МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**Національний університет «Запорізька політехніка»**

Кафедра програмних засобів

(найменування кафедри)

**КУРСОВИЙ ПРОЕКТ**

з дисципліни «Об’єктно-орієнтоване програмування»

(назва дисципліни)

на тему: «Чат-бот для підтримки міжнародної академічної мобільності»

Студентів 2 курсу групи КНТ-229

спеціальності Комп'ютерні науки

Сепетий О. В.

(прізвище та ініціали)

Колбаса Д. О.

(прізвище та ініціали)

Ващенко М. І.

(прізвище та ініціали)

Момот В. І.

(прізвище та ініціали)

Керівник к.т.н., проф. Табунщик Г.В.

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

Національна шкала

Кількість балів: \_\_\_\_\_\_\_ Оцінка: ECTS

Члени комісії Табунщик Г. В.

(підпис) (прізвище та ініціали)

Миронова Н. О.

(підпис) (прізвище та ініціали)

Каплієнко Т. І.

(підпис) (прізвище та ініціали)

м. Запоріжжя

2020 рік

Зміст

//Зробиться після того, як вся пз буде завершена

перелік умовних позначень, символів, одиниць, скорочень і термінів

ООП - об’єктно-орієнтоване програмування.

Е.Г.І. - елементи графічного інтерфейсу

АМ – академічна мобільність

ЗВО — заклад вищої освіти

ПЗ — програмне забезпечення

STL – Standart Template Library

MinGW – Minimal GNU for Windows

GCC – GNU Compilers Collection

**ЗАВДАННЯ  
на курсовий проект студентів**

Сепетого О. В., Колбаси Д. О., Ващенко М. І., Момота В. І.

**1. Тема проекту:** Чат-бот для підтримки міжнародної академічної мобільності

**2. Термін здачі студентами закінченого проекту:** 14 грудня 2020

**3. Загальні вимоги до курсового проекту:** Розроблений програмний продукт повинний бути новим, розробленим з чистого листа. Для візуалізації поставленого завдання використовувати графічний інтерфейс із використанням інструментальних засобів відповідного середовища розробки. Роботу слід виконати в середовищі розробки Qt Creator, Visual Studio, IntelliJ IDEA. Мови програмування ‒ C++, C#, Java. Для групової роботи використовується GitHub

**Вимоги до інтерфейсу**

* + Необхідно розробити мінімум п'ять форм інтерфейсу, дизайн інтерфейсу.
  + Обов’язково необхідно реалізувати допомогу щодо використання програми, інформацію про розробників.

**Вимоги до класів**

* + Ієрархія класів повинна включати мінімум дванадцять класів, з яких один ‒ абстрактний, три ‒ розроблені класи користувача. Опис класів необхідно оформити у вигляді окремих модулів.

**Вимоги до зберігання даних**

* + Інформація зберігається в звичайних файлах. Інформація завантажується в пам'ять програми при старті, редагується в процесі роботи і зберігається в файли при завершенні. Формат файлу вибирається самостійно.
  + При використанні CSV, JSON та XML будуть додаткові питання на захисті з даних форматів представлення даних.
  + Необхідно взаємодіяти з CSV, JSON або XML (2 з трьох на вибір студентів).
  + За бажанням можна також використовувати базу даних. При використанні СКБД будуть додаткові питання на захисті по завантаженню даних таблиць з бази даних в структуру даних.

**Вимоги до алгоритмів**

* + Необхідно розробити авторських 2-4 алгоритми, наприклад, пошуку або сортування, та порівняти їх за швидкістю виконання зі стандартними реалізаціями.

**Вимоги до оброблення виняткових ситуацій**

* + Для оброблення виняткових ситуацій необхідно розробити авторський клас.
  + Програма повинна обробляти не менш ніж 10 різних помилкових ситуацій.

