

MULTICULTURALPLANET.COM

SOFTWARE REQUIREMENT SPECIFICATION

Date: 11/27/2024

Version: 1

Team Members:

Brooke Law (*bel58*)

Mounica Paladugu (*mp3626*)

Emma Romero (*elr69*)

Praket Shah (*ps3226*)

Rachel Typrin (*ret47*)

Stakeholder:

Juliano Braga Telles

Professor/Advisor: Filippos Vokolos - Senior Design 1

Table of Contents

1. Introduction
 - 1.1. Purpose
 - 1.2. Intended Audience
 - 1.3. Overview
2. Overall Description
 - 2.1. Product Perspective
 - 2.2. Product Features
 - 2.3. User Characteristics
 - 2.4. Constraints
 - 2.5. Assumptions and Dependencies
3. Specific Requirements
 - 3.1. Functional Requirements
 - 3.1.1. User management interface
 - 3.1.2. Content moderation interface
 - 3.1.3. Analytics interface
 - 3.1.4. Free content interface
 - 3.1.5. Premium/paid content interface
 - 3.1.6. Mentorship program interface
 - 3.1.7. Ad management interface
 - 3.1.8. Account login interface
 - 3.1.9. User registration interface
 - 3.1.10. Search discovery features
 - 3.1.11. Community engagement tools
 - 3.1.12. Partner integration tools
 - 3.1.13. Event creation/management tools
 - 3.1.14. Article submission tools
 - 3.1.15. Multimedia and collaboration tools
 - 3.1.16. Event hosting tools
 - 3.2. Non-functional Requirements
 - 3.2.1.
 - 3.3. External Interface Requirements
 - 3.3.1. User Interfaces
 - 3.3.2. Hardware Interfaces
 - 3.3.3. Software Interfaces

1. Introduction

1.1 Purpose

This document details the software requirements for MulticulturalPlanet.com, aimed at fostering cultural exchange, professional development, and community collaboration through a versatile online platform.

1.2 Intended Audience

The primary users of this document are the project development team, stakeholders (including Juliano Braga Telles), academic advisors, and potential platform partners.

1.3 Overview

The platform will offer diverse content, community engagement tools, e-commerce features, and educational resources, catering to cultural enthusiasts, professionals, and general users. It aligns with the project milestones and sprints outlined in the Project Plan, emphasizing phased development, testing, and user engagement strategies.

2. Overall Description

2.1 Product Perspective

Multicultural Planet is an integrated platform that merges elements of social networking, digital marketing, e-commerce, and cultural resource sharing. The platform is designed to offer long-term growth, supported by various revenue streams, advanced marketing tools, and secure payment integrations.

2.2 Product Features

Key features include:

- **Event Listings:** High-priority feature for promoting cultural events.
- **News and Content Posting:** High-priority content for engaging users.
- **Business Directory:** High-priority feature connecting users with cultural professionals.
- **Social Media Communities:** Medium-priority integration for enhanced engagement.
- **Skill Sharing and Professional Networking:** Medium-priority feature for connecting users.

2.3 User Characteristics

Users include:

- **Regular Users:** Access free content and participate in forums.
- **Premium Users:** Paid members with access to exclusive content and features.
- **Content Creators:** Professionals producing cultural content.
- **Event Organizers:** Users managing cultural events.
- **Administrators:** Responsible for platform management and security.

2.4 Constraints

- Budget and resource limitations.
- Compliance with data protection regulations (GDPR).
- Project timeline constraints for each development phase.

2.5 Assumptions and Dependencies

- Availability of internet and hosting resources.
 - Timely stakeholder feedback.
 - Collaboration among team members using project management tools.
-

3. Specific Requirements

3.1 Functional Requirements

1. User Management Interface

- **Implementation Approach:**

- Users shall be able to create accounts, log in, and manage their profiles.
- The system shall support role-based access control (Regular User, Premium User, etc.).
- CRUD operations (Create, Read, Update, Delete) shall be available for profile management.
- User roles shall determine access to specific pages and features using conditional rendering.

2. Content Moderation Interface

- **Implementation Approach:**

- Admins shall be able to view a dashboard displaying user-generated content (e.g., articles, comments).
- Admins shall have options to approve or reject content using the interface buttons.
- Users shall be able to flag inappropriate content, which shall then appear in the admin dashboard for review.

3. Analytics Interface

- **Implementation Approach:**

- Admins shall be able to view user activity metrics, such as views, comments, and likes.

