|  |  |  |  |
| --- | --- | --- | --- |
| МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Тараса Шевченка  ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  **Кафедра програмних систем і технологій**        Дисципліна  **«Емпіричні методи програмної інженерії»**      **Практична робота № 1**  **«ПЛАНУВАННЯ ЕМПІРИЧНИХ ДОСЛІДЖЕНЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»** | | | |
| **Виконав:** | Гоша Д. О | **Перевірила**: | Юрчук Ірина Аркадіївна |
| Група | ІПЗ-23 | Дата перевірки |  |
| Форма навчання | денна | Оцінка |  |
| Спеціальність | 121 |
| 2022 | | | |

# Виконання лабораторної роботи

1. Емпіричні методи інженерії програмного забезпечення мають в собі сукупність дій для отримання знань з метою кращого розуміння аспектів розробки програмного забезпечення. Результатом дій є ряд тверджень щодо визначеного переліку проблем. Ці твердження являються відповідями на поставлені запитання та підтвердженням чи спростуванням гіпотез. Предметом емпіричних досліджень – є дослідження процесів (Життєвий Цикл ПЗ), продуктів ІПЗ та самих методів досліджень. Основна ідея емпіричних досліджень в ПЗ - фокусування на процесах досліджень, проектуванні та структурованості емпіричних досліджень, а не на змісті чи результаті.
2. Емпіричне дослідження включає в собі такі етапи:
   * Поставлення мети (Мета має бути конкретна, вимірювана, досяжна, реалістична, обмежена в часі)
   * Висування гіпотези (Спростовуваність пропонованої гіпотези, загальність прогностичний потенціал, принципова логічна простота, спадковий зв’язок гіпотези)
   * Добування фактів (добування фактів, обробка отриманих даних, аналіз)
   * Обробка отриманих даних (ціль-питання-метрика, модель відповідальної особи, стандартизовані метрики)
   * Аналіз даних (Спостереження, індукція, дедукція, перевірка матеріалу, оцінка)
   * Формування висновку (Підтвердження чи спростовування гіпотези)
   * Корегування чи зміна гіпотези
3. Мною було обрано 20 проектів, у 1 групі містяться 10 програмних кодів різних текстових редакторів на об’єктно орієнтованій мові програмування С#, а у 2 групі містяться теж тектові редактори , тільки на різних мовах.Цу популярні редактори з багато тисячними кодами, тому повноцінно оцінити якість архітектури я не зможу, на це попреедні розробники витрачали роки або десятиліття. Але ми зможемо провести кореляційний аналіз між пет-проектами (Власне студенскірозробки та міні проекти) з маштабними та професійними редакторами , якими користуються мільйони людей по всьому світу.

