

FIRE ESCAPE SIMULATION APP

User Requirements Specifications



Students:

Son Duong - 3471381 - s.duong@student.fontys.nl

Imran Touqi - 3475379 - [i.altouqi@student.fontys.nl](mailto:i.altouqi@student.fontys.nl)

Dean Selivestrov - 3021270 - d.selivestrov@student.fontys.nl

Huy Bui - 2790106 - h.bui@student.fontys.nl

Abraham Ackom-Mensah - 2983214 - a.ackommensah@student.fontys.nl

Redzhep Molaahmed - r.molaahmed@student.fontys.nl

Contents

[Version History](#_heading=h.1fob9te) **2**

[Introduction](#_heading=h.3znysh7) **3**

[Requirements](#_heading=h.2et92p0) **3**

[Functional Requirements](#_heading=h.tyjcwt) 3

[Non-Functional Requirements](#_heading=h.vyn8ogb3j088) 4

[Use-Cases](#_heading=h.1t3h5sf) **5**

[User Interface](#_heading=h.bnslczszoa6w) 9

# Version History

|  |  |  |
| --- | --- | --- |
| Version Number | Date | New changes |
| Version 2019.1 | 09/14/2019 | * The first iteration of URS |
| Version 2019.2 | 09/21/2019 | * Adding requirements |
| Version 2019.3 | 09/29/2019 | * Modify use cases and functional requirements |
| Version 2019.4 | 08/10/2019 | * Add new use cases and UI |
| Version 2019.5  Version 2020.1 | 25/10/2019  07/01/2020 | * Updated version * Use cases and Updated UI |

# Introduction

This document states all the requirements the client specifies and what we think is necessary in order to give out a good product. It includes the functional requirements as to how the app is going to behave during run-time and what is needed in order to function that way. As Well as non-functional requirements as to how reliable, maintainability, usability and maybe security that all help the app to run well and overall improve the performance of the app.

# Requirements

## Functional Requirements

**Must have:**

**Automated Simulation:** The simulation will run automatically. This includes placements of the fire extinguishers, people and fires and how they escape the building.

**Generate Results:** The app will display all the statistical details of the simulation through the results tab.

**Simulation Options:** The user can edit the settings of the simulation. This includes the number of people on the floor, the number of fire extinguishers, number of fires and level of fire. Aswell of placements of fire extinguishers that would randomly appear on that floor.

**Saving results:** The statistics from different simulations over time are going to be saved in the database in order to provide further access and can be displayed with a graph.

**Could have:**

1. Define the number of people in groups within the selected floor
2. When a fire occurs, the screen changes its colour and the fire alarm sound appears in the background (audio added to the app).
3. Visual options, such as colours to show the state of the people as more time goes by and the fires expand.

**Would have:**

1. The user would have to log in to access some parts of the app (saving data,..).

## Non-Functional Requirements

**Stability:** The simulation should run smoothly includes the performance of the application, the movements of people - as objects in the simulation,.etc. The application should start as soon as the start button is pressed.

**Flexibility:** The application contains a panel where a user could change various variables in order to simulate as many scenarios as possible

**Extensibility**: the app is composed of loosely coupled components so that the application can be easily extended in the future with new functionality

# Use-Cases

**1. Selecting the Number of people**

Goal: Place the desired number of people on the simulation interface

Actor: User

Pre-condition: Simulation app is open and the number of simulations is 1.

MSS:

1. The user selects a number of groups on the floor.
2. The user inputs the number of people within each group
3. The user selects the create button
4. The application randomly allocates the groups icons to the desired positions on the simulation interface.

5. The system displays the selected number of people to the floor in their desired positions.

**2. Selecting the Number of Fire**

Goal: Generating the fire in the floor plan.

Actor: User

Pre-condition: System displays the interface of the app

MSS:

1. The user selects a number of fire in the floor.
2. The user inputs the intensity of the fire.
3. The user selects the create button.
4. Simulation application randomly allocates the fire icons to desired positions on the simulation interface.

**3. Running one simulation**

Goal: Display and run the simulation with the desired options

Actor: User

Pre-condition: Use case 1 and 2 are completed.

Post-condition: Simulation results are shown (Use case 4)

MSS:

1. User clicks on the start button.
2. The system displays a visual simulation of the fire escape.

**4. Displaying results**

Goal: Display result tab of the simulation

Actor: User

Pre-condition: Use case 3 is completed.

MSS:

1. Simulation Application shows the results tab

**5. Displaying singular simulation statistics**

Goal: Display statistical results of the simulation run

Actor: User

Pre-condition: Use case 4 is completed.

MSS:

1. The user selects the show statistics button
2. Simulation application shows the Statistics of the last run application

**6. Running multiple simulations**

Goal: Display and run the simulation with the desired options

Actor: User

Pre-condition: All the required fields have been filled by the user.

MSS:

1. The user inputs a total number of simulations which they want to run.
2. The user inputs a total number of groups.
3. User inputs total of people within each group.
4. The user inputs the number of fires.
5. User inputs the fire intensity.
6. System randomly allocates the position of the groups and fires.
7. User clicks on the start button.
8. The system displays a loading bar.

Alternate path:

1. 5a. User does not input the total number of people Nor number of groups
   1. Systems display an error pop up asking users to input the desired fields
   2. System returns user the simulation options interface

**7. Reviewing results from a database**

Goal: Display previous results from previous statistics

Actor: User

Post-condition: Use case 3 or 6 must have been done once before

MSS:

1. User clicks on the history tab
2. Simulation App Displays the history tab with all previously simulations.
3. The user selects the column of the simulation they would like to view
4. Simulation App Displays the statistics of the previously saved simulation.

**8. Showing the optimized location of the recommended fire extinguisher.**

Goal: Displaying the fire extinguisher location of the floor.

