

### 3 РЕЗУЛЬТАТЫ РАЗРАБОТКИ И ИСПЫТАНИЙ СИСТЕМЫ

#### 3.1 Выбор средств для разработки

Компьютер: HP Pavilion Gaming, Intel Core i5-8300H CPU 2.30GHz , 16Gb, x64.

Операционная система: Windows 10 Pro, версия 22H2.

Среда разработки: Microsoft Visual Studio Code.

Задание выполнялось на языке программирования PHP, HTML, CSS.

Для выполнения задачи потребовалось подключение библиотек:

- 1) JQuery.min.js – библиотека позволяет изменять содержимое HTML документов путем манипулирования объектами модели, создаваемой браузерами в процессе обработки HTML-кода;
- 2) Bootstrap.min.js – бесплатный HTML/CSS/JavaScript фреймворк для быстрого создания адаптивного дизайна сайта;
- 3) JQuery.dataTables.min.js – мощный плагин jQuery для создания списков таблиц и добавления к ним взаимодействий. Он обеспечивает поиск, сортировку и разбиение на страницы без какой-либо конфигурации;
- 4) DataTables.bootstrap4.min.js – это плагин для библиотеки jQuery Javascript. Он добавляет расширенные функции к любой таблице HTML;
- 5) Dompdf – библиотека с помощью которой можно создавать PDF файлы из HTML кода.

В разработке применялся WampServer – сборка веб-сервера, содержащая Apache, MySQL, интерпретатор скриптов PHP, phpMyAdmin и другие дополнения, предназначенная для web-разработки под Windows.

Для создания база данных использовался phpMyAdmin — веб-приложение с открытым кодом, написанное на языке PHP и представляющее собой веб-интерфейс для администрирования СУБД MySQL.

#### 3.2 Реализация системы

В постановке задачи были обозначены требования к системе, следуя этим задачам обозначим основные алгоритмы используемые в средстве автоматизации «Система учета затрат на жилищно-коммунальное обслуживание населения»:

- 1) Алгоритм подключения к базе данных;
- 2) Алгоритм санкционированного доступа;
- 3) Алгоритм регистрации;
- 4) Алгоритм авторизации;

- 5) Алгоритм выборки из базы данных;
- 6) Алгоритм добавления в базу данных;
- 7) Алгоритм изменения в базе данных;
- 8) Алгоритм удаления из базы данных;
- 9) Алгоритм деактивации/активации пользователей;
- 10) Алгоритм формирования pdf файла;
- 11) Алгоритм изменения пароля.

### **Алгоритм подключения к базе данных**

Описание алгоритма: выполнение алгоритма, в котором происходит открытие подключения к локальной базе данных MySQL находится в database.php. Если соединение не было установлено выводится советующая ошибка.

```
public function __construct(){
    if (!isset($this->pdo)) {
        try { $link = new PDO('mysql:host='.DB_HOST.'; dbname='.DB_NAME,
DB_USER, DB_PASS);
            $link->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
            $link->exec("SET CHARACTER SET utf8");
            $this->pdo = $link;
        } catch (PDOException $e) {
            die("Connection error...".$e->getMessage());
        } } }
```

Данные или настройки для входа и подключения находятся в config.php.

```
define('DB_HOST', 'localhost'); define('DB_NAME', 'db4');
define('DB_USER', 'root'); define('DB_PASS', '');
```

### **Алгоритм санкционированного доступа:**

Описание алгоритма: выполнение алгоритма подразумевает собой в таблице базы данных «tbl\_users» отобразить возможные роли доступа. В нашем случае в таблице «tbl\_role» были выделены 3 роли: админ, пользователь, новый клиент. В процессе работы программы мы получаем id авторизованного пользователя в данный момент времени и условием проверяем принадлежность к роли доступа. При определении прав доступа выводится соответствующая информация, согласно его деятельности.

```
<?php if (Session::get('id') == TRUE) {
    if (Session::get('roleid') == '1') { } } ?>
```

### **Алгоритм регистрации:**

Описание алгоритма: выполнение алгоритма осуществляется, когда пользователь переходит на соответствующую форму register.php. В данной форме находятся поля для ввода, ввода показания и нажав на кнопку, методом post, данные

отправляются в функцию userRegistrarion. Данная функция принимает значения из формы и проверяет корректность введенных данных.

```
$name = $data['name'];
$username = $data['username'];
$email = $data['email'];
$mobile = $data['mobile'];
$roleid = $data['roleid'];
$password = $data['password'];
```

Частично представим проверку и отображения сообщений, например ниже представлена проверка на наличие полученных пустых полей и проверку минимального количества введенных символов в username и др..

```
if ($name == "" || $username == "" || $email == "" || $mobile == "" ||
$password == "") {
    $msg = '<div class="alert alert-danger alert-dismissible mt-3"
id="flash-msg">
    <strong>Error !</strong> Please, User Registration field must not be
Empty !</div>'; return $msg;
} elseif (strlen($username) < 3) {
    $msg = '<div class="alert alert-danger alert-dismissible mt-3"
id="flash-msg">
    <strong>Error !</strong> Username is too short, at least 3 Characters
!</div>'; return $msg; }
```

Если все возможные условия учтены и выполнены, устанавливается соединения с базой данной и переданные значения добавляются в соответствующую базу данных.

```
$sql = "INSERT INTO tbl_users(name, username, email, password, mobile,
roleid) VALUES(:name, :username, :email, :password, :mobile, :roleid)";
$stmt = $this->db->pdo->prepare($sql);
$stmt->bindValue(':name', $name);
$stmt->bindValue(':username', $username);
$stmt->bindValue(':email', $email);
$stmt->bindValue(':password', SHA1($password));
$stmt->bindValue(':mobile', $mobile);
$stmt->bindValue(':roleid', $roleid);
$result = $stmt->execute();
```

Если данные запрос на добавление в базу данных выполненлся выводится соответствующее сообщение.

### **Алгоритм авторизации:**

Описание алгоритма: выполнение алгоритма осуществляется, когда пользователь переходит на соответствующую форму login.php. В данной форме находятся поля для ввода логина и пароля, введя показания и нажав на кнопку, методом

post, данные отправляются в функцию userLoginAuthorication. Данная функция принимает значения из формы и проверяет корректность введенных данных.

```
$email = $data['email'];  
$password = $data['password'];
```

Проводим проверку корректности введенных данных по аналогии, как было выше, и отображения сообщений, например проверка на наличие полученных пустых полей и проверку минимального количества введенных символов и т.д.

Если все возможные условия учтены и выполнены, устанавливается соединения с базой данной и переданные значения сравниваются с базой данных, а также проверяется активен ли пользователь, выводится соответствующее сообщение.

```
$logResult = $this->userLoginAutho($email, $password);  
$chkActive = $this->CheckActiveUser($email);  
if ($chkActive == TRUE) {  
    $msg = '<div class=" alert-dismissible mt-3" id="flash-msg">  
        <strong>Error !</strong> Sorry, Your account is Diactivated,  
        Contact with Admin !</div>'; return $msg;
```

Если все условия выполнены, мы передаем программе все результаты и информацию о зарегистрированном пользователе, в последствии эти данные пригодятся.

```
elseif ($logResult) {  
    Session::init();  
    Session::set('login', TRUE);  
    Session::set('id', $logResult->id);  
    Session::set('roleid', $logResult->roleid);  
    Session::set('name', $logResult->name);  
    Session::set('email', $logResult->email);  
    Session::set('username', $logResult->username);  
    Session::set('logMsg', '<div class="mt-3" id="flash-msg">  
<strong>Success !</strong> You are Logged In Successfully !</div>');  
    echo "<script>location.href='index.php';</script>";
```

В итоге мы получаем авторизованного пользователя, который согласуется с санкционированным пользователем, в процессе пользования программы все данные будут идентифицироваться и проверяться для того, чтобы все пользователи системы имели собственные права на доступ в систему и взаимодействие с ней.

