SAINT LOUIS UNIVERSITY
School of Accountancy, Management, Computing and Information Studies

Academic Year 2024 - 2025
Maryheights Campus, Bakakeng, 2600, Baguio City, Philippines

Documentation:

Python CORBA Client for LETTRBOX "What's The Word" Game

Submitted By:
9322 TEAM 4
CS 222 MTh 9:30-12:00

CALULUT, FRANCIS JOSH
DELA PEÑA, KERNEL ALAN
DORIA, FEIL JASPER
ESPEJO, JASMINE ROSE
GARCIA, ALWIN JOHN
GERONIMO, JOHN REY
TOLENTINO, ALVIN JOHN

Submitted To:
MR. RODERICK MAKIL
INSTRUCTOR

MAY 2025

Python Documentation

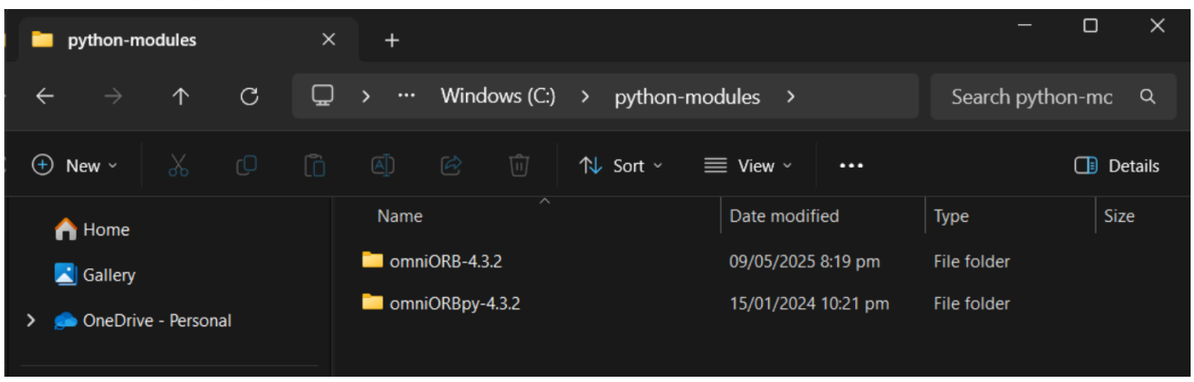
I. Steps to Run the Python Client Program

Prerequisites: Download and Install Visual Studio and Cygwin.

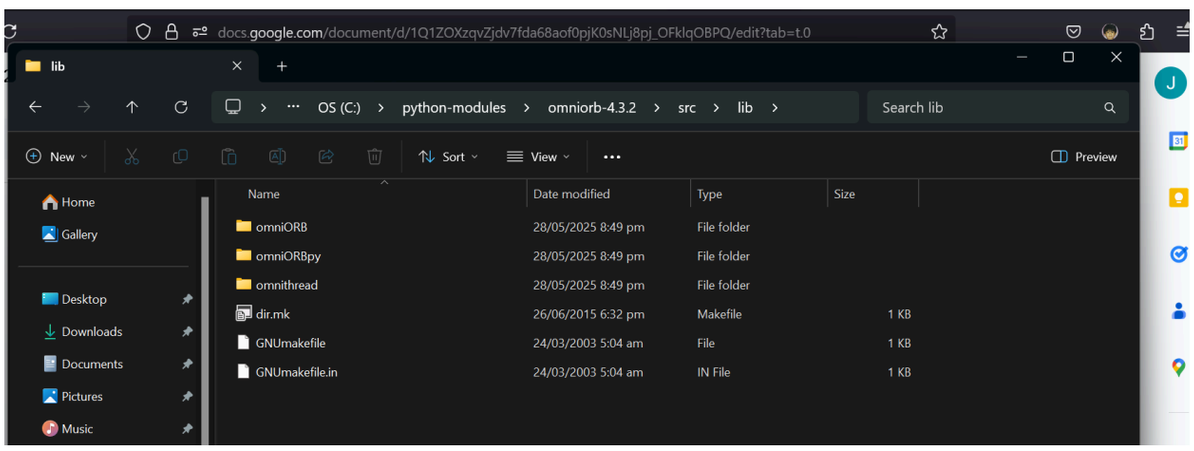
1. Install Python dependencies (omniORB, omniORBpy & omniidl)

a. Download the preferred version of omniORB and omniORBpy from this link: https://omniorb.sourceforge.io/
The group used the 4.3.2 version of omniORB and omniORBpy which can downloaded from these links:
omniORB 4.3.2 :
https://sourceforge.net/projects/omniorb/files/omniORB/omniORB-4.3.2/
omniORBpy 4.3.2 :
https://sourceforge.net/projects/omniorb/files/omniORBpy/omniORBpy-4.3.2/

