner	Int	

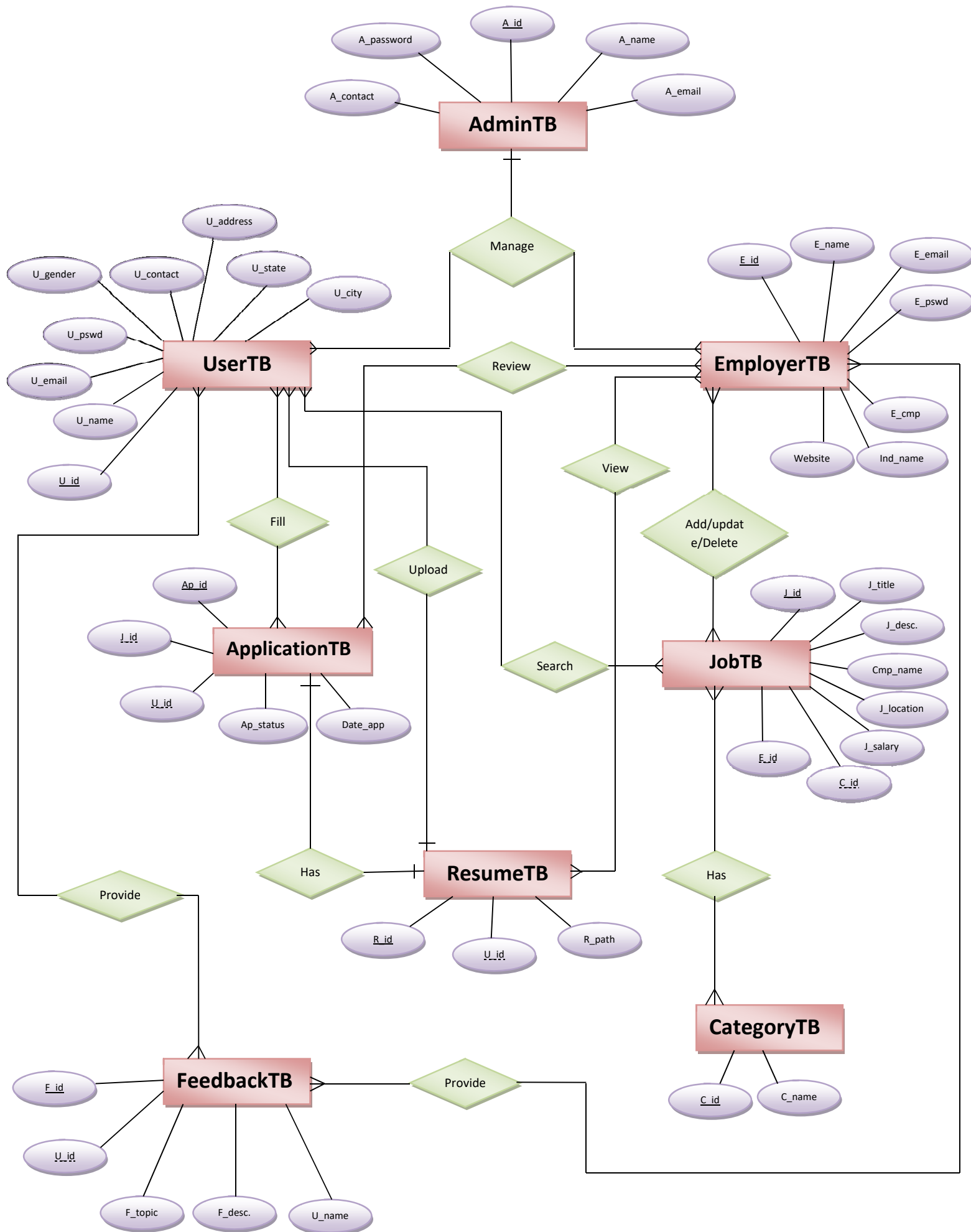
### 8> Feedback

Feedback_id	Int	Primary Key
User_id	Int	Foreign Key , UserTB
Feedback_Topic	Varchar	
Feedback_Description	Varchar	
User_Name	Varchar	

