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 |

**ИНСТРУКЦИЯ ДЛЯ РАЗРАБОТЧИКА**

к системе

|  |
| --- |
| по учету коммуникаций |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнила |  | АСУб-20-2 |  |  |  | А.В. Арбакова |
|  |  | шифр группы |  | подпись |  | И.О. Фамилия |

Иркутск 2023 г.

Содержание

[Аннотация 3](#_Toc154100934)

[Технические требования 3](#_Toc154100935)

[1. Аппаратные требования 3](#_Toc154100936)

[2. Программные требования 3](#_Toc154100937)

[Использованные в разработке программные средства 3](#_Toc154100938)

[Архитектура приложения 4](#_Toc154100939)

[Хранилище данных (БД) 6](#_Toc154100940)

[Структура пользовательского интерфейса 12](#_Toc154100941)

# **Аннотация**

Система по учёту коммуникаций позволяет отслеживать все коммуникации пользователя, с возможностью планирования и координации различных его коммуникаций и встреч.

Электронный календарь экономит время пользователя и его личные ресурсы, потому что все данные пользователя хранятся в одном месте, что позволит быстро и легко вносить изменения и обновления в расписание.

Система предназначена для помощи пользователю с планированием и анализом его рабочих и личных коммуникаций, и осуществлением напоминаний пользователю о запланированных коммуникациях.

# **Технические требования**

1. **Аппаратные требования**

Для запуска приложения необходим компьютер с нижеприведёнными системными характеристиками или выше:

* 64-разрядный (x64) процессор;
* 4 ГБ оперативной памяти;
* Свободное место на жестком диске 20 Mb.

1. **Программные требования**

Для запуска приложения необходимы следующие программы:

* Git;
* XAMPP;
* СУБД MySQL;
* Операционная система Windows 10.

# **Использованные в разработке программные средства**

Для хранения: GitHub.

В веб-разработке: HTML5, CSS3, Javascript, PHP, MySQL.

**Архитектура приложения**

Архитектура приложения – это структура или организация различных компонентов программного обеспечения, которые вместе выполняют определенные функции.

Схема архитектуры приложения приведена на рисунке 1.

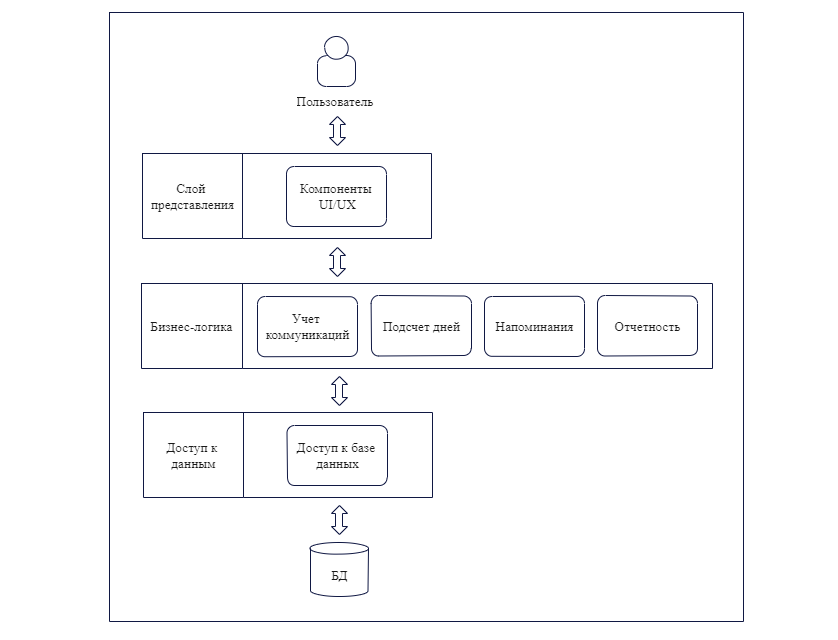
[](https://github.com/ArbakovaAnastasia/AccountingForCommunications/raw/main/%D0%9F%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5/%D0%B0%D1%80%D1%85%D0%B8%D1%82%D0%B5%D0%BA%D1%82%D1%83%D1%80%D0%B0.png)

Рисунок 1 – Схема архитектуры приложения

На схеме представлена взаимосвязь частей архитектуры приложения: слоя представления, бизнес-логика, доступ к данных.