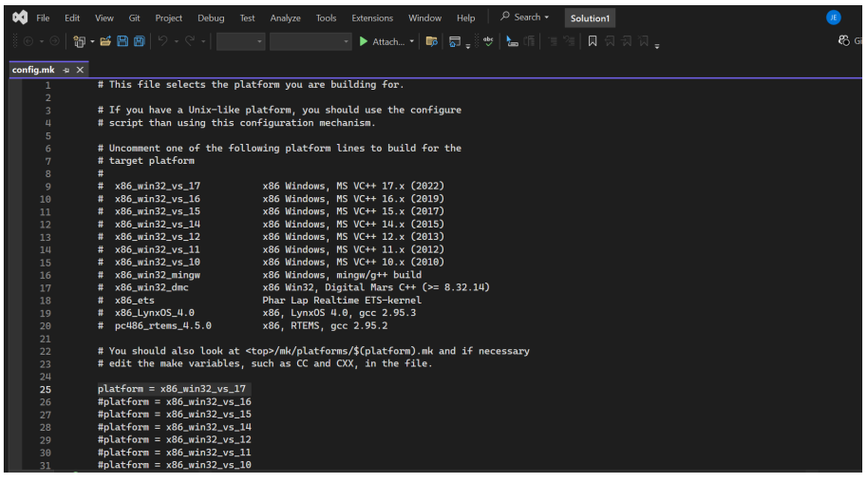
b. Extract the downloaded .zip files
Create a directory named *python-modules* in the path where the zip files will be extracted.



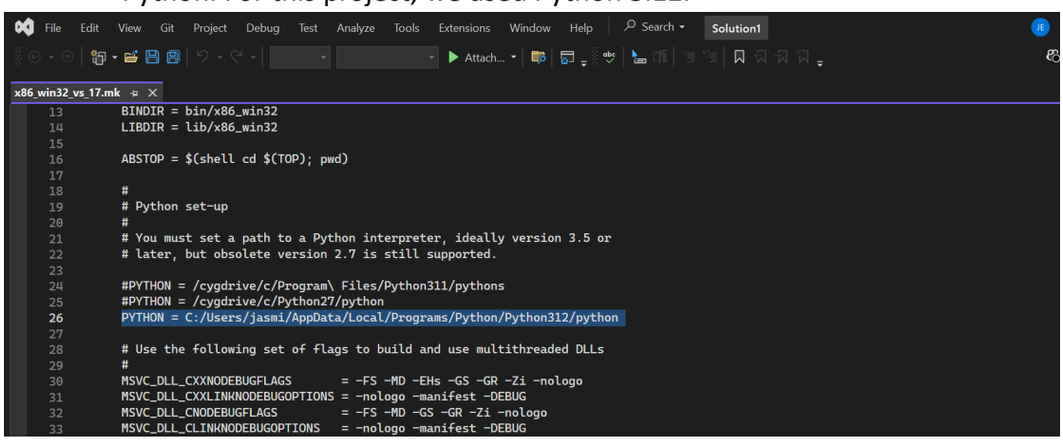
Rename the *omniORBpy-4.3.2* folder to *omniORBpy* and place it under the omniORB-4.3.2/src/lib folder.



