

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»

Кафедра прикладної математики та інформатики

МЕТОДИЧНІ ВКАЗІВКИ

до виконання курсового проекту з дисципліни

«Архітектура та проектування програмного забезпечення»

для студентів спеціальності

121 Інженерія програмного забезпечення

Покровськ, 2020

УДК 004.2:004.41:004.05](072)

М 54

Методичні вказівки до виконання курсового проекту з дисципліни «Архітектура та проектування програмного забезпечення» для студентів спеціальності 121 Інженерія програмного забезпечення [Електронний ресурс] / уклад. Н.О. Маслова., І.В. Ярош. – Покровськ : ДонНТУ, 2020. – 40 с.

Методичні вказівки складено згідно робочої програми дисципліни «Архітектура та проектування програмного забезпечення». У вказівки включено теоретичний матеріал, необхідний для виконання та оформлення курсового проекту, стадії та етапи розробки, вимоги до застосування програмних та технічних засобів, правила відображення діаграм проекту та оформлення пояснювальної записки, варіанти завдань до виконання. Призначені для студентів спеціальності 121 Інженерія програмного забезпечення.

Укладачі:

Н.О. Маслова, доцент, к.т.н., доцент кафедри ПМІ;

І.В. Ярош, старший викладач кафедри ПМІ

Відповідальний за випуск: О.А. Дмитрієва, зав. каф. прикладної математики та інформатики

Рецензент: Ковальов С. О., директор ННІ КНТ, доцент

Затверджено навчально-методичним відділом ДВНЗ ДонНТУ,
протокол № 3 від «24» листопада 2020 р.

Розглянуто на засіданні кафедри прикладної математики та інформатики,
протокол № 13 від 23 листопада 2020 р.

© Донецький національний технічний університет, 2020

ЗМІСТ

Вступ.....	4
1 Підстави для розробки. Призначення проекту.....	5
1.1 Підстави для розробки.....	5
1.2 Призначення проекту.....	5
2 Вимоги до технічного та програмного забезпечення.....	6
2.1 Вимоги до складу та параметрів технічних засобів	6
2.2 Вимоги до надійності.....	6
2.3 Завдання до виконання проекту	6
3 Порядок організації роботи студента.....	10
4 Стадії і етапи розробки.....	11
5 Оформлення пояснювальної записки.....	12
6 Рекомендований зміст пояснювальної записки	15
7 Порядок контролю та приймання.....	16
8 Варіанти завдань	17
Список рекомендованої літератури.....	18
Додаток А Приклади оформлення титульних листів	22
Додаток Б Завдання на курсовий проект	24
Додаток В Обрання інструментарію для побудови UML-діаграм.....	26
1. Cасоо.....	27
2 Creately	28
3. Draw.io	30
4 EdrawSoft.....	32
5. Gliffy	33
6. LucidChart.....	35
7. Rational Rose	36
8. Umbrello.....	38
9. UMLet	40

ВСТУП

Основною метою розробки курсового проекту з дисципліни «Архітектура та проектування програмного забезпечення» є відточування практичних навичок з проектування програмних продуктів і систем, опанування сучасними методами та підходами відображення і опису архітектури створюваного або існуючого програмного забезпечення.

Об'єктом дослідження курсового проекту є існуючий програмний продукт, комп'ютерна online-гра, із зручним інтерфейсом, розвиненим сценарієм, набором різноманітних режимів, атрибутів і акторів, які в сукупності створюють унікальне архітектурне середовище.

Метою роботи є аналіз архітектури обраного відповідно до варіанту програмного продукту (комп'ютерної online-гри) та відображення її особливостей з використанням спеціалізованої мови проектування UML.

Основними завданнями курсового проекту є:

- отримання практичних навичок з виявлення архітектурних принципів побудови програмних продуктів;
- формування необхідних вмінь опису програмних систем з використанням мови проектування UML.

Результатами виконання курсового проекту виступають:

- опис архітектурних і функціональних особливостей програмного продукту або системи;
- відображення виявлених особливостей і режимів роботи за допомогою UML-діаграм;
- надання пояснень щодо побудованих діаграм і їх застосування.

При успішному виконанні курсового проекту студент буде вміти формулювати вимоги до програмного продукту (комп'ютерної гри), виділяти ігрових і неігрових персонажів, визначати їх функції і взаємодію, елементи фізичного розміщення компонентів, будувати UML-діаграми різного виду.

1 ПІДСТАВИ ДЛЯ РОЗРОБКИ. ПРИЗНАЧЕННЯ ПРОЕКТУ

1.1 Підстави для розробки

Курсовий проект виконується на підставі навчального плану підготовки студентів за освітнім ступенем «бакалавр», згідно робочої програми та «Завдання до виконання курсового проекту» за дисципліною «Архітектура та проектування програмного забезпечення» для студентів спеціальності 121 Інженерія програмного забезпечення.

1.2 Призначення проекту

Проект орієнтований на придбання студентами практичних навичок опису архітектурних особливостей програмного продукту, множина характеристик і атрибутів якого повинна відповідати стандарту якості програмного забезпечення ISO/IEC 25010.

В ході аналізу архітектурних особливостей проекту студенти застосовують спеціалізовану мову моделювання UML, створюють UML-діаграми різного типу (варіантів використання, розгортання, поведінки, взаємодії), знайомляться з загальними принципами побудови архітектури комп'ютерних ігрових додатків.

Основні навички та вміння, отримувані студентами в результаті виконання курсового проекту:

- проведення передпроектного обстеження предметної області, системний аналіз об'єкта проектування;
- застосування на практиці ефективних підходів щодо проектування програмного забезпечення;
- застосування на практиці інструментальних програмних засобів для аналізу, проектування, тестування, візуалізації та документування програмного забезпечення.

2 ВИМОГИ ДО ТЕХНІЧНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Вимоги до складу та параметрів технічних засобів

Для виконання курсового проекту необхідний персональний комп'ютер з набором технічних і системних засобів, перелік і конфігурація яких відповідає вимогам програмного продукту, що досліджується в процесі виконання курсового проекту.

2.2 Вимоги до надійності

В процесі виконання роботи застосовуються відкриті програмні системи проектування програмного забезпечення. Зокрема, онлайн-редактори для відображення схем, графіків і створення UML-діаграм різного типу.

У якості досліджуваного програмного продукту (системи) пропонується обрати одну з відомих (розповсюджених на ринку) комп'ютерних online-ігор.

Програмний продукт, що застосовується в процесі дослідження, повинен:

- бути доступним в мережі Інтернет;
- мати зрозумілий інтерфейс і розвинений сценарій;
- включати набір різноманітних режимів, атрибутів, характеристик;
- передбачати наявність декількох діючих персонажів і акторів;
- забезпечувати вирішення завдань, які сформульовані до виконання курсового проекту.

2.3 Завдання до виконання проекту

Основними аспектами, які реалізовані в проекті, повинні бути:

1) загальна інформація щодо об'єкту проектування, з описом теоретичних основ процесу проектування, а також функціональних особливостей комп'ютерного ігрового додатку;

- 2) відображення та опис функціональних можливостей комп'ютерного додатку у вигляді UML-діаграми варіантів використання;
- 3) відображення та опис топології і фізичної структури комп'ютерного додатку у вигляді UML-діаграми розміщення;
- 4) аналіз і опис поведінкових аспектів основних об'єктів комп'ютерного додатку за допомогою діаграм станів;
- 5) аналіз і опис поведінкових аспектів процесу виконання операцій комп'ютерного додатку за допомогою діаграм діяльності;
- 6) аналіз і опис часової взаємодії складових об'єктів проектованого комп'ютерного ігрового додатку з застосуванням діаграм послідовності;
- 7) аналіз та опис взаємодії складових об'єктів у загальному контексті проектованого комп'ютерного ігрового додатку з застосуванням діаграм кооперації;
- 8) демонстраційні матеріали для презентації результатів проведених досліджень.

При наданні загальної інформації щодо об'єкту проектування студент повинен надати короткі теоретичні відомості з основ архітектури та принципів проектування програмних продуктів й, опираючись на правила гри, вказати мету гри, режими, які реалізовано в проекті, виділити типи користувачів (акторів), описати їх функції.

При демонстрації функціональних можливостей програмного продукту у вигляді UML-діаграми варіантів використання необхідно відобразити основні елементи: акторів, варіанти використання, коментарі, відношення між елементами.

При створенні діаграми розміщення необхідно відобразити фізичні взаємозв'язки між програмними та апаратними компонентами системи, застосовуючи основні елементи діаграм цього виду – вузли та з'єднання.

Діаграми станів необхідно використовувати для опису поведінки як окремих екземплярів об'єктів, так і варіантів використання, акторів, підсистем, операцій і методів. Для повноцінного опису системи діаграм станів може бути декілька.

Діаграми діяльності необхідно використовувати для моделювання процесу виконання операцій, виступають доповненням діаграм станів.

При побудові діаграми послідовності та кооперації необхідно визначати персонажів гри, проаналізувати їх поведінку, визначати їх властивості. При цьому на діаграмах послідовності відображається динаміка взаємодії об'єктів в часі. Діаграми кооперації застосовуються для подання структурних особливостей передачі та прийому повідомлень між об'єктами.

