A close-up of a logo

AI-generated content may be incorrect.

**School of Computer Science and Engineering**

**COMMERCE BANK PRO (Static Main)**

**BY**

**ABSTRACT:**

CommerceBankpro is an encompassing online banking project offering a multitude of user-centric functionalities. The platform provides a seamless user experience through distinct pages, each serving a specific purpose. The "Register for the Account" page enables users to create new banking accounts, while the "Sign In" page grants access to registered accounts, unlocking a myriad of services. The pivotal "OpenBankAccount" page facilitates the process of opening new accounts by capturing user-provided details, subsequently storing the data securely in a MySQL Workbench database. Another key feature, the "UPI Payment" page, empowers users to transfer funds between accounts, utilizing unique primary keys known as UPIIDs. Furthermore, the "Split Payment" feature allows users to divide transaction amounts among multiple sources using the UPIID, ensuring flexibility in financial transactions. A comprehensive "Dashboard" consolidates these functionalities, offering an overview of transactions and user profile details. All information and user data are securely stored in the MySQL database, ensuring reliability and security. CommerceBankpro aims to deliver a comprehensive and user-friendly online banking experience, prioritizing convenience, security, and efficient financial management.

**Introduction:**

CommerceBankpro is an innovative banking project designed to offer a comprehensive suite of services catering to users' financial needs. With a user-centric approach, this platform boasts a seamless registration process allowing individuals to create new banking accounts effortlessly. The sign-in page provides secure access, enabling users to explore a myriad of functionalities conveniently. A key feature, the openbankaccount page, facilitates the hassle-free creation of new accounts by capturing and storing user data in a robust MySQL database. Moreover, the UPI-payment page empowers users to conduct swift and secure transactions between accounts using unique identifiers. Additionally, the split-payment functionality allows users to efficiently divide and manage transaction amounts. The dashboard serves as a centralized hub displaying transaction histories and user profiles, providing a holistic view of financial activities. With a focus on data security and user experience, CommerceBankpro redefines banking convenience and reliability.

**Work Breakdown Structure (WBS):**

**A diagram of a company

Description automatically generated**

**Module Composition:**

**Module 1: User Authentication**

* + Responsible for user registration and login functionalities.
  + Includes modules for user registration and user login.
  + Handles authentication and authorization processes.

**Module 2: Account Management**

* + Includes functionalities related to managing user accounts.
  + Sub-modules:
    - **Open Bank Account**:
      * Manages the process of opening a new bank account.
      * Collects user details for account creation.
      * Saves account information into the MySQL database.

**Module 3: Transaction Processing**

* + Manages different types of transactions.
  + Sub-modules:
    - **UPI Payment Page**:
      * Handles the transfer of funds from one account to another using UPI ID.
      * Validates and processes transactions, updating the database accordingly.
    - **Split Payment**:
      * Manages transactions where a total amount is split among multiple users.
      * Uses UPI ID to distribute payment amounts among users.
      * Records transaction details in the database.

**Module 4: Dashboard**

* + Provides an interface to display transaction histories and user profile details.
  + Sub-modules:
    - **Transaction History**:
      * Retrieves and displays transaction records from the database.
    - **Profile Details**:
      * Retrieves and displays user profile information.
    - **Dashboard Page**:
      * Aggregates the above sub-modules to provide a comprehensive dashboard view.

**Module 5: Database Interface**

* + Responsible for handling interactions with the MySQL database.
  + Sub-modules:
    - **Data Storage and Retrieval**:
      * Manages the storage and retrieval of user account details, transaction information, and profile data in the MySQL database.

**Module 6: Main Application**

* + Coordinates the interactions between different modules.
  + Contains the main logic and flow control of the application.
  + Handles user navigation between pages and modules.

**User-Centric Design Principles:**

**Clear and Intuitive Navigation:**

* Implemented a straightforward navigation structure, ensuring users can easily switch between different functionalities/pages like "Register for the Account," "Sign In," "Dashboard," "UPI Payment," etc.
* Utilized consistent placement of navigation elements such as menus, buttons, or links for seamless movement within the application.

**Responsive and Consistent Layout:**

* Ensured the UI is responsive, adapting seamlessly to various screen sizes (desktop, tablet, mobile) for a consistent user experience across devices.
* Maintained a consistent layout, typography, and color scheme throughout the application to establish visual coherence.

**Clear Presentation of Information:**

* Displayed essential information prominently, such as account balances, recent transactions, and user profile details, within the Dashboard.
* Organized transaction histories and account information in a structured and easily scannable format to aid quick comprehension.

