

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ МОНИТОРИНГА И ВИЗУАЛЬНОГО ПРЕДСТАВЛЕНИЯ ЗАНЯТОСТИ УЧЕБНЫХ АУДИТОРИЙ НА ОСНОВЕ РАСПИСАНИЯ И ПЛАНОВ КОРПУСОВ

В. А. Столяров¹, А. Ф. Атнабаев²

¹stolyarov200138@gmail.com, ²Aaf1981@mail.ru

¹⁻²ФГБОУ ВО «Уфимский университет науки и технологий» (УУНиТ)

Аннотация. Статья описывает создание информационной системы, предназначеннной для отслеживания и отображения состояния загрузки учебных классов университетов путем интеграции университетских расписаний занятий и архитектурных планов зданий. Основной задачей является разработка специализированных компонентов, обеспечивающих оперативный мониторинг заполненности аудитории и предоставление удобной визуализации данных о свободных помещениях. Важнейшими направлениями исследований выступают методы извлечения сведений о занятиях, привязка полученных данных к виртуальным объектам учебного корпуса и внедрение удобного интерактивного веб-интерфейса.

Ключевые слова: визуализация данных; парсинг; расписание занятий; занятость; карта; база данных; Python; модуль.

ВВЕДЕНИЕ

В данный момент для получения информации обо всех свободных аудиториях в учебном помещении пользователи должны вручную по отдельности выбирать помещение для получения информации о занятости, что не всегда удобно и эффективно. Часто возникают ситуации, когда необходимо найти любую свободную аудиторию или несколько аудиторий сразу, что довольно затруднительно с текущей системой просмотра занятости. Данный процесс становится еще более трудоемким в период начала учебного семестра, так как большая часть помещений занята, и тяжело предугадать, какая аудитория может быть свободна в определенный промежуток времени. Также возникают ситуации, когда в течение семестра одни аудитории могут быть постоянно заняты, в то время как другие пустуют, что плохо сказывается на распределении аудиторного фонда. Причиной этому служит отсутствие системы анализа занятости аудиторий. Анализ существующей ситуации показал ряд значимых недостатков, препятствующих эффективному управлению учебными площадями.

Отсутствие визуальной составляющей. Информация о расписании занятий предоставляется лишь в табличной форме, без возможности увидеть размещение аудиторий на карте университетского комплекса. Это осложняет ориентацию в пространстве и замедляет поиск нужных помещений, так как пользователи лишены возможности визуально оценить их расположение и доступность.

Потеря времени на поиски свободного пространства. Процесс подбора доступной аудитории сопряжен с большими временными затратами, так как требует последовательного просмотра и ручной проверки расписания по каждому объекту отдельно. Такая ситуация усугубляется, когда срочно необходимо подобрать подходящее помещение для события или лекции.

Ограниченностю сведений о помещениях. Текущие данные не содержат исчерпывающей информации о характеристиках аудиторий. Пользователи не знают, являются ли помещения лекционными залами или лабораториями, какая там инфраструктура и какое количество посадочных мест имеется. Эти факторы важны при выборе места для крупных мероприятий или практических занятий.

Для преодоления указанных трудностей планируется внедрение специализированной информационной системы, которая предоставит интерактивную карту кампусов с актуальной информацией о занятости аудиторий. Система позволит оперативно подбирать свободные помещения, облегчит процедуру поиска и даст объективную картину использования аудиторного фонда. Автоматическое обновление данных обеспечит достоверность предоставляемых сведений, а интерактивная визуализация поможет эффективнее ориентироваться в корпусе.

АНАЛИЗ ПРИМЕНЕНИЯ ГИС-ТЕХНОЛОГИЙ ДЛЯ СОЗДАНИЯ ИНТЕРАКТИВНЫХ КАРТ

На практике во многих университетах и школах создаются интерактивные веб-карты для учебных, организационных и исследовательских целей. Так, интерактивная карта кампуса Университета штата Орегон (OSU) предлагает пользователям больше информации и функций, чем традиционная печатная карта: она оптимизирована под десктоп и мобильные устройства, отображает рендеры зданий, жилых корпусов, спортивных объектов и т.д., а навигация по категориям и поиск по названию упрощают ориентирование на кампусе [1].

