Выпускная квалификационная работа
2025

Разработка веб-приложения

Студент 4-МД-4
Згерский Марк Владимирович

Руководитель ст. преп.
Кокорин Евгений Сергеевич

КАФЕДРА ЦИФРОВЫХ
И АДДИТИВНЫХ ТЕХНОЛОГИЙ

Веб-приложение

Предмет, объект и цель исследования

Объектом исследования является процесс проектирования и реализации веб-приложения для онлайн-школы

Предмет исследования – веб-приложение для онлайн школы «Babich School».

Целью исследования является разработка веб-приложения для онлайн-школы «Babich School».

Цели и задачи

1 / Анализ и планирование

Обсуждение с заказчиком, исследование требований, сбор и анализ требований потенциальных пользователей, изучение аналогов

2 / Проектирование

Составление схем, определение используемых технологий и инструментов, дизайн пользовательского интерфейса, проектирование базы данных

3 / Разработка

Разработка клиентской части (Frontend), разработка серверной части (Backend), интеграция с серверной частью через API, оптимизация производительности

4 / Тестирование

Проверка работы всех функций платформы, устранение найденных багов, проверка производительности

5 / Развертывание

Развертывание платформы в рабочей среде, проведение финальных тестов, обучение пользователей и администраторов

Актуальность

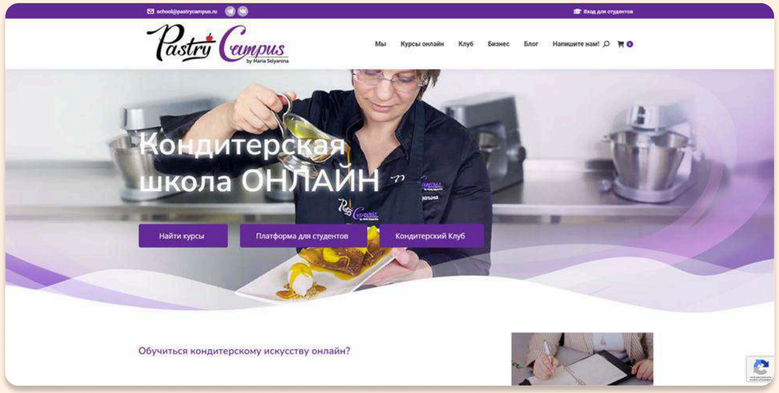
Максим Бабич

- Пекарь с многолетним опытом
- Обучался в Италии, Испании и ведущих пекарей России и стран СНГ
- Работал в ресторане Мишлен
- Бренд-шеф известных проектов «ZOE» и «Текстура»
- Консультант более 50 заведений по всему миру от России и Казахстана до Тайланда и Дубая

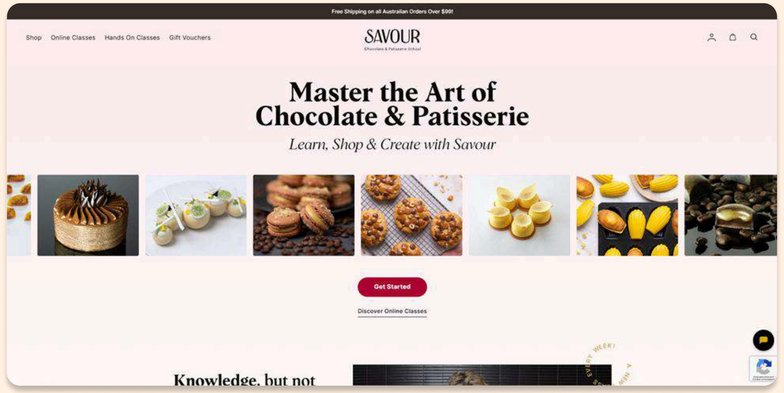
Для расширения спектра своих услуг и повышения узнаваемости личного бренда, Максим Бабич решил расположить в интернете собственную онлайн-школу. После обсуждений с ним было решено реализовать веб-приложение

Обзор аналогов

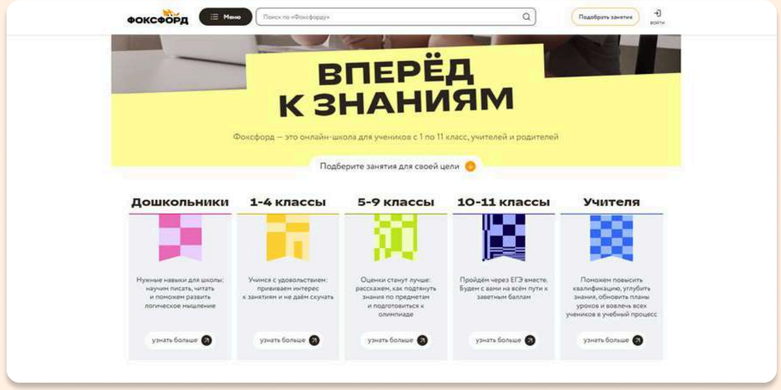
01 / Pastry Campus



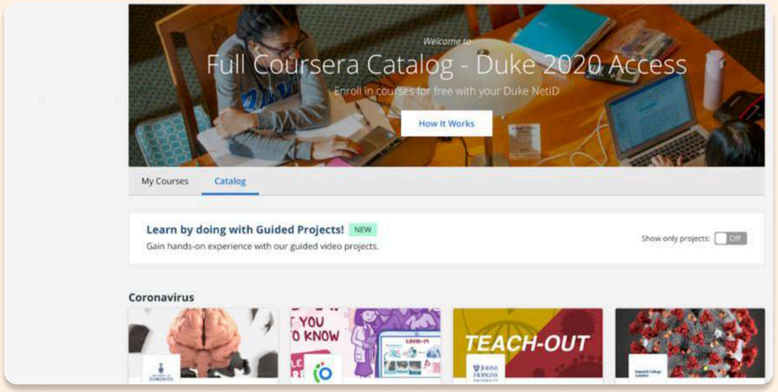
02 / Savour



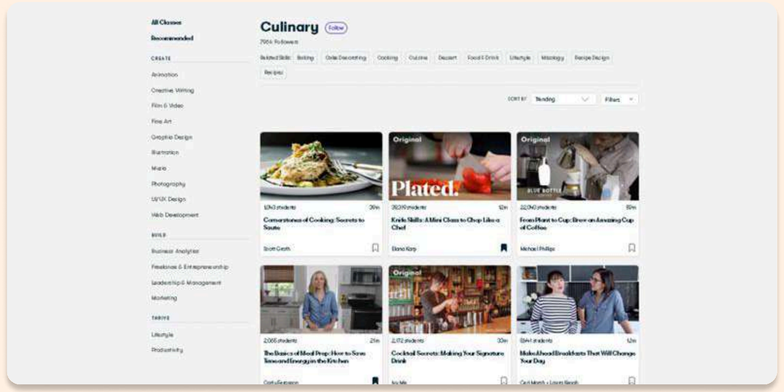
03 / Фоксфорд



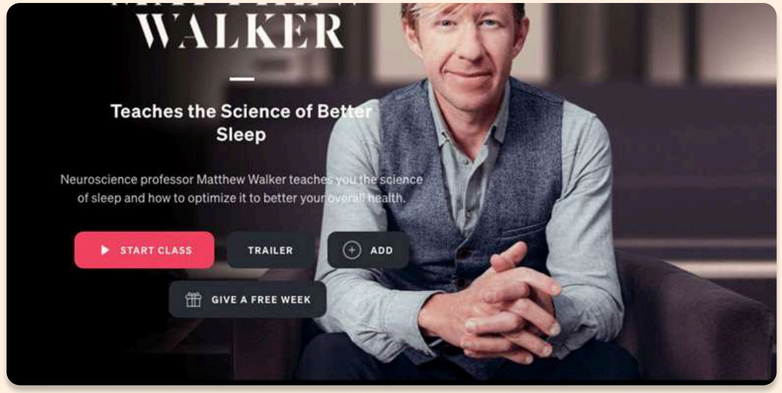
04 / Coursera



05 / SkillShare



06 / MasterClass



Обзор аналогов

Вывод

Основные выявленные проблемы рассмотренных аналогов

- В части проектов устаревший не интуитивный дизайн
- Перегруженные страницы
- Ограниченная обратная связь

Положительные стороны, которые следует учесть

- Качественный пользовательский интерфейс
- Удобная интуитивная навигация
- Адаптивный дизайн для разных устройств

Целевая аудитория

- Возраст: преимущественно 32-40 лет
- Пол: преимущественно женский (около 70%)
- География: Россия и русскоговорящие страны
- Уровень дохода: выше среднего, примерно 120-150 т.р. в месяц
- Интересы аудитории включают кулинарию, здоровое питание, саморазвитие и предпринимательство. Это люди, увлечённые выпечкой и стремящиеся совершенствовать свои навыки — от начинающих кондитеров, осваивающих основы, до профессионалов, желающих повысить квалификацию.
  Среди них — любители домашней выпечки всех возрастов, домохозяйки, экспериментирующие на кухне, и те, кто рассматривает кулинарию как возможность для собственного дела или дополнительного дохода.

