**PARTNER BILLS MOBILE APP**

A Project-II Report

Submitted in partial fulfillment of requirement of the

Degree of

**BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE & ENGINEERING**

BY

**AAYUSH AGRAWAL**

**EN18CS301003**

Under the Guidance of

**Mr. Ashish Kumawat**



**Department of Computer Science & Engineering**

**Faculty of Engineering**

**MEDI-CAPS UNIVERSITY, INDORE- 453331**

**JAN-MAY 2022**

**Report Approval**

The project work **“PARTNER BILLS MOBILE APP”** is hereby approved as a creditable study of an engineering/computer application subject carried out and presented in a manner satisfactory to warrant its acceptance as prerequisite for the Degree for which it has been submitted.

It is to be understood that by this approval the undersigned do not endorse or approved any statement made, opinion expressed, or conclusion drawn there in; but approve the “Project Report” only for the purpose for which it has been submitted.

Internal Examiner

Name: Mr. Ashish Kumawat

Professor

Computer Science Engineering

Medi-Caps University, Indore

External Examiner

Name:

Designation

Affiliation

**Declaration**

I hereby declare that the project entitled **“PARTNER BILLS MOBILE APP”** submittedin partial fulfillment for the award of the degree of Bachelor of Technology in Computer Science & Engineering completed under the supervision of **Mr. Ashish Kumawat, Computer Science,** Faculty of Engineering, Medi-Caps University Indore is an authentic work.

Further, I/we declare that the content of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for the award of any degree or diploma.

**Aayush Agrawal**

**Date: 10-05-2022**

**Certificate**

We, **Mr. Arunkumar K S** , **Mr. Ashish Kumawat** certify that the project entitled **“PARTNER BILLS MOBILE APP”** submittedin partial fulfillment for the award of the degree of Bachelor of Technology by **Aayush Agrawal** istherecordcarried out by him/them under my/our guidance and that the work has not formed the basis of award of any other degree elsewhere.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Mr. Ashish Kumawat Sir **(**Professor**)**

Computer Science & Engineering

Medi-Caps University, Indore

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Mr. Arunkumar KS (Manager)