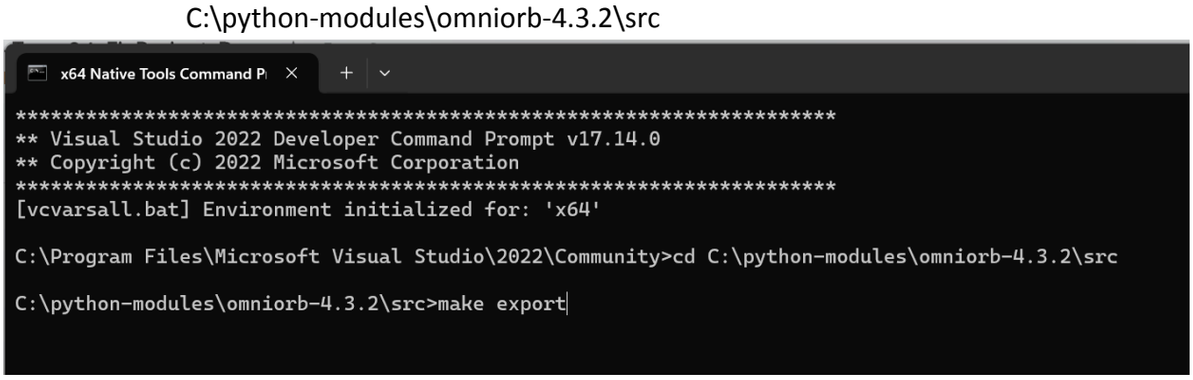
In the python-modules/omniORB-4.3.2/config directory, edit the config.mk file based on the version of Visual Studio that is installed. The group installed Visual Studio 2022 so, the line with platform = x86_win32_vs_17 should be uncommented.



Then, edit the x86_win32_vs_17.mk file from python-modules/omniorb-4.3.2/mk/platform. Set the path of the installed Python. For this project, we used Python 3.12.

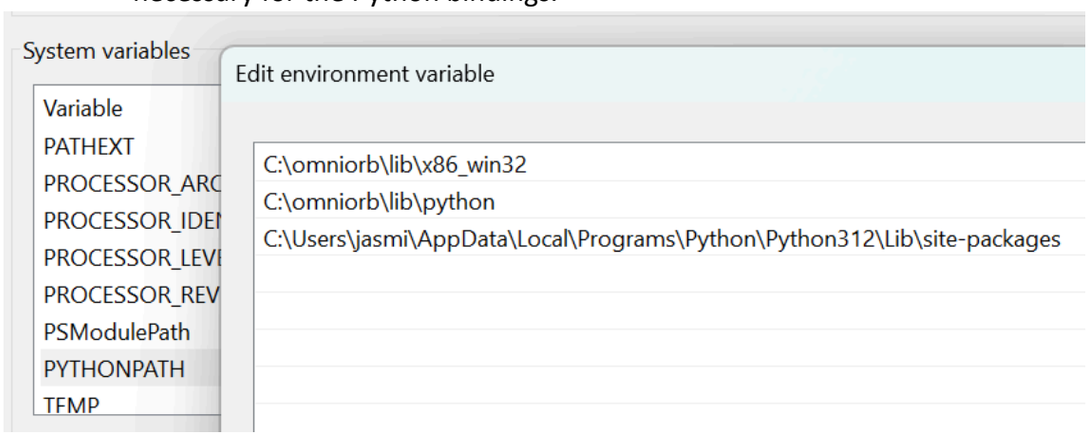


Compile omniORB and omniORBpy using the *make* command in Native Tools Command Prompt for Visual Studio.
For this instance, go to this directory and run *make export:*
C:\python-modules\omniorb-4.3.2\src



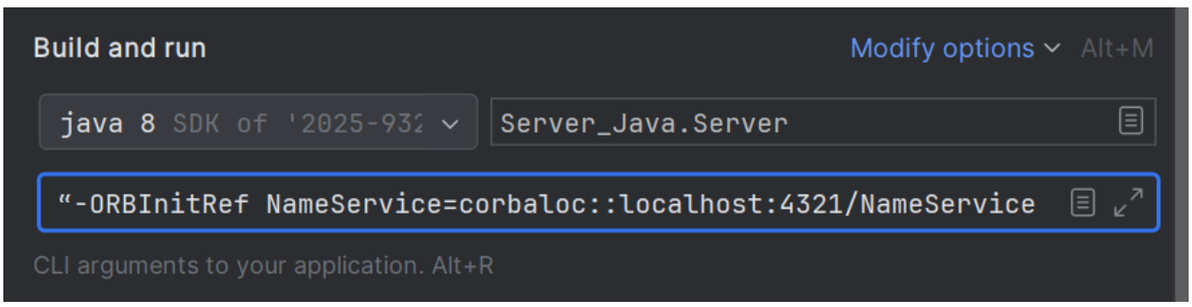
Run the same command in the C:\python-modules\omniorb-4.3.2\src\lib\omniORBpy directory.
The make command is important because it compiles and builds executable programs, libraries, and other outputs that will be necessary for the Python Client Program.