Матеріали для демонстрації результатів проведених досліджень готуються у вигляді презентації та пояснюючої її доповіді.

У презентації необхідно передбачити наступну структуру, що забезпечує виконання вимог до слайдів:

- титульний слайд, що містить інформацію про назву дисципліни, найменування роботи, її тематику, відомості розробника;
- слайд 1 – слайд, що містить вказівку варіанту завдання, об'єкту дослідження, мети та завдань роботи, мови для виконання проектування;
- слайд 2 – слайд, що містить результати передпроектного дослідження предметної області згідно варіанту завдання, підсумки аналізу об'єкта проектування;
- слайд 3 – слайд, що містить обґрунтування вибору комп'ютерного засобу для забезпечення процесу розробки архітектури та проектування комп'ютерного ігрового додатку (з підтримкою нотації UML);
- слайд 4 – слайд, що містить опис функціонального призначення проєктованого комп'ютерного ігрового додатку у вигляді розробленої діаграми варіантів використання (use case diagram);
- слайд 5 – слайд, що містить опис фізичної організації проєктованого комп'ютерного ігрового додатку у вигляді розробленої діаграми розміщення (deployment diagram);
- слайди 6-9 – слайди, що містить опис поведінкових аспектів основних об'єктів проєктованого комп'ютерного ігрового додатку у вигляді розроблених діаграм станів (State Machine diagram) і діаграм діяльності (activity diagram);

- слайди 10-11 – слайди, що містять опис аспектів взаємодії складових об'єктів проєктованого комп'ютерного ігрового додатку у вигляді розроблених діаграм послідовності (sequence diagram) і діаграм кооперації (collaboration diagram);

- слайд 12 – слайд, що містить підсумки виконаної роботи з розробки архітектури та проєктування комп'ютерного ігрового додатку, з вказівкою отриманих результатів і пропозицій/рекомендацій щодо шляхів покращення та оптимізації проєкту (за наявності).

Загальний обсяг презентації не повинен перевищувати 12-15 слайдів і повністю розкривати сутність розробленого проєкту.

З ПОРЯДОК ОРГАНІЗАЦІЇ РОБОТИ СТУДЕНТА

Студент спільно з керівником формулює тему курсового проекту на початку курсового проектування та обирає комп'ютерну online гру, архітектурні особливості якої будуть аналізуватися та описуватися в процесі виконання завдань проектування.

Після затвердження теми заміна її не допускається. Рекомендований перелік програмних продуктів (комп'ютерних ігор) для проектування та аналізу наведено у розділі 8.

Обравши об'єкт проектування, студент разом з керівником складає календарний план, який служить в подальшому документом для контролю та координації робіт з їх виконання. Перелік етапів курсового проектування наведено у розділі 4.

В ході роботи студент вивчає інформацію по заданій тематиці, інструментальну базу, стандарти, робить помітки, виписки, складає необхідні діаграми, komponує матеріали розділів, формує додатки.

Керівник робить позначки про виконану роботу (для денної форми навчання атестації, у тому числі з курсового проектування, як правило, проходять на 8 і 16 тижнях семестру).

Результатом завершального етапу курсового проектування є сформована пояснювальна записка до курсового проекту та презентація для її захисту.

Пояснювальну записку студент представляє на попередній розгляд не пізніше, ніж за тиждень до дати захисту для визначення ступеня підготованості роботи та її відповідності вимогам.

Після попереднього захисту студент усуває наявні недоліки, пов'язані з проведенням аналізом, відображеними діаграмами, оформленням або змістом пояснювальної записки.

Захист курсового проекту проводиться перед комісією, до складу якої входить не менше двох викладачів.

4 СТАДІЇ І ЕТАПИ РОЗРОБКИ

Курсовий проект рекомендовано виконувати згідно графіку, що наведений нижче у таблиці 4.1.

Таблиця 4.1 – Графік виконання курсового проекту

№ з/п	Назва етапів курсового проекту	Терміни виконання
1	Знайомство з програмним продуктом для забезпечення процесу проектування	1-2-й тиждень
2	Створення діаграми варіантів використання	3-4-й тиждень
3	Розробка діаграми розміщення	5-6-й тиждень
4	Розробка діаграм станів	7-8-й тиждень
5	Розробка діаграм діяльності	9-10-й тиждень
6	Розробка діаграм послідовності	11-12-й тиждень
7	Розробка діаграм кооперації	13-14-й тиждень
8	Оформлення пояснювальної записки	15-й тиждень
9	Захист проекту	16-й тиждень

Термін захисту результатів виконаної роботи – не пізніше, ніж за десять днів до початку екзаменаційної сесії.

Термін надання пояснювальної записки до курсового проекту – передостанній тиждень навчального семестру.

5 ОФОРМЛЕННЯ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Загальні вимоги.

Загальними вимогами до текстової частини курсового проекту є:

- послідовність викладення матеріалу повинна бути логічною;
- висновки й рекомендації слід обґрунтовувати;
- формулювання повинні бути точними;
- результатів роботи треба викласти чітко й стисло;
- терміни, застосовані у роботі повинні відповідати стандартам.

При створенні пояснювальної записки не допускається переписування загальних положень і визначень із підручників, навчальних статей, посібників і інших джерел. При необхідності використання в текстовому документі матеріалів з літературних джерел, необхідно робити на них посилання.

При написанні пояснювальної записки використовується шрифт Times New Roman, розмір 14, накреслення Звичайне, міжрядковий інтервал – 1,5. Лист формату А4, розміри полів: верхнє, ліве і нижнє – не менш 20 мм, праве – не менш 10 мм. Заповненість сторінок – не менш 30 %.

Сторінки нумеруються в правому верхньому куті, арабськими цифрами. Нумерація сторінок наскрізна. Титульний аркуш, реферат, лист завдання не нумеруються, але входять у загальне число сторінок.

Пояснювальна записка має складатися зі вступної частини, основної частини та додатків (при необхідності). Додатки розміщують після основної частини пояснювальної записки. Загальна кількість сторінок пояснювальної записки до курсового проекту, з урахуванням додатків, повинна складати 40-45 сторінок.

Заголовний блок проекту складається з:

- титульної сторінки (див. додаток А);
- сторінок завдання, календарного плану (див. додаток Б);
- реферату;

– змісту.

Титульна сторінка є першою сторінкою пояснювальної записки і служить основним джерелом бібліографічної інформації, необхідної для обробки і пошуку документів.

Реферат повинний бути коротким, інформативним і містити інформацію, що дозволяє представити сутність і обсяг роботи у наступній послідовності: об'єкт дослідження; мета роботи; методи дослідження; результати та їхня новизна; значимість роботи та висновки. Реферат необхідно розміщувати на одній сторінці формату А4. Ключові слова (від 5 до 15 слів або словосполучень) формують на основі тексту реферату та розміщують після тексту реферату. Перелік ключових слів друкується прописними літерами в називному відмінку в рядок через коми. Приклад оформлення реферату та титульної сторінки наведено у додатку А.

Зміст розміщують безпосередньо після реферату, з нової сторінки. Зміст включає вступ, послідовно перераховані найменування всіх розділів, підрозділів, пунктів і підпунктів (якщо вони є), висновки, перелік посилань, найменування додатків.

Основна частина складається зі вступу, розділів, що описують суть виконаної роботи, висновків, переліку посилань.

У вступі дається коротка характеристика роботи, формулюється її мета, задачі, що вирішуються для досягнення цієї мети, основні отримані результати. Загальний обсяг вступу – до однієї сторінки тексту.

Розділи пояснювальної записки мають висвітлювати рішення задач, завдяки яким досягається мета роботи. Текст має супроводжуватись ілюстраціями та таблицями. За необхідності в опис розділів можна включити короткі теоретичні відомості, але зловживати цим не слід: загальний обсяг таких відомостей не має перевищувати 20 % обсягу записки.

Основні структурні частини текстового документа (розділи) починають із нового аркуша. Заголовки розділів записуються по центру та прописними буквами. Відстань між заголовком розділу та текстом при виконанні записки повинна бути не менш двох рядків.

Заголовки підрозділів, пунктів і підпунктів записки необхідно починати з абзацного відступу та друкувати, крім першої прописної, рядковими (малими) літерами, не підкреслюючи, без крапки наприкінці.

У висновках характеризуються отримані в роботі результати, визначаються напрями подальшої роботи.

Перелік джерел, на які посилаються в записці, повинний бути приведений наприкінці тексту записки, починаючи з нової сторінки. У відповідних місцях записки повинні бути дані посилання на джерела. Бібліографічні описи в переліку посилань приводять у порядку, у якому вони вперше згадувалися в тексті. Порядкові номери описів у переліку є посиланнями в тексті (номерні посилання). Загальна кількість джерел, застосованих при створенні курсового проекту, включаючи інструкції і Інтернет-посилання, не має бути менш двадцяти одиниць.

Ілюстрації (рисунки, схеми, графіки, діаграми, скріншоти (знімки) екранів, блоки програмного коду) необхідно подавати в роботі безпосередньо після тексту, де вони згадані вперше, або на наступній сторінці з відповідною нумерацією сторінок. Позначають ілюстрації словом «Рисунок» і нумерують послідовно в межах розділу, за винятком ілюстрацій у додатках.

