2020

User Requirement Specification

PROCP – CLIMATE SIMULATION

Kiselkov,Borislav B.K. ; Krumov,Daniel D.K.; Lakov,Plamen P.; Nicolaidou Louttchenko,Kathy K.; Vrachanski, Ognyan O.P. ; Zaykov,Plamen P.Z.

25/02/2020 | VERSION 0.1

Contents

Introduction

This document will introduce our client/mentor with the functional requirements of our application, both functional and non-functional. Additionally, all possible user interactions with the application will be described through use-cases.

Functional Requirements

№	Requirement	Description	MoSCoW
1	Import desirable input	The user can simulate any scenario they want to witness.	Must
2	Configure the simulation	The user can change any of the input variables in order to change/observe the output	Must
3	Export an output	The user can export a finished simulation.	Must
3.1	Import an existing simulation	The user can import an existing simulation from previous or example outputs.	Must
4	Show graphs	The user can see the information depicted on graph statistics.	Must
5	Start/Pause simulation	The user can start or pause the simulation at any time to observe the result in debt.	Must
5.1	Reset simulation	The user can reset the simulation from the beginning with initial variables.	Could
6	User Authentication	User is able to register and login.	Should
7	Animate results	The user can observe the results with proper smooth animation.	Should
8	Compare results	The user can compare results between different outputs.	Could

Non-Functional Requirements

№	Requirement	Description	MoSCoW
1	Robustness	All the errors, such as wrong input or no input at all that can occur will be caught with an exception. Once an exception occurs, there will be a descriptive error message displayed to the user to show them what is wrong while reassuring the application does not crash.	Should
2	Compatibility	The project will be compatible on any machine that can run a common browser, and perhaps other kind of platforms in the future.	Should
3	Usability	The application will have as much as a self-explanatory interface to work with as possible, with backup help/information if needed. This will allow the user to perform the tasks they desire with ease and clarity.	Should
4	Performance	The application will make sure to not unnecessarily overwhelm the memory and processing capabilities of the user's machine, so it runs smoothly. Also, the application should be able to support several users at the same time.	Should
5	Reliability	The application should be consistent in working as expected every time it is used.	Must
6	Accuracy and Precision	The application should operate with real-world data for the best results possible.	Should

Use Cases

Use Case Table

№	Use Case	Description	Functional Req. №
1	Start a new Simulation	Specify the input values.	1, 2, 4, 5
2	Start an Existing Simulation	Choosing an existing scenario to import.	3.1, 4, 5
3	Save simulation	Saving the simulation in the account of the user and the database.	1, 3
4	Interact with Simulation	Choice of changing the predefined scenario input variables.	1, 2, 4, 5 5.1, 7
5	Register account	User makes an account	6
6	Login into account	User accesses his account	6

Use Case Descriptions

Assumption Use Case 1:

User does not need an account to try a simulation.

Use Case 1

Name: Start a new simulation

Actor: User

Pre-Condition: User is on the welcome page

Trigger: User signals to the system that wants to create a new simulation

Main Success Scenario (MSS):

1. System redirects the user to the simulation page.
2. User fills the inputs in.
3. User signals to the system to start the simulation.
4. System starts the simulation.

Extensions:

2a., 3a. The user chooses to cancel:

1. Use case ends

Assumption Use Case 2:

User does NOT need an account to use the preset simulation but NEEDS an account to load existing simulations.

Use Case 2

Name: Start an existing simulation

Actor: User

Pre-Condition: User is on welcome page

Trigger: User signals to the system that want to load a new simulation

Main Success Scenario (MSS):

1. System pops up a window from where the user can choose a simulation.
2. User chooses a simulation.
3. System redirects the user to the simulation page
4. User signals the system that they want to start the simulation.
5. System starts the simulation.

Extensions:

2a. User is not logged in or has no account:

2.1. User chooses the default simulation:

1. Use case goes to step 3 of MSS

2.2. User signals to the systems that wants to login:

1. System redirects user to "***Login***" page. (Use Case 6 – Login)
2. Use case goes to step 1 of MSS depending on the outcome of the sub-use case

2.3. User signals to the systems that wants to register:

1. System redirects user to "***Register***" page. (Use Case 5 – Register)
2. Use case goes to step 1 of MSS depending on the outcome of the sub-use case

2b., 4a. User signals to the system to cancel:

1. System closes the pop-up window
2. Use case ends

Use Case 3.

Name: Save simulation

Actor: User

Pre-Condition: Simulation is already configured, and user is signed in

Trigger: User signals to the system that wants to save the simulation

Main Success Scenario:

1. System asks for name of simulation.
2. User inputs name of simulation.
3. System saves the simulation.

Extension:

2a. User signals to the system to cancel:

1. Use case ends

Assumption Use Case 4:

User will input the required values and start the simulation. User can pause the simulation during runtime. User changes input values either for all countries or for each country by pressing on the desirable country.

Use Case 4.

Name: Interact with simulation

Actor: User

Pre-Condition: Simulation is running

Trigger: User signals to the system that wants to pause the simulation

Main Success Scenario:

1. System pauses simulation.
2. User interacts with the inputs as desired.
3. User signals to the system that wants to resume the simulation.
4. System continues the simulation with the newly defined values.

Extensions:

2a., 3a. User signals to the system to cancel:

1. Use case ends.

2b. User decides not to change the values:

1. Use case goes to step 3 of MSS

Use Case 5.

Name: Register

Actor: User

Pre-Condition: User is on the Register page

Main success scenario (MSS):

1. User fills in all required fields.
2. System confirms the registration.
3. System redirects to the previous page of the user.

Extensions:

1a., 2a. User signals to the system to cancel:

1. System redirects to the previous page of the user

2b User has not filled in all required fields:

1. System displays which field still need to be filled in.
2. Use case goes to step 1 of MSS

3a There is already a user with the same credentials.

1. System displays appropriate message.
2. Use case goes to step 1 of MSS

Use Case 6.

Name: Login

Actor: User

Pre-Condition: User is on the Login page

Main Success Scenario (MSS):

1. User fills in the required field.
2. System confirms credentials.
3. System redirects to the previous page of the user.

Extensions:

1a., 2a. User signals to the system to cancel.

1. System redirects to the previous page of the user
2. Use case ends

2b. User has not filled in all required fields

1. System displays which fields still need to be filled in
2. Use case goes to step 1 of MSS

3a. There is no such user

1. System displays appropriate message
2. Use case goes to step 1 of MSS