

УДК 004.738.5

ПРОЕКТИРОВАНИЕ ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ ЛИЧНОГО ТАЙМ-МЕНЕДЖМЕНТА

Домбровский Я.А.

старший преподаватель

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Терехов Я.И.

студент

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Комаров К.А.

магистрант

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Аннотация.

В статье рассмотрены принципы проектирования веб-приложения для личного тайм-менеджмента, ориентированного на планирование задач, приоритезацию, напоминания и анализ активности пользователя. Актуальность определяется ростом потребности в персональных цифровых инструментах самоорганизации и необходимостью учитывать требования к удобству использования, кроссплатформенности, безопасности обработки данных и доступности интерфейсов. На основе сравнительного анализа популярных решений (Todoist, Trello, Notion, TickTick) сформулированы функциональные требования. Предложен подход к проектированию, объединяющий функциональное моделирование IDEF0 и UML-проектирование, что обеспечивает формализацию

границ системы, пользовательских сценариев и структуры данных. Сформирована концептуальная модель предметной области, спроектирована структура базы данных и разработан прототип пользовательского интерфейса. Рассмотрены организационно-технические меры защиты информации и нормативные требования, релевантные веб-приложениям, обрабатывающим персональные данные. Полученные результаты формируют основу для дальнейшей реализации и апробации приложения.

Ключевые слова: тайм-менеджмент, веб-приложение, SPA, PWA, IDEF0, UML, база данных, UX/UI, доступность, информационная безопасность.

DESIGN OF A WEB APPLICATION FOR PERSONAL TIME MANAGEMENT

Dombrovsky Y.A.

Senior Lecturer

*Kaluga State University named after K.E. Tsiolkovsky,
Kaluga, Russia*

Terekhov Y.I.

Student

*Kaluga State University named after K.E. Tsiolkovsky,
Kaluga, Russia*

Komarov K.A.

Master's student

*Kaluga State University named after K.E. Tsiolkovsky,
Kaluga, Russia*

Abstract.

The paper discusses the design of a web application for personal time management focused on task planning, prioritization, reminders, and activity analytics. The relevance is driven by the growing demand for personal digital self-organization tools and the need to address usability, cross-platform delivery, information security, and accessibility requirements. Based on a comparative analysis of widely used solutions (Todoist, Trello, Notion, TickTick), the functional requirements were formulated. A design approach that combines IDEF0 functional modeling and UML modeling is proposed to formalize system boundaries, user scenarios, and the data model. A conceptual domain model, database schema, and a user interface prototype were developed. Regulatory and organizational-technical measures relevant to personal data processing in web applications are also considered. The results provide a basis for implementation and further evaluation of the application.

Keywords: time management, web application, SPA, PWA, IDEF0, UML, database, UX/UI, accessibility, information security.

Задачи личного планирования (ведение списка дел, контроль сроков, расстановка приоритетов, фиксация выполненных действий) в настоящее время решаются преимущественно с использованием цифровых сервисов. Однако массовые решения нередко ориентированы на командную работу, включают избыточные функции или предъявляют повышенные требования к освоению интерфейса, что снижает их эффективность в индивидуальных сценариях. В связи с этим актуальной является разработка веб-приложения для персонального тайм-менеджмента с акцентом на простоту интерфейса, кроссплатформенность и корректную организацию обработки пользовательских данных [4,13].

Проектирование выполнено на основе сочетания методов сравнительного анализа, функционального моделирования и объектно-ориентированного проектирования. На первом этапе рассмотрены архитектурные подходы к веб-

приложениям и обоснован выбор SPA как базовой модели пользовательского интерфейса с возможностью расширения функциональности за счёт элементов PWA (кэширование, офлайн-режим, push-уведомления).

На втором этапе выполнен сравнительный анализ популярных решений для личного и командного планирования (Todoist, Trello, Notion, TickTick) по критериям: удобство ведения задач и проектов, качество напоминаний, наличие аналитики, сложность интерфейса и кроссплатформенность. Результаты анализа использованы для уточнения функциональных требований и пользовательских сценариев.

Для определения границ системы и этапов её разработки построены модели IDEF0: контекстная диаграмма А-0 и диаграмма декомпозиции А0. Для детализации сценариев и структуры данных применено UML-моделирование (диаграммы вариантов использования и классов) [3].

