**SIMPLIFIED - MAY 19**

* **Case Study 1 - Revamping Banking Experience for a F500 Financial Services Company**
  + **Challenge**
    - Users of the legacy financial transaction system of our Fortune 500 banking client were dissatisfied with the service and often complained abou the transaction errors, cross-platform compatibility issues and application outages
  + **Solution**
    - Creospan provided a team of consultants (8) and they followed our standardized framework (Assessment, Gap Analysis, Solution Proposal, Execution, Iteration), identified the root cause of this issue was linked with the limitations associated with 1) API Connections 2) Application Cache (Mark Logic)
    - Our team deployed a robust software performance monitoring system (Dyna Trace) to automate and accelerate the root cause identification process
    - After thorough analysis of data seen in Dyna Trace, team learned that the intra-component API connections in the web application and the application cache contributed heavily to these issues,
    - Team scaled API connections and eliminiated the application cache (Mark Logic) which was negatively impacting the database and thus preventing the users from accessing their data in a consistent manner with accuracy
    - Team also refactored the code, redesigned the APIs, and aligned the architecture to ensure holistic performance
  + **Impact**
    - Customer Satisfaction - 80,000 users & 70k monthly transactions
    - Robust Database Operation - Sig. drop in complaints and support tickets
    - Operational Efficiency
  + **Tech Stack**
    - React, SpringBoot, Oracle DB, Eureka, Redis

**DETAILS**

**Revamping the Banking Experience: A User-Centric Approach to Enhancing Legacy Web Application**

**Business Challenge:**

* Our client, a prominent Fortune 500 financial institution, offered a web application for investment management. However, they encountered a substantial surge in customer complaints pertaining to the user interface and performance of their existing application.
* Recognizing the urgency to address these issues, the client sought a swift solution to minimize customer terminations and prevent potential revenue loss.

Describe more about the application

Add monthly user, and transactions

**Our Approach & Solution:**

• Consulting Session

* Through an extensive consulting session, we identified gaps in the system architecture, leading us to the conclusion that rearchitecting system APIs and components was necessary to minimize transaction failures and reduce latency.

• Strategy

Based on our assessment we devised the following strategies:

* Adopt a bi-weekly Sprint approach.
* Implement software tracking to monitor User Experience effectively.
* Refactor microservices using industry best practices.
* Optimize queries to enhance efficiency and performance.
* Implement comprehensive test coverage.
* Focus on creating Minimum Viable Products (MVPs) more frequently, rather than waiting for quarterly feature deployments.

Analyze architecture, analyze queries

Reword

solution

**Execution:**

Strategy should be addressed in execution

• After analyzing the client's requirements and the web application, we assembled a team of 6 experts, including an Architect, Scrum Master, QA, and 3 Full Stack Developers.

• This team was responsible for improving performance through API rearchitecting, tech stack upgrades, and component optimization, aimed at reducing transaction failures and latency.

• Additionally, the team evaluated the root cause of application crashes by introducing a robust software monitoring tool – Dynatrace.

• By utilizing Dynatrace, we discovered that the crashes were linked with connections between different parts of the application. The purpose of these connections is to enable communication and data transfer between various parts of the application. Scaling the number of active connections from 4 to 8 resolved the crashes, improving the application's stability and performance.

• In addition to this, the team found that the existing data management platform – MarkLogic, was not properly implemented and hence was impacting software operations and reliability.

• To make the system more reliable and stable, we decided to sunset MarkLogic instances and thus enhance software performances.

Provide more context

**Business Benefits:**

**Improving Customer Satisfaction**

• The integration of Dynatrace provided a comprehensive tracking system to monitor user experience, enabling real-time insights into customer interactions and facilitating prompt identification and resolution of any issues, thereby enhancing overall customer satisfaction.

• By creating MVPs more frequently, rather than waiting for quarterly deployments, the bank achieved accelerated time-to-market for new features, which contributed to higher customer satisfaction and a competitive advantage.

**Enhancing Application Performance**

• Refactoring the microservices using industry best practices enhanced the system architecture, resulting in improved scalability, maintainability, and code quality.

• The optimization of queries led to enhanced performance and efficiency, enabling faster data retrieval and processing, ultimately benefiting both the users and the bank.

• Adopting TDD and writing extensive test cases ensured robustness and reduced the risk of regression, thus enhancing software reliability and stability.

• Removing MarkLogic eliminated the need for additional licensing fees, maintenance costs, and infrastructure requirements associated with its implementation. This resulted in immediate cost savings for our client, improving their financial position.

Overall, these measures resulted in tangible business benefits, including an improved user experience, increased development speed, enhanced product quality, and ultimately, greater customer satisfaction.

More about application

Any number of performance improvement data

**Technology Stack: ReactJS, TypeScript, Java, Spring Boot, Spring Gateway, Eureka, Config Server, Redis Microservices, Bamboo, Oracle DB**

**Project Timeline: 1 year**

**Industry: Finance | Domain: Accounts**

**Add entire team, application name**

**Key Contributors: Sunil Kashetty**