# System Analysis and Modeling

---

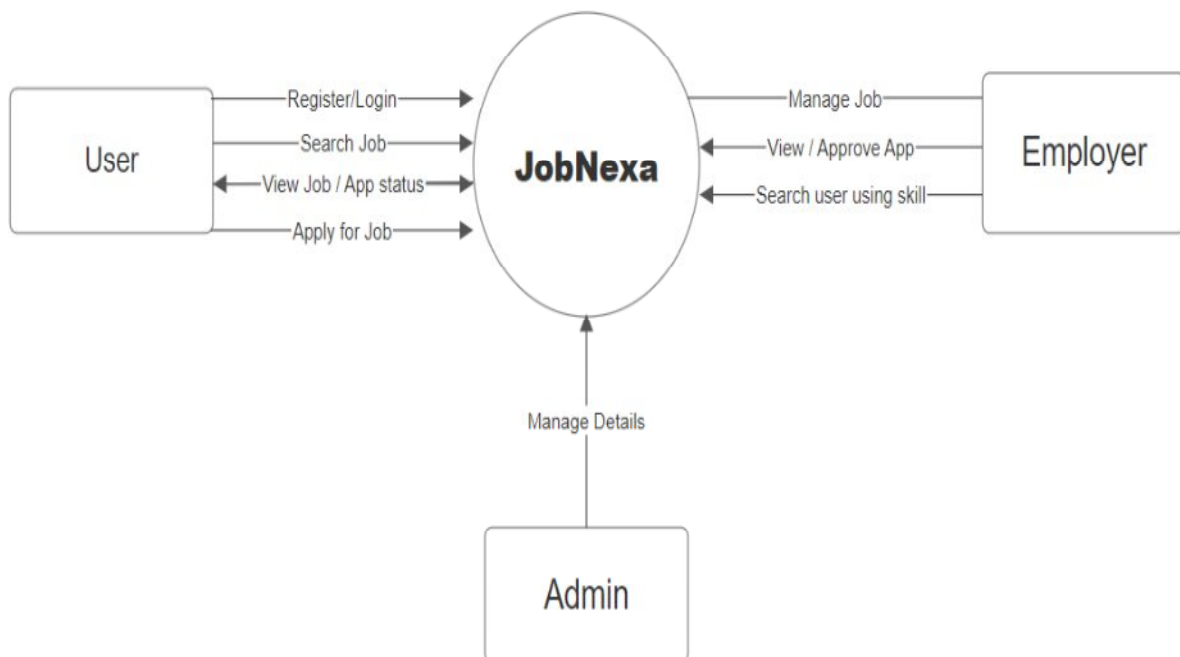
ER Diagram