Actor: User

Pre-condition: Use case 6 has been done.

MSS:

1. User clicks on a display extinguisher.
2. The system displays the floor plan with the built-in extinguisher on it.

**9. Showing multiple simulation statistics.**

Goal: Giving the user an overview of all scenarios outcome.

Actor: User

Pre-condition: Use case 6 has been done.

MSS:

1. The system shows a button “show results” after finishing all simulations.
2. The user clicks on the aforementioned button.
3. The system displays related data in graphs.

# User Interface

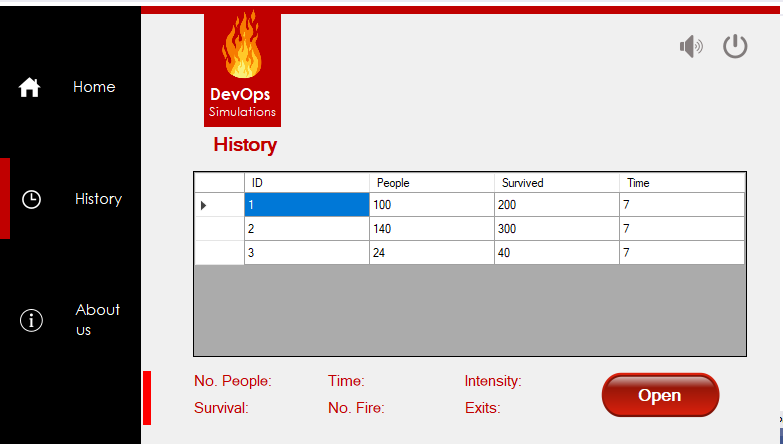
**Home Page**

The following page will provide general information about the Simulation app as well as a Start Button, The page will also have side navigations in which the user can navigate. These pages include Home, History and About us page.

****

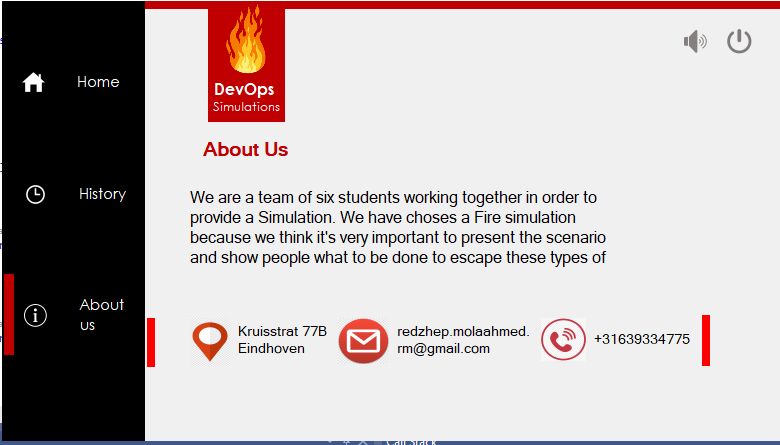
**History**

This page will display previous Simulations that the app has run. This is just a placeholder for now as will be updated later.



**About us**

This page will provide general information about the team, including contact details.



**Simulation**

This is where the simulation takes place, the simulation will instantly run when opened. The person would make its way to the shortest exit possible.

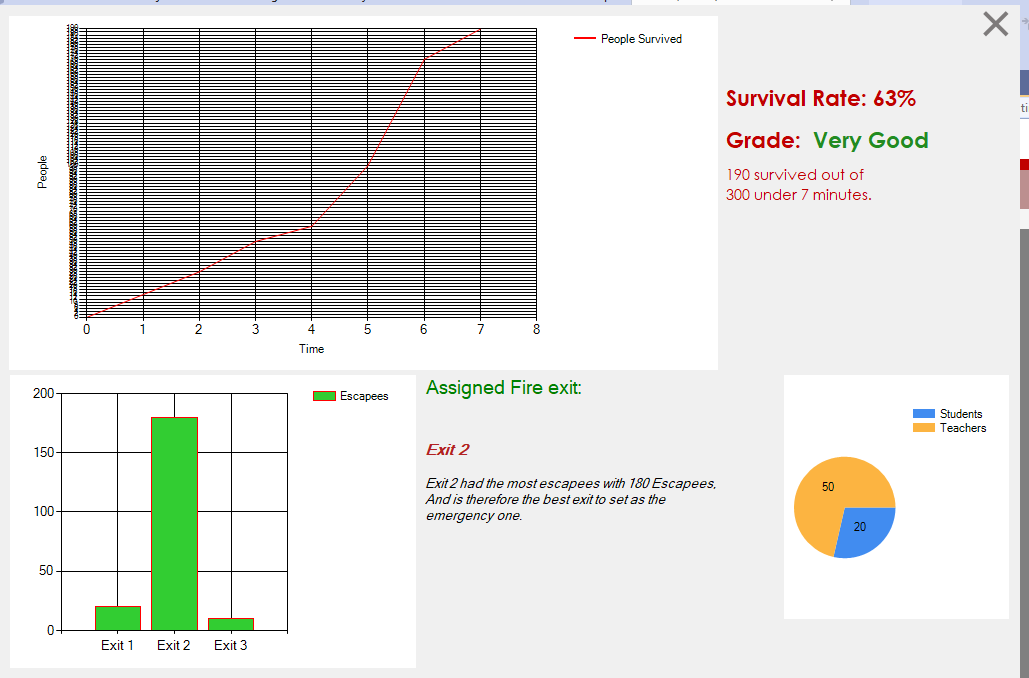
# 

**Results and statistics**

After each simulation phase is completed users are presented with the results tab, where they can select if done or they want to see a graphical representation of the results.



* Singular simulation statistic



* Multiple simulation statistics