Project 3 – T1 Group submission  
  
  
Image source: <https://pakipackages.com/choosing-a-web-application-development-company/>

Members:  
  
Qhama Dyushu – 222 709 006

Zukisani Lale – 222 356 510

Phelo Madubela – 213 229 153

Oluhle Makhaye – 222 419 636

Zintle Mgqongose – 214 280 829

Aimee Paulus – 222 814 969

1. PROBLEM DEFINITION

Problem definition  
  
The beauty industry, particularly small and emerging salons, lacks a structured booking system for services such as hair and nail appointments. Many businesses rely on social media and messaging apps like WhatsApp to schedule appointments and process payments. This approach presents several challenges:

* Time Wasters and Fraudulent Transactions: Without a secure booking and payment verification system, businesses are vulnerable to no-shows and fake proof of payment scams.
* Operational Inefficiencies: Staff members must manually manage bookings, constantly monitor messages, and relay appointment details to the appropriate hairstylists, barbers, or nail technicians.
* Device Dependence: Salons must maintain an active smartphone to handle customer interactions, which increases the risk of missed messages and delays in communication.

**Project Goals and Objectives**

* Automate and Streamline Appointment Scheduling: Develop a user-friendly online booking system that allows clients to schedule appointments without relying on social media or direct messaging. Prevent scheduling conflicts and double bookings through real-time availability updates.
* Secure and Efficient Payment Processing: Integrate reliable payment gateways to facilitate secure transactions and eliminate fraudulent proof of payment issues. Support multiple payment methods, including credit/debit cards, mobile wallets, and EFTs, ensuring convenience for clients.
* Enhance Communication and Customer Engagement: Implement automated appointment confirmations, reminders, and rescheduling notifications via SMS or email. Reduce reliance on manual messaging, improving response times and overall customer satisfaction.
* Optimize Business Operations and Staff Management: Provide a structured system to manage staff schedules, ensuring hairstylists, barbers, and nail technicians are efficiently assigned to appointments. Centralize customer records, enabling seamless tracking of booking history, preferences, and loyalty rewards.
* Increase Customer Trust and Retention: Offer a professional and seamless booking experience, allowing clients to view service availability, select preferred stylists, and manage their appointments effortlessly. Introduce a review and rating system to enhance service quality and build customer confidence.

This system will revolutionize how small beauty businesses manage their operations, reducing administrative workload, minimizing financial risks, and enhancing the overall customer experience.

## Functional Requirements

**Planned Features for Users (Customers)**

## **User Registration and Profile Management**

* Users will be able to create and manage profiles, including personal details, beauty preferences, and service history.

1. **Services**

* Users will browse a list of beauty services with detailed descriptions, pricing, images, and customer reviews.

1. **Booking System**

* Users will schedule beauty appointments by selecting a service, date, and time.
* Home Service Option: Users will have the ability to request beauty specialists to visit their location.
* Emergency Beauty Bookings: The system will support last-minute appointments for urgent beauty needs.
* Group Booking Feature: Users will be able to book multiple services for a group in a single transaction (e.g., bridal makeup, group spa days).

1. **Payment Process**

* Secure payments with credit/debit cards, mobile wallets.
* Loyalty Program: Users will be able to earn points through referrals and frequent bookings, redeemable for discounts or free services.

1. **Notifications and Reminders**

* The system will automatically send booking confirmations and reminders via SMS, email, or push notifications.
* Users may receive personalized beauty tips and promotional offers.

1. **Feedback and Reviews**

* A rating and review system will allow customers to leave feedback on services received.

1. **Referral and Reward Program**

* A referral system will allow users to invite friends and earn discounts or beauty credits.
* Loyalty points will be rewarded for frequent bookings and referrals.

**Planned Features for Beauty Professionals**

1. **Profile Management**

* Beauty professionals will have the ability to create and edit profiles, showcasing their services, pricing, and availability.
* They may choose to offer in-salon, home services, or emergency bookings.

