**АРХАНГЕЛЬСКИЙ КОЛЛЕДЖ ТЕЛЕКОММУНИКАЦИЙ (ФИЛИАЛ) СПбГУТ**

**(АКТ (ф) СПбГУТ)**

**Отчеты по лабораторным и практическим работам**

**по МДК.02.01**

Студент ИСПП 11 19.12.2024 Е.А.Антипин

Преподаватель Ю.С.Маломан

Архангельск 2024

**Лабораторная работа №1**

**Инспекция программного кода на предмет соответствия стандартам кодирования**

**1 Цель работы**

1.1 Изучить процесс выполнения инспекции программного кода.

**2 Ход работы**

2.1 Проверка кода соответствия критериям качества ПО

| критерий качества ПО | что анализируется | доказательство соответствия | рекомендации по улучшению ПО |
| --- | --- | --- | --- |
| функциональная пригодность | Функция получения финансовых показателей компании | доказательство соответствия показано на рисунке 1. |  |
| уровень производительности | Скорость ответа от сервера при обращении к конечной точке, отвечающей за получении данных о компании | Ответ от сервера приходит за 0,21 секунду | Добавить кеширование данных |
| совместимость | Работа в разных браузерах | Приложение открывается в разных браузерах |  |
| удобство использования | Время на выполнение функции получения финансовых показателей компании компании apple.  Цветовая гамма.  Адаптивноть. | Нажать на кнопку Search Companies, в поле поиска ввести appl, среди результатов поиска выбрать компанию Apple Inc. Время на выполнение этих действий – 3 секунды.  Цвета сочетаются.  Текст читаемый. | Адаптировать интерфейс под мобильные устройства. |
| надёжность | Частота сбоев | Сбои не выявлены |  |
| защищённость | Хранение паролей в хешированном виде | Пароли хранятся в захешированном виде | добавить протокол шифрования https |
| сопровождаемость | соответствие кода codestyle | Написаны комментарии к коду, названия переменных соответствуют стилю именования в языке программирования C#, как показано на рисунке 2. |  |
| переносимость (мобильность) | Анализ возможности переноса на другие платформы | Приложение написано с использованием технологии контейнеризации Docker, что позволяет запустить приложение на устройствах с ОС Linux, Windows, MacOS при наличии программы Docker |  |

