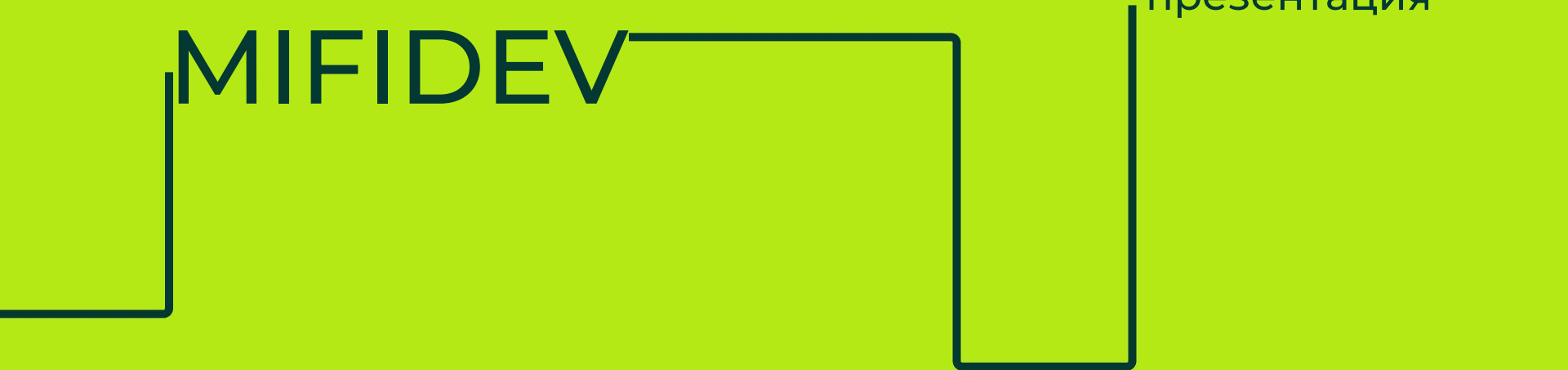


Хакатон MIFIDEV

Итоговая
презентация



Команда



Владислав
Почернин



Александр
Рыжиков



Даниил
Ткаченко



Борис
Заграй



Владимир
Трубарев

Задача

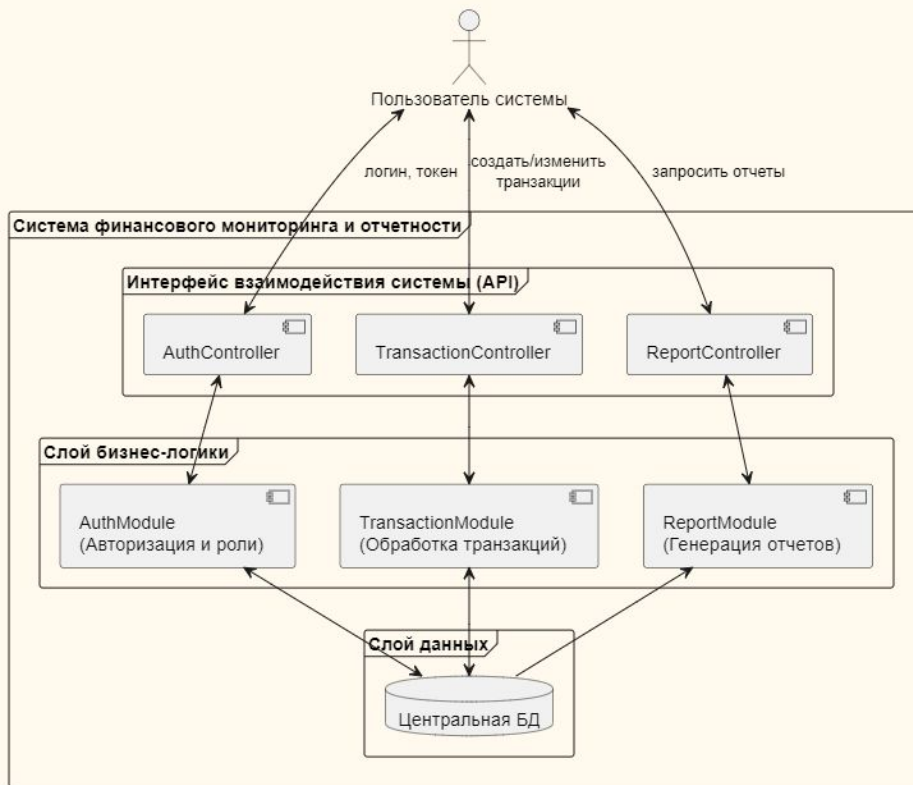
Бриф: “Финансовый мониторинг и отчетность” от партнера Globus IT.

