COS 301
Main Project

Q & A

Project Overview

Worlds consist of cells which have different states. Rules affect different states and cells act accordingly.

Vision

Interface
- Gamification
- User-friendly

Shareable
- Academic ideas

12TERA

Project Introduction

Roles

Project Overview

Cellular Automaton

A cellular automaton consists of a regular grid of cells, each in one of a finite number of states, such as on and off (in contrast to a coupled map lattice). The grid can be in any finite number of dimensions. For each cell, a set of cells called its neighborhood is defined relative to the specified cell. An initial state (time t=0) is selected by assigning a state for each cell. A new generation is created (advancing t by 1), according to some fixed rule (generally, a mathematical function) that determines the new state of each cell in terms of the current state of the cell and the states of the cells in its neighborhood.

A good example of this would be: **Game of Life by John Conway**

Requirements

Functional
- A user can sign in anywhere in the world
- A user can manage (create, edit, delete) cells, states, worlds and rules
- A user is able to share his/her ideas with other users

Non-functional
- Accessability - Create a server to be available online
- Privacy - Use Spring Security for authentication and access-control
- Response time - Robust REST back-end, rich JavaScript front-end
- Portability - Can be accessed on different browsers (Chrome, Safari, Opera, Firefox, etc.)
- Usability - User-friendly interface (Gamification)

Team Members

Christo Brits
11080923

Stephan Viljoen
11008408

Chris Crossman

Client

entelect

Our long term vision is to be the world's best software engineering and solutions company, measured by the value we deliver to our clients, our employees and the greater community.

Contact Persons
Reshal Hurbans (Team Lead, Software Engineer)
Sheldon Lyne (General Manager of Operations)

Implementation

Languages
- Java
- JavaScript
- HTML5
- CSS3

Network Structure
- Robust REST back-end
- Rich JavaScript front-end

COS 301
Main Project

Q & A

Project Overview

Worlds consist of cells which have different states. Rules affect different states and cells act accordingly.

Vision

Interface
- Gamification
- User-friendly

Shareable
- Academic ideas

12TERA

Project Introduction

Roles

Project Overview

Cellular Automaton

A cellular automaton consists of a regular grid of cells, each in one of a finite number of states, such as on and off (in contrast to a coupled map lattice). The grid can be in any finite number of dimensions. For each cell, a set of cells called its neighborhood is defined relative to the specified cell. An initial state (time t=0) is selected by assigning a state for each cell. A new generation is created (advancing t by 1), according to some fixed rule (generally, a mathematical function) that determines the new state of each cell in terms of the current state of the cell and the states of the cells in its neighborhood.

A good example of this would be: **Game of Life by John Conway**

Requirements

Functional
- A user can sign in anywhere in the world
- A user can manage (create, edit, delete) cells, states, worlds and rules
- A user is able to share his/her ideas with other users

Non-functional
- Accessability - Create a server to be available online
- Privacy - Use Spring Security for authentication and access-control
- Response time - Robust REST back-end, rich JavaScript front-end
- Portability - Can be accessed on different browsers (Chrome, Safari, Opera, Firefox, etc.)
- Usability - User-friendly interface (Gamification)

Team Members

Christo Brits
11080923

Stephan Viljoen
11008408

Chris Crossman

Client

entelect

Our long term vision is to be the world's best software engineering and solutions company, measured by the value we deliver to our clients, our employees and the greater community.

Contact Persons
Reshal Hurbans (Team Lead, Software Engineer)
Sheldon Lyne (General Manager of Operations)

