

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики
Кафедра прикладної математики
ПРЕДПРОЕКТНЕ ДОСЛІДЖЕННЯ
з дисципліни «Бази даних та інформаційні системи»
на тему
Програмування

Виконав
Студент групи км-42
Кобилянський Р. В.
Керівник

Київ - 2017

Зміст

| | | |
|-----|-------------------------------|---|
| 1 | ПРЕДПРОЕКТНЕ ДОСЛІДЖЕННЯ..... | 3 |
| 1.1 | Мета | 3 |
| 1.2 | Опис програми | 3 |
| 1.3 | Граничні умови | 3 |
| 2 | SRCUM ПЛАНУВАННЯ..... | 5 |

1 ПРЕДПРОЕКТНЕ ДОСЛІДЖЕННЯ

1.1 Мета

Полегшення та прискорення виконання базових операцій при написанні коду на мовах програмування.

1.2 Опис програми

Програма, яка полегшує написання програм, яку звичайно називають інтегрованою системою розробки (Integrated Development Environment) чи IDE. Процес написання програм полягає в написанні операторів, послідовне виконання яких повинно приводити до конкретного результату. З цією задачею може допомогти програмний продукт, який являю собою текстовий редактор з функціями які полегшують написання коду та керування файловою системою.

1.3 Граничні умови

IDE буде підтримувати тільки обмежену кількість мов програмування.

IDE буде поставлятися в вигляді portable версії чи в вигляді інсталятора.

Користувачу IDE будуть доступні наступні дії:

а) обрання мови програмування:

- 1) джерело інформації – наявні файли налаштування для мов, наприклад для Java, C++, C;
- 2) структура для збереження – XML, property, YAML;
- 3) результатом є закріплення данної мови за конкретним проектом.

б) виконання операцій CRUD при роботі з файлами:

- 1) джерело інформації - наявні файли та папки в папці проекту;
- 2) структура для збереження – файлова система;
- 3) результат є внесення змін в файлову систему.

в) автоматична збірка кодів в файли виконання та їх автоматичне виконання:

- 1) час виконання в залежності від розміру проекту;
- 2) джерело інформації - наявні файли та папки в папці проекту;
- 3) допоміжні засоби – встановлений компілятор в операційній системі;

4) результатом є файл виконня, який можна запустити в IDE.

г) перевірка на помилки часу компіляції під час редагування файлів; та додаткові підказки під час редагування файлів:

- 1) час виконання в залежності від розміру файлу але менше 2 сек;
- 2) джерело інформації – файл з налаштування (той же який і для обирання мови), який описує правила написання коду, ключові та зарезервовані слова, та також сам файл з кодом;
- 3) результатом є підсвічення проблемних місць з поясненням.

2 SRCUM ПЛАНУВАННЯ

Довжина спрінта буде становити 10 робочих днів , чи 2 тижні. В перерахуванні на кількість розробників це 80 годин. Code Freezing з середи 2 тижня скраму.

Перший спрінт буде присвячений на розробку двигуна текстового редактора, який повинен містити функціонал для роботи з файлами. Чи в разі можливості знаходження текстового редактора, який можна доповнити потрібним функціоналом.

Критерії прийняття, відкриття , редагування та зберігання файлів, створення та видалення папок.

