

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧОРНОМОРСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ
ПЕТРА МОГИЛИ

ЧЕРНЕНКО ОЛЕКСАНДР ЮРІЙОВИЧ

УДК 004.8

**РОЗРОБКА КОМП'ЮТЕРНОЇ ГРИ НА БАЗІ
КРОСПЛАТФОРМНОГО СЕРЕДОВИЩА UNITY 3D**

Галузь знань 12 «Інформаційні технології» за спеціальністю

122 «Комп'ютерні науки»

122 - БКР.А - 401.21710130

Автореферат

бакалаврської кваліфікаційної роботи на здобуття освітньої
кваліфікації

«бакалавр з комп'ютерних наук»

Миколаїв – 2021

Бакалаврська кваліфікаційна робота є рукопис.

Робота виконана в Чорноморському національному університеті імені Петра Могили Міністерства освіти і науки України на кафедрі інтелектуальних інформаційних систем

Науковий керівник:

старший викладач.

С.В. Дворецька

Рецензент:

к.т.н., доцент, доцент кафедри ПЗ

К.О. Кірей

Захист відбудеться «_25_» червня 2021 р. о ___год. на засіданні екзаменаційної комісії (ауд. 2-403) у Чорноморському національному університеті імені Петра Могили за адресою: 54003, м. Миколаїв, вул. 68-ми Десантників, 10.

З бакалаврською кваліфікаційною роботою можна ознайомитися в бібліотеці Чорноморського національного університету імені Петра Могили за адресою: 54003, м. Миколаїв, вул. 68-ми Десантників, 10.

Автореферат представлений «___» червня 2021 р.

Секретар

екзаменаційної комісії,

викладач кафедри ІС

М.О. Таранов

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність теми. Тема бакалаврської кваліфікаційної роботи є актуальною, оскільки напряму пов'язана з одним із найпопулярніших на сьогоднішній день виду проведення вільного часу, а саме ігровими додатками.

Метою бакалаврської кваліфікаційної роботи. Збільшення кількості альтернатив цікавого проведення вільного часу шляхом розробки кросплатформної 3D комп'ютерної гри «Castle Defense», що є колаборацію жанрів Tower Defense і стратегії.

Практичне значення отриманих результатів. Під час виконання бакалаврської кваліфікаційної роботи було розроблено кросплатформенний ігровий додаток «Castle Defense».

Структура кваліфікаційної роботи. Пояснювальна записка до бакалаврської кваліфікаційної складається із вступу, 4 розділів, висновків, додатків. Загальний обсяг роботи складає сторінок, рисунків, таблиць та посилань на літературні джерела.

ОСНОВНИЙ ЗМІСТ РОБОТИ

В сучасному світі кожного дня людину оточує безліч різноманітного програмного забезпечення, починаючи від серйозних технічних рішень і закінчуючи соціальними мережами і іграми.

Із підвищенням доступності різного роду девайсів, таких як смартфони, персональні комп'ютери, ігрові приставки, тощо, рівень зацікавленості населення в іграх росте шокуючими темпами.

Перший розділ.

Відеоігра - це електронна гра, яка передбачає взаємодію з користувальницьким інтерфейсом або пристроєм введення - наприклад, джойстиком, контролером, клавіатурою або пристроєм, що зчитує рух - для створення візуального зворотного зв'язку для гравця. Цей відгук відображається на пристрої для відображення відео, наприклад на телевізорі, моніторі, сенсорному екрані або гарнітурі віртуальної реальності. Відеоігри часто доповнюються звуковим зворотним зв'язком, що надходить через динаміки або навушники, а іноді й іншими типами зворотного зв'язку, включаючи тактильні технології.

Відеоігри визначаються на основі їх платформи, яка включає аркадні ігри, консольні ігри для ПК. Зовсім недавно галузь розширилася на мобільні ігри за допомогою смартфонів та планшетних комп'ютерів, систем віртуальної та доповненої реальності та віддалених хмарних ігор. Відеоігри класифікуються на широкий діапазон жанрів залежно від їх типу гри та призначення.

