### **CHAPTER 1**

### **COMPANY PROFILE**



**1.1 History of the Organization**

This chapter gives brief introduction about company its vision, mission values and inspirer of the company. This also highlights the services offered, products developed by the company. Overview of the Company ParvaM was founded by a team of passionate folks from diverse platforms with the intent of delivering the valued services to keep the future ready with various software solutions with cutting edge technologies in the market. We are here to deliver Quality Technical Products, Development & Services for all your Current and Future software requirements. We are also committed to providing Quality Staffing and Consulting services.

ParvaM has served the customers with a delighting experience with our expert services on a competitive edge in delivering the unique experiences on technology, consulting and software solutions in the areas of implementing enterprise applications, professional web sites, customized applications and technology workshops on a different trending technology .ParvaM Consul-tech Pvt. Ltd. is a software development and IT consulting company based in Bengaluru, Karnataka, India. Established in 2017, the company specializes in providing enterprise application implementations, professional website development, customized software solutions, and technology workshops on emerging technologies. With a team size ranging from 11 to 50 employees, ParvaM aims to deliver unique and competitive experiences to its clients. Their services cater to various industries, focusing on enhancing technological capabilities and business processes.

**CHAPTER 2**

### **ABOUT THE DEPARTMENT**

* + 1. **Objectives**

**1. Provide Innovative Technology Consulting**

To deliver advanced IT and business consulting solutions that address complex challenges for clients across various industries.

**2. Support Local and Global Businesses**

To empower startups, SMEs, and enterprises in Bangalore and beyond with scalable, customized technology services.

**3. Promote Digital Transformation**

To assist organizations in adopting digital technologies like cloud computing, AI, IoT, and automation for smarter operations.

**4. Foster a Culture of Excellence**

To uphold high standards of professionalism, quality, and ethical conduct in every engagement.

**5. Enable Business Growth Through IT Solutions**

To offer cost-effective and performance-driven software, data analytics, and digital strategy services that drive business value.

* + 1. **Operations of the Organization**

**1. Web Design & Development**

Creating responsive, mobile-friendly websites with features like SSL integration, SEO optimization, and custom designs to enhance online presence.

**2. Mobile App Development**

Developing custom mobile applications for iOS, Android, and hybrid platforms using an agile approach to ensure user-friendly and secure apps

**3. Digital Marketing & Branding**

Offering services such as SEO, social media marketing, and content creation to boost brand visibility and engagement.

**4. Custom Software Solutions**

Providing tailored software solutions, including ERP and CRM systems, to streamline business processes.

**5. IT Staffing & Training**

Assisting companies in recruiting suitable candidates and offering training and internship programs to develop skilled professionals.

**6. Research & Development**

Investing in R&D to stay ahead with innovative solutions and emerging technologies.

* + 1. **Major Milestones**

ParvaM Consul-Tech Pvt. Ltd., established on October 13, 2017, in Bengaluru, Karnataka, has achieved several significant milestones that highlight its growth and impact in the IT sector.

* **Financial Growth**
* **Revenue Increase**: In the fiscal year 2022, the company reported a remarkable **184.58% increase in revenue**.
* **Profit Surge**: Profit saw a substantial rise of **237.81%**, indicating improved operational efficiency.
* **Net Worth Growth**: The company's net worth grew by **32.90%**, reflecting a strong financial position.

**Service Expansion**

* ParvaM has diversified its offerings to include:
* **Web Design & Development**: Over 100 satisfied clients across Karnataka and globally.
* **Mobile App Development**: Expertise in iOS, Android, and Hybrid apps.
* **Digital Marketing**: Strategies to enhance online presence and engagement.
* **Custom Software Solutions**: Tailored software to meet unique business needs.
* **Graphic Designing**: Creative designs to strengthen brand identity.
* **Content Writing**: SEO-friendly content to boost organic traffic.
* **IT Staffing**: Assistance in identifying suitable candidates for recruitment.
* **Training & Internships**: Programs to nurture future professionals.

**Team Development**

* **Employee Satisfaction**: ParvaM is rated 4.7 out of 5 on AmbitionBox, based on employee reviews.
* **Skill Development**: Employees commend the company for fostering skill development and providing career growth opportunities.
  + 1. **Structure of the Organization**

**Leadership Team**

* Naveen Mariyappa – Co-founder and Director since October 13, 2017.
* Muniyappa Muniramegowda – Co-founder and Director since September 30, 2018.
* Ram Nisarg Manohar – Co-founder and Director.

**Core Service Teams**

* + Web Design & Development – Focused on creating user-friendly websites.
  + Mobile App Development – Expertise in iOS, Android, and Hybrid applications.
  + Digital Marketing – Strategies to enhance online presence.
  + Custom Software Solutions – Tailored software to meet business needs.
  + Graphic Designing – Creative designs to strengthen brand identity.
  + Content Writing – SEO-friendly content to boost organic traffic.
  + IT Staffing – Assistance in identifying suitable candidates for recruitment.
  + Training & Internships – Programs to nurture future professionals.

**Organizational Culture**

ParvaM emphasizes a collaborative and innovative work environment. The company is committed to delivering customized solutions with a future-focused approach, implementing cutting-edge technologies to meet client needs.

* + 1. **Services Offered**

**1. Web Design & Development**

* Custom websites using modern UI/UX principles
* E-commerce platforms
* CMS solutions (e.g., WordPress, Joomla)

**2. Mobile App Development**

* Android and iOS app development
* Hybrid app solutions (e.g., React Native, Flutter)
* App testing and deployment support

**3. Digital Marketing**

* SEO (Search Engine Optimization)
* Social Media Marketing (SMM)
* Google Ads and PPC Campaigns
* Email marketing
* Online reputation management

**4. Custom Software Solutions**

* Enterprise software applications
* CRM and ERP systems
* Cloud-based applications

**5. Graphic Designing**

* Logo and brand identity design
* UI/UX design for apps and websites
* Marketing and promotional materials

**6. Content Writing**

* SEO-optimized website content
* Blog writing and management
* Technical and marketing copywriting

**7. IT Staffing**

* Recruitment solutions for tech roles
* On-demand developer outsourcing
* HR and payroll support for IT talent

**8. Training & Internships**

* Web and app development training
* Digital marketing internships
* Certification programs for fresh graduates

### **CHAPTER 3**

### **TASKS PERFORMED**

**3.1 FRONTEND DEVELOPMENT**

* **HTML:** Hyper Text Markup Language forms the backbone of any web application, including those built with Spring Boot. It serves as the structural framework, defining the content and layout of a webpage. HTML consists of various elements such as headings, paragraphs, lists, images, tables, and forms, which help organize information in a meaningful way. It uses tags to mark different components of a webpage, ensuring they are properly displayed in browsers. In a Spring Boot application, HTML can be used with templating engines like Thymeleaf, which allows dynamic content rendering by embedding Java logic within the markup. This enables developers to create reusable and interactive web pages while integrating data from backend services efficiently.
* **CSS:** Cascading Style Sheets is responsible for styling and enhancing the visual appeal of a web application. It controls aspects such as colors, fonts, spacing, positioning, and responsiveness, ensuring an intuitive user experience across different devices. CSS can be written inline, within a tag in an HTML document, or externally in a separate stylesheet. Spring Boot applications often use CSS to style static HTML files or dynamically rendered pages. Additionally, modern CSS frameworks like Bootstrap or Tailwind CSS can be integrated to provide pre-designed components and utility classes, reducing development time while maintaining a professional and consistent design.
* **JavaScript:** Is a powerful scripting language that adds interactivity and dynamic behaviour to web applications. It enables functionalities such as user input validation, animations, event handling, and asynchronous communication with backend APIs. In a Spring Boot application, JavaScript can be used to interact with RESTful services, fetching data from the server and updating the webpage without requiring a full reload. This is commonly achieved using AJAX or modern frontend frameworks like React, Angular, or Vue.js, which enhance the application’s interactivity. JavaScript’s ability to manipulate the Document Object Model (DOM) allows developers to create smooth user experiences, making it an indispensable tool in modern web development.

