# INDEX

| Sr.No | Description | Date | CO | Sign |
|---|---|---|---|---|
| 1. | Study of UI life cycle | | CO1 | |
| 2. | Study of open source UX Tools | | CO1 | |
| 3. | Prepare Project **Proposal and Requirement Gathering (Choose the project)**<br>The project should be a web, desktop, or mobile interface. If the chosen project is a mobile application, note that it must at least be possible to simulate the project, since one of the prototypes will be such a simulation that can be evaluated. | | CO2 | |
| 4. | **Analysis**<br>**Problem statement:**<br>Briefly state the problem(s) that the project will seek to solve. Take the user's point of view. Consider what the user's goals are, and what obstacles in the way.<br>**Output:**<br>• Write up a user analysis, task analysis (identify three tasks of the chosen problem), and domain analysis clearly, concisely, and completely.<br>• A problem object model or entity-relationship diagram. | | CO1 | |
| 5. | Create a Social model of the chosen Project. | | CO2 | |
| 6. | Identify the Users and Design a User persona. | | CO2 | |
| 7. | **Design Creation of Scenario**<br>Write a scenario that involves all three of the tasks identified for the chosen project.<br>Output:<br>• Explain the Scenario<br>• Sketch the scenario (use any tool or hand sketches) | | CO2 | |
| 8. | Draw a mental model for the above drawn scenario. | | CO2 | |
| 9. | Create High-Fidelity prototype (Wire Frame) using Figma tool. | | CO3 | |
| 10 | Create Prototype for Chosen Project. | | CO3 | |
| 11 | Design Customer Journey map. | | CO4 | |
| 12 | Perform UX Evaluation of Chosen Project.<br>Testing of User Interface from Third Party (Test scripts). | | CO4 | |

**PRACTICAL 1: - INTRODUCTION TO UI LIFE CYCLE AND UI TOOLS.**

## Introduction

A user interface also called a "UI" or simply an "interface," is how a person controls a software application or hardware device.

● A user interface is the point of human-computer interaction and communication on a device, webpage, or app.

● This can include display screens, keyboards, a mouse, and the appearance of a desktop. User interfaces enable users to effectively control the computer or device they are interacting with.

● User interface is important to meet user expectations and support effective functionality.

● A successful user interface should be intuitive, efficient, and user-friendly.

● Nearly all software programs have a graphical user interface or GUI. This means the program includes graphical controls, which the user can select using a mouse or keyboard.

● A typical GUI of a software program includes a menu bar, toolbar, windows, buttons, and other controls.

● User Interface Design is the craft and process of designing what a user interacts with when communicating with software.

## Types of UI

User Interface (UI) refers to the visual elements, controls, and interactions that users experience when interacting with software or hardware. There are several types of UI, each serving different purposes and contexts. Here are some common types of UI:

1. **Graphical User Interface (GUI):** GUI is the most prevalent type of

UI. It uses graphical elements such as icons, buttons, windows, and menus to allow users to interact with a system. Operating systems like Windows, macOS, and Linux, as well as many software applications, use GUI.

2. **Command Line Interface (CLI):** CLI relies on text commands to communicate with a system. Users enter commands through a terminal or command prompt to perform tasks. While it may seem less user-friendly to

   some, it can be powerful and efficient for users familiar with specific commands.

3. **Voice User Interface (VUI):** VUI allows users to interact with a system using spoken commands. Popular examples include virtual assistants like Amazon Alexa, Google Assistant, and Apple's Siri. VUI is becoming increasingly common in smart homes, cars, and mobile devices.

4. **Touch User Interface (TUI):** TUI relies on touch gestures to control and interact with a system. Common in smartphones, tablets, and touch-enabled devices, TUI enables users to tap, swipe, pinch, and perform other gestures to navigate and manipulate content.

5. **Augmented Reality (AR) and Virtual Reality (VR) Interfaces:** AR and VR interfaces provide immersive experiences. AR overlays digital information onto the real world, while VR creates a completely virtual environment. Both use specialized hardware, such as headsets, to deliver interactive experiences.

6. **Gesture-Based Interface:** This interface type allows users to interact with a system using gestures, which can include movements like swiping, waving, or other physical actions. Devices like Microsoft's

Kinect and some smart TVs use gesture-based interfaces.

7. **Web User Interface (WUI):** WUI refers to the user interface elements found on websites. It includes navigation menus, buttons, forms, and other elements that users interact with while browsing the web.
8. **Natural Language Interface:** This type allows users to interact with a system using natural language, such as text or speech. Chatbots and virtual assistants often use natural language interfaces to understand and respond to user queries.
9. **Haptic User Interface (HUI**): Haptic interfaces provide tactile feedback to users, allowing them to feel sensations or vibrations. This is commonly used in gaming controllers, virtual reality devices, and some touchscreen interfaces.
10. **Biometric User Interface**: This type involves the use of biometric data (such as fingerprints, facial recognition, or iris scanning) for user authentication and interaction. Biometric UI is commonly found in smartphones, access control systems, and security applications.
11. **Brain-Computer Interface (BCI)**: BCI allows users to control devices or software using brain signals. Electroencephalography (EEG) is often used to detect brain activity, enabling users to interact with computers or other devices through their thoughts.
12. **Tangible User Interface (TUI):** TUI involves physical objects or manipulatives to interact with a digital system. For example, a tabletop interface where users can manipulate physical objects that are tracked and interpreted by the system.
13. **Multi-Modal User Interface:** Combining multiple modes of interaction, such as voice, touch, gesture, and more, into a single interface. This

approach aims to provide a richer and more flexible user experience.

14. **Adaptive User Interface**: An interface that dynamically adjusts its layout, content, or behavior based on user preferences, behavior, or contextual information. This helps personalize the user experience and cater to individual needs.
15. **Contextual User Interface**: This type of interface adapts based on the context of use. It takes into account factors such as location, device capabilities, user preferences, and other environmental conditions to optimize the user interface.
16. **Kiosk User Interface:** Commonly found in public places, kiosk interfaces are designed for self-service transactions. Users interact with a fixed, standalone terminal to perform tasks such as information retrieval, ticketing, or payment.
17. **Wearable User Interface**: Interfaces designed for wearable devices, such as smartwatches or fitness trackers. They often have limited screen space and utilize gestures, touch, or voice for interaction.
18. **Responsive User Interface**: An interface that adjusts its layout and design to accommodate various screen sizes and resolutions. This is crucial for delivering a consistent user experience across different devices, such as desktops, tablets, and smartphones.

## UI Life Cycle:

The User Interface (UI) life cycle refers to the different stages and processes involved in designing, implementing, testing, and maintaining a user interface for a software application or system. The UI life cycle typically includes the following phases:

1. **User Research and Analysis:**

    - **User Research:** Understanding the target audience, their needs,

preferences, and expectations through methods like surveys, interviews, and observations.

- **Competitor Analysis:** Analyzing the user interfaces of similar products or systems to identify trends and best practices.

2. **Requirement Gathering:**

- **Define Objectives:** Establishing the goals and objectives of the user interface based on user needs and business requirements.
- **Functional Requirements:** Identifying the specific features and functionalities that the UI must support.

3. **UI Design:**

- **Information Architecture:** Organizing and structuring information to create a logical and intuitive navigation flow.
- **Wireframing:** Creating low-fidelity sketches or wireframes to outline the basic layout and structure of the UI.
- **Visual Design:** Defining the aesthetics, including colors, typography, images, and overall visual style.
- **Prototyping:** Developing interactive prototypes to simulate the user experience and gather feedback.

4. **Implementation/Development:**

- **Front-end Development:** Translating the design into actual code using web technologies (HTML, CSS, JavaScript) or other programming languages.
- **Back-end Integration:** Connecting the UI with the application's

back- end services and databases.

- Usability Testing during Development: Conducting iterative testing to identify and address usability issues as the UI is developed.

5. Testing:

- Usability Testing: Evaluating the UI with actual users to identify any usability issues and gather feedback for improvements.
- Cross-Browser and Cross-Platform Testing: Ensuring the UI functions correctly on different browsers and platforms.
- Performance Testing: Assessing the speed and responsiveness of the UI under various conditions.

6. Deployment:

- Release Planning: Planning the deployment of the UI, considering factors such as timing, user communication, and potential impact on existing users.
- Rollout: Deploying the UI to production, making it accessible to users.

7. Monitoring and Maintenance:

- Monitoring Usage: Analyzing user interactions and feedback to identify areas for improvement.
- Bug Fixing: Addressing any issues or bugs that arise after deployment.
- Updates and Enhancements: Implementing updates or enhancements based on user feedback, changing requirements, or emerging trends.

# PRACTICAL 2: STUDY OF UI TOOLS

1. **Sketch:**
    - *Key Features:*
        - Vector-based design for creating scalable interfaces.
        - Extensive library of plugins for additional functionalities.
        - Artboards for organizing and presenting designs.
        - Robust symbols system for reusability.
        - Pixel-perfect design and export features.
2. **Figma:**
    - *Key Features:*
        - Web-based, allowing collaborative design in real-time.
        - Cross-platform accessibility (works on Windows, macOS, Linux).
        - Auto Layout and constraints for responsive design.
        - Prototyping features for creating interactive user flows.
        - Design systems and components for consistency.
3. **Adobe XD:**
    - *Key Features:*
        - Part of the Adobe Creative Cloud, facilitating integration with other Adobe tools.
        - Vector-based design with support for prototyping.
        - Voice prototyping for designing voice-enabled experiences.
        - Plugins and integrations with third-party tools.
        - Collaboration features for design teams.

4. **InVision:**
    - *Key Features:*
        - Prototyping and animation tools for creating interactive experiences.
        - User testing and collaboration features.
        - Inspect mode for developers to extract design specifications.
        - Design System Manager for maintaining design consistency.
        - Integrations with popular design tools.
5. **Axure RP:**
    - *Key Features:*
    - Advanced prototyping with dynamic content and logic.

Conditional flows and interactions for complex user journeys

- Annotations and documentation features for detailed specifications.
- Team collaboration and version control.
- Integration with other design tools.

6. **Zeplin:**
    - *Key Features:*
        - Bridges the gap between designers and developers.
        - Export designs with style guides and assets.
        - Version history for design iterations.
        - Commenting and collaboration features.
        - Integrations with various design tools.
1. **Marvel:**
    - *Key Features:*
        - User-friendly interface for designing and prototyping.

- Collaboration features for remote teams.
- User testing and feedback collection.
- Integrations with popular design tools.

Design versioning and history.

7. **Proto.io:**
    - *Key Features:*
        - Web-based prototyping tool for web and mobile applications.
        - Rich library of UI components and interactions.
        - Real-time collaboration and user testing.
        - Animation and gesture support.

    Integrations with design and project management tool
8. **Balsamiq:**
    - *Key Features:*
        - Low-fidelity wireframing tool for quick ideation.
        - Focus on simplicity and ease of use.
        - Sketch-style wireframes for early-stage design concepts.
        - Collaboration features for team projects.
        - Integration with Jira and other project management tools.
9. **Autodesk SketchBook:**
    - *Key Features:*
        - Drawing and sketching tool with a variety of brushes and tools.
        - Cross-platform support (Windows, macOS, iOS, Android).
        - Customizable brushes and drawing settings.
        - Layers for organizing and editing sketches.

- Suitable for concept sketching and ideation.