Для реализации интерактивных карт в образовании используются разные программные решения. ArcGIS Online/ArcGIS Enterprise от компании ESRI – одна из ведущих облачных платформ для картографирования и анализа данных. Многие университеты интегрируют ArcGIS в учебную инфраструктуру. Для подготовки материалов широко применяется ArcGIS Pro и ArcGIS StoryMaps [2].

Для создания веб-карт без привязки к конкретной ГИС-платформе применяются JavaScript-библиотеки. Например, Leaflet – «ведущая библиотека с открытым исходным кодом для интерактивных мобильных карт». OpenLayers – ещё одна мощная бесплатная библиотека для динамических веб-карт. Обе позволяют добавлять на карту тайлы, векторные слои, метки и всплывающие окна. Студенты и разработчики могут с их помощью создавать собственные учебные приложения: рисовать маршруты, отображать тематические данные и так далее [3].

ПРОЕКТИРОВАНИЕ ФУНКЦИОНАЛЬНОЙ МОДЕЛИ

В данной работе был выбран структурный подход к проектированию информационной системы благодаря следующим преимуществам: методология IDEF0 зарекомендовала себя как проверенная временем и зрелая технология, получившая широкое распространение среди специалистов в сфере разработки информационных систем; использование понятных и простых визуальных схем делает представление системы доступным как для пользователей, так и для разработчиков; основанный на классических принципах системного анализа, этот подход применим не только к информационным, но и к другим видам систем, идеально подходит для изучения предметной области; четкая последовательность этапов способствует эффективному управлению проектом и повышает качество создаваемых систем.

Создание функциональной модели начинается с формирования контекстной диаграммы. Данная диаграмма служит верхним уровнем представления системы, изображая объект моделирования единственным прямоугольником, связанным внешними стрелками. Блок контекстной диаграммы под названием «Разработать информационную систему отображения занятости аудиторий» (A0) представлен на рис. 1.

Данная диаграмма была декомпозирована для более точного понимания процесса разработки информационной системы. Декомпозиция первого уровня разделена на следующие блоки: «Оцифровать помещение», «Заполнить базу данных», «Разработать

функционал веб-приложения», «Разработать интерфейс приложения». Декомпозиция первого уровня представлена на рис. 2.

