

Заключение. В результате выполнения работы был разработан модуль, позволяющий пользователям системы ссылаться на элементы тестов, наблюдать за прохождением теста и общаться через чат.

На данный момент продолжается работа над добавлением недостающего функционала и улучшением производительности приложения.

1. Официальная документация Angular [Электронный ресурс] – Режим доступа: <https://angular.io/docs>. – Дата доступа: 23.03.2021.
2. Официальная документация библиотеки SignalR Core [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/en-us/aspnet/core/signalr/introduction?view=aspnetcore-5.0>. – Дата доступа: 23.03.2021.

ПРОГРАММИРОВАНИЕ RESTFUL ПРИЛОЖЕНИЙ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ GOLANG

Кутузов К.О.,

*учащийся 4 курса Оршанского колледжа ВГУ имени П.М. Машерова,
г. Орша, Республика Беларусь*

Научный руководитель – Трибис А.В., преп.

При разработке крупных веб-приложений необходимо планировать дальнейшее развитие веб-проекта. Следует учитывать, что не стоит создавать модули, которые будут хранить в себе много legacy-кода, чтобы в последствии полностью переписывать весь проект. Для этого веб-проект разделяется на отдельные модули, такие как серверы-шаблонизаторы, прокси-сервера, restful сервера, другие веб-сервисы и микросервисы, а также сервера базы данных. Для этих целей можно выбрать язык программирования Golang, который имеет в себе возможность кросс-компиляции, что позволяет компилировать приложения, написанные на golang даже под android.

Цель исследования – изучить возможности языка программирования Golang для создания веб-приложений.

Материалы и методы. В основном язык Golang используется при разработке различных RESTful API веб-сервисов. REST API – это прикладной программный интерфейс (API), который использует HTTP-запросы для получения, извлечения, размещения и удаления данных.

Для изучения возможностей языка программирования Golang мною было разработано RESTful приложение для веб-проекта «KinoLab».

Создание RESTful API на языке программирования Golang сопровождается использованием библиотек «из-под коробки» для работы с HTTP протоколом. Но для упрощённой разработки можно воспользоваться фреймворком «Gin», который является обёрткой над стандартной библиотекой http [2].

Приложения RESTful в основном основаны на архитектурном паттерне MVC (Model View Controller), где Model – отдельный модуль для работы с данными и базой данных, Controller – обработчик запросов, который определяет операции, производимые при получении определённого запроса сервером, View – модуль, определяющий выполнение определённых обработчиков на каждый Endpoint. Поэтому RESTful приложение, разрабатываемое в данной статье, придерживается данного паттерна.

Для примера, ниже представлен код, в котором разрабатываемое RESTful приложение запрашивает информацию о списке таблиц в базе данных СУБД PostgreSQL и возвращает результат в виде JSON.

Листинг 1 – Пример кода.

```
//Контроллер – в данном случае определяет выполняемые функции url'ам и роутит сервер
r := gin.New()
r.GET("/tables", getAllTables)
r.Run(":Host:Port")
//Вид – определяет данные, предоставляемые контроллеру
func FetchAll(sql string) []string {
    var rows []string
    rows_result, err := conn.Query(context.Background(), sql)
    if err != nil {
        fmt.Println(err)
    }
}
```

```

    }
    for rows_result.Next() {
        var row string
        err = rows_result.Scan(&row)
        if err != nil {
            fmt.Println(err)
            continue
        }
        rows = append(rows, row)
    }
    return rows
}
//Модель – обращается к базе данных, получает и возвращает список таблиц базы данных
func getAllTables(c *gin.Context) {
    sql := "SELECT table_name FROM information_schema.tables where table_schema='public' ORDER BY table_name;"
    tables := storage.FetchAll(sql)
    c.JSON(200, tables)
}

```

Результаты и их обсуждение. Результатом исследования является веб-сервис RESTful API, который реализует общение с сервером базы данных, а также выдаёт необходимую информацию в виде JSON, запрашиваемого клиентом.

Клиентом непосредственно может являться отдельное веб-приложение, которое напрямую использует весь функционал данного RESTful приложения, не нагружая себя лишней задачей, такой как общение с сервером базы данных, приём файлов через POST запросы, а также роутинг файловой системы, для просмотра файлов по HTTP протоколу.

К проблемам разработки приложений на языке Golang можно отнести особый подход, который требуется от разработчика. В данном языке отсутствует ООП, перегрузка методов/функций, исключения и обобщенное программирование, что доставляет сложность разработчику, который переходит с объектно-ориентированного языка программирования [1].

Закключение. Изучив вопрос реализации веб-сервисов на языке Golang, путем написания RESTful приложения, позволяющего осуществить общение с сервером базы данных и передачу запрашиваемой информации в виде JSON клиенту или стороннему серверу, а также написания веб-приложения «Админ-панель», реализующего отправку запросов на RESTful API сервер, можно сделать вывод: язык программирования Golang со всеми своими особенностями отлично подходит для написания различного рода веб-приложений.

1. За что ругают Golang и как с этим бороться? [Электронный ресурс]. URL: <https://habr.com/ru/post/282588/> (Дата обращения: 29.03.2021)
2. Введение в REST API – RESTful веб-сервисы: [Электронный ресурс]. URL: <https://habr.com/ru/post/483202/> (Дата обращения: 29.03.2021)

СЕРВИС PLICKERS КАК РЕСУРС ОПТИМИЗАЦИИ ВЗАИМОДЕЙСТВИЯ СУБЪЕКТОВ ОБРАЗОВАТЕЛЬНОЙ СРЕДЫ

Литвин В.В.,

студент 4 курса ВГУ имени П.М. Машерова, г. Витебск, Республика Беларусь

Научный руководитель – Галузо И.В., канд. пед. наук, доцент

В настоящее время возрастает интерес к организации информационных технологий в образовательную среду. В статье рассматривается такой ресурс оптимизации взаимодействия субъектов образовательной среды как карточки QR – кодов Plickers, описывается ее функционал и проводится анализ того, как ее внедрять в учебный процесс с целью повышения эффективности усвоения знаний и получения учителем объективной, оперативной информации о ходе учебно-познавательной деятельности учащихся. Цель работы – описать и проанализировать метод оперативного тематического контроля знаний по физике.