

УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Отчет

Об учебной практике

*Разработка алгоритма для автоматизированного участия в
аукционных сессиях*

Студент

Митрофанов Е. Ю.

P34101

Руководитель практики

Перл И. А.

Санкт-Петербург, 2023 г.

Оглавление

Введение	3
Тема	3
Цель	3
Задачи и этапы выполнения	3
Основная часть	4
1. Разработка внешней системы для интеграции и тестирования алгоритма	4
1.1. Описание идеи	4
1.2. Разработка прототипа	5
1.3. Разработка внешней платформы	7
2. Анализ существующих ботов и разработка алгоритма	9
2.1. Почему существующие решения не подходят	9
2.2. Экспоненциальная функция	10
2.3. Почему предложенный алгоритм работает	10
3. Разработка бота и интеграция с внешней платформой	11
3.1. Интеграция с внешней системой	11
3.2. Тестирования и проверка работоспособности	12
Выводы	13

Введение

Тема

Разработка алгоритма для автоматизированного участия в аукционных сессиях.

Цель

Разработка алгоритма и бота, реализующего алгоритм, для автоматизации процесса участия в аукционных сессиях. В процессе также необходимо разработать внешнюю систему для интеграции и тестирования разработанного бота.

Задачи и этапы выполнения

Порядковый номер этапа	Наименование этапа	Задание этапа
1	Инструктаж обучающегося	Инструктаж обучающегося по ознакомлению с требованиями охраны труда, техники безопасности, пожарной безопасности, а также правилами внутреннего трудового распорядка
2	Разработка внешней системы для интеграции и тестирования алгоритма	Разработка интерфейса и серверной части приложения. Приложение должно содержать необходимые функции - начало аукциона, ограничение по времени, ставка, отмена ставки, назначение цены на ставку, завершение аукциона и определение победителя
3	Анализ существующих ботов и разработка алгоритма	Анализ существующих ботов, выявление недостатков, разработка собственного алгоритма, его математическое обоснование и теоретическое доказательство
4	Внедрение алгоритма и создание бота	Создание бота, реализующего разработанную стратегию трейдинга, внедрение в ранее разработанную платформу, проведение тестов и доказательство работоспособности

5	Оформление отчета	Оформление отчетных документов в соответствии с требованиями
---	-------------------	--

Основная часть

1. Разработка внешней системы для интеграции и тестирования алгоритма

1.1. Описание идеи

Для интеграции разработанного бота и тестирования алгоритма его работы необходимо создать контролируруемую платформу, реализующую все необходимые функции

В качестве ориентира взят существующий «Портал Поставщиков» города Москвы. Веб-сайт предназначен для компаний, предоставляющим какие-либо услуги или товары (поставщики) и заказчиков, которые размещают заявки. Поставщики предлагают меньшую цену и «соперничают» между собой за право выполнить заказ, а сам заказчик по итогу уменьшает затраты, так как находит поставщика, готового выполнить заказ за более низкую цену.

Из описанного режима работы необходимо вынести минимально необходимый функционал, который должна поддерживать разрабатываемая внешняя платформа:

1. Разделение ролей на поставщиков и заказчиков
 - a. Регистрация с указанием роли
 - b. Вход в платформу с личным кабинетом
2. Создание новой аукционной сессии
 - a. Назначение начальной цены
 - b. Установка срока окончания сессии
3. Просмотр существующих сессий
 - a. Получение информации о текущей цене
 - b. Получение информации о начальной цене
 - c. Просмотр общей информации о сессии (создатель, время проведения, сколько осталось до завершения)
4. Возможность принять участие в сессии
 - a. Установка новой ставки с указанием цены
 - b. Отмена участия
 - c. Уведомление в случае победы
5. Дополнительные желательные функции платформы

