FineAnnotator

Software Requirement Specification (SRS)

Version: 1.0

Collaborators

Nazmun Nahar Tui - **2018331047**
Nishat Rahman - **2018331071**
Nafis Fuad Abir - **2018331087**
Mirza Nihal Baig - **2018331095**
Nazmus Sadat - **2018331099**

1. Introduction

FineAnnotator is a platform that connects researchers and companies with data annotation jobs to students seeking flexible part-time work. It aims to:

- For Researchers and Companies: Provide a platform to find a large workforce of students for data annotation tasks.
- For Students: Offer a flexible and rewarding way to earn money by annotating data.

1.1 General Objectives

1. Facilitate Collaboration

- **Connect Researchers and Annotators:** Provide a user-friendly platform that efficiently connects Researchers/Companies needing data annotations with Students/Annotators seeking part-time job opportunities.

2. Enhance Accessibility and Flexibility

- **Accessible Job Listings:** Enable Researchers/Companies to post diverse annotation tasks across different data types (Text, Images, etc.) to accommodate a wide range of projects.
- **Flexible Opportunities:** Offer flexible part-time job opportunities for Students/Annotators, allowing them to select tasks based on their preferences, such as payment, work hours, and deadlines.

3. Ensure Quality Annotations

- **Quality Assurance:** Ensure the delivery of accurate, high-quality annotated data to Researchers/Companies by implementing rating systems, recommendations, and verification processes for Annotators.
- **Clarity and Communication:** To reduce ambiguity and enhance task understanding, foster clear communication channels between Researchers/Companies and Annotators.

2.1 Scope

The scope of FineAnnotator encompasses a comprehensive platform designed to connect Researchers/Companies in need of data annotations with Students/Annotators seeking flexible part-time job opportunities. The software will facilitate job postings across diverse categories like Text, Images, and potentially other data types, allowing Researchers/Companies

to specify annotator requirements and payment structures. Annotators can create detailed profiles, apply for listed jobs based on their preferences, and interact with Researchers/Companies for task clarifications. The platform will ensure secure payment handling and privacy controls, with administrators overseeing account management, dispute resolution, and scam prevention. An intuitive user interface will ensure ease of use for all stakeholders, with scalability considerations for future growth and potential enhancements to accommodate evolving user needs. Overall, FineAnnotator aims to create a trustworthy, efficient, and user-friendly environment for data annotation tasks while fostering collaboration between Researchers/Companies and Students/Annotators.

2.2 Stakeholders

- **Researchers/Companies:** Initiators of data annotation tasks, posting requirements, and payments. They seek accurate annotations for their projects and contribute tasks to the platform.
- **Students/Annotators:** Perform data annotation tasks. They seek flexible, part-time job opportunities, apply for tasks, and contribute to the completion of annotations.
- **Administrators/Moderators:** Oversee platform activities, handle disputes, ensure security, and maintain the platform's integrity for all users' safety and satisfaction.
- **Technology Team/Developers:** Build, maintain, and update the platform. They ensure the platform's functionality, security, and user experience.
- **Legal/Compliance Team:** Ensure the platform complies with laws, regulations, and data privacy standards, maintaining its trustworthiness and legality.
- **Financial Team:** Manage payment systems, resolve payment-related issues, and ensure smooth financial transactions, fostering trust in the platform's payment processes.

2. General Description

FineAnnotator will be a web-based platform accessible through a web browser. It will offer a user-friendly interface for both researchers and students to:

2.1 Software Features

FineAnnotator provides a lot of features regarding its field. Both researchers and annotators can do a handful amount of things that make this platform a great use. The main users of FineAnnotator are students/annotators and researchers/companies. Below are the features for both types of users that are going to use FineAnnotator.

Researchers/Companies:

- Post data annotation jobs with details such as category, requirements, payment, and deadline.
- Choose the number of annotators they need.
- View profiles and select suitable annotators for their jobs.
- Manage payments to annotators.
- Give annotators a rating based on their performance.

Students/Annotators:

- Search for jobs based on criteria such as payment, work hours, deadline, and category.
- Apply for available jobs.
- Create and manage their profiles with CVs and other details.
- Receive ratings and recommendations from researchers and companies.
- Ask questions and clarify job details with researchers and companies.
- Receive payments through the platform after completing jobs.