Слой представления в архитектуре приложения необходим для связи между пользовательским интерфейсом и бизнес-логикой. Он отвечает за отображение данных на экране, обработку пользовательского ввода и передачу информации на другие уровни приложения. Основные функции слоя представления включают:

* Отображение данных из модели на экране;
* Обработка событий, таких как нажатия кнопок, ввод текста и т. д.;
* Валидация пользовательского ввода;
* Передача данных на уровень бизнес-логики или модели;
* Обновление пользовательского интерфейса в ответ на изменения в данных или состоянии приложения.

Бизнес-логика приложения состоит из задач: учет коммуникаций, подсчет дней, напоминания, отчетность. Слой бизнес-логики необходим для выполнения сложных операций и алгоритмов, связанных с обработкой данных и выполнением задач, определенных функциональностью приложения. Он включает в себя правила и процедуры, которые управляют работой приложения и обеспечивают выполнение бизнес-требований. Этот слой отделяет представление и доступ к данным от бизнес-правил и предоставляет возможность легко изменять и обновлять логику без влияния на другие компоненты системы.

Слой доступа к данным обеспечивает взаимодействие приложения с источниками данных, такими как базы данных, файловые системы, API и другие. В данной случае, с базой данных. Он отвечает за получение данных из источников, их преобразование и передачу на следующий уровень архитектуры. Основными функциями слоя доступа к данным являются:

* Подключение к источникам данных;
* Выполнение запросов к базам данных;
* Трансформация и форматирование данных для передачи на следующий уровень;
* Обеспечение безопасности и контроля доступа к данным;
* Обработка ошибок и исключений, возникающих при работе с данными.

**Хранилище данных (БД)**

Сущность БД – объект, который может быть однозначно идентифицирован в рамках предметной области. Сущности используются для организации данных в базе данных и для установления связей между ними. Для предметной области, определены сущности: пользователь; коммуникация; человек, с которым коммуникация.

Пара сущностей – это два объекта, которые связаны друг с другом и имеют определенные отношения между собой.

На рисунке 2 изображены пары сущностей, между которыми можно установить связь.

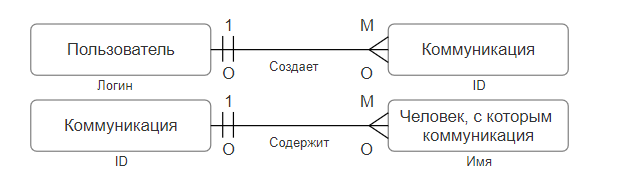


Рисунок 2 – Пары сущностей

В таблицах 1, 2 и 3 описаны данные хранящиеся для сущностей «Пользователь», «Коммуникация», «Человек с которым коммуникация».

Таблица 1 – Сущность «Пользователь»

| **Пользователь** |
| --- |
| Логин (PK) |
| Пароль |
| Дата регистрации |
| Коммуникация (FK) |

Таблица 2 – Сущность «Коммуникация»

| **Коммуникация** |
| --- |
| ID (PK) |
| Название |
| Дата планирования |
| Время планирования |
| Тип коммуникации |
| Человек, с которым коммуникация (FK) |
| Описание |
| Статус |

Таблица 3 – Сущность «Человек с которым коммуникация»

| **Человек с которым коммуникация** |
| --- |
| Имя (PK) |

Логическая модель базы данных – это описание структуры данных, включая таблицы, поля и связи между ними, в терминах, не зависящих от конкретной СУБД. Она нужна для того, чтобы определить, как данные организованы и связаны друг с другом в приложении, и чтобы разработчики могли понять, какие данные требуются для выполнения определенных функций.

Логическая модель базы данных представлена на рисунке 3.

