
TCU Computer Science Dept.

**ReadySet Go
Vision**

Version 1.4

ReadySet Go	Version: 1.4
Vision	Date: 2/3/2021
0001	

Revision History

Date	Version	Description	Author
09/23/2020	1.0	All sections started	Ryan Moncrief
09/25/2020	1.1	Update Introduction	Kien Nguyen
9/26/2020	1.2	Finished Parts 4 and 6	Ryan Moncrief
10/1/2020	1.3	Updated all segments to reflect new project requirements	Ryan Moncrief
2/3/2021	1.4	Fixed various typos	Ryan Moncrief

ReadySet Go	Version: 1.4
Vision	Date: 2/3/2021
0001	

Table of Contents

1.	Introduction	4
	1.1 Background	4
	1.2 References	4
2.	Positioning	4
	2.1 Business Opportunity/Problem Statement	4
	2.2 Product Vision/Position Statement	4
3.	Stakeholder Profiles and User Descriptions	5
	3.1 Stakeholder Summary	5
	3.2 User Environment	5
	3.3 Summary of Key Stakeholder or User Needs	5
	3.4 Alternatives and Competition	6
4.	Product Overview	6
	4.1 Product Perspective	6
	4.2 Deployment Considerations	6
	4.3 Assumptions and Dependencies	6
5.	Product Features / Scope	7
6.	Other Product Requirements	7

ReadySet Go	Version: 1.4
Vision	Date: 2/3/2021
0001	

Vision (Small Project)

1. Introduction

The product aims to develop an educational Go platform that helps amateur Go players to learn the game via playing with Artificial Intelligence. The platform also helps researchers to run small scaled training on the Go engines.

1.1 Background

In March 2016, AlphaGo, a Go engine created by Google DeepMind, defeated the best world Go player then, Lee Sedol. This remarkable event was then followed by a research project at TCU led by Dr. Ze-Li Dou and Dr. Liran Ma, the clients of this product. During the research project, the team was able to reproduce some strong AIs with different levels of strength. The team discovered that sometimes the AI was able to produce moves that were not in the line of human's intuition but were still optimal. With such findings and Dr. Dou's passion to teach Go, an extremely complex board game, to other people, the clients decided to build an educational Go platform that targeted students at TCU and amateur Go players to learn from AI. Besides, the new product will provide the clients, or other Go researchers, a way to easily interact with the AI training process.

1.2 References

(Leave this for later when the document is done)

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; Dieleman, S.; Grewe, D.; Nham, J.; Kalchbrenner, N.; Sutskever, I.; Lillicrap, T.; Leach, M.; Kavukcuoglu, K.; Graepel, T.; and Hassabis, D. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529: 484–503. (AlphaGo paper)

2. Positioning

2.1 Business Opportunity/Problem Statement

The problem of	There is no user-friendly interface to play Go against AI developed by this research project.
affects	Go AI researchers and Go players
the impact of which is	Research with Go AI's is tedious and time consuming.
a successful solution would be	an web-based platform capable of interacting with and training Go AI's.

2.2 Product Vision/Position Statement

For	Go Researchers
Who	Need to train and interface with Go AI's
The (product name)	online Go game GUI
That	trains Go AI's and allows users to play Go
Unlike	OGS and Pandanet
Our product	Train Go AI and generate reports Play versus AI Rank players and track progress

ReadySet Go	Version: 1.4
Vision	Date: 2/3/2021
0001	

3. Stakeholder Profiles and User Descriptions

3.1 Stakeholder Summary

Name	Description	Major value or benefit from this product	Major features of interest	Constraints that must be accommodated	Direct User or not?
Dr. Liran Ma	Administrator	Improved Productivity, Improved Usability,	UI for training AI		Yes
Go Players	Individual wanting to play Go	Improved Usability	UI for playing Go as well as progress tracking	User can play with AI	Yes
Dr. Ze-Li Dou	Administrator	Improved Productivity, Improved Usability,	UI for training AI	Facilitate setting parameters for AI training	Yes

3.2 User Environment

Currently, the AlphaGo engine is operated using a command line interface. Individuals can play Go with AI's but doing so is very complicated through the command line. The administrators can train the Go AI's using a string of commands as well as parameters. This is very inefficient and very challenging for Dr. Ze-Li Dou as well as any non tech-savvy individuals. Our product intends to simplify the process of training AI's as well as implement new features using a web based application. There will be an interface that is very easy to use in order to train the AI's as well as interact with the AlphaGo program. The application will also allow individuals to play Go with AI. The application will be accessible through all platforms using any web browser. There can only be 20-50 players concurrently using the application.

