

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего профессионального образования
«Южно-Уральский государственный университет (национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра системного программирования

Разработка серверной части приложения для столовой «Восточный дракон»

Научный руководитель:
ст. преподаватель кафедры СП
Н.С. Силкина

Автор:
студент группы КЭ-401
И.С. Лазарев

Челябинск-2024

АКТУАЛЬНОСТЬ

- Автоматизация бизнес-процессов
- Сбор статистики о клиентах и их покупках
- Необходимость внедрения онлайн-заказов
- Дороговизна использования готовых сервисов

ЦЕЛИ И ЗАДАЧИ

Цель:

Разработка серверной части приложения для столовой «Восточный дракон»

Задачи:

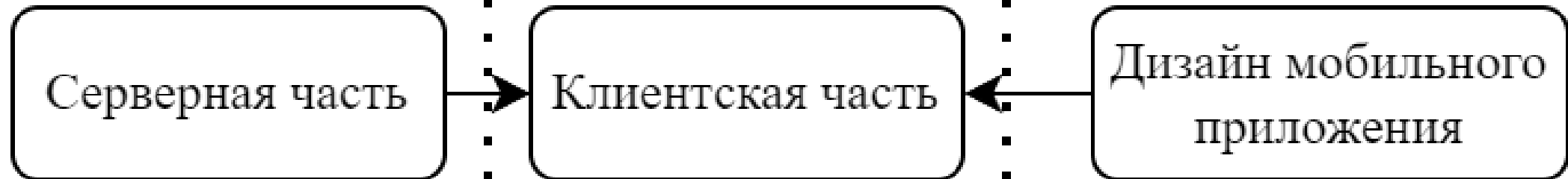
1. Провести анализ предметной области и обзор аналогичных систем
2. Разработать структуру таблиц базы данных
3. Разработать API
4. В рамках реализации API реализовать интеграции с сервисами онлайн-оплаты и push-уведомлений
5. Протестировать API

КОМАНДА ПРОЕКТА

Лазарев Иван
студент 4 курса
ВШЭКН

Стецкий Никита
студент 4 курса
ВШЭКН

Губанова Виктория
студентка 4 курса
ВШЭКН



СТОЛОВАЯ «ВОСТОЧНЫЙ ДРАКОН»

В данном заведении люди могут прийти и сделать заказ на месте. Можно выбрать как готовые товары на витринах, так и заказать горячие блюда или собрать целый обед (товар, с фиксированной ценой и возможностью выбора его различных составляющих)



АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

В столовой «Восточный дракон» можно заказывать блюда только на месте

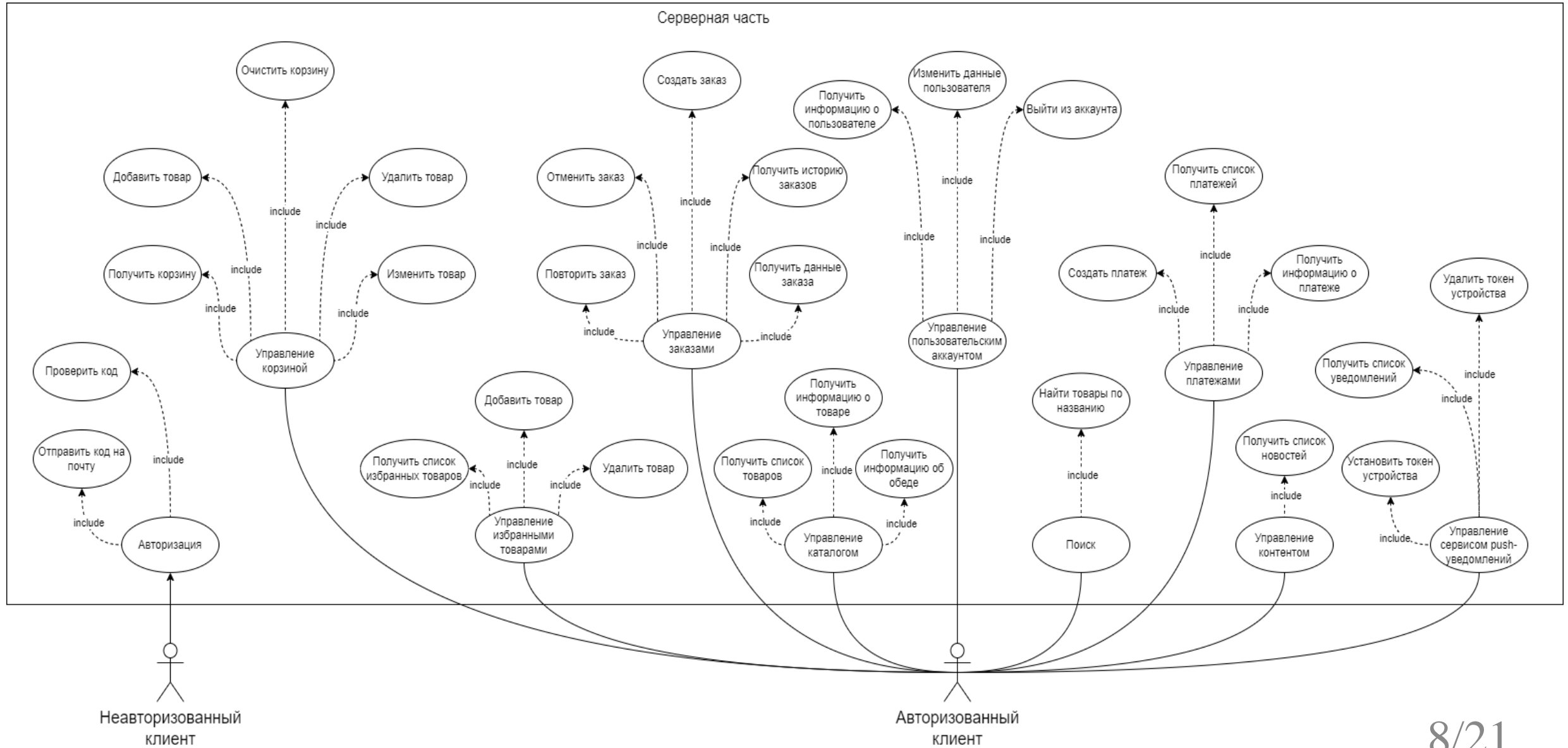
- Длинные очереди в часы-пик из-за подготовки, упаковки заказов и отсутствия состава блюд
- Ухудшает общее впечатление клиентов от посещения заведения

Приложение для онлайн-заказов с серверной частью предоставляющей API позволит решить данную проблему, автоматизировать работу заведения и легко интегрироваться с другими сервисами

СРАВНЕНИЕ ФУНКЦИОНАЛЬНОСТИ ПРИЛОЖЕНИЙ

№	Функционал	Gordan	Kamakura	Разрабатываемое приложение
1.	Избранное	—	+	+
2.	Оформление заказа	+	+	+
3.	Вывод каталога	+	+	+
4.	Вывод детальной карточки товара	+	+	+
5.	Редактирование комбо товара	+	—	+
6.	Вывод новостей и акций	+	+	+
7.	Авторизация	+	+	+
8.	Редактирование личного кабинета	+	+	+
9.	Вывод контактной информации	+	+	+
10.	Просмотр заказа	+	+	+
11.	Просмотр истории заказов	+	+	+
12.	Поиск	—	+	+
13.	Управление корзиной товаров	+	+	+
14.	Уведомления	+	+	+
15.	Онлайн-оплата	+	+	+