**3.2 BACKEND DEVELOPMENT**

A backend Spring Boot application using Maven for web development provides a robust and scalable framework for building enterprise-grade applications. Spring Boot simplifies the development process by offering built-in configurations, auto-configuration, and dependency management, making it easier to create production-ready applications with minimal setup. Maven acts as the project management tool, handling dependencies, build lifecycle, and packaging. In a typical Spring Boot web application, Maven manages libraries such as Spring Web for handling HTTP requests, Spring Data for database interactions, and Thymeleaf or RESTful services for communication between frontend and backend. The Spring Boot framework includes an embedded Tomcat server, allowing developers to run the application without requiring additional deployment configurations. The application follows a layered architecture, where controllers process requests, services contain business logic, and repositories manage data persistence with JPA or Hibernate. With Maven’s dependency management, developers can easily integrate authentication, logging, and third-party APIs, streamlining the development process. Additionally, Spring Boot provides extensive support for RESTful APIs, enabling seamless interaction between the frontend and backend through JSON-based communication. By combining Spring Boot with Maven, developers can build secure, scalable, and efficient web applications that are well-suited for modern software development.

**3.3 DATABASE MANAGEMENT**

In a Spring Boot application, MySQL database management using XAMPP provides a flexible and efficient approach for handling data. XAMPP is a local server environment that includes MySQL, Apache, and PHP My Admin, making database administration and development more convenient. MySQL serves as the database system, storing structured data and enabling efficient retrieval and manipulation using SQL queries. Spring Boot integrates with MySQL seamlessly through JPA (Java Persistence API) and Hibernate, allowing developers to interact with the database using object-relational mapping (ORM). By configuring the application. properties or application. yml file, developers specify the database connection details, including the MySQL driver, username, password, and schema. The application can then define entity classes representing database tables and use Spring Data JPA repositories to perform CRUD (Create, Read, Update, Delete) operations effortlessly. With XAMPP, MySQL can be managed using PHP My Admin, providing a user-friendly graphical interface for handling tables, queries, and database users. Additionally, Spring Boot applications often use RESTful APIs to retrieve and store data dynamically, ensuring seamless interaction between the backend and frontend. By leveraging MySQL with XAMPP in Spring Boot, developers gain a robust and efficient way to manage data for web applications, enhancing performance, scalability, and usability.

### **CHAPTER 4**

### **REFLECTION NOTES**

**4.1 EXPERIENCE**

Working on the IPL Auction Bidding App using Java Spring Boot for the backend and HTML, CSS, and JavaScript for the frontend was a great learning experience for me. I learned how to build a complete web application by doing both backend and frontend work. On the backend, I used Java Spring Boot to create APIs that handled important features like login, player management, bidding, and team information. I also used Spring Security to keep the system safe, so only admins could control the auction and team owners could place bids. It helped me understand how to manage different users and give them access based on their roles. I used a MySQL database to store all the data, such as player details, bids, teams, and users. Spring Data JPA made it easy to connect the backend with the database and manage all the records. I also followed the MVC (Model-View-Controller) pattern, which helped me keep the code clean and well organized. On the frontend, I used HTML to make the layout of the web pages, CSS to design the look of the pages, and JavaScript to make the pages interactive. I created simple and clean forms for login, player entry, and bidding. JavaScript helped me to update the data on the screen without refreshing the whole page. I also added features to show the current highest bid, remaining budget, and bidding status, so users could follow the auction easily. This project helped me understand how the frontend and backend work together and how to build real-time web applications. I improved my skills in Java, web design, and database handling. Overall, this was a fun and useful project that gave me the confidence to build more advanced applications in the future.

**4.2 TECHNICAL OUTCOMES**

**4.2.1 SYSTEM REQUIREMENT SPECIFICATION**

**4.2.2 HARDWARE REQUIREMENTS**

* Processor: Intel i5 or above (Quad-core recommended)
* Hard Disk: 100 GB (minimum)
* Memory (RAM): 8 GB (minimum), 16 GB recommended

**4.2.3 SOFTWARE REQUIREMENTS**

* Web Server: Embedded Apache Tomcat (included with Spring Boot)
* Operating System: Windows
* Platform: Java 17+, Spring Boot Framework
* Build Tool: Maven
* Frontend Technologies: HTML5, CSS3, JavaScript
* IDE (Editor): VS Code
* Database: MySQL / PostgreSQL / H2 (for development)
* JDK: Java Development Kit 17 or later

**4.3 SYSTEM ANALYSIS AND DESIGN**

**4.3.1 EXISTING SYSTEM**

In the development of a construction website using the Spring Boot framework, an existing system typically consists of a well-defined architecture that integrates frontend, backend, and database management efficiently. The backend, built using Spring Boot, provides RESTful APIs to manage various features such as project listings, service details, customer inquiries, and testimonials. It incorporates Spring Security for authentication and authorization, ensuring secure access to user data. The frontend is developed using HTML, CSS, and JavaScript, enhancing responsiveness and interactivity for users accessing construction project details. MySQL serves as the database management system, storing structured data related to ongoing and completed projects, service offerings, and client feedback. The system often includes an admin panel for construction firms to update project statuses, add new services, and manage customer interactions seamlessly. Spring Data JPA and Hibernate facilitate database interactions, allowing efficient CRUD operations without complex SQL queries. Additionally, Maven is used for dependency management, ensuring smooth integration of various Spring Boot modules and third-party libraries. The existing system may also leverage cloud deployment services to enhance scalability and reliability, ensuring seamless performance even with high user traffic. With these components working together, a construction website built on Spring Boot delivers a dynamic, secure, and efficient platform for managing construction projects while providing an intuitive user experience.

**4.3.2 DISADVANTAGES OF THE EXISTING SYSTEM**

1. **Manual Auction Control** – The admin must control all auction stages manually, which is time-consuming and prone to human error.
2. **Weak Real-Time Performance** – The system struggles to handle multiple bids simultaneously, especially under high load, due to limited WebSocket optimization.
3. **Basic User Interface** – The app often lacks a modern, responsive design, making it less engaging and harder to use on mobile devices.
4. **No Automated Bid Validation** – There's insufficient enforcement of team budgets or rules, allowing invalid or duplicate bids to slip through.
5. **Lack of Analytics** – Users and admins can't view reports or trends, making strategic decisions difficult.
6. **Poor Error Handling** – The system provides minimal feedback during errors or disconnects, which can disrupt bidding.
7. **Scalability Issues** – The architecture isn't optimized for scaling to handle many concurrent users or auctions.
8. **Security Gaps** – Without advanced security features like audit logging and rate limiting, the app may be vulnerable to misuse.
9. **Limited Extensibility** – Adding new features or integrating with mobile apps and third-party tools is difficult due to a tightly coupled backend.

**4.3.3 PROPOSED SYSTEM**

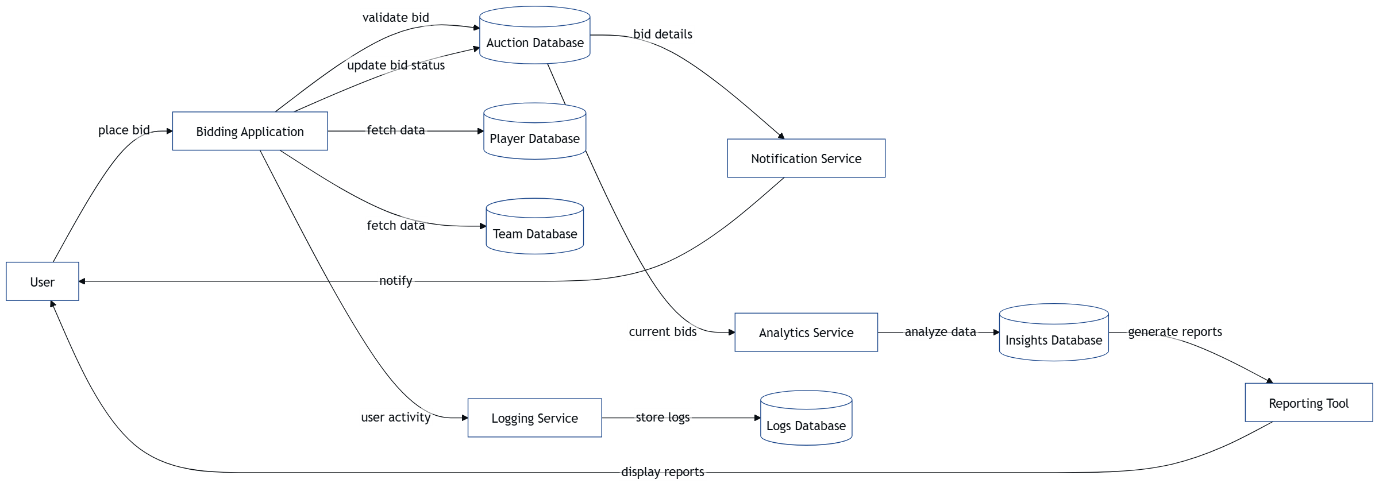
The proposed IPL Auction Bidding App using Java Spring Boot is designed to overcome the limitations of the existing system by providing a fully automated, real-time, and user-friendly solution. In this system, the auction process is automated—players are placed into auction rounds with countdown timers, and the system automatically assigns players to the highest bidder when time runs out. Real-time bidding is enabled using WebSocket technology, which allows all team owners to see live updates and bid without delays. The backend is built with Java Spring Boot and uses REST APIs for standard operations and WebSockets for live communication. Security is strengthened through JWT-based authentication and role-based access control, ensuring that only authorized users can access specific features. The user interface, built with React or Angular, is modern, responsive, and mobile-friendly, allowing team owners to participate in auctions from any device. The system also enforces team budget limits in real time to prevent invalid bids. Additionally, analytics features like dashboards, bid history, and team performance summaries provide useful insights for both admins and users. This proposed system is scalable, secure, and efficient, making it ideal for professional IPL-style auctions.