Add the PYTHON PATH to the system variables of the terminal. This will be necessary for the Python bindings.

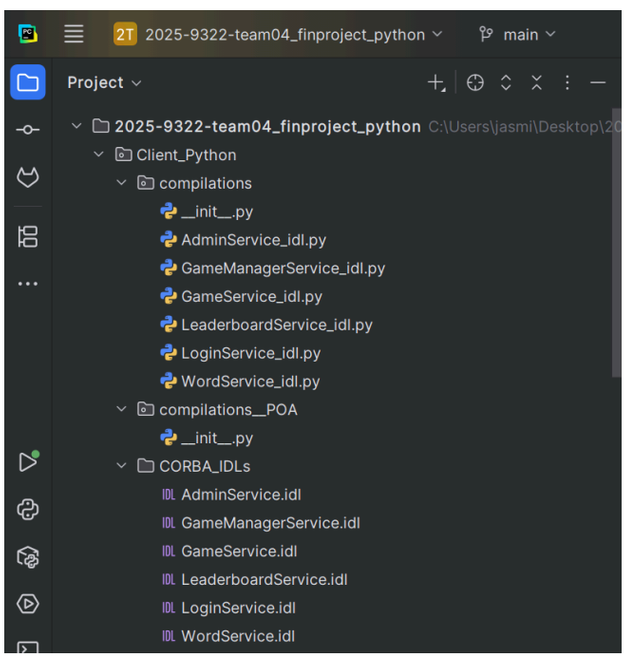


Note: The omniORB package already has the omniidl module.

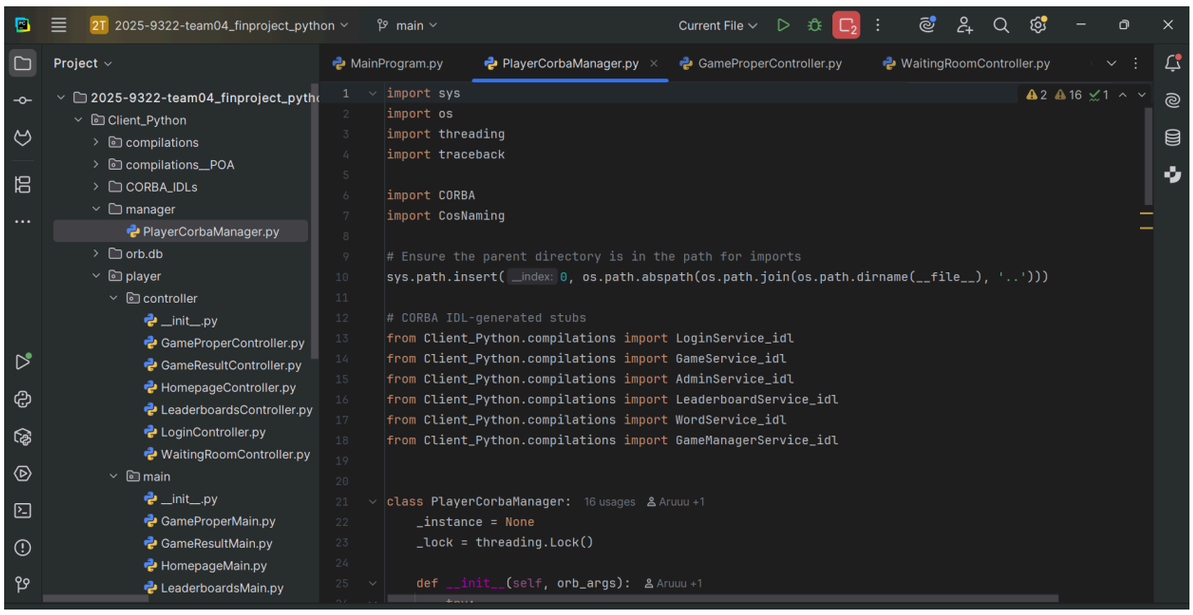
Connect the Server and Java Client to Name Service daemon by putting "-ORBInitRef NameService=corbaloc::<hostname>:<port>/NameService" in the program arguments. For this instance, we utilized "localhost" and port "4321".



