



**UNIVERSITY
OF SUSSEX**

ADVANCED SOFTWARE ENGINEERING

PROJECT MANAGEMENT PLAN

GROUP 6: N- COMPLETION

DATE: 24/10/2023

Table of Contents

- PURPOSE..... 1**
- 1 ORGANISATION PLAN..... 1**
- 2 DEVELOPMENT PLAN..... 1**
 - 2.1 Architecture 1
 - 2.2 Methodology 1
 - 2.3 Frameworks 2
- 3 CONFLICT RESOLUTION PLAN..... 2**
 - 3.1 A Member of the Group Drops Out 2
 - 3.2 Deadline Slippage 2
 - 3.3 Handling Disagreements on Approach..... 2
 - 3.4 Technical Issues 2

PURPOSE

The intended audiences are Professor (Referred to as **client**) and the members of Project Group 6. The purpose of the project is to build a web app designed to solve various types of puzzles. At this point in time, the only designed feature of the application is to solve the *n-queens* puzzle although this will be updated over the following weeks.

1 ORGANISATION PLAN

This organizational plan is for GROUP 6 which officially consists of 8 People. To prevent a single point of failure, each member of the project will be involved in several different aspects of the project. These responsibilities are as follows:

Project Leader <i>Ensures alignment with client requirements and communication with client, manages project timelines and deliverables.</i> Rohan Kadam			
Overseer <i>Co-ordinates team activities and meetings</i> Aman			
Coding Tom, Junkai, Rimjhim, Mya, Aman	Documentation Tom, Rimjhim, Mya, Veerpal	Quality Assurance (QA) and Testing Tom, Mya, Rimjhim, Veerpal, Junkai	Database/Timeline Jawad

2 DEVELOPMENT PLAN

2.1 ARCHITECTURE

Will follow the Model-View-Template (M-V-T) pattern as our development architecture for this task as it is straight forward. After reviewing with the client, this architecture might change to microservices or service-oriented types.

2.2 METHODOLOGY

To simulate a continuous software development project, we will adopt the agile methodology to deliver feedback and features that are requested by the client. We will use GitHub for all development tracking with the customer. Feedback provided on the GitHub site will be preferred rather than physical appointments if needed.

References of code will be given where necessary from where we took inspirations from as it is a good practice to prevent plagiarism even, though it is open source.

We will use GitHub issues and new request option quite often to maintain the flow of constant feedback from the professor and also use it as a chat space for our own by QA's.

2.3 FRAMEWORKS

We will use Django as a framework for this project. Frontend will use CSS/HTML where necessary and the backend will consist of Python and JavaScript.

NB: Due to the agile nature of the project, this plan will change over the course of the term based on the client's constant feedback and additional tasks that he will provide throughout the semester.

3 CONFLICT RESOLUTION PLAN

3.1 A MEMBER OF THE GROUP DROPS OUT

In the event of a team member's dropout, we will redistribute their tasks among remaining team members to maintain project stability. If needed, we'll adjust the project timeline and keep the client informed. After resolving the issue, we'll update project documentation to record the dropout's contributions and the steps taken to ensure project continuity.

3.2 DEADLINE SLIPPAGE

In the event that a team member consistently misses deadlines, our approach involves first understanding the root causes, whether they be technical challenges, personal issues, or workload management problems, through open and empathetic communication. Once identified, we provide necessary support and assistance, which may include additional guidance or adjusting their responsibilities. If necessary, we will reallocate tasks among the team to accommodate their needs and revise the project timeline accordingly. Following issue resolution, we will document the steps taken to address the problem comprehensively.

3.3 HANDLING DISAGREEMENTS ON APPROACH

When disagreements arise about how to approach a task or project, our process involves having a thorough discussion to understand differing viewpoints. We aim to reach an agreement by considering everyone's perspective, evaluating the proposed methods based on project goals, and looking for compromises. If we can't resolve the issue, we might involve higher-ups or seek mediation while keeping open and clear communication throughout.

3.4 TECHNICAL ISSUES

In the event of a GitHub repository issue, our plan includes identifying the problem's nature, attempting rollback changes if feasible, and, if necessary, restoring data from locally backed-up repository copies while maintaining thorough documentation. In the case of a data breach or security incident, our response encompasses immediate containment, investigation, notification, remediation, and comprehensive documentation to address the situation effectively.