

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ

по научно-исследовательской работе

Тема: Применение Spring framework для разработки веб приложения на примере корпоративной социальной сети.

Студент _____

гр. 5304 Фомченко О.А.

Руководитель _____

Лисс А.А.

Санкт-Петербург 2020

ЗАДАНИЕ НА НАУЧНО-ИССЛЕДОВАТЕЛЬСКУЮ РАБОТУ

Студент Фомченко О.А.

Группа 5304

Тема НИР: Применение Spring framework для разработки веб приложения на примере корпоративной социальной сети.

Задание на НИР: Создание веб-приложения корпоративной социальной сети для компаний, занимающихся разработкой программного обеспечения на основе Spring.

Сроки выполнения НИР: 01.09.2020 – 20.12.2020

Дата сдачи отчета: 20.12.2020

Дата защиты отчета: 20.12.2020

Студент гр. 5304

Фомченко О.А.

Руководитель

Лисс А.А.

АННОТАЦИЯ

Целью работы является разработка структуры базы данных для разрабатываемого сервиса и проектирование интерфейса. В данной работе представлены основные принципы построения структуры базы данных и проектирования интерфейса.

SUMMARY

The main purpose of this work is to develop database and viewer interface for target application and to show its main development principals.

Contents

ПОСТАНОВКА ЗАДАЧИ	5
Актуальность	5
Цель работы	5
Основные задачи	5
РЕЗУЛЬТАТЫ РАБОТЫ В ОСЕННЕМ СЕМЕСТРЕ	6
Структура базы данных	6
Главный модуль	6
Мэссенджер	9
Трекер задач	12
Итоговая структура базы данных	13
Пользовательский интерфейс	14
Мэсенджер	14
Аккаунт	15
Трекер задач	16
Задача	17
ПЛАН РАБОТЫ НА ВЕСЕННИЙ СЕМЕСТР	18
ЗАКЛЮЧЕНИЕ	19
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ	20

ПОСТАНОВКА ЗАДАЧИ

Актуальность

Spring – это архитектурный фреймворк существующий не столько для выполнения какой-то прикладной задачи, сколько для обеспечения лучшей масштабируемости, возможности более простого тестирования и более простой интеграции с другими фреймворками (например Struts или Hibernate). Благодаря этому писать большие приложения становится проще - разработчики избегают ряда проблем, связанных с созданием enterprise-приложений, вместо того, чтобы их решать.

Так же само приложение является крайне полезным инструментом для компаний, занимающихся разработкой ПО, так как объединяет в себе не только систему коммуникации и менеджмента команд, но и систему трекинга задач, что сильно облегчает и ускоряет процессы планирования этапов разработки.

Цель работы

Разработка эффективного и масштабируемого продукта для коммуникации и менеджмента команд с использованием Spring на языке Java.

Основные задачи

- 1) Описать процесс разработки приложения.
- 2) Произвести обзор существующих инструментов и фреймворков, позволяющих получить наиболее качественный продукт.
- 3) Сформировать перечень требований к разрабатываемому приложению.
- 4) Описать схему базы данных, используемую приложением.
- 5) Описать способы развертывания приложения.
- 6) Спроектировать графический интерфейс приложения.
- 7) Реализовать приложение.

РЕЗУЛЬТАТЫ РАБОТЫ В ОСЕННЕМ СЕМЕСТРЕ

Разрабатываемое приложение имеет микросервисную архитектуру и состоит из следующих основных сервисов:

- Мэссенджер – сервис, отвечающий за систему коммуникаций между пользователями приложения.
- Трекер задач – сервис, отвечающий за управление, создание и распределение целей и задач команды.
- Графический интерфейс – сервис, отвечающий за отображение информации в браузере.
- Главный модуль – сервис, отвечающий за управление зависимостями между командами и пользователями.

Структура базы данных

Графический интерфейс является front-end сервисом, он никак не взаимодействует с базой данных напрямую. В то время как главный модуль, мэссенджер и трекер задач имеют индивидуальные структуры базы данных.

Мэссенджер и трекер задач являются независимыми друг от друга сервисами и зависят только от состояния главного модуля.

Таким образом мы можем обеспечить минимальную связанность между сервисами, что дает нам высокую гибкость и масштабируемость, что очень важно для подобных проектов, так как у него бесконечное множество путей развития и модифицирования.

Так как все сервисы зависят от состояния главного модуля, начнем именно с его рассмотрения.

Главный модуль

Главной сущностью приложения, является сущность user, являющаяся отражением конечного пользователя в приложении:

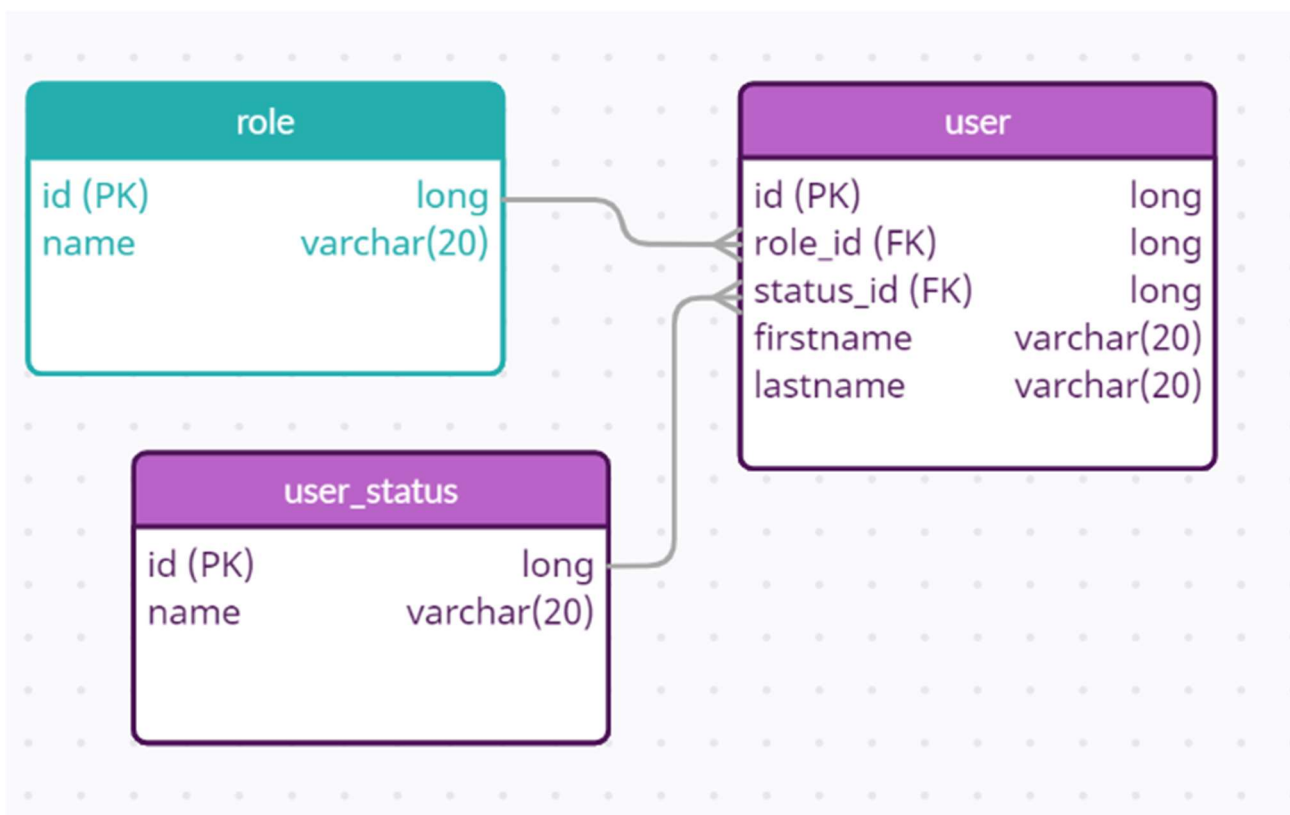


Рис 1. Сущность user в БД.

Ее состояние зависит от таблиц user_status и role, которые, по сути, являются enum-ами. У каждого пользователя в текущий момент времени может быть только один status и role, по-этому связи между ними один к многим.

Так же в приложении существует система аутентификации и система ролей, которые относятся к модулю безопасности, встроенному в главный модуль.

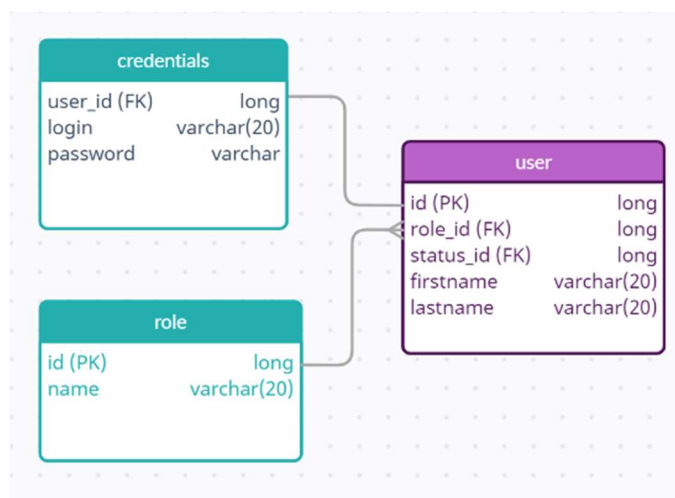


Рис 2. Сущности модуля безопасности.

Role отвечает за систему ролей в приложении, а credentials за систему аутентификации. Так как у каждого пользователя должны быть уникальные данные для авторизации, связь один к одному.

Пользователи, работающих над определенным набором задач объединяется сущность под названием team. Команды, работающие над схожими наборами задач объединяются в сущность под названием group.