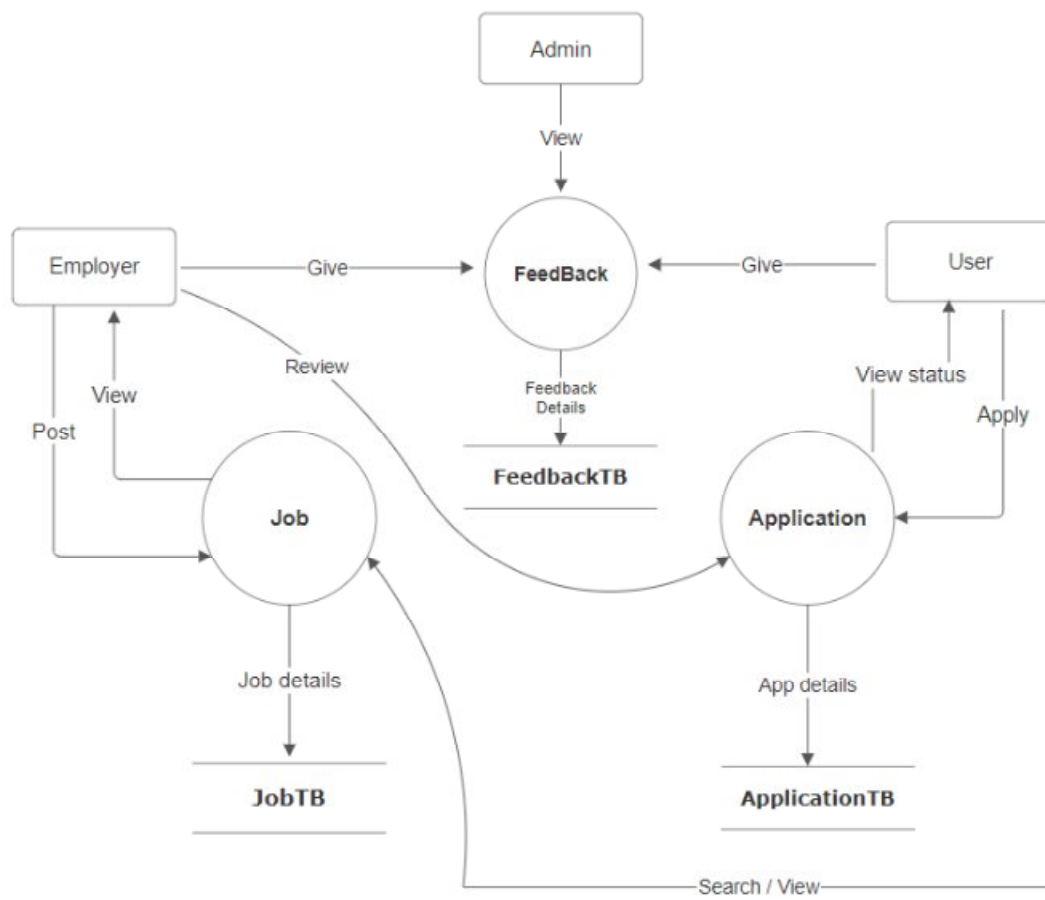


# Data Flow Diagram

## Level: 0