Номер ілюстрації складається з номера розділу та порядкового номера ілюстрації в розділі, що розділяються крапкою. Наприклад, «Рисунок 2.1 – Діаграма розміщення». Номер ілюстрації, її назва та пояснювальні підписи розміщуються послідовно під ілюстрацією. На кожен ілюстрацію у тексті повинно бути посилання (наприклад, «на рисунку 1.1 зображено ...»).

Розташовуються ілюстрації по центру сторінки, їх підписи (з вказівкою назви) – по центру рисунку. Ілюстрація відокремлюється від основного тексту роботи порожнім рядком – до ілюстрації та після назви ілюстрації.

В цілому оформлення пояснювальної записки виконується згідно міжнародних і вітчизняних стандартів оформлення документації і звітів в сфері науки та техніки. Зокрема, це – стандарти «ДСТУ 3008-95. Структура та правила оформлення» і «ISO 5966: 1982 Documentation-Presentation of scientific and technical reports».

6 РЕКОМЕНДОВАНИЙ ЗМІСТ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Структура пояснювальної записки до курсового проекту повинна відповідати рекомендаціям, що наведені далі.

Вступ

1 Загальна інформація стосовно об'єкту проектування

1.1 Поняття архітектури програмного забезпечення

1.2 Особливості проектування програмних продуктів

1.3 Опис комп'ютерної гри

1.4 Застосування мови UML для проектування програмних продуктів

2 Формулювання вимог до функціональних можливостей гри у вигляді UML-діаграм прецедентів

2.1 Діаграма варіантів використання

2.2 Діаграма розміщення

3 Розробка діаграм поведінки для комп'ютерного ігрового додатку

3.1 Діаграма станів

3.2 Діаграма діяльності

4 Побудова діаграм взаємодії проєктованого комп'ютерного ігрового додатку

4.1 Діаграма послідовності

4.2 Діаграма кооперації

Висновки

Список використаних джерел

Додатки

7 ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ

Курсовий проект виконується п'ятнадцять тижнів. Пояснювальна записка до курсового проекту надається на перевірку викладачу не менш, ніж за три робочі дні до дати захисту роботи.

Захист відбувається в присутності комісії в складі 2-3 чоловік і включає:

- а) доповідь, що охоплює всі етапи курсового проектування;
- б) презентацію, що містить основні ілюстрації, необхідні для розкриття змісту роботи;
- в) відповіді на запитання комісії.

Контроль виконання курсового проекту включає поточний контроль за виконанням розрахунків за п'ятьма розділами та захист перед комісією. Оцінка виконання та захисту курсового проекту проводиться за 100-бальною шкалою.

Таблиця 7.1 – Розподіл балів, які отримують студенти за виконання курсового проекту

Пояснювальна записка	Захист проекту	Сума
до 40	до 60	100

8 ВАРІАНТИ ЗАВДАНЬ

Рекомендований перелік завдань наведений в таблиці 9.1.

Таблиця 8.1 – Запропоновані варіанти завдань

№ вар.	Назва ігрового комп'ютерного додатку
1	Word of Tanks
2	Відьмак
3	Strike Force Heroes (Герої ударного загону)
4	Drag Racing (Вуличні гонки)
5	Расыbits Fut
6	Ninja: Cool Adventures! (Ніндзя: класні пригоди!)
7	FlatOut2
8	God of War (Бог війни)
9	XCOM (UFO Defense/Terror from the Deep/Apocalypse/Interceptor/Enforcer/Enemy Unknown/Chimera Squad)
10	Portal
11	Dark Souls (Темні душі)
12	Hitman (Найманий вбивця)
13	Final Fantasy (Остання фантазія)
14	Dragon's Dogma
15	Doom
16	Battlefield: Bad Company
17	Civilization (Цивілізація)
18	Pokemon: Let's Go
19	Cities: Skylines
20	Stellaris
21	Fishdom

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Арлоу, Джим. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование / Джим Арлоу, Айла Нейштадт. – М. : Символ-плюс, 2013. – 624 с.
2. Бевз, О.М. Проектування програмних засобів систем управління : навч. посіб. / О.М. Бевз, В.М. Папінов, Ю.А. Скидан. – Вінниця : ВНТУ, 2010. – 125 с.
3. Боггс, М. UML и Rational Rose / М. Боггс. – М. : РГГУ, 2016. – 438 с.
4. Буч, Грейди. Язык UML. Руководство пользователя / Грейди Буч, Джеймс Рамбо, Айвар Джекобсон. – М. : ДМК, 2013. – 432 с.
5. Гома, Хассан. UML. Проектирование систем реального времени, параллельных и распределенных приложений / Хассан Гома. – М. : ДМК Пресс, 2016. – 700 с.
6. Зиглер, К. Методы проектирования программных систем / К. Зиглер ; [пер. с англ.]. – М. : Мир, 1985. – 328 с.
7. Иванов, Д.Ю. Уніфікована мова моделювання UML : учеб. пособие / Д.Ю. Иванов, Ф.А. Новиков. – СПб. : Изд-во Политехн. ун-та, 2010. – 249 с.
8. Казарін, О.В. Надійність і безпека програмного забезпечення / О.В. Казарін, І.Б. Шубинський. – М. : Юрайт, 2018. – 342 с.
9. Ларман, Крэг. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ, проектирование и итеративную разработку / Крэг Ларман. – М. : Гостехиздат, 2017. – 736 с.
10. Леоненков, А. Самоучитель UML / А. Леоненков. – СПб. : БХВ-Петербург, 2017. – 576 с.
11. Мацяшек, Л.А. Анализ и проектирование информационных систем с помощью UML 2.0 / Л.А. Мацяшек ; [пер. с англ.]. – 3-е изд. – М. : Вильямс, 2008. – 816 с.
12. Мюллер, Р.Дж. Базы данных и UML. Проектирование / Р.Дж. Мюллер. – М. : ЛОРИ, 2017. – 420 с.

13. Пайлон, Д. UML 2 для программистов / Д. Пайлон, Н. Питмен. – СПб. : Питер, 2012. – 272 с.
14. Терехов, А.Н. Технология программирования : учеб. пособие / А.Н. Терехов. – М. : Интернет-Ун-т Информ. технологий, 2012. – 149 с.
15. Фаулер, М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования / М. Фаулер. – М. : Символ-плюс, 2013. – 192 с.
16. Фримен, Э. Патерны проектирования / Э. Фримен, К. Сьерра, Б. Бейтс. – М. : Фабула, 2020. – 672 с.
17. Эванс, Эрик. Предметно-ориентированное проектирование (DDD). Структуризация сложных программных систем / Эрик Эванс. – М. : Вильямс, 2016. – 448 с.
18. Як створити діаграму IPO [Електронний ресурс]. – Режим доступу : <https://uk.computersm.com/10-how-to-create-an-ipo-chart-29517>. – Назва з екрану.
19. Rumbaugh, James. The unified modeling language reference manual [Electronic resource] / James Rumbaugh, Ivar Jacobson, Grady Booch. – Addison Wesley Longman Inc., 1999. – 586 p. – Access mode : http://www.temida.si/~bojan/IPIT_2014/literatura/UML_Reference_Manual.pdf. – Назва з екрану.
20. Unified Modeling Language : Superstructure. Object Management Group. 2007 [Electronic resource]. – Access mode : <https://www.omg.org/spec/UML/2.2/Superstructure/PDF>. – Назва з екрану.
21. Методичні вказівки до виконання курсової роботи з дисципліни «Архітектура та проектування програмного забезпечення» для студентів спеціальності 121 Інженерія програмного забезпечення [Електронний ресурс] / уклад. Н.С. Костюкова, О.А. Тихонова . – Покровськ : ДонНТУ, 2019. – 47 с.
22. Планета інформатики [Электронный ресурс]. – Режим доступа : <https://inf1.info/>. – Название с экрана.
23. Портал знань [Електронний ресурс]. – Режим доступу : <http://www.znannya.org/>. – Назва з екрану.

24. UML [Електронний ресурс]. – Режим доступу : <http://www.uml.org/>
– Назва з екрану.
25. Umbrello UML Modeller : підручник [Електронний ресурс]. – Режим доступу : <https://docs.kde.org/trunk4/uk/kdesdk/umbrello/index.html>. – Назва з екрану.
26. Rumbaugh, J. The Unified Modeling Language. User manual [Електронний ресурс] / J. Rumbaugh, I. Jacobson, G. Booch. – Режим доступу : http://www.temida.si/~bojan/IPIT_2014/literatura/UML_Reference_Manual.pdf. – Назва з екрану.
27. UML-Tutorial [Електронний ресурс]. – Режим доступу : <https://www.tutorialspoint.com/uml/index.htm>. – Назва з екрану.
28. Шаблони або патерни проектування звичною мовою. Ч.1. [Електронний ресурс]. – Режим доступу : <https://echo.lviv.ua/dev/5432>. – Назва з екрану.
29. Shvets, Alexander. Патерни проектування українською [Електронний ресурс] / Alexander, Shvets. – Режим доступу : <https://dou.ua/forums/topic/23683/>. – Назва з екрану.
30. Патерни проектування [Електронний ресурс]. – Режим доступу : <https://refactoring.guru/uk/design-patterns>. – Назва з екрану.
31. Уніфікована мова моделювання UML [Електронний ресурс]. – Режим доступу : <http://www.znannya.org/?view=uml>. – Назва з екрану.
32. Основы UML [Електронный ресурс]. – Режим доступа : <http://techn.sstu.ru/kafedri/%D0%BF%D0%BE%D0%B4%D1%80%D0%B0%D0%B7%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F/1/MetMat/murashev/oop/lec/lec13.htm>. – Название с экрана.
33. Якість програмного забезпечення [Електронний ресурс]. – Режим доступу : https://stud.com.ua/128171/informatika/yakist_programnogo_zabezpechennya. – Назва з екрану.
34. Аналіз якості програмного забезпечення [Електронний ресурс]. – Режим доступу : <https://pidru4niki.com/1194121347734/informatika/>

[analiz_yakosti_programnogo_zabezpechennya](#). – Назва з екрану.