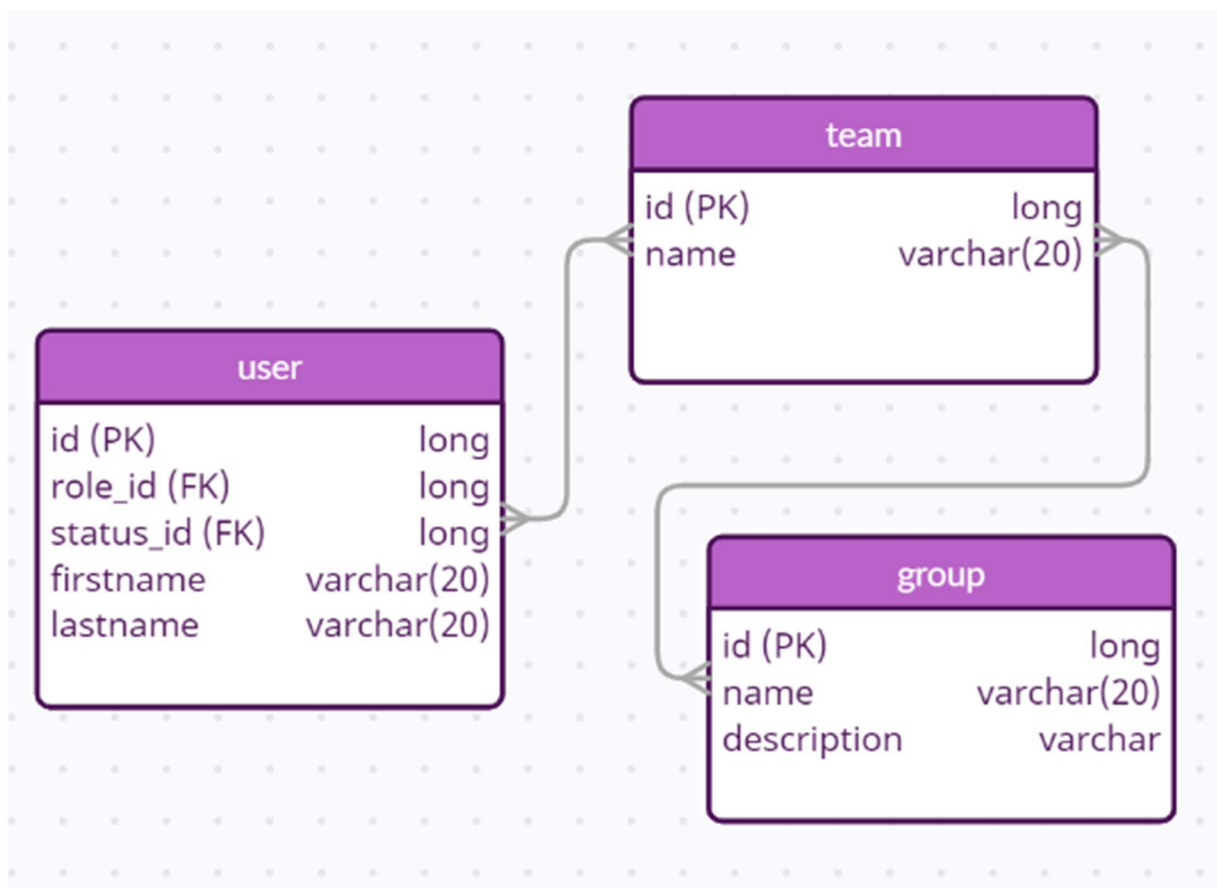


Рис 3. Связи между user team и group.

Так как пользователь может принадлежать к нескольким командам, а команда может принадлежать к нескольким группам, то связи между ними многие к многим.

Отношения между пользователями разрешаются через сущность user_relationships, где указаны объекты отношения и его состояние, определенное в таблице user_relation, которая является enum-ом.

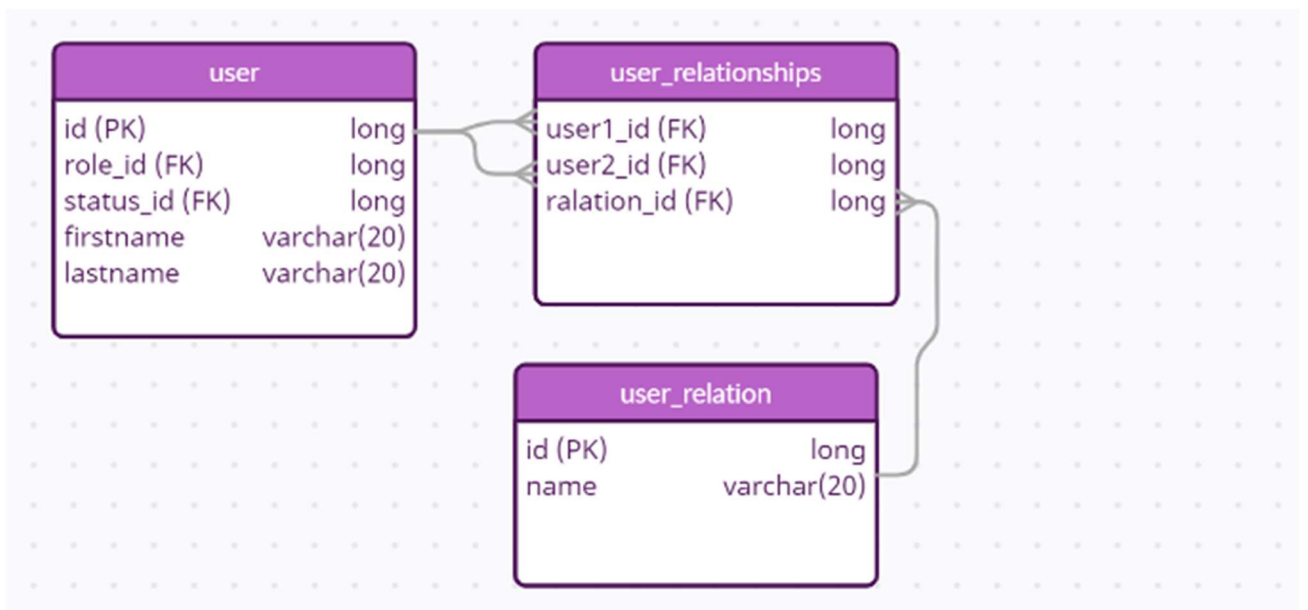


Рис 4. Сущности отношения пользователей.

Таким образом, структура БД для главного модуля спроектирована и подсвечена фиолетовым и лазурным цветом:

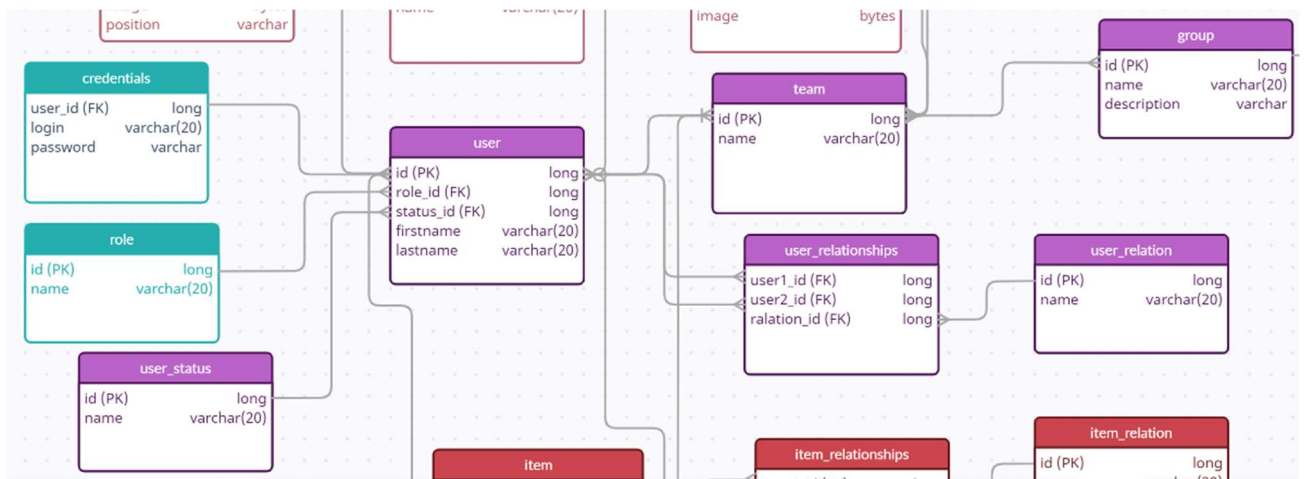


Рис 5. Структура БД для главного модуля.