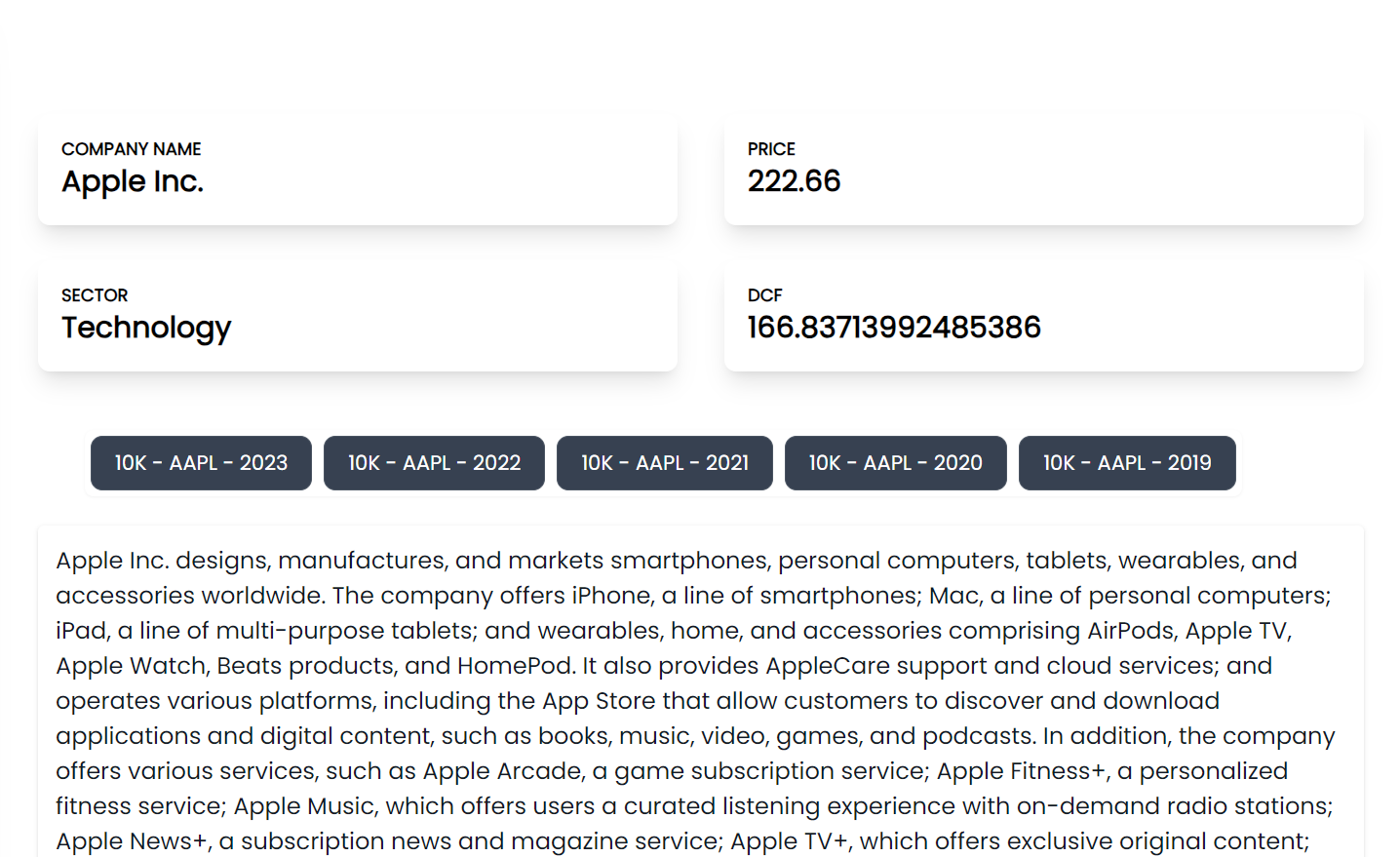


Рисунок 1 – Финансовые показатели компании

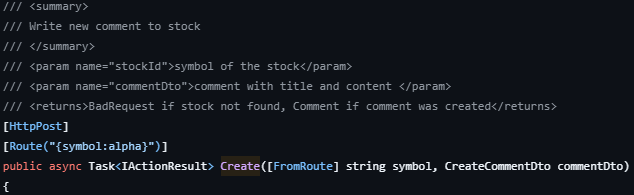


Рисунок 2 – Фрагмент функции добавления комментария

2.2 Парное программирование

Егор создал оконное приложение, установил пакеты: Microsoft.EntityFrameworkCore.Tools, Microsoft.EntityFrameworkCore.Design, Microsoft.EntityFrameworkCore.SqlServer. Написал класс User и класс AppDbContext для конфигурации базы данных с помощью EFCore. Добавил окна для авторизации и регистрации пользователей, написал функцию для проверки правильности логина и пароля.

Виталий отслеживал ошибки, исправлял некорректно названные переменные и оформлял отчет о работе.

Смена ролей.

Виталий создал окно просмотра списка пользователей и написал функции для импорта и экспорта данных.

Егор исправлял ошибки и проводил code-review.

Во время написания кода возникла ошибка, показанная на рисунке 3.

Ошибка исправлена путем внесения изменений в класс UsersRepository.

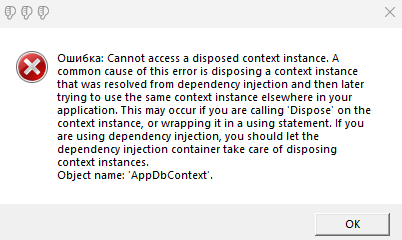


Рисунок 3 - Вид окна ошибки

2.3 Пинг-понг программирование

При разработке окна регистрации Егор написал модульный тест ValidateRegisterData\_ValidInput\_ReturnsTrue для метода валидации данных. Модульный тест проверяет, что name, email, login, password не пустые. Виталий написал метод ValidateRegisterData так, чтобы он проходил модульный тест.

