A Mini Project Synopsis on

Freelancing Platform with Recommendations

T.E. – Computer Science and Engineering-Data Science

Submitted By

Rohan Waghode 21107008

Varad Joshi 21107002

Meet Jamsutkar 22207004

Aman Dhumal 21107036

Under The Guidance Of

Prof. Anagha Aher

DEPARTMENT OF CSE-DATA SCIENCE

A.P.SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic Year : 2022-23

CERTIFICATE

This to certify that the Mini Project report on Freelancing platform using Recommendations has been submitted by Rohan Waghode(21107008), Varad Joshi(21107002), Meet Jamsutkar(22207004) and Aman Dhumal(21107036) who are a Bonafede students of A. P. Shah Institute of Technology, Thane, Mumbai, as a partial fulfilment of the requirement for the degree in Computer Science and Engineering(Data Science), during the academic year 2023-2024 in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Prof. Anagha Aher
Guide

Prof. Anagha Aher
Head Department of CSE-Data Science

Dr. Uttam D.Kolekar
Principal

External Examiner(s)
1.
2.

Place: A. P. Shah Institute of Technology, Thane
Date:

TABLE OF CONTENTS

Chapter 1

Introduction

Welcome to the Freelancing web app Project, a visionary initiative that promises to revolutionize the landscape of online freelancing. At its heart, this project is dedicated to simplifying and enhancing the way freelancers and clients collaborate in the digital age. In a world where remote work and freelancing have become increasingly prevalent, this platform aims to bridge the gap between talented individuals seeking work and businesses in need of their skills.

Our core mission is to facilitate seamless connections between freelancers and clients, creating a dynamic and thriving marketplace that empowers both parties. We recognize the potential of the freelance economy and its ability to bring opportunities to a global scale. With the Freelancing web app Project, we aim to maximize the potential of this workforce, enabling freelancers to showcase their talents and allowing businesses to access a diverse pool of talent.

One of our primary goals is to streamline the process of finding, hiring, and managing freelancers. We aim to offer a user-friendly platform that makes it easy for clients to browse profiles, assess skills, and select the perfect fit for their projects. Simultaneously, freelancers will gain a platform to showcase their expertise and connect with clients who appreciate their skills. Furthermore, our scope extends beyond mere matchmaking. We envision a comprehensive ecosystem that encompasses project management tools, secure payment systems, and robust communication channels, ensuring that collaborations are efficient and transparent. In essence, the Freelancing web app Project is about transforming the world of work by providing a space where freelancers and businesses can not only connect but also collaborate, communicate, and manage their projects with utmost ease. This project's promise lies in the potential to unlock a new era of work, enabling freelancers and businesses to thrive in an interconnected digital world.

1.1 Purpose

The core essence of the project lies in its ambition to establish a digital haven where talented freelancers and eager clients converge. It's a visionary platform meticulously designed to act as a dynamic bridge connecting those in need of assistance with individuals willing to lend their expertise. For freelancers, this means access to a vast array of exciting work opportunities across a multitude of industries. They can showcase their skills, experience, and passion, paving

the way for a fulfilling career in the freelance world. On the other side, clients are presented with a user-friendly environment to easily identify the perfect individuals for their projects. It's a space where businesses can access a global pool of talent, making their search for the right skills seamless and efficient. In a fast-evolving digital age, this project embodies accessibility, efficiency, and a global reach. It transcends geographical boundaries, bringing talent and opportunities together like never before. It's about unlocking the potential of the freelance economy and creating a platform that empowers both freelancers and clients to prosper in a rapidly changing world of work.

1.2 Objectives

The objectives of our project underscore our commitment to delivering a top-tier freelancing platform that caters to the needs and aspirations of both freelancers and clients. Let's delve deeper into each of these key objectives:

1. Enhance Job Matching Accuracy:

   This objective aims to improve the precision and relevance of job matches for job seekers. To achieve this:

   - Implement used cosine similarity algorithm and interaction history models to analyze job postings and match them with the skills and qualifications of job seekers.
   - Collect and use feedback from users to continuously refine the matching process as per the user interests.

2. Provide Personalized Job Recommendations:

   Personalized job recommendations increase the likelihood of job seekers finding suitable positions. This objective involves:

   - Gathering user data and preferences, such as location, job history, and career goals.
   - Employing recommendation algorithms to suggest jobs that closely align with the user's profile.
   - Offering options for users to refine their preferences and criteria for job recommendations.

3. Create a User-Friendly Freelancing Platform:

   A user-friendly platform can attract and retain both freelancers and clients. To accomplish this objective:

   - Develop an intuitive and visually appealing user interface.
   - Prioritize responsive design for users.
   - Provide comprehensive onboarding and support materials to help users navigate the platform.

4. Promote Skill Development:

   This objective focuses on helping users improve their skills and qualifications. Actions may include:

   - Facilitating access to mentorship and networking opportunities.

5. Ensure Data Security:

   Data security is paramount to protect user information and build trust. Key elements include:

   - Implementing strong encryption and access control measures to safeguard personal and sensitive data.
   - Regularly auditing and assessing the platform's security for vulnerabilities.
   - Complying with relevant data protection regulations and ensuring transparent privacy policies.

6. Implement Payment Integration:

   Integrating payment capabilities is crucial for a freelancing platform. This objective involves:

   - Ensuring a seamless and secure payment process for clients and freelancers using Razorpay payment gateway.
   - Allowing the freelancer to set the amount for the services and get paid for the service provided.

1.3 Scope

Out Project Includes, a comprehensive solution designed to empower freelancers and clients alike. We understand the challenges of freelancing and have tailored our platform to address them, providing a seamless experience for both parties. Whether you're a freelancer showcasing your skills or a client looking for the perfect match for your project, we've got you covered. Our platform offers a range of features to facilitate your journey:

- User Profiles: Create a profile that shows off your skills, experience, and past work.
- Job Listings: Post jobs with all the details you need to find the perfect match.
- Smart Recommendations: We'll suggest jobs to freelancers based on what they're great at and what they've done before.
- Easy Communication: Talk directly with clients or freelancers right on our platform.
- Payments Made Simple: We're integrating secure payment options so you can focus on the work, not the hassle.
- Learn and Grow: Find courses and tutorials to help you improve your skills.
- Community and Networking: Connect with others in the freelancing community for advice and inspiration.
- Privacy and Security: Your data is safe with us.
- User-Friendly Everywhere: Access our platform easily from your computer, phone, or tablet.
- Transparency: We're all about making sure you know what's going on with your projects.

Chapter 2

Problem Definition

The freelancing landscape is undoubtedly a dynamic and increasingly prominent segment of the modern workforce. However, the existing inefficiencies and challenges have become roadblocks that hinder its optimal functioning. One of the primary issues is the fragmentation of the freelancing market. This fragmentation is partly due to the proliferation of freelance platforms, each with its own unique set of rules, regulations, and requirements. For both freelancers and clients, navigating this vast sea of options can be incredibly time-consuming and overwhelming. This not only hampers the ability to find the right match but also results in missed opportunities for both parties.

Furthermore, communication within the ecosystem often relies on external tools such as email, messaging apps, or project management software. This dependence on various external platforms leads to fragmented and inefficient collaboration. It can also hinder the establishment of trust, as there is no standardized way to evaluate a freelancer's capabilities and track record. Payment management and invoicing processes are also convoluted and lack transparency. This opacity in financial transactions can result in payment disputes, leading to dissatisfaction among freelancers and clients. Project management issues, including disorganization, miscommunication, and disputes, are commonplace due to the lack of a unified platform for project management.