Выбор программного обеспечения

HTML 5

Язык разметки,
для расположения информации
на сайте

CSS 3

Дает возможность
стилизовать веб-приложение

TypeScript

Типизированный язык
для создания динамических
страниц

Vue.js 3

Фреймворк, позволяющий
создавать реактивные страницы
без перезагрузок

PostgreSQL

Реляционная база данных
для хранения информации

Java

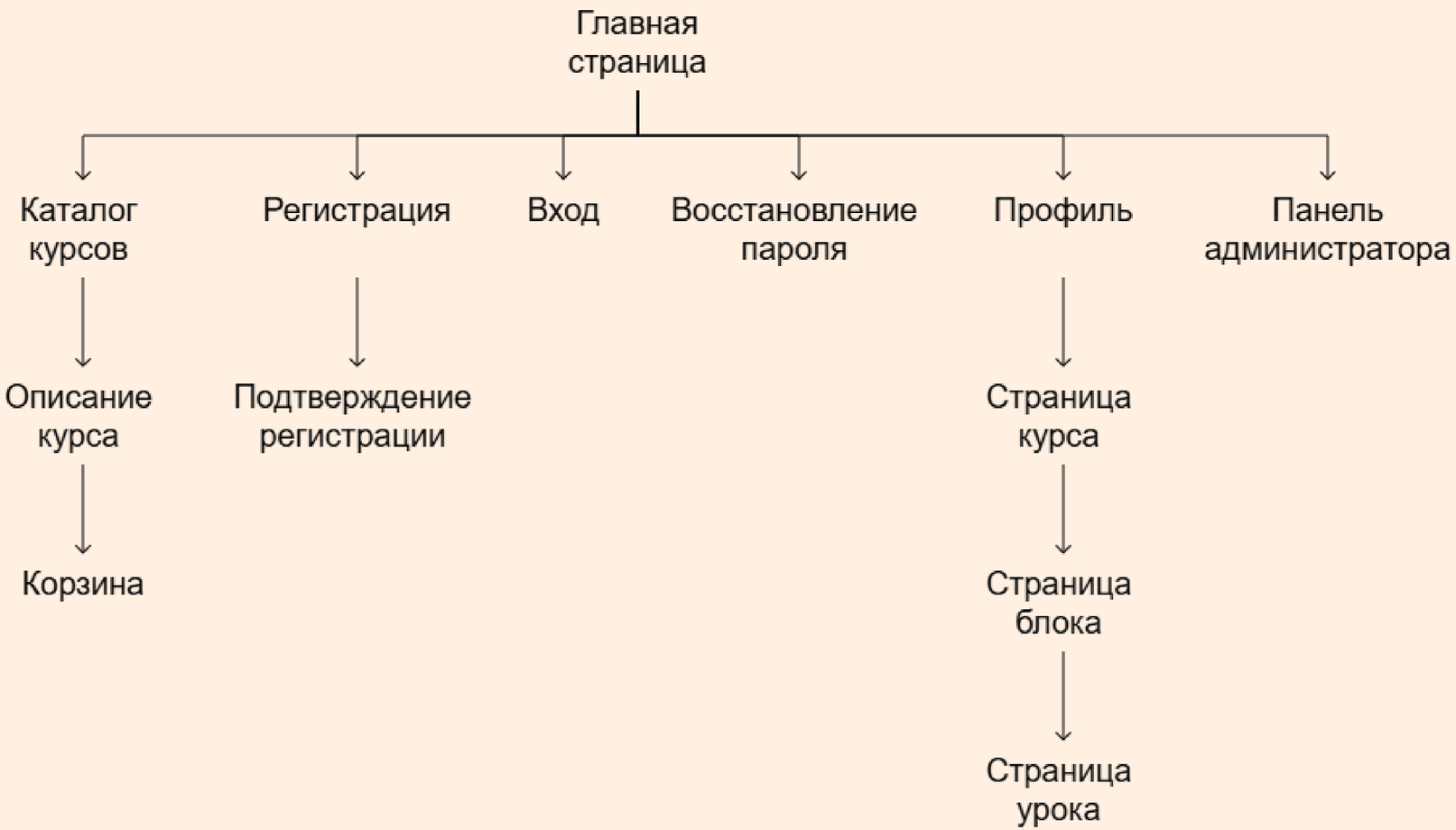
Язык программирования
для бэкенд разработки

