Fontys University of Applied Sciences

Eindhoven, The Netherlands

***Airport Baggage Simulation***

Project Plan



Team members: Tutor:

Aleksandar Staykov, Mr. Mikaeil Shaghelani  
Boris Tsvetanov,  
Dimitar Dyakov,  
Nikolas Zhmakin,  
Velin Ekupov.

# Version History

The following table provides information about the development of the Project Plan, including the main changes in the Project Plan (adding and editing information) and the dates for the approval of this information:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID &**  **Version #** | **Prepared by** | **Revision Date** | **Approved by** | **Reason** |
| 1.0 | The team | 11.02.2019 | Mikaeil Shaghelani | Project Plan draft |
| 2.0 | The team | 11.03.2019 |  | Final Project Plan |

Table of Contents

[Version History 1](#_Toc3155199)

[1. Introduction 3](#_Toc3155200)

[Formal Client 3](#_Toc3155201)

[Tutor 3](#_Toc3155202)

[Team 4](#_Toc3155203)

[Current Situation 5](#_Toc3155204)

[Problem 5](#_Toc3155205)

[Project Goal 5](#_Toc3155206)

[Deliverables and Non-Deliverables 5](#_Toc3155207)

[Constraints 6](#_Toc3155208)

[Risks 6](#_Toc3155209)

[2. Project Phasing 8](#_Toc3155210)

[Activities and Sprints Planning 8](#_Toc3155211)

[First Period 8](#_Toc3155212)

[2.1.2 Second Period 10](#_Toc3155213)

[Time Distribution 13](#_Toc3155214)

[3. MOSQUITO 13](#_Toc3155215)

[Skills 13](#_Toc3155216)

[Quality 14](#_Toc3155217)

[Information 15](#_Toc3155218)

[Team Organization 16](#_Toc3155219)

# Introduction

## Formal Client

**Name:** Basjan Schouwenaars

**Occupation:** Teacher at Fontys University of Applied Sciences

**Email:** b.schouwenaars@fontys.nl

**Telephone** **number:** +31885075375

**Role:** Serving as a formal representative of SIM Software Inc.

## Tutor

**Client** **name:** Mr. Mikaeil Shaghelani

**Occupation:** Teacher at Fontys University of Applied Sciences

**Email:** m.shaghelanilor@fontys.nl

**Telephone number:** +31885074259

**Role:** Advisor of “Prodavam Metla Solutions”

## Team

**Team Name**: Prodavam Metla Solutions

**Description**: Young software developers working on various projects. Developers of a simulation software in the current context.

**Project leader name**: Velin Ekupov

**Occupation**: ICT & Software Engineering Fontys Student.

**Email**: v.ekupov@student.fontys.nl

**Project secretary name**: Boris Tsvetanov  
**Occupation**: ICT & Software Engineering Fontys Student.  
**Email**: b.tstvetanov@student.fontys.nl

**Team members**: Dimitar Dyakov, Nikolas Zhmakin, Aleksandar Staykov.  
**Occupation**: ICT & Software Engineering Fontys Students.  
**Emails**: d.dyakov@student.fontys.nl, a.staykov@student.fontys.nl, n.zhmakin@student.fontys.nl

## Project Justification

SIM Software Inc. work with simulations in focused traffic. They want to expand their expertise and have a couple of ideas for the direction of the expansion. One of them is building an Airport Baggage Handling System Simulation. As innovators looking for a challenge, Prodavam Metla undertook the initiative to develop a product according to the needs of SIM Software. This document contains the plan we will build our solution upon.

## Current Situation

SIM Software has hired our team to develop an application which will simulate an airport luggage system and will provide the means to find the optimal resource allocation via statistics related to how the luggage has been processed.

## Problem

SIM Software is a company which is growing at a fast ratio and they would like to expand their business to cover a broader area of simulation software. Without this new simulation software SIM Software will not be able to grow. So far SIM Software has focused on traffic simulation and lack any solutions whatsoever for potential clients in the airport baggage handling sector.

## Project Goal

The goal of the project is to create a simulation software which will administer the complete process of luggage handling in any given airport. The software will provide meaningful statistics which will help with optimizing resource allocation.

## Deliverables and Non-Deliverables

Deliverables:

* Documentation, including URS, Project Plan, Proof of Concept and Design Document.
* Source code.
* Prototypes of the end of each iteration.
* An object-oriented application which will perform simulations of a baggage handling system.
* Process flow documentation (i.e. Meeting Agenda documents)

Non-deliverables:

* Technical equipment, this includes anything technical which is needed to transport the luggage from one point to another.
* Manuals for usage of the software.

## Constraints

*Constraint 1:* ​Time

The project should be completed within 5 months.

*Constraint 2:* ​Programming language

The applications should be programmed in an object-oriented language, which all the team members are familiar with, e.g. C#. The applications should obey the principles of the object-oriented approach.

*Constraint 3:* ​Version control

Working with a version control system (GitLab) is required.