35. Критерії якості програмного забезпечення та систем [Електронний ресурс]. – Режим доступу : [http://www.dut.edu.ua/ uploads/l_1220_38068797.pdf](http://www.dut.edu.ua/uploads/l_1220_38068797.pdf). – Назва з екрану.

36. Диаграммы реализации [Электронный ресурс]. – Режим доступа : <http://www.maksakov-sa.ru/ModelUML/DiagrReal/index.html>. – Название с экрана.

37. Проектування інформаційних систем на основі уніфікованої мови моделювання [Електронний ресурс]. – Режим доступу : [https:// elearning.sumdu.edu.ua/free_content/lectured:de1c9452f2a161439391120eef364dd8ce4d8e5e/20151203140326/204841/index.html](https://elearning.sumdu.edu.ua/free_content/lectured:de1c9452f2a161439391120eef364dd8ce4d8e5e/20151203140326/204841/index.html). – Назва з екрану.

38. Экономика информационной безопасности [Электронный ресурс]. – Режим доступа : <http://www.trn.ua/articles/2081/>. – Название с экрана.

39. Підручник з Umbrello UML Modeller [Електронний ресурс] / [пер. укр. Юрій Чорноіван]. – Режим доступу : [https://docs.kde.org/ stable5/ uk/ k d e sdk/umbrello/index.html](https://docs.kde.org/stable5/uk/kdesdk/umbrello/index.html). – Назва з екрану.

40. Як працювати з Umbrello UML Modeller [Електронний ресурс]. – Режим доступу : <https://docs.kde.org/trunk5/uk/kdesdk/umbrello/working-with-umbrello.html>. – Назва з екрану.

41. Підручник з Umbrello UML Modeller [Електронний ресурс]. – Режим доступу : http://www.institut.cn.ua/uploads/files/m_23/Modelyuvannya-ta-analz-PZ-Pdruchnik-UML-Modeller.pdf . – Назва з екрану.

ДОДАТОК А

Приклади оформлення титульних листів

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДВНЗ «ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»
Навчально-науковий інститут комп'ютерних наук і технологій
Кафедра прикладної математики та інформатики

ПОЯСНЮВАЛЬНА ЗАПИСКА

до курсового проекту
з дисципліни: «Архітектура та проектування програмного забезпечення»
на тему: «Розробка архітектури програмної системи на базі ігрового
додатку

« _____ »
(назва програмного продукту)

Варіант _____

Студента (-тки) .гр. _____
_____ (дата, підпис) _____ (ПІБ)

Керівники проекту:

_____ (дата, підпис) _____ (ПІБ)

_____ (дата, підпис) _____ (ПІБ)

Оцінка за національною шкалою _____

Загальна кількість балів _____

Покровськ-2020

РЕФЕРАТ

Пояснювальна записка до курсового проекту містить: ____ сторінок, ____ рисунків, ____ таблиць, ____ додатків.

Об'єкт дослідження – це ...

Мета проекту – це ...

Основні задачі – це ...

Результати курсового проекту – це ...

КЛЮЧОВІ СЛОВА

ДОДАТОК Б

Завдання на курсовий проект

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»

ЗАТВЕРДЖУЮ
зав. каф. ПМІ, проф., д.т.н.
_____ О.А. Дмитрієва
«_____» _____ 20__ р.

ЗАВДАННЯ
на курсовий проект
з дисципліни: «Архітектура та проектування програмного забезпечення»
студенту _____ -
Навчально-наукового інституту комп'ютерних наук і технологій
групи _____
на тему: «Розробка архітектури програмної системи на базі ігрового додатку
«_____»
(назва програмного продукту)
Варіант _____

1. Дата видачі завдання: « ____ » _____ 20____ року

2. ЗАВДАННЯ:

Створити архітектуру програмного ігрового додатку у вигляді комплекту UML-діаграм прецедентів, реалізації, поведінки та взаємодії.

Варіант завдання: _____.

3. Термін захисту роботи: « ____ » _____ 20____ р.

Термін здачі пояснювальної записки: « ____ » _____ 20____ р.

4. Графік виконання курсового проекту

Етап	Термін виконання (номер тижня)	Примітка про виконання
1. Вивчення програмного додатку	2	
2. Створення діаграми варіантів використання	4	
3. Розробка діаграми розміщення	6	
4. Розробка діаграм станів	8	
5. Розробка діаграм діяльності	10	
6. Розробка діаграм послідовності	12	
7. Розробка діаграм кооперації	14	
8. Оформлення пояснювальної записки	15	
9. Захист роботи	16	

Студент _____ (_____) (підпис) (ПБ)

Керівники роботи _____ (_____) (підпис) (ПБ)

_____ (_____) (підпис) (ПБ)

ДОДАТОК В

Обрання інструментарію для побудови UML-діаграм

Архітектура системи – це структурна схема компонентів системи, що взаємодіють між собою через інтерфейси. У ній ідентифікуються загальні частини системи, в тому числі готові програмні продукти і наново розроблені компоненти, а також компоненти, багаторазово використовувані в інших додатках.

Мова UML (Unified Method Language) – загальноцільова мова візуального моделювання, яка розроблена для специфікації, візуалізації, проектування та документування компонентів програмного забезпечення, бізнес-процесів і інших систем.

Вділяють наступні види UML-діаграм:

- 1) діаграма варіантів використання (use case diagram);
- 2) діаграма класів (class diagram);
- 3) діаграми поведінки (behavior diagrams):
 - діаграма станів (statechart diagram);
 - діаграма діяльності (activity diagram);
- 4) діаграми взаємодії (interaction diagrams):
 - діаграма послідовності (sequence diagram);
 - діаграма кооперації (collaboration diagram);
- 5) діаграми реалізації (implementation diagrams):
 - діаграма компонентів (component diagram);
 - діаграма розгортання (deployment diagram).

Існує значна кількість інструментальних засобів для створення та відображення UML-діаграм. Більшість з них містять шаблони стандартних UML діаграм. Деякі онлайн-дизайнери UML підтримують роботу в режимі реального часу створення UML діаграм у групі з колегами.

Прикладами сучасних інструментальних засобів для відображення UML-діаграм, є системи Cасoo, Creately, Diagramo, Draw.io, EdrawSoft,

GenMyModel, Gliffy, LucidChart, Microsoft Visio, Gliffy, Rational Rose, Visual Paradigm, Umbrello, UMLet та багато інших.

Нижче наведено короткі відомості та описано особливості деяких з означених інструментальних засобів створення, редагування, документування та відображення UML-діаграм.

1. Cасоо.

Сасоо – це зручний онлайн інструмент малювання, який дозволяє створювати різні діаграми, такі як карти сайту, каркасні схеми, UML і мережеві графіки. Сасоо можна використовувати безкоштовно. Домашня сторінка сайту відображена на рис. В.1.

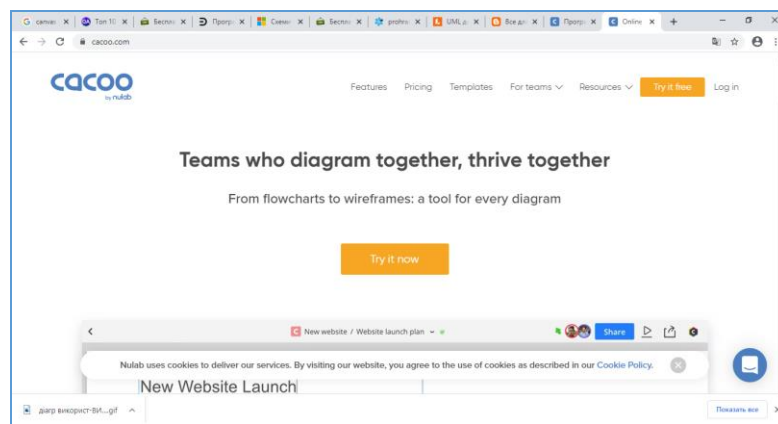


Рисунок В.1 – Домашня сторінка сайту Сасоо

В Сасоо доступні різноманітні трафарети, які дозволяють малювати не тільки UML- та ER-діаграми, електричні схеми, а й карти сайту, блок-схеми, офісне обладнання, схеми мережи, планування приміщень й інше.

Сервіс має безкоштовну версію. В ній можна одночасно розроблювати до 6 схем і взаємодіяти з 15 особами в режимі реального часу.

