

"НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО"
(УНИВЕРСИТЕТ ИТМО)

ЗАДАНИЕ НА КУРСОВОЙ ПРОЕКТ (РАБОТУ)

Студент	Волобуев И. К.		
		(Фамилия, И., О.)	
Факультет	ПИиКТ	Группа	P41081
Направление (специальность)	09.04.04 Программная инженерия		
Руководитель	Государев И.Б., к.п.н., доцент		
	(Фамилия, И., О., должность,)		
Дисциплина	Проектирование и анализ языков веб-решений		
Наименование темы:	Разработка микросервиса по подготовке промышленной документации для элеваторного предприятия на языке программирования Golang		
	(Наименование сайта)		

Задание Разработать микросервисное веб-приложение по подготовке промышленной документации для элеваторного предприятия на языке программирования Golang

Краткие методические указания (задачи работы)

Проанализировать особенности предметной области, для которой будет разрабатываться будущее веб-приложение

Изучить особенности построения микросервисных веб-приложений

Спроектировать архитектуру будущего микросервисного веб-приложения

Спроектировать архитектуру микросервиса выполняющего задачу подготовки промышленной документации

Проанализировать возможности языка программирования Golang в сфере веб-разработки

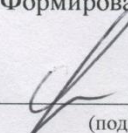
Спроектировать API будущего веб-приложения

Разработать веб-приложение

Содержание пояснительной записки

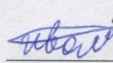
Содержание. Введение. Проектирование архитектуры микросервисного приложения. Разработка сервиса «Формирование отчётов». Вывод. Список источников. References.

Руководитель


(подпись)

И. Б. Государев

Студент


(подпись)

И. К. Волобуев

(Фамилия И.О.)

"НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО"
(УНИВЕРСИТЕТ ИТМО)

ГРАФИК КУРСОВОГО ПРОЕКТА (РАБОТЫ)

Студент Волобуев И. К.

(Фамилия, И., О.)

Факультет ПИиКТ Группа Р41081

Направление 09.04.04 Программная инженерия
(специальность)

Руководитель Государев И. Б., к.п.н., доцент

(Фамилия, И., О., должность)

Дисциплина Проектирование и анализ языков веб-решений

Наименование темы: Разработка микросервиса по подготовке промышленной документации для элеваторного предприятия на языке программирования Golang

(Наименование сайта)

№ п/п	Наименование этапа	Дата завершения		Оценка и подпись руководителя
		Планируемая	Фактическая	
1.	Проектирование микросервисного приложения по ведению производственного учёта и архитектуры сервиса для подготовки промышленной документации.	апрель	апрель	
2.	Разработка и тестирование спроектированного сервиса. Подготовка отчёта. Защита проекта.	май	май	

Руководитель

(подпись)

И. Б. Государев

Студент

(подпись)

И. К. Волобуев

(Фамилия И.О.)

"НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО"
(УНИВЕРСИТЕТ ИТМО)

АННОТАЦИЯ К КУРСОВОМУ ПРОЕКТУ (РАБОТЕ)

Студент Волобуев И. К.

(Фамилия, И., О.)

Факультет ПИиКТ Группа Р41081

Направление 09.04.04 Программная инженерия
(специальность)

Руководитель Государев И. Б., к.п.н., доцент

(Фамилия, И., О., должность)

Дисциплина Проектирование и анализ языков веб-решений

Наименование темы: Разработка микросервиса по подготовке промышленной документации для элеваторного предприятия на языке программирования Golang

(Наименование сайта)

ХАРАКТЕРИСТИКА КУРСОВОГО ПРОЕКТА (РАБОТЫ)

**1. Цель и задачи
работы**

☒ Предложены
студентом

☐ Определены
руководителем

Цель работы — спроектировать и создать сервис для будущего микросервисного приложения на языке программирования Golang

Задачи работы:

1. Проанализировать особенности предметной области, для которой будет разрабатываться будущее веб-приложения
2. Изучить особенности построения микросервисных веб-приложений
3. Спроектировать архитектуру будущего микросервисного веб-приложения
4. Спроектировать архитектуру микросервиса выполняющего задачу подготовки промышленной документации
5. Проанализировать возможности языка программирования Golang в сфере веб-разработки
6. Спроектировать API будущего веб-приложения

7. Разработать веб-приложение

2. Характер работы

☐ Расчет ☐ Конструирование ☐ Моделирование ☒ Другое

3. Содержание работы

Использованы полученные знания по дисциплине для создания веб-ресурса и
разработан веб-сервис, который выполняет задачу по подготовке промыш -
ленной документации и является частью единой системы обеспечивающий
производственный учёт элеваторного предприятия.