При проектировании учитывались требования к кроссбраузерной совместимости, а также нормативные требования и рекомендации в области защиты информации и доступности интернет-ресурсов [2].

Разрабатываемое приложение предназначено для индивидуального использования и включает: регистрацию и аутентификацию, создание и редактирование задач, назначение сроков и приоритетов, группировку по категориям/проектам, настройку напоминаний, просмотр расписания и журнала действий, а также вывод агрегированных показателей активности.

Выбор SPA обеспечивает отзывчивость интерфейса при частых операциях с задачами, а элементы PWA рассматриваются как механизм повышения автономности и доставки уведомлений.

Контекстная диаграмма IDEF0 (А-0) фиксирует назначение системы и основные потоки: входные данные (потребности пользователей, пользовательские сценарии, данные о задачах и расписаниях) преобразуются в выходные результаты (интерфейс планирования, отчёты об активности,

функционал напоминаний). Управляющими воздействиями выступают рекомендации по UI/UX, требования безопасности и нормативные требования к обработке данных, а механизмами – средства разработки и инфраструктура хранения данных (рисунок 1).

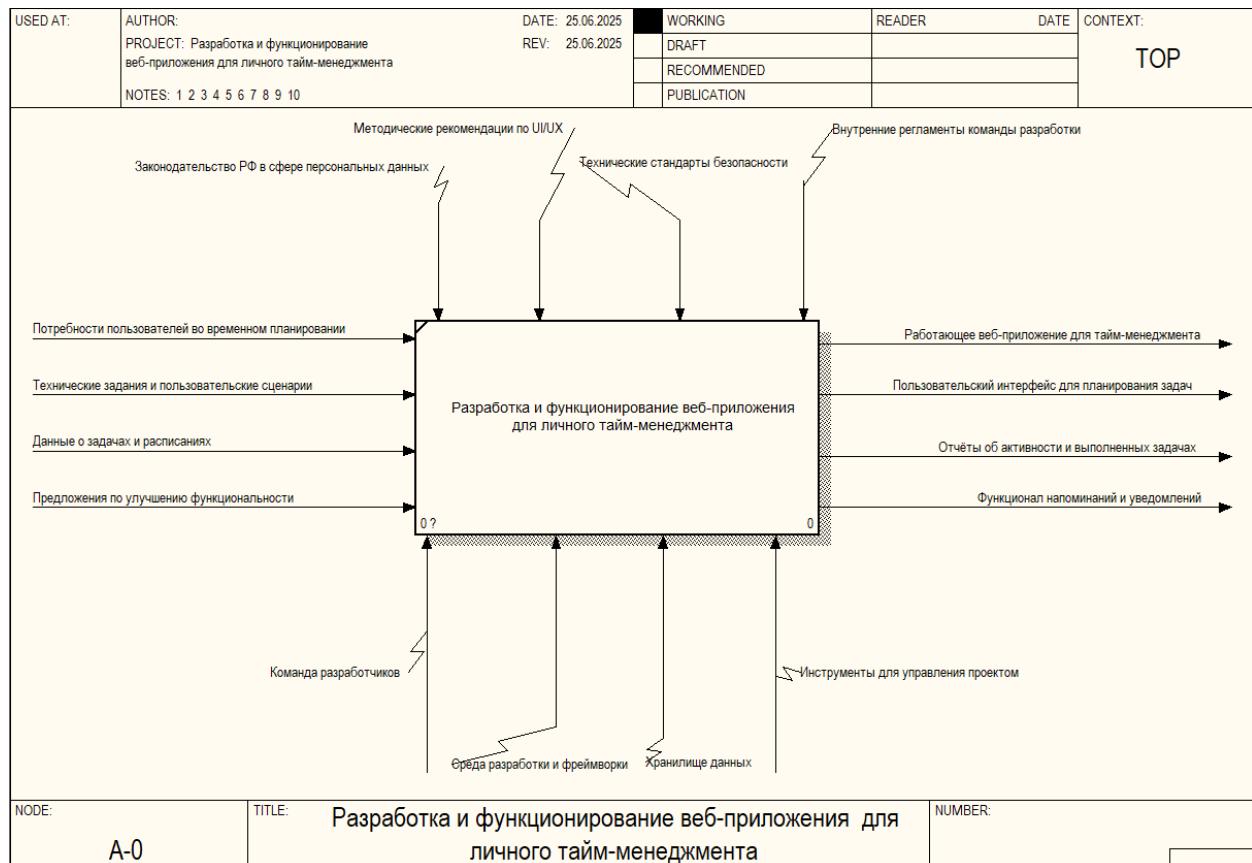


Рис. 1 – Контекстная диаграмма IDEF0 (A-0) системы личного тайм-менеджмента (составлено авторами)

