

Find Recipes on the Go: Mobile Application Using REST API

¹Ritika Kaushal, ²Prof. Suprava Patnaik

¹ Electronics and Telecommunications, Xavier Institute of Engineering, Mumbai, Maharashtra, India

² Electronics and Telecommunications, Xavier Institute of Engineering, Mumbai, Maharashtra, India

Abstract - REST APIs provided easy exchange of data over the internet using certain standards and protocols. This work presents the implementation of a mobile system with the primary objective to aid to an easy categorical search for recipes on the go. This mobile application system was developed using Android Studio. Stateless software architecture was used to make this system light and quick in comparison to such an application with a locally stored database. This paper summarizes the implementation details and presents results of the system in practice. An application prototype was also created demonstrating the mentioned concept.

Keywords-Mobile Application, recipes, REST API, stateless architecture, android

1. Introduction

With the increase in the number of jobs that require more than 8-9 hours of working, the working class has migrated more towards unhealthy eating alternatives like ready to cook meals or merely junk food. This shift has led to adverse health effects such as the prevalence of overweight and obesity has more than doubled during the last couple of decades in contemporary Western societies. As of now, the statistics being, around 39% of adults are classified as overweight and 13% as obese, in accordance to a WHO study of 2014[6]. One of the reasons for obesity is the unavailability of healthy options. There are some applications on the play store with over 500,000 recipes to choose from and have millions of downloads on the Google play-store, although these options make the application extremely big in size. The size of such applications is as high as 74mb. One of the other factors that also needs to be taken into consideration is the speed of the created application, which seems to be have been compromised in many such big name applications. Hence, to provide multiple healthy as well as snack like recipes on the go, a light, extremely interactive and easy application came to life. The idea behind it being an application that has something for everyone be it a working parent, a student staying away from their family or even a professional chef. This application was not only aimed to be extremely easy to use and interactive but was also fast enough with multiple recipe options.

The target audience for this application includes mainly the working class and people who are always on the go and might not have a lot of time on their hands to look through

food and recipe related books and websites. As mentioned before this mobile application has something for every single user. The division of recipes by country, type of food, full/latest meals and ingredients makes the applications extremely easy to navigate.

2. Literature Review

2.1 Mobile applications versus websites

Multiple web applications are widely available with a primary functionality of finding recipes simply by their name or with a functionality of searching recipes around specific ingredients. In comparison to the large number of available websites, there aren't a lot of mobile applications that aid the easy finding of new recipes. Mobile apps are usually 1.5 times faster than web applications, one of the reasons being that mobile applications store data locally, unlike the websites that use web servers [3]. This isn't a clear indication that mobile sites are better, this fact just means that mobile apps provide greater business value in this particular aspect. Another major advantage of mobile applications its ability to show a show a concise display of respective information. The screen isn't overpopulated with advertisements and unnecessary browsing options and data. This is hence an implementation to create a mobile interface to provide the same or more functionalities than the web applications in a faster, robust and cheaper format. Hence the implementation of this REST API based mobile app to add to the scarcely available faster alternatives.

2.2 Database call versus REST API

Below enlisted advantages provide a first-hand testimony for REST API being the best decision over database in this case.

Security: Database access isn't granted to anyone but the web service. This proves to be extremely useful when the data to be maintained is huge.

DB load reduction: Web service can easily cache the data it retrieves from the database. This property aids in fast responses.

Scalability: REST stands for Representational State Transfer; by stateless it means that the server does not store any state about the client session on the server side. The *client session* is stored on the client and not the server and is passed around from the client to every place it needs to be.

The server is stateless means that every server can service any client at any time which means that there is no *session affinity* or *sticky sessions*. The relevant session information is stored on the client and passed to the server as needed.

The client can transfer the state around instead of having the server store it. This is the way to scale to millions of concurrent users.

Encapsulation: You can change underlying DB implementation without impacting service users.

