Утверждаю

Руководитель предприятия

Гончар А.А. \_

(Ф.И.О.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, печать предприятия)

« \_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_2022 Г.

Учреждение образования

«Белорусский государственный технологический университет»

Факультет информационных технологий

Кафедра информационных систем и технологий

Специальность 1-40 05 01 «Информационные системы и технологии»

ОТЧЕТ

о производственной технологической практике

на \_ООО «А2 Консалтинг» в период с «27» июня 2022 г. по «22» июля 2022 г.\_\_\_\_\_\_\_\_\_\_

(наименование предприятия, сроки практики)

Исполнитель

студента 3 курса \_\_3\_\_\_группы \_\_\_\_\_\_\_\_\_\_\_\_\_ Кравжуль А.Д.

(подпись, дата) (Ф.И.О.)

Руководитель практики

от предприятия

руководитель центра обучения \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_ Краенкова К.И.\_\_\_

(должность, печать предприятия) (подпись, дата) (Ф.И.О.)

Руководитель практики

от университета

ассистент \_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ Берников В. О.\_\_\_

(должность, уч. звание) (подпись, дата) (Ф.И.О.)

Отчет защищен с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Минск 2022

**Содержание**

[Введение 3](#_Toc109241113)

[1. Общая характеристика предприятия. 4](#_Toc109241114)

[1.1. Общие сведения о предприятии, его структуре 4](#_Toc109241115)

[1.2. Применяемые информационные технологии. 4](#_Toc109241116)

[1.3. Разрабатываемое на предприятии ПО 4](#_Toc109241117)

[2. Организационный подход к практике. 6](#_Toc109241118)

[3. Индивидуальное задание 7](#_Toc109241119)

[3.1. Разработка и внедрение уведомлений в проект на Next.js 7](#_Toc109241115)

[3.2. Разработка и внедрение заметок в проект на Next.js 12](#_Toc109241115)

[Заключение 16](#_Toc109241121)

[Список использованных источников информации 17](#_Toc109241122)

[ПРИЛОЖЕНИЕ А 18](#_Toc109241123)

[ПРИЛОЖЕНИЕ Б 19](#_Toc109241123)

Введение

Технологическая практика представляет собой углубленное изучение хозяйственной деятельности предприятия, что позволяет студенту детально ознакомиться с тонкостями и нюансами выбранной профессии. Она является обязательной частью образовательной программы подготовки дипломированного специалиста. Основными целями практики являются закрепление знаний и навыков, полученных в ходе теоретического обучения, а также приобретение студентами практического опыта в решении реальных задач.

В процессе технологической практики студент должен ознакомиться с:

* структурой организации, где проходит практика;
* проектной деятельностью и основными принципами создания проектов;
* действующими стандартами, документацией и инструкциями, применяемыми в организации.

Также студенту необходимо освоить:

* особенности использования Next js;
* принцип построения приложения на Next js.

Основные задачи практики включают:

* изучение организационной структуры компании;
* анализ информационных ресурсов и технологий, используемых в организации;
* составление отчета на основе выполненной работы и собранной информации в соответствии с программой практики.

Местом прохождения технологической практики является ООО «СТЭКЛЭВЭЛ ГРУПП».

Общая характеристика предприятия.

Общие сведения о предприятии, его структуре

ООО «СТЭКЛЭВЭЛ ГРУПП» — это международная IT-компания, специализирующаяся на разработке программного обеспечения на заказ для бизнеса. Компания была основана с целью предоставления высококачественных решений для клиентов по всему миру. Основные направления деятельности включают разработку веб- и мобильных приложений, корпоративных систем, а также решений на основе современных технологий, таких как искусственный интеллект и машинное обучение.

Компания «СТЭКЛЭВЭЛ ГРУПП» имеет международное присутствие, предоставляя услуги клиентам в различных регионах, включая Европу, Северную Америку и другие части мира. В структуру компании входят отделы разработки, тестирования, управления проектами и клиентской поддержки, что позволяет эффективно управлять проектами на всех этапах — от анализа требований до сопровождения готового продукта.

Команда компании состоит из более чем 6 высококвалифицированных специалистов, обладающих экспертизой в различных областях IT. «СТЭКЛЭВЭЛ ГРУПП» активно привлекает талантливых разработчиков, инженеров и менеджеров проектов, что позволяет компании предлагать инновационные и эффективные решения для различных отраслей, включая финансы, здравоохранение, ритейл и логистику.