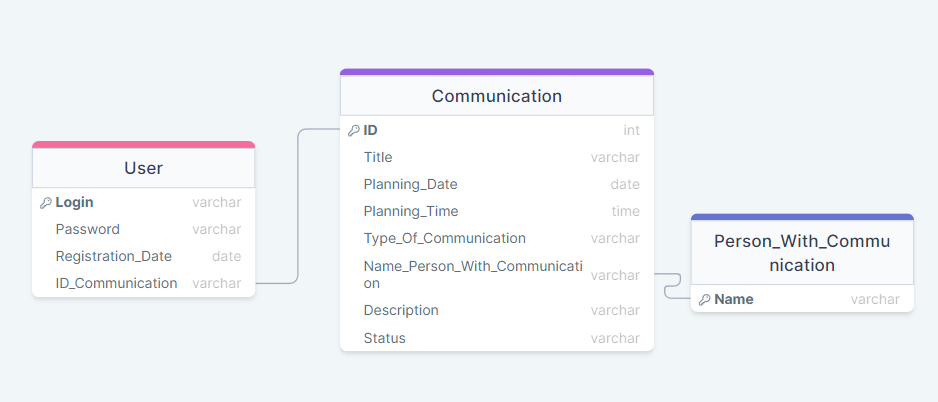


Рисунок 3 – Логическая модель БД

Для подключения к базе данных необходимо установить следующую программу XAMPP. Данная программа содержит: веб-сервер PHP, БД MySQL, и СУБД PhpMyAdmin. Установите XAMPP (см. рисунок 4).

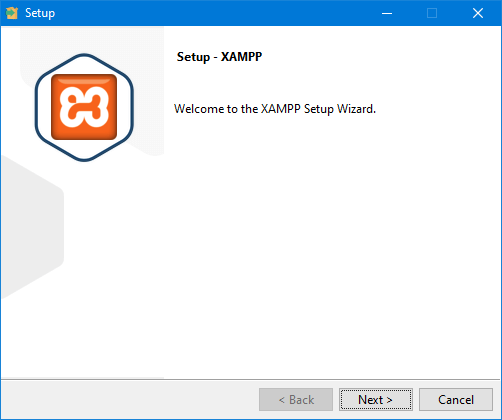


Рисунок 4 – Установка XAMPP

Запустите XAMPP. Откроется контрольная панель комплекса (см. рисунок 5).

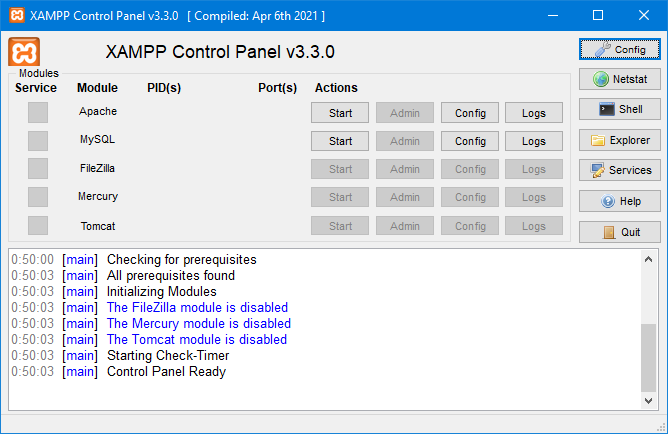


Рисунок 5 – Контрольная панель XAMPP

Запустите Apache и MySQL (см. рисунок 6).

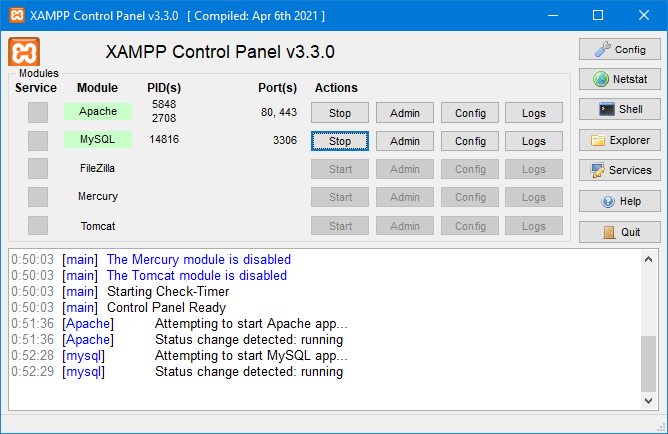


Рисунок 6 – Запуск Apache и MySQL

Далее нажмите на кнопку «Explorer», откроется пустая папка, в которую нужно загрузить файлы системы. Для того чтобы загрузить файлы системы, следует ввести в адресную строку команду «cmd». Данная команда откроет консоль операционной системы, где необходимо ввести команду «git clone [https://github.com/ArbakovaAnastasia/AccountingForCommunications.git»](https://github.com/ArbakovaAnastasia/AccountingForCommunications.git) (см. рисунок 7).