1. **Appointment Management**

* Beauty professionals will be able to manage their schedules, view bookings, and update availability.
* They may accept or decline emergency requests based on their schedule.

1. **Dashboard**

* A dashboard will display booking trends, customer feedback, and performance.

1. **Customer Service Tools**

* An in-app chat feature will allow pre-booking consultations with customers.
* The system will facilitate managing group reservations, including assigning multiple professionals for large bookings.

### **User Stories**

##### **As a Customer, I want to:**

##### Create an account to easily book beauty services.

##### Browse available services and select the best option.

##### Book home services for added convenience.

##### Schedule emergency beauty appointments for urgent needs.

##### Book group appointments for events like bridal parties or spa days.

##### Refer friends and earn rewards to save on future bookings.

##### Pay securely online using various payment options.

##### **As a Beauty Professional, I want to:**

##### Manage my profile to accurately showcase my services.

##### Offer home services and emergency bookings to attract more clients.

##### View and optimize my schedule for better time management.

##### Receive customer feedback to improve my services.

##### Accept and efficiently manage group bookings.

**Prioritization of Features**

**Must-Have Features:**

* User registration and profile management
* Service directory with service descriptions and reviews
* Booking system (in-salon, home services, emergency, and group bookings)
* Payment processing with secure and flexible options
* Notifications and reminders

**Should-Have Features:**

* Feedback and review system
* Beauty professional appointment management
* Dashboard for beauty professionals
* In-app chat for customer consultations

**Nice-to-Have Features:**

* Referral and rewards program (earn beauty credits)
* Loyalty program for frequent users

## End Users and Stakeholders

The primary end user would be clients and customers.

The systems internal users are the employees and the managers. The roles of the internal users are to manage appointments, set availability and provide services. It is the internal user's responsibility to keep schedules updated to prevent overlapping and miscommunication and to respond to bookings. The internal users need scheduling tools and client management

The external users are the customers and clients. The roles of the external users are to book appointments and purchase products. It is the internal user’s responsibility to provide accurate booking information and to make payments. They need easy booking interface, real-time availability and secure payment procedures.

The stakeholders are sponsors, app developers, project managers, customer support, beauticians, salons, clients, investors.  
  
**Stakeholders involvement within the project lifecycle:**

1. Initiation phase:

* Sponsors and investors will approve project charter and project idea, they will provide funding and define project objectives
* Project manager will develop business case and identify key stakeholders
* Industry experts will provide market analysis and competitive research

1. Planning phase:

* Project managers will create a project plan, timeline and budget to adhere to
* App developers will analyse required functions of the project
* UX designers will create wireframes
* Salon owners will define business rules and pricing structures

1. Execution phase:

* Development team build app features according to specifications
* QA Testers execute test cases

1. Monitoring control:

* Project manages tracks progress against milestones and manages risks
* Financial Team monitors budget expenses

1. Closure phase:

* Marketing analyses launch date
* Sponsors review final deliverables and the return on investment
* Customer support reviews possible customer queries

1. TECHNOLOGY AND SCOPE

Technology Selection:  
Overview of Chosen Technologies and Their Capabilities:  
For this project, we have chosen JavaScript for the backend. HTML, JavaScript and CSS for the frontend and MongoDB as the database solution. These technologies were chosen due to their flexibility, efficiency, and suitability for a web-based booking system with an integrated online-store.

**Backend: JavaScript (Framework - Node.js)**

*JavaScript* through *Node.js*, is an excellent backend choice because it is asynchronous, event-driven, and non-blocking, making it ideal for handling multiple booking requests simultaneously. *Express.js*, a lightweight web framework for *Node.js*, can further simplify API development and routing. It has high performance due to asynchronous execution, large ecosystem with many open-source libraries (for example, authentication, payment processing), full-stack *JavaScript* ensures consistency across frontend and backend development. It is suitable and ideal for handling real-time appointment scheduling and notifications, easy integration with *MongoDB* using *Mongoose*, a powerful Object Data Modelling library.

