

Основи ООП.
Лабораторна 1.
Звіт.

Виконав студент ІПС-21
Міцкевич Костянтин Олександрович

Завдання 1a

Design/Implementation Modeling

1. У якості існуючого проєкту було обрано роботу CV-Creator створений задля ознайомлення з React+node.js+express:
<https://github.com/Kirolen/CV-Creator>
Останній коміт перед рефакторингом: 2597f26
2. Було реалізовано тести для серверної частина за допомогою бібліотеки JEST.
3. Побудовано UML діаграми: **Use Case, Class, Activity**. Інші з існуючих типів діаграм було вирішено не реалізовувати через недоцільність. За умовою завдання зроблено версію діаграм після рефакторингу коду. Також реалізовано **глоссарій** проєкту у вигляді однойменного розділу у README.md файлі проєкту.
4. Було запропоновано наступні зміни:
 - Додати можливість редагувати інформацію одразу в багатьох CV одночасно, тобто: 1) добавляти конкретні поля з інформацією; 2) видаляти поля з конкретним значенням 3) змінювати значення конкретних полів: Ім'я, телефон і т.п
 - Видалити непотрібне отримання даних /user-data.
 - виправити баги, зокрема появу CV з інших акаунтів після зміни акаунту без перезапуску сторінки.
 - Оптимізувати функцію відкриття перегляду всіх CV користувача, щоб уникнути зайвого декодування даних з токена, що значно покращило швидкодію сайту.
 - Замість передавання масиву через токен, створити окрему функцію, яка повертатиме масив CV.
5. Усі перелічені зміни реалізовано.
6. Нова версія програми повністю зберігає попередню функціональність і коректну поведінку, у цьому можна впевнитися за допомогою реалізованих раніше unit tests.
7. Хоча benchmark не було реалізовано, але після внесення змін відгук сайт на дії користувача значно пришвидшився. Тепер сайт реагує ментально, на відміну від реалізації раніше, коли деякі дії займали близько 2 секунд.
8. На мою думку, у даній програмі строго дотримано принципів ООП та принципу DRY. Використання патернів, про які варто згадати, не було реалізовано.

* Було реалізовано створення документації коду з використанням JSDOC.
Згенеровану документацію викладено в репозиторій
<https://github.com/Kirolen/CV-Creator>

Завдання 1d

У процесі виконання лабораторної роботи №1 було використано інструмент UML-моделювання draw.io

1. Чому було обрано саме цей інструмент, а не аналоги?

Draw.io було обрано завдяки його простоті у використанні та безкоштовності.

2. Наскільки просто та зрозуміло було отримати, встановити, налаштувати та почати використовувати цей інструмент?

Отримати доступ до draw.io було дуже просто. Хоча існує десктопна версія, для виконання завдань лабораторної роботи було достатньо веб-версії, яка не потребує встановлення.

3. Наскільки зрозумілою та корисною була документація інструменту?

Документація виявилася зрозумілою та корисною. Якщо мати базові знання UML та принципів ООП, проблем із її сприйняттям не виникає. Вона допомагала у виборі відповідних елементів для створення діаграм.

4. Наскільки було зрозуміло, як саме використовувати інструмент, які функції/засоби/вікна/елементи керування використовувати для вирішення поставлених задач?

Інтерфейс інтуїтивно зрозумілий, тож освоєння не викликало труднощів. Основні елементи UML-діаграм можна легко знайти на бічній панелі.

5. Чи всі 14 типів діаграм з UML 2.5 підтримує інструмент? Якщо ні – вкажіть, які саме не підтримуються

Інструмент дозволяє створювати всі типи UML-діаграм, хоча шаблони є

лише для деяких з них, зокрема Activity, Sequence, State та Class діаграм.

6. Чи всі можливості, доступні на кожному типі діаграм, підтримує інструмент? Якщо ні – вкажіть, що саме не підтримується, та для якого типу діаграм.

У межах виконаних завдань усі необхідні функції працювали без обмежень.

7. Чи використовували якісь додаткові можливості інструменту, наприклад генерацію коду з діаграм чи відновлення діаграм з коду? Наскільки гарно та правильно працюють ці можливості?

Додаткові можливості, такі як генерація коду чи відновлення діаграм з коду, не застосовувалися, оскільки в цьому не було потреби.

8. Наскільки зручно було використовувати інструмент, чи не треба було виконувати багато надлишкових дій?

Загалом робота з інструментом була комфортною, однак розміщення стрілок на діаграмі іноді потребувало додаткового часу для точного налаштування.

9. Наскільки зрозумілою була поведінка інструменту в різних ситуаціях? Чи не виникали ситуації, коли незрозуміло, чому були виконані якісь дії чи як досягли певного стану?

Більшість функцій працювали логічно, хоча одного разу виникла незрозуміла ситуація з форматуванням тексту після вставки з буфера обміну.

10. Чи виникали якісь проблеми з використанням інструменту? Чи вдалось їх вирішити, як саме?

Суттєвих проблем, які могли б вплинути на кінцевий результат, не

ВИНИКЛО.

- 11.Що хорошого можна сказати про цей інструмент, які були позитивні аспекти використання інструменту?

Серед переваг: наявність зручних шаблонів для діаграм, інтуїтивний інтерфейс, широкий вибір форматів експорту, а також можливість роботи без встановлення через веб-версію.

- 12.Що поганого можна сказати про цей інструмент, які були негативні аспекти використання інструменту?

Єдиним суттєвим недоліком було певне ускладнення при налаштуванні стрілок на діаграмах.

- 13.Якби довелось вирішувати аналогічну задачу, але вже враховуючи досвід використання в цій лабораторній роботі, що варто було б робити так само, а що змінити? Можливо, використати інший інструмент, чи використати інші можливості цього інструменту, чи інакше організувати процес розробки діаграм, чи ще щось?

У майбутньому варто було б використати більше можливостей інструменту та спробувати працювати з менш поширеними типами UML-діаграм. В цілому, вибір draw.io залишається оптимальним.