Subex, Bangalore

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

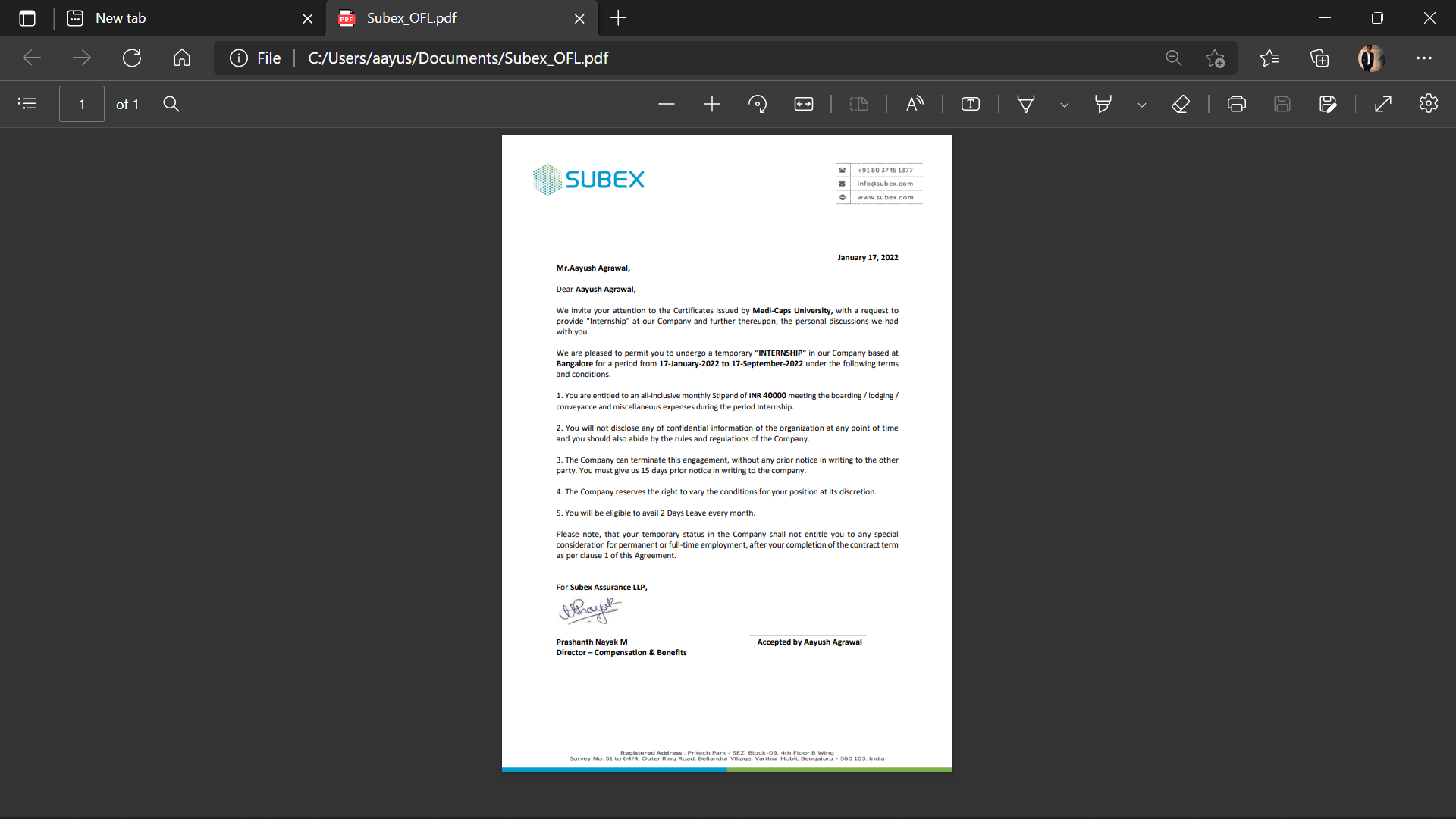
Dr. Pramod S. Nair

Head of the Department

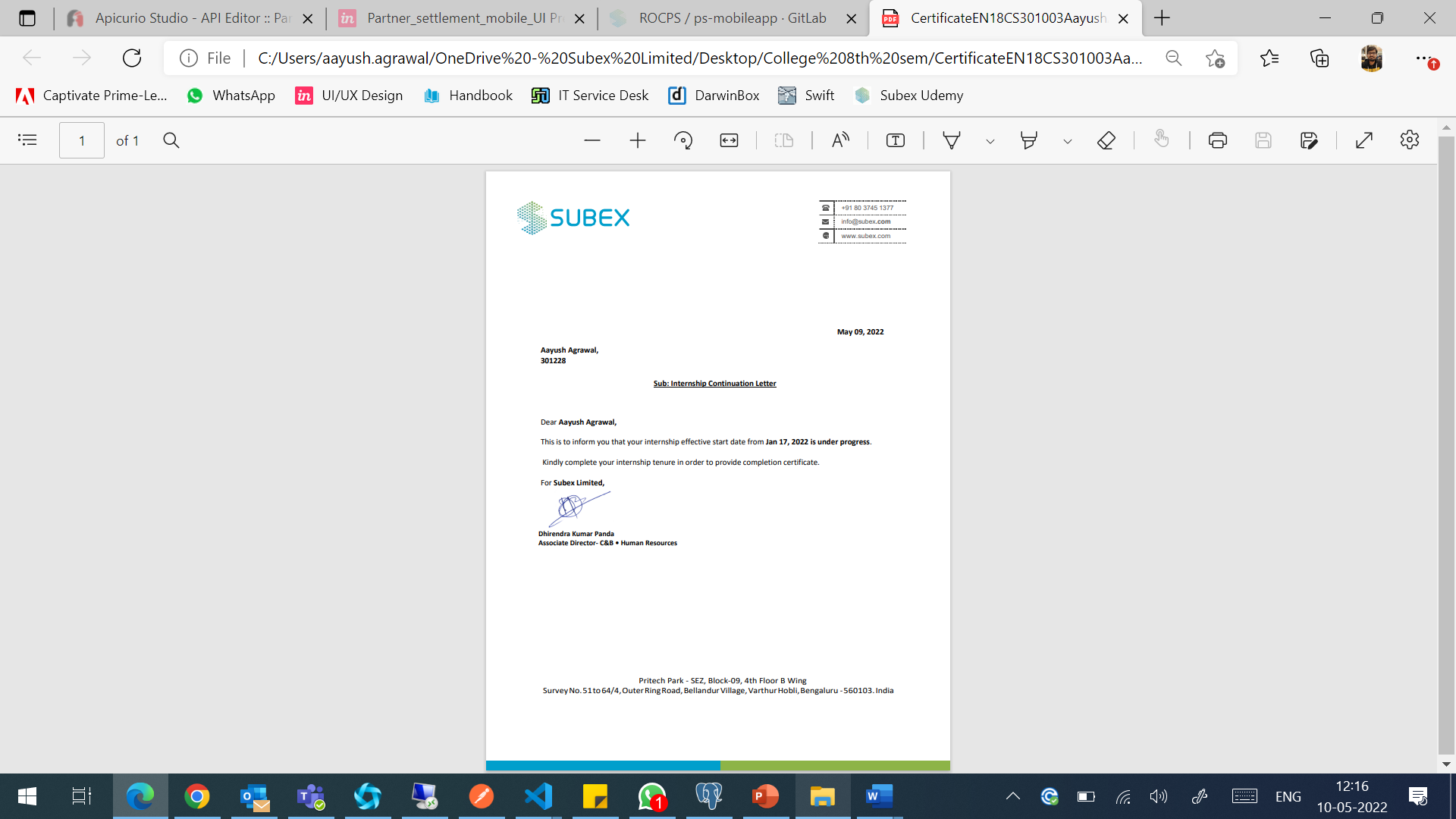
Computer Science & Engineering

Medi-Caps University, Indore

**Offer Letter of the Project work-II/Internship**



**Continuation Letter**

****

**Acknowledgements**

I would like to express my deepest gratitude to Honourable Chancellor, **Shri R C Mittal,** who has provided me with every facility to successfully carry out this project, and my profound indebtedness to **Prof. (Dr.) D. K. Patnaik,** Vice Chancellor, Medi-Caps University, whose unfailing support and enthusiasm has always boosted up my morale. I also thank **Prof. (Dr.) Suresh Jain,** Dean, Faculty of Engineering, Medi-Caps University, for giving me a chance to work on this project. I would also like to thank my Head of the Department **Dr. Pramod Nair** for his continuous encouragement for betterment of the project.

It is their help and support, due to which we became able to complete the design and technical report

Without their support this report would not have been possible.

**Aayush Agrawal**

B.Tech. IV Year

Department of Computer Science & Engineering

Faculty of Engineering

Medi-Caps University, Indore

**Executive Summary**

1. Introduction

a. Large amount of uncertainty involved in choosing to adopt a new technology

b. Based off the fact that we all are more interested in stability than we are in the uncertainty created by new technologies

2. Reasons for Adoption of New Technology

a. New technology adoption is often necessary for the survival of a business firm.

b. Driven by industry-level changes that create a new technological standard

c. Firm must adopt the technology in order to avoid developing a competitive disadvantage in their industry.

d.Adopted in order to give the firm a competitive advantage, not just to avoid competitive disadvantages.

e. Occurs when a firm adopts a new technology ahead of the rest of their industry

f. Could result in wasted capital, if the technology does not give the firm advantage.

g. Gain a competitive advantage if the new technology allows the firm to achieve one or more business objective, such as operational excellence or customer/supplier intimacy.

h. Firms keep these scenarios in mind when decided to adopt a technology such as mobile applications

3. Need for Mobile Application Adoption

a. The firm must ask themselves if this investment is worthwhile to them or if it would simply result in an unnecessary waste of capital

b. Evident that the use of mobile applications for business is becoming more and more necessary for the survival of the firm

c. Different industries require different levels of integration for mobile applications

d. In the near future, it will be extremely necessary for all firms in all industries to adopt this new form of technology

e. Firms must now begin thinking about how and when they will integrate mobile applications into their business processes in order to ensure profits today and survival tomorrow

4. Adoption Decisions

a. Switching to a new technology is not a simple task, especially with a technology as diverse as mobile applications.

b. Many different decisions that need to be made before a firm can begin to take advantage the multiple possible uses of mobile applications

5. Adoption Process

a. Current and future requirements of the firm

i. The firm must understand the current and future needs of their organization.

ii. Firm decides how they want their business to run in five or seven years and integrate mobile applications into this future vision of the cooperation (Gold 1).

b. Firms must evaluate the assets that they currently have available for their use

i. These assets can include existing devices, network access and application development tools.

ii. Firms develop realistic strategy, not just an unrealistic vision of what they wish to achieve but would not be able to achieve

c. Firms must assess the overall capabilities and skills of the organization

i. This includes decisions about the skills of their workers and IT support staff.

ii. Decide which use mobile applications would be possible to implement with the human capital that the firm currently has at it’s disposal.

d. Firm must get support from the management and users

i. If management does not support the adoption of this new technology or users are sceptical about using mobile applications, it may not be a wise decision for the firm to implement such technologies (Gold 1)

**Table of Contents**

|  |  |  |
| --- | --- | --- |
|  |  | **Page No.** |
|  | **Title Page** | i |
|  | **Report Approval** | ii |
|  | **Declaration** | iii |
|  | **Certificate** | iv |
|  | **Offer Letter of the internship** | v |
|  | **Continuation Letter** | vi |
|  | **Acknowledgements** | vii |
|  | **Executive Summary** | viii |
|  | **Table of Contents** | x |
|  | **List of Figures** | xii |
|  | **List of Tables** | xiv |
|  | **Abbreviations** | xiv |
|  |  |  |
| **Chapter 1** | **Introduction** | 1 |
|  | 1.1 Introduction | 1 |
|  | 1.2 Literature Review | 2 |
|  | 1.3 Objectives | 3 |
|  | 1.4 Significance | 3 |
|  | 1.5 Research Design | 4 |
| **Chapter 2** | **Setup & Procedure** | 8 |
|  | 2.1 Experimental Set-up | 8 |
|  | 2.2 Procedures Adopted | 9 |
| **Chapter 3** | **Main Chapter** | 11 |
|  | 3.1 Implementation | 11 |
|  | 3.2 Testing | 12 |
| **Chapter 4** | **Main Chapter** | 15 |
|  | 4.1 Tools & Technologies Used | 15 |
|  | 4.1.1 Frontend Technology Stack | 15 |
|  | 4.1.2 Backend Technology Stack | 32 |
|  | 4.1.3 Other Tools | 35 |
| **Chapter 5** | **Results & Discussions** | 39 |
| **Chapter 6** | **Summary & Discussions** | 40 |
| **Chapter 7** | **Future Scope** | 41 |
|  | **References** | 42 |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Title** | **Page No.** |
| Figure 1.1 | Front end architecture | 4 |
| Figure 1.2 | Backend architecture | 5 |
| Figure 1.3 | Application workflow | 5 |
| Figure 1.4 | JWT login authentication flowchart | 6 |
| Figure 1.5 | Sequence Diagram | 7 |
| Figure 1.6 | Use case diagram | 7 |
| Figure 2.1 | Agile Methodology | 9 |
| Figure 4.1 | Basic HTML tags | 15 |
| Figure 4.2 | CSS | 16 |
| Figure 4.3 | Features of typescript | 17 |
| Figure 4.4 | Typescript compilation | 18 |
| Figure 4.5 | Significance of angular | 19 |
| Figure 4.6 | DOM | 20 |
| Figure 4.7 | Typescript in angular | 21 |
| Figure 4.8 | Angular architecture | 22 |
| Figure 4.9 | Advantages of angular | 24 |
| Figure 4.10 | Limitations of angular | 25 |
| Figure 4.11 | Material UI | 26 |
| Figure 4.12 | Capacitor | 27 |
| Figure 4.13 | Angular to mobile app | 27 |
| Figure 4.14 | Angular default app | 28 |
| Figure 4.15 | Package.json file | 29 |
| Figure 4.16 | angular.json file | 31 |
| Figure 4.17 | Angular mobile app | 32 |
| Figure 4.18 | Spring boot architecture | 34 |
| Figure 4.19 | PostgreSQL | 34 |
| Figure 4.20 | DB Basics | 35 |
| Figure 4.21 | Features of APICURIO | 36 |
| Figure 4.22 | API Structure | 36 |
| Figure 4.23 | First API-GET Method | 37 |
| Figure 4.24 | First API-Response Method | 37 |
| Figure 4.25 | Second API-Details | 37 |
| Figure 4.26 | Second GET-Method | 38 |
| Figure 4.27 | Second API-Response | 38 |
| Figure 4.28 | Third API-POST Method | 38 |
| Figure 4.29 | Third API-Request and Response | 38 |

**List of Tables**

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Title** | **Page No.** |
| Table 3.1 | Unit Testing of login page buttons | 12 |
| Table 3.2 | Unit Testing of Menu Bar | 12 |
| Table 3.3 | Unit Testing of Side Nav Bar (Menu Bar) | 13 |
| Table 3.4 | Unit Testing of the search bar | 13 |
| Table 3.5 | Unit Testing of Filters | 14 |
| Table 3.6 | Unit Testing of the bills display area | 14 |

**Abbreviations**

Apps - Applications

HTML-Hyper Text Markup Language

XML – Extensible Markup Language

UI – User Interface

Js – JavaScript

**Chapter 1**

* 1. **Introduction**

Smartphone users have increased after Jio launched in India. With the increase in users of smartphones there comes the demand for mobile application. Mobile application for android should be unique and understandable by users.

The Scope of android app developers has increased and will increase in the coming years in India. Big companies like Samsung, Vivo, One Plus are launching mobile phones with a new update every year, that has increased the demand for android. There will be no loss in Android App Development in the coming years. Indian app developers are cost effective. Android & Angular is an open-source software stack that includes the operating system and middleware.

