МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТ імені ВОЛОДИМИРА ДАЛЯ

Факультет інформаційних технологій та електроніки Кафедра комп’ютерних наук та інженерії Освітньо-кваліфікаційний рівень бакалавр Спеціальність 123 «Комп'ютерна інженерія»

(шифр і назва)

КУРСОВИЙ ПРОЕКТ (РОБОТА)

З дисципліни: Системне програмне забезпечення (назва дисципліни)

на тему: Розроблення програмних додатків за технологією «клієнт-сервер». Метод комунікації : поштова скринька.

здобувача вищої освіти 3 курсу , групи КІ-19д

(прізвище та ініціали) (підпис)

Керівник проекту

доц.Деркач М.В.

(посада, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Результати захисту:

Національна шкала Кількість балів: Оцінка: ECTS

Члени комісії:

(прізвище та ініціали) (підпис)

(прізвище та ініціали) (підпис)

(прізвище та ініціали) (підпис)

Дата

Сєвєродонецьк – 2021

Східноукраїнський національний університет імені Володимира Даля

Факультет ІТЕ Кафедра КНІ Спеціальність 123 Комп’ютерна інженерія

ЗАТВЕРДЖУЮ:

Зав. кафедри

КНІ

І.С.Скарга-Бандурова

« » 20 р.

**З А В Д А Н Н Я**

**НА КУРСОВУ РОБОТУ**

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розроблення програмних додатків за технологією

«клієнт-сервер». Метод комунікації : поштова скринька.

затверджена наказом по інституту від « » 2020 р. №

1. Термін подання закінченого проекту (роботи) 17 тиждень
2. Початкові дані до проекту (роботи) Розробити окремі додатки: додаток- сервер і два додатки-клієнти. Функції додатку-серверу :а) створення поштової скриньки; б) отримання даних від додатків-клієнтів; в) вивід на екран отриманих даних. Функції додатку-клієнту 1: а) відкрити поштову скриньку; б)підготувати і передати серверу такі дані: ім'я комп’ютера, ім'я користувача, версію операційної систем. Функції додатку-клієнту 2: а) відкрити поштову скриньку; б)підготувати і передати серверу такі дані: відсоток фізичної пам’яті, що використовується; відсоток віртуальної пам'яті, що використовується. За необхідності синхронізації процесів вибраний засіб синхронізації обґрунтувати.
3. Зміст розрахунково-пояснювальної записки (перелік питань, які підлягають розробленню) структура та оформлення пояснювальної записки повинна відповідати вимогам методичних вказівок до курсової роботи і ДСТУ. Основна частина записки повинна вміщувати: постановку задачі; короткі теоретичні відомості з теми курсової роботи (сутність, область застосування, переваги и недоліки архітектури «клієнт-сервер»; огляд методу комунікації через поштову скриньку); розробку та опис алгоритму роботи додатків; результати застосування додатків у вигляді копій екрану. У додатку до записки навести роздруківку програмних додатків.
4. Перелік графічного матеріалу (з точною вказівкою обов'язкових креслень) графічні матеріали до курсової роботи не вимагаються
5. Дата видачі завдання 4 тиждень

КАЛЕНДАРНИЙ ПЛАН

|  |  |  |  |
| --- | --- | --- | --- |
| № з/п | Назва етапів курсового проектування | Строк виконання  етапів | Примітка |
| 1 | Підготовчий етап. Уточнення постановки задачі. Аналітичний огляд науково-технічної  літератури | 24.09.2020 –  27.09.2020 |  |
| 2 | Проектний етап. Розроблення алгоритмів роботи програми. Обґрунтування необхідності і  вибір засобів синхронізації | 28.09.2020 –  09.10.2020 |  |
| 3 | Реалізаційний етап. Реалізація розроблених алгоритмів засобами вибраної мови  програмування, остаточне налагодження | 10.10.2020 –  30.11.2020 |  |
| 4 | Оформлювальний етап. Оформлення  пояснювальної записки відповідно до вимог методичних вказівок і ДСТУ | 01.12.2020 –  14.12.2020 |  |
| 5 | Заключний етап. Захист курсової роботи. | до  24.12.2020 |  |

Здобувач вищої освіти

( підпис ) (прізвище та ініціали)

Керівник проекту (роботи) Деркач М.В.

**РЕФЕРАТ**

Пояснювальна записка до курсової роботи містить: 30 сторінок, 5 рисунків, 3 додатка, 5 джерел.

**Мета роботи:** розробка програм на платформі Win32 для дослідження взаємодії між процесами через метод поштової скриньки.