**User-Friendly Forms and Input Fields:**

* Designed user input forms (e.g., registration, transaction details) with clear labels, placeholders, and error messages for a smooth data entry experience.
* Implemented form validations to provide real-time feedback and prevent submission errors, enhancing usability.

**Visual Cues and Feedback:**

* Incorporated visual cues (such as icons, tooltips, and animations) to guide users and provide feedback on interactions, ensuring a more intuitive experience.
* Employed loading indicators or progress bars for actions that might take time to complete, preventing user confusion or frustration.

**Accessibility and Readability:**

* Ensured sufficient color contrast and font sizes for readability, complying with accessibility standards.
* Utilized descriptive alt text for images and appropriate labels for screen readers, making the application more accessible to users with disabilities.

**Security Indicators:**

* Incorporated visual cues or indicators to reassure users about the security measures in place, such as SSL encryption, secure login indicators, or transaction security notifications.

**User Assistance and Help Sections:**

* Included contextual help sections, FAQs, or tooltips to assist users in understanding functionalities or resolving queries without leaving the interface.

**Interactive Components:**

**Dynamic Dashboard:**

* Developed an interactive dashboard presenting summarized account information and transaction histories through charts, graphs, or tabular representations for a comprehensive view of financial data.

**User Profile Management:**

* Enabled users to manage their profiles easily by providing options to update personal information, change passwords, or modify account settings within the interface.

**Transaction Features:**

* Incorporated intuitive interfaces for initiating UPI payments, splitting transactions, and conducting fund transfers between accounts with clear steps and confirmations.

**Responsive UI Elements:**

* Utilized interactive UI elements like buttons, dropdowns, sliders, and checkboxes to facilitate user interactions, ensuring a smooth and engaging experience.

**Iterative Improvement Process:**

**Usability Testing Sessions:**

* Conducted usability testing sessions with a group of targeted users, including both new users and those familiar with the application.
* Observed users interacting with the UI, noting areas of confusion, difficulties in navigation, or any hurdles encountered during tasks like account registration, fund transfer, or accessing the dashboard.

**Feedback Collection Mechanisms:**

* Implemented feedback collection mechanisms within the application, such as feedback forms, rating prompts after specific interactions, or surveys sent via email to gather user opinions and suggestions.
* Analyzed user support tickets or inquiries to identify recurring issues or user pain points reported through customer service channels.

**Iterative UI Refinements:**

**Prioritization of Improvements:**

* Identified high-priority issues or pain points reported by users based on the severity of impact on user experience and frequency of occurrence.
* Prioritized improvements that could significantly enhance user satisfaction or streamline critical functionalities.

**UI Adjustments and Feature Enhancements:**

* Addressed identified issues by making UI adjustments, such as redesigning confusing interfaces, improving form validations, or simplifying navigation flows.
* Implemented feature enhancements or additions based on user requests or suggestions to enhance the overall user experience.

**Prototype Testing and Validation:**

* Developed prototypes or mockups incorporating proposed UI changes and enhancements.
* Conducted validation testing sessions with users to gather feedback on the revised UI prototypes and ensure that the proposed changes effectively addressed identified issues.

**Continuous Monitoring and Future Iterations:**

* Implemented the refined UI changes into the live application and continued to monitor user feedback and analytics to assess the impact of improvements.
* Planned for future iterations, considering ongoing user feedback and technological advancements to further optimize the UI and maintain a user-centric approach.

**Platform or Software Used:**

**Operating System:** Microsoft Windows

**Frontend Development:**

* **HTML/CSS/JavaScript**: For creating the user interface and interactivity of the web pages.
* **React.js**: Frontend framework for building dynamic and responsive user interfaces.
* **Bootstrap**: Frontend frameworks for design and layout.

**Backend Development:**

* **Node.js**: For server-side scripting and handling backend functionalities.
* **MySQL Database:** For storing user account details, transaction information, and other relevant data.

**Development Tools:**

* **Git/GitHub:** Version control for collaborative development.
* **IDE:** Visual Studio Code for code development.

**User Experience (UX) and User Interface (UI) Design:**

* **Design Tools**: Figma for creating intuitive and user-friendly interfaces.
* **Responsive Design**: Ensures compatibility across various devices (desktop, tablet, mobile).

**Database Management:**

**MySQL Workbench:** To design and manage the MySQL database.

**Testing and Quality Assurance:**

* **Security Measures:** Implementation of security protocols to ensure data safety and prevent vulnerabilities.
* **Error Logging and Handling:** Implementation of error logging mechanisms.
* **Testing:** Unit testing, integration testing, and end-to-end testing for the application.