Today people prefer to get data in their mobile. Every business owner like to develop their identity on android platform. Android mobile apps can be partially or fully personalized to suit your business needs. Android apps are developed with the Java coding language and the Android Software Development Kit, which runs on Mac, PC or Linux. Technological disruption in the app development industry is impacting almost every industry and region today.

Regardless of the industry, most companies today need a mobile app to stay in front of prospects and customers and do business with them. That’s why we can say we will provide our users a more seamless experience directly on their mobile screens.

* 1. **Literature Review**

Usage of frameworks is a crucial technology to be successful in developing business, Angular has exploded in usage because of its unique features. To give an as accurate knowledge as possible this essay contains a literature study of reliable references. The essay also contains an empirical study based on developers own perspective of Angular.

You have the opportunity to choose which version of Angular suits your company best. An article based on AngularJS: A Modern MVC Framework in JavaScript and discussed the various features of angularjs and also the comparative analysis of angularjs, backbone.js, and ember.js. they also give suggestion for why to choose angularjs. An article based on Understanding the Differences in versions of AngularJS vs. Angular 2 vs. Angular 4.it gives features, history, advantages. find out what has changed in Angular and why migrating to the latest version is a good idea.it gives an in-depth comparison so you can understand the differences and make an informed decision.

There is a survey study on angularjs performance, it gives the report the results of a survey with 95 professional developers about performance issues of single page web applications.it gives reports i.e. followed by developers to avoid performance problems (e.g., use of third-party or custom components), the general causes of performance problems in single page web applications, and the technical and specific causes of performance problems

Each part in this article is not intended to be a complete guide, but rather an overview of the basics to get you Up and running so you can get to know how to collect things together and understand what the framework has to offer.

* 1. **Objectives**

Various facilities and features have been planned to be incorporated in our mobile app to make it more seamless and feasible for users.

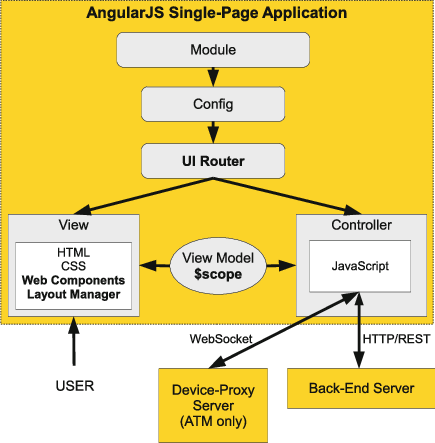
We will attempt to attain these objectives in our mobile app:

1. Specific to business users
2. Seamless user experience with enhanced UI
3. One code base for both android and IOS platforms
4. Optimized to use less storage and make less API calls to server
5. Same codebase can also be used to make progressive web and web applications
6. To provide important bill details directly on user’s mobile screen
7. More attractive UI.
   1. **Significance**

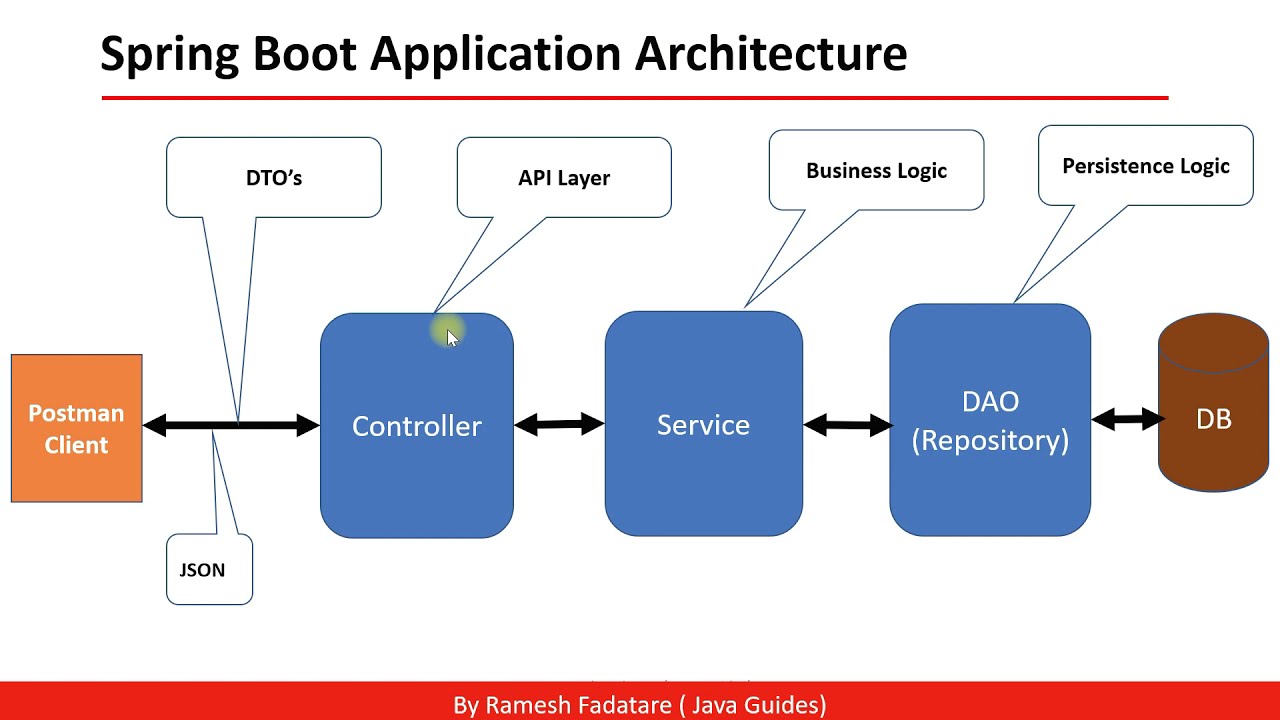
We must not overlook an application's usefulness and usability. They are so important and reliable that we cannot voluntarily ignore them. They are essential for businesses, but they also shape our daily lives.

Mobile apps **permit the users to have functional access to products, information, process, and services that they would demand in real-time**. Moreover, it enables the business to send notifications about changes in products and services or something new.

