|  |  |  |
| --- | --- | --- |
| جـامعـــة الأزهــــر  كليـــــة الهندســـــة (بنين)  قسم هندسة النظم والحاسبات | Logo  Description automatically generated |  |
| Al-Azhar University |
| Faculty of Engineering (boys) |
| Systems and Computer Engineering |
|  |

**Graduation Project**

Logo

Description automatically generated **Freelance Platform**

**A Project Submitted in Partial Fulfillment**

**of the Requirements for the Degree of Bachelor of Science**

**in Systems and Computer Engineering**

**Submitted By**

|  |
| --- |
|  |
| **Ahmed Mohamed Sabry 404020** |
| **Muhammad Ahmed Mndour 404101** |
| **Mohamed Ali Ahmed 404102** |
| **Mahmoud Saadallah Ibrahim 404128** |
| **Mahmoud Abdelfatah Ali 404131** |

**Supervised by**

**Dr. Fawzy Al-Rifai**

**2024**

## Examiner Committee

|  |  |  |
| --- | --- | --- |
| **Name** | **Rule** | **Signature** |
| Prof. Dr. Ali Rashid | President |  |
| Dr. Khaled Al-Shafi’i | President |  |
| Dr. Fawzy Al-Rifai | Supervisor |  |

**Date**

1th of July, 2024

## ABSTRACT

-The freelance industry has experienced significant growth in recent years, driven by the proliferation of digital platforms that connect clients with skilled professionals. These abstract aims to provide a concise overview of the key findings and insights presented in this study.

- The study explores the evolution of freelance platforms and their impact on the gig economy, focusing on the dynamics of client-freelancer interactions, user experiences, and emerging trends. Through a comprehensive literature review, various aspects of online freelance platforms were examined, including their advantages, challenges, and implications for both clients and freelancers.

-Furthermore, the study investigates effective communication strategies and security measures on freelance platforms, addressing concerns related to trust, payment security, and privacy. By synthesizing existing literature and analyzing data from surveys and interviews, this study offers valuable insights into the factors influencing the success and effectiveness of freelance platforms.

-In conclusion, this study contributes to the understanding of the freelance industry and provides recommendations for improving the design and functionality of freelance platforms. It underscores the importance of creating a conducive environment for client-freelancer collaboration while ensuring the security and satisfaction of all stakeholders involved.

### KEYWORDS: Client-Side; Service; Server-Side; ASP.Net; SQL Server;

## ACKNOWLEDGEMENTS

First and foremost, we thank and praise **Allah** for providing us with the patience, strength, well-being, and skills to complete this thesis at one of the most important stages in our lives.

We would like to thank our supervisor, Professor. **Fawzy Ibrahim Elrefai,** for his support and guidance.

We would also like to acknowledge Al-Azhar University for granting us to be among its respectful students.

We would also like to thank **Dr. Ali Halawa** for our participation in the mid-term.

## TABLE OF CONTENTS

[ABSTRACT… ii](#_bookmark0)i

[ACKNOWLEDGEMENTS iv](#_bookmark1)

[TABLE OF CONTENTS v](#_TOC_250026)

[CHAPTER 1 INTRODUCTION… 1](#_TOC_250025)

* 1. [Background 2](#_bookmark2)
  2. [Motivation 2](#_bookmark3)
  3. [Problem description 2](#_bookmark4)
  4. [Aims and Objectives 3](#_TOC_250024)
  5. [Methodology 3](#_bookmark5)

[CHAPTER 2 LITERATURE REVIEW. 5](#_TOC_250023)

* 1. Introduction 5
  2. [Previous Methodology 6](#_TOC_250022)

[CHAPTER 3 SYSTEM DESIGN 11](#_TOC_250021)

* 1. Introduction 11

[System Architecture 11](#_TOC_250020)

* 1. [Requirements 12](#_TOC_250019)
     1. [User requirements 12](#_TOC_250018)
     2. [System requirements 13](#_TOC_250017)
  2. [Process and Sequence Diagram 16](#_TOC_250016)
     1. [Use Case Diagram 17](#_TOC_250015)
  3. [Database Design 19](#_TOC_250013)
     1. Entity-Relationship Diagram 20
     2. Relationships 26
     3. Database tables 35
  4. [MVC Architecture 38](#_TOC_250012)
  5. [HTTP verbs 39](#_TOC_250011)
  6. [APIs 39](#_TOC_250010)
     1. What Is an API 39
     2. API endpoint 40

## CHAPTER 4 IMPLEMENTATION… .............................................................. 42

4.1 Introduction.................................................................................................................. 42

4.2 Tools & Technologies ................................................................................................. 43

4.3 Ui designs ................................................................................................................... 44

* + 1. Home page.....................................................................................................44
    2. Registration.....................................................................................................46
       1. Freelancer Registration ........................................................................46
       2. Client Registration................................................................................47
    3. Sign in ..............................................................................................................48
    4. Forget Password and Reconfirm email.............................................................48
    5. Change Password..............................................................................................49
    6. Account.............................................................................................................49
       1. Freelancer Account................................................................................49
       2. Update Freelancer Account ..................................................................50
       3. User Account..........................................................................................50
    7. Job Posts............................................................................................................52
       1. All job posts...........................................................................................52
       2. Upload job..............................................................................................52
       3. Client job posts.......................................................................................52
       4. Applicants on a certain job....................................................................53
       5. Edit job post...........................................................................................53
       6. Search for job.........................................................................................53
       7. Freelancer Favorite jobs.........................................................................54
    8. Apply for task ....................................................................................................54
       1. Freelancer applied tasks..........................................................................55
       2. Freelancer accepted tasks........................................................................55
       3. Client offers for a job..............................................................................55
    9. Contract...............................................................................................................56
       1. All contracts............................................................................................56
       2. Create contract........................................................................................56
       3. Confirm the contract...............................................................................57
       4. Update contract.......................................................................................57
    10. Payment..............................................................................................................58
    11. Chat....................................................................................................................59
    12. Ask AI................................................................................................................59

4.4 Dataset and Preprocessing ............................................................................................ 60

4.4.1 Data cleaning: ............................................................................................... 60

4.4.2 Feature engineering ...................................................................................... 60

4.4.3 Data normalization........................................................................................ 60

4.4.4 Encoding categorical variables ..................................................................... 60

4.4.5 Splitting the dataset....................................................................................... 61

4.5 Source code.....................................................................................................................61

## CHAPTER 5 CONCLUSION AND FUTURE WORK… ................................................62

5.1 Conclusion ....................................................................................................................................... 62

5.2 Future Work ..................................................................................................................................... 62

[**REFERENCES**… 64](#_TOC_250002)

[**APPENDICES. .65**](#_TOC_250001)

## CHAPTER 1 INTRODUCTION

-When we decided to build a website to connect user with qualified freelancer, we took several key considerations into account.

-Evolution of the Freelance Market: The freelance market has witnessed rapid growth in recent years, with many individuals preferring to work as freelancers rather than adhering to traditional employment structures. Freelance platforms facilitate this growth by providing a digital environment for clients and freelancers to communicate and complete transactions easily.

-Challenges and Opportunities: Despite the clear benefits of freelance platforms, they also face challenges. These include intense competition, difficulties in finding suitable job opportunities, and risks related to health insurance and retirement. On the other hand, freelance platforms also open new doors for opportunities, such as work diversity, schedule flexibility, and the ability to earn additional income.

-Security and Trust: Building trust between clients and freelancers is crucial for the success of freelance platforms. These platforms must provide mechanisms to verify the identity of freelancers and clients, as well as effectively and fairly resolve disputes. Additionally, platforms must also provide a secure and trustworthy environment for financial transactions.

-Innovation and Development: Continuous innovation and development in freelance platforms Are essential to attract users and differentiate from competitors. This requires the use of new technologies such as artificial intelligence and machine learning to improve user experience and provide more diverse and effective services.

-In summary, freelance platforms play a vital role in shaping the future of work, offering tremendous opportunities for both freelancers and clients alike. With continuous advancements in this field, freelance platforms are expected to continue evolving and expanding to meet the needs of the ever-changing job market.

### Background

-Freelancing platforms are places where professionals can use technology to offer their services remotely.

-These platforms have grown a lot in recent years, showing how remote work is changing.

-By using communication technologies, freelancers can work with clients from anywhere, giving them more flexibility and making it easier to connect.

-Freelancing has many benefits that people recognize, such as easier communication between freelancers and clients, the ability to work together effectively, and the opportunity to balance work and personal life.

-In the world of freelancing today, machine learning is making a big impact. It helps with things like managing projects, matching freelancers with the right jobs, and sparking innovation in different industries.

-When freelancers use machine learning, it opens new chances for them to grow and succeed in the ddigital marketplace that is always changing.

### Motivation

The motivation behind developing a freelance platform there are more and more people want to work on their own terms and have control over their schedules. These platforms allow individuals to use their skills and knowledge while giving employers the opportunity to find talented people from around the world.

### Problem description

### -Freelance platforms serve as online marketplaces connecting freelancers and clients, but they face several challenges. Quality control, trust and safety, pricing and payment, communication and collaboration, competition and saturation, scalability and technical infrastructure, legal and regulatory compliance, and dispute resolution are common problems encountered by freelance platforms.

### Aims and Objectives

### 1. Enhance Accessibility: The platform is easy to use for both freelancers and clients.

### Freelancers can easily find work and clients can get the work done they need.

### 2. Ensuring Security and Trust: Prioritizing security measures and implementing robust authentication protocols to ensure the safety of transactions and build trust between freelancers and clients on the platform.

### 3. Facilitate Collaboration: Creating features and tools that make it easy for freelancers and clients to communicate and work together smoothly.

### Methodology

1. User Registration: Freelancers and clients create accounts on the website, providing their personal or business details and skills. This information is valuable for platform to connect freelancers with the right projects and help clients find right talent for their needs.

2. Profile Creation: Freelancers create detailed profiles showcasing their experience, portfolio, and rates. Clients can browse these profiles to evaluate freelancers and make informed hiring decisions. The platform also offers features like reviews and ratings to establish credibility and trust.

3. Job Posting: Clients post job listings specifying their requirements, project details, budget, and timeline. Freelancers can browse available jobs and submit proposals, showing their qualifications, related experience, and suggested pricing.

4. Project Management: Once a freelancer is selected, the platform facilitates the project management process. It includes tools for communication, this tools make the communication between for freelancers and clients easier

5.Payment Processing**:** Develop a secure and reliable payment system that allows clients to pay freelancers for their work.

6.Feedback and Ratings**:** After project completion, clients have the opportunity to provide feedback and ratings for the freelancers they worked with. This feedback helps build the reputation and credibility of freelancers on the platform and assists future clients in making hiring decisions.

7.Chat AI on freelance platforms enhances communication, providing instant, accurate responses to queries, automating routine tasks, and improving overall efficiency. This leads to faster project completion, better client-freelancer interactions, and streamlined workflow management.

**CHAPTER 2**

**LITERATURE REVIEW**

**2.1 Introduction**

The freelance market has experienced unprecedented growth, fueled by technological advancements, and changing attitudes towards work. Today, freelancers represent a substantial segment of the global workforce, offering their expertise and services to clients worldwide. This shift towards freelancing reflects a desire for flexibility, autonomy, and diverse career opportunities among professionals.

At the heart of this freelance revolution are online platforms that serve as intermediaries between clients seeking services and freelancers offering their skills. These platforms provide a digital marketplace where clients can post projects, browse profiles, and hire freelancers based on their qualifications and expertise. Similarly, freelancers can showcase their skills, bid on projects, and build a portfolio of work through these platforms.

One of the key pillars of successful freelance platforms is effective communication. Clear and transparent communication between clients and freelancers is essential for understanding project requirements, setting expectations, and ensuring timely delivery of work. Moreover, trust is paramount in the freelance ecosystem, as clients rely on freelancers to deliver high-quality work, and freelancers depend on clients for timely payments and positive reviews.

In addition to communication and trust-building, freelance platforms must address various challenges, including competition, pricing transparency, and security concerns. Freelancers must navigate through a crowded marketplace to find suitable opportunities, while clients seek assurance of the quality and reliability of freelancers' work.

As the freelance market continues to evolve, freelance platforms play a crucial role in shaping the future of work. By providing a platform for seamless collaboration and talent acquisition, these platforms empower freelancers to thrive in a dynamic and competitive landscape. Moreover, they enable clients to access a diverse pool of talent and scale their projects according to their needs.

In conclusion, freelance platforms represent a transformative force in the modern workforce, offering opportunities for both clients and freelancers to succeed in an increasingly digital and interconnected world. Through effective communication, trust-building, and innovation, freelance platforms are poised to drive continued growth and innovation in the freelance economy.

**2.2** **Previous Methodology**

1. User registration: is a process through which freelancers and clients create accounts on a website by providing their personal or business details and skills. This information is essential for the platform to effectively connect freelancers with suitable projects and assist clients in finding the right talent for their specific needs.

-During the registration process, freelancers typically provide information such as their name, contact details, professional experience, education, portfolio samples, and a description of their skills and expertise. They may also specify their rates, availability, and preferred project types. This information helps the platform understand the freelancer's background and capabilities, enabling it to match them with relevant job opportunities and clients.

-On the other hand, clients registering on the platform provide details about their name ,country and contact information.

-By gathering and analyzing this data from both freelancers and clients, the platform can employ various algorithms and matching techniques to facilitate effective connections. It can suggest freelancers to clients based on their project requirements, and conversely, notify freelancers about relevant job opportunities that align with their skills and preferences.

2. Profile creation is an essential step for freelancers on a platform, as it allows them to showcase their experience, portfolio, and rates to potential clients. By creating detailed profiles, freelancers can provide a comprehensive overview of their skills, expertise, and past work.

-Freelancers typically include information such as their professional background, education, certifications, and any relevant industry experience. They may also highlight their specific skills, such as programming languages, design tools, writing styles, or marketing strategies. This information helps clients assess the freelancer's qualifications and suitability for their projects.

-In addition to textual information, freelancers often use their profiles to showcase samples of their previous work. These portfolio samples could be in the form of images, videos, written articles, website links, or any other relevant media, depending on the freelancer's field. Displaying a diverse range of high-quality work allows clients to evaluate the freelancer's capabilities and style.

-Freelancers also specify their rates and pricing structures in their profiles. This transparency helps clients understand the freelancer's financial expectations and determine if it aligns with their budget for the project. By indicating their rates upfront, freelancers can attract clients who are comfortable with their pricing.

-Overall, detailed profiles allow freelancers to present themselves effectively to clients, showcasing their skills, experience, portfolio, and rates. This information empowers clients to make informed hiring decisions by evaluating freelancers' qualifications, work samples, and reputation through reviews and ratings.

3. Job posting is a crucial feature offered by platforms where clients can create listings to communicate their specific project requirements, budgets, and timelines. By posting detailed job listings, clients can attract freelancers who are suitable for their projects.

-When creating a job listing, clients typically provide a clear description of the project, including the scope of work, desired outcomes, and any specific skills or qualifications they are looking for in a freelancer. They may also specify the estimated budget range for the project and the expected timeframe for completion. Additionally, clients may include any additional information or attachments that can help freelancers understand the project requirements better.

-Freelancers, on the other hand, can browse through the available job listings on the platform and identify projects that match their skills and interests. They can review the job descriptions, project details, and client requirements to determine if they are well-suited for the job.

-If a freelancer finds a job listing that aligns with their expertise and availability, they can submit a proposal to the client. In their proposal, freelancers typically showcase their qualifications, related experience, and a detailed explanation of how they plan to approach the project. They may also include pricing information, such as their proposed rates or a cost estimate for completing the project.

-The proposal serves as an opportunity for freelancers to demonstrate their understanding of the project requirements and convince the client of their capabilities to successfully deliver the desired outcomes. It allows freelancers to highlight their relevant skills, past work experience, and any unique selling points that set them apart from other applicants.

-The client then reviews the submitted proposals and evaluates each freelancer's qualifications, experience, and proposed pricing. Based on this evaluation, the client can choose to contact and engage with the freelancer they believe is the best fit for the project.

-Overall, the job posting feature enables clients to communicate their project requirements, budgets, and timelines effectively. It allows freelancers to browse available jobs, submit proposals to showcase their qualifications, related experience, and suggested pricing. This streamlined process facilitates effective matchmaking between clients and freelancers, leading to successful project collaborations.

4. Project management is an integral part of the platform's functionality, facilitating smooth collaboration between freelancers and clients. It typically includes various tools and features to streamline communication, and project progress tracking.

-To support effective communication, the platform offers messaging systems or chat interfaces that allow freelancers and clients to communicate directly within the platform. This eliminates the need for external communication channels and provides a centralized space for discussing project details, clarifying requirements, and addressing any questions or concerns.

5. Payment processing is another critical aspect of the platform's functionality. A secure and reliable payment system is developed to facilitate transactions between clients and freelancers. This system ensures that freelancers receive timely and accurate compensation for their work while providing clients with a convenient and trustworthy way to make payments.

-The payment system typically handles processes such as invoicing, payment tracking, and secure payment gateways. It may support various payment methods, including credit cards, bank transfers, or digital wallets, depending on the platform's capabilities and the preferences of freelancers and clients.

-The platform ensures that the payment system is secure, protecting sensitive financial information and preventing unauthorized access or fraudulent activities. It may employ encryption techniques, comply with relevant security standards, and implement measures such as two-factor authentication to enhance the security of financial transactions.

-By providing a secure and reliable payment system, the platform fosters trust between freelancers and clients, ensuring a fair and transparent process for compensating freelancers for their work.

6. Feedback and ratings are critical components of the platform's ecosystem as they play a crucial role in establishing the reputation of freelancers and enabling informed decision-making for clients.

-Once a project is completed, clients have the opportunity to provide feedback on their experience working with the freelancer. This feedback can cover various aspects such as the quality of work, adherence to deadlines, communication skills, professionalism, and overall satisfaction with the collaboration.

-The feedback system serves multiple purposes. Firstly, it allows clients to express their opinion and share their experience, providing valuable insights into the freelancer's performance. This feedback can help freelancers understand their strengths and areas for improvement, enabling them to enhance their skills and refine their approach to future projects.

-Feedback and ratings contribute to establishing a freelancer's reputation on the platform. Positive feedback and high ratings from satisfied clients serve as endorsements for the freelancer's work and professionalism. This reputation-building aspect is crucial for attracting new clients and securing future project opportunities.

