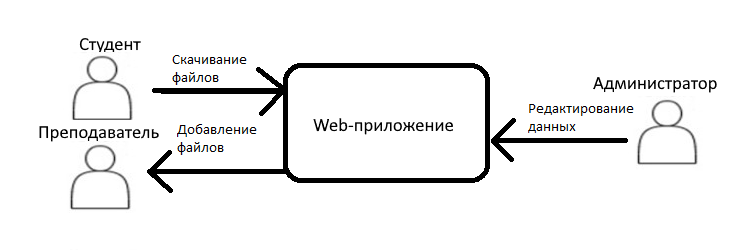
**Глава 3. Проектирование программного средства**

**3.1 Введение**

Проектирование программного средства – это последний этап разработки ПО, контролируемый заказчиком. В нем утверждаются все подсистемы системы, потоки данных между подсистемами, основные функции и подфункции пользователей. Также прогнозируется результат ответа системы в любой момент работы. Описывается связь объектов системы. Процесс проектирования сопровождается построением разных диаграмм, отражающих пользовательское восприятие системы и поясняющих разработчику поставленные перед ним задачи.

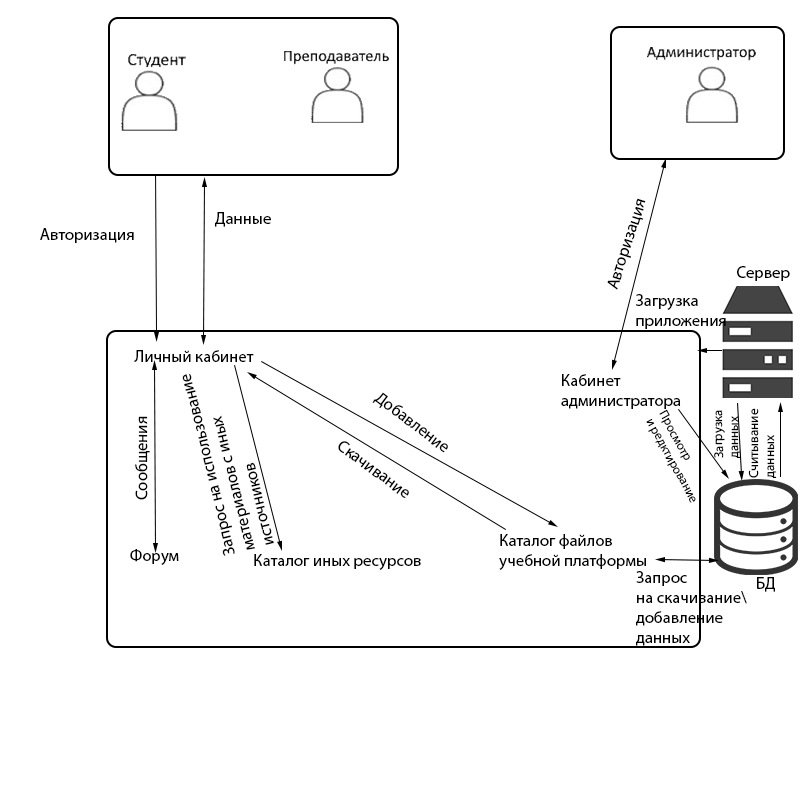
На основании анализа предметной области и маркетингового исследования необходимо построить архитектурно-контекстную диаграмму системы, диаграмму прецедентов, диаграмму потоков данных, диаграмму связи объектов, диаграмму перехода из состояния в состояние, диаграмму классов. Также необходимо протестировать пользовательские функции, выявленные в ходе построения диаграммы прецедентов. 

1) Web-приложение – сайт, обеспечивающая взаимосвязь студентов, пруодавателей и администартора.

2) Студент –участник учебного процесса, пользователь образовательной платформы.

3) Преподаватель – сотрудник ИМиКН, участник учебного процесса, пользователь образовательной платформы.

4) Системный администратор – человек, осуществляющий контроль за файлообменной системой, приоритетный пользователь образовательной платформы.

**4.3 Архитектурная диаграмма потоков**

Основными пользователями приложения являются студенты и преподаватели. Основной функцией является связь преподавателей и студентов, а так же обмен данными: добавление и скачивание полезных источников литературы. Прежде чем начать работу, пользователь должен пройти процедуру авторизации.