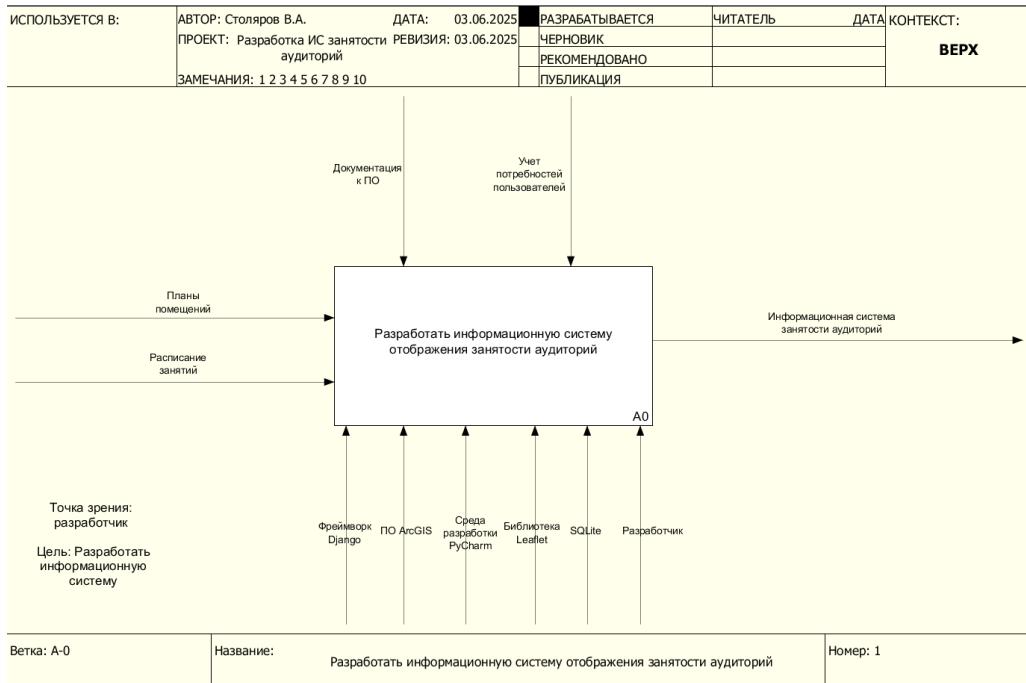


Рис. 1 Контекстная диаграмма функциональной модели

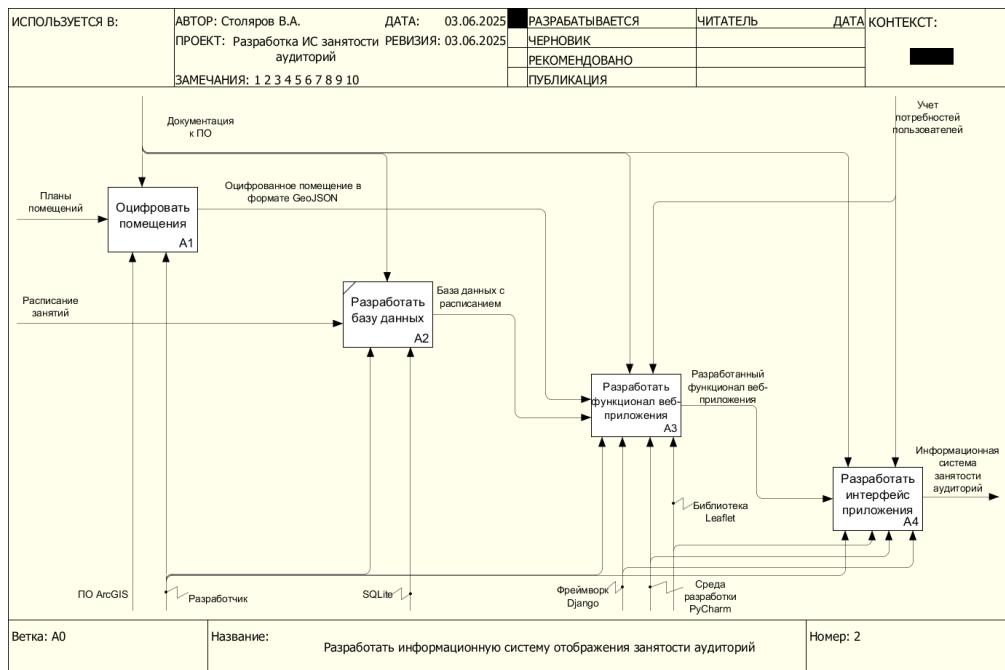


Рис. 2 Декомпозиция первого уровня

ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ МОДЕЛИ

В рамках исследования предметной области была спроектирована диаграмма «Сущность-связь», в ходе проектирования были выделены следующие сущности, атрибуты и связи: сущность «Расписание» имеет следующие поля: «id» (первичный ключ), «id_аудитории» (вторичный ключ), «День», «Время», «Недели». И имеет связь «многие-к-одному» с сущностью «Аудитория». Сущность «Аудитория» имеет следующие поля: «id_аудитории»

(первичный ключ), `id_корпуса` (вторичный ключ), «№ Корпуса», «Тип аудитории», «Количество мест», «Наличие проектора», «Кафедра». И имеет связь «многие-к-одному» с сущностью «Корпус». Сущность «Корпус» имеет следующие поля: `id_корпуса` (первичный ключ), `№ Корпуса`, `Адрес`. Информационная модель предметной области представлена на рис. 3.