Spring

Фреймворк, который упрощает
процесс бэкенд разработки

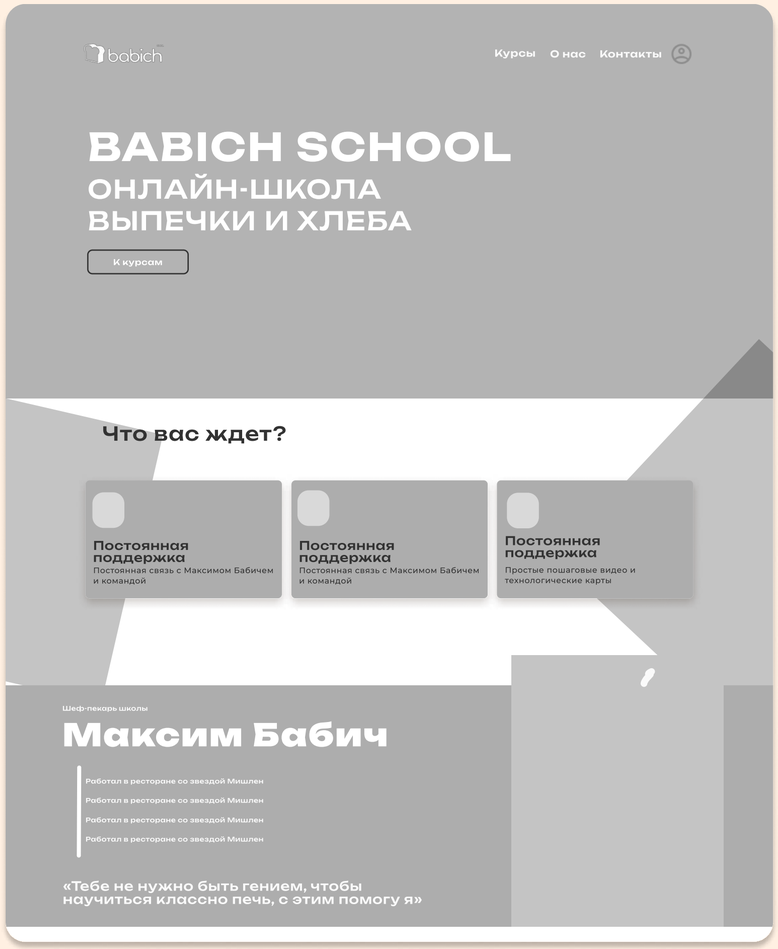
Разработка user-flow

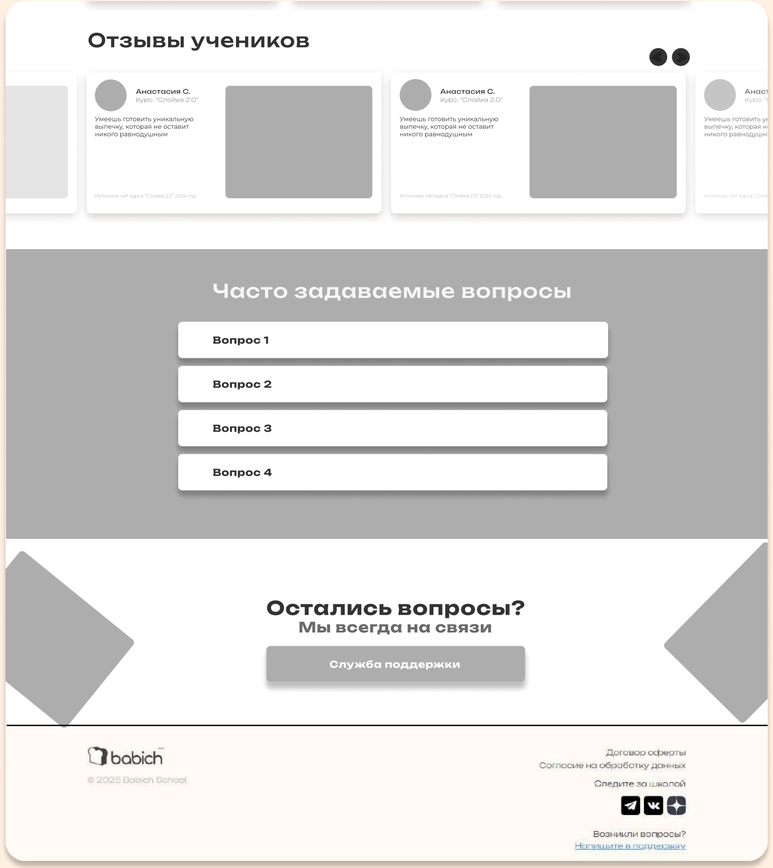
Структура веб-приложения



Разработка Wireframe

Десктопная версия главной



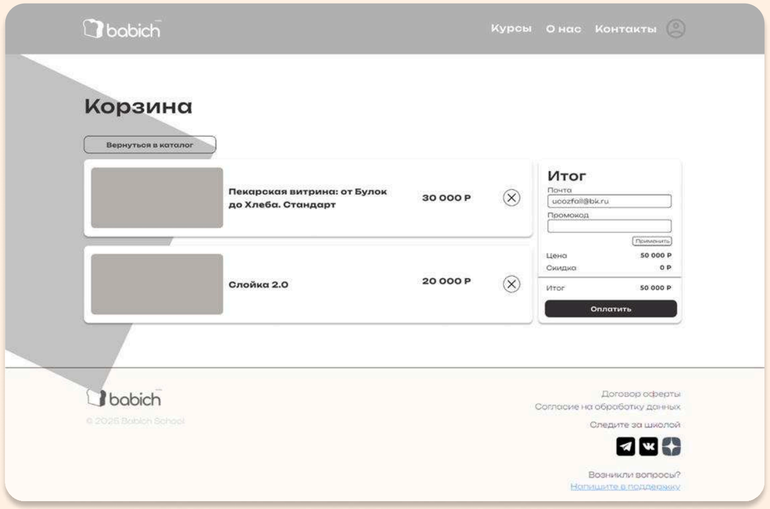


Разработка Wireframe

Мобильная версия главной

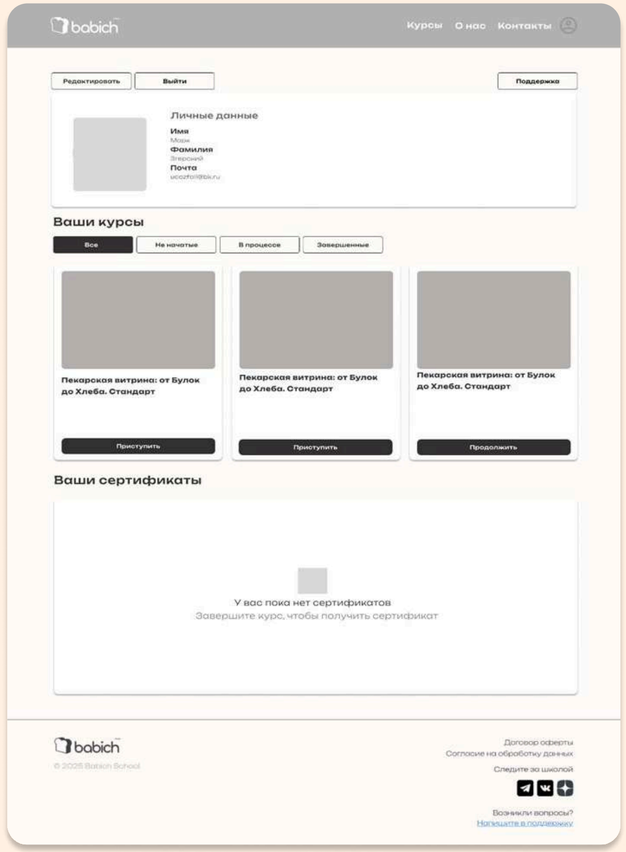
Разработка Wireframe

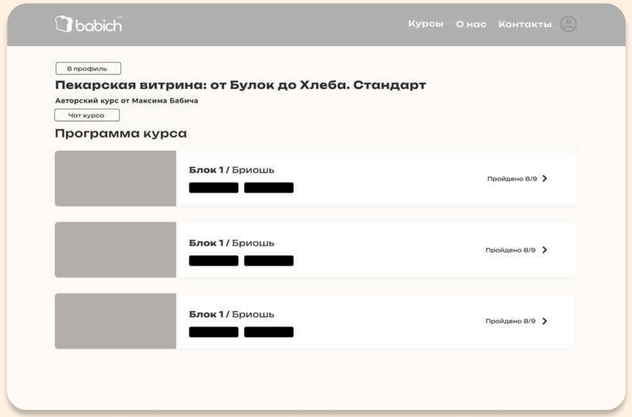
Каталог / Корзина / Вход

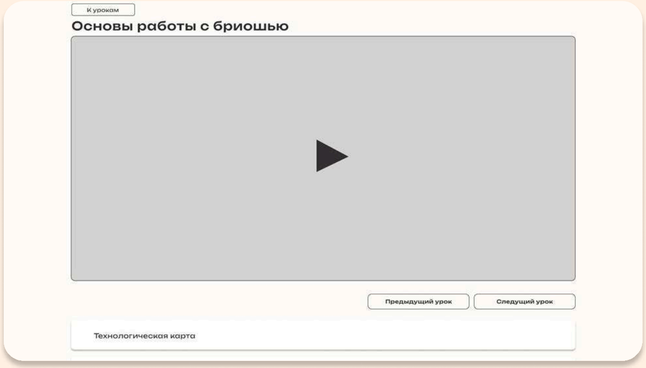


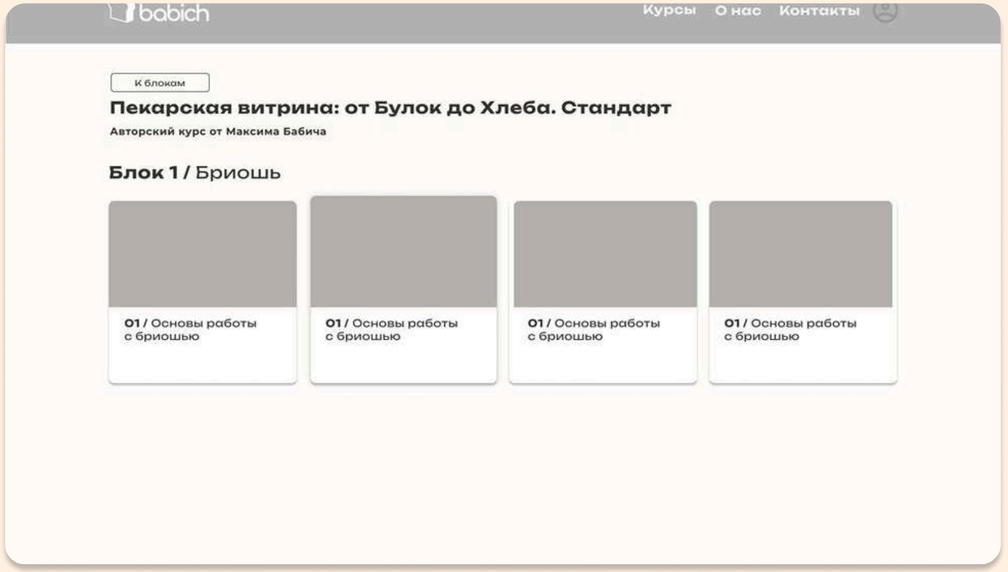
Разработка Wireframe

Профиль / Курс / Блок / Урок









UI-кит

Логотип

babich SCHOOL

Кнопки

Дефолт | При наведении | Нажатие

Купить | Купить | Купить

Подробнее | Подробнее | Подробнее

Все курсы | Все курсы | Все курсы