Мэссенджер

Главной сущностью мэссенджера является **message**, которая соотносится с сущностью **attachment**.

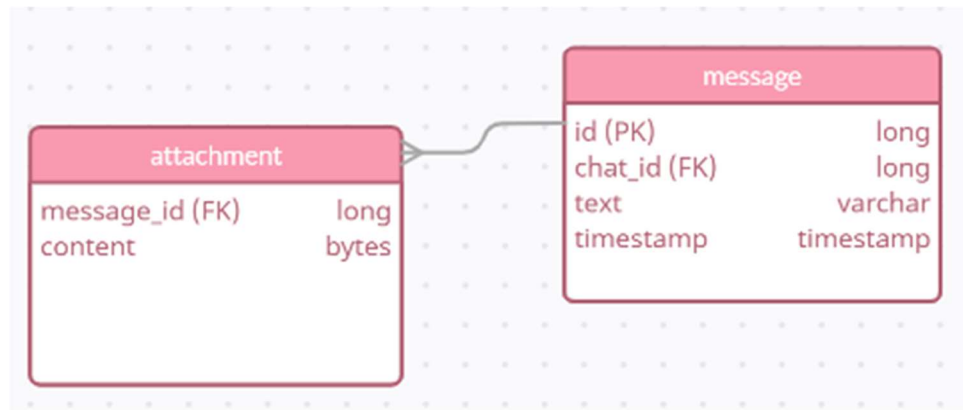


Рис 6. Сущность message.

Ее основные характеристики это текст, прикрепленные материалы и время сообщения.

Непосредственно для осуществления коммуникации используется сущность chat, которая относится к трем возможным типам чата, **user_chat**, **team_chat** и **group_chat**.

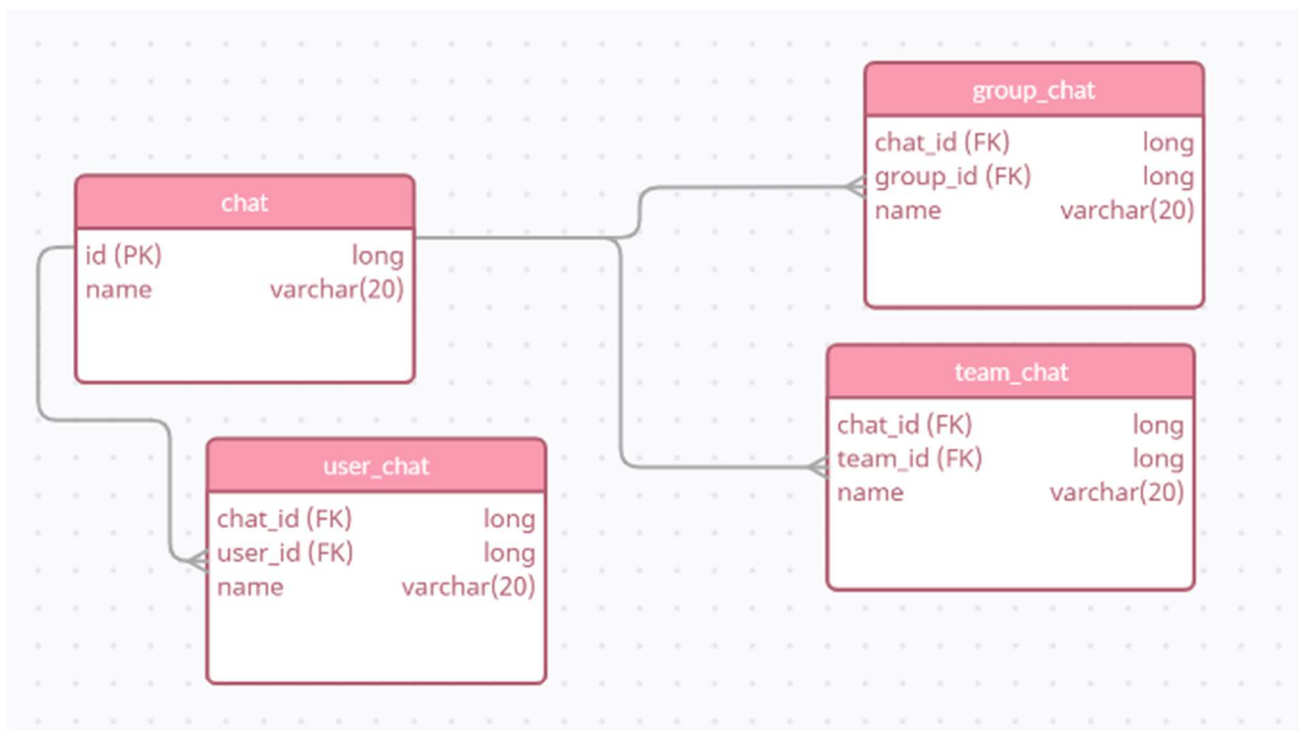


Рис 7. Сущность chat.

Так как у пользователя, команды и группа команд может быть множество чатов, но чат может принадлежать только к одному типу чатов, то тип связи один к многим.

Так же у пользователя и команды есть аккаунты, в которых хранится дополнительная информация. В БД они выражены в виде сущностей `user_account` и `team_account`.

