Webmob Software Solutions
Engineering the Future

DIGITAL WEALTH MANAGEMENT PLATFORM

TABLE OF CONTENT

Case Study

DIGITAL WEALTH MANAGEMENT PLATFORM

Webmob Software Solutions

Bestech Business Tower, Suite No. A-829, Sector - 66, Mohali, Punjab 160066

3030 K Street NW, Suite 102 Washington, DC 20007

0172-4045981

info@webmobinfo.ch

OVERVIEW

The platform is designed as a centralized marketplace for banking products, focusing on Fixed Deposits (FDs) and Bonds. It brings together offerings from multiple banks and NBFCs into one seamless digital experience, eliminating the need for investors to navigate individual portals. Users can easily explore and compare interest rates, evaluate options, and make investments securely within the application.

Integrated APIs ensure real-time updates on product availability, maturity, and redemption, while a dedicated portfolio dashboard allows users to track and manage their investments in one place. To enhance transparency and engagement, the platform features an automated communication system with SMS and email notifications, complemented by manual outreach options when needed. By combining convenience, security, and compliance, the solution redefines how individuals invest in fixed-income products.

ABOUT WEBMOB

Webmob has emerged as a service delivery pioneer in this dynamic fintech industry, serving numerous laurelled clients in Europe and the Middle East. With AI/ML-powered, Cloud-native, and Blockchain in our stack, Webmob provides cutting-edge solutions to fulfil the customer's advanced and disruptive requirements.

Particularly for the FINTECH industry, Webmob offers unparalleled robust solutions in Trade Finance, Money Market, Fiduciary, Commercial Real Estate Loan Tokenization, and NFT marketplace on top Blockchains. As of today, Webmob is equipped with a fully equipped R&D lab, aka WikiDLT.com, and consulting certified professionals, especially to explore new possibilities for innovative Blockchain implementation.

Business Needs

- A single platform to compare and invest in FDs and Bonds from multiple banks and NBFCs.
- API integrations for real-time product details, bookings, and redemption updates.
- Automated SMS and email alerts, along with manual communication options.
- Strong compliance and data security for all transactions.
- A dashboard to track portfolios, maturity dates, and withdrawals.

Solutions

- Developed a unified digital marketplace to simplify FD and Bond investments.
- Integrated APIs with banks and NBFCs for secure booking and real-time updates.
- Built automated SMS and email notifications with manual messaging options.
- Implemented strong security measures to ensure compliance and protect user data.
- Designed a portfolio dashboard for users to manage investments and monitor maturity.

Benefits

- Simplified investment journey with access to multiple banks and NBFCs on one platform.
- Faster booking process with real-time updates on products, maturity, and withdrawals.
- Improved user engagement through timely automated and manual communication.
- Enhanced trust and reliability with secure, compliant transaction handling.
- Clear portfolio visibility, helping users track and manage fixed-income investments with ease.

OUR SOLUTIONS

To address the client's requirements, Webmob delivered a comprehensive platform that combines convenience, security, and compliance. The solution not only simplifies how users discover and book Fixed Deposits and Bonds but also ensures transparency through real-time updates and automated communication. With strong API integrations and a dedicated portfolio dashboard, the platform provides a complete digital investment experience.

01

Centralized Marketplace Development

We built a unified platform where users can explore, compare, and invest in Fixed Deposits and Bonds offered by multiple banks and NBFCs. This eliminated the need to navigate between separate portals, creating a seamless digital investment experience.

02

Robust API Integrations

Secure APIs were integrated with financial institutions to fetch real-time product details, process investment bookings, and update redemption or premature withdrawal statuses. This ensured accuracy, transparency, and reliability across the system.

03

Automated Communication Module

An intelligent notification system was designed to trigger SMS and email alerts at critical stages such as onboarding, booking confirmations, maturity, and redemption. Manual communication options were also added to support customer teams.

04

Security and Compliance Measures

Strong data protection protocols, encryption mechanisms, and compliance-focused processes were implemented to safeguard transactions and maintain trust. The platform strictly adheres to financial regulatory standards.

