

которые несут информацию в виде изображений – визуально понятное и приятное оформление при этом всегда должны учитываться, ориентируясь на самого неопытного пользователя. В результате была построена структура мобильного приложения, разработано само приложение, произведено его функциональное наполнение, организован удобный поиск рецептов, реализована возможность пополнять приложение рецептами пользователей. Приложение в своем наполнении содержит авторские рецепты, для приготовления которых достаточно иметь самые распространенные и доступные продукты.

Мобильное приложение спроектировано с помощью интегрированной среды разработки Android Studio, пользовательский интерфейс приложения разрабатывался с использованием языка программирования XML, с помощью объектно-ориентированного языка Java реализовано функциональное наполнение интерфейса. Разработка предназначена для более приятного и конструктивного сопровождения процесса приготовления еды для пользователей в повседневной жизни, что в конечном итоге, со временем, отразится в положительную сторону на его общем самочувствии и здоровье.

Н. А. Коледа

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **А. В. Воруев**, канд. техн. наук, доцент

ПРИМЕНЕНИЕ CODELESS ETL TOOL

Для построения ETL пайплайнов необходимы знания различных языков программирования, таких как python, java, scala, а так же дополнительных инструментов для работы с данными, например apache spark. При этом возникает вопрос эффективности построенной системы в расходе ресурсов, таких как RAM, CPU и время. Одним из решений этой проблемы является приложение, которое позволяет через веб-интерфейс конфигурировать последовательности загрузки, трансформации и выгрузки данных.

Джоб представляет собой цепь отдельных стейджей, связанных между собой. Он описывает поток данных от источника данных к цели данных. Обычно стейдж имеет как минимум один вход данных и/или один выход данных. Однако некоторые этапы могут принимать более одного входа и иметь несколько выходов. Пайплайн – это соединение нескольких джобов, которое можно запустить.

Исходный код оборачивается в 2 докер образа. Один из которых основное приложение, содержащее spring boot приложение, второй содержит скала-спарк код для выполнения ETL преобразований. Далее эти образы помещаются в любое хранилище образов. Приложение запускается в kubernetes кластере в отдельном пространстве имен на основе, сохраненного ранее, образа. Каждое создание или изменение сущности на клиенте влечет за собой изменение соответствующей сущности в Kubernetes. Так при создании нового проекта, создается новое пространство имен, при создании джоба – configmap, при создании пайплайна – workflow template. Взаимодействие с образом содержащим спарк код происходит при запуске джоба или пайплайна. При этом создается под с контейнером и как переменные окружения добавляются данные из configmap, созданной на этапе конфигурации джоба пользователем. Такой под работает как управляющий узел для исполнителей. Их количество может быть изменено пользователем.

При запуске пайплайна создается workflow с ссылкой на ранее созданные workflow template. При необходимости, меняются параметры, влияющие на быстродействие и последовательность выполнения.

Н. А. Коноплич

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. В. Васильевич**, ст. преподаватель

РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ МЕДИЦИНСКОГО КОНСУЛЬТАНТА С ИСПОЛЬЗОВАНИЕМ СУБД POSTGRESQL И NODE.JS

Веб-приложения – это приложения, с которым можно взаимодействовать с помощью браузера, которое состоит из Frontend и Backend части. В данной статье будет представлен мой опыт разработки такого приложения.

В разработке такого приложения есть три важных этапа: разработка клиентской части (Frontend), разработка серверной части (Backend) и создание базы данных (Database). Для разработки клиентской части я выбрал Java-Script библиотеку React, для разработки серверной части я выбрал модуль Node.js – Express, а для создания базы данных была выбрана СУБД PostgreSQL.

Первоочередной задачей является создание базы данных. Для создания медицинского консультанта понадобилось две таблицы, в одной из которых находятся симптомы (symptoms), с соответствующими