A Project Report
On

Operating a Theme Park

Submitted by

Harsh Ajay: 210303108048

Laxit Khanapra: 210303108039

Dhaval S Bavda: 210303108262

Ruttvik Padhara: 210303108041

Aagam Shah: 210303108217

Under the Guidance of

Narasimha N

Introduction

Imagine a world of thrilling rides, captivating shows, and unforgettable experiences. This is the essence of a theme park, a place where families and friends come together to create lasting memories. But in today's digital age, the magic extends beyond the park gates. A user-friendly and engaging website acts as the park's digital ambassador, captivating visitors, streamlining their experience, and ultimately contributing to the park's success.

This project embarks on the exciting journey of crafting a theme park website built on the MERN stack (MongoDB, Express.js, React.js, Node.js).

1. Stakeholders and Roles

1.1 Customer

- To follow park rules and regulations
- To respect the park's property and staff
- To report any safety concerns or incidents to park staff
- To provide feedback and suggestions to improve the park experience
- To be aware of and comply with any health and safety guidelines

1.2 Staff

- Frontline staff : Interact directly with guests, handle inquiries, provide assistance, and ensure a smooth guest experience.
- ride operators: Operate rides, ensure guest safety, and manage ride queues.
- ticketing agents: Sell tickets, process payments, and manage ticketing systems.

1.3 Software Developers

- Design and develop the system to meet specified requirements and ensure it integrates well with existing processes.

1.4 UI/UX Designers

- Focus on making the system intuitive and easy to use for all stakeholders.

2. Relationship between Case Study and Stakeholders

2.1 Business Goals

- Increase overall theme park attendance through engaging online promotions and informative content about the park's offerings.
- Generate additional revenue through online ticket sales, merchandise purchases, and potential sponsored content or advertising partnerships.
- Improve brand awareness and create a positive online presence for the theme park.

2.2 Target Users

- **Families with young children:** This group seeks a fun and engaging experience for their children. The website should provide information on kid-friendly rides, shows, and character meet-and-greets.
- **Thrill-seekers:** This group prioritizes adrenaline-pumping rides and attractions. The website should highlight these features with detailed descriptions and wait time information.
- **Adults seeking a day out:** This group may be interested in shows, dining options, special events, and relaxing activities. The website should showcase these offers and provide booking options for reservations.

2.3 Features

- Reserved parking slots
- Booking tickets (It will include food, combos, parking)
- Rides / Attractive Review &Feedback
- Navigation /Parking Map
- Tracking the crowed in the rides
- Analyzing the rides preference
- Payment gateway
- Contact
- Safety & security measures

3. Challenges and Solutions

3.1 Challenges

3.1.1 Coding challenges

- Creating a responsive design that adapts to various screen sizes and resolutions.
- Developing a website that can handle high traffic and large volumes of user data.

3.1.2 Teamwork challenges

- Ensuring effective communication and collaboration among team members.

3.1.3 Security

- Safeguarding user information (payment details, personal details) and website functionality from cyberattacks is crucial.

3.1.4 Integration with Existing Systems

- The theme park website may need to integrate with existing ticketing systems, reservation platforms, or loyalty programs.

3.2 Need & Alternatives

3.2.1Need

- Streamline the visitor experience with online ticket purchase, combos and packages and reservation options.

3.2.2 Alternatives

- Basic Website: A low-cost, static website with basic information might suffice for smaller parks, but lacks the interactive features and potential revenue generation benefits of a MERN stack website.

3.3 Standing Out from Competitors

- User Experience (UX) Focus
- Prioritize intuitive navigation
- Clear information architecture
- Engaging content to keep visitors coming back.

3.4 Risk Management & Solutions

3.4.1 Development risk

- Project goal is not defined properly

- Selecting efficient host provide
- Choosing right technologies
- Regular meeting and updates

3.4.2 Security Risks

- Choose a reputable hosting provider with robust security measures.
- Implement best practices for data protection and user authentication.

3.4.3 Content risk

- Establish clear content guidelines and standards to ensure consistency and quality.
- Ensure that all website content is accurate, up-to-date, and relevant to the target audience.

3.4.4 Negative user experience

- Provide clear and concise user instructions and feedback to minimize confusion and frustration.

