ОТЧЕТ ПО ПЕРВОЙ ПОЛОВИНЕ УЧЕБНОЙ ПРАКТИКИ

24.06.24

1. Ознакомительная часть

Консультация по учебной практике. Ознакомление с программой учебной практики, заполнение дневников.

Целью учебной вычислительной (ознакомительной) практики является закрепление полученных знаний за первый год обучения посредством выполнения специальных учебных заданий, участия в работе над командным проектом.

Задачи практики:

1.Знакомство с IT кампанией, возможностями обучения и повышения квалификации, знакомство с разными направлениями профессиональной подготовки IT специалистов.

2.Изучение инструментов для совместной разработки проектов и контроля версий.

3.Подготовка и настройка программных сред и средств тестирования для прохождения тренинга по выбранному направлению.

4.Выполнение стандартных заданий.

5.Выполнение заданий повышенного уровня.

6.Разработка приложения.

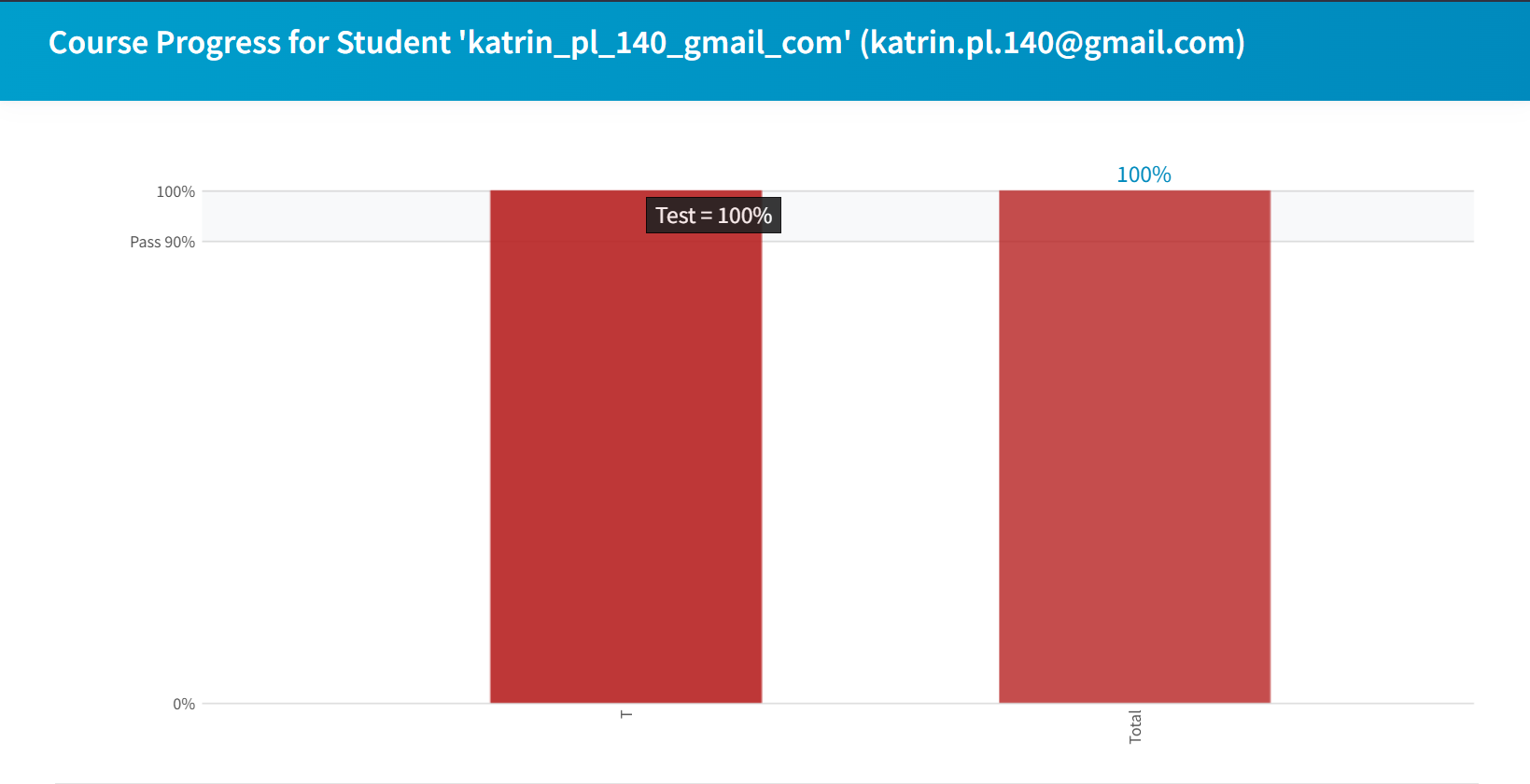
7.Оформление отчета.

Конференция с кампанией Soft Club.

В рамках встречи с SoftClub мы обсудили различные аспекты разработки программного обеспечения. Были рассмотрены вопросы выбора языка программирования, архитектуры приложения, методов тестирования и отладки. Мы поделились опытом работы над совместными проектами и обсудили проблемы, возникающие при разработке.

1. Система контроля версий Git

Прошла курс по Git на платформе [Home | Learn (epam.com)](https://learn.epam.com/start)



Также зарегистрировалась на GitHub [KatePlo (github.com)](https://github.com/KatePlo) и создала свой репозиторий [KatePlo/git-demo (github.com)](https://github.com/KatePlo/git-demo)

25.06.24

3. Модульное тестирование (Unit Testing)

В ходе разработки программного обеспечения было применено модульное тестирование (unit testing). Тесты были написаны для отдельных функций и классов, позволяя гарантировать их корректную работу.

Преимущества модульного тестирования:

• Раннее обнаружение ошибок: Тесты запускаются часто, что позволяет выявлять ошибки на ранних этапах разработки.

• Упрощение рефакторинга: Тесты служат защитой от нежелательных изменений в коде, которые могут привести к ошибкам.

В ходе занятия были рассмотрены примеры и применено модульное тестирование к различным типам программ с использованием Visual Studio и QT Creator.

В том числе к консольным приложениям, приложениям с графическим интерфейсом.

26.06.24

4. Создание GUI приложения в Qt

Основные этапы разработки GUI приложения в Qt:

• Проектирование интерфейса: Создание макета приложения, определение элементов интерфейса и их связей.

• Разработка логики: Написание кода для обработки событий, взаимодействия с пользователем и реализации функциональности приложения.

• Тестирование: Проверка работы интерфейса и всех его компонентов.

Результаты

В результате работы была разработана функциональная программа, включающая в себя GUI с интуитивно понятным интерфейсом, модульные тесты для проверки функциональности кода, а также эффективная система контроля версий.

27.06.24

5. Разработка приложения, отражающего табличные данные с возможностью корректировки ( на основе шаблона Модель-Вид), вывода и ввода данных

Функциональность приложения:

• Вывод данных: Есть возможность загрузить уже существующий документ.

• Ввод данных: Пользователь может изменять данные в таблице и сохранять изменения как в виде текстового документа, так и ввиде Exel таблицы.

• Добавление строк: Пользователь может добавлять новые строки в таблицу.

• Удаление строк: Пользователь может удалять строки из таблицы.

28.06.24

6. Разработка SDI-приложения

SDI-приложения характеризуются наличием одного основного окна, в котором отображается единственный документ.

Функциональность SDI-приложения:

• Создание нового документа

• Открытие существующего документа

• Сохранение документа