1.2 Применяемые информационные технологии

В работе «СТЭКЛЭВЭЛ ГРУПП» применяются передовые информационные технологии и инструменты, обеспечивающие высокую производительность и надежность разрабатываемого программного обеспечения.

Сотрудники компании владеют такими языками программирования и технологиями, как PHP7, Symfony, Doctrine, Docker, RabbitMQ, Sphinx, ApacheSolr, REST, а также Ruby, RubyonRails, Node.js, React.js, Angular, Vue.js, и ReactNative. Компания также имеет значительный опыт работы с реляционными и нереляционными базами данных, включая MySQL, PostgreSQL, Oracle, Redis, и MongoDB, и успешно использует средства репликации и шардинга.

Кроме того, «СТЭКЛЭВЭЛ ГРУПП» активно внедряет практики DevOps, что позволяет интегрировать процессы разработки и эксплуатации, ускоряя доставку продукта на рынок и повышая его качество. В своей работе компания также применяет технологии обработки больших данных (Big Data), что помогает клиентам эффективно анализировать и использовать большие объемы информации.

1.3 Разрабатываемое на предприятии ПО

Компания «СТЭКЛЭВЭЛ ГРУПП» занимается разработкой разнообразного программного обеспечения, включая кастомизированные веб- и мобильные приложения, созданные с учетом уникальных потребностей каждого клиента. В процессе разработки используются современные технологии и языки программирования, такие как PHP7, Symfony, Node.js, React.js, Angular, и другие. Компания также специализируется на создании сложных корпоративных решений, включая CRM, ERP-системы и платформы для электронной коммерции.

Кроме разработки собственного программного обеспечения, «СТЭКЛЭВЭЛ ГРУПП» активно занимается интеграцией и адаптацией существующих решений для своих клиентов. В частности, компания предоставляет услуги по настройке и внедрению популярных технологий и инструментов, таких как Docker, RabbitMQ, и ApacheSolr, что позволяет клиентам эффективно управлять своими бизнес-процессами и повышать производительность.

Организационный подход к практике

Практика в компании ООО «СТЭКЛЭВЭЛ ГРУПП» была организована как полноценное погружение в реальную работу команды разработчиков. С первого дня нас воспринимали как полноправных участников проекта, что позволило нам ощутить себя частью команды и приобрести опыт, максимально приближенный к реальной работе в сфере разработки программного обеспечения.

Ежедневно мы принимали участие в командных созвонах, которые проходили в Microsoft Teams. На этих встречах обсуждались текущие задачи, проблемы и достижения команды. Это позволило нам быть в курсе всех аспектов работы и понимать, как наша работа интегрируется в общий проект.

Важным элементом нашей практики был наш ментор, который не только давал нам конкретные задачи через Trello, но и сопровождал нас на каждом этапе их выполнения. Ментор следил за качеством нашего кода, давал ценные советы и помогал исправлять ошибки, что способствовало нашему профессиональному росту.

Задачи, которые мы выполняли, охватывали различные аспекты разработки: от написания кода до его тестирования и интеграции с другими модулями. Все задания, которые мы получали, были реальными задачами, стоящими перед командой, что делало наш вклад важным и значимым.

Каждое выполненное задание обсуждалось с ментором, который давал обратную связь, помогал улучшить качество кода и предлагал новые подходы к решению задач. Такой формат работы позволил нам не только углубить свои знания и навыки, но и научиться эффективно работать в команде, оперативно решать возникающие проблемы и справляться с реальными вызовами, стоящими перед разработчиками.

В итоге практика в ООО «СТЭКЛЭВЭЛ ГРУПП» стала для нас не только возможностью применить полученные знания на практике, но и ценным опытом работы в команде, нацеленной на достижение общего результата.

# **Индивидуальное задание**

* 1. Разработка и внедрение уведомлений в проект на Next js

В первые 2 недели практики основной задачей было добавление функциональности уведомлений на технологии next js. В качестве базы данных использовалась нереляционная база данных MongoDB. Также в рамках задачи необходимо было разработать интерфейс для уведомлений.