**USE-CASE DIAGRAM:**

**A diagram of a user

Description automatically generated**

**Analysis model:**

A diagram of a computer

Description automatically generated

**Sequence Diagram:**

**A diagram of a company

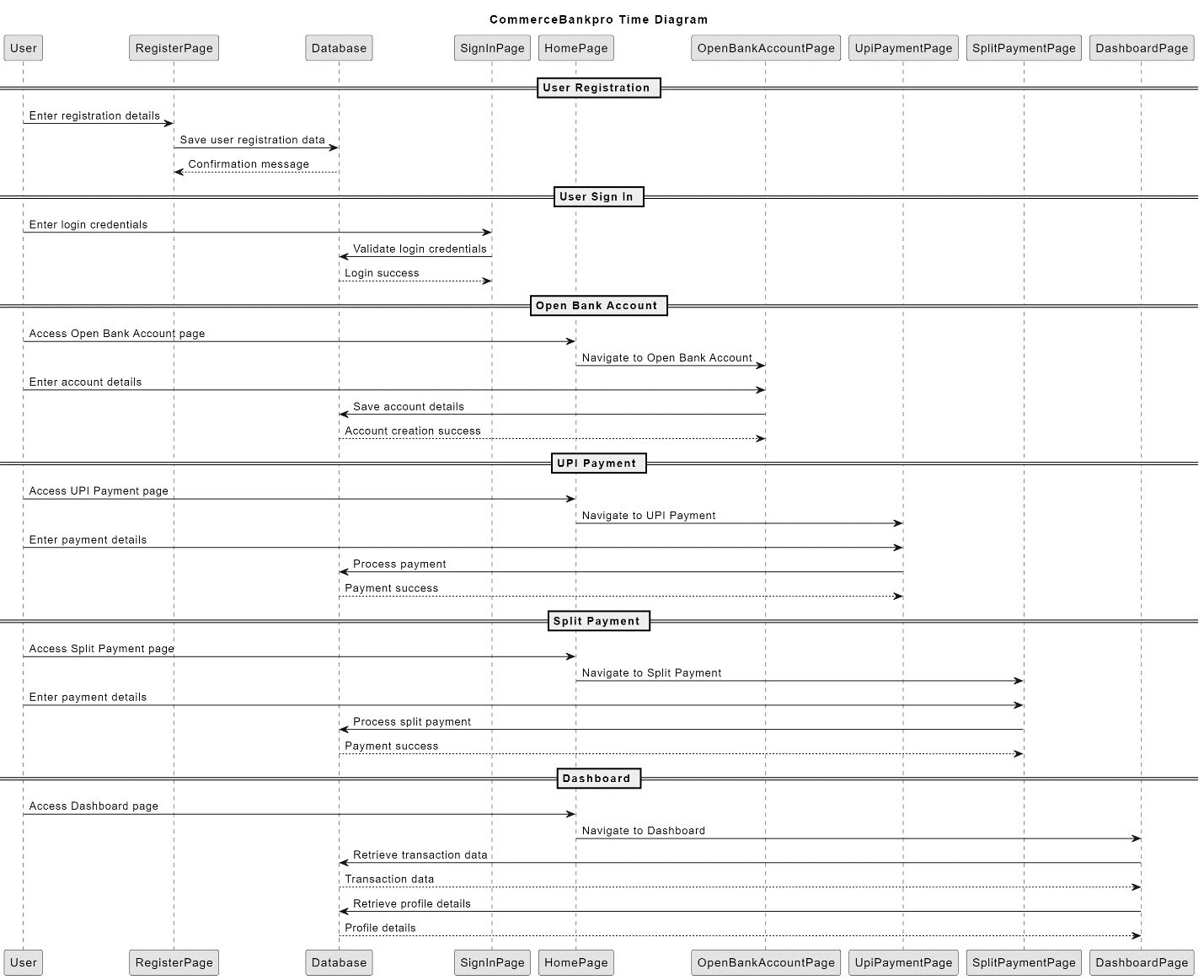
Description automatically generated**

**Class Diagram:**

**A diagram of a company

Description automatically generated with medium confidence**

**Time Diagram:**



**Object Diagram:**

A diagram of a network

Description automatically generated

**State Machine Diagram (User):**

A diagram of a website

Description automatically generated

**Data Flow Diagram:**

**A diagram of a payment method

Description automatically generated**