2.2 Technologies

Technologies that are going to be used for FineAnnotator are mentioned below.

1. **Frontend Development:**
   HTML5/CSS3: Fundamental languages for creating the structure and visual design of web pages. Used to define content layout, styling, and overall appearance of the platform's user interface.
   JavaScript: Essential for client-side scripting, enabling interactive features, dynamic content, and handling user interactions within the web application.
   Frontend Frameworks/Libraries:

- **React.js**: These frameworks can be utilized to build reusable UI components, manage state efficiently, and create a responsive, interactive user interface for Fineannotate.
- **TailwindCSS and MaterialUI:** Frontend frameworks providing pre-designed UI components, grids, and styling elements for consistent and responsive design across the platform.

2. Backend Development:

Server-Side Language and Framework:

- **Node.js with Express.js**: Used to create the backend server, handle HTTP requests, and build RESTful APIs for managing tasks, user profiles, job postings, and interactions between Researchers/Companies and Annotators.

Database Management:

- **NoSQL (e.g., MongoDB):** Depending on the data structure and scalability needs, the chosen database system will store user data, job details, annotations, and other related information.

API Development:

- **RESTful API:** Designed to allow frontend components to communicate with the backend, facilitating actions such as job postings, applications, profile updates, and payment processing.

3. Authentication and Security:

Authentication Libraries:

- **JSON Web Tokens (JWT):** Implemented to manage user authentication and authorization, ensuring secure access to platform features and user data.

Security Measures:

- **HTTPS, SSL/TLS, CSRF Protection:** Ensuring secure communication and implementing measures to prevent common web vulnerabilities like Cross-Site Request Forgery (CSRF).

4. Other Technologies:

Version Control Systems:

- **Git:** Utilized for managing codebase versions, enabling collaboration among developers, and maintaining a history of changes to the application code.

These technologies collectively form the backbone of Fineannotate, enabling the development of its frontend and backend components, ensuring secure data handling, and user authentication, and providing tools for efficient collaboration among developers. The selection of specific technologies would depend on the project's requirements, scalability needs, and the development team's expertise.

2.3 System Models

Below is the Use-Case diagram for our software, FineAnnotator. Our website is about three different types of users, which are Researchers, Annotators, and Admin. Here are three different use-case diagrams for these three types of users.