Диаграмма декомпозиции (A0) отражает основные этапы жизненного цикла решения: сбор и анализ требований, проектирование архитектуры, разработка пользовательского интерфейса, реализация серверной и клиентской логики, развёртывание и техническая поддержка (рисунок 2).

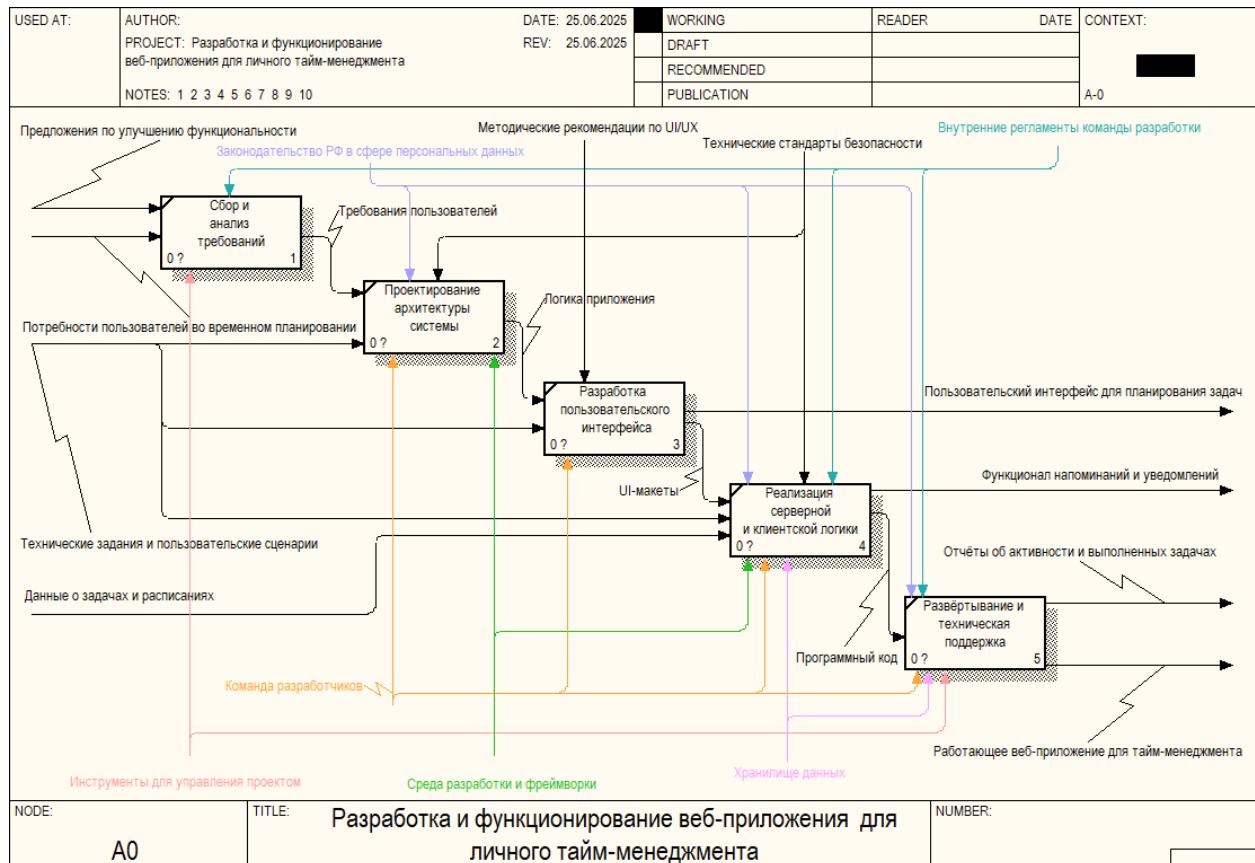


Рис. 2 – Диаграмма декомпозиции IDEF0 (A0) проектирования и разработки приложения (составлено авторами)

UML-диаграмма вариантов использования описывает ключевые действия пользователя: регистрация и аутентификация, операции с задачами, настройка напоминаний, просмотр расписания и журнала действий. Диаграмма используется для согласования функциональных требований и уточнения пользовательских сценариев (рисунок 3).

