A report on

Customer Relationship Management System

Project report submitted in partial fulfilment of the requirement for

the degree of

Bachelor of Technology

in

Computer Science and Engineering

By

Rishu Verma (151370)

Under the supervision of

Mr. Faizan Buch

to

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology,

Waknaghat, Solan

Himachal Pradesh, 173234

CANDIDATE’S DECLARATION

I hereby declare that the work presented in this report entitled "**Customer relationship management system**" in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out from February 2019 till date under the supervision of **Mr. Faizan Buch,** Product Manager, Jungleworks, Click Labs Pvt. Limited.

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

(Student Signature)

Rishu Verma- 151370

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

[illegible]

(Supervisor Signature)

Mr. Faizan Buch,

Product Manager,

Jungleworks, Click Labs Pvt. Limited.

Dated:

ACKNOWLEDGMENT

It is true that one needs to work hard to succeed in his/her life but to be true it is not just the hard work that pays but sometimes what matters is the right guidance and also the right attitude to grab the opportunity. I believe this is all due to the guidance and co-operation of my seniors, mentors, friends and parents that motivated me to work hard enough to grab this opportunity to work on the live project in industry.

First and foremost, I would like to express my gratefulness to Prof. Dr. Satya Prakash Ghrera, FBCS, SMIEEE Professor, Brig (Retd.) and Head Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology and Mr. Pankaj Kumar, Training and Placement Officer, Jaypee University of Information Technology for providing me the opportunity to work with such passionate organization like Click Labs in my final semester of bachelors course. It gives me immense pleasure to express my deepest gratitude and respect to Mr. Faizan Buch, Product Manager (Bulbul), Jungleworks for always being a great manager and support whenever required. I would like to thank entire Bulbul team (Mr. Pradeep Kumar, Mr. Shubham Sharma, Mr. Srijan Dubey, Ms. Aditi Bansal) for not only imparting their knowledge but also their constant supervision, advice and guidance whenever required, without which this internship wouldn’t have been possible. I would also like to thank Mr. Jatin Kapil who guided me during the training period at Click Labs and introduced us with possible real-life problems in the development. Working with this organization was indeed a great experience, especial at this crucial time of my life.

I would also like to thank all other department faculty at Jaypee University of Information Technology. Not only did they taught me and made me capable enough to undergo this industrial experience but were always there at the need of the hour and provided with all the help, facilities and co-operation, which was much required to reach this level.

A special mention to Ravi Raina Sir and Sanjeev Kumar Sir who assist the project lab and guided us towards all the minor issues. They were always there to inform us at times regarding all the details form presentation dates to report format.

Last but not the least, I would like to express my thanks to my parents and family members for their support at every step of my life.

Special thanks to divine Shri Kunj Priya Lal Ji, my father and my elder brother.

<u>**TABLE OF CONTENT**</u>

LIST OF ABBREVIATIONS

API	Application Program Interface
CRM	Customer Relationship Management
IT	Information Technology
HTML	Hypertext Markup Language
CSS	Cascading Style Sheet
ES6	EcmaSscript 6
JS	JavaScript
SQL	Structured Query Language
SRS	Software requirements specifications
SDL	Software development lifecycle
UI	User Interface
SPA	Single page application
CLI	Command line interface
DOM	Document object model
DNS	Domain Name System
IP	Internet Protocol
URL	Uniform Resource Locator

LIST OF FIGURES

ABSTRACT

In present days technology has owned like every second of our life from morning alarms to late night movies ever minor to major things is digital and involves some innovation from some brilliant minds, who invested their days of hard work, several nights so as to make our daily life easy. Sometimes it is worth that take in order to make million other lives easy and sound. It is important for every nation to grow in terms of economy and strengthen their armed forces with latest technologies. But every minor step count to take a major one. So it is important to digitalize our day to day work and contribute in this wave of digitilization. This will help not only in contribution of making our country more digital but also help in making our work easy, more productive in less time, more automates, less chances of any kind of security breach and corruptions because the best part of digitalization is the foot prints it leaves around.

This era of smartphones has made it easy for the user to get things gone in seconds and that too with digital footprints all around. It take seconds to transfer funds, milliseconds to call a cab and as estimated, by 2020 for every person on earth around 1.7MB of data will be created every second. So, it is time to tighten the shoe laces and get ready to find out the way how one can remeber all this especially when the data is all about the business and million customers around. How to keep track of the deals one make, how to remember hundreds of follow-ups because missing one may turn lacs or crores to zero. With this idea in mind and considering several other real-life problems the startup Jungleworks came up with a product named Bulbul, which is a CRM (Custom Relationship Management) Application.

The project Bulbul is an excellent thought to execution product in market under the name of very well flurished startup Jungleworks. Many employees work hard to make this product better day by day. Bulbul is a customer relationship management product, which best suits the sales department where the employees need to keep a track of their deals and followup. Keeping manual records on pen and paper of the deals and follow-ups can be a quite messed up way of managing the records and also one may lose the details. This product takes care of every user needs and provides everything in the hand of the business owner. This product is not only helpful for the sales in particular but logically considering, may it be any business or any department of the company, some way or other it will involve sale and if one sale one get a deal. So **sale**, **deal** are the two code concepts on which the project holds.

The entire product revolves around digitializing the relationship that the employee holds with the customer. This is a quite advanced and user friendly web based platform with various features to make thing easy. The technology stack used in this product is AngularJs for UI development, NodeJs for backend APIs, SQL and Mongo DB for managing the databases.

Chapter-1

INTRODUCTION

1.1 Introduction

Custom Relationship Management **(**CRM) system are the integration of various system parts and processses that help to maintian a healthy relationship between the employee and the customer, throught the use of latest technology and digitalization. The system helps in maintaining the records of the deals and the followups. The key features of this system are that it keeps in mind all the happy and corner cases of the entire system design, like every minor thought has been implemented as a feature to make the flow easy and business more productive.