В Сасоо можна зберігати декілька версій однієї і тієї ж схеми UML діаграми, а також залишати коментарі. Початкова сторінка проектувальника відображена на рис. В.2.

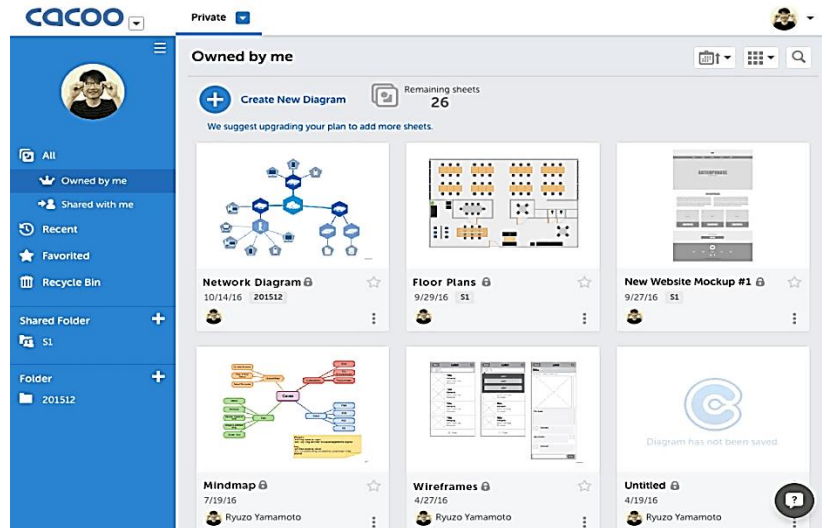


Рисунок В.2 – Початкова сторінка проєкту Cacoo

Передбачена можливість експорту проєктів у різних форматах, інтеграція з Microsoft Teams, Google Drive, Google Docs, TypeTalk, Dropbox, Adobe, AWS, Slack, OneDrive та ін..

Розробник – Nulab Inc. (Японія),

Перший випуск 2009 р.

Інтернет сайт <https://cacoo.com/>

Сумісна з Windows, Linux, Android, iOS

2. Creately.

Creately – це вебсервіс з безліччю спеціалізованих зручних інструментів, який допомагає створювати професійні діаграми та динамічні схеми.

Зазначений засіб відомий характерною простотою і швидкодією; дозволяє малювати та відображати блок-схеми, карти сайтів, організаційні діаграми, UML-діаграми, мережеві діаграми, діаграми Венна, діаграми SWOT-аналізу, карти зв'язків, моделі бізнес-процесів і багато іншого (рис. В.3).

Сучасна версія продукту – Creately Next (2019 р.).

Creately Next – не просто онлайн-інструмент для побудови діаграм – це візуальний робочий простір для спільної роботи команди, канал, де команди

збираються разом, щоб думати, спілкуватися, проводити мозковий штурм, планувати, аналізувати і запроваджувати нововведення.

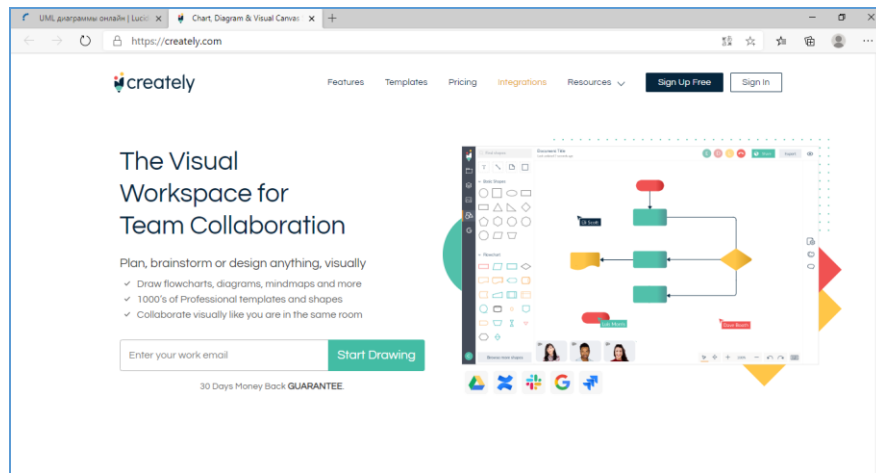


Рисунок В.3 – Веб-сервіс Creately

Creately Next включає спеціалізовані форми для створення понад сімдесяти типів діаграм, бібліотеку, що може розширюватися, швидкі клавіші, більше тисячі шаблонів, має постійну підтримку – і це ще не повний перелік можливостей системи.

Наявна безкоштовна версія даного засобу, в якій можна одночасно працювати над п'ятьма діаграмами та кооперуватися з трьома партнерами.

Сервіс має красивий інтерфейс, який плавно реагує на кожну дію користувача. Елементи, які можна використовувати в діаграмах, поділені на категорії, тому знайти потрібний елемент не є проблемою. Якщо ж деякий елемент відсутній, то його можна додати вручну.

Створену діаграму можна експортувати в формат PDF, PNG або JPG.

Згідно меню Creately підтримує наступні UML-діаграми: Class (діаграми класів), Object (діаграми об'єктів), Activity (діаграми активності), State (діаграми станів), Sequence (діаграми послідовності), Component (діаграми компонентів), Deployment (діаграми розгортання), Communications (діаграми взаємодії).

Основні характеристики продукту:

- простота у використанні і інтерактивний інтерфейс;
- командна робота в режимі реального часу;
- професійно розроблені шаблони для швидкого початку роботи;
- значна кількість форм, згрупованих в бібліотеки;
- вбудований Google Image і іконка пошуку Finder;
- генерація коду для вбудовування діаграм на сайти;
- організація співтовариства для швидкого пошуку діаграм;
- можливість імпорту зображень;
- підтримка Drag & Drop;
- бібліотека зображень.

Сервіс працює з Google- диском, є інтеграція з Slack, створена версія для Confluence (анонсовано на 2020 рік), існує взаємодія з Zapier та OneDrive.

Інтернет-посилання на офіційний сайт продукту: <https://creately.com/>

Фірма-розробник – Cinergix Pty. Ltd (australia)

Сумісність з операційними системами: Windows, Linux і Mac OS.

3. Draw.io.

Draw.io – це безкоштовне онлайн-програмне забезпечення для створення діаграм і блок-схем, діаграм процесів, організаційних діаграм, UML-, ER- і мережевих діаграм.

Draw.io – технологія моделювання програмного забезпечення з відкритим вхідним кодом і додаток для кінцевих користувачів. Має позначку «Найкраще програмне забезпечення для блок-схем (2020)».

При використанні даного засобу неможна редагувати один і той же документ двом людям одночасно, проте можна зберігати файли в будь-якому сховищі – Google Drive, OneDrive, Dropbox, GitHub, GitLab.

Всі необхідні символи UML-діаграм компактно розміщені на панелі інструментів, що розташована збоку. Класифікуються вони за типами UML-діаграм. Переміщаються drag-n-drop. Зовнішній вигляд діаграм можна

налаштувати на панелі «Формат». Є варіанти зміни стилю фігури та стилю тексту. Фрагмент діаграми компонентів подано на рис. В.4.

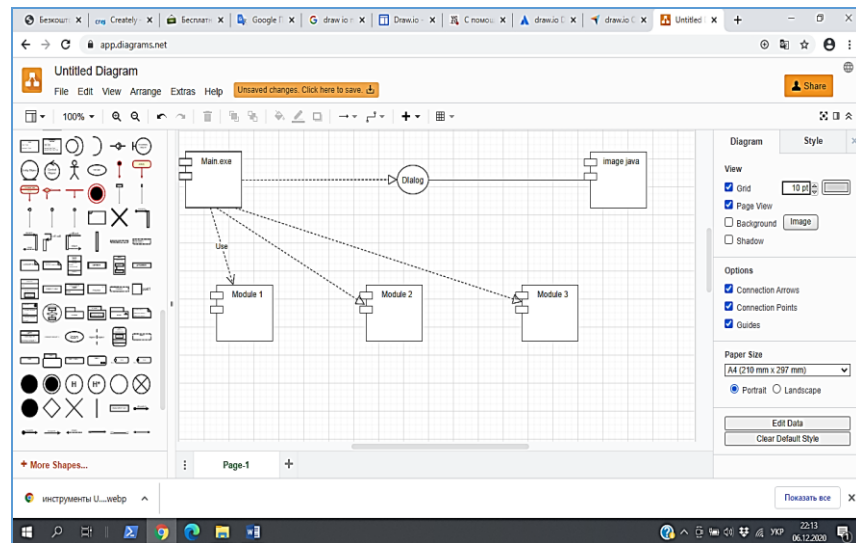


Рисунок В.4 – Фрагмент діаграми компонентів, розробленої з застосуванням програмного продукту Draw.io

Переваги сервісу Draw.io:

- велика кількість шаблонів, що забезпечує можливість оперативного та швидкісного створення потрібної діаграми без особливих зусиль;
- простий інтуїтивний інтерфейс;
- відсутність обмежень та умов на здійснення процедури завантаження завершеного проекту на комп'ютер.

