Федеральное государственное образовательное бюджетное учреждение

высшего профессионального образования

**Финансовый университет при Правительстве Российской Федерации**

**(Финансовый университет)**

**Департамент анализа данных и машинного обучения**

**Факультета информационных технологий и анализа больших данных**

**Лабораторная работа №2**

**по дисциплине «Управление качеством программных систем»**

Направление подготовки - 09.03.09 «Прикладная информатика»

на тему: Работа с классификацией видов

тестирования.

**Подготовила**

студентка группы ЗБ-ПИ20-2

Булдакова Кира Александровна

(Ф.И.О.)

**Проверил**:

ассистент ДАДиМО ФИТиАБД,

Клочков Евгений Юрьевич

(Ф.И.О.)

Москва 2024

# **Цель работы:**

Целью данной лабораторной работы является классификация видов тестирования. Данная работа направлена на изучение и составление отчета о разных сторонах тестирования выбранного для исследования программного продукта. В процессе анализа будут изучены классификация тестирования, его разновидности и типы, а также будут подробно описаны методы и уровни тестирования, применяемые в данном проекте.

# **Описание программного проекта:**

Программный проект представляет собой приложение для демонстрации возможностей машинного обучения в прогнозировании эффективности инвестиционных проектов, а также, привидения примера работы механизма машинного обучения с помощью языка программирования Python (Приложение №1).

# **Список видов, типов, методов, уровней тестирования, применимых для выбранного проекта:**

1. **Виды тестирования:**
   1. Функциональное тестирование
   2. Нефункциональное тестирование (такое как производительность, безопасность, доступность)
   3. Регрессионное тестирование
   4. Бета-тестирование
2. **Типы тестирования:**
   1. Черный ящик (тестирование без знания внутренней структуры системы)
   2. Белый ящик (тестирование с полным пониманием внутренней структуры системы)
   3. Серый ящик (комбинация черного и белого ящика)
3. **Методы тестирования:**
   1. Ручное тестирование
   2. Автоматизированное тестирование
   3. Тестирование на основе моделей
4. **Уровни тестирования:**
   1. Модульное тестирование
   2. Интеграционное тестирование
   3. Системное тестирование
   4. Тестирование приемки

# **Примеры дефектов, характерных для каждого вида тестирования:**

1. **Функциональное тестирование** - проблемы с функциональностью: при запуске приложения на устройствах с другой версией Python (отличные от 3.10) наблюдаются сбои в работе (приложение не запускается или непроизвольно прекращает работу).
2. **Нефункциональное тестирование -** проблемы с производительностью, безопасностью и стабильностью: приложение работает медленно при большой нагрузке (в тестовом режиме были загружены данные о продажах акций за 10 лет только по одному эмитенту, при добавлении эмитентов или увеличения рассматриваемого периода скорость работы приложения сильно снижается).
3. **Регрессионное тестирование -** проблемы, которые возникли после внесения изменений в код: расчет среднеквадратичной ошибки (функция rmse) при добавлении данных не работает, так как требуется разделить расчет значения по каждому эмитенту отдельно.
4. **Бета-тестирование** - проблемы, которые не были замечены во время предыдущих этапов тестирования: при увеличении рассматриваемого периода (более 20 лет) полученные прогнозируемые значения продажи акций (цена закрытия - y\_train) отличаются от реальных значений (x\_train) более, чем на полученную среднеквадратичную ошибку. Вероятно, данный дефект системы связан с тем, что в выбранный при разработке программы период не наблюдалось резких изменений рынка, связанных с социо-политической обстановкой.

# **Выводы по работе:**

Изучение различных видов тестирования — это важный шаг в процессе разработки программного обеспечения. Каждый из видов тестирования помогает обнаружить определенные виды ошибок и дефектов в системе. По результатам тестирования можно сделать вывод, что тщательное и всестороннее тестирование — это ключ к созданию качественного и надежного программного обеспечения. Ошибки и дефекты, обнаруженные в процессе тестирования, помогают улучшить систему и предотвратить возможные проблемы в будущем.

# **Список используемых источников:**

1. Учебное пособие «Основы управления качеством программных средств»
2. В. Н. Пероцкая, Д. А. Градусов «Качество и надёжность программных систем»

# **Приложения:**

1. Булдакова\_ЗБ-ПИ20-2\_КР\_ПР№1