Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение   
высшего образования

**Иркутский национальный исследовательский**

**технический университет**

|  |
| --- |
| Институт информационных технологий и анализа данных |
| наименование института |

Допускаю к защите

|  |  |
| --- | --- |
| Руководитель |  |
|  | подпись |
|  | З. А. Бахвалова |
|  | И.О. Фамилия |

|  |
| --- |
|  |
| Разработка прикладного программного обеспечения |
| наименование темы |

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе по дисциплине

|  |
| --- |
| Технология разработки программных комплексов |

|  |
| --- |
| 1.022.00.00 ПЗ |
| обозначение документа |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил студент |  | ИСТб-20-1 |  |  |  | Е.В. Соболевский |
|  |  | шифр группы |  | подпись |  | И.О. Фамилия |
| Нормоконтроль |  |  |  |  |  | З. А. Бахвалова |
|  |  |  |  | подпись |  | И.О. Фамилия |
| Курсовая работа защищена с оценкой | | | |  | | |

Иркутск 2023 г.

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

ЗАДАНИЕ

на курсовую работу

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| По курсу | Технология разработки программных комплексов | | | | | | | | | | | | | | | | | |
| Студенту | Соболевскому Е.В. | | | | | | | | | | | | | | | | | |
|  | (фамилия, инициалы) | | | | | | | | | | | | | | | | | |
| Тема работы | | Разработка модуля отчетности для ответственного по направлению для проекта «Рейтинг студентов» | | | | | | | | | | | | | | | | |
| Исходные данные | | | Вариант 22 | | | | | | | | | | | | | | | |
| Рекомендуемая литература: | | | | | | | |  | | | | | | | | | | |
| 1. Иванова Г.С. Технология программирования: Учебник. — М: Кнорус, 2010. 2. Лэнгсам Й., Огенстайн М., Тененбаум А. Структуры данных для персональных ЭВМ. — М: Мир, 1989-568с. 3. Диго С.М., Клешко Г.Н., Мишенин А.И., Петров Е.А. Сборник задач по курсу "Информационные системы и структуры данных". — М: Статистика,1981. — 144с. | | | | | | | | | | | | | | | | | | |
| Графическая часть на | | | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ листах. | | | | | | | | | | | | | | |
| Дата выдачи задания | | | | « | 15 | » | февраля | | | | | 2023 г. | | | | |
|  | | | | | | | | | | | | | | | | | | |
| Задание получил | | | | | | | | |  | | | | |  | | Е.В. Соболевский | | |
|  | | | | | | | | | подпись | | | | |  | | И.О. Фамилия | | |
|  | | | | | | | | | |  |  | |  | |  | | |  |
| Дата представления работы руководителю | | | | | | | | | | « | 20 | | » | | мая | | | 2023 г. |
|  | | | | | | | | | | | | | | | | | | |
| Руководитель курсовой работы | | | | | | | | |  | | | | |  | | З.А. Бахвалова | | |
|  | | | | | | | | | подпись | | | | |  | | И.О. Фамилия | | |

Оглавление

[Введение 5](#_Toc146481483)

[1. Формирование требований 6](#_Toc146481484)

[1.1. Описание предметной области 6](#_Toc146481485)

[1.2. Описание проблемы 6](#_Toc146481486)

[1.3. Цель 6](#_Toc146481487)

[1.4. Постановка задачи 6](#_Toc146481488)

[1.5. Словари 6](#_Toc146481489)

[1.6. Функциональность ПО 9](#_Toc146481490)

[1. Анализ функциональных требований 10](#_Toc146481491)

[2.1. Модель предметной области (IDEF0) 10](#_Toc146481494)

[2.2. Варианты использования UML 15](#_Toc146481495)

[2.3. DFD - диаграмма потоков данных 16](#_Toc146481496)

[2.4. Концептуальная модель хранилища данных 18](#_Toc146481497)

[2.5. Логическая модель БД 18](#_Toc146481498)

[2.6. Физическая модель БД 19](#_Toc146481499)

[2.7. Варианты использования 19](#_Toc146481500)

[2.8. Описание интерфейса 24](#_Toc146481501)

[3. Проектирование программного продукта 26](#_Toc146481502)

[3.1. Используемый инструментарий 26](#_Toc146481503)

[3.2. Фреймворки и библиотеки 26](#_Toc146481504)

[3.3. Описание архитектуры разрабатываемого ПО 27](#_Toc146481505)

[4. Пользовательский интерфейс 29](#_Toc146481506)

[5. Программная реализация 31](#_Toc146481507)

[5.1. Таблица спецификаций класса SearchTeamDto 31](#_Toc146481508)

[5.2. Таблица спецификаций класса TeamsController 31](#_Toc146481509)

[Заключение 33](#_Toc146481510)

# **Введение**

В данной работе необходимо разработать модуль страницы коллектива для проекта «Рейтинг студентов».

Рейтинговая стипендия или ПГАС (Повышенная государственная академическая стипендия) – повышенная стипендия, которую может получить студент за какие-либо достижения в различных направлениях деятельности в университете.

Одним из основных способов получения рейтинговой стипендии является участие в жизни каких-либо коллективов.

Для того, чтобы получить информацию о коллективах, изучить их подробнее и создается данный модуль.

На странице будет собрана основная информация о коллективе и будет возможность изучить его деятельность с целью вступления.

# **Формирование** **требований**

## **Описание предметной области**

Компания заказчик занимается распределением рейтинга для начисления рейтинговой стипендии. Программный продукт представляет из себя сайт с большим количеством модулей для вычисления вовлеченности в деятельность коллективов и начисления рейтинговой стипендии за заслуги, полученные в коллективе. А также предоставляет информацию о всех коллективах и возможностью вступить в один из них на странице, изучив информацию о нем.

## **Описание проблемы**

На сегодняшний день, не существует удобного и структурированного способа получения всей информации о коллективе. Создание данного программного продукта позволит в одном месте получить всю информацию о коллективах, расписании мероприятий и фото с прошедших мероприятий и выбрать для себя интересующий.

## 

## **Цель**

Основной целью данного продукта является учет вовлеченности студентов в деятельность коллективов.

## **Постановка задачи**

Программный продукт должен содержать базу всех коллективов с полной и структурированной информацией о них, такой как: краткая информация о деятельности коллектива, расписание мероприятий, фото, данные о руководителе и описание прошедших мероприятий. На главной странице будет предоставлена основная информация о коллективе, такая как: контактные данные, ознакомительная информация и фотографии коллектива. Так же будет вкладка с расписанием, на которой можно будет узнать о датах проведения тренировок, репетиций и мероприятий и вкладка с новостями коллектива, где будет краткая информация о прошедших мероприятиях. Дать руководителю проекта возможность добавлять фотографии, редактировать новости и расписание.

## **Словари**

* + 1. **Словарь предметной области**

|  |  |  |
| --- | --- | --- |
| **№** | **Сущность** | **Описание** |
| 1 | Коллектив | Студенческое объединение, которое занимается какой-либо деятельностью и имеет определенную цель. Коллектив должен подать заявку на регистрацию и быть одобрен ИрНИТУ. Коллектив обязан иметь руководителя. |
| 2 | Мероприятие | Событие, для которого необходимы участники и которое является одним из способов получения баллов. Мероприятие характеризуется конкурсной основой и/или концертной деятельностью, которое, как правило, проходит много раз. Мероприятия проводятся только от лица ИрНИТУ. |
| 3 | Уровень мероприятия | Уровень субъекта, на базе которого проводится мероприятие. |
| 4 | Новости | Актуальная информация о прошедших мероприятиях и какие-то локальные события внутри коллектива. |
| 5 | Фотоотчет | Фотографии с прошедших мероприятий, организованных коллективом или в которых было принято участие. |
| 6 | Требования | Краткий список требований для вступления в коллектив, таких как: наличие КМС, танцевальные или вокальные данные. |
| 7 | Местонахождение | Информация о нахождении коллектива, как правило аудитория, где базируется коллектив. |
| 8 | Расписание | Даты и места проведения мероприятий в которых принимает участие коллектив или организует. |
| 9 | Название | Наименование проекта |
| 10 | Редактирование | Изменение или обновление информации о коллективе. |

* + 1. **Словарь по Абботу**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Сущность** | **Глагол** | **Пояснение глагола** |
| 1 | Коллектив | Выбирать, показывать, искать |  |
| 2 | Мероприятие | Показывать |  |
| 3 | Уровень мероприятия | Показывать |  |
| 4 | Новости | Показывать, редактировать |  |
| 5 | Фотоотчет | Показывать, редактировать |  |
| 6 | Требования | Показывать, распределять |  |
| 7 | Местонахождение | Показывать |  |
| 8 | Расписание | Показывать, редактировать |  |
| 9 | Название | Показывать |  |
| 10 | Редактирование |  |  |

* + 1. **Объектно-ориентированный словарь**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Сущность** | **Данные** | **Методы** |
| 1 | Коллектив | Название, деятельность коллектива, проводимые мероприятия, руководитель коллектива, новости, фото, требования, местонахождение, расписание. | Выбирать, смотреть, искать |
| 2 | Мероприятие | События проводимые коллективом или в которых коллектив принимает участие | Посещать, Показывать |
| 3 | Уровень мероприятия | Уровень субъекта, на базе которого проводится мероприятие.  Они бывают:   1. Вузовский; 2. Городской; 3. Областной; 4. Межрегиональный; 5. Всероссийский; 6. Международный. |  |
| 4 | Новости | Информация о мероприятиях и локальных событиях. | Показывать, редактировать |
| 5 | Фотоотчет | Фотографии с прошедших мероприятий | Показывать, редактировать |
| 6 | Требования | Требования для вступления в коллектив | Показывать, распределять |
| 7 | Местонахождение | Местонахождение коллектива в ИрНИТУ | Показывать |
| 8 | Расписание | Расписание мероприятий коллектива | Показывать, редактировать |
| 9 | Название | Название коллектива | Показывать |
| 10 | Редактирование |  |  |

## **Функциональность ПО**

* + 1. **Просмотр информации о коллективе**

На странице должна содержаться информация: какие мероприятия проводит коллектив, контактные данные коллектива и местонахождение, и актуальные новости. Так же фотографии с мероприятий для дополнительной визуальной информации. На странице будет находится несколько вкладок, отвечающих за предоставление той или иной информации.  
Главная страница предоставляет основную информацию о коллективе и фотографии с коллектива, на ней же и расположена кнопка с возможностью подать заявку. Так же вкладка с расписанием, на которой отображаются все мероприятия, проводимые коллективом, и вкладка с новостями, на которой можно получить все основные новости о событиях, в которых принимал участие коллектив.