**4.3.4 ADVANTAGES OF THE PROPOSED SYSTEM**

1. **Auto Real-Time Bidding:** The system ensures immediate bid updates across all users using WebSocket or STOMP, creating a dynamic auction experience.
2. **Enhanced Security:** With Spring Security and JWT, the system guarantees secure authentication and role-based access, protecting sensitive data and operations.
3. **Improved User Interface:** Using modern frontend technologies, the system provides a responsive and user-friendly interface for a seamless auction experience.
4. **Scalability**: The system can scale efficiently to handle increased users and data by leveraging cloud platforms like AWS or Azure.
5. **Efficient Database Management:** Spring Data JPA and a robust database ensure fast, reliable access to player and auction data.
6. **Automated Budget Tracking:** The system automatically tracks each team’s spending during the auction, preventing budget overruns and ensuring fairness.
7. **Admin Control**: Admins gain greater control with a powerful dashboard to manage the auction process and view real-time metrics.
8. **Cross-Platform Accessibility:** The web-based platform allows users to participate from any device, improving accessibility and user convenience.
9. **Efficient Performance:** Spring Boot’s backend architecture ensures high performance, even during high-traffic periods, providing a smooth auction experience

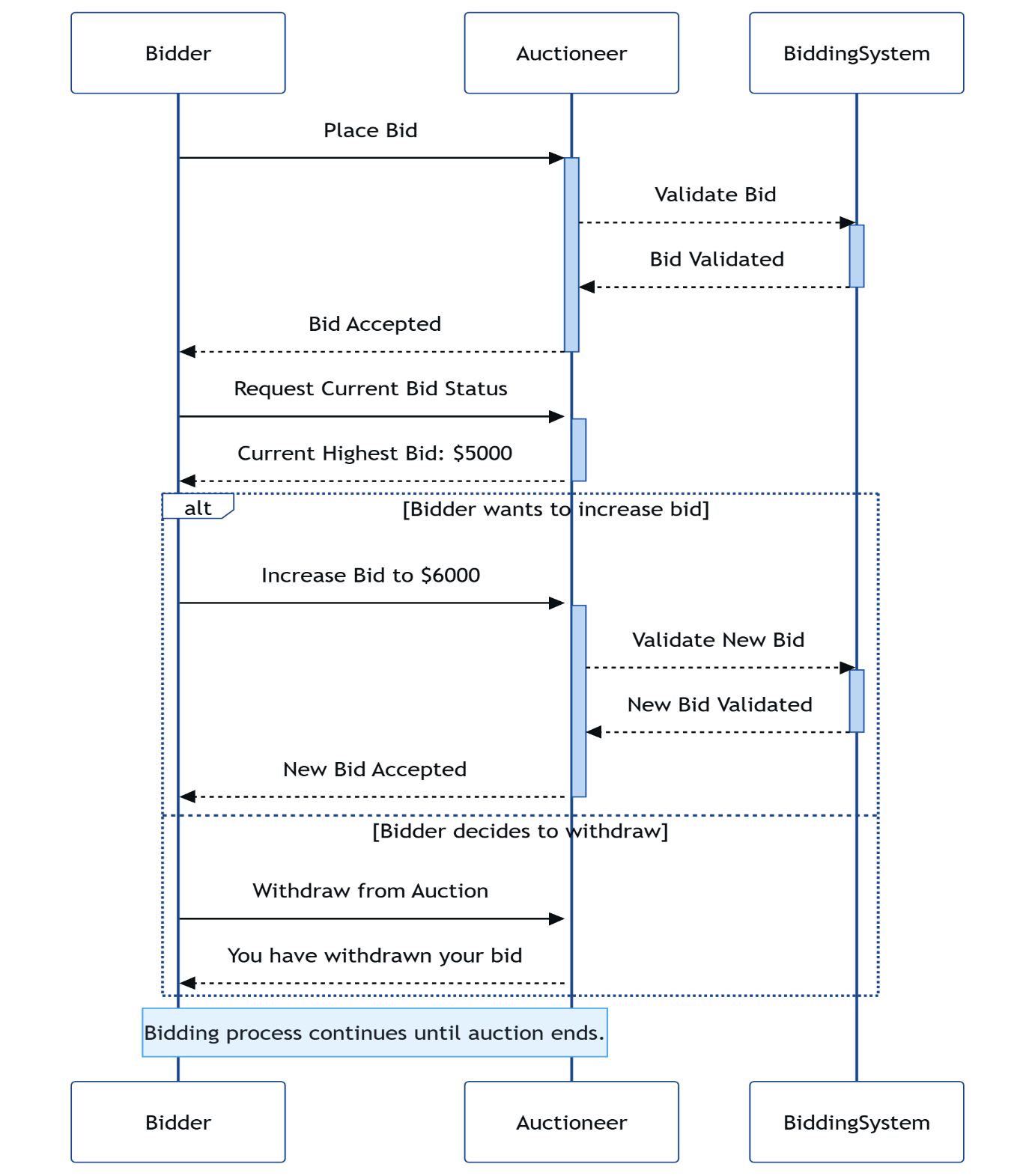
**4.4 SYSTEM ARCHITECTURE**

**4.4.1 DATA FLOW DIAGRAM**

****

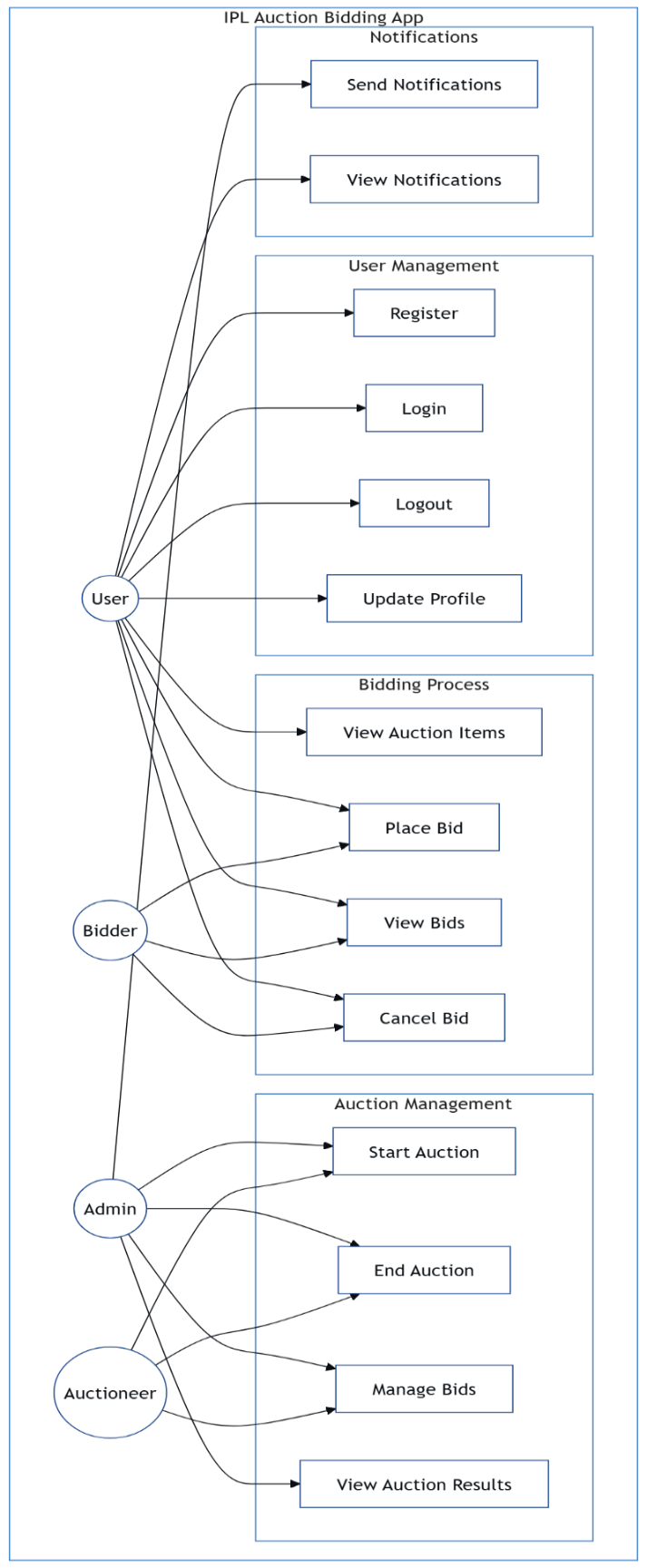
**Fig 4.4.1 Data Flow Diagram**

**4.4.2 UML DIAGRAM**

****

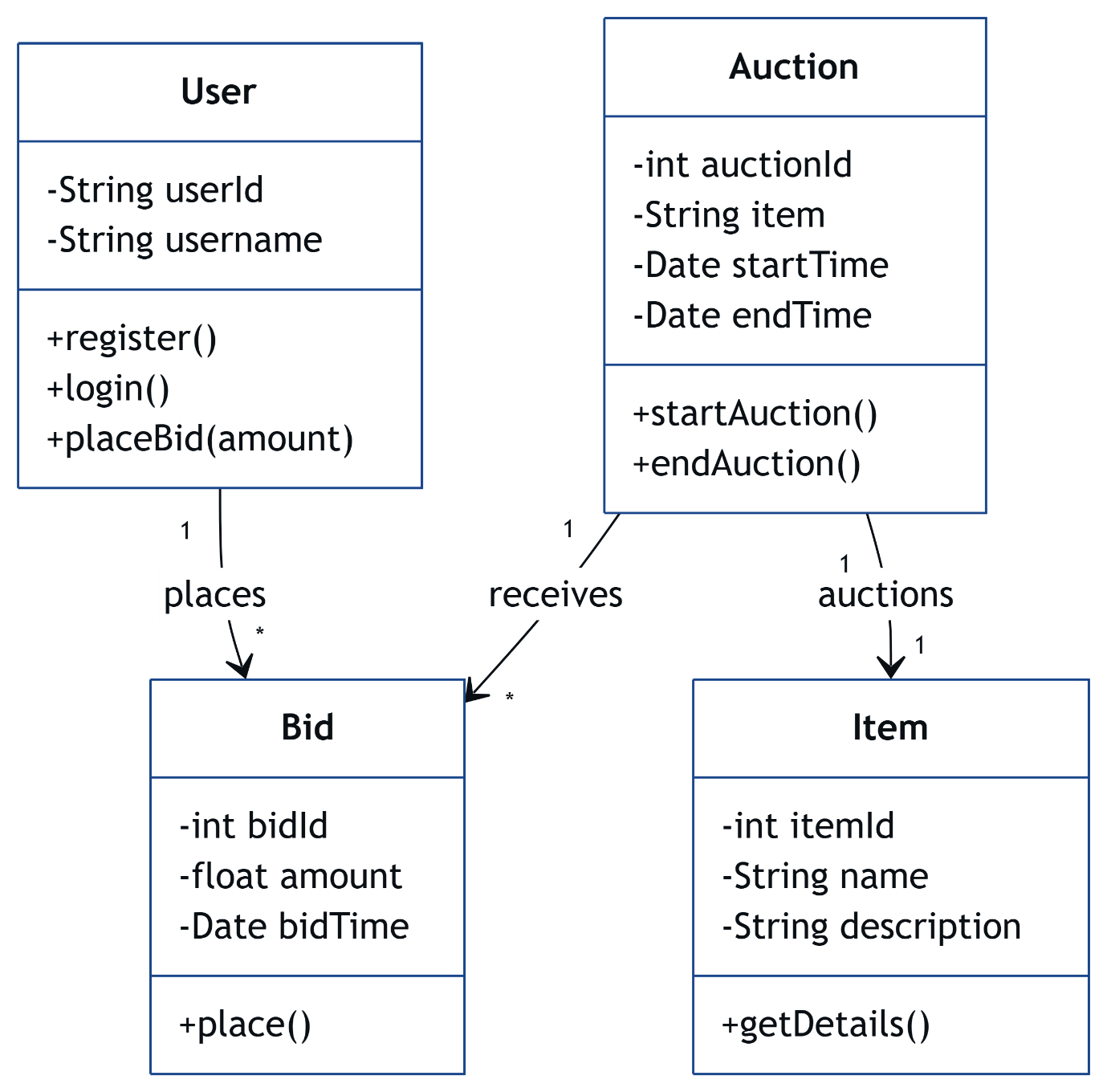
**Fig 4.4.2 UML Diagram**

**4.4.3 USE CASE DIAGRAM**

****

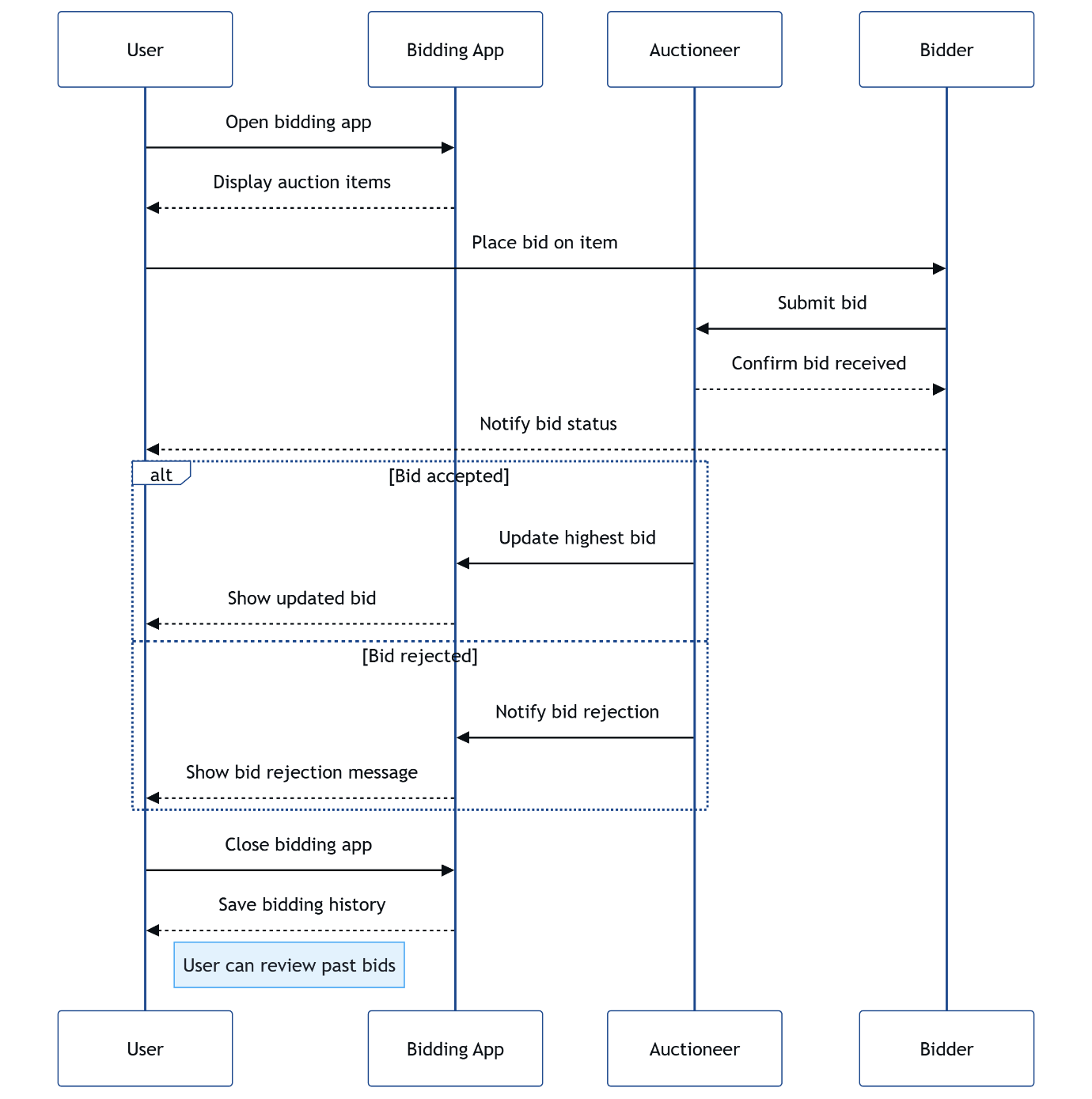
**Fig 4.4.3 Use Case Diagram**

**4.4.4 CLASS DIAGRAM**

****

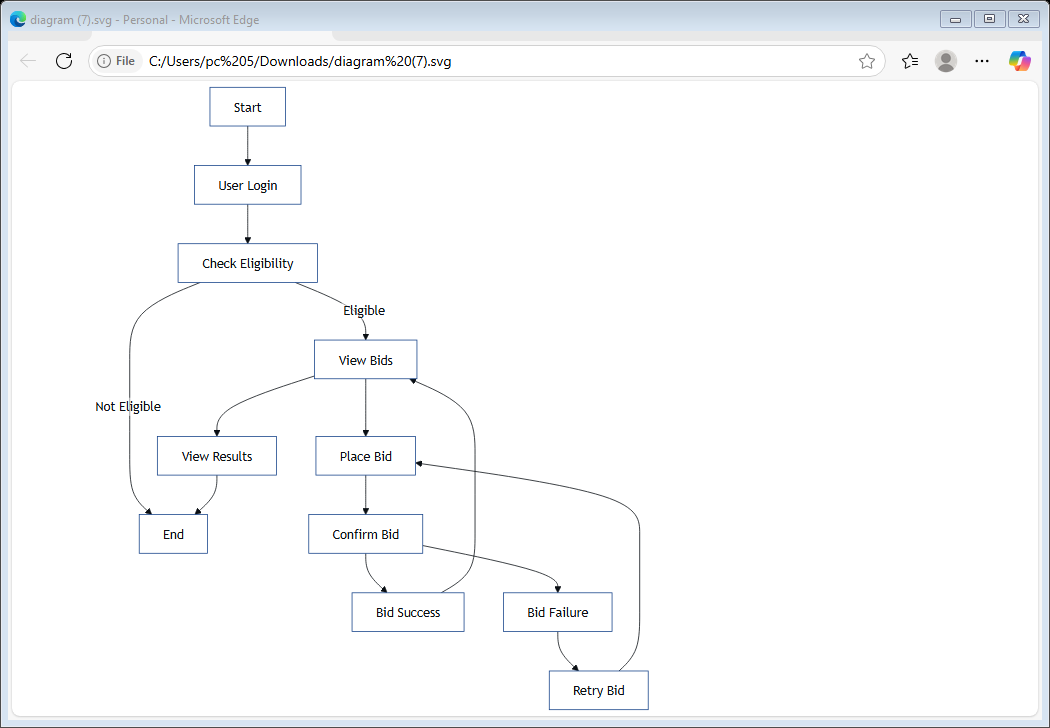
**Fig 4.4.4 Class Diagram**

**4.4.5 SEQUENCE DIAGRAM**

****

**Fig 4.4.5 Sequence Diagram**

**4.4.6 ACTIVITY DIAGR**

****

**Fig 4.4.6 Activity Diagram**

**4.5 IMPLEMENTATION**

**4.5.1 MODULES**

**1. Authentication Module**

* Admin login
* Player registration page

**2. Project Module**

* Team Management (add/update/delete team)
* Upload Team Logo
* Set Team Budget
* Player Management (add/update/delete team)
* Upload Player Image
* Set Base Price
* Set Role (Batsman/Bowler/All-Rounder/Wicket-Keeper)

**3. Services Module**

* Team Management(get team/update team/delete team)
* Player Management(get team/update team/delete team)