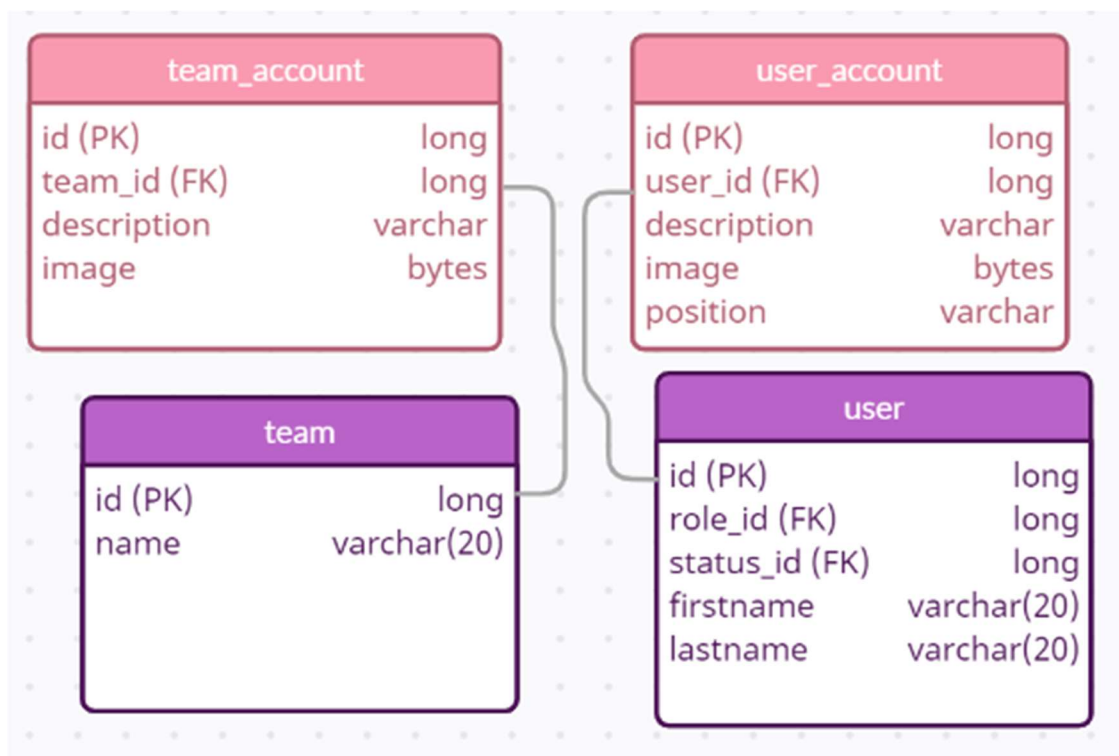


Рис 8. Account сущности.

Так как у пользователя может быть только один уникальный аккаунт, тип связи один к одному.

Таким образом мы получили полную картину структуры БД для мессенджера, подсвеченную розовым цветом:

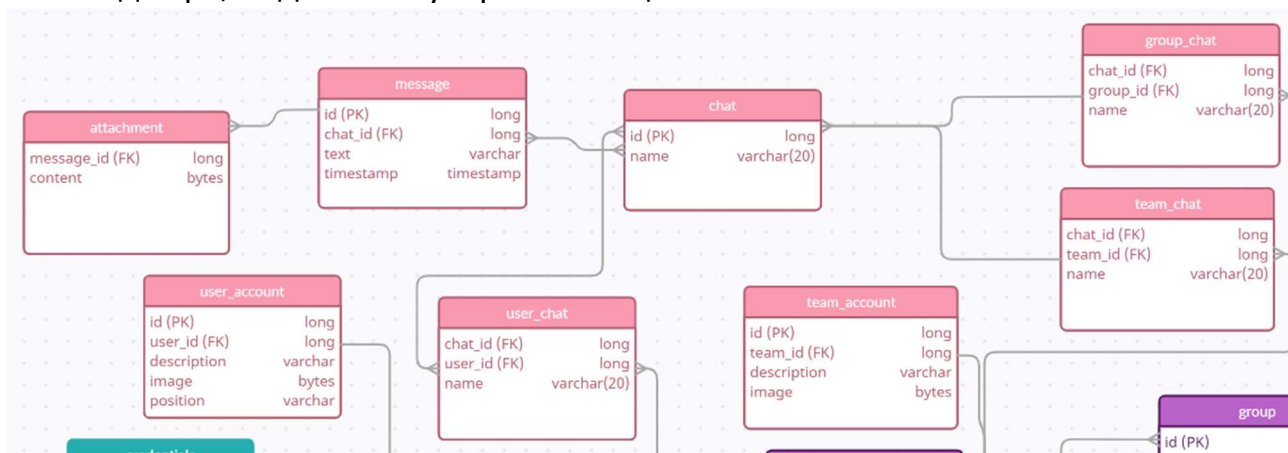


Рис 9. Структура БД для мессенджера.

Трекер задач

Главной сущностью для трекера задач является item, состояние которой зависит от item_status, являющейся enum-ом.

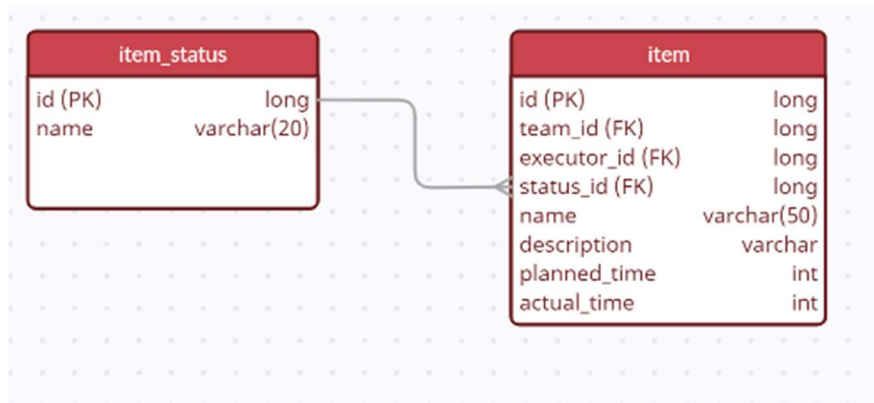


Рис 10. Сущность item.

Так как задача может иметь только один статус, тип связи один к многим.

Отношения между задачами разрешается при помощи сущности item_relationships, содержащий статус отношения, хранящийся в enum-е item_relation.