З появою смартфонів та планшетних комп'ютерів, стандартизованих для операційних систем iOS та Android, мобільні ігри стали важливою платформою. Ці ігри можуть використовувати унікальні функції мобільних пристроїв, які не є необхідними на інших платформах, такі як акселерометри, інформація про глобальне розміщення та камери для підтримки геймплею доповненої реальності.

Захист башти (TD) - це піджанр стратегічних ігор, метою яких є захист територій або володінь гравця шляхом перешкоджання ворожим нападникам або зупинки ворогів до виходу, як правило, це досягається розміщенням захисних споруд на або вздовж їх шляху атаки. [1] Зазвичай це означає будівництво різноманітних структур, які служать для автоматичного блокування, перешкоджання, нападу або знищення ворогів. Захист вежі розглядається як піджанр стратегічних відеоігор у реальному часі завдяки їх походженням у реальному часі [2, 3], хоча багато сучасних ігор оборони включають аспекти покрокової стратегії. Стратегічний вибір та позиціонування елементів оборони є важливою стратегією жанру.

До кінця буму більшість ігор на захист башти все ще застрягли в боковому прокручуванні, ізометричному або графічному носії зверху вниз. Iron Grip: Warlord, випущений в листопаді 2008 року, безуспішно став піонером шутера від першої особи в жанрі. [25] Незручне поєднання експериментальної механіки оборони башти з тривимірною графікою було сприйнято не дуже добре, але пізніші автори вдосконалили її виконання, відкривши шлях для популярної нової породи ігор. Dungeon Defenders, випущена в жовтні 2010 року, була однією з перших ігор для захисту башти, яка представила жанр у перспективі від третьої особи. За перші два тижні випуску було продано понад 250 000 примірників [26], а до кінця 2011 року - понад 600 000 примірників [27]. У 2010 році вийшли SteamWorld Tower Defense, Protect Me Knight, The Tales of Bearsworth Manor, Revenge of the Titans, Arrow of Laputa, Toy Soldiers і Robocalypse: Defense of Beaver.

Стратегічна відеоігра - жанр відеоігор, що фокусується на вмілому мисленні та плануванні досягнення перемоги. [1] Він підкреслює стратегічні, тактичні та інколи логістичні завдання. Багато ігор також пропонують економічні проблеми та дослідження. Зазвичай їх класифікують на чотири підтипи, залежно від того, покрокова гра чи в режимі реального часу, і чи орієнтована гра на стратегію чи тактику.

Стратегічні відеоігри - жанр відеоігор, що підкреслюють вміле мислення та планування досягнення перемоги. [1] Зокрема, гравець повинен спланувати

низку дій проти одного або кількох супротивників, і зменшення сили ворога, як правило, є метою. Перемога досягається завдяки чудовому плануванню, а елемент випадковості відіграє меншу роль. [2] У більшості стратегічних відеоігор гравцеві дається богоподібний погляд на ігровий світ і опосередковано керує ігровими одиницями під їх команду. [1] Таким чином, більшість стратегічних ігор включають елементи війни в різній мірі [2] і містять поєднання тактичних та стратегічних міркувань. [3] На додаток до бою, ці ігри часто кидають виклик здатності гравця досліджувати економіку чи керувати нею. [2]

Другий розділ.

C # - це мова програмування багатопарадигми загального призначення, що охоплює статичну типізацію, сильну типізацію, лексичну область дії, імперативну, декларативну, функціональну, загальну, об'єктно-орієнтовану (на основі класів) та компонентну дисципліни орієнтованого програмування.

C # був розроблений Андерсом Хейлсбергом, а його командою розробників в даний час керує Мадс Торгерсен. Остання версія - 8.0, випущена в 2019 році разом з Visual Studio 2019 версії 16.3.

Microsoft Visual Studio - це інтегроване середовище розробки (IDE) від Microsoft. Він використовується для розробки комп'ютерних програм, а також веб-сайтів, веб-додатків, веб-служб та мобільних додатків. Visual Studio використовує платформи Microsoft для розробки програмного забезпечення, такі як API Windows, Windows Forms, Foundation Presentation Foundation, Windows Store та Microsoft Silverlight. Він може створювати як власний код, так і керований код.