1. **Annotators/Students**

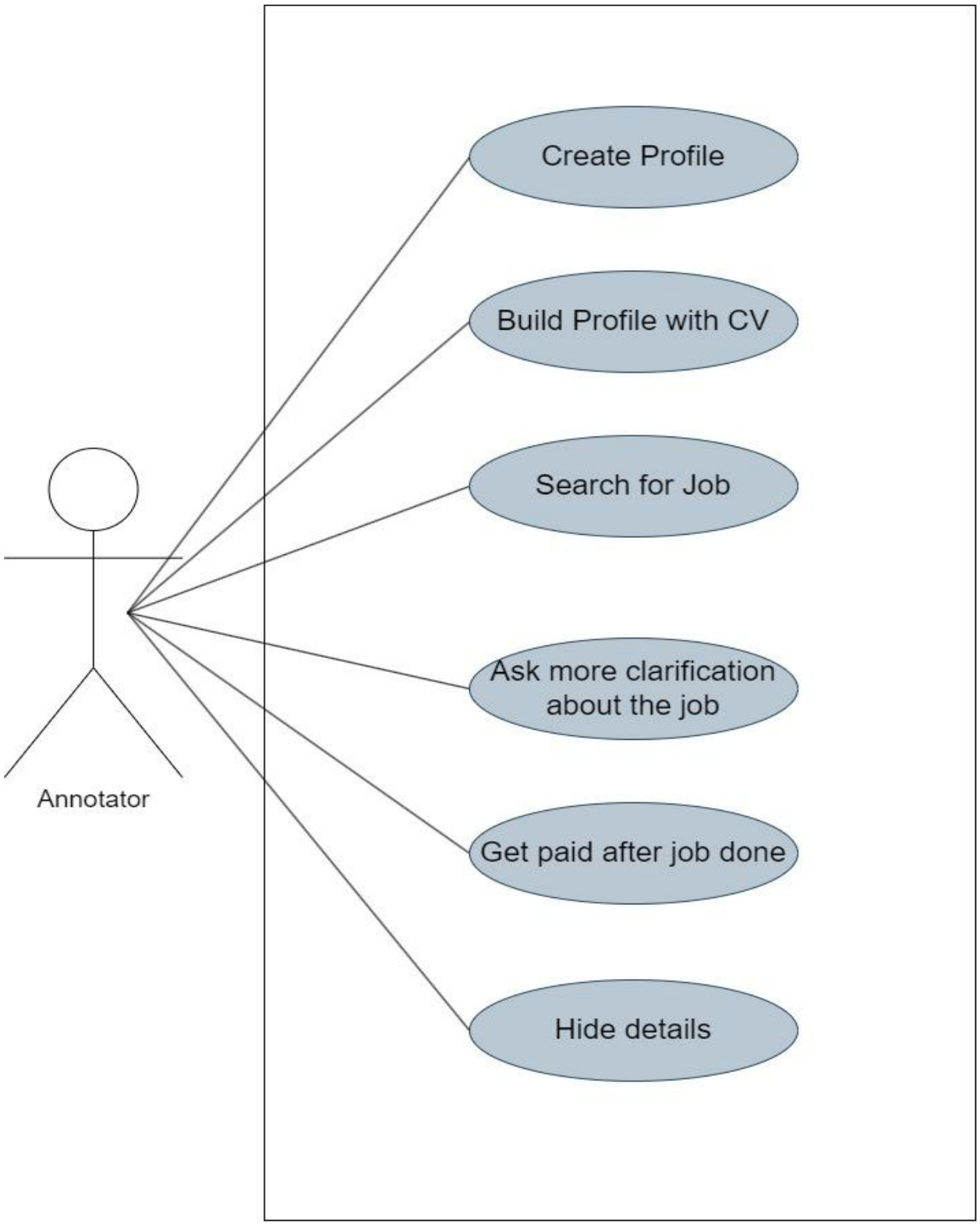

Figure: Use-case diagram for Annotators

2. Researchers/Companies

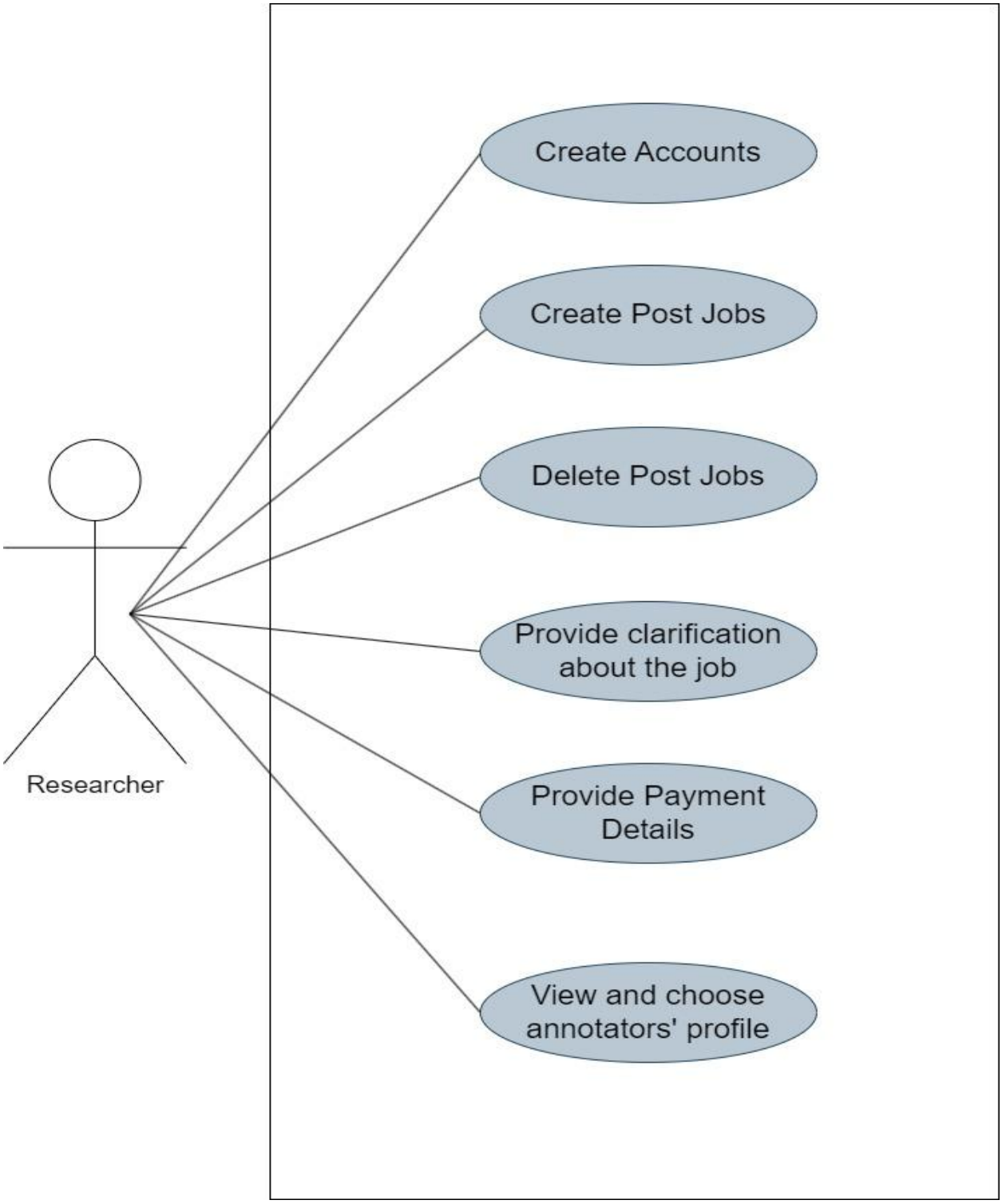

Figure: Use-case diagram for the Researchers

3. Admin

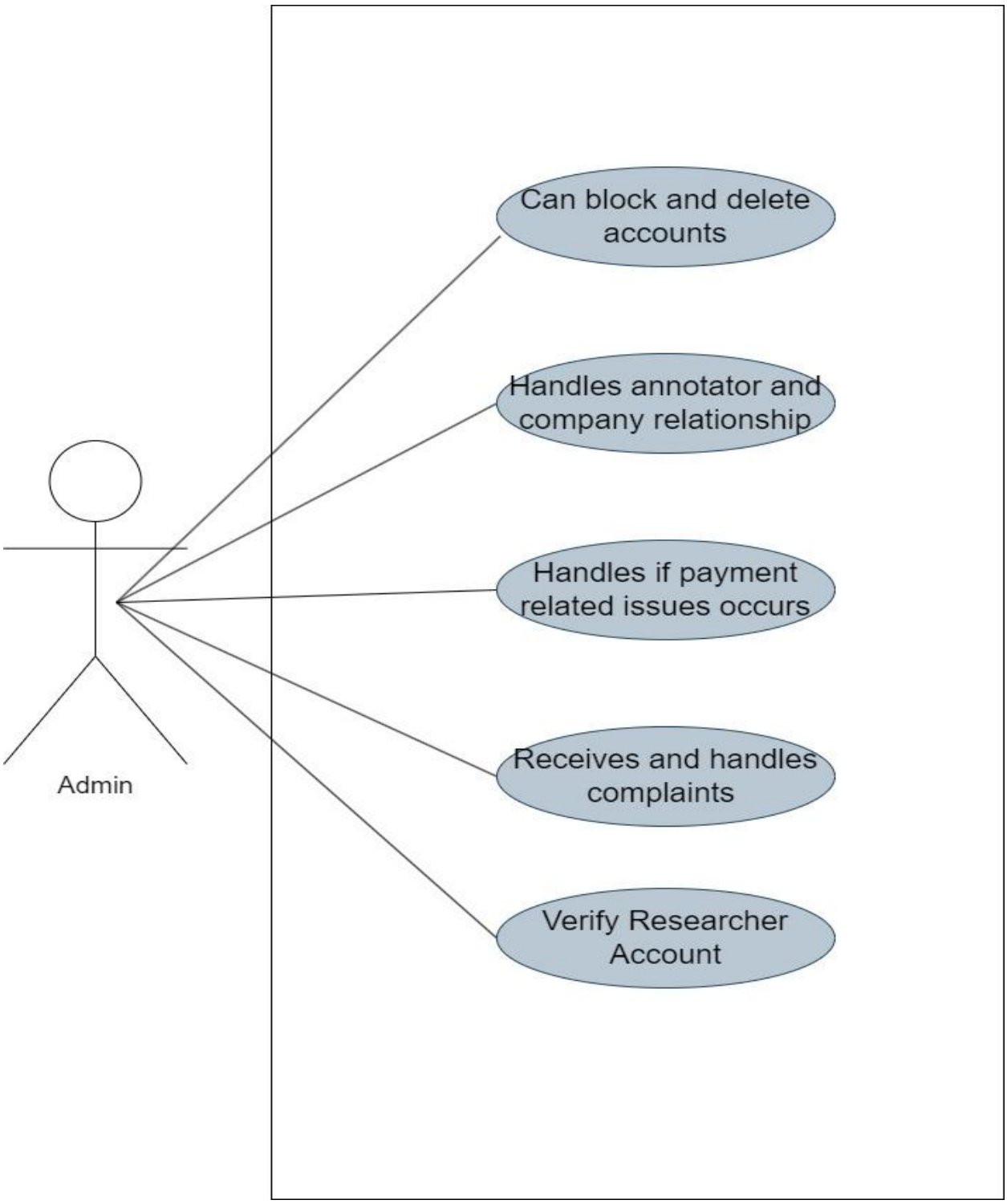

Figure: Use-case diagram for Admin

3. Functional Requirements

3.1 User Management

- Users can register and create profiles.
- User profiles should include information such as name, email, phone number, and CV (optional).
- Users can manage their account settings and contact information.
- Admins can block and delete user accounts.

3.2 Job Management

- Researchers and companies can post data annotation jobs.
- Job postings should include category, requirements, payment, work hours, deadline, and instructions.
- Researchers and companies can choose the number of annotators they need.
- Annotators can search for jobs based on various criteria.
- Annotators can apply for available jobs.
