Timeline Manager
Project Plan

1 Introduction

The project explained within this document is a JavaFX application to act as a Timeline manager. The goal with the application is to let the user create a timeline and populate it with events. It will allow users to create and manage multiple timelines with separate start and end times. The application will also allow the timelines to be saved into a text / binary file for later use. The project will lastly be built into a JAR-file to allow users use it straight out of the box.

This document is here to provide a plan for how this project should be managed and how the goals mentioned above should be pursued. The following sections will give a description of how the development process is structured for the team and technical aspects of the project. This document will also supply an overview of the milestones and objectives to keep track on while working with this project.

2 Project organization

This project is going to consist of seven members listed in a table below. The project is not divided into roles more than a project manager which has been selected to be Oskar Mendel. The project roles are divided into tasks which each member selects and works on for the week. Each member is responsible for their own tasks. A weekly meeting is held to discuss the progress together with the project in general. The working process is managed through Github where each team member creates a new branch and publish their changes to that branch, in the end of the week a pull request for each branch is submitted and the project manager reviews the branches and merges them to the master branch.

The github repository is located at the URL: *https://github.com/brokenprogrammer/Timeline-Manager*

Team members
Mendel Oskar
Alhrazy Wael
Argyriou Dimitrios
Mironov Marinela G.
Modic Milan
Nasrini Mohammed B,
Zhao Shizhen

3 Project practices and measurements

The project will be built using the Eclipse IDE together with the JUnit testing library. The project will be divided into weekly tasks which are handed out to team members and are worked on as a weekly goal. Progress will be tracked through github and diverse documentation provided like this project plan.

JavaFX has been chosen as the GUI library of choice since it is the library that every member of the project have experience with and it's the most up to date Java library right now. We are also going to use FXML files to display our views since that lets us take advantage of the Scene Builder application which we want to use to simplify the UI implementation together with reducing code.

The project structure is going to follow a MVC pattern since JavaFX allows the use of controllers for its views and a model to share data between the controllers is then easy to add, it also helps with keeping the project object-oriented.

4 Project milestones and objectives

Seminar	Primary objectives	Deadline
Week 1	Project Preparation	26/3
Week 2	Analysis	6/4
Week 3	High-level Design	12/4
Week 4	Detailed Design, Implementation & Testing	19/4
Week 5	Detailed Design, Implementation & Testing	27/4
Week 6	Detailed Design, Implementation & Testing	4/5
Week 7	Detailed Design, Implementation & Testing	11/5
Week 8	Detailed Design, Implementation & Testing	18/5
Week 9	Delivery testing, Delivery	23/5

5 Deployment

[Outline the strategy for deploying the software (and its updates) into the production environment.]

6 Lessons learned

[List lessons learned from the retrospective, with special emphasis on actions to be taken to improve, for example: the development environment, the process, or team collaboration.]