### **Алгоритм выборки из базы данных:**

Описание алгоритма: выполнение алгоритма осуществляется, когда пользователь переходит на соответствующую форму. По аналогии работают соответствующие формы index.php, address.php, client.php, contract.php, details.php, indication.php, notification.php, post.php, service.php, unit.php, worker.php. В данных формах находится таблица для вывода информации из базы данных.

```

<tr>
    <th class="text-center">SL</th>
    <th class="text-center">Name</th>
    <th class="text-center">Username</th>
    <th class="text-center">Email address</th>
    <th class="text-center">Mobile</th>
    <th class="text-center">Status</th>
    <th class="text-center">Created</th>
    <th width='25%' class="text-center">Action</th>
</tr>

```

Для заполнения таблицы вызываем соответствующую функцию. Согласно требующейся задаче и форме вызываются следующие функции по аналогии: selectAllUserData(), selectAllAddress(), selectAllClient(), selectAllContract(), selectAllDetails(), selectAllIndication(), selectAllNotification(), selectAllPost(), selectAllService(), selectAllUnit(), selectAllWorker(). В функции создается соединение с базой данных и производится выборка из БД с помощью запроса SELECT.

```

$sql = "SELECT * FROM tbl_users ORDER BY id DESC";
$stmt = $this->db->pdo->prepare($sql);
$stmt->execute();
return $stmt->fetchAll(PDO::FETCH_OBJ);

```

Соответственно полученный результат выводится в форму.

```

<?php if ($allUser) {
    $i = 0;
    foreach ($allUser as $value) {
        $i++;    } ?>
<td><?php echo $i; ?></td>
    <td><?php echo $value->name; ?></td>
    <td><?php echo $value->username; ?> <br>
    <?php if ($value->roleid == '1') {
        echo "<span class='info text-white'>Admin</span>";
    } else if ($value->roleid == '2') {
        echo "<span class='badge-dark text-white'>Editor</span>";
    } else if ($value->roleid == '3') {
        echo "<span class=' badge-dark text-white'>User Only</span>";
    } ?>
</td>

```

Аналогично заполняются все остальные поля таблицы.

### Алгоритм добавления в базу данных:

Описание алгоритма: выполнение алгоритма осуществляется, когда пользователь переходит на соответствующую форму. По аналогии работают соответствующие формы userAdd.php, addressAdd.php, clientAdd.php, contractAdd.php, detailsAdd.php, indicationAdd.php, notificationAdd.php, postAdd.php, serviceAdd.php,

unitAdd.php, workerAdd.php. В данных формах находятся поля для ввода информации в базу данных.

```
<form class="" action="" method="post">
  <div class="form-group pt-3">
    <label for="name">Your name</label>
    <input type="text" name="name" class="form-control">
  </div>
  <div class="form-group">
    <label for="username">Your username</label>
    <input type="text" name="username" class="form-control">
  </div>
  <div class="form-group">
    <label for="email">Email address</label>
    <input type="email" name="email" class="form-control">
  </div>
</form>
```

Для нажатия на соответствующую кнопку вызывается соответствующую функцию. Согласно требующейся задаче и форме вызываются следующие функции по аналогии: addNewUserByAdmin(), addNewAddressByAdmin(), addNewClientByAdmin(), addNewContractByAdmin(), addNewDetailsByAdmin(), addNewIndicationByAdmin(), addNewNotificationByAdmin(), addNewPostByAdmin(), addNewServiceByAdmin(), addNewUnitByAdmin(), addNewWorkerByAdmin(). В функции создается соединение с базой данных и производится добавление в БД с помощью запроса INSERT.

```
$name = $data['name'];
$username = $data['username'];
$email = $data['email'];
$mobile = $data['mobile'];
$roleid = $data['roleid'];
$password = $data['password'];
```

Проводим проверку корректности введенных данных по аналогии, как было выше, и отображения сообщений, например проверка на наличие полученных пустых полей и проверку минимального количества введенных символов и т.д.

Если все возможные условия учтены и выполнены, устанавливается соединения с базой данной и переданные значения добавляются в соответствующую базу данных.

```
$sql = "INSERT INTO tbl_users(name, username, email, password, mobile,
roleid) VALUES(:name, :username, :email, :password, :mobile, :roleid)";
$stmt = $this->db->pdo->prepare($sql);
$stmt->bindValue(':name', $name);
$stmt->bindValue(':username', $username);
$stmt->bindValue(':email', $email);
$stmt->bindValue(':password', SHA1($password));
$stmt->bindValue(':mobile', $mobile);
$stmt->bindValue(':roleid', $roleid);
$result = $stmt->execute();
```

Если данные запрос на добавление в базу данных выполненлся выводится соответствующее сообщение.

### **Алгоритм изменения в базе данных:**

Описание алгоритма: выполнение алгоритма осуществляется, когда пользователь переходит на соответствующую форму. По аналогии работают соответствующие формы profile.php, addressVER.php, clientVER.php, contractVER.php, detailsVER.php, indicationVER.php, notificationVER.php, postVER.php, serviceVER.php, unitVER.php, workerVER.php. При переходе на эту страницу срабатывает функция, которая определяет конкретную запись для изменения. По аналогии работают следующие функции: getUserInfoById(\$userid), getIndicationInfoById(\$userid), getAddressInfoById(\$userid), getClientInfoById(\$userid), getContractInfoById(\$userid), getDetailsInfoById(\$userid), getNotificationInfoById(\$userid), getPostInfoById(\$userid), getServiceInfoById(\$userid), getUnitInfoById(\$userid), getWorkerInfoById(\$userid). В данных формах находятся поля для ввода информации в базу данных.

```
<?php
    $getUinfo = $users->getUserInfoById($userid);
    if ($getUinfo) {    ?>
        <form class="" action="" method="POST">
            <div class="form-group">
                <label for="name">Your name</label>
                <input type="text" name="name" value="<?php echo
$getUserInfo->name; ?>" class="form-control"> </div>
            <div class="form-group">
                <label for="username">Your username</label>
                <input type="text" name="username" value="<?php echo
$getUserInfo->username; ?>" class="form-control"> </div>
```

Для нажатия на соответствующую кнопку вызывается соответствующую функцию.

```
if ($_SERVER['REQUEST_METHOD'] == 'POST' && isset($_POST['update'])) {
    $updateUser = $users->updateUserByIdInfo($userid, $_POST);
    if (isset($updateUser)) { echo $updateUser; }
```

Согласно требующейся задаче и форме вызываются следующие функции по аналогии: updateUserByIdInfo(\$userid, \$data), updateIndicationByIdInfo(\$userid, \$data), updateAddressByIdInfo(\$userid, \$data), updateClientByIdInfo(\$userid, \$data), updateContractByIdInfo(\$userid, \$data), updateDetailsByIdInfo(\$userid, \$data), updateNotificationByIdInfo(\$userid, \$data), updatePostByIdInfo(\$userid, \$data), updateServiceByIdInfo(\$userid, \$data), updateUnitByIdInfo(\$userid, \$data), updateWorkerByIdInfo(\$userid, \$data). В функции создается соединение с базой данных и производится добавление в БД с помощью запроса INSERT.

```

$name = $data['name'];
$username = $data['username'];
$email = $data['email'];
$mobile = $data['mobile'];
$roleid = $data['roleid'];

```

Проводим проверку корректности введенных данных по аналогии, как было выше, и отображения сообщений, например проверка на наличие полученных пустых полей и проверку минимального количества введенных символов и т.д.