4. Выводы

Требования к проекту реализованы

Руководитель

(подпись)

И. Б. Государев

Студент

(подпись)

И. К. Волобуев
(Фамилия И.О.)

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ

«Национальный исследовательский университет ИТМО»
(Университет ИТМО)

Факультет программной инженерии и компьютерной техники

Направление (специальность) — 09.04.04 Программная инженерия

Специализация — Веб-технологии

Дисциплина — Инновационные исследования в вебе

Курсовой проект (работа)

ТЕМА: Разработка микросервиса по подготовке промышленной
документации для элеваторного предприятия на языке
программирования Golang

ВЫПОЛНИЛ

Студент группы

Р41081

№ группы



подпись

Волобуев Игорь Константинович

ФИО

ПРОВЕРИЛ

к.п.н., доцент

ученая степень, должность



подпись

Государев Илья Борисович

ФИО

САНКТ-ПЕТЕРБУРГ

2022 г.

Содержание

Введение	7
Проектирование архитектуры микросервисного приложения	9
Разработка микросервиса «Формирование отчётов»	11
Вывод	13
Список источников	13
References	14

Введение

Golang – многопоточный компилируемый язык программирования с открытым исходным кодом, разработанный в стенах компании Google. В ней же изначально использовался в качестве инструмента для решения внутренних задач. Авторы языка придерживались двух основных постулатов в процессе его проектирования. Первый принцип – разрабатываемый язык должен быть эффективным в плане возможностей для реализаций программных решений, работающих на распределённых системах и многоядерных процессорах. Относительно многих своих предшественников, Go можно оценить как новый язык программирования. Соответственно, Golang должен был как минимум не уступать в своих возможностях другим языкам, и в идеале предложить куда лучшие решения, чем уже имеющиеся. Вторым важным фактором была простота Go. Он должен быть прост в изучении как людям начинающим изучать основы программирования, так и тем, кто уже успел попробовать себя в других языках. [1]

В качестве своего основного ориентира и отправной точки разработчики выбрали язык программирования C. Далее, начали наращивать вокруг него функционал, одновременно с этим избавляясь на взгляд разработчиков от очевидных недостатков. На момент написания статьи, актуальной версией Go является «go 1.18». Ниже перечислены некоторые функциональные возможности, важные, в частности, для серверной веб-разработки:

- Горутины (goroutines) – реализация параллельных вычислений в экосистеме Go. Горутины куда более легковесны чем стандартные потоки, а также весьма практичны с точки зрения интеграции параллельных вычислений в код. Сюда же можно отнести возможность откладывать выполнение функций при помощи оператора «defer»;

- Сборщик мусора – система автоматической очистки памяти от ненужных объектов, тем самым избавляя разработчика от обязанности контроля за памятью;
- Стандартная библиотека, содержащая в себе всё необходимое, чтобы разработать серверное веб-приложение: начиная от инструментария для создания HTTP-сервера для отправки и получения запросов, и заканчивая интерфейсом для подключения и работы с базой данных;
- Встроенный менеджер пакетов «Go Modules» для управления внешними зависимостями веб-проекта;
- Встроенные средства тестирования для проверки правильной работоспособности программного кода. [2]

Для того чтобы достигнуть простоты и читабельности кода, авторы добивались, чтобы любая поставленная задача могла решаться только одним единственным способом. Таким образом, это могло бы поспособствовать программистам, которые включаются в разработку какого-либо проекта не с самого начала, быстро освоиться в нём и начать реализовывать свои идеи. Также, для этого разработчики языка задались целью, чтобы каждая конструкция языка могла быть представлена только в одном варианте. Так, например, из всех возможных способов реализации циклов в языках программирования, в Go присутствует только цикл с определённым числом шагов. Помимо этого, авторы отошли от концепции классов, отказавшись от наследования и полиморфизма.