ВАРИАНТЫ ИСПОЛЬЗОВАНИЯ



КОМПОНЕНТЫ СИСТЕМЫ

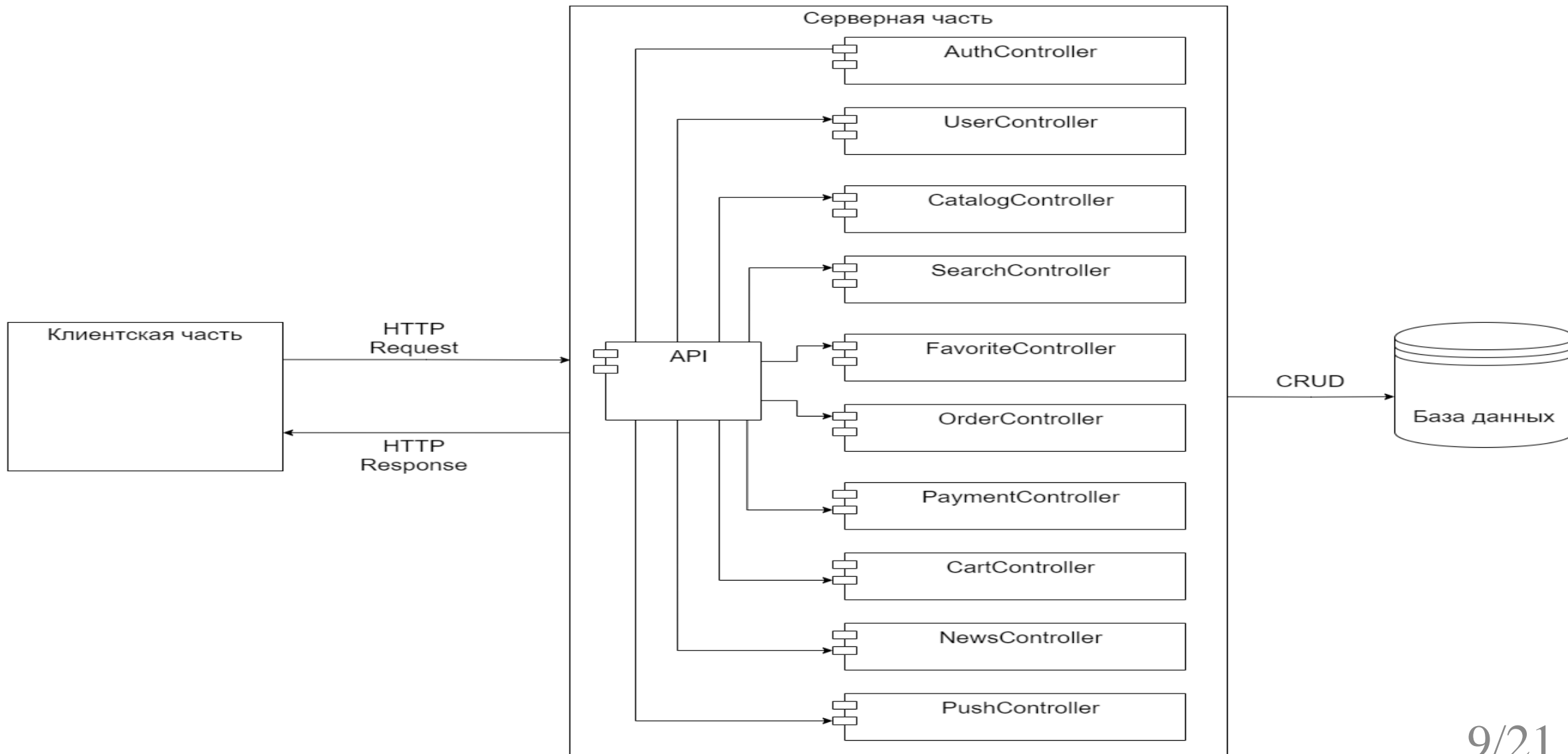
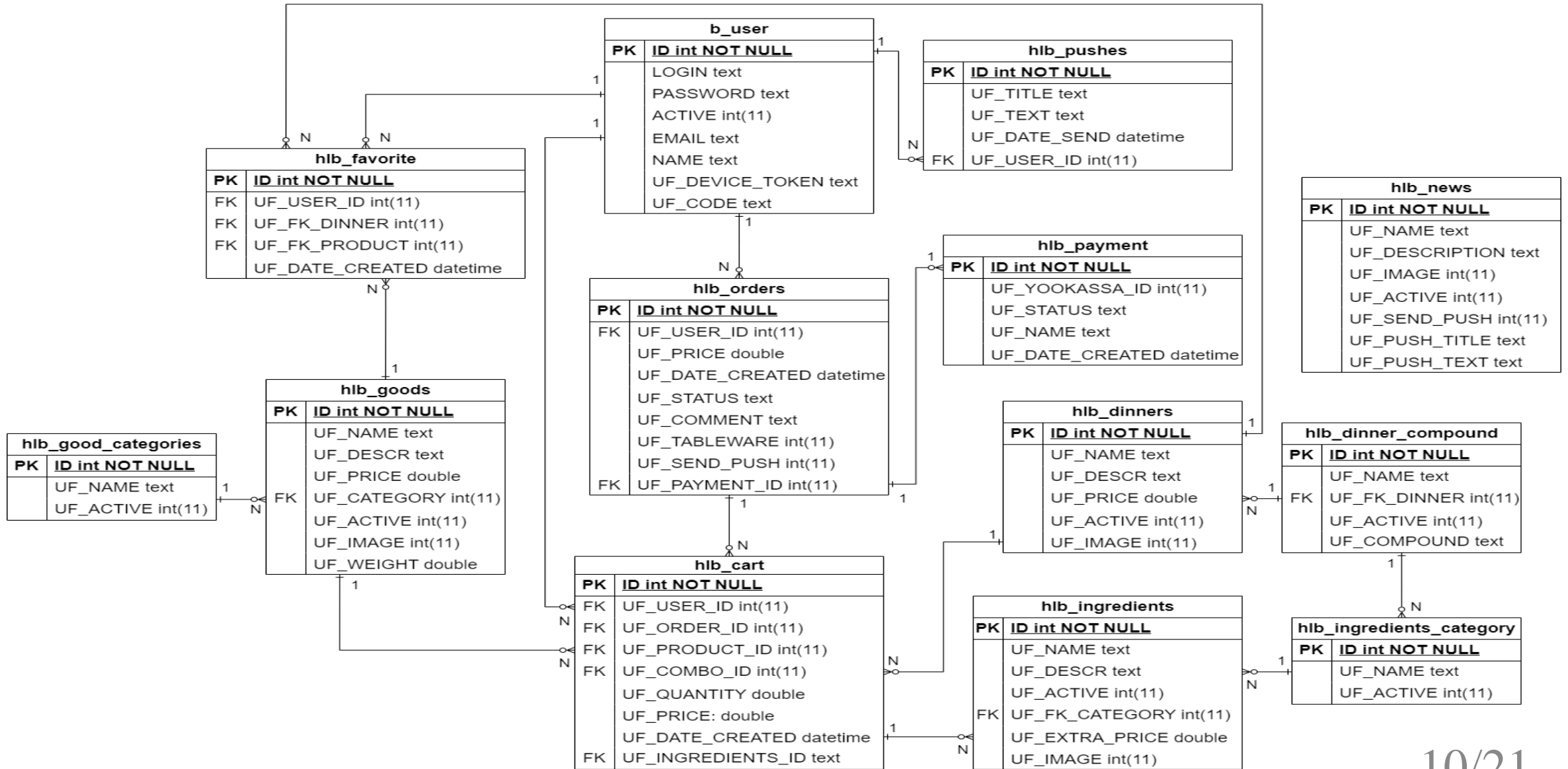