-For clients, feedback and ratings provide a means to assess the credibility and reliability of freelancers. By reviewing the feedback left by previous clients, potential clients can gain insights into the freelancer's track record, work ethics, and ability to meet project expectations. This information helps clients make informed decisions when selecting freelancers for their own projects.

-Overall, feedback and ratings are essential for freelancers to establish a good reputation and gain trust from clients on the platform. They allow clients to share their experience and provide valuable insights into the freelancer's performance. This feedback helps freelancers improve their skills and professionalism, while also assisting clients in making informed decisions when hiring freelancers for their projects.

**CHAPTER 3**

**SYSTEM DESIGN**

* 1. **Introduction**

The system design of the website and application included a robust and secure platform that made the services easy to access. The goal is to develop an easy-to-use interface that meets the specific needs of users to ensure the privacy and security of their sensitive medical data.

### System Architecture

1.Client side: The client-side architecture includes the user's web browser or application. Interacts with server-side components to request and display information to the user. Use HTML, CSS, and JavaScript to create a responsive, easy-to-use interface.

2.Web server: The web server handles requests from clients and directs them to the appropriate components. It can be used like Apache or Nginx. The web server sends requests to the application process.

3.Application server: The application server hosts the business logic and core functions of the website. It receives and processes requests from the web server and interacts with the database and external services. Technologies such as Node.js, Java or Python were used to implement the application server.

4.Database: The database stores and manages user data, appointment information, medical records, and other relevant data. Suitable database management system (DBMS) Use SQL based on project requirements. Ensure data security and privacy by implementing appropriate encryption and access controls.

5.External Services: Integrate with external services such as payment gateways, electronic health record systems, or telemedicine platforms if required. This allows for seamless functionality and enhances user experience.

6.Security layer: implement security measures to protect user data and ensure system confidentiality, integrity and availability. Use encryption technologies and secure communication protocols (HTTPS) and implement user authentication and authorization mechanisms.

7.Scale and Load Balancing: The system is designed to handle a large number of users and traffic. It applies load balancing techniques to distribute the workload across many, ensuring high availability and performance.

8.Monitoring and Logging: Monitoring tools to track system performance, detect problems, and create logs for debugging and auditing purposes. Server health, database performance, and user activity are monitored to identify and resolve any potential security bottlenecks or threats.

9.Cloud Infrastructure: In the future, the use of cloud infrastructure services such as AWS or Google Cloud has been considered to host and scale the system. Cloud services provide flexibility, scalability, and reliability.

* 1. **Requirements**

### 3.2.1 User Requirements

### 3.2.1.1 Login/Sign up Requirements:

- Users shall be able to sign up with their email, name and password.

- Registered users shall be able to login with their email and password.

### 3.2.1.2 Freelancer Requirements:

-Freelancers shall be able to edit their profile page.

-Freelancers shall be able to add and remove default profile segments.

-Freelancers shall be able to choose the interest areas and tags they want to be notified of.

- Freelancers shall be able to add a project to their favorites.

-Freelancers shall be able to rate a project that they have completed.

-Freelancers shall be able to pin and remove previous contracts to their account.

- Freelancers shall be able to filter search results down to their certain categories, tags, and budget range.

-Freelancers shall be able to bid for a project.

-Freelancers shall be able to inform clients about project completion by using messaging service.

- Freelancers shall be able to upload their works to system.

### 3.2.1.3 Client Requirements:

-Clients shall be able to create project postings.

-Clients shall be able to set a deadline for their project postings.

-Clients shall be able to add tags to their project postings.

-Clients shall be able to choose tags from a searchable list of preexisting tags and create new tags (not used before) for their project postings.

- Clients shall be able to choose a budget range for their project postings.

- Clients shall be able to add project details to their postings using a rich text editor.

- Clients shall be able to propose a project to a certain freelancer.

- Clients shall be able to decide on any bid placed on their project.

- Clients shall be able to delete a project.

-Clients shall be able to contract any freelancer who placed a bid on the project.

-Clients shall be able to cancel a contract if freelancer misses the deadline.

### 3.2.1.4 Feedback Requirements:

- Users shall be able to comment and post feedback on the project

* Clients shall be able to post feedback for working process, final product and also a text input for detailed elaboration.
* Freelancers shall be able to post feedback for payment, communication, project specifications/description.

- Users shall be able to rate and comment about users they worked with.

### 3.2.1.5 User profile Requirements:

- Users shall be able to view public profiles of other freelancers and clients.

- Users shall be able to pin feedback to top of their profile.

- Users shall be able to enable notifications through e-mail to get notified of messages and actions related to them.

- Users shall be provided a way of account recovery option in case they lose access to their account.

### 3.2.1.6 Annotation Requirements:

- Users shall be able to annotate the text in the project description.

- Users shall be able to annotate the images in the project descriptions.

- Owner user of an annotation shall be able to edit or delete his/her annotation.

**3.2.2 System Requirements**

### 3.2.2.1 Bidding Requirements:

-Given bids to a project with their amounts and given dates shall not be visible to any user except the client.

-The system shall notify the users upon a change in the post details.

-In case of a change to post details, the post shall be suspended temporarily and be put in moderation queue to confirm the change.

### 3.2.2.2 Recommendation Engine Requirements:

-Freelancers shall be given a recommendation of project postings that are related to their previous jobs, fields, interests and tags they prefer.

-Clients shall be suggested of top and rising freelancers that may be interested in their posting.

-During project posting creation, a list of successfully completed projects of similar topics shall be presented to the user to help them better specify the scope of the project.

#### **3.2.2.3 Payment Requirements:**

-The advance payment that the client promises shall be withdrawn from their account and be held until both parties agree that the conditions to said payment is met.

-Client shall not be able to contract with a freelancer if he/she doesn't have enough cash in his/her account.

#### **3.2.2.4 Payment Requirements:**

-The system shall support both basic text and semantic search using the keywords user has typed in.

-The system shall provide filtering mechanism by considering the title and description of the contents.

#### **3.2.2.5 User Creation Requirements:**

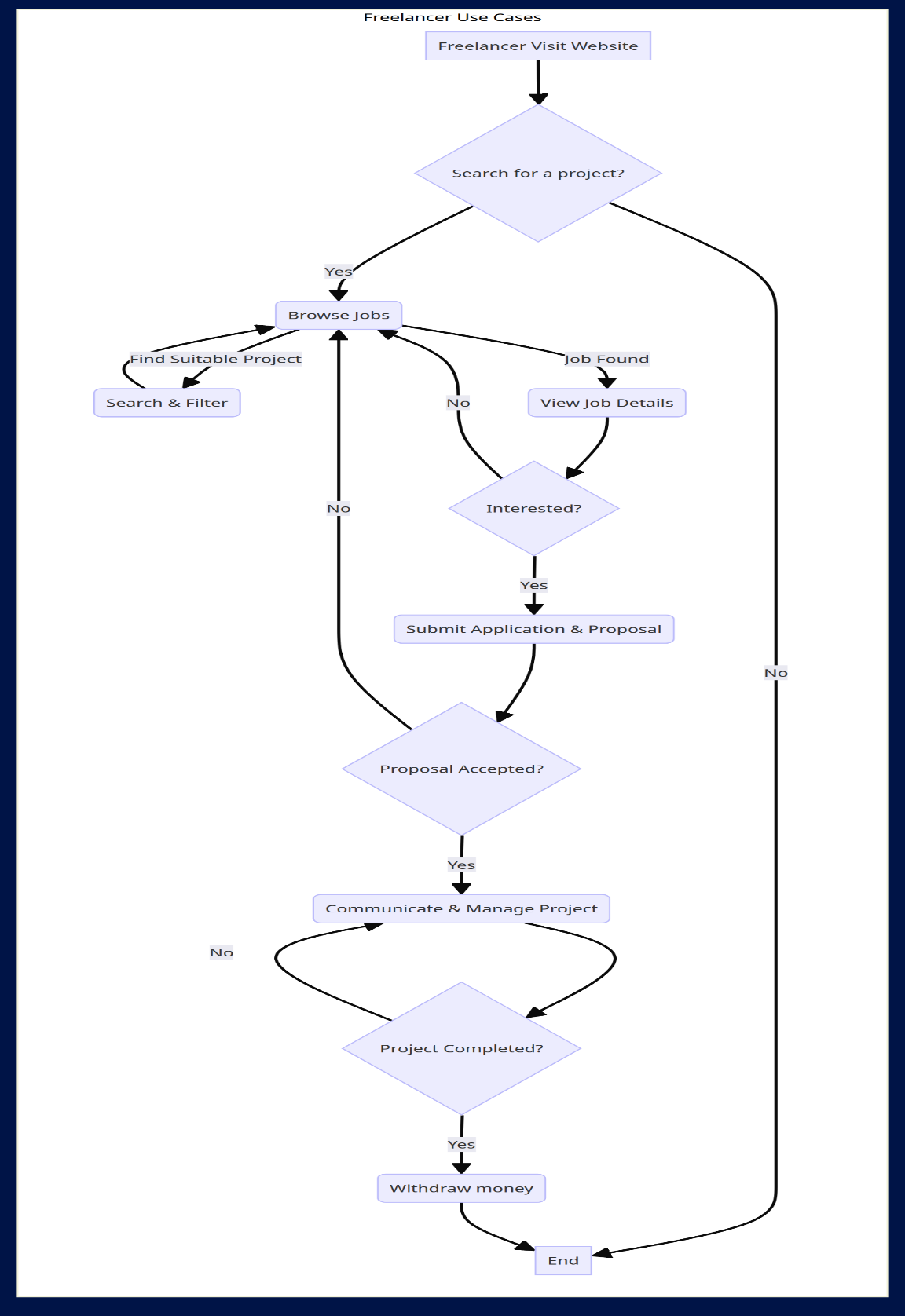
-The system shall suggest user to create an account only when they want to commit to creating a project posting.

### 3.3 Process and Sequence Diagram

### Authentication

### 3.3.1 Use Case Diagram

### Use Case-Freelancer

**Use Case-Freelancer**  
  


### 3.4 Database Design

### Full schema SchemaWithoutPropsWithout Background 3.4.1 Entity-Relationship Diagram

### User Entity User Prop no background Reviews Entity Reviews no back ground Category Entity Category prop no background Reports Entity Reports prop no background

### Notification Entity NotificaionProp no background Apply Task Entity ApplyTaskProp no background JobPost Entity JobPost prop no background

### Languages Entity Language prop no background

### Favorites Entity Favorites no background

### Portfolio Entity Protfolio prop no background

### Contract Entity ContractProp no background

### Skills Entity Skills prop no background

### 3.4.2 Database Relationship

### User Apply Task Relation - User may apply for zero or more task.

### - Apply Task may has one or more user applayed to it. User Contract Relation

### - User (Freelancer) has zero or more contract.

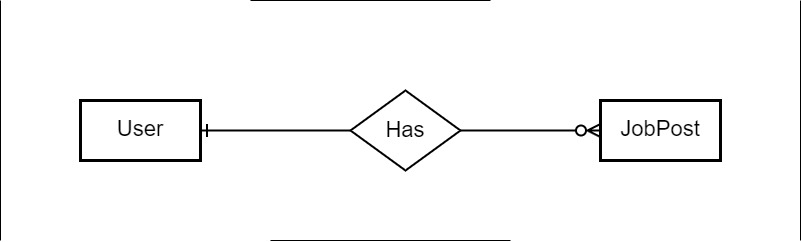
### - User (Client) make zero or more contract.

### - Contract has one or more user.

### 

### User Favorites Relation - User has zero or Favorites.

### - Favorites must be applied to one user.

  
  
  
  
 **User job Post Relation** 

### - Job Post must be applied to user.

### - User has zero or more Job Post.

### User Languages Relation - User has one or more Language.

### - language spoken by zero or more users

### 

### 

### User Notification Relation - User has zero or more Notification.

### - Notification send to one or more users.

### User Languages Relation - User (Client) Make zero or more Payment.

### -User (Freelancer) receive zero or more payment.

### - payment must belong to user (Freelancer and Client).

### User Portfolio Relation -User has zero or Portfolio (Projects).

### -portfolio must be belong to one user.

### -User Reports Relation -User (Freelancer) has zero or more Reports.

### -User (Client) makes zero or more Reports.

### -Report must belong to one user.

**User Reviews Relation**

### 

### -User (Freelancer) Has zero or more Reviews.

### -User (Client) makes zero or more Reviews.

### - Reviews must belong to User.

**User Skills Relation**

### - User has zero or more skill.

### - skills belong to zero or more users.

### 

### Skills Category Relation - Skill belong to zero or more Category.

### - Category has one or more skill.

### 

### Favorite Job Post Relation - Job Post has zero or more Favorite.

### - Favorite must belong to one JobPost.

### 

### Apply Task Job Post Relation - Apply Task must belong to one jobpost.

### - Job post has zero or more ApplyTask. Category Job Post Relation

### - Category has zero or more jobpost.

### - Job Post must belong to one category.

### Contract Job Post Relation - Contract must belong to jobpost.

### - job post has zero or more contract.

### Notification Job Post Relation - Job post has zero or more Notification.

### - Notification belong zero or one job post.

### Skills Job Post Relation

### - skills belong to zero or more jobpost.

### - job post must need one or more skills.

### Contract Payment Relation

### -Contract must have one payment method.

### -Payment must belong to one contract.

### 

### Apply Task Notification Relation -Apply task has one or more Notification.

### -Notification may belong to Apply task.

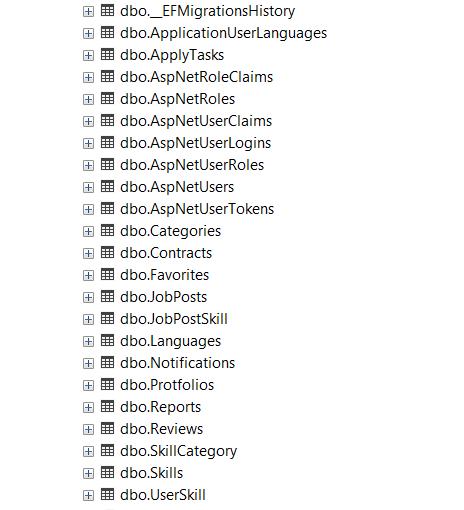
### 

### Apply Task Reviews Relation

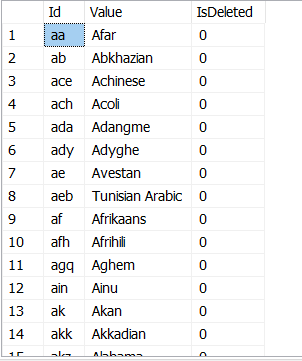
### -Apply Task has zero or more Reviews.