## PRACTICAL 3: PROJECT PROPOSAL AND REQUIREMENT GATHERING

**Project Title**: Job Connect Portal

**Project Type**: Web Application

**Business Purpose**: Job Connect aims to streamline the recruitment process by providing a centralized platform for job seekers and employers. Users can create profiles, search and apply for jobs, while employers can post openings, manage applications, and schedule interviews.

**Target Users**:

- Job Seekers: Individuals actively searching for jobs or internships.
  Employers: Companies and recruiters looking to hire candidates.

**Requirement Gathering:**

1. **Functional Requirements:**

- User registration/login via Email
- Profile creation with resume upload
- Job browsing and advanced filtering
- Job application and status tracking

2. **Non-Functional Requirements:**

- Responsive web interface (React.js)
- Secure backend with Node.js and Express
- Data storage using MongoDB

3**. Tools & Technologies:**

- Frontend: React.js
- Backend: Node.js + Express.js
- Database: MongoDB

4**. Expected Output**: A working web-based job portal where users can register, apply for jobs, and employers can post jobs and interact with candidates

# PRACTICAL 4: ANALYSIS

**Problem Statement**: Many job seekers struggle to find appropriate opportunities and track their applications. Employers face difficulties in efficiently screening and managing a large volume of candidates.

**User Goals:**

- Find relevant job listings easily
- Apply quickly and track application status
- Communicate with employers

**Obstacles:**

- Overwhelming job platforms with poor filters
- Lack of feedback or status updates
- No direct communication channel

**Solution**: Job Connect Portal provides intuitive navigation, filtered search, and application status tracking in one platform

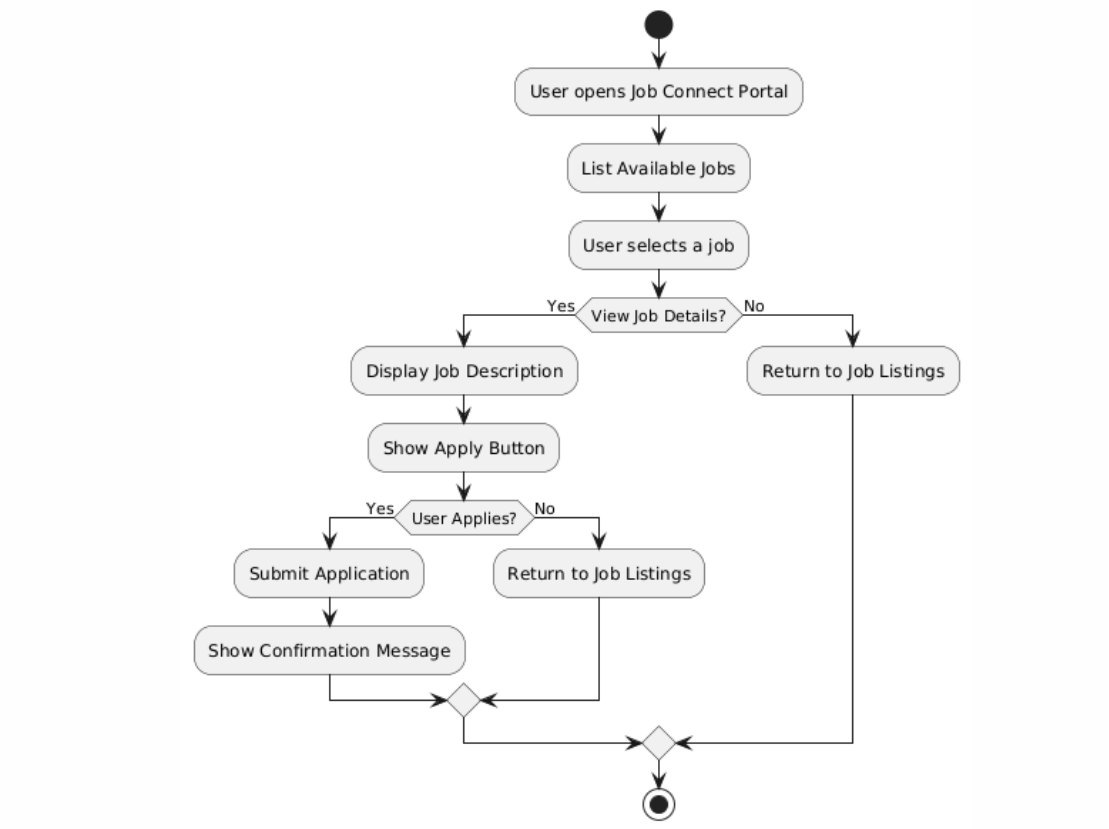
## SYSTEM DESIGN:

The **Job Connect Portal** follows a **three-tier architecture** consisting of:

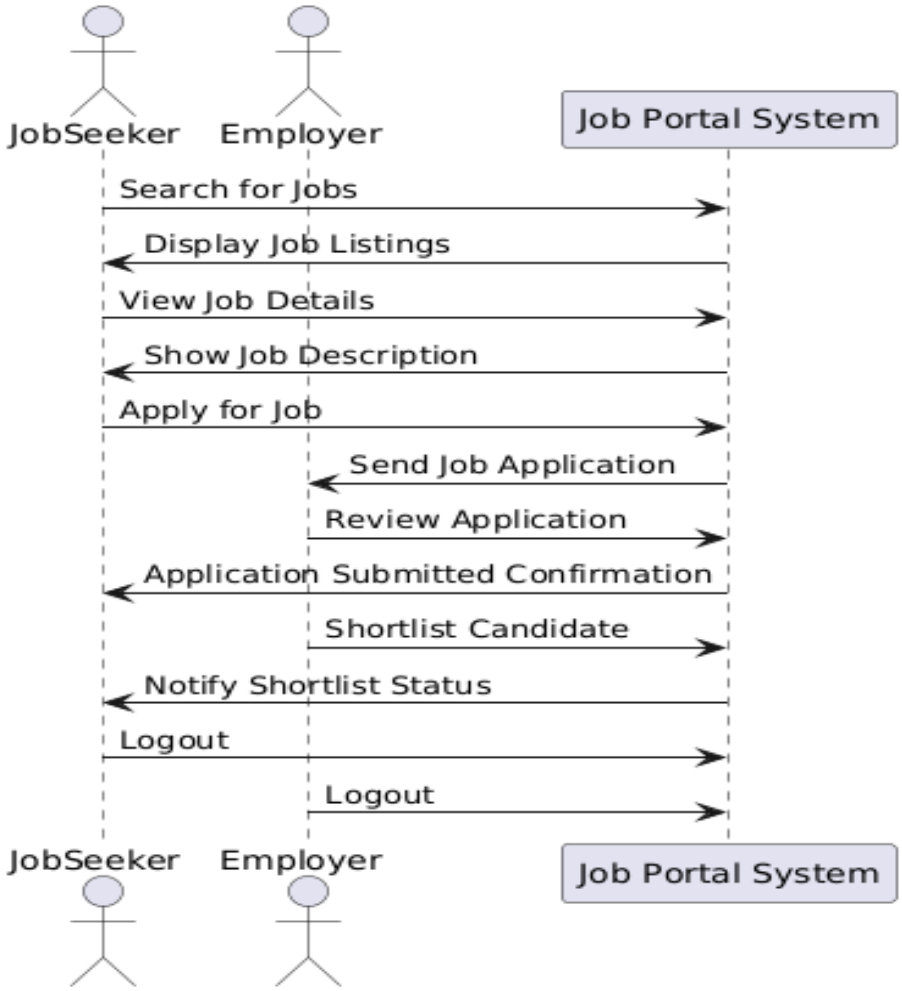
1. **Presentation Layer (Frontend):**
    - Handles user interactions.
    - Developed using **React.js** for a dynamic and responsive UI.
2. **Business Logic Layer (Backend):**
    - Manages job postings, applications, and user data.
    - Developed using **Node.js with Express.js** for handling API requests.
3. **Data Layer (Database):**
    - Stores job listings, user profiles, and application records.
    - Uses **MongoDB** for efficient data management.

| Diagram Type | Purpose | Focus | Representation |
|---|---|---|---|
| **Activity Diagram** | Represents job application workflows/processes | Job search, application flow | Start/End nodes, actions, decisions |
| **Sequence Diagram** | Models interaction between users and system | Communication between users, job status updates | Objects, lifelines, messages |
| **Data Flow Diagram** | Shows how job data moves in the system | Job posting, application, resume processing | Processes, data flows, data stores |
| **Architecture Diagram** | Depicts system structure | Frontend, backend, database interactions | Layers, components, connectors |