Задача по внедрению уведомлений включала разработку интуитивно понятного пользовательского интерфейса, позволяющего отображать конкретные уведомления лично для пользователя. Также, отдельно отображаются непрочитанные уведомления. Пользовательский интерфейс уведомлений представлен на рисунке 3.1.

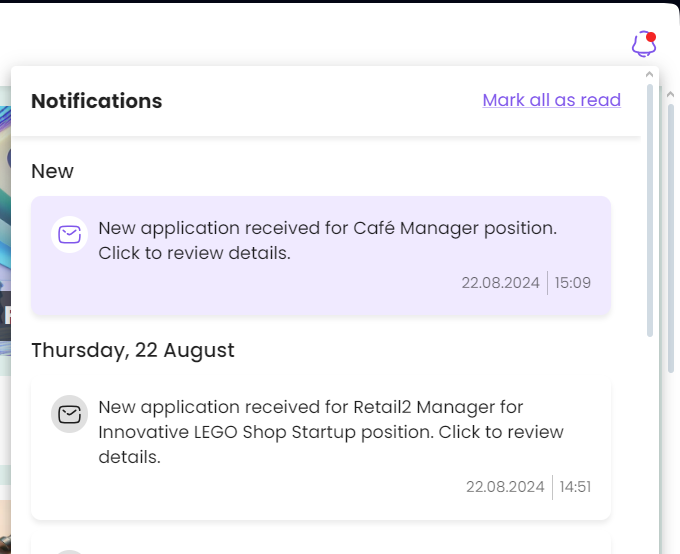


Рисунок 3.1 — Интерфейс уведомлений

Одной из ключевых задач являлась создание самих уведомлений. Необходимо было создать коллекцию с уведомлениями в базе данных. А также, необходимо добавлять записи в эту коллекцию после завершения каких-то событий. Пример кода добавления уведомления представлен в приложении А.

Далее необходимо было написать graphql запрос, для получения только необходимых полей и достижения меньшей загруженности сети а также написать эндпоинты для добавления уведомления и получения всех уведомлений. Пример graphql запроса для получения всех уведомлений представлен на рисунке 3.2.

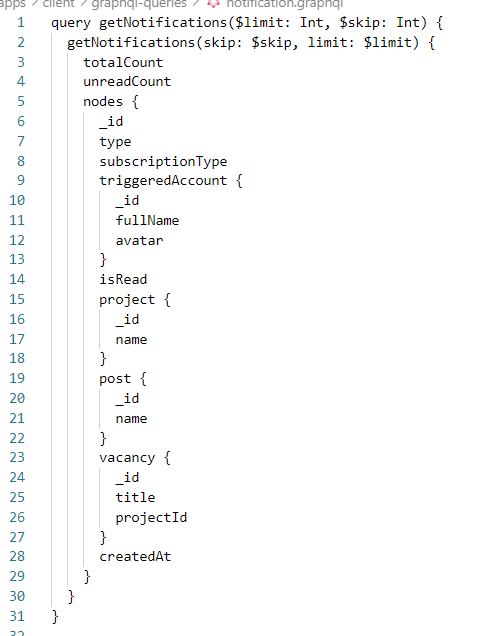


Рисунок 3.2 — Пример graphql запроса для получения всех уведомлений

Также необходимо было написать graphql запрос для того, чтобы отметить уведомления как прочитанные

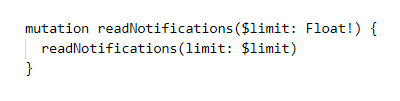


Рисунок 3.3 — Пример graphql запроса чтобы сделать все уведомления прочитанными

Были также написаны эндпоинты на Nest js, в результате которых запрашивались все уведомления для пользователя и отмечались как прочитанные.