**4. Зміст розрахунково-пояснювальної записки:**

1. Аналіз предметної області;

2. Аналіз програмних засобів;

3. Основні рішення з реалізації компонентів системи;

4. Керівництво програміста;

5. Керівництво користувача;

Висновки

Додаток А. Код програми

Додаток Б. Інтерфейс програми

Додаток В. Слайди презентації

**5. Дата видачі завдання:** 11 вересня 2020 року

Вступ

Під час навчання в університеті, у студентів є можливість спробувати себе у закордонних університетах. На жаль про це мало говорять викладачі та куратор. Деяку інформацію можна знайти на офіційному сайті університету, але це не так зручно і зрозуміло, адже наш сайт має дуже багато розділів і часом там легко загубитися або важко знайти потрібний розділ. Також за нашим баченням на сайті не були розкриті деякі ключові питання типу ціни, мови та іншого.

Вирішенням даної проблеми є створення спеціалізованого чат-бота для підтримки міжнародної академічної мобільності. Найголовніша мета нашого чат-бота - донести цікаву, важливу та легку в сприйняттіінформацію до користувача та максимально залучити до участі в АМ.

Чат-боти – це універсальні помічники, які можуть бути використані у різноманітних сферах людського життя, в тому числі і наданні інформації про академічну мобільність.

Мета даного курсового проекту є – навчитись використовувати засоби об'єктно-орієнтованого програмування для створення чат-ботів.

Проведено аналіз предметної області, досліджено аналогічне програмне забезпечення, а також існуючі методи та програмні засоби для вирішення завдання.

Для реалізації програмного продукту використовувалася мова програмування С++ та середа розробки Qt Creator з її власними зручними класами.

Здійснено опис прийнятих рішень, наведено опис полів та методів реалізованих класів.

1 Аналіз предметної області

В даному розділі розглянуті існуючі методи вирішення завдання підтримки АМ та програми, що реалізують дану потребу, а саме чат-боти що спрямовані на спрощення взаємодії між студентами та університетом, серед можливостей яких, зокрема є підтримка АМ .

1.1 Аналіз академічної мобільності, як процесу що потребує автоматизації

Міжнародна академічна мобільність являє собою короткострокові програми обміну: кредитна мобільність на навчання (3-12 міс.) та на практику (2-12 міс.) для українських студентів: (молодших) бакалаврів, магістрів, аспірантів, докторантів.

Академічна мобільність для викладання або підвищення кваліфікації або стажування (від 5 днів до 2 міс.) для викладачів та працівників закладів вищої освіти.

Організація цієї події проходить між одним ЗВО з країни-члена Програми та одним ЗВО країни-партнера Програми Еразмус+.

Заявником - координатором проектів з мобільності виступають заклад вищої освіти країн-членів програми Еразмус+.  
 Координують напрям діяльності Національні Агентства Еразмус+ країн-членів Програми (децентралізований напрям).

Брати участь у АМ можуть: студенти (молодші бакалаври, бакалаври, магістри, аспіранти, докторанти), викладачі і адміністративний персонал.  
 Щоб взяти участь, необхідно пройти співбесіду з представником університету, в якому особа навчається, або працює і з представником партнерського університету. Але першими і головними кроками є розуміння що таке АМ і реєстрація для проходження співбесіди. Для полегшення цих кроків і створений наш чат-бот з підтримки АМ.

1.2 Огляд існуючих методів вирішення завдання та аналогів програмного забезпечення

***1.2.1 Передмова***

Звісно одним з найочевидніших шляхів донесення студентів (викладачів та інших працівників закладів вищої освіти) є загальна презентація про процес АМ та процес взяття в ньому участі. Але недоліком цього методу витрати часу на підготовку цього заходу і на власне презентування.

Іншим методом є застосування чат-бота, що доступний цілодобово і має відповіді на всі запитання загального характеру, який швидко і зрозуміло донесе до людини потрібну їй інформацію, а також зможе надати можливість реєстрації на співбесіду і сформує перелік охочих.

