SE 4485: Software Engineering Projects

Fall 2024

Project Management Plan

|  |  |
| --- | --- |
| Group Number | 7 |
| Project Title | Build a Private Architecture Assessment LLM Using Past FCG Architectural Documents |
| Sponsoring Company | The Fellows Consulting Group (FCG) |
| Sponsor(s) | Tom Hill |
| Students | Gehrig French  Brandon Hernandez  Debra Samia  Samuel Williford  Bilal Zubair |

**Software Project Management Plan**

**Software Engineering Capstone Project**

Build a Private Architecture Assessment LLM Using Past FCG Architectural Documents

**Group 7**

Gehrig French

Brandon Hernandez

Debra Samia

Sam Williford

Bilal Zubair

Department of Computer Science

University of Texas at Dallas

800 West Campbell Road

Richardson, TX 75080

**Industry Sponsor**

Tom Hill

The Fellows Consulting Group (FCG)

7356 Lane Park Ct

Dallas, TX 75225

**ABSTRACT**

This document provides an overview of the key components necessary for the successful completion of the University of Texas at Dallas software engineering capstone project. It details the project organization, lifecycle, risk analysis, resource requirements, deliverables, and schedule. Additionally, it outlines the mechanisms for monitoring, reporting, and controlling the project, as well as the professional and engineering standards adhered to throughout the process.

**TABLE OF CONTENTS**

Title 1

Abstract 2

Table of Contents 3

List of Figures 3

List of Tables 3

Introduction 4

Product Overview 4

Document Overview 4

Project Organization 4

Life Model Used 4

Risk Analysis 5

Software and Hardware Resources Requirements 6

Deliverables and Schedules 7

WBS for Project Development 7

Project Deliverables 8

WBS for Project Deliverables 8

Monitoring, Reporting, and Controlling Mechanisms 10

Professional Standards 10

Evidence the Document has been placed under configuration management 11

Engineering Standards and Multiple Constraints 13

Engineering Standards 13

Final Project Constraints 13

Additional References 13

Appendix A 13

Appendix B. 14

# **LIST OF FIGURES**

# Figure 1. Risk management (Table)

# Figure 2. Deliverable and due dates (table)

# Figure 3. Project management plan (table)

# **LIST OF TABLES**

# Risk management table

# **Project management plan table**

# **Deliverable and due dates table**

**INTRODUCTION**

This Software Project Management Plan (SPMP) outlines the technical and managerial processes necessary for the successful completion of the University of Texas at Dallas software engineering capstone project. It serves as the controlling document for all project management decisions, ensuring the development of software that adequately meets the needs and expectations of our industry sponsor.

**PRODUCT OVERVIEW**

The Fellows Consulting Group (FCG) has a database of customer architectural assessments spanning forty years and aims to provide CIOs with conversational access to this valuable data using a large language model (LLM). Due to the presence of confidential client information, existing proprietary LLMs like ChatGPT cannot be used. The project involves researching open-source LLMs that can be run locally, designing an obfuscation technique to remove confidential information, and training the localized LLM on the obfuscated data. The final step is to test the LLM with a simulated conversation and demonstrate its effectiveness.

**DOCUMENT OVERVIEW**

This document outlines the organization of the development team, including roles, responsibilities, and the rationale behind the team's structure. It details the chosen software development lifecycle model and risk analysis, including potential risks, likelihood, and mitigation strategies. The necessary software and hardware resources are identified, with justifications for their selection. The project’s schedule, activities, dependencies, and team member allocations are presented, along with the mechanisms for monitoring and controlling progress. The document also covers the professional standards expected of team members and confirms that all project deliverables have been placed under configuration management. Additional engineering standards, constraints, and references are included, with a guideline on addressing unacceptable behavior provided in an appendix.

**PROJECT ORGANIZATION**

Our team has opted for a flexible structure, where each member contributes based on their skills and project needs, rather than predefined roles. Task ownership will shift as necessary to maintain efficiency and adapt to changing requirements. Everyone is involved in decision-making and problem-solving, with specific tasks being led by individuals as needed. However, the team remains responsible for meeting deadlines and completing deliverables. Team members will work across different areas of the project—such as research, development, documentation, and testing—depending on what the situation requires. This approach allows us to stay flexible while ensuring that all tasks are handled effectively.