Additionally, the dearth of comprehensive data and analytics further compounds these problems. Without access to meaningful insights, freelancers and clients are left in the dark when it comes to making informed decisions about pricing, project viability, and market trends. Addressing these issues through the development of an integrated platform is not just a desirable goal but a necessary one. Such a platform could streamline the process of finding freelance talent, simplify communication and collaboration, establish standardized evaluation mechanisms, and provide transparent payment processes. Ultimately, this would lead to reduced project delays, cost savings, and a vastly improved freelancing experience for all parties involved, thus fostering growth and innovation in the freelancing landscape.

Chapter 3

Proposed System

The proposed management system for your freelancing platform is a comprehensive solution designed to enhance user experience and efficiency. It incorporates a recommendation system based on collaborative filtering, implementing cosine similarity. This advanced algorithm analyzes user behavior and preferences to provide personalized recommendations, enhancing the platform's usability and relevance for each individual user.

The system also includes a real-time messaging feature using WebSockets, enabling direct and instant communication between clients and freelancers. This feature facilitates clear understanding of project requirements and expectations, fostering effective collaboration. Moreover, the system integrates third-party OAuth for user authentication. This feature allows users to conveniently register and log in using their existing social media accounts, enhancing the platform's accessibility and user-friendliness. This proposed management system leverages advanced technologies and algorithms to create a user-centric, efficient, and secure environment for freelancing.

3.1 System Architecture and Working:

In this chapter, we will delve into the core components and mechanisms of a recommendation system. We will explore how user interests are dynamically generated and updated through the Score Generator and how user feedback is incorporated to enhance accuracy. Additionally, we will uncover the intricate process of recommendation generation, which leverages both collaborative filtering and content-based filtering for optimal results.

The Score Generator is a critical component of the recommendation system that dynamically generates user interests based on their interactions with the system. The Score Generator uses a combination of explicit and implicit feedback to generate a score for each item in the system. Explicit feedback is provided by users when they rate or review an item, while implicit feedback is inferred from user behaviour such as clicks, views, and purchases.

User feedback is incorporated into the system to enhance accuracy. The system uses a variety of techniques to incorporate feedback, including collaborative filtering and content-based filtering. Collaborative filtering uses similarities between users and items simultaneously to provide recommendations. This allows for serendipitous recommendations; that is, collaborative filtering models can recommend an item to user A based on the interests of a

similar user B. Content-based filtering, on the other hand, recommends items based on their similarity to items that the user has already interacted with.

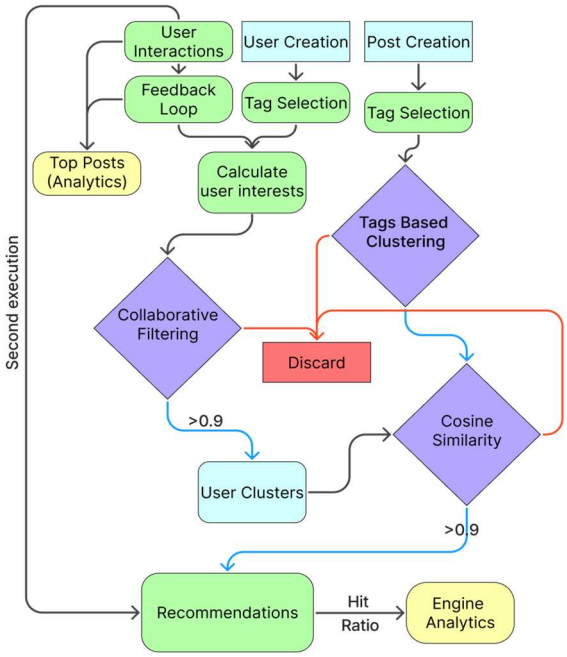

Figure 1: Tag Based Recommendation System

The implementation works on a custom-built algorithm increasing user retention by a considerable margin. For convenience's sake, we divide the engine into three distinct parts:

1. Interest Generation:

The Engine leverages user interactions in tandem with existing user interests to keep creating and updating interests. The user interests would consist of the tags user is interested in, and the score of each tag, i.e. the score of how much a user is interested in a particular tag. The user interactions and user interests are compared using cosine similarity to generate new User interests. These User interests come into effect later into the recommendation part.

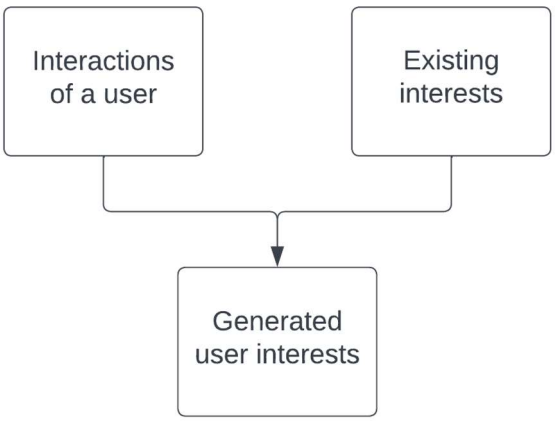

Figure 2: Generation of User interests

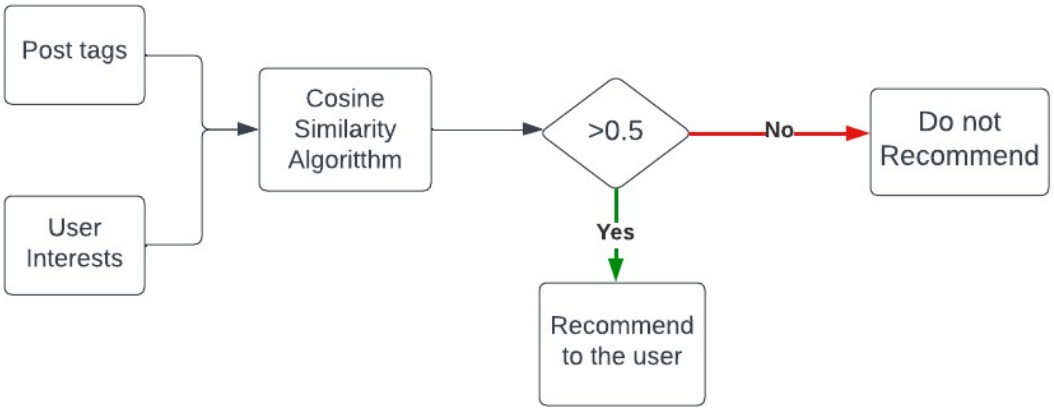

Figure 3: Generation of Recommendations

2. Recommendation Generation:

The aforementioned User interests are then compared with Post tags(tags that a certain post has) using cosine similarity to obtain a score that is between 0 and 1. In this process every single user interest is compared with every single post tag and posts with post tags with a score greater than 0.5 are recommended to the User.