Figure 1.1: Sales CRM

The system has various features and is a long term project, many developers and other brilliant minds are working on this system to improve it day by day and add new features to it. But the main focus in kept on features that market shouts out for and these include Team collaboration, Client Management, Actionable insights and email tracking. No doubt the CSM system which is the product of Jungleworks names as Bulbul is a CSM developed keeping in mind the basic rules of UI/UX design and the best part of this product is that one can start working on it without any training to this. It is easy to use all thanks to the features and the designers who worked on it and are still working on it.

The CSM is sufficient enough to manage a team working together on a singe deal, one can set activities, check deal status and change the role and permissions of the team members as well, feature included keeping in mind the team hierarchy (usually a team has a team leader and other members as per the experience). The user can create the client's profile with all the client's details in it which work as the directory, the system keeps record of all the email conversations one do. One of the best feature that can be considered top featues in the ability of this system to get leads from multiple channels, the system allows the user to capture leads directly from the website or webforms by using the APIs provided by the system backend. The system also provides with the feature of chat support which is the integration of another Jungleworks product names Hippo.

The team can also see the performance of the members by seeing the status of the leads and the stats if the member won the deal or lost or any other stats. The platform also provides the feature to make calls by the third party API (Plivo and Ring Central) and also provide Whistle which is a browser extension for Bulbul. One can easily sync the official email and then can create the email templates and email journey dependind on the lead.

So this CRM system named bulbul in itself is a well established product with various amazing and mind-blowing featues in it.

Figure 12: CRM Icon

1.2 Problem Statement

The main problem which the team is trying to get the solution is the addition of new features on the CSM and trying to make it more user friendly and user convinient. For this the team is working on to add new features to already established product. Not only keeping the system up to date to the user requirement is the challenge the team is working on, but also the team is trying to make this system available on smartphones.

The challenge on this point of time is not the development of product from very scratch, but to think all the corner cases of the way the normal employee managers it’s leads and then turning that cases in the form of features. Another challenge is the production rate, due to the increasing market demand it is important for the team to keep in mind the deadlines and also the whole hierarchy of the product development and deployment. It is always kept in mind while developing that the demanding feature should be developed two to three days before the deadline and then the feature is put on test and after testing all the flows of the feature (happy flow, pleasant flow and all corner cases) and no major bugs are found then the product is put on beta so that the feature can be tested by sales department of the company itself. After that if the tester finds no bugs and the feature performs good at beta, it is made live. So these are the few major steps that cannot be ignored in real-life development, and it is a challenge to deliver the product on time.

Also, the team is planning to make this system avaliable on the smartphones as well, before this it was just a web application. For this first the major challenge was to selection of the framework depending on the amount of time avaliable. The framework we choose was Native Script which supports Hybrid App Development (i.e the developer writes one code and it builts a web application, an android application and an iOS application at same time). This is a great take but the challenge here is that the framework is quite new and also it requires the code base in angular but the CSM we have had a codebase of AngularJS. So we are working around to figure the things out. We are trying to find the solution of all these real-life production problems and working on the features and the hybrid applications.

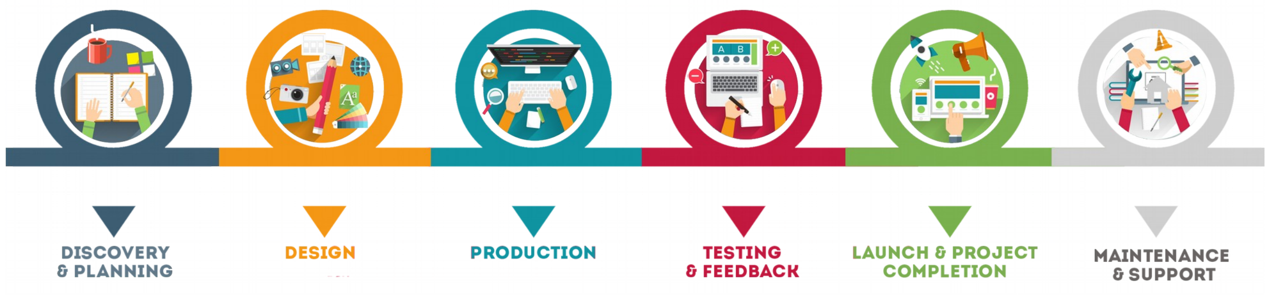


Figure 1.3: Major steps in production environment.

1.3 Objectives

The Objective of this project are:

1. To keep the product upto the market and user’s demand by adding features to it time to time.
2. To increase the product efficiency and keep maintaining it timely so that the time it takes rendering the pages after the API hit can be reduced.
3. Keep unit testing the product with every feature to avoid reaching major bugs when the deployment dates are near.
4. To develop an hybrid application keeping in mind the avaliability of resource and the delivery time of the application.
5. Optimizing the line of code to as to community standards so as to make it easy for every new developer to understand.
6. Considering a team work one need to manage all the updates on git so working on gitlabs to make the development easy and timely.

This way by using the knowledge of web development, page rendering, deployment, VPN, git, grunt and development required frameworks such as AngularJs, Native Script in essence with the code standards we have tried to add new features to the product and make it more user friendly and upto date.

1.4 Methodology

The various methodologies used involve the frameworks namely AngularJS , Angular, Native Script. The use of technologies and languages like HTML, CSS, Bootstrap, Javascript, AngularJS material library and grunt. Apart from this keep the code on git and managing the code in a proper manner was gone on gitlabs. All this together helped in compiling and executing the project to practical implementation.

1.4.1 Frameworks

In computer systems the frameworks are the layered structure that indicate the kind of system to be built or can be built and the interrelate between them. Some of the new computer system frameworks include actual programs, specifying program interfaces while some do offer programming tools for the use of the framework. Frameworks also help in standardizing the communication at some level of the network and vice versa. One can say that tthe frameworks are more complete than the protocols and are more dictatorial than a structure.

More or less the frameworks are the standard way to deploy and built any kind of the application software. It is universal hence the standards are maintained, also it is reusable in terms of software environment providing some particular functionalities as a prt of the larger software platforms in order to facilitate the development of the software products, applications and solutions. There are various frameworks used while by me while working on frontend of the project are Angular, AngularJS, Native Script. But considering the entire product the backend uses NodeJS for development.