### -Reviews belong to zero or one ApplyTask.

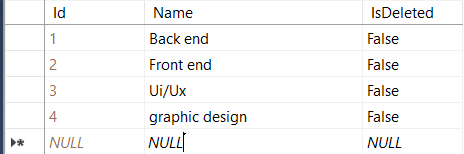
**3.4.3 Database Tables**



Languages

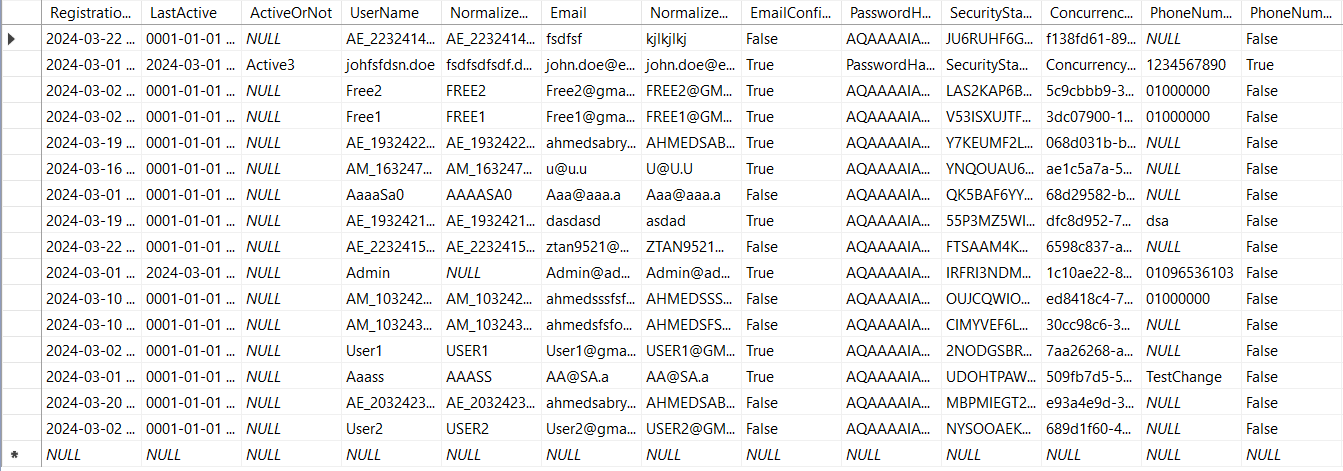


Categories

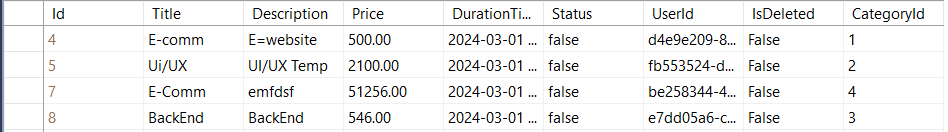


User

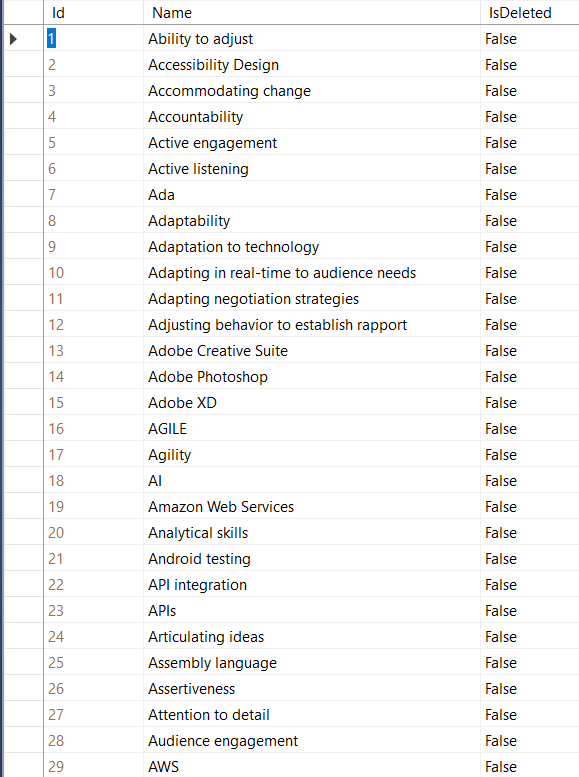




**Job post**

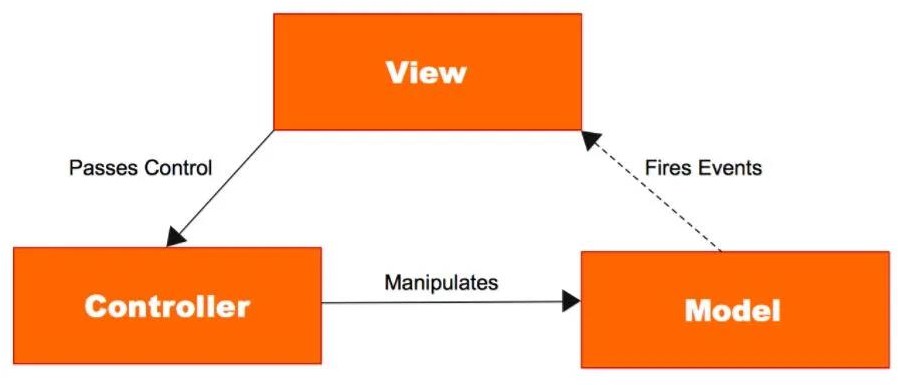


**Skills**

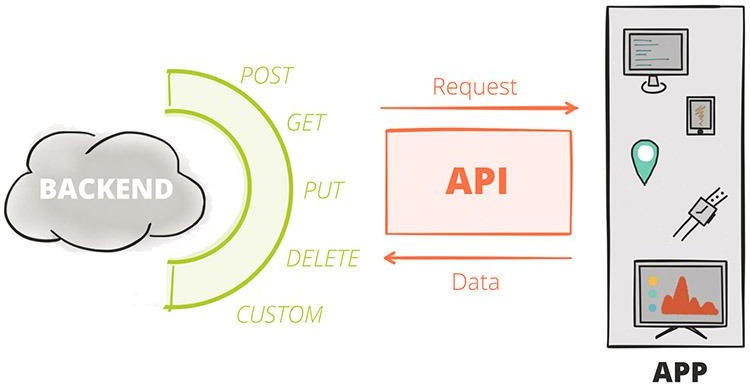


### 3.5 MVC Architecture

* MVC (Model-View-Controller) architecture is a design pattern used in software development.
* It separates the application into three interconnected components: model, view ,and controller.
* The model represents the data and business logic, the view handles the user interface, and the controller mediates between the two.
* MVC promotes code reusability, maintainability, and separation of concerns.
* It is widely used in web development frameworks like Laravel, and ASP.NET MVC.
* It is also applicable to desktop and mobile application development.

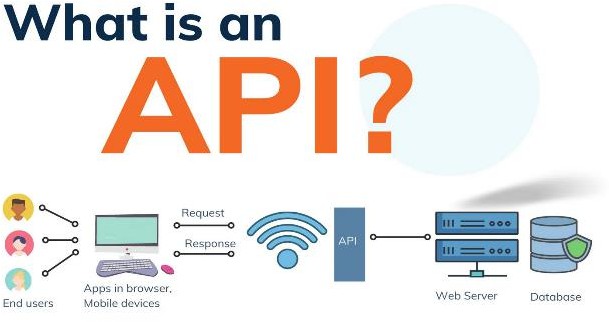


### 3.6 HTTP verbs

HTTP verbs, also known as HTTP methods, are used to indicate the type of action to be performed on a resource. The most commonly used HTTP verbs are GET (retrieve data), POST (submit data), and DELETE (remove data). Other verbs include PUT (update data) and PATCH (partially update data).

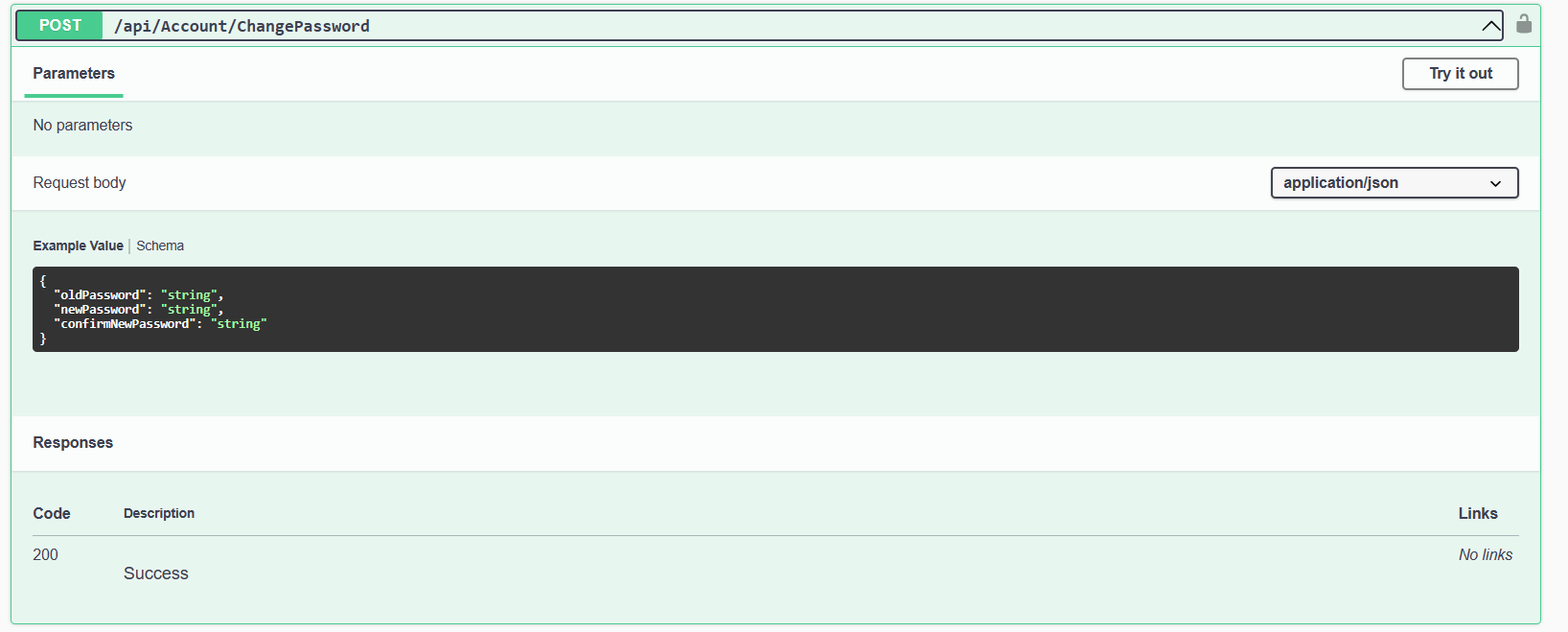
### 3.7 APIs

**3.7.1 What Is An API**

* + - * API (Application Programming Interface ) is a set of rules and protocols that enable software applications to communicate and interact with each other.
      * It defines how different components should interact, what data can be exchanged, and what operations can be performed .
      * APIs allow applications to access functionalities of a system or service without knowing the underlying implementation details. They are commonly used in web development to connect frontend and backend systems and integrate different services and platforms.

### 3.7.2 API endpoint

An API endpoint is a specific URL or URI (Uniform Resource Identifier) that an API (Application Programming Interface) exposes to enable interaction with a particular functionality or resource. It represents a specific operation or action that can be performed by making an HTTP request to that endpoint. API endpoints can be used to retrieve data, send data, update resources, or perform other operations depending on the API's design and functionality.



**CHAPTER 4**

**IMPLEMENTATION**

**4.1 Introduction**

The implementation phase of creating a website to connect clients with freelancers for various services is a crucial step in turning the system design into a functional and operational platform. This phase involves translating the design specifications into actual code, integrating various components, and ensuring the smooth functioning of the website. This section provides an overview of the key considerations and steps involved in the implementation process.

1. Development Environment Setup:

* Set up the necessary development environment, including selecting the appropriate programming languages, frameworks, and tools based on the system design requirements.
* Install and configure the development environment, ensuring compatibility and smooth integration of different components.

2. Front-end Development:

* Begin by implementing the user interface design using HTML, CSS, and JavaScript.
* Develop responsive and user-friendly web pages that align with the design specifications.
* Implement interactive features, such as search functionality, Profile Creation and Management, Job Posting and Search, Messaging and Communication, Ratings and Reviews and user profile management.

3. Back-end Development:

* Implement the necessary APIs and web services to handle user authentication, data retrieval, and communication between the front-end and back-end components.
* Develop the database schema and implement data models to store and retrieve user information.

4. Integration and Testing:

* Integrate the front-end and back-end components to ensure seamless communication and functionality.
* Test the website's performance, security, and scalability to ensure it can handle the expected user load and maintain data integrity.

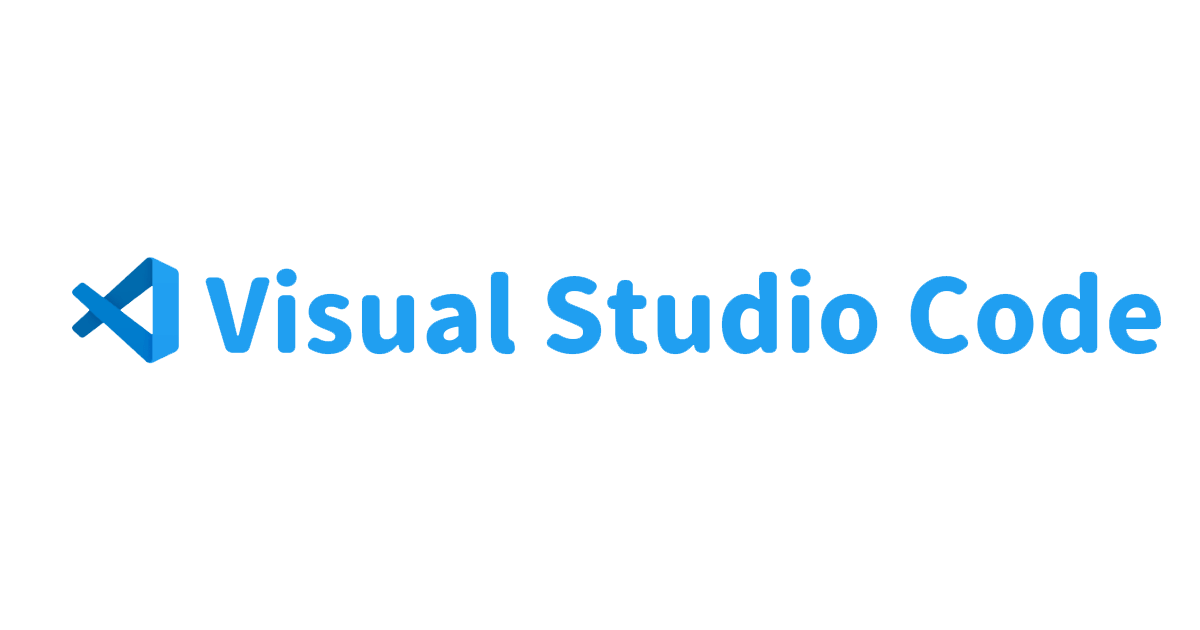
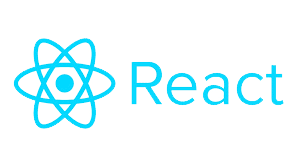
5. Maintenance:

* Monitor the website's performance, address any issues or bugs that arise, and apply regular updates and patches to maintain its functionality and security.
* Provide ongoing maintenance and support, including addressing user feedback, implementing new features, and ensuring the website remains up-to-date with evolving technologies and best practices.

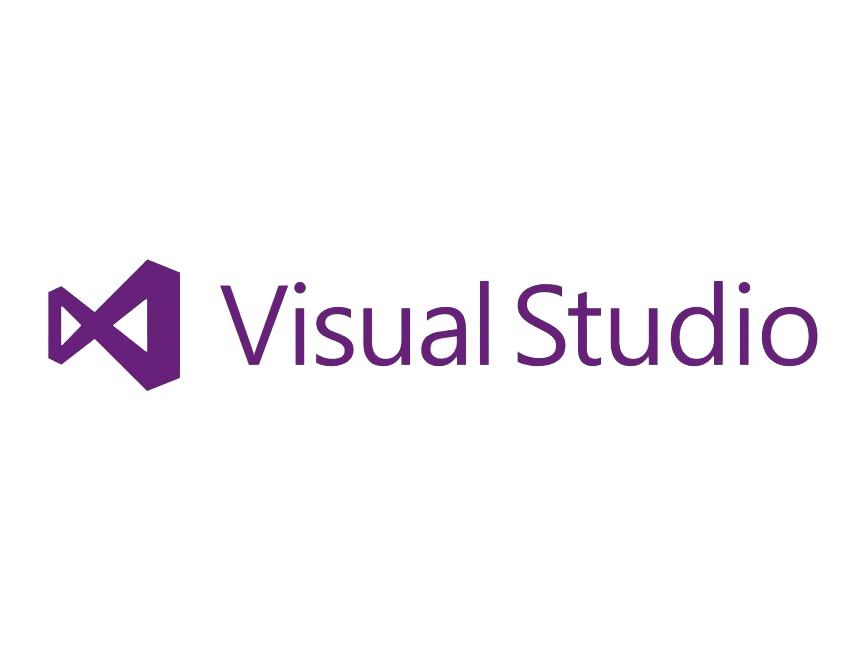
By following a systematic and iterative approach during the implementation phase, the website for connecting freelancers and clients can be developed and deployed successfully. It is important to collaborate closely with the development team, conduct regular testing, and prioritize user feedback to ensure the website meets the needs of both freelancers and clients.

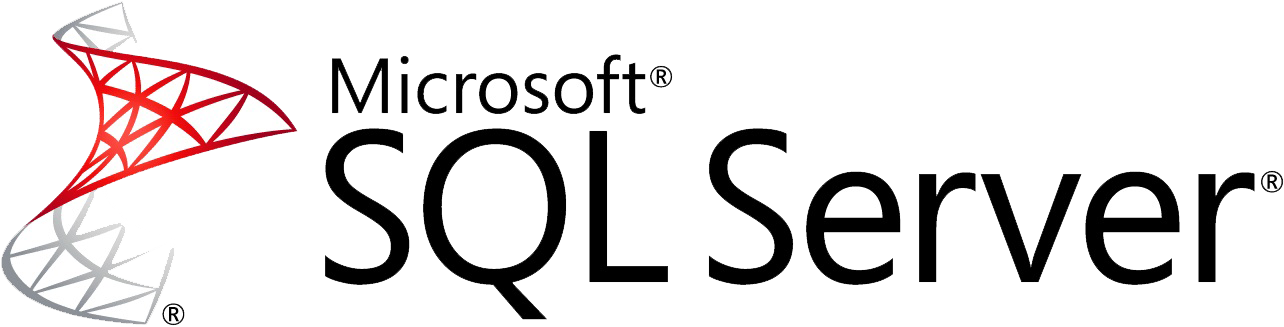
**4.2 Tools & Technologies**

* In Frontend design We used:

* + - * Visual Studio Code as a code editor
      * React js as a framework
* In backend design We used:

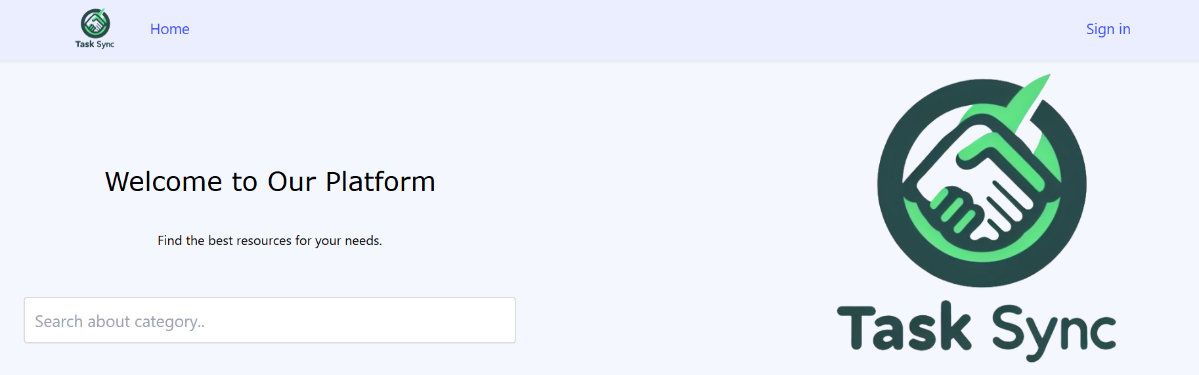
- Microsoft SQL Server: A relational database management system used for storing and managing the application's data.

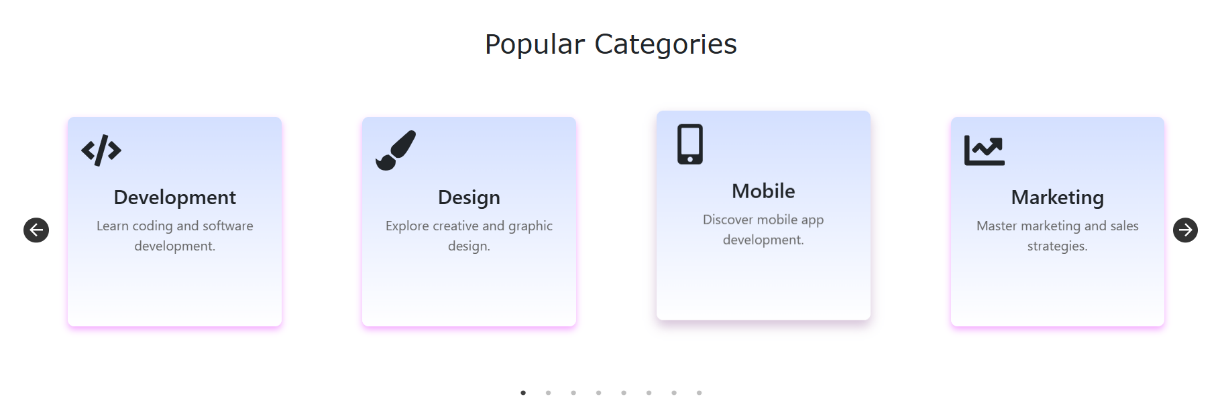
.NET Core: A cross-platform framework for building and running backend services and web applications.