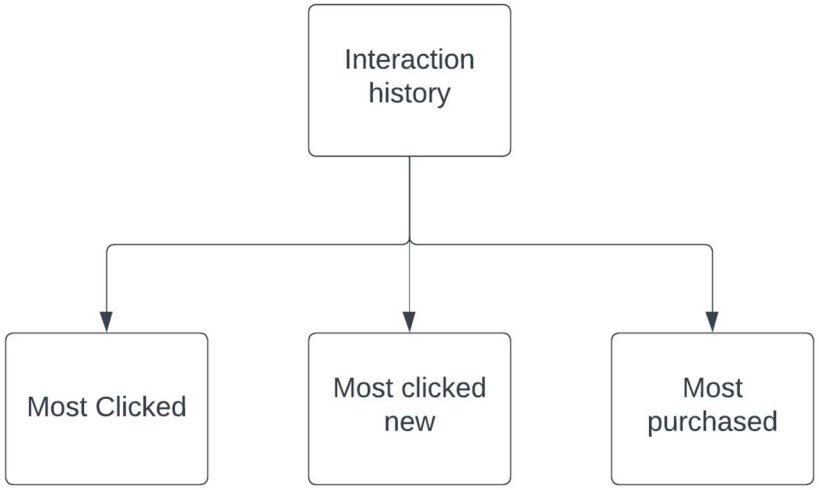

Figure 4: Generating Top Charts/ Analytics

3. Top charts:

This implements user interactions to keep tabs on how many times a post is clicked on and/or purchased to rank the post to create a view that may show the most popular posts on the platform.

We will now take an in-depth overview of how each component works together.

a. Post:

   A Post is a piece of content that will be recommended to the User based on the decision made by the Recommendation Engine.

   post_tags={postid=post1: tag1=yes, tag2=no, tag3=no ... tagn=yes}

   While processing, all "yes" values become 10 and no become 0 for cosine similarity calculation.

b. Tags:

   Tags are the attributes given to our Posts which help us establish an analogy between a post and the User. Each post can have multiple tags describing itself. Tags can either be set by reviewers, post author or automatically by using NLP based on different use cases. In our testing, the Post Author gets to set its Tags.

c. User Interests:

   User interests track the interest of a user in different categories of Posts by assigning a value, hereon referred to as Score, to each Tag. This is generated by the "Score Generator" part of the recommendation engine and serves as a basis for generating Recommendations.

   user_interests={user_id=u1: tag1=score, tag2=score, ... tagn=score}

   where the value of score ranges between 0 to 10.

d. User Interactions:

   User Interactions refer to the various things a user might do on the platform. In our case, we are using a simple implementation which records the timestamp, post id, user id and action. Based on what action is being done, there is provision in the recommendation engine to provide more or less weightage to the interest of the user in that post. Interactions are also our main basis for calculating Analytics, both Post analytics as

well as Engine analytics. User interactions are also used by the Score generator to generate User Interests. For more advanced analytics, we can add more user data such as region, language, gender, age group and get analytics based on those factors.

user_interactions={userid=u1, postid=post1, event:click}

e. Score Generator:

The score generator is the first part of our recommendation engine which generates the User Interests which can then be compared with Post Tags in order to decide whether to recommend a post or not. Following is the example of how the score generator generates user interests.

Iterate over users:

Retrieve existing User Interests from the database and store them in array A.

Let us assume some values for array A.

A = {0, 1, 5, 3, 10} ------- *(3.5.1)*

Next, we check the database for user interactions and retrieve the tags of every post the user has interacted with since the last score generation. Let us assume this data is saved in a two-dimensional array B where the rows represent Posts and columns represent tags. Let us assume data for array B

B = {0, 10, 0, 10, 0

10, 0, 10, 0, 10

0, 0, 10, 0, 0

10, 10, 0, 0, 0

10, 0, 10, 0, 0} ------- *(3.5.2)*

Now, the score generator takes summation of all rows and stores it in an array C.

C = {30, 20, 30, 10, 10} ------- *(3.5.3)*

Array C represents the latest interests of the User. The array C is now normalized to 10 by dividing throughout by 3.

C = {10, 6.67, 10, 3.34, 3.34} ------- *(3.5.4)*

Now, we have array A representing the user's historical interests and array C representing the user's latest interests. We average both the arrays, and the resulting data is updated into the database. In this case, let us assume the result is array D.

D = {5, 8.34, 7.5, 3.17, 1.67} ------- *(3.5.6)*

Rounding to nearest integer,

D = {5, 8, 7, 3, 2} ------- *(3.5.7)*

This way, the scores never become stagnated, and the user never gets recommended something that he is no longer interested in.

f. Feedback Loop:

Our primary way of generating latest user interests is by aggregating past user interests and their latest interactions. However, sometimes a stray tag might get too weighted while the user isn't really interested in it. This is why we introduced a Feedback Loop which the user can use for letting the system know that they are not interested in a particular post/tag. These posts/tags are negated while calculating user interests by the score generator. Doing this improves the accuracy of User Interests, thereby reducing the error factor while generating recommendations and enhancing prediction quality.

g. Recommendation Generation:

For generating recommendations, we use the cosine similarity algorithm along with clustering of Users as well as Posts, thus making it a hybrid system. A hybrid system is the one that implements both, collaborative filtering as well as content-based filtering.

Cosine Similarity: Cosine similarity is a metric used to measure the similarity between two vectors. We can use Post Tags and User Interests as the dimensions for 2 distinct vectors and compare them using cosine similarity, as follows.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Figure 3.7.1: Cosine Similarity Formula

Let array A represent User Interests and array B represent Post Tags. Assuming data,

A = {0, 10, 10, 0, 10, 0, 0, 10} ------- *(5)*

B = {0, 4, 6, 1, 4, 0, 3, 10} ------- *(6)*

By formula, we first need to find the dot product between A and B vectors as well as the magnitude of A and B.

A.B = 240

|A| = 20

|B| = 13.341

Now, *cosine similarity = 240/(20*13.341) = 0.8988*

Based on the value of cosine similarity (0.8988), we can assume that vectors A and B are similar. The closer the value of cosine similarity is to 1, the better fit it is.

Collaborative Filtering: Collaborative filtering is a method which analyses the history of Users with similar interests and recommends them the same Posts. In our implementation, we use collaborative filtering to divide Users into categories by using cosine similarity to find users with similar interests.

Tag Based Clustering: Tag-based clustering is a data analysis technique used to group or categorize Posts on the basis of Tags. We already have categories in place and hence do not need to do any additional processing for this.

Using the methods given above, we obtain User clusters and Categories. In order to generate recommendations, the first step is to generate scores for Categories using the same score generator used for users. Once obtained, we can compare User clusters with Categories using cosine similarity and rank the outcomes on the basis of the outcome of cosine similarity. These recommendations can be served to all the users from a cluster.

3.2 Features and Functionalities:

Personalized Experience: This feature records all user interactions. The data is then used by a Recommendation Engine to tailor the user experience. This could mean suggesting services or freelancers that the user might be interested in based on their past behavior.

Top Charts: This displays various top services, categories, and freelancers based on statistics. It's a great way for users to see what's popular or trending in the community.

Email Alerts: This sends notifications to users via email. The alerts could be about new messages, project updates, or other important information.

Ease of communication: It allows for real-time communication between users on the platform. It uses WebSockets, a protocol that provides full-duplex communication channels over a single TCP connection. This means users can send and receive messages in real-time, improving the responsiveness and user experience of the platform.

Password less authentication: Using allauth based single sign on(SSO) password less authentication to make login easier, providing SSO using google OAuth.

Payment Integration: Implementation of a razor pay payment gateway to allow in app purchases.

3.3 Advantages:

The proposed system for your freelancing web application offers numerous advantages:

- It promises improved efficiency through automation. This could mean automating the process of matching freelancers with suitable projects, thereby saving time and effort.

- The system provides a centralized platform where freelancers and clients can collaborate effectively. This could involve real-time communication, project tracking, and more.
- The user experience is enhanced by transparency and user-friendliness. For instance, the system could provide clear information about project requirements, payment terms, etc.
- Powerful reporting and analytics tools are offered that not only help reduce operational costs but also provide valuable insights. These insights could be used to further improve the platform or to gain a competitive edge in the market.
- Importantly, the system is highly scalable, making it suitable for handling a large number of users and projects without compromising on performance or user experience. This makes it suitable for both small businesses and large organizations.

