

SIWOSU

+ CLOUD
♥

ХСLOUD: ИСТОРИЯ СОЗДАНИЯ

PRESNTATION



ПЛАН НА СЕГОДНЯ

1337

- | | | | |
|----|---------------|----|-------------------|
| 03 | ОБО МНЕ | 08 | ПРИМЕР КОДА #1 |
| 04 | ЦЕЛИ И ЗАДАЧИ | 09 | ПРИМЕР КОДА #2 |
| 05 | СЛОЖНОСТИ | 10 | КАК РАБОТАЕТ ПР-Е |
| 06 | БИБЛИОТЕКИ | 11 | ЗАВЕРШЕНИЕ |
| 07 | ПЛАТФОРМА | | |

ОБО МНЕ

Absolute??



Не первый год интересуюсь программированием, за эти 3-4 года сделал несколько проектов/программ. Такие как: шифрование текста, калькулятор систем счислений, система входа и регистрации и многое другое.

Сам же учусь в 9 классе, готовлюсь к экзаменам и большую часть времени потратил на данный проект который в будущем будет развиваться и развиваться.

SIWOSU

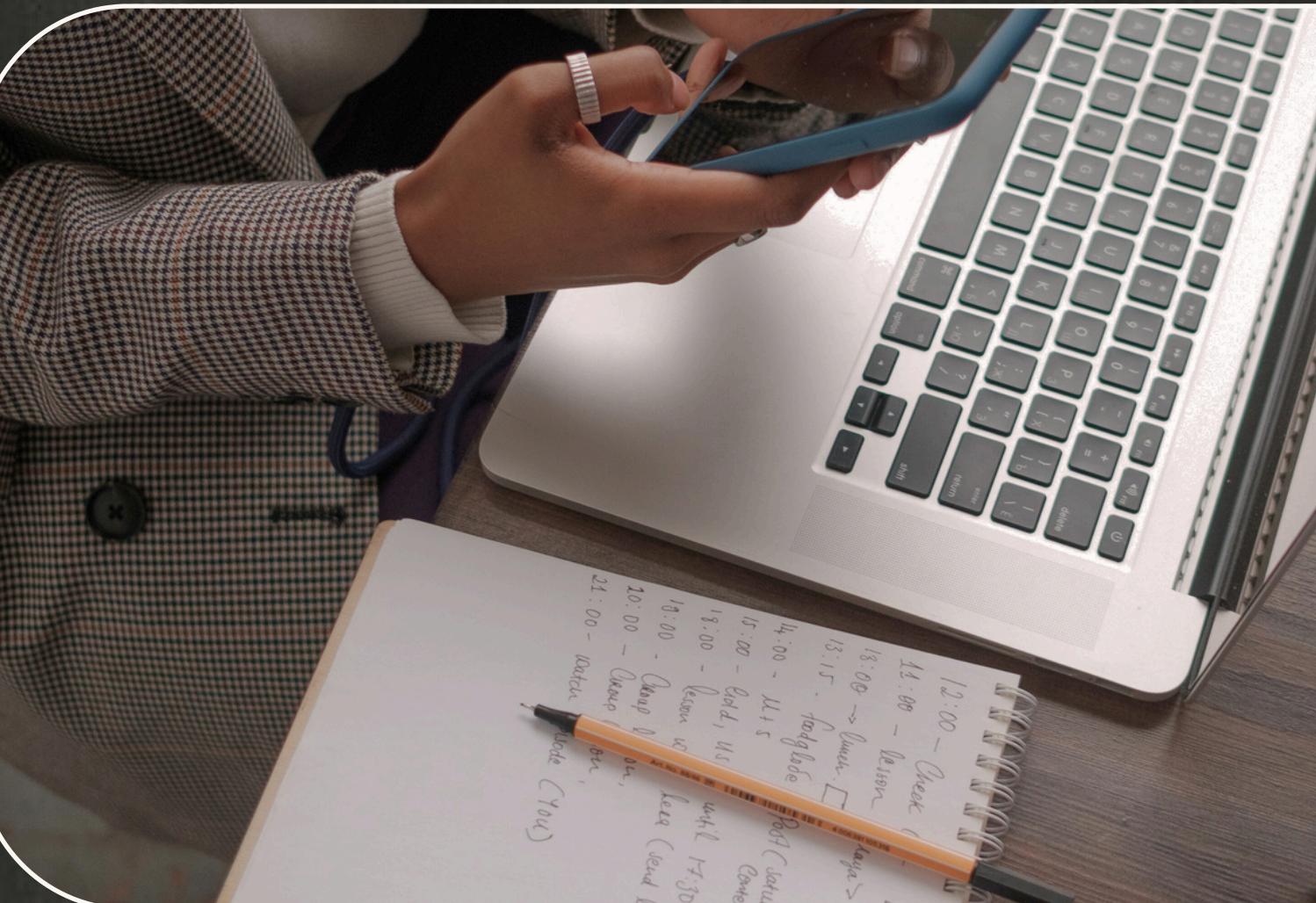
оз

ЦЕЛИ И ЗАДАЧИ

Моя цель заключалась в том, чтобы любой человек мог достаточно просто разобраться с моей программой и у него/неё не возникало вопросов. Я реализовал практически всё, что предполагал. Но, как я говорил ранее, я планирую развивать свой проект.