Implementation

Languages
- Java
- JavaScript
- HTML5
- CSS3

Network Structure
- Robust REST back-end
- Rich JavaScript front-end


12TERA
Project Introduction
Prezi

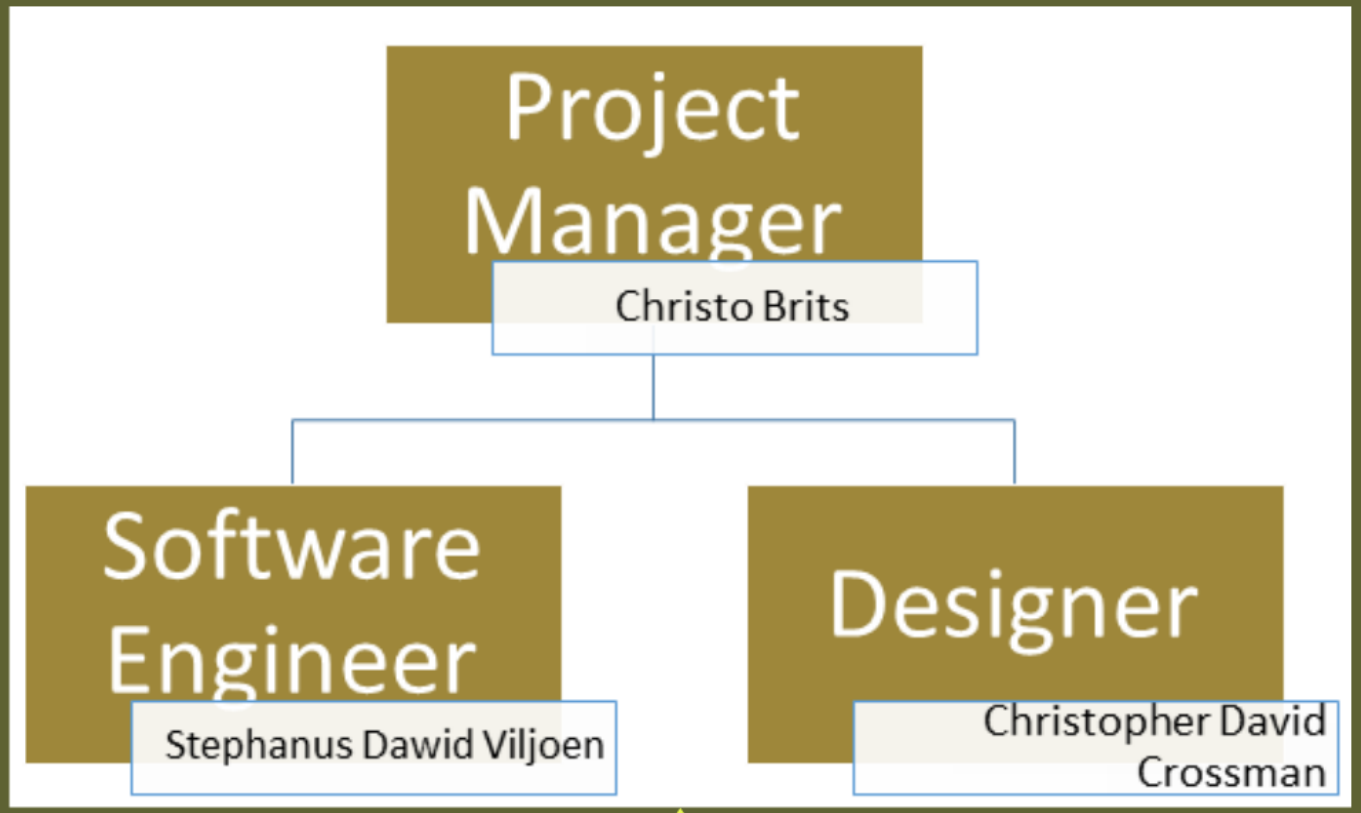
Team Members

Christo Brits
11080923

Stephan Viljoen
11008408

Chris Crossman

Roles


Project Manager
Christo Brits
Software Engineer
Stephanus Dawid Viljoen
Designer
Christopher David Crossman

Client

Our long term vision is to be the world's best software engineering and solutions company, measured by the value we deliver to our clients, our employees and the greater community.

Contact Persons
Reshal Hurbans (Team Lead, Software Engineer)
Sheldon Lyne (General Manager of Operations)

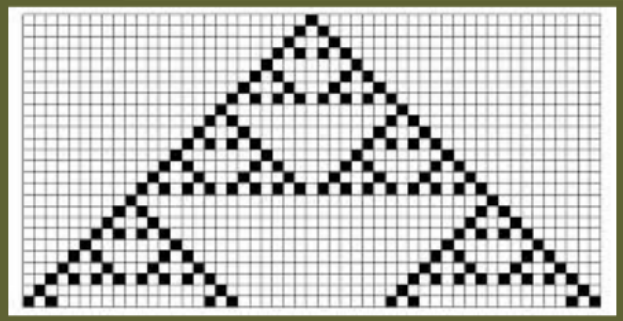
Project Overview

Cellular Automaton

A cellular automaton consists of a regular grid of cells, each in one of a finite number of states, such as on and off (in contrast to a coupled map lattice). The grid can be in any finite number of dimensions. For each cell, a set of cells called its neighborhood is defined relative to the specified cell. An initial state (time t=0) is selected by assigning a state for each cell. A new generation is created (advancing t by 1), according to some fixed rule (generally, a mathematical function) that determines the new state of each cell in terms of the current state of the cell and the states of the cells in its neighborhood

