**Software Requirement Specification**

for

**Neural Evolution and ML Implementation in Application**

*Jan-May 2022*

Prepared by

|  |  |  |
| --- | --- | --- |
| **Specialization** | **SAP ID** | **Name** |
| CCVT | 500077169 | Yash Grover |
| AIML | 500075430 | Abhishant Singh |
| AIML | 500076108 | Akash Jaiswal |
| CCVT | 500077195 | Shivani Srivastava |

**Mentor**

Ms Richa Choudhary

Assistant Professor

Department of Computer Science



Department of Virtualization

School Of Computer Science

UNIVERSITY OF PETROLEUM & ENERGY STUDIES,

DEHRADUN- 248007. Uttarakhand

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Topic** | | **Page No** |
| Table of Content | | 2 |
| Revision History | | 3 |
| 1 | Introduction | 3 |
|  | 1.1 Purpose of the Project | 4 |
|  | 1.2 Target Beneficiary | 4 |
|  | 1.3 Project Scope | 5 |
|  | 1.4 References | 5 |
| 2 | Project Description | 6 |
|  | 2.1 Reference Algorithm | 6 |
|  | 2.2 Data/ Data structure | 7 |
|  | 2.3 SWOT Analysis | 9 |
|  | 2.4 Project Features | 10 |
|  | 2.5 User Classes and Characteristics | 11 |
|  | 2.6 Design and Implementation Constraints | 11 |
|  | 2.7 Design diagrams | 12 |
| 3 | System Requirements | 12 |
|  | 3.1 User Interface | 13 |
|  | 3.2 Software Interface | 13 |
|  | 3.3 Database Interface | 13 |
|  | 3.4 Protocols | 14 |
| 4 | Non-functional Requirements | 14 |
|  | 4.1 Performance requirements | 15 |
|  | 4.2 Security requirements | 15 |
|  | 4.3 Software Quality Attributes | 16 |
| 5 | Other Requirements | 16 |
| Analysis Model | | 17 |
| Issues List | | 17 |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Change** | **Reason for Changes** | **Mentor Signature** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* INTRODUCTION

1.1 Purpose of the Project

To demonstrate the implementation of **Reinforced Machine Learning** using evolutional neural networks in an interactive application. Additionally, the project will also have a functional model of facial recognition model.

1.2 Target Beneficiary

Although holding a lot of potential, **Reinforced Machine Learning** is not so frequently implemented and utilized in the current scenario of machine learning. Currently, the convenient and most used options are supervised and unsupervised machine learning. The **Reinforced Machine Learning** will be a revolution in the world of machine learning.

This would be a boon for the data scientists, machine learning enthusiasts and modellers.

1.3 Project Scope

The objective here is to acknowledge the immense potential Reinforced Machine Learning Algorithm has and implement it along with Neuro Evolution of Augmenting Topologies (NEAT) library in application. Studying its advantages  over other types of neural networks along with its implementation in an app​

1.4 References

* <https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78>
* <https://pytorch.org/>
* <https://techitsmart.ca/2020/05/18/python-deep-learning-for-flappy-bird-game/>​
* <https://towardsdatascience.com/neural-network-plays-flappy-bird-e585b1e49d97>
* PROJECT DESCRIPTION

2.1 Reference Algorithm

Neuro Evolution Neural Network – NEAT(Neuro Evolution with Augmented Topologies)

2.3 SWOT Analysis

Strength-

1. Self-learning​
2. Amends topology if feed-forward neural network​
3. 50 genomes released at time​
4. 100% efficiency achieved in less than 5 generations​
5. Up to 20 faces data stored​
6. (optional pixelated face conversion)

Weaknesses-

a. Different (depends on type) genome mutation techniques can result in increased computation

Opportunities-

a. Learning about different algorithms.

