Федеральное государственное автономное образовательное учреждение высшего образования

«Нижегородский государственный университет

им. Н.И. Лобачевского»

Факультет института информационных технологий, математики и механики

Отчёт по проекту

«Офисная система»

Выполнили:

студент ф-та ИИТММ гр. 381908-4

Китаев П. И.

студент ф-та ИИТММ гр. 381908-1

Мясникова В. С.

Научный руководитель:

Лебедев И. Г.

Нижний Новгород

2021 г.

Содержание

[Содержание 2](#_Toc71106862)

[Введение 3](#_Toc71106863)

[Постановка задачи 5](#_Toc71106864)

[Руководство пользователя 6](#_Toc71106865)

[Руководство программиста 8](#_Toc71106866)

[Описание структуры программы 8](#_Toc71106867)

[Описание структур данных 10](#_Toc71106868)

[Описание алгоритмов 11](#_Toc71106869)

[Эксперименты 14](#_Toc71106870)

[Работа сервера 14](#_Toc71106871)

[Работа клиентской программы 14](#_Toc71106872)

[Заключение 20](#_Toc71106873)

[Литература 21](#_Toc71106874)

Введение

Существует множество способов взаимодействия работников в организации, самые распространенные – звонки по служебному телефону (АТС), электронные письма, личная беседа.

У организации, которая только начинает свой нелегкий путь достаточно трат и без приобретения АТС, поэтому данный способ подходит не всем, и к тому же может быть не удобен в конкретных ситуациях.

Личная беседа доступна почти всегда, но ведь часто сотрудник занят своими делами, а путь и диалог, который может плавно перейти в дружескую беседу отнимут значимое количество ценного времени. Да и услышанная информация может попросту забыться, придется тратить время на восполнение пробелов. Сомнительная затея.

В наше время всегда можно написать сообщение в популярный мессенджер, но что делать, если человек банально забыл телефон дома, или руководство ругает за социальные сети на рабочем месте, а если доступ вовсе заблокирован? Опять не вариант.

На фоне всего перечисленного идеальным вариантом кажется корпоративный чат. Мгновенные сообщения с уведомлениями, с информацией о том, кто онлайн и историей переписки. Взамен, или в дополнение к традиционным собраниям, руководство может разослать задания всем своим подчиненным.

А что, если к чату добавить еще несколько полезных функций?

Например, личные заметки, информацию об организации, чтобы всегда знать почту, телефон, ИНН и прочие данные о месте работы. Функцию безопасности для службы охраны.

Всё это предоставляет офисная система, которая позволяет слаженно работать команде за счет быстрой коммуникации и выполнять индивидуальные задания качественно и в срок.

Если говорить подробнее, то функция использования заметок позволяет сотрудникам компании структурировать свои задачи и самостоятельно контролировать их выполнение, что экономит время и стикеры в офисе.

Функция безопасности дает возможность службе охраны контролировать открытие и закрытие дверей в том числе в ночное время. Благодаря этому снижается риск незаконного проникновения в помещения организации.

Актуальность данной офисной системы заключается в упрощении организации как личной, так и командной работы в компании. Четкое выполнение заданий сотрудниками позволяет улучшить качество и повысить эффективность работы коллектива, экономя время персоналу, потребителям продукции данной компании, а также деловым партнерам.

Постановка задачи

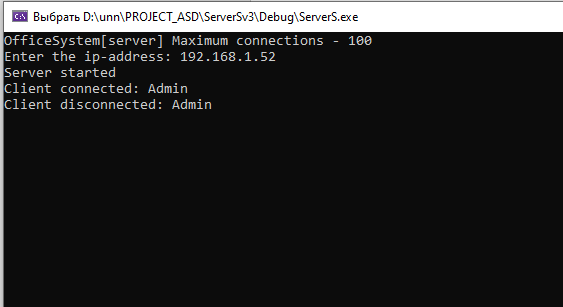
Необходимо реализовать комплекс программ, объединенных в приложение, которое позволит ускорить процесс работы в офисе за счет простой коммуникации и использования заметок, а также поможет предотвратить незаконные проникновения.

Для успешной реализации проекта, нужно:

* Составить требования к программе,
* Разработать необходимые алгоритмы,
* Определить инструменты, позволяющие реализовать необходимые функции,
* Изучить необходимый материал,
* Составить модель общения клиента с сервером,
* Разработать серверное приложение, которое будет отвечать за обработку пользовательских запросов,
* Реализовать клиентское приложение с минимальным функционалом,
* Реализовать работу микроконтроллера для снятия показаний с датчиков безопасности,
* Реализовать чат сотрудников,
* Реализовать систему заметок,
* Реализовать функцию охраны помещений,
* Произвести отладку, тестирование и устранение недочетов.

Руководство пользователя

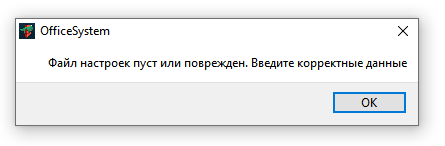
Работа сервера

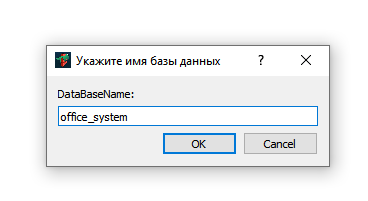


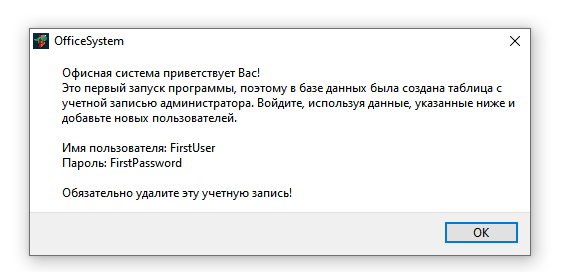
При запуске сервера требуется ввести IP-адрес.