1.4.1.1 AngularJS [9]

Figure 1.4: AngularJS logo

AngularJS is an frontend web (javascript based), open source framework. The framework is principally maintained by Google alone with the community of individuals. All these community individuals and corporations address various challenges that other users of the framework face while development of single page applications. So angularJS framework is majorly used for SPAs (single page applications). The major reason why some companies prefer it over React are that angularJS alone is sufficient to make a SPA and one don’t need anything else while ins React framework one need to get additional libraries to make an enitre application. The best about this framework is that it is quite easy to work on this as one can easily get the enitre start and concept explained really really well in the official documentation, the documentations of both angularJS and angular are tutorial based and hence explains all concepts really well. AngularJS is one such framework that works on imporving the vocabulary of the HTML.

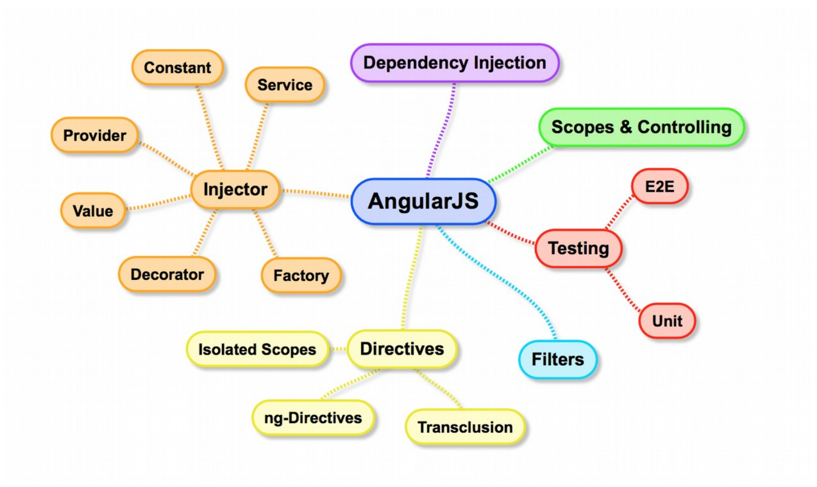


Figure 1.5: AngularJS features

With AngularJS, HTML is no more just declaring static document renderer now it can render dynamic views as AngularJs provides the gift to HTML by letting it add loops and more HTML-Javascript integration in the HTML document. So yes, with AngularJs HTML can now render the dynamic views pretty well. Also, unlike another frameworks which never worked on the fact that HTML is never made for the purpose of rendering the dynamic views and always focused on the HTML's flaws either by proving the repeaitive way to manipulate and render the DOM or by providing million other abstraction layers on HTML, JavaScript and CSS. So AngularJS is really a toolkit which has nearly everything and if it lags somewhere then it provides full support and is highly adjustable to work well with other libraries.

1.4.1.2 Angular [10]

Angular is nothing but improved and updated version of angularJS. Angular version 1.0 is named as AngularJS by the community whereas all other versions above that are named as Angular further specified as Angular 2.0, Angular 3.0 and so on. The latest one is Angular 7.0 also known as Angular CLI (Command Line Interface), called so after the new updation in the framework that is, now the user can easily create the new project, modules, components, services and also automatically link and declare the created modules, components etc. That means the user just have to write one simple command and nothing else, no more efforts to maintain the file structure. Apart from this Angular CLI is capable of making mobile (hybrid Android+iOS) as well as desktop applications with one code, all this possible with the help on frameworks like NativeScript that works with Angular, React and Vue.

ANGULAR

Figure 1.6: Angular Logo

As mentioned on of the feature of Angular CLI is code reusability, one code can be used for both mobile and desktop application development. As Angular unlike AngularJS, which works on the concept of Promises, works on the concept of Observables which make data fetch from api even faster and hence increase the performance of the SPA. Angular provides it's plugins for nearly ever ide and hence also allows working on localserver with autorefresh feature. As AngularJS it also extends the HTML vocabulary but now we make use of our own components and also make use of several other existing components. Also AngularJS makes use of JavaScript whereas Angular works on TypeScript. TypeScript is a super-set of JavaScript.

1.4.1.3 NativeScript [11]

NativeScript is another really efficient open source framework which is used for building native mobile applications with Angular, Vue.js. It make use of TypeScript when used with Angular and JavaScript can also be used for development, but in case of JavaScript the developer should be well known worth XML. In this project we worked on NativeScript with Angular as the entire project is based on AngularJS so the plan is to shift the codebase to Angular, but as it is not easy to shift thousands of files and already deployed product to some other framework so easily so that work is being started side by side. NativeScript with Angular integration has all the features of Angular CLI from routing support to code generation via CLI, webpacks etc.

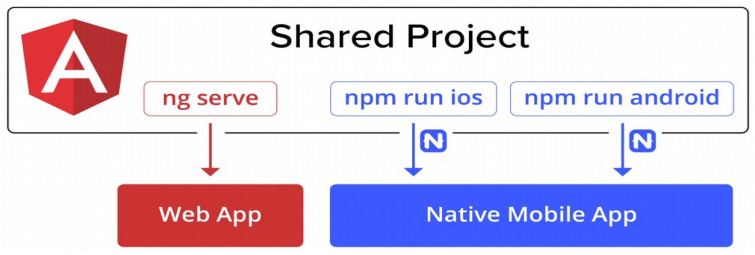


Figure 1.7: NativeScript with Angular

No doubt it is not important to use Angular with NativeScript but it makes things more easy and on can easily reuse the code for both web and mobile applications, side by side maintaining the file structure. So this integration is the lastest and the fastest integration. Angular with Native makes it possible to create a high quality code that is maintainable and reusable in terms of DI (dependency injection), routing etc. The best thing about this is both Angular and Native have a vast and passionate community that responses to all the queries within no time. The only hectic process while working on the NativeScript application is system setup, but the entire process is explained really well in the documentation and if stuck one can find the answer to the problem on the community forums.