**Frontend: HTML, CSS, JavaScript**

The frontend will be built using standard *HTML*, *CSS*, and *JavaScript*, ensuring a responsive and interactive user experience. React.js or Vue.js could be integrated for a more dynamic UI if needed. JavaScript allows for dynamic interactions and real-time updates, *CSS* frameworks like *Tailwind* *CSS* or *Bootstrap* enhance design efficiency, works seamlessly with the backend since they both use *JavaScript*. It is suitable because it provides an intuitive and visually appealing interface for customers, supports interactive features like appointment selection, payment processing, and notifications.

**Database: MongoDB**

We chose *MongoDB*, a *NoSQL* database, because of its scalability, flexibility, and ease of integration with JavaScript-based backend frameworks. Unlike relational databases, *MongoDB* stores data in JSON-like documents, making it a natural fit for handling appointment details, user profiles, and transactions. It is a schema-less structure allows for flexible data storage, easily scales for future growth as the business expands, integrates seamlessly with *Node.js* through the *Mongoose* Object Data Modelling. It is suitable andideal for handling dynamic, unstructured data such as customer bookings and real-time availability updates, facilitates fast read/write operations, making it efficient for frequent booking transactions.

**Feasibility and Scalability of Chosen Technologies**

Availability of Resources: *JavaScript (Node.js*) and *MongoDB* are widely adopted in the developer community, ensuring strong documentation, tutorials, and third-party support. Numerous open-source libraries reduce development time (e.g., Passport.js for authentication, Stripe for payments). Cloud hosting services like *MongoDB* *Atlas* and *AWS* ensure easy deployment and scalability.

Scalability for Future Growth: Node.js and MongoDB support horizontal scaling, allowing the system to handle more users as the business grows. Microservices architecture can be implemented in the future for better modularity and performance. Potential for integrating mobile apps using React Native for cross-platform booking accessibility.

**Considered Alternatives and Why They Were Not Chosen**

Java is a powerful, scalable and secure language but requires a complect setup and is a steep learning curve. Python has simple syntax, it's good for quick development but it is slower for high performance and real time applications. PHP is good for Web applications and easy hosting, but it is less suited for modern and interactive User interfaces. CassandraDB has high availability and is a NoSQL database, but its overhead is too high for a small-scale application.

Final Decision: JavaScript (Node.js) and MongoDB were chosen because it balances performance, ease of use, scalability, and ecosystem support, making it ideal for the web-based beauty appointment system. This selection ensures a cost-effective, efficient, and scalable platform that meets the needs of small beauty businesses while allowing room for future growth.

## Scope of the project:

The principal aim of this project is to offer a core centre (a "Hub") through which different beauty businesses can handle bookings and sales of their products. The system will allow such businesses to handle bookings of their customers, order for beauty products, and receive payment in an easy-to-use, convenient environment.

**What's Included:**

Business Hub: The feature to enable businesses to list and manage their services (e.g., haircuts, nails, makeup) and products on a single platform.

Service Booking System: The customers can choose the business, book appointments for various beauty services (haircuts, nails, makeup), and see available time slots.

Product Ordering: Customers can view beauty products from various businesses and order. The system will generate invoices automatically and provide secure payments for bookings and product purchases.

Multi-business Support: The system will support multiple businesses operating under one platform. Chatbot for Inquiries: A basic customer support chatbot for queries and quick responses.

**Excluded:**  
Mobile App: Web-based only (no app).  
Advanced AI Chatbot: FAQ bot only, no AI-powered.  
Complex E-commerce: Beauty products only, not multicity e-commerce stores.  
Analytics: Simple features only, no advanced reporting to begin with.  
  
**Why These Exclusions:**  
  
A mobile app adds complexity and expense.  
Higher time and effort for advanced AI chatbots.  
E-commerce of anything other than beauty products is more than simplicity and scope.  
Analytics can be added later when the platform is stabilized.  
E-commerce for products other than beauty products is beyond scope due to simplicity and focus.  
Analytics can be added later when the platform grows.

