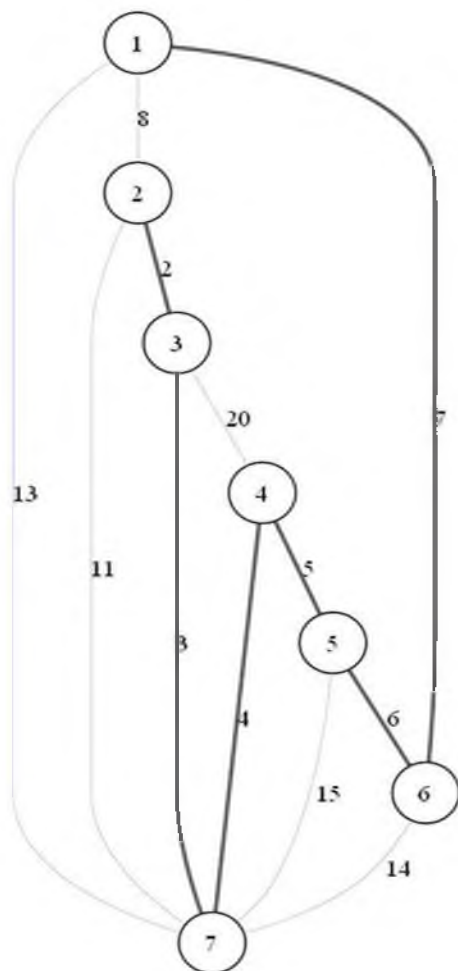




Сочетание C++ и Graphviz обеспечило высокую производительность и гибкость.

Подобным образом мы можем представить абстрактные данные в виде наглядных моделей, что ускоряет анализ, улучшает понимание систем, а также помогает находить неочевидные решения. Без таких инструментов работа с крупными или сложно связанными данными менее эффективна.

Список использованных источников

1. Graphviz Official Website [Электронный ресурс]. – Режим доступа: <https://graphviz.org/>. – Дата доступа: 03.04.2025.
2. Алгоритмы: построение и анализ / Кормен Т. [и др.]; пер. с англ. - 3-е изд. – М.: ООО «И. Д. Вильямс», 2013. – 1328 с.
3. Страуструп, Б. Язык программирования C++. Специальное издание. Пер. с англ. – М.: Издательский дом Бином, 2019. – 1136 с.

Рисунок 3 – Сгенерированное программой изображение

УДК 004.4

ИНФОРМАЦИОННАЯ СИСТЕМА УПРАВЛЕНИЯ ОБЩЕЖИТИЯМИ

Руммо Д. С., студ., Казаков В. Е., к.т.н., доц., Ринейский К. Н., начальник ЦИТ

Витебский государственный технологический университет,

г. Витебск, Республика Беларусь

Реферат. В статье представлен обзор разработки клиентской части информационной системы управления общежитиями, представлены предпосылки его внедрения в учреждение образования и круг решаемых им задач.

Ключевые слова: front-end, управление общежитиями, TypeScript, React.

Управление общежитиями – это информационная подсистема, предназначенная для автоматизации процессов учёта проживания студентов в общежитиях. Она регламентирует структуру и порядок взаимодействия с данными о комнатах, проживающих в них лицах, а также предоставляет механизмы заселения, переселения, выселения и подтверждения проживания. В системе реализован просмотр информации о комнатах и их текущем состоянии, оформление заявок на заселение или переселение, подтверждение факта заселения либо его отмена.

Данный функционал направлен на повышение прозрачности, эффективности и контроля процессов управления жилым фондом общежитий учебного заведения.

В УО «ВГТУ» разработан REST-сервис, обеспечивающий управление данными об общежитиях [1]. Разработана клиентская часть данной информационной системы, которая представляет собой браузерное приложение, не требующее инсталляции. Приложение разработано на языке TypeScript на основе фреймворка React.js [2]. Применялись также: библиотека стилизации компонентов MantineUI; средство организации централизованного хранилища приложения redux toolkit; для обеспечения механизма Dependency Injection использовалась библиотека inversify; библиотека доступа к REST-сервисам axios.

