PROJECT SYNOPSIS

ON

RozgarSetu

SUBMITTED

TO

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FOR

Full Stack Engineering (22CS037)

Submitted By:

Nitin Choudhary (2210991999), Nitin Kumar (2210992001),

Omil Goel (2210992007), Piyush Goyal (2210992044), Prince Kumar (2210992091)

Semester: 6’th

Session: 2022-2026

**Submitted To: Mr. Rahul Rajput**

# Index

Sr. no Topic Page No

1. Problem Statement 3
2. Title of project 3
3. Objective & Key Learning’s 3
4. Options available to execute the project 3-4
5. Advantages/ Disadvantages 4
6. References 5

1. Problem Statement:

In today’s digital age, many individuals—especially daily wage workers, small-scale vendors, and freelancers—struggle to find consistent employment opportunities due to lack of accessible, trustworthy platforms. Traditional job portals often fail to cater to this unorganized sector. There's also a disconnect between people seeking services and those offering them locally. The project **RozgarSetu** aims to bridge this gap by providing a secure, feature-rich, and community-driven platform that connects job seekers and employers, facilitates skill discovery, manages profiles, and supports transparent communication—all while promoting local employment and empowerment.

1. Title of project: RozgarSetu

Description: A Full-Stack Employment & Service Connector Platform

1. Objective & Key Learnings:

The objective of RozgarSetu is to build a digital platform where users can register, post or find job opportunities, interact with service providers, manage referrals, and view detailed work history—all within a community-focused ecosystem. It supports real-time updates, secure authentication, and personalized user experiences.

Key Learnings:

* Implementing user authentication using Passport.js and Mongoose.
* Designing a real-time job and services feed.
* Handling file uploads for user profiles (Multer).
* Developing scalable routes for various user functionalities.
* Building interactive dashboards: referrals, transactions, orders, and investments.
* Real-time updates on posts, comments, and job requests.
* Creating a privacy-respecting and intuitive profile system.
* Using MongoDB for efficient data structuring and storage.

1. Options available to execute the project:

 Web**-Based Application (Node.js + Express + MongoDB + EJS):**

* Cross-platform access via browser.
* Dynamic content rendering and route handling.
* Supports secure user sessions.

 Mobile **App Extension (Future Scope – Flutter/React Native):**

* Increased accessibility and mobility.
* Push notifications for job alerts and updates.

 Cloud **Deployment (MongoDB Atlas, Render, or Fly.io):**

* Scalable backend infrastructure.
* Persistent cloud-hosted database and media storage.

 Modular **Architecture:**

* Routes for job management, orders, holdings, referrals, etc.
* Separate files for user image uploads and post handling.

1. Advantages/ Disadvantages:

Advantages:

* + **Connects job seekers with verified employers and clients.**
  + **Real-time interaction through posts, comments, and messages.**
  + **Referrals and offers boost engagement and visibility.**
  + **Encourages local hiring and self-employment.**
  + **Profile-based trust-building and transparency.**
  + **Secure data handling and login mechanisms.**

Disadvantages:

* + Requires digital literacy among users.
  + Initial traffic may be low without outreach or marketing.
  + Moderation and trust verification are necessary.
  + May need mobile app for broader adoption.
  + Real-time updates require robust backend handling.

# REFERENCES

* **Node.js:** [Official Documentation](https://nodejs.org/docs/latest/api/)
* **Express.js:** [Documentation](https://expressjs.com/)
* **MongoDB:** [Basics](https://docs.mongodb.com/manual/)
* **EJS:** [Documentation](https://www.ejs.co/)
* **GitHub Actions:** [Documentation](https://docs.github.com/en/actions)
* **React:** [Documentation](https://react.dev/)