Дефолт | При наведении

Цвета

Основная палитра

#FF8B19 | #FFF8F0 | #333132

Дополнительные цвета

#C4DB87 | #BA1F22 | #5D81DE

#FED85D

Аватар

Без фотографии

Профиль

Header

С фотографией

Профиль

Header

Поля ввода

Дефолт

Логин

Активное

Логин

mail@example.com

Иконки

Видимо | Скрыто | Редактировать | Удалить

Стиль текста

Unbounded

Bold 32px

Заголовок <h1>

Medium 30px

Заголовок <h2>

Medium 24px

Заголовок <h3>

Medium 20px

Заголовок <h4>

Regular 16px

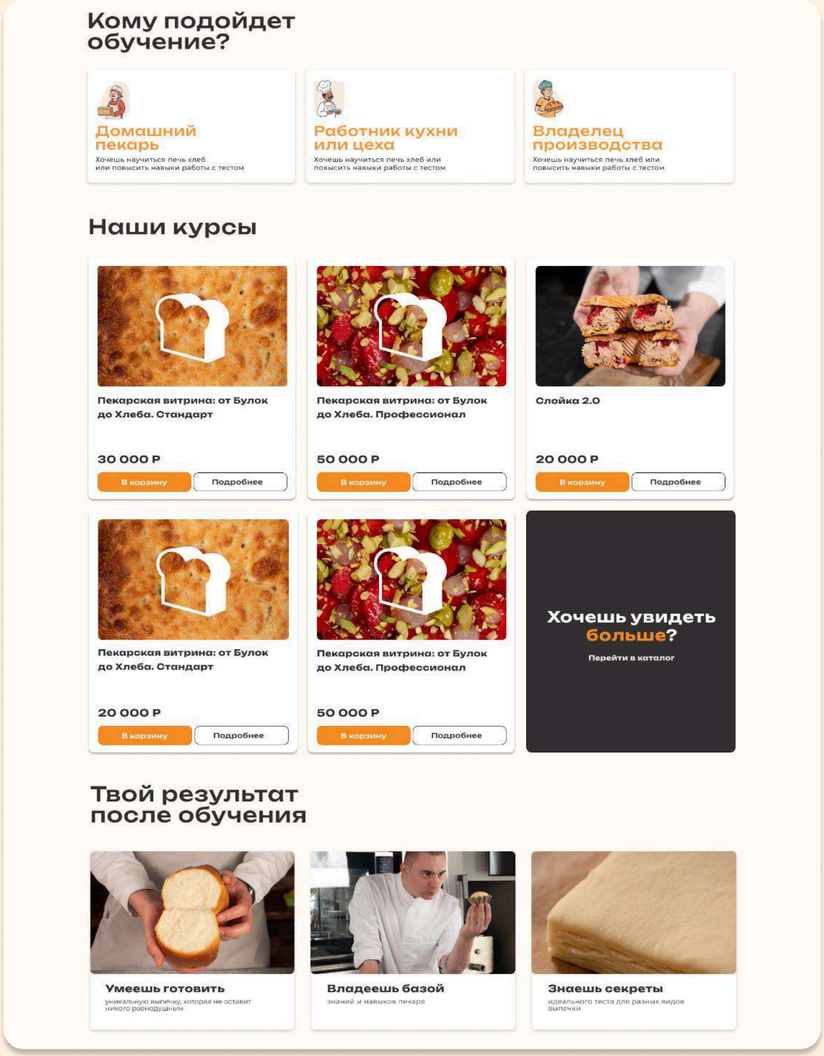
Текст <p>

Light 20px

Текст для действий <button>

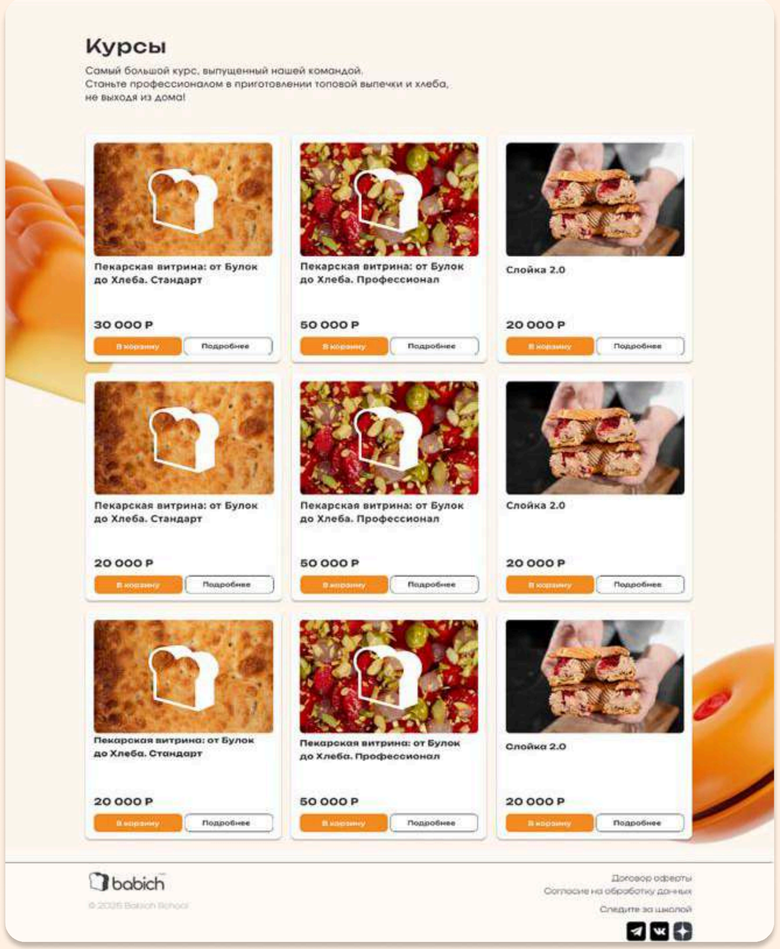
Прототипы

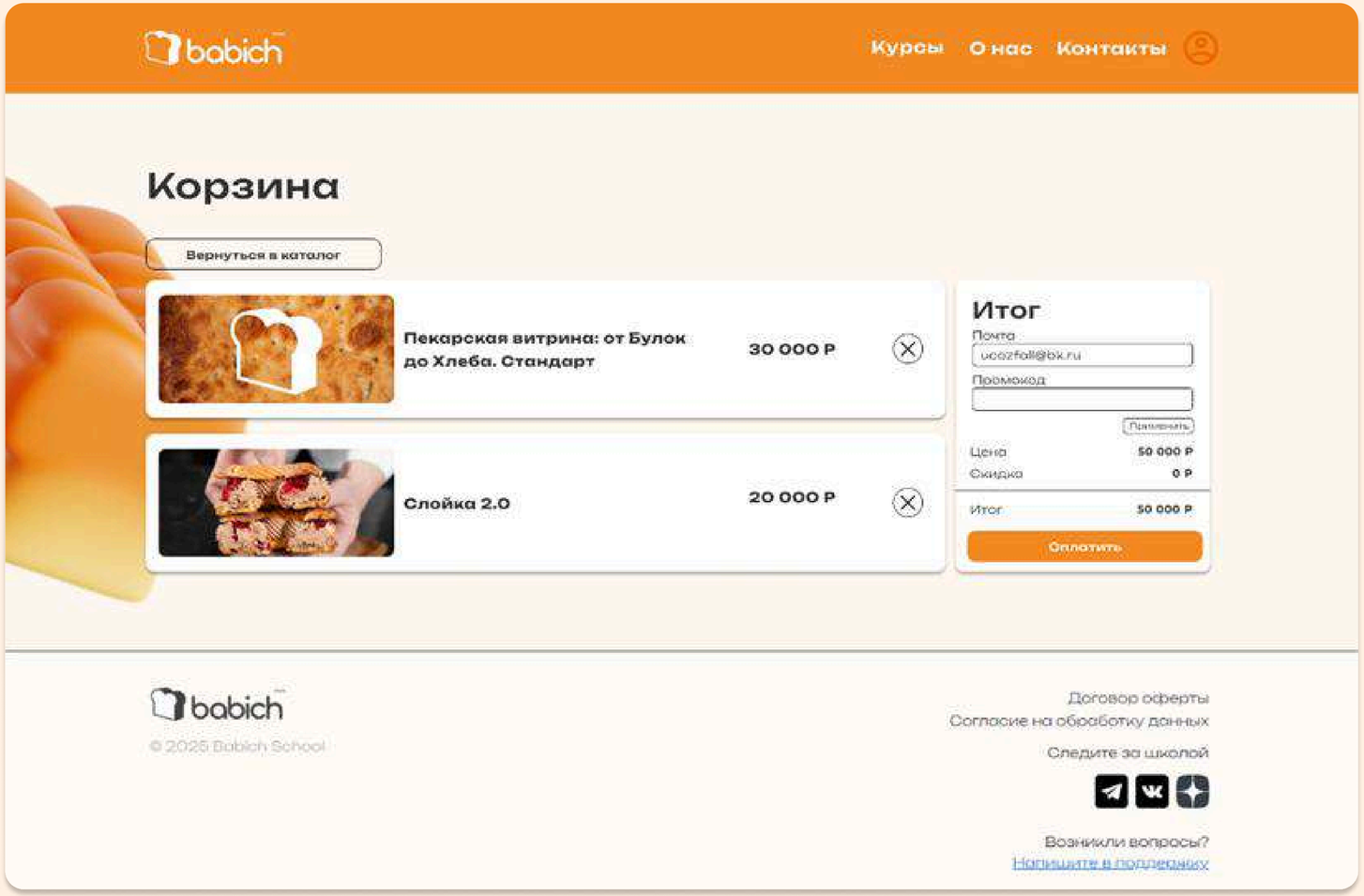
Главная



Прототипы

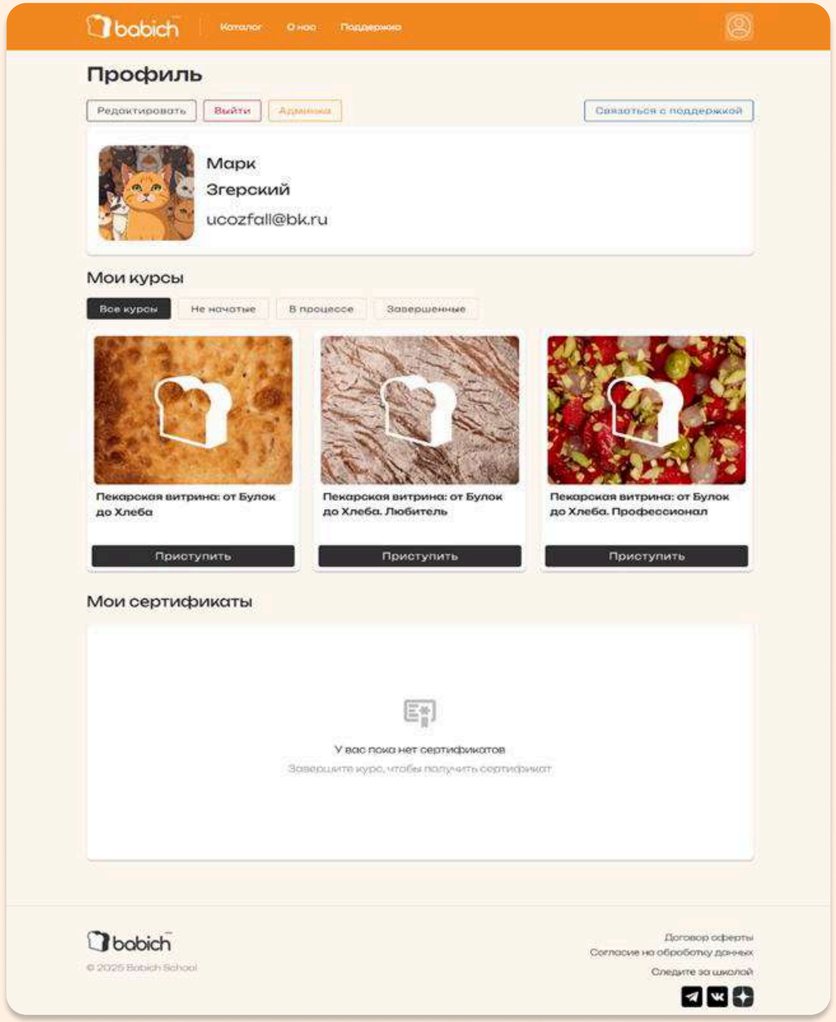
Каталог / Корзина