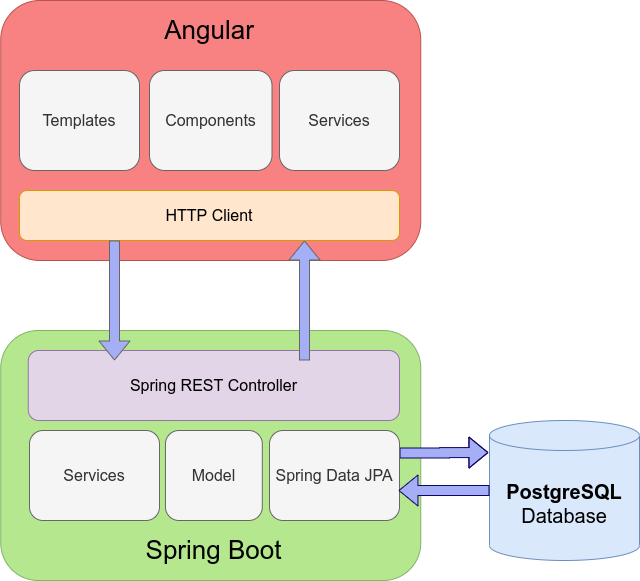
* 1. **Research Design**



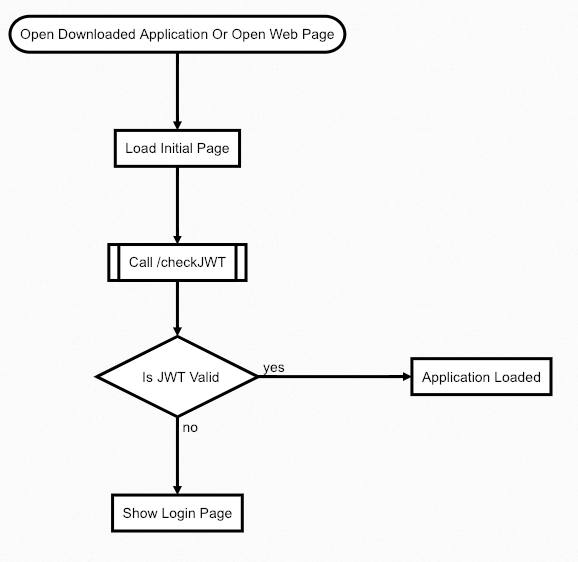
**Figure 1.1 Front end architecture**



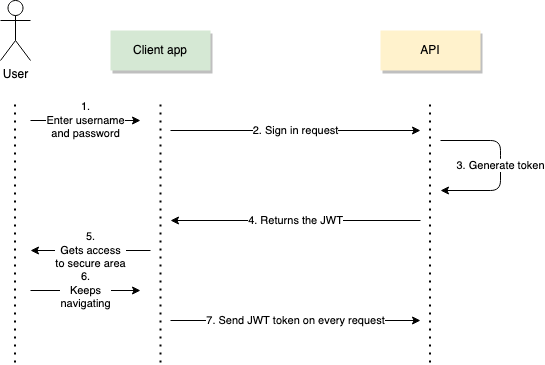
**Figure 1.2 Backend architecture**



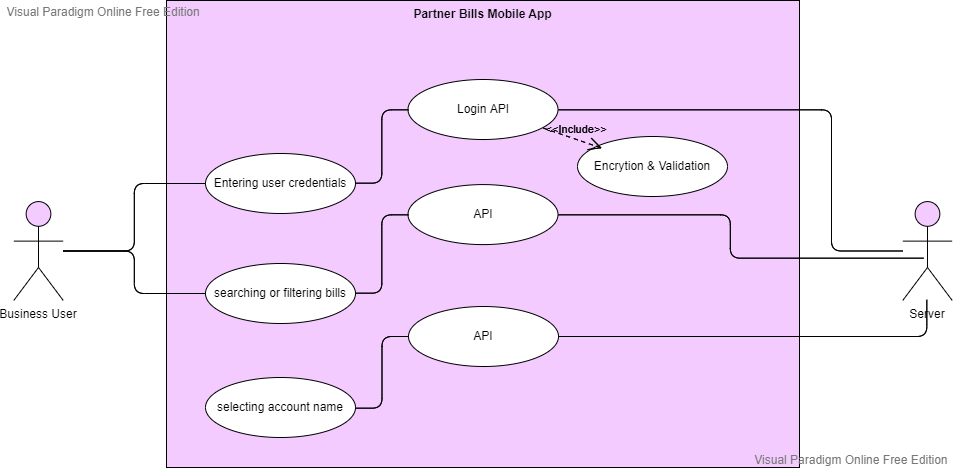
**Figure 1.3 Application workflow**



**Figure 1.4 JWT login authentication flowchart**



**Figure 1.5 Sequence Diagram**



**Figure 1.6 Use case diagram**

**Chapter 2**

**2.1 Experimental Setup**

**Feasibility:**

**Technical Feasibility**: To design a mobile app which meets the user requirements, a deep understand of JAVA language, spring boot and angular, typescript and other technical knowledge are needed. Based on the related technology information and resources for Angular and Spring boot on the market and equipped with technical personnel of technology and the spirit of willing to learn, the technology is feasible.

**Social Feasibility**: With the rapid development of the mobile phone and computer market, all kinds of mobile apps are widely circulated on the Internet. These resources seem ordinary but have gradually become an indispensable part of people life, which derived the development of all kinds of mobile phone/desktop apps. Powerful mobile application is a good thing, but a lot of functions are actually useless for most users. Aimed at these problems, developing mobile app which owns the features of simplified functions, meeting the needs of most users, less required memory and maximises the optimisation in performance.

**Software/Hardware Requirement:**

**Hardware Requirement:** Hardware is the backbone of any up and running software. Every system demands a good hardware to support its functions.  The hardware requirements are as follows:

a) Processor

Minimum: core i3 for better performance

b) Memory

Minimum: 4 GB RAM

Expected: 8 GB RAM for better performance

Minimum disk space: 10 GB free space

**Software Requirement:** Any android phones and for desktop windows 7 or above.

The required software of the developing environment

Windows 10 OS

Node Package Manager

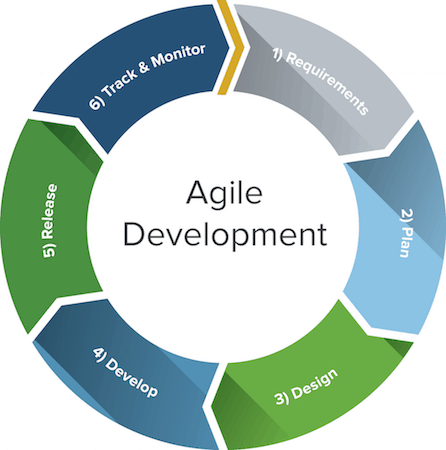
Java SDK

Angular CLI

Capacitor CLI

**2.2 Procedure Adopted**

In this project, the agile development cycle will be used to guide the development process. The reason for using agile methods is that desktop applications have a short software life cycle and rapidly changing technologies, so users will constantly change their requirement and needs in response to technological changes. Therefore, the agile development cycle is more suitable for desktop application development because of iterative and flexible, so it can adapt effectively to changing customers.



**Figure 2.1 Agile Methodology**

The agile development cycle contains 6 phrase which is requirement analysis, planning, design, implementation or development, testing, and deployment.

**Requirement** **analysis:** At this stage, we will review the requirements of business users. After the review, we will find out what current users need and collect their comments and suggestions for further analysis.

**Planning:** In the planning stage, we should first try to explore out the features that our mobile application can have. Next, we will eliminate the features that users feel no really useful or low cost-effective. Finally, each feature is prioritized and assigned to an iteration.

**Design:** The design stage is prepared according to the requirements of users. Since there are many details and problems encountered during development to be considered for each feature. Therefore, we will discuss and formulate solutions and test strategies to verify the product at this stage.

**Implementation** **or** **Development:** During the development phase, we will iteratively implement each of the features listed during the planning phase. At this stage, there will be many setbacks and obstacle, so the team needs to constantly overcome these obstacles. Moreover, we will prioritize the most important features and need to make intelligent trade-offs between the depth of completeness of a single feature and the breadth of implementation of multiple features.

**Testing**: In this stage, we will test the performance of each feature in order to check whether it meets the requirements of users. For example, we will test whether the application can be properly installed and run on a real device, and check whether any errors occur in the running process and each feature is up to standard.

**Deployment:** In this final phase, we will begin to deliver this application to the customer. For instance, we will deploy this mobile application. In addition, we will anticipate that users will encounter unpredictable problems when using the player in this process, so we will solve these problems in a future version.

**Chapter 3**

**3.1 Implementation**

The proposed application completed the debugging task during the testing phase, then it should enter the deployment phase. In the deployment phase, the developer needs to publish the application's installation package which is the “APK” file, to a platform such as Google Paly Store for users to download. However, due to the number of users are limit so far and the proposed application is not in the final public version, there are still many modules that should be improved and updated. Therefore, it will be uploaded to the relevant platform to promote to users after the final public version is released. In addition, users can execute the app in a non-network state.

Below are the steps to describe how a new user will execute the proposed application:

1. The user first install the APK and opens the application.

2. Users can log in by entering their credentials in the login screen.

3. In the next coming screen, bills will get loaded.

4. Users can scroll down to load next set of bills.

5. The user can use the search bar to search a particular account.

6. The user can use the side nav bar for navigation.

7. The user can logout by clicking on the logout button in side nav menu.

**3.2 Testing**

**Unit Testing 1: Login page buttons**

Test Objective: To test the functioning of login and password visibility buttons.

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Output** | **Actual Output** |
| Click “toggle” button in the password field | Password should be visible on one click then again invisible in next click | Pass |
| Click “login” button in the login page | On entering correct credentials, user should navigate to next page. And jwt token is generating. | Pass |

**Table 3.1 Unit Testing of login page buttons**

**Unit Testing 2: Menu Bar on bills page**

Test Objective: To test menu bar functions are working correctly.

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Output** | **Actual Output** |
| Click “side nav icon” in the menu bar | Side navigation menu should be opened | Pass |

**Table 3.2 Unit Testing of Menu Bar**

**Unit Testing 3: Side Nav Bar (Menu Bar)**

Test Objective: To test nav menu functions are working correctly.

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Output** | **Actual Output** |
| Click on menu item name in the Tools | It would navigate you to that page | Pass |
| Hover on menu items | That much portion becomes black | Pass |
| Click “logout” in the footer | Logging out of the application | Pass |

**Table 3.3 Unit Testing of Side Nav Bar (Menu Bar)**

**Unit Testing 4: The Search bar**

Test Objective: To test the functionality of search bar

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Output** | **Actual Output** |
| Entering an account name | It would display that account bill details on screen | Pass |

**Table 3.4 Unit Testing of the search bar**

**Unit Testing 5: The filter button & filters**

Test Objective: To test the functionalities of filters

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Output** | **Actual Output** |
| Click “filter icon” button | Filter menu should open | Pass |
| Click “on account dropdown” button | Accounts menu dropdown should open | Pass |
| Click “cancel button in accounts filed” button | Remove all selected accounts | Pass |
| Click “clear all” button in accounts filed | Remove all selected accounts | Pass |
| Click “date picker” button in filter menu | Should select the date | Pass |
| Click “status” radio button in filter menu | Should select the status | Pass |
| Click “Cancel” button | Remove all selected filters | Pass |
| Click “Apply” button | Apply all selected filters | Pass |
| Click “Close” button | Close the opened filter menu | Pass |
| Click outside the filter menu | Close the opened filter menu | Pass |

**Table 3.5 Unit Testing of Filters**

**Unit Testing 6: The bills display area**

Test Objective: To test the functionalities display area

|  |  |  |
| --- | --- | --- |
| **Input** | **Expected Output** | **Actual Output** |
| Scroll down in the area | Load the next set of records from the database | Pass |

**Table 3.6 Unit Testing of the bills display area**

**Chapter 4**

**4.1 Tools & Technologies Used**

**4.1.1 Frontend Technology Stack:**

**HTML:**

[HTML is a](https://www.educba.com/what-is-html/)markup language heavily utilized for creating web pages and web applications. HTML, when combined with JavaScript and CSS, has become a [milestone for web development](https://www.educba.com/web-development-interview-questions/). One of the useful aspects of HTML is, it can embed programs [written in a scripting language](https://www.educba.com/programming-languages-vs-scripting-languages/) like TypeScript, which is responsible for affecting the behaviour and content of web pages. CSS inclusion would affect the layout and appearance of the content. The basic building blocks of any HTML pages are HTML elements. A structured document can be created with the help of structural-semantic text like heading, paragraph, list, link, and other items. Browser indeed does not display the HTML tags but utilize them to interpret the content of the page. One needs to study various tags and then understand their behaviour.

