

Целью работы является разработка веб-приложения «Облачное хранилище», которое реализует хранение и передачу данных с отдельного сервера на сервер ASP.NetCore, а тот в свою очередь на сервер React. Js.

Модель данных реализована с помощью фреймворка Entity Framework, который предоставляет более гибкий подход работы с данными. Проект состоит из двух серверов. Сервер на React.Js реализует отображение веб-страниц в браузере, его логика написана на языке программирования JS. Сервер на Asp.netCore реализует обращение к данным, находящимся в базе данных и передачу этих данных на сервер React.Js, его логика написана на языке программирования C#.

Функционально веб-приложение решает следующие задачи: авторизация и регистрация через внешние сервисы; обеспечивает загрузку и скачивание файлов различных форматов; просмотр данных; взаимодействие с файловой системой разделов; персистентность разделов; навигация по файловой системе; обеспечивает доступ к данным других пользователей, если они этого пожелают; обеспечивает безопасность от внешних хакерских атак (Ddos и Brutforce атаки).

Разработанное веб-приложение ориентировано на пользователей, которые хотят хранить свои данные в сети Интернет, а не на физических носителях. Данное приложение поможет надежно обеспечивать доступ к файлам, так как сервер надежно защищен от нескольких популярных хакерских атак.

УДК 004.056

Студ. В.А. Реут

Науч. рук. ст. преп. А.С. Наркевич
(кафедра программной инженерии, БГТУ)

РАЗРАБОТКА СЕРВИСА ДЛЯ СОХРАНЕНИЯ ЗАМЕТОК NOTES-EVERYWHERE

В настоящее время благодаря развитию технологий у человечества появилось множество новых возможностей, однако, вместе с тем и много проблем. Во всем информационном потоке нужно как-то разбираться и не потерять мысль, в этом поможет запись своих мыслей и создание заметок. На самом деле, ведение записей может оказаться полезным для многих. Изложение мыслей в письменной форме помогает улучшить настроение, самочувствие и повысить стрессоустойчивость. Но не всегда есть место под рукой, где можно оставить запись. Главное

неудобство приложений, которые помогают решить эту проблему, то, что они ограничены в устройствах: разрядился телефон, нет интернета на компьютере, и вы уже не сможете записать все что хотели, что может привести к потере важной для пользователя информации.

Цель работы создать сервис сохранения заметок, с возможностью: регистрации и авторизации, доступа с помощью телеграмм-бота и веб приложения, синхронизацией между клиентами и взаимодействием с заметками даже после отключения сервера или отключения интернета у клиента.

Исходя из поставленной цели, реализованы сервер, клиент и телеграмм-бот.

Для выполнения всех вышеперечисленных условий был выбран следующий набор технологий. Технологии для клиентской части приложения: thymleaf, JavaScript. Технологии для серверной части приложения: Springboot, Mysql, Hibernate. Клиент, сервер взаимодействуют между собой с помощью передачи сообщений в формате json, клиент телеграмм-бота взаимодействует с тем же сервером, что и веб-приложение с помощью специальной java библиотеки Telegram-bot-api.

Основной задачей, которая ставилась для проекта, была возможность нормального функционирования при возможном отключении от интернета на время или временной недоступности сервера. В этих случаях помогает телеграмм, который дает возможность серверу обработать сообщение от пользователя, даже если во время запроса к боту был отключен интернет, а, если интернет был отключен у пользователя, клиентское приложение телеграмма отошлет сообщение при следующем подключением к сети.

Обработкой всех данных, формированием и передачей их по удаленным каналам отвечает сервер. Клиент посылает запросы на сервер по HTTP протоколу версии 1.1. Все процессы, отвечающие за хранение данных, возложены на MySQL.

ЛИТЕРАТУРА

1. Документация по Hibernate ORM [Электронный ресурс] – Режим доступа: <https://hibernate.org/orm/documentation/6.0/> – Дата доступа: 23.03.2022.
2. Документация по TheSpringFramework [Электронный ресурс] – Режим доступа: <https://docs.spring.io/springframework/docs/2.5.x/spring-reference.pdf/> – Дата доступа: 20.03.2022.