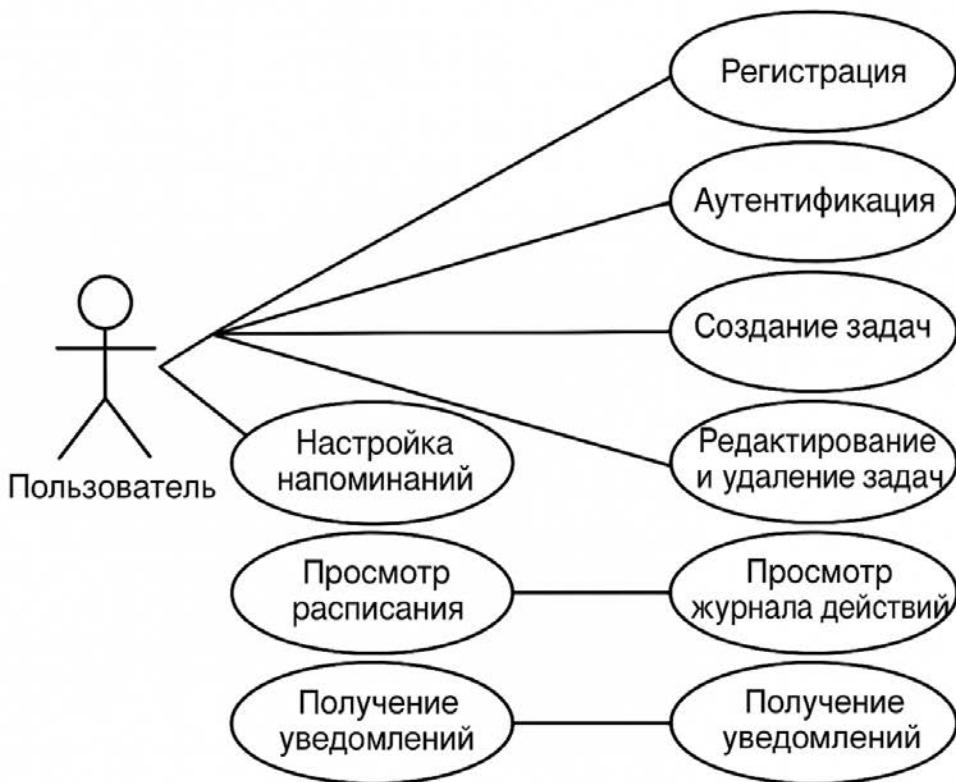


Рис. 3 – UML-диаграмма вариантов использования приложения
(составлено авторами)

UML-диаграмма классов отражает сущности предметной области: User, Task, Category, Reminder и ActivityLog (рисунок 4). Связи между сущностями обеспечивают хранение задач пользователя, группировку по категориям, постановку напоминаний и фиксацию событий в журнале активности, что является основой для проектирования базы данных и интерфейсов взаимодействия [6].