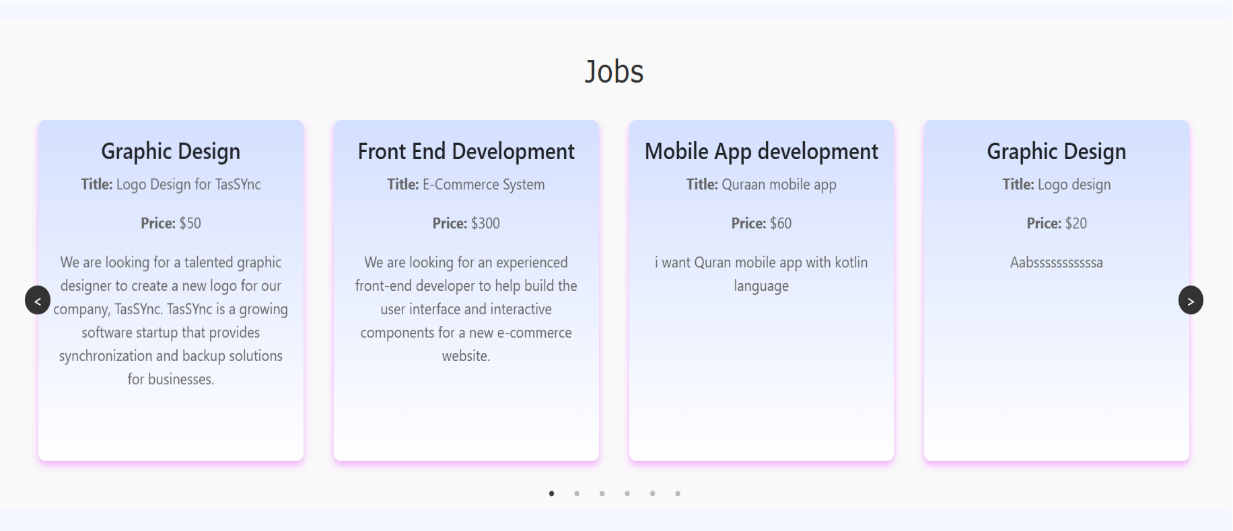
Swagger: An API testing tool used to develop, test, and debug APIs by sending HTTP requests and analyzing responses.

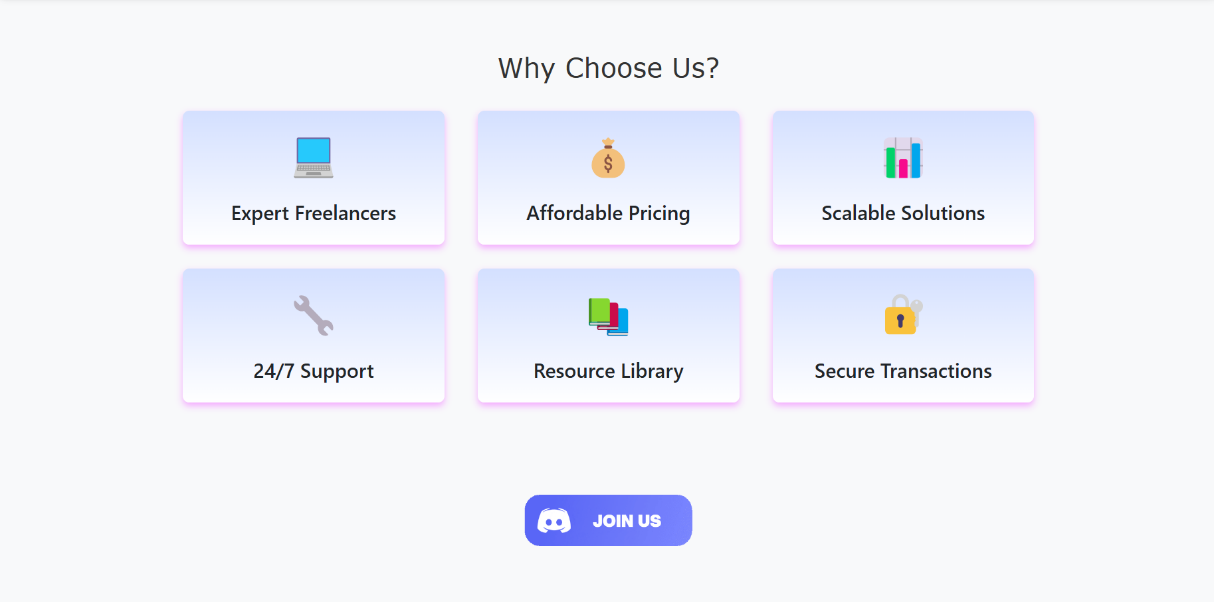
Microsoft Visual Studio: An integrated development environment (IDE) for writing, debugging, and managing .NET backend code and projects.

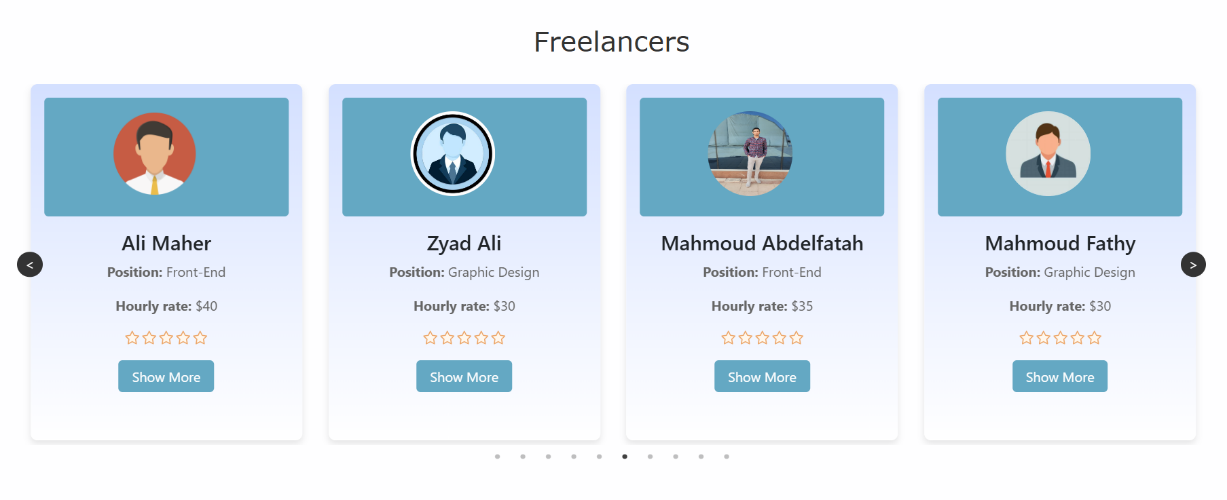
* 1. **Ui designs** 
     1. **Home page**