- а. Просмотр своих ставок
- б. Настройка профиля
- с. Журнал уведомлений

1.2. Разработка прототипа

Для представления разрабатываемого приложения создан прототип в Figma

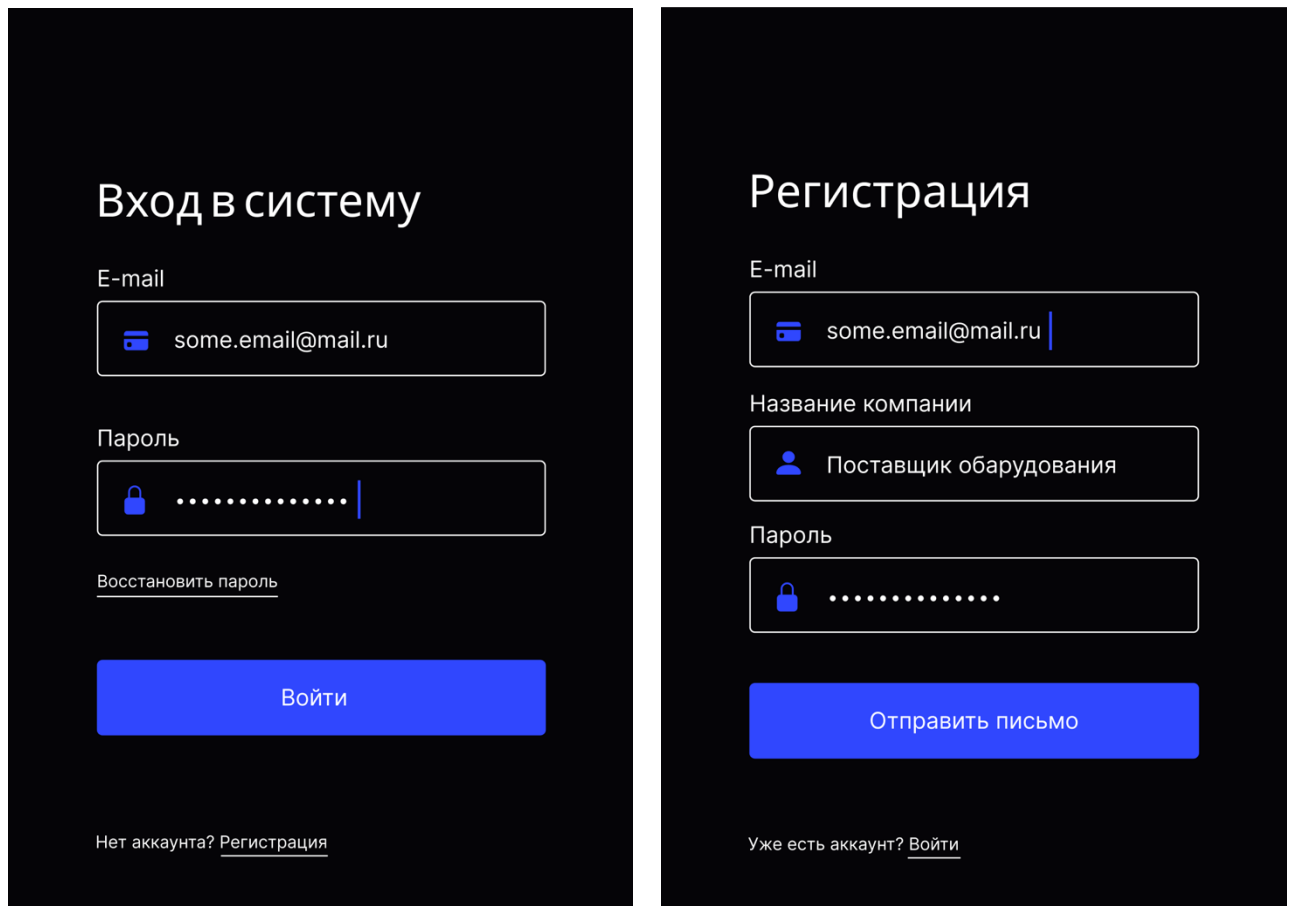


Рисунок 1, экраны регистрации и входа