1.4.2 Technology and Languages As in real life language is important for communation in the same we have programming languages in the computers that help the developer to communicate with the system. So programming languages are more of set of rules as we have in our grammer and it is vocabulary of the computing devices. As there are many languages in real life in the same way we have many languages for computing devices such as C, COBOL, C++, Java etc.

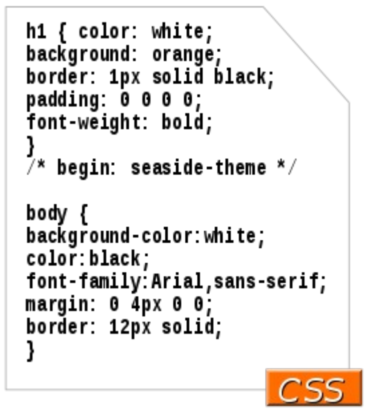
Also when a team or individual is working on a specific project then the first and foremost step is to decide the technology stack, which is basically the set of technologies that will be majorly used when developing the system or application. Like for backend development the technology stack includes web frameworks (can be any depending on the developer’s feasibility, need of the operating system (mostly developers prefer ubuntu or macOS for the development purposes) and then comes the programming languages and servers to deploy the appilication on world wide web.

1.4.2.1 HTML

HTML- Hypertext markup language is the most basic and required language to work on web development. It is the language designed for maipulation of static documents, but now with the improvement in technologies we have thousands of frameworks are make HTML capable of manipulating the dynamic views. This language is based on the tags and attributes that together in the html code helps in describing the structure of the web page. So the tags are nothing but the representation of the HTML elements. Not only this, the browsers help in rendering the HTML by understanding the sementics of the HTML, so it do not display the HTML tags or elements on the screen but rather render the page accrordingly. It is very important to use the HTML properly as per the standards because it helps the text-to-speech applications (which usually works for users who are specially abled) to read the content well in proper sementic order. HTML is always considered not so major language as it is easy to learn and use, but this is the root of any web based application. So it is not important to use the language, but important is to use the language in the standard format because there are many other applications that work on these standards.

1.4.2.2 CSS

CSS- Cascading Style Sheet is a programming language usually used with HTML for designing and presentation of the page. With CSS the look and presentation of page can be done million times better. CSS controls the styling and layout of multiple web pages at the same time. It can manipulate the texts colors, text fonts, positions and hence make the web page decorated and look sobber. Now a days, CSS is widely used arcoss the developing community for making doodles and other pattterns that catches user's attention and this is really creative to make things done with CSS. There are three ways in which CSS can be added to HTML, inline, internal and external. <style> is the tag used for internal addition of CSS to HTML. But the prefered and standard way is to make a sperate CSS file and use inline addition, as it helps keeping the code clean and visually appealing and more understandable. Also Bootstrap which is an open source CSS framework, contains CSS and JavaScript templates, (but mostly CSS) works on the concept of dividing the screen to 12 columns. But the concept of flexbox and grid has made CSS more easy. Now we have several CSS extensions as Sass, SCSS etc that make the writing CSS more easy and professional.



```
h1 { color: white;
background: orange;
border: 1px solid black;
padding: 0 0 0 0;
font-weight: bold;
}
/* begin: seaside-theme */

body {
background-color:white;
color:black;
font-family:Arial,sans-serif;
margin: 0 4px 0 0;
border: 12px solid;
}
```

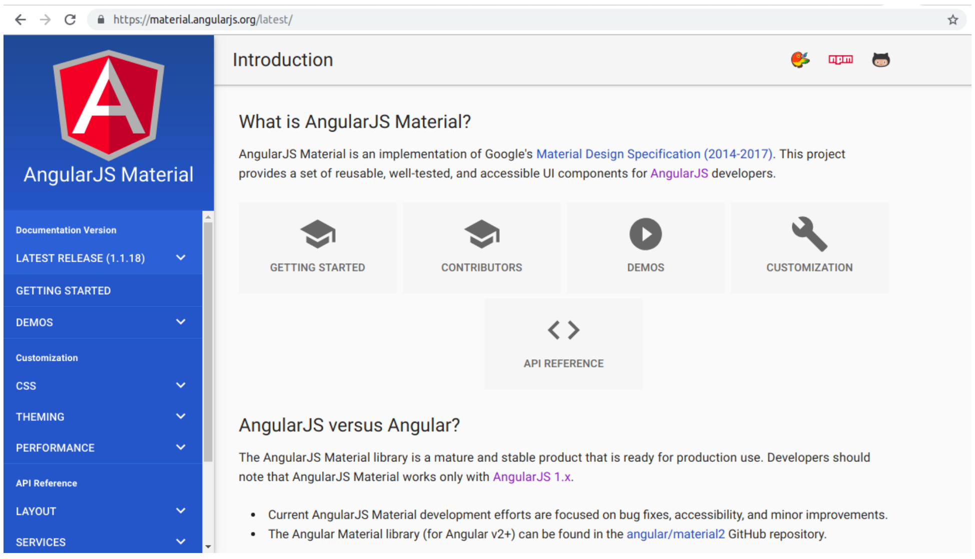
1.4.2.3 JavaScript: [8]

JavaScript abbreviated as JS is a lightweight interpreted, prototype-based, high-level programming language which is most widely used because of the ease of learning and syntax of the language. This language complies with the rules and standards of ECMAScript specifications with curly-bracket syntax and just-in-time compilation language. Not only it works well in browsing environment but also works well with many non-browsing environment. It is dynamic language which supports object-oriented styles along with declarative or functional programming. This languages helps very well for APIs that helps rendering of the HTML page dynamic and fast. TypeScript

(.ts) is the superset of JavaScript used in Angular framework. Also there are thousands of third-party libraries (like Jquery etc) and frameworks made on this language. Also, this language have many features (but it is not capable of multiprocessing and multithreading) with which the developer can manage date, time, browsers's window and the operating system from which the user is using the application, all this can be easily done with this scripting language.Though it cannot be used directly for the networking applications but it can we used with other third-party libraries to do that as well. With JavaScript one can do anything with DOM manipulation from managing static forms to creating dynamic forms to managing SQL queries all can be done with JavaScript. The language is so vast and high performance that all the browsers not only support the language but also helps in debugging it.