Chapter 4

Project Outcomes

The implementation of the proposed Freelancing Platform is expected to yield several significant outcomes, benefiting both freelancers and clients alike. These outcomes encompass enhanced efficiency, improved quality of collaborations, streamlined communication, and more. Here's a breakdown of the anticipated project outcomes:

1. **Enhanced Efficiency**: By automating time-consuming tasks, the platform will empower users to dedicate more time to strategic activities, ultimately reducing the time required to find the right freelancers or projects.

2. **Better-Quality Matches**: Leveraging advanced filtering and screening capabilities, the platform will excel at identifying the most qualified freelancers for specific projects. This will result in better-quality matches, ensuring that clients and freelancers are well-suited for their collaborations.

3. **Improved Freelancer and Client Experience**: The platform will offer streamlined communication channels and easy access to project information. This will lead to an improved experience for both freelancers and clients, fostering smoother and more productive collaborations.

4. **Cost Savings**: Through the automation and centralization of freelancing processes, organizations can anticipate reduced operational costs. This, in turn, will enhance return on investment (ROI) for both freelancers and clients, making freelancing more cost-effective.

5. **Data-Driven Decision-Making**: The platform's robust reporting and analytics capabilities will provide valuable insights into freelancing processes. This data-driven approach will enable users to make informed decisions and continuously refine their strategies.

6. **Competitive Advantage**: Organizations that leverage the Freelancing Platform will gain a competitive edge by enhancing their freelancing processes and attracting top-tier talent in the market. This advantage will position them as leaders in their respective industries, making them a preferred choice for both freelancers and clients

Chapter 5

Software Requirements

The success of the Freelancing web app Project hinges on its software requirements, which play a pivotal role in ensuring seamless functionality and a user-friendly experience. Let's delve into the software requirements for this visionary initiative based on the software stack. The project's software should be compatible with Windows and Windows 11 to ensure accessibility for a broad range of users. It's crucial to develop a platform that runs smoothly on these popular operating systems, accommodating both freelancers and clients.

The web app's frontend should be designed using HTML and HTML5 to create a modern, responsive, and interactive user interface. HTML5's advanced features can enhance the user experience, making the platform more engaging and intuitive. BULMA CSS is a popular CSS framework that can be utilized to ensure a visually appealing and consistent design across the platform. Its responsive grid system and extensive styling options can be leveraged to create a visually appealing and user-friendly web app. Redis, as an in-memory data store, can be instrumental in improving the platform's performance. It can be used for caching frequently accessed data, thus reducing the load on the backend server, and ensuring swift response times. Redis can also aid in real-time data processing and messaging.

The development ensures compatibility with Visual Studio Code to facilitate efficient coding, debugging, and version control, making it easier to manage the project's source code. Django is a robust web framework for Python, known for its security, scalability, and extensive library of reusable components. It should be utilized to build the backend of the platform. Django provides a reliable foundation for user authentication, data management, and the implementation of various features like user profiles, messaging systems, and project management.

In the context of the Freelancing web app Project, the software stack serves as the backbone for realizing the project's vision. Here's how these software requirements contribute to the project's success:

Compatibility: Ensuring compatibility with Windows and Windows 11 broadens the user base, enabling freelancers and clients to access the platform regardless of their operating system.

User-Friendly Frontend: HTML and HTML5, combined with BULMA CSS, contribute to an intuitive and visually appealing frontend. A clean and responsive design is essential for providing a positive user experience.

Performance Optimization: Redis 7.2.0 is employed to optimize the platform's performance. It enhances responsiveness by caching frequently used data and can handle real-time communications efficiently.

Development Efficiency: Visual Studio Code 17.7 as the IDE simplifies the development process, allowing the development team to write, debug, and manage code effectively.

Robust Backend: Django 4.2.4, a powerful web framework, serves as the foundation for the backend. It ensures security, scalability, and facilitates the implementation of essential features like user management, project tracking, and communication tools.

Components	Sub-component	Specification
Operating System	Windows	Windows 11
Languages	HTML	HTML 5
	Python	Pyhton 3/
	BULMA CSS	BULMA CSS 0.9.4
Backend	Redis	Redis 7.2.0
IDE	Visual Studio Code	Visual Studio Code 17.7
Libraries	Django	Django 4.2.4

Table 1: System Requirements

Chapter 6

Project Design

Software design for the AI Fitness Trainer involves the systematic process of defining the software's structure and behavior, ensuring it aligns with user requirements. This high-level design creates a blueprint for translating abstract requirements into specific data and functionalities. The interface design encompasses how the software communicates internally, with external systems, and with user behaviors, ensuring seamless interactions and user-friendly experiences.