Если все возможные условия учтены и выполнены, устанавливается соединения с базой данной и переданные значения обновляются в соответствующую базу данных.

```

$sql = "UPDATE tbl_users SET name = :name, username = :username, email = :email, mobile = :mobile, roleid = :roleid WHERE id = :id";
$stmt = $this->db->pdo->prepare($sql);
$stmt->bindValue(':name', $name);
$stmt->bindValue(':username', $username);
$stmt->bindValue(':email', $email);
$stmt->bindValue(':mobile', $mobile);
$stmt->bindValue(':roleid', $roleid);
$stmt->bindValue(':id', $userid);
$result = $stmt->execute();

```

Если данные запрос на изменение в базе данных выполнен, выводится соответствующее сообщение.

### Алгоритм удаления в базу данных:

Описание алгоритма: выполнение алгоритма осуществляется, когда пользователь нажимает на кнопку на соответствующей форме.

```

<a onclick="return confirm('Are you sure To Delete ?')" class="danger
<?php if (Session::get("id") == $value->id) { echo "disabled"; } ?>
    btn-sm " href="?remove=<?php echo $value->id;?>">Remove</a>
<?php if (isset($_GET['remove'])) {
    $remove = preg_replace('/^[a-zA-Z0-9-]/', '', (int)$_GET['remove']);
    $removeUser = $users->deleteUserById($remove);} ?>

```

По аналогии работают соответствующие формы deleteUserById.php, deleteAddressById.php, deleteClientById.php, deleteContractById.php, deleteDetailsById.php, deleteIndicationById.php, deleteNotificationById.php, deletePostById.php, deleteServiceById.php, deleteUnitById.php, deleteWorkerById.php. В функции создается соединение с базой данных и производится удаление из БД соответствующая запись с помощью запроса DELETE.

```

$sql = "DELETE FROM tbl_users WHERE id = :id ";
$stmt = $this->db->pdo->prepare($sql);
$stmt->bindValue(':id', $remove);
$result = $stmt->execute();

```



Если запрос на удалении записи из базы данных выполнен, выводится соответствующее сообщение.

### **Алгоритм деактивации/активации пользователей:**

Описание алгоритма: выполнение алгоритма осуществляется, когда администратор системы нажимает на кнопку на форме.

```
<?php if (isset($_GET['deactive'])) {  
    $deactive = preg_replace('/[^a-zA-Z0-9-]/', '',  
(int)$_GET['deactive']);  
    $deactiveId = $users->userDeactiveByAdmin($deactive); }  
if (isset($_GET['active'])) {  
    $active = preg_replace('/[^a-zA-Z0-9-]/', '', (int)$_GET['active']);  
    $activeId = $users->userActiveByAdmin($active); } ?>
```

В функции создается соединение с базой данных и производится деактивация/активация аккаунта в БД с помощью запроса UPDATE.

```
$sql = "UPDATE tbl_users SET  
    isActive=:isActive  
    WHERE id = :id";  
$stmt = $this->db->pdo->prepare($sql);  
$stmt->bindValue(':isActive', 1);  
$stmt->bindValue(':id', $deactive);  
$result = $stmt->execute();
```

Если запрос на деактивацию/активацию в базе данных выполнен, выводится соответствующее сообщение.

### **Алгоритм формирования pdf файла:**

Описание алгоритма: выполнение алгоритма осуществляется, когда администратор системы нажимает на кнопку на форме. Срабатывает ссылка на конфигурационный файл библиотеки dompdf, где указывается файл, который будет генерироваться в pdf, в нашем случае это select.php. А также устанавливаются настройки преобразования.

```
<?php require 'dompdf/vendor/autoload.php'; //reference Dompdf namespace  
use Dompdf\Dompdf; // instantiate and use the dompdf class  
$dompdf = new Dompdf();  
ob_start();  
require('select.php');  
$content = ob_get_clean();  
$dompdf->loadHtml($content);  
$dompdf->setPaper('A4', 'portrait'); // Render the HTML as PDF  
$dompdf->render(); // Output the generated PDF to Browser ?>
```

В файле select.php находятся запросы к базе данных и выборка соответствующих данных, и отправка этих данных в файл index.html, в котором находится форма документа со стилистикой. Если генерация pdf файла выполнен, браузер начинает скачивание файла pdf, который в последствии администратор системы может распечатать и использовать в повседневной работе.

На двух чертеж «Схема работы приложения» мы можем наблюдать схему работы программы. На первом листе изображена схема авторизации/регистрации, а также сценарии и функции работы от потребителя. На втором листе чертежа изображено сценарий и функции работы системы от администратора (см. чертеж «Схема работы системы» 1-2 лист).

### **Алгоритм изменения пароля пользователя:**

Описание алгоритма: выполнение алгоритма осуществляется, когда пользователь или администратор нажимает на кнопку «Change password» на форме.

```
<?php
if ($_SERVER['REQUEST_METHOD'] == 'POST' && isset($_POST['changepass'])) {
    $changePass = $users->changePasswordBysingelUserId($userid, $_POST); } ?>
```

В функции создается соединение с базой данных и производится проверка введенных данных, затем вызывается функция проверки старого пароля на соответствие.

```
$old_pass = SHA1($old_pass);
$sql = "SELECT password FROM tbl_users WHERE password = :password
AND id =:id";
$stmt = $this->db->pdo->prepare($sql);
$stmt->bindValue(':password', $old_pass);
$stmt->bindValue(':id', $userid);
$stmt->execute();
```

Если функция вернула не успешный вариант проверки, выводим сообщение, если же успешно, мы обновляем данные в базе данных, при чем пароль зашифруем с помощью SHA1 – который возвращает хэш строки пароля, вычисленный по алгоритму US Secure Hash Algorithm.

```
$new_pass = SHA1($new_pass);
$sql = "UPDATE tbl_users SET
password=:password
WHERE id = :id";
$stmt = $this->db->pdo->prepare($sql);
$stmt->bindValue(':password', $new_pass);
$stmt->bindValue(':id', $userid);
$result = $stmt->execute();
```

Если запрос на изменение выполнен, выводится соответствующее сообщение.

### 3.3 Результаты испытаний

Тестирование – одна из основных процедур при разработке любого программного обеспечения. Веб-приложения имеют свою методику тестирования. Методика тестирования веб-приложений включает в себя три этапа: тестирование клиентской части приложения, тестирование серверной части приложения, тестирование БД.

