

**Tic-Tac-Toe**

**Made by:**

**Dávid Arthur**

**Kelemen Dániel**

**Leading teacher:**

**Szántó Zoltán**

**1. Introduction**

The people observe their environment through their senses. We have five main senses: sight, hearing, smell, taste, touch. The most important of these is sight. The eye delivers 80 percent of the information we need to perceive the world around us. We recognize colors, shapes, movements with the help of our eyes. It recognizes gestures as well, so does our game, which is Tic-Tac-Toe, but not just the simple game, it is controlled with hand gestures. The game uses a camera to detect the movements of the players hand. Tic-Tac-Toe is a simple game, but we made it more fun with this feature, and also you don’t have to be close to your friend, because now you can play it through the internet.

**2. Goals**

Our goals for this project were to make an enjoyable, but easy to use game. We wanted it to be a game which can be played to pass time and it doesn’t involve much preparing and learning the mechanics of the game, or a high-end electronic device.

**3. Requirements**

3.1 User requirements

In order to play our game, the user needs to register a new account or if he has one already, he needs to log-in. To register, the user needs to fill out a registration form which a username and a password. The user must be aware that the credentials are encrypted, and his public IP address is used for logging purposes. The credentials provided by the Player should be noted down, because at the time the game doesn’t have a password recovery feature.

After the Login/Registration procedure, the player can adjust the game settings in the ‘Settings’ page, and after that he can start a game, from where he should choose the game mode he wants to play.

3.2 Use cases

3.3 System requirements

3.3.1 Functional requirements

The system requirements vary according to the way the user wants to play the game. In order to play online the player needs internet connection, if he wants to control the game with hand gestures, the system has to be connected to a camera.

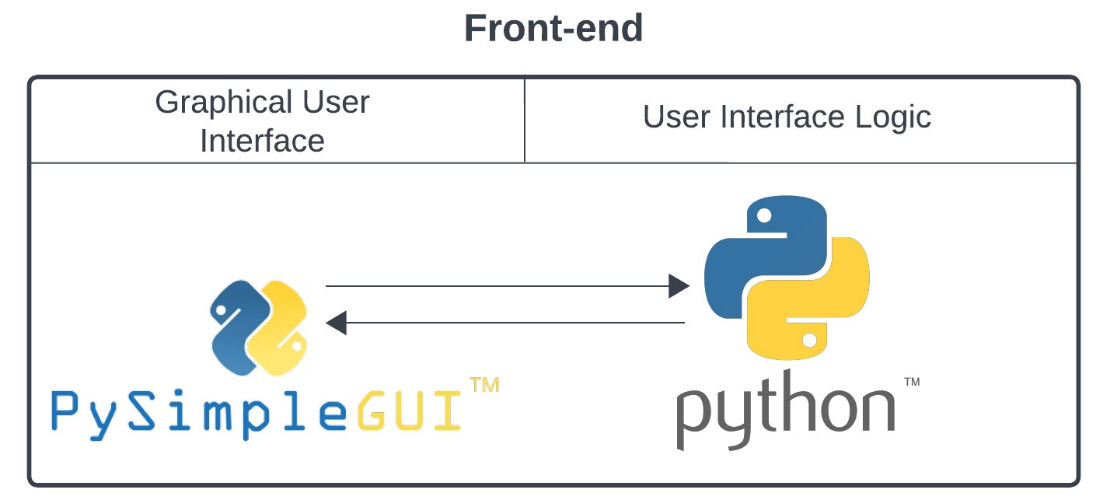
* User authentication: The Game allow players to save their scores, and use them on multiple devices(not simultaneously). Also saves their unique username that the Player can select.
* Personalized game experience: Is allowed to change the Background color of the game in the ‘Settings’ menu, and also if the player has multiple Webcams connected, they can use which they want. If using camera, the grid view can be enabled in the view.
* Multiple game mode types: The game provides multiple game modes that you can choose from, an play it: PVE(Player vs. Environment), SamePC(2 players on the same device), PVP(Player vs Player)
* Player selection: The Player can choose it’s enemy if he plays PVP. He can choose one from the list of Online-Players
* Leaderboard: The game provides possibility to check the all-time all-player leaderboard
* Tutorial: The new Players can read a brief walkthrough of the game explaining the base mechanics, and game modes.

