### Page 1 Scrum Model for Professional Search

## Sprint 1 - Week 1

## **©** Sprint Goal:

Set up the project environment and implement user authentication so users can register and log in.

# **✓** Epic 1: User Authentication

#### **User Stories:**

- As a user, I want to register with my email and password so I can create an account.
- 2. As a user, I want to log in securely so I can access my account.
- 3. As a user, I want to reset my password if I forget it.
- 4. As a user, I want to log in using Google or LinkedIn so I can sign in quickly.

#### **☆** Tasks:

- 1. Set up Git repository & CI/CD pipeline
- 2. Create Figma designs for Login & Registration pages
- 3. Initialize frontend (React) and backend (Node.js/Express)
- 4. Implement backend authentication API with JWT/OAuth
- 5. Implement frontend forms with validation
- 6. Integrate Google & LinkedIn OAuth APIs
- 7. Forgot Password email flow
- 8. Basic UI testing

## Sprint 2 – Week 2

## **©** Sprint Goal:

Build the **Home Page** and **Profile Management** so users can view the platform's offerings and manage their profiles.

### Page | 2 Scrum Model for Professional Search

# **☑** Epic 2: Home Page

#### **User Stories:**

- 1. As a user, I want to see trending professionals and categories so I can quickly find what I need.
- 2. As a user, I want a search bar so I can search for jobs or professionals.
- 3. As a visitor, I want to learn how the platform works so I can understand its benefits before signing up.

## **☑** Epic 3: Profile Management

#### **User Stories:**

- 1. As a freelancer, I want to create and update my profile so I can showcase my skills.
- 2. As an employer, I want to manage my company profile so jobseekers can know more about us.
- 3. As a user, I want to upload a profile picture and documents.

#### **☆** Tasks:

- 1. Design home page layout in Figma
- 2. Implement search bar (frontend + backend API)
- 3. Add dynamic trending list & categories (backend integration)
- 4. Add "How It Works" section (Figma → frontend)
- 5. Design profile management UI for both user types
- 6. Implement file/document upload feature
- 7. Create backend API for profile CRUD operations
- 8. Connect profile data with UI

## Sprint 3 – Week 3

### Page | 3 Scrum Model for Professional Search

# **③** Sprint Goal:

Enable **Job Posting, Matching, and Notifications** so employers can post jobs and freelancers get matched.

# **☑** Epic 4: Job Posting & Matching

#### **User Stories:**

- 1. As an employer, I want to post a job so freelancers can apply.
- 2. As a freelancer, I want to see matched jobs based on my skills.
- 3. As a user, I want to filter results so I can find relevant opportunities quickly.

## **∠** Epic 5: Notifications

#### **User Stories:**

- 1. As a user, I want to receive notifications for job matches, messages, and updates.
- 2. As a user, I want to mark notifications as read/unread.

#### **☆** Tasks:

- 1. Job posting form UI (React)
- 2. Backend API for job posting & retrieval
- 3. Implement job matching algorithm (skills-based)
- 4. Create filter options (frontend)
- 5. Implement notification backend (WebSocket or polling)
- 6. Create notification UI list (frontend)
- 7. Read/unread toggle feature
- 8. Testing job & notification features

## Sprint 4 – Week 4

# **③** Sprint Goal:

Add **Messaging**, **Settings**, **and Help** so users can communicate and get support.

### Page 14 Scrum Model for Professional Search

# ✓ Epic 6: Messaging System

#### **User Stories:**

- 1. As a user, I want to chat with employers/freelancers so we can discuss projects.
- 2. As a user, I want to send files through chat.

## ✓ Epic 7: Settings & Preferences

#### sUser Stories:

- As a user, I want to update my account details so I can keep my information accurate.
- 2. As a user, I want to control my privacy and notification settings.

## Epic 8: Help & Support

#### **User Stories:**

- 1. As a user, I want to search FAQs so I can find answers quickly.
- 2. As a user, I want to contact support if I have a problem.

#### **☆** Tasks:

- 1. Design chat UI in Figma
- 2. Implement real-time messaging with WebSockets
- 3. Enable file sharing in chat
- 4. Build settings page UI & backend integration
- 5. Add notification preference controls
- 6. Help page UI with searchable FAQs
- 7. Contact form integration
- 8. Deploy & final bug fixes