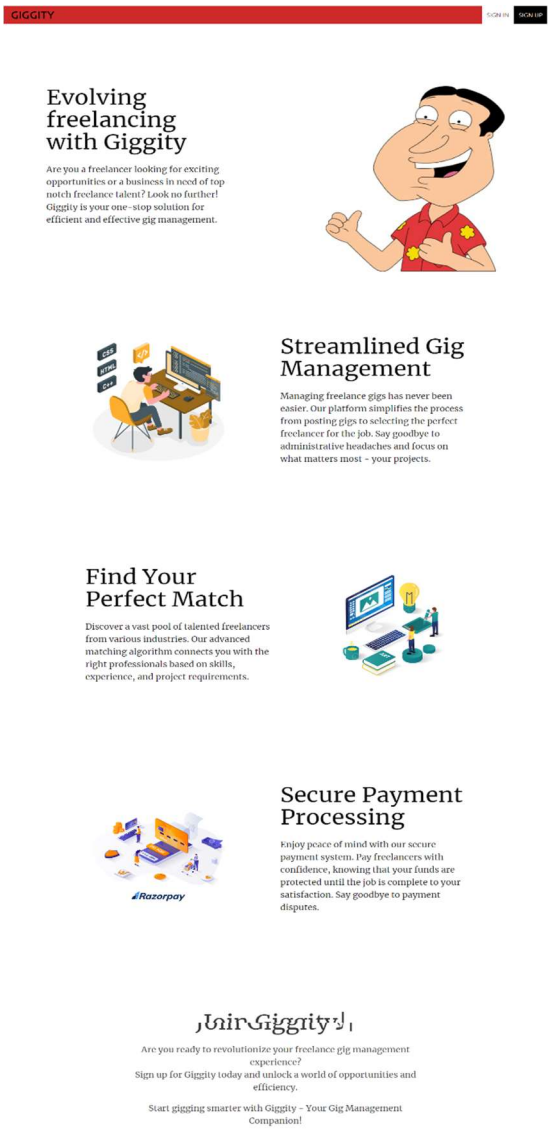

Figure 5: Freelancing Platform – Landing Page

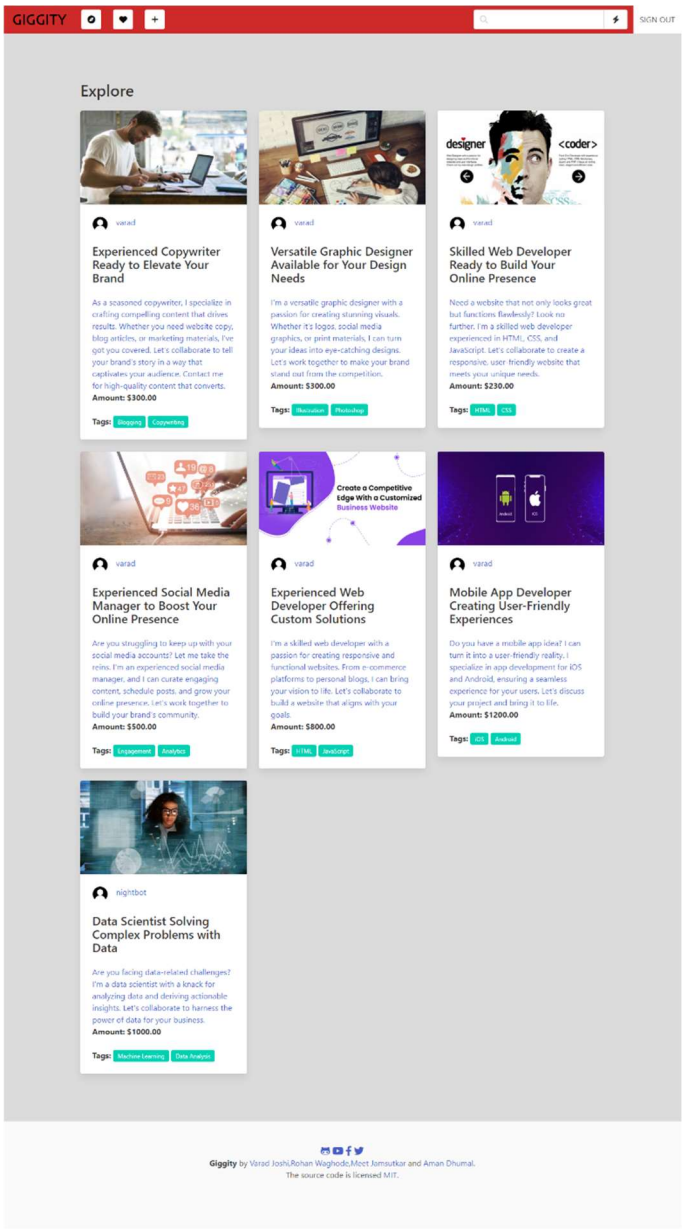

Figure 6: Freelancing Platform – Explore

The freelancing platform's landing page boasts a modern, user-friendly design with a dynamic search bar, enticing users to explore diverse job categories, secure payment options, and global access. It swiftly engages visitors, urging them to join the community. On the "Explore" page, detailed service profiles, project descriptions, and transparent pricing information offer insights for potential collaborations. This page builds trust and encourages seamless connections between freelancers and employers, fostering engagement within the vast marketplace.

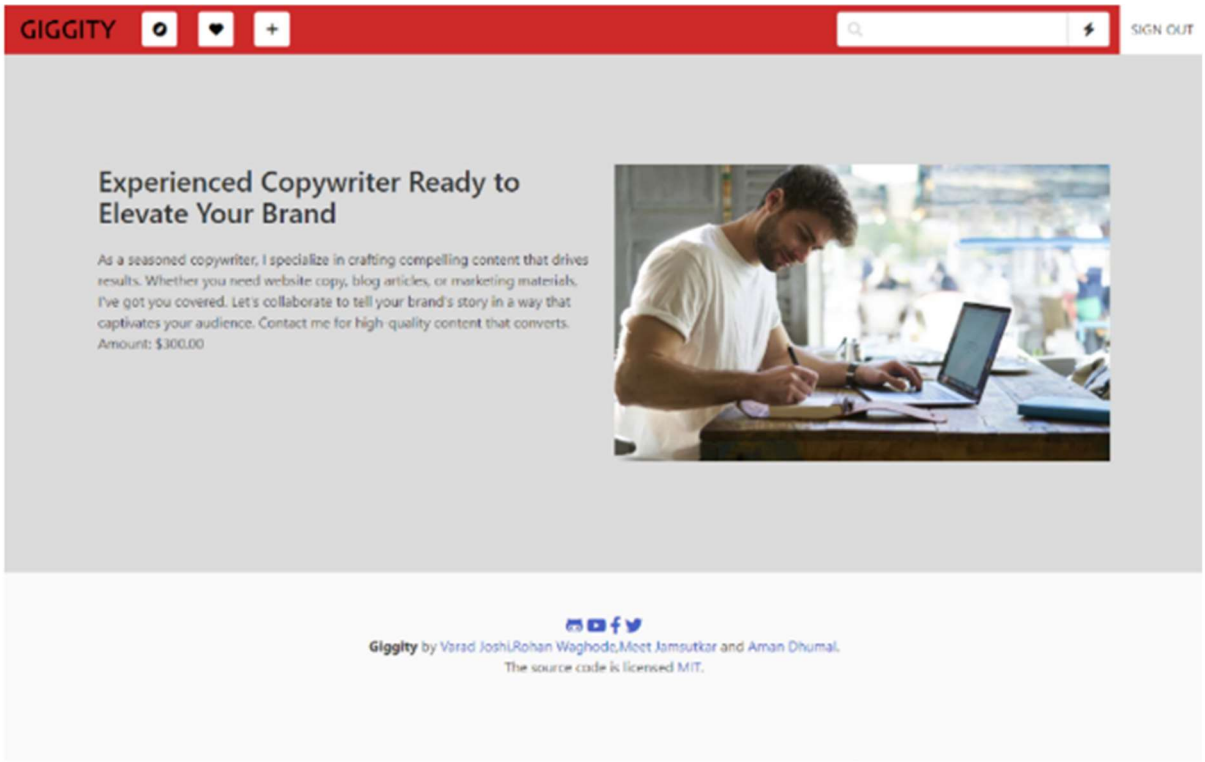

Figure 7: Freelancing Platform – Post Details

On the Post Detail Page, you'll find a comprehensive description of the project, including its requirements, budget, and timeline. Get a clear understanding of the opportunity to submit your proposal. It's your gateway to making informed decisions and securing your next freelance gig.