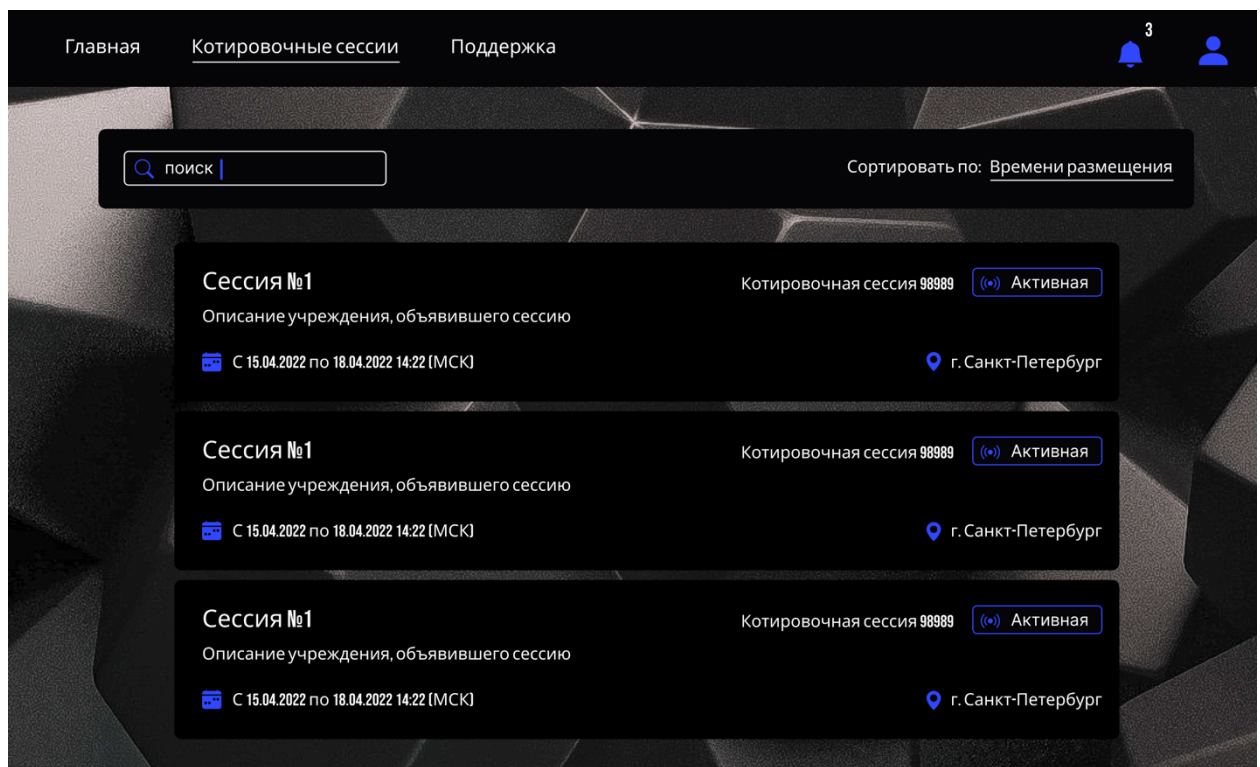


Рисунок 2. Основной экран просмотра активных сессий

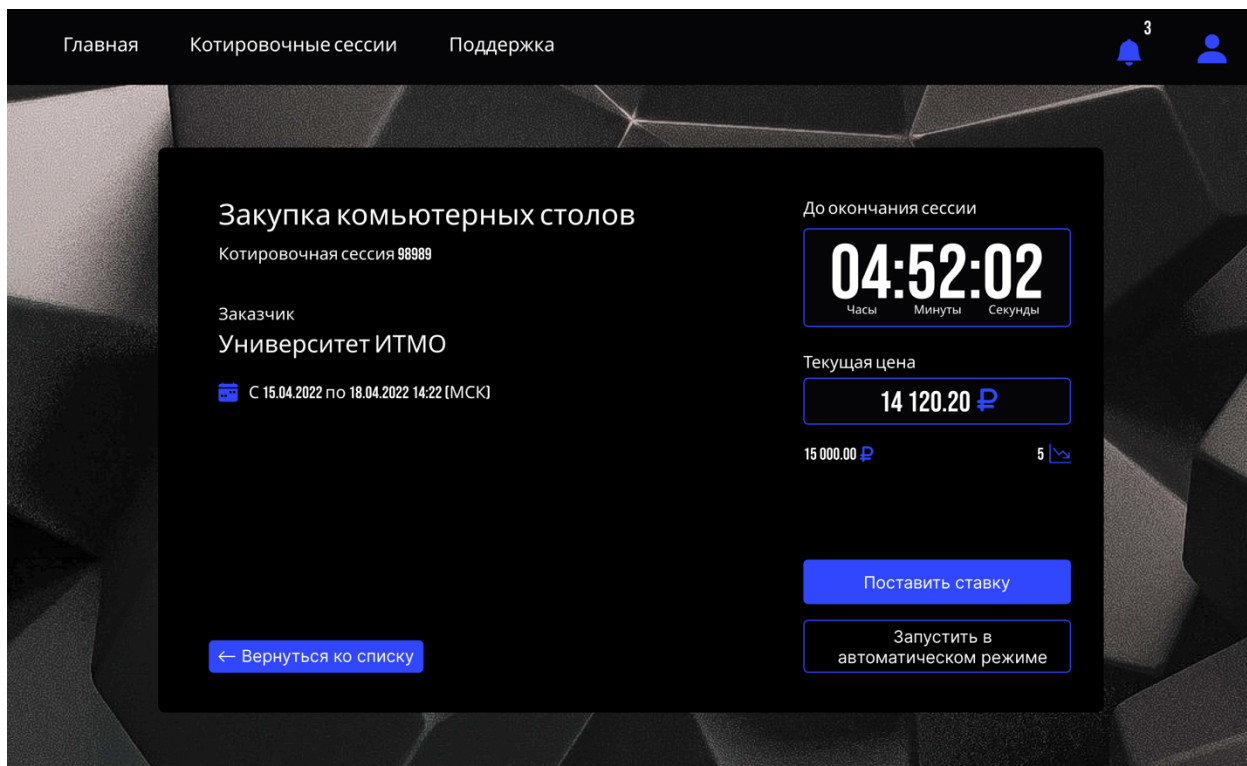


Рисунок 3, Просмотр сессии с возможностью участия

На экране просмотра сессий можно поставить свою ставку, выбрав новую цену в диалоговом окне

