Mobile Application Development Proposal

Business Analysis and Information

14 November 2023

Table of Contents

[Document Control 3](#_Toc149639142)

[Proprietary Information 3](#_Toc149639143)

[Document Version Control 3](#_Toc149639144)

[Document History 3](#_Toc149639145)

[Contact Information 3](#_Toc149639146)

[Introduction 4](#_Toc149639147)

[Project Overview 4](#_Toc149639148)

[Project Objectives 4](#_Toc149639149)

[Stakeholders 5](#_Toc149639150)

[Scope 5](#_Toc149639151)

[Description 5](#_Toc149639152)

[Deliverables 5](#_Toc149639153)

[Exclusions 5](#_Toc149639154)

[Requirements 6](#_Toc149639155)

[Functional Requirements 6](#_Toc149639156)

[Non-Functional Requirements 6](#_Toc149639157)

[UX Requirements 6](#_Toc149639158)

[Timeline 6](#_Toc149639159)

[Milestones & Schedule 6](#_Toc149639160)

[Budget 7](#_Toc149639161)

[Cost Estimation 7](#_Toc149639162)

[Budget Constraints 7](#_Toc149639163)

[Risk Management 7](#_Toc149639164)

[Risk Identification 7](#_Toc149639165)

[Risk Analysis 7](#_Toc149639166)

[Quality Assurance 8](#_Toc149639167)

[Quality Goals 8](#_Toc149639168)

[Quality Control Measures 8](#_Toc149639169)

[Resource Plans and Strategy 9](#_Toc149639170)

[Test Strategy 9](#_Toc149639171)

[Story Board 10](#_Toc149639172)

[Site Map 11](#_Toc149639173)

# Document Control

## Proprietary Information

The contents of this document and any information preceding this document is considered proprietary information and should only be disclosed outside of the company with trusted and approved service providers via suitable secure trusted channels.

## Document Version Control

|  |  |
| --- | --- |
| **Document Information** | |
| **Data Classification** | **Confidential** |
| **Document Title** | **Mobile Application Development Proposal** |
| **Author** | Dean Reid |

## Document History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Status** | **Date** |  | **Author** |
| 0.1 | DRAFT | 30-Oct | Initial Document | Dean Reid |
|  |  |  |  |  |

## Contact Information

|  |  |  |
| --- | --- | --- |
| **Document Information** | | |
| **Primary Contact** | **Title** | **Email** |
| Dean Reid | Project Owner | xxx@xxx.com |
| **Secondary Contact** | **Title** | **Email** |
| Bob Geldof | Software Engineer Manager | xxx@xxx.com |

# Introduction

## Project Overview

The project is centred around the development of an innovative application that aims to redefine the online shopping experience for various digital services. The primary goal is to simplify the process of purchasing and managing Virtual Private Servers (VPS), Bare Metal services, web hosting, game hosting, and server hardware.

The application will feature a user-friendly interface that makes it easy for users to navigate through a wide range of digital services and products. This includes everything from VPS and Bare Metal services to web and game hosting solutions. Users can browse through these categories, compare offerings from different providers, and choose the one that best suits their needs.

In addition to hosting services, users can also purchase server hardware directly through the application. The platform will provide detailed specifications and pricing for a variety of servers, allowing users to make informed decisions based on their specific requirements.

Once a purchase is made, users can manage their products or services directly within the application. This includes setting up and configuring their VPS or Bare Metal servers, managing their web or game hosting settings, and scheduling maintenance or upgrades for their server hardware.

The application will also incorporate a secure payment system for safe and easy transactions. Whether users are buying a new server or paying for their hosting service, they can complete their purchase with just a few clicks.

In essence, this project aims to create a comprehensive platform that serves as a one-stop-shop for all your digital service needs, combining convenience, variety, competitive pricing, and top-notch customer service.

## Project Objectives

1. **Simplify Digital Services Shopping**: To create an intuitive and user-friendly platform that simplifies the process of purchasing and managing various digital services such as VPS, Bare Metal services, web hosting, game hosting, and server hardware.
2. **Wide Range of Services**: To provide users with a wide range of digital services and products to choose from, ensuring they can find the solutions that best fit their needs within a single platform.
3. **Informed Decision Making**: To enable users to make informed decisions by providing detailed specifications, pricing comparisons, and reviews for various digital services and server hardware.
4. **Efficient Management Tools**: To offer efficient tools for users to manage their purchased services or products directly within the application, including setting up and configuring servers, managing hosting settings, and scheduling maintenance or upgrades.
5. **Secure Transactions**: To ensure all transactions made within the application are secure, always protecting users’ personal and financial information.
6. **Excellent Customer Service**: To provide top-notch customer service, ensuring user queries and issues are addressed promptly and effectively.