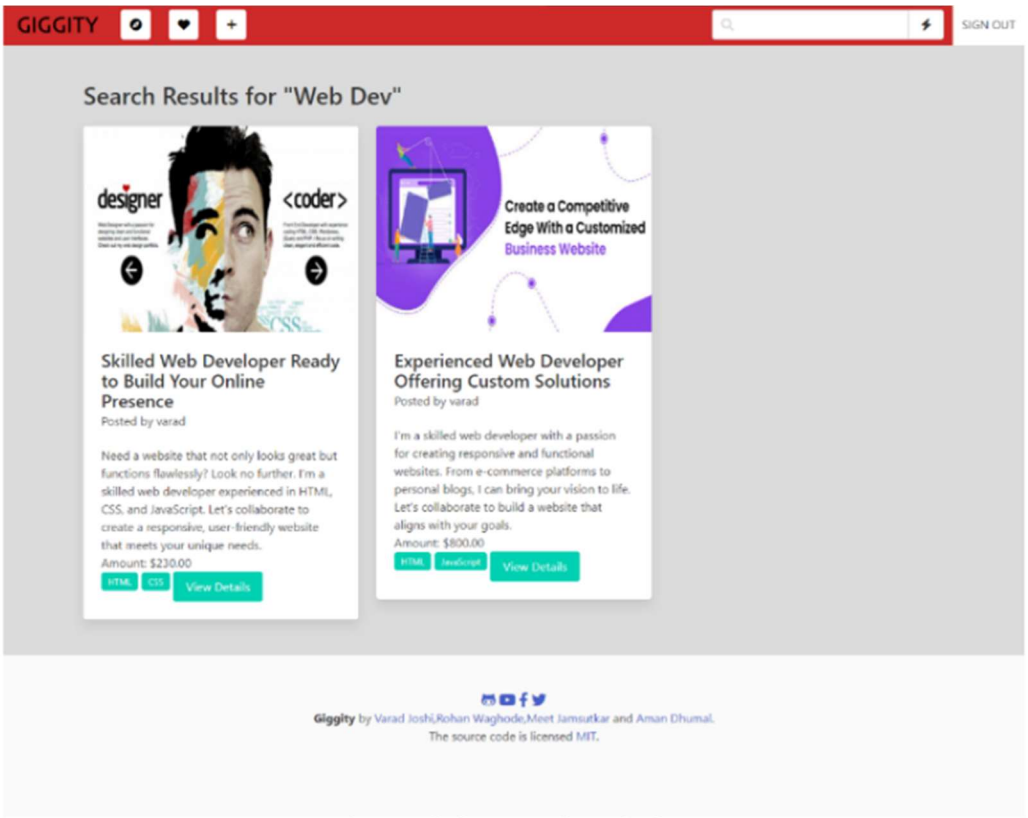

Figure 8: Freelancing Platform –Search Result of Posts using tags

The Search Page on our Freelancing Platform allows you to explore tailored search results of posts using relevant tags. Find the right projects or talents effortlessly by using tags that match your interests and needs. It's your shortcut to a world of opportunities and the perfect way to connect with the right people or projects

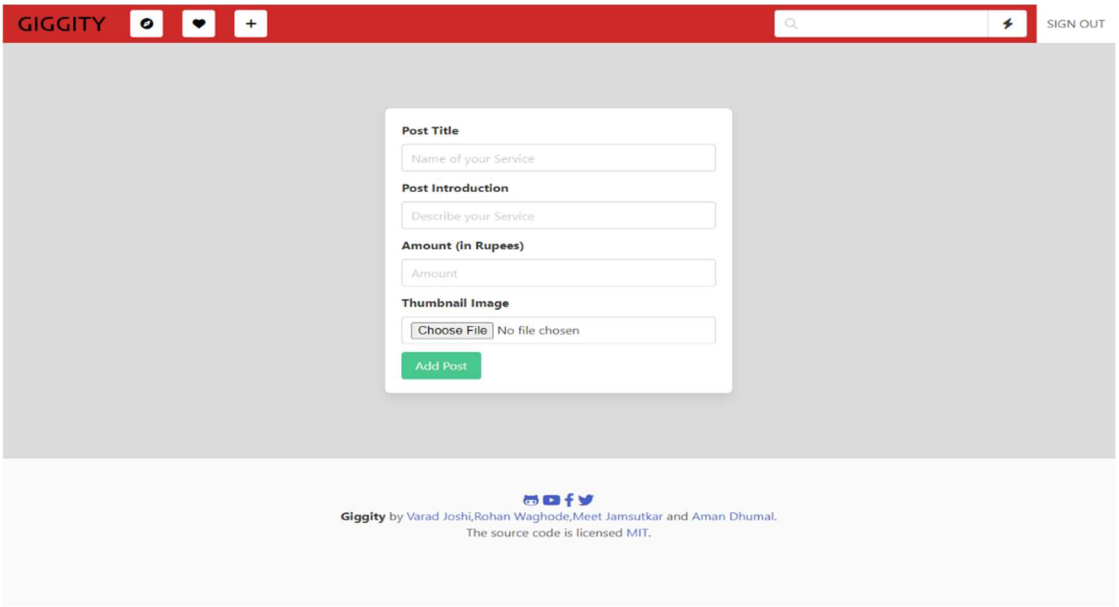

Figure 9: Freelancing Platform – Post a Service

The Create Post Page empowers you to showcase your skills and services to the world. Craft detailed project descriptions, set your pricing, and add tags to make your post easily discoverable. Whether you're a freelancer offering your talents or a client in need of services, this page is where opportunities are born.

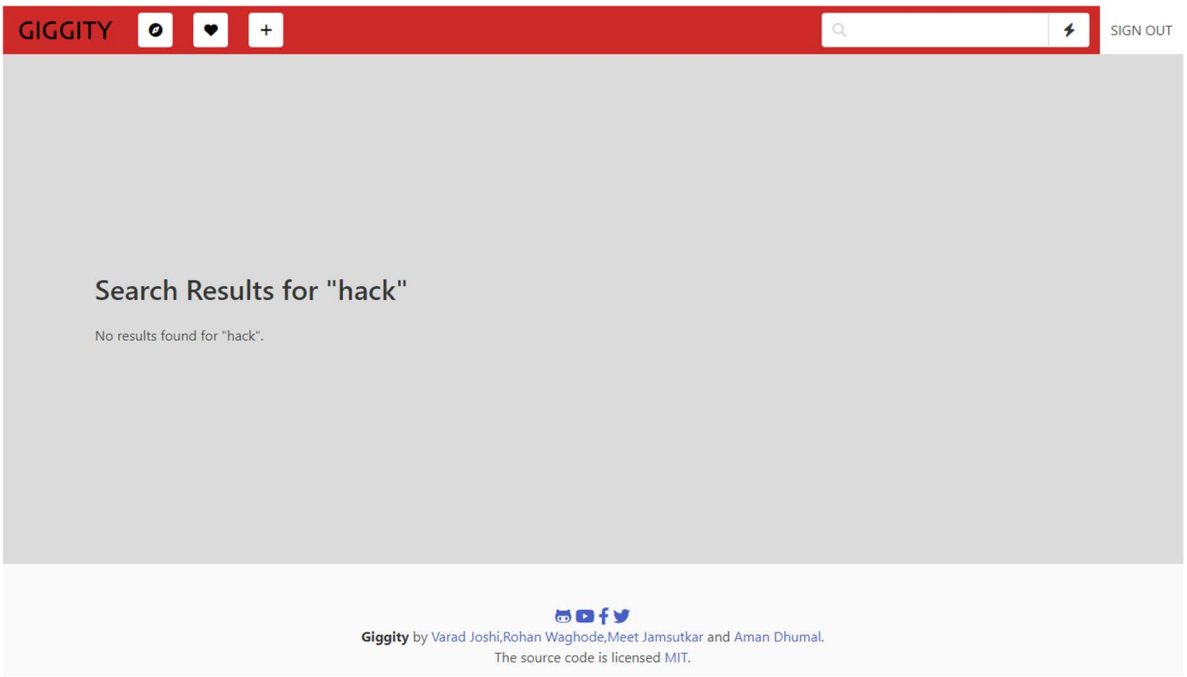

Figure 10: Freelancing Platform – Search Result of Undefined tag/Category

When curiosity leads to uncharted territories. The 'Result Not Found' Page graciously guides you when you search for an undefined or non-existent tag. Sometimes, the freelancing world may not have ventured there yet. It's a reminder that each tag represents unique skills and services, and the path to discovery starts with the right keywords.