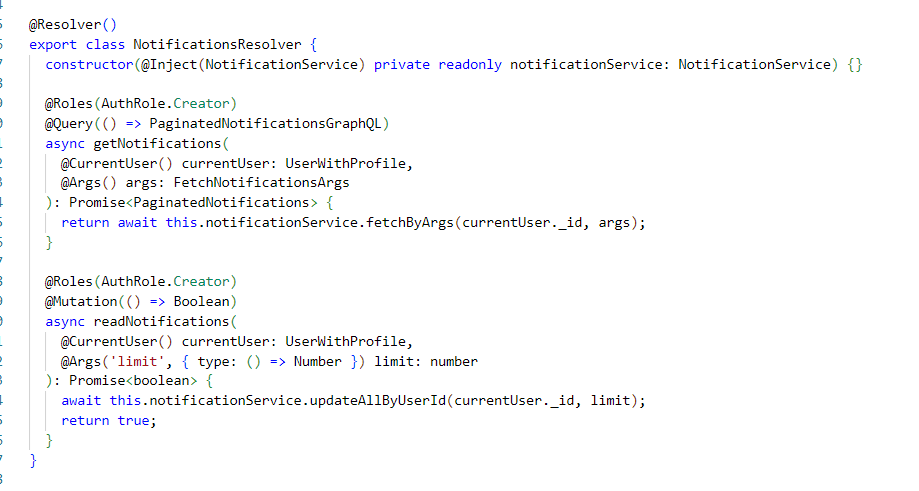


Рисунок 3.4 — Пример эндпоитов на Nest js, связанные с уведомлениями

После написания всей бизнес-логики, необходимо было разработать пользовательский интерфейс. Интерфейс представлял из себя иконку колокольчика, возле которой, при наличии непрочитанных уведомлений, показывалась красная точка.

При нажатии на колокольчик появлялось всплывающее окно, в котором показывались все уведомления, связанные с авторизованным пользователем. Предоставлялась дополнительная информация: дата и время создания уведомления, а также содержание уведомления. На рисунке 3.5 представлен колокольчик.



Рисунок 3.5 — Колокольчик, когда всплывающее окно закрыто

На рисунке 3.6 представлено всплывающее окно со всеми уведомлениями.

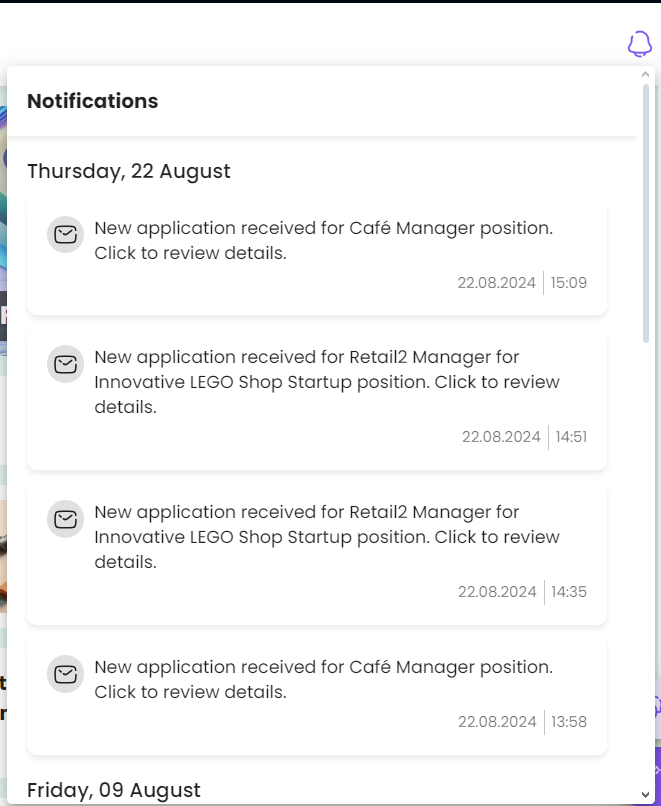


Рисунок 3.6 — Колокольчик c открытым всплывающим окном

В ходе работы над проектом активно использовался Github для версионного контроля и совместной работы. Каждый участник команды создавал и вел свою ветку, куда вносил изменения, после чего осуществлялось слияние этих веток в основную (main) через pull requests. Такой подход позволил команде эффективно сотрудничать, контролировать версионность проекта и поддерживать стабильность финальной версии программного обеспечения.

Также были добавлены необходимые файлы, которые были логически организованы. Ниже изображена структура папок. (Рисунок 3.7)

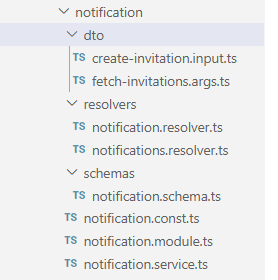


Рисунок 3.7 — Структура папок

В результате задания была добавлена и успешно внедрена функциональность уведомлений, удовлетворяющая всем поставленным требованиям и успешно интегрированная в общий проект на платформе Github.

