**Requirement Analysis Report**

**1. Introduction**

**Project Title:**

BIRD LEARNING – Using open source python libraries and learning algorithms for machine learning

**Project Definition:**

In this project, we are coding a flappy bird game that learns how to play. The bird which in the game is learning how to cross the barriers and it tries to be alive.  
We are using deep learning algorithm via python language. We will use pyGame, TenserFlow libraries. We decided to use PyCharm IDE because it`s intellisense. We decided to use Waterfall V-Model. Because, this project is close to change.

**Project Goal:**

Coding a flappy bird bot which learns playing by itself. Firstly, this bird didn’t know how to play this game. Then, we will write some codes with using some libraries and learning algorithms. At the end of project, this bird must learn to play Flappy Bird without players. So, this project’s goal is a teaching the bird how to learn.

**Project Scope:**

It’s an example of artificial intelligence, sometimes called machine intelligence, is intelligence demonstrated by machines, in contrast to the natural intelligence displayed by humans. The term artificial intelligence is a very comprehensive subject. Artificial intelligence has been a growing resource for video games for years now. Most video games — whether they’re racing games, shooting games, or strategy games — have various elements that are controlled by AI, such as the enemy bots or neutral characters. Even the ambiguous characters that don’t seem to be doing much are programmed to add more depth to the game, and to give you clues about what your next steps should be. It can be integrated into many games. In any video game environment, AI can be programmed to enhance a player’s experience. Machine learning runs on the stores of data available to it and uses this information to create a world where characters can live and conduct basic actions. All the basic information collected through machine learning can then be used to create game environments and characters that appear more realistic and natural. There are lots of learning algorithms for these games. It learns by making mistakes.

**Application Areas:**

Machine learning has been applied various areas such as virtual personal assistants, predictions while commuting, videos surveillance, social media services, email spam and malware filtering, online customer support, Search Engine Result Refining, **product recommendations, online fraud detection.**AI has also been applied to video games, for example video game bots, which are designed to stand in as opponents where humans aren't available or desired. Additionally, it uses in game industries. Computer games can be used to give a better form of education and can even make computers become the unique tools of learning. The paper will present the current scenarios of computer games being used for the purposes of education and training.

**Background :**

In this Project, the main programming language is Python. Development environment is pyCharm IDE. When we are programming the game, we will use pyGame and TensorFlow libraries. When we apply the learning algorithms, we will use QLearning.

**2. Team Organization**

Front-end --> Hümeyra BİLGİ

Back-end --> Oğuzhan SOLAK

Machine Learning --> Gökçen TÜRKÖZ, Hasan DİNÇKURT, Yasin Tarık SUYABATMAZ

**3. Project Schedule**

Week-1

• Research a project topic

• Create a team web site

• Create a GitHub account

Week-2

• Decide the project

• Research and decide the programming language

• Setup the required IDE and libraries

• Update the web site and GitHub account

Week-3

• Research about deep learning algorithms

• Research about Python language

• Start to create the game

• Create a project schedule

• Update the web site and GitHub account

Week-4

• Progress on the game

• Update the web site and GitHub account

Week-5

• Finish the game

• Research about deep learning algorithms

• Update the web site and GitHub account

Week-6

• Decide and start to work about the deep learning algorithm

• Update the web site and GitHub account

Week-7

• Progress on the deep learning algorithm

• Update the web site and GitHub account

Week-8

• Finish the project

• Update the web site and GitHub account

Week-9

• Test the project

• Fix the bugs

• Update the web site and GitHub account

**4. Process Model**



In this Project, we decided to use Waterfall V-Model. Because, our project isn’t open to change so weekly meetings are not necessary and Agile models cause to lose time and money.

But Waterfall models is fit for us. In our Project, the most important part is testing. V-Model doesn’t go to linearly. We can test the project after every step.

In V model, based on the requirements the System test cases are prepared, and based on the HLD (High level document)the Integration Test cases are prepared, and based on the LLD(Low-level document)the Integration Test cases are prepared. And then the coding is done.

Once coding is completed, unit, integration and system testing happens in the sequence. In V-model, gives relationship between each development stages and Testing stages.

While a useful if simplistic model of system or software development, the traditional V model does not adequately capture development from the tester's point of view. This blog discusses three variations of the traditional V model of system/software development that make it more useful to testers, quality engineers, and other stakeholders interested in the use of testing as a verification and validation method.

**5. System Requirements**

During this initial phase, system requirements and analysis are performed to determine the feature set and needs of users. Just as with the same phase from the waterfall model or other similar methods, spending enough time and creating thorough user requirement documentation is critical during this phase, as it only occurs once.

**1)Functional Requirements**

The bird should play the game (Flappy Bird) itself. But the bird doesn’t know how to play this game before this project doesn`t have a learning algorithm. By using this algorithm, the bird will learn this game and it will play without losing.

**2)Non-functional Requirements**

--> Performance: It should be well because this project needs to higher system requirements. Such as strong RAM, processor. Otherwise, bird learns slowly.

--> Scalability: These requirements depend on bird's learning. Also, system performance effects it.

--> Security: This project doesn't need a high security because we don't need to keep any personal information.

--> Maintainability: There will no need to maintain because the project needs any change.

--> Usability: This project`s learning algorithm can use on other games, if learning algorithm to be revised.

--> Availability: The Q-Learning algorithm that we`ll use, is available on all gaming platforms.

**6. Use Cases**

**Use Case Name:** Bird Learning

**Actors:**

* System
* The Bird

**Pre-condition:**

* The bird will hit to a barrier or the floor.

**Post-conditions:**

* The System will get the data from last failed game.
* The system will rearrange the coordinates.
* The bird will fly.

**Normal Flow:**

1. The bird will start to fly.
2. The bird will hit to a barrier.
3. The game will be over.
4. The system will get the data from last failed game.
5. The system will rearrange the coordinates.
6. The bird will start to fly.
7. The use case returns to step 2 and continues until it.

**Alternate Flows:**

**2A1:** The algorithm won`t work correctly.

1. The bird will hit to the floor.
2. The use case returns to step 3 and continues until it.

**6A1:** The bird won`t hit anywhere.

1. The bird will fly without hitting till forever.