HTML is used for a web document, internet navigation, etc. In this Uses of HTML article, we shall focus on top uses of HTML.

Diagram

Description automatically generated

**Figure 4.1 Basic HTML tags**

**CSS:**

**C**ascading **S**tyle **S**heets, fondly referred to as **CSS**, is a simply designed language intended to simplify the process of making web pages presentable. CSS allows you to apply styles to web pages. More importantly, CSS enables you to do this independent of the HTML that makes up each web page.  
CSS is easy to learn and understand, but it provides powerful control over the presentation of an HTML document.

**WHY CSS?**

* **CSS saves time:**You can write CSS once and reuse the same sheet in multiple HTML pages.
* **Easy Maintenance:**To make a global change simply change the style, and all elements in all the webpages will be updated automatically.
* **Search Engines:**CSS is considered a clean coding technique, which means search engines won’t have to struggle to “read” its content.
* **Superior styles to HTML:**CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
* **Offline Browsing:**CSS can store web applications locally with the help of an offline cache. Using this we can view offline websites.



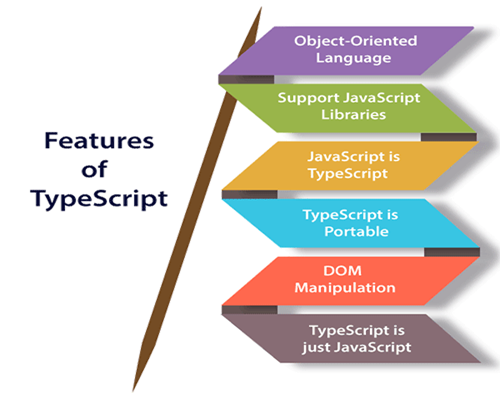
**Figure 4.2 CSS**

**TypeScript:**

TypeScript stands in an unusual relationship to JavaScript. TypeScript offers all of JavaScript’s features, and an additional layer on top of these: TypeScript’s type system.

For example, JavaScript provides language primitives like string and number, but it doesn’t check that you’ve consistently assigned these. TypeScript does.

This means that your existing working JavaScript code is also TypeScript code. The main benefit of TypeScript is that it can highlight unexpected behaviour in your code, lowering the chance of bugs.



**Figure 4.3 Features of typescript**

TypeScript cannot run directly on the browser. It needs a compiler to compile the file and generate it in JavaScript file, which can run directly on the browser. The TypeScript source file is in ".ts" extension. We can use any valid ".js" file by renaming it to ".ts" file. TypeScript uses TSC (TypeScript Compiler) compiler, which convert Typescript code (.ts file) to JavaScript (.js file)



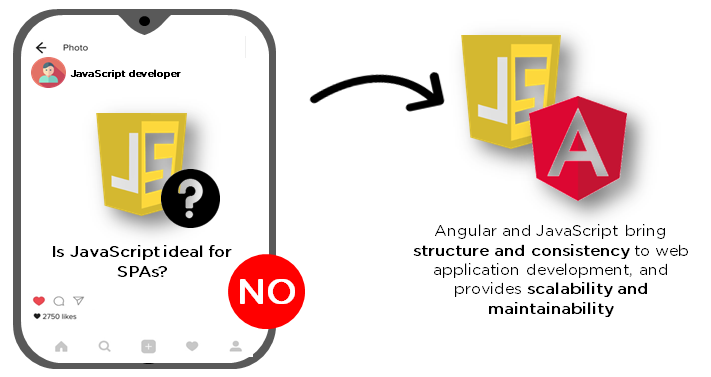
**Figure 4.4 Typescript compilation**

**Angular**

Angular is an open-source, [JavaScript](https://www.simplilearn.com/tutorials/javascript-tutorial/introduction-to-javascript) framework written in [TypeScript](https://www.simplilearn.com/tutorials/typescript-tutorial/typescript-interview-questions). Google maintains it, and its primary purpose is to develop single-page applications. As a framework, Angular has clear advantages while also providing a standard structure for developers to work with. It enables users to create large applications in a maintainable manner.

Frameworks in general boost web development efficiency and performance by providing a consistent structure so that developers don’t have to keep rebuilding code from scratch. Frameworks are time savers that offer developers a host of extra features that can be added to software without requiring extra effort.

**Why Angular?**



**Figure 4.5 Significance of angular**

[JavaScript](https://www.simplilearn.com/reasons-to-learn-javascript-article) is the most used client-side scripting language. It is written into [HTML](https://www.simplilearn.com/tutorials/html-tutorial/what-is-html) documents to enable interactions with web pages in many unique ways. As a relatively easy-to-learn language with pervasive support, it is well-suited to develop modern applications.

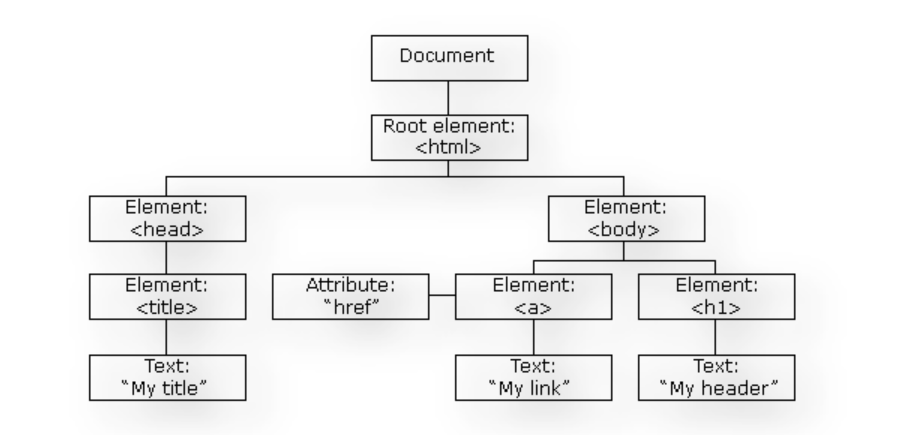
But is JavaScript ideal for developing single-page applications that require modularity, testability, and developer productivity? Perhaps not.

These days, we have a variety of frameworks and libraries designed to provide alternative solutions. With respect to front-end web development, Angular addresses many, if not all, of the issues developers face when using JavaScript on its own.

## Features of Angular

### **1. Document Object Model**

DOM (Document Object Model) treats an [XML](https://www.simplilearn.com/tutorials/programming-tutorial/what-is-xml) or HTML document as a tree structure in which each node represents a part of the document.

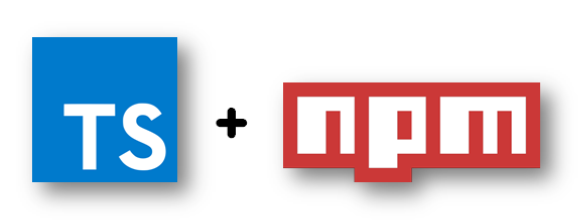


**Figure 4.6 DOM**

Angular uses regular DOM. Consider that ten updates are made on the same HTML page. Instead of updating the ones that were already updated, Angular will update the entire tree structure of HTML tags.

**2. TypeScript**

TypeScript defines a set of types to JavaScript, which helps users write JavaScript code that is easier to understand. All the TypeScript code compiles with JavaScript and can run smoothly on any platform. TypeScript is not compulsory for developing an Angular application. However, it is highly recommended as it offers better syntactic structure—while making the codebase easier to understand and maintain.