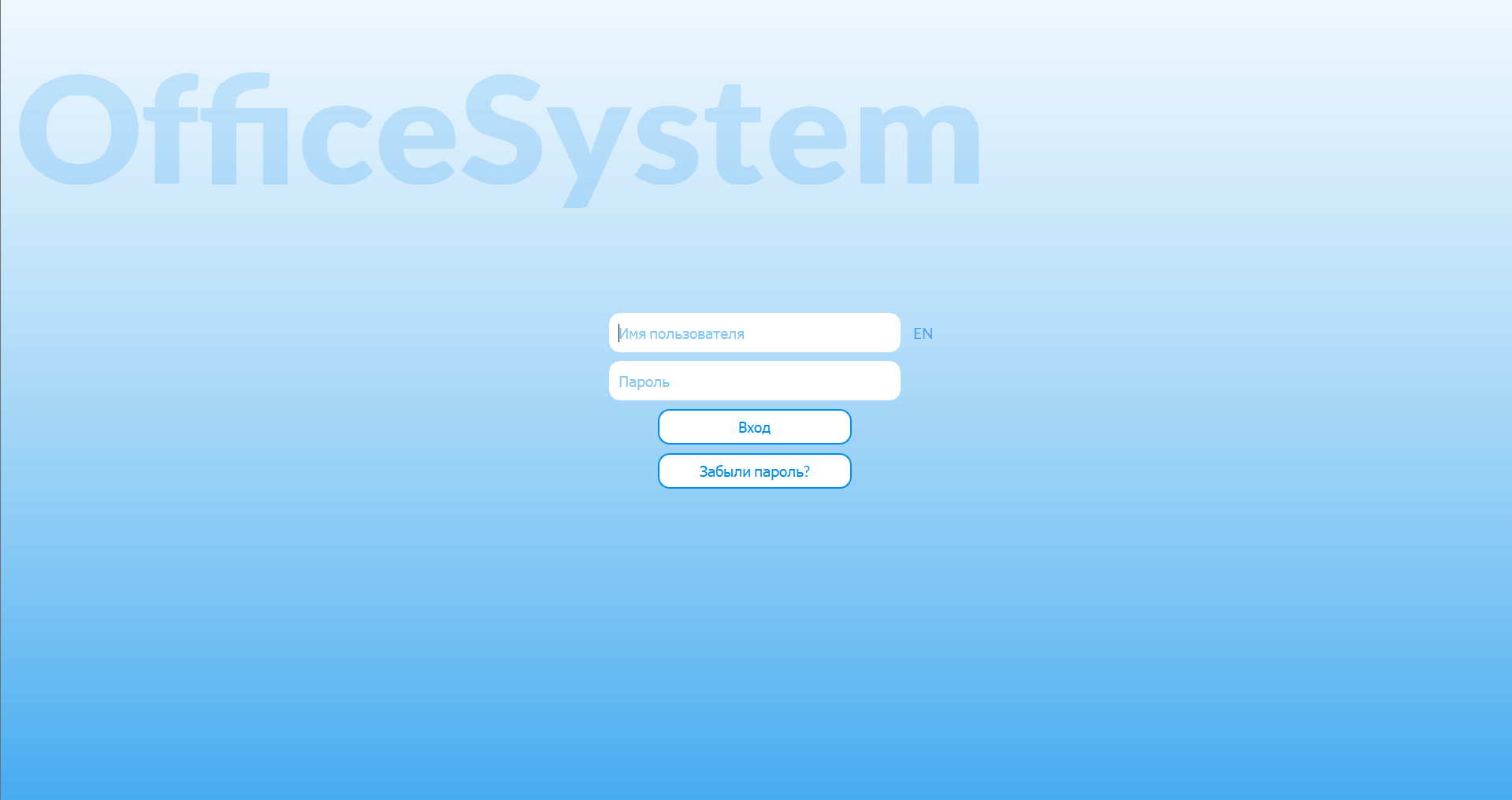
Работа клиентской программы

Уведомление о пустом или поврежденном файле настроек и его первичное заполнение:

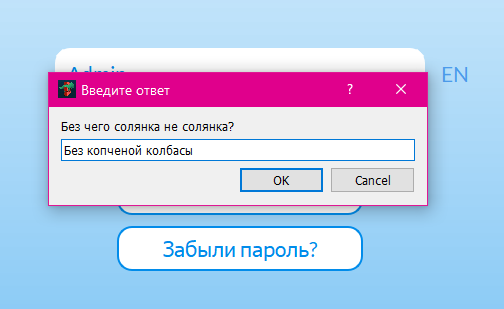
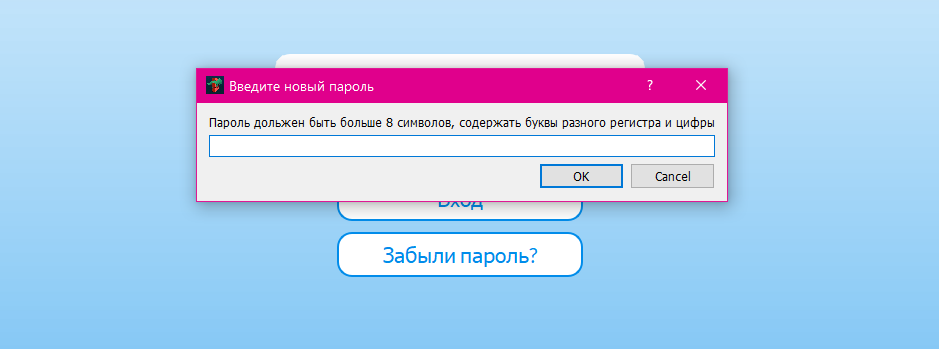




Приветственное окно при первом запуске:

При входе в приложение Вам предлагается ввести свой логин и пароль. 

Далее, если Вы забыли пароль, то для того, чтобы его восстановить, необходимо ответить на ранее установленный секретный вопрос. В случае его отсутствия нужно обратиться к администратору для сброса пароля.