Как известно, существует несколько основных браузеров, которые реализованы на различных платформах, что в следствии, каждый из них, интерфейс может отображать отлично друг от друга. Для исключения данных ситуаций, проводят тестирование клиентской части приложения в некоторых популярных браузерах, проверяя работоспособность интерфейса пользователя. Тестирование пользовательского интерфейса проводилось браузерах Chrome, Opera, Mozilla FireFox. (см. плакат «Результаты испытаний»)

На рисунке 3.1 изображена спроектированная структура базы данных.

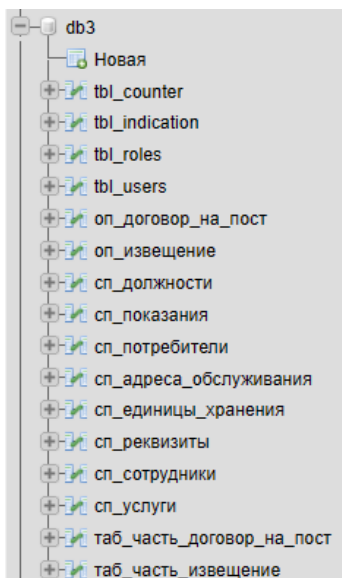


Рисунок 3.1 – Структура базы данных MySQL

#### Тест 1: «Регистрация»

Ожидаемый результат: пользователь регистрируется в системе.

Описание: тестирование правильности чтения информации из полей формы. Регистрация происходит после нажатия кнопки «Registration». Для тестирования отобразим базу данных до добавления (см рисунок 3.2). Если данные были введены корректно (см. рисунок 3.3), ожидается дополнение нового пользователя в базу данных (см. рисунок 3.4). В таблице присутствует поле «roleid», где мы можем наблюдать цифры, которые имеют значение: 1 – роль администратора, 2 – роль нового пользователя, 3 – роль пользователя. Также присутствует поле isActive в котором 0 означает активного пользователя, а 1 – неактивного, т.е. доступ к системе у него заблокирован администратором. Соответствующее сообщение он увидит при регистрации.

Изм.	Лист	№ докум.	Подп.	Дата

ДП.ПО5.190363 – 07 81 00

Лист

45

id	name	username	email	password	mobile	roleid	isActive	created_at	updated_at
23	admin@admin1	admin@admin1	admin@admin1	6c7ca345	54852852	1	0	2020-12-19 17:35:56	2020-12-19 17:35:56
24	ahmed	benahmed	achme@gmail.com	7f0c9d56	54277528	3	0	2020-12-19 18:13:39	2020-12-19 18:13:39
30	root@user1	root@user1	root@user1	eb794434	6442254	3	0	2023-04-15 22:37:38	2023-04-15 22:37:38
31	Editor	editor	Editor@editor	b9b85b9c	123456789	3	0	2023-05-04 18:37:42	2023-05-04 18:37:42
34	root@root1	root@root1	root@root1	8f8cc717	123456789	3	0	2023-05-10 13:52:24	2023-05-10 13:52:24
37	Виктор Викторович Крощук		vtas20011@inbox.ru	8ed2c560	+375333256425	3	0	2023-06-01 15:10:03	2023-06-01 15:10:03

Рисунок 3.2 – Данные таблицы пользователей до регистрации

СИСТЕМА УЧЕТА ЖКХ

Register Login

User Registration

Your name

Иван Степанович

Your username

admin@admin1

Email address

ivanov@gmail.com

Mobile Number

+375291234567

Password

\*\*\*\*\*

Register

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023 Разработчик Крощук В.В. студент БрГТУ, ФЗИС, гр.ПО-5

Рисунок 3.3 – Данные вводимые с окна регистрации

id	name	username	email	password	mobile	roleid	isActive	created_at	updated_at
23	admin@admin1	admin@admin1	admin@admin1	6c7ca345f	54852852	1	0	2020-12-19 17:35:56	2020-12-19 17:35:56
24	ahmed	benahmed	achme@gmail.com	7f0c9d56d	54277528	3	0	2020-12-19 18:13:39	2020-12-19 18:13:39
30	root@user1	root@user1	root@user1	eb794434f	6442254	3	0	2023-04-15 22:37:38	2023-04-15 22:37:38
31	Editor	editor	Editor@editor	b9b85b9de	123456789	3	0	2023-05-04 18:37:42	2023-05-04 18:37:42
34	root@root1	root@root1	root@root1	8f8cc717a	123456789	3	0	2023-05-10 13:52:24	2023-05-10 13:52:24
37	Виктор Викторович Крощук		vtas20011@inbox.ru	8ed2c560e	+375333256425	3	0	2023-06-01 15:10:03	2023-06-01 15:10:03
38	Иван Степанович	admin@admin1	ivanov@gmail.com	00c612dd4	+375291234567	3	0	2023-06-01 15:31:55	2023-06-01 15:31:55

Рисунок 3.4 – Данные таблицы пользователей после регистрации

**Вывод:** регистрация работает корректно, при корректном вводе информации были добавлены в базу данных новый пользователь. Ожидаемый результат совпал с полученным.

## Тест 2: «Авторизация» и «Санкционированный доступ»

**Ожидаемый результат:** пользователь авторизуется в системе и получает доступ согласно его правам доступа.

**Описание:** тестирование правильности чтение информации из полей формы авторизации. Авторизация происходит после нажатия кнопки «Login». Для тестирования отобразим окно авторизации администратора, пользователя и нового пользователя с введенными данными (см рисунок 3.5). Если данные были введены корректно ожидается вход в систему и вывод информации согласно правам доступа (см. рисунок 3.6). Если данные были введены неверно выводиться сообщение (см. рисунок 3.7)

СИСТЕМА УЧЕТА ЖКХ

User login

Email address

admin@admin1

Password

\*\*\*\*\*

Login

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.5 – Окна регистрации администратора, пользователя и нового пользователя

СИСТЕМА УЧЕТА ЖКХ

Success! You are Logged In Successfully!

Меню Welcome! admin@admin1

Показания

Потребители

Адреса обслуживания

Сотрудники

Должности

Услуги

Единицы хранения

Реквизиты

Извещение

Договор на поставку

User lists

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.6 – Вывод информации согласно правам доступа администратора, пользователя и нового пользователя

Error! Email did not Found, use Register email or password please!

Рисунок 3.7 – Сообщение об ошибке некорректности ввода данных

**Вывод:** авторизация и санкционированный доступ работает корректно, при корректном вводе информации были сверка введенных данных и предоставлен доступ. Ожидаемый результат совпал с полученным. В дальнейшем права доступа также будут отображены.

**Тест 3:** «Вывод информации для администратора системы из таблиц базы данных»

**Ожидаемый результат:** администратор получает доступ к информации системы согласно его правам доступа.

**Описание:** тестирование правильности чтение информации из таблиц базы данных. Переход на нужную таблицу базы данных осуществляется после нажатия соответствующей

кнопки из меню. Для тестирования отобразим все возможные таблицы и вывод информации согласно правам доступа администратор (см. рисунки 3.8-3.10).