Unity - це крос-платформний ігровий двигун, розроблений Unity Technologies, вперше анонсований і випущений в червні 2005 року на всесвітній конференції розробників Apple Inc. як ексклюзивний ігровий движок для Mac OS X. Станом на 2018 рік двигун був розширений для підтримки більш ніж 25 платформ. Механізм може бути використаний для створення тривимірних, двовимірних, ігор у віртуальну реальність та доповнену реальність, а також моделювання та іншого досвіду. Двигун був

прийнятий на виробництво за межами відеоігор, таких як кіно, автомобілебудування, архітектура, машинобудування та будівництво.

Unity - це крос-платформний двигун. Редактор Unity підтримується на Windows, macOS та платформі Linux, тоді як сам движок в даний час підтримує створення ігор для більш ніж 25 різних платформ, включаючи мобільні, настільні, консольні та віртуальну реальність. Платформи включають iOS, Android, Tizen, Windows, Універсальну платформу Windows, Mac, Linux, WebGL, PlayStation 4, PlayStation Vita, Xbox One, 3DS, Oculus Rift, Google Cardboard, Steam VR, PlayStation VR, Gear VR, Windows Mixed Reality, Daydream, Android TV, Samsung Smart TV, tvOS, Nintendo Switch, Fire OS, Facebook Gameroom, Apple ARKit, Google ARCore, Vuforia, і Magic Leap.

Unity надає користувачам можливість створювати ігри як у 2D, так і в 3D, а движок пропонує основний API сценаріїв на C# як для редактора Unity у формі плагінів, так і для самих ігор, а також функцію перетягування. До того, як C# був основною мовою програмування, що використовувалася для движка, він раніше підтримував Boo, який був видалений випуском Unity 5 та версію JavaScript під назвою UnityScript, яка була припинена в серпні 2017 року після випуску Unity 2017.1, на користь C#.

Третій розділ.

Діаграма використання найпростіша - це представлення взаємодії користувача із системою, яка показує взаємозв'язок між користувачем та різними випадками використання, в яких користувач бере участь. Діаграма випадків використання може ідентифікувати різні типи користувачів системи та різні випадки використання, і часто вона супроводжується також іншими типами діаграм. Варіанти використання представлені кругами або еліпсами.

Незважаючи на те, що сам випадок використання може детально вивчити кожну можливість, діаграма прикладів використання може допомогти забезпечити огляд системи на більш високому рівні. Раніше вже було сказано, що "схеми використання - це принципи вашої системи".

Через їх спрощений характер, схеми використання можуть бути хорошим інструментом комунікації для зацікавлених сторін. Креслення

намагаються імітувати реальний світ і дають зацікавленій стороні уявлення про те, як буде розроблена система. Сіау та Лі провели дослідження, щоб визначити, чи взагалі існувала дійсна ситуація для схем використання або вони були непотрібними. Було виявлено, що діаграми випадків використання передають намір системи більш спрощеним чином зацікавленим сторонам і що вони "інтерпретуються більш повно, ніж діаграми класів".

Елементи:

- рамки системи (англ. system border) - прямокутник із назвою у верхніх частинах та еліпсами (прецедентами) всередині. Часто може бути опущено без корисної інформації про полезну інформацію,
- актор (англ. actor) - стилізований людський персонаж, обзначаючий набір ролей користувача (розуміється в широкому змісті: людина, зовнішня сутність, клас, інша система), взаємодіючого з деякою сутністю (системною, підсистемою, класом). Актори не можуть бути пов'язані між собою з іншим (за вимкнення відносин щодо обробки / дослідження),
- прецедент - еліпс із надписом, що означає виконувану систематичну дію (може включати можливі варіанти), що призводить до спостережуваних акторами результатів. Надпис може бути ім'ям або описом (з точки зору актора) того, "що" робить система (а не "як"). Ім прецедента зв'язано з неперервним (атомарним) сценарієм - конкретною послідовністю дій, ілюструючою повіддю. Під час сценарію актори обмінюються із систематичними повідомленнями. Сценарій може бути приведений на діаграмі прецедентів у відео UML-коментарі. З одним прецедентом може бути пов'язано кілька різних сценаріїв

У програмній інженерії діаграма класів в Уніфікованій мові моделювання (UML) - це тип статичної структурної діаграми, що описує структуру системи, показуючи класи системи, їх атрибути, операції (або методи) та взаємозв'язки між об'єктами.