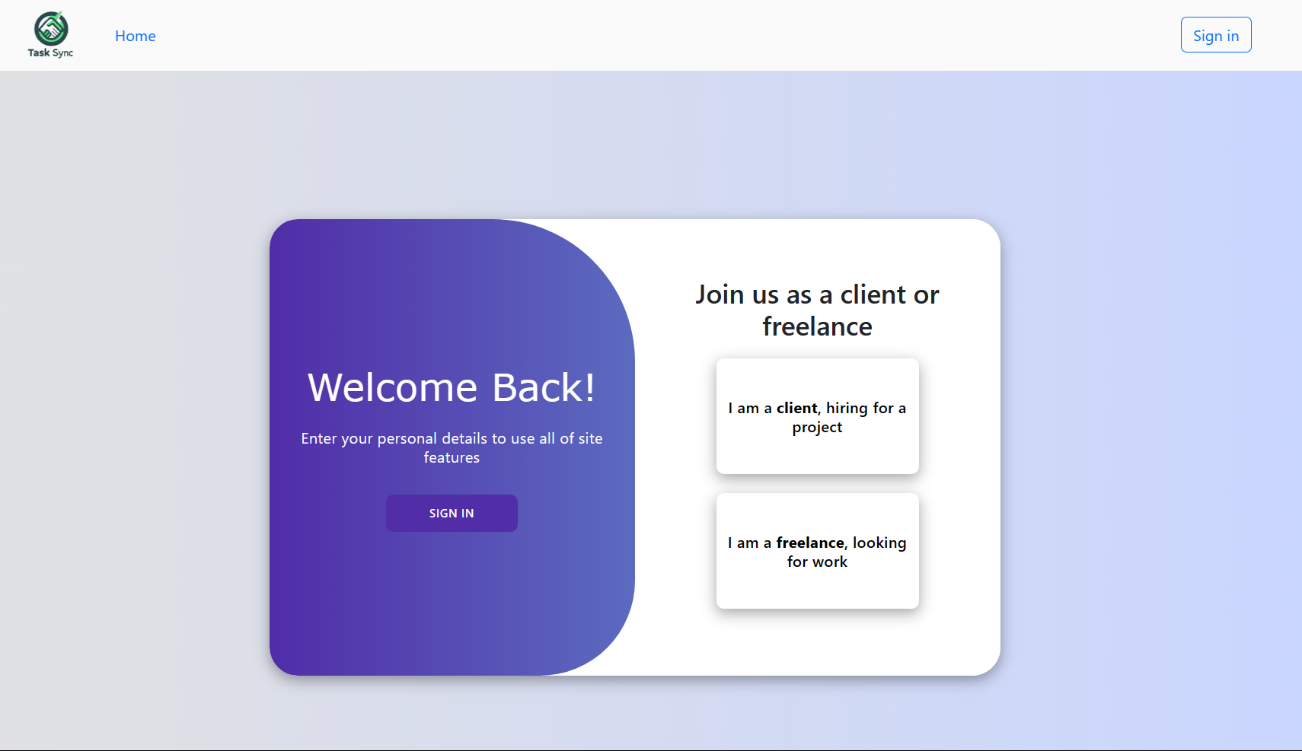




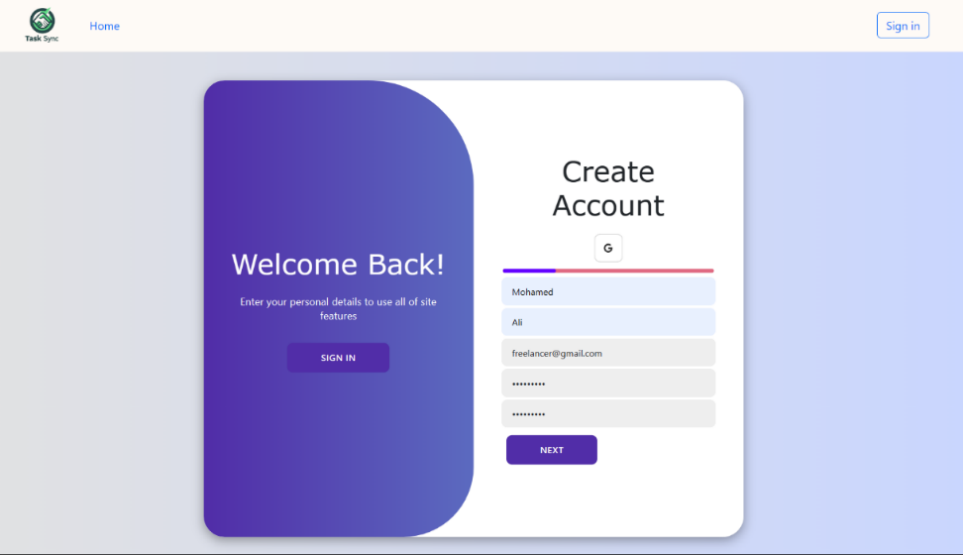


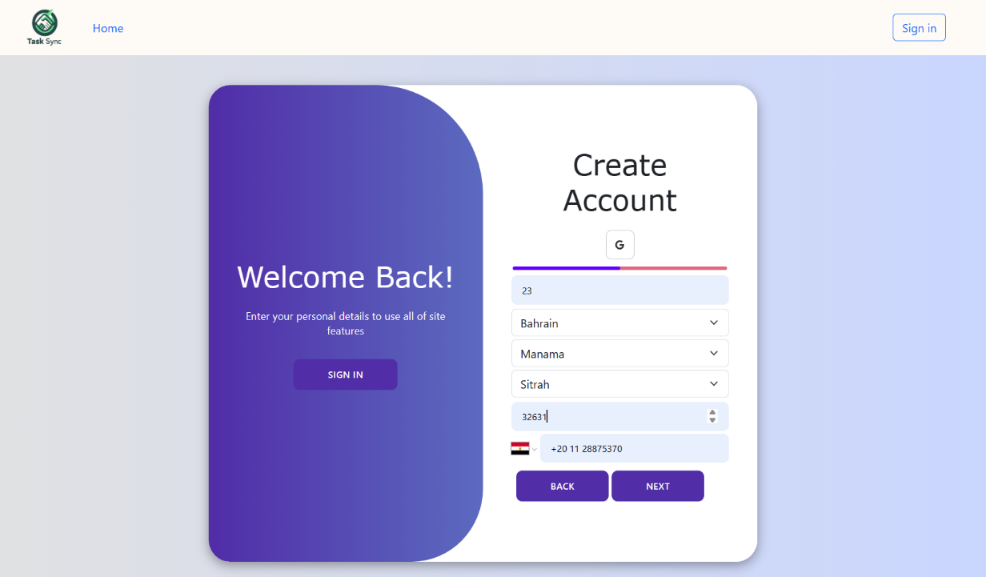


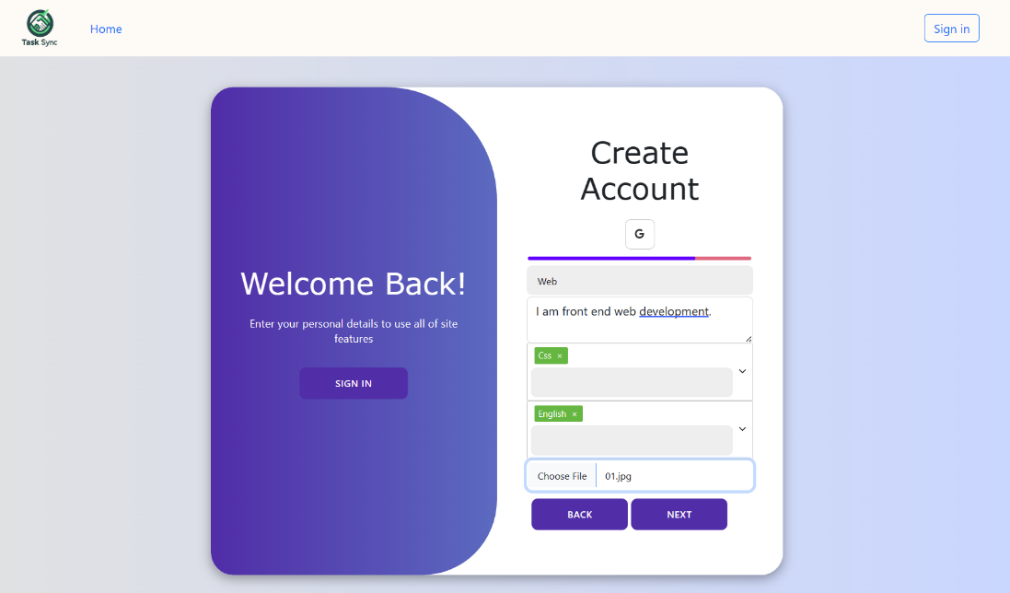
* + 1. Registration

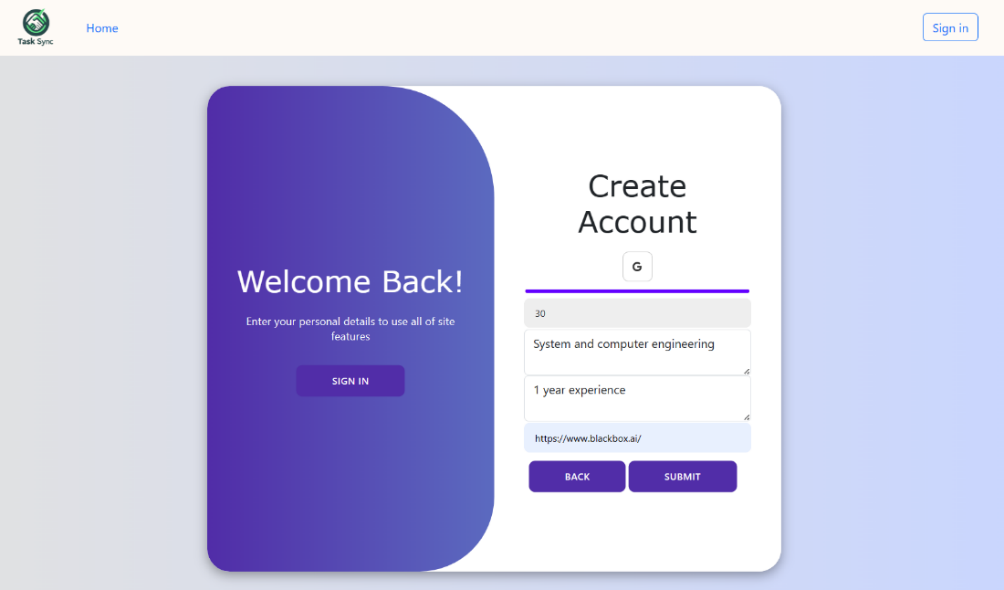


* + - 1. Freelancer Registration

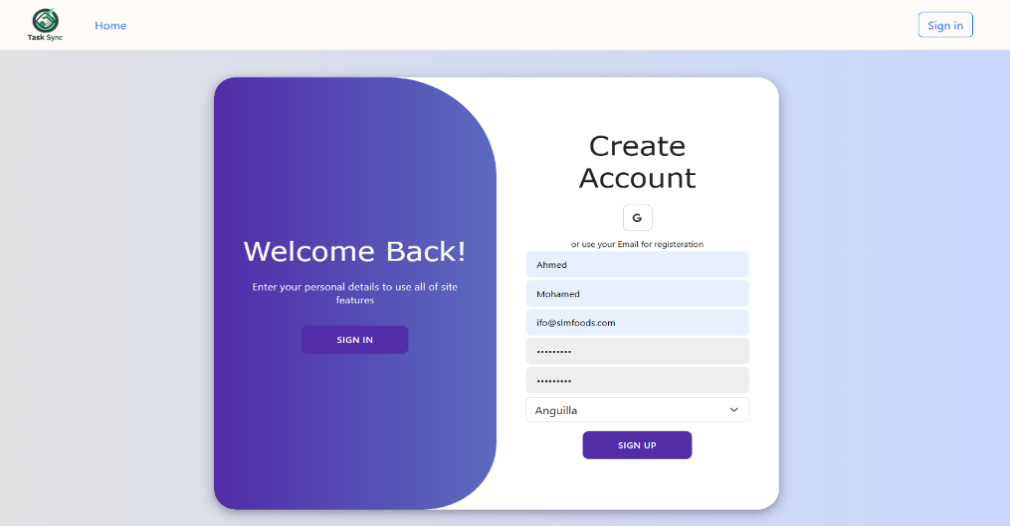




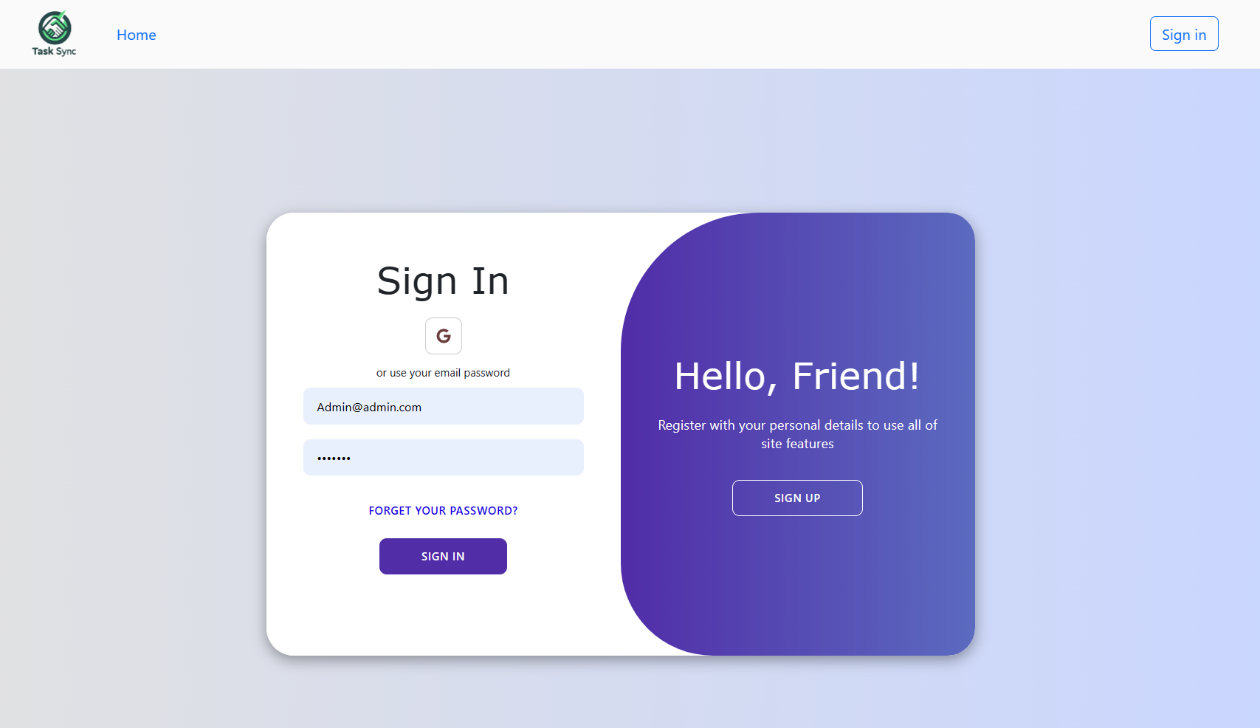




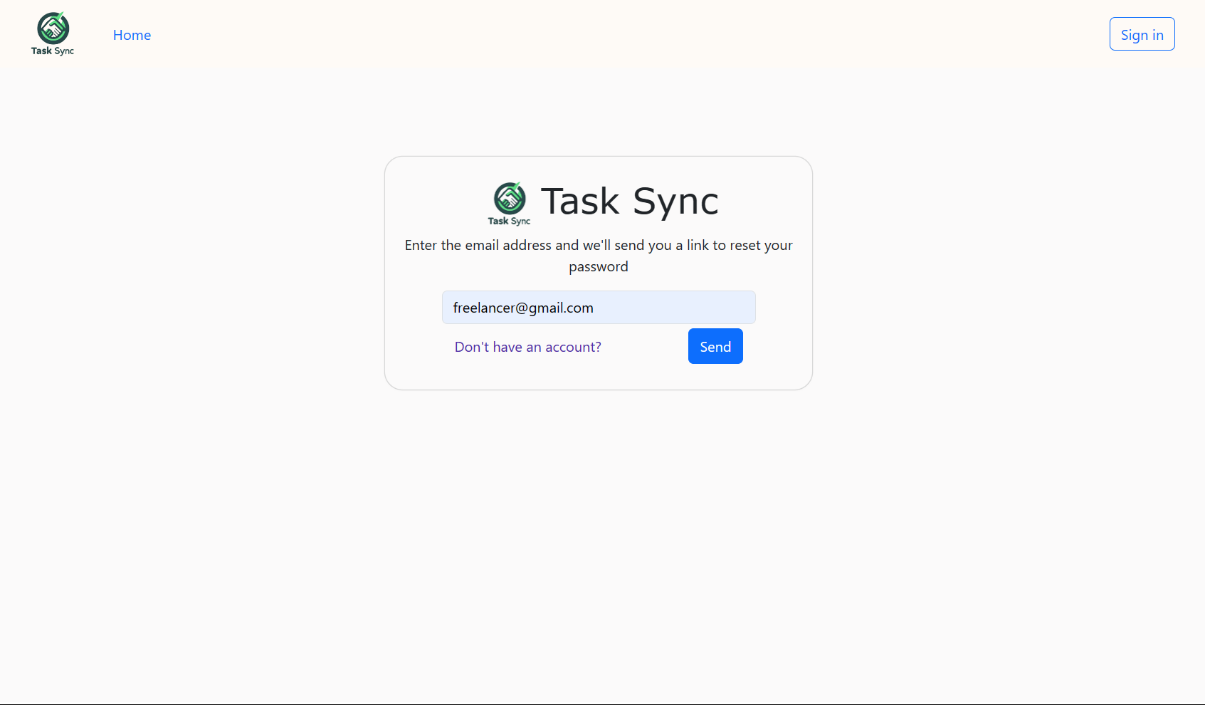
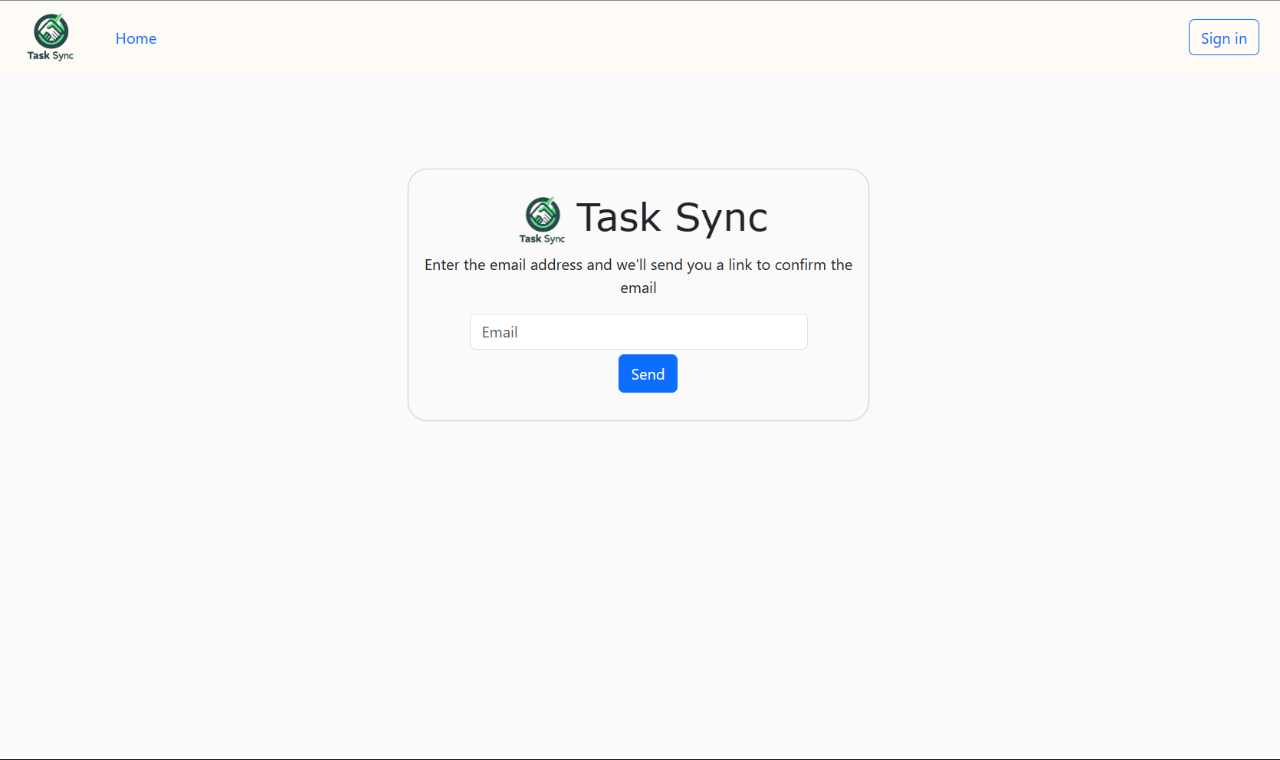
* + - 1. Client Registration



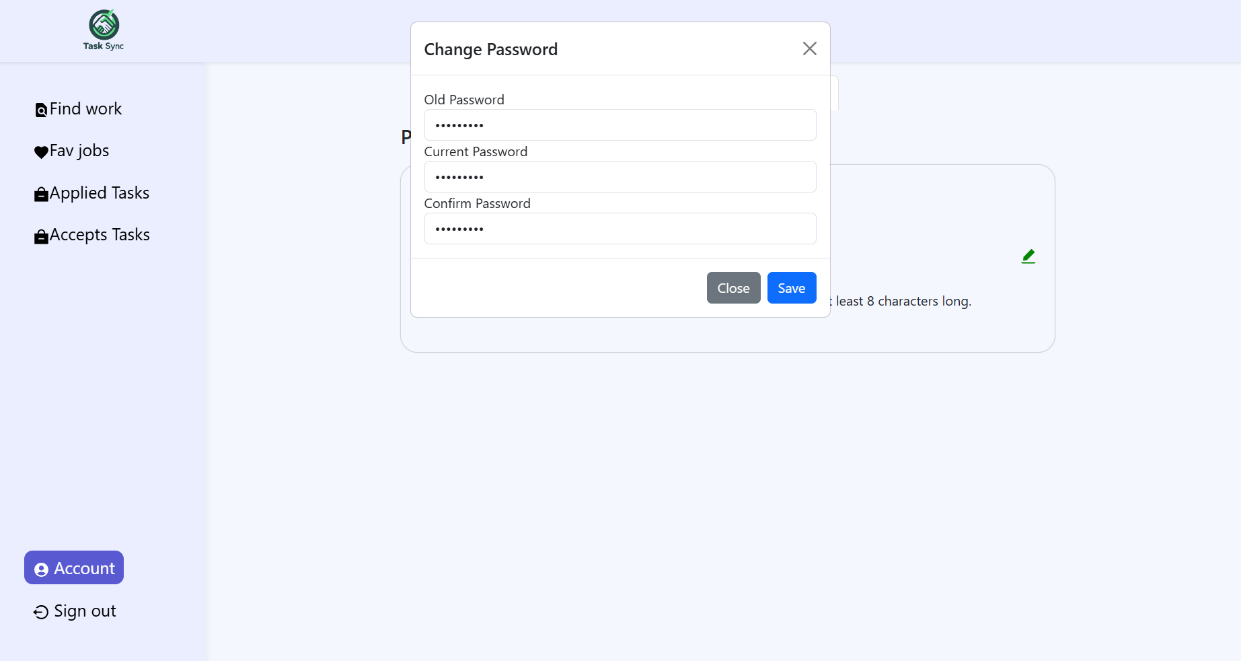
* + 1. Sign in



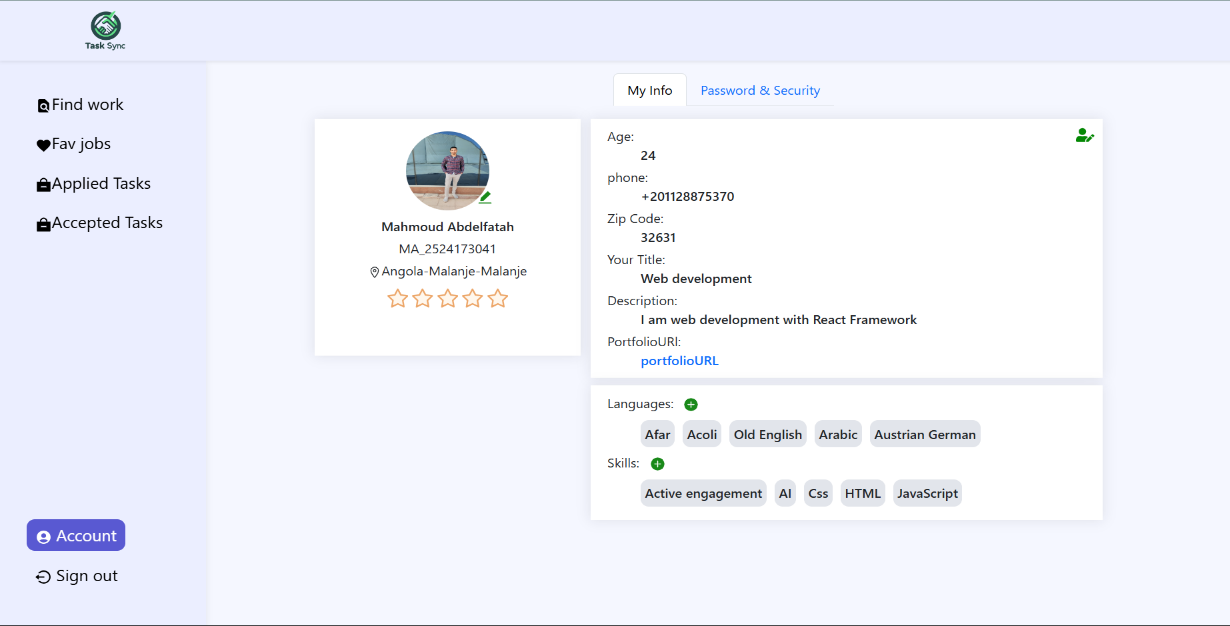
* + 1. Forget Password and Reconfirm email



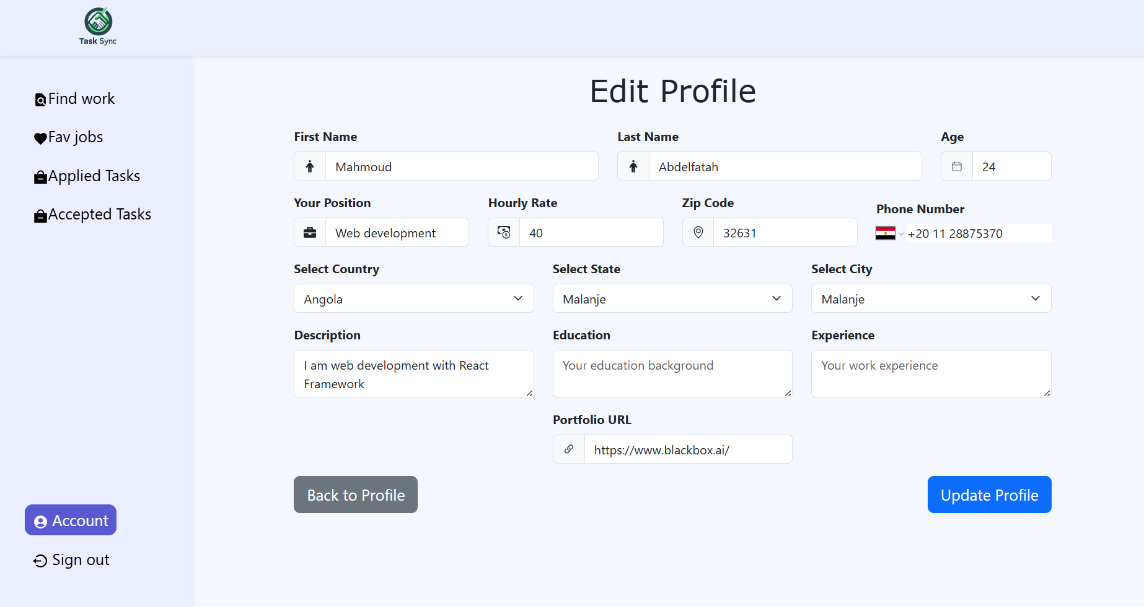
* + 1. Change Password



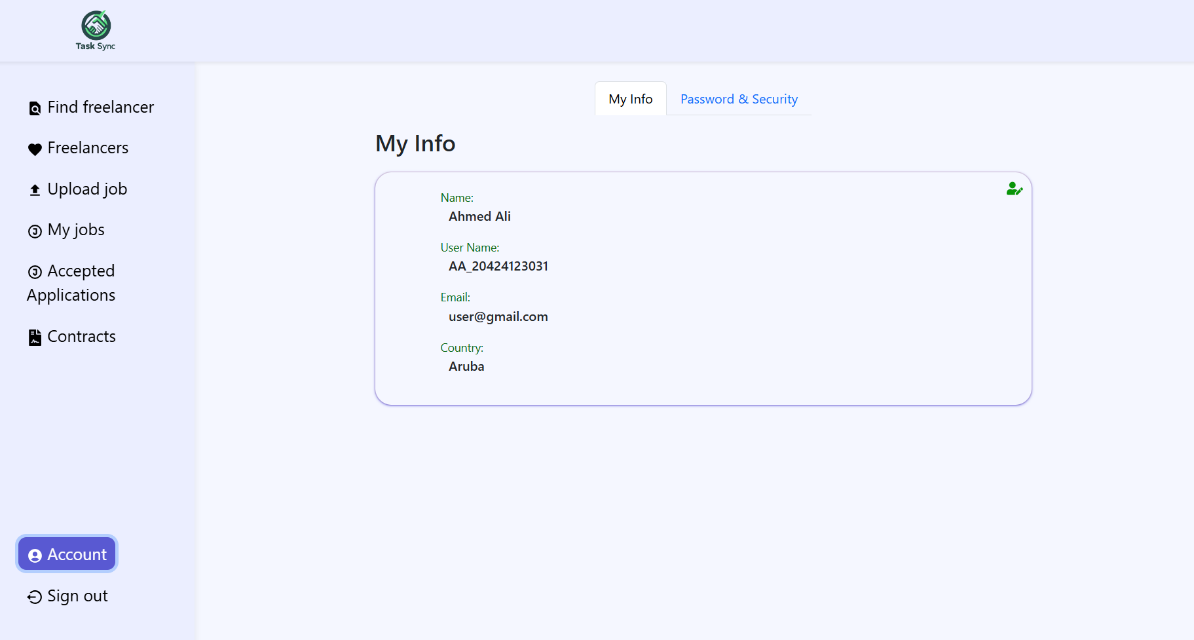
* + 1. Account
       1. Freelancer Account

