**Структура проекта.**

Проект разделён на отдельные библиотеки(проекты), каждая из которых предоставляет основные функции программы.

Bot

├───API

│ GetStateActive.cs

│ GetStateMode.cs

│

├───Bot

│ Bot.cs

│

├───DataBase

│ DBLogic.cs

│ Enemy.cs

│ Event.cs

│ User.cs

│

└───Services

IntegrationAddons.cs

IntegrationAPI.cs

SendMessage.cs

**Классы.**

**Библиотека Database:**

* **DBLogic.cs** – класс отвечает за логику работы занесения информации в базу данных.
* **Enemy.cs** – класс хранит в себе информацию о текущем враге.
* **Event.cs** – класс хранит в себе информацию о текущем ивенте.
* **User.cs** – класс хранит в себе информацию о состоянии персонажа игрока.

**Библиотека Bot:**

* **Bot.cs** – точка входа программы, синтезирует в себе всю логику бота.

**Библиотека API:**

* **GetStateActive.cs** – обеспечивает возможность получать состояние активности бота и время работы бота. Не имеет возможности воздействовать на бота.
* **GetStateMode.cs** – обеспечивает возможность получать состояние активного режима бота, а также менять активный режим на альтернативный.

**Библиотека Services:**

* **IntegrationAddons.cs** – обеспечивает возможность добавлять сторонние аддоны для расширения возможностей игры.
* **IntegrationAPI** – предоставляет API бота.
* **SendMessage.cs** – предоставляет возможность отправки сообщения в телеграмме.

**Репозиторий.**

Содержит три основные ветки:

* main: Релизная ветка.
* dev/UI: В этой ветке ведётся работа над улучшением интерфейса и добавлением новых возможностей.
* dev/Bot: В этой ветка ведётся работа логикой бота.