Перед розглядом аналогів, треба зазначити, що програмного забезпечення, яке б задовольняло назві “Програмне забезпечення для підтримки академічної мобільності” або “Чат-бот для підтримки академічної мобільності” не знайшлося, що свідчить про те, що конкретних аналогів немає або їх дуже мало або про те що ПЗ з таким функціоналом нині не є дуже затребуваним. Підтримка АМ є частиною декількох знайдених чат-ботів. Але слід зазначити, що ПЗ, яке немає цього функціоналу аналізувалося на предмет загальної організації побудови взаємодії з користувачем.

Програмами для підтримки академічної мобільності є:

* Програма “MoveOn”;
* “Chatbot TARS”;

Чат-бот, що не має підтримки АМ, але є цікавими прикладами ПЗ для університетів:

* “Virtual spirits chat bot”;

Критерії аналізу аналогів ПЗ:

* точне донесення до користувача інформації щодо АМ, або університет (якщо чат бот не підтримує АМ)
* побудова діалогу з користувачем;
* наявність безкоштовної ліцензії;
* зручність і простота інтерфейсу.

1.2.2 Програма “MoveOn”

Компанія-видавець: QS UniSolution

Посилання: https://www.qs-unisolution.com/moveon/

Короткий опис: MoveON - це платформа на основі аналітичних даних для міжнародних офісів вищих навчальних закладів, для покращення процесів їхньої мобільності та ефективніше відстеження міжнародних партнерських відносин. Вона надає можливість створити свій шлях обміну студентами та побудувати стратегічне партнерство для досягнення цілей інтернаціоналізації закладу.

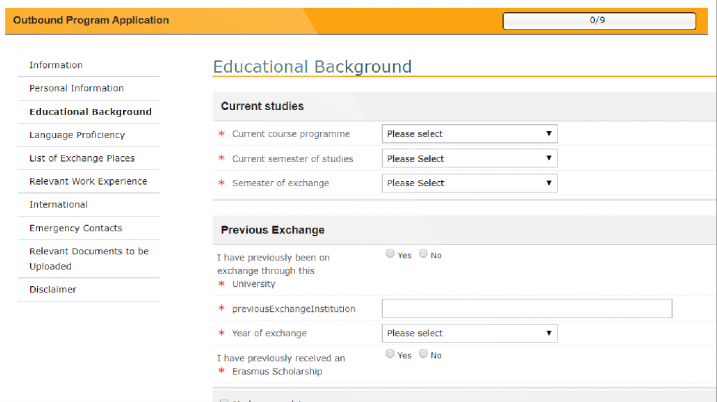
Переваги:

* Запуск програми проходить швидко і легко
* Має досить простий і зрозумілий інтерфейс
* Дозволяє легко керувати програмами обміну студентів
* Співпрацює з Еразмус+

Недоліки

* ПЗ спрямовано на полегшення роботи адміністрації університету, але ні як не на комунікацію зі студентами.
* Має платну ліцензію

Робота програми продемонстрована на рисунку 1.1.

 Рисунок 1.1 – Робота програми «MoveOn»

**1.2.3 Chatbot TARS**

Компанія-видавець: Tars.

Посилання:https://hellotars.com/chatbot-templates/education/SJmmXq/  
university-search-chatbot-for-studying-abroad

Чат-бот допомагає навчальним закладам автоматизувати процес планування зустрічей, щоб заощадити час та зробити більш привабливими форми реєстрації на курси

Переваги:

* Чат-бот підтримує АМ
* Діалог з користувачем проходить доволі зручно і зрозуміло
* Головним способом комунікації з програмою є панель кнопок, що робіть комунікацією простою
* має безкоштовну пробну версію

Недоліки:

* Не надає інформацію про університети.

Робота програми продемонстрована на рисунку 1.2.