<u>1.4.3 Libraries, Softwares, Operating System, IDE and Plug-ins</u> The most basic requirement when one start developing is IDE in this project development we have worked on **Visual Studio Code** running on the Ubuntu 18.04 operating system. VSCode is Microsoft's IDE that has many intgrations and framework support available in it, it even supports git. It has various themes and terminal support makes the work more interesting and easy. It is highly optimized IDE developed for the purpose of building and debugging. The library in computer systems is the collection of stable resources often used by computer programs majorly for software development. These include type specifications, subroutines, templates etc. Plug-ins are the software extensions or software components that add some addition specific feature to already existing program.

1.4.3.1 Angular Material [6]



What is AngularJS Material?

AngularJS Material is an implementation of Google's Material Design Specification (2014-2017). This project provides a set of reusable, well-tested, and accessible UI components for AngularJS developers.

AngularJS versus Angular?

The AngularJS Material library is a mature and stable product that is ready for production use. Developers should note that AngularJS Material works only with AngularJS 1.x.

- Current AngularJS Material development efforts are focused on bug fixes, accessibility, and minor improvements.
- The Angular Material library (for Angular v2+) can be found in the angular/material2 GitHub repository.

<u>Figure 1.8: Angular Material</u>

AngularJS material is the application of Google’s material design specification. This provide an approachable UI components which are reusable in nature for AngularJS developers. Material designs provide the visual, more realistic and interactive designs that can easily be made fit to different devices and are responsive by nature. It is designed keeping in mind to provide the lightweight set of UI elements that helps in making the SPAs more efficient and attractive. Also the CDN version of angularJS material in avaliable which makes it even more easy to use AngularJS material because by CDN developer don’t have to bother about the local copies of the files distributed globally.

1.4.3.2 Grunt

Grunt is a JavaScript task runner which is used to reduce the work by automation of few works like minification and uglification. Gruntjs is used for automizing unit testing, compilation, linting and hence clean code is deployed on the server there by reducing the load on the servers. So Grunt helps doing all the humdrum works for the team with almost zero efforts. It itself is written in Node.js and uses a CLI to run all the custom tasks defined in the grunt file.

1.4.3.3 Git

Git is widely used system software which is usually used when large number of people are working on a single project parallelly. This is a distributed version-control system which keeps track of of all the changes in the code during the enitre process of software development. The motive with which this system came into existence is to speed the process of development between the team, to support the non-linear workflows and to maintain data integrity. Git is a concept based thing and hence is easy to understand and work on, it works on the concept of branches, merges, rebasing etc. It helps maintaing the work on local as well as global and origin.

1.5 Organization

The project report is organised in such a way that in

Chapter 1 basic introduction of the entire system covering the problems statement that what led the comapany to this product and all other real-life project deployment problems that were to be kept in mind while development of the project. Also, basic introduction to all the major technologies, libraries, softwares and frameworks is given.

Chapter 2 is all about the literature survey and the study that was done before starting the development process. All about the way HTML is rendered, the way browser deals with the Angular framework, watches, digests. The concept of promises and other techniques to work through the concept of the asynchronous API hits.

Chapter 3 describes the system design, deployment process and other architecture of the web application and the hybrid application.

Chapter 4 descibes all the work done during the entire internship period and analysis of the problems encountered.

Chapter 5 concludes the project completed till date and about the furture scope and other features that are in process of being developed.

Chapter-2

LITERATURE SURVEY

It is always a good practice to be well know on the topic you are about to work on. The same practice is also considered in case of development. The more the developer knows about the roots of the technology and understands how it works, more efficient work the developer will be able to perform, more easily the developer will be able to find the solution to the problems they are facing in the real-life working environment. For this purpose all the interns who joined the company had to undergo training so that the interns get aware about the problems that the developers face in their daily life. Not only this but for such kind of problems the interns were asked to find the solution and solve them in the form of assignments. In this chapter I have described all those research work I did at the time of training and also at the time of production experience when I got stuck in some kind of error, or and production problem.

So here the main focus is on the few core concepts that one usually forgets to notice beacuse of the layers of abstraction that the new frameworks have. It is noticed that 98% of the times the problem comes in picture due to the fact that the developer is not known to how the flow works and how things are coming into picture. Due to this reason we were first made to focus on the very basic languages and core concepts of those languages, so that even if we don’t have a good hold on the framework we are working on, we still can solve the problem with the knowledge and understanding of the basic native languages. So the entire training process was based on the practical knowledge and understanding.

Managing the API hits sequence and considering the case when there is millions of data at the time of rendering then, what is to be done. How can one avoid the situation when the user is on screen and the screen is blank for like few minutes, and as obvious reasons this is really a huge mistake and voilation on UI/UX rules, as it is highly unpleasent to show a blank screen to a user of to even show a user a loader for few minutes. So starting from very basic the content will first discuss about the way HTML is rendered from whole DNS, IP, domains and all. Then we will move on to the concept of digests and watches that are present in AngularJS and how browser works for that. Next we will discuss about the asynchrounous calling, callbacks, the concept of promises in AngularJS and the observables in Angular.

2.1 Browsers and HTML rendering

It is always the best practice to learn the internals of how the browser operates when the user types some ‘www.xyzab.com’ in the address bar in the browser because it will make the developer more practical and more close to the understanding of the entire process of development. As always said treasures are found when we dig deep inside the ground, so in the same way the more we backtrack the concepts and technologies, the more we dig deep in the abstraction layers of the frameworks and technologies the more we will be able to discover, that will help one to relate and find the solutions to all the current problems and justifies the entire process behind the development. There are many browsers but the few mainly used like 71% desktop browsing is done mostly on Safari (macOS), Chrome and Firefox. On mobiles, mostly the work is done on android broswers and Iphones again we have Safari.