**User list**

Welcome! **admin@admin1**

Show  entries
 Search:

Sl	Name	Username	Email address	Mobile	Status	Created	Action
1	Иван Степанович	admin@admin1 <small>Editor</small>	ivanov@gmail.com	+375291234567	Active	2023-06-01 15:31:55	View Edit Remove Disable
2	Виктор Викторович	Крошук <small>User Only</small>	vtas20011@inbox.ru	+375333256425	Active	2023-06-01 15:10:03	View Edit Remove Disable
3	root@root1	root@root1 <small>User Only</small>	root@root1	123456789	Active	2023-05-10 13:52:24	View Edit Remove Disable
4	Editor	editor <small>User Only</small>	Editor@editor	123456789	Active	2023-05-04 18:37:42	View Edit Remove Disable
5	root@user1	root@user1 <small>User Only</small>	root@user1	6442254	Active	2023-04-15 22:37:38	View Edit Remove Disable
6	ahmed	benahmed <small>User Only</small>	achme@gmail.com	54277528	Active	2020-12-19 18:13:39	View Edit Remove Disable
7	admin@admin1	admin@admin1 <small>Admin</small>	admin@admin1	54852853	Active	2020-12-19 17:25:56	View Edit Remove Disable

Showing 1 to 7 of 7 entries
 Previous 1 Next

**Извещение**

Welcome! **admin@admin1**

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
 Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5
 

**Сформировать новое извещение**

Show  entries
 Search: 

Showing 1 to 3 of 3 entries
 Previous 1 Next

**Услуги**

Welcome! **admin@admin1**

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
 Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5
 

**Добавить услугу**

Show  entries
 Search: 

Showing 1 to 5 of 5 entries
 Previous 1 Next

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
 Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5
 

Рисунок 3.8 – Вывод таблиц «Аккаунты», «Извещение», «Услуги» из базы данных

Договор на поставку услуг

Welcome! **admin@admin1**  
[Оформить новый договор](#)

Show 10 entries

Search:

ID	Номер договора	Дата составления	Потребитель	ID user	Лицевой счет	Адрес обслуживания	Организация	Оформил	Action
1	1	2023-05-22 00:00:00	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	Брест ЖКО-1	Иванов Иван Иванович	
2	3	2023-05-26 20:36:09	Ляшевич Тимофей Александрович	31	5555	Брест ул.Киевская д.1	Брест ЖКО-2	Егоров Егор Егорович	

Showing 1 to 2 of 2 entries

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Адреса обслуживания

Welcome! **admin@admin1**  
[Добавить адрес](#)

Show 10 entries

Search:

ID	Адрес обслуживания	Потребитель	ID user	Лицевой счет	Action
1	Брест ул.Березовская д.46	Крошук Виктор Викторович	37	1234	
2	Брест ул.Лирическая д.5	Крошук Виктор Викторович	37	1234	
3	Брест ул.Харитоновская д.28	Крошук Марк Викторович	34	1111	
4	Брест ул.Киевская д.1	Ляшевич Тимофей Александрович	31	5555	

Showing 1 to 4 of 4 entries

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Единицы измерения

Welcome! **admin@admin1**  
[Добавить единицу измерения](#)

Show 10 entries

Search:

ID	Единица измерения	Action
1	куб.м.	
2	кВт*ч	
3	Гкал	

Showing 1 to 3 of 3 entries

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Реквизиты организации

Welcome! **admin@admin1**  
[Добавить реквизиты](#)

Show 10 entries

Search:

ID	Организация	УНП Организации	Адрес	Action
1	Брест ЖКО-1	123456789	Брест, Пионерская 5	
2	Брест ЖКО-2	123123123	Брест ул.Хацапетовская д.55	

Showing 1 to 2 of 2 entries

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.9 – Вывод таблиц «Договор на поставку», «Адреса обслуживания», «Единицы измерения», «Реквизиты организации» из базы данных согласно правам администратора

Сотрудники

Welcome! admin@admin1

Добавить сотрудника

Show 10 entries

Search:

ID	Сотрудник	Должность	Action
1	Иванов Иван Иванович	Менеджер	<div><div>View</div><div>Edit</div><div>Remove</div></div>
2	Егоров Егор Егорович	Контролер	<div><div>View</div><div>Edit</div><div>Remove</div></div>
3	Викторов Виктор Викторович	Бухгалтер	<div><div>View</div><div>Edit</div><div>Remove</div></div>
4	Курилюк Иоанн Семенович	Работник	<div><div>View</div><div>Edit</div><div>Remove</div></div>

Showing 1 to 4 of 4 entries

Previous

1

Next

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Должности

Welcome! admin@admin1

Добавить должность

Show 10 entries

Search:

ID	Наименование должности	Action
1	Менеджер	<div><div>View</div><div>Edit</div><div>Remove</div></div>
2	Контролер	<div><div>View</div><div>Edit</div><div>Remove</div></div>
3	Бухгалтер	<div><div>View</div><div>Edit</div><div>Remove</div></div>
4	Работник	<div><div>View</div><div>Edit</div><div>Remove</div></div>

Showing 1 to 4 of 4 entries

Previous

1

Next

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Потребители

Welcome! admin@admin1

Добавить потребителя

ID	Потребитель	Лицевой счет	Адрес	Action
1	Крошук Виктор Викторович	37	1234 Брест ул.Березовская д.46	<div><div>View</div><div>Edit</div><div>Remove</div></div>
2	Крошук Виктор Викторович	37	1234 Брест ул.Лирическая д.5	<div><div>View</div><div>Edit</div><div>Remove</div></div>
3	Крошук Марк Викторович	34	1111 Брест ул.Харитоновская д.28	<div><div>View</div><div>Edit</div><div>Remove</div></div>
4	Ляшевич Тимофей Александрович	31	5555 Брест ул.Киевская д.1	<div><div>View</div><div>Edit</div><div>Remove</div></div>

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Показания

Welcome! admin@admin1

Добавить показания

Show 10 entries

Search:

ID	Потребитель	ID user	Лицевой счет	Адрес	Дата	Услуга	Ед.хран.	Тариф	Количество	Итого_сумма	Action
1	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	2023-05-22 00:00:00	Газоснабжение	куб.м.	0.2062	100	20.6200	<div><div>View</div><div>Edit</div><div>Remove</div></div>
2	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	2023-05-25 00:00:00	Холодное водоснабжение	куб.м.	1.3456	100	134.5600	<div><div>View</div><div>Edit</div><div>Remove</div></div>
3	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	2023-05-22 00:00:00	Электроснабжение	кВт*ч	0.2705	100	27.0500	<div><div>View</div><div>Edit</div><div>Remove</div></div>
4	Крошук Марк Викторович	34	1111	Брест ул.Харитоновская д.28	2023-05-26 23:30:49	Обогрев помещения	Гкал	0.0010	1000	1.0000	<div><div>View</div><div>Edit</div><div>Remove</div></div>
5	Крошук Марк Викторович	34	1111	Брест ул.Харитоновская д.28	2023-05-26 23:31:29	Канализация	куб.м.	1.1312	10	11.3120	<div><div>View</div><div>Edit</div><div>Remove</div></div>
6	Ляшевич Тимофей Александрович	31	5555	Брест ул.Киевская д.1	2023-05-27 15:03:35	Газоснабжение	куб.м.	0.2062	1000	206.2000	<div><div>View</div><div>Edit</div><div>Remove</div></div>

Рисунок 3.10 – Вывод таблиц «Сотрудники», «Должности», «Потребители», «Показания» из базы данных согласно правам доступа администратора

					ДП.ПО5.190363 – 07 81 00	Лист 50
Изм.	Лист	№ докум.	Подп.	Дата		




Вывод: отображение всей информации и санкционированный доступ для администратора работает корректно. Ожидаемый результат совпал с полученным.


