

тексте разработка приложения для составления расписания учебных занятий на факультете становится актуальной и необходимой задачей. Такое приложение поможет администрации факультета легче организовывать учебный процесс, минимизировать конфликты в расписании и повысить общую эффективность обучения.

Приложение позволит автоматизировать проверку занятости преподавателей, контролировать их общую недельную нагрузку, а также занятость аудиторий в определенный момент. Полезной может оказаться функция учёта количества проведенных занятий по дисциплинам, которая позволит контролировать выполнение учебного плана специальности факультета.

Приложение будет поддерживать такие базовые функции, как сохранение/загрузка расписания, создание Excel файла расписания.

Для разработки фронтенд-части, а также бизнес-логики приложения был выбран Godot Engine, имеющий в своём распоряжении готовые компоненты пользовательского интерфейса. К сожалению, Godot не имеет библиотек для работы с Microsoft Excel, а потому для скриптов экспорта расписания в файлы формата xlsx был выбран язык программирования Python.

Используя тот же Python, можно автоматизировать размещение расписания на сайте факультета, загрузки расписания в базу данных или импорт дисциплин, которые будут преподаваться в данном семестре, в приложение.

Приложение потребляет незначительное количество ресурсов компьютера, не требует установки и может быть легко перенесено на другой компьютер обычным копированием корневой папки.

**Д. А. Моторин, Г. Л. Карасёва**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ВЕБ-АРІ ИНТЕРНЕТ МАГАЗИНА**

Интернет стал неотъемлемой частью нашей жизни. Люди используют его каждый день, для общения со всем земным шаром, для скачивания файлов и приложений, для поиска информации по требуемой теме и для посещения сайтов. По статистике агентства

We Are Social за 2023 год интернетом пользуется 64,4 % мирового населения, то есть 4,62 млрд. человек. Поэтому разработка сайтов является одним из важнейших направлений ИТ-индустрии.

Основная цель исследования заключалась в анализе принципов работы интернет магазина и основных принципов создания серверной части сайта. Главной задачей является обеспечение возможности заказа товара, просмотра заказанных товаров и управление базой данных этого магазина вместе с обеспечением необходимых функций магазина.

С помощью платформы .Net, а в частности Asp.Net Web API, создаётся проект Web-API интернет магазина. Проект имеет n-уровневую архитектуру и состоит из 3-х частей:

1. DAL (Data Access Layer) – уровень доступа к данным, хранит модели, описывающие используемые сущности, хранит отдельные классы (например, класс контекста Entity Framework). Также тут хранятся репозитории, через которые уровень бизнес логики взаимодействует с базой данных.

2. BLL (Business Logic Layer) – уровень, отвечающий за обработку информации и передачу её на уровень представления.

3. Presentation Layer – уровень представления, отвечающий за обработку запроса пользователя и предоставление ответа на эти запросы.

Для проверки работы используется Swagger – инструмент, с помощью которого можно проверить работу Web-API.

Информация об интернет магазине хранится с SQL Server. Получение доступа и использование базы данных происходит с помощью Entity Framework.

Разработанный WEB-API является следствием глубокого анализа необходимых функций и возможностей, необходимых для интернет магазина. Проект имеет удобную для понимания структуру и предоставляет все возможности интернет магазина. Возможно создание полноценного сайта на основе этого проекта при добавлении пользовательского интерфейса.