Пользователь вводит свой логин и пароль. После попадает в личный кабинет, где может редактировать личные данные. Из личного кабинета пользователь может добавлять и скачивать файлы, общаться на форуме (добавлять сообщение и читать сообщения других пользователей), искать необходимые источники информации в Интернете. Для последнего существует специализированный каталог иных источников, который пополняется пользователями. Данный каталог содержит ссылки на источники литературы в сети Интернет.

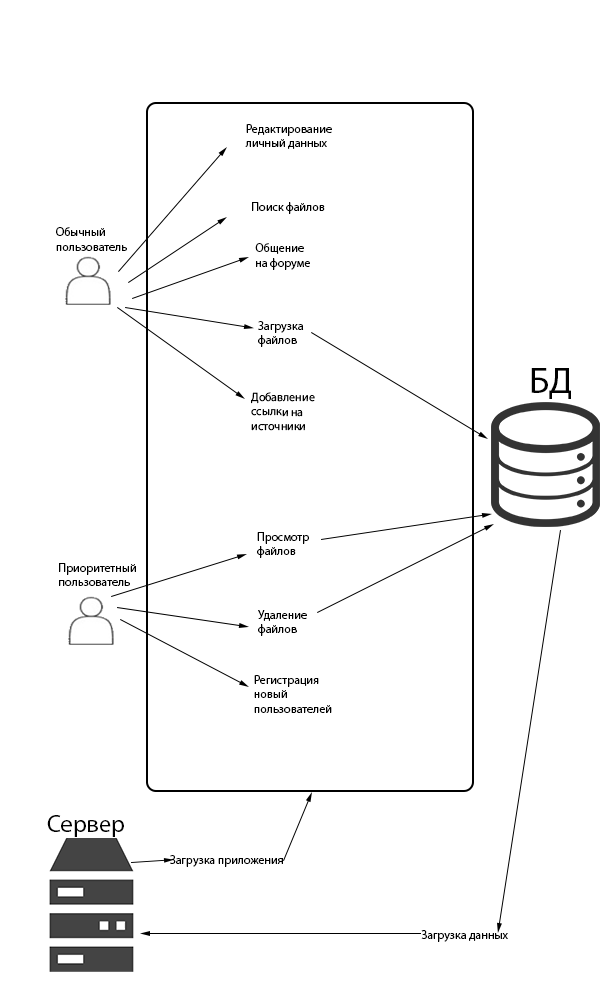
После добавление сообщения на форум и добавления нового источника в каталог иных источников данные сохраняются на сервере.

Из личного кабинета пользователь может просматривать каталог файлов. Каталог файлов, хранящихся в БД. При добавлении или скачивании файлов пользователь создает запрос. База данных хранится на сервере. Данные в базу данных считываются с сервера и наоборот загружаются на сервер.

Загрузка всего приложения происходит с сервера.

Существует еще приоритетный пользователь - администратор . Администратор просматривает добавленные файлы и сообщения на форум из личного кабинета администратора. Он также должен пройти процедуру авторизации. Администратор может удалять добавленные файлы и сообщения

**4.4 Диаграмма прецедентов (вариантов использования)**



1) Обычный пользователь – пользователь системы.

2) Приоритетный пользователь – приоритетный пользователь системы.

3) Сервер – сервер, на котором хранятся данные и web-приложение.

4) БД – база данных файлов, хранящихся на сервере.

5) Редактирование личных данных – процесс изменения данных личного кабинета (ФИО, дата рождения, пол, специальность, номер курса, номер группы, фотография, пароль, фото, статус).

6) Поиск файлов – процесс нахождения файлов по заданному ключу (основные критерии поиска: название предмета; дополнительные: фамилия преподавателя, фамилия автора, специальность).

7) Общение на форуме – процесс обмена сообщениями между пользователями образовательной платформы.

8) Загрузка файлов – процесс добавления файлов в БД.

9) Добавление ссылок на иные источники – процесс расширения БД файлообменной системы по средствам сети Интернет.