Діаграма класів є основним будівельним елементом об'єктно-орієнтованого моделювання. Він використовується для загального концептуального моделювання структури програми та для детального

моделювання переведення моделей у програмовий код. Діаграми класів також можуть бути використані для моделювання даних. Класи на діаграмі класів представляють як основні елементи, взаємодії в програмі, так і класи, що програмуються.

При заході в гру відкривається головне меню (грає фонова музика, музика всієї гри). На сцені меню є 3 кнопки яким керує скрипт `MainMenuController`. Кнопка вихід, закриває додаток. Кнопка налаштування, відкриває нове вікно яким керує скрипт `MusicController`, який дозволяє поставити прапорці на звук музики і звук ефектів при активних прапорцях є звуки, при неактивних — їх немає. Окрім є два слайдера якими можна налаштувати гучність. Є кнопка скинути рівні якою управляє `MainMenuController`, яка скидає наше проходження рівнів. При натисканні на кнопку грати ми переходимо в сцену вибору рівня, в якій є скрипт `LevelSelector` який управляє логікою відкриття і закриття рівнів в залежності від того пройдені вони чи ні. Ще він керує кнопкою «назад», яка переходить назад в головне меню. Кожен рівень це кнопка яка зовні виглядає як зруйнований або цілий замок, що символізує пройдений рівень і чи ні. При вдалому проходженні рівня і перемозі при запуску сцени вибору рівня `LevelSelector` відкриває доступ до нових рівнів. Вибравши рівень ми переходимо на основну ігрову сцену, в кожному рівні є своя сцена, яка відрізняється тільки наявністю різного ландшафту і патрульних юнітів ворога.

На сцені ігрового рівня видно в лівому нижньому кутку кнопки покупки наших юнітів ними керує скрипт `GameController`, він керує всіма кнопками і майже всією грою. Він зберігає в собі інформацію про всіх юнітів на сцені, про параметри сили здібностей, інформацію про управління золотом, поточних юнітів, які є на сцені, він в собі має посилання майже на всі об'єкти, управляє смугою магії.

Мета гри знищити замок ворога і не втратити свій. Замками управляє скрипт `Castle` який успадковується від `Building`, по кнопці зліва знизу екрану. Біля замків з'являється кнопки апгрейда і будівництва башт замку, якими

управляє скрипт TowerBuilding. На замках як і на юнітах є HealthScr який керує їхнім здоров'ям і відображає в UI отримання шкоди або лікування.

У лівому верхньому кутку смуга здоров'я нашого замку, 3 види магії, на які витрачається енергія при заповненій смузі енергії. Перша здатність наносить шкоду по області, друга здатність це лікування, а третя здібність це лікування вашого замку. Ще UI відображає кількість золота і кількість дружніх юнітів. Виклик магії управляється гейм контролером, а сама магія скриптом Magic.

У правому верхньому куті видно кнопку паузи і здоров'я ворожого замку. При відкритті кнопки паузи гра зупиняється на паузу, щоб вийти з режиму паузи потрібно натиснути на цю ж кнопку. У меню паузи ми можемо налаштувати звуки гри, і зробити рестарт рівня або вийти в меню вибору рівнів.

Графічний інтерфейс користувача складається з основних елементів, а саме:

- головне меню;
- меню налаштувань;
- меню рівнів;
- ігрова сцена;
- ігрове меню.

Для початку гри користувачеві слід відкрити додаток і натиснути кнопку «Играть». Для змін параметрів гучності звуків і музики необхідно натиснути кнопку «Настройки» і змінити відповідні налаштування.

Після переходу на сцену вибору рівня слід натиснути на необхідний рівень і очікувати переходу до ігрової сцени.

