Covid-19 Automata Simulator

Project Proposal & Plan

1. Introduction

In the last few years, COVID19 has affected the lives of people across the world. Our goal with this project is to provide visualization for people to further understand the consequences of the deadly pandemic. We hope to use this as a visual and educational representation simulating the spread of the virus and its effect on the general population. We want to do this through a cellular automata model but plan on extending the model to be more realistic and representative of the real world. We hope our project can be used as a tool that positively impacts the world, and the outlook on the pandemic.

1.1 Project Overview and Statement of Proposal

The project will consist of several key parts. A User Interface (UI) for the user to interact with the simulation, such as to set initial conditions, configure parameters, design environments, etc. The simulation itself will use cellular automata to model the spread of a virus and may be extended to a larger, more comprehensive geospatial model as a tool for international/worldwide epidemic study if time allows.

Several of the parameters that might be included for users to modify are vaccination rate, initial vaccinated population, viral mutations/variants, infection mitigation measures (mask usage, social distancing, etc.), and other potential factors not explicitly enumerated here as progress is made on the software. Some data visualization would also be provided, such as SIR curves, approximations for infectivity rate (R_0), and rolling graphs of the infected population.

In order to accomplish these design goals, we intend to adopt an agile development process. This is so we can rapidly produce functional prototypes that will gradually have features added or allow for quick response to bugs or unforeseen circumstances during the development process. We intend to maintain a flexible Gantt chart with hard deadlines for deliverables, but with room for additional tasks to be added or existing tasks to be extended as necessary.

> **Statement of Proposal:** *We propose to develop a software application to simulate viral spread in a sample population across various environments. The proposed software will utilize cellular automata and allow for user configurable parameters to model different cases and provide the user with data and visualizations.*

1.2 Project Scope and Objectives

- Provide a robust UI where the user can…
    - Configure settings related to infection rate, virality, population, and simulation environment
- Simulate the spread of COVID19 through a populated environment with a cellular automata model, where spread is influenced by the various factors mentioned above
- Output data for the results of each simulation and display it to the user in a readable format

2. Risk Management Strategy

2.1 Risk Table

Risk	Category	Probability	Impact	RMMM
Project is too large to be completed in time	PS	1	4	4
Project fails to satisfy scope of requirements	PS	1	4	4
Group fails to effectively collaborate	PR	2	3	6
Group member leaves	ST	0	4	0
Group member lacks necessary skills/experience	ST	1	2	2
Project design process is insufficient	PR	2	2	4
An ineffective development environment is chosen	DE	2	3	6

Category values:

PS – Product Size Risk

BU – Business Impact Risk

CU – Customer Relations Risk

PR – Process Risk

TE – Technology Risk

DE – Development Environment Risk

ST – Risk Associated with Staff Size and Experience

Impact values:

4 – catastrophic

3 – critical

2 – marginal

1 – negligible

Probability Values:

0 – No risk

1 – Low risk

2 – Medium risk

3 – Medium-High risk

4 – High risk

2.2 Discussion of Risks to Be Managed

For this project, the main risk factors stem from the relative inexperience of the team members. As the team is composed of college students with little to no actual experience with software design, there are certainly going to be issues regarding planning. There may also be some issues that arise from lack of programming skill, but this is less likely to happen. Although the team members have limited experience working collaboratively on a large project, all have plenty of experience with programming. The final major source of risk is in the selection of a development environment, as the limited experience of the group could lead to the selection of a language that lacks some of the features the project will require.

As there is no actual client or business involved in the development of this project, risks normally related to business impact and customer relations are not an issue. Technology risks are also not likely to be prevalent, simply because the size of the project should not cause computational power to become a limiting factor. Overall, the main risks for this project will stem from the inexperience of the group, rather than issues external to the project.

2.3 Risk Mitigation, Monitoring, and Management Plan

2.3.1 Risk Mitigation

- *Project is too large to be completed in time*
  Starting with a conservative scope should ensure that the initial requirements of the project can be completed in time.
- *Project fails to satisfy scope of requirements*
  Approval of the project by the professor should assure that the project is large enough to satisfy the requirements.
- *Group fails to effectively collaborate*
  In order to prevent poor collaboration, frequent meetings and consistent communication will be maintained throughout the duration of the project.
- *Group member leaves*
  Meetings will assure that all group members are happy with the direction of the project, so that they do not want to leave. All group members are also hard workers, so the likelihood of them dropping out of the class is minimal.
- *Group member lacks necessary skills/experience*
  Assignment of responsibilities will take strengths and weaknesses of individual group members into account. Training time will be included to mend the gaps in skills/experience per project member.
- *Project design process is insufficient*
  An agile planning approach will be used, as this strategy is likely better suited for groups with less experience in software design.
- *An ineffective development environment is chosen*
  Care will be taken when selecting an environment to assure that it can meet all of the proposed requirements for the project.

