**Пояснительная записка**

**Файл «main\_window.py» - front-end**

**Класс MainWindow, является полностью моей собственной разработкой по улучшению работы с модулем PyGame в рамках проектной работы.**

**Константы файла:**

* **COLOR\_ACTIVE\_BUTTON –** кортеж цвета (RGB) кнопки, когда на неё наведён курсор
* **PLAYERS\_COLOR –** кортеж цвета (RGB) цвета колонии игрока
* **ENEMY\_COLOR –** кортеж цвета (RGB) цвета колонии оппонента (ИИ)

**Данный класс также содержит функции (в дальнейшем методы):**

* **\_\_init\_\_ -** метод, отвечающий за инициализацию данного класса, с именованными (имена переменных говорящие) переменными
* **initUI –** основной метод класса, отвечающий за использование функций, создающих или изменяющих главное окно
* **check\_pos\_on\_button –** метод, проверяющий положение курсора мыши на зоне кнопки
* **initialization\_buttons –** метод, инициализирующий объекты класса Button
* **init\_colonies –** метод, в последствии инициализирующий игровые колонии, как объекты класса файла “back-end”
* **render\_starter\_menu –** метод, “рисующий” стартовое меню
* **render\_menu\_download\_game –** метод, “рисующий” меню загрузки игры
* **render\_settings\_menu –** метод, “рисующий” меню настроек
* **render\_rules –** метод, “рисующий” меню правил игры и сути игры
* **render\_game\_lvl –** метод, “рисующий” игровой уровень
* **render\_endgame –** метод, “рисующий” конец игры

**Класс Button, используемый для "рисованя” кнопок.**

**Данный класс также содержит функции (в дальнейшем методы):**

* **\_\_init\_\_ -** метод, отвечающий за инициализацию данного класса, с именованными (имена переменных говорящие) переменными
* **print\_text\_on\_button –** метод, для написания текста на кнопках
* **draw\_button –** метод, для “рисования” кнопки
* **draw\_active\_button –** метод, для “рисования” активной (на которую навели курсор) кнопки

**Класс DrawBoard, используемый для “рисования” игрового поля.**

**Данный класс также содержит функции (в дальнейшем методы):**

* **\_\_init\_\_ -** метод, отвечающий за инициализацию данного класса, с именованными (имена переменных говорящие) переменными
* **draw –** метод, для “рисования” поля
* **draw\_player –** метод, для “рисования” колонии игрока
* **draw\_enemy –** метод, для “рисования” колонии оппонента
* **draw\_enemy –** метод, для “рисования” колонии оппонента
* **get\_array -** метод, возвращающий массив всей необходимой информации о каждой из колоний на рисунке поле

**Файл «backend.py» - back-end**

**Класс ActionWithTable является классом для работы с файлами формата «.csv».**

**Данный класс также содержит функции (в дальнейшем методы):**

* **\_\_init\_\_ -** метод, отвечающий за инициализацию данного класса, с именованными (имена переменных говорящие) переменными
* **get\_dict –** метод, отвечающий за возвращение словаря из необходимого файла “.csv”
* **rewrite\_file –** метод, переписывающий необходимый файл “.csv”
* **get\_dict –** метод, отвечающий за возвращение матрицы из необходимого файла “.csv”
* **get\_array –** метод, отвечающий за возвращение списка чисел из необходимого файла “.csv”

**Классы PlayersColony и EnemyColony являются классами для осуществления взаимодействия между игровыми колониями/колониями ИИ, в дальнейшем могут получить сложение между собой, но изначально задумка заключалась в их разделении с некоторыми особенностями.**

**Данный класс также содержит функции (в дальнейшем методы):**

* **\_\_init\_\_ -** метод, отвечающий за инициализацию данного класса, с именованными (имена переменных говорящие) переменными
* **\_\_iadd\_\_ –** метод, осуществляющий сложение класса и числа, для удобства в коде
* **remove\_lives –** метод, удаляющий население колонии, далее взаимодействует с программой по средству логических значений

**Класс PeacefulColony является классом для осуществления игровых взаимодействий с мирными колониями (которые не могут производить захват), в дальнейшем могут получить сложение между собой, но изначально задумка заключалась в их разделении с некоторыми особенностями.**

**Данный класс также содержит функции (в дальнейшем методы):**

* **\_\_init\_\_ -** метод, отвечающий за инициализацию данного класса, с именованными (имена переменных говорящие) переменными
* **\_\_iadd\_\_ –** метод, осуществляющий сложение класса и числа, для удобства в коде (ограничение на население - 10 человек на одну колонию)
* **remove\_lives –** метод, удаляющий население колонии, далее взаимодействует с программой по средству логических значений

**Файл «artificial\_intelligence.py» - back-end**

**Класс GameBot является искусственным интеллектом для игры.**

**Данный класс также содержит функции (в дальнейшем методы):**

* **\_\_init\_\_ -** метод, отвечающий за инициализацию данного класса, с именованными (имена переменных говорящие) переменными