

Версіонування та збереження проектів САПР у хмарних сервісах

Калініченко Євген

Керівник: к.т.н. Смаковський Д.С.

Об'єкт та предмет дослідження

- Об'єкт дослідження - комп'ютерні технології організації збереження та версіонування даних в хмарних сервісах
- Предмет дослідження - комп'ютерні технології організації збереження та версіонування даних в хмарних сервісах для формату проектів САПР

Проблеми при роботі з САПР проектами

- Складність одночасної роботи з одним проектом декількох проектувальників
- Перенесення файлів для роботи з однієї локальної машини на іншу займає багато часу
- Важко контролювати та слідкувати за зробленими змінами

Існуючі програмні засоби

- Система **Autodesk Vault**
- Система керування версіями **Git**
- Використання готових програмних рішень реалізації хмарного сховища, наприклад **Яндекс.Диск**

Система Autodesk Vault

- Переваги
 - Архівація та відновлення будь-яких версій
 - Підтримка захисту файлів та користувачів
- Недоліки
 - Прив'язаність до продукту Autocad, немає можливості використання зі стороннім САПР
 - Складність налаштування

Система керування версіями Git

- Переваги
 - Потужний інструмент контролю версій
- Недоліки
 - Неможливість порівняння файлів САПР проекту, тобто немає можливості перегляду змін

Використання готових програмних рішень реалізації хмарного сховища

- Переваги
 - Простота роботи для користувача, рішення працює з коробки
- Недоліки
 - Відсутня можливість перегляду змін в SVG файлах
 - Обмежена одночасна робота декількох користувачів
 - Автоматична синхронізація, що призводить до розмноження різних версій файлів з мінімальними змінами
 - Обмежена підтримка версіонування

Існуючі програмні засоби

- Всі існуючі програмні засоби мають свої переваги та недоліки
- Жодна система не підтримує простого збереження файлів до хмарного сховища
- Не має зручної можливості перегляду зроблених змін

Хмарні сервіси

- Набирають свою популярність з 2008 року
- Висока надійність збереження даних
- Розподіленість даних та швидкий доступ

Мета дослідження

- Виявлення підходу до організації версіонування проекту САПР SVG формату двомірного креслення та вибір оптимального сервісу для організації збереження проекту до хмари

Постановка задачі

- Проаналізувати методи пошуку змін між ієрархічними деревами з метою виявлення можливостей адаптації алгоритмів для пошуку змін між ієрархічними елементами SVG файлу
- Проаналізувати існуючі технології побудови хмарних сервісів та їх придатність для організації сховища САПР проектів
- Розробити алгоритм пошуку змін в дереві елементів SVG файлу
- Здійснити програмну реалізацію системи збереження та версіонування даних використовуючи обраний сервіс та реалізацію алгоритму пошуку змін

SVG формат

- SVG - уніфіковане майбутнє для розробки 2D CAD систем
- Повна стандартизація W3C, відкриті специфікації
- Елементи креслення описуються мовою XML
- Підтримка формату всіма сучасними САПР
- Векторне представлення даних

Amazon S3

- Наявність безкоштовного тарифного плану
- Зручне та відкрите SDK
- Вбудована підтримка версіонування, з необмеженою кількістю версій файлу
- Можливість територіального вибору основної локації розміщення сховища
- Популярність та технічна підтримка



Пошук змін в SVG файлі

- Для зручного порівняння змін зроблених в проекті було розроблено модуль для порівняння та пошуку змін в SVG файлах
- SVG є XML подібним документом
- Зміст SVG файлу можна представити у вигляді дерева елементів
- Таким чином для реалізації алгоритму порівняння використано підхід порівняння дерев елементів двох SVG файлів

Діаграма діяльності алгоритму пошуку змін



Математичний опис алгоритму пошуку змін

Нехай X та Y - зв'язані ациклічні графи (дерева), що описують елементи порівнюваних файлів

1. Знаходження бієкції двох дерев, тобто:

$B = f: X \leftrightarrow Y$, тобто

1. Має місце ін'єктивність:

$$\forall x_1 \in X, \forall x_2 \in X, x_1 \neq x_2 \Rightarrow f(x_1) \neq f(x_2)$$

