**Проект на PyQT5**

**Название проекта: «Морской бой»**

Выполнил:

Ходжаев Дамир

1 Часть. Серверная часть

Процесс запуска:

1) Запускаем файл main.exe расположенного в пути «где распаковали…\App\Server\hall\_server\hall\_server.exe»

2) Создалось два файла «settings.inf» и «accounts.db», файл с настройками сохраним на флэшке.

Как работает:

1)Обработка данных(принципа было 2):

1. На сервер поступает запрос по примеру “need\_to\_do?param1=value1& param2=value2& param3=value3&”

И создается словарь с ключами и значениями d= {param\_N: value\_N}

Происходит обработка данных, и возвращается “True?some\_data\_1” или “False?some\_data\_2”

1. На сервер поступает строка, с помощью библиотеки json, преобразуем в словарь

Как на примере в (а), и дальше как с (а) только возвращаем не просто строку, а словарь.

2) Соединение игроков: при авторизации игроков на сервер у них статус «в меню», при нажатии на кнопку «поиск игроков» статус меняется на «в поиске игры»,

Когда другой человек будет искать игру, а в словаре уже есть игрок у которого статус «в поиске игры», сервер возьмет данные игроков (из словаря) проверит оба ли игрока еще присутствуют на сервере, если да, то в отдельном потоке запуститься сервер для битвы, где будут играть игроки.

2 Часть. Клиентская часть

Процесс запуска:

1. Файл с настройками который мы скопировали вставляем в папку расположенную в пути

«где распаковали…\App\Client\settings», если спрашивают заменить ли смело жмите ДА

2) Далее запускайте клиент расположенный в пути «где распаковали…\App\Client\main\main.exe»

Процесс использование:

1. Для входа в аккаунт вам нужно для начало его зарегистрировать, для этого жмем на кнопку «Зарегистрировать»

Возможные вывод:

«Success register», если все ОК.

«Какое то поле имеет запрещенный символ (?&=)»

«Такой логин уже зарегистрирован»

«Какое-то поле пустое»

1. После того как авторизуемся можно смело входить в аккаунт в главной форме авторизации, которая появляется после входа в клиент или после регистрации.

Возможные выводы:

«Какое-то поле пустое»

«Login is already using», если логин уже используется.

«Какое то поле имеет запрещенный символ (?&=)

«Не верный пароль», если аккаунт с таким логином есть, но пароль не верный

«Не верный логин», если нету аккаунта с таким логином

1. Если вы забыли пароль есть Label («Забыл пароль»), нажав на который выйдет диалоговое окно в котором нужно ввести логин и секретное слово, которое вы указывали при регистрации.

Возможные выводы:

«Какое-то поле пустое»

«Не верный логин», если логина нету в бд

«Не верный секретный ключ», если введенный секретный ключ не совпадает с тем что в бд

1. Если авторизация была успешной выйдет игровое меню, на котором показаны ваши победы и поражения, а также находиться кнопка для поиска противника
2. Когда нажимается кнопка поиска противника, запускается второй поток в котором выполняются запросы на сервер, и пока не найдется второй игрок, второй поток ждет ответа от сервера.
3. Ответ полученный от сервера(главного) будет содержать (ip, port) сервера на котором будет сражение, после вызывается окно с расстановкой кораблей
4. После расстановки кораблей нажимаете на «готов» и приложение начинает ждать оппонента.
5. Вы сражаетесь с друг-другом, и у того кто выиграл выведется диалоговое окно сообщающее о победе, другому о поражении, и обоих игроков выкинет в главное меню

Особенности:

Использовалось СУБД SQLite3, json, socket, threading (на сервере, на клиенте QThread), PyQT5

Из шаблонного программирования: Декоратор (Для вывода ошибок в диалоговое окно)