Draw.io дозволяє створити наступні типи UML-діаграм: Class (діаграми класів); Object (діаграми об'єктів); Activity (діаграма активності); State (діаграми станів); DFD Data Flow Diagram (діаграма потоків даних); Collaboration (діаграми кооперації); Use Case (Юз-кейс-діаграми, діаграми варіантів використання, сценаріїв, діаграма прецедентів); Deployment (діаграми розгортання).

Інтернет-посилання на офіційний сайт продукту: <https://draw.io>.

Формати зберігання UML-діаграм: JPG, PNG, SVG та ін.

Сумісність з операційними системами: Windows, Mac, Ios, Android.

4. EdrawSoft : Edraw Max.

Edraw Max – це графічне програмне забезпечення, яке може сприяти створенню більше двохсот типів діаграм і графічних представлень у вигляді бізнес-презентацій, планів будівель, карт розуму, наукових ілюстрацій, дизайну одягу, діаграм UML, робочих процесів, каркасів, електричних діаграм, діаграм r&id, діаграм спрямованості, діаграм бази даних тощо. На рис. В.5 відображено вікно завантаження версії програми для різних операційних систем.

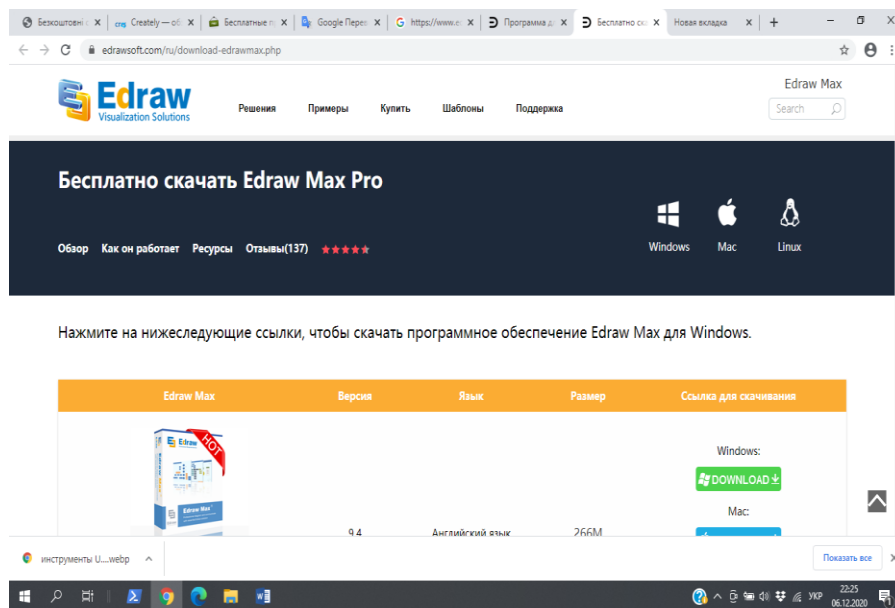


Рисунок В.5 – Вікно завантаження Edraw

UML-діаграми в Edraw Max – це діаграми послідовності, використання, класів, діяльності, розгортання і багато інших.

Функціональні особливості Edraw Max:

- підтримка імпорту файлів Visio XML;
- швидке створення професійних діаграм на основі стилів;
- побудова діаграм для оптимального представлення;
- автоматичне вирівнювання і узгодження розташування елементів;
- векторне малювання, прискорює створення організаційних графіків, блок-схем, мережових діаграм;

- можливість вибору кольору, шрифту, форми і стилю тексту, картинки і символу для кожного об'єкта діаграми;
- інтеграція та інтерфейс в тилі Microsoft Office;
- можливість прив'язки діаграм до даних;
- підтримка основних форматів графіки.

Програма має велику кількість бібліотек і понад 5000 векторних символів, завдяки чому процес креслення в Edraw Max Professional значно спрощується. Користувачі Edraw Max мають доступ до численних шаблонів, форм і інструментами, можливість перенести діаграми в Word, Excel, PowerPoint, PDF, SVG, PNG, підтримка імпорту XML-файлів Visio.

Інтернет-посилання на офіційний сайт продукту:
<https://www.edrawsoft.com/ru/uml/>.

Сумісність з операційними системами: Windows, Mac, Linux.

5. Gliffy.

Gliffy – онлайн-редактор для схем, графіків і діаграм, реалізований за допомогою використання хмарного додатка. Редактор є умовно безкоштовним, має достатню функціональність. Сервіс відрізняється зрозумілим дизайном і широким функціоналом (рис. В.6).

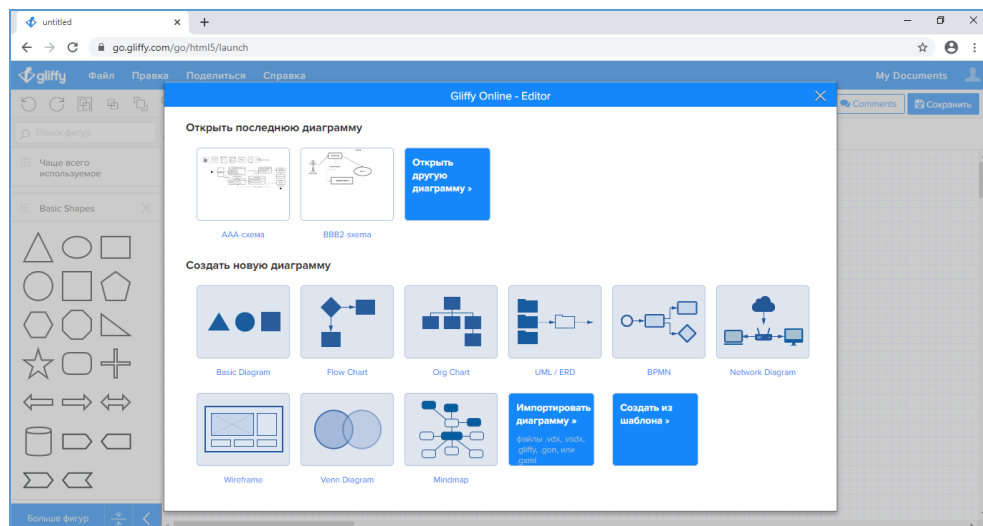


Рисунок В.6 – Вікно-запрошення програмного продукту Gliffy

Завершені проекти можна зберігати на сервері Gliffy і переглядати за згенерованим посиланням. Для перенесення діаграми в файл власного проекту рекомендовано застосовувати скріншоти екрану. Детальний опис та приклади застосування редактору надано в [21].

Нові користувачі отримують на термін в тридцять днів про-акаунт. По завершенню зазначеного терміну можна придбати повноцінний «Pro» або ж продовжити користування Gliffy в звичайному режимі, при цьому список доступних функцій зменшиться, але для середньостатистичного користувача їх все одно буде достатньо.

Роботу можна почати як з нуля, так і з використання шаблону. Завершені проекти можна зберігати на сервері Gliffy і переглядати за згенерованими посиланнями (рис. В.7).

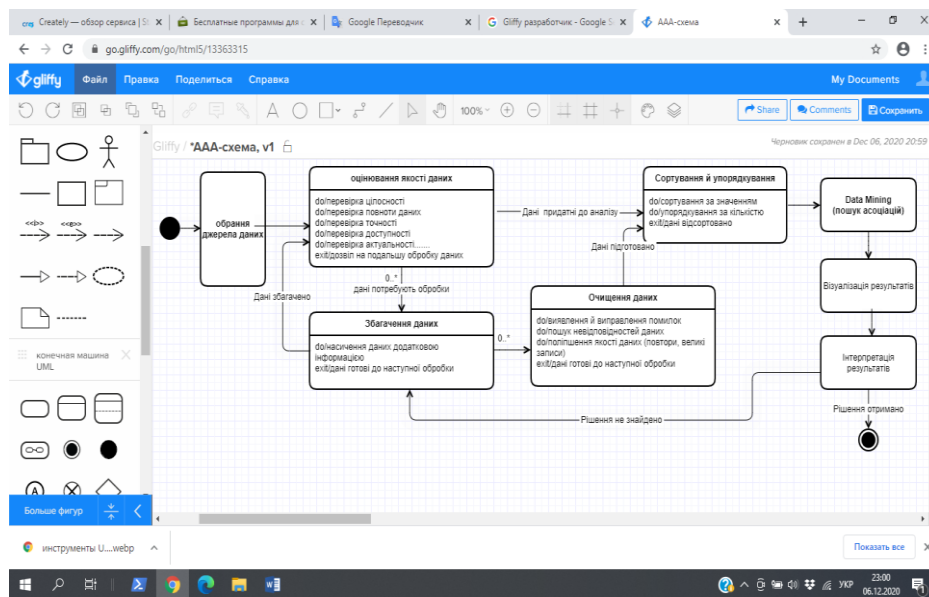


Рисунок В.7 – Приклад згенерованої діаграми в програмному продукті Gliffy

У програмі представлено тринадцять категорій елементів, що відносяться до різних видів діаграм і схем. Робота з елементами здійснюється за принципом Drag and Drop. Передбачена можливість зміни масштабу, розмірів зображень, порядку шарів, можливе налаштування стилю, додавання

та форматування текстової інформації. Готові діаграми можна відправляти на друк або експортувати у формати SVG, GXML, PNG, JPG.

Вебсервіс підтримує командну роботу. Обмежений функціонал редактора доступний безкоштовно, розширений набір сервісів – за наявності платної підписки.