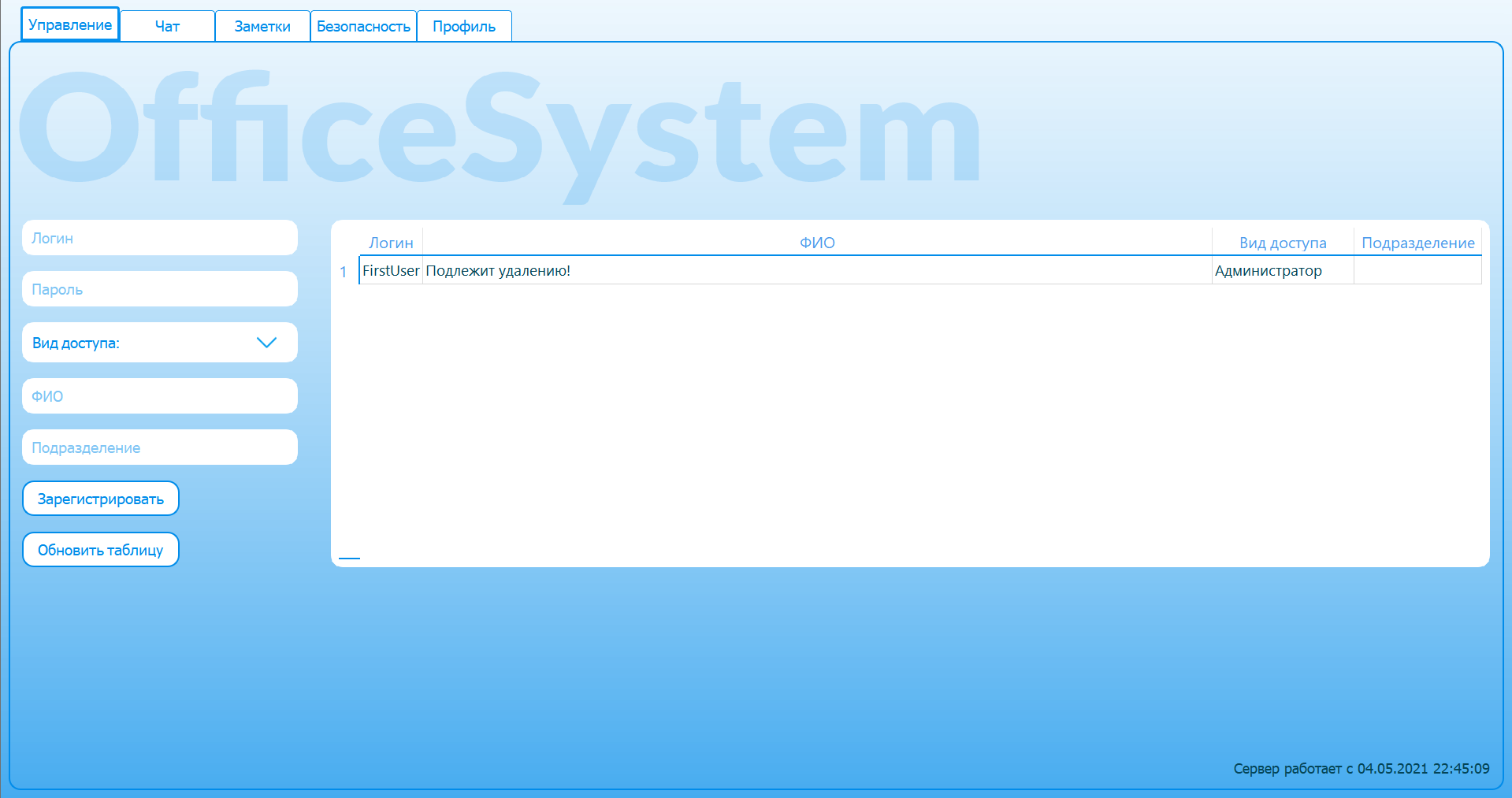
Если авторизация успешна, то в зависимости от вида доступа, который назначил администратор, Вам будут доступны следующие функции:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Управление | Чат | Заметки | Безопасность | Профиль |
| Администратор | + | + | + | + | + |
| Пользователь |  | + | + |  | + |
| Охрана |  | + | + | + | + |

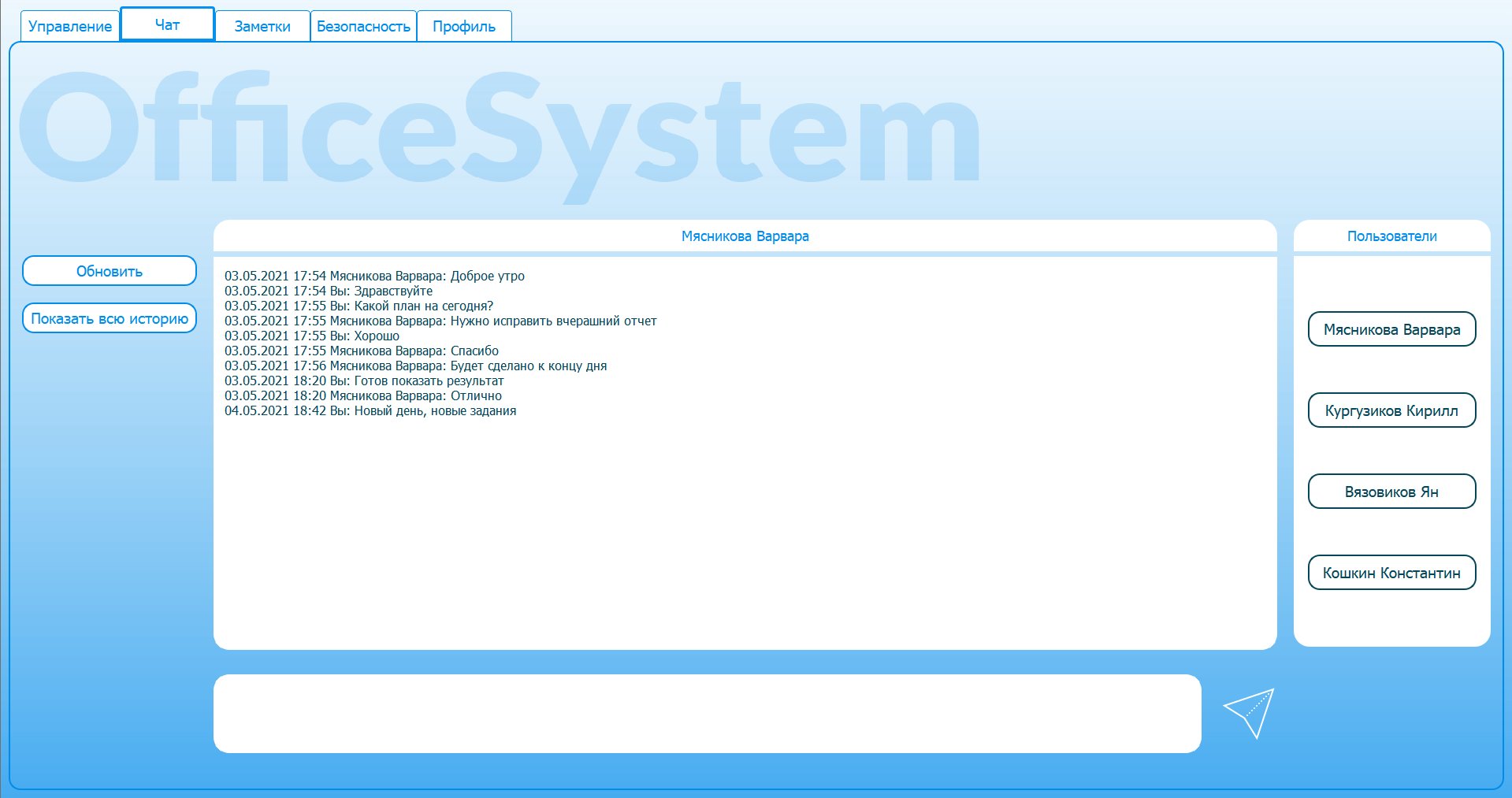
*Таблица 1. Соответствие функций видам доступа.*