## Stakeholders

|  |  |  |
| --- | --- | --- |
| **Primary Stakeholders** | | |
| **Dean Reid** | **Project Owner / Manager** | [xxx.xxx@xxx.com](mailto:xxx.xxx@xxx.com) |
| Bruce Wayne | Lead Developer | [xxx.xxx@xxx.com](mailto:xxx.xxx@xxx.com) |
| **Thor Odinson** | **UI / UX Designer** | [xxx.xxx@xxx.com](mailto:xxx.xxx@xxx.com) |
| Reed Richards | Quality Assurance Lead | [xxx.xxx@xxx.com](mailto:xxx.xxx@xxx.com) |
| **Tony Stark** | **Marketing Lead** | [xxx.xxx@xxx.com](mailto:xxx.xxx@xxx.com) |

# Scope

## Description

The project involves the development of a web application that provides web hosting, Virtual Private Server (VPS), and bare metal services. The application will allow users to manage their hosting services, configure their VPS or bare metal servers, and access customer support.

## Deliverables

1. **Web Application**: A fully functional web application that allows users to purchase and manage web hosting, VPS, and bare metal services.
2. **User Interface (UI)**: An intuitive and user-friendly interface that provides easy navigation and management of services.
3. **Payment Gateway Integration**: Integration with popular payment gateways for secure transactions.
4. **Customer Support Module**: A module for users to raise tickets, chat with support agents, and access help documentation.
5. **Server Management Tools**: Tools for users to manage their VPS or bare metal servers, including start/stop/restart functions, OS re-installation, and usage statistics.
6. **Security Features**: Implementation of security measures such as SSL, firewall configurations, and DDoS protection.
7. **Documentation**: Comprehensive user guides and technical documentation for the web application.

## Exclusions

1. **Physical Infrastructure**: The project does not include the provision of physical infrastructure such as data centres or servers.
2. **Third-Party Software Licenses**: Any third-party software licenses required for the server configurations are not included.
3. **Maintenance Services**: Ongoing maintenance and support services post-deployment are not part of this project scope.
4. **Data Migration**: Migration of user data from other hosting services is not included in this project.

# Requirements

## Functional Requirements

1. **User Registration and Authentication**: Users should be able to register, log in, and manage their accounts.
2. **Service Selection**: Users should be able to browse, select, and purchase web hosting, VPS, and bare metal services.
3. **Server Management**: Users should have the ability to manage their servers including start/stop/restart functions, OS re-installation, and viewing usage statistics.
4. **Payment Processing**: The application should integrate with a payment gateway for processing transactions.
5. **Customer Support**: Users should be able to raise tickets, chat with support agents, and access help documentation.

## Non-Functional Requirements

1. **Performance**: The application should load quickly and respond to user actions without delay.
2. **Security**: User data should be protected with encryption, and transactions should be secure.
3. **Scalability**: The application should be able to handle many users and high traffic.
4. **Reliability**: The application should have high uptime and low failure rate.
5. **Maintainability**: The code should be well-structured and documented for easy maintenance.

## UX Requirements

1. **Intuitive Navigation**: The application should have a clear and intuitive navigation structure.
2. **Responsive Design**: The application should be accessible and usable on various devices and screen sizes.
3. **Consistent Design**: The design elements such as colour scheme, typography, and layout should be consistent across the application.
4. **Feedback Mechanisms**: The application should provide clear feedback to user actions.
5. **Accessibility**: The application should comply with accessibility standards to cater to users with disabilities.

# Timeline

## Milestones & Schedule

1. **Project Kick-off**: Week 1
2. **Requirement Gathering and Analysis**: Weeks 2-3
3. **Design Phase**: Weeks 4-6
4. **Development Phase**: Weeks 7-14
5. **Testing Phase**: Weeks 15-17
6. **Deployment**: Week 18
7. **Post-Deployment Support**: Week 19 onwards

# Budget

## Cost Estimation

The cost estimation for the project is as follows:

1. **Project Management**: £15,000
2. **Requirement Gathering and Analysis**: £5,000
3. **Design**: £15,000
4. **Development**: £50,000
5. **Testing**: £10,000
6. **Deployment and Support**: £10,000

This brings the total estimated cost to £105,000.

## Budget Constraints

1. **Time Constraints**: Any changes in the requirements or scope of the project can lead to delays and increase the project timeline.
2. **Budget Constraints**: Unforeseen complexities during the development or testing phase may lead to additional costs.
3. **Resource Constraints**: Availability of resources (both human and technical) can impact the project timeline and costs.

# Risk Management

## Risk Identification

1. **Scope Creep**: Changes or uncontrolled growth in a project’s scope can lead to delays, cost overruns, and decreased quality.
2. **Technical Risks**: There may be technical issues or complexities that arise during the development phase which could impact the project timeline and budget.
3. **Resource Availability**: Unavailability of key resources, both human and technical, can pose a risk to the project.
4. **Security Risks**: As the project involves handling sensitive user data and transactions, there are inherent security risks involved.
5. **Vendor Dependability**: The project’s success depends on third-party vendors for payment gateway integration, which could pose a risk if the vendor fails to deliver as expected.
6. **Project Management Risks**: Inadequate project management can lead to poor communication, lack of coordination, and ultimately project failure.
7. **Legal and Compliance Risks**: Non-compliance with legal requirements or industry standards can lead to legal issues and penalties.
8. **Market Risks**: Changes in market conditions or customer preferences can impact the project’s success.