**Figure 4.7 Typescript in angular**

You can install TypeScript as an NPM package with the following command:

npm install -g typescript

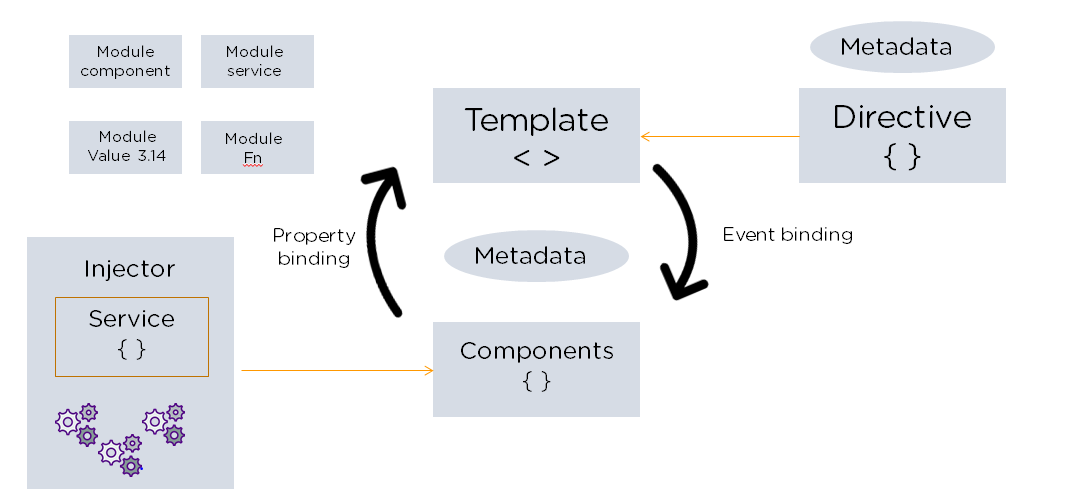
### **3. Data Binding**

[Data binding](https://www.simplilearn.com/tutorials/angular-tutorial/angular-data-binding) is a process that enables users to manipulate web page elements through a web browser. It employs dynamic HTML and does not require complex scripting or [programming](https://www.simplilearn.com/how-to-learn-programming-article). Data binding is used in web pages that include interactive components, such as calculators, tutorials, forums, and games. It also enables a better incremental display of a web page when pages contain a large amount of data.

Angular uses the two-way binding. The model state reflects any changes made in the corresponding UI elements. Conversely, the UI state reflects any changes in the model state. This feature enables the framework to connect the DOM to the model data through the controller.

**Angular Architecture**

Angular is a full-fledged model-view-controller (MVC) framework. It provides clear guidance on how the application should be structured and offers bi-directional data flow while providing real DOM.



**Figure 4.8 Angular architecture**

**The following are the eight building blocks of an Angular application:**

### **1. Modules**

An Angular app has a root module, named AppModule, which provides the bootstrap mechanism to launch the application.

### **2. Components**

Each [component](https://www.simplilearn.com/tutorials/angular-tutorial/angular-components) in the application defines a class that holds the application logic and data. A component generally defines a part of the user interface (UI).

### **3. Templates**

The Angular template combines the Angular markup with HTML to modify HTML elements before they are displayed. There are two types of data binding:

1. Event binding: Lets your app respond to user input in the target environment by updating your application data.
2. Property binding: Enables users to interpolate values that are computed from your application data into the HTML.

### **4. Metadata**

Metadata tells Angular how to process a class. It is used to decorate the class so that it can configure the expected behaviour of a class.

### **5. Services**

When you have data or logic that isn’t associated with the view but has to be shared across components, a [service](https://www.simplilearn.com/tutorials/angular-tutorial/angular-service) class is created. The class is always associated with the @Injectible decorator.

### **6. Dependency Injection**

[This feature](https://www.simplilearn.com/tutorials/angular-tutorial/angular-dependency-injection) lets you keep your component classes crisp and efficient. It does not fetch data from a server, validate the user input, or log directly to the console. Instead, it delegates such tasks to the services.

## Advantages of Angular



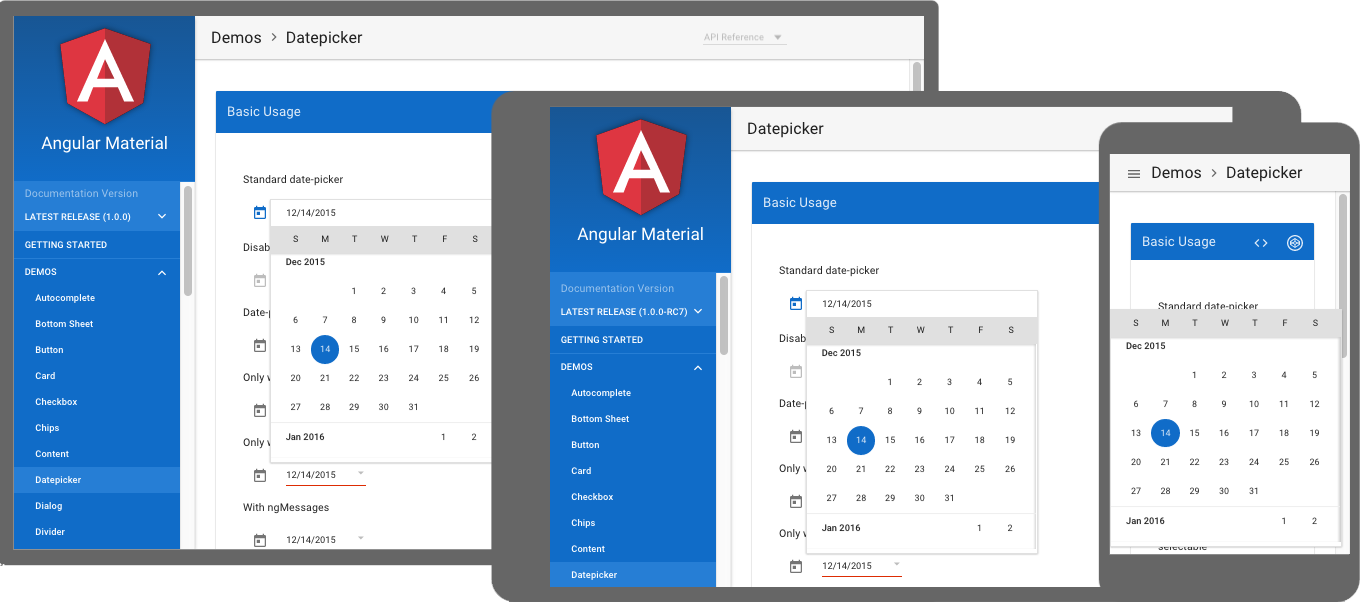
**Figure 4.9 Advantages of angular**

## Limitations of Angular



**Figure 4.10 Limitations of angular**

**Angular Material:**



**Figure 4.11 Material UI**

**Angular Material** is a UI library component developed by Google in 2014. It is specially designed for AngularJS developers. It helps to design the application in a structured manner. Its components help to construct attractive, consistent, and functional web pages and web applications. It is used to create a responsive and faster website.

## Features of Angular Material

* It is an In-built responsive design.
* Angular Material has standard CSS.
* The new version of UI Components which are **buttons, checkboxes**, and **text fields** is used to follow Angular Material Design concepts.
* It has specialized features such as **cards, toolbar, speed dial, side nav, swipe, side nav**, and more.
* It is a cross-platform browser, and it is used to create web components.

### **Responsive Design**

* Angular Material has a responsive design so that its website can fit in any size.
* The websites created by Material are compatible with **Android, iPhone, tablets**, and **laptops**.

### **Extensible**

* It is easy to add new **CSS** rules to overwrite the existing **CSS**
* It supports shadows and colors.
* The colors and shades are uniform in Angular Material.
* Angular Material is free to use.

**Capacitor:**

# Capacitor: Cross-platform Native Runtime for Web Apps

Capacitor is a cross-platform native runtime that makes it easy to build modern web apps that run natively on iOS, Android, and the Web. Representing the next evolution of Hybrid apps, Capacitor creates **Web Native apps**, providing a modern native container approach for teams who want to build web-first without sacrificing full access to native SDKs when they need it.