Інтернет-посилання на офіційний сайт продукту:
<https://www.gliffy.com/>.

Розробник – Gliffy, Inc.

Сумісність з операційними системами: Windows, Mac, Android, iOS, Windows Phone.

6. LucidChart.

Інтерфейс онлайн-редактора UML-дизайну LucidChart представлений на рисунку В.8. Символи та графічні блоки UML-діаграм поділені на блоки, що розміщені в панелі інструментів ліворуч. Існує підтримка Drag&Drop, можна перетягувати елементи UML-діаграми з панелі інструментів на полотно (рис. В.8). Кількість шаблонів достатньо велика. Готові проекти можна зберігати в форматах PDF, JPG, PNG і деяких інших.

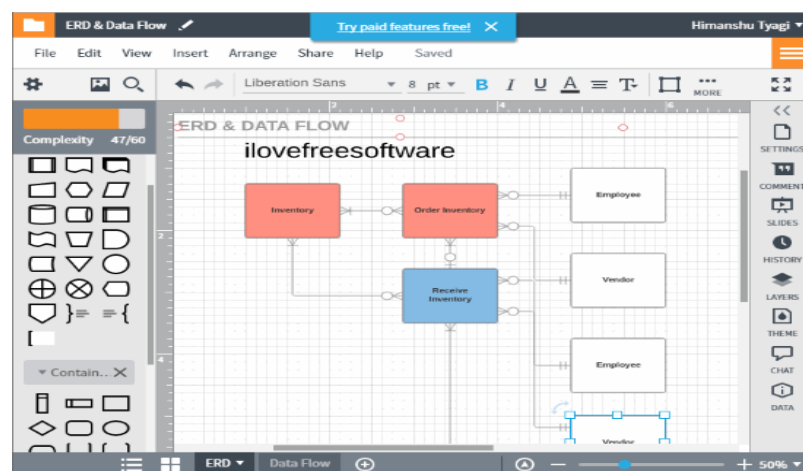


Рисунок В.8 – Інтерфейс онлайн-редактора LucidChart

Існує платна та безкоштовна версії LucidChart. В безкоштовному варіанті можна зберігати не більше трьох активних діаграм. Однак

можливість взаємодіяти в режимі реального часу з іншими розробниками проекту залишається.

Діаграми, які підтримуються Lucidchart: ERD & Data Flow (діаграми зв'язків та потоків даних); Activity (діаграми активності); Sequence (діаграми послідовності); Component (діаграми компонентів); Class diagram (діаграми класів); Use Case (Юз-кейз діаграми, діаграми варіантів використання, сценаріїв, діаграма прецедентів); State Diagram (діаграми станів).

Згідно інформації розробників сервісом користується вже більше, ніж одинадцять мільйонів людей по всьому світу. На YouTube є відеоуроки зі створення UML-діаграм у LucidChart. У грудні 2010 року редактор LucidChart був включений до інтернет-магазину Chrome, і є однією з найпопулярніших платних програм.

Інтернет-посилання на офіційний сайт продукту:
<https://www.lucidchart.com>.

Розробник – Lucid Software.

Сумісність з операційними системами: Microsoft Windows, Mac OS X та Linux.

7. IBM Rational Rose.

IBM Rational Rose – об'єктно-орієнтований CASE-засіб проектування інформаційних систем, набір інструментів моделювання UML для розробки програмного забезпечення. Сторінка сайту фірми IBM, де можна знайти опис засобів моделювання та скачати демо-версію, наведено на рисунку В.9.

Один з інструментів Rational Rose – Rational Software Architect – це середовище моделювання та розробки, яке використовує уніфіковану мову моделювання (UML) для проектування архітектури додатків і вебсервісів, C++ і Java EE (JEE).

Rational Software Architect побудований на платформі програмного забезпечення з відкритим вихідним кодом Eclipse і включає можливості, орієнтовані на аналіз архітектурного коду, C++ і розробку на основі моделей (MDD) з UML для створення додатків і вебсервісів.

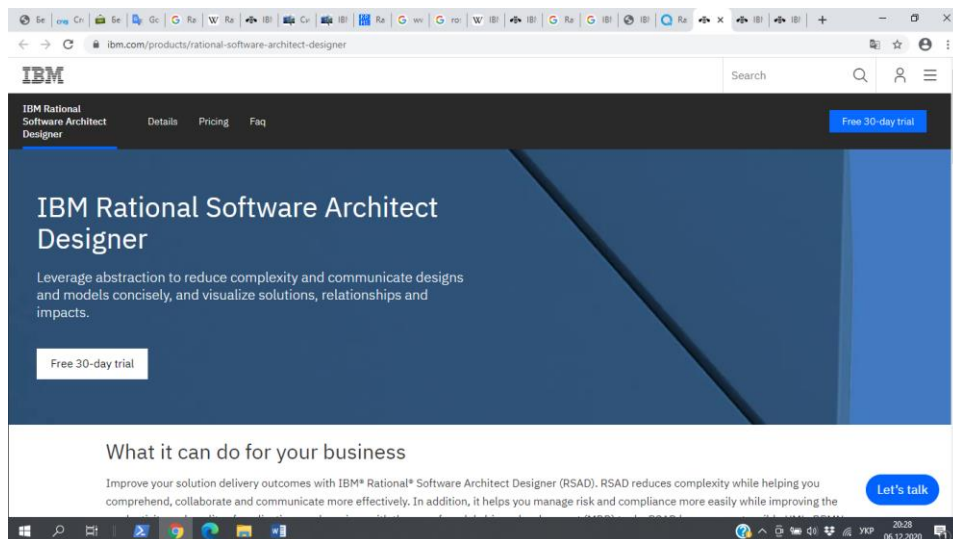


Рисунок В.9 – Сторінка сайту з засобами моделювання Rational Rose

Rational Software Architect – це сімейство з трьох основних продуктів:

- Rational Software Architect: надає інтегрований інструмент проєктування та розробки з підтримкою UML;
- Rational Software Architect для WebSphere Software: платформа, оптимізована для додатків SOA, JEE і Websphere;
- Rational Software Architect Design Manager: інструмент із забезпеченням спільної роботи для обміну проєктною інформацією та управління нею.

Rational Software Architect підтримує UML версії 2.1, дозволяє управляти моделями для паралельної розробки та архітектурного рефакторинга, наприклад, розділяти, комбінувати, порівнювати та об'єднувати моделі й фрагменти моделей. Надає інструменти візуального конструювання для прискорення проєктування та розробки програмного забезпечення.

Rational Rose має ряд переваг, серед яких: зручний графічний інтерфейс, відкритий для доповнень; багатоплатформність; інтеграція з Microsoft Visual Studio та з усіма продуктами Rational; підтримка технологій Microsoft, CORBA і Java-додатків; проєктування систем будь-якої складності; підтримка мови UML; можливості автоматичного контролю, надання розгорнутої інформації про проєкт (документування); можливості

генерування коду (стандартний список модулів Включає C++, ADA, CORBA, Visual Basic, XML, COM, Oracle); зворотне проектування систем.

Загальний вид екрану моделювання Rational Rose наведено на рисунку В.10.

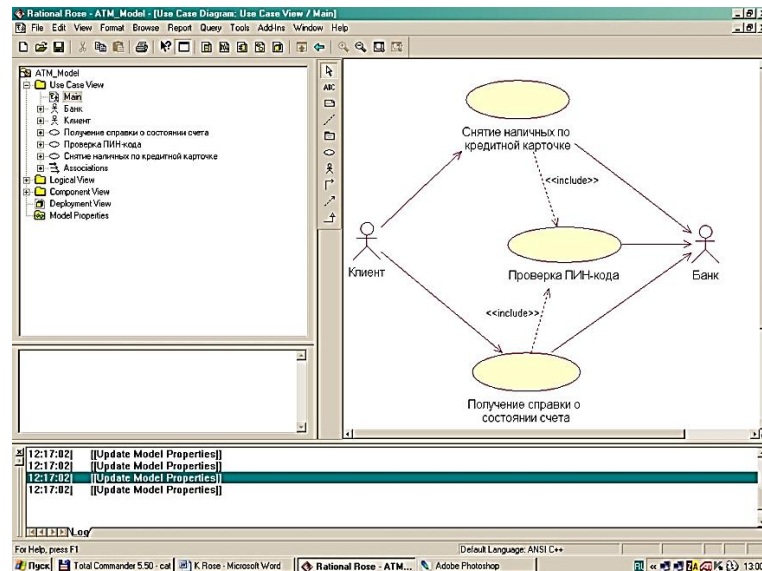


Рисунок В.10 – Загальний вигляд вікна моделювання Rational Rose

Інтернет-посилання на офіційний сайт продукту:
<https://www.ibm.com/products/rational-software-architect-designer/details>.

Розробник – Rational Software (з 2003 р. – поглинула корпорація IBM).

Діюча версія: Rational Software Architect – V9.7.0.2/19.11.2019.

Сумісність з операційними системами: Microsoft Windows, Linux, OS X.
8. Umbrello.

Umbrello – середовище UML-моделювання, призначено для побудови UML-діаграм на платформі Unix (рис. В.11).