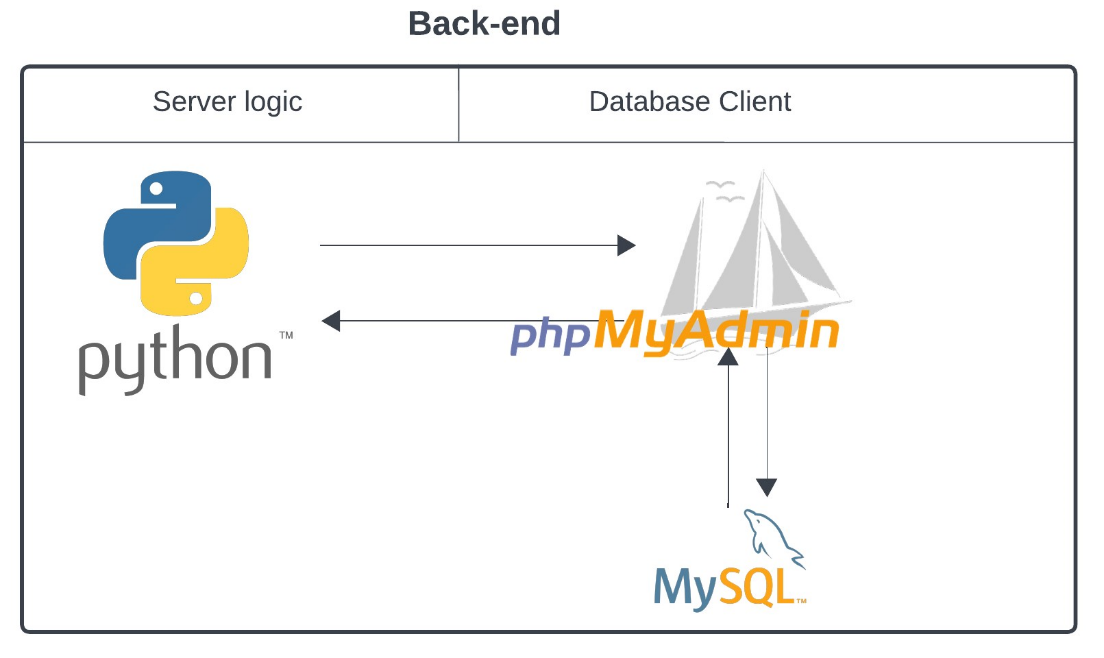
3.3.2 Non-functional requirements

* PC or Laptop running Linux/Windows OS
* Stable internet-connection
* Python 3.10 release or higher
* Storage space: 450 MB at least
* Memory: 512 MB RAM at least
* Dedicated or Integrated Webcam with a minimum resolution of 480x640
* Dedicated or Integrated Graphics Card

**4. System Design & Architecture**

4.1 System architecture

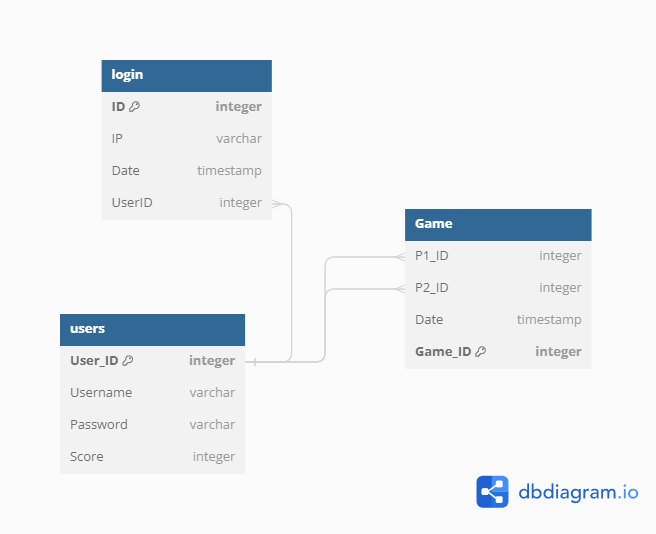




As the figure shows above, we used Python as our main programming language, because it had the most libraries we needed to create our game, and we didn’t had to go on low-level programming.

For realising the GUI(Graphical User interface) we used PySimpleGUI, which was easy to use and enough for what we needed it to be.

