

Project Design Document

27/05/2021
Simple platformer

Project Concept

1

Player Control

You control a

Man with a gun

in this

Side view, like a platformer

game

where

As, space and lmc

makes the player

Move player, makes it jump and allow to shoot

2

Basic Gameplay

During the game,

Enemies and hp potions

appear

from

Right and top

and the goal of the game is to

Find exit

3

Sound & Effects

There will be sound effects

When player shooting

and particle effects

When player hits the enemy

[optional] There will also be

description of any other expected special effects or animation in the project.

4

Gameplay Mechanics

As the game progresses,

Player will fight with enemies and moving throught the level

making it

[optional] There will also be

There will be hp porion.

5

User Interface

The

Player hp

will

decrease

whenever

Enemy shooting or colliding with him

At the start of the game, the title

"Working title

will appear

and the game will end when

Player find exit/

6

Other Features

Any other notes about the project that you don't feel were addressed in the above.

Project Timeline

Milestone	Description	Due
#1	- Создание проекта. Поиск шаблонов для игрока, фона для задников и противников.	3/06
#2	- Размещение персонажа, реализация, перемещения между "Локациями".	10/06
#3	- Реализация хп, урона и лечения игрока	17/06
#4	- Создание генерации противников.	01/07
#5	- Проработка механики пошагового боя.	15/07
#6	- Добавление звуковых эффектов	22/07
#7	- Добавление проигрыша и выигрыша	29/07
#8	- Добавление меню	05/08
Backlog	- Feature on backlog - not a part of the minimum viable product - Feature on backlog - not a part of the minimum viable product - Feature on backlog - not a part of the minimum viable product	mm/dd

Project Sketch



В конце уровня
выход. Движение
вправо.
Новые враги
появляются
справа.



Основные классы

PlayerManager()

Здесь будет реализовано управление игроком. Чтобы показать на практике абстракцию, часть кода, отвечающая за движение персонажа будет реализована в отдельных методах.

Enemy()

Это родительский класс для нескольких видов врагов. Чтобы показать умение пользоваться наследованием и полиморфизмом, у класса будут дочерние классы. В основной класс также будет виртуальный метод для движения врагов, который будет реализован отдельно в каждом наследнике.

GameManager()

Скорее всего, синглтон. Будет использоваться для отслеживания состояния игры, проигрыша, выигрыша и хранения данных внутри сессии, а также между сессиями.

JumpPad()

Класс-родитель для различных джамп-педов.

Item()

Класс-родитель для предметов.

StartMenu()

Для работы UI главного экрана

GameUI()

Для работы UI внутри сцены с игрой.