СХЕМА БАЗЫ ДАННЫХ



ЗАПРОСЫ API

Формат передачи данных – multipart/form-data, стандартный тип содержимого (Content-Type)

Формат ответа от сервера – JSON

Реализовано: 30 запросов

GET: {API_URL}/api/catalog/:id – Получение товара

POST: {API_URL}/api/user/code/ – Проверка кода

PUT: {API_URL}/api/user/ – Изменить данные пользователя

DELETE: {API_URL}/api/cart/:id – Удалить товар из корзины

*{API_URL} – адрес до API на сервере

СРЕДСТВА РЕАЛИЗАЦИИ ПРОЕКТА

- CMS 1С-Bitrix
- Хостинг reddock.ru
- Язык программирования PHP 8.2
- СУБД MySQL
- Среда разработки PhpStorm 2024



ТАБЛИЦА В АДМИНИСТРАТИВНОЙ ПАНЕЛИ

Highload-блок "Товары": Список записей ☆

Фильтр



ID

Найти

Отменить



+ Добавить запись



<input type="checkbox"/>		ID ▲	Название	Описание	Цена	Вес	Категория	Активность
<input type="checkbox"/>	☰	1	Рамен острый	Лапша, мясо, перец	450	300	Готовые блюда [1]	да
<input type="checkbox"/>	☰	2	Fanta		150	500	Напитки [3]	да
<input type="checkbox"/>	☰	3	Кола	Добрый кола	100	100	Напитки [3]	да
<input type="checkbox"/>	☰	4	Спрайт	Добрый спрайт	100	100	Напитки [3]	да
<input type="checkbox"/>	☰	5	Сок "Добрый" яблочный	0.25 л, в стакане	50	25	Напитки [3]	да
<input type="checkbox"/>	☰	6	Рисовая лепешка	Рисовая мука, вода, соль, растительное масло	60	50	Готовые блюда [1]	да
<input type="checkbox"/>	☰	7	Рис пареный	Рис, вода	160	600	Готовые блюда [1]	да

Для всех



ВЗАИМОДЕЙСТВИЕ С БАЗОЙ ДАННЫХ

Bitrix ORM ядра D7

```
public static function getList(array $filter): ProductsCollection | null
{
    $items = Product::getTableEntity()::getList([
        'filter' => $filter,
        'select' => [
'ID', 'UF_NAME', 'UF_ACTIVE', 'UF_CATEGORY', 'UF_PRICE', 'UF_DESCR', 'UF_WEIGHT', 'UF_IMAGE', 'CATEGORY',
        ]
    ])->fetchCollection();

    $collection = new ProductsCollection();
    foreach($items as $item)
    {
        $collection->push(new Product($item));
    }

    return $collection;
}
```

ОБРАБОТКА АДРЕСА

GET: {API_URL}/api/catalog/:id – Получение товара

```
$api->handlerGet("$baseUrlPart/{id}", [  
    'callback' => \Project\Api\Catalog\CatalogController::getItem(...),  
    'needAuth' => true,  
    'exceptionMessage' => $api->exceptionMessage,  
]);
```