2. **Compile IDL files for the Python Client Program**
   In the command prompt, go to the directory of the IDL files that will be compiled and run this command: "omniidl.exe -bpython <IDL filename>.idl". This should generate the stub files in the "compilations" module.

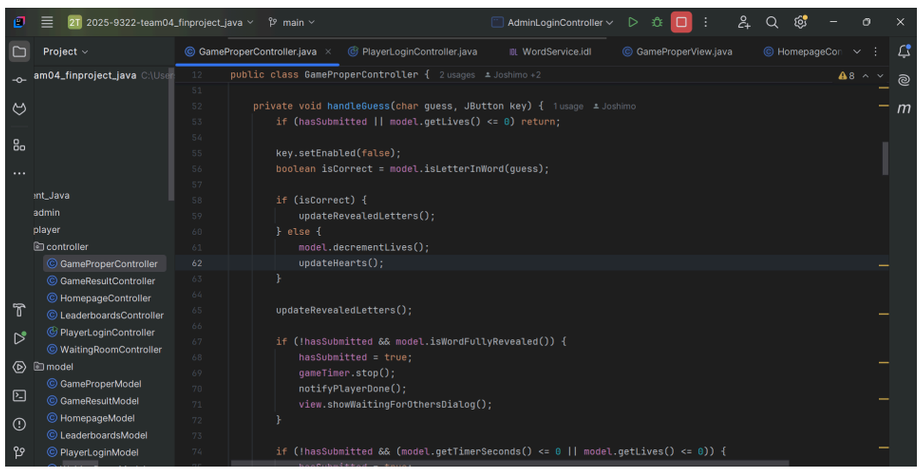


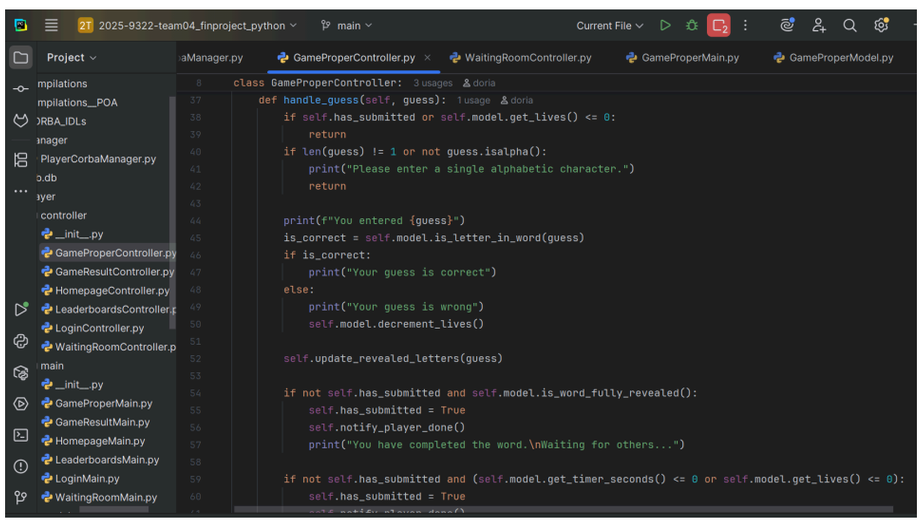
3. **Implement Python Client Program**
   In the “PlayerCorbaManager” class, import the generated stubs and connect to the CORBA server.