3.4.5 Accessibility issue with website

- Implement accessibility best practices, such as providing alternative text for images, using descriptive headings, and ensuring sufficient color contrast

3.6 Avoiding Failure

- Proper planning.
- Proper implementation.
- Select the right SDLC method.
- Choose updated resources so it will be Compatible with every system.

4. Areas for Investigation, Solution Design, and Resources

4.1 Investigation

- **Target Audience:** Identify the park's primary visitor demographics (families, thrill-seekers, etc.) and their online behavior patterns.
- **Park Offerings:** Understand the park's attractions, shows, dining options, and unique selling points.

- **User Needs:** Conduct user research (surveys, user testing) to understand visitor pain points and information needs.

4.2 Solution Design

- **Information Architecture:** Develop a clear and user-friendly website structure for easy navigation and information access.
- **User Interface (UI) & User Experience (UX) Design:** Design an engaging and intuitive interface that prioritizes a seamless user experience across devices.

4.3 Resources

- **Development Team:** Skilled developers with expertise in MERN stack (MongoDB, Express.js, React.js, Node.js).
- **UI/UX Designers:** To create a visually appealing and user-friendly website interface.

4.4 Technologies for Theme Park Website

4.4.1 MERN Stack (Our Chosen Technology)

- Pros: MERN stack is a complete development environment that offers flexibility, scalability, and a large community of developers who can help.
- Cons: It requires knowledge of multiple technologies, and new developers might need time to learn it.

4.4.2 LAMP Stack (Linux, Apache, MySQL, PHP)

- Pros: LAMP stack is a well-established way to build websites, and many developers are familiar with PHP.
- Cons: It might not be as scalable as Node.js, and PHP can have security issues if not used carefully.

4.4.3 Python Django

- Pros: Django is a popular web framework for building dynamic websites and web applications with Python.
- Cons: Django might not be able to handle high-traffic websites as efficiently as other technologies, which could be a concern for a theme park website that expects a large number of visitors.

5. Breakdown Structure and Potential Areas for Investigation

5.2 MERN Stack Architecture Breakdown:

5.2.1. MongoDB

Acts as the website's database, storing information like,

- Ride details (descriptions, height requirements, wait times)
- Show schedules and descriptions
- Restaurant menus and locations
- User information (if they create an account for purchases or reservations)

5.2.2. Express.js

Acts as the backend framework, handling tasks like,

- Processing user requests (e.g., ticket purchases, reservation bookings)
- Communicating with the MongoDB database to retrieve and store data
- Delivering data to the frontend (React) for display

5.2.3. React.js

Creates the website's user interface (what visitors see and interact with).

- Interactive elements like maps, show schedules, and wait time displays
- User-friendly forms for ticket purchases and reservations
- Personalized recommendations for rides, shows, and restaurants (based on user preferences)

5.2.4. Node.js

- Acts as the runtime environment that allows JavaScript code to run on the server (backend).
- It essentially powers both Express.js (backend framework) and potentially some functionalities within React (frontend).

6. Clear Goals and Success Criteria

6.1 Goals

- Launch the website within 8 weeks.
- Build a user-friendly website.
- Integrate all necessary features.

- Ensure high performance and reliability.

6.2 Success Criteria

- User satisfaction surveys and feedback.
- High user engagement and retention
- Smooth operational integration

7.Realisation of Benefits and Business Case

7.1 Business Case

- The theme park website will increase online ticket sales, improve user experience, and enhance theme park visibility.

7.2 Measurements and Metrics:

- Online ticket sales revenue.
- Website traffic and engagement metrics.
- User satisfaction ratings.

7.3 Vital Business Functions

- Online ticket purchasing.
- Ride information and scheduling.
- Special offers and promotions.

8. Objectives and Outcomes

8.1 Objectives

- Deliver a high-quality website
- Meet all functional requirements

8.2 Outcomes

- Increased visitor satisfaction
- Improved operational efficiency
- Enhanced Park management capabilities

9. Learning Statement

9.1 Learning Outcomes

- Gain experience with MERN stack development.
- Learned Api integration, payment gateway, Ui/UX Design, Documentation, management of users.
- Enhance communication skills.

9.2 Current Capabilities

- Proficient in Documentation.
- Basic with Node.js and Express framework.
- Moderate knowledge of MongoDB and front-end development.

9.3 Importance and Relevance

- Improve ability to work with diverse technologies and teams.
- Contribute to the development of a real-world project.
- It will be used to get a good package's job.
- Make an impact on resume.