Виталий написал модульный тест IsValidPasswordLength\_ShortPasswords\_ReturnsFalse для метода валидации длины пароля. Модульный тест проверяет, что длина пароля больше или равна 5 символам. Егор написал метод IsValidPasswordLength так, чтобы он проходил модульный тест.

2.4 Просмотр кода (code review)

Код одногруппника расположен по адресу:

<https://github.com/x1roko/bk654/blob/master/bk654/StartWindow.xaml.cs>

Замечания к методу, показанному на рисунке 4:

- метод не содержит обработки исключений,

- метод не проверяет, был ли найден результат запроса перед установкой текста в nameTextBlock,

- метод использует форматирование строки для создания SQL-запроса, что небезопасно,

- отсутствуют комментарии к коду.

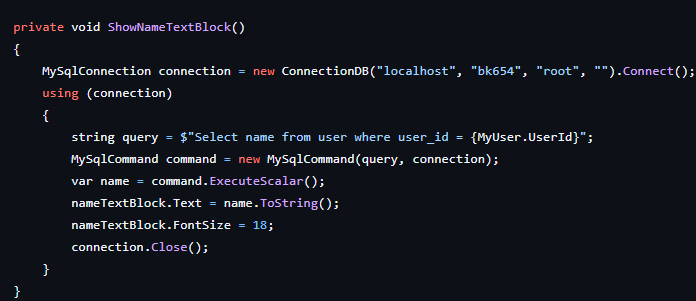


Рисунок 4 - Фрагмент кода одногрупника

Код, находящийся в открытом репозитории, расположен по адресу:

https://github.com/4qiz/wsr/blob/develop/Desktop/ScheduleApp/ScheduleApp/Windows/LoginWindow.xaml.cs

Замечания к методу, показанному на рисунке 5:

- метод не содержит обработки исключений,

- метод не соответствует принципу Single Responsibility, так как выполняет получение данных из БД, хеширование пароля, выдачу роли пользователю, что делает его менее поддерживаемым,

- роль пользователя хранится в виде строки, что может привести к ошибкам,

- отсутствуют комментарии к коду.



Рисунок 5 - Фрагмент кода из открытого репозитория

2.5 Применение специализированного ПО для инспекции кода

Инспекция кода выполнена с помощью сервиса Aikido Security. Результат показан на рисунке 6. Сервис выявил устаревшие версии библиотек, например Npgsql 8.0.2. Также выявлена возможная утечка секретных ключей в файле appsetting,json, как показано на рисунке 7. Нажатие на кнопку «Create Autofix» позволяет автоматически исправить уязвимости.

Сервис Aikido Security выбран, так как он бесплатный, позволяет удобно провести анализ кода, имеет интеграцию с GitHub.



Рисунок 6 - Вид окна «Aikido Security»

