1. **ScriptableObjects**

Наличие в Ресурсах

|  |  |  |  |
| --- | --- | --- | --- |
| Object: | GameData | SaveData | GameTypes |
| Обязательно ли наличие: | Да | Да | Нет |

|  |  |
| --- | --- |
| GameDataObject | |
| static GameDataObject GetData() | Берёт данные из GameTypes, если их нет то возвращает стандартный “GameData”. Иначе возвращает название объекта GameDataObject из списка GameTypes. Если его нет в GameTypes то тоже “GameData” |
| static GDOMain GetMain() | Получить главные данные из GameDataObject с помощью GetData(). |

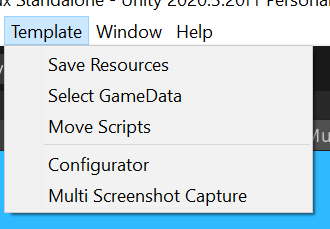
|  |  |
| --- | --- |
| GameDatasManagerObject | |
| static string GetGameDataByLevel() | Возвращает название GameDataObject из списка по номеру уровня который хранится в SaveData. Если GameDatasManagerObject не существует в ресурсах то всегда возвращает “GameData” |

|  |
| --- |
| AbstractSavesDataObject |
| Prefs – Список всех возможных префсов.  PrefType – Список всех типы префсов.  Для создания нового плеер префса просто нужно добавить его в Prefs, а после в массив в SavesData – PrefsValues  Хранится в **GameData.main.saves** |

|  |
| --- |
| SavesDataObject |
| Наследник от абстракции AbstractSavesDataObject. При создании других наследников можно будет сделать оверрайд виртуальных методов для реализации разных типов сохранений. |

1. **Editors**

Все функции отображаются в Unity Window>Menu>Template.



1. Configurator Window – Позволяет сконфигурировать проект если вы загрузили шаблон из юнити пакета.

|  |  |  |
| --- | --- | --- |
|  | Configure Resources | Создаёт папку Resources. В папке создаёт GameData и SaveData. После конфигурирует GameData и делает автопоиск всего. |
| Configure Graphics | Изменяет Quality и конфигурирует его под мобильные устройства. |
| Configure Folders | Создаёт дерево папок, если есть его часть то дополняет. |
| Configure All | Вызывает 3 вышеописанных метода. |
| Separate Scripts | Переносит скрипты из Assets в нужные папки. |

1. Select GameData – Пингует GameData в Project.
2. Move Scripts – Выполняет метод Separate Scripts
3. Configurator – вызывает Configurator Window
4. Multi Screenshot Capture – Вызывает окно Screenshot
5. **Менеджеры**

|  |  |  |
| --- | --- | --- |
| GameManger | | |
| Расположен на главной сцене, обращается к GameData, в зависимости от данных строит уровень, управляет уровнями и вызывает список эвентов:   * StartGame = delegate { }; * EndGame = delegate { }; * TapToPlayUI = delegate { }; | | |
| Статика | | |
| static void OnLevelStarted | Инициализирует геймлей, вызывает эвент | |
| static void OnLevelEnd | Оканчивает левел и вызывает эвент | |
| static void Restart | Рестартит | |
| static void NextLevel | Грузит следующий левел и лупит уровни. | |
| Статические свойства | | |
| GameManger instance | | **Singleton** |
| static LevelManager currentLevel { get; private set; } | | Менеджер уровня |
| static GameObject player { get; private set; } | | Текущий игрок |
| static Canvas canvas { get; private set; } | | Текущий Canvas |

|  |
| --- |
| LevelManager |
| Инициализирует уровень. Ищет штуки в нём, выполняет функции логики уровня. Должен быть на префабе каждого левела.  Получать через Singleton GameManager (GameManager.currentLevel) |