Назва

Ігровий програмний застосунок у жанрах Roguelike та Stealth.

Керівник (product owner)

Новіков Юрій Сергійович

Проблема(и), яку вирішує програмний продукт

Можливість тренувати ігрові навички гравця для подолання ігрових перешкод. Можливість головного героя пересуватися, взаємодіяти з ігровими предметами, персонажами гри.

Анотація (мета, основні функції, складові)

Ігровий застосунок з основним механіками ігор в жанрі roguelike: хаотично генеровані рівні, необоротність програшу гравця та механізм дослідження локації; у поєднанням із механіками жанру stealth: механізм виявлення гравця та взаємодія з ігровим світом для досягнення цілі.

Основні функції:

- система генерації ігрового рівня;
- механізм виявлення гравця;
- механізм ігрового звуку: видавання головним героєм звуків та їх вплив на ігрове середовище.
- система ігрових предметів;
- система ігрових монстрів.

Додаткові функції:

- система ігрових досягнень,

Складові системи:

- клієнтська частина;
- серверна частина.

Початкові дані

- референс та диференціація;
- game design document;
- таблиці балансування.

Джерела

Системи з аналогічними функціями:

1. https://store.steampowered.com/app/588650/Dead_Cells/
2. https://store.steampowered.com/app/243970/Invisible_Inc/
3. https://store.steampowered.com/app/214490/Alien_Isolation/
4. https://apps.apple.com/us/app/king-of-thieves/id952715194

Склад команди (ролі, завдання)

У складі може бути одна особа:

1) розробка серверної частини ігрового застосунку;
2) розробка клієнтської частини ігрового застосунку;

Мають також бути виконані функції: менеджеру проєкту, гейм-дизайнера, тестувальника та ігрового аналітика.

Технології (методології)

- управління проектом - Agile. Scrum;
- програмування ігрового застосунку – Unity та C#;
- програмування серверного застосунку – C# або Node.js;
- база даних - безкоштовна з легким розгортанням

- (наприклад Google Firebase);
- тестування залежить від обраних технологій.

Критерії (метрики) успішності

Для визначення успішної реалізації проекту необхідно повністю реалізувати та протестувати наступні функції в ігровій системі:

- інтерфейс;
- система генерації рівнів;
- система виявлення гравця та можливість сховати головного героя;
- система ігрових монстрів різної складності;
- система ігрових предметів.

Графік виконання

- до 15 січня- сформована команда (розподіл ролей), затверджене GDD;
- кожні 2 тижня SCRUM зустрічі;
- до 20 лютого - створення прототипів кожних з частин;
- до 20 березня - тестування;
- до 20 квітня - апробація.

Обов’язкові умови

1. Програмний код публікується у GitHub, як публічний репозитарій відкритий для всіх. У файлі Readme має бути вказано, що “**проєкт створено як складову бакалаврської кваліфікаційної роботи студента групи … ПІБ спеціальності “Інженерія програмного забезпечення” кафедри Програмної Інженерії ХНУРЕ (https://se121.nure.ua)”.**

2. Кафедра Програмної Інженерії ХНУРЕ має право на безкоштовне використання цього програмного продукту в PR-цілях та в учбовому процесі без обмеження у часі.