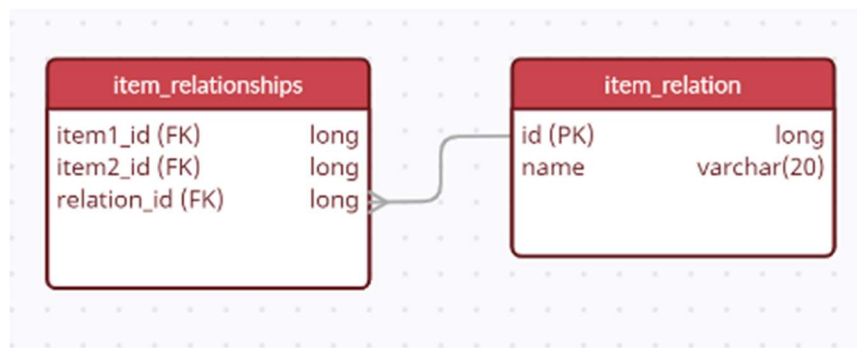


Рис 11. Сущность item_relationships

Таким образом, мы получили схему БД для сервиса трекера задач, подсвеченного красным цветом:

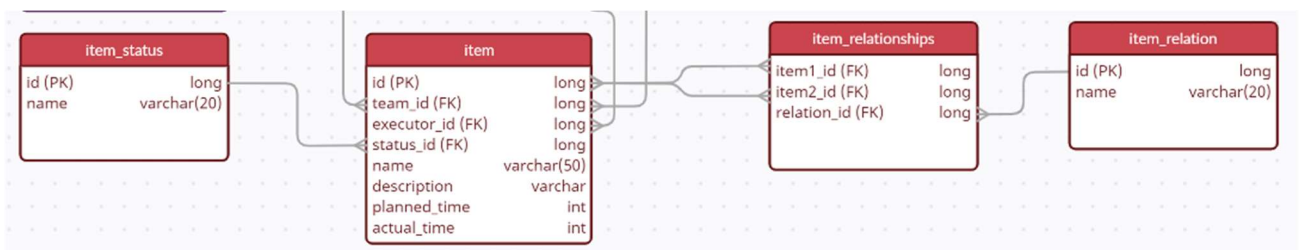


Рис 12. Структура БД для трекера задач.

Итоговая структура базы данных

В результате работы по проектированию структуры базы данных была получена ее итоговая схема, состоящая из четырех модулей: Главного, подсвеченного фиолетовым; Модулем безопасности, являющегося частью главного и подсвеченного лазурным; Мэссенджера, подсвеченного розовым; Трекера задач, подсвеченного красным:

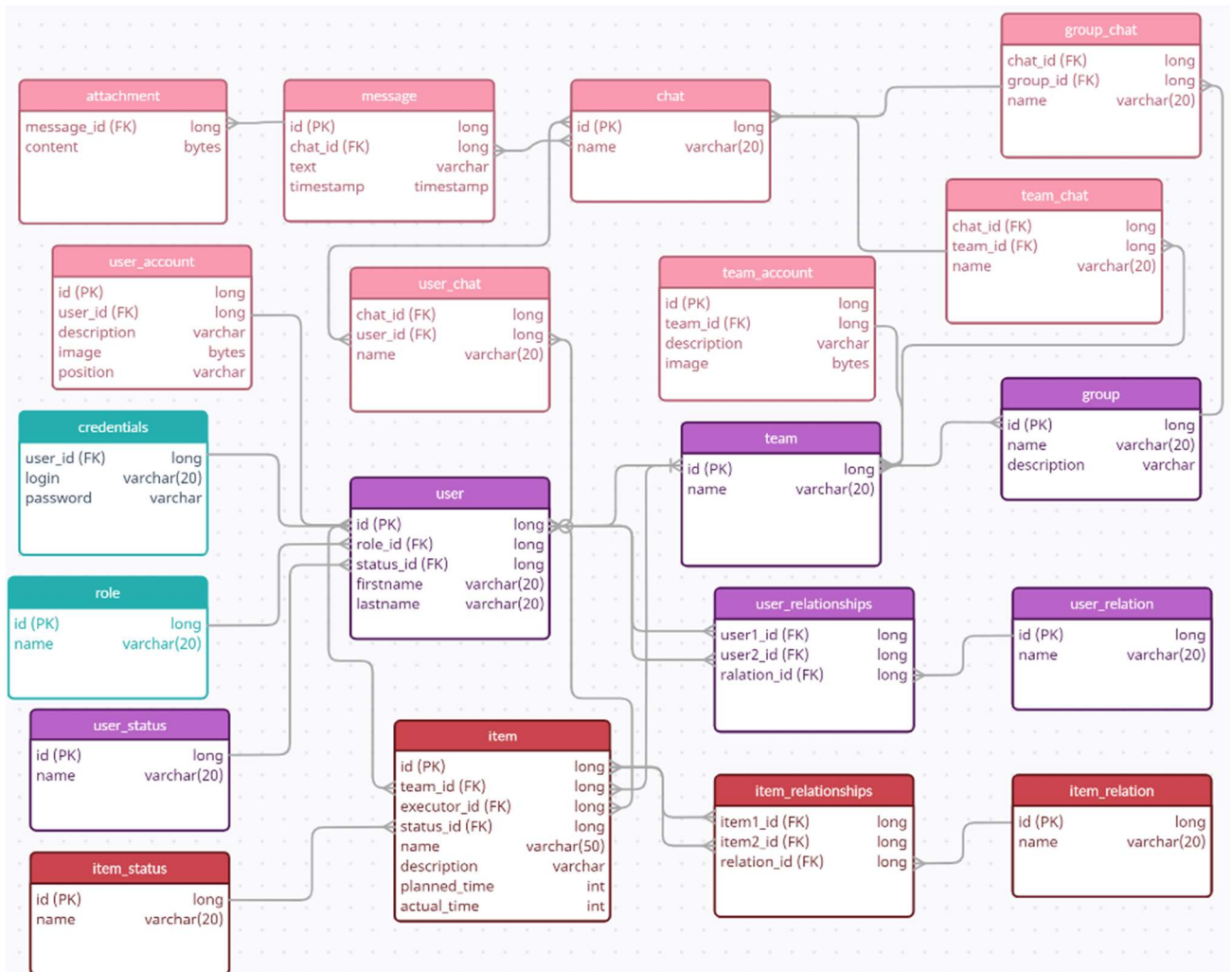


Рис 13. Общая структура БД приложения.

Пользовательский интерфейс

Так как для главного модуля пользовательский интерфейс не требуется, то были разработаны четыре основные страница пользовательского интерфейса:

- Мэссенджер
- Аккаунт
- Трекер задач
- Задача

Для удобства навигации по приложению на каждой странице присутствует навигационная панель с логотипом, ссылкой на основные страницы приложения и аккаунт пользователя с drop-down списком.

Мэссенджер

Основная задача пользовательского интерфейса мэссенджера является обеспечения удобного способа переключения между чатами, их чтения и написания сообщений.