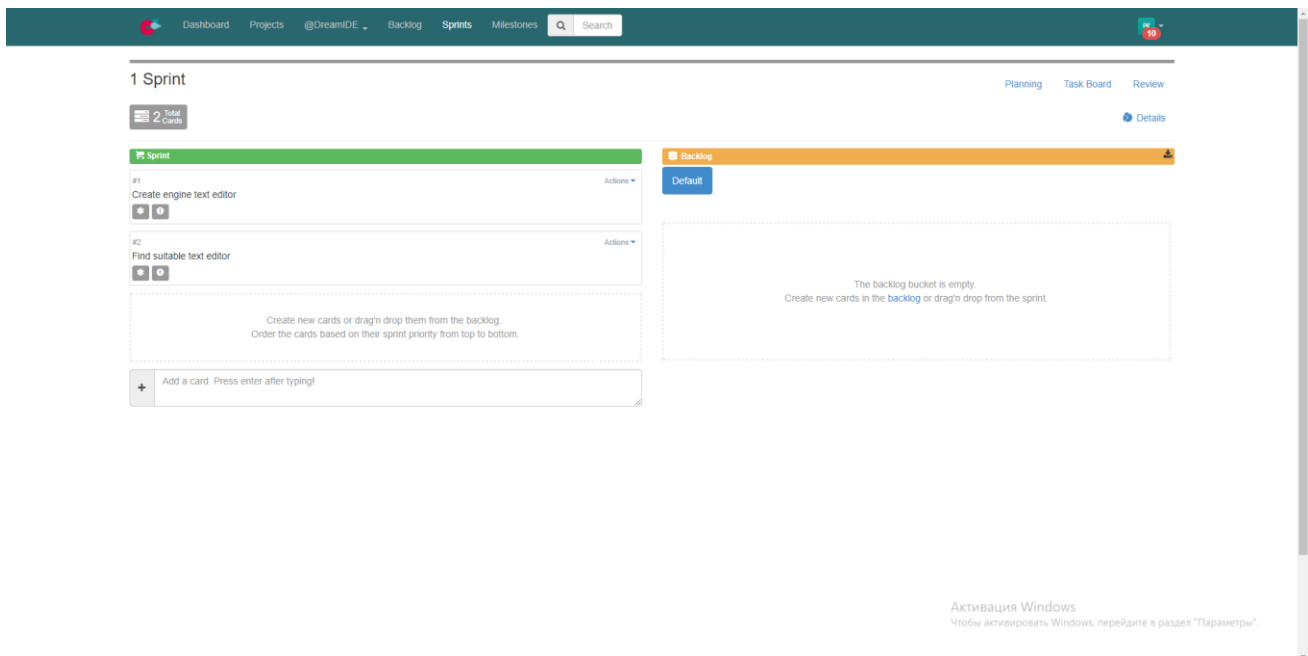


Рисунок 2.1 – Спрінт 1

В другому спрінту повинно бути впровадження закріплення компіляторів за текстовим редактором та компіляція файлів. Також, в якості спайку , розробка формату конфігураційних файлів.

Критерії прийняття : компіляція коду та готове рішення для конфігураційних файлів.

