

## **Dream Jam Demo**

Направление: Софтуерни приложения

**Автор:**

**Христо Симеонов Узунов**

**ЕГН:** 0543106441

**Адрес:** София, ж.к. „Студентски град“,  
ул. „Фредерик Шопен“, бл.27,ет. 12, ап. 57

**Телефон:** 0899077841

**E-mail:** [hu99549591@edu.mon.bg](mailto:hu99549591@edu.mon.bg)

**Училище:** НПМГ „Акад. Л. Чакалов“,  
гр. София

**Клас:** 10

**Ръководител:**

**Мирослава Спасова Николова**

**Телефон:** +359886877696

**E-mail:** [miroslava.nikolova@npmg.org](mailto:miroslava.nikolova@npmg.org)

**Длъжност:** Учител по информатика и ИТ

## 1. Цели и целева група

Файловете на мултимедийната игра са достъпни на адрес:

<https://github.com/3DCreativity/Dream-Jam>

### 1.1.Цели

- 1.1.1. Dream Jam Demo е игра в която се разказва за Game Developer, който участва в Game Jam. Когато той мисли каква да му бъде играта той ляга да спи с идеята, че идеята му ще дойде от съня. Но бива изненадан, когато открива много повече от просто игра там. Целта на играча е да избяга от съня като победи Бъг, който е превзел цялата игра.

### 1.2.Целева група

- Геймъри търсещи не типичен стил на игра

## 2. Основни етапи

- 2.1.Проучване за съществуващи примери по предварително избрана тема
- 2.2.Замисъл на главна идея на играта
- 2.3.Направа на проект в Unity и тестова сцена
- 2.4.Направа на базови функции и механики
- 2.5.Направа на потребителски интерфейс
- 2.6.Направа на нива
- 2.7.Тестване
- 2.8.Популяризиране

## 3. Ниво на сложност

- 3.1. При създаването на мултимедийната игра възникнаха следните проблеми:
  - 3.1.1. При направата на картата на което и да е ниво се появява проблем при използването на стандартния Tilemap Collider 2D. Collider-а бива изграден от множество Collider-и за всяко заето пространство на мрежата на картата. Това предизвиква при Квадратен Collider да възприеме дори най-малкото изместване на калкулацията при всяка една част, предизвиквайки неочаквано спиране като при удряне в стена или в случай на Кръгъл или Капсуловиден Collider – неочаквано подскачане при движение.
  - 3.1.2. При направата на базовите механики на играта някои ефекти свързани с камерата не се вграждаха правилно.
  - 3.1.3. При използването на виртуални камери се получи проблем при Parallax ефекта(ефект при който се получава чувство за дълбочина и разстояние използвайки неравномерно движение на обекти спрямо камерата) на задните обекти.

- 3.1.4. При направата на потребителски интерфейс се появиха следните проблеми:
    - 3.1.4.1. Неправилно конфигуриране на настройки
    - 3.1.4.2. Автоматична грешка при наслояване на обекти при Dropdown менюта
    - 3.1.4.3. Неправилно възприемане на свързано устройство при контролери
    - 3.1.4.4. Проблеми при вграждането на анимации в UI елементи
    - 3.1.4.5. Трудности при промяната на езика за всички UI елементи в конкретната сцена и цялата игра
  - 3.1.5. При направата на нивата бе направен опит да се сложи масова анимация на детайли от нивото, но това задави системата и накара играта да даде грешка и да се затвори.
  - 3.1.6. При направата на диалогов мениджър се появиха затруднения и проблеми с разпознаването на героя, разпознаването на името на главния герой от игровия мениджър, постепенното написване на текста, и правилното подаване на текст към мениджъра съответно със стандартен текст на различни езици.
  - 3.1.7. При прилагане на еднакви материали в нивата се синхронизират техните променливи правейки ги трудна пречка за преодоляване ако искаме да променим променливата на материала само на един обект в играта.
  - 3.1.8. При вграждане на Post Processing ефекти не се подава винаги автоматично правилната светлинна карта от спрайта към shader-a
  - 3.1.9. При смяната на нивата имаше проблем да остане на зареждащ екран докато играча не приеме превключването на друго ниво.
  - 3.1.10. При атака 1 на героя се появяваше проблем при привличането на врагове към центъра на кълбото подобно на черна дупка и изстрелване на врага във въздуха ако е прекалено близо.
- 3.2. Решени ли са проблемите?:
- 3.2.1. Решен - Unity са направили специален инструмент, който слива тези Collider-и, елиминирайки източника на проблема
  - 3.2.2. Решен – Чрез използването на добавката Cinemachine се решават доста от проблемите на камерата плюс правейки следването на героя и смяната на камери да изглежда плавно.
  - 3.2.3. Решен – При използването на Cinemachine камерата бива автоматично настроена според настройките на активната виртуална камера. Кода, който написах използва местоположението на камерата и понеже пренаписването на характеристиките на камерата става по едно и също време с кода се получава насечен ефект поради объркването на кода коя от двете локации на камерата да използва. Намерих параметър на Cinemachine, който определя в кой момент се прилагат промените и го промених.
  - 3.2.4. Решени – всички

