1-Software Requirement Specification(SRS)

SRS is a detailed description of the system purpose and it consist of many branches like : Introduction, Overall Description, System Features, External Interface Requirements and finally Functional and non Functional Requirements but now we will focus on Functional and non Functional Requirements.

1.1 Functional Requirements

Functional Requirements have to do with the functionality of the system and as our system here is Tic Tac Toe game so we should talk about the main functions that achieve that and now we will divide the main functions into two parts : main code functions and database functions.

a- Main code functions

1- PlayerMove: this function returns bool and its job is to see the place played at is empty or not if empty returns true and if not return false and it save all the moves for the players throw the game to can be saved later in the database.

2- CheckGameState: this function returns int and from its name it retkurns the state of the game so if the game not finished yet it returns continue_state (0) and if the game ends and one of the players wins it returns Winner_state(1) and if the game ended in draw it returns the Tie_stste(2)

3- AiMOVE: this function is called if the user wants to play against AI.

4- Minimax: this function returns int and it is related to the AI part so its function is to return the best place that can be played at after some calculation can reach 50000 for one mpve.

b- Database functions

1- FileExists: tis function returns bool and its job is to return true if the file opened without any problem and retunes false if not.

2- SaveUserData: this function is called to save every thing about the user data in the file opened before.

3- RegisterUser: this function is called when user wants to create a new account so what it does that it takes the user name and the password the user entered and send them to the function SaveUserData to save them in the file.

4- LoadUserData: this function is called to load the user data from the file.

5- Login: this function returns bool and its job is to compare the username and password the users enter with the save username and password so it returns true if the user did have an account and returns false if not.

6- SaveLastGame: this function is called when the game ends and it collects every thing about the game data(game_state, first to play, game_time, array of moves and won) and sends them to SaveUserData function to save them in the file.

We can see the game rules in some functions like RegisterUser which forces the user to create an account if he\she does not have, Login which forces the user to remember the username and password he\she entered at the first time to can open the game, PlayerMove which forces the user to play just in the empty places and forces him to choose either playing with AI or with another player and we can see the system behavior by looking at Fig.1 which is a flow chart that shows the system behavior for main functions.

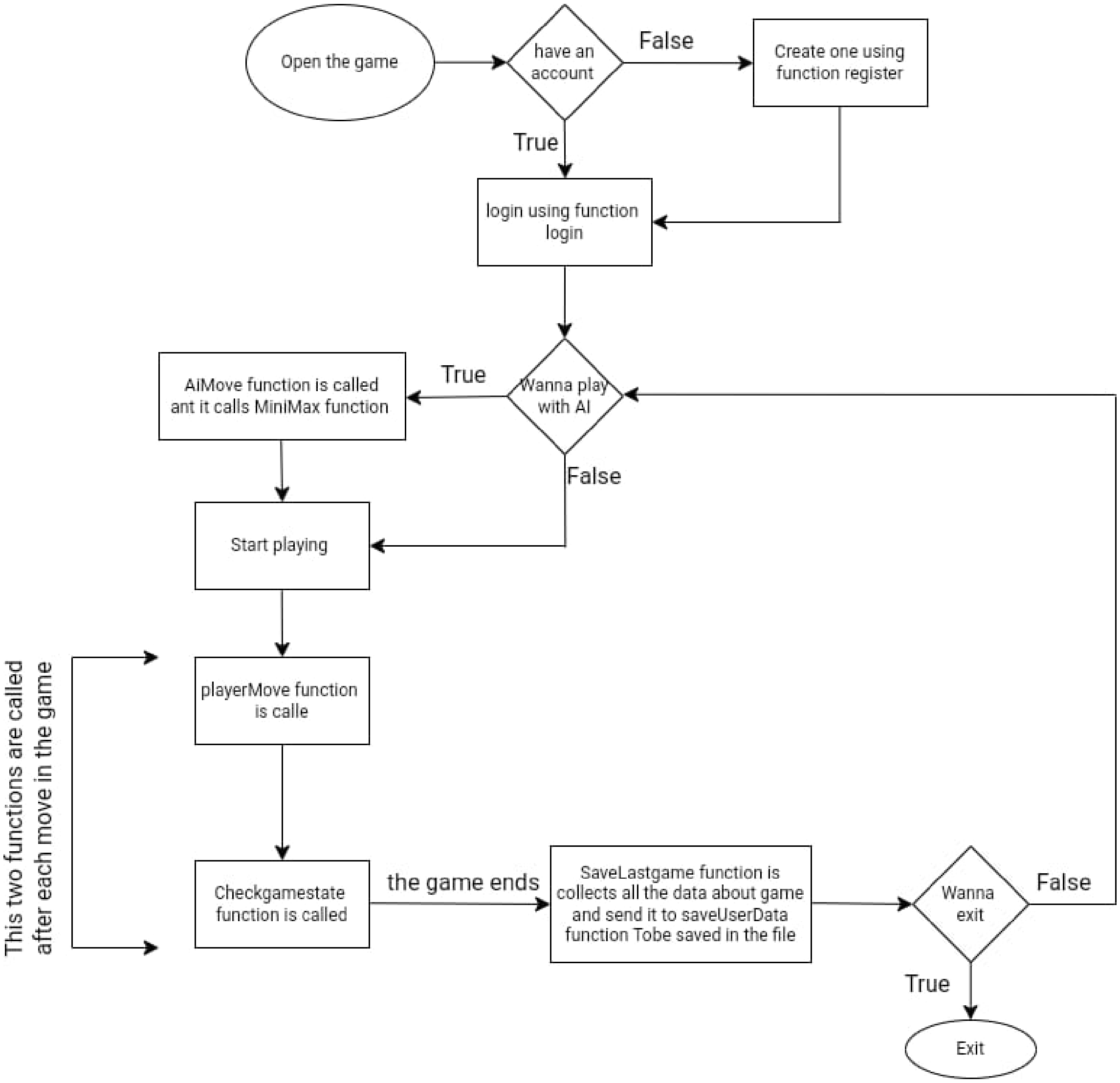
Open the game
have an account
False
Create one using function register
True
login using function login
AiMove function is called ant it calls MiniMax function
True
Wanna play with AI
False
Start playing
playerMove function is calle
This two functions are called after each move in the game
Checkgamestate function is called
the game ends
SaveLastgame function is collects all the data about game and send it to saveUserData function Tobe saved in the file
Wanna exit
False
True
Exit

Fig.1.Flow chart shows the system behaviornby showing the main functions for the system

1-2 Non Functional Requirements

Non Functional Requirements refer to the system non functional properties like cost, life time, reusability, availability, security, flexibility, performance and more other things and now we will talk about some of them.

1- Performance: we tried to improve the performance by trying to save memory as at the first we were saving two arrays in the code one to save the moves in the game and the second to save each move is X or O but we dealt with that by saving one array which contains the moves of the game and int that indicates to who played first X or O so we saved big memory for each game.

2- Security: and that is done by hashing some user data and that is done by using the md5 which is a hashing algorithm from github

3- Showing the score of wins, loses and ties by using the functions wins, loses and ties and the history for each game played before.