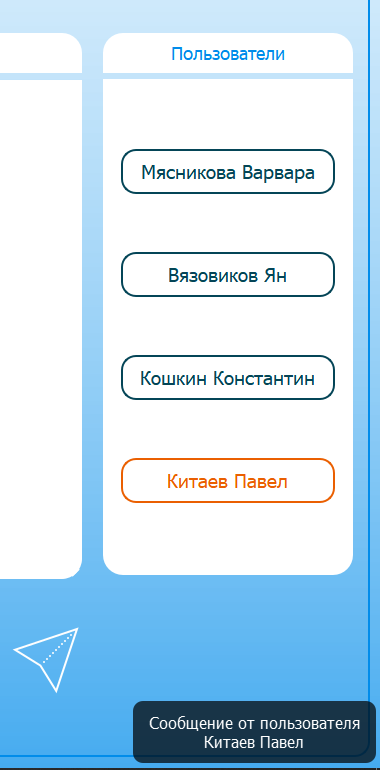
Рассмотрим все возможности интерфейса.

На вкладке управления есть таблица с данными всех пользователей системы. Здесь администратор может управлять данными пользователей, добавлять новых или удалять уже существующих пользователей, а также сбрасывать пароль.

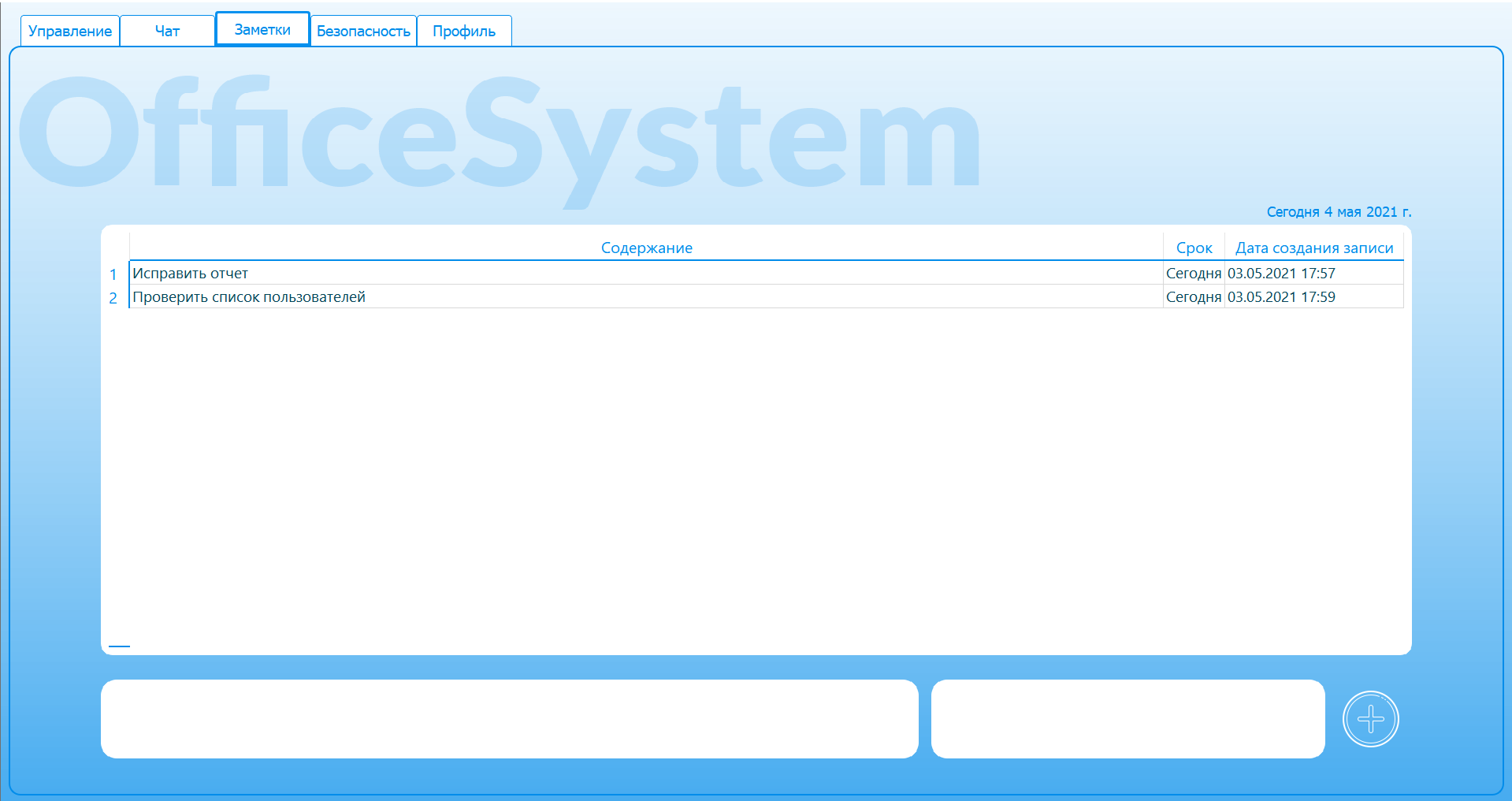


Вкладка чат подразумевает взаимодействие сотрудников. Через чат можно договориться с коллегой о