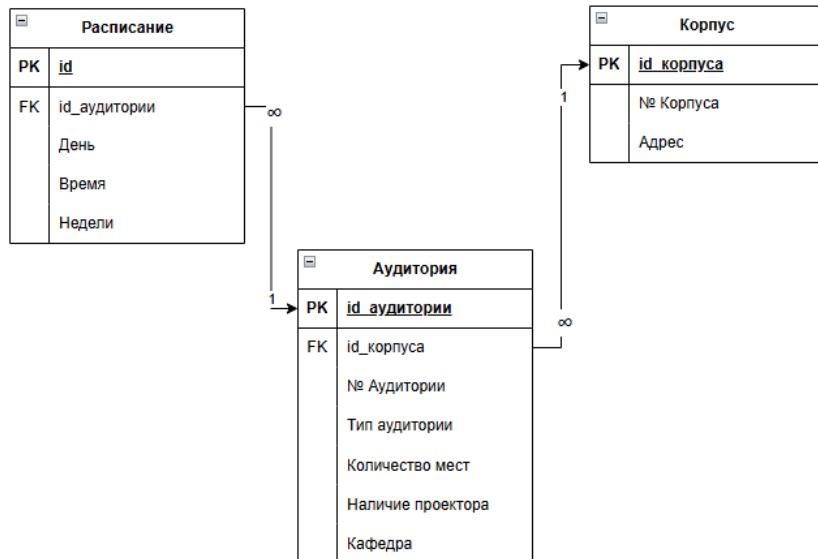


Рис. 3 Информационная модель предметной области

СОЗДАНИЕ ОЦИФРОВАННОЙ МОДЕЛИ ПОМЕЩЕНИЙ

Первым шагом в разработке веб-приложения стала цифровая обработка помещений. Данный этап предусматривает конвертацию растровой графики планов помещений в удобный цифровой вид, пригодный для дальнейшего использования в программе. В ходе процесса каждому помещению присваивались необходимые характеристики, такие как номер комнаты, этаж расположения, назначение помещения и другие сведения.

Поскольку существующие растровые планы могли утратить свою актуальность и не соответствовать современному состоянию помещений, была проведена проверка состояния здания. Итогом проведенных мероприятий стала цифровая модель этажей здания с внесенными характеристиками всех объектов. Полученные данные будут применяться в приложении, повышая точность и полезность предоставляемой информации.

Цифровое изображение первого этажа шестого корпуса «УУНиТ» представлено на рис. 5.

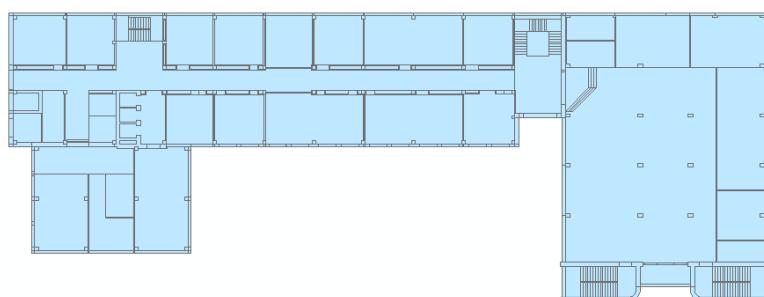


Рис. 5 Оцифрованный первый этаж 6 корпуса «УУНиТ»

РАЗРАБОТКА БАЗЫ ДАННЫХ

Эффективная организация данных является ключевым фактором успешного функционирования современных информационных систем. Прежде чем приступить

непосредственно к проектированию базы данных, был проведен детальный анализ бизнес-процессов образовательного учреждения. Определены основные информационные сущности, необходимые для реализации функционала приложения. В результате проведенного анализа выделены следующие центральные сущности: аудитория (rooms). Хранение сведений о физическом пространстве, используемом для проведения занятий. Атрибуты включают уникальный идентификатор аудитории, номер комнаты, вместимость аудитории и статус занятости. Расписание (Schedule). Содержит детализированную информацию о запланированных занятиях: уникальные идентификаторы записей, наименования предметов, даты и время проведения занятия, принадлежность к корпусу, а также связь с конкретной аудиторией. Корпус (Building). Отражает физическую инфраструктуру организации, включает адрес и номер корпуса. Каждая сущность представлена соответствующей таблицей в реляционной базе данных, содержащей набор полей для фиксации значимой информации.

РАЗРАБОТКА МОДУЛЕЙ ДЛЯ СИСТЕМЫ

Первым делом был реализован проект с использованием популярного веб-фреймворка Django, написанный на языке программирования Python. Этот выбор обусловлен высокой производительностью, надёжностью и гибкостью Django, позволяющими быстро создавать мощные и безопасные веб-приложения. Django предоставляет широкий спектр встроенных инструментов и библиотек, облегчающих разработку сложных проектов, таких как система аутентификации, работа с базами данных, механизм шаблонизации и поддержка RESTful API.