05

Portfolio Management Dashboard

A user-friendly dashboard was developed to provide complete visibility of active and mature investments. Users can track returns, monitor maturity dates, and manage withdrawals efficiently from a single interface.

TECHNOLOGY

Frontend – React.js

We used React.js to build a responsive and user-friendly interface. It ensures a smooth experience while users browse FD and Bond options, compare returns, or track investments through the portfolio dashboard.

Mobile – Flutter

The mobile application was developed using Flutter to deliver a consistent cross-platform experience. It allows both Android and iOS users to access investment features and manage their portfolios seamlessly.

Backend – Node.js with Express.js

The backend was built using Node.js and Express.js to handle server-side operations and manage communication between banks, NBFCs, and the app. It ensures fast, secure, and reliable processing of all transactions.

Database – PostgreSQL and MongoDB

PostgreSQL was implemented to manage structured financial and transactional data with accuracy. MongoDB complements it by efficiently handling unstructured data, logs, and user communication records.

Hosting – AWS (Amazon Web Services)

The platform is hosted on AWS to guarantee scalability, security, and uninterrupted availability. AWS infrastructure ensures smooth performance even as the user base and transaction volume grow.

Communication Services – SMS and Email Gateways

Trusted gateways were integrated for automated SMS and email notifications. This keeps users informed throughout the investment lifecycle, from onboarding to redemption.

Challenges

Before the platform was developed, investors faced significant challenges when exploring Fixed Deposits and Bonds. The first major issue was fragmented accessibility. Users had to visit multiple banks or NBFC websites individually to compare rates, check terms, and initiate investments. This not only consumed time but also created confusion, as information was scattered across different sources. There was no single interface that provided a consolidated view of options, making decision-making cumbersome and inefficient.

The second challenge was a lack of transparency and communication. Once investments were booked, customers often received limited updates about their status, maturity, or redemption timelines. Manual communication from institutions led to delays, while portfolio tracking across different providers was nearly impossible. These gaps highlighted the need for a unified, secure, and transparent digital solution.

QA PROCESS

Our QA process involves a systematic approach encompassing various stages to thoroughly assess the money market platform's functionality, security, and user experience.

01 Test Planning
We defined the scope of testing, identified objectives, allocated resources, and developed a comprehensive test plan outlining our approach, timelines, and deliverables.

02 Requirement Analysis
We reviewed the requirements documentation to understand the platform's expected behaviour and ensured our team accurately captured all functional and non-functional requirements.

03 Test Environment Setup
We established testing environments mirroring the production environment, installed the necessary software, configured databases, and ensured the availability of test data representing various marketplace scenarios.

04 Test Case Design
We developed detailed test scenarios and cases covering functional workflows, boundary conditions, error handling, and exception scenarios, prioritising based on criticality and risk.

05 Functional Testing
We executed test cases to verify the functionality of different modules and features, including various workflows, integration with external systems, and compliance with regulatory standards.

06 User Interface Testing
We evaluated the user interface for usability, accessibility, and responsiveness, ensuring consistency in design elements, layouts, and navigation across different screens.

