

# Software Requirements Specification for Game of continuous life: An advanced version of the Conway's Game of Life

Baptiste Pignier

February 5, 2025

# Contents

<b>1</b>	<b>Reference Material</b>	<b>1</b>
1.1	Table of Units . . . . .	1
1.2	Table of Symbols . . . . .	1
1.3	Abbreviations and Acronyms . . . . .	1
<b>2</b>	<b>Introduction</b>	<b>1</b>
2.1	Purpose of Document . . . . .	1
2.2	Scope of Requirements . . . . .	2
2.3	Characteristics of Intended Reader . . . . .	2
2.4	Organization of Document . . . . .	2
<b>3</b>	<b>General System Description</b>	<b>2</b>
3.1	System Context . . . . .	2
3.2	User Characteristics . . . . .	3
3.3	System Constraints . . . . .	3
<b>4</b>	<b>Specific System Description</b>	<b>3</b>
4.1	Problem Description . . . . .	3
4.1.1	Terminology and Definitions . . . . .	3
4.1.2	Physical System Description . . . . .	4
4.1.3	Goal Statements . . . . .	4
4.2	Solution Characteristics Specification . . . . .	4
4.2.1	Assumptions . . . . .	5
4.2.2	Theoretical Models . . . . .	5
4.2.3	General Definitions . . . . .	5
4.2.4	Data Definitions . . . . .	6
4.2.5	Instance Models . . . . .	6
4.2.6	Input Data Constraints . . . . .	7
4.2.7	Properties of a Correct Solution . . . . .	7
<b>5</b>	<b>Requirements</b>	<b>8</b>
5.1	Functional Requirements . . . . .	8
5.2	Nonfunctional Requirements . . . . .	8
5.3	Rationale . . . . .	8
<b>6</b>	<b>Likely Changes</b>	<b>8</b>
<b>7</b>	<b>Unlikely Changes</b>	<b>9</b>
<b>8</b>	<b>Traceability Matrices and Graphs</b>	<b>9</b>



## Revision History

Date	Version	Notes
01/02/2025	1.0	First release

# 1 Reference Material

This section records information for easy reference.

## 1.1 Table of Units

No special units will be used

## 1.2 Table of Symbols

No special symbols will be used

## 1.3 Abbreviations and Acronyms

symbol	description
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
TM	Theoretical Model

# 2 Introduction

Conway's Game of Life is the result of applying simple rules of evolution. It allows the observation of an emerging phenomenon called cellular automaton. These cellular automata can be made more complex by making the rules of evolution more complex to observe more complex structure.

This section will develop the characteristics of the SRS document, more than those of the project itself.

## 2.1 Purpose of Document

The purpose of this Software Requirements Specification (SRS) document is to define the functional and non-functional requirements of the project in a clear and structured manner.

This document serves as a communication tool between stakeholders, including clients, developers, and project managers, to ensure a shared understanding of the project's scope and objectives.

## **2.2 Scope of Requirements**

This project aims to develop a simulation in a simplistic artificial environment. As such, the simulated elements will not be governed by any other physical law than those implemented by the environment. No assumptions will be made compared to a classic physical simulation model. The presentation of assumptions is therefore not applicable to this project.

## **2.3 Characteristics of Intended Reader**

Reviewers of this document should have undergraduate mathematical knowledge as well as a good understanding of the physical approximations that will be implemented. He will also need to be aware of the different libraries used for this project.

## **2.4 Organization of Document**

This document is structured as follows. Section 3 covers general information about the project. Section 4 explains this information in detail. Section 5 covers the project requirements. Section 6 lists the likely changes and Section 7 lists the unlikely changes. Section 8 presents the requirements traceability. Section 9 presents the project development plan.

# **3 General System Description**

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics and lists the system constraints.

## **3.1 System Context**

- User Responsibilities:
  - The user will have no responsibility for the data he transmits to the system since they will be included in a range of values that the system will provide him.
- Game of continuous life Responsibilities:
  - Provide the user with a simulation that faithfully matches the data the user has given to the system.

This software is intended for educational and/or artistic use. As such, its components must not be used for uses requiring security or critical requirements.