**Об’єкт розробки** – програмне забезпечення, що містить програму виконуючу функції сервера і дві програми, виконуючи функції клієнтів. Розробка програм виконується на платформі Win32 у середовищі Microsoft Visual Studio, з використанням для між процесорної взаємодії метод поштової скриньки (mailslots).

Архітектура «клієнт-сервер» застосовується для організації обміну інформації між двома об’єктами.

У результаті розробки курсової роботи було визначено переваги та недоліки архітектури «клієнт-сервер»; зроблено огляд системних засобів комунікації і синхронізації процесів. Розроблено графічний інтерфейс користувача. Налагоджена програма.

WIN32 API, MAILSLOTS, ПОШТОВА СКРИНЬКА, МЬЮТЕКС, БУФЕР, ПРОЦЕСИ, МІЖПРОЦЕСОРНІ ВЗАЄМОДІЇ, АЛГОРИТМ, ПРОГРАМА.

# ЗМІСТ

[**ВСТУП 5**](#_Toc73948352)

[**1.** **АНАЛІЗ І ПОСТАНОВКА ЗАДАЧІ 6**](#_Toc73948353)

[1.1 Визначення ООП 6](#_Toc73948354)

[1.2 Основні поняття Об'єктно-Орієнтованого Програмування 6](#_Toc73948355)

[1.3 Регулярні вирази 12](#_Toc73948356)

[1.4 Технічне завдання на розробку 13](#_Toc73948357)

[**2.** **ПОБУДОВА ТА РЕАЛІЗАЦІЯ ПРОГРАМИ 14**](#_Toc73948358)

[2.1 Вибір засобів розробки 14](#_Toc73948359)

[2.2 Реалізація базового класу “TextStats” 15](#_Toc73948360)

[2.3 Реалізація похідного класу “Csymbol” 15](#_Toc73948361)

[2.4 Реалізація похідного класу “CsymbolNonSpace” 16](#_Toc73948362)

[2.5 Реалізація похідного класу “Cletter” 16](#_Toc73948363)

[2.6 Реалізація похідного класу “Cwords” 17](#_Toc73948364)

[2.7 Реалізація похідного класу “Csentence” 18](#_Toc73948365)

[2.8 Реалізація похідного класу “Cnumbers” 18](#_Toc73948366)

[2.9 Реалізація похідного класу “CtheNumbers” 19](#_Toc73948367)

[2.10 Реалізація класу “CText” 19](#_Toc73948368)

[**2.10.1 Реалізація конструктору класу** 20](#_Toc73948369)

[**2.10.2 Реалізація деструктору** 20](#_Toc73948370)

[**2.10.3 Реалізація функції “loadFile”** 20](#_Toc73948371)

[**2.10.4 Реалізація функції “calculate”** 21](#_Toc73948372)

[**2.10.5 Реалізація функції “print”** 22](#_Toc73948373)

[2.11 Реалізація головної функції програми 22](#_Toc73948374)

[2.12 Приклади результатів 22](#_Toc73948375)

[ВИСНОВКИ 24](#_Toc73948376)

[**ВИКОРИСТАНІ ДЖЕРЕЛА: 25**](#_Toc73948377)

[**Додаток А 26**](#_Toc73948378)

**ВСТУП**

Зі зростанням розвитку технологій та поширеності Інтернету робота з віддаленою інформацією стала звичним явищем. Технологія клієнт-сервер широко застосовується під час роботи з віддаленою інформацією в мережі. Термін «клієнт-сервер» застосовується до архітектури програмного забезпечення, що описує розподіл процесу виконання за принципом взаємодії двох програмних процесів, один із яких у цій моделі називався «клієнтом», а інший – «сервером». Ця модель утворює єдину систему, що забезпечує розподілені обчислення, аналіз та представлення даних. Клієнт відправляє запит на сервер, де він обробляється, і готовий результат зворотно відправляється клієнтові. У можливості сервера також входить одночасне обслуговування кількох клієнтів.

Клієнт-серверну архітектуру можна визначити, як концепцію інформаційної мережі в якій основна частина її ресурсів зосереджена на серверах, які обслуговують своїх клієнтів. Така архітектура визначає такі компоненти:

* **Клієнт**,який виконує надсилання запиту на сервер для можливості надання даних або виконання певної групи дій;
* **Сервер**, який виконує роботу за клієнтськими запитами, вирішує певного кола завдань та надає користувачам доступ до певних системних ресурсів;
* **Мережа**, яка забезпечує взаємодію між клієнтами та серверами.