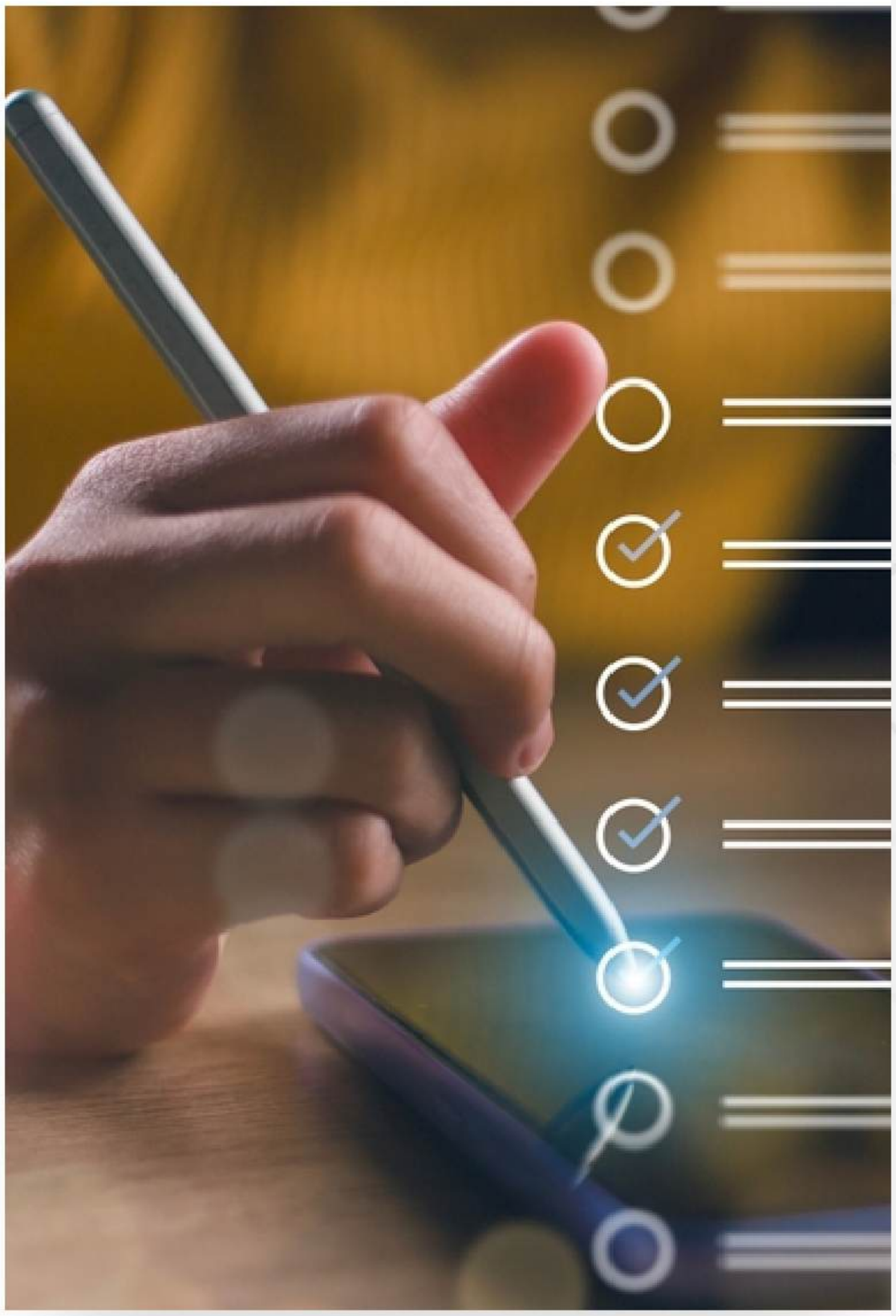
07 Security Testing
We performed security assessments, testing authentication and authorisation mechanisms, data encryption, and secure communication protocols, and conducted penetration testing to assess resilience to security breaches.

08 Regression Testing
We re-ran previously executed test cases to ensure new changes did not introduce regressions, automating regression test cases where feasible and validating backward compatibility.

09 Integration Testing
We tested data exchange mechanisms and validated data consistency and integrity across integrated systems, including file uploads, API calls, and message queues.

10 Documentation and Reporting
We maintained a detailed documentation of test cases, results, and defects, generated test reports summarising test coverage and provided stakeholders with regular updates on testing progress and identified issues.

11 User Acceptance Testing (UAT)
We collaborated with end-users and stakeholders to conduct UAT, obtaining feedback on the platform's functionality, usability, and performance, ensuring alignment with user expectations.



SECURITY TESTING OF THE PLATFORM

01 API Testing

Objective: Evaluate the functionality, reliability, security, and performance of APIs used in the platform.

Tools:

- **Postman:** Automated testing tool for API automation testing, enabling comprehensive testing of API endpoints and payloads.
- **SoapUI:** Another automated testing tool suitable for API testing, providing features for functional testing, load testing, and security testing.