выполнении какого-либо задания, или просто позвать на перерыв, разумеется в установленное для этого время. Так же руководитель, к примеру, с легкостью может разослать всем своим подчиненным задания на день, а адресаты получат уведомления о новых сообщениях.



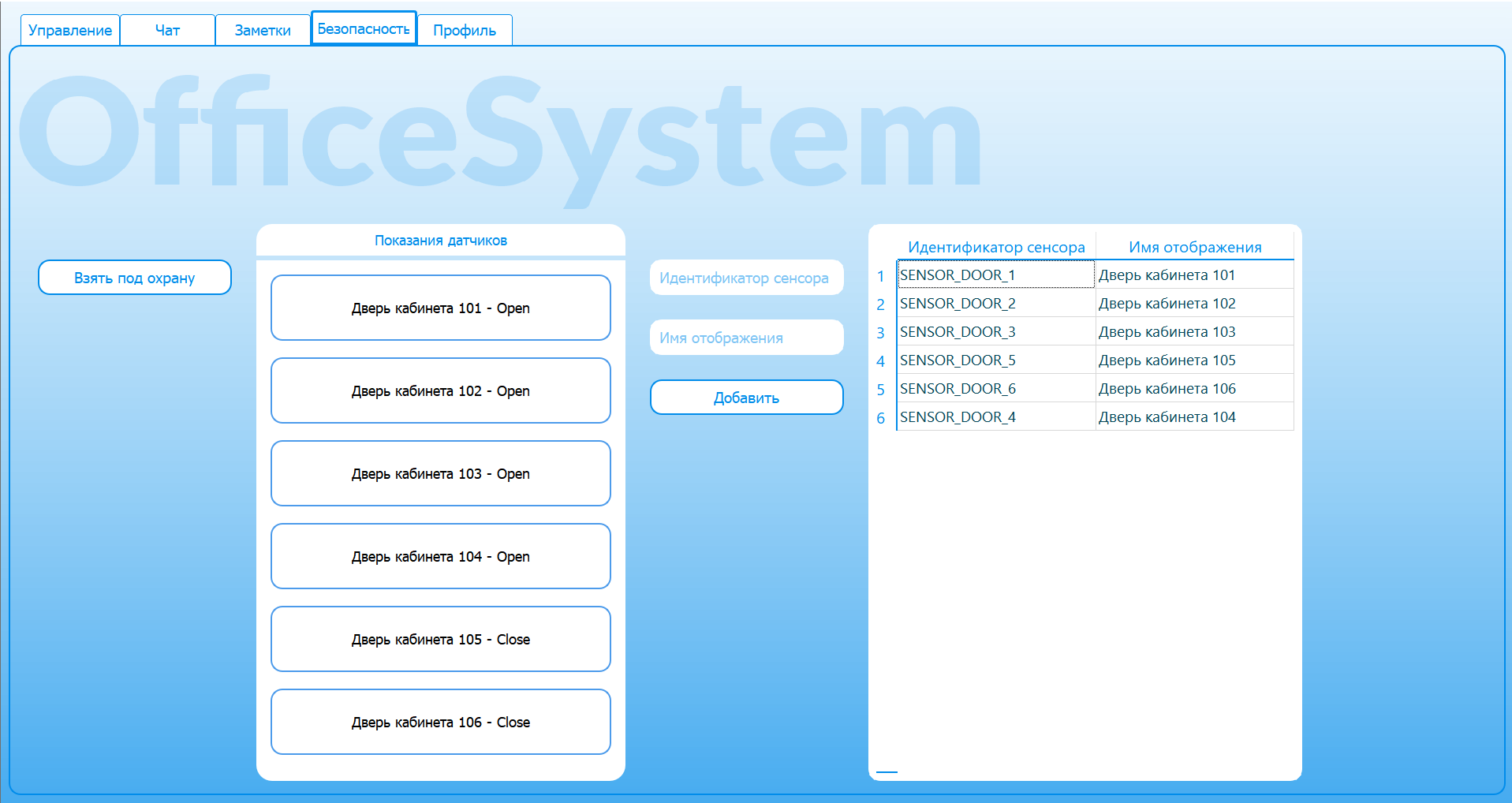


Заметки позволяют зафиксировать задания и срок их выполнения. Без проблем подойдут и для записи любой важной для пользователя информации.