2. Та сюр'єктивність:

$$\forall y \in Y, \exists x \in X, f(x) = y$$

2. Розрахунок $Cost$ - оцінка вузлів X та Y які не входять в множину B , тобто:

$x \in X$, де $x \notin B$ та $y \in Y$, де $y \notin B$

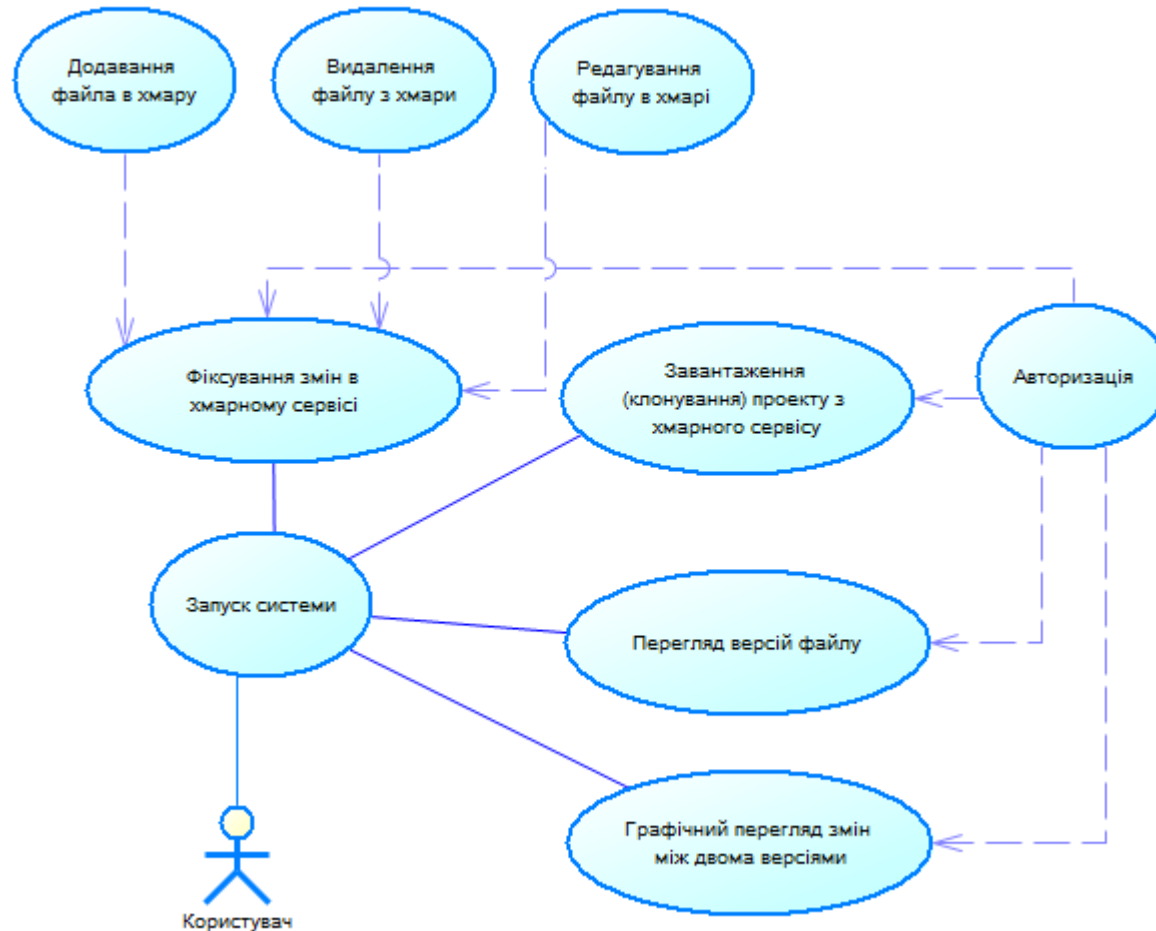
Оцінка базується на кількості однакових атрибутів та їх значень для елементів $x \in X$, де $x \notin B$ та $y \in Y$, де $y \notin B$,