## Risks

*Not all project features are finished on time*

* Probability: Low
* Impact: High
* To prevent: ​​Communicate often with the client
* Solution: ​​Focus on the must haves

*Conflict between team members disrupts the project*

* Probability: Low
* Impact: Medium
* To prevent: ​​Discuss the entire project together, encourage team communication
* Solution: ​​Have a meeting, find problems and then find the solutions to those problems together with the team.

*Team might not possess the required knowledge to successfully complete the project*

* Probability: Medium
* Impact: High
* To prevent: Inform tutor on time
* Solution: Contact tutor and ask for assistance

# Project Phasing[[1]](#footnote-1)

In this chapter the overview of the project tasks is shown distributed over the time the team is given for the project.

## Activities and Sprints Planning

### First Period

#### For Sprint S1 – Kick-off Phase (week 1 - 2):

Activities:

* Team forming.
* Research on algorithms that need to be implemented.
* Preparation for meeting with client.

Deliverables:

* Draft of Project Plan.
* Application Proposal.

#### For Sprint S2 – Initial Phase (week 3 - 4):

Activities:

* Create Project Plan for Iteration 1
* Update version of Project Plan & URS.
* Prepare for meeting with client/tutor.
* Create work division report.

Deliverables:

* Final version of Project Plan for Iteration #1
* Updated URS.
* Work Division Report.

#### For Sprint S3 – First half of Iteration 1 (week 5 - 7):

Activities:

* Present, Discuss and Update URS according to tutor/client wishes.
* Work on the application (code) and database.
* Create Plan for Iteration #2
* Update Work Division Report
* Prepare for meeting with client/tutor.

Deliverables:

* Final URS Version for Iteration #1
* Final version of Project Plan for Iteration #2
* Source code of proof of concept.
* Proof of concept.
* Updated work division report.
* UML Class & Sequence Diagrams.

#### For Sprint S4 – Second half of Iteration 1 (week 10):

Activities:

* UML Class diagram(s) & the non-trivial sequence diagram(s)

Deliverables:

* UML Class diagram(s) & non-trivial sequence diagram(s) of proof of concept

#### For Sprint S5 – Calibration (week 11):

Activities:

* Prepare for presentation in front of board of tutors.
* Update URS.
* Start working on Design Document.

Deliverables:

* Updated URS.
* Design Document & Testing Plan for Iteration #2

### Second Period

#### For Sprint S6 – Iteration 2 (week 12 - 14):

Activities:

* Work on finalizing URS, Design Document & test reports of Iteration #2
* Work on Project Plan for Iteration #3
* Work on creating the first prototype of the system.
* Work on cleaning source code and bugs in the application by conducting various tests.
* Update Work Division report.

Deliverables:

* Final URS, Design Document & Test Report for Iteration #2
* Final version of Project Plan for Iteration #3
* Source code, unit tests of prototype and prototype itself.
* Updated work division report.

#### For Sprint S7 – Iteration 3 (week 15 – week 17):

Activities:

* Finalize all URS, Design Document and Test report for iteration #3
* Work on cleaning bugs, conducting tests on current application functionality.
* IF there is free time, add some extra functionality.
* Finalize all work on documentation & coding.
* Create Process Report.

Deliverables:

* Final URS, Design Document & Test Report for Iteration #3.
* Source code, unit tests of final product and final product itself.
* Final version of work division report.
* Process Report.

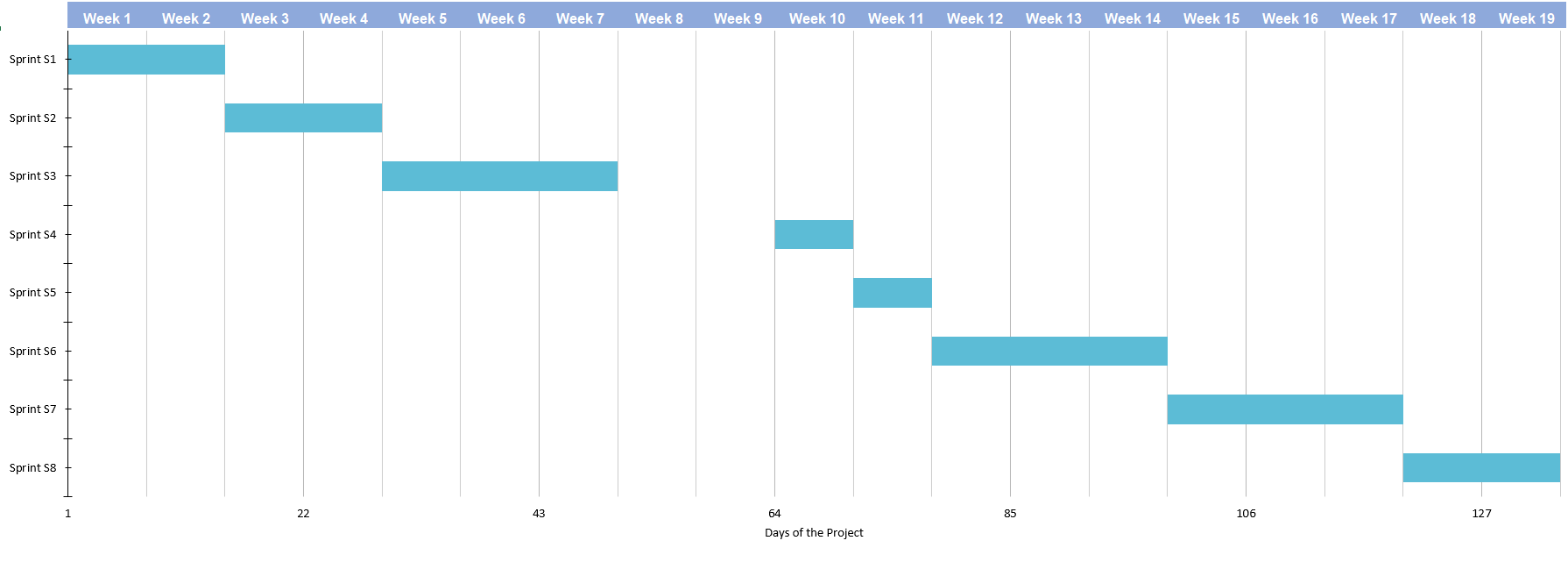
#### For Sprint S8 – End Phase (week 18/19):

