Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

**Отчет по лабораторной работе №8**

Дисциплина: «Разработка профессиональных приложений»

Вариант №19

Выполнил:

студент группы ИВТАПбд-21

Романов Н.Ю

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

Задание по варианту

Необходимо модифицировать лабораторную работу №7:

1. Добавить средства редактирования для ВСЕХ сущностей.

2. Добавить минимум 2 роли, соответствующих реальным ролям в выбранной предметной области. Добавить пользователей, обладающих этими ролями.

3. Реализовать средства регистрации пользователей, аутенфикации.

4. Добавить средства авторизации и разграничения прав доступа разным ролям к разным сущностям.

Описание реализации

В рамках данной работы были созданы две роли пользователей: "Сотрудник" и "Обычный пользователь".

"Сотрудник" имеет полный доступ к операциям создания, чтения, обновления и удаления записей во всех таблицах базы данных. Это значит, что он может добавлять новые магазины в систему, изменять информацию о уже существующих магазинах, сотрудниках, чеках и удалять ненужные записи.

"Обычный пользователь", с другой стороны, имеет только ограниченный доступ к информации. Он может просмотреть некоторые таблицы, которые содержат информацию о магазинах, такие как адреса, товары. Обычный пользователь не может выполнять действия в системе, такие как изменение или удаление записей.

Для аутентификации пользователей и их регистрации в системе использовались стандартные Django формы, которые были адаптированы под дизайн сайта. Таким образом, пользователи могут зарегистрироваться в системе, чтобы получить доступ к соответствующей информации в зависимости от их роли и прав доступа.

**Описание возникших затруднений**

Затруднения возникли с регистрацией, поскольку для неё пришлось сделать отдельную форму и отображение. А также было потрачено немного времени на то чтобы узнать как определить роль пользователя после авторизации.

Описание альтернативных способов решения

Можно было не использовать готовые формы Django, а написать всё полностью самому, т.е. сделать модель пользователя и сделать миграции, после чего уже отдельно работать с формами. А также можно было вместо проверки роли проверять группу, в которой находится пользователь.