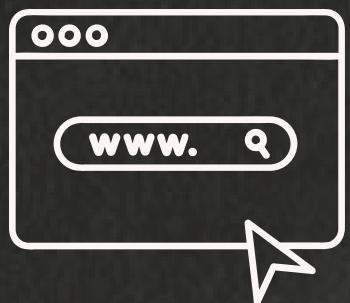
Задачи были таковы:

- 1) Найти подходящую платформу для создания программы
- 2) Разобраться с интерфейсом платформы
- 3) Найти документации/материалы которые мне помогут
- 4) Дизайн программы и многое другое.



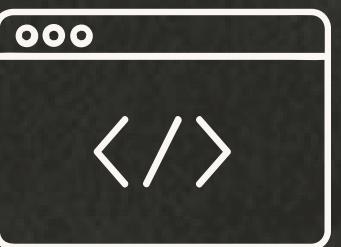
СЛОЖНОСТИ

АРТЕМ LOVE <3



CONNECT WITH DROPBOX

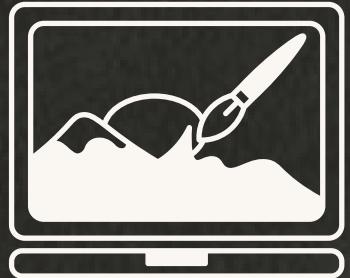
Изначально, я хотел использовать Google Cloud. Так, как этой платформой пользуется сама компания Apple для работы своего iCloud. Но из-за того, что Google Cloud работает не корректно в России, я отказался от этого варианта и выбрал DropBox.



RESET PASSWORD

Для сброса пароля, я хотел использовать SMTP от Google, но вариант оказался не удобен из за не точной настройки под C++ и фреймворк Qt. Альтернативой, я сделал бота в Telegram который присыпает 6-ти значный код на ваш аккаунт.

БИБЛИОТЕКИ



QSQL, QMYSQ “DATABSE”

Прежде всего, нужно было определиться с базой данных, я выбрал MariaDB. Это серверная база данных, предназначенная для более крупных проектов. Для просмотра и создания таблицы я использовал DBeaver и HeidiSQL.



HASH “SHA256”

Для безопасности пользователей я создал хеширование паролей через Sha256. Его плюсы: безопасность, быстрота выполнения, и фиксированная длина в 32 бита, что обеспечивает небольшой объем и размер выходных данных.

ПЛАТФОРМА

ПОЧЕМУ QT?



На Qt можно создавать программы для ОС таких как: Window, macOS и для разных дистрибутивов.



На данной платформе создано много популярных программ такие как: Telegram, Skype и т.д.



Мощный фреймворк GUI который позволяет легко разобраться с интерфейсом.



ПРИМЕР КОДА #1

AUTORIZATION

```
51 void MainWindow::on_pushButton_clicked()
52 {
53     // обработчик нажатия кнопки "Войти"
54
55     QString login = ui->login->text();
56     QString password = ui->password->text();
57
58     if (login.isEmpty() || password.isEmpty()) {
59         QMessageBox::critical(this, "Ошибка", "Пожалуйста, введите логин и пароль");
60         return;
61     }
62
63     // Выполняем запрос SELECT с условием на логин и хэшпароля
64     QSqlQuery selectQuery(db);
65     if (selectQuery.prepare("SELECT * FROM users WHERE login = :login AND password = :password")) {
66         selectQuery.bindValue(":login", login);
67         selectQuery.bindValue(":password", hashPassword(password));
```

Вот малый п-р кода. Здесь происходит обработчик кнопки “Войти”.

Так же идёт проверка на то, есть ли такой Логин/Пароль в базе данных.