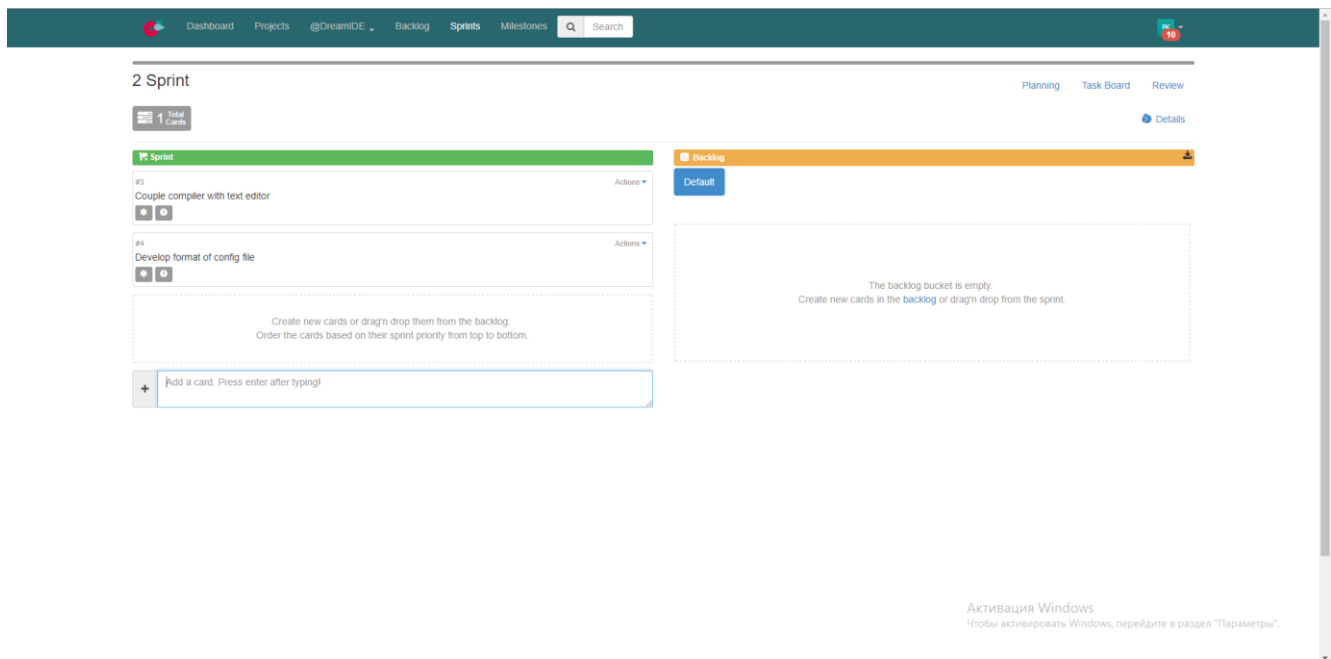


Рисунок 2.2 – Спрінт 2

Третій спрінт повинен складатися з створення дефолтних конфігураційних файлів. Витягування з файлів конфігурацій інформації про валідацію, та про ключові слова. Реалізація впливаючих підказок по ключовим словам.

Критерії прийняття: створено конфігураційні файли, отримання з них правил валідацій, там можливість отримати вказівку по ключовим словам.

