

веб-камеры и получения кадров, позже совместно с keras и numpy происходит обнаружение лица на кадре и определение эмоции, а также на кадр добавляется рамка и эмоция, обнаруженная на данном кадре. Так как результатом работы модели является кортеж из пяти элементов с размером от нуля до единицы, то далее идёт постройка двух графиков, первый показывает величину каждой эмоции на кадре, а второй – величины всех эмоций на протяжении нескольких кадров. Для постройки графиков используются библиотеки pandas и matplotlib.

Далее с помощью tkinter и pil создаём GUI и реализуем вывод изображения с рамкой лица и эмоцией черно-белого изображения, которое видит нейронная сеть, график эмоций за один кадр и график эмоций за несколько кадров.

Для того чтобы запустить приложение, нужно иметь установленный язык программирования Python. А также для корректной работы требуется наличие веб-камеры. Также есть возможность создать исполняемый файл и запускать приложение через него.

**П. А. Гомонов, Е. А. Ружицкая**  
(ГГУ им. Ф. Скорины, Гомель)

## **РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ WEB-ПРИЛОЖЕНИЯ «ТРЕЙДИНГОВАЯ ПЛАТФОРМА»**

Всё большую популярность набирают web-приложения, которые используются в различных отраслях. Отрасль торговли также нуждается в автоматизации работы.

Разработана серверная часть web-приложения для трейдинговой платформы. Приложение использует микросервисную архитектуру и предназначено для организации торговли.

В приложении пользователи создают предложения о покупке или продаже товаров, сервер находит подходящие предложения и автоматически обменивает товары каждого из пользователей, входящих в процесс обмена. После того, как сервер находит такие подходящие приложения, каждому из пользователей отправляется сообщение о покупке или продаже товара.

В приложении реализована возможность регистрации новых пользователей или авторизации в системе, если учетная запись уже

создана, а также валидация данных. При разработке системы авторизации и регистрации использовался JSON Web Token.

В качестве базы данных использовалась объектно-реляционная СУБД PostgreSQL, преимуществом которой являются высокая надёжность и производительность. Связь с базой данных осуществляется через ORM. Разработана база, состоящая из 8 таблиц, содержащих следующую информацию: список пользователей, список типов валюты, предмет торговли, список цен предметов, информация о продавце, просмотренные пользователем товары, предложения пользователей, совершенные сделки.

Приложение разработано с использованием фреймворка Django, основанном на языке Python. В Django встроена система контроля версий баз данных, которая позволяет отслеживать, управлять и изменять все изменения схемы базы данных. Приложение не зависит от операционной системы, на которой оно запущено.

**А. С. Гончаров, Д. С. Кузьменков**  
(ГГУ им. Ф. Скорины, Гомель)

## **РАЗРАБОТКА КЛИЕНТСКОЙ ЧАСТИ СИСТЕМЫ УПРАВЛЕНИЯ ВЗАИМООТНОШЕНИЯМИ С КЛИЕНТАМИ**

В современном мире достаточно хорошо развит средний и малый бизнес. Когда идёшь по улице, всегда можно заметить парикмахерскую, салон красоты, какой-то медицинский центр и др. Многие из этих организаций используют какие-то программы для составления расписания работы сотрудников, учёта их больничных, отпусков, зарплаты, записи клиентов, составления различной статистики. К сожалению, зачастую люди не используют систему управления взаимоотношениями с клиентами (далее СУВК). Некоторые используют Microsoft Excel, некоторые записывают в тетради или в блокноте, а кто-то использует несколько приложений, так как зачастую не хватает возможностей в существующих СУВК.

Для упрощения управления взаимодействия с клиентами, было разработано специальное приложение с простым и удобным пользовательским интерфейсом, множеством новых возможностей, которых не хватает в конкурирующих приложениях. В данной СУВК появляется возможность следить за количеством расходных материалов, их