де x та y – вузли з однаковими ім'ям

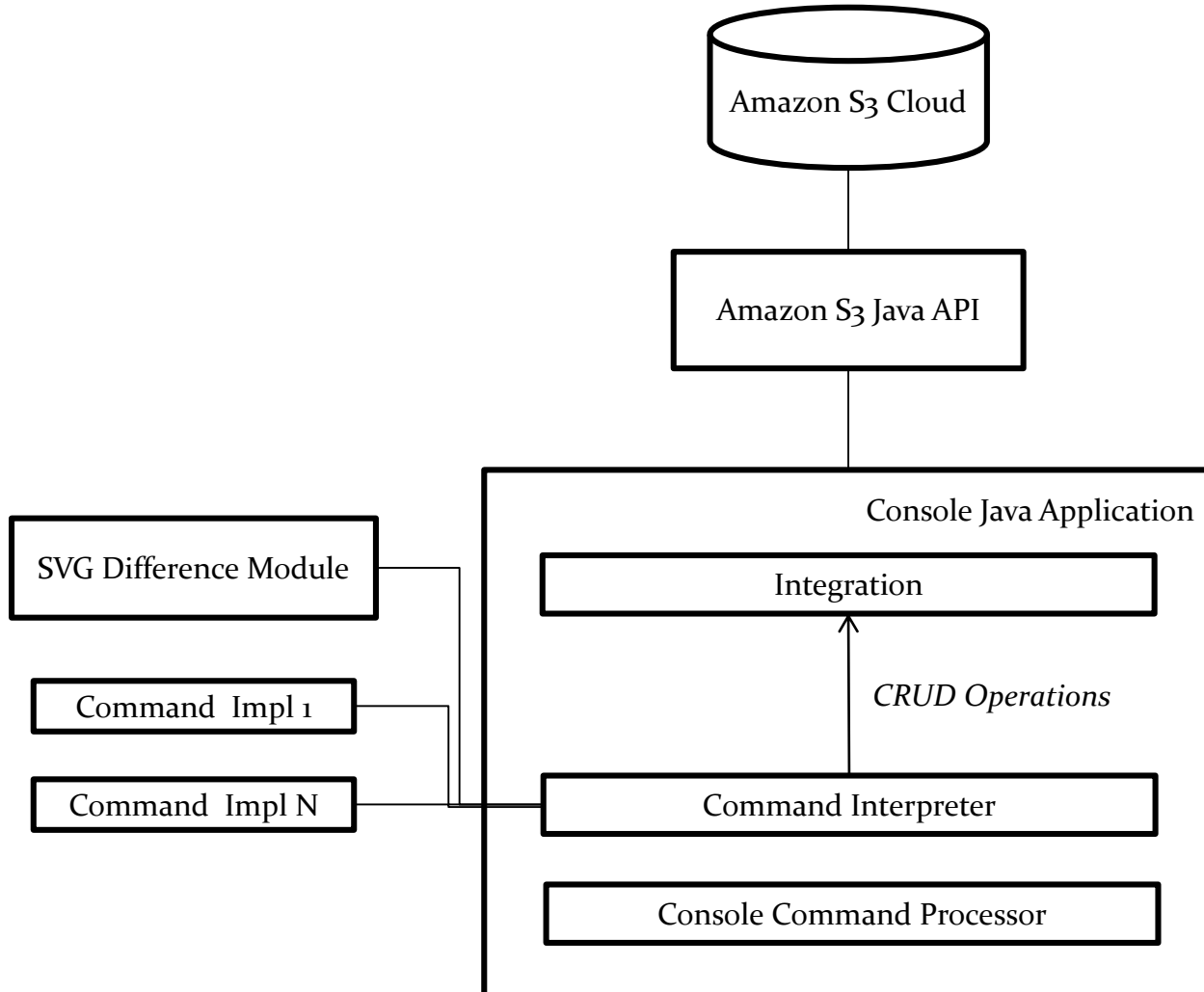
3. Вузли з $Cost > 0$ помічаються як змінені з відповідною кількістю елементів.