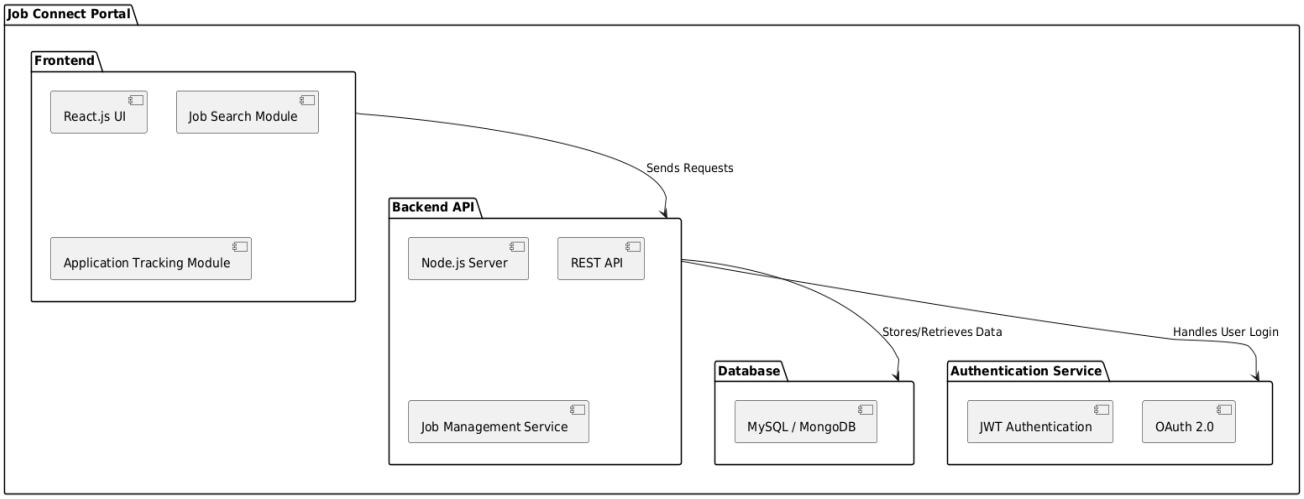
## ACTIVITY DIAGRAM



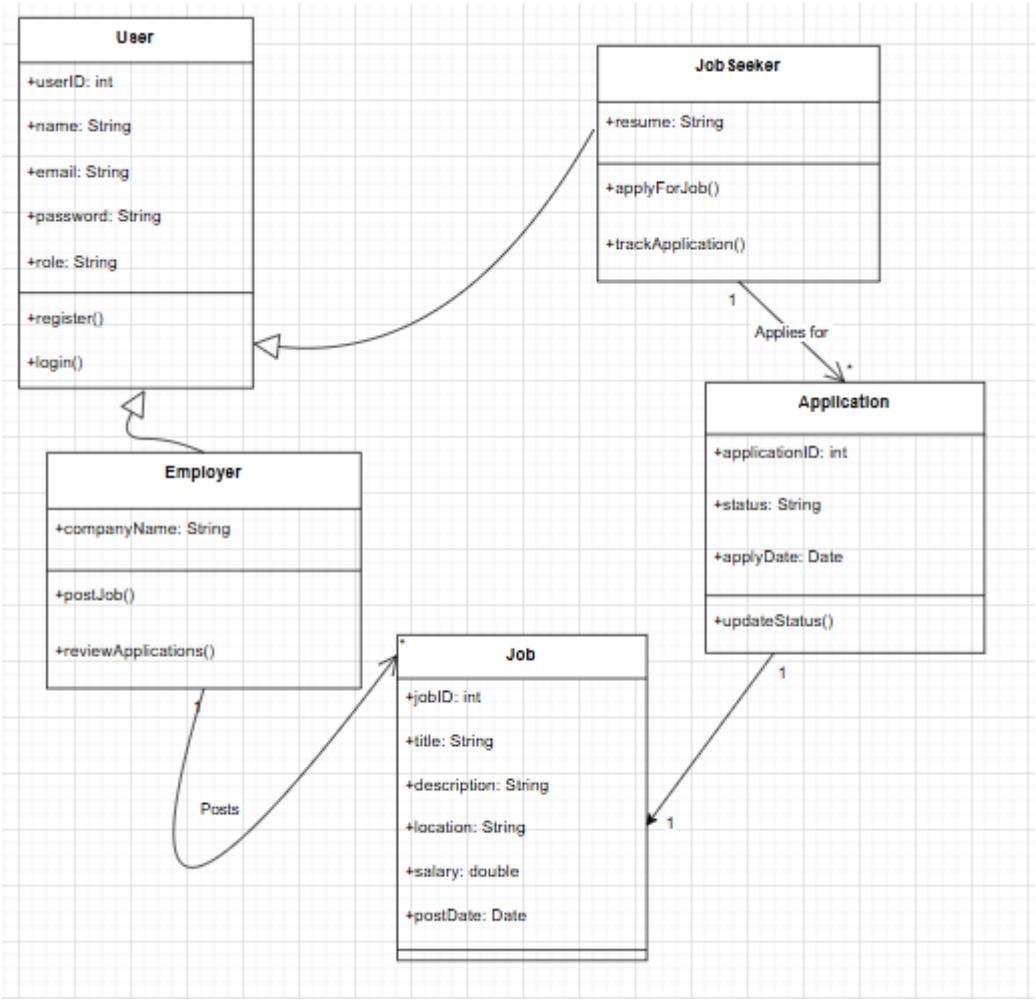
## SEQUENCE DIAGRAM:



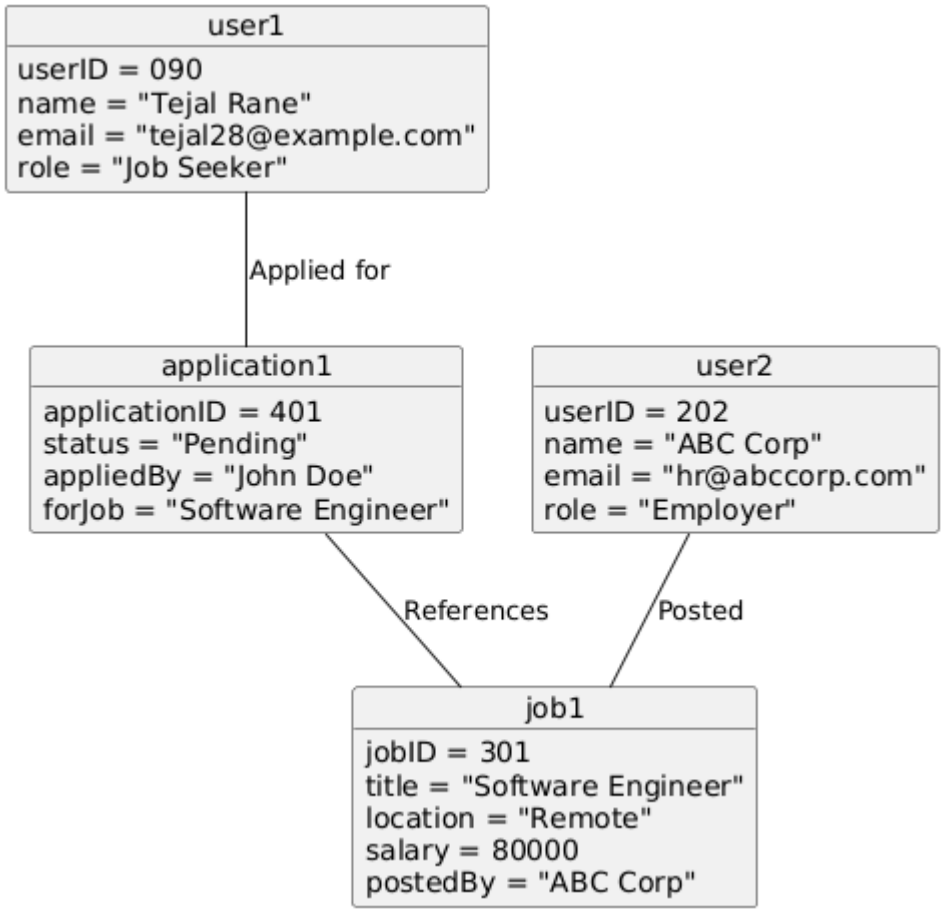

## ARCHITECTURE DESIGN



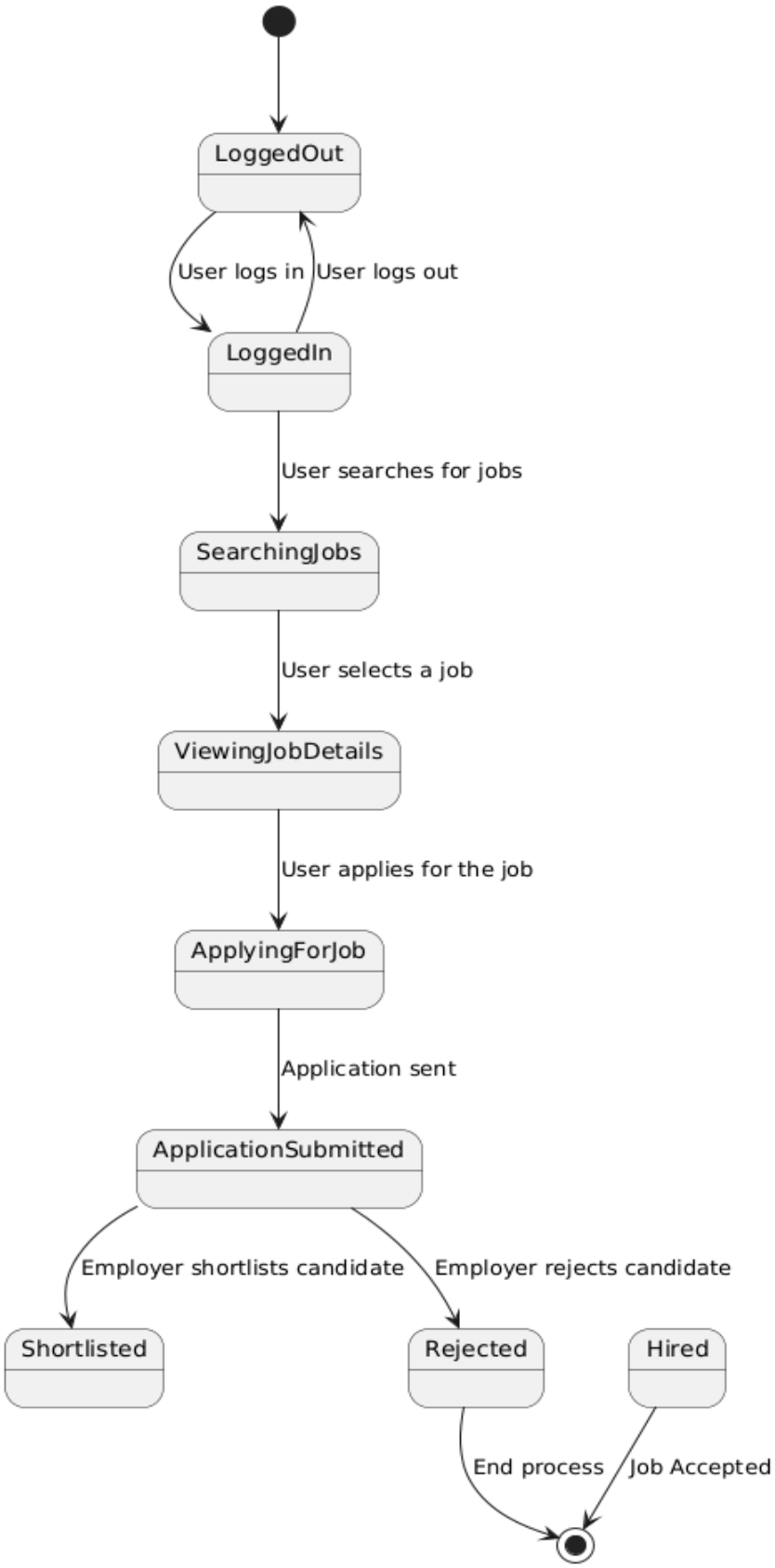
# CLASS DIAGRAM



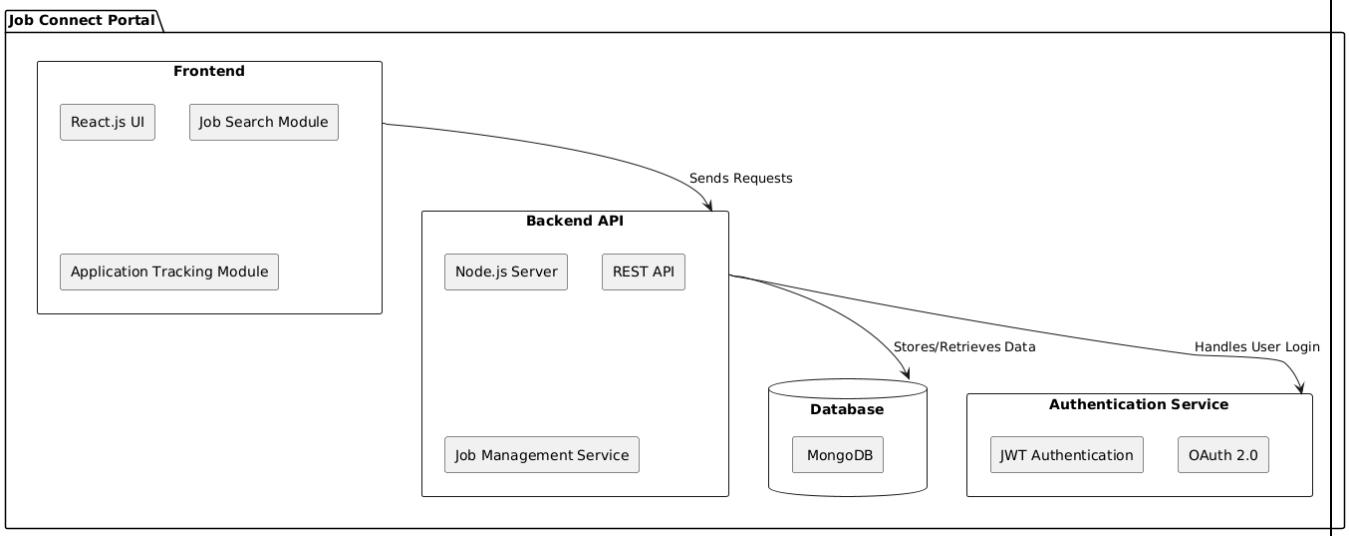
## OBJECT DIAGRAM



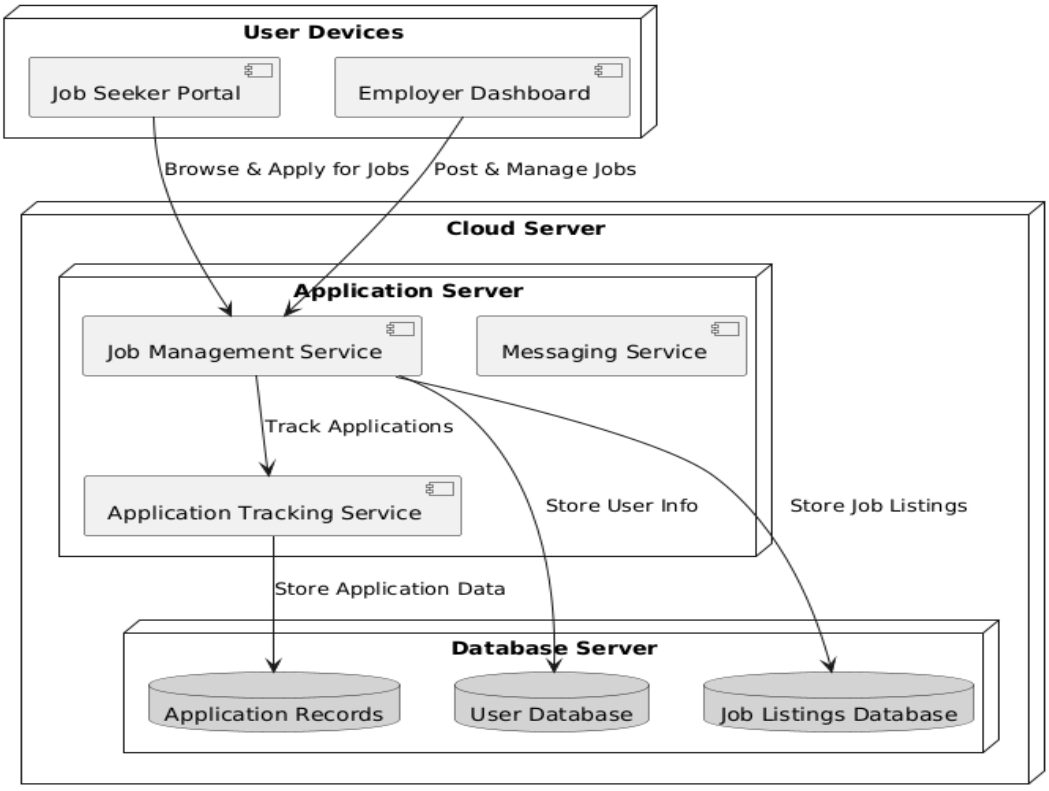
## STATE DIAGRAM



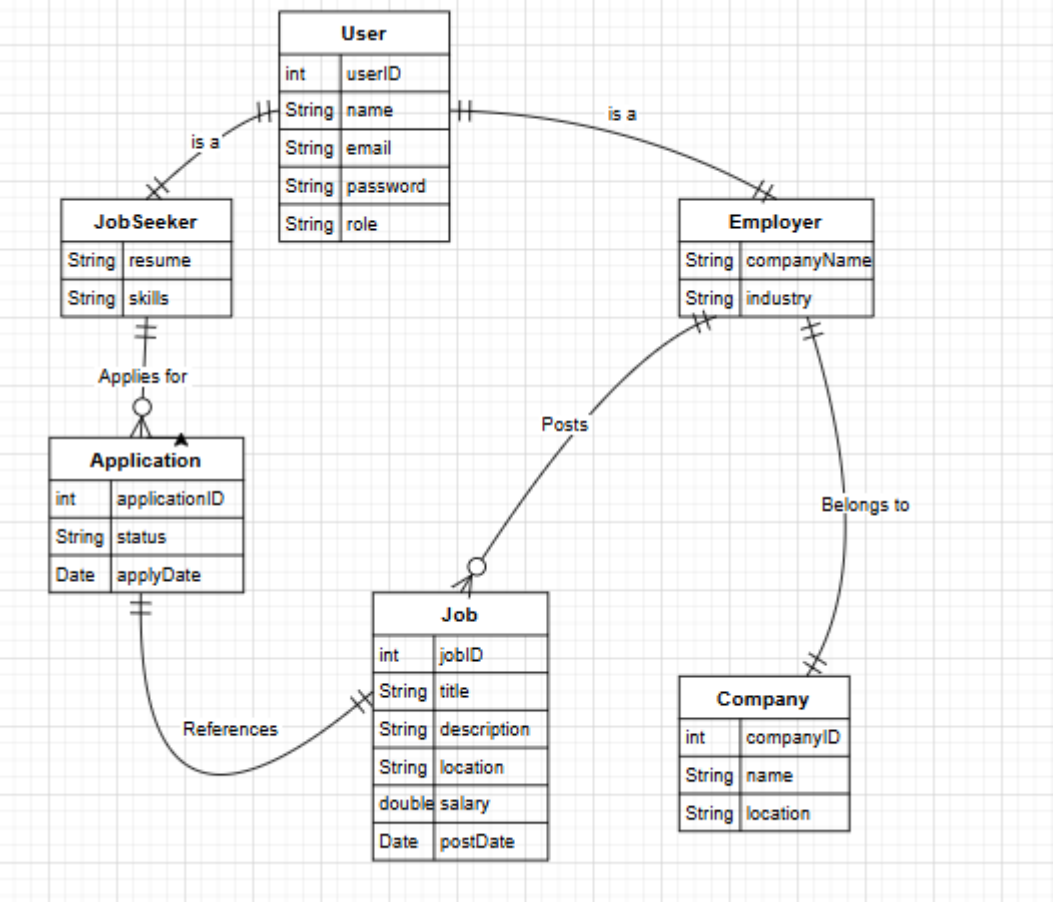
# COMPONENT DIAGRAM




# DEPLOYMENT DIAGRAM



# ER DIAGRAM


User
int userID
String name
String email
String password
String role
is a
is a
JobSeeker
String resume
String skills
Applies for
Application
int applicationID
String status
Date applyDate
Employer
String companyName
String industry
Posts
Belongs to
Job
int jobID
String title
String description
String location
double salary
Date postDate
References
Company
int companyID
String name
String location

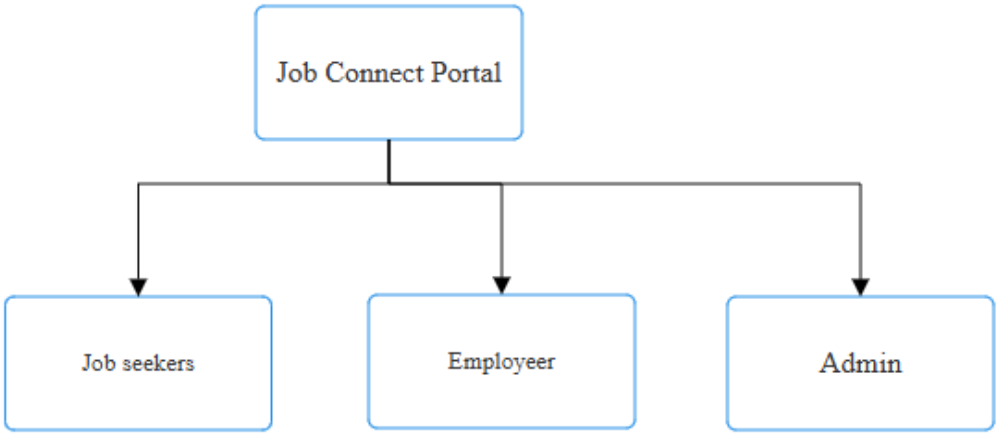
# PRACTICAL 5: SOCIAL MODEL OF THE PROJECT

**Project Title**: Job Connect Portal

**Social Model Overview**: This model identifies stakeholders and their interactions with the system.

**Stakeholders and Their Roles:**

| **Stakeholder** | **Role** |
|---|---|
| Job Seekers | Create profiles, upload resumes, apply for jobs, track applications |
| Employers | Post job listings, manage applicants, schedule interviews |
| Admin | Monitor platform activity, handle reports, manage disputes |




**Interaction Flow:**

Job Seeker ↔ Portal: Search, Apply, Communicate

Employer ↔ Portal: Post Jobs, Review, Schedule

**Social Impact:**

For Job Seekers: Streamlined application process.

For Employers: Efficient recruitment management.

For Society: Better employment matching and faster hiring.

## PRACTICAL 6: IDENTIFY THE USERS AND DESIGN A USER PERSONA

**Identified Users:**

1. Job Seekers:

- Looking for jobs based on skills

- Want a clean, fast application interface

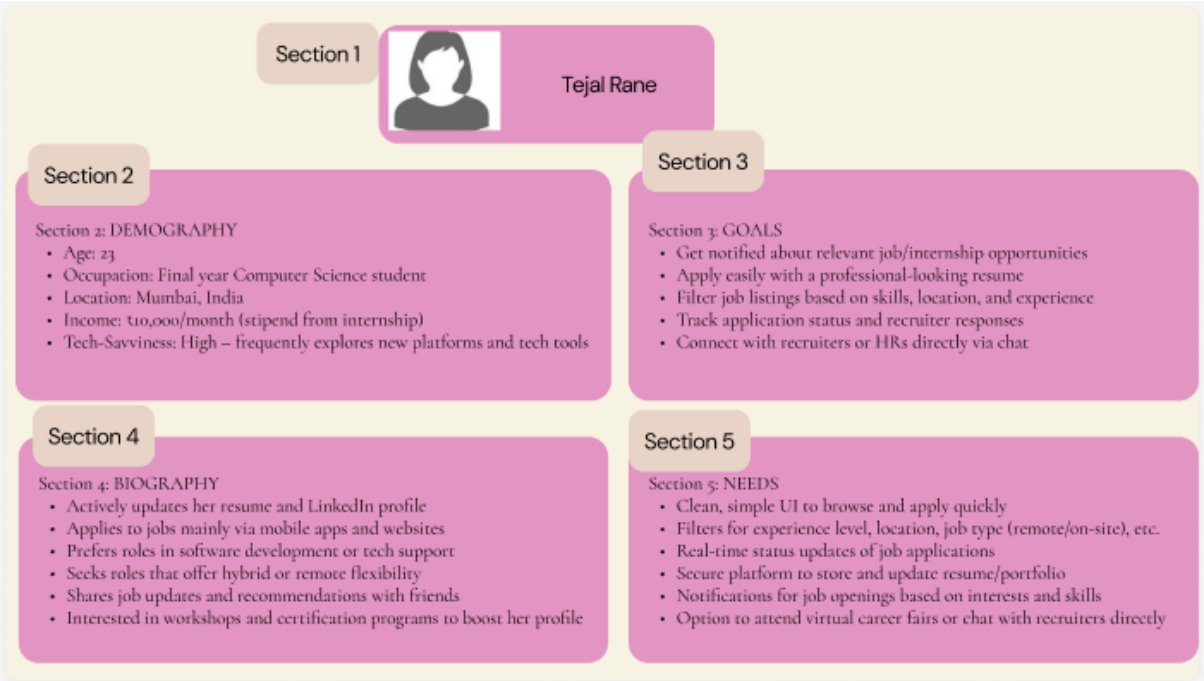
- Prefer to track status and receive updates