The question that is really very common and comes across many a times is that why do we ever need browsers, like okay the developers can deploy their code on the severs and the other users can directly connect to that IP and it’s done. No, it’s not that simple because the code that the developer writes is not the code in layman’s language and also if the website is global then obviously if the code is written in english then other non-english people won’t be able to understand the things. Nevermind, it is not possible to have a code in a commonly spoken language untill processed by AI (that is again a layer of abstraction, but practically the code has some sementics and some rules that are to be followed in order that the computer understands it and so the required tasks.

If we don’t have any browser then we will (considering the example of any web application) see the HTML tags and elements on the screen and a common man will not understand all these and consider there is some error in the webpage. For this purpose of rendering the webpage (maybe an HTML document or an image or any kindof pdf file or any other content), the user choose, from server and display it on the browser’s window. Now comes there are millions of websites across hosted on world wide web, then it is quite cumbersome and tiring task to allotte and kind of serial number or maintain the record of the names, also the probability of mistake incerased. Not only this, but also the changes of the vulnerability and security breaches increases. For this purpose the system has URI – Uniform Resource Indentifier where the location of all the resources is specified by the user[1].

W3C- World Wide Web Consortium has specified all the details of how the browser interprets the HTML and CSS before displaying it on the browser’s window. All the browsers have few common interface elements as the address bar, bookmark option, refresh or stop button, back button, forward button, home button, a few more but these are the few that can be easily noticed when one visits the broswer window. All this is quite odd, as there standard or specifications that how the broswer should be made, all this because of the few metioned specifications in the W3C [3].

The other main components of the browser apart from the UI are:

1. Browser engine
2. Rendering engine
3. Networking
4. JS interpreter
5. UI backend
6. Data Storage

All these components work together for the rendering of one single web resource. The browser engine organizes all the actions between the user interface and the rendering engine, next comes the work of the rendering engine. Rendering engine is responsible for displaying the content the user has requested, depending on the type of content the user has asked for, can be HTML, CSS, pdf, image etc. Networking helps in API hits and other network calls like the HTTP requests, here the major concern is the type of platform from which the request is being made, if it is secured or not, saying cross-platform requests which is again a next big topic to consider when considering the HTTP requests.

The user interface backend is used for displaying basic widgets like the windows etc, thst is all the generic interface that is not platform dependent or platform specific is displayed by the backend. Also this make use of the operating system UI methods. The JS interpreter is used for parsing the JS code in the web resources. The data storage is used for the process of storing cookies or cache, basically for local storage of data that most of the developer use to make the rendering fast. Apart from all this the browsers also support IndexedDS, fileSystems etc.[2]

Now the browsers are so advanced and are trying to enhance the performance in terms of speed by running multiple instances of the same rendering engine so that the user could surf multiple tabs and each tabs works independently and each tab in itself is a seperate process. In the below figure we can easily see and understand the components of the browser with the help of the layers and hierarchy in which they work. Also, by default the rendering engine alone without any extensions is capable enough to display XML and HTML documents and images for any other type of content plug-ins or extensions are required.

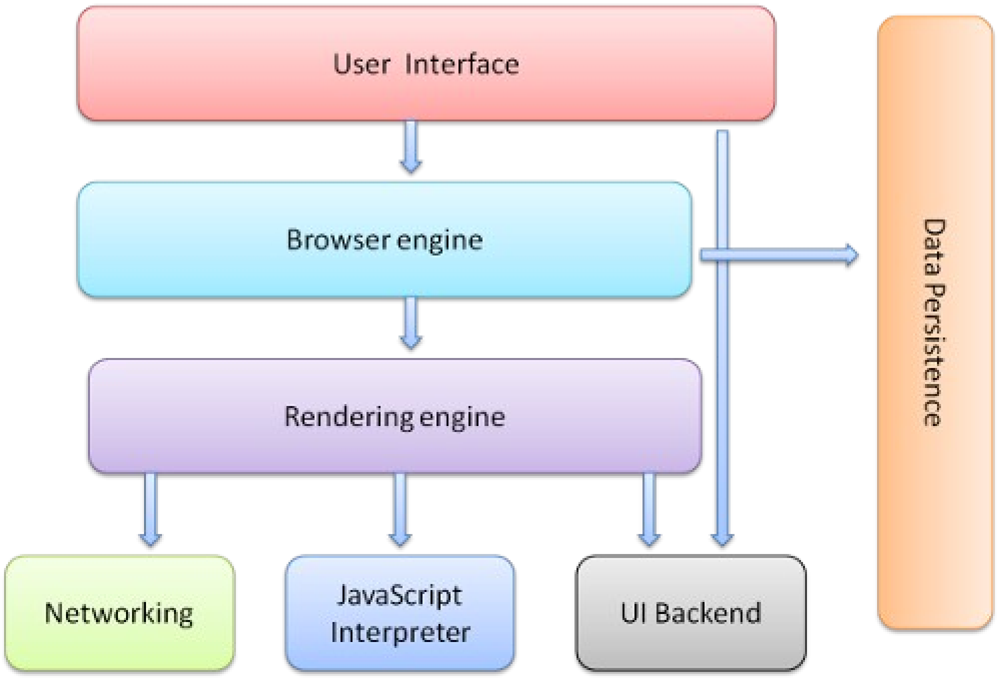


Figure 2.1: Broswer Components

Different browsers have different rendering engines, Chrome and Firefox the mostly used browsers use Blink and Gecko respectively, Blink is a fork or say addon of WebKit which is an open-source rendering engine used in Safari.

The basic flow of the rendering engine can be seen from the below image:

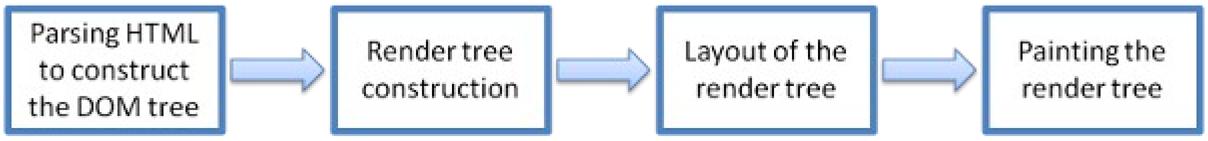

Parsing HTML to construct the DOM tree
Render tree construction
Layout of the render tree
Painting the render tree

Figure 2.2: Flow of rendering engine

The rendering engine first parses the HTML then it gets out the elements from that HTML and convert them to the DOM nodes in **content tree**. Also if there is any styling information that is set to another tree names **render tree** (all this parsing quite similar to the concept how compilers work, one having knowledge about how compilers work can easily relate). Then the entire layout process in done with the help of the render tree with exact coordinates pointed to on screen and the very next step is painting using the UI backend layer on the browser’s window.

