

Отчёт по проделанной работе на семинаре по мобильной разработке (18.05.2022).

(выполнил Валяев Георгий Анатольевич)

React-native – один из интересных и богатых фреймворков, наследующих большинство свойств фреймворка React (в рамках разработки нативных веб-приложений), разработанный командой Facebook для создания и написания приложений под мобильные устройства с поддержкой ОС Android, iOS и других систем.

В этой работе мы получим навыки работы с данными в плане их передачи с одного экрана приложения на другой в процессе взаимодействия с самим пользователем.

Полноценный код всего проекта расположен в моём репозитории GitHub:
... (ссылка на репозиторий).

Работа, на которой будет основана данная разработка передачи данных, расположена также на GitHub, но в другом репозитории:

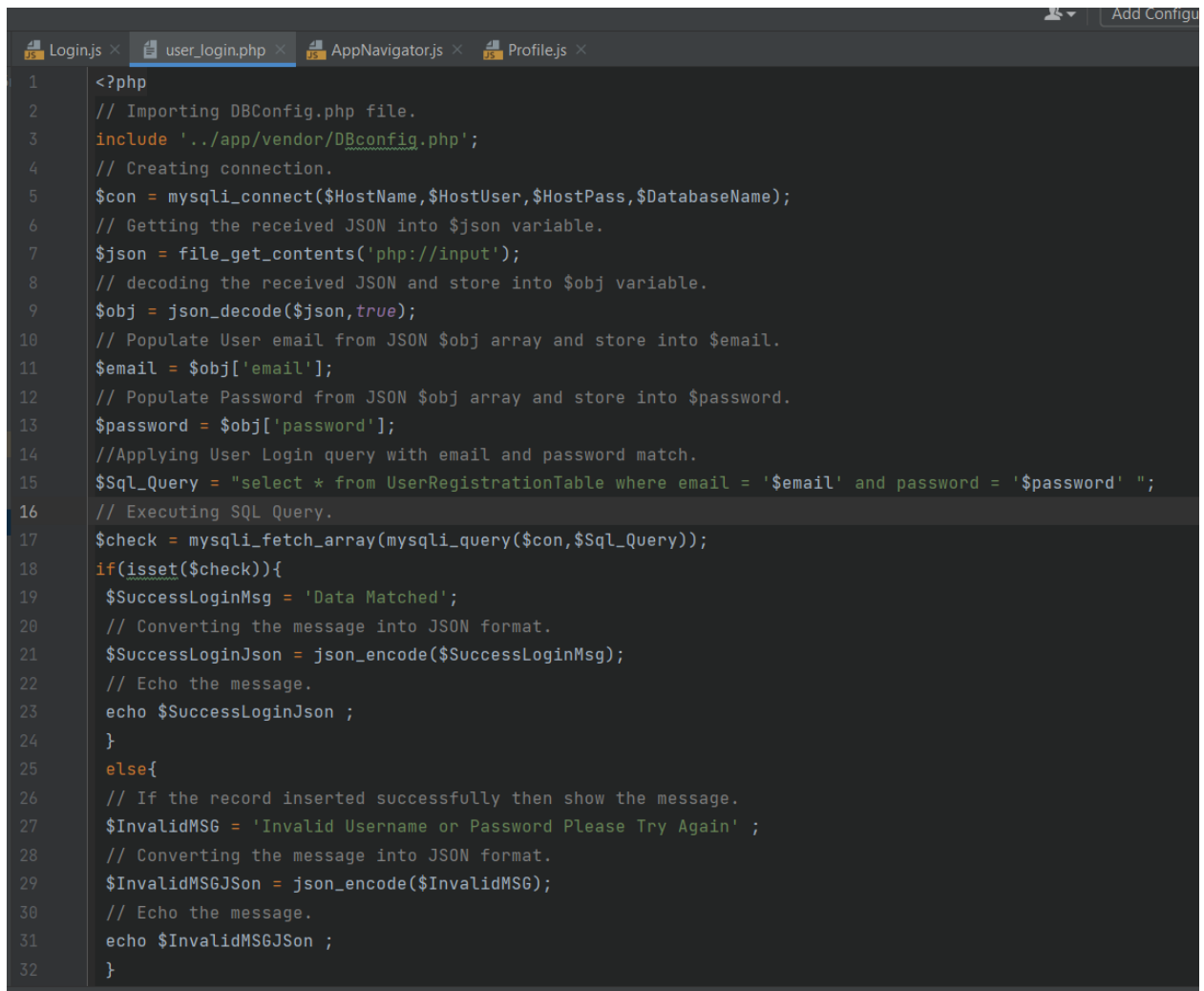
https://github.com/Valyaevgeorgiy/Android_projects/tree/main/React-Native/Forms_reg.