- Summary statistics, including total views and engagement rates, shall be displayed.
- Data visualization charts shall present engagement trends over time.
- 4. **Free Content Interface**
 - **Implementation Approach:**
 - Users shall be able to browse free articles and event listings in a blog-like layout.
 - Admins shall have the ability to create, update, and manage free content via a CMS or CRUD operations.
 - Users shall be able to search and filter content by categories such as “language learning” or “cultural news.”
- 5. **Premium/Paid Content Interface**
 - **Implementation Approach:**
 - Access to premium content shall be restricted to users with a “Premium” role.
 - Non-premium users shall see “Upgrade to Premium” messages when accessing restricted content.
 - The system shall support secure payment processing for premium subscriptions.
- 6. **Mentorship Program Interface**
 - **Implementation Approach:**
 - Mentors and mentees shall be able to create profiles, specifying interests and preferences.
 - The system shall suggest potential mentor-mentee matches based on tags or manual selection.
 - Mentors shall be able to schedule sessions using a calendar interface.
- 7. **Ad Management Interface**
 - **Implementation Approach:**
 - Advertisers shall be able to upload and manage ads using a simple form with fields for title, description, and image link
 - Ad metadata (title, description, image link) shall be stored in a centralized database
 - Ad performance metrics shall be trackable by integrating third-party ad services, such as Google Ads
- 8. **Account Login Interface**
 - **Implementation Approach:**
 - Users shall be able to log in using their registered email and password
 - Users shall be able to see an error message indicating invalid credentials after failed login attempt(s)
 - Users shall be able to reset their password with an email link if they forget their credentials
- 9. **User Registration Interface**
 - **Implementation Approach:**
 - Users shall be able to register by providing an email, password, and selecting a user role (Regular or Premium) in the respective fields
 - Users shall see a message after email validation indicating whether the email is valid or invalid based on format and uniqueness

- Users shall see a message after entering a password indicating whether it meets complexity requirements (minimum length and character variety)
- Users shall be able to press a "Create Account" button only if all fields are completed
- Users shall be able to confirm account creation through a verification email sent to the registered email

10. Search and Discovery Features

- **Implementation Approach:**
 - Users shall be able to search for content, events, or users by using a search bar
 - Users shall be able to filter search results with options including content type (articles, events, multimedia) and categories
 - Users shall be able to sort search results by relevance, date, or popularity
 - Users shall see a message for no results indicating no matches were found, along with suggested terms or recent popular searches

11. Community Engagement Tools

- **Implementation Approach:**
 - Users shall be able to create, read, update/edit, and delete (CRUD model) topics and replies in discussion forums
 - Users shall be able to upvote/like topics and replies, with the ability to reverse their upvote/like
 - Users shall be able to report posts and replies for moderation using a "Report" button
 - Moderators shall be able to lock or delete reported topics and replies

12. Partner Integration Tools

- **Implementation Approach:**
 - Partner users shall be able to create, post, and delete ads
 - Partner users shall be able to upload content for advertising
 - Partner users shall be able to create and submit advertisements to database for admin approval
 - Partner users shall be able to view analytics for tracking engagement including views, interactions, etc.

13. Event Creation/Management Tools

- **Implementation Approach:**
 - Users shall be able to input an event *NAME*, *DATE*, *TIME*, *LOCATION*, *DESCRIPTION*, and *PREMIUM/NOT PREMIUM* when creating an event
 - Dates shall be displayed in the mm/dd/yyyy format
 - Times shall be displayed in the HH:MM TIME ZONE format
 - Description shall be less than 1000 words
 - Events shall be marked as either *PREMIUM* for premium members, and *NOT PREMIUM* for all members.
 - Users shall be able to edit all fields in an event
 - Users shall be able to delete existing events
 - Users shall be able to save changes to events after modifying

14. Article Submission Tools

- **Implementation Approach:**
 - Users shall be able to upload text files (.pdf, .docx, .doc, .txt.) at a maximum size of 10M
 - Users shall be able to submit added text files to a content database for admin review and approval

15. Multimedia Tools

- **Implementation Approach:**
 - Users shall be able to upload files with extensions .jpeg/.jpg, .png, .mp4, .mov, .pdf, .gif, .wav, .mp3
 - Users shall be able to upload folders/portfolios containing the listed file types

16. Event Hosting Tools

- **Implementation Approach:**
 - Users shall be able to add other users as hosts for existing events
 - Users shall be able to add up to 4 users as hosts, 5 total including the event creator who is always a host
 - Users shall be able to remove users as host
 - Users shall be able to save changes to event hosts

3.2 Non-Functional Requirements

- **Performance:** Implement caching for frequently accessed data and optimize database queries to ensure scalability.
 - **Usability:** Prioritize clean, responsive layouts using frameworks like Bootstrap or Material-UI.
 - **Security:** Use HTTPS for secure transactions and implement input sanitization to prevent common vulnerabilities (SQL injection).
 - **Reliability:** Schedule regular database backups and implement error-handling routines.
 - **SEO Optimization:** Include metadata tags and content optimization strategies.
-

Performance

1. **Caching Efficiency:**
 - **Cache Hit Ratio:** The percentage of cache hits vs. total cache accesses.
 - **Target:** Aim for a cache hit ratio of 80-90%.
 - **Test:** Measure the number of cache hits versus cache misses over time.
2. **Database Query Efficiency:**
 - **Query Response Time:** Average time taken by the database to execute queries.
 - **Target:** Keep the average query response time under 100ms for critical operations.