2.3.2 Risk Monitoring

- *Project is too large to be completed in time* and *Project fails to satisfy scope of requirements*
  Regular meetings will be held to monitor the project progress, so that a slow pace can be noted as early as possible.
- *Group fails to effectively collaborate*
  Discussions will be held openly with the whole group to ensure that it is clear if anyone is not on the same page. All members are expected to communicate issues to be resolved as a group.
- *Group member leaves*
  As a courtesy, group members will let people know if they are considering leaving the project for any reason. All members are expected to communicate issues to be resolved as a group and satisfaction will be checked at each meeting.
- *Group member lacks necessary skills/experience*
  Group members will be given the opportunity to discuss difficulties they are having with the project at regular meetings.
- *Project design process is insufficient*
  Part of the regular meetings will revolve around reviewing the current design, and seeing if it seems sufficient for the project.

- *An ineffective development environment is chosen*
  The effectiveness of the development environment will also be reviewed at meetings.

2.3.3 Risk Management (Contingency Plans)

- *Project is too large to be completed in time*
  Non-essential aspects of the project will be cut to save time.
- *Project fails to satisfy scope of requirements*
  Additional features and complications will be added to the plan to ensure that it meets the requirements.
- *Group fails to effectively collaborate*
  An emergency meeting will be held to get everybody back on track and revise any design differences.
- *Group member leaves*
  The responsibilities previously left to that member will be split as evenly as possible amongst the group.
- *Group member lacks necessary skills/experience*
  Tasks and sub-teams will be reassigned as necessary if members have substantial difficulty with their assignment.
- *Project design process is insufficient*
  The project goals will be re-analyzed, and a new development plan will be made based on the situation.
- *An ineffective development environment is chosen*
  Alternative solutions will be explored, and existing/former software deployments maintained until a transition to a new environment is made.

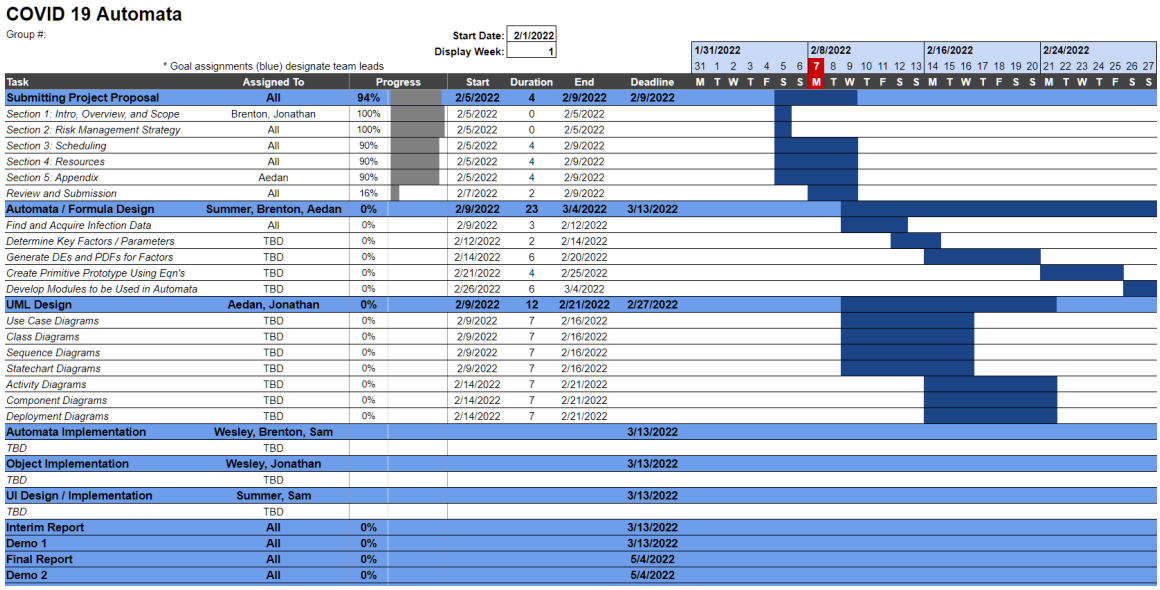
3. Schedule

3.1 Task List

1. Requirement Analysis
2. System Design
3. Design what and how factors should influence virus spread (automata rules)
4. Design UML diagrams for objects and how they relate
5. Implement automata, including grid visualization, and rules for infection
6. Implement object designs
7. UI Design
8. Integration
9. Testing

3.2 Timeline Chart

A responsive Gantt Chart is maintained at a Google Sheets link. Below is a sample screenshot to demonstrate its functions:



COVID 19 Automata

Group #:

Start Date: 2/1/2022

Display Week: 1

* Goal assignments (blue) designate team leads

Task	Assigned To	Progress	Start	Duration	End	Deadline
Submitting Project Proposal	**All**	**94%**	**2/5/2022**	**4**	**2/9/2022**	**2/9/2022**
Section 1: Intro, Overview, and Scope	Brenton, Jonathan	100%	2/5/2022	0	2/5/2022	
Section 2: Risk Management Strategy	All	100%	2/5/2022	0	2/5/2022	
Section 3: Scheduling	All	90%	2/5/2022	4	2/9/2022	
Section 4: Resources	All	90%	2/5/2022	4	2/9/2022	
Section 5: Appendix	Aedan	90%	2/5/2022	4	2/9/2022	
Review and Submission	All	16%	2/7/2022	2	2/9/2022	
Automata / Formula Design	**Summer, Brenton, Aedan**	**0%**	**2/9/2022**	**23**	**3/4/2022**	**3/13/2022**
Find and Acquire Infection Data	All	0%	2/9/2022	3	2/12/2022	
Determine Key Factors / Parameters	TBD	0%	2/12/2022	2	2/14/2022	
Generate DEs and PDFs for Factors	TBD	0%	2/14/2022	6	2/20/2022	
Create Primitive Prototype Using Eqn's	TBD	0%	2/21/2022	4	2/25/2022	
Develop Modules to be Used in Automata	TBD	0%	2/26/2022	6	3/4/2022	
UML Design	**Aedan, Jonathan**	**0%**	**2/9/2022**	**12**	**2/21/2022**	**2/27/2022**
Use Case Diagrams	TBD	0%	2/9/2022	7	2/16/2022	
Class Diagrams	TBD	0%	2/9/2022	7	2/16/2022	
Sequence Diagrams	TBD	0%	2/9/2022	7	2/16/2022	
Statechart Diagrams	TBD	0%	2/9/2022	7	2/16/2022	
Activity Diagrams	TBD	0%	2/14/2022	7	2/21/2022	
Component Diagrams	TBD	0%	2/14/2022	7	2/21/2022	
Deployment Diagrams	TBD	0%	2/14/2022	7	2/21/2022	
Automata Implementation	**Wesley, Brenton, Sam**					**3/13/2022**
TBD	TBD					
Object Implementation	**Wesley, Jonathan**					**3/13/2022**
TBD	TBD					
UI Design / Implementation	**Summer, Sam**					**3/13/2022**
TBD	TBD					
Interim Report	**All**	**0%**				**3/13/2022**
Demo 1	**All**	**0%**				**3/13/2022**
Final Report	**All**	**0%**				**5/4/2022**
Demo 2	**All**	**0%**				**5/4/2022**

Link to access our Gantt chart is available in the appendix.

3.3 Resource Table

Task	People	Hardware & Software	Special
Requirement Analysis	All Members	N/A	N/A
System Design	Aedan, Jonathan,Wesley	IntelliJ	N/A
Design of cellular automata and formulas	Brenton, Aedan, Summer	N/A	Data from the CDC and other reputable disease research agencies
Design UML diagrams	Aedan, Jonathan	Netbeans/Eclipse	N/A
Implement automata, including grid visualization, and rules for infection	Wesley, Sam, Brenton	IntelliJ	N/A
Implement object designs	Jonathan, Wesley	IntelliJ	N/A
UI Design	Summer, Sam	Netbeans	N/A
Testing	All Members	IntelliJ	N/A

4. Project Resources

4.1 People

1. Summer Bronson

2. Wesley Camphouse
3. Brenton Candelaria
4. Jonathan Carsten
5. Samuel Nix
6. Aedan Wells

4.2 Hardware and Software Resources

Our project will not require any specific hardware, however, there will be multiple software elements in place to best facilitate production. Java will be our chosen programming language, so an IDE like IntelliJ and Netbeans will be used for system coding, UI, and UML design and creation. Github will be used for version control and Discord/email will be used for team communication. Members' personal computers should suffice for development and deployment when it comes to hardware.

4.3 Special Resources

Simulation software based on Covid-19 spread, especially with the different attributes of different variants, requires data from the CDC and other reputable disease research agencies to make the simulations as accurate as possible. This information will be crucial towards providing the best simulations.

5. Appendices

For this report, all team members contributed equally.

2/1/22 meeting:

The group met to discuss project ideas developed from the concept exploration assignment. Each group member proposed their main idea and a secondary idea for our software engineering. We discussed our top choices and secondary choices, settling on a Covid19 simulation/visualization software with a Vitamin K tracker as a backup choice. Each member was assigned tasks to explore methods of setting up and executing the project before we met to write the proposal. A google drive, discord server, and github repository page was created for this project.

2/5/22 meeting:

The group met to write a rough draft of the project proposal. Each group member took a section of the report and discussed our execution plan. A summary was set from the concept exploration page, however, all other elements were developed individually since the previous meeting. Risk was a debated topic, comparing an industry quality risk to our school project was challenging but worthwhile in the end.

We have decided to implement an Agile development cycle for our software product as the simple prototypes should be easy to construct and improvements will be steady. Story cards will help us improve the prototype to be a reliable product. To determine the best possible times for meeting, a google calendar was created with everyone's classes and meetings.

The gantt chart is available for review at the following link:

https://docs.google.com/spreadsheets/d/1K6KE9ma6MrcFe8U3UUh_3Kht0pm6ZRa3SO_QFqx9YR8/edit?usp=sharing