2. Employers:

- Need to post and manage job openings

- Want to review and shortlist candidates

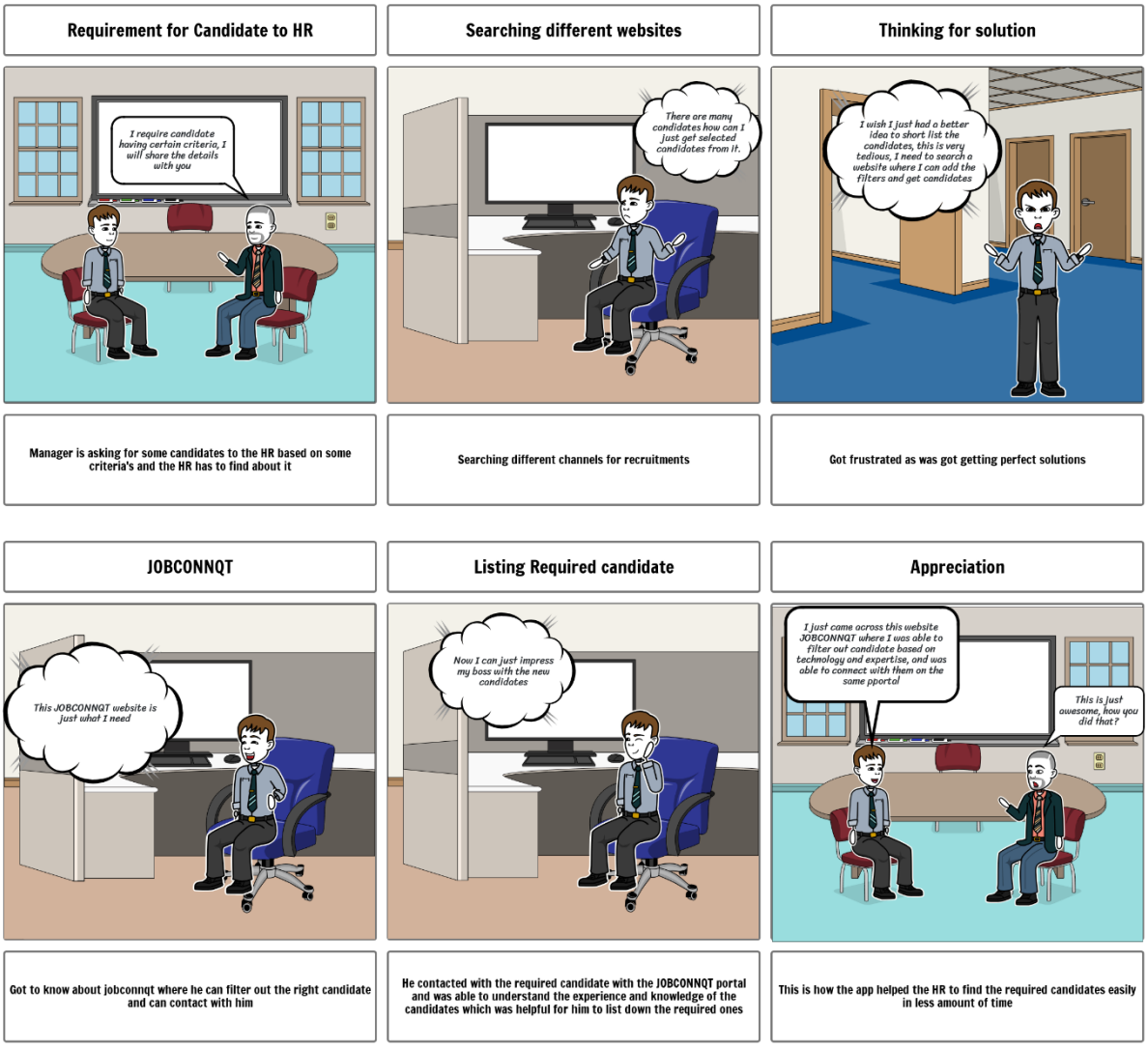
- Prefer simple communication and scheduling tools



# PRACTICAL 7: DESIGN — SCENARIO CREATION

**Scenario**: "From Job Search To Hiring"

An HR executive was tasked by his manager to find candidates matching specific criteria. Frustrated with searching across multiple job portals and not finding the right fit, he wished for a smarter solution. That's when he discovered **JOBCONNQT**, a powerful job portal that allowed him to filter candidates based on experience, skills, and qualifications. Using its intuitive interface, he quickly identified suitable applicants, contacted them directly, and created a shortlist. In the next meeting, the manager was impressed with the efficient results. Thanks to JOBCONNQT, the hiring process became faster, more accurate, and stress-free.



**PRACTICAL 8: DRAW MENTAL MODEL FOR THE ABOVE DRAWN SCENARIO**

## Mental Model of Job Connqt Portal

### 1. USER GOAL

• Find and apply to relevant job opportunities efficiently as a job seeker.

• Post jobs and find suitable candidates as an employer

### 2. PAIN POINTS (FRUSTRATIONS)

• Difficulty finding jobs or candidates that match preferences.

• Confusing or cluttered platforms with irrelevant listings.

• Time-consuming application or shortlisting process.

• Lack of updates after applying or posting.

• No personalized recommendations.

### 3. MOTIVATION TRIGGERS

• Recommendation from college, friends, or online groups.

• Need for a better-organized portal tailored for students and employers.

• Promise of faster, more relevant job/candidate matches.

### 4. SOLUTION (Job Connqt Portal)

**Features (Job Seeker Side):**

• Easy signup/login

• Job filtering by role, location, company, and skills

• Resume upload & tracking application status

• Personalized job suggestions

**Value Proposition:**

• Smooth, intuitive interface

• Customized experience for students and companies

• Real-time status updates and notifications

• Time-saving, streamlined hiring process

### 5. POSITIVE OUTCOMES

• Seeker finds suitable jobs with less effort

• Employer shortlists ideal candidates faster

• Faster communication between both parties

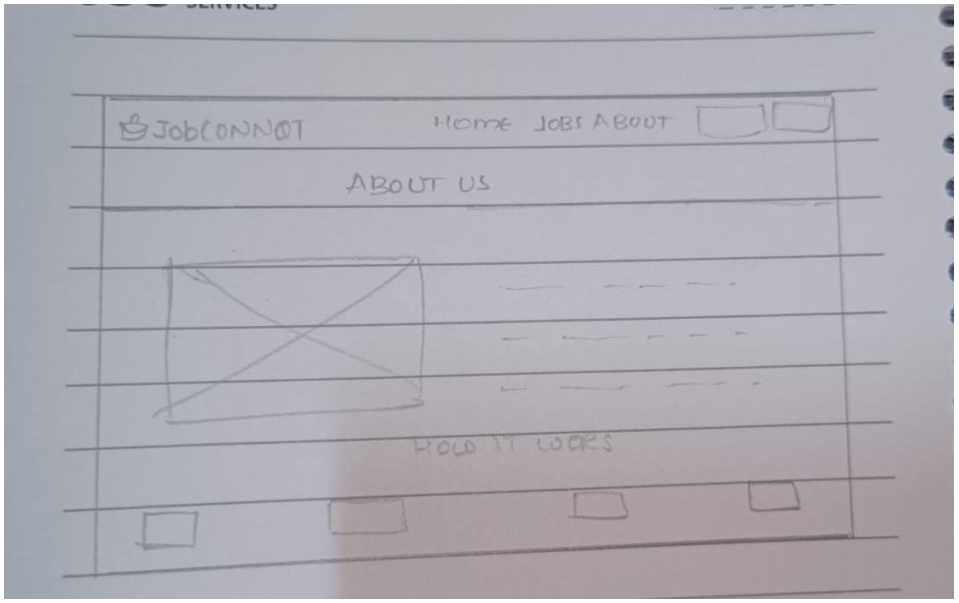
• Student gets placed and gains career confidence

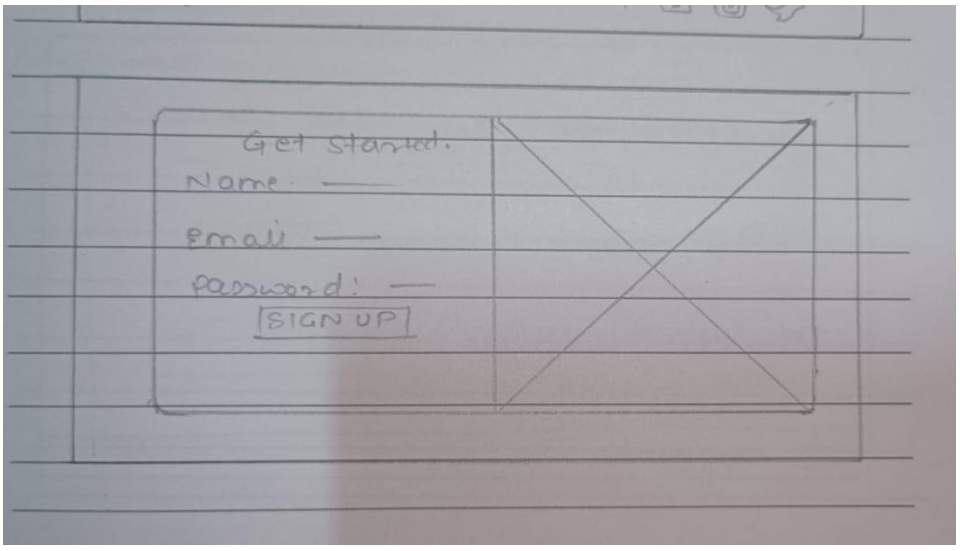
• Employer builds a reliable talent pipeline