* + 1. **Загрузка фотографий в альбом.**

При загрузке фотографий ей автоматически присваивается идентификационный номер на основе даты добавления, после этого новые фотографии помещаются в альбом.

* + 1. **Предоставление фотографий на странице.**

На странице находятся фотографии, фотографии подгружаются на страницу из сформированных альбомов. Фотографии отображаются в отдельном блоке «карусель», где их можно просмотреть.

* + 1. **Формирование архивов с фотографиями.**

При добавлении новых фотографий в систему, каждой из них автоматически присваивается уникальный номер на основе даты добавления. При обновлении информации о фотографии, предыдущие версии сохраняются в архиве. Таким образом, возможно формирование архивов с фотографиями

* + 1. **Формирование новостей.**

Блок с новостями формирует и редактирует руководитель. В новостном блоке есть возможность добавить новости в блок, а предыдущие новости уходят в архив.

* + 1. **Показать новости.**

**Необходимо создать отдельную страницу, где будут публиковаться** новости. На этой странице будет список последних трёх новостей, созданных руководителем. Для каждой новости будет краткое описание события и прикреплённая фотография. Фотографии так же подгружаются из альбомов. При добавлении новой новости, она отображается на странице, заменяя.

# **Анализ функциональных требований**



## **Модель предметной области (IDEF0)**

Данная модель IDEF0 описывает процесс «Предоставление информации о коллективе».

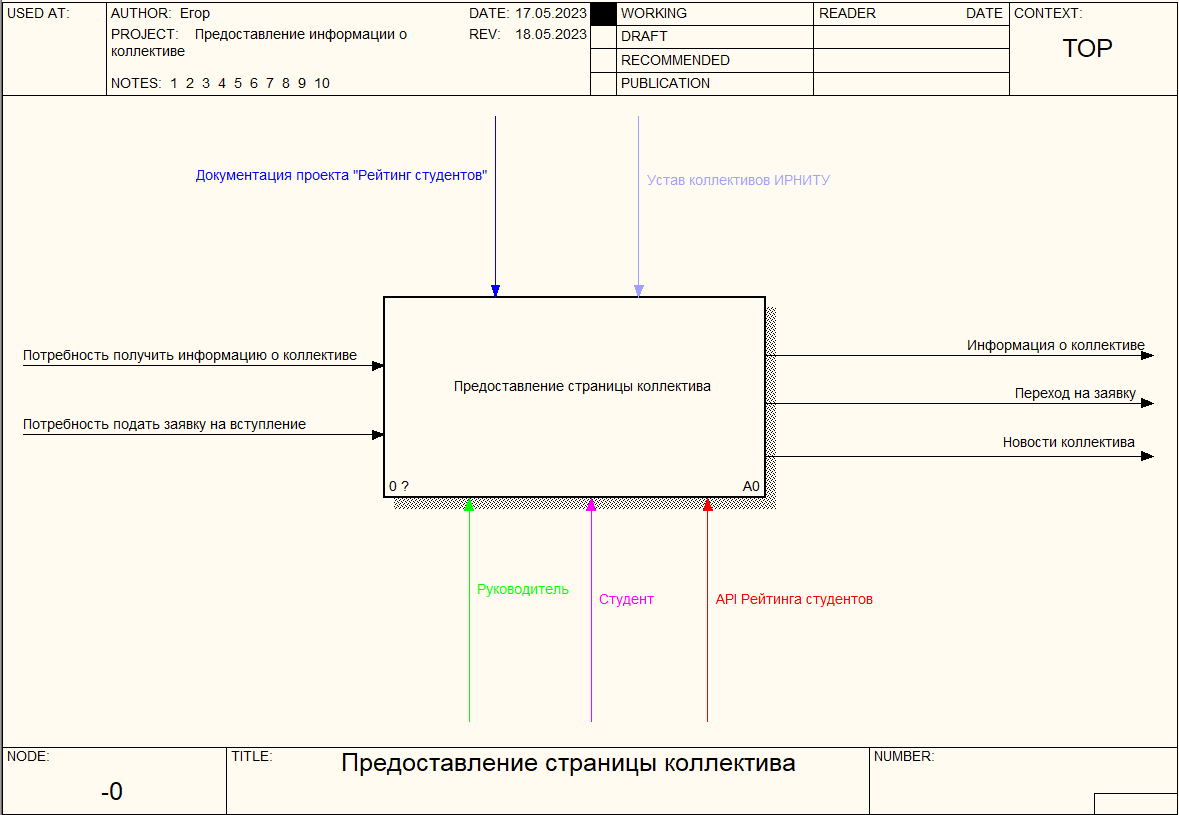


Рисунок 1 – Модель IDEF0. Уровень А-0

На вход данный процесс поступает потребность получить информацию о коллективе и потребность подать заявку на вступление. Ресурсами являются студент, руководитель и API рейтинга студентов. API позволяет работать с базой данных проекта.

Состоит из трёх подпроцессов: А1 «Предоставить главную страницу», А2 «Предоставить страницу с новостями», А3 «добавить фотографии». Данная детализация изображена на рисунке 2.

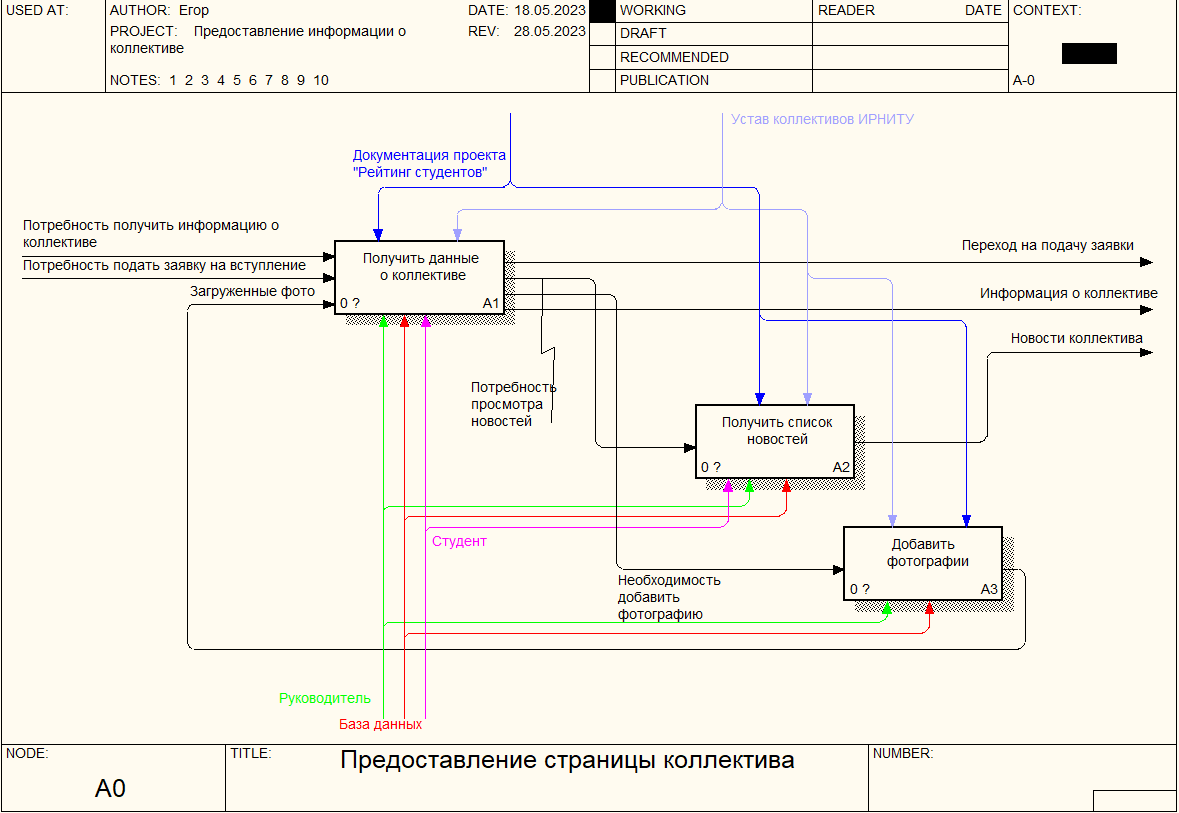


Рисунок 2 – Модель IDEF0. Уровень А0

Процесс А1 «Предоставить главную страницу» имеет детализацию, представленную на рисунке 3.

Этот процесс обусловлен потребностью предоставить главную страницу коллектива с основной информацией.

Процесс А2 «Предоставить страницу с новостями» имеет детализацию, представленную на рисунке 4.

Процесс А3 «Добавить фотографии» имеет детализацию, представленную на рисунке 5.