**Тест 4:** «Вывод информации для пользователя системы из таблиц базы данных»

Ожидаемый результат: пользователь получает доступ к личной информации системы согласно его правам доступа.

Описание: тестирование правильности чтения информации из таблиц базы данных. Переход на нужную таблицу базы данных осуществляется после нажатия соответствующей кнопки из меню. Для тестирования отобразим все возможные таблицы и вывод информации согласно правам доступа пользователя (см. рисунки 3.11-3.12).

 Показания

Welcome! Крошук

 Добавить показания

Ваш ID пользователя: №37

ID	Потребитель	ID user	Лицевой счет	Адрес	Дата	Услуга	Ед.хран.	Тариф	Количество	Итого_сумма	Action
1	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	2023-05-22 00:00:00	Газоснабжение	куб.м.	0.2062	100	20.6200	<a href="#">View</a> <a href="#">Edit</a>
2	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	2023-05-25 00:00:00	Холодное водоснабжение	куб.м.	1.3456	100	134.5600	<a href="#">View</a> <a href="#">Edit</a>
3	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	2023-05-22 00:00:00	Электроснабжение	кВт*ч	0.2705	100	27.0500	<a href="#">View</a> <a href="#">Edit</a>
8	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	2023-04-30 16:05:22	Обогрев помещения	Гкал	0.0010	10000	10.0000	<a href="#">View</a> <a href="#">Edit</a>

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

 Адреса обслуживания

Welcome! Крошук

Ваш ID пользователя: №37

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.11 – Вывод таблиц «Показания», «Адреса обслуживания» из базы данных согласно правам доступа пользователя

Show 10 entries

Search:

Showing 1 to 5 of 5 entries

Previous 1 Next

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Номер договора	Дата составления	Потребитель	ID user	Лицевой_счет	Адрес обслуживания	Организация	Оформил	Action
1	2023-05-22 00:00:00	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	Брест ЖКХ-1	Иванов Иван Иванович	<button>Услуги</button>

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Show 10 entries

Search:

Showing 1 to 3 of 3 entries

Previous 1 Next

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Вывод: отображение всей информации и санкционированный доступ для пользователя работает корректно. Ожидаемый результат совпал с полученным.

### Тест 5: «Добавление в базу данных»

Ожидаемый результат: администратор добавит (составит) новый договор на поставку услуг, которая отобразится в базе данных.

Описание: Для тестирования возьмем администратора и от него составим новый договор. По аналогии происходят все функции добавления как от администратора, так и от пользователя в разных таблицах системы. Добавление происходит после нажатия кнопки «Добавить». Для тестирования отобразим таблицу до добавления (см рисунок 3.13). Если данные были введены корректно (см. рисунок 3.14), ожидается дополнение нового договора в базу данных, следовательно результат отобразится в таблице (см. рисунок 3.15).

Договор на поставку услуг

Welcome!

admin@admin1

+

Оформить новый договор

Show

10

entries

Search:

ID	Номер договора	Дата составления	Потребитель	ID user	Лицевой_счет	Адрес обслуживания	Организация	Оформил	Action
1	1	2023-05-22 00:00:00	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	Брест ЖКХ-1	Иванов Иван Иванович	<div>Услуги</div> <div>View</div> <div>Edit</div> <div>Remove</div>
2	3	2023-05-26 20:36:09	Ляшевич Тимофей Александрович	31	5555	Брест ул.Киевская д.1	Брест ЖКХ-2	Егоров Егор Егорович	<div>Услуги</div> <div>View</div> <div>Edit</div> <div>Remove</div>

Showing 1 to 2 of 2 entries

Previous

1

Next

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.13 – Данные таблицы «Договор на поставку» до добавления

Success ! Ура, вы зарегистрировали нового договор на поставку услуг !

### Добавить договор на поставку

Адрес обслуживания

2 - Брест ул.Лирическая д.5 - Крошук Виктор Викторович

Реквизиты организации

2 - Брест ЖКХ-2 - 123123123 - Брест ул.Хацапетовская д.55

Оформляющий сотрудник

4 - Курилюк Иоанн Семенович - Работник

Добавить

Назад

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023  
Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.14 – Данные вводимые с окна добавления нового договора на поставку



 Договор на поставку услуг					Welcome! <span>admin@admin1</span>  <a href="#">Оформить новый договор</a>				
Show <input type="text" value="10"/> entries					Search: <input type="text"/>				
ID ↑↓	Номер договора ↑↓	Дата составления ↑↓	Потребитель ↑↓	ID user ↑↓	Лицевой_счет ↑↓	Адрес обслуживания ↑↓	Организация ↑↓	Оформил ↑↓	Action ↑↓
1	1	2023-05-22 00:00:00	Крошук Виктор Викторович	37	1234	Брест ул.Березовская д.46	Брест ЖКХ-1	Иванов Иван Иванович	<div>Услуги</div> <div>View</div> <div>Edit</div> <div>Remove</div>
2	3	2023-05-26 20:36:09	Ляшевич Тимофей Александрович	31	5555	Брест ул.Киевская д.1	Брест ЖКХ-2	Егоров Егор Егорович	<div>Услуги</div> <div>View</div> <div>Edit</div> <div>Remove</div>
3	4	2023-06-01 17:34:54	Крошук Виктор Викторович	37	1234	Брест ул.Лирическая д.5	Брест ЖКХ-2	Курилюк Иоанн Семенович	<div>Услуги</div> <div>View</div> <div>Edit</div> <div>Remove</div>
Showing 1 to 3 of 3 entries								Previous <span>1</span> Next	
© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023 Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5									

Рисунок 3.15 – Данные таблицы «Договор на поставку» после добавления

Вывод: при корректном вводе информации добавление нового договора в базу данных работает корректно. Ожидаемый результат совпал с полученным.

#### Тест 6: «Изменение в базе данных»

Ожидаемый результат: администратор изменить ФИО и роль нового пользователя в системе, изменения должны отобразиться в базе данных.

Описание: Для тестирования возьмем администратора, посмотрим все данные о всех пользователях (см. рисунок 3.16), выберем нового пользователя нажав на кнопку «View» или «Edit» увидим его оригинальные данные (см. рисунок 3.17). Изменим его данные и зарегистрируем (см. рисунок 3.18). По аналогии происходят все функции изменения как от администратора, так и от пользователя в разных таблицах системы в зависимости от прав доступа. Если данные были введены корректно ожидается изменение в базе данных, следовательно результат отобразиться в таблице (см. рисунок 3.19).