ОБРАБОТКА ЗАПРОСА

GET: {API_URL}/api/catalog/:id – Получение товара

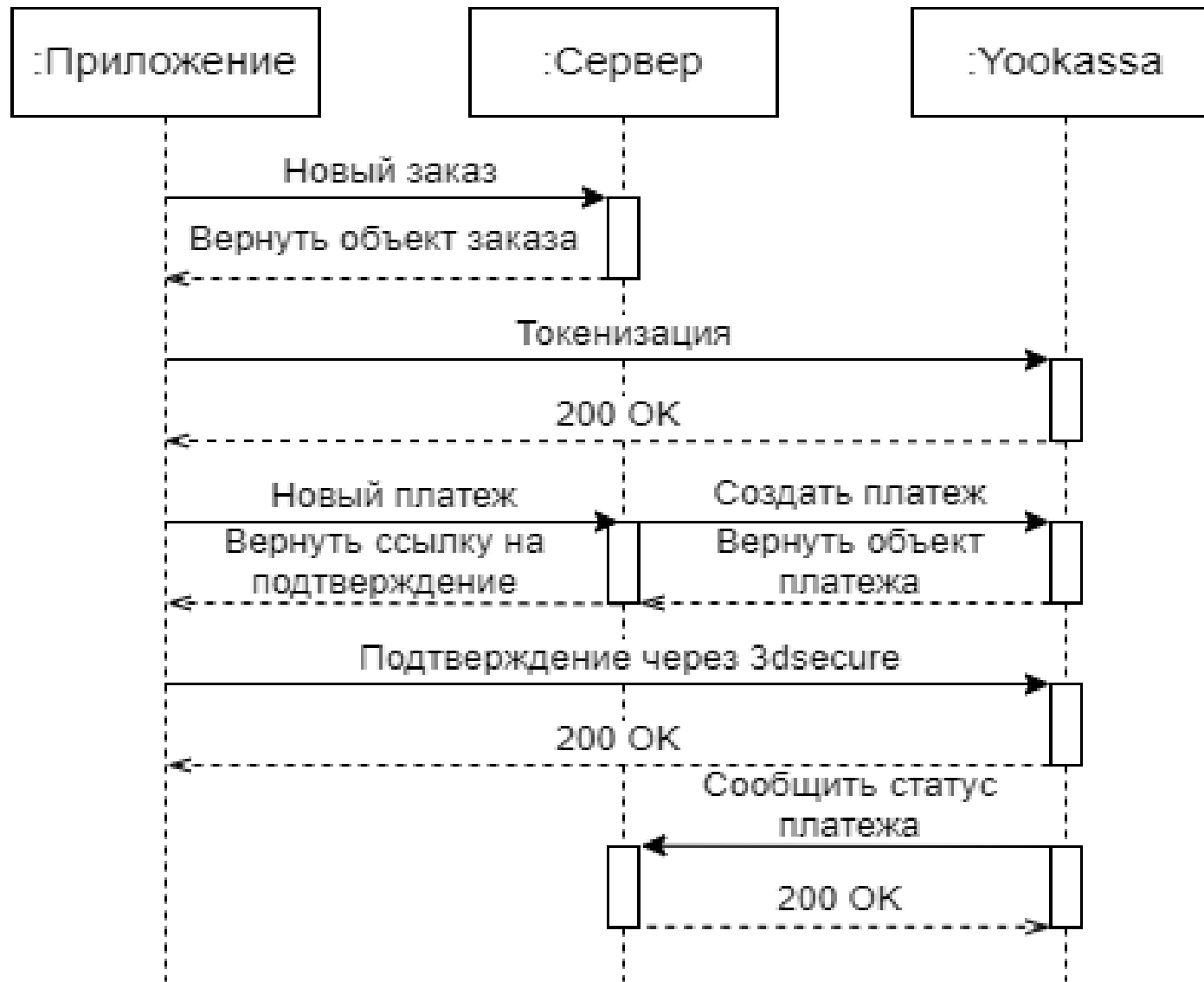
```
public static function getItem($urlParams = []): array
{
    if (empty($urlParams['id']) || !is_numeric($urlParams['id'])) {
        throw new Exception('Не указан id товара или указан в неверном формате');
    }
    $item = ProductRepository::getById($urlParams['id']);
    $image = $item->getImage();
    return [
        'id' => $item->getId(),
        'name' => $item->getName(),
        'price' => $item->getPrice(),
        'compound' => $item->getDescription(),
        'weights' => $item->getWeight(),
        'image' => $image['src'] ?? '',
    ];
}
```


ИНТЕГРАЦИИ С ВНЕШНИМИ СЕРВИСАМИ

- Сервис онлайн-оплаты Yookassa
- Сервис отправки push-уведомлений Firebase Cloud Message



ОПЛАТА ЗАКАЗА



ОТПРАВКА УВЕДОМЛЕНИЙ



ТЕСТИРОВАНИЕ ПРИЛОЖЕНИЯ

Тестирование проводилось при помощи сервиса Postman

Всего было реализовано 119 тестов. В среднем каждый запрос содержит 3-5 тестов.

Все тесты были пройдены и результаты представлены в отчете Postman

Eastern Dragon - Run results

 Ran today at 16:26:05 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	none	1	4s 974ms	119	90 ms

All Tests Passed (119) Failed (0) Skipped (0)

GET Информация о товаре

<https://vh80732.rdock.ru/api/catalog/1>

- PASS Response status code is 200
- PASS Content type is application/json
- PASS Data object has required fields
- PASS Price is a non-negative number
- PASS Message is null and success is true

ОСНОВНЫЕ РЕЗУЛЬТАТЫ

1. Проведен анализ предметной области и аналогичных проектов
2. Определены варианты использования и компоненты системы
3. Спроектирована база данных и REST API
4. Реализованы API и интеграции с сервисами онлайн-оплаты Yookassa и отправки push-уведомлений Firebase
5. Выполнено тестирование с помощью сервиса Postman