10) Просмотр файлов – процесс проверки, добавляемых пользователями файлов на наличие вирусов, актуальность нахождения в БД.

11) Удаление файлов – процесс удаления устаревших файлов из БД системы.

12) Регистрация новых пользователей – процесс, позволяющий создавать новых пользователей системы, путем создания нового личного кабинета и внесения в него личных данных.

13) Загрузка данных – процесс передачи ссылок на добавленные и удаленные файлы от БД к серверу.

14) Загрузка приложения – процесс запуска приложения при подключении пользователя к системе.

**4.5 Диаграмма связи объектов**

В работе с файлообменной системой участвуют 5 объектов: пользователь, персональный компьютер пользователя, сервер, персональный компьютер администратора и администратор.

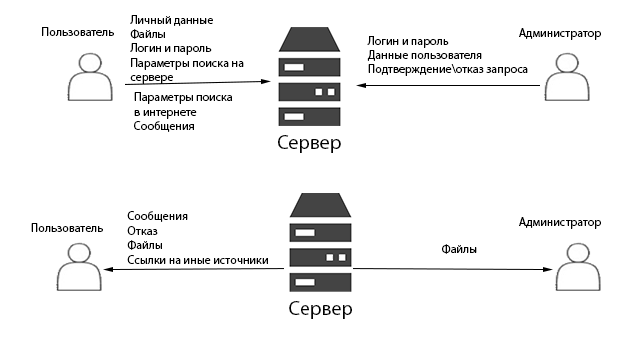
Пользователь, использую свой компьютер, загружает сервер, и входит в web-приложение. С другой стороны, администратор использует свой компьютер, загружает сервер и входит в личный кабинет администратор.

Приложение осуществляет доступ в интернет для поиска иных источников информации, также загружает личные данные пользователя в личном кабинете. Приложение, взаимодействую с БД, создает каталог файлов, необходимых пользователю.

**4.6 Диаграмма потоков данных**

На основании результатов выявленных при построении архитектурно-контекстной диаграммы и диаграммы прецедентов можно спроектировать диаграмму потоков данных.

Прежде чем приступать к проектированию ДПД, необходимо уточнить данные, передаваемые между пользователем и системой, между системным администратором и системой. Для этого построим расширенную АКД.

 В расширенной АКД показаны конкретные потоки информации между пользователями и системой.

Пользователь передает системе:

1. Личные данные – для редактирования данных личного кабинета.
2. Файл – для пополнения БД системы.
3. Логин и пароль – для входа в систему (процесс авторизации).
4. Запрос на про-р форума – для просмотра форума.
5. Параметры поиска на серв. – критерии поиска необходимых файлов.
6. Параметры поиска в инт. – критерии поиска необходимых файлов.
7. Сообщение – для добавления на форум.

Администратор передает системе:

* 1. Логин и пароль – для входа в систему (процесс авторизации).
  2. Данные пользователя – для создания нового личного кабинета пользователя.
  3. Подтверждение/отказ запрос – для разрешения добавлять или не добавлять файл, загруженный пользователем.

Пользователь получает от системы:

* + 1. Сообщения – сообщения, находящиеся на форуме.
    2. Отказ – неуспешное добавление файлов.
    3. Файлы – при успешном поиске необходимых источников информации.
    4. Ссылки на иные источники – при успешном поиске в каталоге иных источников.

**4.8 Заключение**

На основе выявленных требований в предыдущей главе был сформирован проект программного средства, позволяющий пользователю и разработчику окончательно договориться об однозначном понимании системы. Проект позволяет разработчику приступать к дальнейшему переходу к реализации.

# Глава 5. Проектирование БД

## 5.1 Введение

Данный веб-сайт – многопользовательский интерфейс с большим объемом данных, загружаемых определенными пользователями, и большим объемом пользователей. Для успешной работы приложения требуется разработать СУБД, основанную на анализе предметной области и проектировании программного средства. Система должна включать все объекты предметной области, учитывать взаимосвязи между ними. В связи с большим потоком пользователей и ограниченных аппаратных средствах, СУБД должна быть быстрой и безопасной в использовании.