## Technology implementation plan:

This plan lays out the processes required to create a beauty services platform with Vue.js, Node.js, and MongoDB. The project will be divided into different stages to ensure well-organized development and implementation.

## **Step 1: Setting Up Development Environment**

## **Frontend Setup**

* Install Node.js and npm
* Initialize Vue Project
* Install Development Tools (Visual Studio Code, Install Vue Devtools for effective debugging)
* **Backend Setup**
* Set Up Node.js Server for handling requests.
* Organize server structure for routes and controllers.
* Use MongoDB as your NoSQL database.

**Step 2: Building the Frontend**

* **Vue.js Structure**
* Use reusable components to break down the application (for example, ServiceList, BookingForm, and UserProfile).
* **Routing Setup**
* Use Vue Router to navigate between components.
* Create routes for home, services, bookings, and user profiles.
* **State Management**
* Use Vuex to manage application state, especially user data and booking information.
* **Styling**
* Use Bootstrap or Tailwind CSS for responsive design.
* Use scoped styles in Vue components to keep styles modular.
* **JavaScript Functionality**
* To communicate with the backend API, send HTTP requests over the application.
* Configure Vue Router for client-side navigation.

**Step 3: Integrating Third-Party Libraries or APIs**

**Define API Endpoints**

* Create RESTful API endpoints for:
* User registration and authentication.
* Services (CRUD operations).
* Booking management.

**Database Models**

* Define Mongoose models for:
* Users
* Services
* Booking
* Reviews

**Testing the Backend**

* To test API endpoints, use tools such as Postman or Insomnia.
* Create unit tests using Mocha or Jest.

**Step 4: Configuring Deployment**

* **Prepare for Launch**
* Optimize images and files to improve loading times.
* Test the application on various devices to ensure compatibility and functionality.
* Use technologies such as Webpack to simplify JavaScript and CSS files.
* Ensure that all Mongoose models are defined and tested.
* Configure and execute database migrations as needed.
* **Deploy the Application**
* Choose a suitable hosting platform and follow the deployment procedures to make the application live.
* Frontend: Deploy with Netlify or Vercel.
* Backend: Deploy with Heroku, AWS, or DigitalOcean.
* Database: For database hosting, use MongoDB Atlas.
* Ensure that the environment settings and database connection settings are correct.
* **Set Up Automation**
* Use tools like GitHub Actions to automate testing and streamline code releases, ensuring a smooth deployment process.

**Potential challenges or risks associated with technology implementation**

* **Browser Compatibility**
* **Challenge:** Ensuring the application works consistently across different browsers can be difficult.
* **Mitigation:** Conduct thorough testing on major browsers (Chrome, Firefox, Safari) during development to identify and fix issues early.
* **Performance Issues**
* **Challenge:** Slow loading times can lead to a poor user experience.
* **Mitigation:** Optimize images and files, implement lazy loading, and utilize performance monitoring tools to identify bottlenecks.
* **Security Risks**
* **Challenge:** Protecting sensitive user data is critical, especially with personal information.
* **Mitigation:** Enforce HTTPS, validate user input, and regularly update dependencies to address vulnerabilities.
* **Team Skill Gaps**
* **Challenge:** Some team members may lack familiarity with specific technologies or tools.
* **Mitigation:** Provide training sessions, recommend online courses, and encourage knowledge sharing among team members.