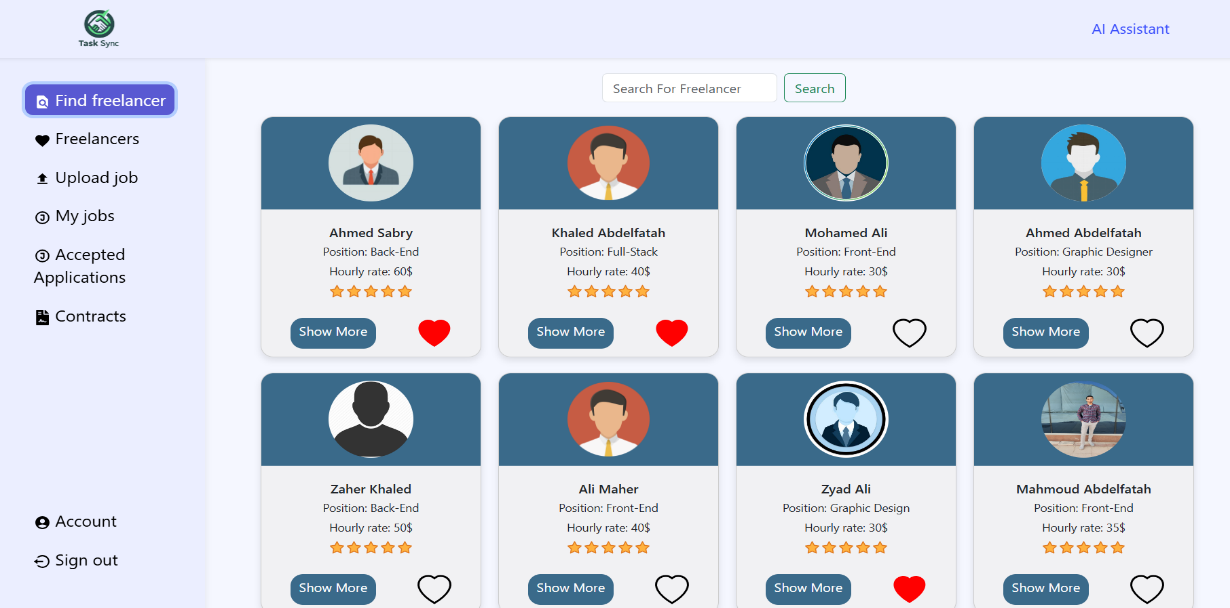


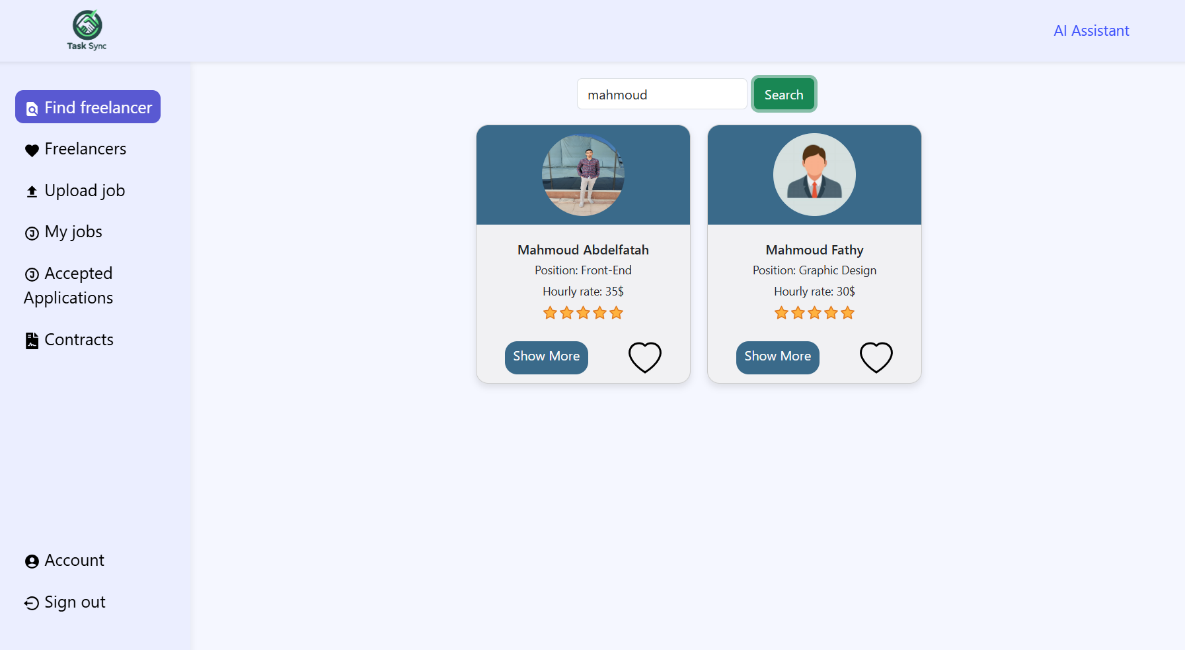
* + - 1. Update Freelancer Account

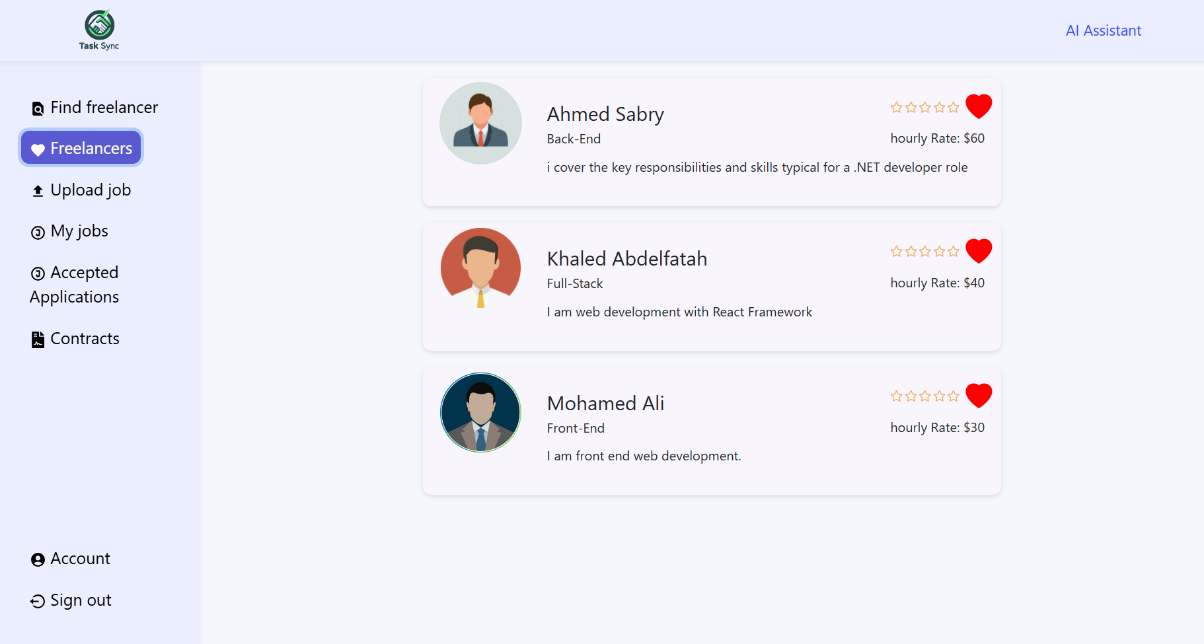


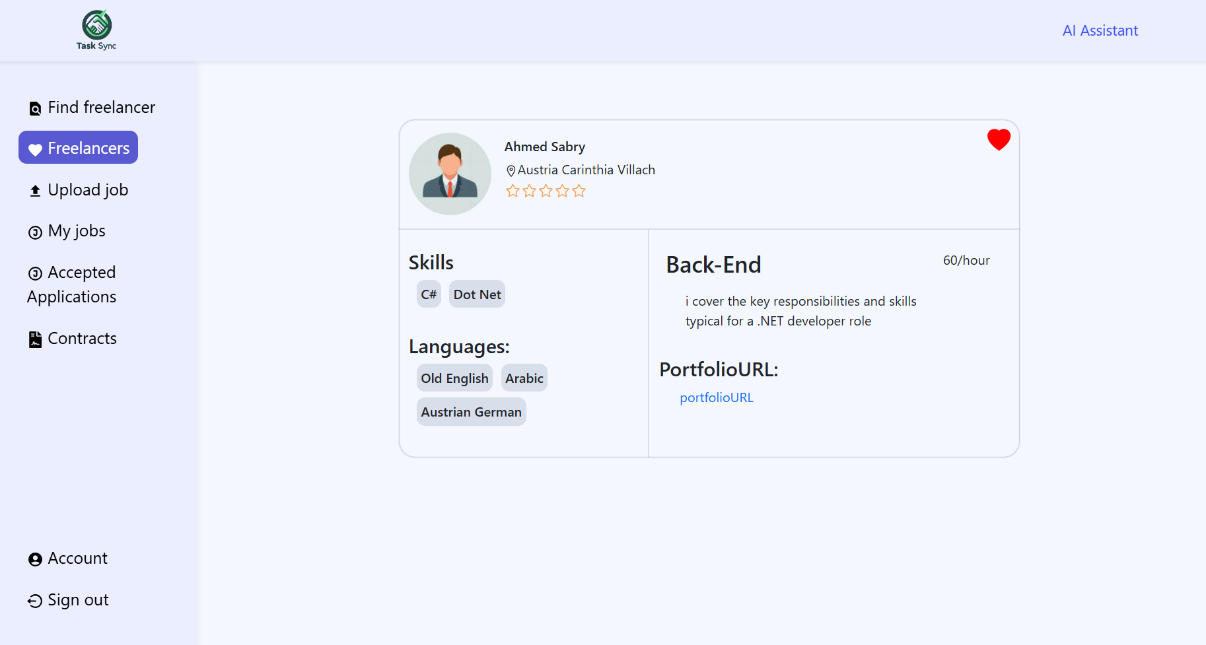
* + - 1. User Account



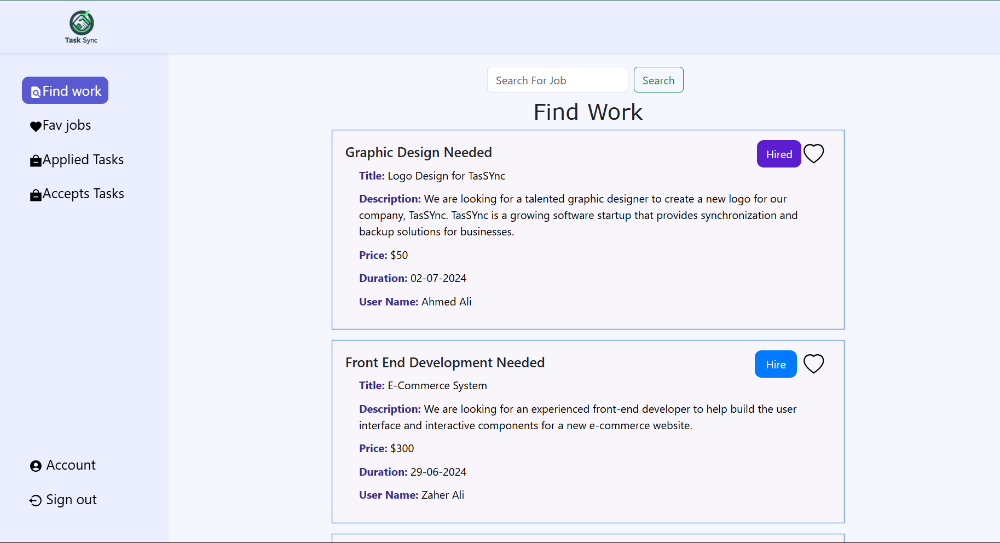




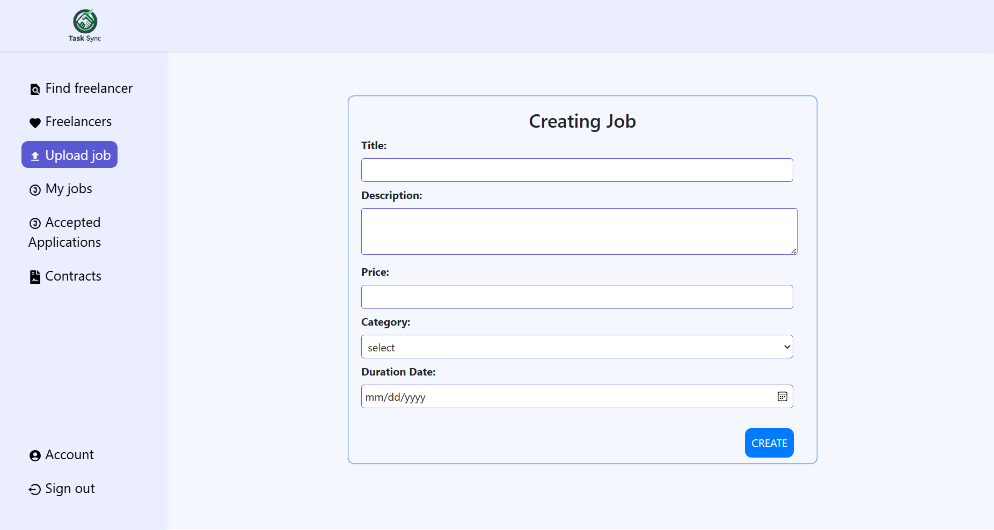




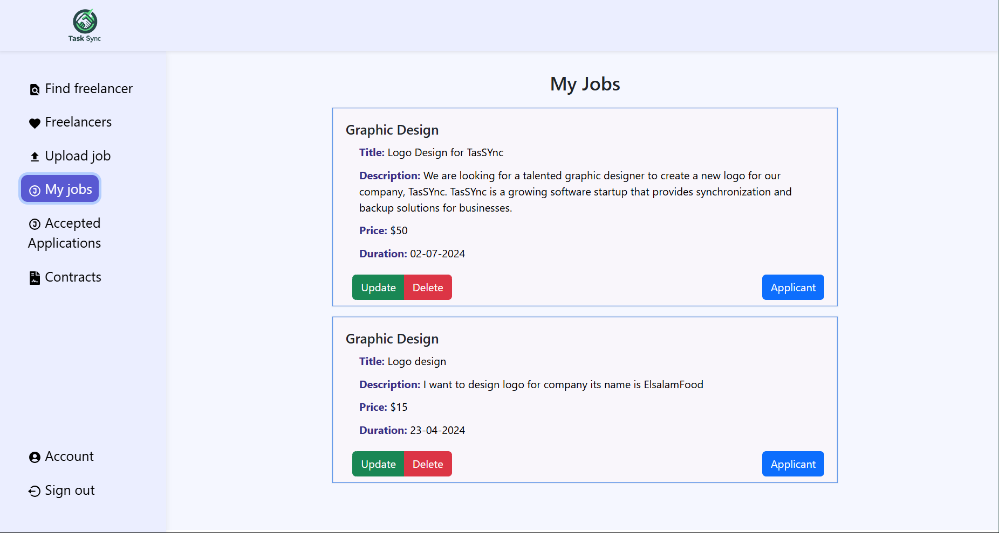
* + 1. Job Posts
       1. All job posts



