Посилання на github: <https://github.com/pmluxoftlab/ProjectPractice.git>

Дисциплина «Проектный практикум»

Практическое задание №3.

Проектирование программного продукта. Часть 2

1. Алгоритмічна складність програми:

Нижче представлена схема алгоритму для функції Convert to Graph(Рис 3.3):



Рисунок 3.3 – схема алгоритму Convert to Graph

При оцінці алгоритмічної складності треба враховувати середній час доступу до даних у пам’яті, наявність циклів під час виконання операцій та час збереження даних. Нижче зображено схему алгоритму для функції Convert to XML(рисунок 3.4):



Рисунок 3.4 – схема алгоритму для функції Convert to XML

У наведених схемах алгоритмів більшість елементарних блоків виконують операції зчитування/запису даних. Воно відбувається за середній час O(n\*log(n)). Також на схемах наявні цикли, складність яких O(n)(середній випадой). У найгіршому разі складність програмної системи: O(n\*n). Середній випадок: O(n\*log(n)).

1. Опис інтерфейсу користувача:

У початку роботи програми користувач бачить пусте вікно зі стрічкою меню. Нова діаграма створюється наступним чином: користувач натискає пункт меню File -> New -> Static View(діаграма класів або діаграма кооперацій/коммунікацій) або Dynamic View( діаграма використання або діаграма послідовностей). Далі користувач додає блоки діаграм на графічну сцену та поєднує їх лініями шляхом перетягування блоків на графічну сцену(метод drag`n`drop). Збереження діаграми відбувається таким чином: користувач натискає пункт меню File -> Save та задає ім’я файлу. Для конвертації у файл формату XML/граф необхідно вибрати пункт меню File -> Save as та вибрати ім’я та формат файлу( XML або граф).для відкриття вже створеної діаграми необхідно вибрати пункт меню File -> Open та вибрати файл с діаграмою.

Дана програма дозволяє створювати наступні види діаграм:

Діаграми варіантів використання(рисунок 3.5):

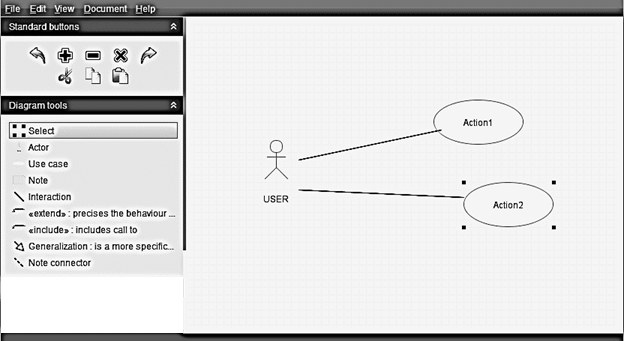


Рисунок 3.5 – Діаграма варіантів використання

Діаграми класів(рисунок 3.6):

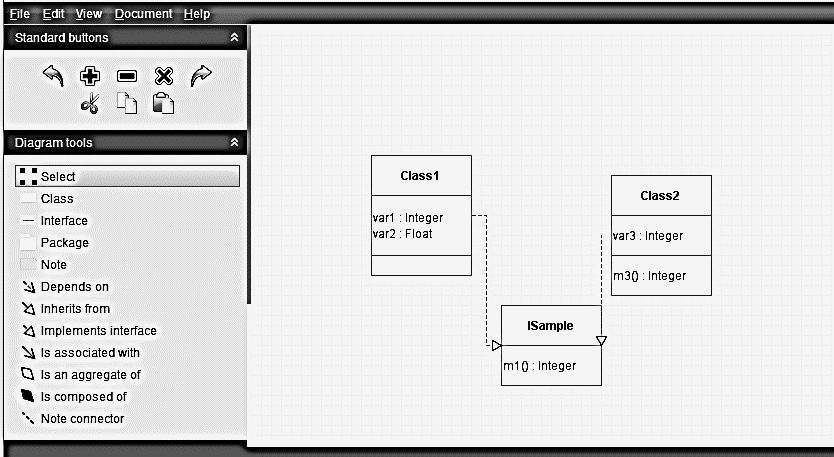


Рисунок 3.6 – Діаграма класів

Діаграми послідовностей(рисунок 3.7):

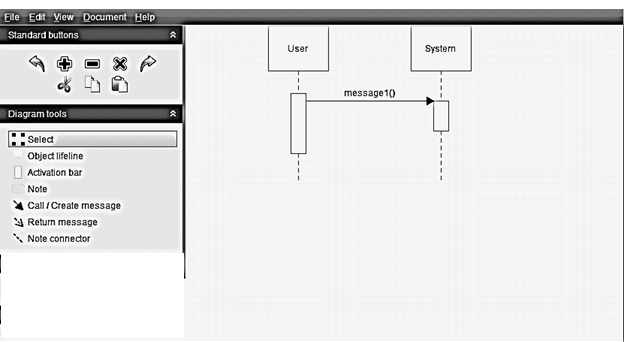


Рисунок 3.7 – Діаграма послідовностей

Діаграма кооперацій(рисунок 3.8):

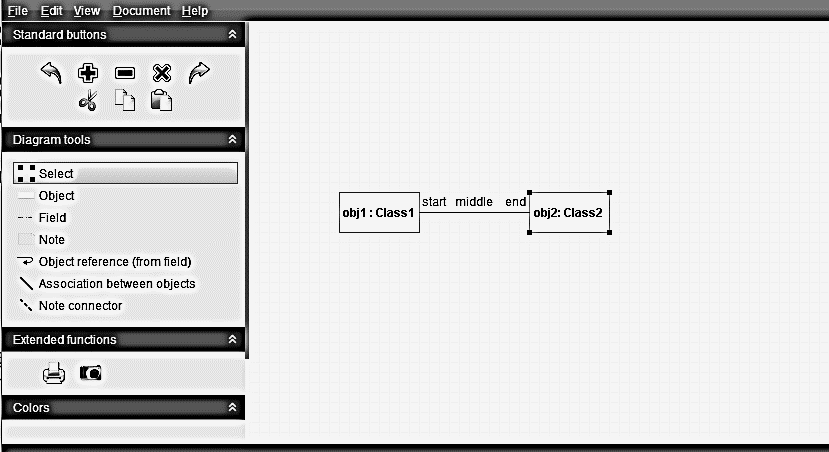


Рисунок 3.8 – Діаграма кооперацій