Module 5- Computer Systems (2021-22)
Project

UNIVERSITY OF TWENTE.

Requirement Analysis Document Template

Project Name: ChessMate	**Team Members: Hein Huijskes, Julia van der Geest, Ujjwal Dodeja, Jose Gavilanes, Maouheb Bessi, Mengmeng Li**
Team ID: 17	**Mentor(s): Ramish Bhutto, Wenjie Zhao**

1. Introduction

ChessMate is a voice based game which allows a user to play board games using voice commands. A user can perform every task related to the game using their voice. For moving a player from one position to another, a user has to pronounce the coordinates of the present position followed by the coordinates of the destination. For example: to move a pawn by two steps, a user has to speak - E2, E4. Here, E2 is the current position of the pawn and E4 is the destination for the pawn.

There are existing applications which provide chess games through voice controls. The game keeps track of the scores, therefore we will implement an ELO rating with a leaderboard.

1.1. **Purpose**:

The purpose of this platform is to play games using voice commands. This way less capable people can also play games and enjoy them. We are aiming for a very user-friendly experience in which most commands can be given using a voice. The platform should recognize the voice, which is a security aspect of the project.

1.2. **Limitations of the current system(If any)**:

- The current system has a poor UI, which can be improved upon
- The current system is limited to only 1 game (chess)
- The current system doesn't provide are fair rating system and a leaderboard
- The current system has a very limited set of voice commands
- The current system has a lacking and buggy online multiplayer
- The current system has no security measures

1.3. **Intended Audience**

The users in our target audience are people who would like to play simple games, but either prefer to use their voice or more specifically cannot use their hands. For these people a voice controlled system would be ideal, since most things can be controlled using voice only. This also means that the system should work quickly and smoothly using voice, without the immediate need of manual interference.

The stakeholders are our TA's, our project group itself, and the university. Other parties may get access to the system in the future, but are not significant to the project's development as of writing this document.

1.4. **Define SMART Goals**:
The goals for the ChessMate project are as follows:

*1. To improve the **usability** of the system by having a user-friendly application.*	*To see if users interact with the system in the intended way.*	*To test the system with the improved application.*	*To enable users easy access to set up and if necessary reboot the system.*	*To finish the task between Week 8-Week 9.*
*2. To improve the **productivity** of the system by adding a microphone.*	*To see if the microphone records sound.*	*To test if the microphone correctly records sound.*	*To record voices of users to be parsed to usable text.*	*To finish the task between Week 4-Week 5.*
*3. To improve the **user experience** by having an application that looks clean, finished and well thought out.*	*To see if users can navigate through the app.*	*To test if potential users like the implementation.*	*To ease the use of the system and improve the experience.*	*To finish the task between Week 6-Week 7.*
*4. To improve the **security** of the system by using log-in for the application.*	*To see if a user can log-in.*	*To test log-in using real and fake users.*	*To protect the account and settings of the user from malicious parties.*	*To finish the task between Week 4-Week 5.*
*5. To improve the **security** of the database by making sure we hash the password and salt it as well.*	*To see if the passwords are hashed and salted*	*To test the quality of the hashed passwords. They should be unbreakable.*	*To be able to deny requests made to the database by unauthorized users.*	*To finish the task between Week 4-Week 5.*

1.5. **Scope:**

- System boundaries (Software and hardware):
  - Software: Python language, Mobile application, Voice recognition software, Voice parsing software, Log-in, (Cloud) Database
  - Hardware: Raspberry Pi 4, Microphone, Power supply, Cable access to internet or WiFi access, Monitor, Speaker
- Interfaces:
  - Internet via WiFi or cable
- Limitations:
  - This project is completed within 10 weeks
  - This project uses basic voice recognition libraries
  - This project will have a maximum of 2 functional games attached (unless a lot of extra time allows for more)

- This project allows only for simple online games to be played, that are set up per game. Other games are local.
- This project only records scores, ratings and user information on a database

2. **Product features:**

2.1 **Functional requirements:**

1. The system should be able to record voices using a microphone
2. The system should be able to relay messages using a speaker
3. The system should be able to host offline games
4. The system should be able to play co-op games.

2.2 **Nonfunctional requirements:**

5. The system should be able to connect to an online elo rating system for global leaderboards by sending match results to a server/database
6. The system should be able to convert voice commands into moves in the games
7. The system should include an extensive list of voice commands for the player to use to play the games
8. The system should include a nice interface on which to show leaderboards and games
9. The system should include multiple levels of AI difficulty to play against
10. The system should include secure player accounts protected by passwords or voice
11. The system should be able to undo moves (in case they were made maliciously by the opponent)

3. **Appropriate security controls**

1. The system should include secure player accounts protected by passwords or voice

We need voice recognition software, log-in, and a database to store the information. We need a microphone to execute these functionalities. We need authentication software and password management software.

The user will use voice recognition to log in to the application as well as to control the game.

Security Policy				
Security Requirements	**Security mechanisms(List down for your application)**	**Remarks on why you considered these requirements**	**Supplement requirements for your application(user story/abuse story)**	**Risk identification/Threat Assessment(At least one risk identification/abuse case)**
Authentication	checking passwords	User can use account to login the application	user story "as an User, I want to register and login the application." Abuse	The length of passwords is not very strong

			story :"as an attacker , I can enter the default passwords to access the application"	
Authorization	Access control role_based	The general users only play the game, Well the admin can manage the application	User story :"The system should be manually accessible via securely logging in to an application using an admin account." Abuse story: "general users change the set of application"	the admin forget his password
Audit	protection of database	to protect users information and account when attacker try to access database	User story :"As an user, I want to protect my data" Abuse story :" As an attacker, I want to steal users data "	SQL injection

4. **Conclusion:**

ChessMate is going to be a project intended for people who want to use voice control to control a game. The project is going to convert voice commands into moves in the game. We intend to make it as user-friendly as we can. The game that we will implement is Chess. We will add a leaderboard and a rating system. There will also be missions to complete in games that can earn the player additional rating.

5. **Reference**: List the existing literature (documents/articles/blogs/research papers) references you have considered for finalizing the project idea.

- Sanjeev, A. (2018, 23 march). The best voice recognition software out of three we tested, and how to set it up on Raspberry Pi. maker. https://maker.pro/raspberry-pi/tutorial/the-best-voice-recognition-software-for-raspberry-pi
- E. (2021, 9 februari). *Setup a Raspberry Pi MYSQL Database*. Pi My Life Up. https://pimylifeup.com/raspberry-pi-mysql/
- Corke, T. (2021, 15 augustus). Building Your Own Team Rating System. Matter of Stats. http://www.matterofstats.com/mafl-stats-journal/2013/10/13/building-your-own-team-rating-system.html