# **LIFECYCLE MODEL USED**

Our project follows a hybrid lifecycle model that blends Agile practices with a Waterfall structure. While we have distinct phases, the work within these phases is driven by Agile principles. We hold weekly meetings where we conduct retrospectives, review progress, and prioritize tasks from our backlog. This iterative approach ensures that we remain flexible and can adjust our work as needed throughout each phase. However, the project still progresses through clear stages, allowing us to maintain focus on deliverables while staying adaptive. This hybrid model helps us stay organized through defined phases while benefiting from the responsiveness of Agile to keep the project moving efficiently.

# **RISK ANALYSIS (Figure 2)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| **Project - Issues and Risks** | | | | | | |
| **#** | **Risk Level** | **Description** | **Resolution Plan** | **Contingency Plan** | **Due Date** | **Risk Owner** |
| R-1 | Low | Team members failing to attend scheduled meetings or fulfill commitments can disrupt progress and hinder collaboration. | If a team member cannot attend a meeting, they should notify the group promptly via online communication or text. This allows the team to adjust and reorganize with minimal disruption. | Create a group chat or text and add all group members. Ensure that push notifications are enabled so everyone receives real-time updates. | 9/13/24 | Team |
| R-2 | High | Loss of internet access or power outages, especially during winter, can disrupt work and delay project progress. | Two potential mitigation plans for internet outages involve driving to access reliable Wi-Fi. The first option is driving to UTD, allowing work to be submitted on time. The second option is driving to local libraries, coffee shops, or fast-food locations like McDonald's that offer free Wi-Fi. | Team members can make sure ahead of time to have a car, bike, or another form of transportation. Locate nearby places that offer Wi-Fi, and once there, connect and continue working as usual. | 9/13/24 | Team |
| R-3 | Med | External work or lack of time may cause delays in completing project tasks, leading to missed deadlines and falling behind schedule. | External stress and workload can sometimes lead to late submissions, which are not accepted. In cases where a student faces multiple deadlines, such as two exams and a project assignment, time management becomes crucial. The student should prioritize tasks and break their study time into manageable blocks to handle the workload effectively. | To manage heavy workloads, team members should use a calendar or scheduling app to create study blocks. Breaking tasks into manageable chunks helps prioritize deadlines and ensures timely completion of exams and projects. | 9/13/24 | Team |
| R-4 | High | Unexpected illness or emergencies can prevent team members from participating, potentially causing delays in project progress. | Illness or emergencies are unpredictable and pose a high risk to progress. However, with online communication tools like Microsoft Teams, we can stay in touch with the affected team members to ensure the project stays on track. | Create a group chat or text and add all team members. Ensure push notifications are enabled so everyone receives updates immediately when they check their phones. | 9/13/24 | Team |

# **SOFTWARE AND HARDWARE RESOURCE REQUIREMENTS**

**HARDWARE REQUIREMENTS**

Each team member must have a development machine capable of handling the computational load of natural language processing tasks. The hardware should be performant enough to support development work efficiently, especially for tasks involving machine learning or data processing.

**SOFTWARE REQUIREMENTS**

**Collaboration Tools:**

We will use Discord for communication and coordination, GitHub for version control and code sharing, and Google Drive for document storage and file management.

**Development Tools:**

We will require an IDE or code editor for software development. Potential tools include VSCode, PyCharm, or Jupyter Notebook, depending on individual preferences and project needs.

**Network Connection:**

A reliable internet connection is essential for research, collaboration, and communication. Since a stable connection is a critical requirement, we will consider backup solutions in case of outages.

# **DELIVERABLES AND SCHEDULE**

This section begins with the Work Breakdown Structure (WBS) for project development, outlining the key technical tasks required to complete the project. Following this, a list of project deliverables and their due dates is presented. The section concludes with the WBS for the project deliverables, detailing the tasks needed to complete the project documentation.