**4. Team Module**

* Name, position, social links, bio, photo
* Dynamic team section on “Our Team” page

**5. Testimonials Module**

* Name (or nickname)
* Email (optional, private)
* Optional Image Upload (profile or user screenshot)
* Date auto-generated
* Login required
* Real-time bidding
* Team building
* Carousel display on homepage

**6. Blog Module**

* Add/edit/delete blog posts
* Categories/tags
* Blog thumbnails, rich text content
* Comment system (optional with moderation)
* SEO meta title/description

**7. Contact Module**

* Contact form (name, email, contact number)
* Backend handling with DB storage
* Google Maps integration (optional)
* Admin view of messages

**8. Home Module**

* Dynamic content blocks for home
* Change homepage headlines, intro sections, banners
* Carousel/banner management

**9. Admin Panel (CMS)**

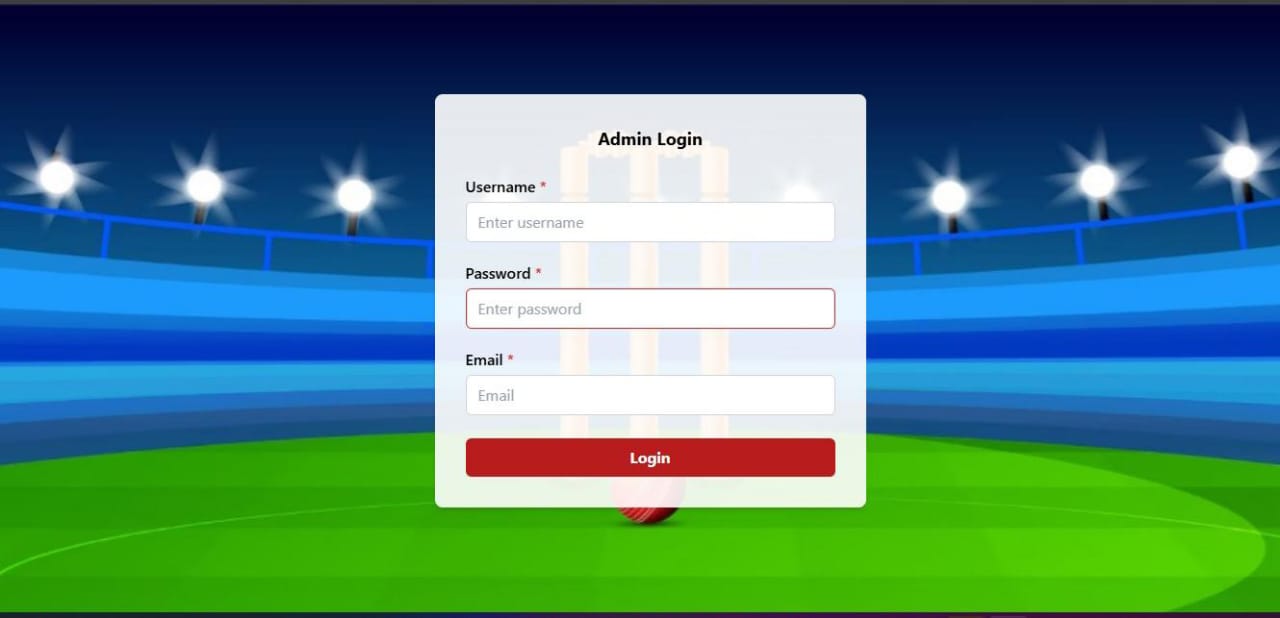
* Central place to manage all above modules
* Rich text editor for projects/services/testimonials
* Image upload with preview
* Role-based control (admin/editor)

**10. Frontend with Thymeleaf**

* Dynamic rendering of all content
* Tailwind CSS or Bootstrap for styling
* SEO-friendly routes and layout fragments (header, footer, etc.)