* 1. Разработка и внедрение заметок в приложение на Next js

На последней неделе необходимо было разработать функциональность заметок, а точнее их создание, редактирование, удаление и закрепление. Заметки представляют собой сущность, где пользователь может сохранить свои идеи, о которых он вскоре может забыть.

Необходимо было разработать весь пользовательский интерфейс, а также написать все эндпоинты, graphql запросы, и создать отдельную коллекцию в базе данных MongoDB под названием “notes”.

Для заметок была создана отдельная страница. Она представляет из себя страницу, где перечислены все заметки пользователя. Перед всеми заметками находится блок, с помощью которого пользователь может добавить заметку. (Рисунок 3.8).

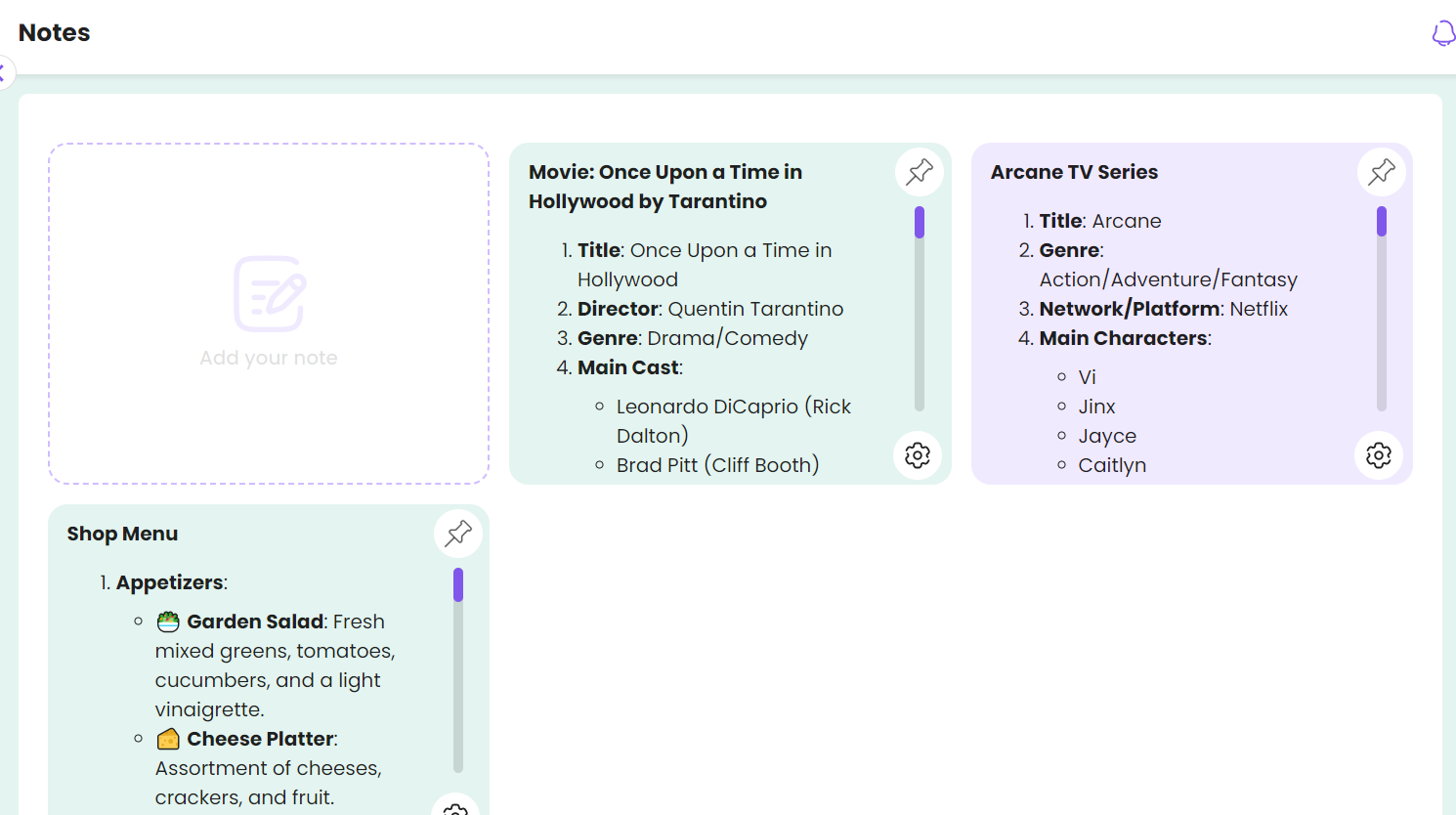


Рисунок 3.8 — Страница Notes

Цвет фона карточек чередуется в зависимости от их порядкового номера. Каждая карточка состоит из контента, иконки закрепления заметки и иконки