Refer to Appendix B for in-depth Project Schedule

**WBS FOR PROJECT DEVELOPMENT**

1. Research and Selection of Open-Source LLMs
   1. Identify Potential Open-Source LLM Models
   2. Evaluate LLMs Against Project Requirements
   3. Select the Most Suitable LLM
2. Setting Up the Chosen LLM to Run Locally
   1. Install Chosen LLM on Local Machines
   2. Verify LLM Installation and Troubleshoot Issues
3. Data Obfuscation
   1. Analyze Customer Interaction Data
   2. Develop Data Obfuscation Technique
   3. Implement and Test Data Obfuscation
4. Train the LLM with Obfuscated Data
   1. Prepare Obfuscated Data for Training
   2. Train the LLM
   3. Monitor and Validate
5. Demonstrate Results
   1. Design and Test Conversation Scenarios
   2. Conduct Test Conversations
   3. Prepare and Present Final Demonstration

**PROJECT DELIVERABLES**

**Figure 1.**

|  |  |
| --- | --- |
| **Deliverable** | **Due Date** |
| Project Management Plan | September 6, 2024 |
| Requirements Documentation | September 20, 2024 |
| Architecture Documentation | October 18, 2024 |
| Detailed Design Documentation | November 1, 2024 |
| Test Plan | November 15, 2024 |
| Final Project Demonstration | November 30, 2024 |
| Final Project Report | December 2, 2024 |

**WBS FOR PROJECT DELIVERABLES**

1. Project Management Plan
   1. Title Page, Abstract, Table of Contents, List of Figures, List of Tables
   2. Introduction
   3. Project Organization, Project Lifecycle, and Risk Analysis
   4. Software and Hardware Resource Requirements
   5. Deliverables and Schedule
   6. Monitoring, Reporting and Controlling Mechanisms/ Configuration Management
   7. Professional Standards/Appendix A
   8. Engineering standards and Multiple Constraints, Additional References
   9. Proofread and final formatting
2. Requirements Documentation
   1. Title Page, Abstract, Table of Contents, List of Figures, List of Tables
   2. Introduction
   3. Use Case Model for Functional Requirements
   4. Rationale for Use Case Model
   5. Non-Functional Requirements
   6. Configuration Management
   7. Engineering standards and Multiple Constraints, Additional References
3. Architecture Documentation
   1. Title Page, Abstract, Table of Contents, List of Figures, List of Tables
   2. Introduction
   3. Architectural Styles
   4. Architectural Model
   5. Technology, Software, Hardware
   6. Rationale for Architectural Style and Model
   7. Traceability from Requirements to Architecture
   8. Configuration Management
   9. Engineering Standards and Multiple Constraints, Additional References
4. Detailed Design Documentation
   1. Title Page, Abstract, Table of Contents, List of Figures, List of Tables
   2. Introduction
   3. GUI Design
   4. Static Model
   5. Dynamic Model
   6. Rationale for Detailed Design Model
   7. Traceability from Requirements to Detailed Design Model
   8. Configuration Management
   9. Engineering Standards and Multiple Constraints, Additional References
5. Test Plan
   1. Title Page, Abstract, Table of Contents, List of Figures, List of Tables
   2. Introduction
   3. Requirements/Specifications-Based System Level Test Cases
   4. Techniques for Test Generation
   5. Traceability of Test Cases from Use Cases
   6. Configuration Management
   7. Engineering Standards and Multiple Constraints, Additional References
6. Final Project Report
   1. Title Page, Executive Summary, Table of Contents, List of Figures, List of Tables
   2. Introduction
   3. Project Management Plan
   4. Requirements Specifications
   5. Architecture
   6. Design
   7. Test Plan
   8. Engineering Standards and Multiple Constraints, Additional References
   9. Acknowledgements

# **MONITORING, REPORTING, AND CONTROLLING MECHANISMS**

Team progress and task management will be monitored through weekly meetings, where each member will provide updates on their progress, discuss upcoming deadlines, and address any concerns or challenges. These meetings will take place in person, ensuring that all members are actively engaged and accountable.