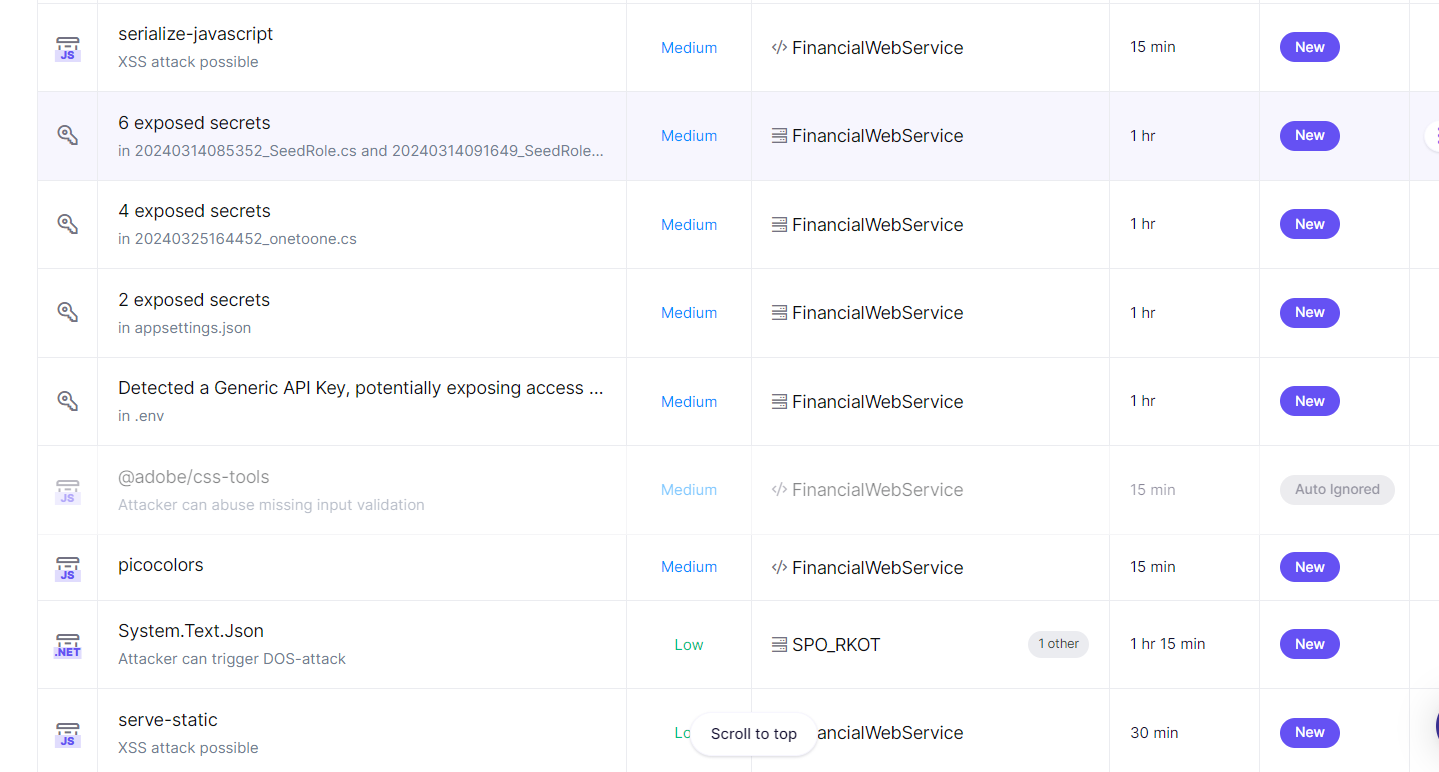


Рисунок 7 - Вид окна «Aikido Security»

**3 Контрольные вопросы**

8.1 Что такое «инспекция программного кода»?

Инспекция программного кода - это процесс анализа и оценки качества кода программы с целью выявления потенциальных проблем, улучшения его структуры и повышения общего уровня надежности.

8.2 Что такое «code review»?

Code review - это процесс проверки кода разработчиком или группой разработчиков перед его интеграцией в основную линию разработки.

8.3 Какие критерии учитываются при оценке качества ПО?

Функциональная пригодность, уровень производительности, совместимость, удобство использования, надёжность, защищённость, сопровождаемость, переносимость (мобильность).

8.4 Какие преимущества у парного программирования?

Быстрее обнаружение ошибок. Ускорение процесса разработки. Улучшение качества кода. Обучение и обмен опытом.

8.5 Какие типы инспекций кода существуют?

Автоматические инспекции, Code Review.

8.6 Какое ПО позволяет автоматизировать инспекцию программного кода?

Aikido Security, Upsource, embold.

**4 Вывод**

4.1 В ходе лабораторной работы изучили процесс выполнения инспекции программного кода.