По результатам общественного опроса в 2021 году разработчиков на Go компанией JetBrains, специализирующейся на выпуске IDE (интегрированных сред разработки) для самых разных языков программирования, Golang чаще всего используется для реализации веб-приложений, утилит и IT-инфраструктур. Также, большей части разработчиков для выполнения своих задач достаточно стандартных средств языка Go, за исключением Go-роутера. [3]

Проектирование архитектуры микросервисного приложения

В качестве предметной области проектируемой системы была выбрана элеваторная промышленность. Это отрасль агропромышленного комплекса, специализирующаяся на подработке и хранении зерновых культур. Приложение проектировалось в соответствии с нуждами компании «ООО Фасад-Комплект», предоставляющая услуги элеваторов. В ходе общения с сотрудниками, были сформированы следующие требования к функционалу приложения:

- возможность ведения учёта в процессе приёмки и отгрузки зерна;
- ведение учёта во время хранения зерновой культуры;
- ведение бухгалтерского учёта;
- ведение учёта персонала предприятия;
- составление заявок на детали для комплексов и оборудования элеватора;
- создание отчётов на основе данных бухгалтерского и элеваторного учёта.

На основе этих требований, было выделено пять микросервисов, составляющих единую систему приложения. Архитектура разрабатываемого приложения, со всеми сервисами и взаимосвязями между ними, представлена на рисунке 1.

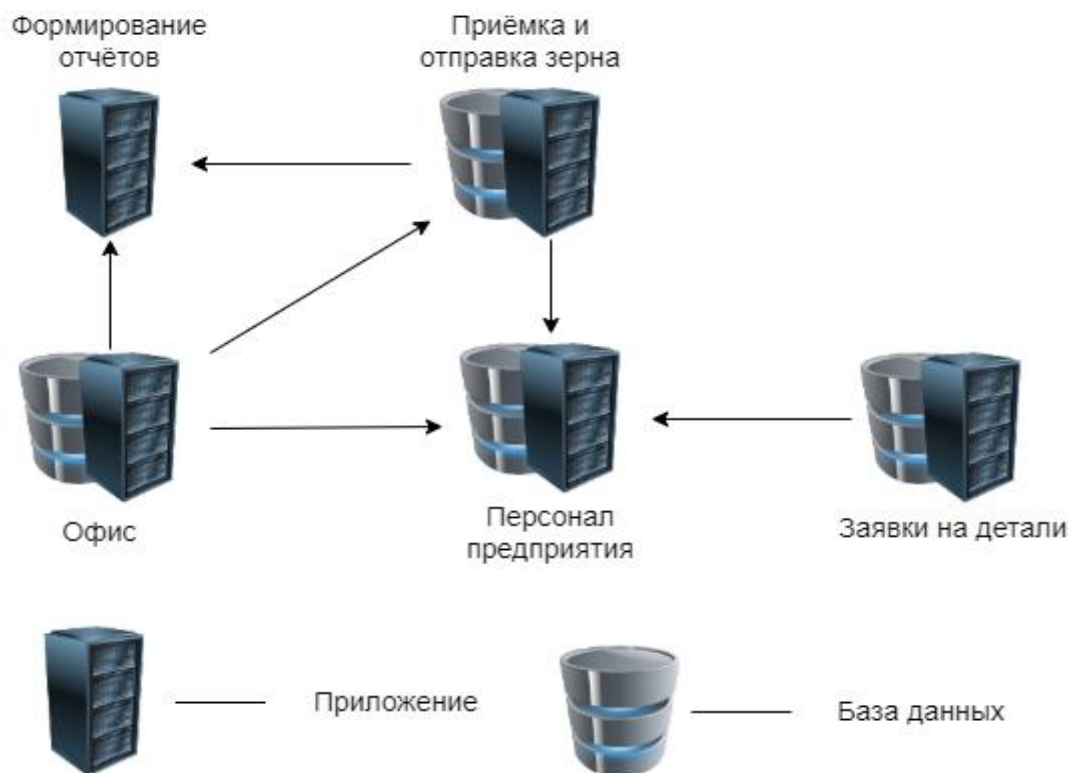


Рисунок 1. – Архитектура приложения

Можно заметить, что к четырём микросервисам в данной системе прилагается база данных. Её наличие обуславливается необходимостью хранения определённых записей присущих специфике сервиса:

- «Офис» – данные бухгалтерского учёта;
- «Приёмка и отправка зерна» – записи учёта хранимых зерновых культур клиентов;
- «Заявки на детали» – обращения работников по вопросам снабжения комплексов элеватора комплектующими деталями;
- «Персонал предприятия» – учёт рабочего персонала, а также сервис авторизации пользователей других микросервисов.