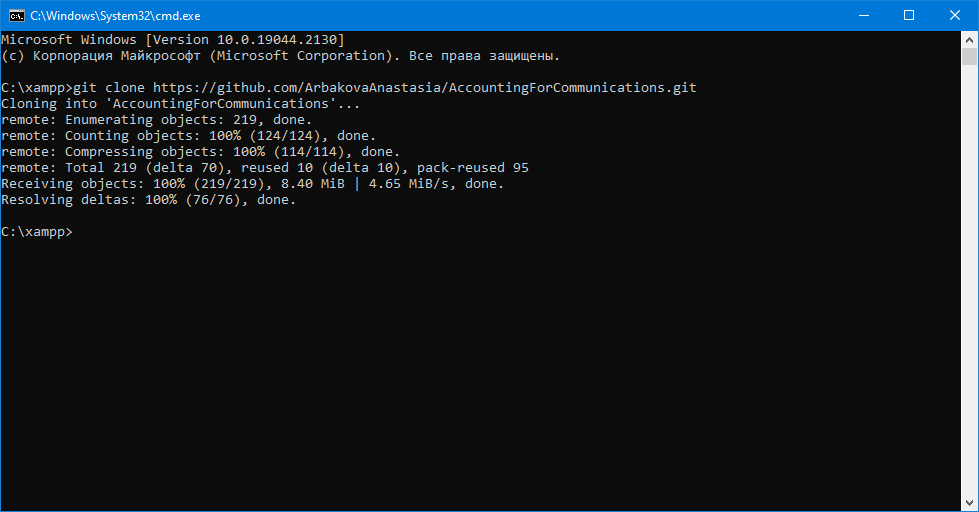


Рисунок 7 – Клонирование репозитория

После выполнения команды консоль закрывается, а в папке появляются файлы системы (см. рисунок 8).

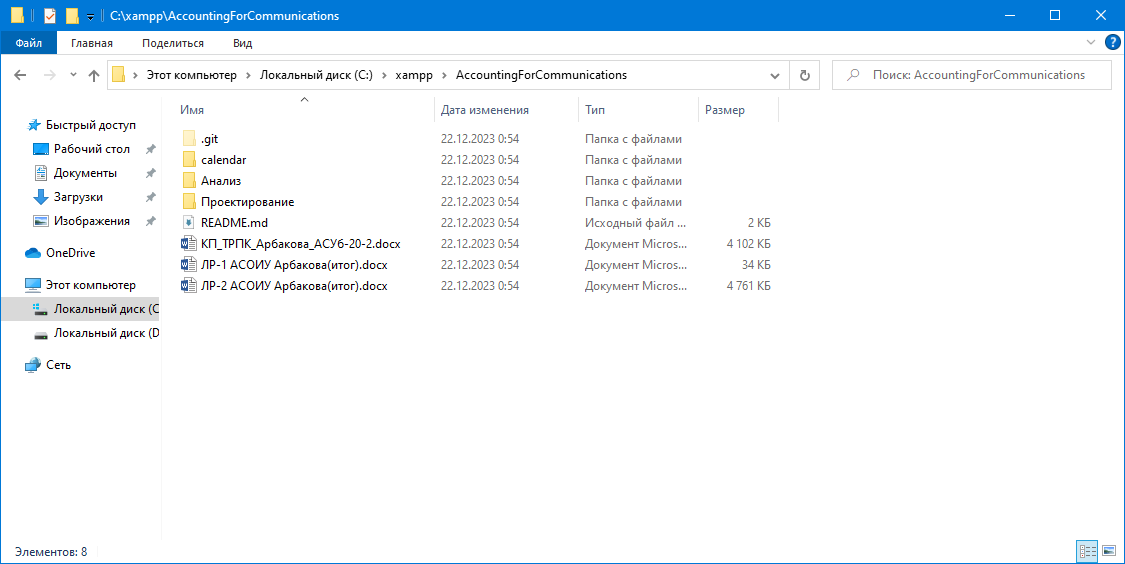


Рисунок 8 – Клонированные файлы

После необходимо настроить конфигурацию системы. Для этого создайте файл «application-configuration.cfg» и заполнить данные БД в файл конфигурации:

database-address: LOCALHOST – адрес сервера базы данных.

database-port: 3306 – порт на котором запущена БД.

database-login: root – логин базы данных.

database-password: root – пароль базы данных.

datebase-name: EVENTS – название базы данных.