A picture containing sky, outdoor, person, blue

Description automatically generated

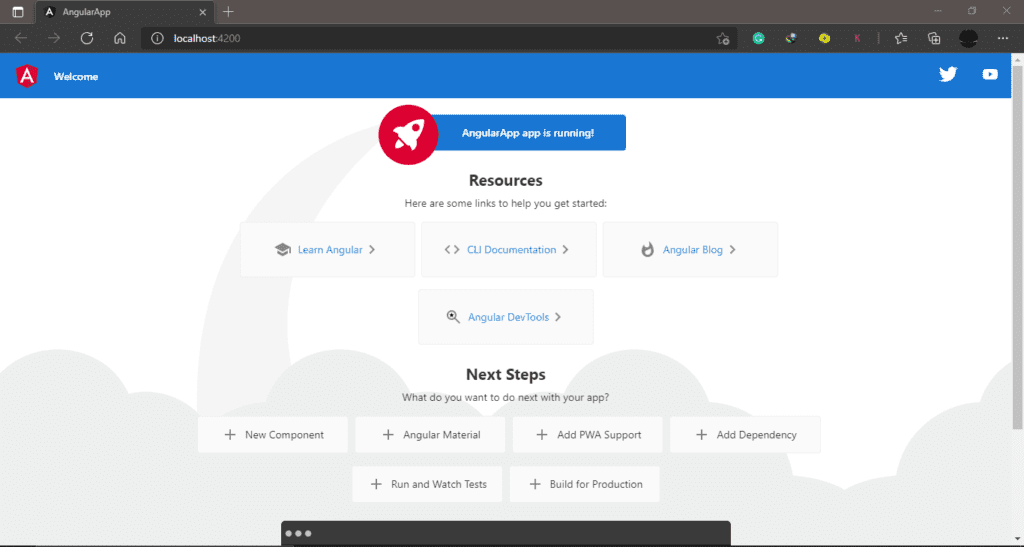
**Figure 4.12 Capacitor**

A picture containing text, electronics, black, computer

Description automatically generated

**Figure 4.13 Angular to mobile app**

# **How to convert Angular project to Android app**



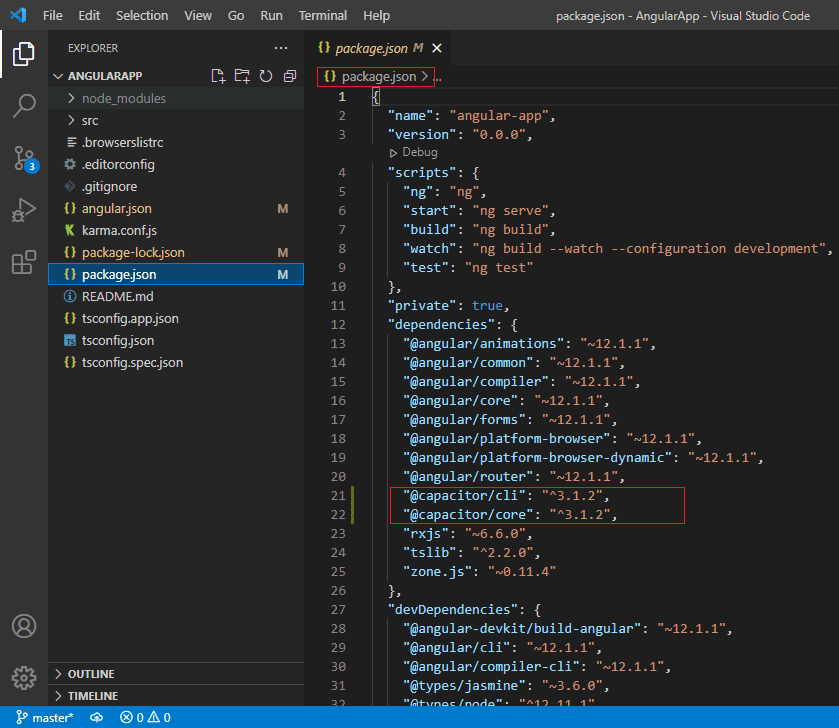
**Figure 4.14 Angular default app**

#### **Step 1: Install Capacitor in your project**

After you created, go to the project location and open CMD from there. If you have an existing Angular project open the CMD inside the project folder. Now we are going to install “Capacitor”. Run the below code in the CMD.

npm install --save @capacitor/core @capacitor/cli

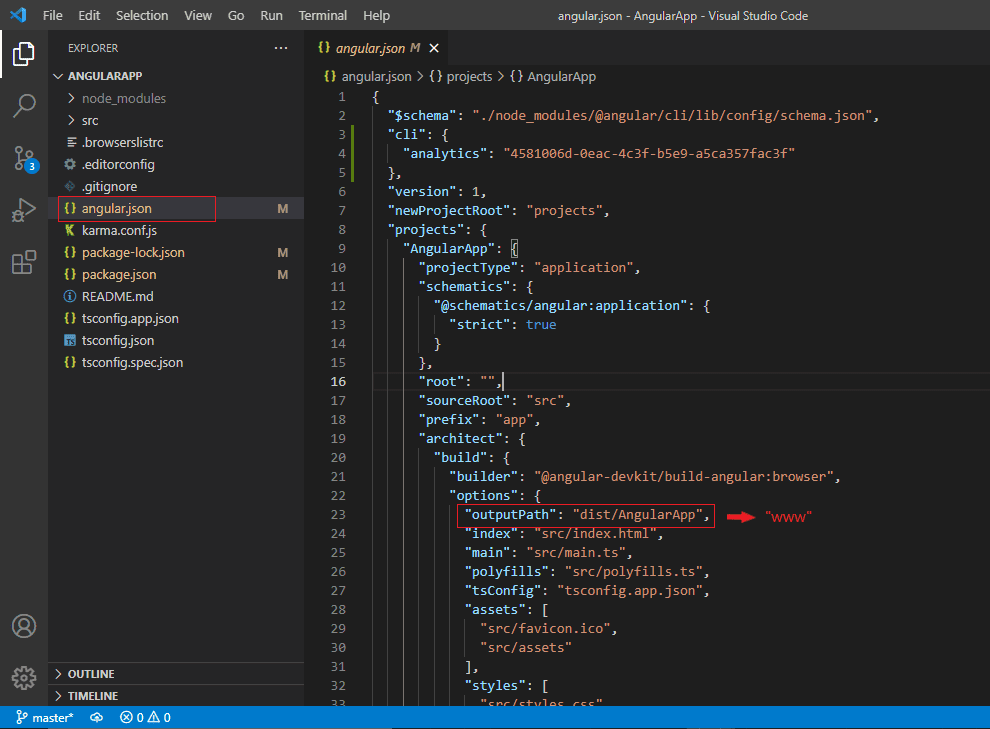
After successfully installed the capacitor, you see in the package.json file .



**Figure 4.15 Package.json file**

#### **Step 2: Change Output Path to “www”**

Then open the “**angular.json**” file and change **“outputPath”: “dist/AngularApp”** to **“outputPath”: “www”.**



**Figure 4.16 angular.json file**

#### **Step 3: Build Product**

Then save the project and run the below code to build the product. That compiles the Angular app into an output directory.

ng build

#### **Step 4: Initialize Capacitor in your project**

npx cap init

#### **Step 5: Add Android platform**

npx cap add android

After that, if you change anything in the web-based code, you can use the below code to copy those things to the native app.

npx cap copy android

Then run the below code to open the application in Android Studio.

npx cap open android

After the run “**npx cap open android** “, android studio automatically open and then you can run the app on the emulator or your phone using [USB debugging](https://maztars.com/usb-debugging-is-it-safe/). Then you can see the output like below in your phone or emulator.

Graphical user interface, application

Description automatically generated

**Figure 4.17 Angular mobile app**

### **Update converted Android app**

After you update your web application, you can run the below codes in CMD or code editor terminal.

ng b --prod

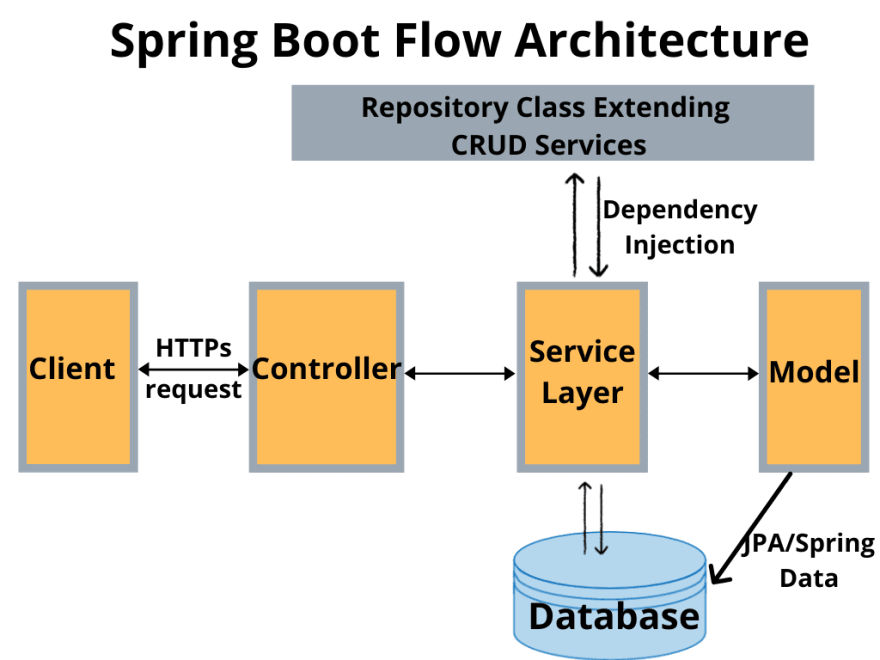
npx cap copy android

npx cap open android

**4.1.2 Backend Technology Stack:**

**Java & Spring Boot:**

Spring Boot is an open-source Java-based framework used to create a micro Service. It is developed by Pivotal Team and is used to build stand-alone and production ready spring applications. This chapter will give you an introduction to Spring Boot and familiarizes you with its basic concepts.



**Figure 4.18 Spring boot architecture**

## What is Micro Service?

Micro Service is an architecture that allows the developers to develop and deploy services independently. Each service running has its own process and this achieves the lightweight model to support business applications.

### **Advantages**

Micro services offer the following advantages to its developers −

* Easy deployment
* Simple scalability
* Compatible with Containers
* Minimum configuration
* Lesser production time

Spring Boot provides a good platform for Java developers to develop a stand-alone and production-grade spring application that you can **just run**. You can get started with minimum configurations without the need for an entire Spring configuration setup.