4. **Follow the same logic from the Java Client Program to the Python Client Program**

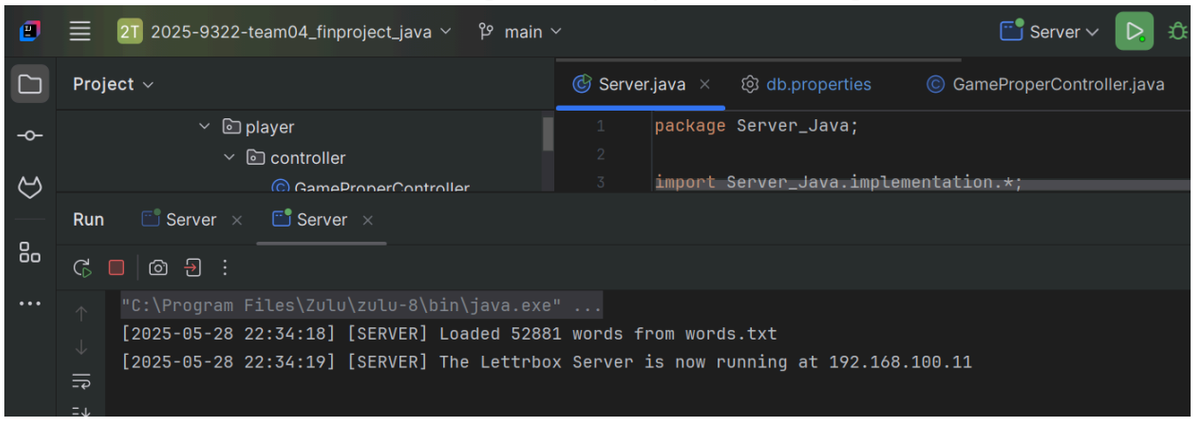
   Example:



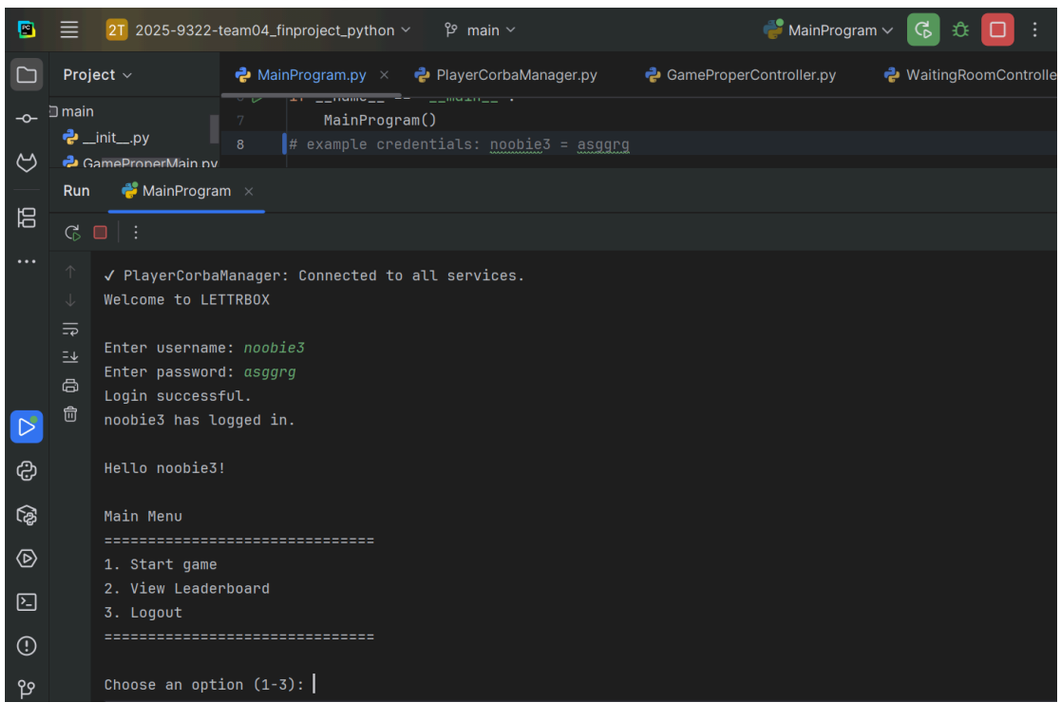


5. **Run the program**

Ensure that the server is running to run the Python Client Program.



Run the main Python Program.