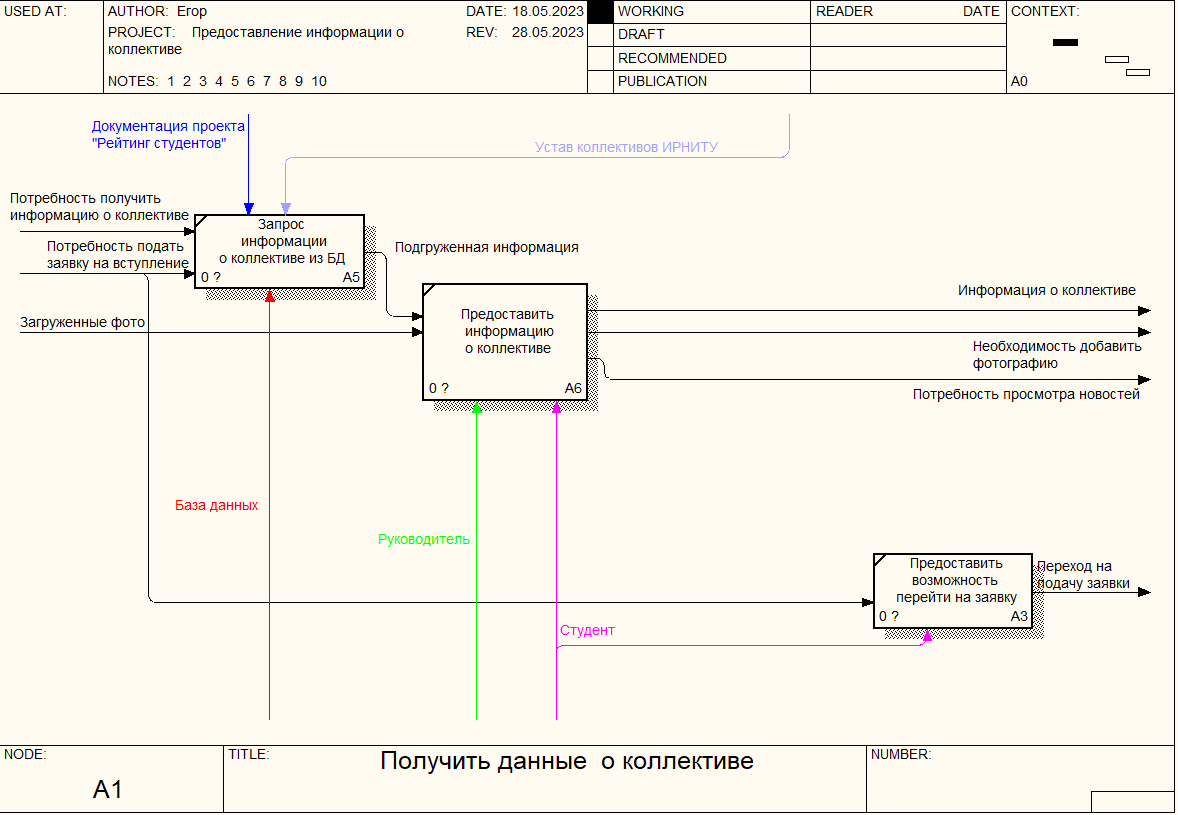


Рисунок 3 – Модель IDEF0. Уровень А1

На главной странице представлена основная информация о коллективе, которая подгружается из с помощь Api. Так же предоставляется возможность перейти на форму с оформлением заявки на вступление в коллектив и возможность перейти на странницу новостей.

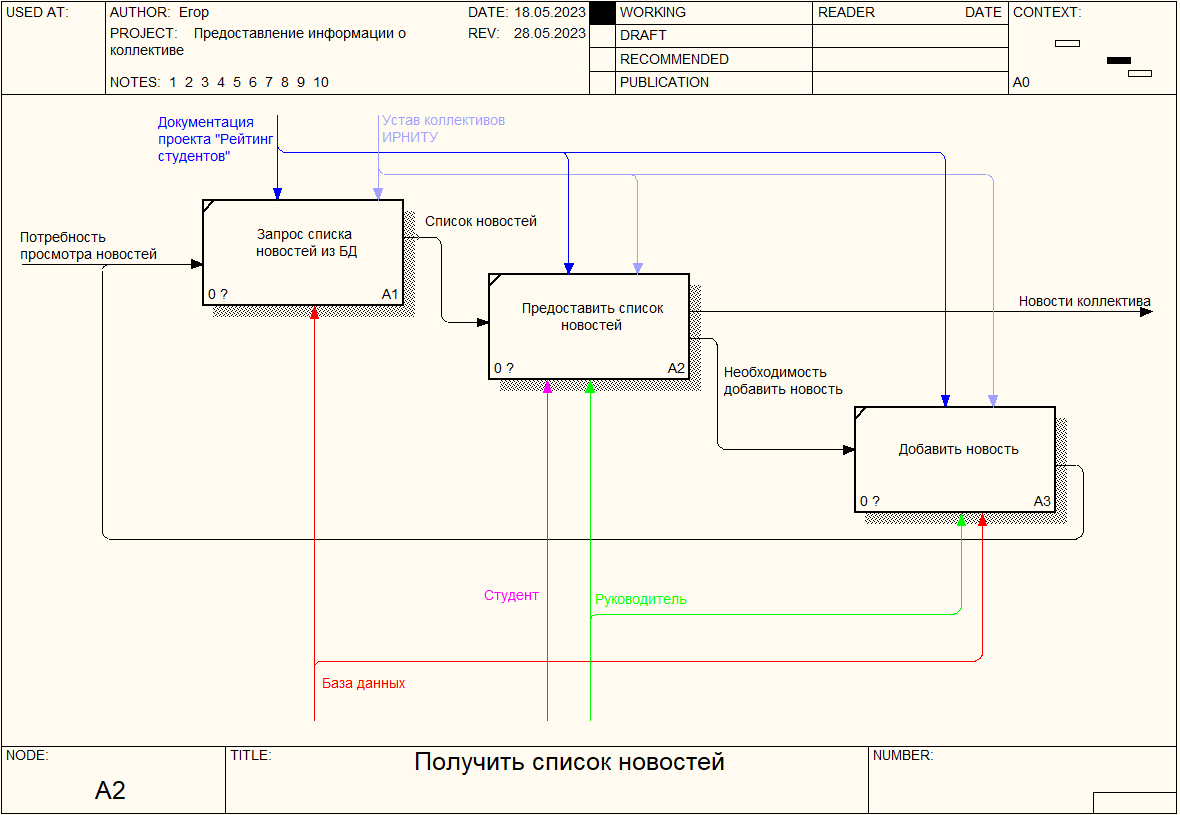


Рисунок 4 – Модель IDEF0. Уровень А2.

Процесс обусловлен потребностью просмотре и редактировании страницы с новостями.

Список новостей подгружается из API и потом отображается на главной странице. При необходимости руководитель может добавить новость и страница обновится с новой новостью в списке.

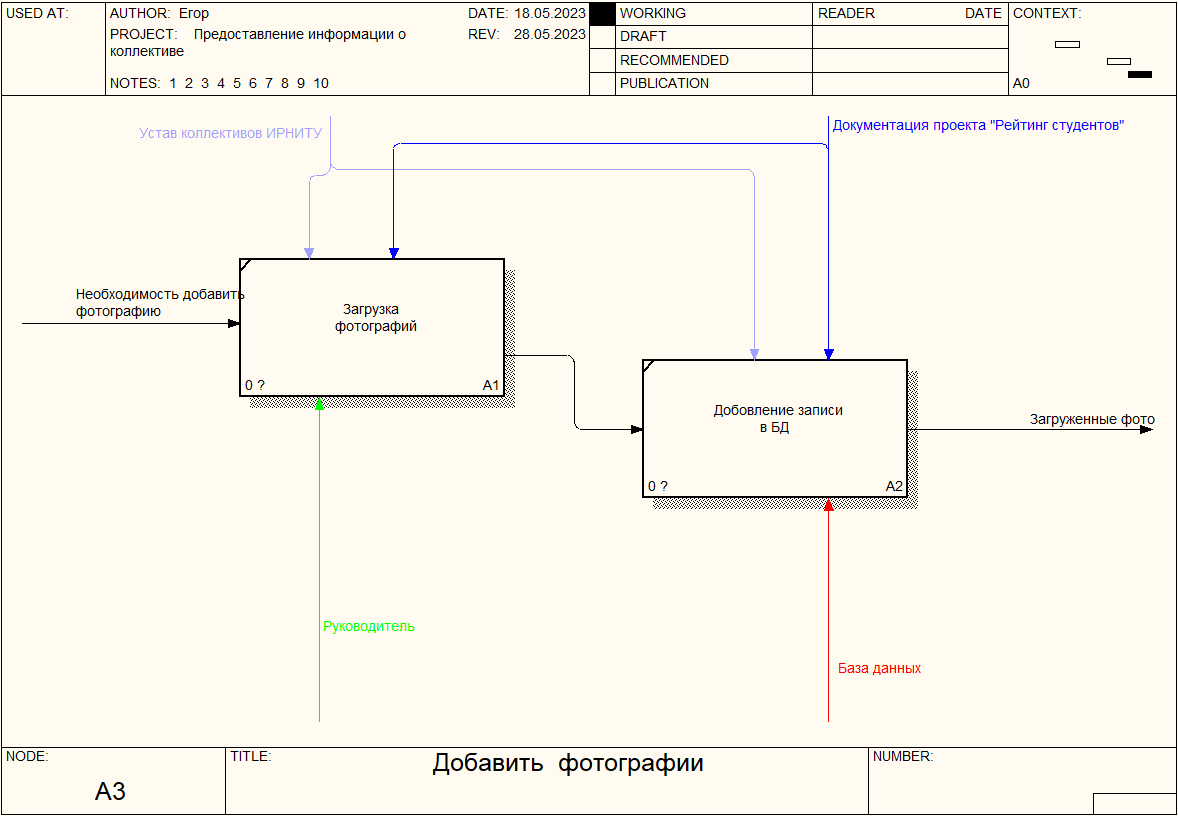


Рисунок 5 – Модель IDEF0. Уровень А3

Процесс обусловлен потребностью в загрузке фотографий в карусель.

