**Проектування**

 Вибір мови програмування

Згідно ТЗ, в даному курсовому проекті буде використовуватися мова програмування Python 3.

Python 3. - високорівнева мова програмування загального призначення, орієнтований на підвищення продуктивності розробника і читання коду. Синтаксис ядра Python мінімалістичний. У той же час стандартна бібліотека включає великий обсяг корисних функцій, таких як:

Math - стандратний модуль, що дозволяє проводити розширені арифметичні операції над змінними.

Time - стандартний модуль, що дозволяє працювати програмі з тимчасовими значеннями.

Fractions - модуль, що дозволяє проводити операції над раціональними числами, в тому числі пошук найбільшого загального дільника.

Numpy - модуль, що полегшує роботу з простими числами.

Вибір середовища розробки

При написанні програми буде використане середовище розробки PyCharm 2016.3 Professional, яка дозволяє писати програми на мові Python 3. PyCharm - це інтегроване середовище розробки для мови програмування Python. Надає засоби для аналізу коду, графічний відладчик, інструмент для запуску юніт-тестів і підтримує веб-розробку на Django. PyCharm розроблена компанією JetBrains на основі IntelliJ IDEA.

можливості:

• Статичний аналіз коду, підсвічування синтаксису і помилок.

• Навігація по проекту і вихідного коду: відображення файлової структури проекту, швидкий перехід між файлами, класами, методами і використаннями методів.

• Рефакторинг: перейменування, вилучення методу, введення змінної, введення константи, підйом і спуск методу і т. Д.

• Інструменти для веб-розробки з використанням фреймворку Django

• Вбудований відладчик для Python

• Вбудовані інструменти для юніт-тестування

• Розробка з використанням Google App Engine

• Підтримка систем контролю версій: загальний користувальницький інтерфейс для Mercurial, Git, Subversion, Perforce і CVS з підтримкою списків змін і злиття

## Вибір використаних технологій

## В даному курсовому проекті буде використовуватися інтерфейс програмування додатків PyQt 5.

## PyQt5 - це набір Python бібліотек для створення графічного інтерфейсу на базі платформи Qt5 від компанії Digia.і Він доступний для Python 2.x і 3.x.

## Бібліотека Qt є однією з найпотужніших бібліотек GUI (графічного інтерфейсу користувача).

## PyQt5 реалізований у вигляді набору python-модулів. Ця бібліотека має понад 620 класів і 6000 функцій і методів.

## Це мультиплатформенна бібліотека, яка працює на всіх основних операційних системах, в тому числі Unix, Windows і Mac OS.

# **Розробка**

# Вибір системної архітектури

# Системна архітектура додатка визначає, як взаємодіють елементи програми та які функції вони надають. Існують три типи системної архітектури: одно-, дво- і багаторівнева. Багаторівневі додатки. Багаторівневі додатки реалізуються розподілом по безлічі комп'ютерів в мережі. Додатки цього типу називають також розподіленими або n-рівневим. Подібне додаток є особливим випадком трирівневого, у якого один і більше рівнів розбивається на додаткові рівні, що забезпечує більш високу масштабованість додатки. У багаторівневих додатках презентаційний сервіс, прикладна логіка, і сервіс даних відокремлені один від одного. Однак рівнів може бути не три, а більше. Логічні рівні не обов'язково повинні відповідати їх фізичному місцезнаходженням в мережі. У багаторівневому додатку клієнту надається тільки для користувача інтерфейс. Прикладна логіка реалізується проміжним рівнем, який розміщується між призначеним для користувача інтерфейсом і системою зберігання даних. Це і дозволяє виділяти кожен тип сервісу в окремий рівень. Перевага такої моделі в тому, що прикладна логіка зосереджена в одному місці і може бути легко модифікована. Презентаційний рівень відповідає фактично тільки за взаємодію з користувачем. У багаторівневому додатку покупець не звертається до системи зберігання даних безпосередньо. Поділ всіх сервісів дозволяє модифікувати будь-який рівень системи, не вносячи зміни в інші.

# Однорівнева архітектура - архітектура, що підтримує як призначений для користувача інтерфейс, так і прикладну логіку. До останньої належать різноманітні математичні функції: перевірка, правопису та ін. На тому ж рівні містяться і підпрограми, які забезпечують збереження і доступ до файлів даних. Однорівнева системна архітектура не передбачає поділ на рівні.

# Дворівнева архітектура - архітектура, що підтримує як призначений для користувача інтерфейс, так і прикладну логіку, які при цьому реалізовані на різних рівнях.

# Для трирівневої архітектури характерна наявність призначеного для користувача інтерфейсу, реалізації рішення задачі, бази даних. Даний тип архітектури застосовується в програмному забезпеченні, яке здійснює роботу, з будь-якою базою даних.

# Проаналізувавши види систем архітектури, найкращим рішенням є вибір дворівневої архітектури для програми так як програма повинна реалізовувати підрахунок результату і взаємодіяти з користувачем через веб-інтерфейс.

Діаграма прецедентів

Діаграма прецедентів (англ. Use case diagram, діаграма варіантів використання) в UML - діаграма, що відображає відносини між акторами і прецедентами і є складовою частиною моделі прецедентів, що дозволяє описати систему на концептуальному рівні.

      Прецедент - можливість модельованої системи (частина її функціональності), завдяки якій користувач може отримати конкретний, вимірний і потрібний йому результат. Прецедент відповідає окремому сервісу системи, визначає один з варіантів її використання і описує типовий спосіб взаємодії користувача з системою. Варіанти використання зазвичай застосовуються для специфікації зовнішніх вимог до системи.

      Відповідно до завдання на курсовий проект була побудована UML діаграма прецедентів рисунок 3.2.



Рисунок 3.2-UML діаграма прецедентів

1) Користувач розпочинає проходження тесту.

2) В ході тестування користувач вибирає показники які підходять його випадку.

3) На основі показників вибраних клієнтом, видається результат опитування у вигляді назви методики.