Далее был разработан скрипт-парсер, предназначенный для автоматического сбора и сохранения данных расписания. Используемые библиотеки: Requests: библиотека для отправки HTTP-запросов к серверу с целью получения страниц с интересующей информацией; BeautifulSoup: инструмент для разборки HTML-кода страницы, позволяющий выделять нужные элементы и извлекать необходимую информацию; SQLite3: модуль Python для взаимодействия с базой данных SQLite, что упрощает сохранение собранных данных.

Алгоритм работы скрипта заключается в следующем: отправляется GET-запрос к ресурсу, предоставляющему расписание, передаётся значение параметра building_id. Из полученного HTML-документа выделяется список аудиторий с использованием элемента <select name="room">; далее для каждой найденной аудитории формируются запросы по специальному шаблону URL, содержащему значения room_id. Запросы отправляются к API сервера для получения расписания. Ответ обрабатывается с извлечением необходимых данных: дни недели, время занятий и номера недель. Собранные данные сохраняются в трёх связанных таблицах базы данных: buildings (корпуса), rooms (аудитории) и schedule (расписание). Чтобы избежать повторения данных, используется команда SQL INSERT OR IGNORE. Обработчик ошибок проверяет код состояния HTTP-ответа. Если возникают проблемы, выполняется откат транзакций, и сообщение об ошибке отображается в терминале. Данный скрипт был настроен на ежедневное выполнение в определённый промежуток времени для хранения свежих данных о расписании. Для проверки работы скрипта было выполнено ручное тестирование. Результат представлен на рис. 6.

```
(.venv) PS C:\Users\stoly\Desktop\SheduleMap> cd .\scheduleMap\
(.venv) PS C:\Users\stoly\Desktop\SheduleMap\scheduleMap> python manage.py parse_schedule
Данные для 33 аудиторий сохранены!
```

Рис. 6 Результат ручного выполнения скрипта

Далее был разработан модуль анализа, задача которого состояла в оценке продуктивности использования каждой учебной аудитории. Были сформулированы конкретные цели: сбор статистики по числу проведенных занятий в каждой аудитории; Представление полученных данных как в виде текста, так и в наглядной графической форме;

Реализация динамического интерфейса, обновляющего информацию без полной перезагрузки страницы; Возможность расширения функционала посредством добавления фильтров по этажам, кафедрам и другим критериям.

Был создан скрипт occupancy.py, внутри которого была реализована функция calculate_occupancy(). Эта функция с помощью метода annotate() подсчитывает, сколько раз каждая аудитория встречается в расписании, а функция get_chart_data() формирует структуру данных для диаграммы. На стороне клиента был добавлен JavaScript-модуль occupancy.js, который отправляет AJAX-запрос к серверу при нажатии; получает от сервера агрегированные данные; отображает данные в виде диаграммы с использованием Charts.js; показывает текстовый список загруженности аудиторий. В HTML-шаблон были добавлены контейнеры для отображения результатов и диаграммы, подключение скриптов и стилей, отображение начальных данных, полученных при первом рендеринге страницы.

Назначение ключевых компонентов: Calculate_occupancy() – Получение текста: «Аудитория x – занята n раз»; Get_chart_data() – Формирование данных для графика; Main_view() – (в views.py) Обработка POST и GET запросов, рендеринг шаблона; Occupancy.js – Асинхронное взаимодействие и рендеринг диаграммы; Main.html – Основной интерфейс приложения; Chart.css – Визуальное оформление блока анализа.

Для наглядного представления взаимодействия клиентской и серверной частей в процессе анализа занятости аудиторий была составлена диаграмма (рис. 7).