Руководитель может добавить фотографию на главной странице и загрузить ее в карусель, так же фотография добавиться в базу.

## **Варианты использования UML**

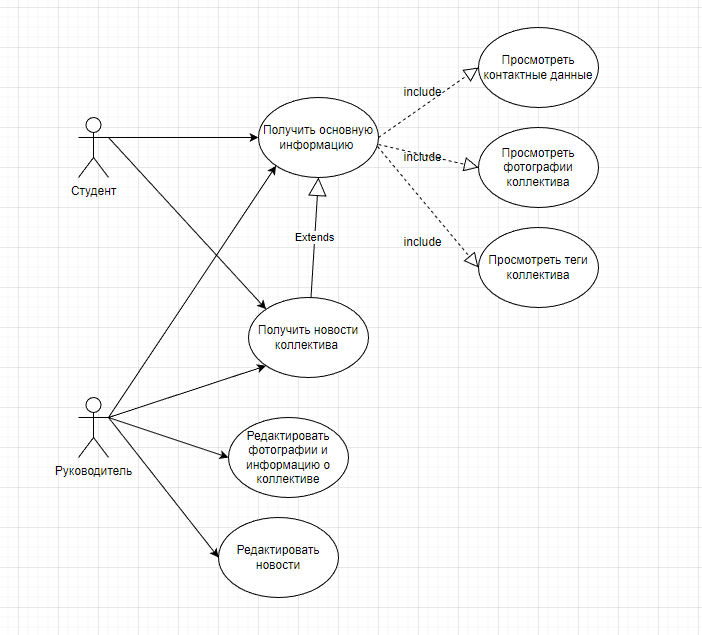


Рисунок 6 – Модель use-case

Пользователи могут выполнять функции, которые отображены на рисунке 6.

Пользователем может являться, как студент, так и руководитель. Все те лица, кто сейчас посещает страницу коллектива.

Вариант использования «Получить основную информацию» включает в себя варианты использования «Посмотреть теги коллектива» и «Просмотреть контактные данные», «Просмотреть фотографии коллектива».

Руководитель так же может редактировать фотографии, информацию и новости.

## **DFD - диаграмма потоков данных**

На диаграмме была выделена сущность «студент». Процесс описывает потребности студента в получении информации о коллективе на странице коллектива. Процесс «Предоставление страницы коллектива» предоставляет студенту информацию о коллективе, новости коллектива и возможность перейти на заявку на вступление.

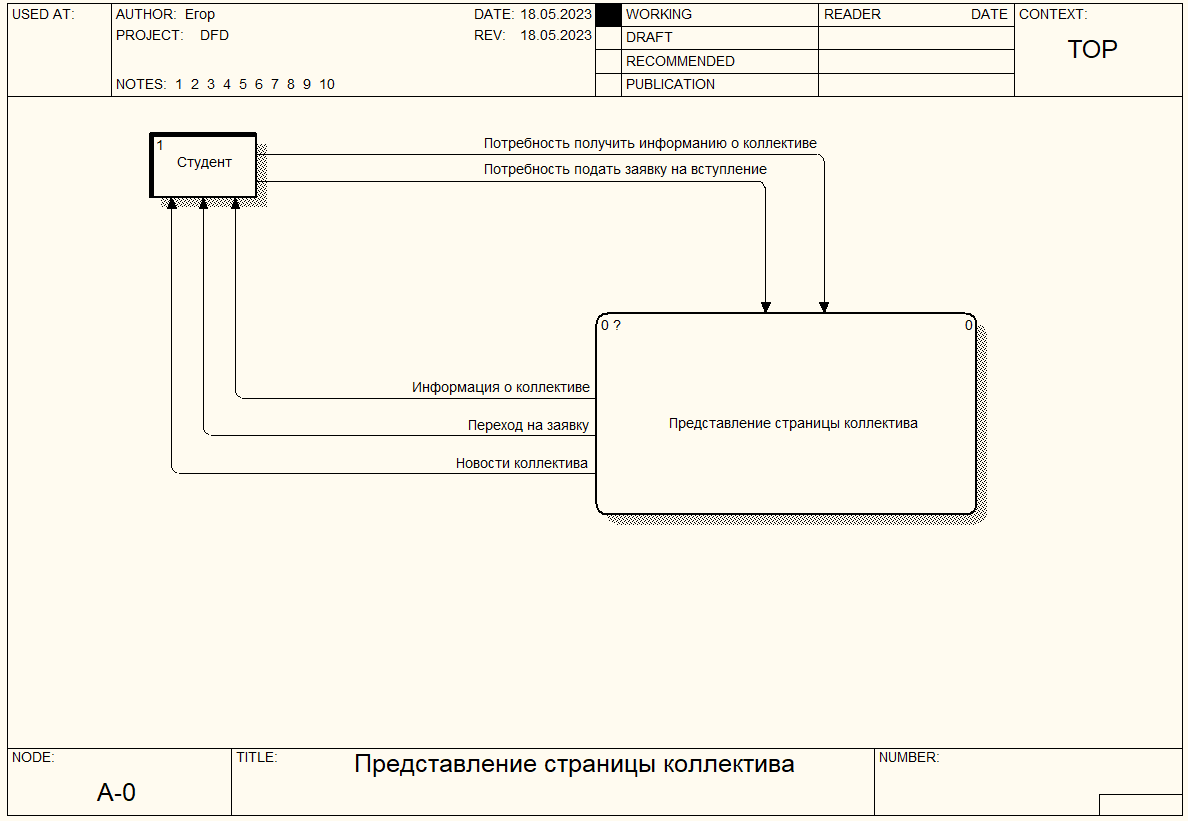


Рисунок 7 – диаграмма поток данных

На рисунке 7 представлена детализация уровня A0, где выявлены 3 основных процесса.

1. «Предоставить главную страницу»

* Информация о коллективе – сущность хранилища данных «Коллективы»

1. «Добавить фотографии»

* Новое фото – сущность хранилища данных «Коллективы»

1. «Предоставить страницу с новостями»

* Список новостей – сущность хранилища данных «Новости»
* Новая новость - сущность хранилища данных «Новости»

