

# VirtuWork: A Global Hub for Task-Based Services

## 1. Modules

### a. User Management Module

- **Features:**
  - User registration and login (clients and freelancers).
  - Profile management (skills, preferences, feedback).
  - View earned rewards and task history.

### b. Task Management Module

- **Features:**
  - Clients post tasks with specific requirements.
  - Freelancers browse and bid for tasks.
  - Machine learning algorithms recommend tasks to users.

### c. Admin Management Module

- **Features:**
  - Admin facilitates communication between clients and freelancers.
  - Admin moderates disputes and ensures task progress tracking.
  - Admin receives a 10% commission from payments.

### d. Reward System Module

- **Features:**
  - Rewards are calculated based on client ratings and feedback.
  - Rewards can be redeemed during payment transactions.

### e. Payment and Transaction Module

- **Features:**
  - Secure payment integration using gateways like Stripe or PayPal.
  - Option to apply rewards for task payments.
  - Automatic transfer of admin fees.

### f. Task Progress Module

- **Features:**
  - Dashboards to track task milestones, deadlines, and deliverables.

- Notifications for updates and task completion.

#### **g. Machine Learning Module**

- **Features:**
  - Task recommendation based on user profiles, past activities, and preferences.
  - Task pricing optimization based on historical data and complexity.

## **2. Datasets / Tables in the Database**

### **Database Tables:**

- 1. Users Table:**
  - userId, name, email, password, role (client/freelancer), skills, preferences, rewards, rating, taskHistory.
- 2. Tasks Table:**
  - taskId, clientId, description, category, skillsRequired, deadline, price, status (open/in-progress/completed).
- 3. Bids Table:**
  - bidId, taskId, freelancerId, bidAmount, status (pending/accepted/rejected).
- 4. Payments Table:**
  - paymentId, taskId, amount, adminFee, rewardUsed, date.
- 5. Admin Table:**
  - adminId, name, contact, email.
- 6. Ratings and Feedback Table:**
  - ratingId, taskId, freelancerId, clientId, rating, feedback, date.

## **3. Reference List of Papers, Journals, and Blogs**

### **Papers and Journals**

1. "A Survey of Machine Learning Applications in Freelancer Platforms"
  - *DOI:* 10.1109/xyz2021.12345
2. "Recommendation Systems: User-Centered Algorithms and Evaluations"
  - *Author:* F. Ricci et al., 2015.
3. "Secure Payment Gateways in E-Commerce Applications"
  - *Journal:* International Journal of Computer Applications, 2022.

### **Blogs and Articles**

1. Freelance Task Platforms: Trends and Challenges
2. How to Build a Reward System in Web Applications
3. Machine Learning in Recommendation Systems.

#### **4. Objectives**

1. Build a platform for seamless interaction between clients and freelancers for remote tasks.
2. Implement machine learning for task recommendations .
3. Provide transparency and accountability through progress tracking dashboards.
4. Introduce a reward system to incentivize quality task completions.
5. Ensure platform sustainability by integrating an admin fee model.

#### **5. Problem Statement**

- **Existing Issues:**
  1. Lack of reliable platforms for connecting clients and freelancers for remote tasks.
  2. Difficulty in finding suitable tasks or freelancers due to inefficient matching systems.
  3. Lack of incentive systems for quality task delivery.
  4. Issues in payment transparency and disputes between clients and freelancers.

#### **6. Innovative Ideas in the Project**

1. **Reward System:**
  - Incentivizing freelancers through redeemable rewards based on ratings and feedback.
2. **Admin-Mediated Communication:**
  - Restrict direct communication to maintain professionalism and security.
3. **Progress Tracking:**
  - Real-time dashboards for clients and freelancers to monitor task progress effectively.
4. **Task Recommendation Engine:**
  - Personalized recommendations based on user profiles and historical activities.