|  |  |
| --- | --- |
| 1. Blazored TextEditor (https://github.com/Blazored/TextEditor) | Вихідний код це веб застосунок , написаний на плаформі asp.net з додаванням JS. Тобто призначений для інтегрування на веб сервіси з подальшим спрощенням написання контенту адміністраторами.  Код виглядає яксіно та лаконічно, немає нічого зайвого , все закоментовано.  Що до інтерфейсу , він достатньо простий , зрозумілий але не функціональний , через замалу кількість можливостей. |
| 1. AvalonEdit (https://github.com/icsharpcode/AvalonEdit) | Застосунок , що повністью написаний на C# .net framework , базується на основі WPF. Проблему , що вирішує застосунок це недостатня функціональність textbox та richtextbox у WPF. Тому він створенйи для розширеня модливостей цих текстових полів , додаваня коліру , товщини та стилів тексту.  Код написаний дуже якісно , має свою власну документацію та веб сайт. Багато популярних проетів інтегрують його , це свідчить про якість ПЗ.  Інтерфейс, на мою думку, трохи застарілий , але при цьому зручний на простий у використані. |
| 1. Limetext (https://github.com/limetext/lime/tree/old\_master) | Редактор , проект якого включає декалька мов програмування , але теж об’єктно орієнтовний підхід , тож проаналізуємо його. Проект створений як конкурент Sublime Text 2 editor. Тому це аналог але з відритим вихідним кодом.  Двигун рендерінгу шрифтів використовує WebKit, тобто являє собою механізм веб браузеру, який вирішує багато пробем компановкою , кольором та високоякісним рендерінгом. Являється кроссплаформеним.  Інтерфейс нагадую Sublime , має своє фйлове дерево, підсвідка синтаксису та швидкий перехід до коду. Тому це не поганий редактор коду. |
| 1. FastColoredTextBox (https://github.com/PavelTorgashov/FastColoredTextBox) | Fast Colored TextBox — це компонент текстового редактора для .NET. Дозволяє створювати власний текстовий редактор з підсвічуванням синтаксису. Він добре працює з маленькими, середніми, великими та дуже-дуже великими файлами. Він має такі налаштування, як колір переднього плану, стиль шрифту, колір фону, які можна налаштувати для довільно вибраних текстових символів. За допомогою регулярних виразів можна легко отримати доступ до тексту. Також підтримуються WordWrap, Find/Replace, Code folding і багаторівневі скасування/повторення.  Код написан не погано , але не закоментований та не має власної документації.  Інтерфейс у прогамі відсутній. |
| 1. AvaloniaEdit (https://github.com/AvaloniaUI/AvaloniaEdit) | Цей проект є портом текстового редактора AvalonEdit на основі WPF для Avalonia. AvaloniaEdit підтримує такі функції, як: Підсвічування синтаксису за допомогою граматик TextMate Нумерація рядків Прокрутка нижче документа Гіперпосилання і багато іншого. Являється представником релакторів коду.  Має власну дркументацію та якісно написаний код , що повністю ролкоментован. Програма зручна до маштабування та код інтуєтивно зрозумілий.  Інтерфейс не гнучий , чимость нагадує VSCode. Тим самим дуже простий та зрозумылий для звичайних користувачів. |
| 1. Text Editor - Windows Forms (https://github.com/hjohnson12/TextEditor) | Простий текстовий редактор , написаний в цілях дипломної роботи о 2016 році. Являє собою WPF проект , з XAML інтерфейсом. Має звичайний функціонал такий як: зміна кольору тексту, шрифту розміру , гарячі клавіши.  Документації , як і коментарів не має. Код написан просто , в деяких місцях з незначними помилками , більш у функціональному стилі.  Інтерфейс простий написаний на XAML , чимось нагадує ворд, якщо б цого писали на він формах. Проте має повний необхідний функціонал , та інтуїтивно зрозумілий |
| 1. SuperbEdit (https://github.com/mattiascibien/SuperbEdit) | SuperbEdit — це вдосконалений текстовий редактор для програмістів з інтерфейсом із вкладками, написаним на C# та WPF. Він створеший для того, щоб бути надзвичайно настроюваним та розширюваним. Має модульну архітектуру. Фронтенд написаний на CSS та XAML.  Має документацію та Doxyfile . Код написаний професійно , протестований UNIT тестами. Присутні коментарі у коді , тож стороньому програмісту буде не складно інтегрувати деякий функціонал у вигляді модулів.  Інтерфейс розроблен чудово . Має оглядач об’єктів справа. Та підвсітку синтаксису коду. Являє собою повноцій редактор коду. Також функціонал включає спеціальні можливості у вигляді комбінацій клавіш для виконування деяких дій з тектом. |
| 1. EditorEngine | EditorEngine — це проект із відкритим кодом, який підтримує ContinuousTests команда www.continuoustests.com. Призначення цього інструменту – надати набір функцій різноманітних текстових редакторів. Він містить плагін для кожного підтримуваний редактор. Функції - це базовий зворотний зв'язок, як-от натискання клавіш і діїяк відкритий файл у спеціальному місці. Він був розроблений в основному як загальний бекенд для проекту OpenIDENet.  Код написаний нормально , без видимих помилок або неуточнень. Без коментарів та поясненьу самому коді , але певно, має власну документацію.  Інтерфейс відсутній. |
| 1. Simple-Editor-de-Texto (https://github.com/LuisArmando9/Simple-Editor-de-Texto) | Простий текстовий редактор, C# містить багато помилок, створіть прикладний сервер, щоб використовувати його.  Код написаний непогано. Але без коментарів та документації.  Інтерфейс мінімалістичний. Простий ,зрозумілий. |
| 1. MemoEditor (https://github.com/damulhan/MemoEditor) | Текстовий редактор на основі WPF з навігацією в дереві. Використання MVVMLight редактор пам'яток; лівий провідник файлів заснований на «Редагуванні TreeView на місці в MVVM. Цей проект складається з Treeview і простого компонента TextEdit. Перегляд дерева представляє папки, а текстовий редактор — їхні текстові файли (.txt). Це призводить до простого редактора нотаток у стилі MS Windows Explorer і файлового менеджера. Легко створюйте нагадування \*.txt, і воно автоматично зберігається, коли фокус дерева переміщується на інший вузол або завершується програма. А також натиснувши кнопку зберегти. Збережений файл знаходиться у файлі .txt, а не в базі даних.  Повністью написаний на C#. Хочу відмітити доволі не погану архітектуру проекту, що може слугувати зручною для маштабуваня та рерайту.  Інтерфейс чимось схожий на редактор коду + ворд , виконаний у мінімалістичному стилі , але поєдную широкий функціонал. |