II. Issues Encountered and Resolutions

1. Mismatched Logic with the Java Client Program

Issue:

a. The Python Program does not have GUI unlike the Java Program which affected how user interactions and program flow were handled, causing some difference on how the overall program works.
b. Some logic implemented in the Java client was initially missing or inconsistent in the Python client.

Resolution:

a. The members assigned for the Python side thoroughly reviewed and went over the flow and logic of the Java Client Program and replicated the same logic in the Python version using Command Line Interface or CLI.

2. Setting Up omniORB

Issue:

a. Due to unfamiliarity and first time users of omniORB, setting it up was difficult and took most of the time.
b. The initial approach using “pip install” did not work properly due to missing dependencies or incompatible configurations.

Resolution:

a. Instead of using the “pip install” command for setting up, the members manually installed and compiled the necessary files for omniORB and omniORBpy as well as setting up the environment variables.

3. Using the right python version

Issue:

a. The initial version of Python, Python 3.13, used for the project is incompatible with omniORB and other dependencies.

Resolution:

a. Installed a compatible version of Python, which are versions 3.12 and 3.10.

4. Debugging (Especially Imports and Return Types)

Issues:

a. Frequent errors occurred during the import of CORBA stubs and when receiving or sending data through CORBA methods. This included ImportError, attribute errors, and mismatched return types between the server and Python client.

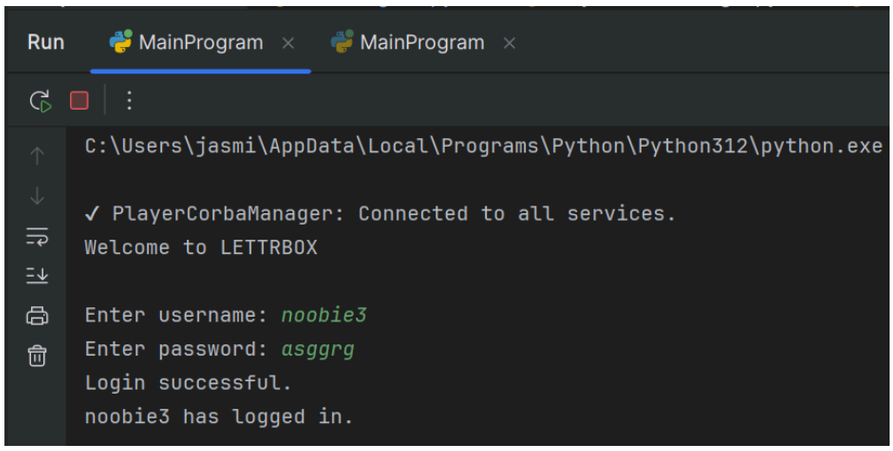
Resolution:

a. Carefully reviewed the structure of the generated Python stubs and ensured proper import paths were used.
b. Referred to the IDLs to ensure consistent return types.
c. Used try-except blocks to handle unexpected behavior gracefully, leading to more stable communication.

III. Python Client Program Navigation

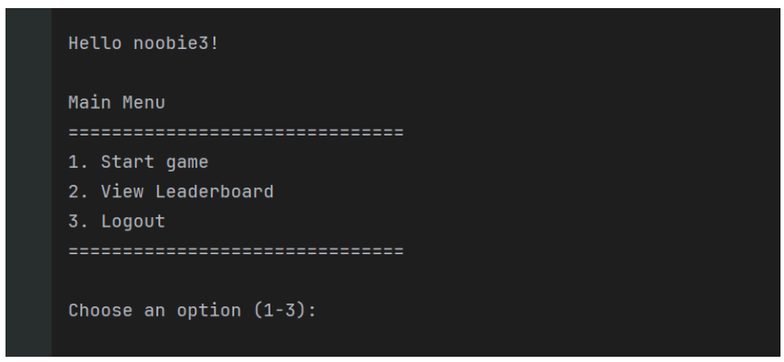
1. Login Page

The user should input a valid username and password.