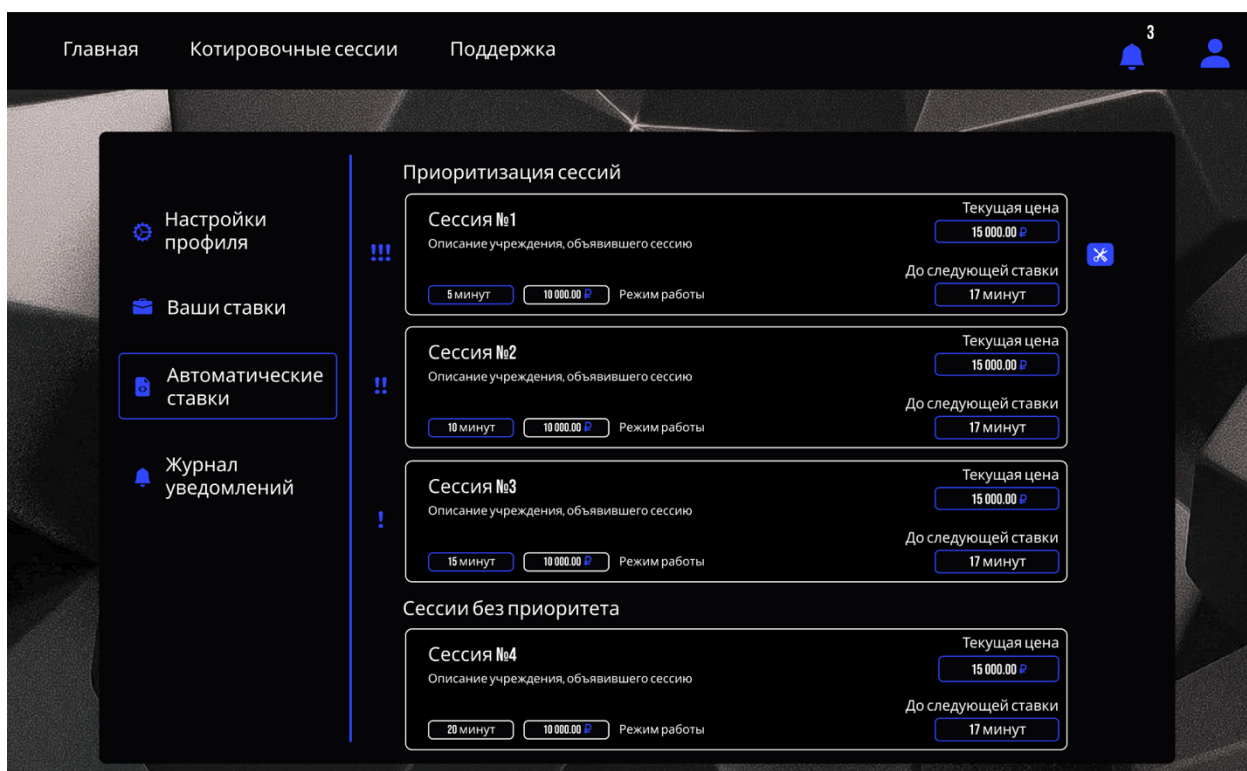


Рисунок 4, Личный кабинет с информацией о текущих сессиях

А личном кабинете есть возможность отменить любую свою ставку, а также функции автоматизации, которые будут описаны позже.

1.3. Разработка внешней платформы

Для разработки внешней системы был использован следующий стек технологий:

1. Серверное приложение:
 - a. Spring Boot – для запуска и развертывания
 - b. Spring Security + JWT для обеспечения механизма разделения ролей, авторизации и аутентификации
 - c. Postgres + Flyway - база данных и система контроля миграций для хранения всех данных
 - d. Swagger 2.0 для описания и интерактивного взаимодействия с API разработанного приложения
2. Клиентское приложение:
 - a. React MobX
 - b. Type Script
 - c. Chakra UI, Styled Component

Очень важно подготовить качественную и понятную OpenApi спецификацию для дальнейшего использования при интеграции бота