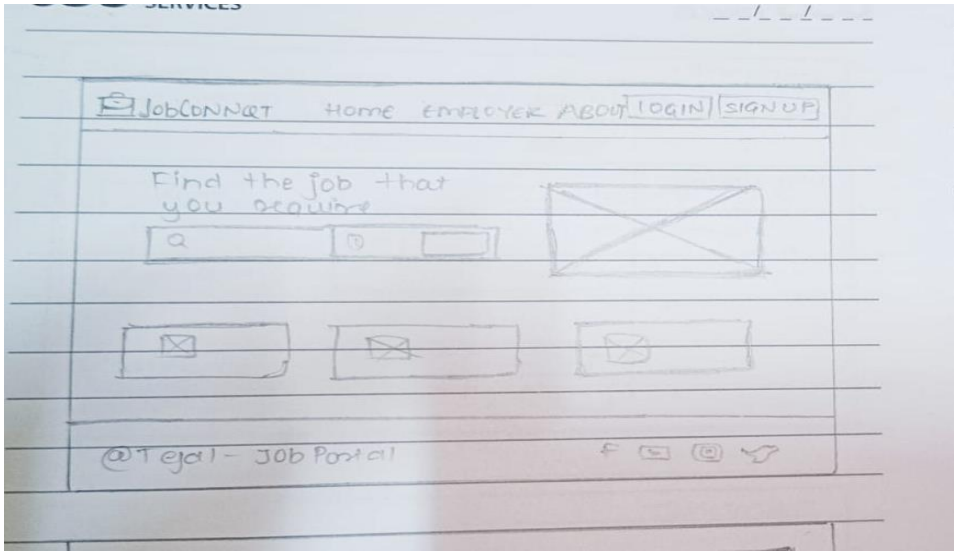
# PRACTICAL :9

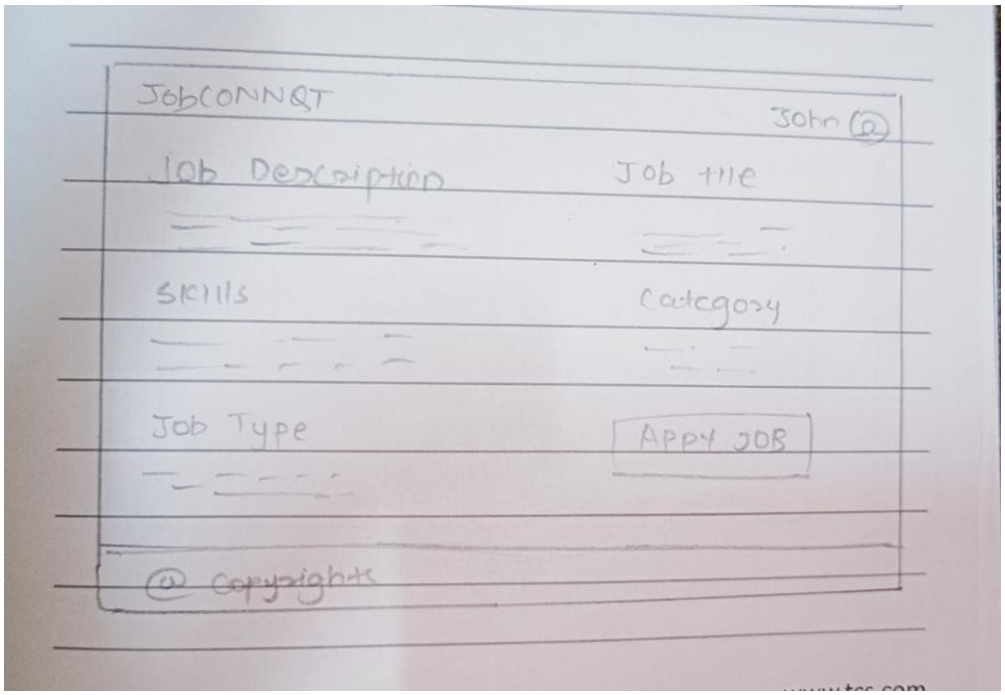
## CREATE HIGH-FIDELITY PROTOTYPE (WIRE FRAME) USING FIGMA TOOL.