Отчёт этой работы находится здесь:

https://github.com/Valyaevgeorgiy/Android_projects/blob/main/Methodics_reports/%D0%A1%D0%B5%D0%BC_%D0%BF%D0%BE_%D0%BC%D0%BE%D0%B1%D0%B8%D0%BB%D0%BA%D0%B5__RN_reg_form.pdf.

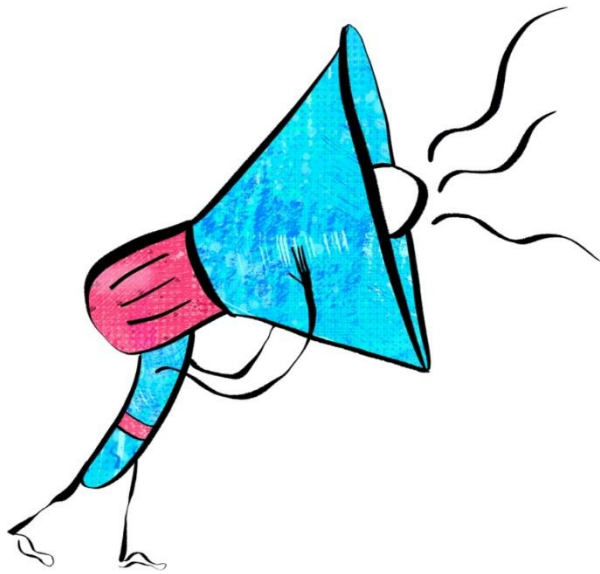
В данном отчёте отражено описание работы функционала для регистрации и Docker контейнеров. Сейчас в контейнере nginx был создан PHP-файл (), в

котором прописан весь функционал автоматизации по рекомендации <https://reactnativecode.com/react-native-user-login-using-php-mysql/>:

A screenshot of a code editor with a dark theme. The editor has four tabs at the top: 'Login.js', 'user_login.php' (which is the active tab), 'AppNavigator.js', and 'Profile.js'. The code in the active tab is a PHP script for user login. It starts with a PHP opening tag, imports a database configuration file, creates a MySQL connection, gets the received JSON data from a file, decodes it, and extracts email and password. It then constructs an SQL query to check the user's credentials in a table named 'UserRegistrationTable'. If the credentials match, it returns a success message; otherwise, it returns an invalid username or password message. The code is numbered from 1 to 32.

```
1 <?php
2 // Importing DBConfig.php file.
3 include '../app/vendor/DBconfig.php';
4 // Creating connection.
5 $con = mysqli_connect($HostName,$HostUser,$HostPass,$DatabaseName);
6 // Getting the received JSON into $json variable.
7 $json = file_get_contents('php://input');
8 // decoding the received JSON and store into $obj variable.
9 $obj = json_decode($json,true);
10 // Populate User email from JSON $obj array and store into $email.
11 $email = $obj['email'];
12 // Populate Password from JSON $obj array and store into $password.
13 $password = $obj['password'];
14 //Applying User Login query with email and password match.
15 $Sql_Query = "select * from UserRegistrationTable where email = '$email' and password = '$password' ";
16 // Executing SQL Query.
17 $check = mysqli_fetch_array(mysqli_query($con,$Sql_Query));
18 if(isset($check)){
19     $SuccessLoginMsg = 'Data Matched';
20     // Converting the message into JSON format.
21     $SuccessLoginJson = json_encode($SuccessLoginMsg);
22     // Echo the message.
23     echo $SuccessLoginJson ;
24 }
25 else{
26     // If the record inserted successfully then show the message.
27     $InvalidMSG = 'Invalid Username or Password Please Try Again' ;
28     // Converting the message into JSON format.
29     $InvalidMSGJson = json_encode($InvalidMSG);
30     // Echo the message.
31     echo $InvalidMSGJson ;
32 }
```

Получившийся результат работы приложения выглядит следующим образом:



Вход

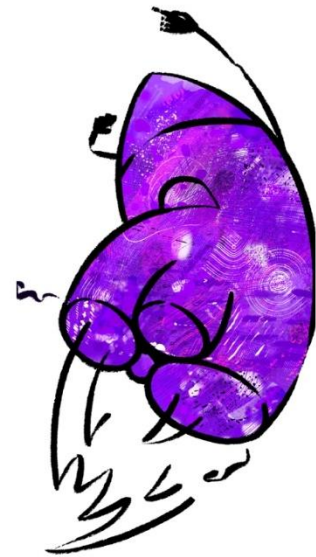
Введите логин и пароль, который создавали ранее при регистрации.