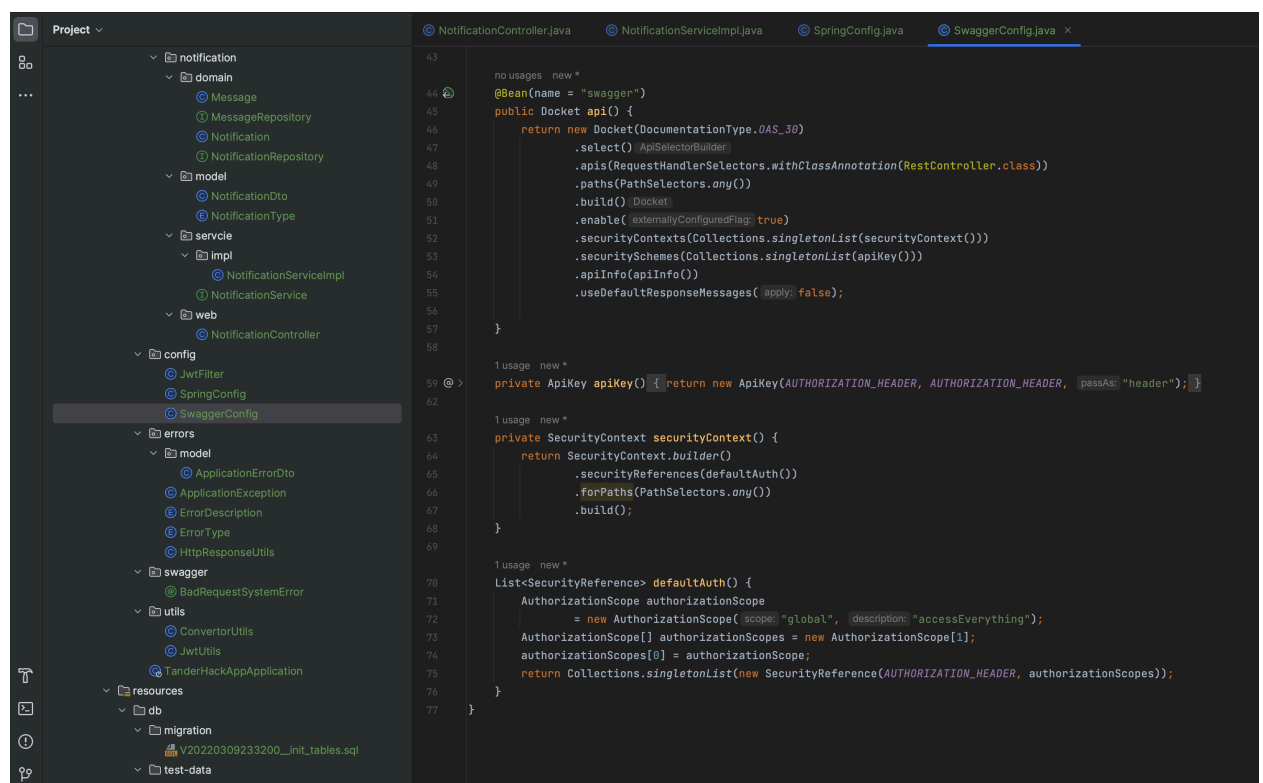


Рисунок 5, исходный код внешней платформы

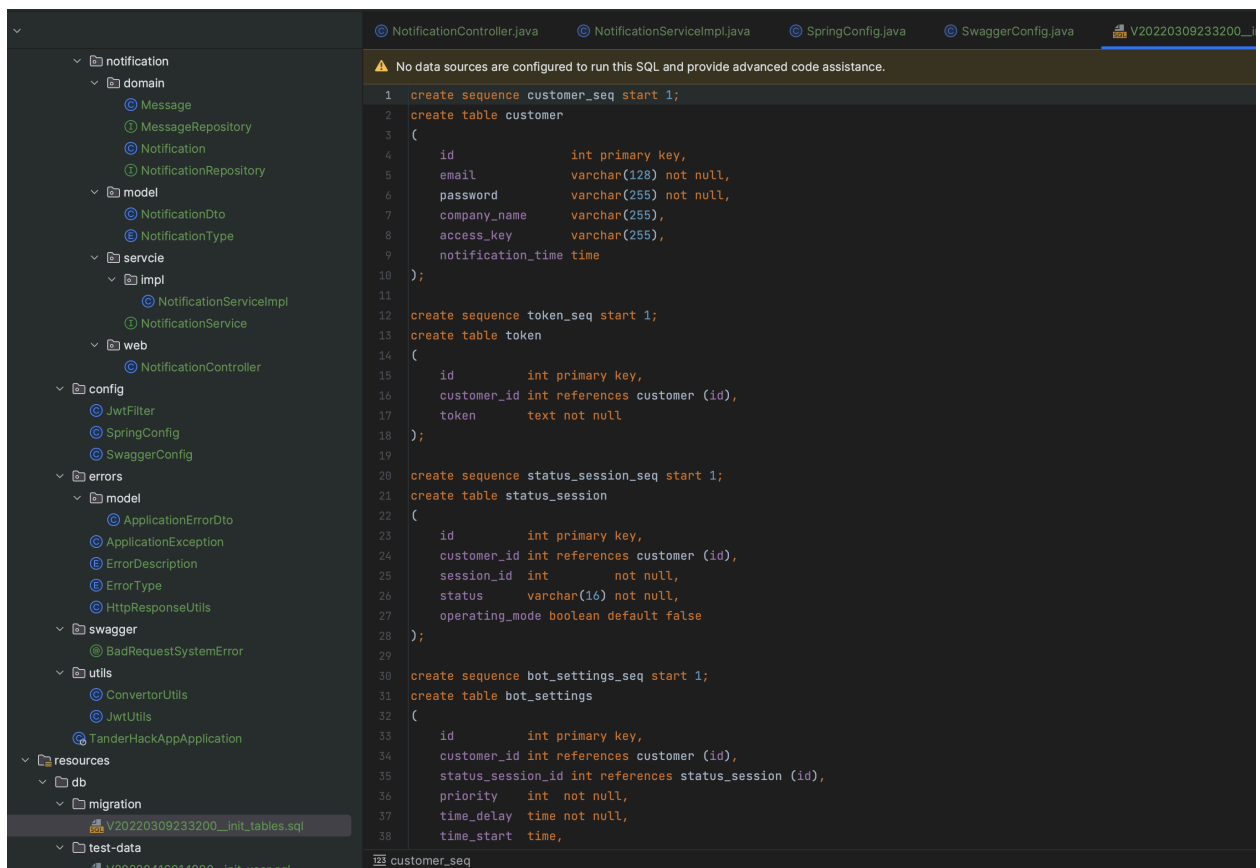


Рисунок 6, создание и управление данными с помощью системы контроля миграций

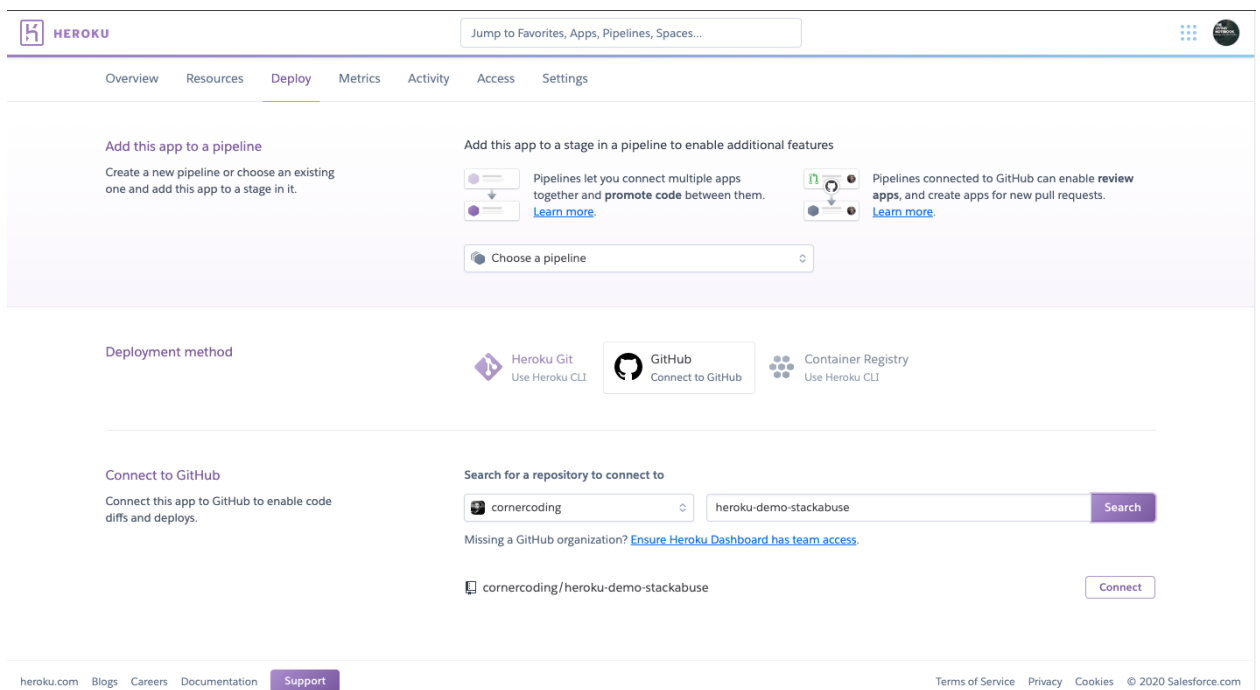


Рисунок 7, развертывание приложения на сервере Heroku

2. Анализ существующих ботов и разработка алгоритма

2.1. Почему существующие решения не подходят

При попытке найти готовое решение, которое будет решать задачу автоматизированного участия в аукционных сессиях, можно заметить, что большинство ботов являются «автокликерами», усложнёнными в разных аспектах.

Такие системы не подходят для использования по нескольким причинам:

- Подобные боты всегда имеют превосходство над человеком, а соревноваться с ними почти невозможно
- Автокликеры сильно нагружают систему, поэтому большинство серверов будут блокировать частую отправку запроса или использовать другие механизмы защиты, такие как сарча или повторная авторизация
- Участие подобных ботов в аукционных сессиях приведет к тому, что цена будет моментально снижена до минимально возможного значения, а в такой ситуации пропадает сам смысл проведения аукционной сессии. Кроме того ограничения по времени перестают что-либо значить, так как для при наличии большого количества ботов на платформе аукцион будет закончен за несколько секунд

Стоит отметить, что поставленная задача – разработать бота, цель которого именно автоматизировать процесс участия, а не перебить остальные ставки. Наша цель – задать некоторые начальные параметры и запустить контролируемый процесс, кроме того, очень важно разработать такой алгоритм работы, который не будет давать непреодолимое преимущество боту и позволит другим участникам все так же участвовать в «ручном» режиме.