Особое внимание в разработке приложения было уделено пользовательскому опыту. Чтобы пользователь имел представление о содержимом страницы до загрузки данных с сервера, были реализованы skeleton-компоненты. Также были добавлены фильтры для поиска комнат, которые сохраняют своё состояние при повторном открытии страницы по той же ссылке. Для создания удобного и лаконичного интерфейса использовалась библиотека Mantine UI [3], предоставляющая широкие возможности для кастомизации темы приложения (цветовая схема, стили и др.), а также для создания собственных компонентов и модальных окон. Интерфейс был адаптирован под мобильные устройства, что обеспечивает комфортное использование приложения на разных экранах. Кроме того, реализована поддержка светлой и тёмной тем, между которыми пользователь может переключаться в зависимости от своих предпочтений или настроек операционной системы.

При проектировании интерфейса учитывались требования сотрудников общежития.

Приложение реализует следующие основные функции: просмотр комнат общежитий с различными фильтрами и сортировками, просмотр подробной информации о комнате, управление проживающими в комнатах, переселение, заселение, подтверждение заселения (рис. 1).

Страница комнат содержит фильтры и таблицу со списком комнат, а также пагинацию. Для удобства пользователя комнаты без свободных мест выделяются цветом.

Номер общежития	Этаж	Номер блока	Комната	Тип блока	Свободно мест
5	2	1	201м	Мужской	1
1	2	2	202б	Женский	2
1	3	1	301м	Мужской	3
2	1	3	101б	Женский	0
2	4	1	401м	Мужской	4
3	5	2	501б	Женский	2
3	3	1	302м	Мужской	0
3	4	2	402б	Женский	1
4	2	3	203м	Мужской	3
4	3	1	303б	Женский	4
5	4	2	402м	Мужской	1
5	5	2	503б	Женский	0
6	2	2	204м	Мужской	2
6	3	1	304б	Женский	3
7	4	2	404м	Мужской	0
7	5	1	504б	Женский	1
8	1	3	101м	Мужской	2
8	2	2	201б	Женский	4
9	3	1	301м	Мужской	1
9	4	2	401б	Женский	0

Рисунок 1 – Вид страницы комнат общежитий

Фильтры применяются автоматически после того, как пользователь введет что-либо в текстовое поле или же выберет какой-то пункт в селекторе, для полей ввода был применен debounce, который отправляет запрос на сервер для получения данных не на каждый ввод символа пользователем, а через некоторое время.

На странице комнаты (рис. 2) можно увидеть различный функционал для управления проживающими, при каком-либо действии появляется модально окно для подтверждения, чтобы сократить нежелательные действия. При каких-либо ошибках, которые возвращает сервер, пользователь так же будет уведомлен с помощью модального окна.