Стэк проекта: Java, Spring Boot, Spring Web MVC, Spring Security, Spring Data, JUnit, PostgreSQL, JWT, Swagger, Docker, Flyway, Maven

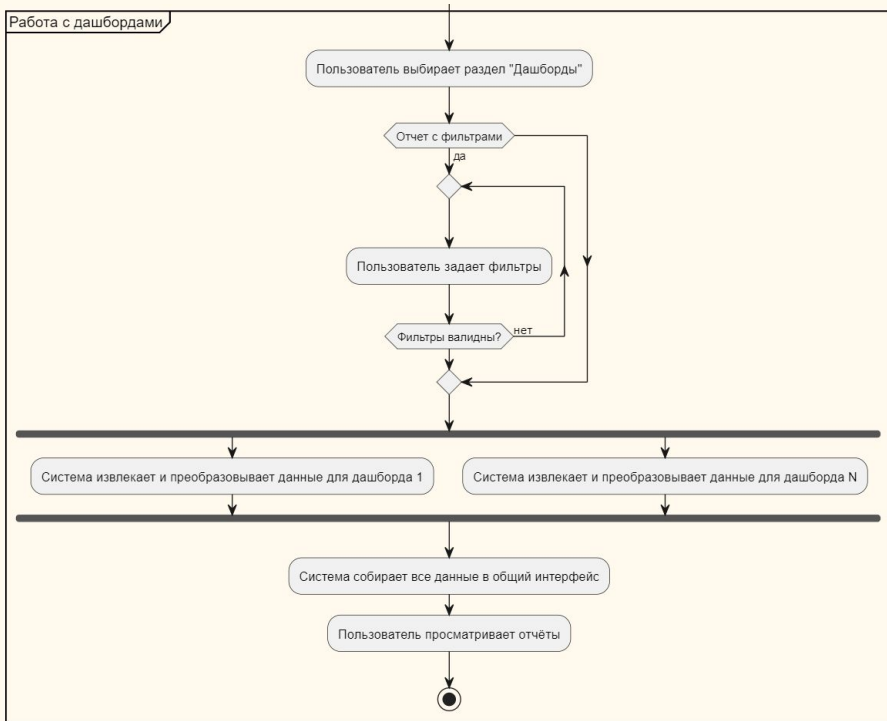
Задача: Реализовать веб-приложение, позволяющее отслеживать доходы и расходы, а также получать аналитические отчеты. Приложение должно включать систему для ввода финансовых данных с возможностью генерации отчетов по различным параметрам для анализа тенденций.

