Лабораторна робота №2

Реалізувати програму з графічним інтерфейсом користувача. Для реалізації інтерфейсу можна використовувати Qt або інший фреймворк (за бажанням студента, погодивши з викладачем).

Далі наведені можливі програми для реалізації, для кожної з яких є основні можливості (які треба обов’язково реалізувати) та список додаткових можливостей (з них треба обрати декілька для реалізації, хоча за бажання можна реалізувати й усі, отримавши за це більше балів). Варіантом цієї лабораторної роботи є номер програми та номери додаткових можливостей, які будуть реалізовані.

***Програма 1 – розумний таймер***

*Базові можливості:*

* Можливість запуску таймерів, звуковий та візуальний сигнал по завершенню часу.
* Таймери на певний проміжок часу (власне таймери) або до заданого моменту часу (будильники) – бажано реалізувати як один список, можливо з додатковою фільтрацією
* Можливість одночасного запуску декількох (довільної кількості) таймерів,
* Зручний інтерфейс для перегляду списку таймерів, керування таймерами

*Додаткові можливості:*

1. Фільтри для вибору таймерів за різними параметрами.
2. Таймери різних типів – дозволяють налаштовувати сигнал, значення часу, інші параметри.
3. Різні формати задання часу таймерів – час від даного моменту, час спрацювання, різні формати часу, інші можливості
4. Підказки під час запуску таймерів – можливість задати певні залежності між таймерами, наприклад, після спрацювання таймера типу 1 рекомендовано запустити таймер типу 2
5. Завчасно налаштовані групи таймерів з можливість швидкого запуску всіх таймерів групи
6. Можливість режиму “Do not disturb” – певного часу, коли сигнали таймерів не спрацьовують.
7. Можливість запам’ятовувати запущені таймери і продовжувати роботу після перезапуску застосунку чи ОС.

***Програма 2 –*** ***система для нотаток***

*Базові можливості:*

* Можливість швидкого створення текстових нотаток (ідей, конспектів якихось ресурсів, завдань/TODO, …)
* Можливість перегляду створених нотаток в хронологічному порядку (за часом створення)
* Можливість переносити менш потрібні/менш актуальні/виконані нотатки до архіву

*Додаткові можливості:*

1. Підтримка груп чи контекстів для нотаток – користувач задає, в якому контексті робляться нотатки (наприклад, робочі, персональні, …)
2. Підтримка ієрархічних контекстів (робочі – проект1 – компонент А – …)
3. Можливість зручного перемикання контекстів – список нещодавніх контекстів, пов’язані контексти
4. Можливість задавати ключові слова, які автоматично перемикають на відповідний контекст
5. Global hotkey – можливість додати нотатки натисканням спеціальної комбінації клавіш, з можливістю вставити текст з буфера обміну.
6. Не лише текстові нотатки – зображення, посилання, аудіо, відео, …
7. Можливість експорту певного набору нотаток в один документ.

***Програма 3 – offline file manager***

*Базові можливості:*

* Можливість зберігання інформації про файли та каталоги на зовнішніх носіях (флеш-диски, зовнішні диски, CD/DVD)
* Можливість додавати метадані до файлів/каталогів
* Якщо носій підключено – можливість відкрити системний переглядач файлів для заданого файлу чи каталогу.

*Додаткові можливості:*

1. Можливість створення віртуальних каталогів, які об’єднують дані з різних джерел
2. Пошук за метаданими (ім’я файлу/каталогу, тип, час створення/редагування, хеш/контрольна сума, користувацькі метадані)
3. Інформація про кількість точних копій файлу чи каталогу на відомих носіях
4. Генерація і запуск команд для копіювання чи синхронізації даних між різними носіями (наприклад, з використанням rsync, robocopy чи інших подібних засобів)

***Програма 4 – демонстрація роботи алгоритмів чи структур даних***

*Базові можливості:*

* Графічна ілюстрація певної структури даних (де зберігаються дані, куди показують вказівники, …)
* Покрокова ілюстрація виконання алгоритму – як змінюється стан структури даних та додаткових змінних на кожному кроці алгоритму
* Можливість рухатись як вперед, так і назад по крокам алгоритму
* Список підтримуваних алгоритмів/структур даних, можливість вибору зі списку для перегляду демонстрації.

*Додаткові можливості:*

1. Можливість користувачу задавати параметри структури даних – наприклад, кількість елементів, значення елементів
2. Можливість збереження та відновлення проміжного стану демонстрації.
3. Можливість вибору якихось елементів демонстрації для відслідковування – вони виділяються, наприклад кольором чи шрифтом, відображається інформація про їх зміни.
4. Можливість підключення нових демонстрацій без перекомпіляції основної програми.

***Програма 5 – демонстрація випадкових подій***

*Базові можливості:*

* Можливість задати кілька випадкових подій – задаються можливі результати та ймовірності кожного результату.
* Запуск моделювання подій заданої кількості разів
* Список результатів моделювання – які події стались на кожному кроці.

*Додаткові можливості:*

1. Підтримка інших видів випадкових подій – вибір елементів зі списку без повернення («колода карт»), випадкові блукання по прямій/площині/графу, випадкові точки в множинах на площині, … (як джерело інших ідей щодо можливих типів подій, можна подивитись <https://www.random.org/> )
2. Збір певної статистики – обчислення середніх значень, кількості подій з певної категорії, …
3. Можливість задати та зберегти seed value генератора випадкових чисел, щоб відтворити ту саму ситуацію.
4. Можливість підключення нових демонстрацій без перекомпіляції основної програми.

***Програма 6 – перевірка завдань з програмування***

*Базові можливості:*

* Можливість задавати кілька джерел коду – репозиторії, онлайн файлові сховища, локальна файлова система.
* Прив’язка коду до студентів.
* Можливість додавати коментарі, прив’язані до конкретного коду
* Генерація звітів з підсумками коментарів

*Додаткові можливості:*

1. Копіювання коду для запуску у спеціально налаштовані середовища
2. Перевірка виконання тестів
3. Перевірка на заданих вхідних даних
4. Запуск статичних аналізаторів коду і збір інформації
5. Виміри продуктивності коду – час виконання, використання пам’яті, …
6. Збір статистики з репозиторію – кількість комітів, кількість гілок, активність за певний період часу
7. Підтримка версіонування для коду, що зберігається не в репозиторіях source control
8. Можливість пошуку спеціальних зразків неправильного коду (налаштовуються користувачем)
9. Можливість створення скриптів для опису додаткових перевірок
10. Можливість створення плагінів для розширення функціональності