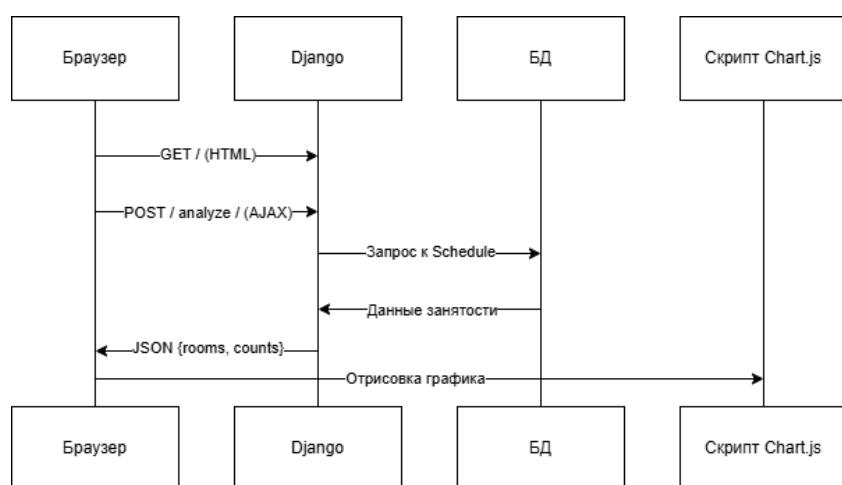


Рис. 7 Механизм взаимодействия клиента и сервера

Результат работы модуля представлен на рис. 8.

```

Data:
  ▾ Object 1
    ▾ counts: (50) [226, 207, 204, 201, 195, 194, 166, 158, 149, 146, 141, ...]
    ▾ rooms: (50) ['111', '218', '305', '509', '307', '207', '101', '107', ...]
    ▾ [[Prototype]]: Object
  
```

Рис. 8 Полученные данные, на основе которых будет строиться диаграмма

РАЗРАБОТКА ИНТЕРФЕЙСА

Разработка качественного пользовательского интерфейса стала одним из приоритетных этапов проекта. Созданный интерфейс учитывает современные тенденции и стандарты UX/UI-дизайна, обеспечивая удобную и комфортную работу с системой на любом устройстве – будь то десктоп, планшет или смартфон. Принцип адаптивной верстки позволил добиться безупречного отображения интерфейса на экранах разных размеров и разрешений. Центральную часть занимает карта, наглядно отражающая распределение

свободных и занятых аудиторий в режиме реального времени. Она облегчает пользователю быстрое принятие решений относительно выбора подходящей аудитории. Дополнительно интерфейс дополнен блоками аналитики, предоставляющими статистику по загрузке аудиторий, графикам проведения занятий и другие полезные метрики. Все важные функции расположены в пределах легкой досягаемости через удобные навигационные меню и кнопки, расположенные таким образом, чтобы обеспечить интуитивность и простоту пользования системой. Внешний вид пользовательского интерфейса представлен на рис. 9.

Также в ходе разработки проекта особое внимание было уделено адаптации пользовательского интерфейса специально для мобильных устройств, учитывая ограниченную область экранов смартфонов и планшетов (рис. 10). Основная цель данной доработки заключалась в обеспечении максимальной эргономичности и легкости восприятия функциональных возможностей приложения пользователями мобильных платформ.