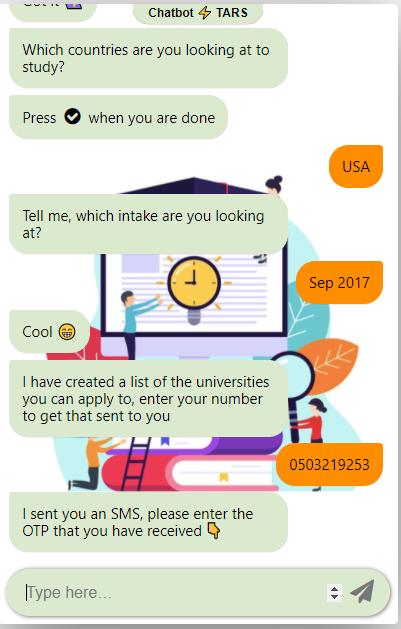


Рисунок 1.2 – Робота з програмою «Chatbot TARS»

1.2.4 «Virtual spirits chat bot»

Компанія-видавець: Virtual spirits.

Посилання: https://www.virtualspirits.com/chatbot-for-university.aspx

Короткий опис: Чат-бот використовується для підтримати студентів та надання миттєвої відповіді на послуги кампусу, реєстрацію, стипендії, навчальні послуги, поради щодо іспитів, ІТ-послуг, усунення несправностей тощо

Переваги:

* Має широкий набір функціоналу.
* Зручний інтерфейс
* Взаємодія виконується за допомоги текстових повідомлень і іноді кнопок (категорій).

Недоліки:

* Нажаль не надає підтримки АМ

Робота програми продемонстрована на рисунку 1.3.

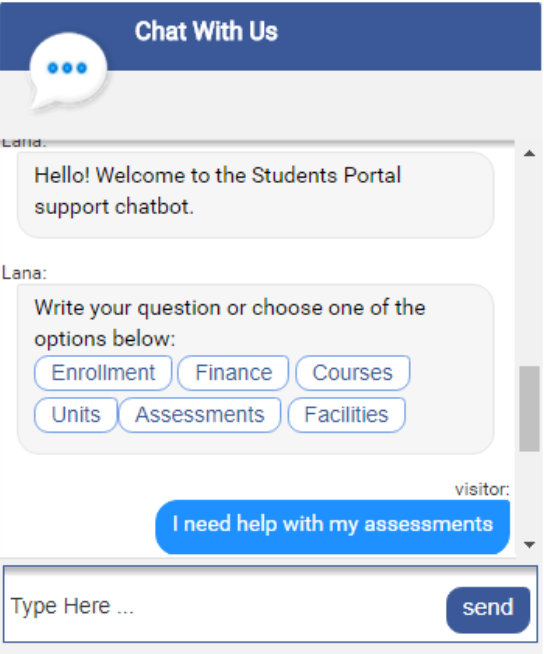


Рисунок 1.3 – Робота з програмою «Virtual spirits chat bot»

2 Аналіз програмних засобів

В даному розділі розглянуті основні особливості програмних засобів, за допомогою який реалізовано курсовий проект.

2.1 Огляд особливостей мови програмування

С++ - мова програмування загального призначення.

С++ підтримує об’єктно-орієнтовану парадигму, що дозволяє представити програму, як набір об’єктів, кожен з яких має свої властивості, поведінку і своє призначення в програмі. Програму, написану з використанням ООП легше модифікувати і підтримувати. Однією з особливостей саме мови С++ в тому, що вона підтримує множинне спадкування. С++ підтримує програмування з використанням шаблонів, що спрощує роботу з даними. Бібліотека Qt, що дозволяє розроблювати графічний інтерфейс, має свою реалізацію STL, свою реалізацію потоків вводу-виводу, та інших надбань стандартної бібліотеки з більшою кількістю можливостей, що робить процес розробли ПЗ більш швидким та продуктивним.