02 Penetration Testing (PenTesting)

Objective: Identify and exploit vulnerabilities in the platform to assess its security posture.

Tools:

- **Burp Suite:** A comprehensive toolkit for web application security testing, including manual and automated vulnerability scanning, request interception, and exploitation of security flaws.
- **Metasploit:** A penetration testing framework offering various exploits and payloads for testing network and application security.

03 Patch Testing

Objective: Verify the effectiveness of security patches applied to the platform.

Process:

- Testing patches on a sandbox or staging environment ensures they do not introduce regressions or new vulnerabilities.
- Automated and manually tested critical functionalities affected by the patch to ensure they operated as expected.

04 Third-Party Testing

Objective: Gain independent verification and validation of the platform's security measures.

Process:

- Engaging external security firms or independent security researchers to conduct thorough security assessments, including penetration testing, code review, and vulnerability scanning.
- Utilising bug bounty programs to incentivise external security researchers to discover and responsibly disclose security vulnerabilities in the platform.

05 Source Code Testing

Objective: Evaluate the security of the platform's source code to identify and remediate vulnerabilities and ensure robust protection against potential threats.

Process:

- The source code testing process for the platform begins with configuring and integrating tools like SonarQube and Checkmarx into the development environment.

Tools:

- **SonarQube:** Analyzes the platform's source code for bugs, vulnerabilities, and code smells, providing insights into code quality and security..
- **Checkmarx:** A static application security testing (SAST) tool that identifies security vulnerabilities in the source code, helping developers remediate potential issues before deployment.

06 Network Testing

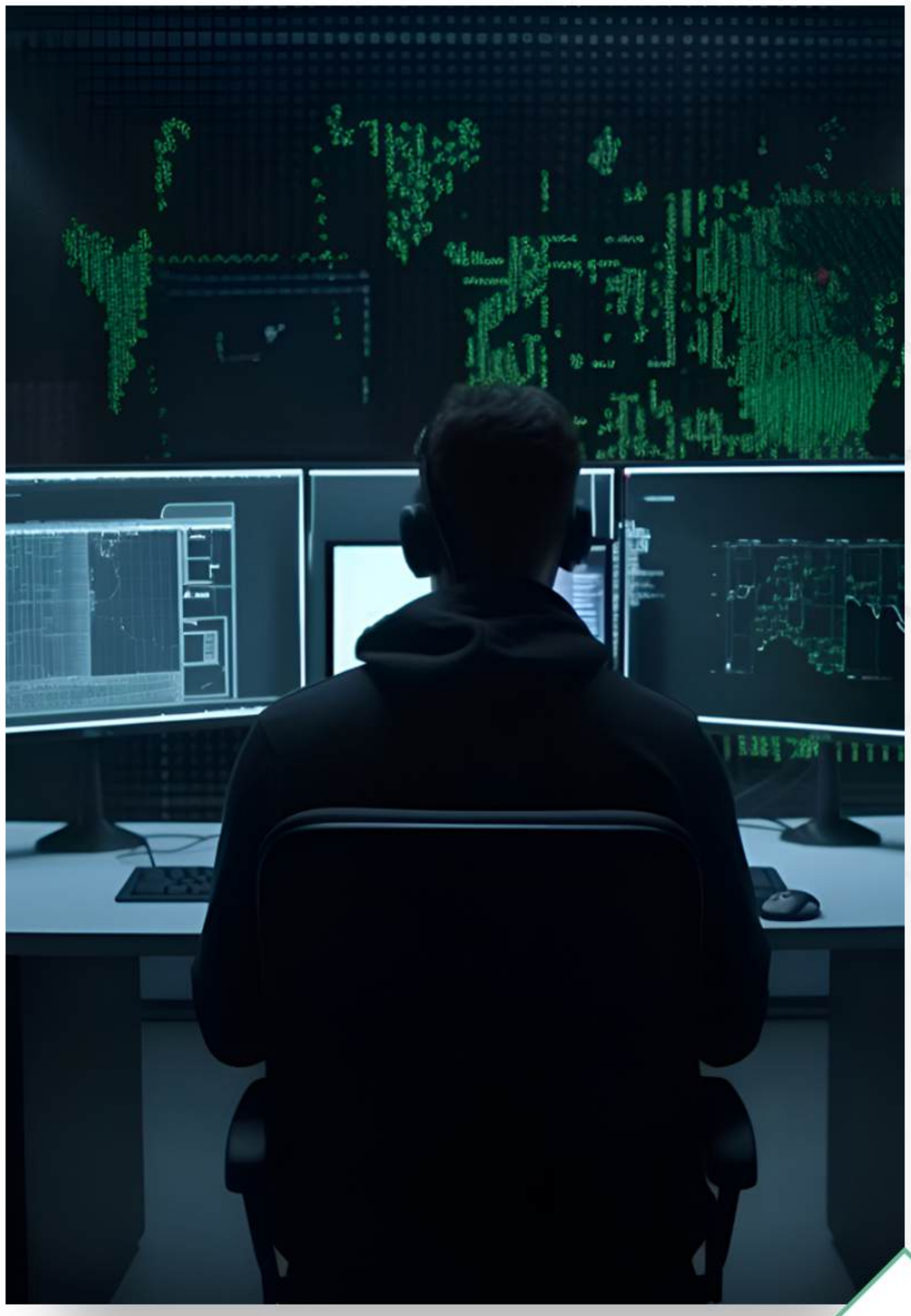
Objective: The primary objective of network testing is to assess the security and resilience of the platform's network infrastructure, ensuring protection against potential threats and vulnerabilities.