4. Вузли з $Cost = 0$ є видалені або додані, в залежності від дерева X та Y для якого не було знайдено відповідний елемент

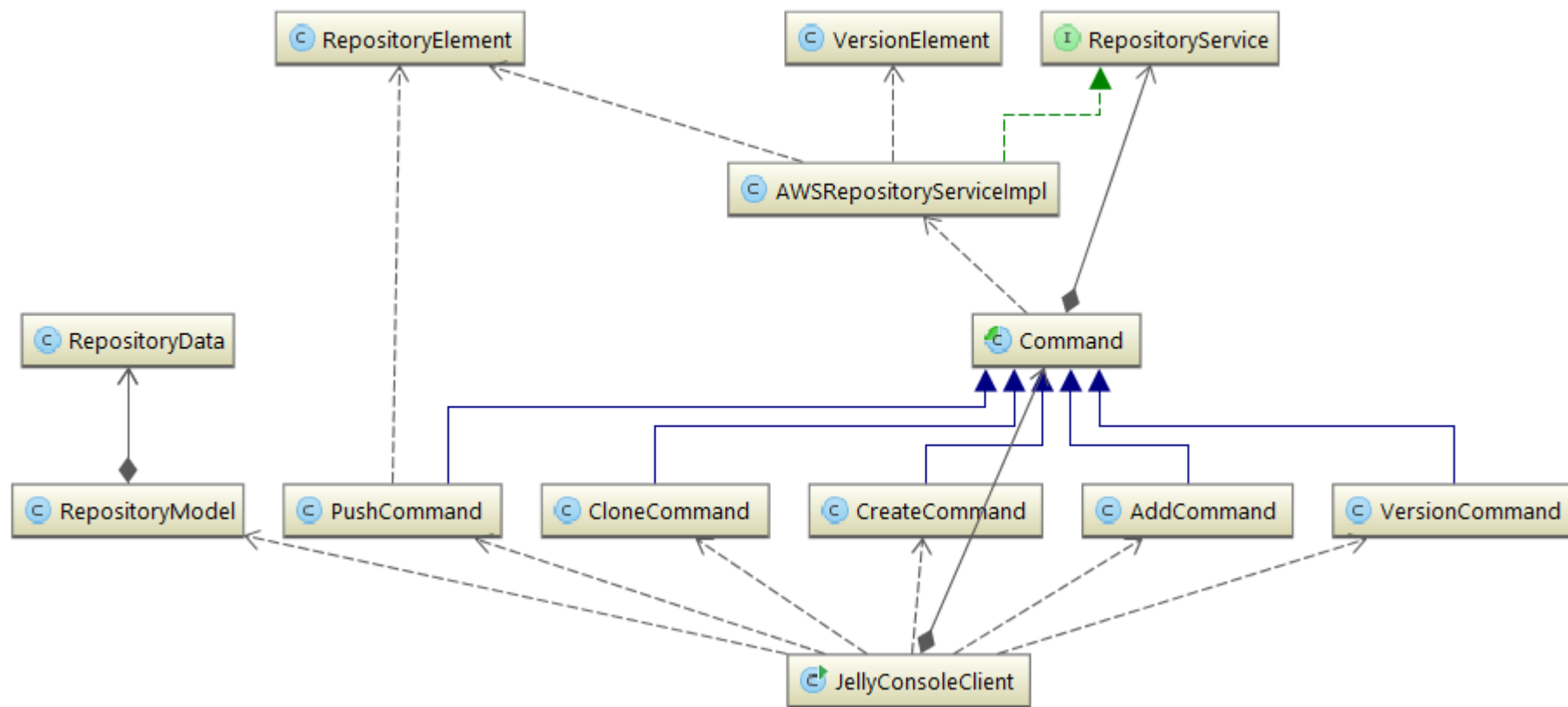
Use Case діаграма



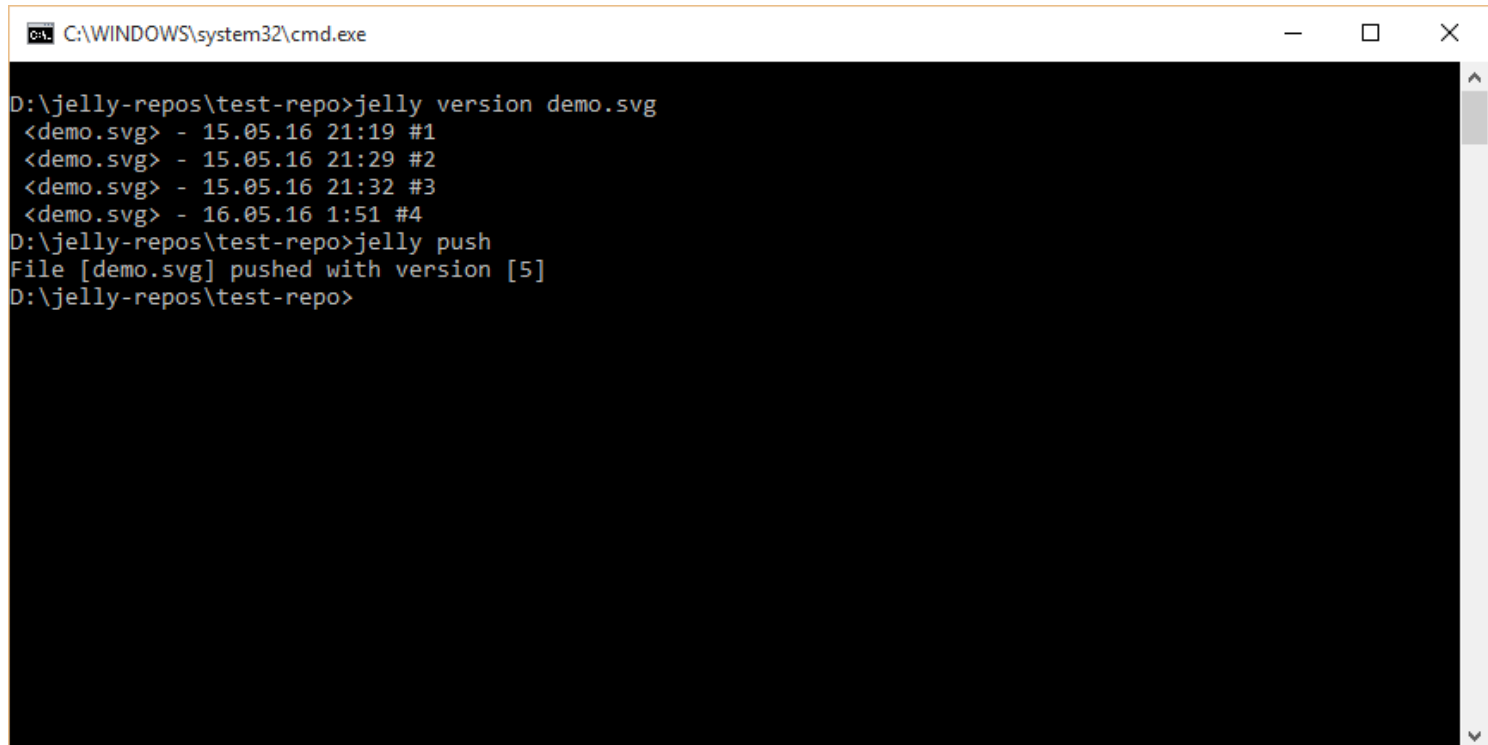
Архітектура розробленої системи



Діаграма класів



Интерфейс програми



```
C:\WINDOWS\system32\cmd.exe

D:\jelly-repos\test-repo>jelly version demo.svg
<demo.svg> - 15.05.16 21:19 #1
<demo.svg> - 15.05.16 21:29 #2
<demo.svg> - 15.05.16 21:32 #3
<demo.svg> - 16.05.16 1:51 #4
D:\jelly-repos\test-repo>jelly push
File [demo.svg] pushed with version [5]
D:\jelly-repos\test-repo>
```