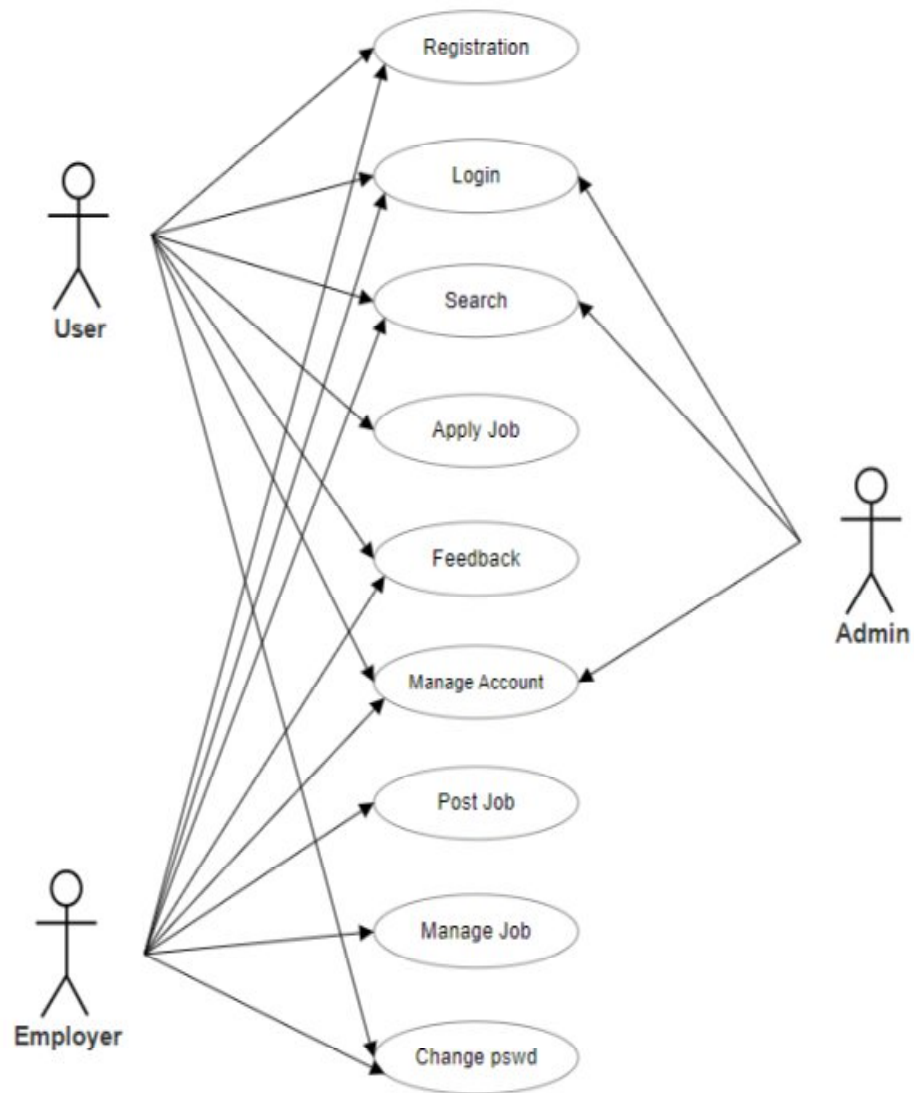


Level: 1

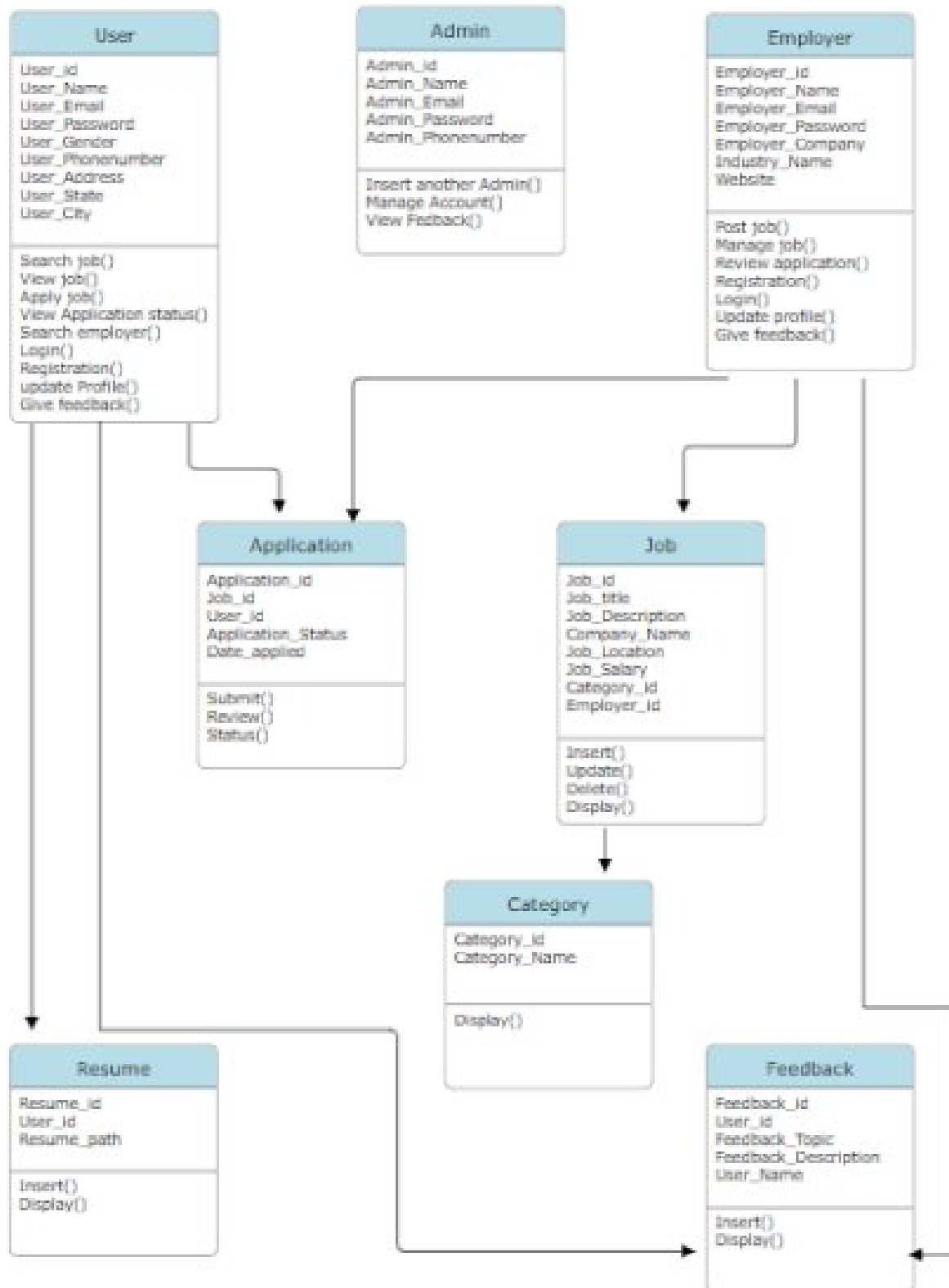