b. Neural Network implementation in working application

c. TensorFlow learning for face recognition

Threats-​

1. Based on deployment

2.4 Project Features:-

Neural Networks Visual Implementation Demonstration

Facial Recognition Demonstration

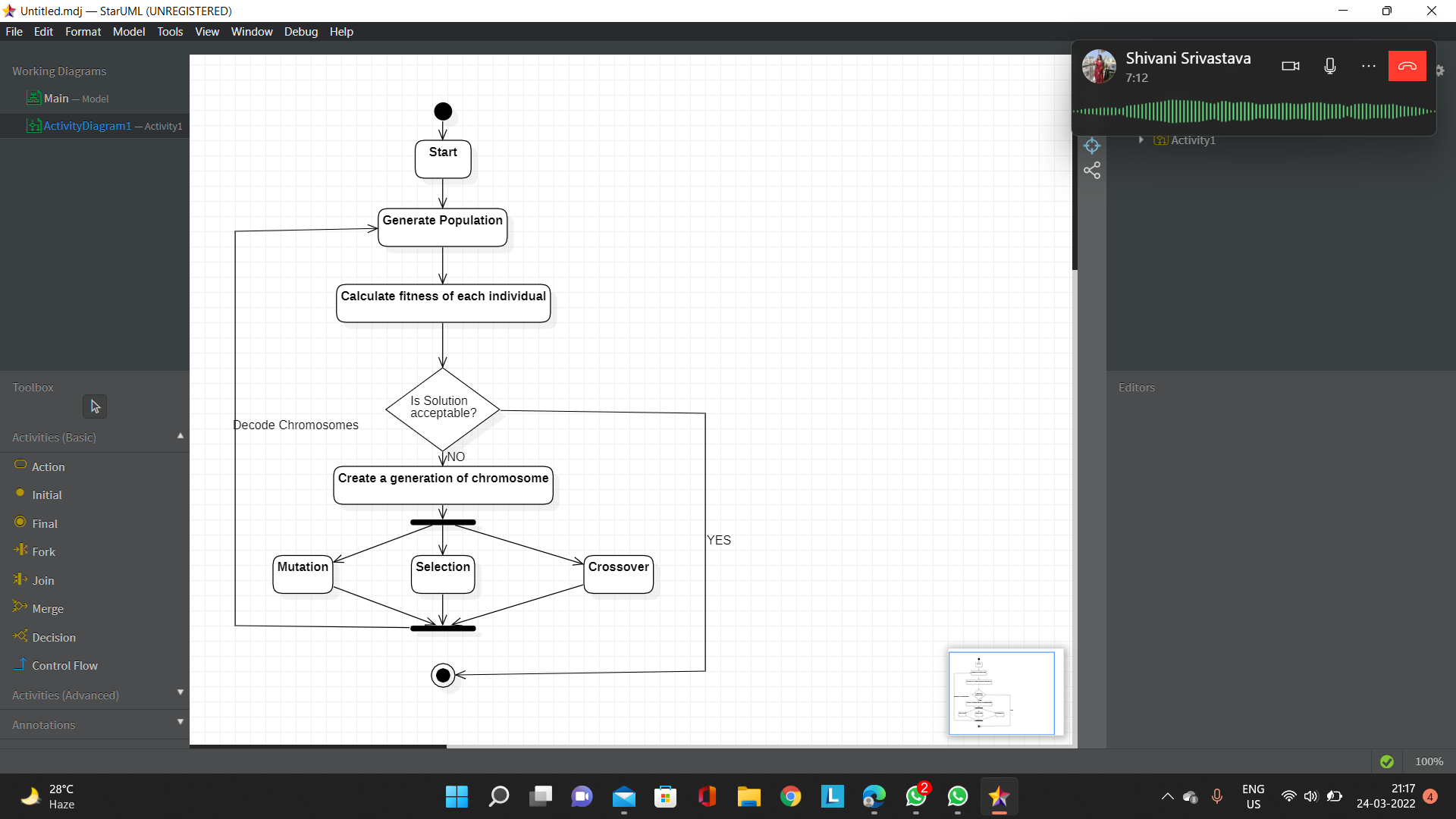
Reinforced Machine Learning

2.5 User Classes and Characteristics:-

* Manufacturing is all about producing goods that can satisfy our basic needs and essential wants. Robot Manufacturers or Manufacturers of Collaborative Robots that can perform various manufacturing tasks with a workforce of more than 100 people,
* ML models like Deep Neural Networks (whose framework is Reinforcement Learning) can be leveraged for simplifying this trending image processing method,
* With games optimization through Reinforcement Learning algorithms, we may expect better performances of our favorite games related to adventure, action, or mystery and many more

2.7 Design diagrams:-

Activity Diagram:



## USER REQUIREMENTS

## 3.1 USER INTERFACE

## We are using Microsoft Windows Operating System, various IDEs like Py charm, anaconda,

## Visual studio code, google Colab. So, it has various GUI functions like syntax highlighting, scroll features, graphical windows etc. These GUI applications provide interactive menus, toolbars, buttons, panes, containers, grids allowing easy controls with the help of various input devices like mouse and keyboard.

## 3.1 SOFTWARE INTERFACE

## Software interfaces (programming interfaces) are the languages, codes and messages that programs use to communicate with each other and to the hardware. Examples are the Microsoft Windows Operating Systems. In this project we are using PYTHON as programming language.

## 3.2 PROTOCOLS

## No such communication, security or encryption issues.

## NON-FUNCTIONAL REQUIREMENTS

## 4.2 SECURITY REQUIREMENTS

## Physical access to the system and the application on it will be limited to the user. The system has no extra security features.

## 4.3 SOFTWARE QUALITY ATTRIBUTES

## RELIABILITY- The mathematical model will dependably produce the proper results. Model’s reliability is determined by how well a project performs in various working environments and conditions.

## MAINTAINABILITY - It is easy to add code to the existing mathematical system, the program is easy to upgrade for new features from time to time.

## USABILITY – The mathematical model is quite easy to use and navigation is quite simple. The analysis program is easy to learn and user friendly.

## CORRECTNESS - The application's functionality, internal calculations, and navigation should all be proper.

## TESTABILITY – It is easy to test and find defects in the system. If required, the system will be easy to divide into smaller modules for the purpose of testing.

## REUSEABILITY – The model can be reused with other states/places if required by changing some of the values and collecting appropriate data.

## INTEROPERABILITY - It should be simple for a product to communicate data or services with other systems. Different system modules should be compatible with various operating systems (MacOS, Windows, Linux) & IDEs.