Далее следует нажать на кнопку «admin». После чего откроется СУБД «PhpMyAdmin» (см. рисунок 9).

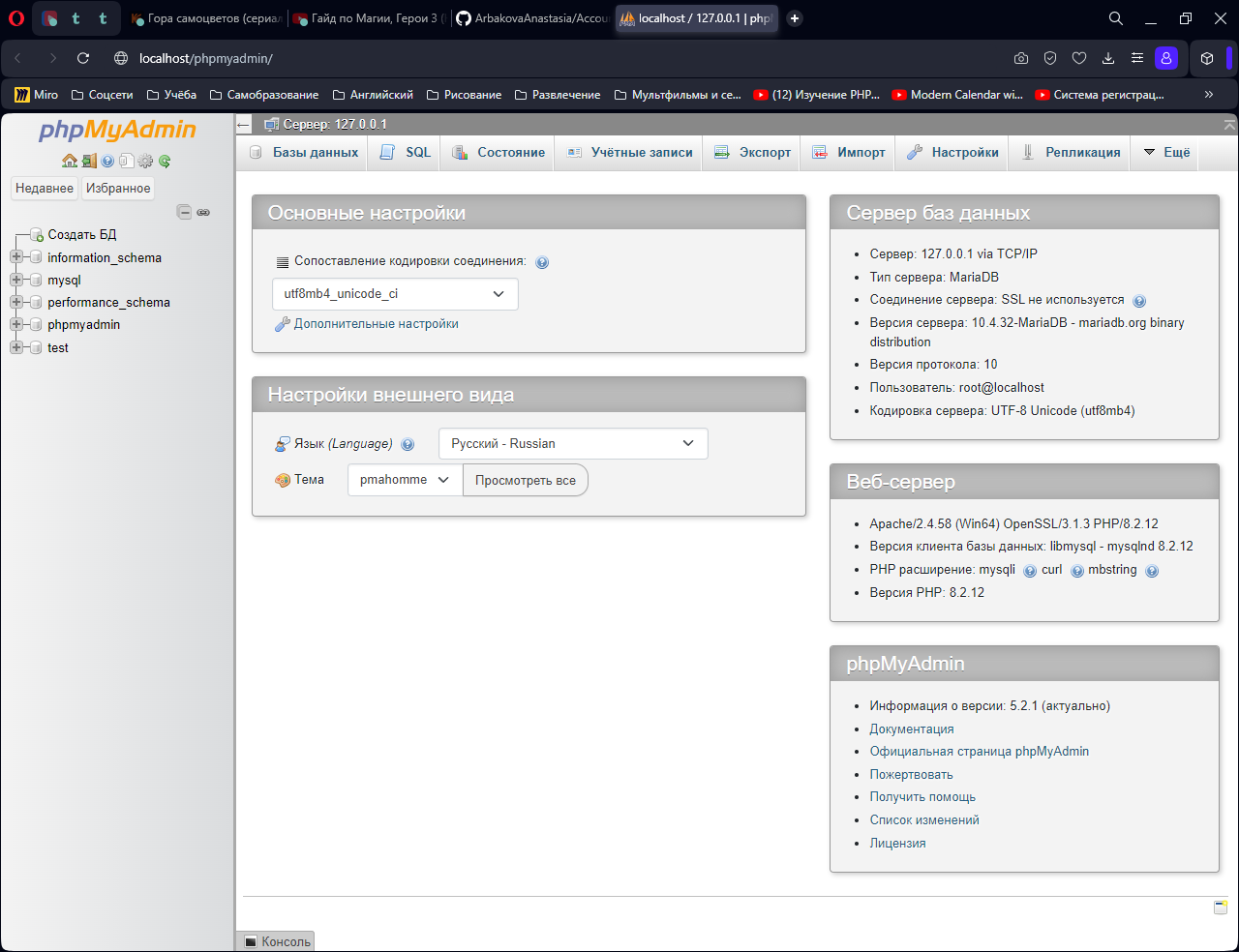


Рисунок 9 – PhpMyAdmin

Дальше необходимо создать БД, нажмите на кнопку «Создать БД» и введите название БД, указанное в файле конфигурации. Последующий шаг, это импорт БД (см. рисунок 10).