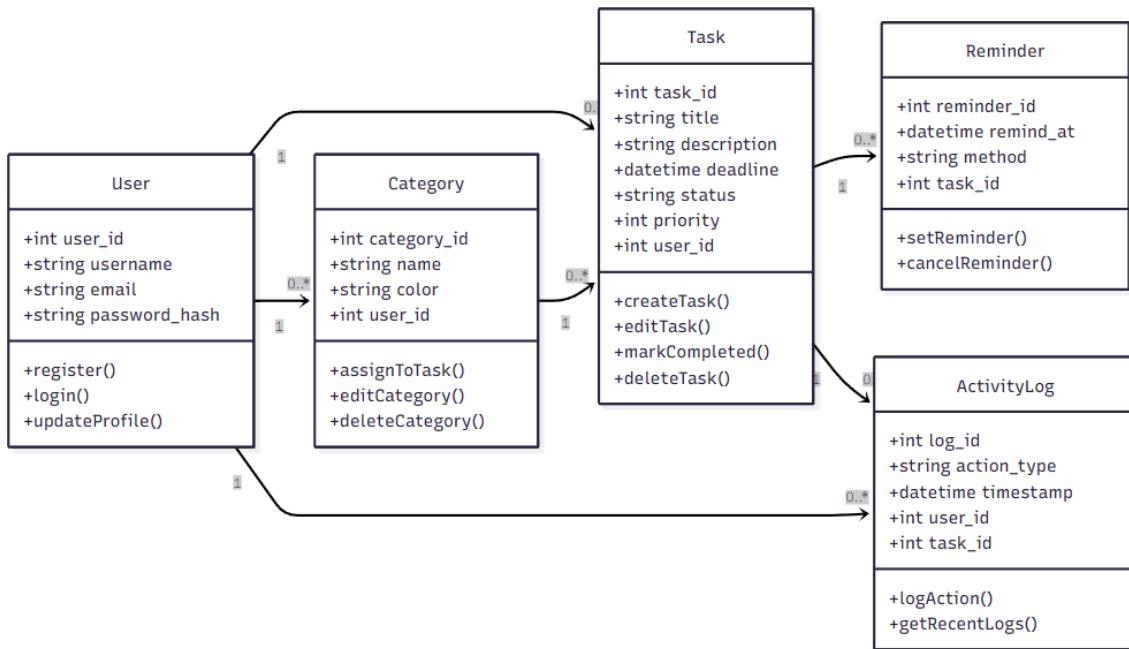


Рис. 4 – UML-диаграмма классов (концептуальная модель данных)
(составлено авторами)

Структура базы данных реализует хранение пользователей, задач, категорий, напоминаний и журнала действий (рисунок 5). При проектировании учитываются практики работы с реляционными СУБД и принципы нормализации данных [1,5].

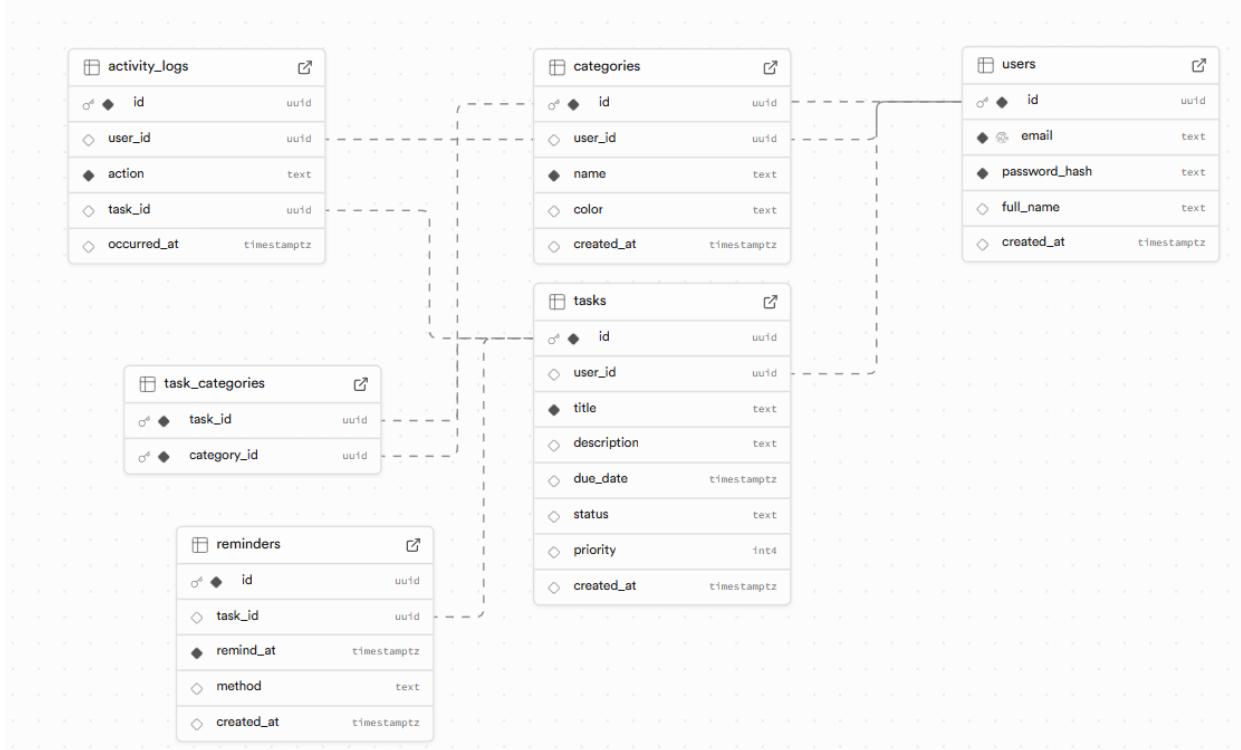


Рис. 5 – Схема базы данных приложения (основные таблицы и связи)
(составлено авторами)