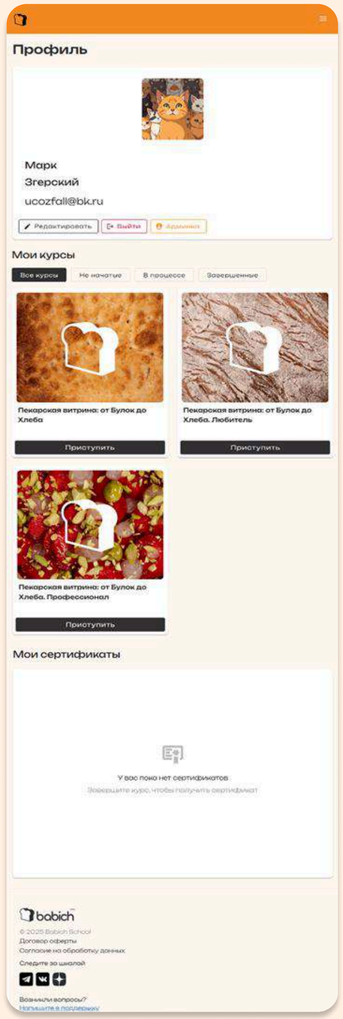




Прототипы

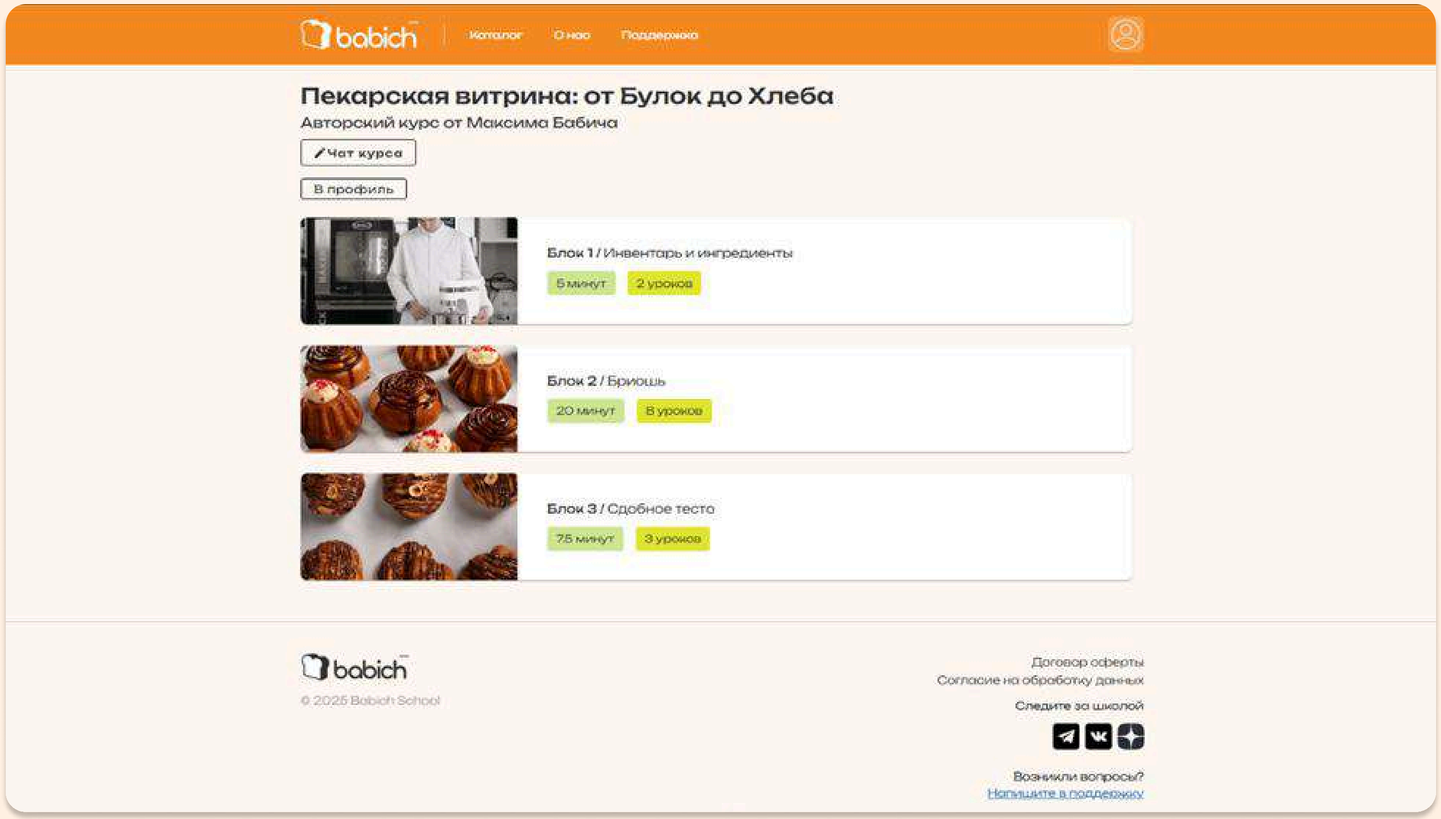
Профиль десктоп / телефон / планшет

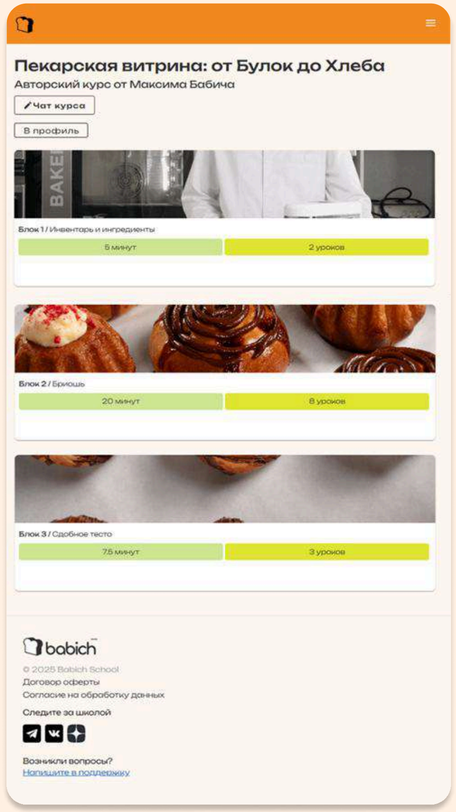




Прототипы

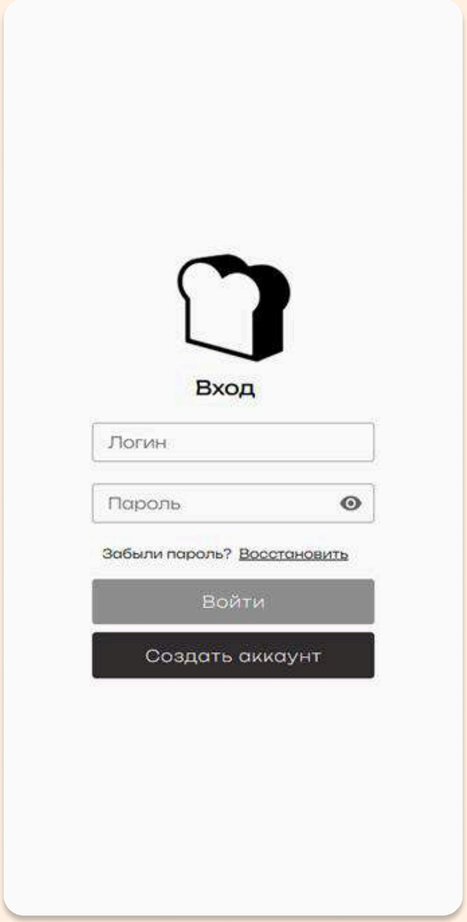
Курс десктоп / телефон / планшет





Разработка

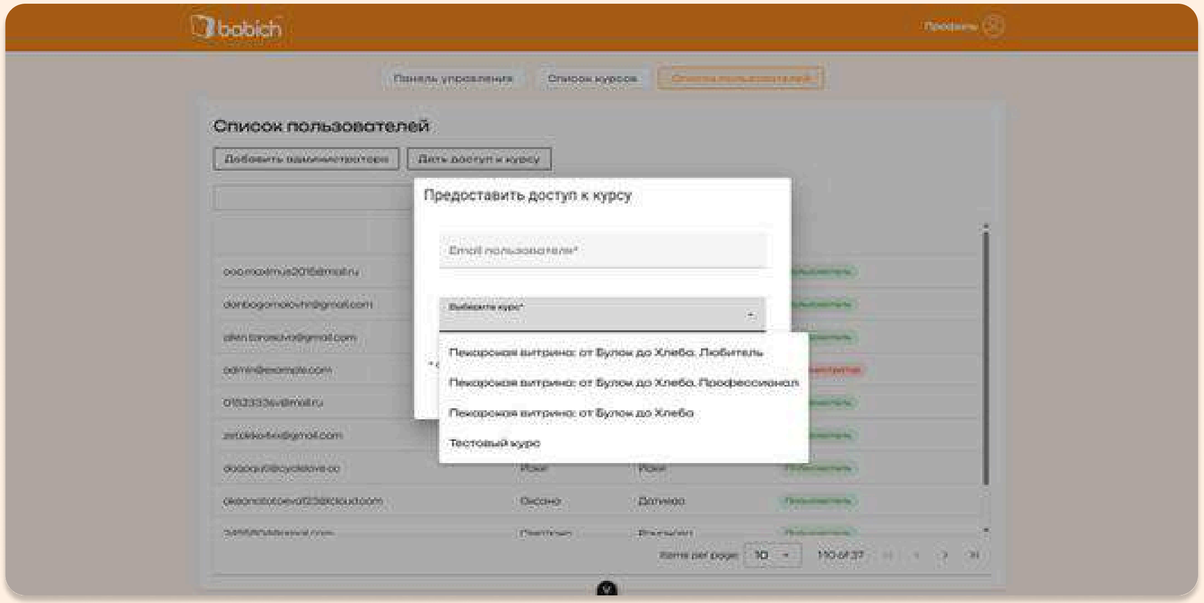
Страница входа / Техническая реализация

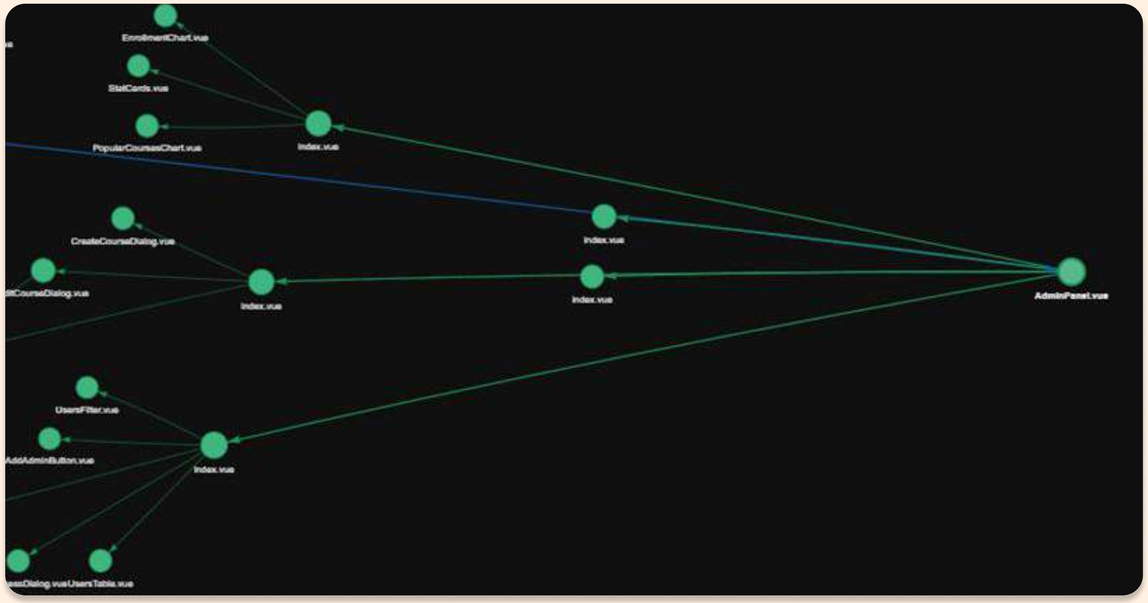



```
<script lang="ts" setup>
const handleLogin = async () => {
  if (!validateForm()) return

  try {
    loading.value = true
    serverError.value = ''
    formErrors.email = ''
    formErrors.password = ''

    const result = await AuthService.login(formData.email, formData.password)

    const redirectUrl = safeGetItem('redirectAfterLogin')
    safeRemoveItem('redirectAfterLogin')

    if (redirectUrl) {
      if (redirectUrl.includes('/admin') && !AuthService.isAdmin()) {
        router.push('/lk')
        safeSetItem('accessError', 'Доступ к административной панели запрещен')
      } else {
        router.push(redirectUrl)
      }
    } else {
      if (AuthService.isAdmin()) {
        router.push('/admin')
      } else {
        router.push('/lk')
      }
    }
  } catch (error: any) {
    if (error.response) {
      if (error.response.status === 401) {
        serverError.value = 'Неверный логин или пароль'
      } else if (error.response.status === 403) {
        serverError.value = 'Доступ запрещен'
      } else if (error.response.data?.message) {
        serverError.value = error.response.data.message
```