- **Test:** Use a database query performance tool (e.g., New Relic, Query Profiler) to track execution times.
- **Database Load:** Track the number of queries executed per minute/hour.
 - **Target:** Ensure no significant spikes beyond the system's capacity.
 - **Test:** Monitor with load testing tools (e.g., Apache JMeter, LoadRunner) under peak usage scenarios.

Usability

1. **Page Load Time:**
 - **Time to Interactive (TTI):** Time it takes for a page to become fully interactive.
 - **Target:** Less than 3 seconds for first load.
 - **Test:** Measure using web performance tools like Google Lighthouse or WebPageTest.
2. **Responsive Design:**
 - **Mobile Usability Score:** A score that indicates how well the website adapts to mobile devices (e.g., Google Mobile-Friendly Test).
 - **Target:** Achieve a score of 90+ out of 100.
 - **Test:** Use Google's Mobile-Friendly Test or Lighthouse for mobile responsiveness evaluation.
3. **User Interaction:**
 - **Click-to-Action Time:** Time it takes for users to complete a specific action (e.g., sign-up, purchase).
 - **Target:** Less than 2 minutes for key user actions.
 - **Test:** Analyze user flow with tools like Hotjar or Google Analytics.

Security

1. **SSL/TLS Coverage:**
 - **HTTPS Implementation:** Percentage of site traffic served over HTTPS.
 - **Target:** 100% of the traffic should be over HTTPS.
 - **Test:** Use SSL Labs' SSL Test to verify secure connection for the entire site.
2. **Vulnerability Scanning:**
 - **Security Flaws Identified:** Number of critical vulnerabilities detected via security scans.
 - **Target:** Zero critical vulnerabilities at any given time.
 - **Test:** Run periodic vulnerability scans using tools like OWASP ZAP, Nessus, or Burp Suite.
3. **Input Validation:**
 - **SQL Injection Detection:** Number of potential SQL injection attempts blocked.
 - **Target:** Zero successful SQL injections.
 - **Test:** Perform penetration testing or automated vulnerability scanning with tools like SQLMap.

Reliability

1. **Uptime:**
 - **Server Uptime:** The percentage of time the server is operational without any downtime.
 - **Target:** 99.9% uptime.
 - **Test:** Monitor uptime with tools like UptimeRobot or Pingdom.
2. **Backup Frequency and Success Rate:**
 - **Backup Frequency:** Percentage of backups completed as scheduled.
 - **Target:** 100% of backups should be completed on schedule.
 - **Test:** Regularly audit backup logs and verify the integrity of backups.
 - **Restore Time:** Time required to restore from the latest backup.
 - **Target:** Restore time should be under 30 minutes.
 - **Test:** Perform disaster recovery drills to test the restore process.
3. **Error Rate:**
 - **Error Rate:** Percentage of failed requests or errors encountered by users.
 - **Target:** Error rate should be less than 0.5% of total requests.
 - **Test:** Monitor errors via an application performance monitoring (APM) tool like New Relic or Datadog.

SEO Optimization

1. **Page Speed:**
 - **First Contentful Paint (FCP):** Time it takes for the first visual element to appear.
 - **Target:** FCP under 2 seconds.
 - **Test:** Use Google Lighthouse or PageSpeed Insights to test.
 2. **Meta Tag Optimization:**
 - **Meta Tags Coverage:** Percentage of pages with proper meta titles, descriptions, and keywords.
 - **Target:** 100% coverage for all pages.
 - **Test:** Use tools like Screaming Frog SEO Spider to audit meta tag presence.
 3. **Content Readability:**
 - **Readability Score:** Average readability score of content on the website (Flesch-Kincaid, etc.).
 - **Target:** Flesch-Kincaid score of 60-70 (easily understandable for a wide audience).
 - **Test:** Use readability analysis tools (e.g., Hemingway Editor, Yoast SEO).
 4. **Backlink Quality:**
 - **Backlink Profile:** Number of high-quality backlinks (e.g., from authoritative domains).
 - **Target:** Achieve a consistent increase in the number of quality backlinks each quarter.
 - **Test:** Use tools like Ahrefs or SEMrush to monitor backlinks.
-

4. External Interface Requirements

4.1 User Interfaces

- Web and mobile-responsive interfaces, intuitive UI components.

4.2 Hardware Interfaces

- Server hosting, compatible user devices.

4.3 Software Interfaces

- APIs for social media, payment gateways, CMS integration.