| 1. Vi / Vim | Vim – це потужний текстовий редактор на основі командного рядка, який розширив функціональні можливості старого текстового редактора Unix Vi. Це один з найпопулярніших текстових редакторів серед системних адміністраторів і програмістів, тому багато користувачів часто називають його редактором для програмістів. Він підтримує підсвічування синтаксису під час написання коду або редагування конфігураційних файлів. |
| 1. Gedit | Це текстовий редактор загального призначення з графічним інтерфейсом, який встановлюється за умовчанням у робочому столі Gnome. Він простий у використанні, має в арсеналі потужні модулі, що підключаються, і сам по собі це потужний редактор з наступними функціями: Підтримка UTF-8 Розмір і колір шрифту, що настроюється. Підсвічування синтаксису, що настроюється Можливість скасування та повтору останніх дій Повернення файлів Видалене редагування файлів Пошук та заміна тексту Підтримка буфера обміну та багато іншого |
| 1. Nano | Nano - це простий у використанні текстовий редактор, хороший як для початківців, так і досвідчених користувачів Linux. Він підвищує зручність використання, надаючи прив'язку клавіш, що настроюється! Nano має такі особливості: Поєднання клавіш, що настроюються. Підсвічування синтаксису. Можливість скасування та повтору останніх дій. Повний дисплей на стандартному виводі. Підтримка Pager (за сторінковим виведенням) для читання форми стандартного введення |
| 1. GNU | Це текстовий редактор, що легко розширюється і налаштовується, який також пропонує інтерпретацію мови програмування Lisp. Що особливо важливо, це те, що можуть бути додані різні розширення для підтримки додаткових функцій редагування тексту. Emacs має такі функції: Хороша користувальницька документація та велика кількість посібників. Підсвічування синтаксису за допомогою кольорів навіть при редагуванні простого тексту. Підтримка безлічі мов. Різні розширення, включаючи пошту та новини, інтерфейс відладчика, календар та багато іншого… |
| 1. Kate/Kwrite | Kate - це багатофункціональний текстовий редактор з широкими можливостями розширення, який постачається з KDesktop Environment (KDE). Проект Kate спрямований на розробку двох основних продуктів: KatePart та Kate.  KatePart - це розширений компонент текстового редактора, включений в багато програм KDE, який може вимагати від користувачів редагування тексту, тоді як Kate - це текстовий редактор з підтримкою всього в кілька документів (MDI). Нижче наведено деякі із загальних особливостей Kate: Розширюється за допомогою сценаріїв. Підтримка кодування, така як режим Unicode. Рендеринг тексту у двонаправленому режимі. Підтримка закінчення рядка з функціями автоматичного визначення. |
| 1. Jed | Це також ще один редактор командного рядка з підтримкою GUI-подібних функцій, таких як меню, що випадають. Він розроблений спеціально для розробки програмного забезпечення, і однією з його важливих функцій є підтримка режиму Unicode. |
| 1. Geany | Geany пропонує базові функції IDE-подібні з акцентом на розробку програмного забезпечення з використанням інструментарію GTK+. Деякі з його функцій, наведені нижче: Підсвічування синтаксису. Змінний інтерфейс Підтримка великої кількості типів файлів . Навігація за кодом. Автозаповнення. Підтримка автоматичного закриття тегів HTML та XML Елементарна функціональність управління проектами та багато іншого. |
| 1. Bluefish | Bluefish – це простий у встановленні та використанні текстовий редактор, призначений для програмістів Linux та веб-розробників. Він пропонує широкий набір функцій, наведених нижче: Інтуїтивно зрозумілий інтерфейс. Інтеграція зовнішніх програм Linux, таких як lint, weblint, make та багато інших . Інтеграція фільтрів, таких як sed, sort, awk та багато інших…. Функція перевірки правопису. Підтримка паралельної роботи над кількома проектами Видалене редагування файлів. Пошук та заміна тексту. Можливість скасування та повтору останніх дій. Авто-відновлення змінених файлів |
| 1. VSCode | VSCode – це надійний сучасний безкоштовний текстовий редактор із відкритим вихідним кодом, створений Microsoft для комп'ютерів Linux, Mac та Windows. Він пропонує безліч потужних функцій, включаючи: Можливість налагодження в інтерактивній консолі, точки зупинки, стеки викликів тощо. Вбудована підтримка Git за допомогою команд Git IntelliSense 100% настроюваність інтерфейсу Підтримка безлічі мов прямо з коробки Макети, що перемикаються Вбудований термінал |
| 1. Notepad++ | Notepad++ – це текстовий редактор, що настроюється, побудований з акцентом на швидкість і мінімальний розмір програми для платформ Windows. Він розроблений на основі текстового редактора Scintilla, і його функціональність може бути розширена за рахунок плагінів. Його функції включають: Редагування у багатьох вкладках. Згортання коду (по блоково). Підтримка закладок. Карта документа. Perl-сумісні регулярні вирази |

# Висновки

В цій лабораторній роботі було розглянуто 20 проектів, в першій підгрупі знаходяться текстові редактори та редактори коду, на об'єктноо орієнтовній мові С#, в другій підгрупі розглядалось теж текстові редактори , але на мові програмування C++. Проекти в кожній групі різнились використаними технологіями та обсягом проекту. Було досліджено кожний проект та проведено аналіз функціональних можливостей вихідного коду , якості написаного коду, що вплине на зручність маштабування проекту іншими розробниками та читаємості коду. Також був проведен аналіз якості і зручності інтерфейсу, що відіграє важливу роль з точки зору користувача. Мною було зрозуміло основні етапи дослідження програмного забезпечення, було сформовано мету дослідження.