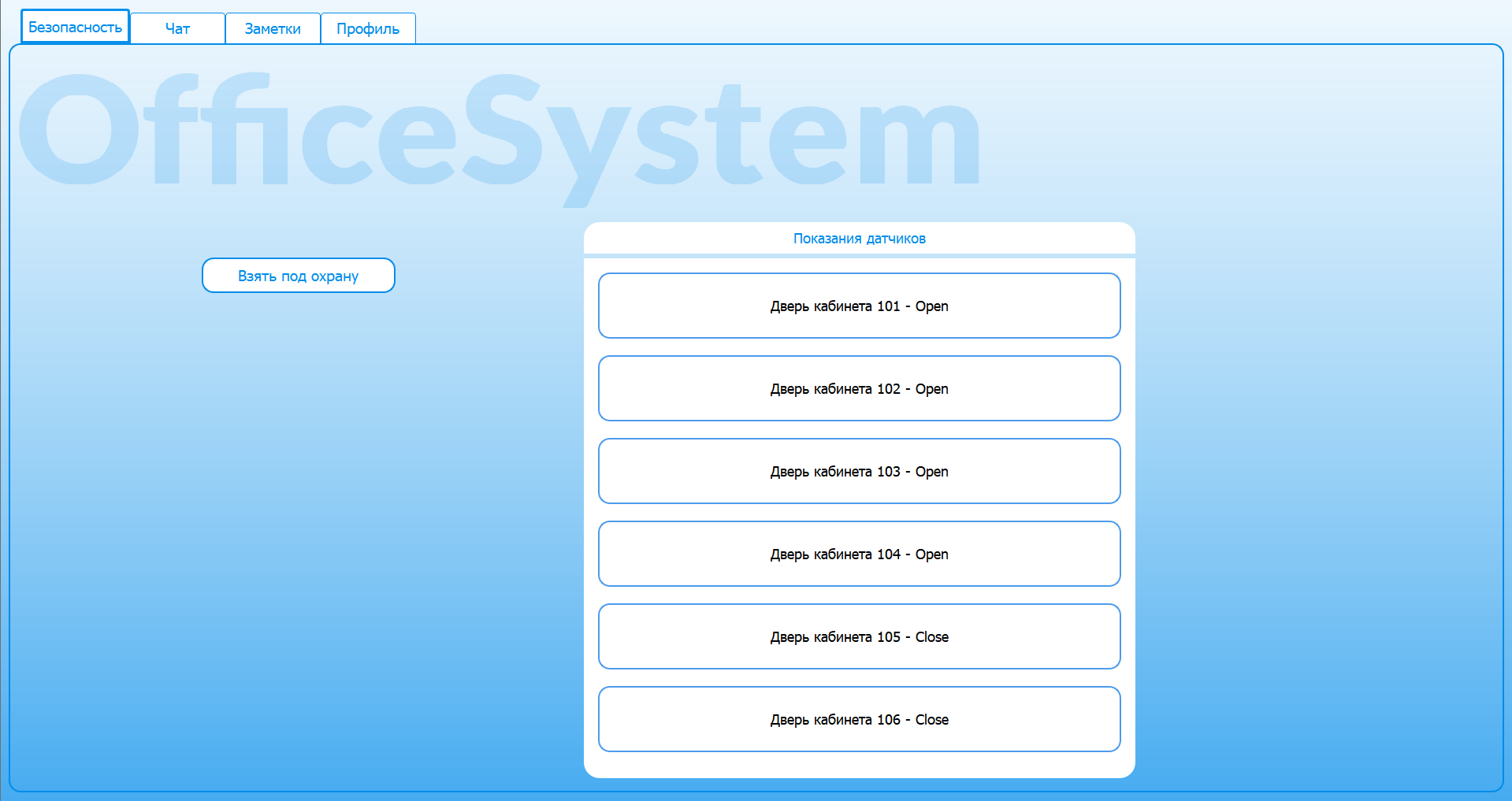


Вкладка безопасности предусмотрена для охраны. Охранник может отслеживать показания датчиков и брать под охрану помещения. Администратор в свою очередь может добавить новые датчики или изменить данные о существующих.

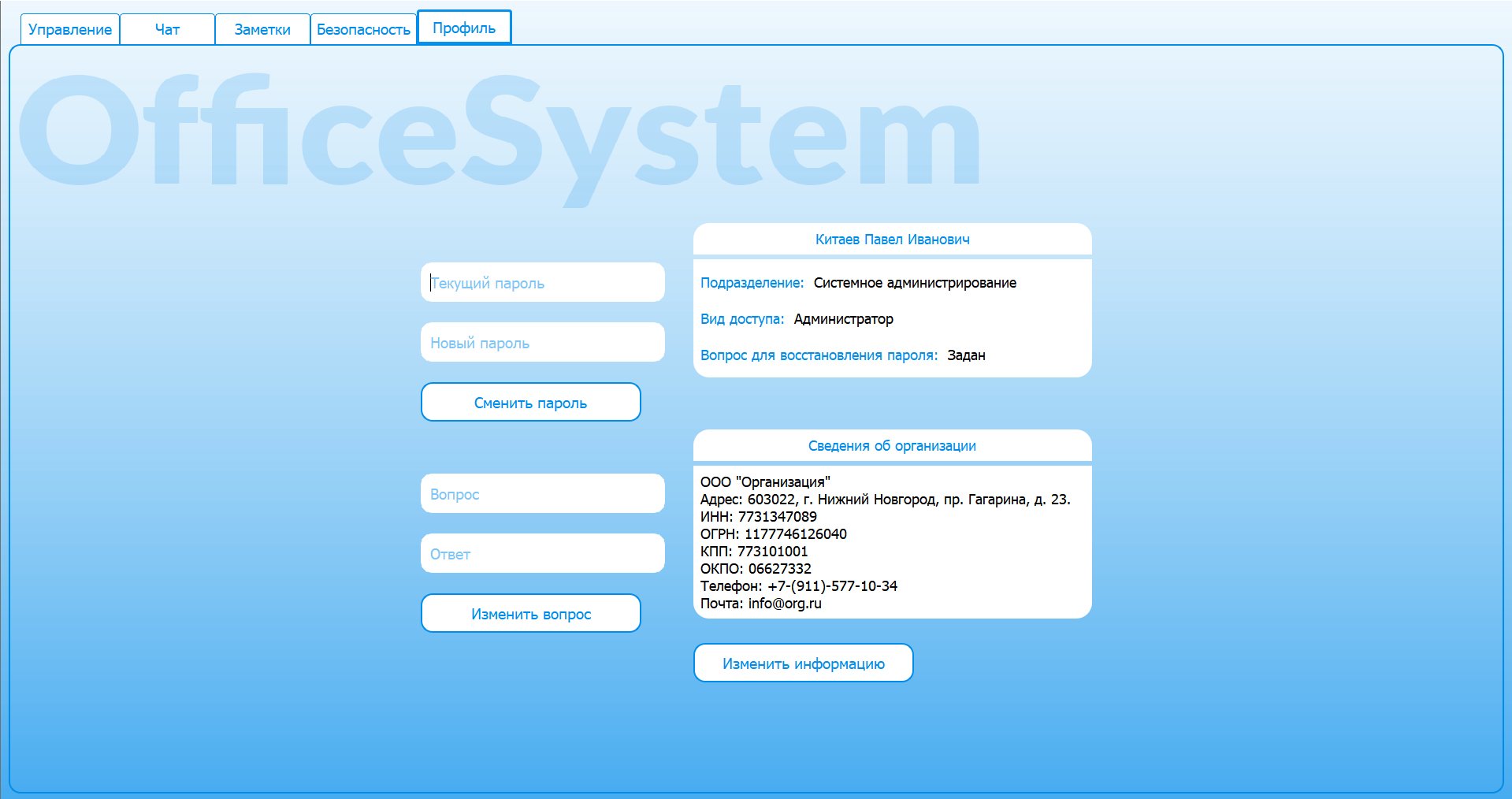
Безопасность (Администратор):