Umbrello UML Modeller є вільним програмним забезпеченням, що має простий та функціональний користувацький інтерфейс.

Umbrello UML Modeller 2.11 передбачає підтримку всіх стандартних типів UML-діаграм: діаграми класів; діаграми послідовностей; діаграми співпраці (кооперації); діаграми випадків використання (прецедентів); діаграми станів; діаграми діяльності; діаграми компонентів; діаграми впровадження (розгортання); діаграми взаємозв'язків сутностей.

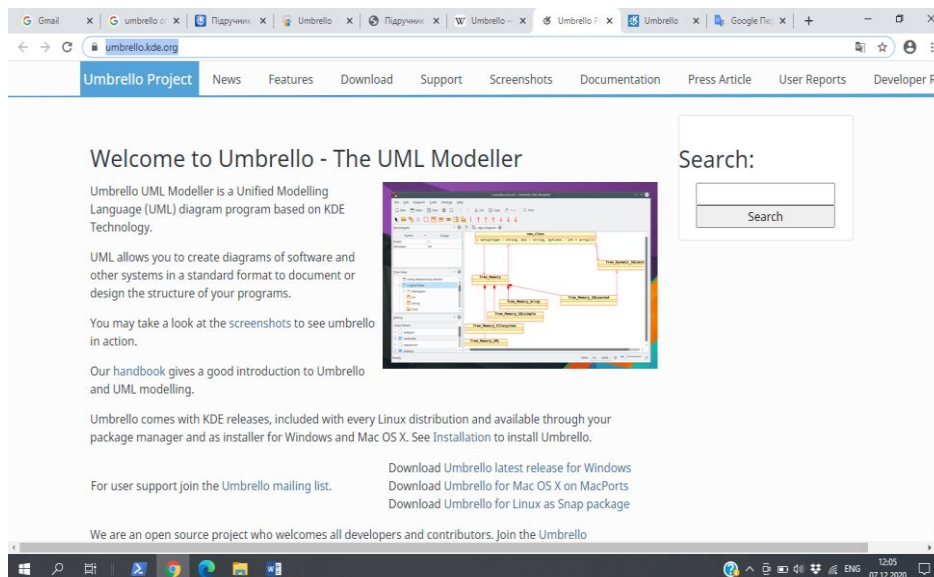


Рисунок В.11 – Вебсторінка з описом Umbrello

Umbrello – частина віконного середовища KDE, але, незважаючи на це, інструмент працює і з іншими віконними менеджерами в інших програмних оточеннях.

Приклад відображення в Umbrello діаграми випадків використання подано на рисунку В.12.

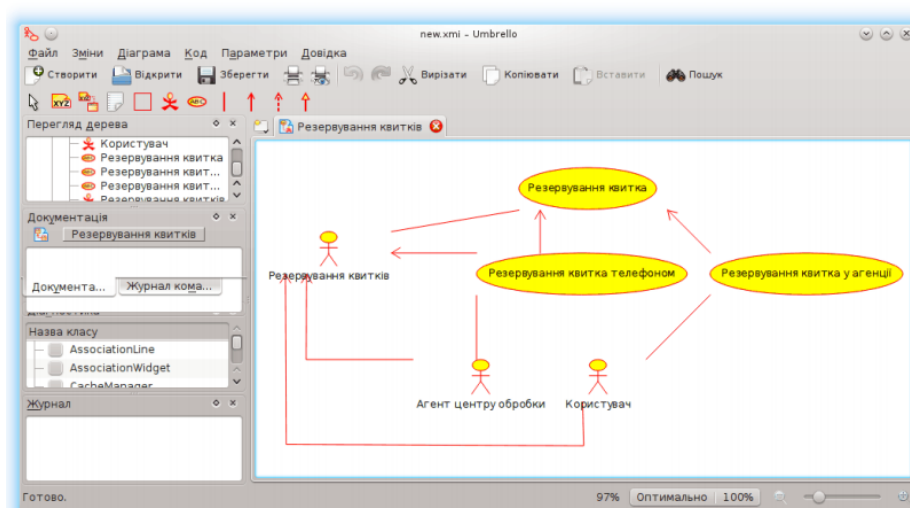


Рисунок В.12 – Діаграма випадків використання[41], створена за допомогою Umbrello

Umbrello дозволяє зберігати дані моделі в форматах DocBook и XHTML. Формат файлу, що використовується при зберіганні діаграм,

Заснований на XMI. Підтримується імпорт та експорт діаграм у різні мови програмування, як то C++, IDL, Pascal / Delphi, Ada, Python, Java, Perl.

Детальний опис і інструкція по роботі з інструментом наведені у інформаційному джерелі [39].

Електронний підручник з UML-моделювання (українською мовою) надається за активним посиланням у заголовку сайту за посиланням <http://docs.kde.org/> [40].

Розробник – міжнародна спільнота KDE.

Інтернет-посилання на офіційний сайт продукту:
<https://umbrello.kde.org/>.

Інтернет-посилання на вебсторінку з підручником (українською мовою): <https://docs.kde.org/stable5/uk/kdesdk/umbrello/index.html>.

Сумісність з операційними системами: Windows, Mac, Linux.

9. UMLet.

UMLet – це безкоштовний інструмент UML з відкритим вихідним кодом і простим призначенням для користувача інтерфейсом (рис. В.13).

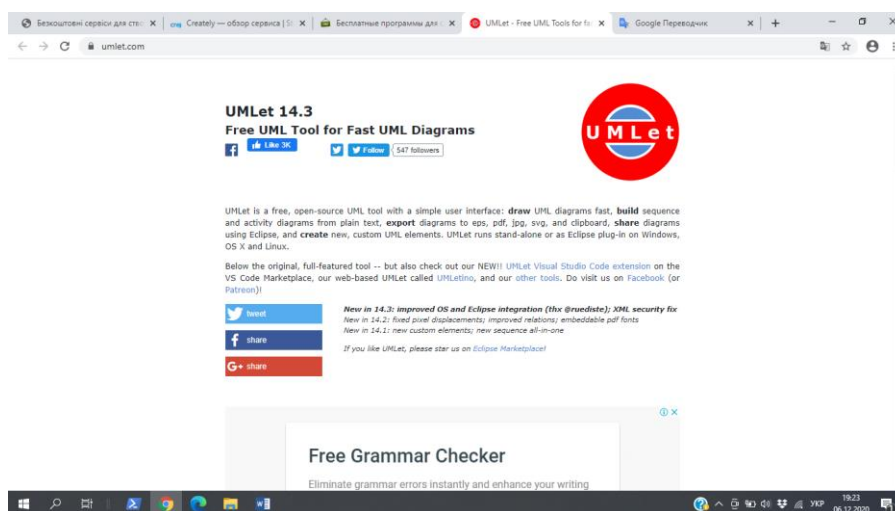


Рисунок В.13 – Домашня сторінка UMLet

Приклад робочого екрану UMLet наведено на рисунку В.14.

Особливості UMLet:

- швидке малювання UML-діаграм всіх типів;
- побудова діаграм послідовності та дій зі звичайного тексту;

- експорт діаграм в EPS, PDF, JPG, SVG і буфер обміну;
- обмін діаграмами за допомогою Eclipse;
- створення нових налаштовуваних елементів UML.

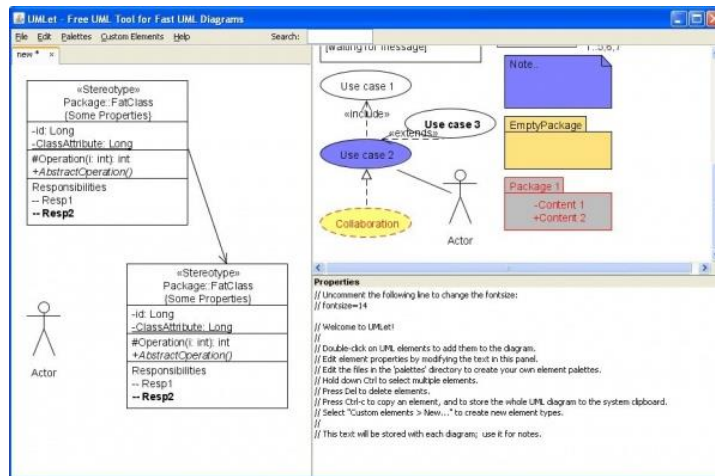


Рисунок В.14 – Приклад побудови діаграми в середовищі UMLet

Середовище UMLet надає можливість розробки власних елементів на основі існуючих стандартних елементів. Для цього необхідно створити новий елемент у вбудованому редакторі, визначити його властивості за допомогою введення коду Java і додати в діаграму (зберегти). На сайті <https://www.umlet.com/> представлений відеоурок по роботі з UMLet і можливість завантаження робочої версії.

Інтернет-посилання на офіційний сайт продукту:
<https://www.umlet.com/>.

UMLet поширюється під ліцензією GNU.UMLet, працює автономно або як окремий плагін Eclipse.

Сумісність з операційними системами: Windows, Linux, FreeBSD, MacOS.

Таким чином, студент має можливість обрати зручний для себе інструмент створення та редагування UML-діаграм і моделювати архітектуру програмної системи на сучасному кваліфікаційному рівні з можливістю колективної роботи над проектом й поширенням наробок у різні професійні середовища.