Android Project

...

Students:

Teyssier Titouan

Pelloin Valentin

February 7, 2019

Contents

1 | Our application : Visual Life Configurator

Our application is a cellular automaton simulation game. The player is free to create new automatons with the rules we want (for example Conway's game of life, Wireworld, ...). For each automaton, the player configures how the automaton work, the types of cells, how neighbours are calculated, and the rules for a cell to be transformed into an other one.

Then, the player can create as many worlds he wants, edit them, and play with them.

Our application is called `VLC`, it stands for Visual Life Configurator.

1.1 Persona

Name John Doe

About John studies computer science. He is interested into algorithms and life simulation. John likes playing games on his phone while waiting for his bus before going to the University.

Goals

- Learn new things

- S

1.2 User stories

- As a player, I want to entertain myself by playing a game ;

- As a user, I want to learn new ;

- As a player, I want to experiment new configurations of cellular automatons

2 | Functional study

2.1 Storyboards
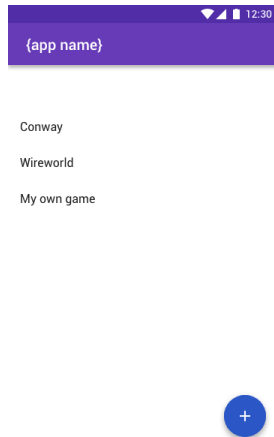
Below, the mock-ups of our application:

Figure 2.1 – The main activity of VLC.
The player can find all the automatons he created, and click on them to edit them or to play on a game.
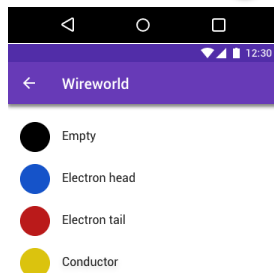The player can also click on the "plus" button to create and configure a new automaton.

Figure 2.2 – Inside a automaton, cells tab.
The player can create and remove cells types for this automaton. He can reorganize the order of the cells. The first cell type is the default one, the type of the cell that will fill the grid when creating a blank world.
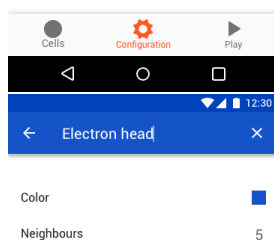
Figure 2.3 – Inside a automaton, configuration tab.
In this tab, the player can access the configuration for each cell.

Figure 2.4 – Inside a automaton, play tab.
In this tab, the user can access to it's saved games. He can also choose to play on a random world.
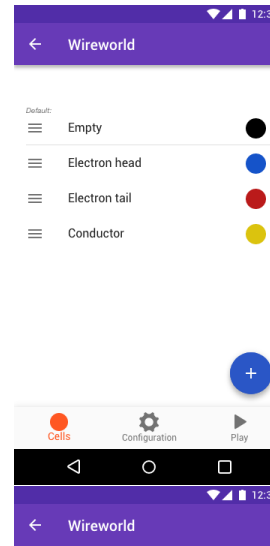
Figure 2.5 – Cell parameter activity.
For this specific cell type, the user can edit its name, its color. He can also access to the neighbours choosing screen.
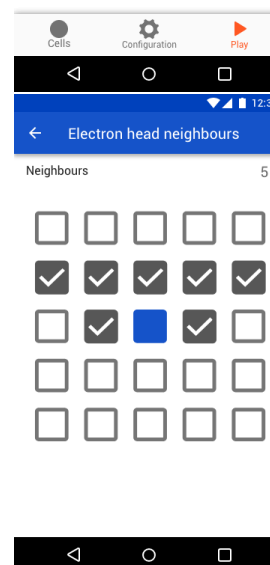
Figure 2.6 – Cell neighbours choosing.
On this activity, the player can choose which cells are being counted by the automaton as neighbours for this specific cell type.
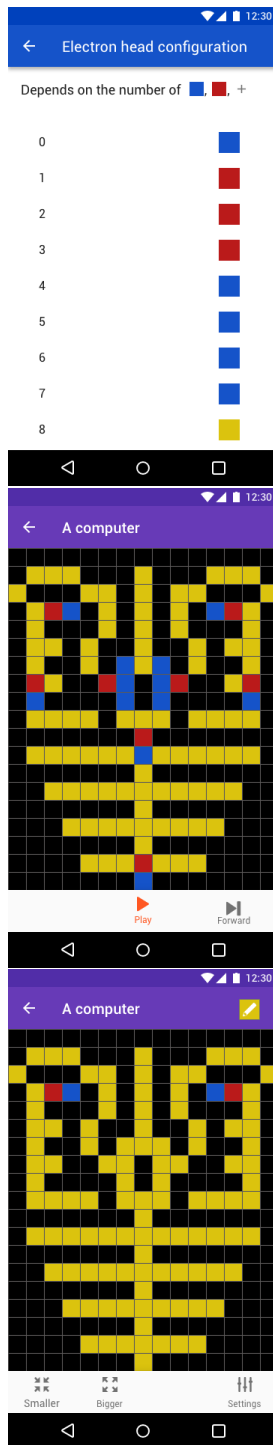
Figure 2.7 – Cell type rule edition.
For this cell type (blue one), the player configured the rules of transformation. For example, if there are 8 neighbours around a blue cell, it will be transformed to a yellow cell in the next generation.
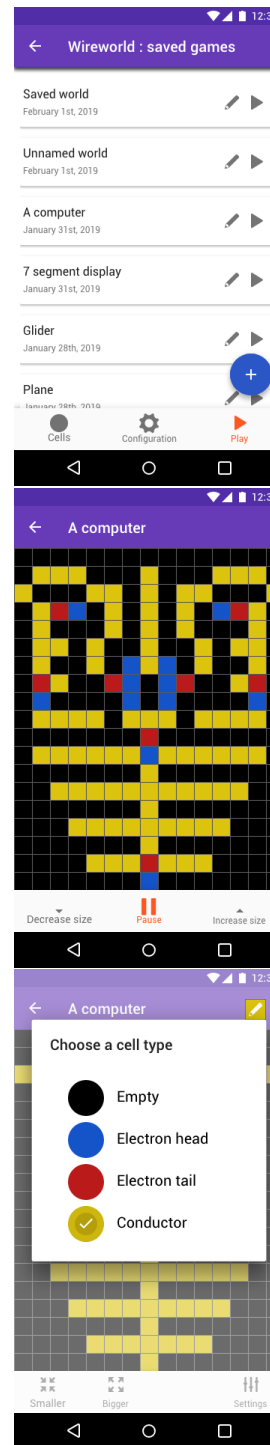

Figure 2.8 – List of saved games.
The player can list worlds that are existing for a automaton. For each world, he can edit it, or play on it. By clicking on the "plus" button, e can also create a new blank world.

Figure 2.9 – Playing game activity : paused.
By clicking on "Forward", the player can generate the next step.

Figure 2.10 – Playing game activity : playing.
The player can zoom in or zoom out.

Figure 2.11 – Editing a game.
The player can drag on cells to color them.

Figure 2.12 – Editing a game.
The player clicked on the pen in toolbar. He can change the color of cells he will be placing.

It is possible to view a demonstration of these mock-ups by clicking on this

2.2   Architecture

d

2.3   Github link

Our project is accessible at `https://github.com/valentinp72/VisualLifeConfigurator`.