Activities:

* Prepare for presentation in front of board of tutors.

Deliverables:

* Presentation about application.



## Time Distribution

The proposed time to complete this project is 180 hours (18 weeks), the team is supposed to work at least 10 hours per week. Below is how the team will allocate time for each task:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sprint** | **Documentation** | **Meetings** | **Application Design** | **Building Application** |
| S1 | 5hrs | 1hrs | 0hrs | 0hrs |
| S2 | 5hrs | 1hrs | 0hrs | 0hrs |
| S3 | 5hrs | 3hrs | 15hrs | 30hrs |
| S4 | 10hrs | 1hrs | 0hrs | 0hrs |
| S5 | 5hrs | 1hrs | 5hrs | 0hrs |
| S6 | 5hrs | 3hrs | 0hrs | 40hrs |
| S7 | 5hrs | 3hrs | 0hrs | 30hrs |
| S8 | 5hrs | 1hrs | 0hrs | 0hrs |
| **TOTAL** | **45hrs** | **15hrs** | **20hrs** | **100hrs** |

# MOSQUITO

The following chapter includes organizational topics, such as money distribution, skills needed for the successful implementation of the project, the quality of the end product and team organization and information.

## Skills

In order for this project to be fulfilled, the team must consist of people with skills in the following areas:

* Project Management – The practice of initiating, planning, executing, controlling, and closing the work of a team to achieve specific goals and meet specific success criteria at the specified time.
* C# - Object-Orient Programming language, serving as the backbone of our product.
* Software Design – The process of creating specifications of a software product.
* Object-Oriented Programming/Development – Programming paradigm based on the concept of “objects”.
* Unified Modelling Language – general-purpose modelling language, intended to provide a visual representation of a system.

## Quality

In order for the client to be content, there are certain quality aspects which need to be met:

* C# Application with complete functionality as desired by the client.
* Bug-free C# Application written in an Object-Oriented way.
* Optimized database.
* User-friendly design of application which would make the user experience more comfortable.

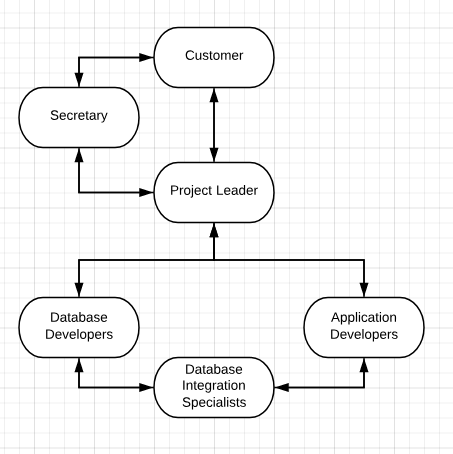
## Information

The following table contains information about the distribution of tasks in the team and the actions to them - to read, approve, write or discuss.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ~ | Project Plan | Application  Development | Database  Development | Database  relationship | Testing &  Documentation |
| Customer | [A] | [A] | [A] | [A] | [R] |
| Project Leader | [W] | [D] | [D] | [D] | [R/D] |
| Designers | [R] | [W] | [D/W] | - | [D/W] |
| C# Developers | [R] | [W] | - | [W] | [D/W] |
| Database Developers | [R] | - | [W] | [W] | [D/W] |

(Legend: R = Read; A = Approve; W = Write; D = Discuss.)

## Team Organization

The scheme below depicts the roles in the team and their connections. 

1. Project Phasing is visualized via a Gantt Chart on page 12. [↑](#footnote-ref-1)