## Risk Analysis

1. **Scope Creep**: This risk can be mitigated by clearly defining the project scope and requirements at the outset and controlling changes to the scope throughout the project.
2. **Technical Risks**: Regular code reviews, using established development practices, and having a skilled development team can help mitigate this risk.
3. **Resource Availability**: This risk can be managed by proper resource planning and having backup resources available.
4. **Security Risks**: Implementing best security practices, regular security audits, and using secure coding practices can help mitigate this risk.
5. **Vendor Dependability**: This risk can be managed by choosing reliable vendors with a proven track record and having backup vendors in case of any issues.
6. **Project Management Risks**: This risk can be mitigated by employing experienced project managers, using proven project management methodologies, and maintaining clear communication among all stakeholders.
7. **Legal and Compliance Risks**: Regular audits, staying updated with the latest legal requirements and industry standards, and consulting with legal experts can help mitigate this risk.
8. **Market Risks**: Regular market research, staying updated with the latest trends, and being flexible to adapt to changing market conditions can help manage this risk.

There will need to be continuous monitoring of these risks throughout the project lifecycle and update the risk management plan as necessary. A contingency plan has been put in place for high-impact risks where necessary.

# Quality Assurance

## Quality Goals

1. **Robust Infrastructure**: Ensure the robustness of your web hosting infrastructure, VPS, and bare metal servers. The system should be resilient to high traffic, data loads, and potential security threats.
2. **Seamless User Experience**: Strive for a seamless user experience across all platforms. This includes fast load times, intuitive navigation, and responsive design.
3. **Data Integrity**: Safeguard the integrity of user data stored on your servers. This includes regular backups and robust recovery procedures.
4. **Scalability**: Your application should be scalable to handle growth in user base or data volume without compromising performance.

## Quality Control Measures

1. **Load Testing**: Regularly conduct load testing to simulate high traffic scenarios and understand how your application performs under stress.
2. **Security Audits**: Regular security audits are crucial to identify potential vulnerabilities in your application or infrastructure.
3. **Disaster Recovery Plan**: Have a disaster recovery plan in place to ensure minimal downtime and data loss in case of any unforeseen incidents.
4. **Monitoring Tools**: Implement monitoring tools to keep track of server health, application performance, and user activities in real-time.
5. **Automated Deployment**: Use automated deployment tools for efficient and error-free deployment of updates or new features.
6. **Server Optimization**: Regularly optimize your servers for better performance and efficient resource utilization.

# Resource Plans and Strategy

## Test Strategy

1. **Unit Testing**: Test individual components of the application in isolation. This includes testing individual functions, methods, and classes to ensure they behave as expected.
2. **Integration Testing**: Test the interaction between different components of the application. This includes testing APIs, data pipelines, and other points of integration.
3. **Functional Testing**: Test the functionality of the application and website. This includes testing user interfaces and workflows to ensure they work as expected.
4. **Performance Testing**: Test the performance of your application under various loads. This includes load testing (simulating multiple users), stress testing (testing the limits of your system), and endurance testing (testing the system over a prolonged period).
5. **Security Testing**: Conduct regular security audits to identify potential vulnerabilities in your application or infrastructure. This includes penetration testing (attempting to breach your own systems) and code reviews.
6. **Disaster Recovery Testing**: Regularly test your disaster recovery procedures to ensure they are effective and that you can recover data quickly in case of a failure.
7. **Usability Testing**: Test the usability of your application from an end-user perspective. This includes testing user interfaces for intuitiveness and ease of use.
8. **Compatibility Testing**: Test your application on various platforms (different operating systems, browsers, devices) to ensure it works consistently across all of them.

## Story Board

|  |  |
| --- | --- |
| Welcome Page | About Us |
|  |  |
| Contact Information Page / Admin Contact | Admin Home Page |
|  |  |

**Welcome Page**:

1. This will be the header of the page, displaying a company branding.
2. This will display a link to the login and register page.
3. This will display articles on the blog, but will not allow any permissions to the articles until logged in.

**About Us Page**:

1. See the Welcome Page references.
2. This will have all the links to specific product pages within the website – e.g. Homepage, Blog, Contact Information Page, Admin Home Page, Logout.
3. This will display the articles on the page but will allow permissions such as edit and delete article for admin –and allow the users to comment on the articles. Admins have all permissions to edit and delete any article or comment on the page.

**Admin Dashboard**:

1. This will be like the Welcome Page but will have an embedded link which will redirect to the blog page.
2. This will contain links to the main web pages, an admin link to edit or delete any comments and posts, along with user edit abilities and the logout function.
3. This will include options to suspend users, promote users, post articles, delete or alter articles, and access to the contact page.

**Contact Page**:

1. See Admin Dashboard.
2. See Blog Page.
3. This will contain text boxes for contact – one for email, one for the query, as well as submit button to forward the form.

# Site Map



