

Team 3

CMSC 495

3/18/2024

Project Plan for Restaurant Reservation System

Prepared By: Michael Bennett and Team 3

Team Members:

Nalani Arita: Project Manager (PM)

Madilyn Jones: Test Director (TD)

Aaron Mitchell: Software Designer (SD)

Daniel Keenan: Requirements Manager (RM)/Technical Writer (TE)

Michael Bennett: Software Designer (SD)

1. Overview

Brief introduction to the restaurant reservation system, its purpose, and the team behind it.

2. Project Scope Management Plan

Project Objectives: Define the main goals for the restaurant reservation system, including user experience and technical requirements.

Deliverables: List of expected outcomes, such as design documents, code repositories, testing reports, and the final application.

Work Breakdown Structure (WBS): Breakdown of tasks assigned to team members, including preliminary research, design, coding, testing, and documentation phases.

Schedule Baseline: Timeline for each phase of the project, emphasizing milestones and deadlines.

Project Verification and Acceptance: Criteria for project success, as agreed upon by team members and stakeholders.

3. Schedule Management Plan

Outline the timeline, including start and end dates for major tasks and deliverables.

4. Cost Management Plan

Although not directly applicable to academic projects, mention any tools or resources that require financial consideration (e.g., server costs, if any).

5. Quality Management Plan

Standards for code quality, design integrity, and user experience. Mention use of code reviews, testing frameworks, and usability testing.

6. Communication Management Plan

Communication tools (Discord, GitHub, HackMD) and meeting schedules (Wednesdays at 7 PM for team meetings).

7. Risk Management Plan

Potential risks (e.g., technical challenges, delays) and mitigation strategies.

8. Requirements Specification

Detailed description of system requirements, including functional and non-functional aspects.

9. Test Plan and Results

Testing strategy, types of tests to be conducted (unit, integration, system), and an overview of the testing process.

10. Design and Alternate Designs

High-level design of the system, including front-end and back-end components. Briefly mention alternative design options considered.

11. Development History

Version control strategy, major development milestones, and a brief history of changes.

12. Conclusions

Lessons learned, strengths and limitations of the final product, and suggestions for future improvement.

Appendices

Include any relevant additional information, such as detailed schedules, test reports, user guides, and references.