Прототип интерфейса включает боковую навигацию и центральную панель задач текущего дня, а также блоки аналитики и проектов (рисунок 6). Дизайн ориентирован на сокращение количества действий при ежедневном планировании и соответствует рекомендациям по юзабилити и визуальной иерархии [1,5].

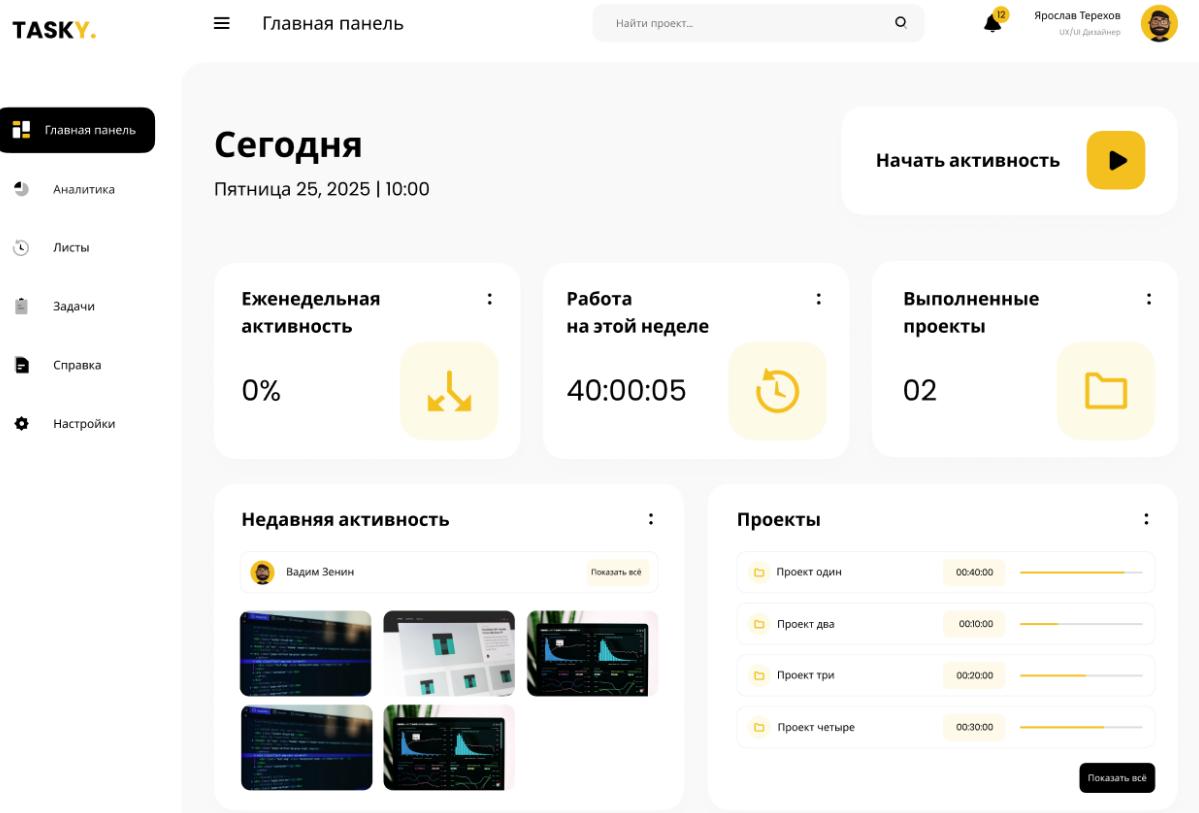


Рис. 6 – Прототип главного экрана веб-приложения (составлено авторами)

Поскольку приложение предполагает хранение пользовательских данных, в проект закладываются меры по обеспечению конфиденциальности и целостности: использование защищённого соединения (HTTPS), хранение паролей в виде хешей, контроль доступа к данным, ведение журналов безопасности, резервное копирование, а также применение типовых защит от XSS/CSRF и инъекций [7,11].