настроек, при нажатии на которую появляется выбор, редактировать или удалить заметку, а сама иконка настроек меняет свой цвет на фиолетовый. (Рисунок 3.9).

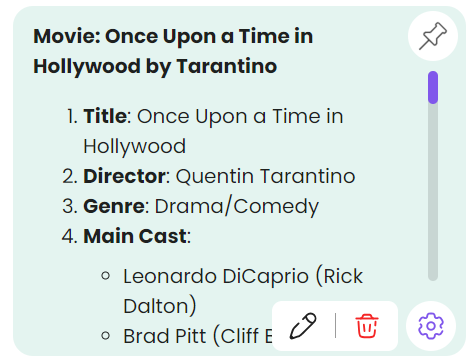


Рисунок 3.9 — Карточка, с открытыми настройками

При нажатии на блок с добавлением заметки появляется контейнер, в котором находится текстовое поле для ввода заметки, иконка мусорки, для очистки содержимого поля и кнопка “Save”, для сохранения заметки. (Рисунок 3.10).

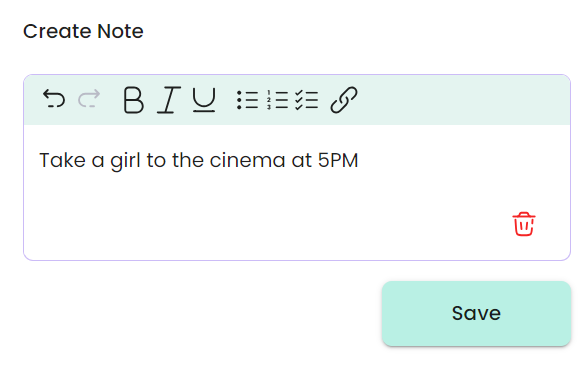


Рисунок 3.10 — Контейнер с текстовым полем, кнопкой очистки и кнопкой сохранения заметки

При нажатии на “Save”, заметка будет создана. При нажатии на иконку редактирования открывается заметка с заранее заполненным текстовым полем, при нажатии на кнопку “Update” она будет обновлена. При нажатии на мусорку заметка будет удалена.

Были написаны необходимые эндпоинты (Рисунки 3.11 и 3.12) и создана удобная структура папок. (Рисунок 3.13).