User list

Welcome!
admin@admin1

Show
10
entries

Search:

SL	Name	Username	Email address	Mobile	Status	Created	Action
1	Иван Степанович	admin@admin1 Editor	ivanov@gmail.com	+375291234567	Active	2023-06-01 15:31:55	View Edit Remove Disable
2	Виктор Викторович	Крошук User Only	vas20011@inbox.ru	+375333256425	Active	2023-06-01 15:10:03	View Edit Remove Disable
3	root@root1	root@root1 User Only	root@root1	123456789	Active	2023-05-10 13:52:24	View Edit Remove Disable
4	Editor	editor User Only	Editor@editor	123456789	Active	2023-05-04 18:37:42	View Edit Remove Disable
5	root@user1	root@user1 User Only	root@user1	6442254	Active	2023-04-15 22:37:38	View Edit Remove Disable
6	ahmed	benahmed User Only	achme@gmail.com	54277528	Active	2020-12-19 18:13:39	View Edit Remove Disable
7	admin@admin1	admin@admin1 Admin	admin@admin1	54852852	Active	2020-12-19 17:35:56	View Edit Remove Disable

Showing 1 to 7 of 7 entries

Previous
1
Next

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.16 – Данные таблицы «Аккаунты пользователей» до изменения

User Profile

Back

Your name

Иван Степанович

Your username

admin@admin1

Email address

ivanov@gmail.com

Mobile Number

+375291234567

Select user Role

Editor

Update

Password change

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.17 – Оригинальные данные, выведенные в окно просмотра/изменения

User Profile

Back

Your name

Дмитрий Геннадьевич

Your username

admin@admin1

Email address

ivanov@gmail.com

Mobile Number

+375291234567

Select user Role

User only

Update

Password change

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.18 – Исправленные данные в полях

User list

Welcome! admin@admin1

Show 10 entries

Search:

SL	Name	Username	Email address	Mobile	Status	Created	Action
1	Дмитрий Геннадьевич	admin@admin1 User Only	ivanov@gmail.com	+375291234567	Active	2023-06-01 15:31:55	View Edit Remove Disable
2	Виктор Викторович	Крошук User Only	vas20011@inbox.ru	+375333256425	Active	2023-06-01 15:10:03	View Edit Remove Disable
3	root@root1	root@root1 User Only	root@root1	123456789	Active	2023-05-10 13:52:24	View Edit Remove Disable
4	Editor	editor User Only	Editor@editor	123456789	Active	2023-05-04 18:37:42	View Edit Remove Disable
5	root@user1	root@user1 User Only	root@user1	6442254	Active	2023-04-15 22:37:38	View Edit Remove Disable
6	ahmed	benahmed User Only	achme@gmail.com	54277528	Active	2020-12-19 18:13:39	View Edit Remove Disable
7	admin@admin1	admin@admin1 Admin	admin@admin1	54852852	Active	2020-12-19 17:35:56	View Edit Remove Disable

Showing 1 to 7 of 7 entries

Previous 1 Next

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.19 – Данные таблицы «Аккаунты пользователей» после изменения

Вывод: изменились данные пользователя в базе данных, а именно имя, отчество и роль прав доступа. Изменение работает корректно. Ожидаемый результат совпал с полученным.

### Тест 7: «Деактивация/активация пользователей»

Ожидаемый результат: администратор может заблокировать/разблокировать пользователя, и он не сможет зайти в систему.

Описание: для тестирования возьмем администратора, посмотрим все данные о всех пользователях (см. рисунок 3.20), выберем пользователя и нажав на кнопку «Disable», увидим сообщение предупреждение, которое подтвердим. Статус изменится в таблице с активного на неактивный (см. рисунок 3.21). Попытаемся авторизоваться со стороны пользователя, статус которого неактивный (см. рисунок 3.22). Аналогичным образом происходит активация пользователя.

User list							Welcome! admin@admin1	
Show 10 entries							Search:	
SL	Name	Username	Email address	Mobile	Status	Created	Action	
1	Дмитрий Геннадьевич	admin@admin1 User Only	ivanov@gmail.com	+375291234567	Active	2023-06-01 15:31:55	View	Edit Remove Disable

Рисунок 3.20 – Данные таблицы «Аккаунты пользователей» до блокировки пользователя

User list							Welcome! admin@admin1	
Show 10 entries							Search:	
SL	Name	Username	Email address	Mobile	Status	Created	Action	
1	Дмитрий Геннадьевич	admin@admin1 User Only	ivanov@gmail.com	+375291234567	Deactive	2023-06-01 15:31:55	View	Edit Remove Active

Рисунок 3.21 – Данные таблицы «Аккаунты пользователей» после блокировки пользователя

Error ! Sorry, Your account is Diactivated, Contact with Admin !

User login

Email address

ivanov@gmail.com

Password

.....

Login

© СРЕДСТВО АВТОМАТИЗАЦИИ "СИСТЕМА УЧЕТА ЗАТРАТ НА ЖИЛИЩНО-КОММУНАЛЬНОЕ ОБСЛУЖИВАНИЕ НАСЕЛЕНИЯ" 2023

Разработчик: Крошук В.В. студент БрГТУ, ФЭИС, гр.ПО-5

Рисунок 3.22 – Авторизация неактивного пользователя

**Вывод:** пользователь был деактивирован, войти в систему он не может, а также получил предупреждение, что его аккаунт заблокирован и требуется обратиться к администратору. Ожидаемый результат совпал с полученным.

#### Тест 8: «Удаление из базы данных»

**Ожидаемый результат:** пользователь может удалить неправильные показания использования услуг.

**Описание:** для тестирования возьмем пользователя, просмотрим все показания, введенные им ранее (см. рисунок 3.23), выберем показание для удаления и нажав на кнопку «Remove», увидим сообщение предупреждение, которое подтвердим. Из таблицы показаний удалиться запись (см. рисунок 3.24). Аналогичным образом происходит удаление в других таблицах как от пользователя, так и от администратора.

ID ↑↓	Потребитель ↑↓	ID user ↑↓	Лицевой счет ↑↓	Адрес ↑↓	Дата ↑↓	Услуга ↑↓	Ед.хран. ↑↓	Тариф ↑↓	Количество ↑↓	Итого_сумма ↑↓	Action ↑↓
6	Ляшевич Тимофей Александрович	31	5555	Брест ул.Киевская д.1	2023- 05-27 15:03:35	Газоснабжение	куб.м.	0.2062	1000	206.2000	View Edit Remove
7	Ляшевич Тимофей Александрович	31	5555	Брест ул.Киевская д.1	2023- 05-27 15:04:30	Обогрев помещения	Гкал	0.0010	2000	2.0000	View Edit Remove
9	Ляшевич Тимофей Александрович	31	5555	Брест ул.Киевская д.1	2023- 05-31 01:06:32	Газоснабжение	куб.м.	0.2062	10	20.6200	View Edit Remove

Рисунок 3.23 – Данные таблицы «Показания» до удаления

ID ↑↓	Потребитель ↑↓	ID user ↑↓	Лицевой счет ↑↓	Адрес ↑↓	Дата ↑↓	Услуга ↑↓	Ед.хран. ↑↓	Тариф ↑↓	Количество ↑↓	Итого_сумма ↑↓	Action ↑↓
7	Ляшевич Тимофей Александрович	31	5555	Брест ул.Киевская д.1	2023- 05-27 15:04:30	Обогрев помещения	Гкал	0.0010	2000	2.0000	View Edit Remove
9	Ляшевич Тимофей Александрович	31	5555	Брест ул.Киевская д.1	2023- 05-31 01:06:32	Газоснабжение	куб.м.	0.2062	10	20.6200	View Edit Remove

Рисунок 3.24 – Данные таблицы «Показания» после удаления

**Вывод:** пользователем одна запись показаний была удалена. Ожидаемый результат совпал с полученным.

#### Тест 9: «Формирование pdf фала»

**Ожидаемый результат:** администратор, нажав на кнопку, получит pdf файл извещения.

**Описание:** для тестирования возьмем администратора, нажав на кнопку «PDF file» перед соответствующей записью (см. рисунок 3.25), начнется загрузка в память компьютера pdf файла (см. рисунок 3.26). Открыв который получаем извещение для оплаты за соответствующий месяц (см. рисунок 3.27). Аналогичным образом происходит формирование pdf файла договора на поставку услуг.