 Логин

 Пароль


Войти


Нет аккаунта? Зарегистрироваться




Регистрация

Введите свои персональные данные для повышения персонализации. Это необходимо только один раз.

 Newuser

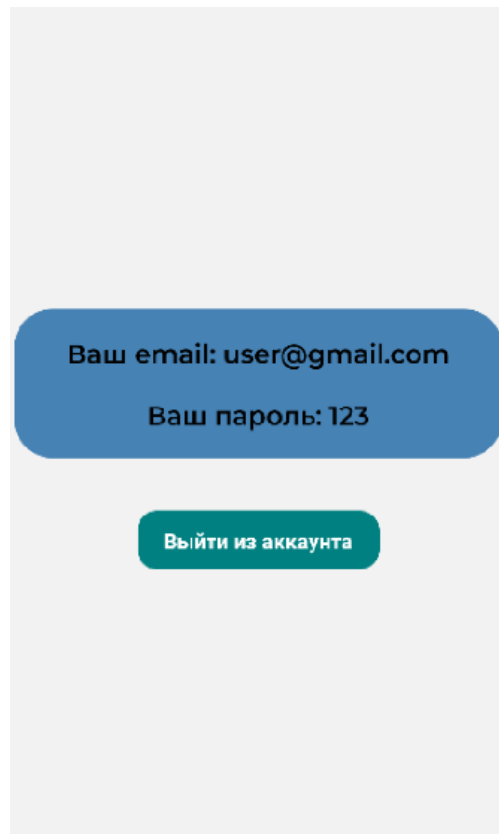
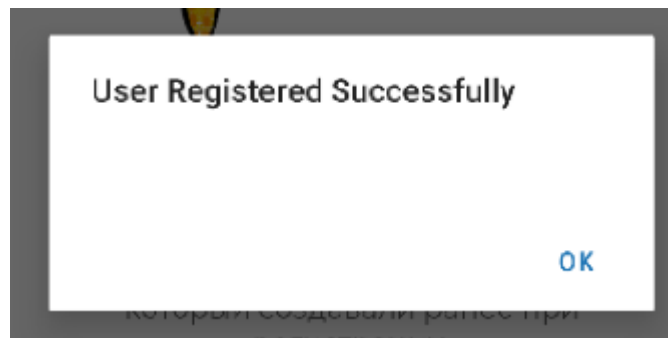
 user@gmail.com

 ...

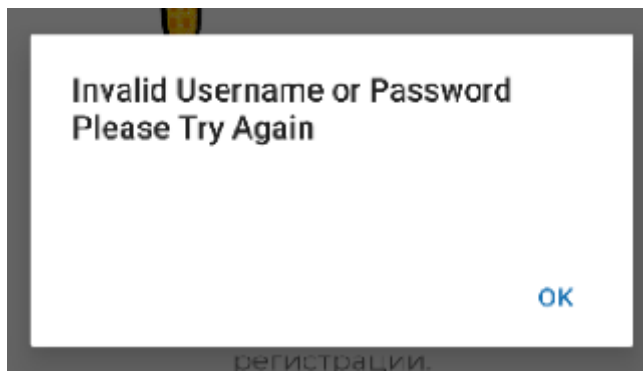
Зарегистрироваться

Уже есть регистрация? Войти

Далее рассмотрим дополнительно разработанную страницу регистрации в целях создания нового пользователя как записи в нашу базу данных.



При процессе регистрации нового пользователя в базе данных обновляется и происходит переход на страницу авторизации, где поля ввода Email и Password автоматически заполняются нужными данными с экрана авторизации, переход и передача данных с экрана регистрации происходят только при положительном ответе сервера.



При таком вводе в поля Email и Password, который по значениям не совпадает с записью в базе данных, выводится уведомление о том, что введенные значения неверные (invalid).

А если ввести корректные данные, то произойдет переход на экран Profile, где выводится информация о пользователе, используя передачу данных с экрана авторизации.