При разработке алгоритма необходимо обозначить основные параметры:

- Время обновления ставки – как часто бот сможет обновлять и проверять ставку
- Шаг снижения цены – насколько сильно бот может снижать цену
- Минимально возможная цена, после которой бот отключиться

2.2. Экспоненциальная функция

В результате работы был выявлен наилучший подход – использование экспоненциальной функции.

Чем больше разница между текущей ценой и минимальной установленной ценой, тем реже бот обновляет ставку. Как только текущая цена начинает приближаться к необходимому значению, интервал ставки сокращается все сильнее.

Кроме того, чтобы не перегружать систему необходимо выставить минимальный доступный интервал в несколько секунд.

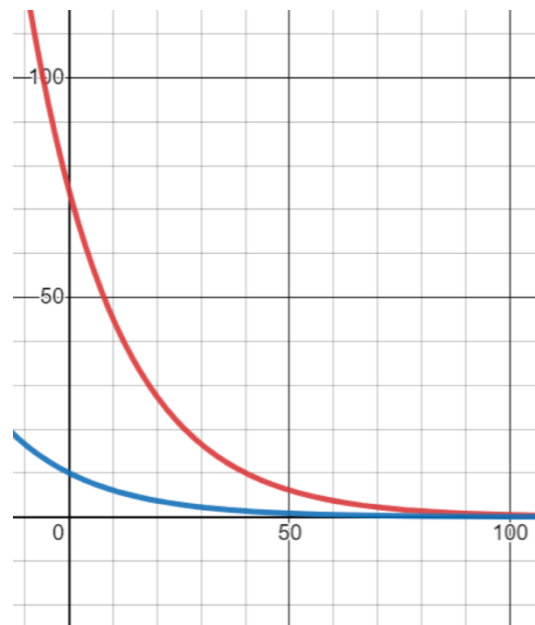


Рисунок 7, Пример экспоненциальной функции

Итоговая функция имеет следующий вид:

$$delay = 0.5 \cdot e^{\left(-0.05 \cdot \left(100 - \frac{\min Price}{currentPrice} \cdot 100\right) + 5\right)}$$

Рисунок 8, Функция вычисления интервала ставки

Основная зависимость происходит между минимально допустимой вставленной ценой и текущей ценой в сессии. Также стоит отметить, что подобранные коэффициенты можно менять, меняя таким образом стратегию трейдинга.

2.3. Почему предложенный алгоритм работает

- Участник, предложивший наименьшую цену, победит с большей вероятностью. Система дает участнику, близкому к минимальной цене, больше возможностей сделать ставку, одновременно с этим ограничивая пользователей, близких к победе

- Участник сам решает, что наиболее важно для него. Настройка стратегии трейдинга и начальных параметров будет напрямую влиять на ведение торгов. Если у пользователя есть возможность дальше снижать цену, будет выбрана более агрессивная стратегия понижения ставки, а если цена приближается к минимально допустимой, то шаг снижается, но повышается частота обновления ставки
- Алгоритм не ломает проведение торгов для других участников. По поведению пользователя даже нельзя сразу понять, что используется автоматизация обновления ставок. Кроме того, в любой момент пользователь может взять управление ведением торгов в свои руки и перевести бота в ручной режим. В таком случае он даже может получить преимущество

3. Разработка бота и интеграция с внешней платформой

3.1. Интеграция с внешней системой

Описанный ранее алгоритм будет реализовывать самостоятельное приложение, использующее ранее разработанную платформу и спецификацию OpenApi.

Бот получает информацию о доступных сессиях, после чего пользователь может участвовать в аукционе в автоматическом режиме, задав начальные параметры для бота. Каждое действие бота сопровождается уведомлением на указанную в профиле почту. Таким образом пользователь всегда знает, какие действия совершаются от его имени и какая стратегия трейдинга сейчас у бота.

Бот хранит контекст пользователя и сессии, для которой он запущен, и после каждого обновления ставки пересчитывает следующий интервал.