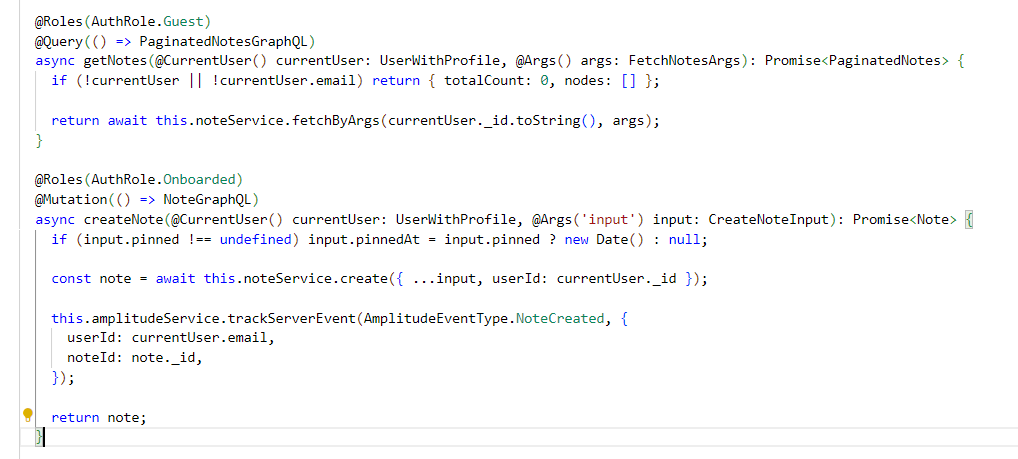


Рисунок 3.11 — Эндпоинты с получением всех заметок и созданием заметки



Рисунок 3.12 — Эндпоинт с удалением заметки

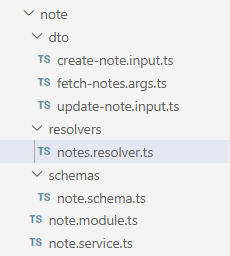


Рисунок 3.13 — Структура папок и файлов для сущности заметки

Также были написаны необходимые graphql запросы, которые будут предоставлены в Приложении Б.

При нажатии на кнопку закрепления заметки сама иконка меняет цвет, а заметка перемещается перед всеми остальными заметками. (Рисунок 3.14)

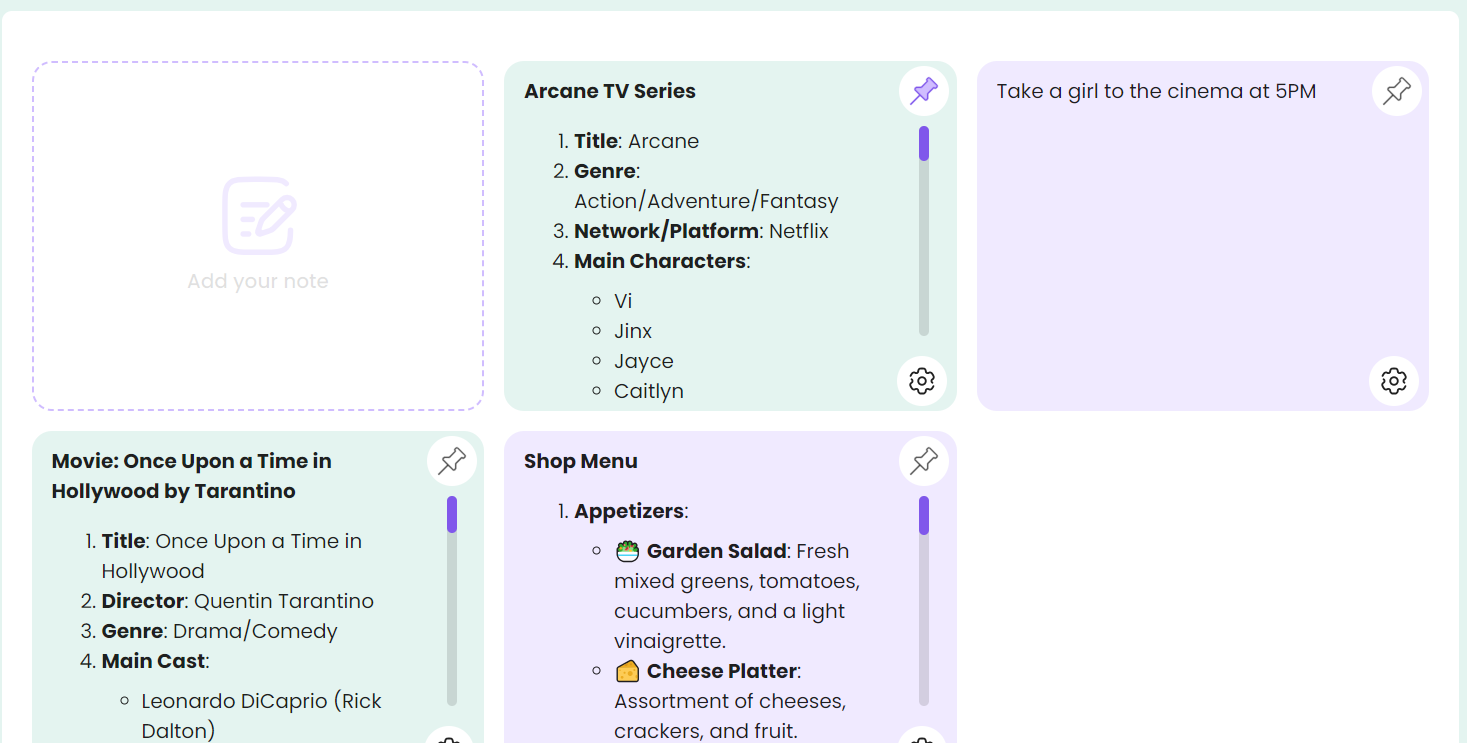


Рисунок 3.14 — Изменение цвета иконки закрепления заметки и смещение заметки в начало всех остальных заметок