- Researchers and companies can view applications and select suitable annotators.
- Annotators can communicate with researchers and companies through a messaging system.
- Researchers and companies can rate and provide feedback on annotators' work.

3.3 Payment Management

- The platform will facilitate secure payments between researchers and annotators.
- Researchers and companies can deposit funds into their accounts.
- Payments to annotators will be processed through the platform after job completion and researcher approval.
- The platform will have a clear fee structure for both researchers and annotators.

3.4 Data Security

- The platform will ensure the security of user data and financial information.
- All data should be stored securely and encrypted.
- The platform will comply with relevant data privacy regulations.

4. Non-Functional Requirements

4.1 Performance

- The platform should be responsive and handle user requests efficiently.
- The job search and application process should be fast and easy.
- The platform should be able to scale to accommodate a large number of users and jobs.

4.2 Usability

- The platform should have a user-friendly interface that is easy to navigate and use.
- The platform should be accessible to users with disabilities.
- User feedback should be used to improve the platform's usability continuously.

4.3 Reliability

- The platform should be reliable and available 24/7.
- The platform should have a high uptime and minimal downtime.
- The platform should have a backup and disaster recovery plan in place.

4.4 Security

- The platform should be secure and protect user data from unauthorized access.
- The platform should have strong authentication and authorization mechanisms.
- The platform should comply with relevant security standards and regulations.

5. System Architecture

Fineannote follows the widely known MVC (Model-View-Controller) architectural pattern as the system architecture. Below is the general diagram for the MVC architectural pattern.