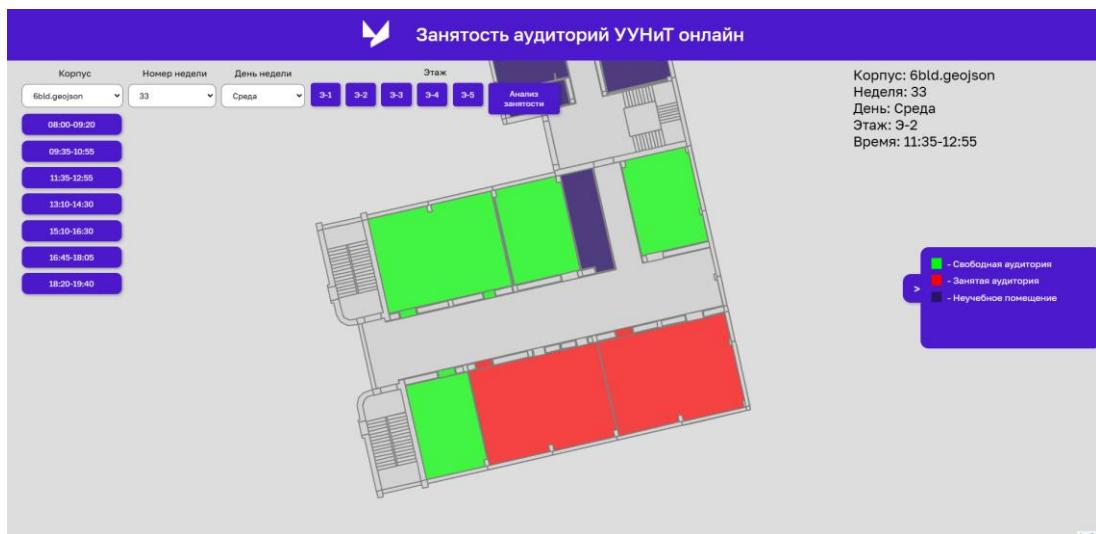


Рис. 9 Внешний вид интерфейса для ПК

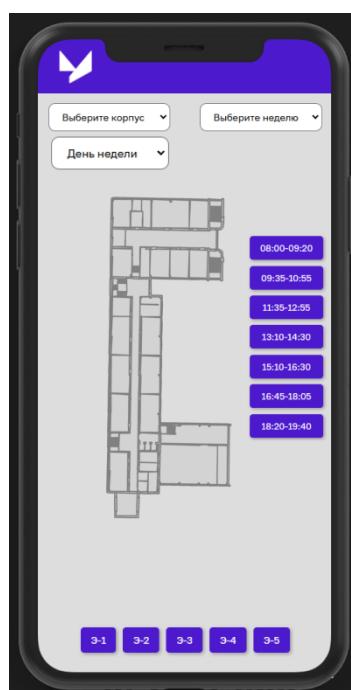


Рис. 10 Внешний вид интерфейса для мобильных устройств

ЗАКЛЮЧЕНИЕ

Представленный проект иллюстрирует целостный метод решения проблемы контроля загруженности учебных помещений в масштабном образовательном учреждении. Использование передовых технических решений и грамотно спроектированной структуры базы данных позволяет разработать действенный инструмент автоматического формирования и демонстрации графика занятий. Применение такой системы существенно облегчает организацию учебного процесса для учащихся и персонала вуза, способствуя повышению общей продуктивности образовательной деятельности. Следующие шаги предполагают расширение функциональности мобильного приложения, введение дополнительных возможностей и увеличение достоверности оперативной информации о статусе аудиторий. Данный опыт может послужить базой для дальнейшей разработки информационных технологий в области образования.

СПИСОК ЛИТЕРАТУРЫ

1. Бурганова Э., Епанешникова Ю. Применение геоинформационных систем в образовании. Елабужский институт КП(Ф)У, 2021. 6 с.
2. Таганов А. И. Геоинформационная система ArcGIS: Учебное пособие / А. И. Таганов, А. Н. Колесенков. Рязань: РГРТУ, 2020. 52 с. — Текст: электронный // Лань: электронно-библиотечная система. URL: <https://e.lanbook.com/book/167982> (дата обращения: 04.06.2025). Режим доступа: для авториз. пользователей.
3. Open Layers // [Электронный ресурс] URL: <https://openlayers.org/#:~:text=OpenLayers%20makes%20it%20easy%20to,also%20known%20as%20the%20FreeBSD>. (дата обращения 20.05.2025)

ОБ АВТОРАХ

СТОЛЯРОВ Владимир Андреевич, студ. каф. ГИС.

АТНАБАЕВ Андрей Фарагатович, к. т. н., доцент ГИС.

METADATA

Title: Development of an information system for monitoring and visualizing the occupancy of classrooms based on schedules and building plans

Author: V.A. Stolyarov¹, A.F. Atnabaev²

Affiliation:

^{1, 2} Ufa University of Science and Technology (UUST), Russia.

Email: stolyarov200138@gmail.com¹, Aaf1981@mail.ru²

Language: Russian.

Source: Molodezhnyj Vestnik UGATU (scientific journal of Ufa University of Science and Technology), no. 3 (34), pp. 134-141, 2025. ISSN 2225-9309 (Print).

Abstract: The article describes the creation of an information system designed to track and display the state of occupancy of university classrooms by integrating university class schedules and architectural plans of buildings. The main task is to develop specialized components that provide operational monitoring of classroom occupancy and convenient visualization of data on available rooms. The most important areas of research are methods of extracting information about classes, linking the obtained data to virtual objects of the educational building and implementing a convenient interactive web-interface.

Key words: Data visualization, parsing, class schedule, occupancy, map, database, Python, module.

About authors:

STOLYAROV Vladimir Andreevich, student, Dept. of GEOGRAPHIC INFORMATION SYSTEMS (UUST).

ATNABAEV Andrey Faragatovich, Ph.D., Assoc. Prof., Dept. of GIS (UUST).