2.2 Огляд особливостей обраного компілятору

Компілятор MinGW – версія компілятора GCC, створена для розробки ПЗ мовою С++ на операційних системах сімейства Windows. Розроблене нами ПЗ не використовує жодних специфічних речей, що можуть робити Unix-подібні операційні системи, чого не може робити Windows, тому необхідність у використанні компілятора Cygwin відпала. Компілятор MSVC постачається разом з Microsoft Visual Studio, але на наш погляд не має істотних переваг. До того ж робота в Visual Studio потребує додаткових операцій з підключенням бібліотеки Qt.

2.3 Огляд класової ієрархії

QString — Клас динамічного типу даних для роботи зі рядками тексту. Використовується для зберігання інформації (рядки тексту), що зчитується з файлів, вводиться користувачем.

QFile – Клас для введення-виведення для зчитування та запису текстових і двійкових файлів і ресурсів. Використовується для роботи з файлами, зчитування та запису текстової інформації, а також для підключення таблиць стилів.

QList – Контейнерний клас Qt на основі двозв’язного списку. Дозволяє виконувати операції вставки та видалення елементу за постійний час. Використовується для зберігання даних про студентів і університети.

QStringList — Нащадок класу QList, але на відміну від нього не є шаблонним класом, тобто це QList з типом заповнювачем QString. Він також дозволяє виконувати операції вставки та видалення елементу за постійний час. Використовується для зберігання рядків тексу які задіяні у локалізації.

QDomDocument – Клас, що реалізує собою роботу з XML документом. За своєю суттю є коренем дерева документа. Використовується для зчитування та запису інформації в XML документ.

QDateTime — Клас, що надає функції дати та часу, тобто об’єднує класи QDate і QTime. Використовується для визначення часу відправки повідомлення.

QRegExp - Клас, що надає порівняння зі зразком за допомоги регулярних виразів. Використовується для зберігання регулярного виразу.

QRegExpValidator — Клас, використовуються для перевірки відповідності рядка регулярному виразу.

2.4 Висновки з розділу

У даному розділі було описано мову програмування та її особливості. Також було описано класи, що були використані в процесі розробки ПЗ та коротко для чого саме були вони використані.

3 основні рішення з реалізації компонентів системи

3.1 Основні рішення щодо розроблених класів

3.2 Основні розроблені алгоритми

3.3 Основні рішення щодо розробки інтерфейсу

3.3.1 Вибір платформи.

Головним рішенням, щодо розробки інтерфейсу, було вибрати нову та сучасну платформу для розробки інтерфейсу Qt. Ця платформа відзначається кросплатформеністю та більш зручним інтерфейсом для програміста.

3.3.2 Основні рішення з впорядкування елементів інтерфейсу

Під час розробки інтерфейсу користувача було використано наступні е.г.і.:

* MainWindow
* VHLayouts
* TextBrowser
* PushButton
* Label
* LineEdit
* MenuBar
* TextEdit
* Frame
* ComboBox

Саме вони були використані тому що, по перше, це одні з найпростіших та найпопулярніших е.г.і. в цілому, по друге, до цих елементів доволі просто застосовувати стилі та інші графічні зміни, по третє, ми вже маємо досвід роботи з цими е.г.і.

Для простоти і лаконічності програми було створено 6 форм, а саме:

* mainwindow.ui
* bug.ui
* developer.ui
* manual.ui
* rateapp.ui
* settings.ui

**mainwindow.ui**

Тут розташовується саме вікно чату, кнопки для взаємодії з ботом, рядок введення і меню для переходу до наступних форм. Було прийнято рішення помістити кнопки в різні шари і відображати конкретні групи при натисканні на відповідну за змістом кнопку. Зроблено це для економії місця і не нагромадження інтерфейсу.

**bug.ui**

У цій формі користувач може ввести повідомлення про знайдену помилку в програмі або повідомити якусь свою пропозицію розробнику.

**developer.ui**

Тут можна знайти основну інформацію про програму, її версію і розробників.