As all this process is a compilation of many other series of sub-process, which makes it slow but for better user experience the rendering engine will not wait for the whole HTML to parse rather it will start displaying as soon as it gets the results while the result of the content and the request’s response keep coming from the network. Now the below two images can very well explain the main flow of the webkit and Gecko the two most common rendering engines.

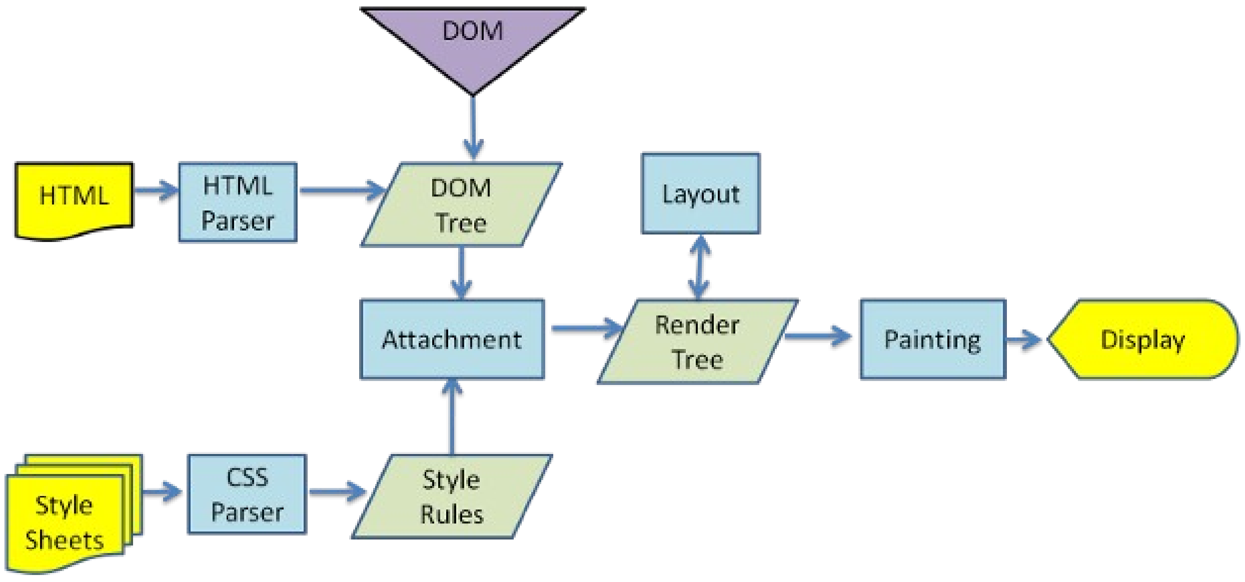


Figure 2.3: Webkit main flow

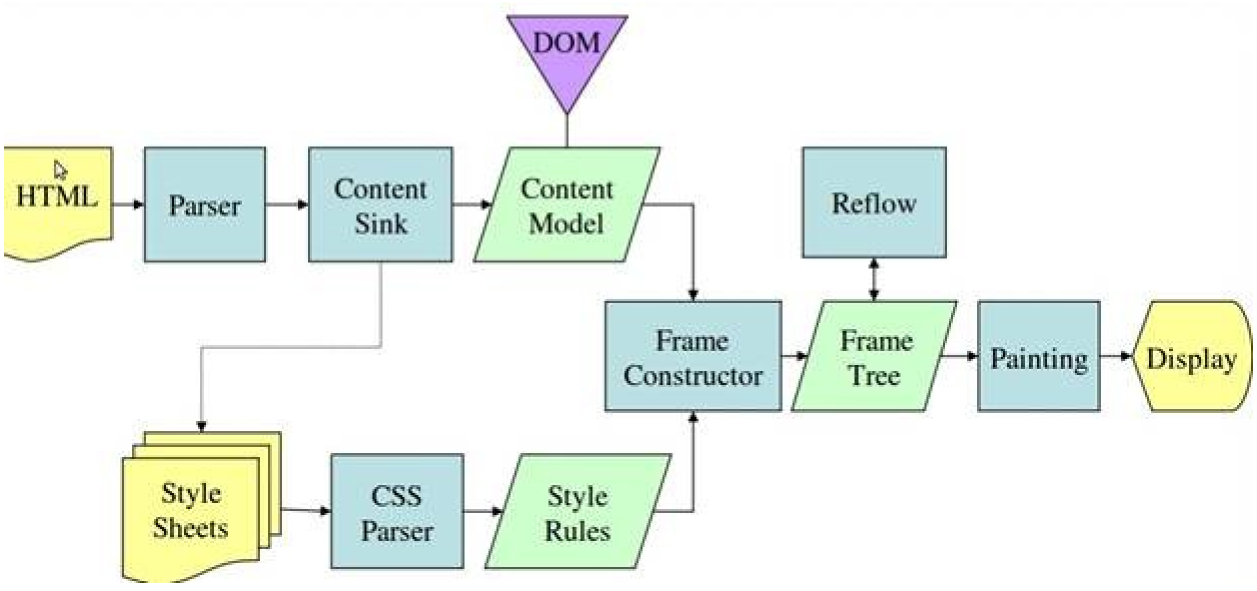


Figure 2.4: Gecko's main flow

Now the below two images can very well explain the main flow of the webkit and Gecko the two most common rendering engines. All the above information is all about the actions that browser takes when the domain name as 'www.xyzAB.com' is put in the address bar of the browser now let's take a look that how the domain name is identified in by the servers and what happens between this domain name to server.