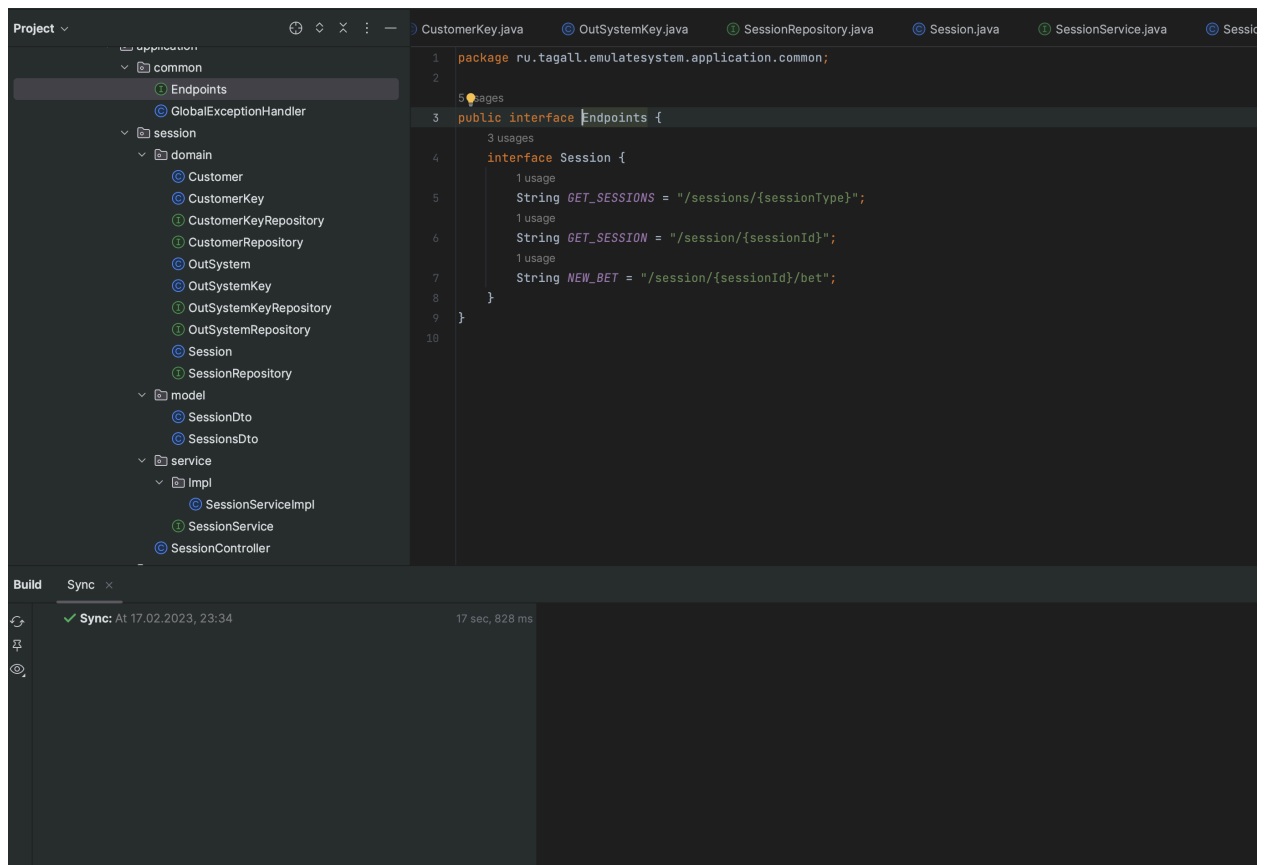


Рисунок 9, интеграция с внешней системой

3.2. Тестирования и проверка работоспособности

Для автоматического тестирования были разработаны интеграционные тесты с помощью библиотеки JUnit 4 и Mockito. Записанные заранее тестовые данные используются для расчета интервалов обновления и проверки корректности вычисляемых значений.

Также бот был протестирован с помощью специально разработанной внешней системы, вместе с реальными пользователями. Несколько тестовых сценариев предполагали участие в одной сессии одного бота и нескольких реальных пользователей, а также несколько ботов одновременно.

Результат тестирования дал понять, что бот не имеет преимущества перед человеком, а лишь автоматизирует конкретный бизнес-процесс, поэтому при соблюдении определенных правил, таких как ограничение минимального времени обновления ставки, его можно использовать и адаптировать и под другие аукционные сессии.

Выводы

По результатам прошедшей учебной практики был разработан и реализован инструмент, позволяющий автоматизировать участие человека в аукционных сессиях. Для тестирования бота была разработана отдельная система интеграции с открыты API. Были проведены тесты, целью которых было сравнение работы бота и реального пользователя.

Разработанный бот достигает поставленных целей, а именно не имеет большого преимущества перед человеком, позволяет участвовать одновременно и реальным пользователям, и автоматизированным системам, а также решает задачу автоматического обновления ставки.

Во время работы я на практике познакомился с несколькими платформами, реализующими процесс аукционных сессий, самостоятельно воссоздал подобные процессы в разработанном приложении, проанализировал существующие подходы к автоматизации и предложил свой, выделив его преимущества. Также я изучил несколько новых инструментов разработки, деплоя и интеграции приложений.