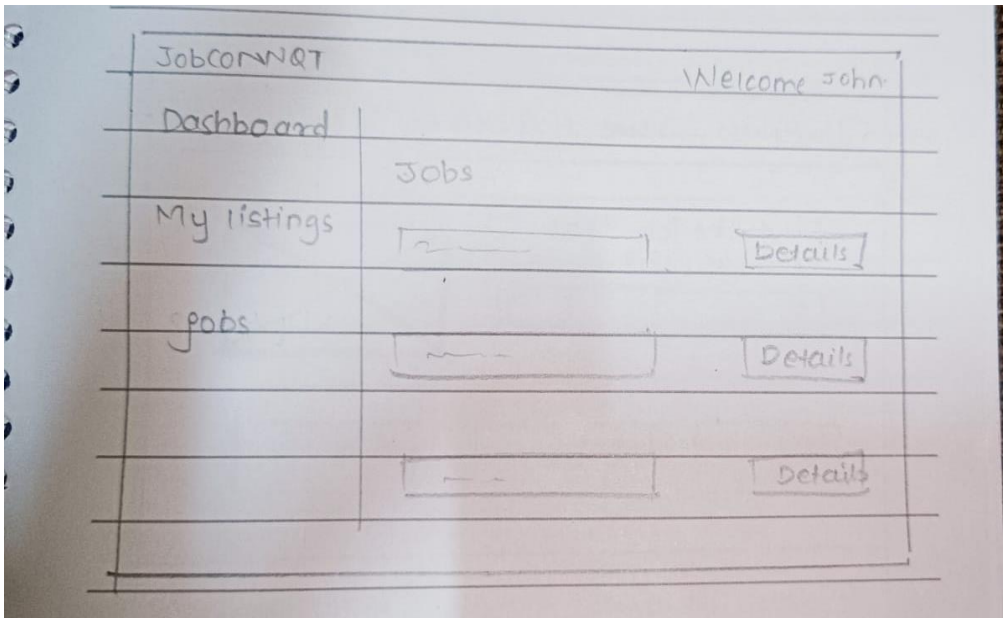
**Paper wireframe:**

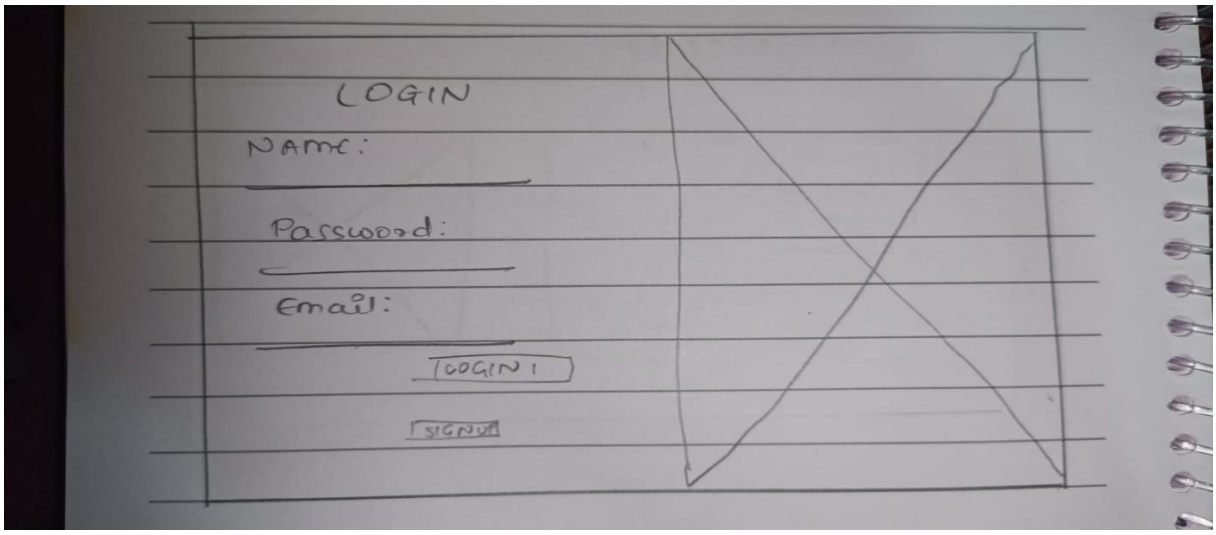


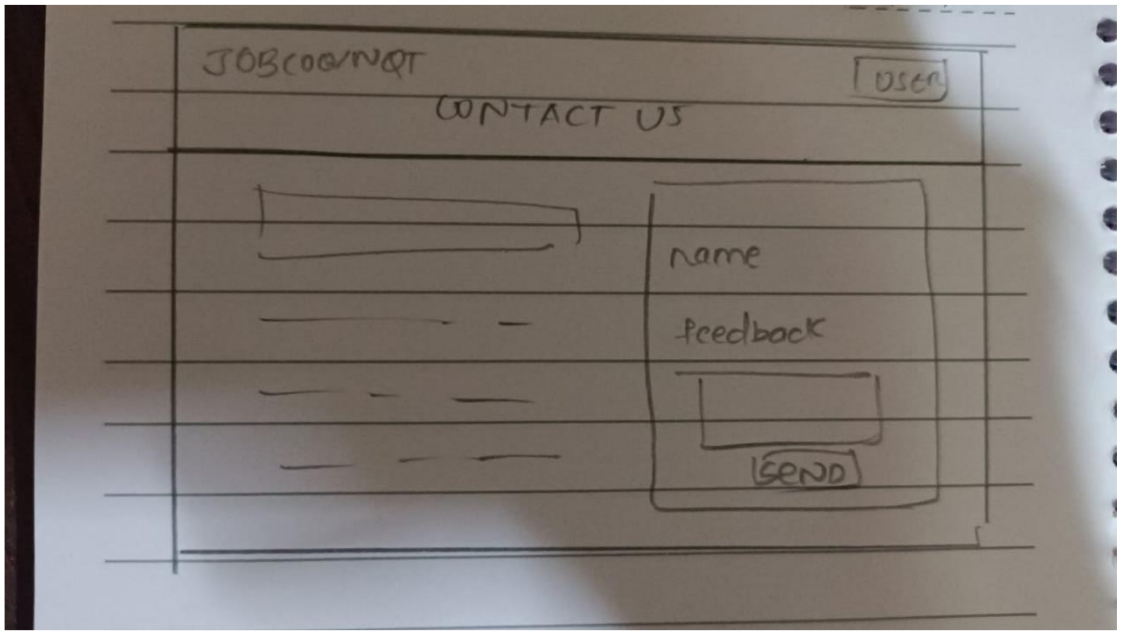



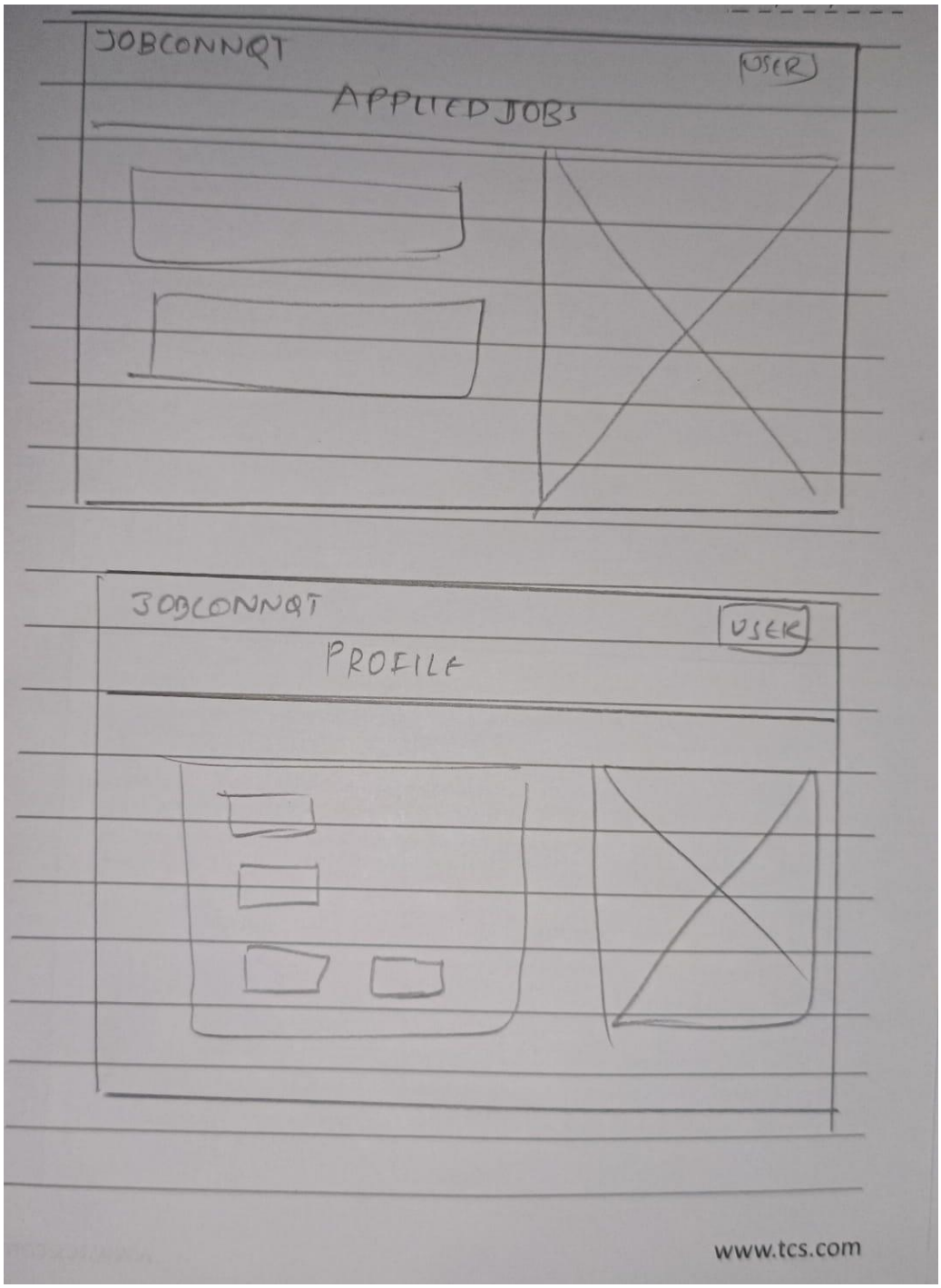

JobCONNQT
Home
EMPLOYER
ABOUT
LOGIN
SIGN UP
Find the job that you acquire
@Tejal - Job Portal


JobCONNQT
John
Job Description
Job title
Skills
Category
Job Type
Apply JOB
@ Copyrights


JobCONNQT
Welcome John
Dashboard
My listings
Jobs
Jobs
Details
Details
Details


LOGIN
NAME:
Password:
Email:
LOGIN
SIGNUP

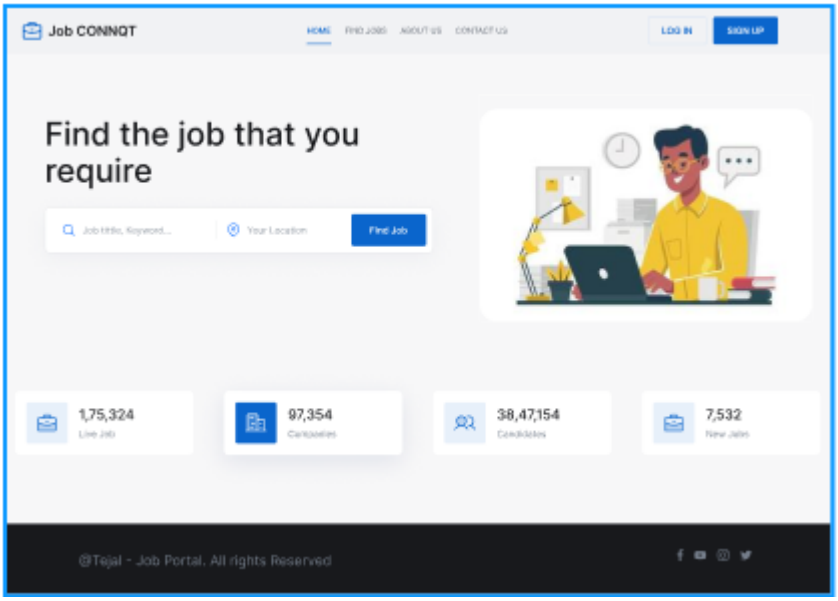

JOBCONNQT
USER
CONTACT US
Name
Feedback
SEND

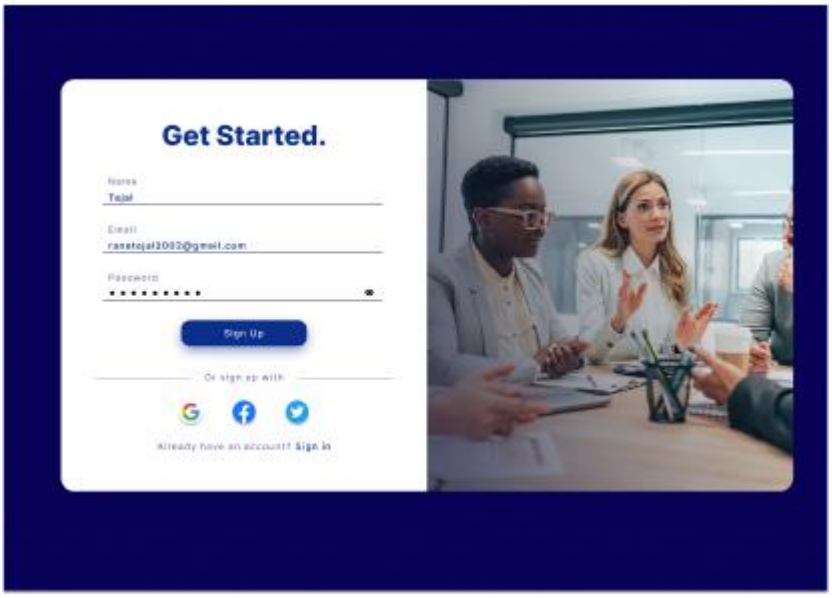



Figma Wireframe:

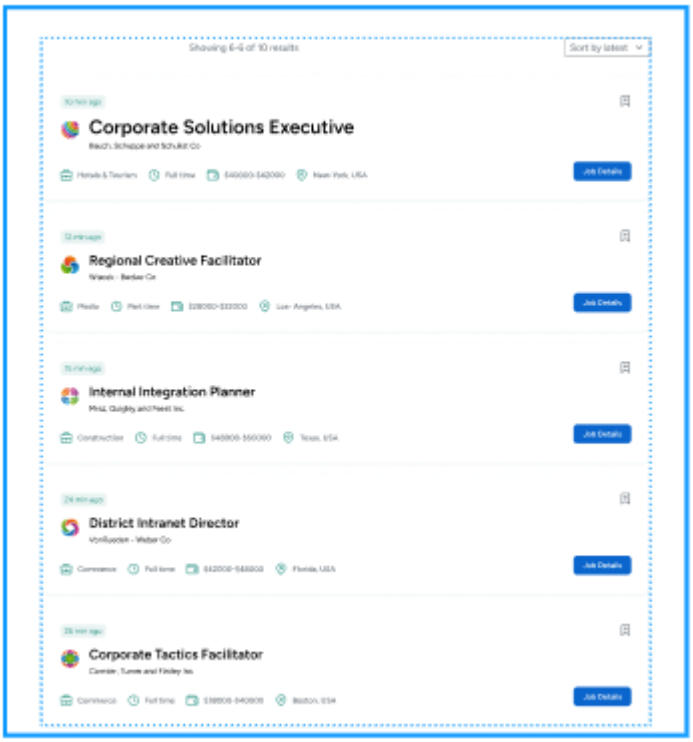

Desktop - 1
Frame 1
Side bar+ cars+ banner
Job Details


Desktop - 1
Frame 1
Desktop - 4
Frame
Side bar+ cars+ banner
Job Details
Desktop - 5
Desktop - 8
Desktop - 3
Desktop - 7
Desktop - 6

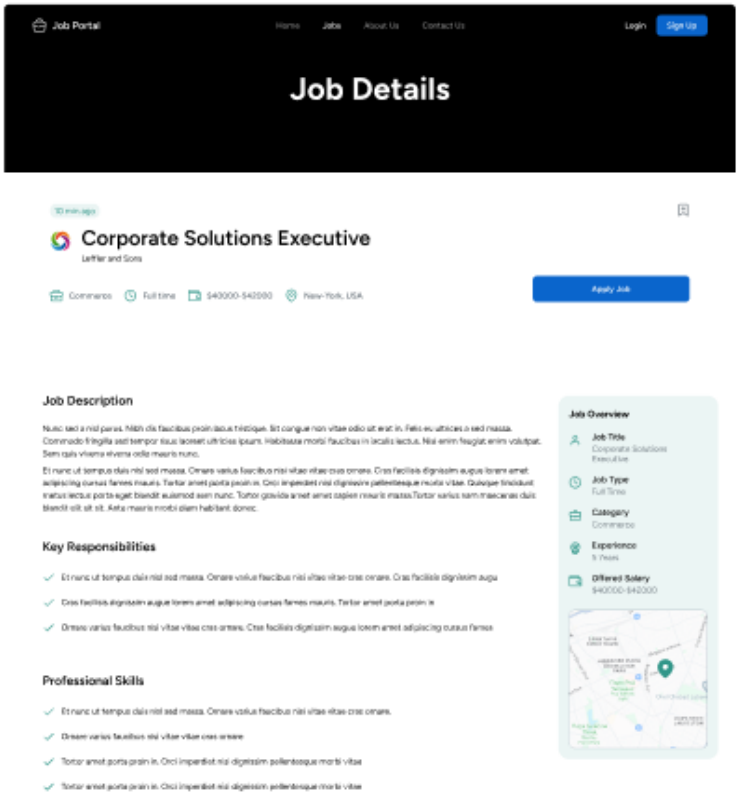

Job CONNQT
HOME
FIND JOBS
ABOUT US
CONTACT US
LOG IN
SIGN UP
Find the job that you require
Find Job
1,75,324
97,354
38,47,154
7,532
@Tejal - Job Portal. All rights Reserved

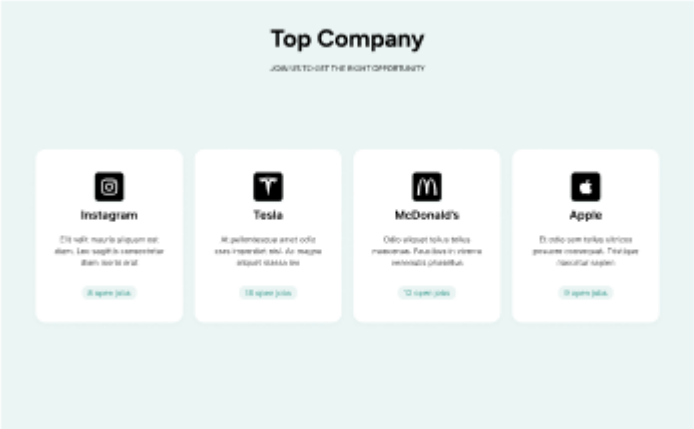

Get Started.
Name
Tejal
Email
Password
Sign Up
Or sign up with
Already have an account? Sign in