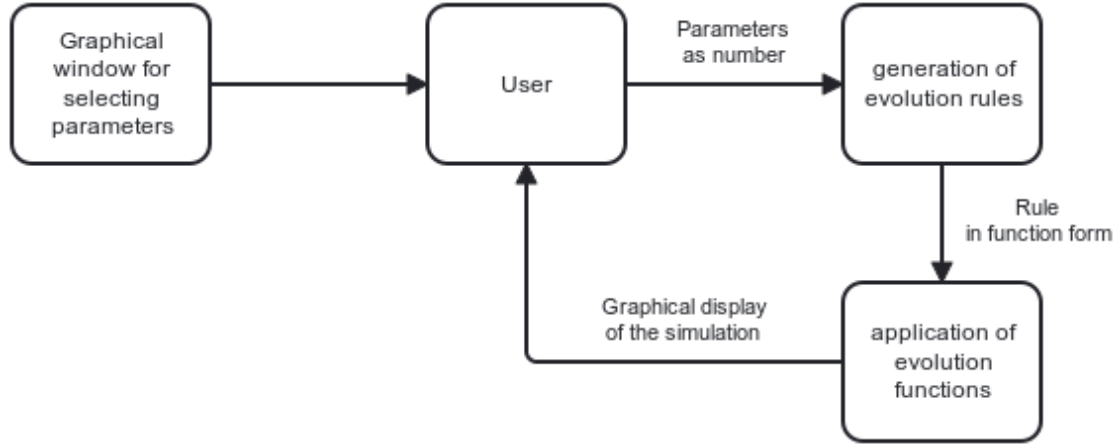


Figure 1: System Context

### 3.2 User Characteristics

The user does not need any special knowledge to use the program. All useful information will be provided by the graphical window for selecting parameters.

### 3.3 System Constraints

No system constraints are identified for this project, as long as its requirements are met.

## 4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models.

### 4.1 Problem Description

Game of continuous life aims to demonstrate the emergence of macroscopic behavior and structure from microscopic generation and evolution rules.

#### 4.1.1 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements:

- A cell : A portion of a display area that contains one or more values

- The grid : Grid cell size as small as possible. The dimensions of this grid are as large as desired.
- Neighborhood : Area around a cell, consisting of the cells in that area. This area is delimited by a neighborhood function
- Neighborhood function: Function that takes a cell as input and returns the cells in its neighborhood.
- Growth function: Function that takes a cell as input and modifies the value(s) contained in the cell based on its neighborhood.
- Structure: Finite set of cells in prolonged interaction

#### **4.1.2 Physical System Description**

The physical system of Game of continuous life, as shown in Figure ?, includes the following elements:

PS1 : Cells

PS2 : Growth function

PS3 : Neighborhood function

#### **4.1.3 Goal Statements**

Given the input parameters, the goal statements are:

GS1: Create the Growth function

GS2: Create the Neighborhood function

GS3: Apply iterative evolution

GS4: Define interesting evolution

GS5: Find some interesting evolution

## **4.2 Solution Characteristics Specification**

The instance models that govern Game of continuous life are presented in Subsection [4.2.5](#). The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.



### 4.2.1 Assumptions

Since the project does not aim to model a physical system by simplifying it, no assumptions of physical simplification will be made. Therefore, this section is not relevant.

### 4.2.2 Theoretical Models

No additional assumptions will be made to modify the equations used. As such, the models are described in section 4.2.5. This section is not relevant.

### 4.2.3 General Definitions

This section collects the laws and equations that will be used in building the instance models.

Number	GD1
Label	<b>Gauss function</b>
SI Units	N/A
Equation	$G(x) = A \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$
Description	This formula will be used for applying the growth and neighborhood function on a cell.
Source	<a href="https://en.wikipedia.org/wiki/Gaussian_function">https://en.wikipedia.org/wiki/Gaussian_function</a>
Ref. By	None

Number	GD2
Label	<b>Laplacian Operator</b>
SI Units	N/A
Equation	$\nabla^2 u_{i,j} = u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}$
Description	This formula is the discrete approximation of the second derivative. This model models the interactions between two concentration values. It will be used only if the simulation includes multiple channels.
Source	<a href="https://en.wikipedia.org/wiki/Finite_difference#Multivariate_finite_differences">https://en.wikipedia.org/wiki/Finite_difference#Multivariate_finite_differences</a>
Ref. By	None