Process:

- Network testing begins by examining the network infrastructure's configuration and setup to identify any potential weaknesses or misconfigurations.
- Comprehensive scans are conducted using specialised tools to analyse server ports, configurations, versions, and subdomains within the network.

Tools:

- **Nessus:** A powerful scanning tool utilised for comprehensive network scans, providing detailed insights into potential security risks and vulnerabilities within the network infrastructure.
- **Nmap:** Another widely used scanning tool that enables thorough examination of network configurations and identifies potential security loopholes and weaknesses.



DEVELOPMENT PHASE

01 Requirement Gathering

Requirements were gathered through meetings and discussions to understand the platform's functional and non-functional aspects.

02 System Design

Based on the gathered requirements, system architecture and design were finalised. It included defining the database schema, application modules, and integrations with external systems.

03 Coding

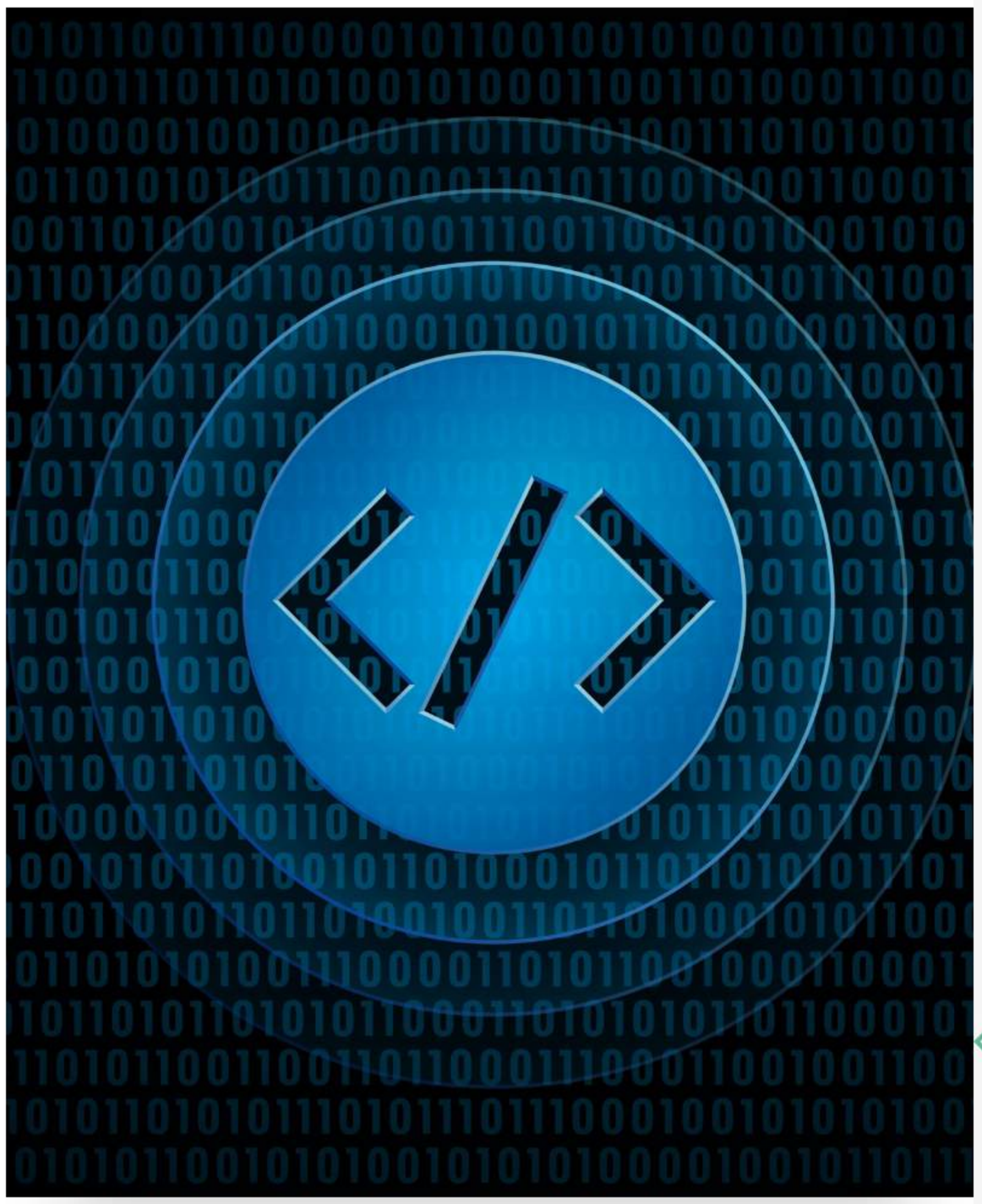
Our developers wrote code according to the design specifications using programming languages & frameworks suitable for the platform's requirements.