**Timeline for Technology-Related Tasks and Milestones**

**Step 1: Setting Up Development Environment**

* **Frontend Setup:** Week 1
* Install Node.js and npm
* Initialize the project
* Install development tools
* **Backend Setup:** Week 1
* Set up Node.js server
* Organize project structure
* **Milestone:** Development environment ready by the end of Week 1

**Step 2: Building the Frontend**

* **HTML Structure:** Week 2
* Create basic structure for index.html
* **Styling:** Week 2 to Week 3
* Implement responsive design using CSS frameworks like Bootstrap or Tailwind CSS.
* **JavaScript Functionality:** Week 3 to Week 5
* Implement main features
* **Milestone:** HTML structure completed by Week 2, responsive design by Week 3, main features by Week 5

**Step 3: Backend Development**

**Define API Endpoints: Week 6**

* Create RESTful API endpoints for:
* User management (registration, login).
* Services (CRUD operations)
* Booking management.

**Database Models: Week 6**

* Define Mongoose models for Users, Services, and Bookings.
* Implement validation and middleware to ensure data integrity.

**Testing the Backend: Week 7**

* Test API endpoints with Postman or other comparable tools.
* Write unit tests with Mocha or Jest.
* **Milestone:** Backend API functional and tested by the end of Week 7.

**Step 4: Configuring Deployment**

* **Prepare for Launch:** Week 8 to Week 9
* Optimize files and test for compatibility
* **Deploy the Application:** Week 10
* Follow deployment procedures on chosen platform
* Use MongoDB Atlas for database hosting.
* **Set Up Automation:** Week 9 to Week 10
* Implement CI/CD with GitHub Actions
* **Milestone:** Application optimized by Week 9, live deployment by Week 10

**Ongoing Monitoring:**

* Start performance monitoring in Week 11 and continue as needed.

1. COMMUNICATION

Our team has decided to use WhatsApp as a means for communication due to ease of access. As lectures were online, the team was not able to have in person meetings due to transport issues. Despite these challenges, members attended group meetings via WhatsApp calls and if a member was unable to attend, they would let the group know in advance and be updated in the minutes of the meeting.

Communication was effective through online meetings with group members as well as online meetings with our supervisor. All decisions were concluded after thorough discussion and analysis. When addressing project ideas, group members shared multiple ideas and after evaluation and advice from our supervisor, we decided to create a beauty appointment booking system. It was challenging to pick a group leader but after further discussion, we voted on Qhama being our group leader via a poll on WhatsApp.

Depending on what needed to be done, group members would communicate their progress weekly and we would have meetings once a week or once every two weeks.  
To ensure we stayed on track in our meetings, an agenda was set up prior to the actual meeting.

We had a few issues with our online meetings as we had to work around each other's schedules as well as loadshedding and connectivity issues. We would have preferred face to face meetings but due to circumstances, we settled for online meetings. Going forward, we will be having in person meetings. Regarding task allocation, tasks were split up to ensure that everyone does an equal amount of work.

We implemented flexible scheduling by asking team members to indicate their availability, which allowed us to arrange meeting times that worked for everyone. This increased attendance and engagement during conversations. We made effective documentation by sending meeting minutes via WhatsApp to all members, even those who couldn't attend. We also created a clear agenda for each meeting, which helped us remain on track and allowed for more productive discussions. We have also included in-person meetings, which has built a deeper sense of teamwork and collaboration. By encouraging open communication and creating a friendly environment, we made it possible for team members to share ideas and concerns, thereby improving our group's overall effectiveness and collaboration.

1. TASK ALLOCATION

**Zintle Mgqongose**(Team member)- Front-End Development:

CSS Styling: Design the layout, colour scheme, fonts, and overall feel to make the site look pretty and responsive on different devices.

JavaScript Interactivity: Add interactive elements such as booking forms, product sliders, and live updating of content.

Frameworks:React.js and vue.js

**Strengths:**  
Good at HTML, CSS, and JavaScript. Good at crafting responsive and user-friendly layouts. Experienced with React.js.  
  
**Weaknesses:**  
Needs help in backend integration and highly complex JavaScript frameworks other than React. HTML Structure: Develop the overall composition of all web pages (Home, Booking, Product, etc.).

**Aimee Paulus**(Scribe) - Front-End Development

UI/UX Design: Prioritize user experience and make page navigation simpler.