Архитектура проекта

Архитектура системы финансового мониторинга и отчетности



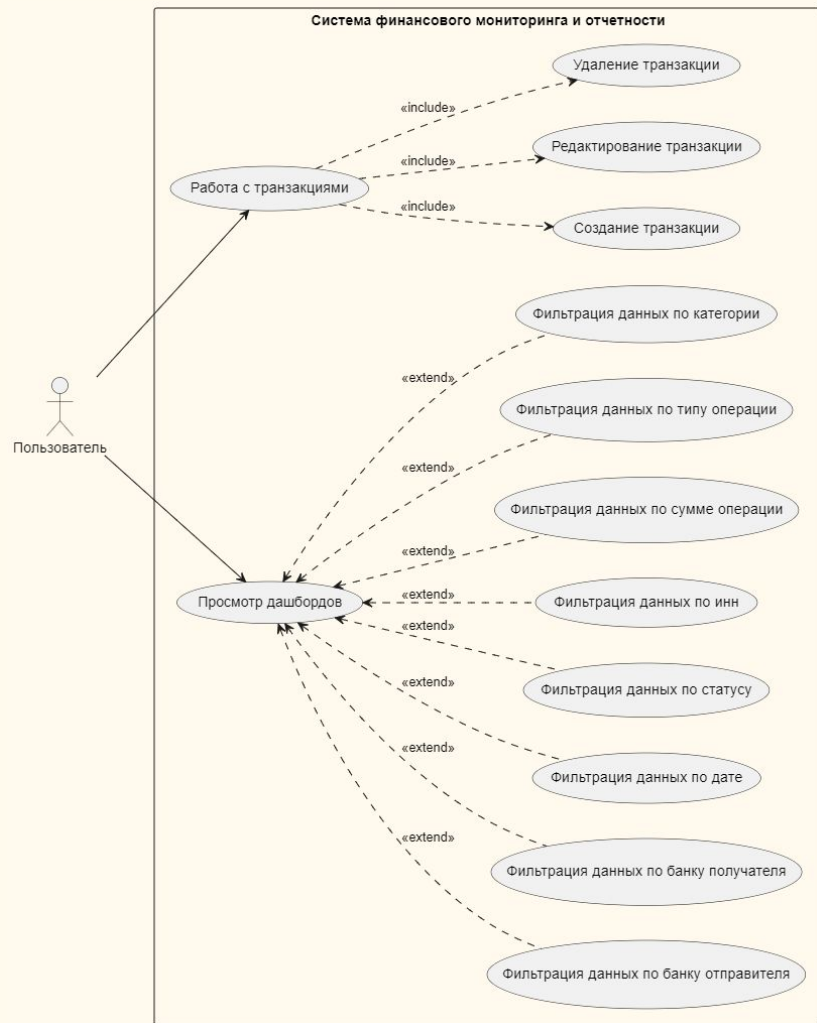
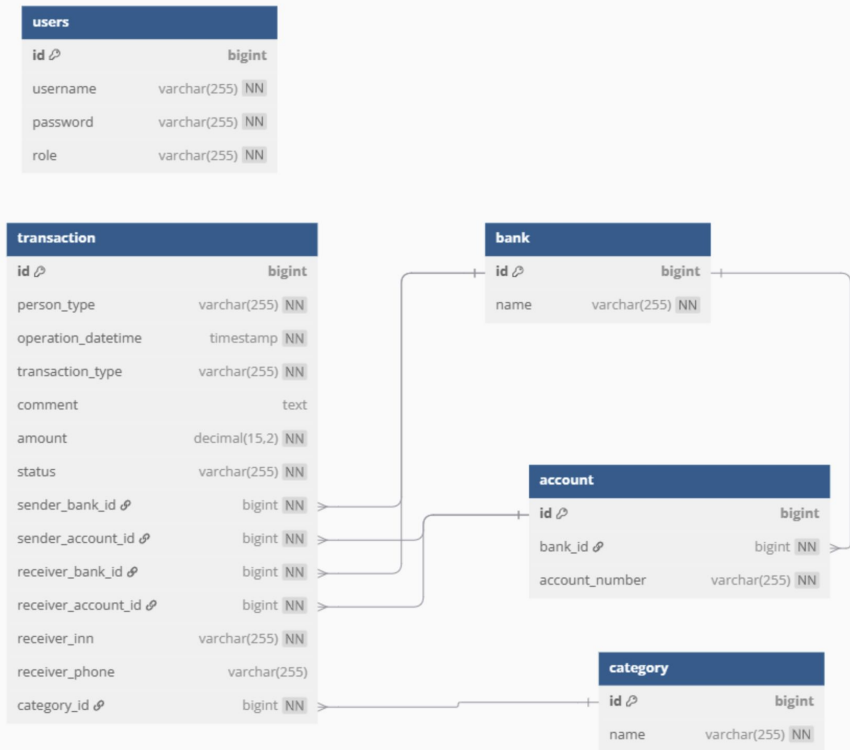
Краткое описание решения



```
{  
  "id": 1,  
  "description": "Пользователь с таким логином уже существует",  
  "errorType": "LOGIN_EXISTS",  
  "additionalInfo": ""  
}
```

```
{  
  "id": 4,  
  "description": "Запрашиваемый элемент не найден",  
  "errorType": "NOT_FOUND",  
  "additionalInfo": "Пользователь со следующим логином не найден: string1"  
}
```

Диаграммы



Тестирование

✓ <default package>	438 ms
✓ FinmonitorApplicationTests	306 ms
✓ contextLoads()	306 ms
✓ TransactionServiceTest	132 ms
✓ delete_NewTransaction_ChangesStatusToDeleted()	120 ms
> ✓ delete_ForbiddenStatus_ThrowsException(String)	11 ms
✓ delete_NonExistentTransaction_ThrowsException()	1 ms
✓ TransactionValidatorTest	
> ✓ validate_ValidDates_NoException(String)	
> ✓ validate_InvalidPhone_ThrowsException(String)	
> ✓ validate_ValidInn_NoException(String)	
> ✓ validate_ValidPhone_NoException(String)	
✓ validate_ValidData_NoException()	
> ✓ validate_InvalidInn_ThrowsException(String)	
✓ validate_InvalidDates_ThrowsException(String)	
✓ [1] date=32.01.2025	
✓ [2] date=00.01.2025	
✓ [3] date=01.13.2025	
✓ [4] date=01.00.2025	
✓ [5] date=01.01.202	
✓ [6] date=01-01-2025	
✓ [7] date=1.1.2025	
✓ [8] date=01.1.2025	
✓ [9] date=1.01.2025	
✓ [10] date=	

Сценарии тестирования

Ниже представлен реестр тест-кейсов для ручного тестирования системы.