Единственный сервис, не нуждающийся в хранении каких-либо данных, это «Формирование отчётов», так как информация, на основе которой и заполняется документация, уже хранятся в базах данных сервисов «Приёмка и отправка зерна» и «Офис». Определив основные критерии и роль микросервиса в приложении, можно перейти следующему шагу – реализации.

Разработка микросервиса «Формирование отчётов»

В архитектуре самого микросервиса можно выделить два ключевых слоя:

- доменная модель, по сути, представляющую из себя конфигурационные данные приложения;
- сетевая часть, отвечающая за приём клиентских запросов и отправку им ответов.

С точки зрения бизнес-логики, задача заполнения документа выполняется путём записи данных в файл-шаблон. Шаблон, в свою очередь, представляет собой excel-файл, в котором адреса ячеек соответствуют пустым полям для определённых значений, которые обходимо заполнить. Например, если взять excel-шаблон «Товарно-транспортной накладной (зерно)», то в нём ячейка «AA5» предназначена для номера этой накладной, а в ячейки «I6», «L6» и «V6» вносится дата составления документа. [4]

Предварительно, составляется конфигурационный файл, содержащий в себе данные о значениях с соответствующими им полями. В процессе инициализации структур приложения информация из конфигурационного файла записывается в структуры данных приложения, откуда позже будут браться значения для записи их в файл. В данном микросервисе было принято решение для конфигурационной документации пользоваться нотацией yaml.

Пример подобного файла и структуры данных для «Товарно-транспортной накладной (зерно)» продемонстрирован в таблице 1.

Таблица 1 – Представление конфигурационных данных в приложении

configuration.yaml (часть)	report: zpp: template: report/forma-sp-31.xlsx excel: id: AA5 day: I6 month: L6
----------------------------	--

	year: V6
Структура данных ZppConfiguration (часть)	<pre> type ZppConfiguration struct { id string day string month string year string } </pre>

Для решения задачи десериализации данных из yaml-файла в программную среду использовался проект с открытым исходным кодом «Viper». Данная библиотека также широко используется для работы со структурами типа JSON, HCL, TOML, а также для чтения данных из переменных окружения, буфера и флагов командной строки. [5]

Заполнение данными excel-шаблонов было реализовано при помощи библиотеки «Excelize». Функционал данной библиотеки поддерживает работу с таблицами самых разных версий: XLAM, XLSM, XLSX, XLTM, XLTX. [6]

Как было оговорено ранее, информация, необходимая для создания файла, приходит от других микросервисов. Разработанная сетевая часть приложения принимает и обрабатывает данные в формате JSON. Сами данные поступают на сервер в виде GET-запроса, так как сущность этих запросов не несёт за собой какое-либо изменение данных, а только получение их в другом виде.

К каждому типу документации присущи два URI: на получение документа в формате docx-файла и pdf-файла. Примеры запросов для таких URI представлены в таблице 2.

Таблица 2 – Примеры запросов.

Запрос	Назначение
GET /zpp/docx	Получение ТТН в виде docx-файла
GET /zpp/pdf	Получение ТТН в виде pdf-файла

Тело JSON в запросе будет содержать данные типа «наименование атрибута-значение». Далее, эти атрибуты соотносятся с теми, что записаны в конфигурационных файлах, и после записываются в определённые ячейки. В качестве ответа клиент получает наименование созданного файла и байтовую последовательность, представляющую содержание документа.

Вывод

Таким образом, средствами языка Golang был разработан микросервис решающий задачу заполнения промышленной документации для элеваторного предприятия. Путём консультации с персоналом предприятия, были сформированы основные требования к будущему микросервисному приложению, а также весь список используемой отраслевой документации и особенности её ведения. На основе уже этой информации, началось проектирование сервиса для подготовки документации.

Были исследованы возможности и особенности разработки серверных веб-приложений на языке программирования Go. Также изучены некоторые проекты с открытым исходным кодом на данном языке, упрощающие процесс веб-разработки. В результате чего был сделан вывод, что Golang обладает функциональными возможностями удовлетворяющие запросам серверной веб-разработки. После чего, были спроектированы API микросервиса и его структура. И последним шагом, стала непосредственная реализация проекта.