Безопасность (Охранник):



Профиль, как чат и заметки, предусмотрен для всех пользователей. Здесь указана информация о пользователе, а именно ФИО, подразделение, вид доступа, наличие секретного вопроса и сведения об организации. Также в этой вкладке можно изменить пароль и свой секретный вопрос.



Руководство программиста

**Описание структуры программы**

1. Серверная часть

Сервер разработан на языке программирования C++ с использованием Visual Studio 2019. За основу взята библиотека WinSock2.

Определяется максимальное количество клиентов и запускается цикл с ожиданием нового подключения. При подключении к серверу создается новый поток, в котором происходит принятие сообщений. При этом, в основном потоке происходит ожидание нового подключения.

В случае отключения клиента, сервер получает соответствующий сигнал в виде команды, производит удаление информации о клиенте и завершает поток. Если клиент разорвал соединение некорректно, например, при отключении питания, то сервер отследит этот момент и произведет отключение пользователя от сервера.

Реализованные команды:

* NEW\_CLIENT::Login – Команда посылается клиентом при подключении. Сервер вносит индекс подключения и полученный логин в соответствующую структуру данных,
* SECURITY\_CLIENT::Login – Вносит индекс подключения и логин в контейнер, который используется для получения клиентов, кому предназначены данные о состоянии датчиков безопасности,
* DISCONNECTIONG\_CLIENT::Login – Команда посылается при отключении клиента от сервера. Происходит удаление из структуры данных индекса подключения и логина,
* GET\_USERS\_ONLINE:: – Команда запроса подключенных пользователей в данный момент,
* MSG::SenderLogin::RecipientLogin::TextMessage – Команда отправки сообщения пользователю. Указывается сначала логин отправителя, затем получателя и в конце само сообщение,
* SERVER\_DATATIME:: – Команда от клиента, на которую ответом является дата и время запуска сервера,
* GET\_SENSORS\_DATA:: - Команда запроса данных с устройства безопасности,
* SAFETY::Data – Команду отправляет микроконтроллер. Поле “Data” содержит данные о состоянии датчиков безопасности.

1. Клиентская часть

Клиентское приложение разработано на языке программирования С++ с использованием фреймворка Qt.

Приложение состоит из 4 основных графических форм:

* mainform.ui – форма авторизации пользователя,
* admform.ui – форма для роли «Администратор»,
* mngform.ui – форма для роли «Пользователь»,
* scrform.ui – форма для роли «Охранник».

После авторизации в форме mainform происходит определение вида доступа и открывается одна из остальных форм – admform, mngform или scrform.

Реализованные классы:

* mainform.h – класс формы авторизации,
* admform.h – класс формы для роли «Администратор»,
* mngform.h – класс формы для роли «Пользователь»,
* scrform.h – класс формы для роли «Охранник»,
* chatbase.h – класс чата и принятия сообщений от сервера,
* notes.h – класс заметок,
* safety.h – класс безопасности,
* userprofile.h – класс профиля,
* log.h – класс журнализирования действий,
* settingfile.h – класс для работы с файлом настроек,
* officesystem.h – статические методы,
* popnotify.h – класс уведомлений.

1. Устройство системы безопасности

Система разработана с использованием микроконтроллера Espressif32 и средой разработки PlatformIO с фреймворком ESP-IDF.

Микроконтроллер подключен к локальной сети с использованием беспроводных технологий. При получении команды от сервера происходит опрос датчиков и формирование итогового сообщения для отправки на сервер и дальнейшей пересылке клиенту с соответствующим видом доступа. Таких клиентов может быть несколько.

**Описание структур данных**

1. Серверная часть

В программе используется контейнер map<int, string> clients для хранения пары индекс – логин. В этот контейнер заносятся данные в момент подключения пользователя. При отключении – удаляются. Служит для хранения информации о клиентах, которые онлайн.

Контейнер map<int, string> securityClients предназначен для хранения индекса с логина клиента, который имеет вид доступа охранник или администратор. Необходим для того, чтобы сервер знал, кому требуется отправить данные о датчиках в случае необходимости.

Есть массив данных SOCKET Connections[MAX\_CLIENTS], куда заносятся подключения, созданные для каждого конкретного пользователя.

1. Клиентская часть

В программе основные данные хранятся в базе данных PostgreSQL.

Данные, необходимые для подключения к базе данных, и серверу хранятся в соответствующем файле – «setting.ini». В случае отсутствия данных в этом файле, программа предложит ввести их вручную, после чего сохранит в файл.