# UML

## ❖ Use case Diagram

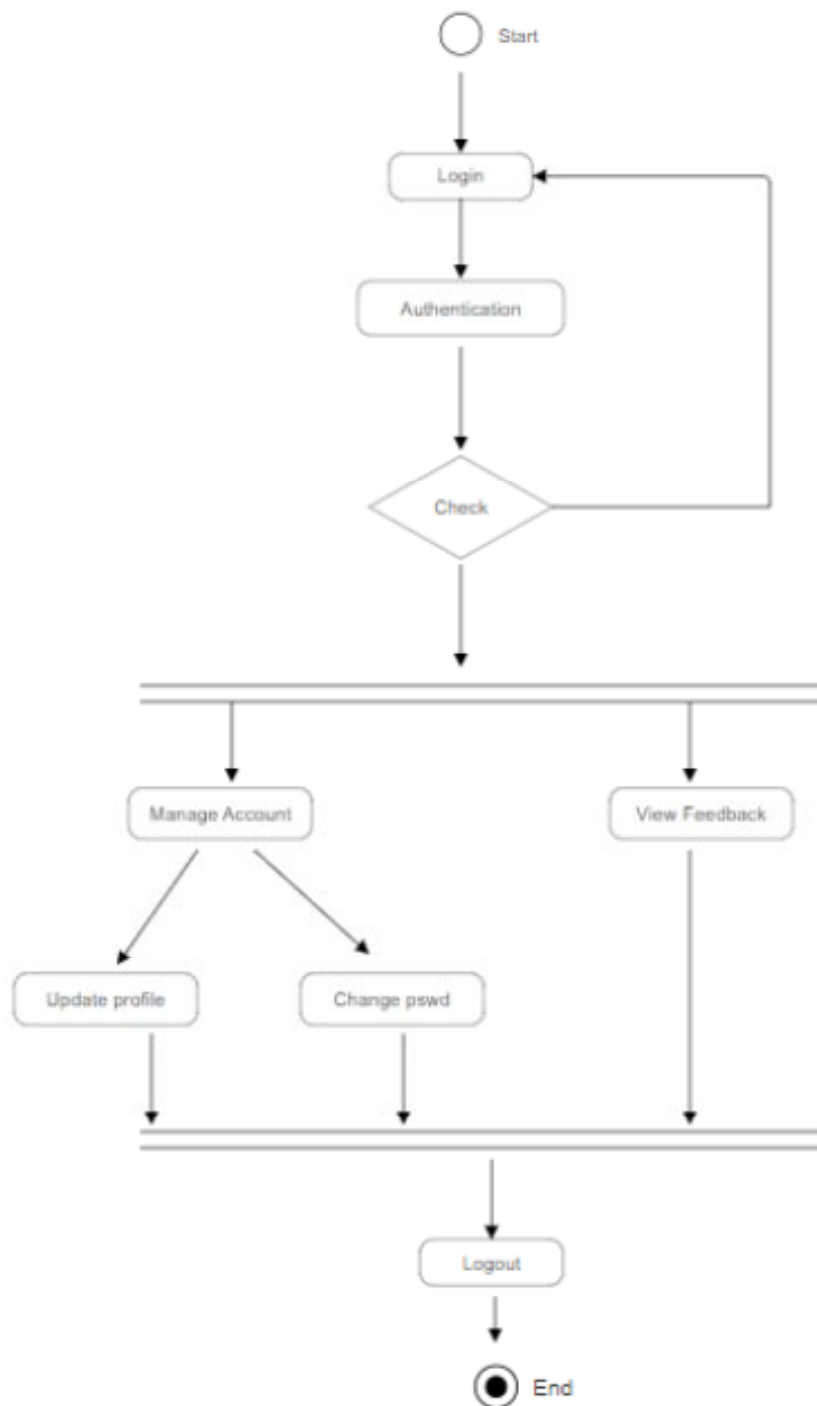


## ❖ Class Diagram