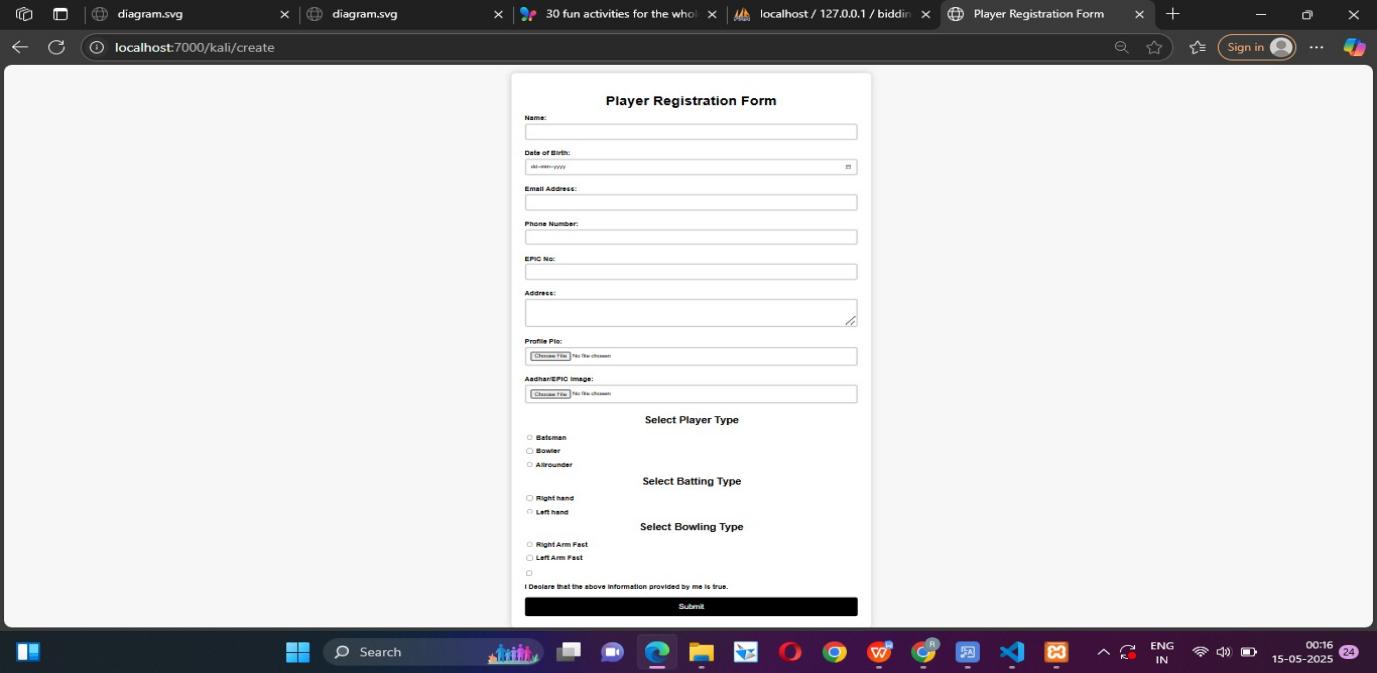
**4.6 SCREENSHOTS**

**4.6.1 Admin login**

****

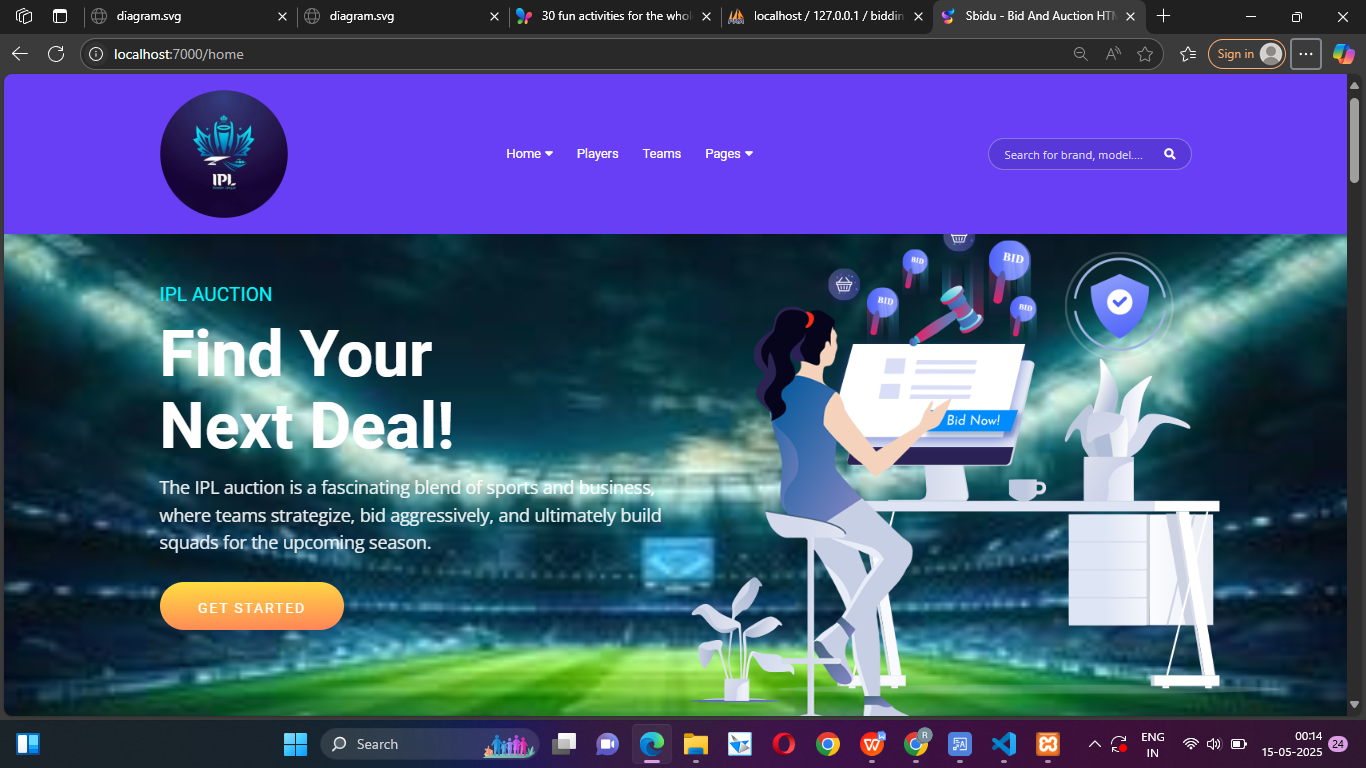
**Fig 4.6.1 Admin login**

**4.6.2 Player registration page**

****

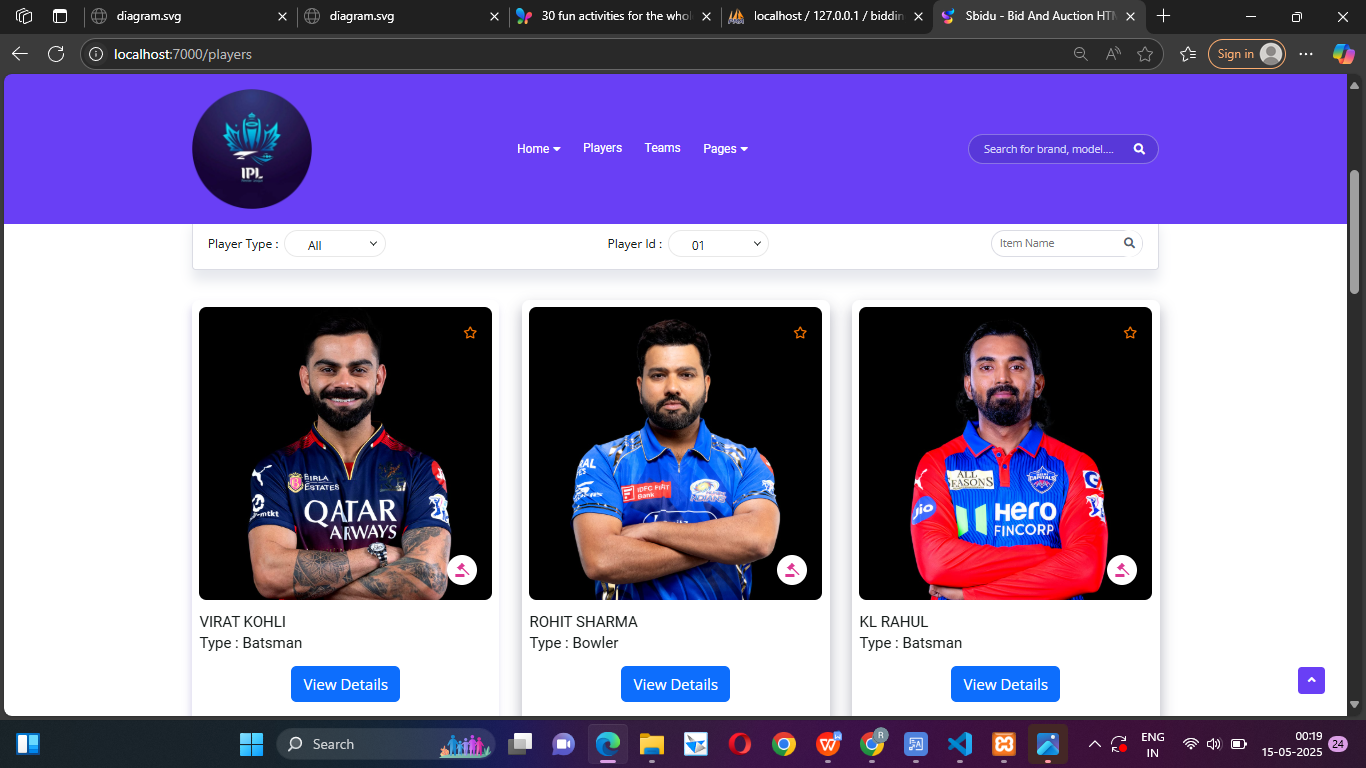
**Fig 4.6.2 Player registration page**

**4.6.3 Home page**

****

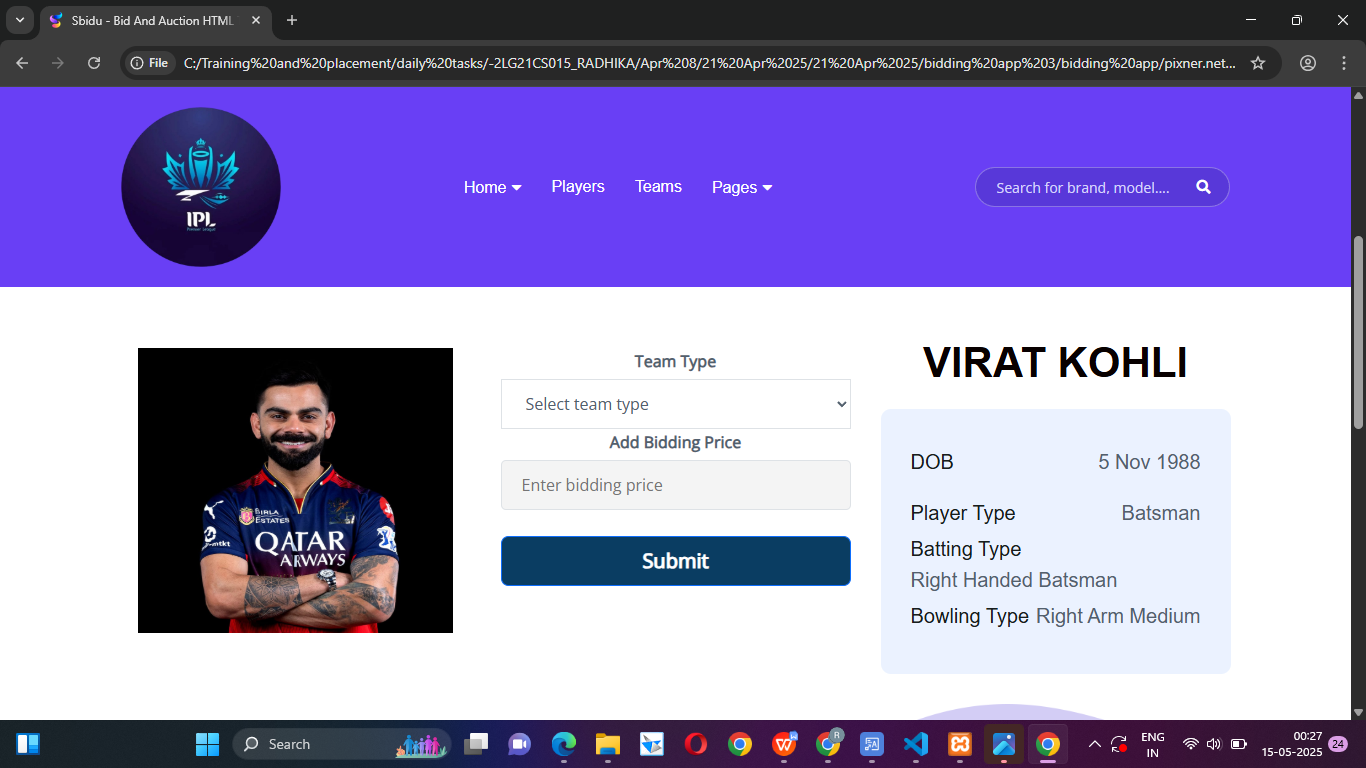
**Fig 4.6.3 Home page**

**4.6.4 Players list**

****

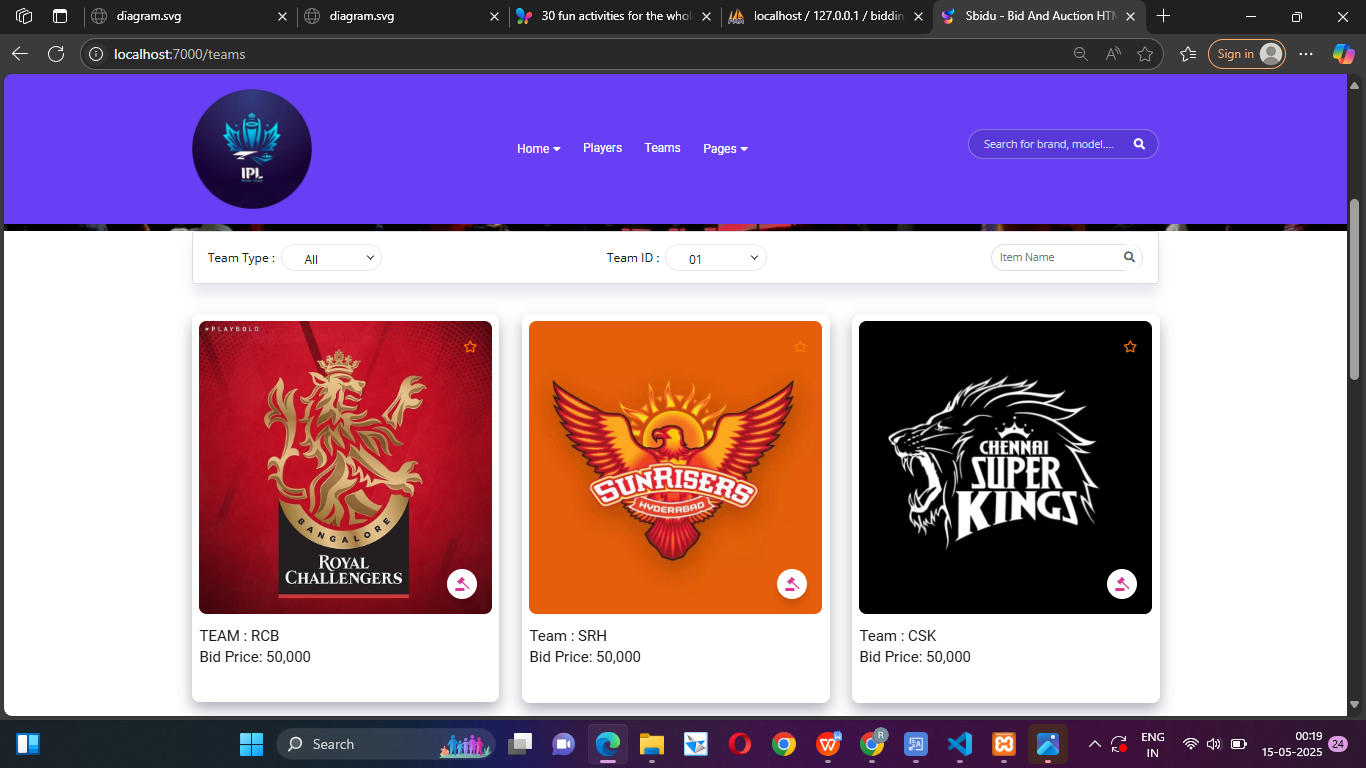
**Fig 4.6.4 Players list**

**4.6.5 Player detail**

****

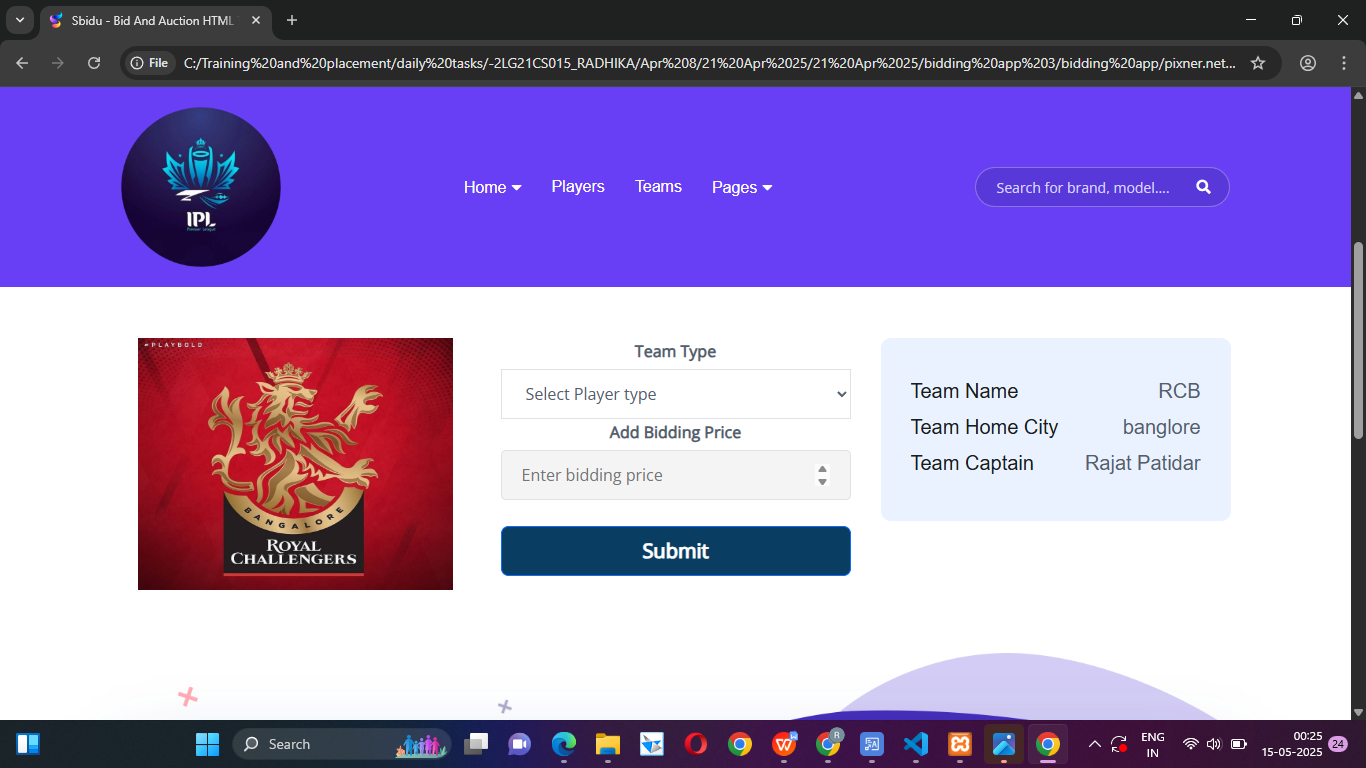
**Fig.4.6.5 Player detail**

**4.6.6 Team list**

****

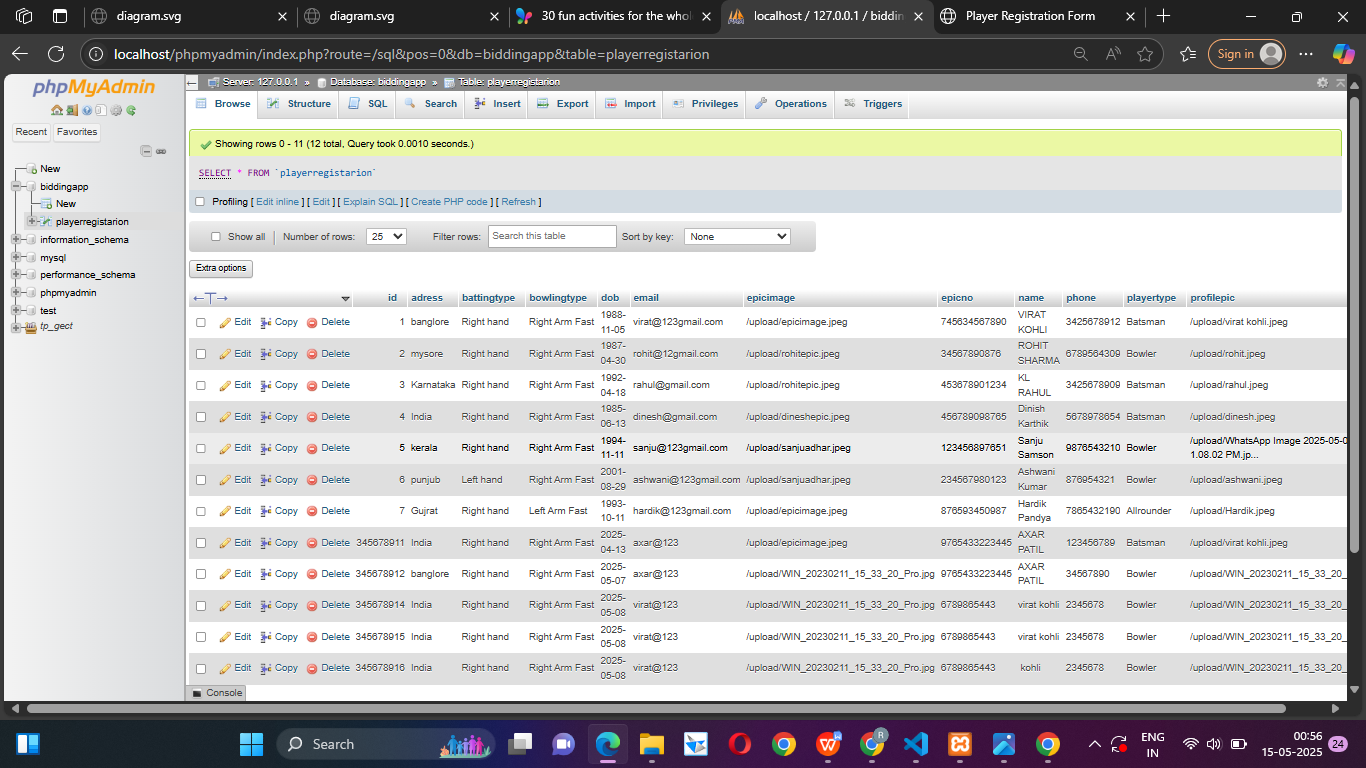
**Fig.4.6.6 Team list**

**4.6.7 Team detail**

****

**Fig.4.6.7 Team detail**

**4.6.8 Database**

****

**Fig.4.6.8 Database**

**CHAPTER 5**

**CONCLUSION**

The IPL Auction Bidding App developed using Java Spring Boot is a powerful and efficient solution for managing the online auction process of IPL players. The system provides a complete environment where administrators can control the flow of the auction, add players, set rules, and monitor team budgets, while team owners can actively participate in live bidding with ease. The backend is built using Spring Boot, which allows fast development, secure access using Spring Security and JWT, and real-time features using WebSocket for instant bid updates. The use of MySQL as the database ensures proper storage