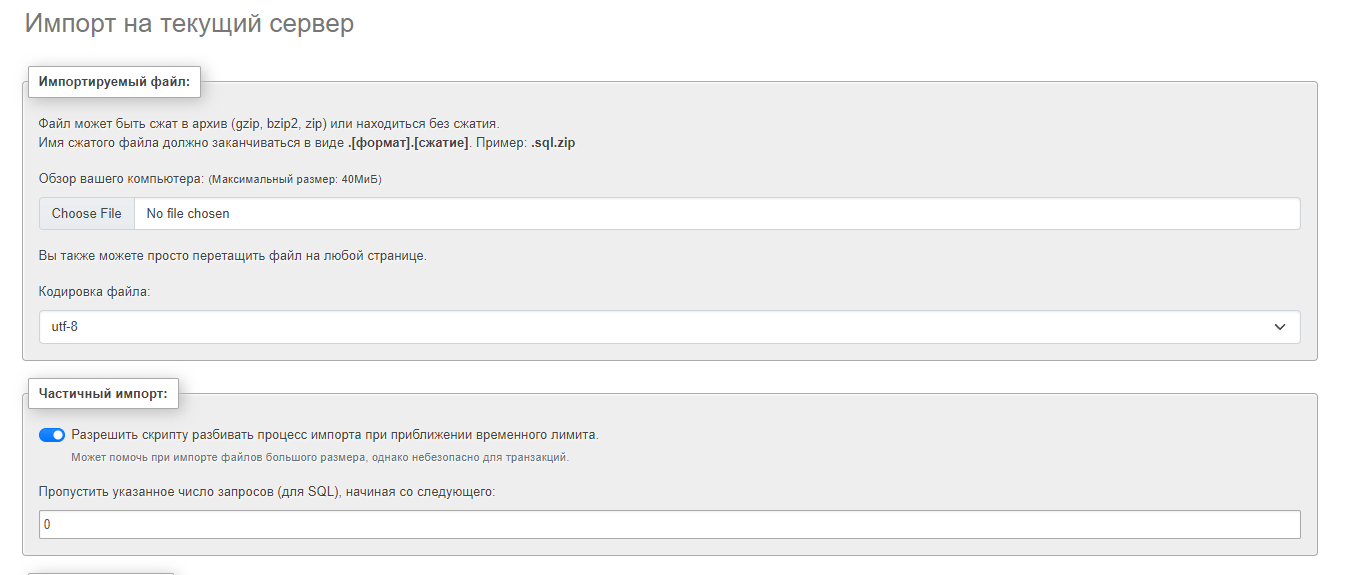


Рисунок 10 – Импорт БД

Конечный вид БД представлен на рисунке 11.