- 3.2.5. Решен – намерих пробен инструмент на Unity, който не е излязъл официално, но е достъпен чрез GitHub, позволявайки да се добави анимация и други екстри към нормалния Tilemap
- 3.2.6. Решен
- 3.2.7. Решен – при началото на код прикрепен към обекта прави нов материал на базата на оригинала правейки го независим от останалите
- 3.2.8. Не е решен – Все още не се подава винаги правилната светлинна карта.
- 3.2.9. Решен
- 3.2.10. Частично решен – Всеки враг в определен радиус бива привлечен към източника, но ако е прекалено близо все още може да се изстреля във въздуха на учудваща височина.

#### **4. Логическо и функционално описание**

- 4.1. Архитектура – използва се Unity Game Engine за създаване на мултимедийната игра
- 4.2. Модули – играта е разделена на множество мениджъри, управляващи и съдържащи определена косвена информация – Диалогов мениджър, Игрови мениджър, Аудио мениджър, Меню Мениджър, Мениджър за настройки, и Мениджър за ниво.
- 4.3. Функции на модулите
  - 4.3.1. Диалогов Мениджър – контролира всякакъв диалог в играта плюс някои малки допълнителни следствия от определени части на диалога
  - 4.3.2. Игрови Мениджър – държи базови настройки, които са нужни за всички нива да бъдат еднакви
  - 4.3.3. Аудио мениджър – контролира всякакви звуци, които се издават по време на играта – всеки звук може да се извика чрез скрипт
  - 4.3.4. Меню Мениджър – отговаря за базовите функции на показаното меню
  - 4.3.5. Мениджър за настройки – отговаря за прилагането на настройките зададени от каквото и да е меню за цялата игра
  - 4.3.6. Мениджър за ниво – отговаря за поведението на нивото по време на игра.

#### **5. Реализация**

- 5.1. Играта е създадена чрез Unity Game Engine като един от най-известните и наскоро най-лесен за изучаване Game Engine.
- 5.2. Скриптовете за играта са направени с Visual Studio поради доброто му взаимодействие с Unity Engine
- 5.3. Paint.net и Photorea се използват за редакция на 2D графика и направата на плакат/постер на играта
- 5.4. Почти цялата 2D графиката не е авторска, но се спазват лицензите обявени от авторите, за които е отделено заслужено място в надписите на играта

5.5. Изображенията и спрайтшийте са взети от: [itch.io](https://itch.io) , [OpenGameArt.org](https://OpenGameArt.org), [CraftPix.net](https://CraftPix.net), и [Pixabay](https://Pixabay);

5.6. За запазване на всички файлове в онлайн пространството се използва GitHub

5.7. Презентацията е направена с MS PowerPoint

## **6. Описание**

6.1. Мултимедийната игра може да се стартира на платформа Windows чрез файла „Dream Jam.exe“ от файловете на играта

6.2. Играта няма инсталатор към себе си – всички файлове се намират папка „Build“ в GitHub. Тя трябва съответно да се свали и играта може да бъде пусната.

## **7. Заключение**

7.1. Dream Jam Demo е кратка пиксел арт 2D платформер игра с динамична история, множество опции за продължение и скрити детайли от значение за историята в бъдеще.

7.2. Играта успява успешно да научи играча на контролите чрез предизвикателства тествайки новите знания.

7.3. Играта има за момента много опции за подобрене и развитие, които се разработват паралелно с останалата част от играта.