

# ОГЛАВЛЕНИЕ

	стр.
ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ .....	2
ВВЕДЕНИЕ .....	2
1. ОБЗОРНАЯ ЧАСТЬ .....	3
1.1. Описание предметной области .....	3
1.1.1. Описание серверной архитектуры .....	4
1.1.2. Описание модели внедрения программных продуктов Agile Scrum.....	4
1.1.3. Схема ручного сопровождения API-документации .....	4
1.2. Исследование существующих технологий.....	4
1.2.1. Перечень функций, подлежащих автоматизации .....	4
1.2.2. Выбор и обоснование критериев качества .....	4
1.2.3. Анализ аналогов и прототипов .....	4
1.2.4. Сравнение аналогов и прототипов.....	5
2. РАСЧЕТНО-КОНСТРУКТОРСКАЯ ЧАСТЬ.....	5
2.1. Определение требований к системе .....	5
2.2. Разработка структуры автоматизированной системы .....	5
2.3. Разработка структуры интерфейса взаимодействия пользователя с системой .....	5
2.4. Разработка алгоритмов программных модулей .....	5
2.5. Разработка плана проведения тестирования .....	5
3. ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ .....	6
3.1. Реализация разработанных алгоритмов.....	6
3.2. Тестирование и отладка системы .....	6
3.3. Руководство пользователя .....	6
ЗАКЛЮЧЕНИЕ.....	6
СПИСОК ЛИТЕРАТУРЫ.....	6

## ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

1. REST API – это набор правил, по которым следует обращаться к серверу для отправки или получения данных.
2. Пользователь – **живой человек**
3. Клиент – любое приложение которое **делает запросы на сервер**.  
Например, в роли клиента может выступать веб браузер, когда пользователь открывает веб-сайт.

## ВВЕДЕНИЕ

На сегодняшний день большинство крупных IT компаний для взаимодействия сервера и клиента используют REST API [1].

Например, компании вроде Яндекса, Google и т.п. Предоставляют открытые API методы своих сервисов чтобы разработчики могли интегрироваться с ними.

Компания ООО «ФорМакс» разрабатывает продукт Fonmix, серверная часть которого полностью базируется на технологии REST API, т.е. общение любого пользователя с сервером Fonmix осуществляется через REST API.

Основными клиентами [3] для сервера Fonmix являются:

1. Веб-сайт fonmix.ru – представляет собой веб интерфейс, в котором пользователи [2] могут управлять музыкой в своих заведениях: создавать плейлисты, составлять музыкальное расписание, добавлять рекламу в перерывах между песнями и т.п.
2. FM.Player – кроссплатформенный медиапроигрыватель разрабатываемый также в компании ООО «ФорМакс», с помощью которого воспроизводится медиа контент правообладателей.

3. Правообладатель – это исполнитель и изготовитель фонограмм, с которым заключается договор о дистрибуции контента и предоставлении отчетов об использовании.

Соответственно, помимо написания кода API метода, в обязанности бэкенд разработчика входит написание и поддержание в актуальном состоянии API-документации.

Поскольку написание документации занимает 1/3 всей разработки, было принято решение автоматизировать данный процесс.

Поскольку данный процесс занимает 1/3 всей разработки, было принято решение автоматизировать данный процесс.

Написание подробной документации как пользоваться API методами входит в обязанности любого бэкенд разработчика и/или технического писателя.

Соответственно написание и поддержание в актуальном состоянии документации как пользоваться API методами для всех клиентов входит в обязанности любого бэкенд разработчика и/или технического писателя.

Однако данный процесс ...

## **1. ОБЗОРНАЯ ЧАСТЬ**

### **1.1. Описание предметной области**

### **1.1.1. Описание серверной архитектуры**

Серверная часть проекта Fonmix на разделена на микросервисы.

Микросервисная архитектура – вариант сервис-ориентированной архитектуры программного обеспечения, направленный на взаимодействие насколько это возможно небольших, слабо связанных и легко изменяемых модулей – микросервисов.

Основными микросервисами являются

### **1.1.2. Описание модели внедрения программных продуктов Agile Scrum**

Текст

### **1.1.3. Схема ручного сопровождения API-документации**

Текст

## **1.2. Исследование существующих технологий**

### **1.2.1. Перечень функций, подлежащих автоматизации**

Текст

### **1.2.2. Выбор и обоснование критериев качества**

Текст

### **1.2.3. Анализ аналогов и прототипов**

Текст

#### **1.2.4. Сравнение аналогов и прототипов**

Текст

## **2. РАСЧЕТНО-КОНСТРУКТОРСКАЯ ЧАСТЬ**

### **2.1. Определение требований к системе**

Текст

### **2.2. Разработка структуры автоматизированной системы**

Текст

### **2.3. Разработка структуры интерфейса взаимодействия пользователя с системой**

Текст

### **2.4. Разработка алгоритмов программных модулей**

Текст

### **2.5. Разработка плана проведения тестирования**

Текст

### **3. ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ**

#### **3.1. Реализация разработанных алгоритмов**

Текст

#### **3.2. Тестирование и отладка системы**

Текст

#### **3.3. Руководство пользователя**

Текст

### **ЗАКЛЮЧЕНИЕ**

Что в итоге получилось.

### **СПИСОК ЛИТЕРАТУРЫ**

1. Python Documentation – [Электронный ресурс]:  
<https://docs.python.org/3/>  
(Дата обращения: 28.09.2020)
2. Bootstrap Documentation – [Электронный ресурс]:  
<https://getbootstrap.com/docs/3.3/> (Дата обращения: 28.09.2020)
3. PostgreSQL Database Documentation – [Электронный ресурс]:  
<https://www.postgresql.org/docs/> (Дата обращения: 28.09.2020)
4. Скотт Б., Нейл Т. Проектирование веб-интерфейсов. – СПб.: Символ-Плюс, 2010. – 352 с.