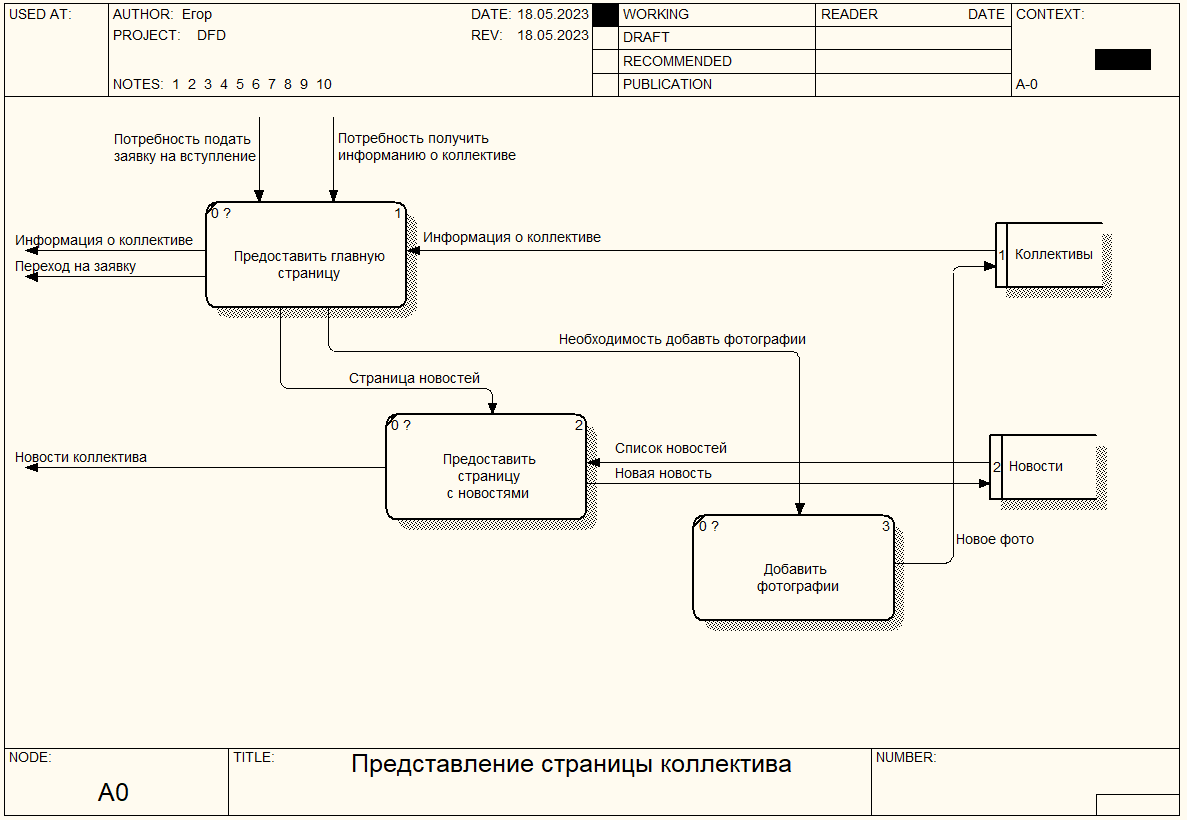


Рисунок 8 – диаграмма поток данных

## **Концептуальная модель хранилища данных**

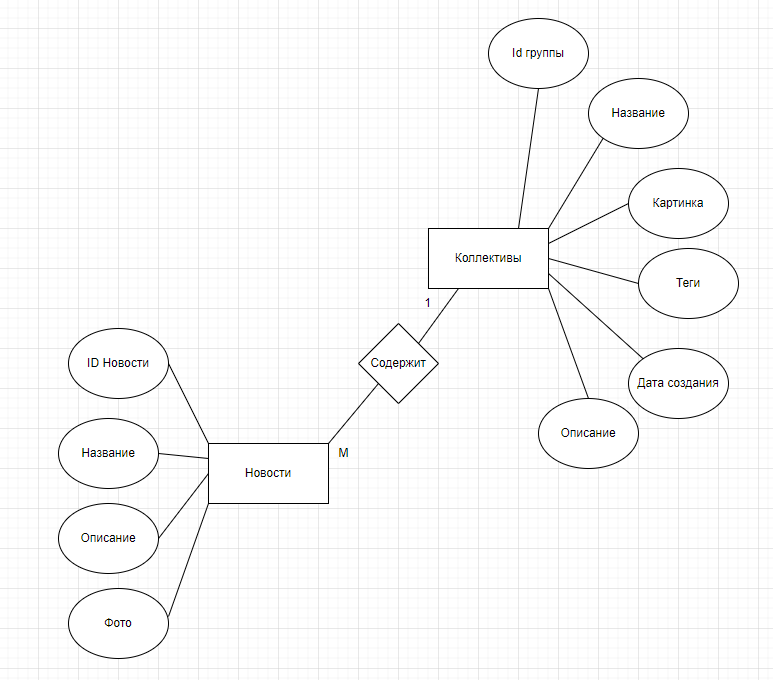


Рисунок 9 – концептуальная модель хранилища данных

## **Логическая модель БД**

На рисунке предоставлена логическая модель базы данных.

Так как данный модуль полностью посвящен странице коллектива, то самой главной сущностью является «коллектив».

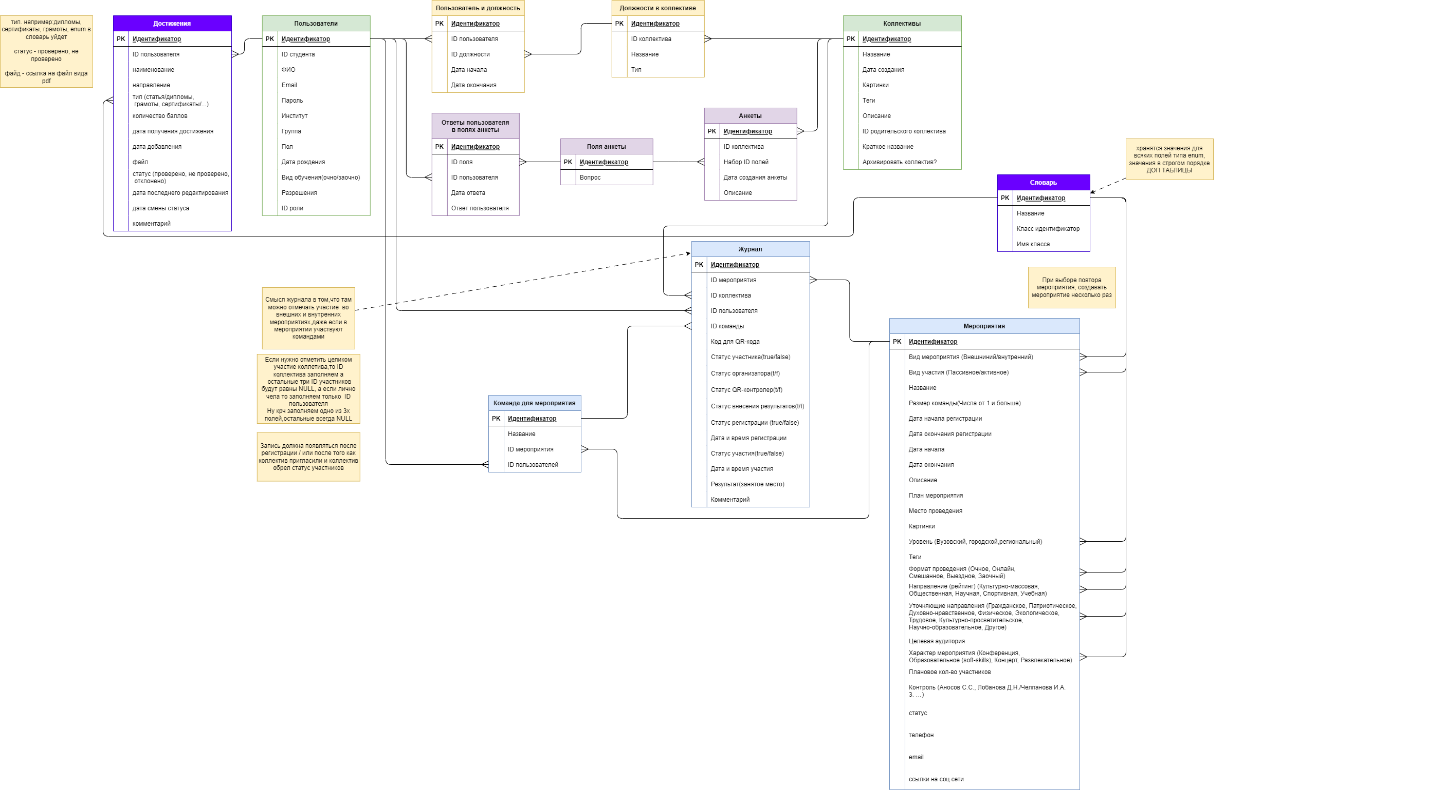
Коллектив имеет: название, дата создания, картинки (например, иконка коллектива), теги (ключевые слова, по которым можно найти определенный коллектив в общем списке коллективов), описание коллектива, id родительского класса – направления не вынесены в отдельную таблицу, т.к. их всего 5 и меняться они не будут, поэтому, чтобы определить к какому направлению относится коллектив, существует эта метка, краткое название коллектива, также коллектива может быть архивирован. 

Рисунок 10 – Логическая модель БД

## **Физическая модель БД**

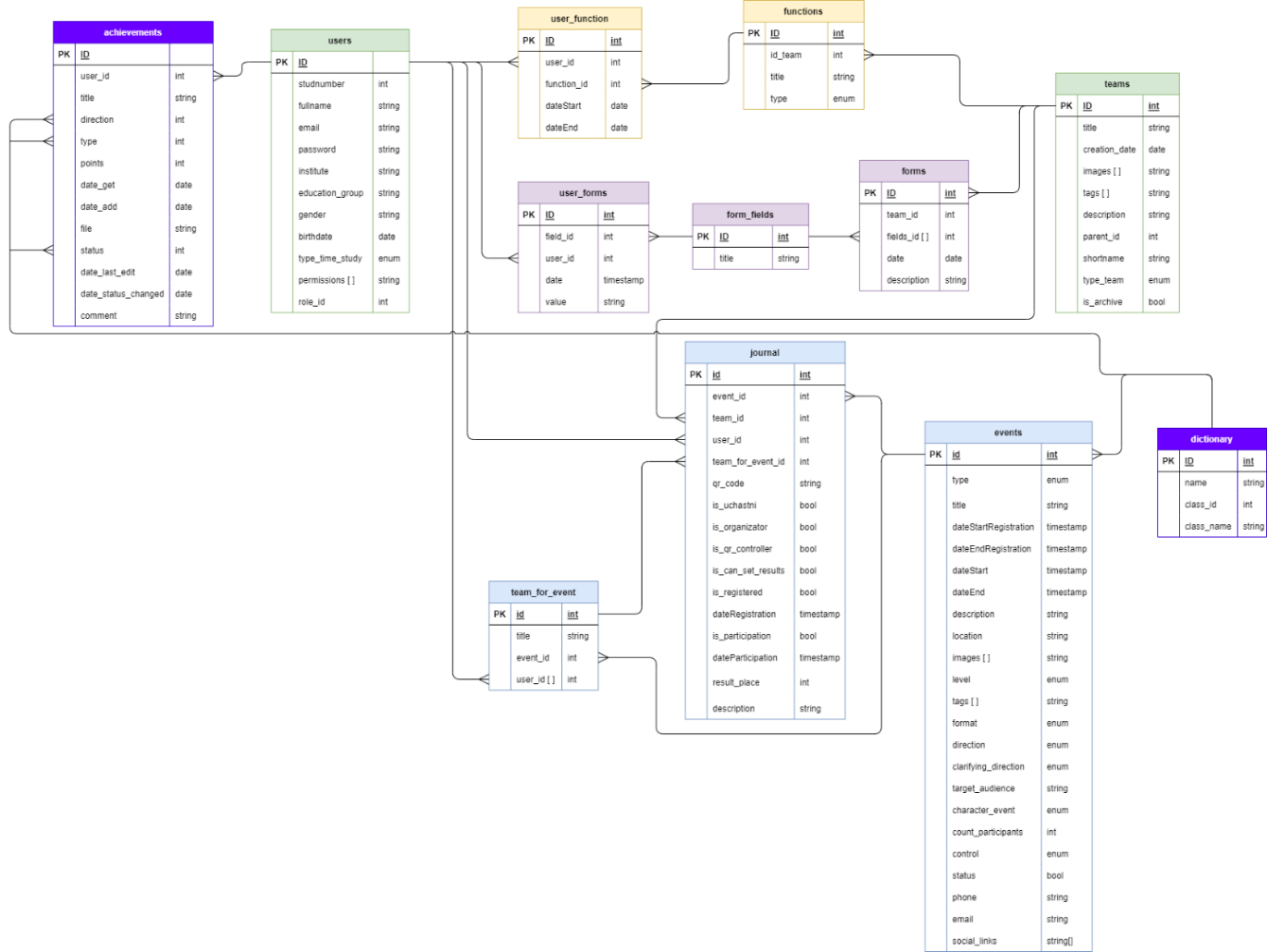


Рисунок 11 – Физическая модель БД

## **Варианты использования**

* + 1. Вариант использования: «Получить основную информацию о коллективе»

**Цель:** получить информацию о коллективе.

**Активные субъекты:** Пользователи.

**Краткое описание:** Активный субъект может получить основную информацию о коллективе.

**Предусловие:** нет.

**Постусловия:** возможность перейти на заявку и возможность перейти на новости.

**Основной поток событий:**

1. Система предоставляет информацию о коллективе.
2. Активный субъект просматривает информацию о коллективе.
3. Система завершает работу варианта использования.