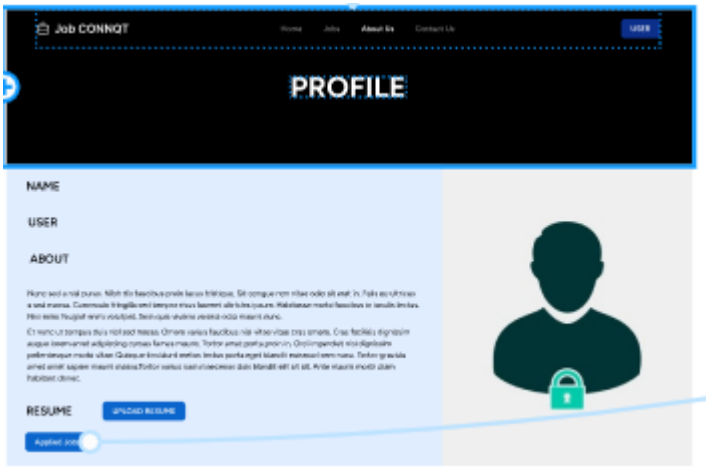
Side bar+ cars+ banner




Job Details

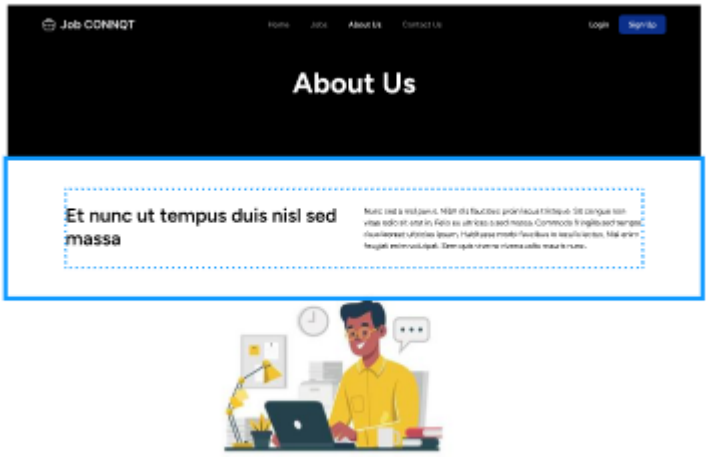



Top Company
Instagram
Tesla
McDonald's
Apple

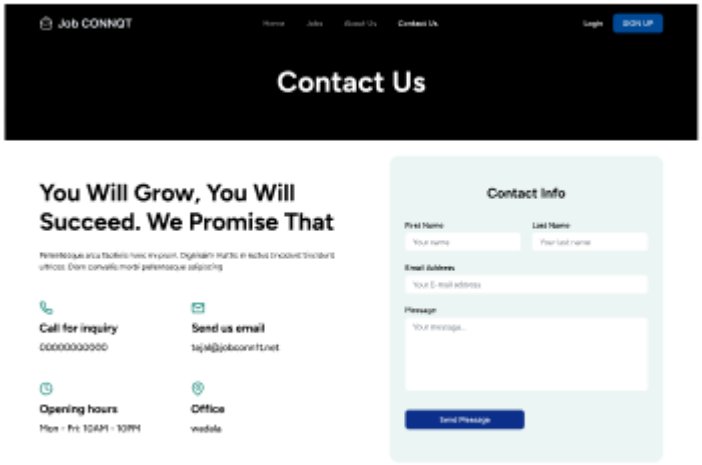

Job CONNQT
PROFILE
NAME
USER
ABOUT
RESUME


LOGIN

Desktop - 3

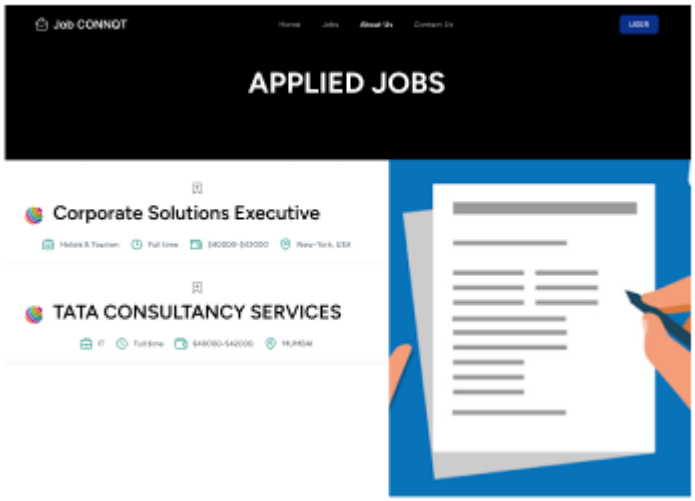



Desktop - 4



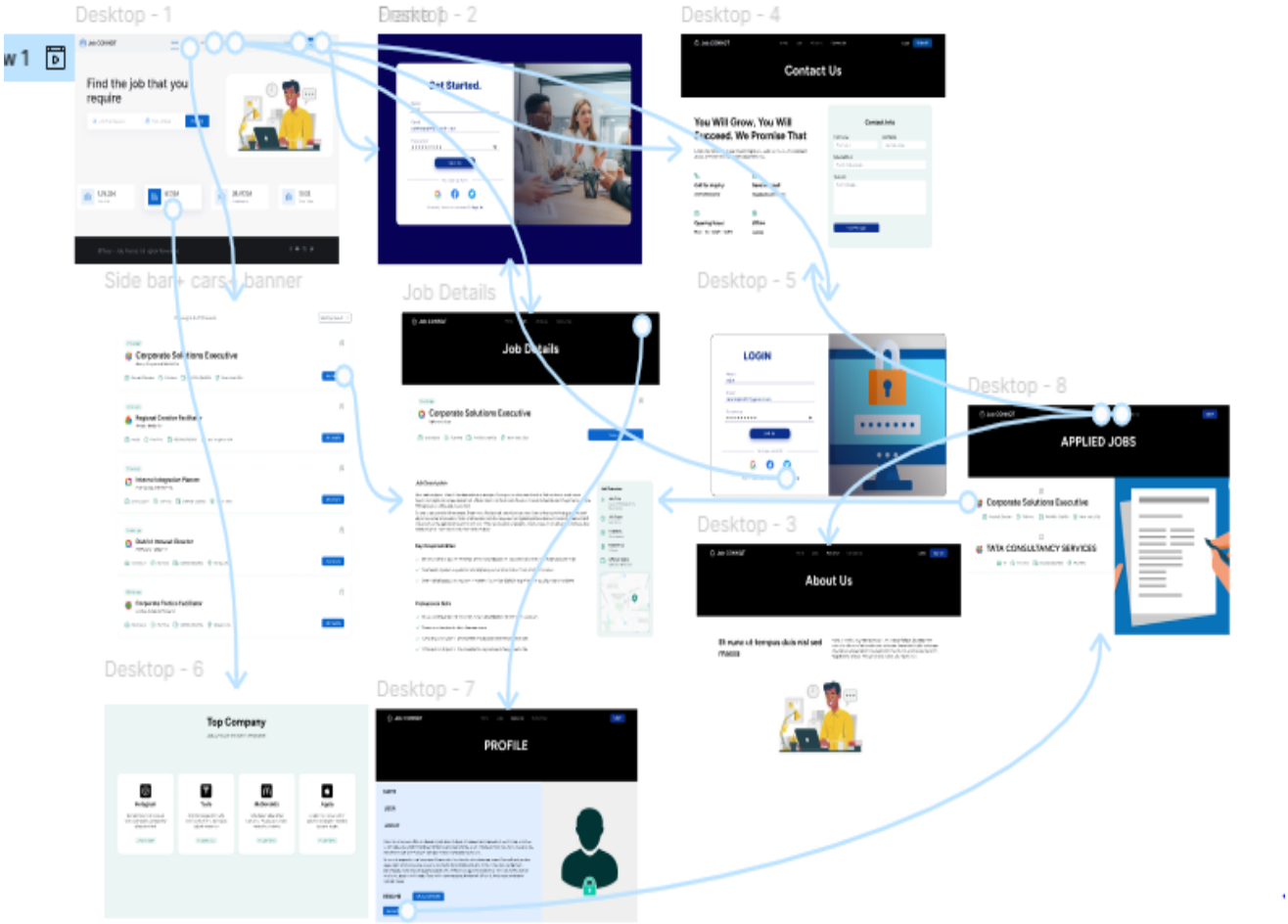

Desktop - 5

Desktop - 8



# PRACTICAL 10:

## CREATE PROTOTYPE FOR CHOSEN PROJECT.




https://www.figma.com/design/bG95EyfKkCFWwFvInAj7C1/JOB-PORTAL-FINAL?node-id=0-1&p=f&t=4c2Xo1wqnLqF52v1-0

# PRACTICAL 11:
# DESIGN CUSTOMER JOURNEY MAP.

## User: Mohan Singh

| Stage | Touchpoints | Actions | Emotions | Pain Points | Opportunities |
|---|---|---|---|---|---|
| **1. Awareness** | College portal, social media, friends | Hears about JobConnqt from college noticeboard & WhatsApp group | Curious | Doesn't know if it's better than other portals | Campus ambassador program, targeted social ads |
| **2. Consideration** | JobConnqt landing page, peer reviews | Visits the site/app, skims UI and available listings | Interested | Skeptical about job quality or relevance | Clear UX, student-focused branding, verified company badges |
| **3. Evaluation** | Search filters, job descriptions, resume upload | Uses filters for "Internship", adds resume, shortlists jobs | Hopeful | Confused about what skills match certain jobs | Resume tips, skill-tagged jobs, smart suggestions |
| **4. Application** | Apply button, confirmation popup | Applies to 2–3 relevant jobs | Motivated | Fear of no callback, repetitive forms | One-click apply, email confirmation, save profile feature |
| **5. Waiting/Tracking** | Dashboard, email, notifications | Checks dashboard for status, gets one interview call | Nervous | No updates from some companies | Real-time status updates, reminders for follow-up |
| **6. Interview/Outcome** | Email, offer letter | Attends interview, receives an internship offer | Excited | Nervousness about performance | Interview prep resources, past Q&A, mentorship call option |
| **7. Post-Placement** | Dashboard, success story email | Updates profile, shares success on social media | Proud | None — process was smooth | Request testimonials, share success, LinkedIn |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | badge integration |
| **8. Advocacy** | College events, peer groups, Instagram | Recommends JobConnqt to batchmates | Loyal | None | Referral rewards, ambassador badge, spotlight success stories |

## PRACTICAL 12:

## PERFORM UX EVALUATION OF CHOSEN PROJECT.

| Test Case ID | Description | Expected Result | Pass/Fail |
|---|---|---|---|
| **TC001** | Launch the JobConnqt portal | Homepage loads with options to login/signup and job categories visible | |
| **TC002** | Sign up as a student user | User account is created, and dashboard is accessible after email verification | |
| **TC003** | Login with valid credentials | Student is taken to personalized dashboard | |
| **TC004** | Search for "Web Developer" jobs | List of matching jobs is displayed | |
| **TC005** | Apply filter: Location = "Remote" | Job list narrows to remote roles only | |
| **TC006** | Click on a job card | Job detail page opens with role, requirements, and company info clearly visible | |
| **TC007** | Upload resume and click "Apply Now" | Resume is uploaded and application confirmation message is shown | |
| **TC008** | Go to "My Applications" section | Shows a list of applied jobs with their statuses (e.g., "Under Review") | |
| **TC009** | Attempt to apply without uploading resume | Error message appears: "Please upload your resume before applying." | |
| **TC010** | Access "Help & Support" from dashboard | Help options, FAQs, and contact details are visible | |