Complexity: For a database query one needs to connect the application to the database and install drivers if necessary. On the contrary, the REST API does not care about the app that is requesting the data, the requesting app can be a web app, a mobile app or a desktop app; A simple request can be drafted and sent to the API just with the help of a simple old URL. The API parses it, queries the database on your behalf (which eliminates the need of drivers and SDKs) and returns a structured response. It does not matter to the REST API as long as the request is a valid-request. API is just an additional layer of abstraction on the DB Query, which prevents the users from caring about the kind of DB used. [4]

Independence: Due to the separation between client and server, the protocol makes it easy for developments across the various areas of a project to take place independently. In addition, the REST API adapts at all times to the working syntax and platform. This offers the opportunity to try several environments while developing. [5]

Flexibility and Portability: With the indispensable requirement for data from one of the requests to be properly sent, it is possible to perform a migration from one server to another or carry out changes on the database at any time.

Front and back can therefore be hosted on different servers, which is a significant management advantage. [5]

2.3 What is REST APIs?

a. What are WEB services?

A web service is a collection of standards and protocols that applications and systems use for exchanging data over the internet [7]. WEB services aren't confined to one particular language and also have an added advantage of OS independency.

b. What is REST?

REST stands for representational state transfer. It's a stateless software architecture that governs the behavior of client and server based applications.

c. When can an API be called REST?

A REST API has the below given key features:

Client-server architecture: The front end and the back-end of the service are independent of each other.

Stateless: No data should be stored on the server during the processing of the request transfer. The state of the session should be saved at the client's end [7].

Cacheable: The client should have the ability to store responses in a cache. This greatly improves the performance of the API. [7]

d. Formal Definitions of a REST API.

A REST API is a web service implemented using HTTP protocol as well as the principles of REST. It is a mere collection of resources that employ HTTP methods i.e. GET, PUT, POST, DELETE. The collection of the resources is then represented in a standardized form that can be any valid Internet media type, provided that it is a valid hypertext standard. [7]

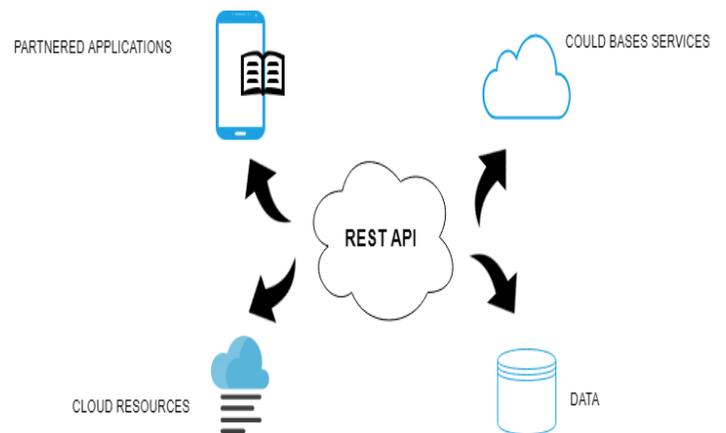


Fig. 1 Versatility of REST APIs

e. Real world examples of best use of REST APIs:
 Many popular websites make use of this powerful tool of REST APIs for the development of their applications, mainly to provide the application features, such as enhanced performance, scalability, simplicity, portability, and reliability. The simplicity in functioning, maintaining and development is a major advantage of using REST APIs.

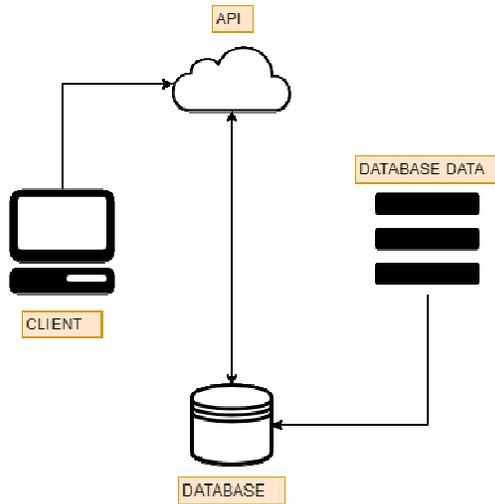


Fig. 2 Simple communication between client and API using HTTP protocol

2.4 Android Studio:

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ’s powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps. Some services offered by Studio include [8]:

- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Instant Run to push changes to your running app without building a new APK
- Code templates and GitHub integration to help you build common app features and import sample code
- Extensive testing tools and frameworks
- Lint tools to catch performance, usability, version compatibility, and other problems

- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine
- A flexible Gradle-based build system

The Android Studio version of 3.1.3 was used to develop the mentioned mobile application.

3. Architecture

3.1 Proposed Architecture:

The mobile application is android based and works using REST API. The proposed architecture has been shown below in the form of a block diagram.

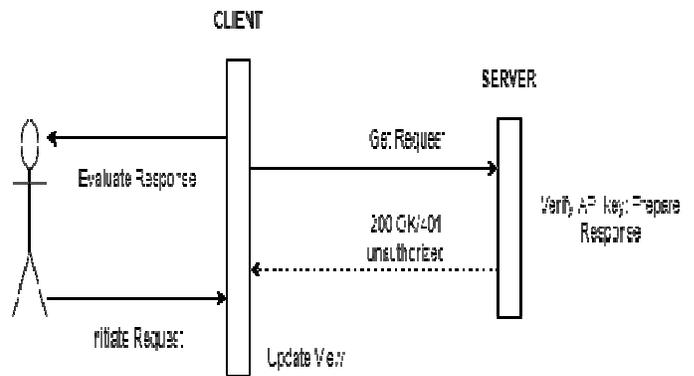


Fig. 3 Block Diagram of the proposed system

The request is initiated by the user and the client/application hits a get request to the server, the API key is verified on the server side, the response to the query is prepared and set back to the client and the view/UI of the application gets updated. No response to a particular query/get request is handled as shown in the block diagram.

4. System Implementation and Results

4.1 APIs used and received responses

Innumerable options have been available that provide hundreds and thousands of recipes in their APIs. The development of the above mentioned mobile application was done using a REST API that is freely available and easy to access. The used API was by “TheMealDB.com”. They offer free JSON APIs with set responses and the data can be filtered and used to meet the mobile applications needs. A snippet from the response of one of the many

available API methods is shown below. The attained response as shown is in the raw form. This response can be filtered and used to suit the needs of the developed UI.



Fig. 4 Response of one API from 'Mealdb.com'



Fig. 5 Response of one API from 'Mealdb.com'

The snippets shown in the figures above have been acquired using the APIs as mentioned below.

- (1) <https://www.themealdb.com/api/json/v1/1/random.php>
- (2) <https://www.themealdb.com/api/json/v1/1/list.php?a=list>

The first API method helps us acquire a random recipe from the database. Its result is a new response every single time the API is hit. The second API method helps us acquire a list of areas/countries that are stored in the database, to find related recipes.

4.2 Mobile Application: User Interface

The mobile application developed has an extremely interactive UI. Various categories such as ingredients, countries, food categories etc. were utilized to make the user experience as quick and easy as possible. These categorical divisions help the user not only find the desired recipes but also all the available ones that are best suited for the user's available ingredients or cuisine choice etc.