ПРИМЕР КОДА # 2

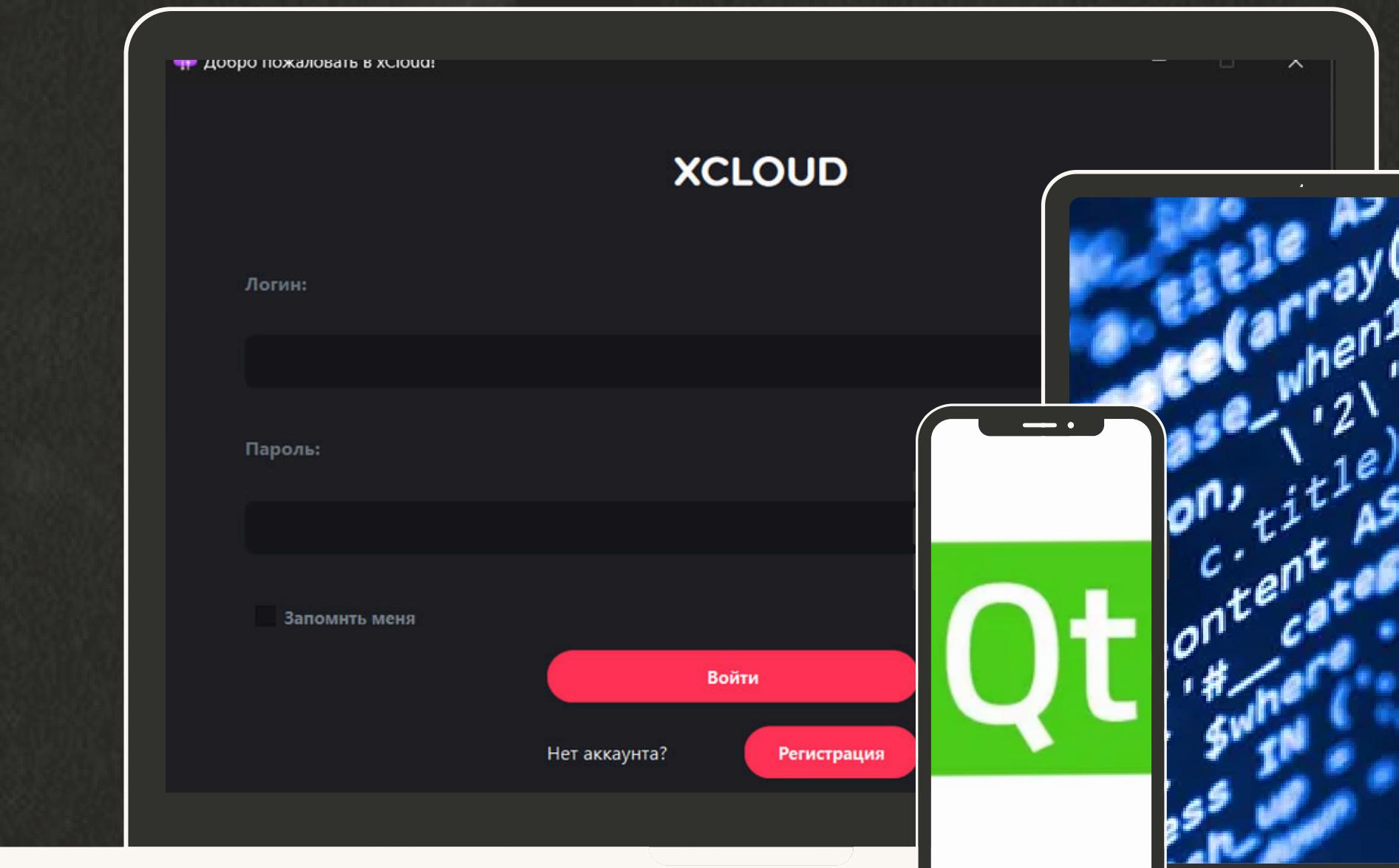
```
39 void registrationwindow::on_pushButton_clicked()
40 {
41     // Объявление статических QRegularExpression
42     static QRegularExpression nameRegex("^[a-zA-Zа-яА-Я]+$");
43     static QRegularExpression emailRegex("^[A-Za-z0-9]+([\\.-]?[A-Za-z0-9]+)*@[A-Za-z0-9]+([\\.-]?[A-Za-z0-9]+)*([\\.]?[A-Za-z]{2,})$");
44     static QRegularExpression loginRegex("^[a-zA-Z][a-zA-Z0-9\\-_.]*$");
45
46     // Проверка имени
47     QString name = ui->name_line->text();
48     if (name.isEmpty()) {
49         QMessageBox::warning(this, "Ошибка", "Введите имя");
50         return;
51     }
52
53     // Проверка размера имени (от 2 до 20 символов)
54     if (name.length() < 2 || name.length() > 20) {
55         QMessageBox::critical(this, "Ошибка", "Имя должно содержать от 2 до 20 символов");
56         return;
57     }
58
59     // Проверка имени на допустимые символы
60     if (!nameRegex.match(name).hasMatch()) {
61         QMessageBox::critical(this, "Ошибка", "Имя может содержать только латинские и русские символы");
62         return;
63     }
64
65     // Проверка почты
66     QString email = ui->Mail_line->text();
67     if (email.isEmpty()) {
68         QMessageBox::critical(this, "Ошибка", "Введите адрес электронной почты");
69         return;
70 }
```

Проверка на допустимые символы. Проверок в приложении достаточно много.

КАК РАБОТАЕТ ПР-Е?

SUDO -RM -RF

Интерфейс у программы достаточно прост и лаконичен. Если Вы впервые открыли мою программу, то сначала нужно зарегистрироваться, после чего войти. В моём приложении вы сможете загрузить любые файлы и не переживать, что с ними что-то случится. Можно не только загружать файлы в облако, но так же скачивать их и удалять из облака.



Вопросы

**СПАСИБО ЗА
ВНИМАНИЕ !**