Form Validation: Validate all forms (Booking, Contact, etc.) on the client-side correctly before submission.

Responsive Design: Make the site responsive, testing across various screen sizes.

JavaScript Features: Add more front-end features such as modals, pop-ups, and form animations.

Frameworks: React.js and vue.js

**Strengths:**  
Good UI/UX designing skills. Experienced in form validation, responsive layout, and React.js.  
  
**Weaknesses:**  
Limited backend experience and extensive knowledge of server-side logic.

**Qhama Dyushu** (Team Leader) – Team leader

Project Management: Manage project milestones, ensure timely completion, and manage client/stakeholder communication.

Technical Leadership: Lead back-end architecture and make high-level technology decisions.

Team Coordination: Coordinate efficient working between team members.

**Strengths:**  
Strong communication, project management, and leadership skills. Strong decision-making and progress-tracking capabilities.  
  
**Weaknesses:**  
Predisposed towards coordination as opposed to coding in general, especially on the frontend.

**Oluhle Sbusiso Makhaye** (Team Member):Back-End Developer

Core Logic: Develop booking, payment, and order processing logic.

Real-Time Updates: Coordinate real-time updates for orders and bookings.

Framework: Node.js,

**Strengths:**Have experience in backend logic and real-time updates. Familiar with Node.js and Express.js.  
  
**Weaknesses:**  
Needs assistance in frontend integration and UI/UX design.

**Phelo Sifiso Mninawe Madubela**(Team Member) : Back-End Developer

System Integrations: Integrates third-party services (payment gateways, email).

Booking Management: Manages booking creation, update, and cancel.

Error Handling: Provides error tracking and data validation.

Security: Manages authentication and session management.

Framework: Node.js,

**Strengths**: Experienced in error handling, security, and session management. **Weaknesses**: Limited front-end development and integration experience.

**Zukisani Lale** (Team Member): Database Administrator

Database Design: Designs users, bookings, orders, and products schema.

Data Integrity: Stores validated, accurate data.

Optimization: Optimizes queries for performance and scalability.

Security: Provides access control and encryption of sensitive data.

**Strengths**: Mastery of database design, optimization, and data integrity maintenance using MongoDB.

**Weaknesses**: Limited front-end experience, will require guidance on full-stack integration. Strengths: Expert in database design, optimization, and ensuring data integrity with MongoDB.

1. Group organization

**Group organisation**

We organized our team as I elaborate below, and it worked in our favour. We can note the success as we were able to effectively accomplish the beauty appointment booking system project:

**Team Leadership and Decision-Making:**

* We chose a group leader by voting by making a poll on WhatsApp with our names. After discussing well, I had more votes, and we all agreed I Qhama Dyushu will be the team leader.
* The responsibilities of the leader included overseeing project progress, organizing meetings, and keeping deadlines.

**Task Distribution and Role Assignment**

* Responsibilities were allocated based on individuals interests to maximize effectiveness.
* Each member had something specific to work on within the project, so the work would be divided equally.

**Communication Tools and Techniques:**

* WhatsApp was selected as the main form of communication based on accessibility and convenience.
* There were regular updates and conversations through WhatsApp messages, voice messages, and WhatsApp calls.

**Meeting Structure and Scheduling:**

* Because of issues like load-shedding and transport, most of the meetings were held online.
* There was a proper agenda for every meeting to guarantee effective discussion and decision-making.
* Face-to-face communication was later added to encourage greater cooperation.

**Collaboration and Teamwork:**

* The group provided a perfect work atmosphere where everyone felt free to share problems and ideas.
* Members who missed meetings were informed through meeting minutes circulated on WhatsApp.

**Problem-Solving:**

* All disagreements were settled by open discussion and voting where required. However, so far, we haven't faced any challenges and are still working well together.
* Monitoring Progress and Quality Assurance Ongoing progress was given within the team to keep track of tasks completed and pending. The project's final document was checked for accuracy and completeness prior to submission.