Разработка

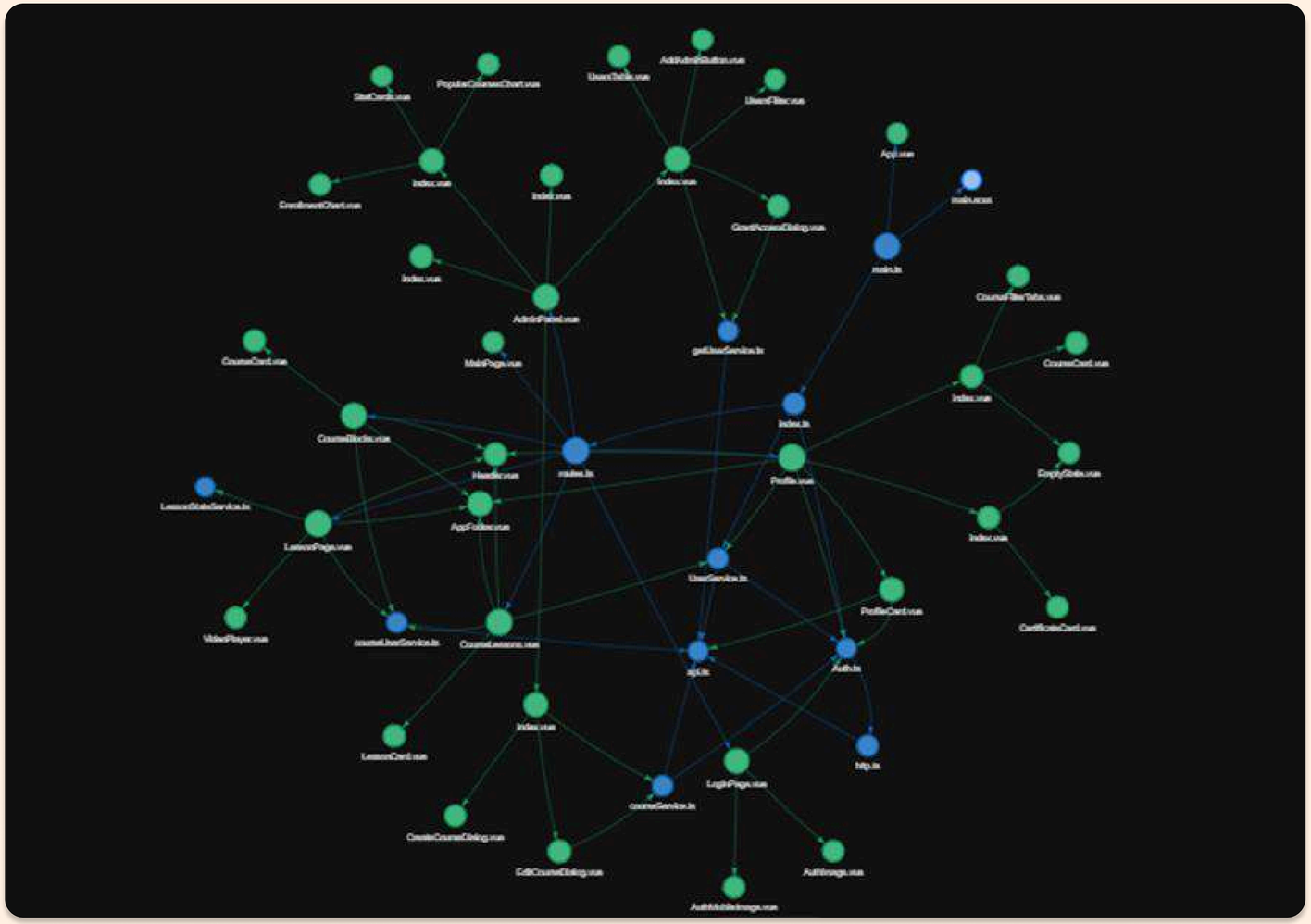
Панель администратора / Техническая реализация





Разработка

Структура компонентов в проекте

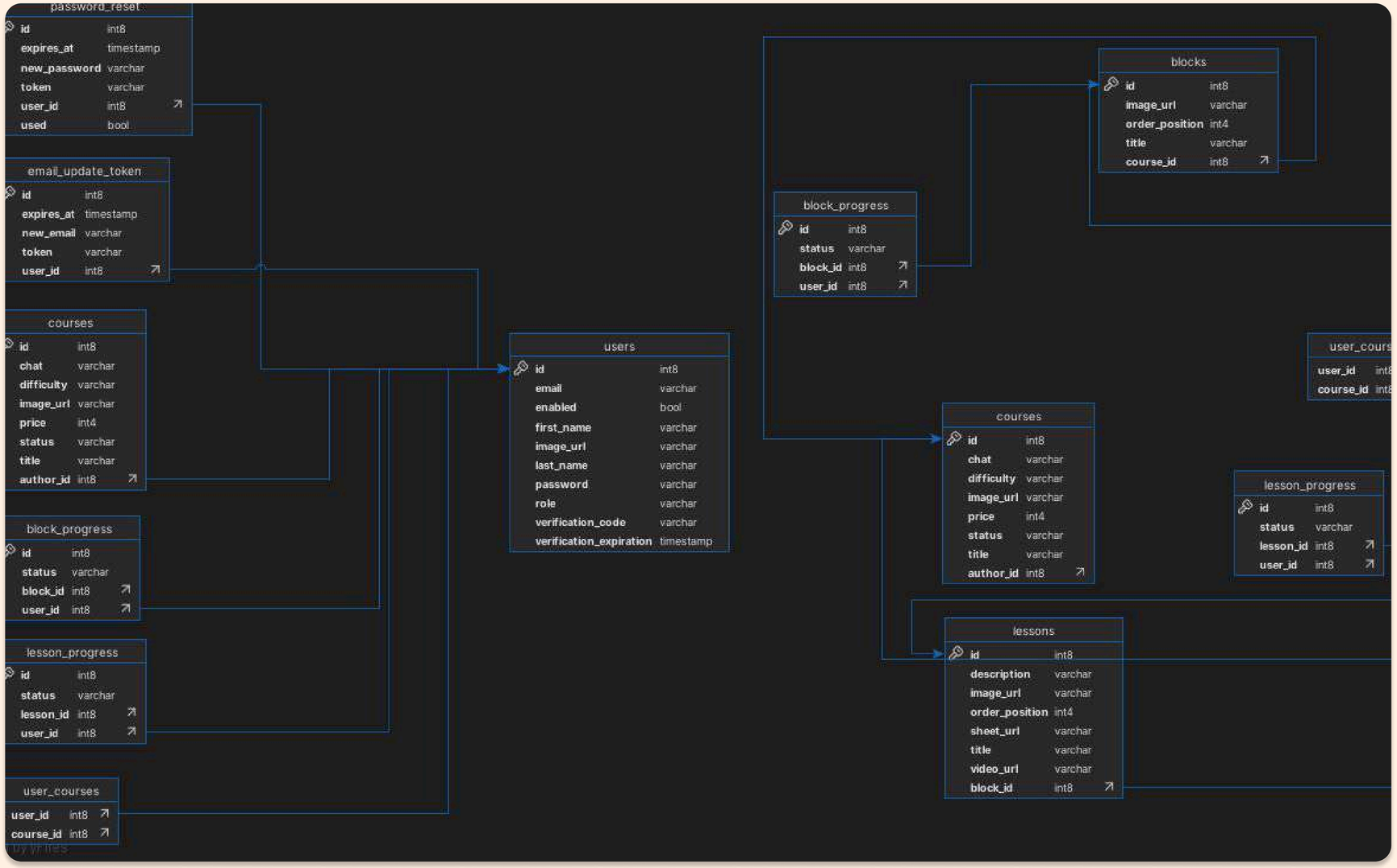


Итоговая струкутра проекта. Суммарно для реализации клиентской части было создано 211 компонентов