Для створення нового юніта слід натиснути на кнопку відповідного юніта внизу екрану. Для використання магичних навичок слід натиснути на одну з навичок на панелі у верхній лівій частині екрану.

Для відкриття меню паузи слід натиснути на кнопку у правому верхньому куті. Для виходу до меню вибору рівня слід в меню паузи натиснути кнопку «Назад».

Розділ з охорони праці.

У даному розділі роботи розглянути питання охорони праці у ФОП Ракоєд М.Е., виконана інтегральна оцінка умов праці та запропоновані заходи, спрямовані на їх покращення.

Приміщення відділення Зд друку розташовано на першому поверсі чотирьох поверхової будівлі, що знаходиться у м. Миколаєві за адресою вулиця Московська, 9, Миколаїв, Миколаївська область, 54000. Розміри приміщення складають $a \times b \times H = 6,0 \times 3,0 \times 2,5$ м. У приміщенні влаштовано металопластикове вікно розмірами $c \times d = 1,8 \times 2,2$ м.

Приміщення розташовано у будівлі, що знаходиться у м. Миколаєві за адресою вулиця Янтарна, 233, Миколаїв, Миколаївська область, 54050. Розміри приміщення складають $a \times b \times H = 5,0 \times 8,0 \times 2,5$ м. У приміщенні влаштовано два металопластикових вікна (з потрійними склопакетами) розмірами $c \times d = 1 \times 1,5$ м.

Приміщення має сучасний офісний інтер'єр. Стеля виконана у вигляді підвісної конструкції із синтетичного матеріалу стилізованого під дерево. Стіни мають гладку поверхню світло-сірого кольору. Підлога являє собою бетонну плиту, покриту панелями, що імітують дерево.

У приміщенні розташовано 3 робочих місця, обладнаних сучасними персональними комп'ютерами з необхідними периферійними пристроями, з умовою наявності моніторів у кількості 2-х штук на кожне робоче місце. Для зберігання робочої документації передбачені шафи та полиці.

На основі існуючих даних проведена **інтегральна оцінка умов праці в обраному виробничому приміщенні**. Здійснена оцінка питомої ваги кожного із представлених факторів виробничого середовища та трудового процесу. Також здійснена оцінка ефективності заходів щодо покращення умов праці. Проведений аналіз показав що заходи можуть призвести до підвищення продуктивності праці співробітників відділу на 7,2 %. Тобто основна мета роботи була досягнута.

ЗАГАЛЬНІ ВИСНОВКИ

Метою роботи було збільшення кількості альтернатив цікавого проведення вільного часу шляхом розробки кросплатформної 3D комп'ютерної гри «Castle Defense», що є колаборацію жанрів Tower Defense і стратегії.

Для виконання поставленої задачі було виконано наступні завдання:

1. Провести огляд предметної області.
2. Обрати засоби реалізації програмного забезпечення.
3. Розробити архітектуру ПЗ.
4. Виконати реалізацію комп'ютерної гри.

В першому розділі роботи було проведено огляд предметної, тобто проаналізовано поняття відеогри, виявлено жанрові особливості майбутньої гри. Вся зібрана інформація допомогла отримати загальне розуміння предметної області і в подальшому лягла в основу розробки власного ігрового додатку.

В другому розділі проведено вибір засобів реалізації і їх опис. В результаті було виявлено, що найбільш підходящою для поставлених задач є мова програмування C# у зв'язці зі спеціалізованим інтегрованим середовищем розробки Visual Studio і ігровим двигуном Unity.

Третій розділ роботи присвячений проектуванню і реалізації програмного продукту. В першому підрозділі проведено проектування програмного продукту за допомогою побудови варіантів використання системи, яка допоможе визначитися з можливим функціоналом, та діаграми класів додатку, яка дасть вичерпне розуміння внутрішньої структури розроблюваного програмного забезпечення. Окрім цього розроблено графічний інтерфейс користувача, для взаємодії кінцевого користувача з програмним продуктом, написано керівництво користувача, описано основні механіки та створено керівництво користувача.