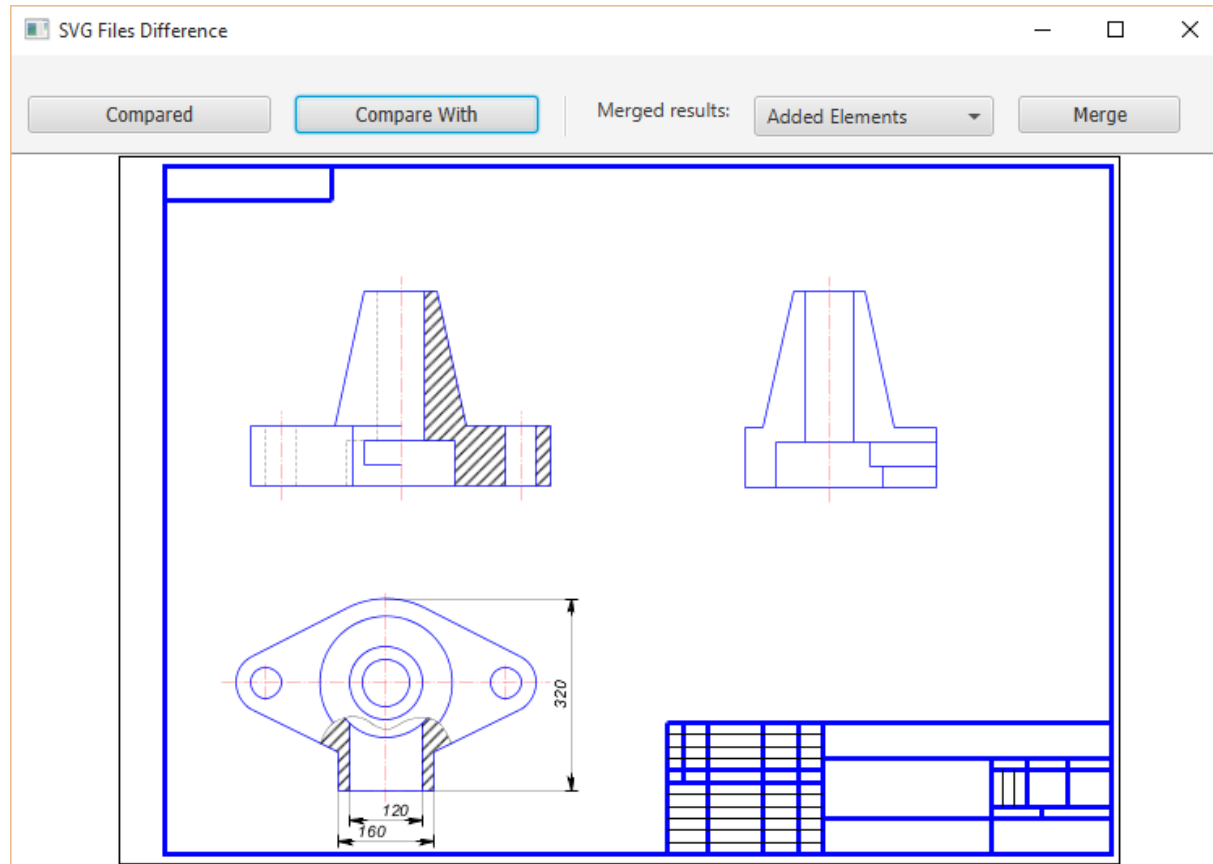
Наявні команди

- *create* – створення репозиторію в хмарі
- *add* – додавання файлу під контроль системи для майбутньої синхронізації
- *push* – фіксування змін в сховищі
- *clone* – клонування репозиторію з хмари на локальний комп'ютер
- *version* – отримання списку версій
- *diff* – відображення різниці між двома версіями файлу