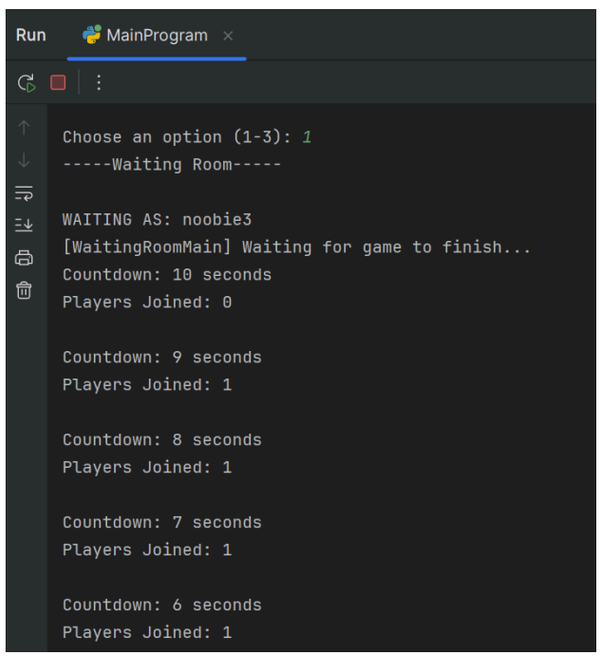
2. Homepage

Shows the main menu and options of what the player wants to do with the program. The players have the option to start the game, view the leaderboard, and logout of the program.



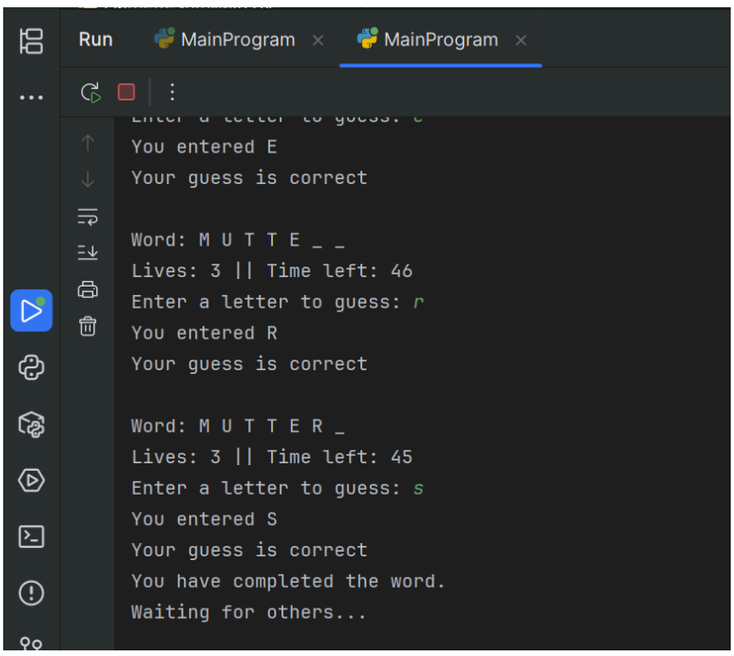
3. Waiting Room

When the players select option 1, which is to start the game, they will enter the waiting room that allows other players to join the game. It displays the time remaining for waiting and how many players are currently waiting.



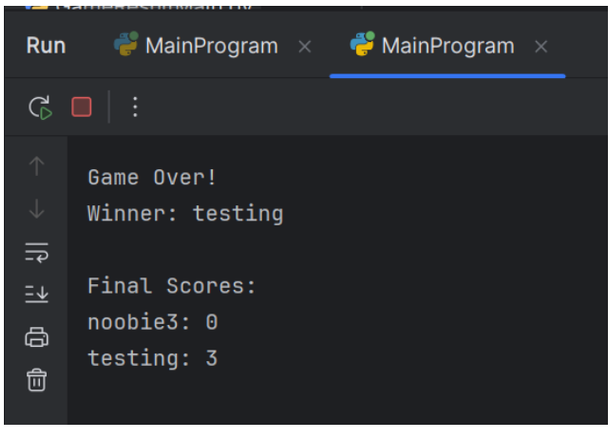
4. **Game Proper**

   The spaces corresponding to the number of letters in the word to guess should be displayed as well as the lives and the time left for the current round. The program should also show if the guess is right or wrong.



5. **End of Game**

   When the game is over, the program shows who is the winner of the game.



6. **Leaderboard**

   The leaderboard shows the list of the top players with the most game wins.