Завдяки чіткому виконанню завдань, поставлених на початку роботи, в результаті було отримано повноцінний програмний продукт, здатний виконувати закладений в нього функціонал і готовий до використання в реальних умовах.

АНОТАЦІЯ

Черненко Олександр Юрійович. РОЗРОБКА КОМП'ЮТЕРНОЇ ГРИ НА БАЗІ КРОСПЛАТФОРМНОГО СЕРЕДОВИЩА UNITY 3D. – На правах рукопису.

Бакалаврська кваліфікаційна робота на здобуття освітньої кваліфікації «бакалавр з комп'ютерних наук» в галузі знань 12 «Інформаційні технології» за спеціальністю 122 «Комп'ютерні науки».

Чорноморський національний університет імені Петра Могили, Миколаїв.

Об'єктом роботи є процес розробки комп'ютерних ігор у середовищі розробки UNITY.

В свою чергу предметом роботи є інструменти та засоби розробки комп'ютерних 3D ігор із використанням рушія UNITY.

Мета роботи - збільшення кількості альтернатив цікавого проведення вільного часу шляхом розробки кроссплатформної 3D комп'ютерної гри «Castle Defense», що є колаборацією жанрів Tower Defense і стратегії.

Робота складається з фахового розділу і спеціальної частини з охорони праці. Пояснювальна записка складається зі вступу, чотирьох розділів, висновків та додатків.

У першому розділі розглядається і описуються теоретико-методологічні основи дослідження, поняття та зміст відеоігор, методи і особливості створення ігрових додатків, особливості жанрів стратегії і Tower Defense, завдяки чому було сформовано загальне розуміння предметної області і виявлено жанрові особливості майбутнього додатку.

У другому розділі проведено вибір інструментарію розробки, тобто вибрано мову програмування, середовище розробки, та ігровий двигун, аналіз відомостей про які сприяв побудові загальної картини підходу до розробки додатку.

У третьому розділі проведено проектування системи за допомогою діаграм та, власне, розробка інформаційної системи; висновків, списку використаних джерел та додатків.

В результаті власний ігровий . власного програмний продукт «Castle Defense», який представляє собою колаборацію жанрів Tower Defense і стратегії.

Бакалаврська кваліфікаційна робота містить 72 сторінок, 7 рисунків, 4 таблиці, 30 використаних джерел та 2 додатки.

До ключових слів належать наступні: стратегія, Tower Defense, відеогра, Unity, C#, Visual Studio.

ABSTRACT

Chernenko Alexander Yurievich. DEVELOPMENT OF A COMPUTER GAME BASED ON THE CROSSPLATFORM ENVIRONMENT UNITY 3D. - On the rights of the manuscript.

Bachelor's qualification work for the educational qualification "Bachelor of Computer Science" in the field of knowledge 12 "Information Technology" in the specialty 122 "Computer Science".

Petro Mohyla Black Sea National University, Mykolaiv.

The object of work is the process of developing computer games in the UNITY development environment.

In turn, the subject of the work are tools and tools for developing computer 3D games using the UNITY engine.

The purpose of the work is to increase the number of alternatives for interesting leisure time by developing a cross-platform 3D computer game "Castle Defense", which is a collaboration of Tower Defense genres and strategy.

The work consists of a professional section and a special part on labor protection. The explanatory note consists of an introduction, four sections, conclusions and appendices.

The first section discusses and describes the theoretical and methodological foundations of research, the concept and content of video games, methods and features of creating game applications, features of strategy genres and Tower Defense, which formed a general understanding of the subject area and identified genre features of the future application.

The second section selects the development tools, ie selected the programming language, development environment, and game engine, the analysis of information about which contributed to the construction of a general picture of the approach to application development.

In the third section, the system is designed using diagrams and, in fact, the development of an information system; conclusions, list of sources and appendices used.

As a result, own gaming. Castle Defense's own software product, which is a collaboration of Tower Defense genres and strategies.

The bachelor's thesis contains 72 pages, 7 figures, 4 tables, 30 sources used and 2 appendices.

Keywords include Strategy, Tower Defense, Video Game, Unity, C #, Visual Studio.