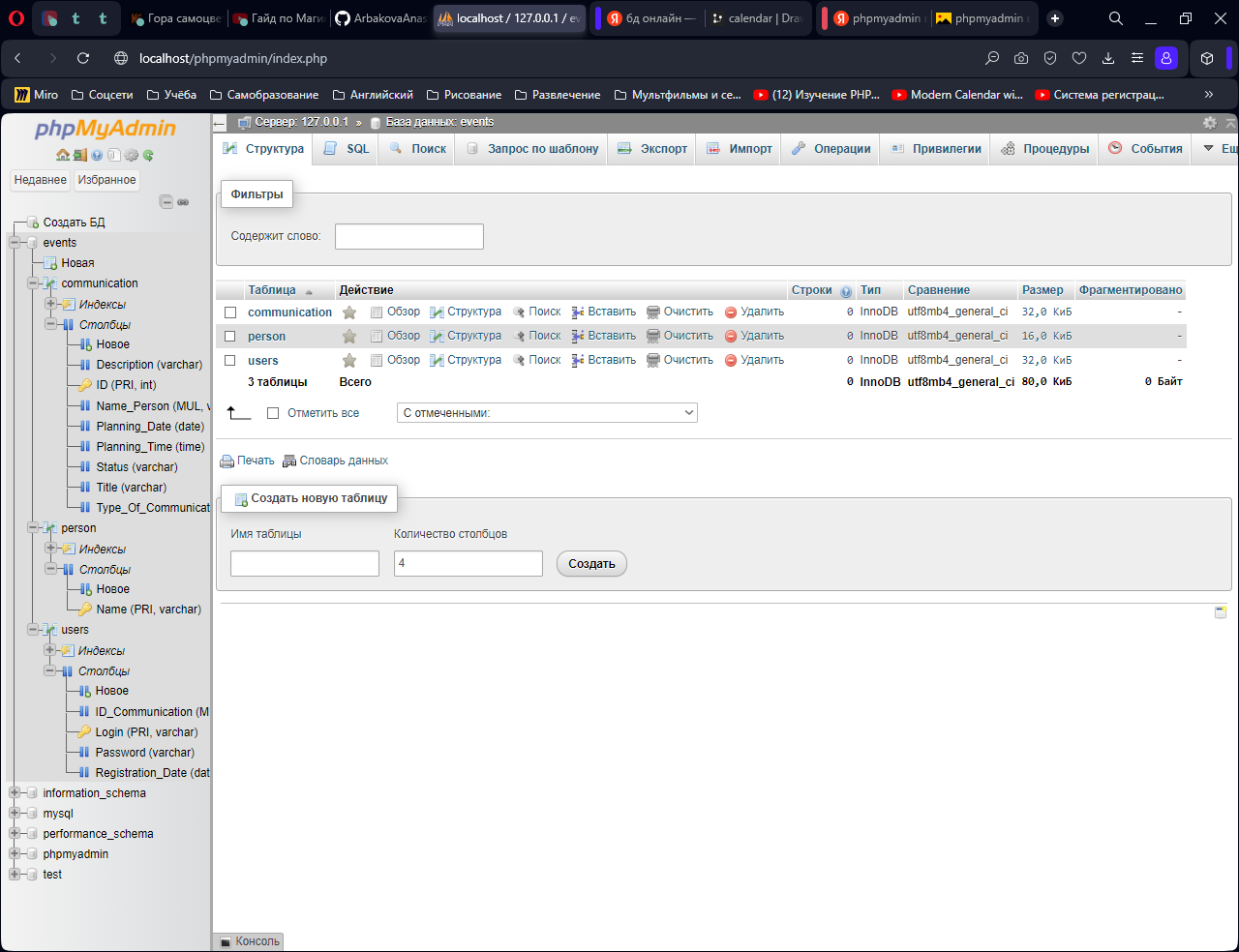


Рисунок 11 – БД в PHPMyAdmin

# **Структура пользовательского интерфейса**

Верхнеуровневое определение экранов – это процесс анализа и определения основных элементов пользовательского интерфейса, которые будут использоваться для взаимодействия пользователя с системой или приложением.

Верхнеуровневое определение экранов представлено на рисунке 12.

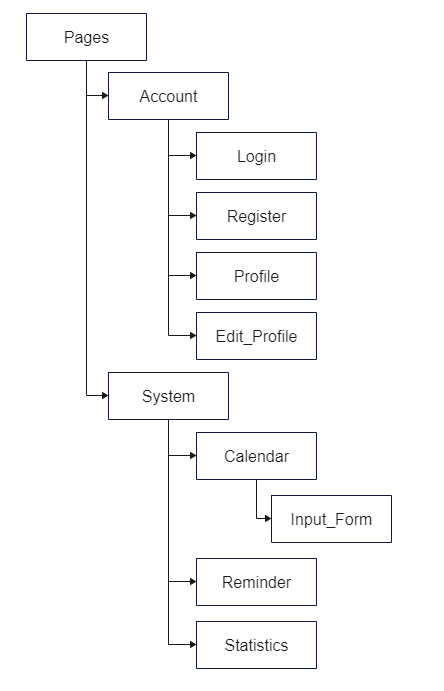


Рисунок 12 – Верхнеуровневое определение экранов

Имеются разделы “Aссount” и “System”.

Раздел “Aссount” состоит из таких частей, как: вход (Login), регистрация (Register), профиль (Profile), редактирование профиля (Edit\_Profile).

Раздел “System” состоит из таких частей, как: календарь (Calendar), напоминания (Reminder), статистика (Statistics).

Из календаря (Calendar) вызывается форма ввода коммуникации (Input\_Form).

Назначение экранов – это процесс определения того, какой экран или экраны будут использоваться для отображения определенной функции или задачи в приложении или системе.

Таблица назначения экранов представлена на рисунке 13.