В этих целях был разработан следующий интерфейс:

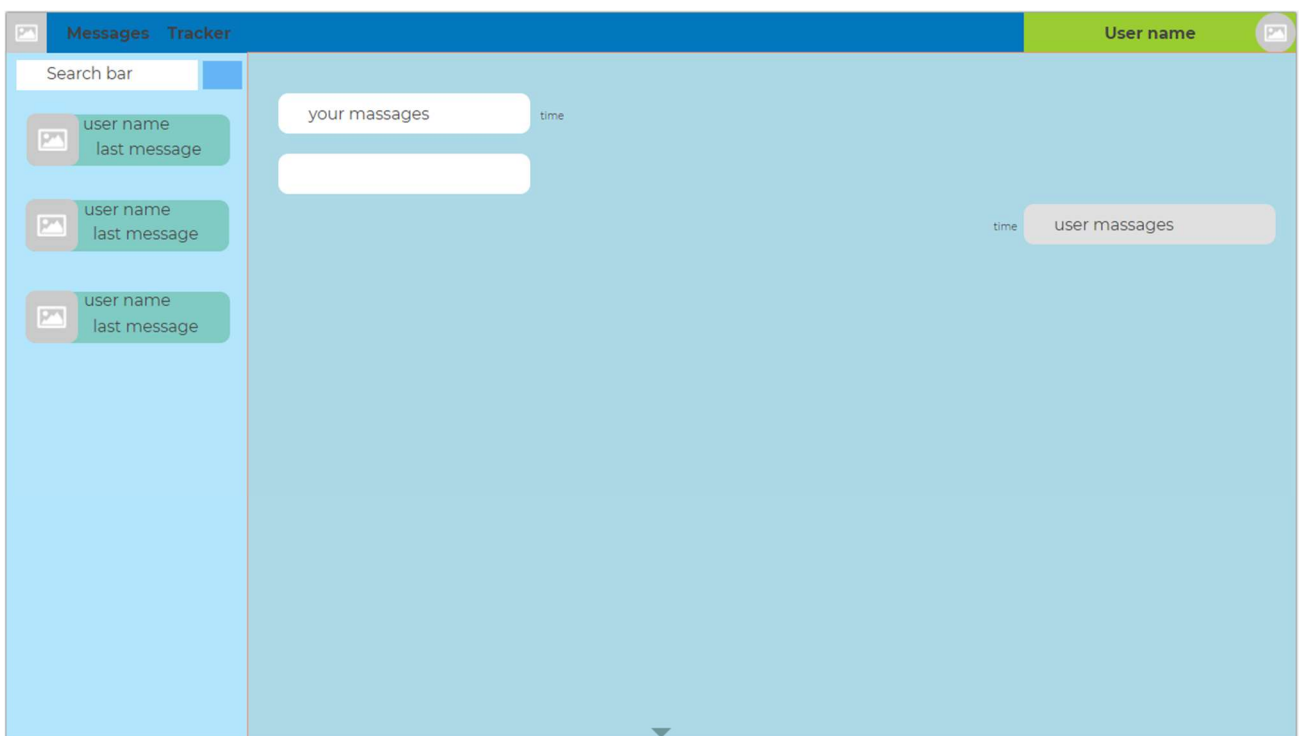
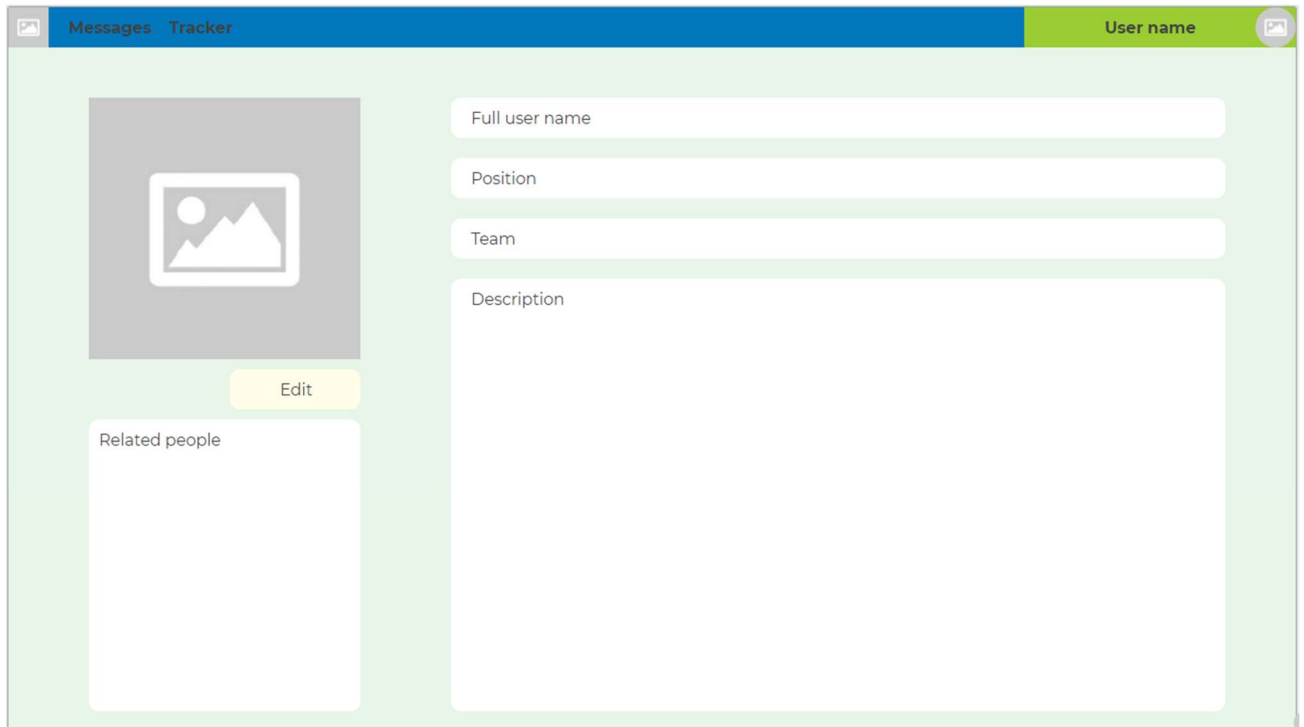


Рис 14. Схематичный мокап страницы мэссенджера.