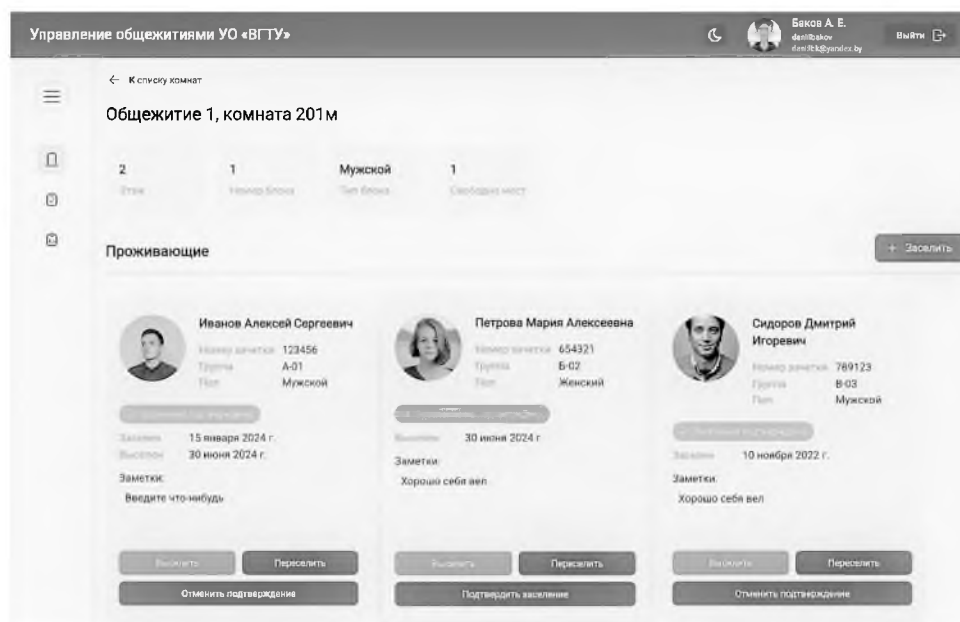


Рисунок 2 – Вид страницы комнаты с данными

Для улучшения пользовательского опыта во время загрузки приложение отображает страницу, представленную на рисунке 3. Страница отображается очень быстро, и состоит из элементов, ожидающих загрузки, и элементов, с которыми пользователь может взаимодействовать, не дожидаясь загрузки остальных элементов.



Рисунок 3 – Вид страницы комнаты во время загрузки данных

Использование сотрудниками общежитий данного модуля информационной системы университета позволит централизованно управлять заселением студентов, оперативно обновлять информацию о проживающих, а также обеспечивать другие модули системы актуальными данными – например, для учета занятости комнат, формирования отчетности или взаимодействия с административными службами в рамках студенческой регистрации и контроля проживания.

Список использованных источников

1. Казаков, В. Е. Разработка back-end приложения «Учебно-методический отдел (учебные планы)» / М. С. Карнилов, В. Е. Казаков // Материалы докладов 55-й Международной научно-технической конференции преподавателей и студентов: в 2 т. / УО «ВГТУ». – Витебск, 2022. – Т. 2. – С. 5–7.
2. Сайт «react.dev» [Электронный ресурс]. – Режим доступа: <https://react.dev/learn>. – Дата доступа: 12.05.2025.
3. Сайт «mantine.dev» [Электронный ресурс]. – Режим доступа: <https://mantine.dev>. – Дата доступа: 12.05.2025.

УДК 004.8:61

НЕЙРОННАЯ СЕТЬ СРЕДСТВАМИ ЯЗЫКА Python ДЛЯ РЕШЕНИЯ ЗАДАЧ КЛАССИФИКАЦИИ

*Тришин Я. Д., студ., Соколова А. С., ст. преп., Черненко Д. В., ст. преп.
Витебский государственный технологический университет,
г. Витебск, Республика Беларусь*

Реферат. Цель работы – создание интеллектуальной системы с помощью искусственной нейронной сети средствами языка Python для классификации объектов, а именно: цифровых изображений. Метод исследования – моделирование. В качестве исходных объектов используется объёмная база данных образцов рукописного написания цифр MNIST. В качестве инструментальных средств разработки использовались: язык программирования Python; открытые библиотеки TensorFlow, Keras, OpenCV.

Ключевые слова: многослойный персептрон, интеллектуальная система, распознавание объектов, моделирование искусственной нейронной сети, задачи машинного обучения или анализа.

Искусственная нейронная сеть всё чаще применяется для решения разного рода задач в повседневной жизни. Прогнозирование, распознавание изображений, речи – неполный ряд функций, которые нейронная сеть выполняет уже сейчас с высокой точностью. Наиболее актуальная задача из имеющихся – задача классификации объектов по цифровым изображениям.

В результате работы изучена архитектура многослойного персептрона, процесс обучения нейронной сети, а также создана и исследована интеллектуальная система для распознавания объектов по цифровым изображениям.

В качестве обучающей выборки была использована база MNIST (сокращение от "Modified National Institute of Standards and Technology"). Это база данных образцов рукописного написания цифр, которая поставляется вместе с Keras. Для доступа к ней понадобится выполнить импорт библиотек языка Python:

matplotlib.pyplot (plt) – библиотека для визуализации данных (графики, изображения);

fetch_openml из sklearn.datasets – функция для загрузки датасетов из OpenML [1].

Загрузим данные MNIST. В базе 60000 изображений в обучающей выборке и 10000 в тестовой. Для примера визуализируем первых 25 изображений. В результате выполнения кода на экране появится сетка 5x5 с первыми 25 изображениями из MNIST (рис. 1).

Для решения задач классификации выполним моделирование искусственной нейронной сети. Полносвязная нейронная сеть состоит из входного, скрытого и выходного слоёв. Нам необходимо подобрать такое количество нейронов в каждом слое, чтобы получить высокую