Number	GD3
Label	<b>Reaction-Diffusion equation</b>
SI Units	N/A
Equation	$\frac{\partial u}{\partial t} = D_u \nabla^2 u + f(u, v)$
Description	This model models the interactions between two concentration values. It will be used only if the simulation includes multiple channels.
Source	<a href="https://www.karlsims.com/rd.html">https://www.karlsims.com/rd.html</a>
Ref. By	None

#### 4.2.4 Data Definitions

Number	DD1
Label	<b>Concentration</b>
Symbol	$u_t$
SI Units	
Description	The concentration of species $u$ in a given cell, at a given time $t$ .
Sources	
Ref. By	IM2, IM1

#### 4.2.5 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.2.4 to replace the abstract symbols in the models identified in Sections 4.2.2 and 4.2.3. The goal GS1 are solved by IM2.

Number	IM1
Label	<b>Reaction Diffusion</b>
Input	$D_u, D_v, u_t, v_t, F, k$
Output	$u_{t+1} = D_u \nabla^2 u_t - u_t \cdot v_t^2 + F(1 - u_t)$ $v_{t+1} = D_v \nabla^2 v_t + u_t \cdot v_t^2 - (F + k)v_t$
Description	<p><math>D_u</math> and <math>D_v</math> are the diffusion rate for <math>u</math> and <math>v</math>.</p> <p><math>u_t</math> and <math>v_t</math> are the concentrations of elements <math>u</math> and <math>v</math> at time <math>t</math></p> <p>We will use the Gray-Scott model to implement the diffusion reaction.</p>
Sources	<a href="https://www.karlsims.com/rd.html">https://www.karlsims.com/rd.html</a>
Ref. By	None

Number	IM2
Label	<b>Growth function</b>
Input	$u_t, \alpha, \beta$
Output	$u_{t+1} = -1 + 2 \exp \left( -\left( \frac{u_t - \alpha}{\beta} \right)^2 \right)$
Description	We'll use a Gaussian function to start. We want an amplitude between -1 and 1 to simplify the calculations.
Sources	None
Ref. By	None

#### 4.2.6 Input Data Constraints

Constraints on the input data will be set by the program itself, via a selection interface. The values of its constraints will be determined in a future revision of this document.

#### 4.2.7 Properties of a Correct Solution

This program does not provide a "solution", but a visualization of the events induced by the parameter choices made by the user. This section is not relevant.

## 5 Requirements

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

### 5.1 Functional Requirements

- R1: Input data will be given by the user before the simulation starts and must be modifiable during the simulation.
- R2: The parameter selection interface allows for understanding by uninitiated users
- R3: The program must calculate step by step the results of applying the growth function to each cell.
- R4: The output should allow the user to correctly distinguish between concentrations of value.

### 5.2 Nonfunctional Requirements

- NFR1: **Accuracy** The accuracy of the computed solutions should meet the level needed for correct visualization
- NFR2: **Usability** This software must be able to be used without any special skills or knowledge.
- NFR3: **Maintainability** The effort required to make any of the likely changes listed for Game of continuous life should be less than  $\epsilon$  of the original development time.
- NFR4: **Portability** This program should run on all operating systems supported by the libraries used.

### 5.3 Rationale

It is clear that significant mathematical and physics skills are required to simulate a biologically accurate cellular automaton development environment. Not having such skills, I limit myself to a basic but correct and useful implementation of a cellular automaton simulation.

## 6 Likely Changes

- LC1: Implementing multiple value channels
- LC2: Stochastic equation implementation for the evolution rules

## 7 Unlikely Changes

None

## 8 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an “X” may have to be modified as well. Table 1 shows the dependencies of general definitions and instance models with each other. Table 2 shows the dependencies of instance models and requirements on each other.

	GD1	GD2
IM2	X	
IM1		X

Table 1: Traceability Matrix Showing the Connections Between Items of Different Sections

	IM2	IM1
R3	X	X

Table 2: Traceability Matrix Showing the Connections Between Requirements and Instance Models

## 9 Values of Auxiliary Constants

Auxiliary Constants	Values
$\epsilon$	$\frac{1}{20}$