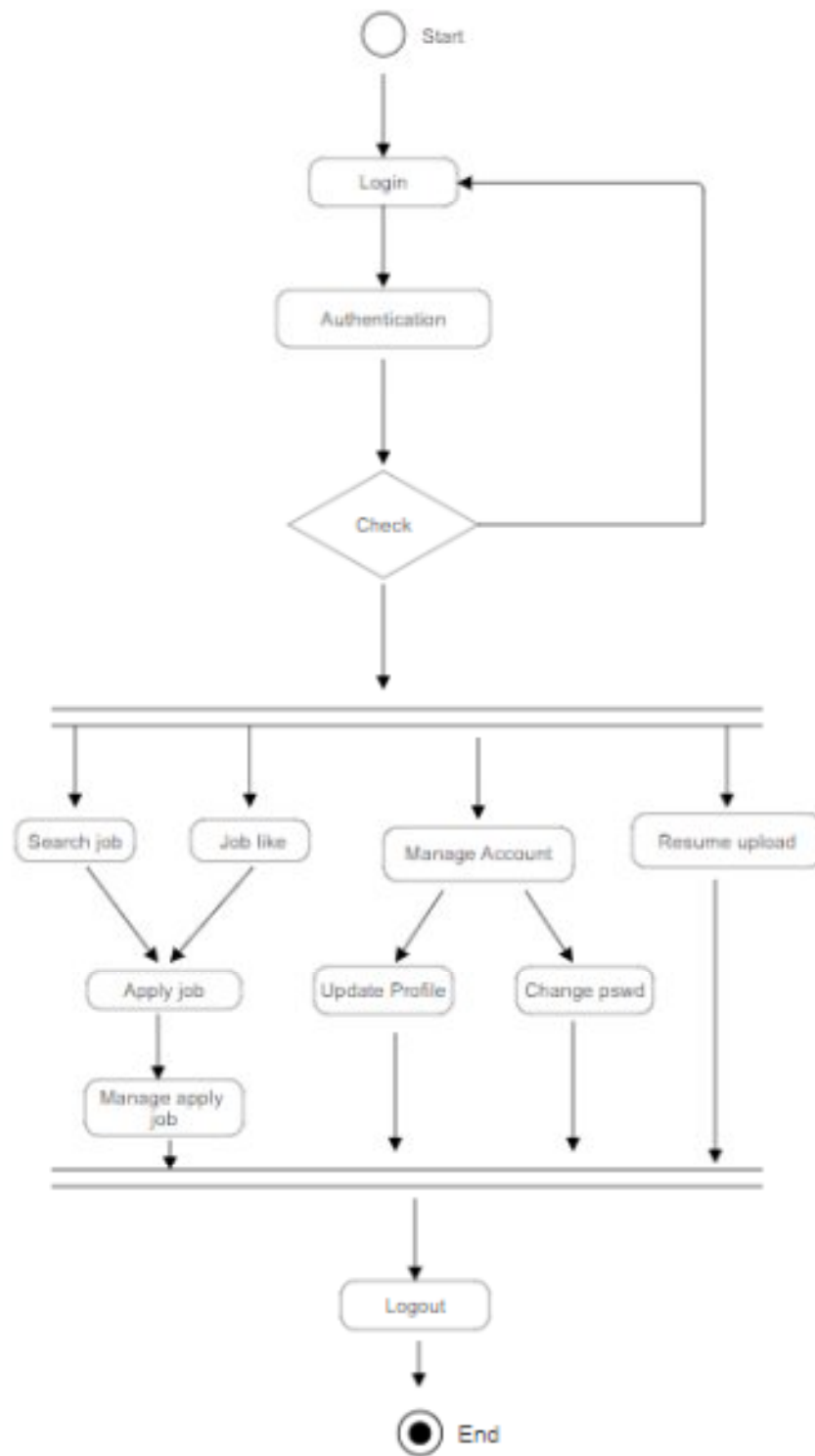


## ❖ Activity Diagram

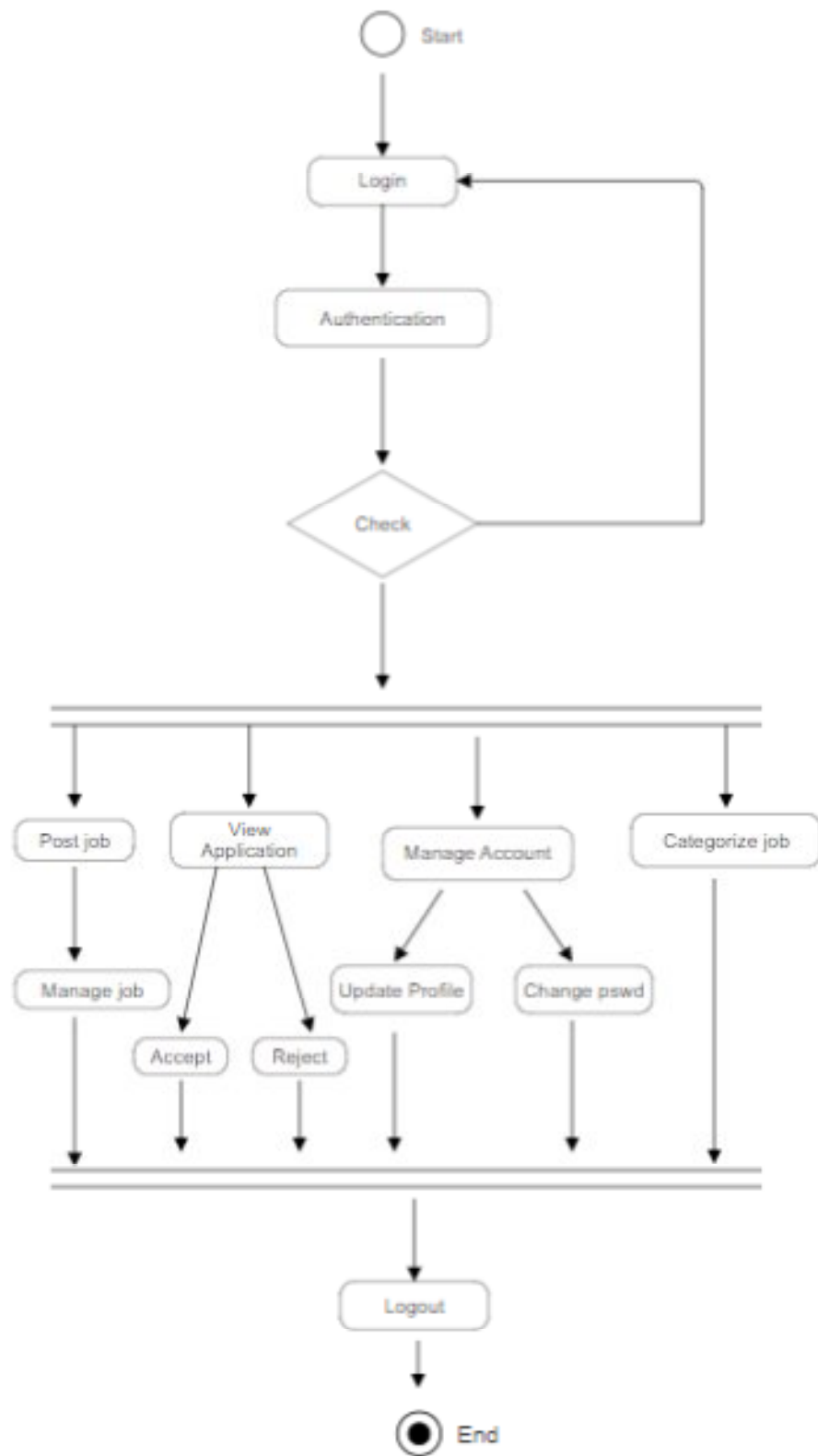