Доступность интерфейса рассматривается как обязательное качество веб-сервиса: поддержка масштабирования, достаточная контрастность, возможность навигации с клавиатуры, корректная семантика элементов и наличие текстовых альтернатив. Требования сопоставляются с действующими нормативными документами и рекомендациями по доступности интернет-ресурсов [9,12].

Анализ аналогов показывает, что для индивидуального тайм-менеджмента востребованы сочетания: низкий порог входа, быстрый ввод задач, наглядное

отображение статуса, надёжные напоминания и базовая аналитика. Предложенный подход проектирования позволяет связать эти требования с формальными моделями процессов и данных, снижая риск функциональных разрывов при переходе к реализации.

Ограничением текущего этапа является отсутствие экспериментальной оценки удобства использования. В дальнейшем целесообразно провести юзабилити-тестирование прототипа и реализованного решения, а также проверку кроссбраузерной совместимости и доступности [8].

Описано проектирование веб-приложения для личного тайм-менеджмента: обоснованы функциональные требования и архитектура (SPA с возможностью использования элементов PWA), построены модели IDEF0 и UML, разработана структура базы данных и прототип пользовательского интерфейса. Рассмотрены требования информационной безопасности и доступности. Полученные результаты формируют основу для последующей реализации и апробации приложения.

Библиографический список:

1. Батенькина, О. В. Юзабилити информационных систем : учебное пособие / О. В. Батенькина, О. Н. Ткаченко. – Омск : ОмГТУ, 2015. – 144 с.
2. Безопасность финансовых (операционных) систем. Защита информации. Общие положения: ГОСТ Р 57580.1–2017. – Введ. 2018-01-01. – М.: Стандартинформ, 2017. – 24 с.
3. Брусникин, Г. Н. Разработка UML-моделей при проектировании информационных систем : учебное пособие / Г. Н. Брусникин, Н. Ю. Соколова. – Москва : МИЭТ, 2023. – 52 с.
4. Виды веб-приложений: выбираем подходящий вариант для вашего бизнеса / [Электронный ресурс] // LEANTech : [сайт]. – URL:

<https://leantech.ai/vebprilozhenie-osnovnye-vidy-i-ih-osobennosti> (дата обращения: 19.12.2025).

5. Габриелян, Т. О. Коммуникативный и мультимедийный дизайн. Графический пользовательский интерфейс : учебно-методическое пособие / Т. О. Габриелян. – Симферополь : КФУ им. В.И. Вернадского, 2021. – 166 с.

6. Леон, У. Разработка веб-приложения GraphQL с React, Node.js и Neo4j / У. Леон ; перевод с английского А. Н. Киселева. – Москва : ДМК Пресс, 2023. – 262 с.

7. Моуэт, Э. Использование Docker / Э. Моуэт ; научный редактор А. А. Маркелов ; перевод с английского А. В. Снастина. – Москва : ДМК Пресс, 2017. – 354 с.

8. Остроух, А. В. Проектирование информационных систем : монография / А. В. Остроух, Н. Е. Суркова. – 2-е изд., стер. – Санкт-Петербург : Лань, 2021. – 164 с.

9. Разработка фронт-енд части веб-ориентированного информационного обеспечения систем управления в организационных системах : учебное пособие / В. Н. Волков, А. В. Демидов, С. В. Новиков [и др.]. – Орел : ОГУ имени И.С. Тургенева, 2023. – 427 с.

10. Романова, И. П. Базы данных: работа с PostgreSQL : учебное пособие / И. П. Романова, П. С. Романов. – Москва : МУИВ, 2023. – 193 с.

11. Старолетов, С. М. Основы тестирования и верификации программного обеспечения / С. М. Старолетов. – 3-е изд., стер. – Санкт-Петербург : Лань, 2023. – 344 с.

12. Сухов, К. К. Node.js. Путеводитель по технологии : учебник / К. К. Сухов. – Москва : ДМК Пресс, 2015. – 416 с.

13. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебник для среднего профессионального образования / А. Ф. Тузовский. – Москва : Издательство Юрайт, 2025. – 219 с.