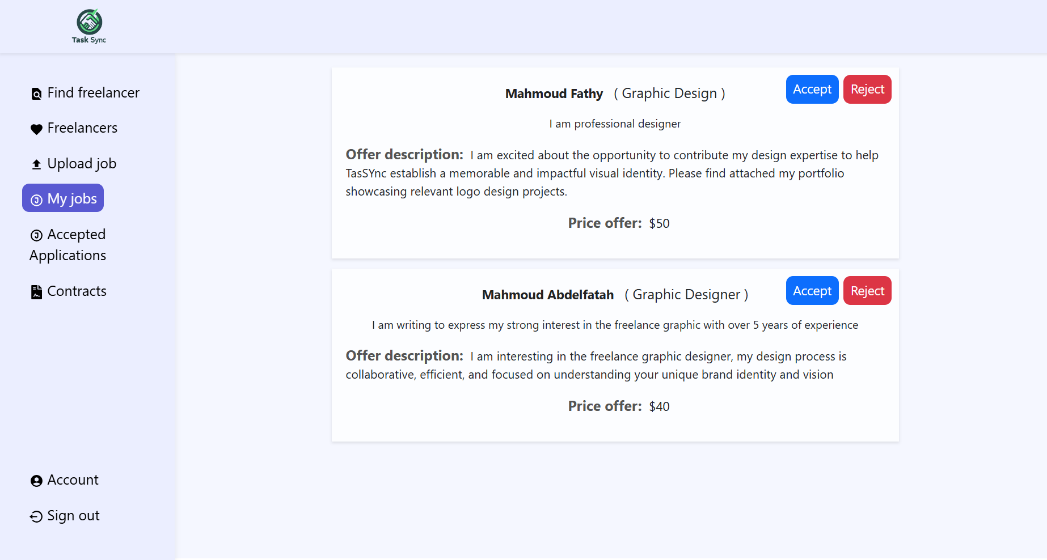
* + - 1. Upload job



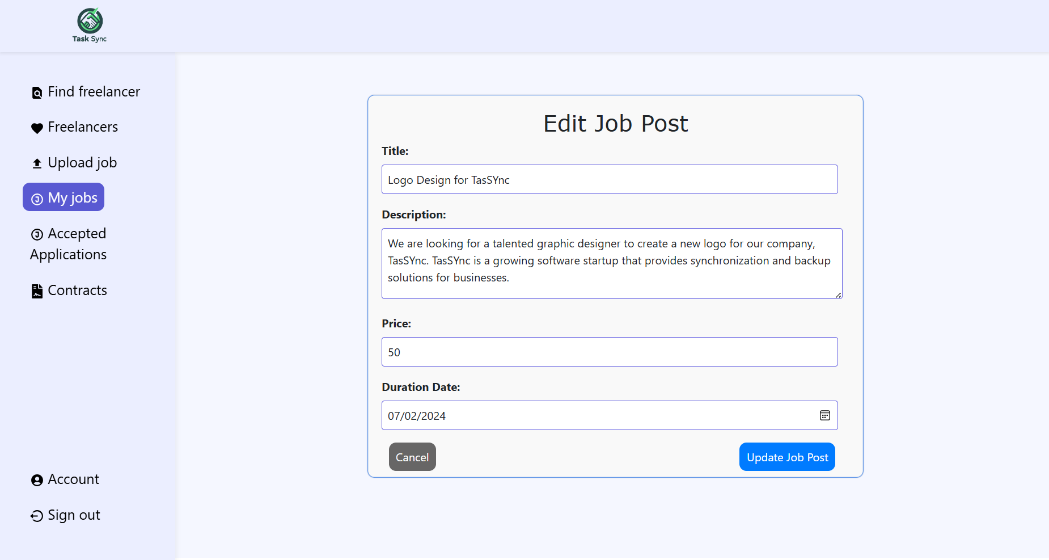
* + - 1. Client job posts



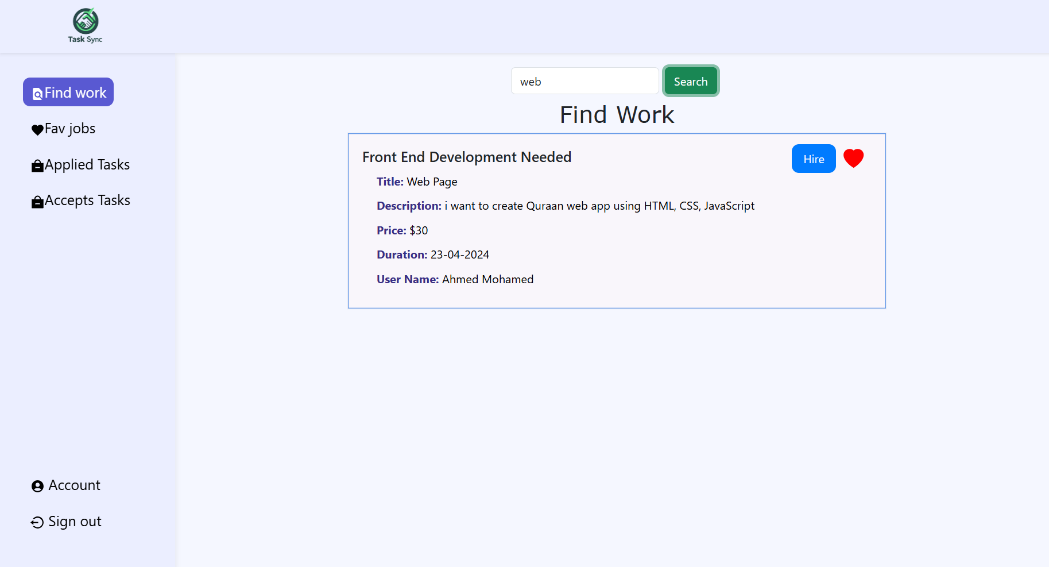
* + - 1. Applicants on a certain job



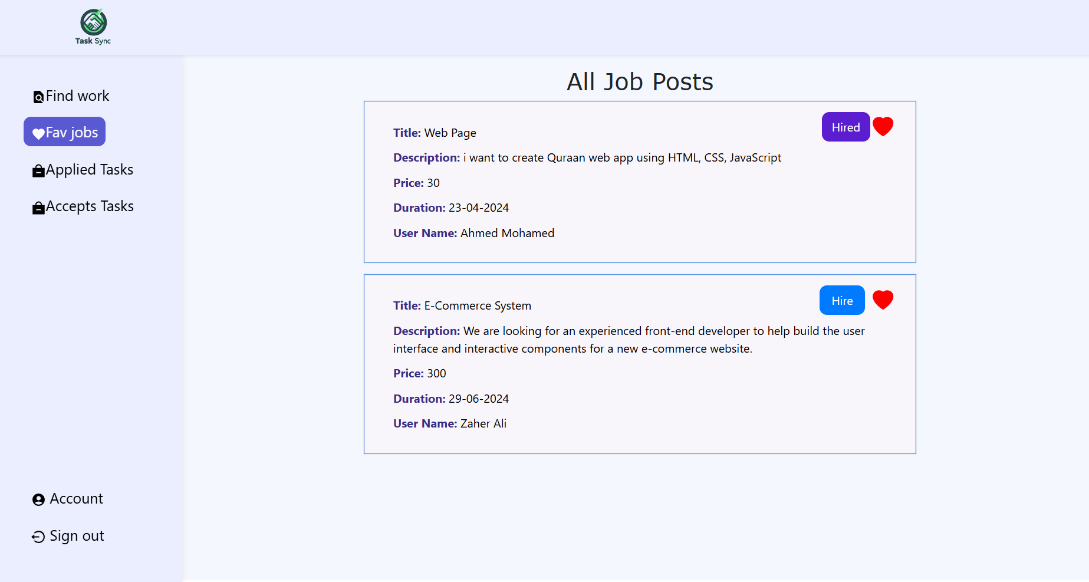
* + - 1. Edit job post



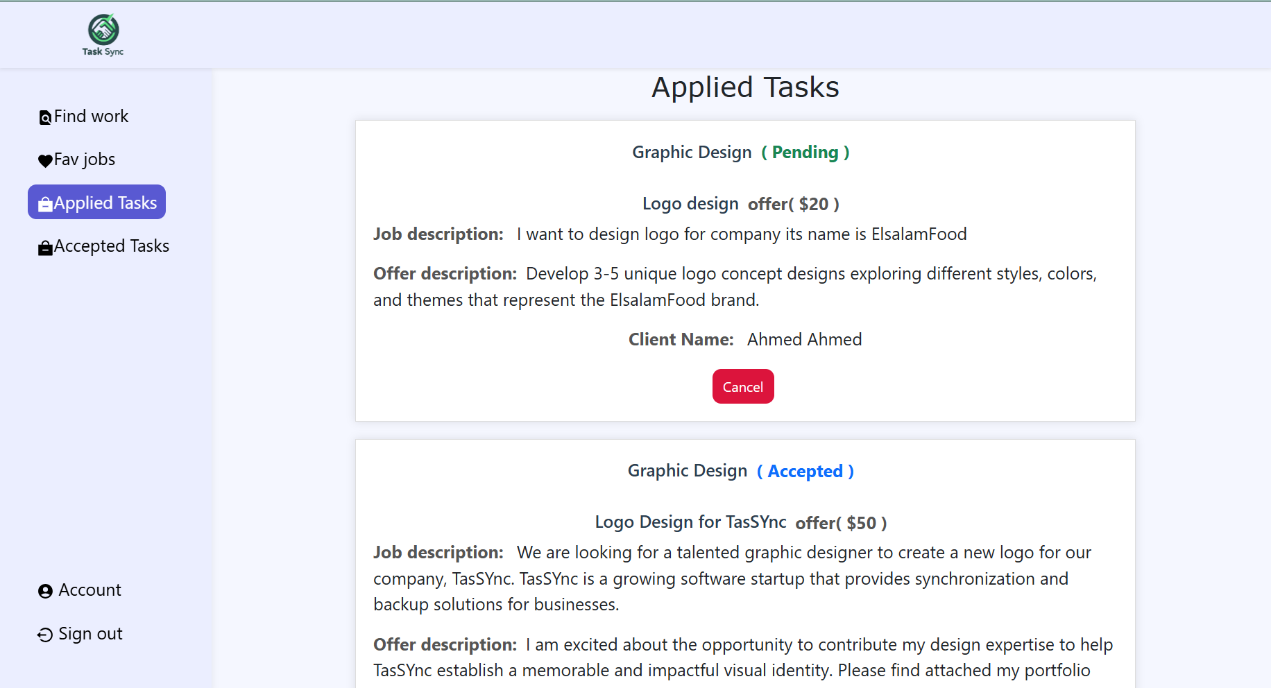
* + - 1. Search for job



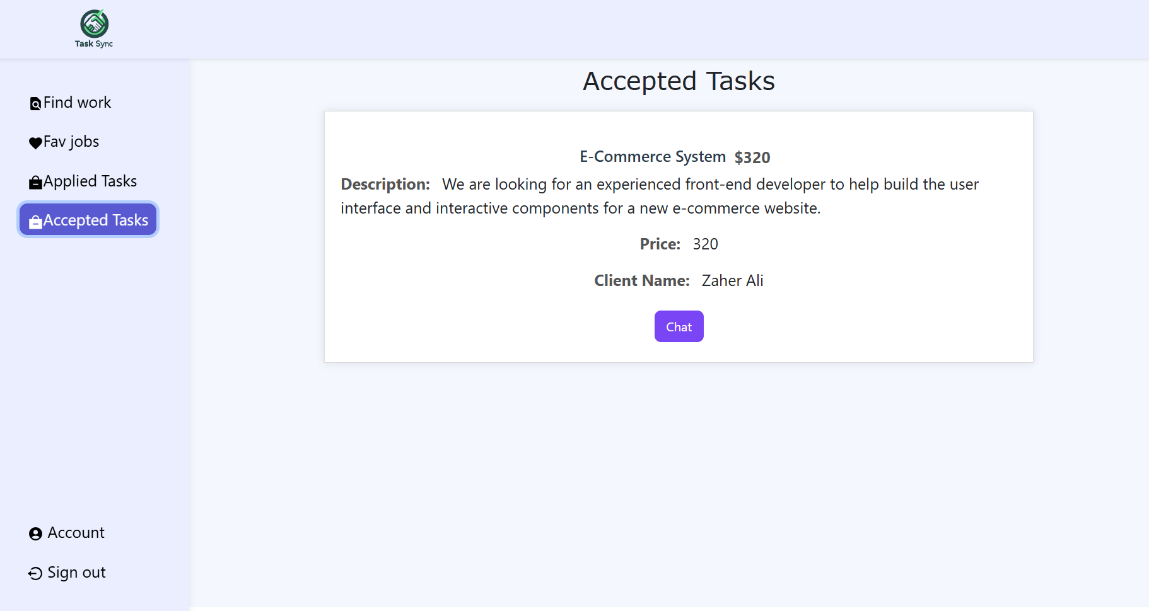
* + - 1. Freelancer Favorite jobs



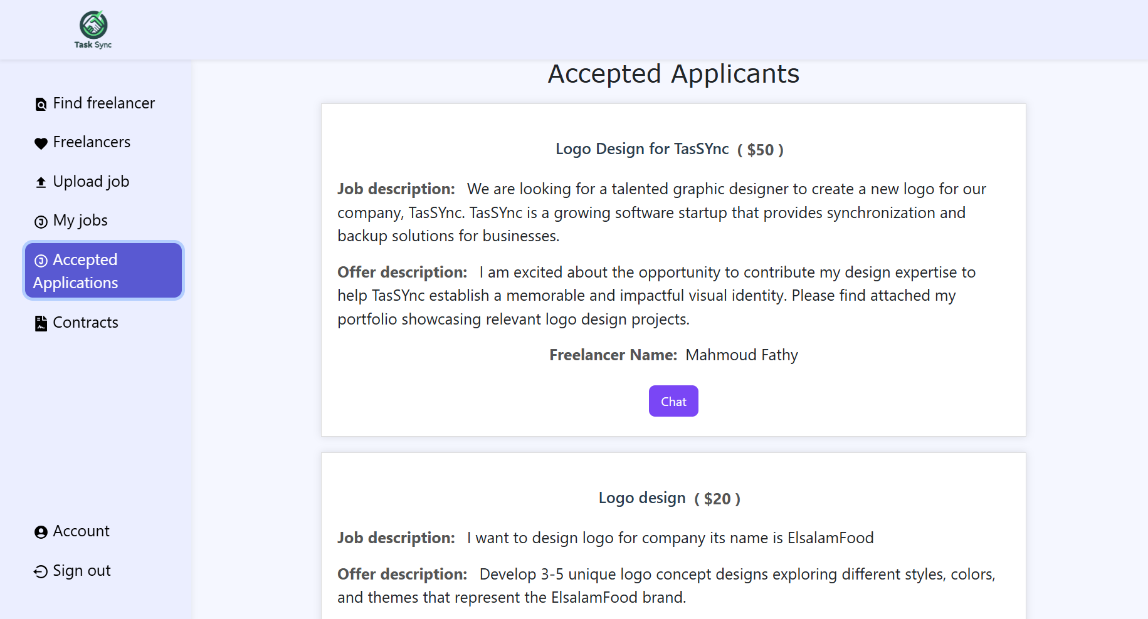
* + 1. Apply for task
       1. Freelancer applied tasks