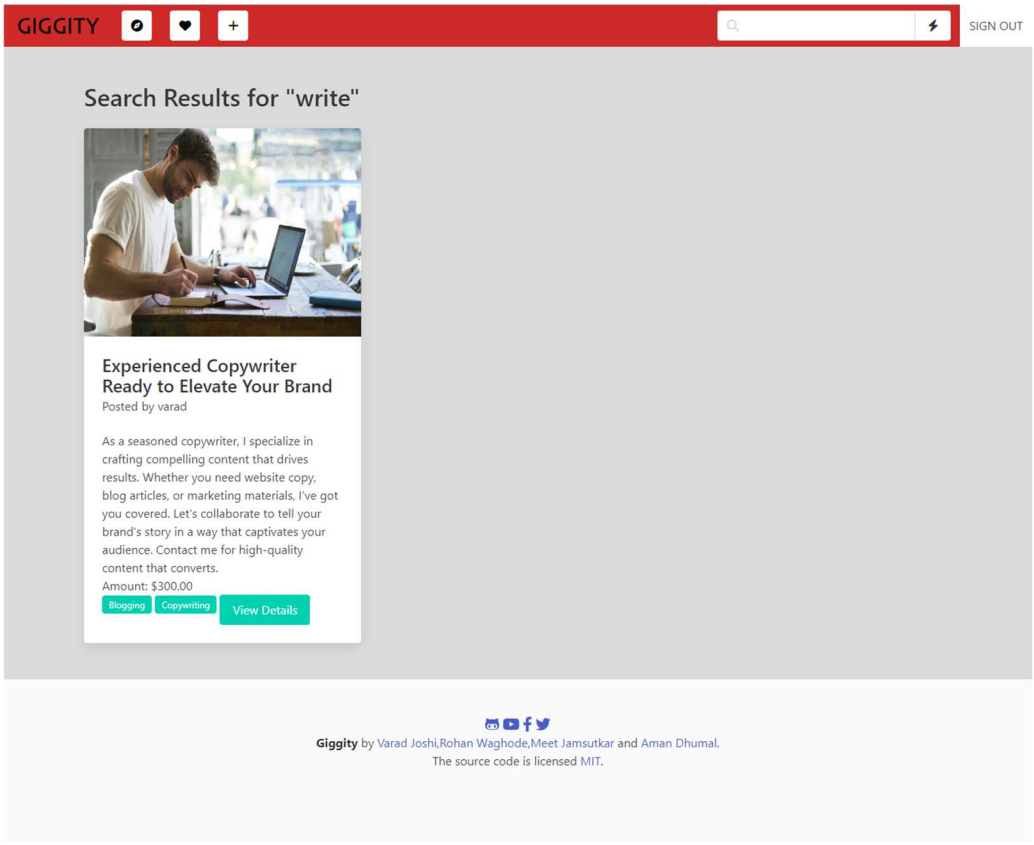

Figure 11: Freelancing Platform – Search Result of Posts using description.

Exploring with words. In addition to tags, our platform allows you to search using project descriptions. Discover the most relevant results based on your specific criteria and interests. It's a flexible way to find the perfect match for your project or services you need.

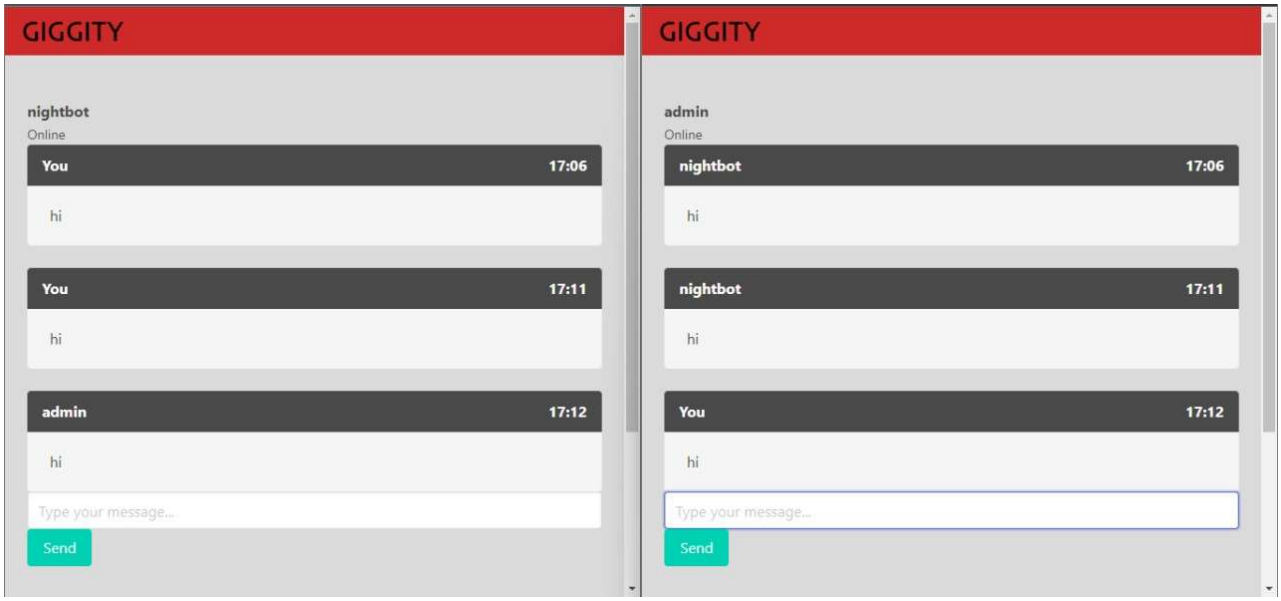

Figure 12: Freelancing Platform – Chatroom:

Real-time communication at your fingertips. Our Chat Room leverages the power of web sockets to provide instant, seamless conversations. Connect with clients, collaborators, or freelancers in real-time, share files, and discuss project details effortlessly. Experience a dynamic, interactive workspace that enhances collaboration and productivity.

Chapter 7

Project Scheduling

In the context of the Freelancing Platform, scheduling plays a vital role in organizing and managing the development process. The project schedule comprises a comprehensive list of milestones, tasks, and deliverables, serving as a roadmap for the project's execution. It outlines the timeline for task completion, allocation of resources and activities.

Sr. No	Group Member	Time Duration	Work to be done
1	Rohan Waghode Varad Joshi Meet Jamsutkar Aman Dhumal	**1st week of August**	Group formation and Topic finalization. Identifying the scope and objectives of the Mini Project.
			Discussing the project topic with the help of a paper prototype.
		3rd week of August	Identifying the functionalities of the Mini Project.
			Designing the Graphical User Interface (GUI).
2	Meet Jamsutkar Varad Joshi Rohan Waghode	**2nd week of September**	Database Design and Recommendation engine processing and working.
3	Rohan Waghode Varad Joshi Meet Jamsutkar Aman Dhumal	**1st week of October**	Database Connectivity and functionalities of modules in the app.
4	Rohan Waghode Varad Joshi Meet Jamsutkar Aman Dhumal	**Last week of October**	Integration of all modules, User Interfaces and Report Writing.