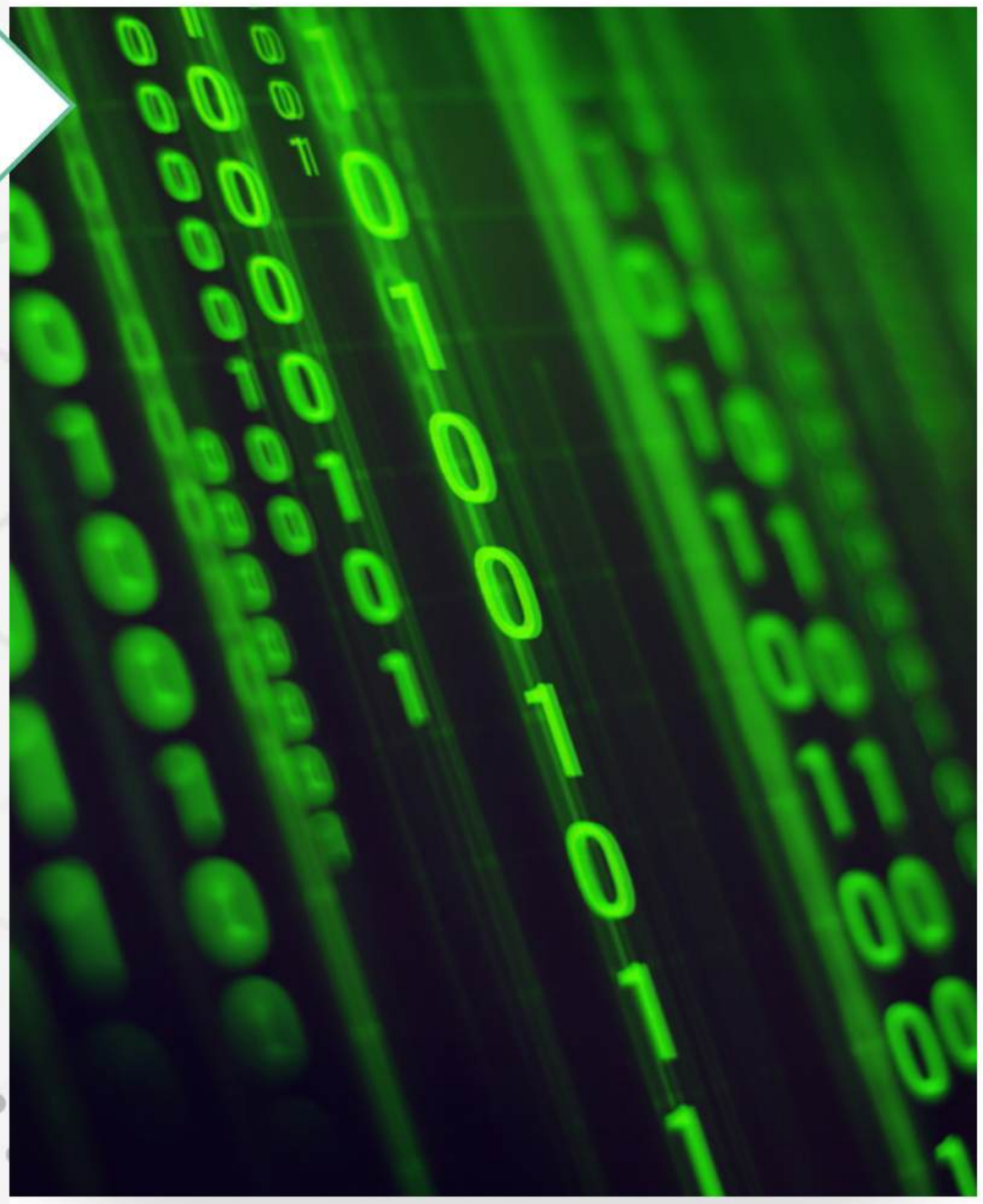
04 Quality Assurance

Our QA engineers conducted comprehensive platform testing, including source code, functional, security, and performance testing, that helped us identify & resolve any defects or issues.

05 Review & Integration

The platform has undergone thorough code reviews to ensure its stability and performance. Our team addressed any feedback or issues identified during testing and made necessary integrations.





DEPLOYMENT PHASE

01 Preparation

The necessary infrastructure and environments were set up, including development, staging and production.

02 Deployment Planning

We have created a pitch-perfect deployment plan outlining the steps and procedures for deploying the platform to the production environment.

03 Release Management

Our team deployed the platform to the product environment following the deployment plan. It involved deploying code, configuring servers, and ensuring all dependencies were met.

04 Monitoring and Optimisation

After deployment, our team continuously monitored the platform for performance, security & stability. We promptly addressed any issues or anomalies and made necessary changes.

05 Post-Deployment Review

We conducted a post-deployment review to assess the deployment process's success and gather user feedback. Additionally, our team documented any lessons learned for future deployments.

PROJECT METHODOLOGY

Our team adhered to an Agile methodology during this project, fostering efficient and iterative development. We structured our workflow around sprints, each lasting two weeks, allowing us to focus on specific features and functionalities. Regular feedback sessions with the client, occurring after every sprint, were integral to our process. It ensured our work aligned with the client's evolving requirements and expectations.

Additionally, we employed the project management tool Trello to streamline collaboration and task management, facilitating transparent communication and real-time progress tracking. These practices enabled us to maintain a dynamic and responsive development approach, ultimately delivering a high-quality solution that effectively met the client's needs.

TIMELINE

01 **Total months:** 5 months (Ongoing)

02 **No. of Resources:** 6 (2 Backend, 2 Frontend, 1 Designer, 1 QA, 1 Mobile App Developer)

03 **Experience of Resources:** Each team member has 4–7 years of relevant domain experience.