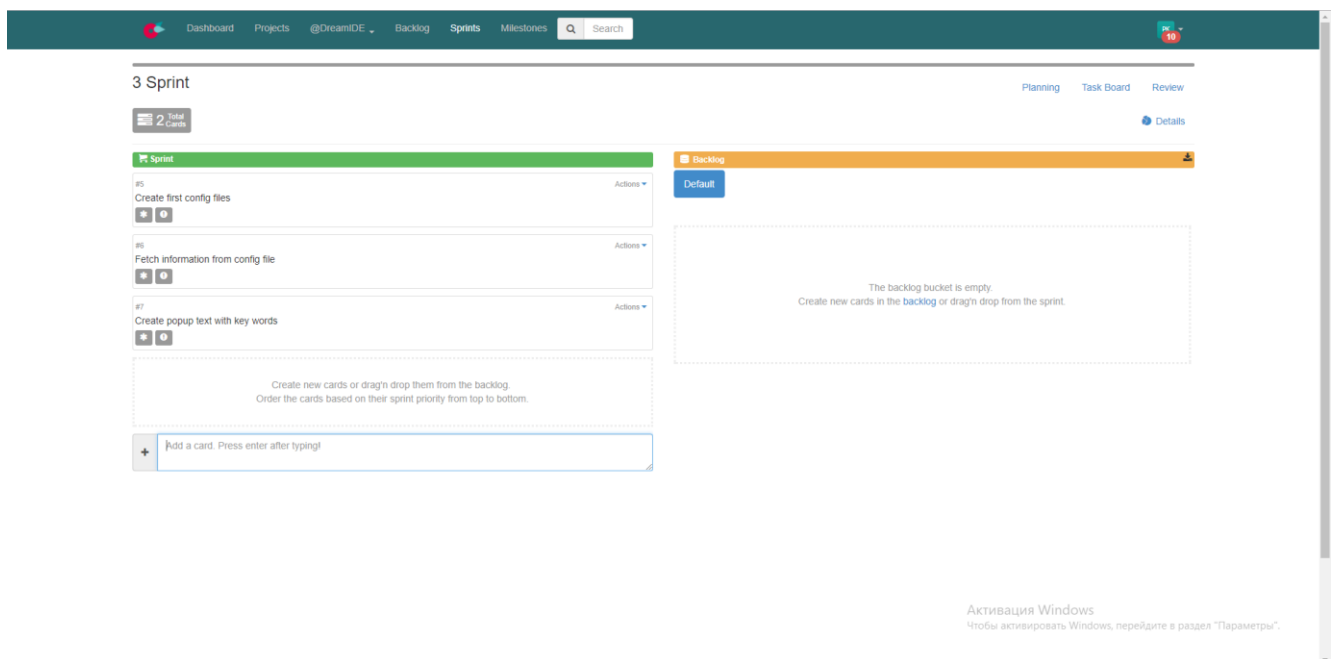


Рисунок 2.3 – Спрінт 3

Четвертий спрінт повинен бути присвячений автоматичній компіляції коду та підсвітки помилок, та запуску програми з IDE.

Критерії прийняття : запуск програми з IDE, підсвітка помилок при невірному вводі.

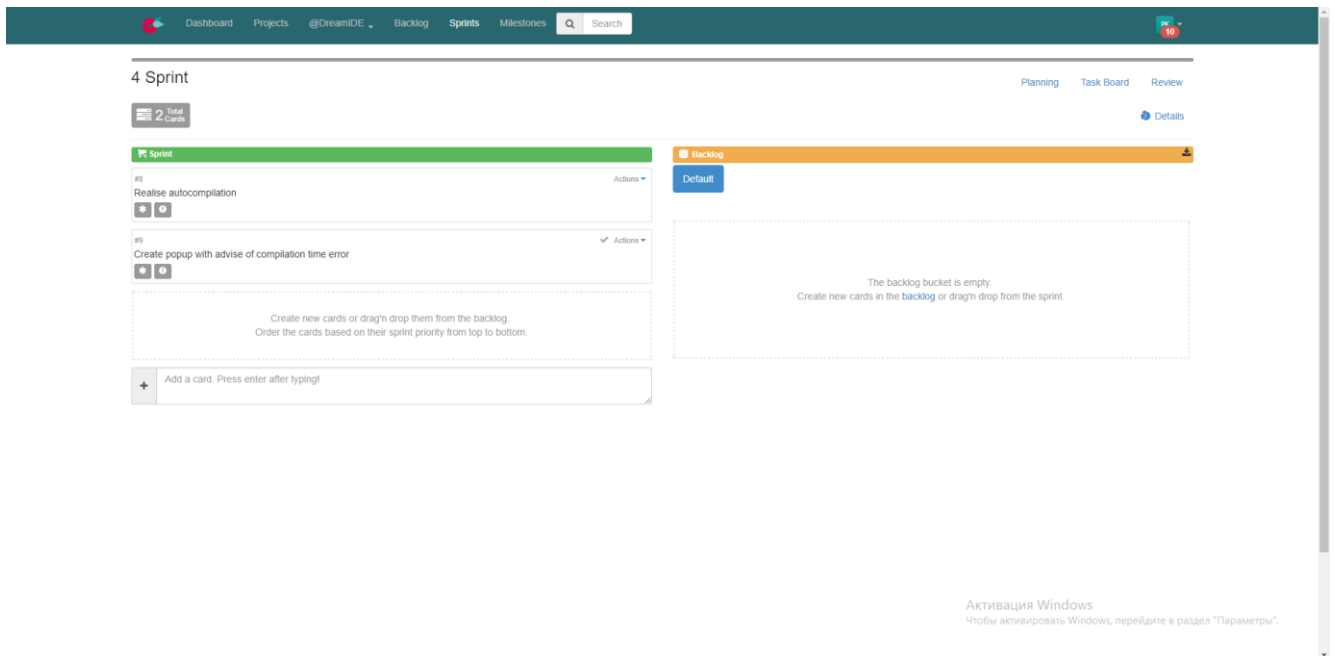


Рисунок 2.4 – Спрінт 4