Аккаунт

Главной целью страницы аккаунта является предоставление пользователю основной информации о владельце аккаунта.

Исходя из этого был разработан следующий мокап:



The mockup shows a web interface for a user profile. At the top, there is a navigation bar with a blue section containing 'Messages' and 'Tracker' with a small icon, and a green section containing 'User name' with a small icon. The main content area has a light green background. On the left, there is a large square placeholder for a profile picture with a camera icon. Below it is a yellow 'Edit' button. Underneath the 'Edit' button is a white box labeled 'Related people'. To the right of the profile picture, there are four input fields: 'Full user name', 'Position', 'Team', and 'Description' (which is a larger text area). The 'Description' field is currently empty.

Рис 15. Схематичный мокап страницы аккаунта.

Трекер задач

Целью страницы трекера задач является показать список задач с их названиями статусами и исполнителями с возможностью редактирования этого состояния.

Принимая во внимания данные требования был разработан следующий вид страницы:

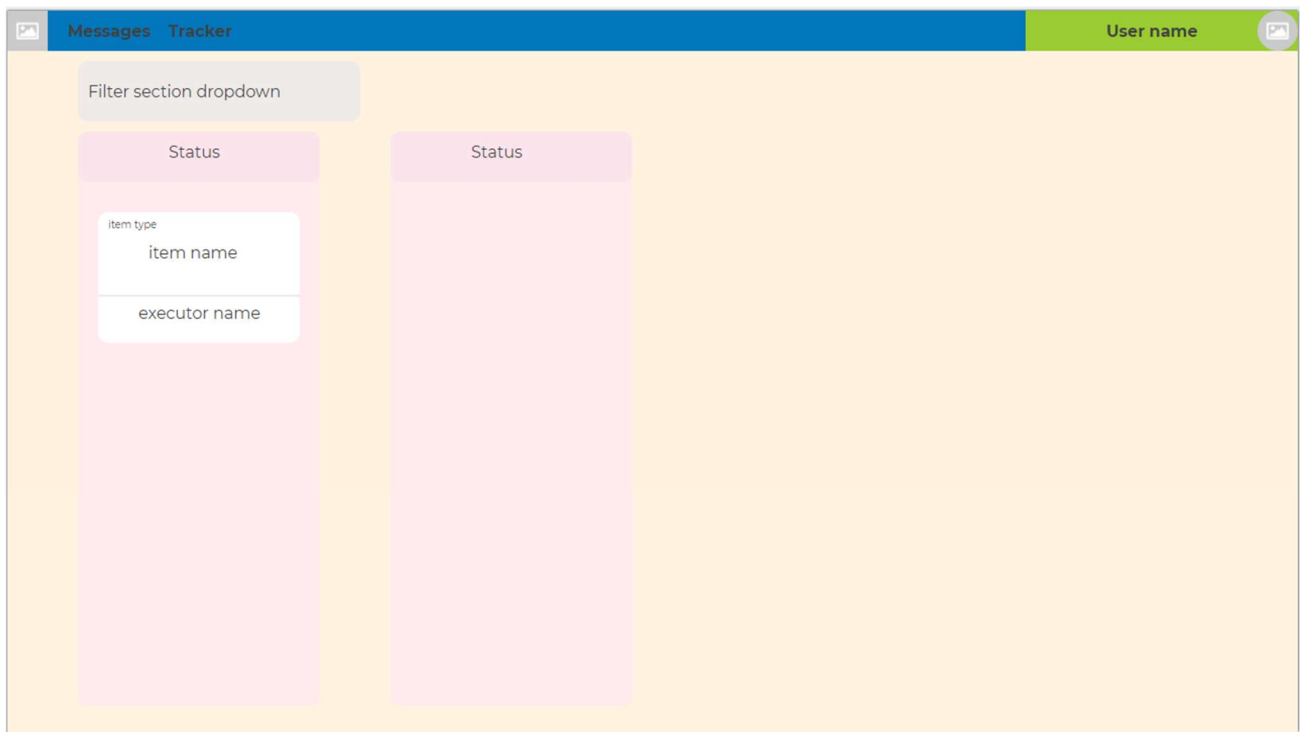


Рис 16. Схематичный мокап страницы трекера задач

Задача

Предназначение страницы для задачи состоит в том, чтобы просмотреть всю информацию по задаче и иметь возможность ее редактировать.

Учитывая эти требования, была разработана pop-up страница для задачи:

The mockup shows a web interface for a 'Messages Tracker'. At the top, there is a blue header bar with the text 'Messages Tracker' and a green bar on the right with 'User name' and a profile icon. Below the header, there is a light orange background area. On the left, there is a 'Filter section dropdown' and a pink sidebar with 'Status' tabs and a list of items with 'item type', 'item name', and 'executor name'. The main content area is a purple-bordered box containing a form with fields for 'item name', 'executor name', 'item type', 'item status', 'description', and 'additional information'. At the bottom of the form are 'previous status' and 'next status' buttons.

Рис 17. Схематичный мокап для страницы задачи.

ПЛАН РАБОТЫ НА ВЕСЕННИЙ СЕМЕСТР

- Настроить окружение приложения.
- Реализовать backend приложения.
- Реализовать frontend приложения.
- Описать процесс разработки приложения.
- Описать способы развертывания приложения.

ЗАКЛЮЧЕНИЕ

В ходе научно-исследовательской работы в осеннем семестре была разработана архитектура приложения, создана структура базы данных и ее uml-диаграмма и спроектированы мокапы основных страниц пользовательского интерфейса приложения.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

- 1) Основы проектирования баз данных // url: <https://habr.com/ru/post/514364/>
- 2) Инструмент для создания структур баз данных // url: <https://creatly.com/ru/lp/>
- 3) Инструмент для создания мокапов сайтов // url: <https://lk.wilda.ru/editor/>
- 4) PostgreSQL // Документация url: <https://postgrespro.ru/docs/postgrespro/9.5/sql>
- 5) Spring // Документация url: <https://spring.io/>
- 6) Spring Boot plus React JS // url: <https://habr.com/ru/company/alfa/blog/340776/>