**Альтернативные потоки событий:** нет.

**Сценарий обработки ошибок:** нет.

**Специальные требования:** нет.

**Дополнительные замечания:** нет.



Рисунок 12 – Диаграмма последовательностей варианта использований «Получить основную информацию о коллективе»

* + 1. Вариант использования: «Редактировать фотографии и информацию о коллективе»

**Цель:** редактировать информацию о коллективе.

**Активные субъекты:** Пользователи.

**Краткое описание:** Активный субъект может редактировать основную информацию о коллективе и фотографии.

**Предусловие:** должен быть выполнен вариант использования «Получить основную информацию о коллективе»

**Постусловия:** должен быть выполнен вариант использования «Получить основную информацию о коллективе»

**Основной поток событий:**

1. Система публикует форму с информацией о коллективе
2. Активный субъект редактирует форму.
3. Активный субъект нажимает на кнопку для редактирования информации.
4. Система формирует обновленные данные.
5. Система завершает работу варианта использования.

**Альтернативные потоки событий:**

1. если активный субъект на шаге 2 передумал редактировать заявку, то система возвращается к варианту использования «Получить основную информацию о коллективе».

**Сценарий обработки ошибок:** нет.

**Специальные требования:** Пользователь должен быть руководителем.

**Дополнительные замечания:** нет.



Рисунок 13 – Диаграмма последовательностей варианта использований «Редактировать фотографии и информацию о коллективе»

* + 1. Вариант использования «Получить новости коллектива»

**Цель:** получить новости коллектива.

**Активные субъекты:** Пользователи.

**Краткое описание:** Активный субъект может получить новости коллектива.

**Предусловие:** должен быть выполнен вариант использования «Получить основную информацию о коллективе»

**Постусловия:** нет.

**Основной поток событий:**

1. Система предоставляет новости коллектива.
2. Активный субъект просматривает новости коллектива.
3. Система завершает работу варианта использования.

**Альтернативные потоки событий:** нет.

**Сценарий обработки ошибок:** нет.

**Специальные требования:** нет.

**Дополнительные замечания:** нет.



Рисунок 14 – Диаграмма последовательностей варианта использований «Получить новости коллектива»

* + 1. Вариант использования «Редактировать новости»

**Цель:** редактировать новости коллектива.

**Активные субъекты:** Пользователи.

**Краткое описание:** Активный субъект может редактировать новостной блок коллектива.

**Предусловие:** должен быть выполнен вариант использования «Получить новости коллектива»

**Постусловия:** должен быть выполнен вариант использования «Получить новости коллектива»

**Основной поток событий:**

1. Система публикует форму с новостями коллектива.
2. Активный субъект редактирует форму.
3. Активный субъект нажимает на кнопку для редактирования новостей.
4. Система формирует обновленные данные.
5. Система завершает работу варианта использования.

**Альтернативные потоки событий:**

1. если активный субъект на шаге 2 передумал редактировать заявку, то система возвращается к варианту использования «Получить новости коллектива».

**Сценарий обработки ошибок:** нет.

**Специальные требования:** Пользователь должен быть руководителем.

**Дополнительные замечания:** нет.



Рисунок 15 – Диаграмма последовательностей варианта использований «Редактировать новости»

## **Описание интерфейса**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Страница** | **Функция** | **Элемент интерфейса** | **Описание элемента** | **Ссылка на другую страницу** |
| Навигационная панель | Переход по вкладкам страницы коллектива | Вкладка «главная» |  |  |
| Вкладка «новости» |
|  | Переход к заявке на вступление | Кнопка «Подать заявку» | Кнопка в шапке страницы, позволяющая перейти на заявку на вступление в коллектив. | Заявка на вступление. |
| Вкладка «главная» | Предоставить основную информацию | «Карусель» фотографий | Блок, где отображаются и прокручиваются все фотографии коллектива |  |
| Кнопка  «Добавить фотографию» |  |
| Теги коллектива |  |
|  |  | Текстовое поле с основной информацией о коллективе |  |  |
| Вкладка «Новости» | Предоставить новости коллектива | Карточка новости | Блок, в котором отображается информация о новостях с названием, фотографией и информацией по новости. |  |



Рисунок 17 – Диаграмма интерфейсных классов

# **Проектирование программного продукта**

## **Используемый инструментарий**

Среда разработки Visual Code

Языки программирования: typescript

Разработка макетов интерфейса: Figma

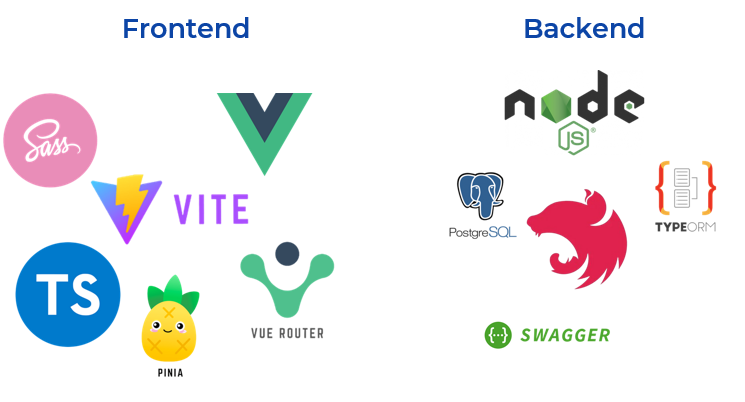


Рисунок 18 – Стек технологий

Node (Node.js) – кроссплатформенная среда исполнения с открытым исходным кодом, которая позволяет разработчикам создавать всевозможные серверные инструменты и приложения используя язык JavaScript.

PostgreSQL — это реляционная база данных с открытым кодом, которая поддерживается в течение 30 лет разработки и является одной из наиболее известных среди всех существующих реляционных баз данных.

## **Фреймворки и библиотеки**

**Бэкенд:**

Nest js – Прогрессивная платформа Node.js для создания эффективных, надежных и масштабируемых серверных приложений.

TypeORM – это простой в использовании ORM для создания новых приложений, подключающихся к базам данных. Функциональность TypeORM – это специфичные для СУБД концепции.

Swagger – это язык описания интерфейса для описания RESTful API, выраженных с помощью JSON.

**Фронтенд:**

Vue – это интерфейсная платформа с открытым кодом для создания пользовательских интерфейсов и одностраничных веб-приложений на JavaScript.

Sass (или Syntactically Awesome Stylesheets) – это скриптовый метаязык (т.е. язык, описывающий другой язык), разработанный для упрощения файлов CSS. Этот модуль входит в Haml (HTML abstraction markup language), который используется для упрощения HTML.

Vite – это инструмент для сборки frontend-проектов на VueJS, который значительно упрощает разработку.

Pinia – легковесная библиотека управления состояниями для Vue.js.

Vue-Router – это пакет JavaScript, который позволяет настроить маршрутизацию для одностраничных приложений (SPA).

TypeScript – это свободный язык программирования с открытым исходным кодом.Он является строгим синтаксическим супермножеством JavaScript и добавляет в язык необязательную статическую типизацию.Он предназначен для разработки больших приложений и транспонируется на JavaScript.