### **Advantages**

Spring Boot offers the following advantages to its developers −

* Easy to understand and develop spring applications
* Increases productivity
* Reduces the development time

### **Goals**

Spring Boot is designed with the following goals −

* To avoid complex XML configuration in Spring
* To develop a production ready Spring applications in an easier way
* To reduce the development time and run the application independently
* Offer an easier way of getting started with the application

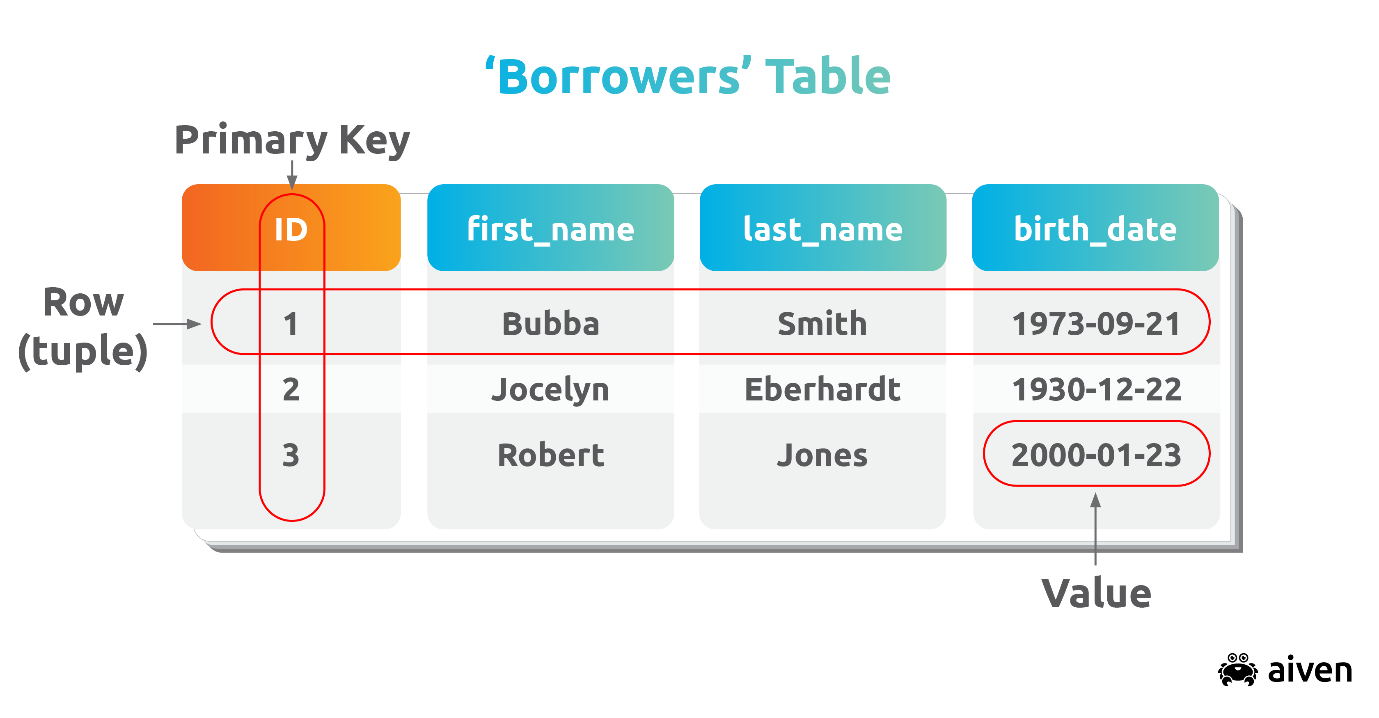
**Database:**

PostgreSQL is a flexible open-source object [relational database](https://en.wikipedia.org/wiki/Relational_database) management system able to cope in a huge variety of [use cases](https://aiven.io/blog/use-cases-for-postgresql), from single machines to data warehouses to web services with many concurrent users. PostgreSQL uses and extends SQL (hence the name) and is broadly extensible to a range of use cases beyond mere transactional data.



**Figure 4.19 PostgreSQL**

As a relational database, PostgreSQL stores data in tables (called relations) containing the tuples representing entities (such as documents and people) and relationships (such as authorship). Relations hold fixed-type attributes representing entity properties (such as a title) along with a primary key. Attribute types can be either atomic (such as integer, floating point, or Boolean) or structured (such as an array, nested JSON, or a procedure).



**Figure 4.20 DB Basics**

PostgreSQL supports transactions with ACID properties. This means transactions must support four attributes:

**Atomicity** — transactions are considered complete units; a transaction can either completely succeed or completely fail - in the case of failure, the database state is left unchanged.

**Consistency** — a database, between transactions, can only exist in a valid state; all data written to the database must adhere to extant constraints, triggers, cascades and related combinations.

**Isolation**, — a function of concurrency control, ensures that data is not corrupted by illegal or concurrent transactions — as transactions are treated as if they happened sequentially.

**Durability** — ensures that a transaction remains committed even when the system fails — typically, completed transactions are recorded

**4.1.3 Other Tools Used:**

**APICURIO:**

Apicurio Studio is a web-based API Design suite, primarily (but not exclusively) built to support design-first REST API development. It provides a way for users to collaboratively and visually design their APIs prior to (or concurrent with) the implementation.

A picture containing text

Description automatically generated**Figure 4.21 Features of APICURIO**

Graphical user interface, text, application, email

Description automatically generated**Figure 4.22 API Structure**

Graphical user interface, application

Description automatically generated

**Figure 4.23 First API-GET Method**

Graphical user interface, application

Description automatically generated**Figure 4.24 First API-Response Method**

Graphical user interface, text, application

Description automatically generated

**Figure 4.25 Second API-Details**

Graphical user interface, text, application, email

Description automatically generated**Figure 4.26 Second GET-Method**

Graphical user interface, application

Description automatically generated

**Figure 4.27 Second API-Response**

Graphical user interface, application

Description automatically generated

**Figure 4.28 Third API-POST Method**Graphical user interface, application

Description automatically generated**Figure 4.29 Third API-Request and Response**

**Chapter 5**

**Results & Discussions**

We have developed the mobile app using the angular as frontend and spring boot as backend framework and working properly till now.

Large amount of uncertainty involved in choosing to adopt a new technology, Reasons for Adoption of New Technology, New technology adoption is often necessary for the survival of a business firm. Firm must adopt the technology in order to avoid developing a competitive disadvantage in their industry.  
  
Occurs when a firm adopts a new technology ahead of the rest of their industry, could result in wasted capital, if the technology does not give the firm advantage. Gain a competitive advantage if the new technology allows the firm to achieve one or more business objective, such as operational excellence or customer/supplier intimacy. In the near future, it will be extremely necessary for all firms in all industries to adopt this new form of technology.

**Chapter 6**

**Summary & Discussions**

The features like scrolling with pagination, flexible with all screen sizes and other features are working fine.

Adoption Decisions, switching to a new technology is not a simple task, especially with a technology as diverse as mobile applications. Firm decides how they want their business to run in five or seven years and integrate mobile applications into this future vision of the cooperation.  
  
Decide which use mobile applications would be possible to implement with the human capital that the firm currently has at its disposal. If management does not support the adoption of this new technology or users are sceptical about using mobile applications, it may not be a wise decision for the firm to implement such technologies, Strategies for implementing the new technology must be communicated to the staff and management.

**Chapter 7**

**Future Scope**

As per Statista report of Q2 2021, users downloaded about [**28 billion**](https://www.statista.com/statistics/695094/quarterly-number-of-mobile-app-downloads-store/) apps from Google Play Store and about [**7.9 million**](https://www.statista.com/statistics/695094/quarterly-number-of-mobile-app-downloads-store/) from the App Store. Additionally, by the end of 2022, US consumers will spend around [**$34 billion**](https://www.statista.com/statistics/695104/worldwide-mobile-app-consumer-spend-by-region/) and even more on apps via the app stores.

As recorded in 2021, worldwide, there are about [**7.1 billion**](https://www.statista.com/statistics/218984/number-of-global-mobile-users-since-2010/) app users. Every smartphone user has installed about 35 and more apps on their devices. Out of those, they uninstall some because of some issues like performance. Still, nothing has stopped the growth of the mobile app development industry. The global app revenue of about[**$581 billion**](https://www.statista.com/statistics/269025/worldwide-mobile-app-revenue-forecast/) in 2020 is expected to reach nearly [**$808 billion**](https://www.statista.com/statistics/269025/worldwide-mobile-app-revenue-forecast/) in 2022.

Noticing the hike in these stats, more and more start-ups and enterprises are emerging to develop mobile [app ideas](https://www.emizentech.com/blog/mobile-app-ideas.html) for their business. Besides, this increased the competition in the mobile app development industry.

So, developing a simple app including standard features will not make your app stand uniquely.

So, to make your app stay ahead of the curve, like most enterprises, you should start adopting the latest technologies and trends. Before that, you need to know such future mobile app development trends and technologies to craft an effective [app development strategy](https://www.emizentech.com/blog/mobile-app-strategies.html) that works. This will result in successful app development for sure.

**References**

1. <https://material.angular.io/components/categories>
2. <https://angular.io/>
3. <https://angular-university.io/>
4. <https://spring.io/>
5. <https://medium.com/angular-in-depth>
6. <https://www.baeldung.com/>
7. <https://www.educative.io/blog/angular-routing-guide#lazy-loading>