SSR to full load	629ms
Page load	414ms
Navigation	119ms

Разработка

Схема базы данных веб-приложения



Разработка

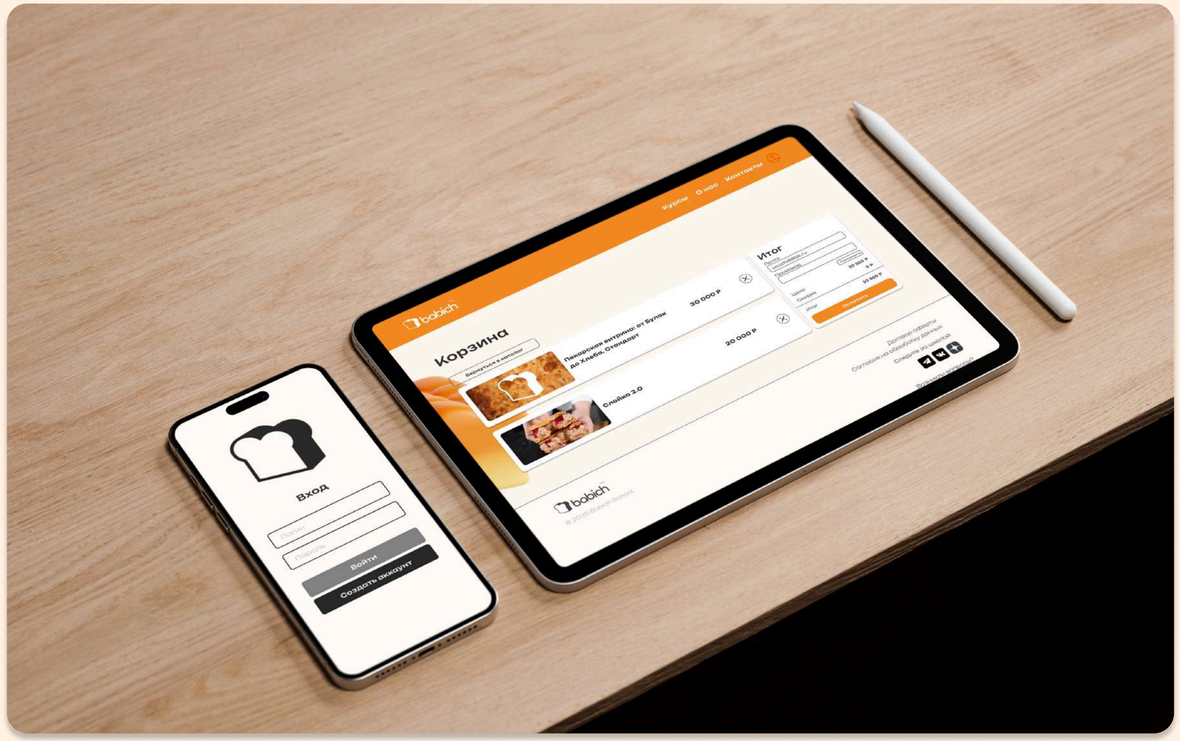
Бэкенд онлайн-школы

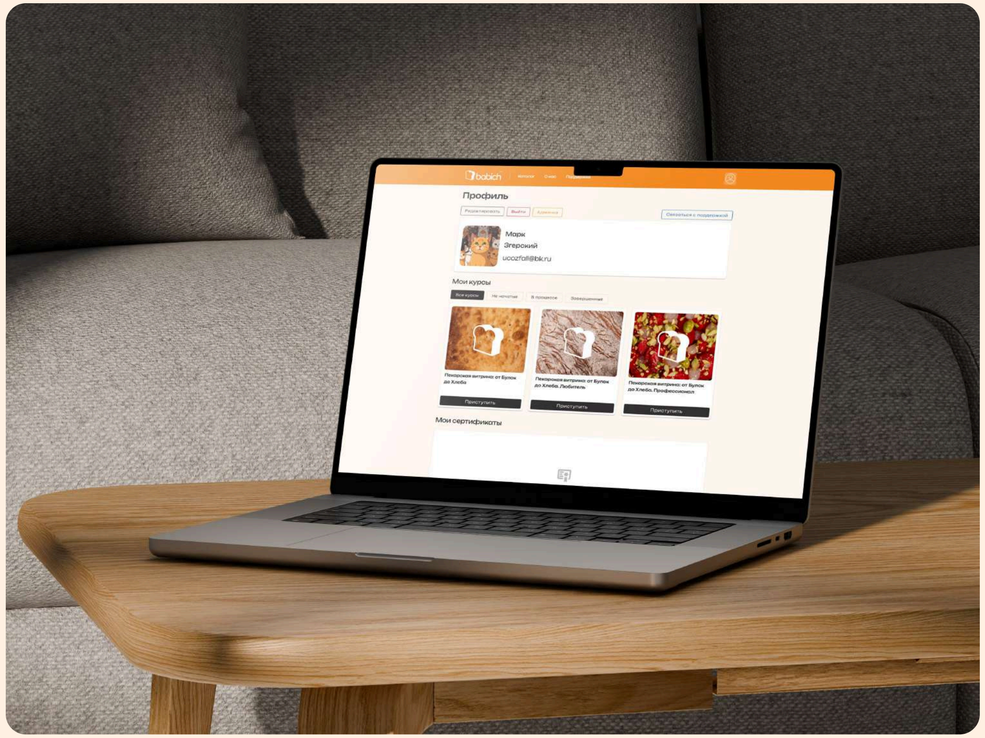

```java
import java.io.IOException;

@Component
public class JwtAuthenticationFilter extends OncePerRequestFilter {
    private final HandlerExceptionResolver handlerExceptionResolver;

    private final JwtService jwtService;
    private final UserDetailsService userDetailsService;

    public JwtAuthenticationFilter(
            JwtService jwtService,
            UserDetailsService userDetailsService,
            HandlerExceptionResolver handlerExceptionResolver
    ) {
        this.jwtService = jwtService;
        this.userDetailsService = userDetailsService;
        this.handlerExceptionResolver = handlerExceptionResolver;
    }

    @Override
    protected void doFilterInternal(
            @NonNull HttpServletRequest request,
            @NonNull HttpServletResponse response,
            @NonNull FilterChain filterChain
```

```java
@RequestMapping("/admin")
@RequiredArgsConstructor
public class AdminController {

    private final AdminService adminService;

    // === Назначение курса пользователю ===
    @PostMapping("/users/{email}/courses/{courseId}")
    public ResponseEntity<Void> assignCourseToUser(
            @PathVariable String email,
            @PathVariable Long courseId) {
        adminService.assignCourseToUser(email, courseId);
        return ResponseEntity.status(HttpStatus.NO_CONTENT).build();
    }

    // === Удаление курса у пользователя ===
    @DeleteMapping("/users/{email}/courses/{courseId}")
    public ResponseEntity<Void> removeCourseFromUser(
            @PathVariable String email,
            @PathVariable Long courseId) {
        adminService.removeCourseFromUser(email, courseId);
        return ResponseEntity.status(HttpStatus.NO_CONTENT).build();
```

```java
// Логика аутентификации и получения JWT токенов
@PostMapping("/login")
public ResponseEntity<LoginResponse> authenticate(@RequestBody LoginUserDto loginUserDto){
    LoginResponse loginResponse = authenticationService.authenticate(loginUserDto);
    return ResponseEntity.ok(loginResponse);
}

@Operation(summary = "Обновление access-токена", responses = {
        @ApiResponse(responseCode = "200", description = "Токен успешно обновлен"),
        @ApiResponse(responseCode = "401", description = "Ошибка валидации токена")
})
// Логика получения нового access токена по refresh токену
@PostMapping("/refresh")
public ResponseEntity<String> refreshAccessToken(@RequestBody RefreshToken refreshToken){
    String newAccessToken = authenticationService.refreshAccessToken(refreshToken.getToken());
    return ResponseEntity.ok(newAccessToken);
}

@Operation(summary = "Подтверждение аккаунта пользователя", responses = {
        @ApiResponse(responseCode = "200", description = "Аккаунт успешно подтвержден"),
        @ApiResponse(responseCode = "400", description = "Неверные данные для подтверждения")
})
```

Итоговая реализация





Заключение

В рамках ВКР было разработано современное веб-приложение для онлайн-школы выпечки "Babich School", направленное на повышение эффективности обучения.

Проект реализован с учётом потребностей целевой аудитории, включает удобный адаптивный интерфейс и необходимый функционал для онлайн-обучения. Решение готово к практическому использованию, имеет потенциал для масштабирования и дальнейшего развития платформы.

Спасибо
за внимание