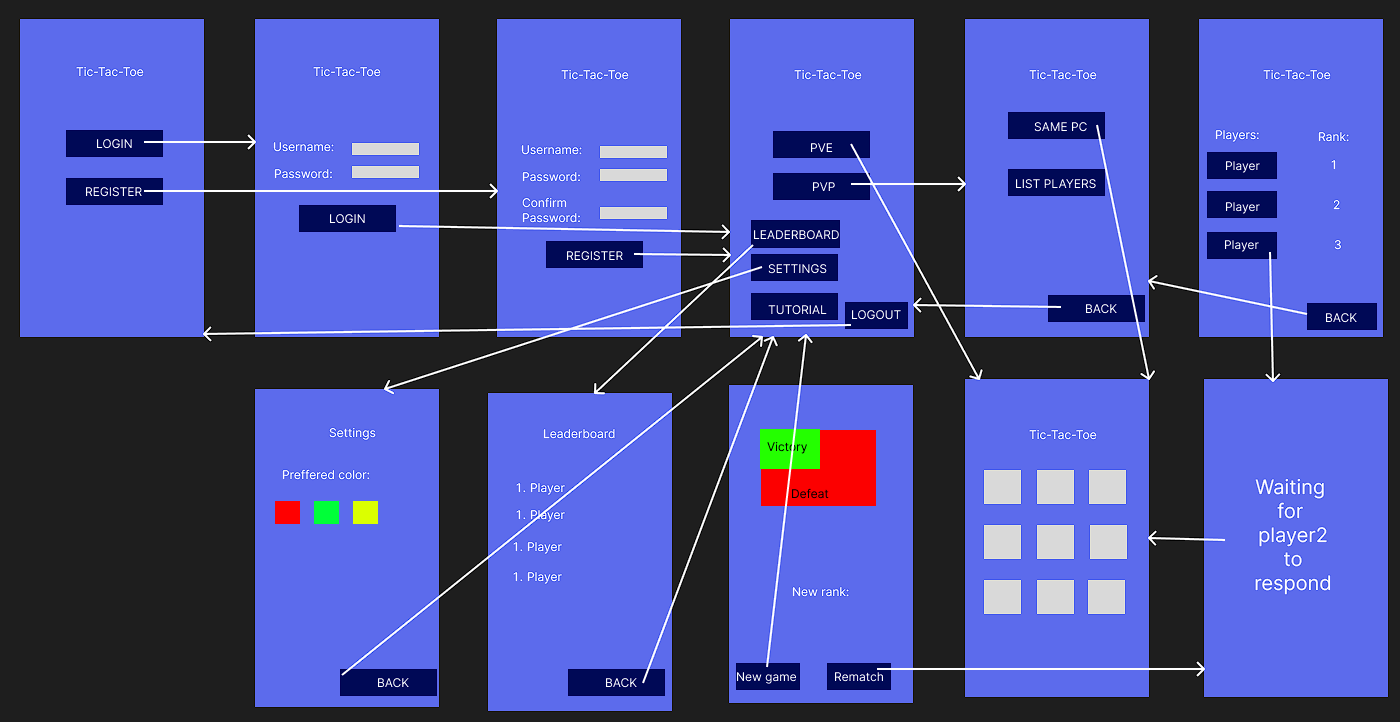
For creating the database we used MySQL. It is a fast database management system and has good connectivity, easy to use, which was needed for us, because the game communicates with a server through the internet. You can see our database plan below:



To integrate the database with Python, we created an ORM(Object–relational mapping) using SQLAlchemy.

4.2 UI Plan

We made the UI plan at the start of the project. It helped us so much, we had something to follow throughout the whole process of creating the UI. The design and some parts of this plan changed along the way, because we came up with new ideas and better solutions. You can see the plan below:



4.3 Management

For keeping track of our work we used Trello, where we initially added the main functionalities to do, after we started to develop certain parts of the initially added fucntions, we added more detailed descriptions of what we need to do. Trello board was a good way to see how we work, what we have to do and what we have done already, so the tracking of the project could go straight-forward without needed to think what should we do next. It was also a good place to divide the work among us.

For version control we used GitHub. It is a good platform for working simultaneously on the same project. Our strategy here, was to create branches for every part of our project or feature. We divided the work so that we could work at the same time, we developed two separate parts of the project at the same time so we didn’t have to wait for each other.

Also the GitHub is very safe in the data-loss perspective, so we didn’t have to worry about overwriting each other codes’, and also we could track every version of the project, and return to a safe version if something bad happened.

**5. Application walkthrough**

**6. Conclusion**

The application in its current version is stable, playable, so we can call it as a first version. As by now the main features of the game are that is

* Multiplayer, so the players can play across the internet
* Hand/gesture recognition, so the players can play by their hands

6.1 Further development possibilities

In the future we would like to extend the features of the game by offline-games, so you don’t have to connect to the server, and don’t need an internet connection, thus your ranks will be updated when synchronized with the server.

Another feature we would like to add, is the cross-platform playing. We want to make a GUI on Android so it can be played not only with a Windows/Linux operating system.