3.3 Summary of Key Stakeholder or User Needs

Need	Priority	Concerns	Current Solution	Proposed Solutions
I need to train AI's	High	current solution requires a lot of time and is complicated	A long string of commands and parameters is used on the command line	An interface that is easy to use that can execute commands and set parameters

ReadySet Go	Version: 1.4
Vision	Date: 2/3/2021
0001	

I need to play Go	High	current solution is very complicated	commands are entered on the command line	An graphical interface that makes playing Go easy and is visually appealing
I need to track my progress	High	No current solution exists	N/A	There will be user registration allowing progress to be tracked for each player

3.4 Alternatives and Competition

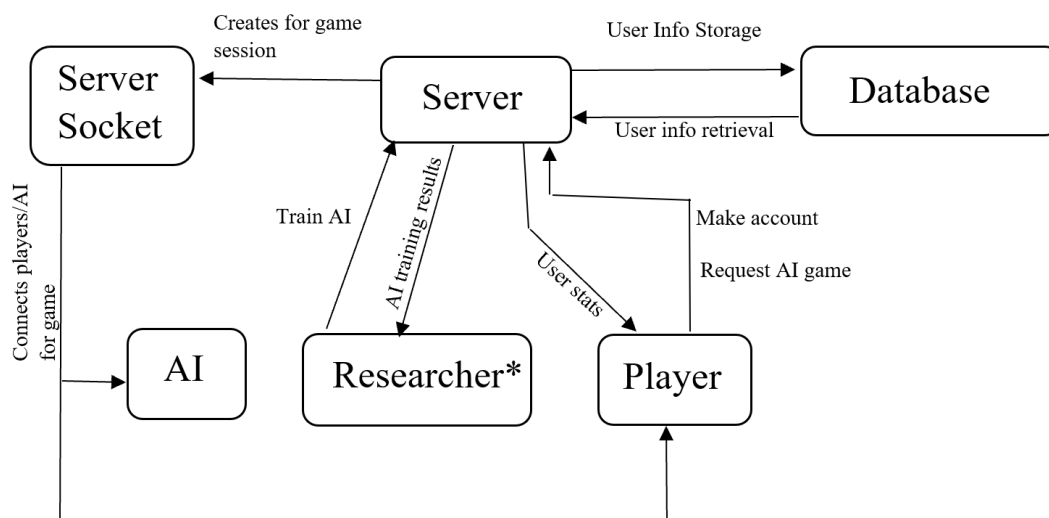
The command line interface is the current solution. Because direct commands are used, it is very easy to communicate with the server and Go engines.. The issue is that the user side is difficult as commands are very long and complicated.It is very time consuming and difficult to do by non tech-savvy individuals. Feedback also isn't very great as it's all delivered in command line or csv files which can be hard to understand. There also isn't any progress tracking as of now for players.

4. Product Overview

This section provides a high-level view of the AlphaGo web application.

4.1 Product Perspective

This project will mainly be concerned with providing an online application and interface for the existing AlphaGo system. The Go AI and testing commands already exist, this project serves as a way to make the system more accessible and user friendly for researchers. In addition, many non-researcher users(players) will be able to create profiles and keep track of their game statistics, as well as play with AI.



*Researchers will also have access to all features that players do

4.2 Deployment Considerations

Users will need an internet connection to access the product. Users can be either players or researchers, and

ReadySet Go	Version: 1.4
Vision	Date: 2/3/2021
0001	

researchers will be able to train AI with different parameters. The system should be able to add and remove many users and researchers. The team will be provided with an Ubuntu web server by the client. Our team is likely going to install Tomcat on that server to manage the web deployment. As for the project itself, the team is planning on using the Java based framework Spring. For the database, the team is planning on using MongoDB.

4.3 Assumptions and Dependencies

1. The user will need a stable, consistent internet connection
2. The project will use the already existing AlphaGo research project AI
3. The project will be hosted on an Ubuntu web server provided by the client
4. The project will have various internal dependencies based on the frameworks used (eg. Spring)

5. Product Features / Scope

Features will include:

- An easy to use GUI (Graphical User Interface) for researchers to interact with so they don't have to input commands manually
- Hardware statistics for researchers
 - For example: GPU/RAM usage so researchers can tell how much system resources they will have at their disposal
- Non-researcher users will be able to create profiles and play games against AI, and can keep track of their statistics.

Since the back-end AI work is already completed, the scope of the project is mostly in:

- Enhancing ease-of-use for researchers in the form of an easy to use front-end.
- Creating back-end features to store statistics that will be displayed in the front-end.

6. Other Product Requirements

1. Environmental- User must have access to stable internet connection- Priority: Critical
2. Performance- Server must be able to handle 20-50 sessions of gameplay concurrently- Priority: High
3. Platform- Frontend UI compatible with mobile devices as well as desktop browsers- Priority: Medium