Фактично клієнт та сервер – це програмне забезпечення. Зазвичай ці програми розташовані на різних обчислювальних машинах і взаємодіють між собою через обчислювальну мережу за допомогою мережевих протоколів, але можуть бути розташовані також і на одній машині.

Архітектура клієнт-сервер визначає лише загальні принципи взаємодії між комп'ютерами, деталі взаємодії визначають різні протоколи. Дана концепція розділяє на клієнтські пристрої, яким завжди щось треба і серверні, які дають те, що треба.

В результаті виконання даної курсової роботи необхідно:

* визначити метод налагодження міжпроцесової взаємодії, та розробити алгоритми побудови додатків сервера і клієнтів;
* розробити та налагодити сервер та клієнти;
* скласти алгоритми основних модулів програми;
* провести аналіз роботи розроблених програм.

1. **АНАЛІЗ І ПОСТАНОВКА ЗАДАЧІ**
   1. **Визначення ООП**
   2. **Основні поняття Об'єктно-Орієнтованого Програмування**
   3. **Регулярні вирази**
   4. **Технічне завдання на розробку**
2. **ПОБУДОВА ТА РЕАЛІЗАЦІЯ ПРОГРАМИ**
   1. **Вибір засобів розробки**
   2. **Реалізація базового класу *“*TextStats*”***
   3. **Реалізація похідного класу “Csymbol”**
   4. **Реалізація похідного класу “CsymbolNonSpace”**
   5. **Реалізація похідного класу “Cletter”**
   6. **Реалізація похідного класу “Cwords”**
   7. **Реалізація похідного класу “Cnumbers”**
   8. **Реалізація похідного класу “CtheNumbers”**
   9. **Реалізація класу “CText”**
      1. **Реалізація конструктору класу**
   10. **Реалізація головної функції програми**
   11. **Приклади результатів**

**ВИСНОВКИ**

.

ВИКОРИСТАНІ ДЖЕРЕЛА:

1. Регулярні вирази [Електронний ресурс] // slideshare.net. Режим доступу: www. URL: <https://www.slideshare.net/bulgakovdmitriy/c-37604265>
2. Регулярні вирази [Електронний ресурс] // Вікіпедія. Режим доступу: www. URL: <https://ru.wikipedia.org/wiki/Регулярные_выражения>
3. Основи регулярних виразів [Електронний ресурс] // ravesli.com. Режим доступу: www. URL:
4. <https://ravesli.com/regulyarnye-vyrazheniya-osnovy/#toc-7>
5. ООП [Електронний ресурс] // Вікіпедія. Режим доступу: www. URL: <https://ru.wikipedia.org/wiki/Объектно-ориентированное_программирование>
6. Основні принципи ООП [Електронний ресурс] // habr.com. Режим доступу: www. URL: <https://habr.com/ru/post/87205/>
7. Модифікатори доступу ООП [Електронний ресурс] // javarush.ru. Режим доступу: www. URL: <https://javarush.ru/groups/posts/1988-modifikatorih-dostupa-private-protected-default-public>
8. ООП [Електронний ресурс] // programming.in.ua. Режим доступу: www. URL: <http://programming.in.ua/programming/basisprogramming/25-oop.html>
9. Основні поняття ООП [Електронний ресурс] // works.doklad.ru. Режим доступу: www. URL: <https://works.doklad.ru/view/o_odpt9kmPo/all.html>
10. Основні принципи ООП [Електронний ресурс] // studme.org. Режим доступу: www. URL: <https://studme.org/341972/informatika/osnovnye_printsipy>
11. Абстракція даних [Електронний ресурс] // Вікіпедія. Режим доступу: www. URL: <https://ru.wikipedia.org/wiki/Абстракция_данных>
12. Основи ООП [Електронний ресурс] // schoolboyprog10.blogspot.com. Режим доступу: www. URL: <https://schoolboyprog10.blogspot.com/p/blog-page_44.html>
13. Спадкування [Електронний ресурс] // metanit.com. Режим доступу: www. URL: <https://metanit.com/cpp/tutorial/5.10.php>
14. Управління доступом. Інкапсуляція [Електронний ресурс] // metanit.com. Режим доступу: www. URL: <https://metanit.com/cpp/tutorial/5.4.php>
15. Конструктори і ініціалізація об'єктів [Електронний ресурс] // metanit.com. Режим доступу: www. URL: <https://metanit.com/cpp/tutorial/5.2.php>

Додаток **А**

Загальний лістинг коду