 Извещение		Welcome! <span>admin@admin1</span>						
		 Сформировать новое извещение						
Show <span>10</span> entries		Search: <input type="text"/>						
ID ↑↓	Номер извещения ↑↓	Дата составления ↑↓	Потребитель ↑↓	Лицевой_счет ↑↓	Адрес обслуживания ↑↓	Организация ↑↓	Оформил ↑↓	Action ↑↓
1	1	2023-05-22 00:00:00	Крошук Виктор Викторович	1234	Брест ул.Березовская д.46	Брест ЖКХ-1	Викторов Виктор Викторович	<div>Показания</div> <div>PDF file</div> <div>Remove</div>

Рисунок 3.25 – Запись из таблицы «Извещения» для формирования pdf файла

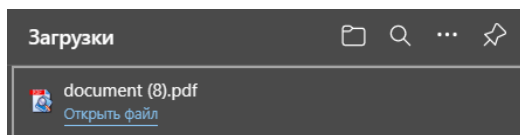


Рисунок 3.26 – Файл формата pdf в загрузках браузера

### ИЗВЕЩЕНИЕ ЖКХ №1 от 2023-05-22

ОРГАНИЗАЦИЯ	МЕСТО ОБСЛУЖИВАНИЯ	ПОТРЕБИТЕЛЬ	ИЗВЕЩЕНИЕ
Наименование: "Брест ЖКХ-1"	Адрес: Брест ул.Березовская д.46	Крошук Виктор Викторович	Номер извещения:1
Адрес: Брест, Пионерская 5		Лицевой счет: 1234	Дата и время: 2023-05-22 00:00:00
УНП: 123456789			

Код показаний	Услуга	Единица хранения	Тариф	Количество	Цена
1	Газоснабжение	куб.м.	0.2062	100	20.6200 BYN
2	Холодное водоснабжение	куб.м.	1.3456	100	134.5600 BYN
3	Электроснабжение	кВт*ч	0.2705	100	27.0500 BYN

**Итого к оплате : BYN: 182.2300**

**ИЗВЕЩЕНИЕ СФОРМИРОВАННО АВТОМАТИЧЕСКИ**

Подпись: .....

Сотрудник: Викторов Виктор Викторович

Должность: Бухгалтер

Дата и время: 2023-05-22 00:00:00

Рисунок 3.27 – Сформированный pdf файл

Вывод: администратором был сформирован pdf файл извещения по соответствующей записи, месяце показаний, пользователе, адресе обслуживания, реквизитам организации. Ожидаемый результат совпал с полученным.

					ДП.ПО5.190363 – 07 81 00	Лист
Изм.	Лист	№ докум.	Подп.	Дата		59

### Тест 10: «Изменение пароля пользователем»

Ожидаемый результат: пользователь, введя свой старый пароль и новый пароль, изменит его в базе данных.

Описание: для тестирования возьмем пользователя, нажав на кнопку «Password change» (см. рисунок 3.28), перейдет в окно, где можно ввести старый пароль и новый пароль, на который изменяется старый (см. рисунок 3.29). После нажатия «Change password», происходит проверка старого пароля, если пароль верен, заменяется на новый, при чем зашифрованный в базе данных (см. рисунок 3.30 – 3.31). Получение сообщение о изменении (см. рисунок 3.32).

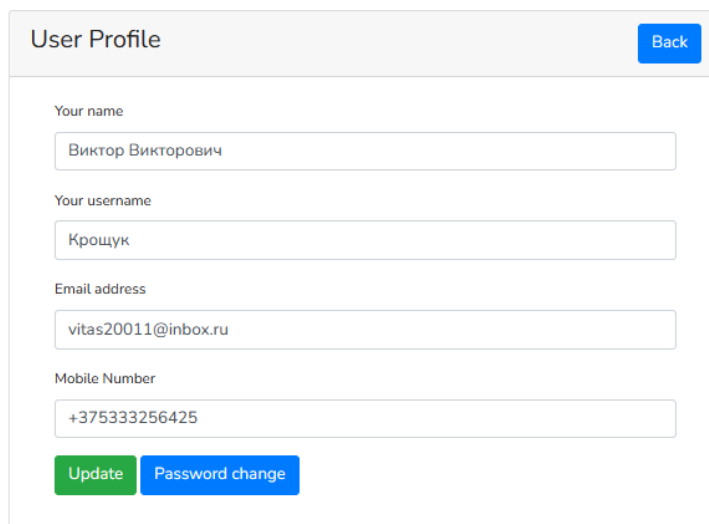


Рисунок 3.28 – Окно изменений в профиле

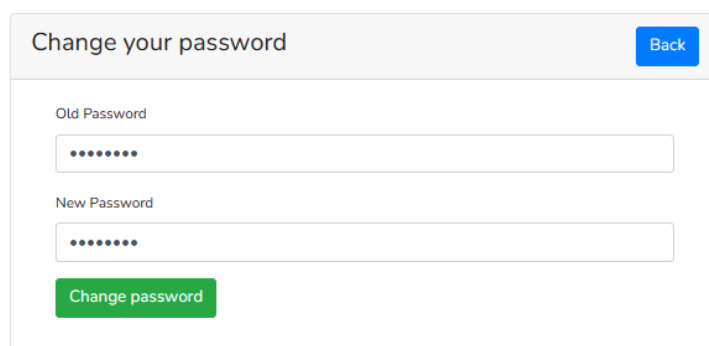


Рисунок 3.29 – Окно ввода старого и нового пароля

37	Виктор Викторович	Крошук	vitas20011@inbox.ru	8ed2c5608218c228b6da35153f6c4275499f2afa	+375333256425	3	0	2023-06-01 15:10:03	2023-06-01 15:10:03
----	-------------------	--------	---------------------	--	---------------	---	---	---------------------	---------------------

Рисунок 3.30 – База данных до изменения пароля

37	Виктор Викторович	Крошук	vitas20011@inbox.ru	fb80eeecf61ec8d9ac6d8a61927a40a61ffbfd46	+375333256425	3	0	2023-06-01 15:10:03	2023-06-01 15:10:03
----	-------------------	--------	---------------------	--	---------------	---	---	---------------------	---------------------

Рисунок 3.31 – База данных после изменения пароля

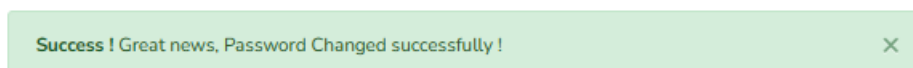


Рисунок 3.32 – Сообщение о успешном изменении пароля

Вывод: пользователем был изменен пароль, который в базе данных зашифрован. Ожидаемый результат совпал с полученным.