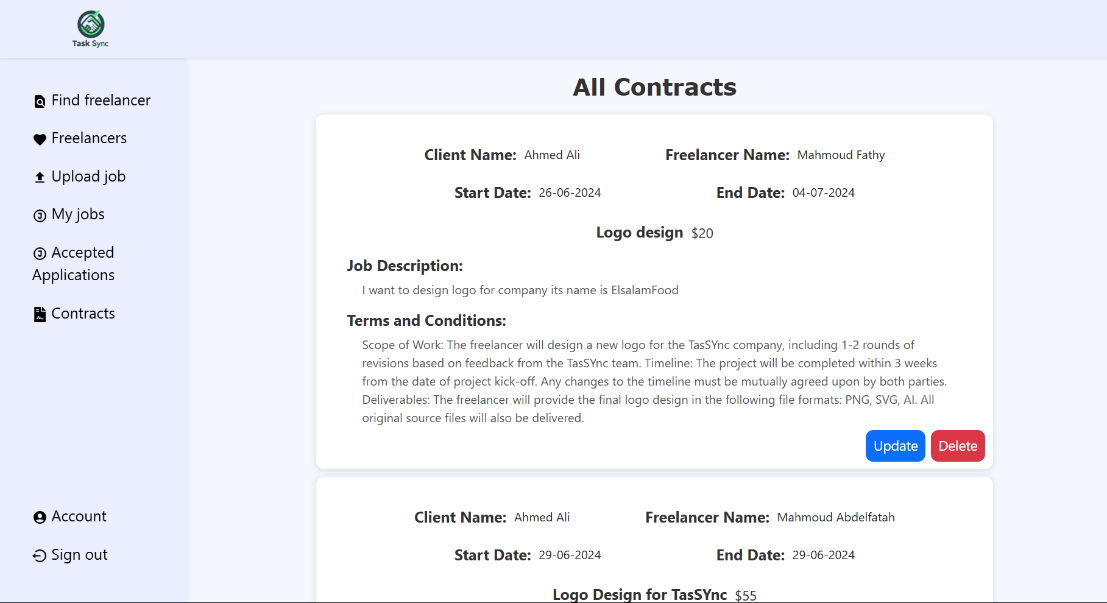
* + - 1. Freelancer accepted tasks



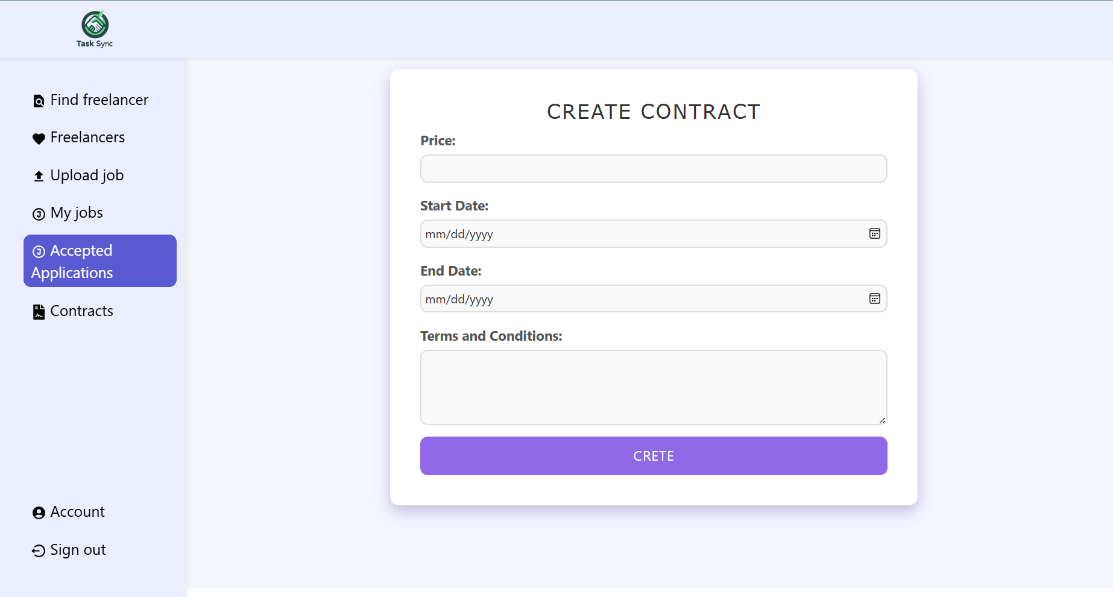
* + - 1. Client offers for a job



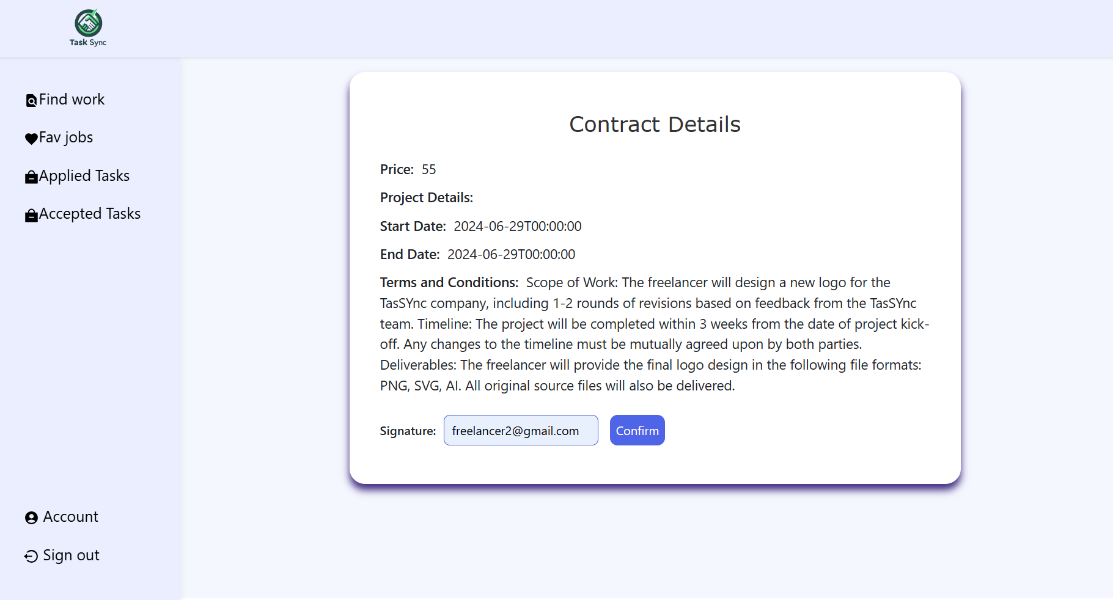
* + 1. Contract
       1. All contracts



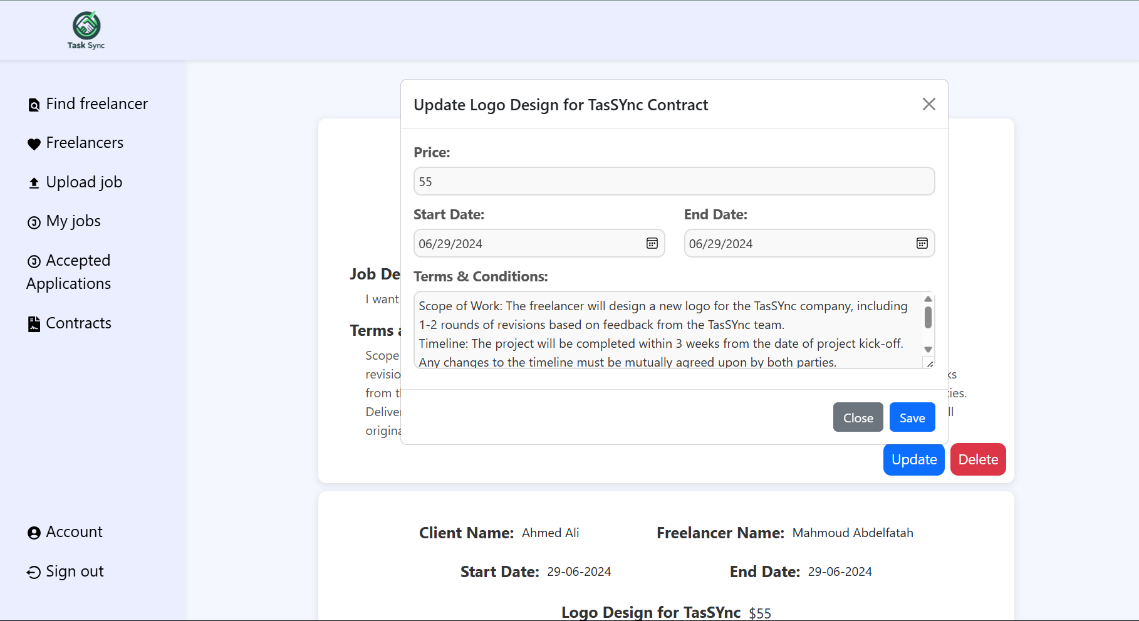
* + - 1. Create contract

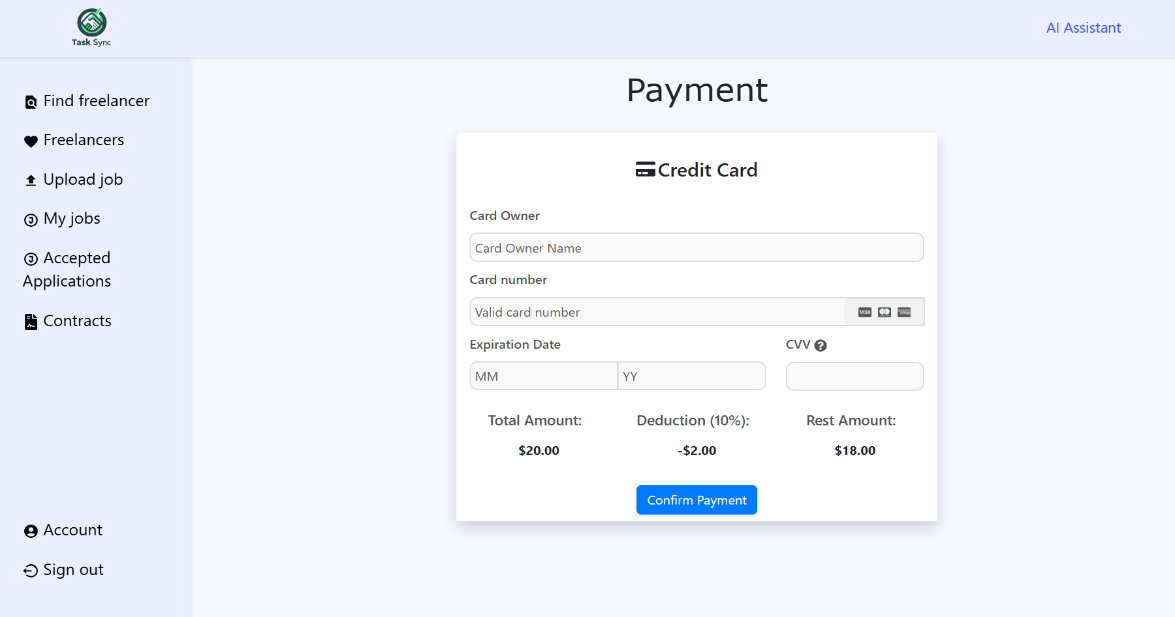


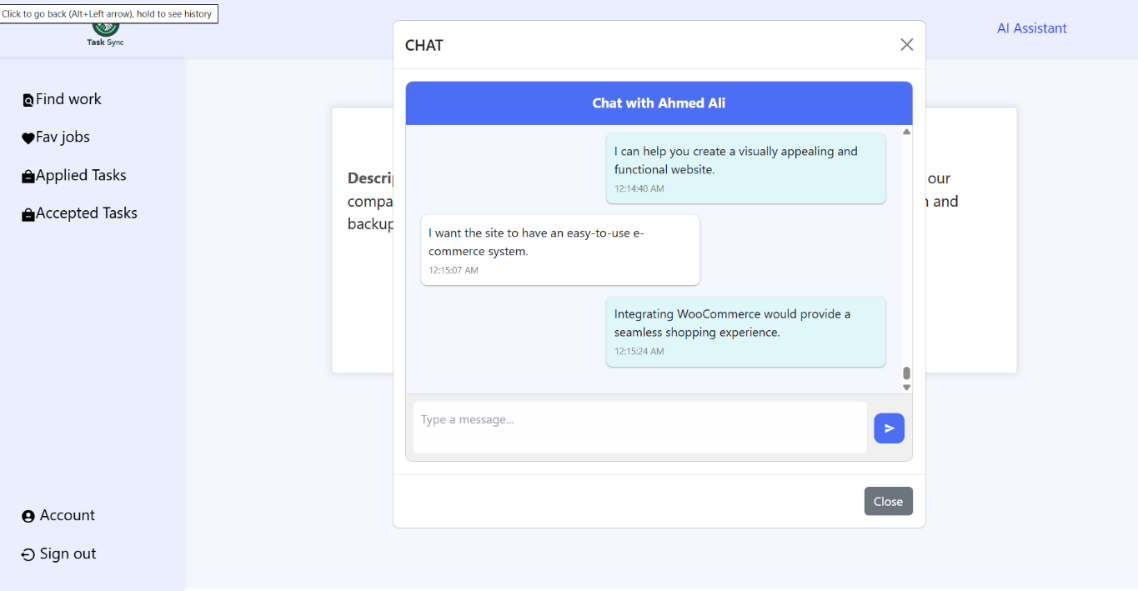
* + - 1. Confirm the contract



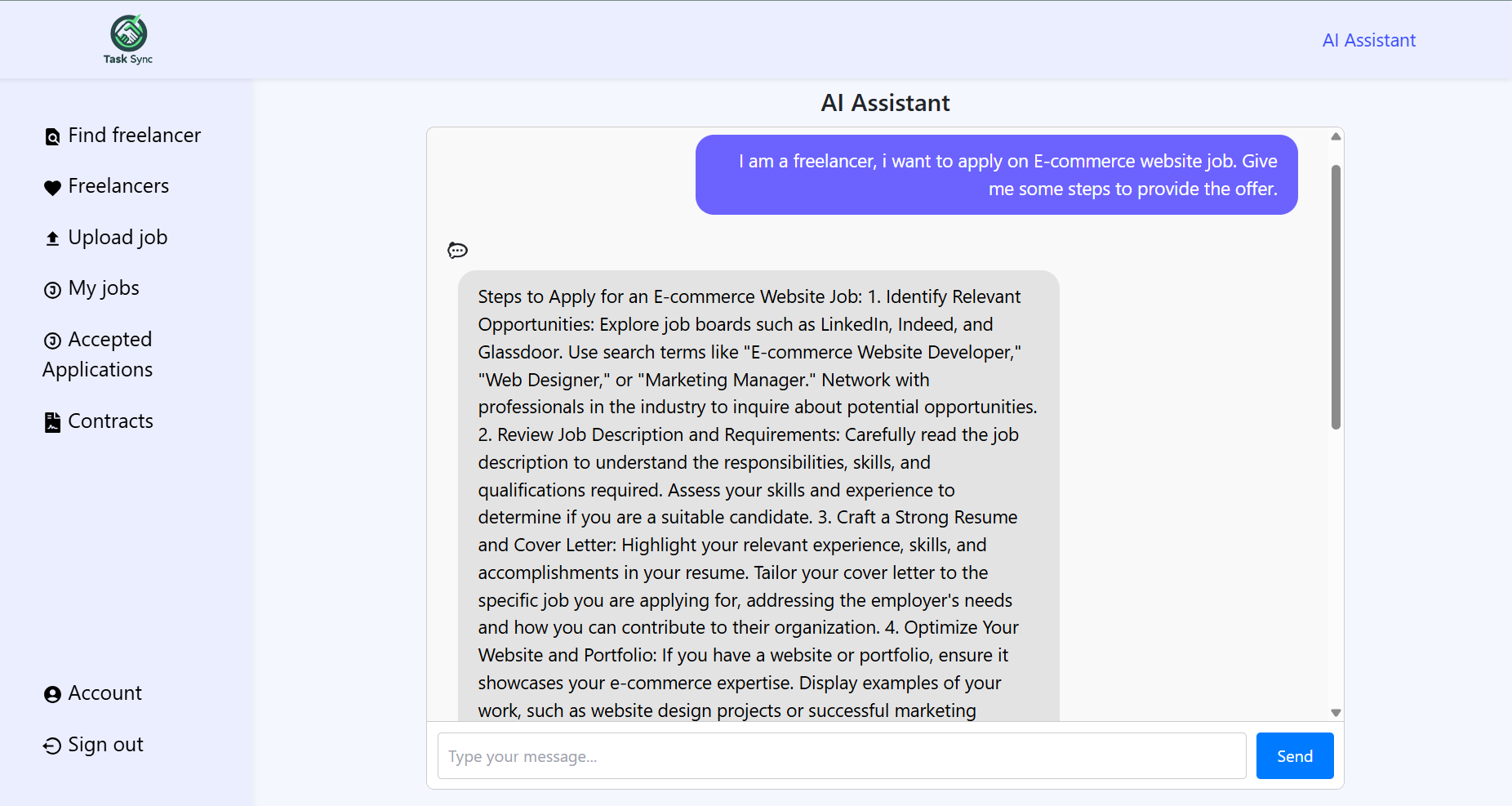
* + - 1. Update contract



* + 1. **Payment** 
    2. **Chat**



* + 1. **Ask AI**



* 1. **Dataset and Preprocessing**

For a freelance platform, we would typically need a dataset that includes information about freelancers, such as their skills, availability, and areas of expertise. Additionally, you would need client data, including their project requirements, budget, and specific needs.

Building such a dataset can involve different approaches. You could collect data through surveys or interviews with freelancers and clients, or you could collaborate with freelance platforms to obtain anonymized data. It's important to ensure that any data collection process adheres to privacy regulations and obtains proper consent from participants.

Once you have collected the dataset, preprocessing is necessary to clean and prepare the data for analysis and modeling. This may involve steps such as removing duplicates, handling missing values, standardizing formats, and encoding categorical variables. Preprocessing techniques can vary depending on the specific requirements of your project and the characteristics of the dataset.

Some common preprocessing steps for a project like this may include:

* + 1. **Data cleaning**: Removing any irrelevant or duplicate entries, handling missing values, and correcting inconsistencies in the data.

* + 1. **Feature engineering**: Creating new features or transforming existing ones to extract meaningful information. For example, you could derive features like freelancer availability based on their working hours or calculate the distance between clients and freelancers based on their locations.

* + 1. **Data normalization**: Scaling numerical features to a common range to avoid bias towards certain variables during analysis.

* + 1. **Encoding categorical variables**: Converting categorical variables, such as freelancer skills or client project types, into numerical representations that can be used in machine learning models.

* + 1. **Splitting the dataset**: Dividing the dataset into training, validation, and testing sets to evaluate the performance of the implemented system.
  1. **Source code**

Backend: https://github.com/A7medSabri/FinalProject

Frontend: https://github.com/7oda-119/Final-Project

**CHAPTER 5**

CONCLUSION AND FUTURE WORK

**5.1 Conclusion**

In conclusion, this software engineering project has successfully achieved its objectives of developing a user-friendly and efficient freelance platform. Through the implementation of robust software engineering practices, we have created a reliable and scalable solution that meets the needs of our users. The project has demonstrated our strong understanding of software engineering concepts and our ability to analyze and solve complex problems.

**5.2 Future Work**

Adding specialized services under the guidance of industry experts.

Using AI to facilitate the matching of freelancers to client projects and improve response times for user interactions.

While the current version of the platform is functional and meets the initial requirements, there are several areas that can be further improved and expanded upon in future iterations. Some potential areas for future work include:

1. Performance Optimization: Conducting performance profiling and identifying bottlenecks to optimize the platform's speed and responsiveness. This could involve implementing caching mechanisms, optimizing database queries, or utilizing more efficient algorithms.
2. Enhanced Security Measures: Strengthening the platform's security by implementing additional authentication and authorization mechanisms, as well as conducting regular security audits to identify and address any vulnerabilities.
3. Integration with Third-Party APIs: Integrating the platform with popular third-party APIs to enhance its functionality and provide users with additional features. This could include integrating with payment gateways, social media platforms, or other relevant services.
4. User Interface Enhancements: Gathering user feedback and conducting usability testing to improve the platform's user interface and overall user experience. This could involve refining the design, streamlining workflows, and implementing intuitive navigation.
5. Localization and Internationalization: Adapting the platform to support multiple languages and cultural preferences to cater to a global user base. This could involve implementing localization frameworks and providing translation support.
6. Continuous Integration and Deployment: Setting up a robust CI/CD pipeline to automate the build and deployment processes. This would ensure faster and more reliable releases, as well as facilitate collaboration among team members.
7. By addressing these areas of future work, we can further enhance the platform's functionality, performance, security, and user experience. This project has provided a strong foundation, and we are committed to continuously improving and expanding upon it to meet the evolving needs of our users.

**References**

**Microsoft Documentation**:

* [ASP.NET Core Documentation](https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-5.0): Official documentation for learning and reference.

**Tutorials**:

* [Build a web API with ASP.NET Core MVC](https://docs.microsoft.com/en-us/learn/modules/build-web-api-aspnet-core/): Step-by-step tutorial on building a web API.
* CRUD Operations in ASP.NET Core: Guide to implementing CRUD operations.

**Books**:

* *Pro ASP.NET Core 3* by Adam Freeman: A comprehensive guide covering ASP.NET Core fundamentals and advanced topics.
* *ASP.NET Core in Action* by Andrew Lock: Another great resource for learning ASP.NET Core from basics to advanced features.

**Community Resources**:

* [Stack Overflow](https://stackoverflow.com/questions/tagged/asp.net-core): A valuable resource for getting help with specific issues.
* [ASP.NET Core GitHub Repository](https://github.com/dotnet/aspnetcore): Check out the official repository for examples and community discussions.

**Sample Projects**:

* [eShopOnWeb](https://github.com/dotnet-architecture/eShopOnWeb): A sample ASP.NET Core reference application, powered by Microsoft, showcasing architectural patterns and techniques.

**APPENDICES**

**Timeline/Milestones (Gantt Chart)**

|  |
| --- |
|  |