Порядок заметок меняется благодаря полям “pinned” и “pinnedAt”, которые прокидываются как поля в функцию обновления заметки. Сортировка заметок происходит по полю “pinnedAt”, а само указание на “закрепленность” поля указывает поле “pinned”. Ниже предоставлен эндпоинт обновления заметки. (Рисунок 3.15).

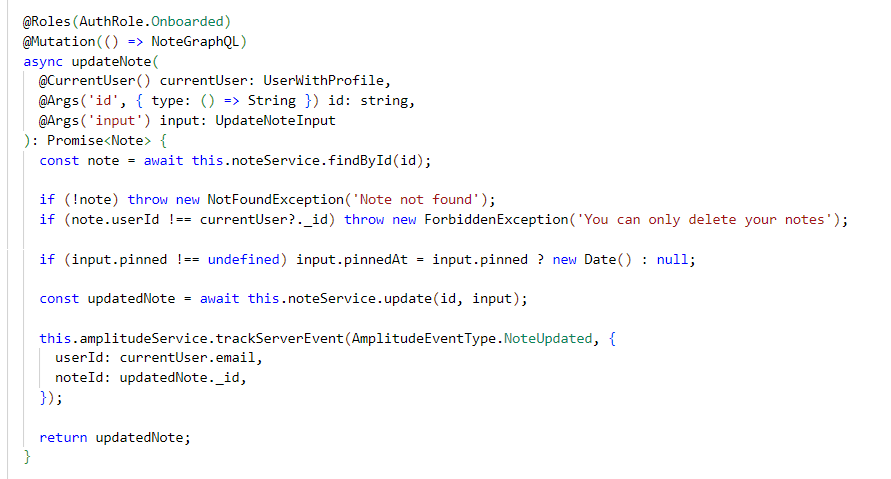


Рисунок 3.15 — Эндпоинт изменения заметки

**Заключение**

В ходе прохождения практики был получен ценный опыт в области командной разработки приложений, что позволило значительно улучшить навыки работы в коллективе и взаимодействия с другими разработчиками.

Были усовершенствованы навыки использования системы контроля версий Git в командной среде, включая создание и слияние веток, а также управление pull requests. Особое внимание было уделено разработке приложений с использованием технологий Next js, Nest js, GraphQl и MongoDB.

Список использованных источников информации

1. Официальная документация Next.js [Электронный ресурс]. – Режим доступа: https://nextjs.org/docs . – Дата доступа: 02.09.2024.
2. Официальная документация Nest.js. [Электронный ресурс].  – Режим доступа: https://docs.nestjs.com/– Дата доступа: 02.09.2024.
3. GitHub [Электронный ресурс]. – Режим доступа: <https://github.com> – Дата доступа: 02.09.2024.
4. Официальная документация GraphQl [Электронный ресурс]. – Режим доступа: https://graphql.org/learn/– Дата доступа: 02.09.2024

# **ПРИЛОЖЕНИЕ А**

|  |
| --- |
| await this.notificationService.create({          type: input.type === CommentType.Post ? NotificationType.Comment : NotificationType.ProjectChat,          postId: comment.postId,          projectId: comment.projectId,          userId: post?.userId || project?.userId,          triggeredBy: currentUser.\_id,          isRead: false,        }); |

# **ПРИЛОЖЕНИЕ Б**

|  |
| --- |
| query getNotes($filter: NoteFilter, $search: String, $sort: NoteSort) {    getNotes(filter: $filter, search: $search, sort: $sort) {      nodes {        \_id        content        createdAt        pinned        userId      }      totalCount    }  }  mutation createNote($input: CreateNoteInput!) {    createNote(input: $input) {      \_id    }  }  mutation updateNote($updateNoteId: String!, $input: UpdateNoteInput!) {    updateNote(id: $updateNoteId, input: $input) {      \_id    }  }  mutation deleteNote($noteId: String!) {    deleteNote(id: $noteId)  } |