In addition to weekly meetings, team members will communicate regularly through Discord, which will serve as our primary platform for day-to-day discussions, task tracking, and collaboration. GitHub will be used to track code contributions and version control, providing visibility into the development process.

# **PROFESSIONAL STANDARDS**

**Attendance and Participation:**

Team members are expected to attend all meetings on time, come prepared, and actively participate. If a member is unable to attend or anticipates being late, they should notify the team in advance. Regular attendance ensures proper coordination, progress tracking, and respect for everyone’s time.

**Task Delivery and Quality:**

All tasks should be completed by the assigned deadlines, meeting established standards of quality. Work should be accurate, thoroughly researched, and well presented. Consistently delivering high-quality work builds trust, reduces the need for rework, and aligns with professional expectations.

**Communication and Collaboration:**

Team members should maintain open and respectful communication. Tasks should be divided fairly, with everyone contributing and supporting one another. Clear communication promotes alignment, improves outcomes, and fosters a positive team environment.

**Accountability and Ownership:**

Each member is responsible for their assigned tasks and should take ownership of their work. If challenges or delays arise, they must be communicated promptly to the team. Accountability ensures transparency, smooth workflow, and timely project completion.

# **EVIDENCE THE DOCUMENT HAS BEEN PLACED UNDER CONFIGURATION MANAGEMENT**

# This document was initially created in Office 365 and shared using OneDrive, with full version history to track all edits and additions made by team members. Each team member was granted editing privileges, and our industrial mentor was provided with view-only access. Below is a screenshot showing the current version history and access control settings.

# 

# 

# 

# To ensure more robust configuration management and better handle the complexity of version control, the document has now been moved to our GitHub repository. GitHub will allow us to implement version control, track changes more efficiently, and provide a clear audit trail for any modifications going forward. Screenshots of the current version history from OneDrive are included, along with confirmation of its migration to GitHub.

# The GitHub repository can be accessed at: <https://github.com/SPWilliford/SE4485-Group7>

**ENGINEERING STANDARDS AND MULTIPLE CONSTRAINTS**

**ENGINEERING STANDARDS:**

* IEEE Std 1058-1998: Software Project Management Plans [[pdf](https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=741937)]
* IEEE Std 1490-2003: A Guide to the Project Management Body of Knowledge [[pdf](https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1302773)]
* IEEE Std 12207: Software Life Cycle Processes [[pdf](https://ieeexplore.ieee.org/document/8742773)]
* IEEE Std 15939: Measurement Process [[pdf](https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4775910)]
* ISO/IEC/IEEE Std 29148-2011: Systems and Software Engineering
* Life Cycle Processes
* Requirements Engineering [[pdf](https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6146379)]

**FINAL PROJECT CONSTRAINTS:**

* **SOURCE CODE:** All source code must be hosted on GitHub.
* **DOCUMENTATION:** All project documentation must be stored in the designated project Google folder.
* **TEST RESULTS:** Include all test results in the project Google folder.

**ADDITIONAL REFERENCES**

* Larson, E. and Gray, C., 2014. Project Management: The Managerial Process. McGraw Hill
* Humphrey, W.S. and Thomas, W.R., 2010. Reflections on Management: How to Manage Your Software Projects, Your Teams, Your Boss, and Yourself. Pearson Education

**Appendix A.**

To ensure the success of the project, all team members are expected to adhere to professional guidelines. Mutual respect is paramount, even in cases of disagreement. We aim to foster a collaborative environment where everyone feels comfortable expressing their opinions and concerns.

All assigned tasks must be completed and delivered on time. Valid reasons for delays are limited to illness, family emergencies, or academic conflicts. Failure to meet these expectations will result in the following disciplinary actions:

**First Offense:** A group discussion will be held to address the issue and collaboratively develop a solution. This will serve as the first warning.

**Second Offense:** A meeting will be scheduled with the instructor, where the issue will be discussed, and a resolution will be sought. This will count as the second warning.

**Third Offense:** Another meeting with the instructor will be scheduled. If the problem persists, the group will decide whether to remove the offending team member. The removed member will receive a grade based solely on their individual contributions to the project.

**Appendix B. (Figure 3)**