## **Описание архитектуры разрабатываемого ПО**

* + 1. **Модель развертывания**

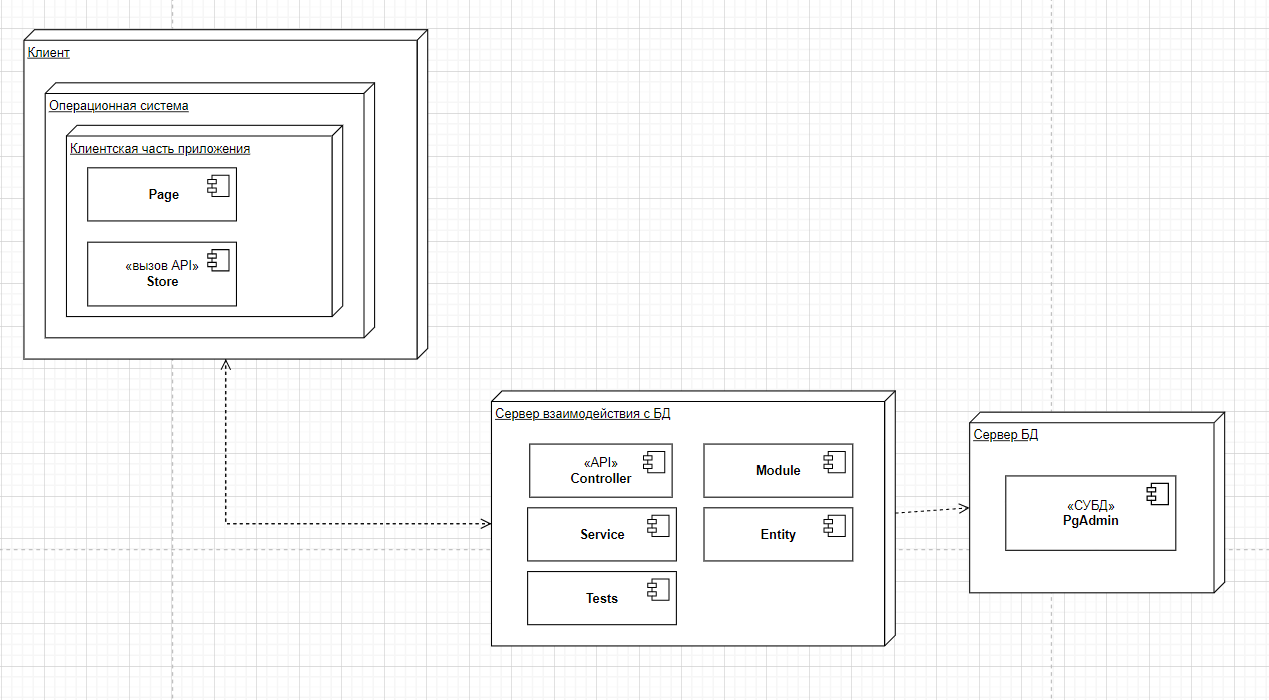


Рисунок 19 – Модель развертывания

В данной системе предполагается наличие серверной и клиентской части приложения. Серверная часть взаимодействует с БД PostgreSQL и включает в себя контроллер, сервис, модуль, сущности и тесты.

* + 1. **Фактическая организация модулей системы**

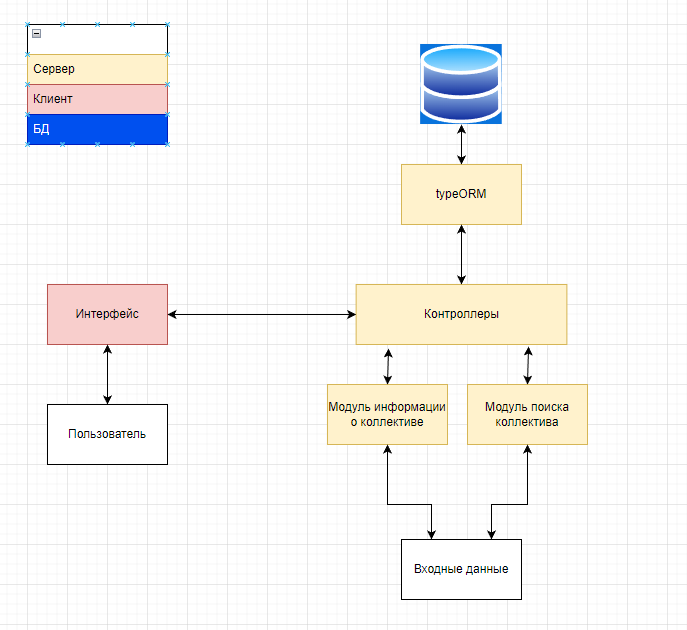
****

Рисунок 20 – Модули системы

Модули были созданы на основе функциональных требований к системе

* + 1. **Подробное описание архитектуры приложения**

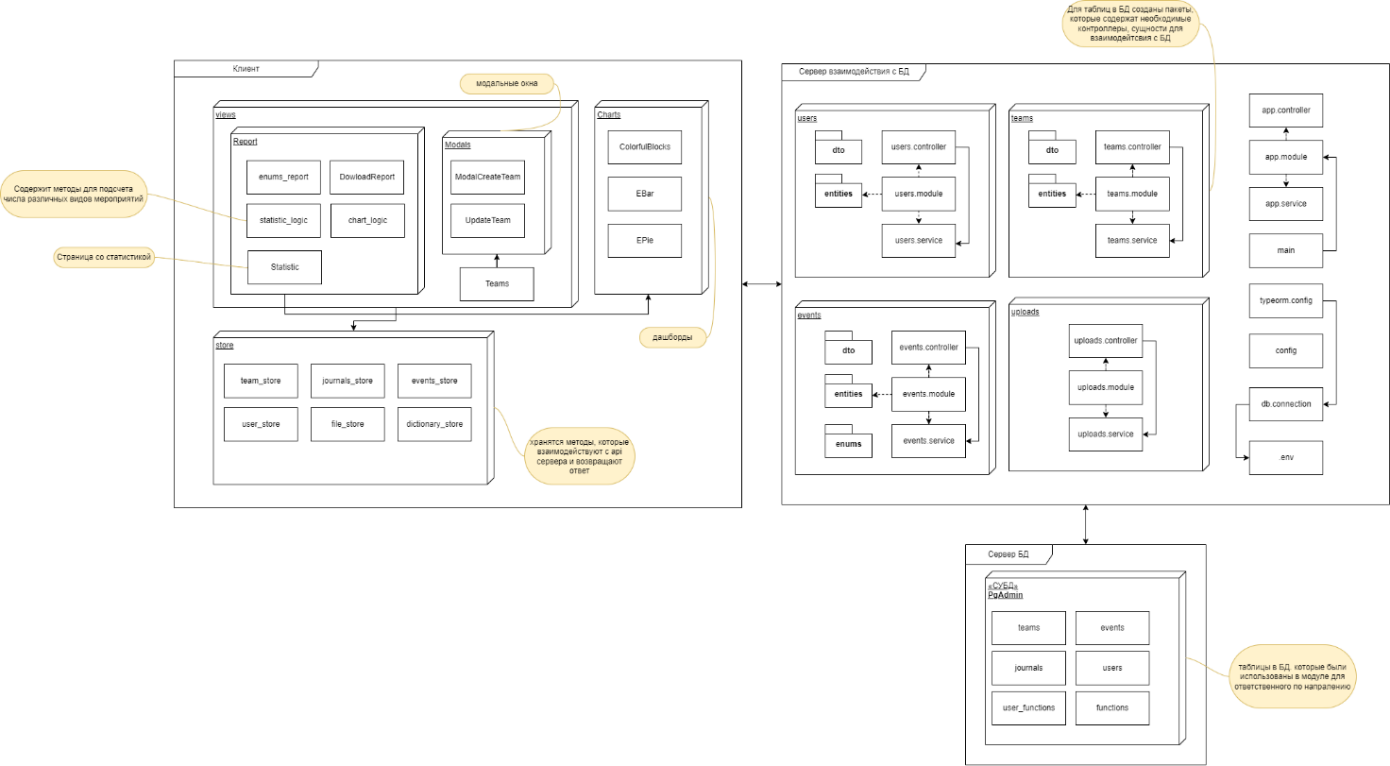


Рисунок 21 – Архитектура приложения

Взаимодействие показано в соответствии с диаграммой развертывания.

* + 1. **Диаграмма состояний**



Рисунок 22 – Диаграмма состояний

# **Пользовательский интерфейс**

С помощью инструмента Figma были разработаны макеты вкладок страницы коллектива.