## 5.2 Инфологическое проектирование

Описание информационных объектов:

\*тут ебать типа таблицы и строки или я хуй его знает и типа их описание\*

## 5.3 Взаимосвязи информационных объектов

\*тут ебать типа такая хуйня что прямоугольники и стрелочки и они типа взаимосвяаны ну как типа в бд хуйня которая JOIN\*

Диаграмма отражает взаимосвязи информационных объектов предметной области. Между объектами реализована связь от многих к многим с помощью вспомогательных таблиц, отражающих соответствие отдельных объектов другим.

**5.4 Формальный способ проектирования инфологической модели данных**

В данном разделе необходимо установить взаимосвязи между информационными объектами, имеющим общие реквизиты. Таким образом, необходимо сконструировать схему БД.

\*тут ебать сама бд\*

## 5.5 Выборы СУБД

MSAccess

Достоинства:

* Доступ делает большую переднюю часть других типов баз данных, таких как SQL сервер, и вы можете очень быстро создавать отличные приложения, управляемые данными, очень быстро
* Есть отличный встроенный пакет отчетов и легкий доступ к другим приложениям MS Office
* Все компоненты в рамках Access предварительно встроены, чтобы сделать взаимодействие с данными максимально простым

Недостатки:

* Наличие хотя бы одного ответа на каждый вопрос, предлагающий использовать что-то другое.
* Нет файла рабочих групп (mdw), что усложняет разграничение прав пользователей и построение защиты БД.
* Перегруженный интерфейс, из-за чего остается мало места для работы

Postgres

Достоинства:

* Является масштабируемым и способен обрабатывать терабайты данных.
* Большое сообщество - существует довольно большое сообщество в котором вы запросто найдёте ответы на свои вопросы
* Большое количество дополнений - несмотря на огромное количество встроенных функций, существует очень много дополнений, позволяющих разрабатывать данные для этой СУБД и управлять ими.
* Существует множество предопределенных функций.
* Объектность - PostrgreSQL это не только реляционная СУБД, но также и объектно-ориентированная с поддержкой наследования и много другого

Недостатки:

* Документация туманна, поэтому, возможно, ответы на некоторые вопросы придется искать в интернете.
* Конфигурация может смутить неподготовленного пользователя.

MongoDB

Достоинства:

* Скорость и простота в использовании.
* Движок поддерживает json и другие традиционные документы NoSQL.
* Данные любой структуры могут быть сохранены/прочитаны быстро и легко..

Недостатки:

* SQL не используется в качестве языка запросов.
* Инструменты для перевода SQL-запросов в MongoDB доступны, но их следует рассматривать именно как дополнение.

Проведя сравнительный анализ, можно сделать вывод, что наиболее подходящей СУБД из претендентов является Postgres. Данная СУБД способна обеспечивать эффективную работу с БД при необходимом количестве записей, СУБД аппаратно-независимая, кроссплатформенная, расширяемая, присутствует возможность управления распределенными БД.

## 4.6 Заключение

На основе данных, полученных при анализе предметной области, было сформировано четкое представление БД необходимой для создания веб приложения. Было составлено детальное описание объектов предметной области, а именно: «Забронированные туры», «Туристы», «Список страховок», «Каталог туров», «Список отелей», «Список стран», «Работники», «Список городов», «Список авиакомпаний», «Пользователи», «Клиенты». Для каждого объекта был определен набор реквизитов и соответствующий тип данных. Были установлены взаимосвязи между объектами. Составлена схема БД . В качестве СУБД была выбрана бесплатная СУБД PostgreSQL, успешно интегрируемая с инструментами реализации веб-приложения Django и Vue. Таким образом, был создан проект БД для веб-приложение туристического агентства.

# Заключение

а основе данных, полученных при анализе предметной области, было сформировано четкое представление БД необходимой для создания файлообменной системы. Было составлено детальное описание объектов предметной области, а именно: \*тут хуйня всякая типа\*. Для каждого объекта был определен набор реквизитов и соответствующий тип данных. Были установлены взаимосвязи между объектами. Составлена схема БД. В качестве СУБД была выбрана бесплатная СУБД \*тут какая выбрана я хуй его знает че вы пикнете наврное постргрегрс. Таким образом, был создан проект БД для образовательной платформы формальдегид.