В классе «chatbase» имеются следующие структуры:

QVector<QPushButton\*> btns – хранит указатели на кнопки, которые используются для отображения пользователей в чате. Необходим для удаления или редактирования существующих кнопок и создания новых,

QVector<QString> unread – хранит логины пользователей, с кем имеется непрочитанный диалог,

QVector<QString> allUsersVec – хранит логины всех пользователей системы,

QVector<QString> userOnline – хранит логины пользователей, которые в данный момент находятся в сети,

QMap<QString, QString> chatName – хранит пару логин – имя пользователя для отображения в списке пользователей.

Класс «safety» содержит контейнеры:

QVectrol<QLabel> lbs – указатели на Label, которые содержат информацию с датчиков, для их дальнейшего редактирования,

QMap<QString, QString> dataTable – хранит соответствия идентификаторов датчиков именам для отображения. Получает информацию из базы данных,

**Описание алгоритмов**

1. Серверная часть

При запуске программа запросит ip-адрес, на котором следует запустить сервер, затем инициализируется библиотека WinSock2, задается ip-адрес и порт, создается сокет. Начинается процесс ожидания подключений.

При подключении клиент получает уникальный сокет и создается новый поток, в который передается индекс сокета. В потоке происходит слушание созданного сокета в ожидании сообщений от клиента. При этом, в главном потоке продолжается ожидание клиентов.

Следует не допускать переполнения подключений, иначе произойдет остановка программы. Сервер рассчитан на 100 одновременных подключений.

1. **Клиентская часть**

При первом запуске программы запрашиваются данные для подключения к базе данных и создаются все нужны для функционирования таблицы в базе. Также создается первая учетная запись с видом доступа «Администратор» для регистрации всех пользователей. Первичная учетная запись в целях безопасности должна быть обязательно удалена!

После добавления администратором пользователей в программу, сотрудники организации могут полноценно использовать систему.

При запуске программы появляется форма с вводом имени пользователя и пароля. При нажатии кнопки «Вход» программа делает запрос к базе данных, где проверяет, существует ли данный логин в системе, и если логин существует, происходит сверка паролей. Если пароли совпадают, программа получает вид доступа для данного пользователя и открывает соответствующую форму.

Если на этом этапе программа не может подключится к серверу, то появляется соответствующее окно с вариантами решения проблемы. Ввести новые данные для подключения или продолжить без сервера. Если выбран второй вариант, то программа продолжит функционировать в штатном режиме, но без возможностей чата и безопасности.

Если нажата кнопка «Забыли пароль» – программа проверит, есть ли информация для восстановления пароля. Если информация имеется, пользователю будет предложено ввести ответ на ранее созданный секретный вопрос и в случае правильного ответа – сменить пароль. Если пользователь не создавал вопрос, или не может на него ответить, ему следует обраться к администратору, который имеет возможность сбрасывать пароли пользователей.

В зависимости от вида доступа будет различаться функционал системы:

* Для пользователя будет доступен чат, заметки, вкладка профиля,
* Для администратора доступны чат, заметки, профиль, вкладка управления, вкладка безопасности с возможностью редактирования данных,
* Для охранника доступны чат, заметки, профиль, вкладка безопасности без возможности редактирования данных.

Вне зависимости от вида доступа везде изначально происходит инициализация необходимых классов, переменных, компонентов и вызывается функция размещения созданных компонентов. Затем в зависимости от действий пользователя выполняются полезные функции.

Также в программе предусмотрена система ошибок, которые записываются в текстовый документ logfile.log:

* #С001 – Несоответствие сетевой библиотеке. Возникает, если версия библиотеки WinSock2 ниже минимально допустимой,
* #L002 – Ошибка подключения к серверу. Сервер недоступен либо отсутствуют данные для подключения,
* #L003 – Данный пользователь уже в сети,
* #L004 – Файл setting.ini поврежден,
* Все ошибки, связанные с базой данных, предоставляет библиотека QScqError.

1. **Устройство системы безопасности**

Устройство подключается к беспроводной сети. Далее устанавливается связь с сервером. Микроконтроллер ожидает команду на передачу данные. При получении команды, происходит опрос датчиков и формирование итогового сообщения и его незамедлительная отправка на сервер. В свою очередь, сервер направляет сообщение клиентам, которые зарегистрированы как «охранники» или администраторы.

Заключение

Выделим, что мы сделали за время проекта.

В процессе реализации проекта было изучено следующее:

1. Основы работы с фреймворком Qt,
2. Принцип работы со стилями Qt для реализации дизайна приложения,
3. Собственные библиотеки Qt,
4. Функции системы сигнал-слот в Qt,
5. Библиотека Winsock2.h в C++,
6. Основы работы с микроконтроллером ESP32.
7. Среда разработки PlatformIO в паре с фреймворком ESP-IDF.

Также были применены знания о работе с базами данных, полученные на соответствующем учебном предмете.

Можно сделать вывод, что данная офисная система станет отличным помощником в организации за счет своей простоты, как в использовании, так и в развертывании. Обеспечит быструю и легкую коммуникацию между сотрудниками, поможет структурировать индивидуальные дела с помощью заметок, а охрану обеспечит дополнительными полезными данными.

Для полноценного использования нашей офисной системы, администратору достаточно создать базу данных и указать программе информацию для подключения к ней, а все необходимы таблицы программа создаст автоматически.

За счет применения простого микроконтроллера и датчиков, которые доступны по очень низким ценам, данную систему может позволить себе абсолютно каждая организация на любом этапе существования.

Литература

«Язык программирования C++. Лекции и упражнения». Автор Прата Стивен. ISBN: 978-5-907114-00-5

«Qt4.5. Профессиональное программирование на C++». Автор Макс Шлее. ISBN: 978-5-9775-0398-3

«Qt4.7+. Практическое программирование на C++». Автор Андрей Боровский. ISBN: 978-5-9775-0757-8

Книга Колбана о ESP32, автор Нил Колбан

Официальная документация Qt – <https://doc.qt.io/qt-5/classes.html>

Официальная документация PostgerSQL – <https://postgrespro.ru/docs/postgrespro/9.5/>