and management of player data, teams, and bids. On the frontend, HTML, CSS, and JavaScript create a user-friendly and responsive interface that allows smooth interaction for all users. Real-time updates and instant feedback keep the auction dynamic and competitive. This project not only improves the traditional manual auction system but also ensures fairness, speed, and accuracy throughout the process. It is scalable, so it can be deployed for larger events or leagues. Overall, the app demonstrates how modern technologies can be used to digitalize complex systems like sports auctions, making them more organized, transparent, and accessible from anywhere in the world.

**BIBLIOGRAPHY**

The development of a bidding app for the IPL Auction draws upon a wide range of technical and domain-specific resources. Information about the IPL auction structure, player data, and franchise operations was primarily sourced from the official website of the Indian Premier League (https://www.iplt20.com) managed by the BCCI. To understand the architectural and functional design of real-time auction systems, the work by Sharma and Kumar (2021) on real-time auction platforms was referenced. Additionally, foundational principles in software development, including requirement gathering, system design, and testing, were guided by "Software Engineering: A Practitioner’s Approach" by Pressman and Maxim (2014), and "Applying UML and Patterns" by Craig Larman (2005). For the technical implementation, open-source repositories from GitHub provided practical insights into WebSocket-based live bidding functionalities.

**APPENDIX**

The appendix provides supplementary information supporting the development of the IPL Auction Bidding App. It includes detailed system requirements, both hardware and software, specifying the use of Flutter for the frontend, Node.js or Django for the backend, and PostgreSQL or MongoDB as the database system. A use case overview describes the roles of administrators, franchise representatives, and optional viewer participation. A sample player data table illustrates the format used for displaying base prices, roles, and final bid outcomes. The auction process flow is outlined step-by-step—from player announcement and bidding initiation to final sale and roster update. Mock-ups and interface sketches (if available) showcase key components like the login screen, bidding dashboard, player analytics view, and admin panel. Additionally, the appendix outlines the AI recommendation module, which assists franchises in making data-driven decisions by analyzing player performance, team composition, and budget using machine learning algorithms such as Random Forest or regression models. These resources together support the app’s functionality, technical feasibility, and user experience strategy.