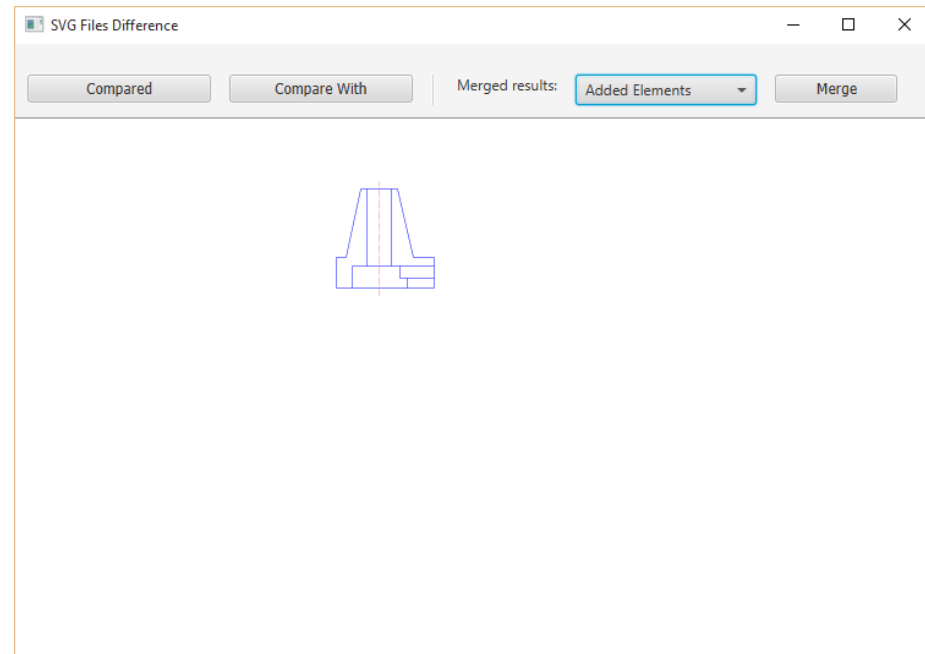
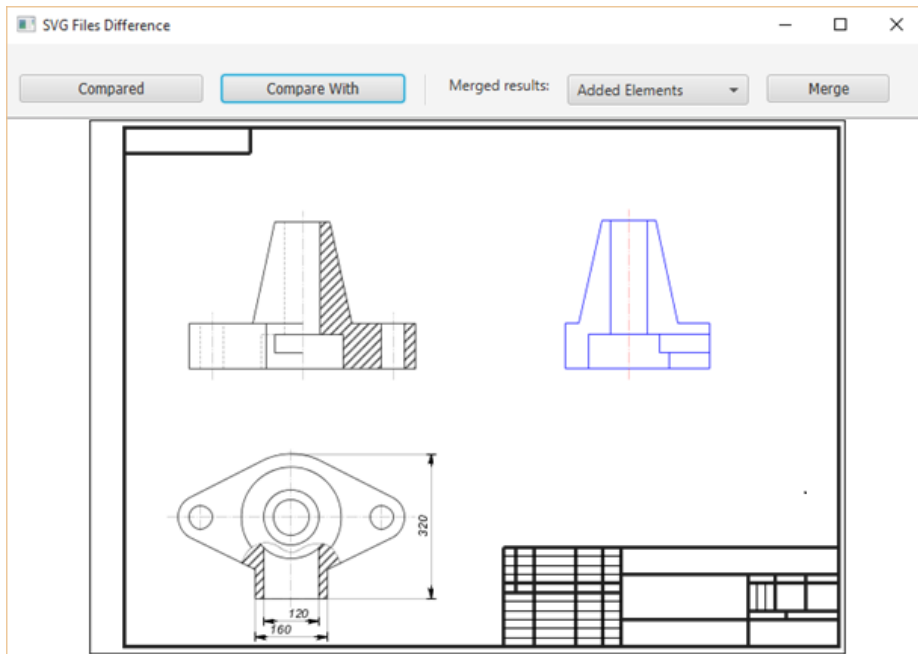
Специфіка реалізації програми

- Реалізовано мапінг хеш коду версії файлу на порядковий номер версії, для зручності використання команди перегляду змін між версіями
- Пошук змін реалізовано у вигляді JavaFX модулю та може бути використано не лише для пошуку змін між певними версіями, а і для порівняння будь-яких SVG файлів

Перегляд змін між версіями



Перегляд змін між версіями



Висновки

- Розроблено систему збереження та версіонування файлів САПР проекту SVG формату до хмарного сервісу Amazon S3.
- Розроблено метод пошуку змін в SVG файлі на основі алгоритму знаходження бієкції графів.
- Система значно спрощує роботу одного або декількох розробників над проектом та покращує продуктивність

Наукова новизна магістерської роботи

- Було запропоновано підхід до порівняння двох SVG файлів та пошуку зроблених змін на основі порівняння дерев елементів, що дозволило знаходити та виділяти змінені елементи SVG файлу
- Набуло подальшого розвитку застосування бієкційних методів пошуку змін в XML-подібних форматах.

Дякую за увагу!