Admin



# User

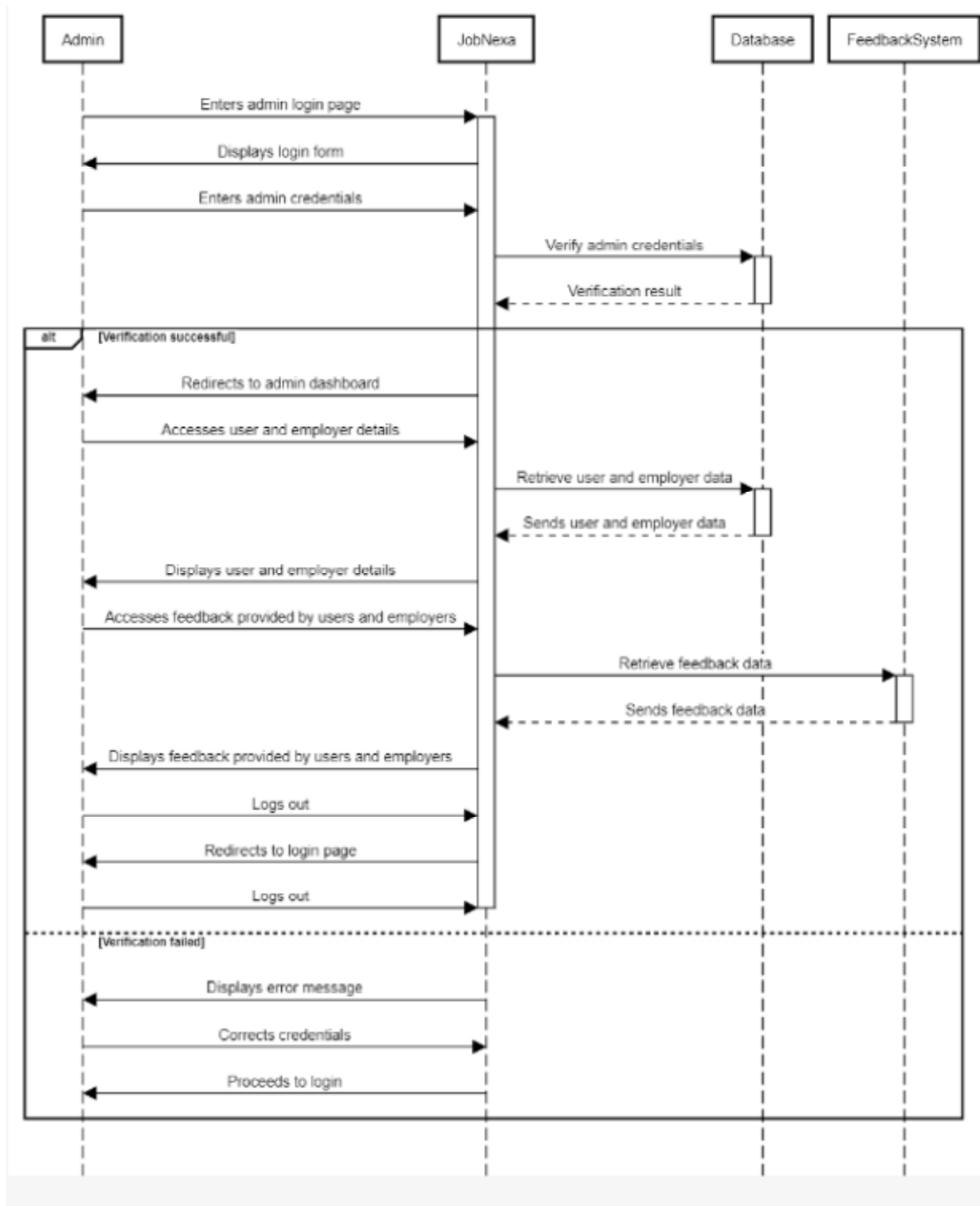


# Employer