A good example of this would be **Game of Life by John Conway**

Project Overview



1D

3D

2D

Worlds consist of cells which have different states. Rules affect different states and cells act accordingly.

Requirements

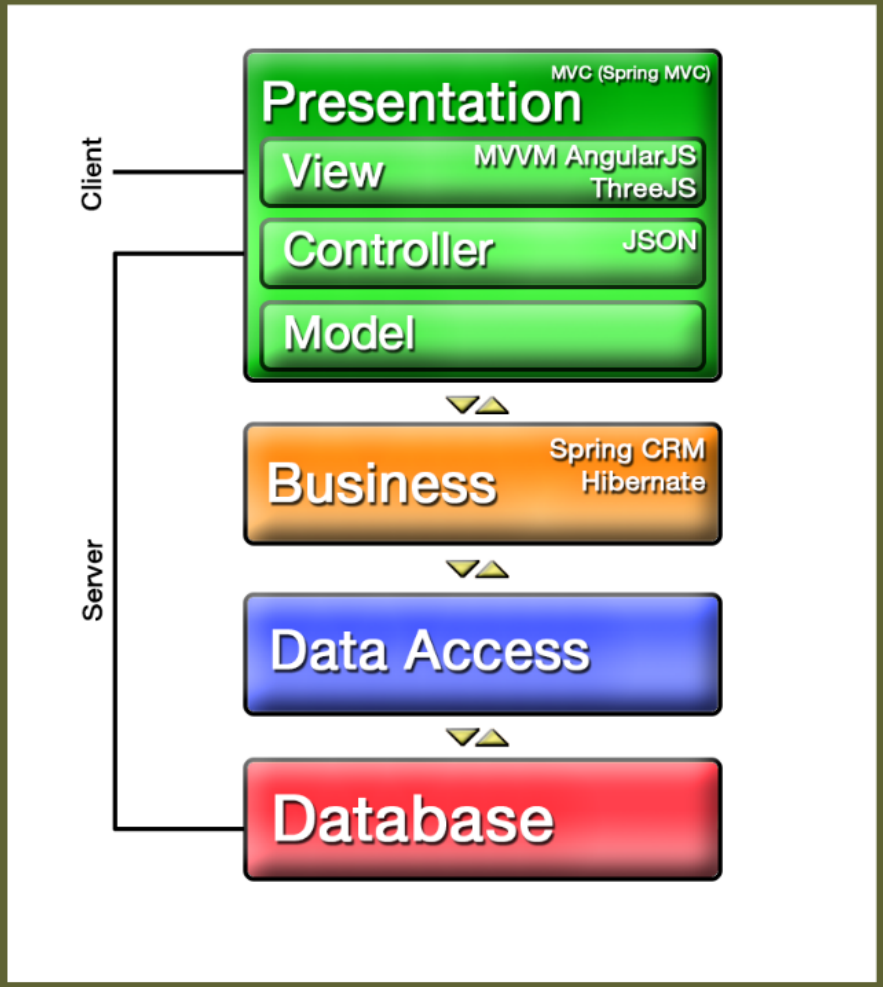
Functional

- A user can sign in anywhere in the world
- A user can manage (create, edit, delete) cells, states, worlds and rules
- A user is able to share his/her ideas with other users

Non-functional

- Accessibility - Create a server to be available online
- Privacy - Use Spring Security for authentication and access-control
- Response time - Robust REST back-end, rich JavaScript front-end
- Portability - Can be accessed on different browsers (Chrome, Safari, Opera, Firefox, etc.)
- Usability - User-friendly interface (Gamefication)

Implementation



Languages

- Java
- JavaScript
- HTML5
- CSS3

Network Structure

- Robust REST back-end
- Rich JavaScript front-end

Vision

Interface

- Gamification
- User-friendly

Shareable

- Academic ideas

Q & A