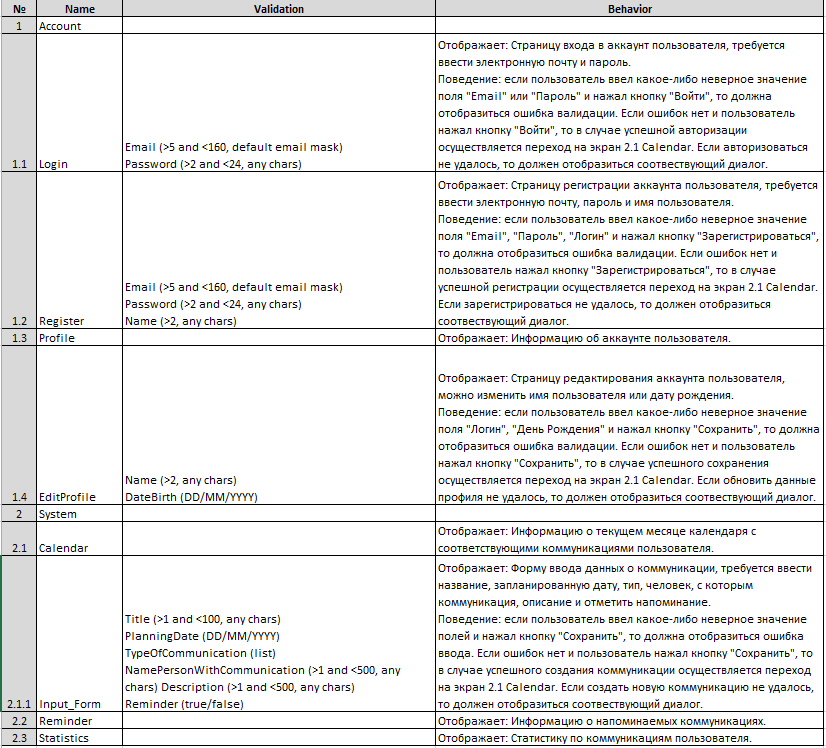


Рисунок 13 – Назначение экранов

Ключевые столбцы таблицы:

1. Номер экрана
2. Краткое название (Name)
3. Поля ввода для валидации (Validation)
4. Описание экрана и его поведения (Behavior)

Номер экрана – используется для упорядочивания экранов в приложении.

Краткое название (Name) – краткое и понятное название экрана, которое помогает другим разработчикам быстро понять, о каком экране идет речь.

Поля ввода для валидации (Validation) – список правил, которые проверяются при вводе данных пользователем. Это помогает гарантировать, что данные, введенные пользователем, соответствуют требованиям приложения.

Описание экрана и его поведения (Behavior) – подробное описание того, как работает экран, какие функции он выполняет и как взаимодействует с другими экранами приложения.

Карта экранов – это визуальное представление всех экранов в приложении или системе, упорядоченных и связанных друг с другом. Карта экранов представлена на рисунке 14.

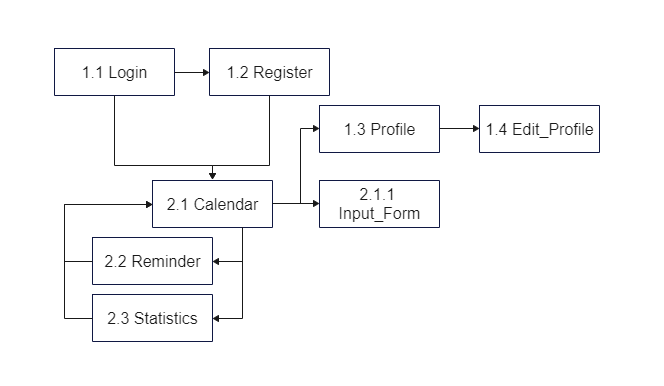
[](https://github.com/ArbakovaAnastasia/AccountingForCommunications/raw/main/%D0%9F%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5/%D0%BA%D0%B0%D1%80%D1%82%D0%B0%20%D1%8D%D0%BA%D1%80%D0%B0%D0%BD%D0%BE%D0%B2.png)

Рисунок 14 – Карта экранов

На карте экранов изображены переходы из экрана входа на экран регистрации и на экран календаря.

Из экрана регистрации также имеется переход на экран календаря.

Из экрана календаря имеются переходы на экраны формы ввода коммуникации, профиля, напоминаний и статистики.

Из формы профиля имеется переход на экран редактирования профиля.

Из экранов напоминаний и статистики можно вернуться на экран календаря.