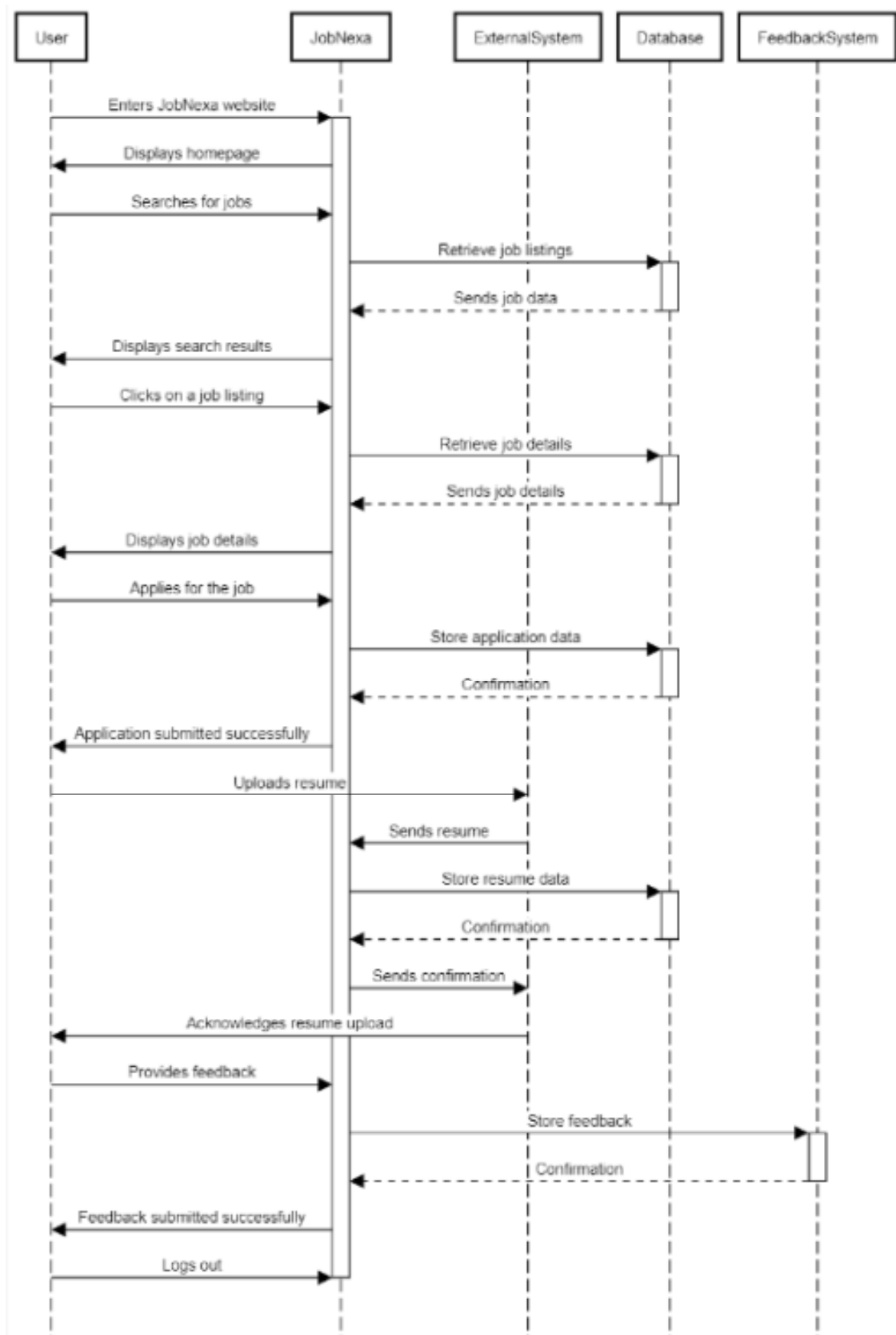
## ❖ Sequence Diagram

Admin



User





Employer