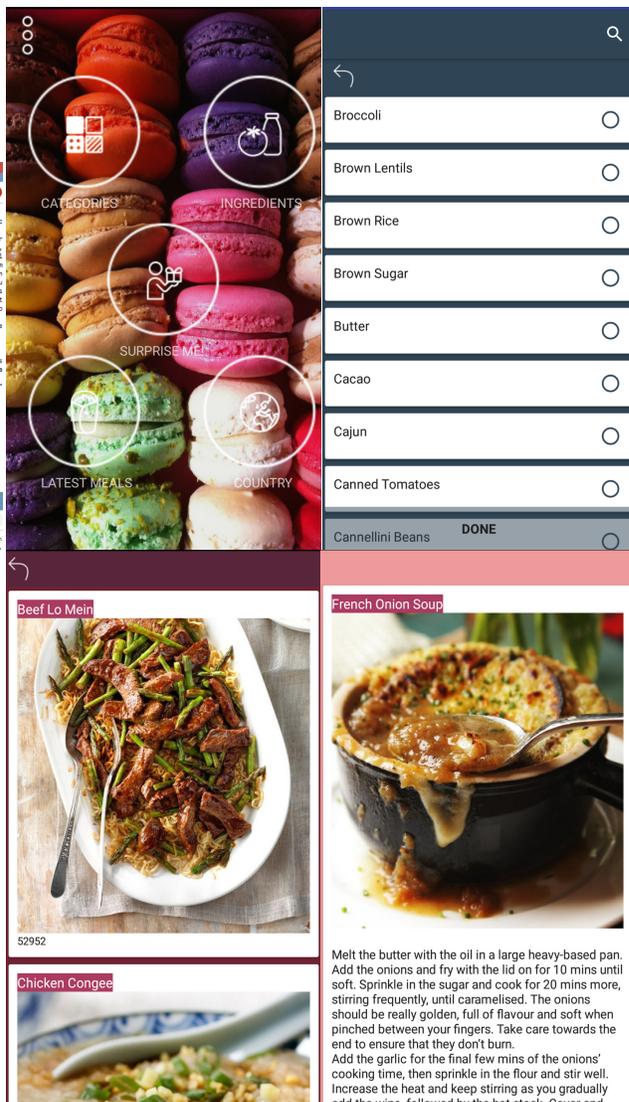


Fig. 6 UI of the android based mobile application

5. Conclusion and Future Scope

The application has been developed using Android studio and REST APIs. The application is extremely suitable for everyone always on the go. It is extremely beneficial in making healthy food choices. The application developed is primarily to ease out the trouble of finding a perfect recipe for a 'Sunday brunch' or a 'Tuesday dinner'. The application provides a list of 500+ ingredients to choose from and hence can come handy for every meal that's to be cooked. The implementations in the feature would help eradicate the shortcomings of the application at the moment.

The addition of features such as download, mark and favorites could be added to make the application all the more efficient.

A bigger API that contains more recipes will increase the number of options available.

Categories such as 'Sunday Brunch' or 'Monday Blues' could be added to the list, this would contain direct recipes and the user would hence be able to access these in less than 2 clicks from the main screen.

A category curated by the users for the users would prove to make the app more user interactive and also help users share their recipes.

Another review section in the bottom of every recipe which would include pictures from the user tried version of the respective recipes.

Acknowledgments

Heartiest thanks to Mr.Zag from Mealdb.com for providing with all the needed support in context to the APIs for the development of this application. Extremely warm thanks to Dr.Suprava Patnaik for her constant support, guidance and encouragement to do better.

References

- [1] <https://www.mulesoft.com/resources/api/what-is-rest-api-design>
- [2] <https://www.sitepoint.com/building-simple-rest-api-mobile-applications/>
- [3] <https://vwo.com/blog/10-reasons-mobile-apps-are-better/>
- [4] <https://www.quora.com/Whats-the-difference-between-consuming-RESTful-api-and-querying-a-database-Both-of-them-returns-the-queried-response>
- [5] <https://www.chakray.com/en/advantages-of-rest-api/>
- [6] Naomi Kakoschke, Eva Kemps & Marika Tiggemann (2017): Differential effects of approach bias and eating style on unhealthy food consumption in overweight and normal weight women, *Psychology & Health*, DOI: 10.1080/08870446.2017.1327587
- [7] <https://dzone.com/articles/an-introduction-to-restful-apis>
- [8] <https://developer.android.com/studio/intro/>
- [9] <https://www.themealdb.com/api.php>