

PROCP - Fire escape simulator

16.02.2020

[**Introduction**](#_t3g99emikqgo) **3**

[Team](#_hf4f9w5dr8zm) 3

[**Current situation**](#_yshnu2frrya2) **3**

[**Problem Definition**](#_iyr045cpsl7g) **3**

[**Project Goals**](#_gd0oiw1ed7f9) **3**

[**Project Scope**](#_6itf3z60tu0q) **4**

[**User Requirement Specification**](#_bdz33wd42yk2) **4**

[MoSCoW (must, should, could, won’t)](#_xbwsurhaox9n) 4

[Functional Requirement](#_rdriyyb4y32m) 4

[Non-functional Requirement](#_7ytl2y3vosgs) 5

[Constraint](#_a9nbfwk9ztf) 5

[**Project Deliverables**](#_zae401pdo5nf) **5**

[**Project Non-deliverables**](#_pzoqwnnd2c1b) **6**

[**Phasing**](#_hcf297qxzovm) **6**

[Activities & Tasks](#_2ldn1xgktiur) **7**

[Project Risk](#_kqz2zwol6tok) 7

# 

# 

# 

# 

# 

# Introduction

Emin Thaqi is the representative of SIM Software Inc. SIM Software would like to extend its expertise to cover a broader area of simulation software. So the company has asked for a software solution in the area of simulation software.

## Team

|  |  |  |
| --- | --- | --- |
| Name | Role | Contact |
| Chandieka Tasbihyantra | Project manager, Software Engineer | c.tasbihyantra@student.fontys.nl |
| David Stanislavov | Software Engineer | d.stanislavov@student.fontys.nl |
| Raymond Hildering | Software Engineer | r.hildering@student.fontys.nl |
| Siyana Stoyanova | Secretary, Software Engineer | siyana.stoyanova@student.fontys.nl |
| Jelle Schroijen | Software Engineer | j.schroijen@student.fontys.nl |
| Vic Verberne | Software Engineer | v.verberne@student.fontys.nl |

# Current situation

Currently, the company cannot guarantee that the placing of fire extinguishers is statistically optimal. This may result in more casualties and/or damages than with (optimal) best solution placement.

# Problem Definition

The current system is not optimal, because the current placement of fire extinguishers is not statistically guaranteed.

# Project Goals

The main goal of the project is to build a working application that can simulate a fire emergency so that it can determine the most optimal placement of fire extinguishers.

# Project Scope

The project will result in a working application, which runs simulations based on user input. The application will determine the most effective placement of fire extinguishers for a given number of fire-extinguisher and people. The project will be completed by the end 4th semester (June).

# User Requirement Specification

## MoSCoW (must, should, could, won’t)

### Functional Requirement

|  |  |  |
| --- | --- | --- |
| **No** | **Requirement** | **Priority** |
| 1 | The system should be able to make the people move to the closest accessible fire extinguisher | Must |
| 2 | The system should be able to time the simulation | Must |
| 3 | The system should be able to place people randomly | Must |
| 4 | The system should be able to place fire randomly | Must |
| 5 | The system should be able to place fire-extinguisher randomly | Must |
| 6 | The user should be able to input the amount of fire-extinguisher | Must |
| 7 | The user can specify the locations of the people | Should |
| 8 | The user can specify the locations of the fires | Should |
| 9 | The user can specify the locations of the fire-extinguishers | Should |
| 10 | The system should be able to make the people with fire extinguishers go to the closest fire. | Must |
| 11 | The system should be able to make the people with fire-extinguishers extinguish the fire | Could |
| 12 | The system should be able to allow the user to design a floor plan | Could |
| 13 | The system will make the people, fire, and fire-extinguisher colour-coded | Should |
| 14 | The application should calculate the most efficient location for fire extinguishers based on the entered layout of the floor. | Should |
| 15 | The application will simulate multiple floors | Won’t |
| 16 | The application will simulate smoke from the fire | Won’t |
| 17 | The application will take light injuries into account | Won’t |

### Non-functional Requirement

|  |  |  |
| --- | --- | --- |
| **No** | **Requirement** | **Priority** |
| 1 | The application will have an intuitive GUI | Should |
| 2 | The application will run smoothly | Must |
| 3 | The application should have a modern looking graphical user interface. | Should |
| 3 | The application will be monkey proof | Should |

### Constraint

|  |  |  |
| --- | --- | --- |
| **No** | **Requirement** | **Priority** |
| 1 | The system should only allow the fire to spread within the predetermined floor | Must |
| 2 | The system should only allow people to move within the predetermined floor | Must |

# Project Deliverables

|  |  |  |
| --- | --- | --- |
| **No** | **Deliverable** | **Deadline** |
| 1 | Project plan |  |
| 2 | Source code (proof of concept, prototype and etc) |  |
| 3 | URS |  |
| 4 | Unit Test Report |  |
| 5 | Design Document |  |
| 6 | Work division report |  |
| 7 | A working application that met the agreeable specification |  |
| 8 | plan for iteration |  |

Note: the deadline specify here is the obligatory one that needs to be hand in.

# Project Non-deliverables

1. Maintenance
2. Manual
3. Training

# Phasing

|  |  |  |
| --- | --- | --- |
| Phase | Deliverables | Start and End Dates |
| Kick-off phase | * Project plan * Proposal | Week 1-2 |
| Initial phase | * URS * Work division report * Iteration 1 planning | Week 3-4 |
| Iteration One | * URS * Working application * Work division report * Iteration 2 planning | Week 5-9 |
| Calibration session | * Progress presentation | Week 10 |
| Iteration Two | * URS * Design document * Working application * Unit tests * Work division report * Iteration 3 planning | Week 11-13 |
| Iteration Three | * URS * Design document * Test report * Working application * Unit tests | Week 14-16 |
| End phase | * Process report * Final presentation | Week 17-18 |

## Activities & Tasks

TBD

## Project Risk

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk | Likelihood | Impact | Mitigating action | Contingent action |
| The task breakdown is not optimal | Medium | Low | Make sure to breakdown each task | Reorganize schedule |
| Delay in project phases jeopardizes the ability to meet fixed date | Low | High | Leave enough time for each of the phases in the planning | Finish the phase as quick as possible and move on the next |
| Added workload or time requirements because of a new direction | High | High | Decide the direction of the solution in the beginning and stick to it | Try to use what is already there and adapt it to the new idea |
| Project deliverables unclear | High | Medium | Decide on the deliverables at the beginning of the project through proper consulting with the team and client | Organize a meeting with the entire team and client present, straighten out deliverables |
| One or multiple team members not available | Medium | Medium | Keep in mind the schedule and availability of each team member | Inform the team in case of a schedule change |
| Inaccurate time estimations | Medium | Medium | In the planning, leave some room available for unexpected activities | Make use of the planned additional time |