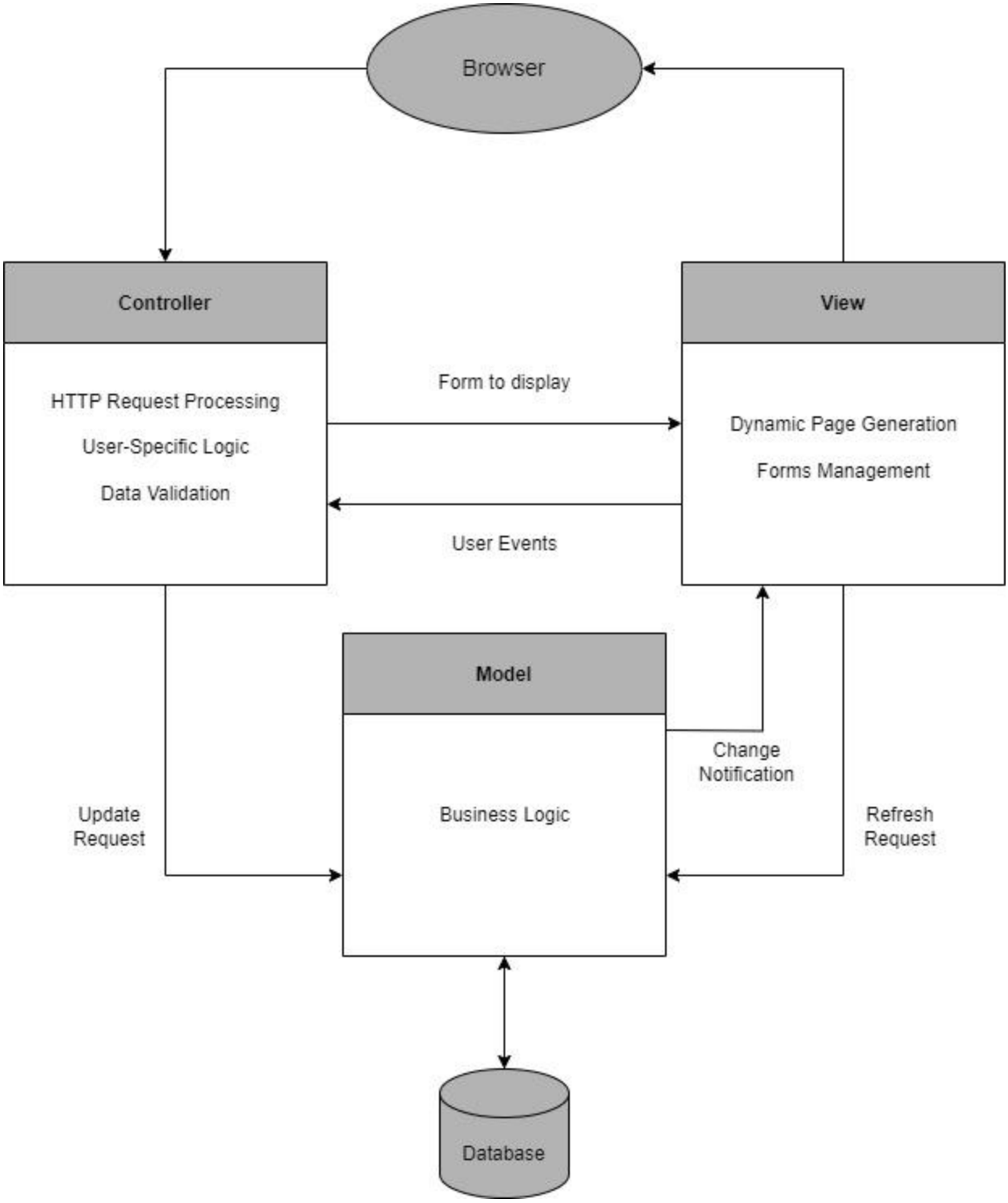

Figure: FineAnnotator System Architecture (MVC)

1. Model:
The Model in FineAnnotator would include classes or structures defining entities like User, Task, Payment, and functions/methods to interact with these entities, perform validations, and manage the data flow.

2. View:
User interface components such as web pages, forms, and elements displaying task details, profile information, application forms, and messaging systems would be part of the View in FineAnnotator.

3. Controller:
In FineAnnotator, controllers would include backend logic or server-side scripts that handle HTTP requests, validate input, trigger operations on the Model (e.g., posting a task, processing a job application), and render the appropriate View responses.

6. Additional Considerations

FineAnnotator may need to interact with other systems, such as:

- Payment gateways for processing payments.
- Cloud storage services for storing data.
- Data annotation tools for specific tasks.

Other than interacting with other systems, there are some other things to keep in consideration, such as:

- The platform should be designed to be flexible and adaptable to future growth and changes.
- The platform should be maintained and updated regularly.
- The platform should have a support system to help users with any questions or problems.

8. Conclusion

This SRS provides a detailed overview of the FineAnnotator platform and its functionalities. It outlines the key requirements for the system and ensures that it meets the needs of both researchers and students. The SRS will guide the development process and ensure that the final platform is a success.