SOFTWARE PROJECT MANAGEMENT PLAN

1. INTRODUCTION

This is the software system proposal document for the project Pivi: An eclipse Plug-In for Visual Parallel Programming sponsored by Dr. Javier Gonzalez. This project is being undertaken by the Group 14. The team is comprised of graduate students majoring in Software Engineering at Arizona State University. The team members are enrolled in a two-semester senior project course Software Factory. Successful delivery of the desired software product will fulfill the senior project requirement for the student team members.

PROJECT SPONSOR

Name Javier Gonzalez

Company ASU

Division/Unit CIDSE

Email [javiergs@asu.edu](mailto:javiergs@asu.edu)

DEVELOPMENT TEAM

Pujitha Kara [pkara1@asu.edu](mailto:pkara1@asu.edu)

Nithya Kogaleru [nkogaler@asu.edu](mailto:nkogaler@asu.edu)

Sri Kiran Panchavati Ganesh [spanchav@asu.edu](mailto:spanchav@asu.edu)

Giridhar Peddabuttaiahgari [gpeddabu@asu.edu](mailto:gpeddabu@asu.edu)

Snehal Shendaware [sshendwa@asu.edu](mailto:sshendwa@asu.edu)

* 1. Purpose

The purpose of [this document](https://en.wikipedia.org/wiki/Project_planning) is to identify the scope of the project, [estimate](https://en.wikipedia.org/wiki/Estimation_in_software_engineering) the [work](https://en.wikipedia.org/wiki/Work_(project_management)) involved, and create a [project schedule](https://en.wikipedia.org/wiki/Gantt_chart).  It provides a baseline of what has to be achieved and gives an overview of the project and deliverables.

* 1. Scope

This document provides project overview, project organization, project management and control, technical process, the activities and schedule. The document does not cover technical details, requirement details. This is an initial proposal of the project details.

* 1. Definitions, Acronyms and Abbreviations
  2. References

1. PROJECT OVERVIEW

This Eclipse plug-in focused on facilitating teaching techniques and implications of parallel, concurrent and multicore computing. It builds inside Eclipse an atmosphere of visual programming based on icons. The tool allows students to create a visual model of a problem (by the interconnection of representative icons of atomic structures) building a graph that represent a complete programmable system. Then, automatically generates code of the problem in Java or C/C++, inlaying the necessary elements for concurrent or parallel systems, to compile, execute and run.

* 1. Project Summary

The overall project is majorly divided into three phases. The first phase includes understanding and gathering the requirements. This phase also includes understanding the purpose and scope of the project. The second phases involve prototyping a design and later implementing the design. The final phase includes testing and a hand over of the project to the sponsor.

* 1. Project Deliverables

The project deliverables can be classified into two phases. The first set of deliverables include the following

* Project Charter
* Project Management Plan
* SRS
* Demo of the project idea.

The second set of deliverables include the following

* Design Document
* Version 1 Rollout
* Version 2 Rollout
* User’s manual
* Hand Over
  1. The Management Plan and the Planning Process

An estimation of deadlines for completing each of the above mentioned phases are as follows,

|  |  |  |
| --- | --- | --- |
| **Phase** | **Estimated Start Date** | **Estimated End Date** |
| Project Charter |  |  |
| Project Management Plan |  |  |
| SRS |  |  |
| Demo |  |  |
| Design Document |  |  |
| Version 1 Rollout |  |  |
| Version 2 Rollout |  |  |
| User Manual |  |  |
| Hand Over |  |  |

The above mentioned estimated dates can be modified depending on the progress made and sponsor’s requirements.

<Dates will be updated after discussing deadlines with sponsor and instructor>

1. PROJECT ORGANIZATION

This part describes the phases of work that will be done during the course of the project. It explains in detail about the process model, organization structure of the team and responsibilities fulfilled in each phase.

* 1. Process Model

For this project we have decided the process model followed would be an agile model. We will be using Taiga as the online scrum board where we can view and act on the product backlog decided by the product owner. Each sprint will last for 15 days and contents of one particular sprint will be added as per the requirement and delivered at the end of the sprint which will move further for quality assurance.

* 1. Organizational Structure and Interfaces

The team consists of 6 members who will interchange roles like developer, tester, integrator every sprint. Although the role of manager is assigned to a single person in the team. The project manager will be point of contact to the sponsor. The developer is responsible for application development and unit testing tester will be taking care of quality assurance.

* 1. Project Responsibilities

During the first phase of the project we would have constant interactions with the product owner and discuss the scope and exact requirements of the project. Then we start drafting a Software requirement specification document and review it with the sponsor to make changes and get a sign off for the final requirements. We will start research on the technologies and practices closely related to the development of the project. This involves brainstorm session, discussions, critical analysis of the technologies that are appropriate for the application development. We will then create a prototype of the original model as a proof of concept to showcase our product to the owner and ask for feedback.

Once this is approved we move forward to implement the actual product on eclipse as a plugin with version 1 being the UI for the application with basic visualizations with in the requirements. Once this is completed we proceed to the second version where development of visualizations for multithreading operations take place and a code for the same is generated behind the scenes. This tool will be developed and tested in perspective of the end user who will be a student using this tool to learn parallel processing and multithreading using visualizations in Java.

On successful completion of this phase it will be put on Eclipse Marketplace for open source usage and if time permits we will move ahead to third version which optimizes second version and enhances it such that it can be used at an enterprise level to help develop other software.

1. PROJECT MANAGEMENT AND CONTROL

We will use Agile methodology for managing and developing this project as it can accommodate changes better. The team members will form a group in which each member will be responsible for one or more of the activities needed. We will use Git and Taiga for monitoring and measuring the progress of the project. We are planning to have two week sprints. First two sprints will be exploratory sprints where we spend time on activities like gathering requirements, understanding requirements, creating project plan, creating software requirement specification, Learning the technologies and validating the requirements. The project is validated by constant monitoring by the sponsor. Completeness of the requirements is measured by showing a proof of concept to the sponsor and accommodating his feedback.