Работа с транзакциями

ТС-001: Создание новой транзакции

Параметр	Значение
Предусловие	Пользователь авторизован
Шаг 1	Перейти в раздел «Транзакции»
Шаг 2	Нажать кнопку «Создать»
Шаг 3	Заполнить обязательные поля (дата, сумма, категория и т.д.)
Шаг 4	Нажать «Сохранить»
Ожидаемый результат	Транзакция отображается в списке, данные сохранены в базе

ТС-002: Редактирование существующей транзакции

Параметр	Значение
Предусловие	В системе есть хотя бы одна транзакция
Шаг 1	Найти транзакцию в списке
Шаг 2	Нажать кнопку «Редактировать»
Шаг 3	Изменить данные (например, сумму)
Шаг 4	Сохранить изменения
Ожидаемый результат	Обновлённые данные видны пользователю и корректно сохранены

ТС-003: Удаление транзакции

Параметр	Значение
Предусловие	Существующая транзакция в списке
Шаг 1	Выбрать нужную транзакцию
Шаг 2	Нажать кнопку «Удалить»
Шаг 3	Подтвердить удаление
Ожидаемый результат	Транзакция удалена из списка и базы

Просмотр отчётов

ТС-004: Просмотр отчёта по умолчанию

Параметр	Значение
Предусловие	Есть хотя бы одна транзакция
Шаг 1	Перейти в раздел «Отчёты»
Ожидаемый результат	Отчёт отображается с агрегированной информацией по всем транзакциям


ТС-005: Фильтрация данных в отчётах

Параметр	Значение
Предусловие	Несколько транзакций с разными параметрами (датами, категориями)
Шаг 1	Перейти в «Отчёты»
Шаг 2	Открыть панель фильтров
Шаг 3	Указать, например, диапазон дат или категорию
Шаг 4	Применить фильтр
Ожидаемый результат	Отчёт перестраивается и отображает только подходящие транзакции

ТС-006: Поведение системы при отсутствии транзакций

Параметр	Значение
Предусловие	Отсутствует
Шаг 1	Перейти в раздел «Транзакции» или «Отчёты», когда данных нет
Ожидаемый результат	Сообщение о пустом списке, возможность создать первую транзакцию

UI

 Swagger
framework for REST APIs

v3/api-docs

Explore

OpenAPI definition v0 OAS 3.1
v3/api-docs

Servers
http://localhost:8080 - Generated server url

Authorize

transaction-controller

GET

/api/transactions/{id}

Получить транзакцию по ID

PUT

/api/transactions/{id}

Обновить транзакцию

DELETE

/api/transactions/{id}

Удалить транзакцию

POST

/api/transactions

Создать транзакцию

POST

/api/transactions/filter

Фильтрация транзакций

POST

/api/transactions/dashboard/operationTypeDynamics

Динамика типов операций

POST

/api/transactions/dashboard/operationDynamics

Динамика операций

POST

/api/transactions/dashboard/operationCount

Количество операций

POST

/api/transactions/dashboard/incomeExpensesComparison

Сравнение доходов и расходов

POST

/api/transactions/dashboard/categoryStatistics

Статистика по категориям

POST

/api/transactions/dashboard/bankStatistics

Статистика по банкам

GET

/api/transactions/all/{limit}

auth-controller

POST

/api/auth/register

Регистрация нового пользователя

POST

/api/auth/login

Авторизация пользователя

auth-controller

POST /api/auth/register Регистрация нового пользователя

Создание нового пользователя в системе. Возвращает информацию о зарегистрированном пользователе.

Parameters

No parameters

Request body required

application/json

```
{  "username": "string",  "password": "string"}
```

Execute

Responses

Code	Description	Links
200	Пользователь успешно зарегистрирован Media type */* Content Accept header Example Value Schema <pre>{ "message": "string", "username": "string"}</pre>	No links
400	Некорректные данные для регистрации Media type */* Example Value Schema <pre>{ "message": "string", "username": "string"}</pre>	No links

Ссылки на проект и материалы

- GitHub (см. README.md) - <https://github.com/vspochernin/finmonitor>
- Документация -
https://github.com/vspochernin/finmonitor/blob/master/documentation/documentation%20mephi%20fin_project.md
- Swagger (работает после запуска проекта) -
<http://localhost:8080/swagger-ui/index.html>
- Видео питча -
<https://drive.google.com/file/d/1RPB29l0nT5EWAwb5TTSYw3uF4k2-Agnu/view?usp=sharing>

Спасибо за внимание!

Команда № 1

SKILLFACTORY