Тестирование



Рис. 1 Страница от лица обычного пользователя

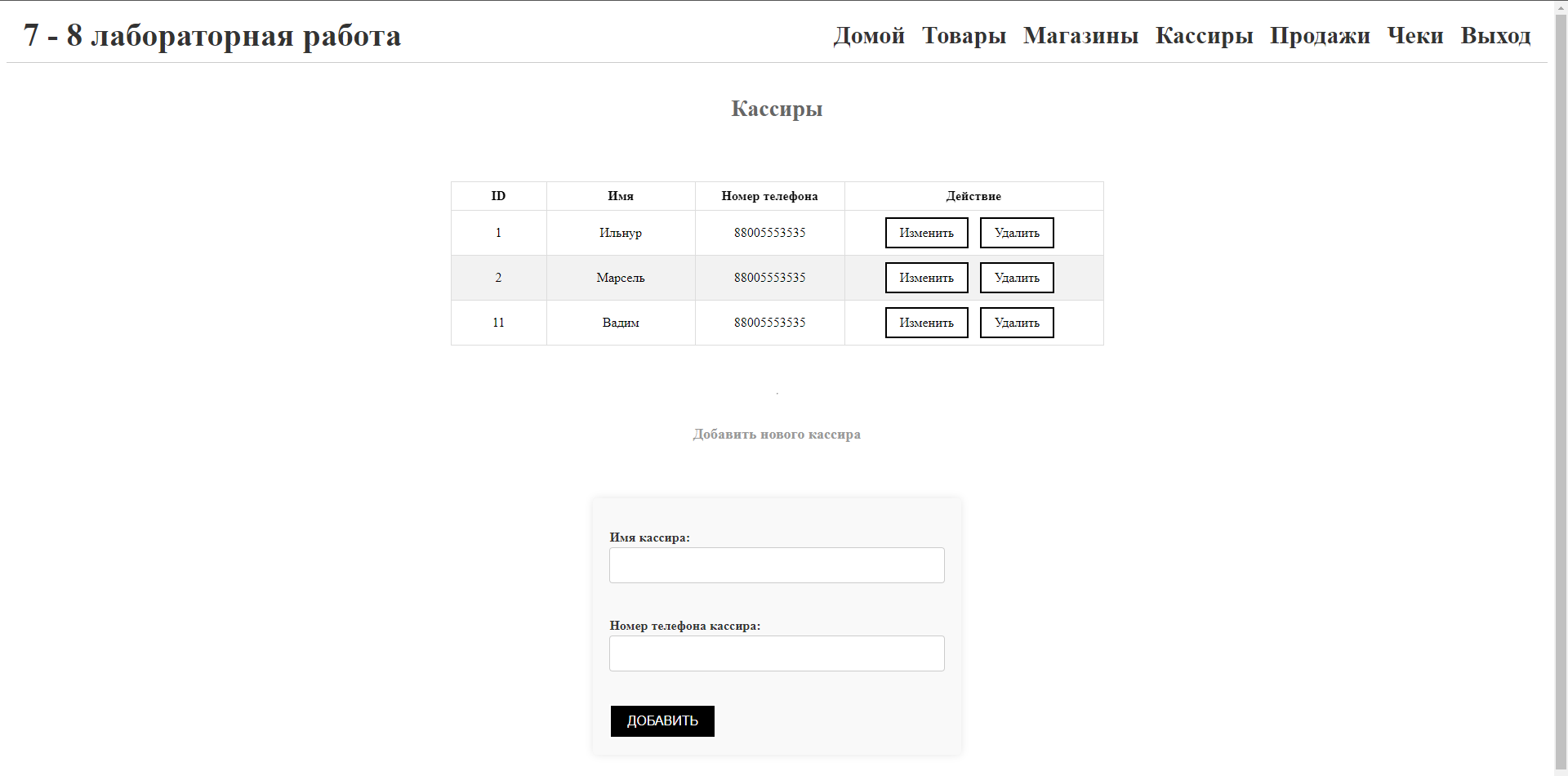


Рис. 2 Страница от лица сотрудника

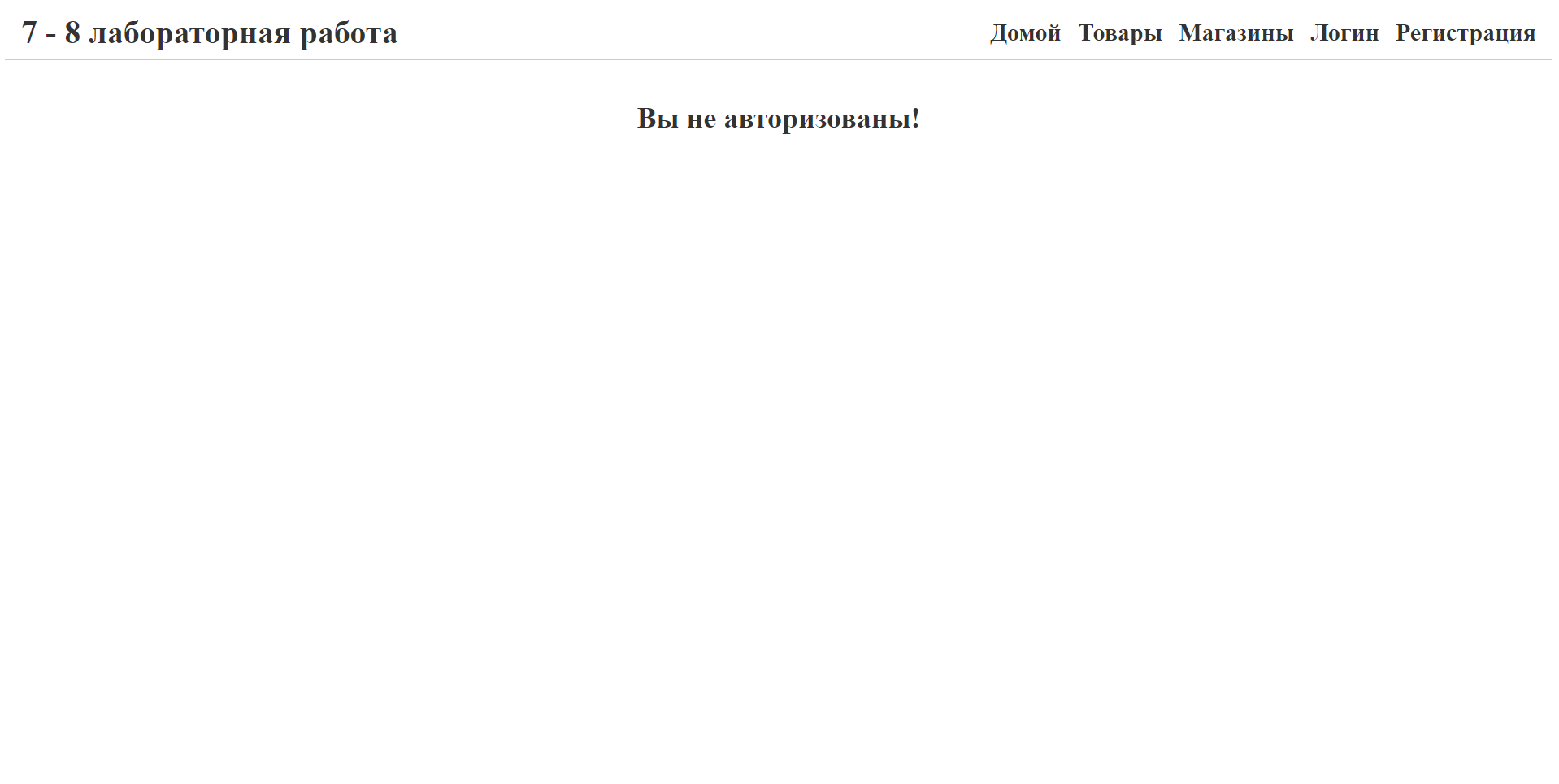


Рис. 3 Страница от неавторизованного пользователя

Вывод

В данной лабораторной работе были приобретены навыки работы с регистрацией и авторизацией на фреймворке Django. В ходе работы была выполнена проверка на роль пользователя, чтобы только сотрудник имел доступ к редактированию данных, а обычный пользователь этого не мог, а также была осуществлена проверка на то, авторизован ли пользователь или нет.

Код программы

login.html

{% extends "base.html" %}

{% block content %}

<h1>Авторизация</h1>

{% if form.errors %}

<p>Ваш логин или пароль был введён неправильно, попробуйте ещё раз.</p>

{% endif %}

{% if next %}

{% if user.is\_authenticated %}

<p>Ваш акаунт не имеет доступа к данной странице.</p>

{% else %}

<p>Пожалуйста войдите, чтобы увидеть эту страницу</p>

{% endif %}

{% endif %}

<form method="post" action="{% url 'login' %}">

{% csrf\_token %}

{{ form.username.label\_tag }}

{{ form.username }}

{{ form.password.label\_tag }}

{{ form.password }}

<div class="buts">

<button class="btn" type="submit">Авторизоваться</button>

<p><a class="btn" href="{% url 'register' %}">Регистрация</a></p>

<input type="hidden" name="next" value="{{ next }}"/>

</div>

</form>

{% endblock %}