**manual.ui**

Коротка і зрозуміла інструкція для користувача як взаємодіяти з програмою.

**rateapp.ui**

В даному вікні можна оцінити чат-бота.

**settings.ui**

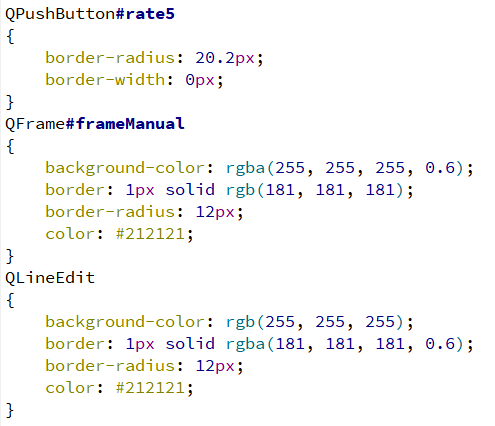
Мабуть одне з найбільш функціональних вікон крім головного – налаштування програми під себе. Можна вибрати колірну тему програми (темну\світлу) і мову інтерфейсу програми. Ми постаралися і локалізували під 4 мови: українська, англійська, німецька та російська. Всі зміни застосовуються в реальному часі. Це досить важливо, тому що користувач не витрачає час на перезапуск програми.

3.3.3 Основні рішення з комунікації з користувачем

Для кращої комунікації з користувачем та рівномірного розподілення місця у додатку були використані кнопки(PushButton) для запиту певної стандартної інформації у бота, та строка вводу (TextEdit, LineEdit) для нестандартної інформації такої як: особиста інформація користувача для надання ії керівникам АМ, або повідомлення про помилку\баг у додатку.

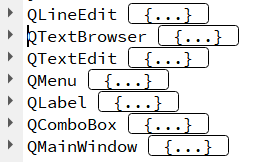
3.3.4 Основні рішення з створенням інтерфейсу

Було активно використано стилі. Тобто відокремлення певної групи елементів і створення для них власного стилю. Приклад даного рішення наведений нижче на рисунку

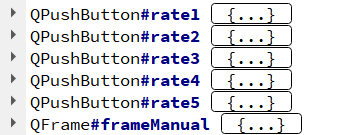


Основним варіантом застосування стилів є використання Qt Style Sheets. Як стає очевидно з назви, це трохи специфічний аналог звичних всім каскадних таблиць стилів CSS(Cascading Style Sheets), без яких немислимий сучасний інтернет. Оскільки в минулому семестрі у нас була дисципліна «WEB-дизайн» і ми активно використовували CSS в своїх роботах, то нам здалося зручним використовувати даний аналог і тут. Також це зручно в тому плані, що всі графічні рішення виводяться в окремий файл, не займають місце в основному коді і довантажуються під час запуску програми, а також потрібний стиль застосовується при виборі теми програми в Налаштуваннях.

Було використано звернення по класу. Це зручно тим, що зміни застосовуються відразу до всіх елементів даного класу. Приклад:



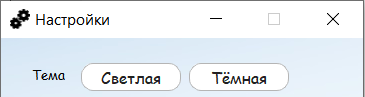
Звернення по імені дає можливість кастомізувати конкретний елемент з усього класу. Приклад:



Псевдо-стан елемента допомагає зробити програму більш динамічною і "живою" для користувача. Приклад:



Також було прийнято рішення додати повністю альтернативну тему для усього додатку, т.з. темний стиль окрім світлого. Ця ідея прийшла до нас у процесі створення програми вночі та ми подумали, що користувачеві буде зручно змінювати колірний стиль усього додатку одним кліком у Налаштуваннях.

3.4 Основні рішення щодо розробки бази даних

4 керівництво програміста

5 керівництво користувача

Висновки

ПЕРЕЛІК ПОСИЛАНЬ

Додаток А Діаграма класів

Додаток Б Код Програми

Додаток В Зображення Програми