NT- Modernization of Financial Dashboard Application for a Leading Banking Client

Background

The client, a major financial banking institution, faced significant challenges with their existing financial dashboard application. This legacy system, crucial for displaying financial information to users, was plagued by reliability issues and an outdated, cluttered user interface. The application's poor performance and non-intuitive design led to frequent user complaints and hindered efficient account management. The system primarily displayed financial plans and account details without natively supporting transactions, further limiting its functionality.

Users Affected

The primary users of this application were the bank's customers who relied on the dashboard to manage their financial accounts and plan their finances. The inefficient system negatively affected customer satisfaction and trust, which are critical in the banking sector.

Approach

To address these issues, our Creospan consulting team embarked on a project to completely overhaul the old system using modern technologies. The project was structured around a clear timeline of 6-7 months, with the following strategic phases:

- **Requirement Gathering:** Collaborating with client stakeholders to understand the essential features and user needs.
- **System Design and Architecture Planning:** Developing a new architecture that supports scalability, performance, and ease of maintenance.
- **Development and Testing:** Implementing the frontend and backend services using React and SpringBoot, respectively, and ensuring thorough testing to avoid bugs that plagued the previous system.
- **Deployment and Monitoring:** Utilizing Pivotal Cloud Foundry for daily deployments and setting up a robust CI/CD pipeline using Bamboo and GitHub.

Solution

To address the identified challenges, Creospan implemented a comprehensive, step-by-step solution leveraging a modernized technology stack:

- **Frontend Design and Development:** React.js was chosen for the frontend to leverage its component-based architecture, which facilitates reusable UI components and enhances the user experience with a responsive and dynamic interface.

- **User Interface (UI):** The new UI was designed to be clean and intuitive, significantly improving on the cluttered and non-intuitive interface of the legacy system. Key features included:

    - **Dashboard:** Customizable panels displaying key financial indicators such as account balances, recent transactions, and investment summaries.
    - **Navigation:** A simplified navigation menu allowing users to easily switch between viewing different financial plans and accounts.
    - **Accessibility Enhancements:** Improved contrast, text resizing, and keyboard navigability to ensure accessibility for all users, including those with disabilities.

- **Backend Development:** SpringBoot was chosen for the backend to take advantage of its comprehensive infrastructure support for developing microservices, including integration with various backend systems and databases.

- **Microservices Architecture:** The backend was segmented into three key microservices. Each microservice was designed to be loosely coupled and independently scalable, ensuring that the application can handle increasing loads efficiently and maintain high availability:
    - **Account Information Service:** Manages real-time retrieval and updates of user account details.
    - **Financial Planning Service:** Handles the logic for financial forecasting and scenario modeling based on user inputs and market data.
    - **User Management Service:** Responsible for authentication, user profile management, and security compliance.

- **Database Integration:** Oracle was retained for its robustness and reliability. A new schema was designed to optimize data retrieval and updates, which are critical for the real-time nature of the financial data presented in the dashboard.

- **Cloud Platform:** The application was deployed on Pivotal Cloud Foundry, enabling easy scaling and management of the application's environment.
    - **Environment Isolation:** Separate environments for development, testing, and production ensured that changes could be deployed and rolled back without affecting the live system.

- **CI/CD Pipeline:** A continuous integration and deployment pipeline was set up using Bamboo and GitHub, facilitating regular updates with minimal downtime. This included:
    - **Automated Testing:** Integration and unit tests are run automatically, ensuring that every release is as stable as possible.
    - **Rollback Capabilities:** Quick rollback mechanisms in case of any issues during new releases to minimize impact on end-users.

- **Security Measures:** Given the sensitive nature of financial data, security was a paramount concern:
    - **Data Encryption:** At-rest and in-transit encryption of sensitive data to meet regulatory and compliance requirements.
    - **Authentication and Authorization:** Implementation of OAuth for secure access and JWT (JSON Web Tokens) for maintaining user session integrity.
    - **Regular Audits:** Continuous security audits and compliance checks to ensure that the application adheres to the latest security standards.

Benefits

The modernized financial dashboard application brought significant business benefits:

- **Enhanced User Satisfaction:** The redesign of the UI improved user experience significantly, leading to a 35% increase in customer satisfaction and a 25% reduction in customer churn. Support queries dropped by 30%, reflecting greater user confidence and fewer issues.

- **Increased System Reliability:** Modern development practices reduced system downtime by 40% and maintenance costs by 20%, enhancing the application's reliability and operational readiness.

- **Operational Efficiency:** The introduction of microservices architecture streamlined updates, reducing deployment times by 50% and allowing for 30% quicker bug resolution, which minimized disruption and enhanced productivity.

- **Scalability:** The cloud-based infrastructure successfully managed a 50% increase in transaction volumes during peak periods without performance loss, demonstrating strong scalability without significant additional investments.

- **Real-Time Data Access:** Improved database integration and microservices architecture enabled real-time access to financial data, reducing data retrieval times by 60% and allowing customers to make timely and informed financial decisions.

- **Enhanced Security:** The implementation of robust security measures, including data encryption and OAuth, ensured that sensitive financial information is protected, increasing customer trust and compliance with regulatory requirements by 20%.

- **Better Customer Support:** The new system's reliability and improved user interface reduced the need for customer support interventions by 30%, freeing up resources to focus on more complex customer issues and improving overall support quality.