Table 2: Timeline Chart

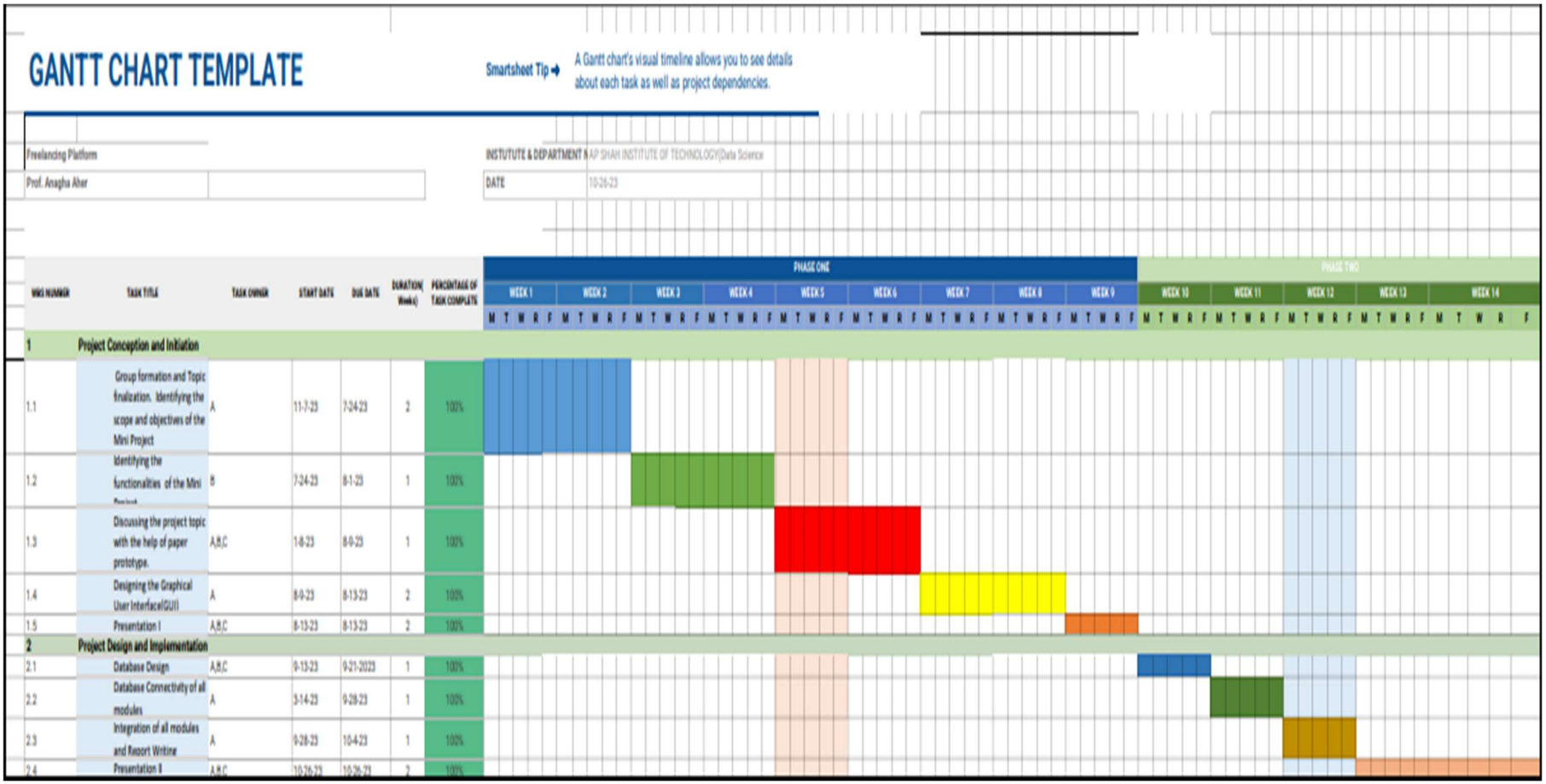

Figure 13: Gantt Chart of Freelancing Platform

To visualize this schedule, a Gantt chart is employed, providing a graphical representation of task durations, start and finish dates, and interactivity. Additionally, Gantt charts help illustrate the project's work breakdown structure and the relationships between activities, ensuring effective project management and progress tracking.

Here in the above figure, the rows of the chart contain the task titles such as the project conception and initialization as well as the project design and implementation which in subdivision contains the group formation, topic finalizing, prototype, GUI designing, backend implementation etc. The columns contain the duration of the task completed, percentage of work completed, number of weeks required to complete a particular task, the specific dates, the team members who contributed towards the completion of tasks

The detailed explanation of the Gantt chart is explained below: The project conception and initiation task were executed by the month end around 26/10/23. The task of initiation included many more sub-tasks such as group formation and topic finalization which was performed during the 1 week of project initialization. The group formed included 4 members Rohan Waghode, Varad Joshi, Meet Jamsutkar, Aman Dhumal and the finalized topic was Freelancing Platform. Further, the upcoming week led to the task of identifying the scope and objectives of the mini projects.

The next sub-task was to identify the functionalities of the project which was done by the two members Meet Jamsutkar, Varad Joshi, Rohan Waghode in a span of one week from 25/07/23 to 01/08/23. The discussion of the project topic with the help of a paper prototype was completed with equal contribution from all the group members within one week from 01/08/23-08/08/23.

The next task, Database Connectivity and functionalities of modules in the app were done by Rohan Waghode, Varad Joshi, Meet Jamsutkar, Aman Dhumal from 13/09/23 to 21/09/23. The Integration of all modules, user interfaces and report writing was completed by Rohan Waghode, Varad Joshi, Meet Jamsutkar, Aman Dhumal from 21/09/23 to 28/09/23. The preparation of final presentation II work was equally shared by all the group members in the time of 2 weeks from 4/10/23 to 26/10/23.

Chapter 8

Conclusion

In conclusion, The Freelancing web app Project is a groundbreaking initiative set to reshape the online freelancing landscape. It serves as a dynamic bridge connecting freelancers and clients, streamlining their collaboration in the digital age. This visionary platform empowers freelancers by providing access to a vast array of work opportunities across diverse industries, while clients benefit from an intuitive environment that simplifies the hiring process. Beyond mere matchmaking, this project envisions a comprehensive ecosystem with project management tools, secure payment systems, and robust communication channels, ensuring efficient and transparent collaborations.

The Freelancing web app Project extends its reach across geographical boundaries, making it a global hub for talent and opportunities. It embodies accessibility, efficiency, and a worldwide presence. The project's core objectives focus on enhancing job matching accuracy, delivering personalized job recommendations, creating a user-friendly platform, promoting skill development, ensuring data security, and implementing seamless payment integration. Its comprehensive scope covers user profiles, job listings, smart recommendations, easy communication, secure payments, skill enhancement, community networking, privacy, and cross-device accessibility.

The software requirements emphasize compatibility, user-friendliness, performance optimization, development efficiency, and a robust backend using a specific software stack. In summary, the Freelancing web app Project is a transformative endeavor that promises to usher in a new era of work, empowering both freelancers and businesses to thrive in the interconnected digital world.

REFERENCES

1) Isinkaye, Folasade Olubusola, Yetunde O. Folajimi, and Bolande Adefowoke Ojokoh. "Recommendation systems: Principles, methods and evaluation." _Egyptian informatics journal_ 16.3 (2015): 261-273.
2) Son, Jieun, and Seoung Bum Kim. "Content-based filtering for recommendation systems using multiattribute networks." _Expert Systems with Applications_ 89 (2017): 404-412
3) Xia, Peipei, Li Zhang, and Fanzhang Li. "Learning similarity with cosine similarity ensemble." _Information sciences_ 307 (2015): 39-52.
4) Dehak, Najim, et al. "Cosine similarity scoring without score normalization techniques." _Odyssey_. 2010.
5) Li, Xin, Lei Guo, and Yihong Eric Zhao. "Tag-based social interest discovery." _Proceedings of the 17th international conference on World Wide Web_. 2008.
6) He, Xiangnan, et al. "Neural collaborative filtering." _Proceedings of the 26th international conference on world wide web_. 2017
7) Koren, Yehuda, Steffen Rendle, and Robert Bell. "Advances in collaborative filtering." _Recommender systems handbook_ (2021): 91-142

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Prof. Anagha Aher** Expressing gratitude towards our HoD, **Prof. Anagha Aher**, and the Department of Computer Science and Engineering (Data Science) for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our teachers who gave us her valuable suggestions and ideas when we were in need of them. Lastly, we would like to thank our peers and colleagues for their helpful suggestions, discussions, and feedback, which enhanced the quality of our work.