Список источников

1. GoDev. The Go Blog. Go, Open Source, Community. [Электронный ресурс]. URL: <https://go.dev/blog/open-source> (дата обращения 09.04.2022)
2. Батчер М., Фарина М. Go на практике / Москва. ДМК Пресс, 2017. С. 23 – 38

3. JetBrains. Экосистема разработки в 2021 году. Go [Электронный ресурс]. URL: <https://www.jetbrains.com/lp/devecosystem-2021/go/> (дата обращения 09.04.2022)
4. КонсультантПлюс. Товарно-транспортная накладная (зерно) (Типовая межотраслевая форма № СП-31) (ОКУД 0325031) [Электронный ресурс]. URL: http://www.consultant.ru/document/cons_doc_LAW_32449/613140af737af0a66a88d5c56e8b105999b40916/ (дата обращения 11.04.2022)
5. GitHub. Viper [Электронный ресурс]. URL: <https://github.com/spf13/viper> (дата обращения 14.04.2022)
6. Pkg.GoDev. Excelize. Go library for reading and writing XLSX files [Электронный ресурс]. URL: <https://pkg.go.dev/github.com/xuri/excelize/v2> (дата обращения 14.04.2022)

References

1. GoDev. The Go Blog. Go, Open Source, Community. [Web resource]. URL: <https://go.dev/blog/open-source> (date of access 09.04.2022)
2. Butcher M., Farina M. Go in practice / Moscow. DMK Press, 2017. С. 23 – 38
3. JetBrains. The State of Developer Ecosystem 2021. Go [Web resource]. URL: <https://www.jetbrains.com/lp/devecosystem-2021/go/> (date of access 09.04.2022)
4. ConsultantPlus. Consignment note (grain) (Standard intersectoral form N SP-31) (OKUD 0325031) [Web resource]. URL: http://www.consultant.ru/document/cons_doc_LAW_32449/613140af737af0a66a88d5c56e8b105999b40916/ (date of access 11.04.2022)
5. GitHub. Viper [Web resource]. URL: <https://github.com/spf13/viper> (date of access 14.04.2022)

6. Pkg.GoDev. Excelize. Go library for reading and writing XLSX files
[Web resource]. URL: <https://pkg.go.dev/github.com/xuri/excelize/v2> (date of
access 14.04.2022)

"НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО"
(УНИВЕРСИТЕТ ИТМО)

ОТЗЫВ РУКОВОДИТЕЛЯ
о выполнении курсового проекта (работы)

Студент Волобуев И. К.
(Фамилия, И., О.)

Факультет ПИиКТ Группа Р41081

Направление (специальность) 09.04.04 Программная инженерия

Руководитель Государев И.Б. к.п.н., доцент
(Фамилия, И., О., должность)

Дисциплина Проектирование и анализ языков веб-решений

Наименование темы: Разработка микросервиса по подготовке промышленной документации для элеваторного предприятия на языке программирования Golang
(Наименование темы)

ОЦЕНКА КУРСОВОГО ПРОЕКТА (РАБОТЫ)

№ п/п	Показатели	Оценка			
		5	4	3	0
1.	Проект создан обучающимся самостоятельно	✓			
2.	Созданные элементы сайта раскрывают тематику и название фирмы	✓			
3.	Проект технологически грамотный	✓			
4.	Оформление отвечает требованиям к отчету	✓			
5.	Во время защиты обучающийся показал умение кратко, доступно представить результаты работы, умение анализировать, аргументировать свою точку зрения, делать обобщение и выводы, адекватно ответить на поставленные вопросы.	✓			
ИТОГОВАЯ ОЦЕНКА		отлично			

Отмеченные достоинства:

В отчете студента отражены полученные в ходе выполнения проекта навыки, соответствующие компетенциям по данной тематике. Студент показал себя личностью пунктуальной, ответственной, готовой к изучению нового материала. В процессе работы студент подтвердил свои навыки в области проектирования веб-ресурсов.

Отмеченные недостатки:

—

Заключение:

Студент подтвердил навыки, полученные за время обучения по указанной специальности.

Руководитель

(подпись)

И.Б. Государев

« 20 » июля 2022