Доклад

По Интернет програмиране и Софтуерно инженерство

Тема: REST API

1. Да се опише какво е rest, api, RESTful api, endpoint
2. Принципи на изграждане на rest аpi и примери

1.Да се опише какво е rest, api, RESTful api, endpoint

REST - Това е архитектурен стил за взаимодействие между компоненти на разпределена система в компютърна мрежа. Просто казано, REST определя стила за взаимодействие (обмен на данни) между различни компоненти на системата, всеки от които може да бъде физически разположен на различни места.

Този архитектурен стил е последователен набор от ограничения, спазвани при проектирането на разпределена система. Тези ограничения понякога се наричат ​​ръководни принципи на REST. Не са много, само 6. За тях ще говорим малко по-късно.

API - (от Application Programming Interfaces), са софтуерни градивни елементи, които позволяват на две приложения да комуникират помежду си. Винаги, когато използвате приложение като Facebook, изпращате директно съобщение или проверявате времето на вашия SmartPhone, използвате API.

API обикновено се състои от набор от рутинни процедури, протоколи и инструменти, използвани при изграждането на софтуерни приложения. Накратко, API определя как трябва да се интегрира софтуер и как този софтуер работи заедно. Освен това API се използват като компоненти на графичния потребителски интерфейс (GUI). Добрите API улесняват разработването на програма, като предоставят готови градивни елементи. След това програмист поставя градивните елементи заедно и захранва API-тата с данни от програмата. Той позволява на програмата да събира данните, които API-та връщат за по-нататъшна обработка.

RESTful API - интерфейс, който две компютърни системи използват за сигурен обмен на информация по интернет. Повечето бизнес приложения трябва да комуникират с други вътрешни приложения и приложения на трети страни, за да изпълняват различни задачи. Например, за да генерирате месечни фишове за заплати, вашата система за вътрешни сметки трябва да споделя данни с банковата система на вашия клиент, за да автоматизира фактурирането и да комуникира с вътрешно приложение за разписание.

RESTful API поддържат този обмен на информация, защото следват сигурни, надеждни и ефективни стандарти за софтуерна комуникация.

Предимствата на RESTful API – Мащабируемост, Гъвкавост, Независимост

Endpoint - Крайните точки са физически устройства, които се свързват и обменят информация с компютърна мрежа. Някои примери за крайни точки са мобилни устройства, настолни компютри, виртуални машини, вградени устройства и сървъри. Устройствата за Интернет на нещата – като камери, осветление, хладилници, системи за сигурност, интелигентни високоговорители и термостати – също са крайни точки. Когато едно устройство се свърже към мрежа, потокът от информация между, например, лаптоп и мрежа, прилича много на разговор между двама души по телефона.

2.

В C# може да изградите REST API, следвайки някои основни принципи на REST. Ето някои от тези принципи:

Ресурси (Resources):

* Идентифицирайте основните ресурси, с които ще работите във вашия REST API.
* Пример: Ако сте разработчик на книжарница, ресурсите могат да бъдат книги, автори и т.н.

URI (Uniform Resource Identifier):

* Използвайте URI за идентификация на ресурсите. URI трябва да бъдат уникални и описат ясно ресурса.
* Пример: /books, /authors.

HTTP Методи:

* Използвайте стандартните HTTP методи за взаимодействие с ресурсите. Най-често използваните са GET, POST, PUT, DELETE.
* Пример: GET за извличане на информация, POST за създаване на нов ресурс, PUT за актуализация и DELETE за изтриване.

Представяне (Representation):

* Изпращайте и получавайте данни в подходящ формат като JSON или XML.
* Пример: Върнете информация за книга във формат JSON.

Състояние (Stateless):

* Дизайнът трябва да бъде безсъстоятелен, което означава, че всички необходими информации за обработка на заявката трябва да бъдат включени в самата заявка, а не съхранявани на сървъра.

Кеширане:

* Използвайте кеширане, където е възможно, за оптимизация на изпълнението.

Сигурност:

* Защитавайте вашия REST API с подходяща аутентикация и авторизация.
* Използвайте HTTPS за криптиране на комуникацията.

Операции върху Ресурси:

* Избягвайте използването на операции в URI (например /getBook), предпочитайте използването на HTTP методи.

Източници:

<https://codegym.cc/bg/groups/posts/bg.286.pregled-na-rest-cast-1-kakvo-e-rest->

<https://bg.itpedia.nl/2018/11/02/wat-zijn-apis-application-programming-interface/#google_vignette>

<https://aws.amazon.com/what-is/restful-api/>

<https://www.microsoft.com/en-us/security/business/security-101/what-is-an-endpoint>

Изготвил: Валентин Кьосев 12 ,,а‘‘ клас