**Пояснительная записка к приложению «Hurry Up»**

**Цель проекта:** Создать 2D-головоломку с видом сверху, суть игры – пройти через лабиринт, полный опасностей и препятствий, собирая награды (звёзды).

Обеспечение возможности сохранения результатов игры пользователя.

**Исполнители:** Спивак Максим Игоревич, Кривова Полина Дмитриевна

**Используемые библиотеки:**

pygame== 2.6.1

**В приложении реализованы следующие возможности:**

**1. Управление персонажем и перемещение по уровням:**

Игрок управляет персонажем, перемещая его по игровому полю с помощью клавиш WASD. Уровни представляют собой лабиринт с различными препятствиями и объектами.

**2. Сбор ключей и открытие дверей:**

На уровнях расположены ключи и двери, для открытия которых требуется собрать соответствующий ключ. Собранные ключи отображаются в инвентаре игрока.

**3. Сбор сундуков:**

На каждом уровне расположены сундуки, которые необходимо собрать для его завершения. При соприкосновении игрока с сундуком он исчезает с уровня.

**4. Ограниченное время и проигрыш:**

Игроку дается ограниченное время на прохождение уровня. Также игра заканчивается проигрышем, если персонаж попадает в воду. При проигрыше игроку предлагается перезапустить уровень.

**Реализованны функции:**

• **load\_image():** Загружает изображение из файла.

• **load\_level():** Загружает данные уровня из текстового файла.

• **create\_level():** Создает объекты уровня на основе загруженных данных.

• **draw\_digit():** Отрисовывает число в виде стилизованных цифр на экране.

• **draw\_text():** Отрисовывает текст на экране.

• **draw\_clock\_face():** Отрисовывает прямоугольник для фона таймера.

• **draw\_pause\_button():** Отрисовывает кнопку паузы/продолжения игры.

**Реализованы следующие классы и их методы:**

**Base(pygame.sprite.Sprite):** Базовый класс для игровых объектов, в атрибуте picture: ключи – закодированные символы объектов, значения - картинки

• **move():** Перемещает объект.

• **cut\_sheet():** Кадрирует картинку для анимации.

• **reverse\_image():** Осевая симметрия картинки.

• **update():** Меняет изображение объекта. Создано для анимации.

**Player(Base):** Класс игрока, наследуется от **Base**. Перегружен **move()** на **pass**

• **is\_collide():** Проверяет столкновение игрока с группой спрайтов.

**Door(Base):** Класс двери, наследуется от **Base**.

• **is\_open:** Атрибут, указывающий открыта ли дверь.

**Monster(Base):** Класс монстра, наследуется от **Base**.

• **trajectory**: Атрибут, содержащий координаты вершин многоугольника, по которому передвигается объект.

• **current\_point\_index**: Атрибут, содержащий номер той вершины, из которой вышел объект.

• **next\_point\_index**: Атрибут, содержащий номер той вершины, в которую идёт объект.

• **speed**: Атрибут, содержащий скорость объекта.

• **load\_trajectory():** Создаёт **self.trajectory.**

• **calculate\_movement():** Возвращает перемещение объекта по пути.

• **update\_monster():** Проверяет передвижение объекта по траектории, задаёт **self. current\_point\_index** и **self.current\_point\_index.**

• **move():** Перемещает объект.

**Board:** Класс для управления игровым уровнем.

• **set\_view():** Устанавливает положение и размер ячейки уровня.

• **render():** Отрисовывает сетку игрового поля.

• **get\_click():** Обрабатывает клик мыши.

• **get\_cell():** Определяет координаты ячейки по клику мыши.

• **on\_click():** Обрабатывает клики.

• **load\_level():** Загружает и создает уровень.

• **move\_level():** Перемещает уровень и проверяет столкновения.

• **move\_monsters():** Передвигает монстров.

• **draw\_level():** Отрисовывает уровень на экране.

• **check\_portal\_collision():** Проверка прохождения уровня при вход в портал.

• **check\_monster\_collision():** Проверка проигрыша при столкновении с монстром.

**Inventory(Board):** Класс для управления инвентарем, наследуется от **Board**.

• **add\_to\_inventory():** Добавляет предмет в инвентарь.

**Button:** Класс для создания кнопок.

• **draw():** Отрисовывает кнопку.

• **handle\_event():** Обрабатывает событие клика по кнопке.

**PopupWindow:** Класс для создания всплывающих окон.

• **close():** Закрывает окно.

• **run():** Запускает цикл обработки событий для окна.

• **render\_text():** Отображает текст.

**StartWindow:** Класс для создания стартового окна.

• **create\_buttons():** Создает кнопки в стартовом окне.

• **start\_game():** Закрывает стартовое окно и запускает игру.

• **show\_about():** Открывает всплывающее окно "О игре".

• **show\_authors():** Открывает всплывающее окно "Об авторах".

• **run():** Запускает цикл обработки событий для стартового окна.