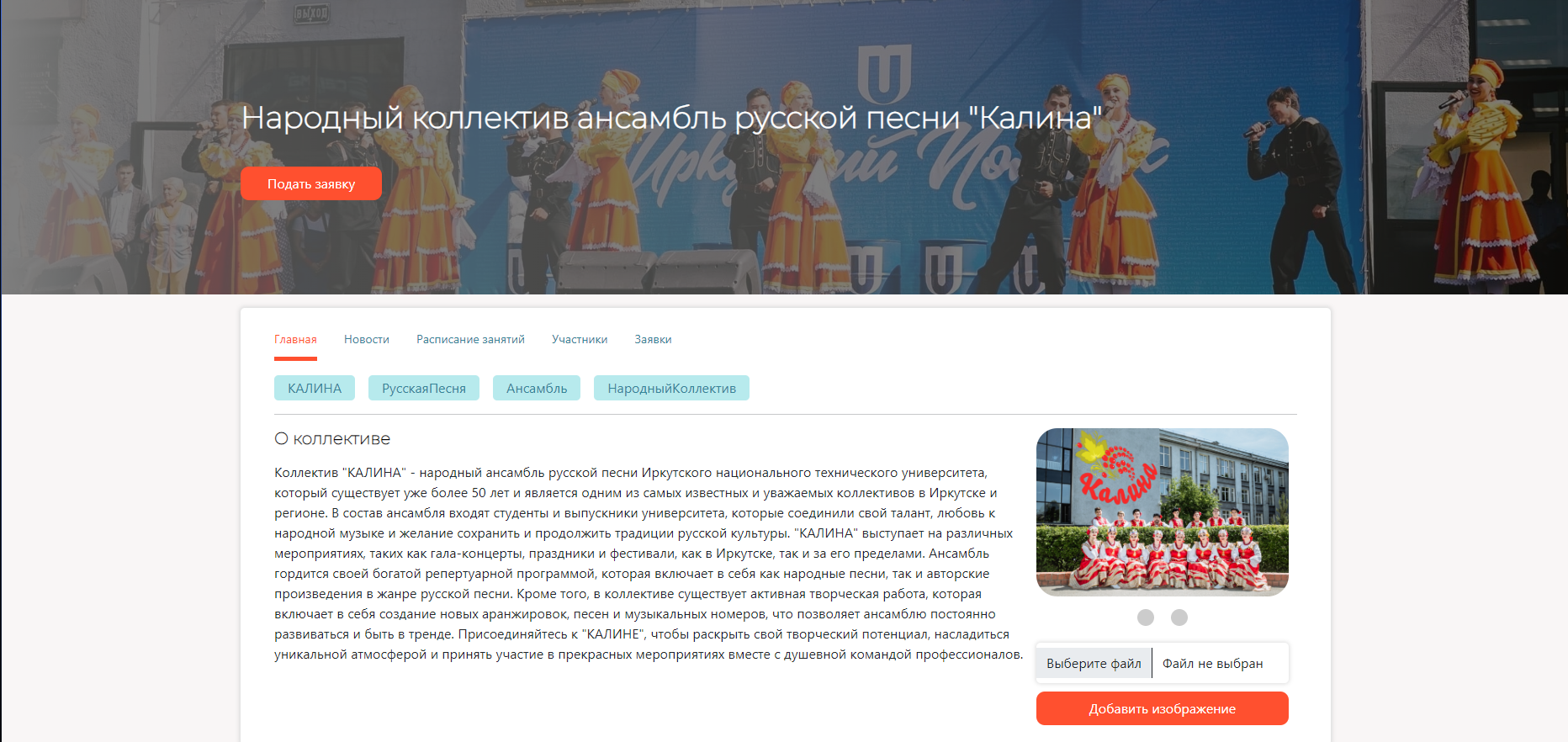


Рисунок 23 – Главная страница

На данной странице показана основная информация о коллективе и фотографии. У руководителя есть возможность добавить фотографии в карусель.

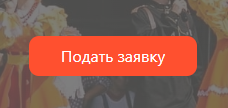


Рисунок 24 – Кнопка «Подать заявку»  
По нажатию на данную кнопку происходит перенаправление на заявку на

вступление в коллектив.

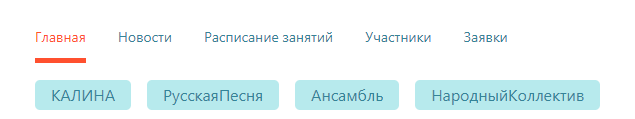


Рисунок 25 – Навигационная панель и теги коллектива

С помощью навигационной панели осуществляется переход по вкладкам на странице коллектива

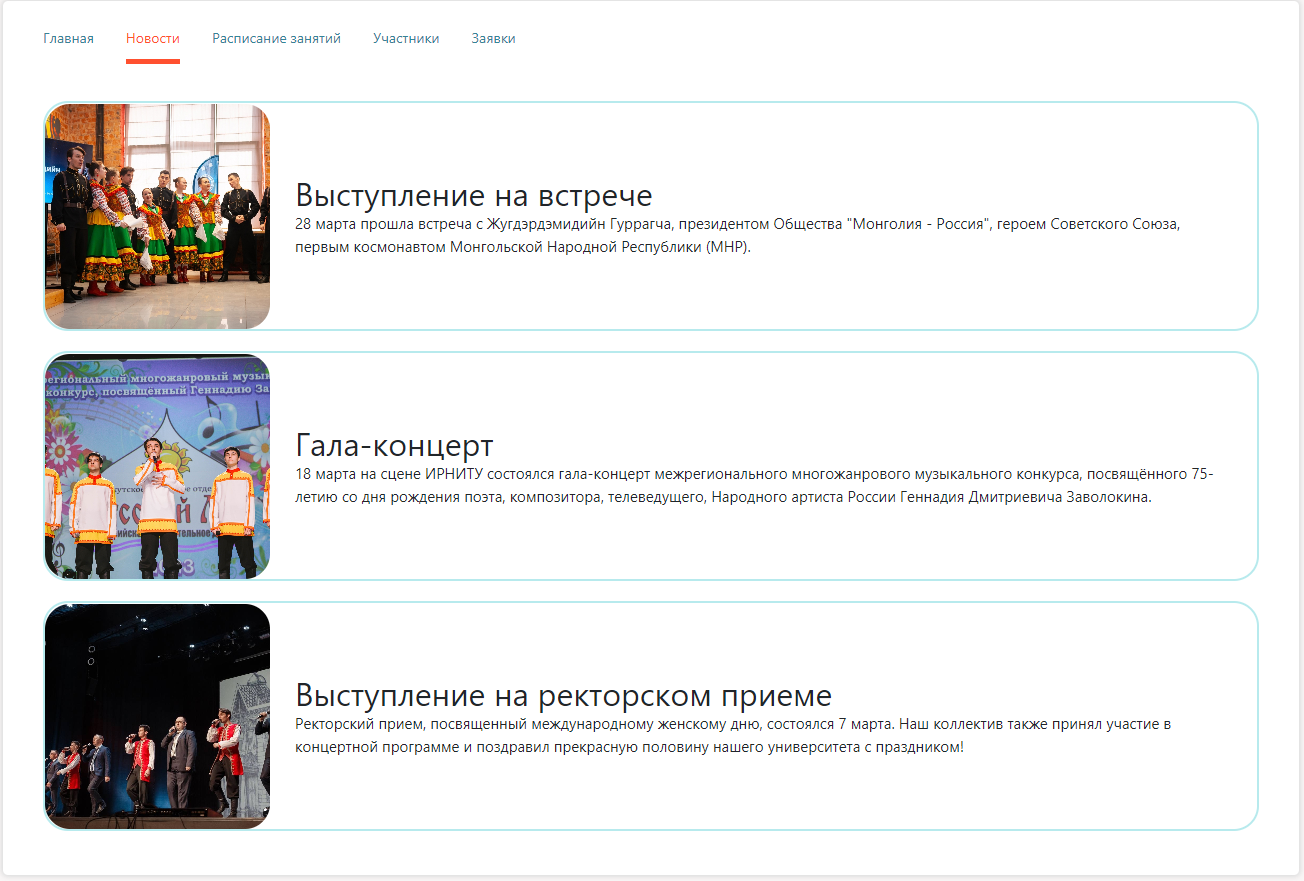


Рисунок 26 – Вкладка новостей коллектива

При нажатии на вкладку «Новости» будет открыта страница с новостями коллектива. На данной странице пользователь может ознакомиться с новостями коллектива.

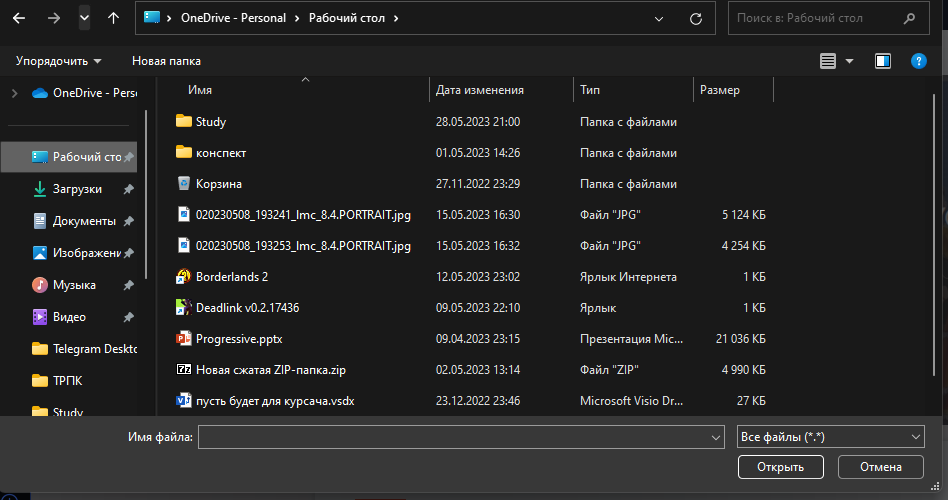


Рисунок 27 – Кнопка «Добавить изображение»

По нажатию на данную кнопку открывается проводник и с устройства можно добавить фотографию в карусель.

# **Программная реализация**

## **Таблица спецификаций класса SearchTeamDto**

Данный класс предназначен для получения нужного коллектива

Таблица 4.2– Переменные класса

|  |  |  |
| --- | --- | --- |
| **Имя** | **Назначение** | **Тип/Диапазон** |
| title | Название коллектива | String |
| description | Описание коллектива | String |
| tags | Теги | String |

## **Таблица спецификаций класса TeamsController**

Api для получения коллективов

Таблица 4.4 – Спецификация методов

|  |  |  |  |
| --- | --- | --- | --- |
| **Имя** | **Назначение** | **Тип** | **Параметры** |
| findAll | Получение списка коллективов с их руководителями | Team[] | @Query() params: SearchTeamDto |
| findAllTeamsOfDirection | Получение списка коллективов c учетом параметров (направление, вид) | [Team[], number] | @Query() params |
| changeArchiveTeam | Архивировать коллектив или наоборот | UpdateResult | @Param("id") id, @Body() data |
| update | Обновить коллектив (ответственный по направлению) | Team | @Param('id') id: number, @UploadedFiles() files, @Body() updateTeamDto: UpdateTeamDto |
| create | Создать новый коллектив (ответственный по направлению) | @UploadedFiles() files, @Body() createTeamDto: CreateTeamDto | @UploadedFiles() files, @Body() createTeamDto: CreateTeamDto |

# **Заключение**

В результате курсовой работы был разработан модуль для сайта «Рейтинг студентов ИРНИТУ». Была спроектирована страница коллективов с основной информацией и новостями коллектива.

Был проведен анализ предметной области, поставлена и описана задача создаваемой системы. Проведено проектирование программного продукта.

В процессе разработки данного модуля были получены навыки анализа предметной области, проектирования системы для данного программного продукта и умение правильно ставить цели для достижения поставленных задач системы.

Страница спроектирована с дальнейшей возможностью развития, добавления вкладок и расширением функционала для руководителя.