2.2 URL to web resources

Internet oftern referred as a global network is a set of millions and trillions of computers in a network connected to each other for the purpose of communication, in this network every computer follows certain set of rules and standards names as Internet Protocols (IP). Ever computer on this network has a unique identity that distinguishes it from other computer or computing devices in the network. These IP addresses are in the format of either IPv4 or IPv6. IPv4 follows a format as xxx.xxx.xxx.xxx with each of the 4 sets ranges between 0-255 whereas in Ipv6 this format extends to 6 sets as xxx.xxx.xxx.xxx.xxx.xxx. So, one can clearly conclude the amout of IP addresses available, few of which are reserved and rest are distributed by the distributer for hosting the websites.[7]

But it is quite confusing and difficult to remember all these IP addresses for a normal user, not exactly a normal user but for anyone (untill someone is extermely good in numbers). So for that domain name came into picture. Imagine a class of say 20 students, then it will be easy for the reached to remeber eash student by the enrollment number, but as the class extends to hundreds or maybe thousands then it is not easy for teachers to remember every student with their enrollment number but it will be easy for the teacher to remember the student by his/her qualities. This same concept works for websites as well, as we know we have like millions and even more websites hosted on the global network, so in order to make this easy for the user to visit any website why not name is by the work it do and associate that name with the IP of the website. It was a great idea, but again how will such long list of association be managed, so there comes DNS (Domain Name System).

Simple explanation of DNS is that it is a system that resolves the domain names to IP addresses. When the user types the address/domain name, in the address bar of the browser and hit enter then this request for the web resources is send to the DNS and it performs the task to locate the corresponding IP address, the entire process of this locating is called DNS lookup. The lookup is done in proper hierarchy and distributed fashion that is if the IP address is not found in the local DNS then it will ask other DNS servers systematically till the time it reaches some conculsion that the IP address is available or not. The entire process involves various cases of actions like if the user visited the site recently then the IP address more specifically the domain will remain in the cache of the operating system or the browser.

If the address is not in the web browser then the request is sent to the DNS resolver, which is basically the local server with the data of the main DNS nameservers. The resolver will check it's cache if the IP address is found then it will continue the further process. If the cache is not found then the forward the request to the root server from bottom to root and then root to step-by-step bottom hierarchy as in the tree like structure. Then comes authoritative name servers which contains all the DNS records specific to that domain, hence will respond to the IP address of the specific domain. At all these steps caching is done so that the next time the entire process need not be repeated. When the IP address is known then the next focus is on the HTTP protocols, the browser follow the protocol and sends a GET request to the server with the returned IP address. As we can see in the image thar how the commution takes place in few steps, how HTTP request is send to the server and server responds to that request.

1. The user who wants to browse to a particular website enters the particular URL at the address bar of the browser.
2. After this the URL which is basically the domain name is looked up for the IP address in local DNS and if not found then in root or authentication servers.
3. Then the HTTP request is sent to that particular IP address, which is located from the DNS.
4. The server then do mapping of the request it has received from the URL, to the files under in the directory of the server.
5. The server send back the web resources in the form of HTTP response which is displayed on the browser window.

All these steps which are again processes wrapped by layers of abstraction, help in getting the data from client to the server.

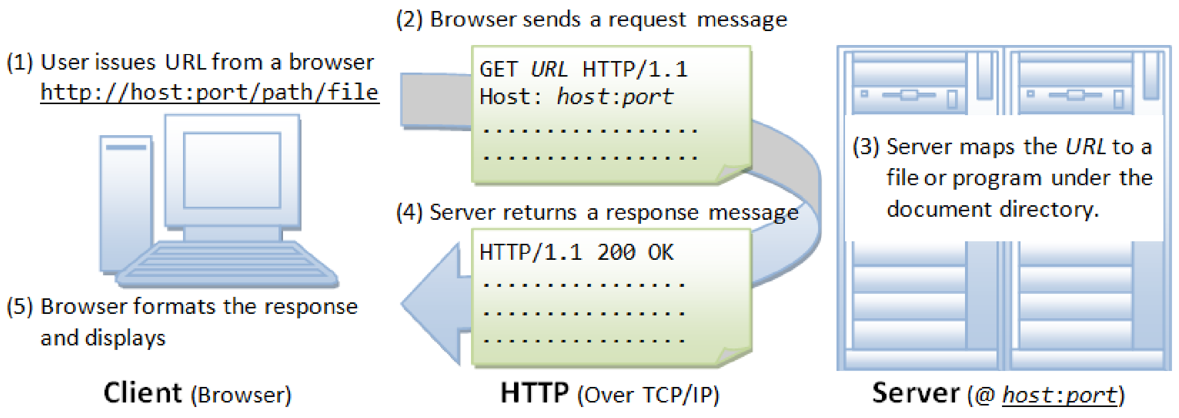

(2) Browser sends a request message
(1) User issues URL from a browser
http://host:port/path/file
GET URL HTTP/1.1
Host: host:port
(3) Server maps the URL to a file or program under the document directory.
(4) Server returns a response message
HTTP/1.1 200 OK
(5) Browser formats the response and displays
Client (Browser)
HTTP (Over TCP/IP)
Server (@ host:port)

Figure 2.5: Client to server netwotk request

There comes to concept of computer networks and TCP/IP protocols. TCP- Transmission Control Protocols working at transport layer as per the OSI model is responsible for creating a secure and highly reliable connection between two ends. This also helps in breaking the data in smaller chunks called packets and reassemble it at the other host. The entire TCP connection depends on the client’s and server’s IP address and client’s and server’s port number which together form a socket. Then comes three-way handshaking, concept of SYN and ACK flags. Though these packets are send in form of small chunks still there are chance of some virus to intrude during the transmission so for that the server's firewall acts as a banner that checks the packets header and other flags to filter the useful and required packets.

After all the filtered packets pass the firewall the major task is to handle the load on the single server, in case the single server is handling all the client requests, then to avoid failure or sudden collapse load balancer is requried. It handles the load of all types of traffic using some load balancing algorithms which are basically based on the concept of hashing and consistent hashing. Same steps are followed for both static and dynamic data but in case of dynamic data then Application servers come in picture.

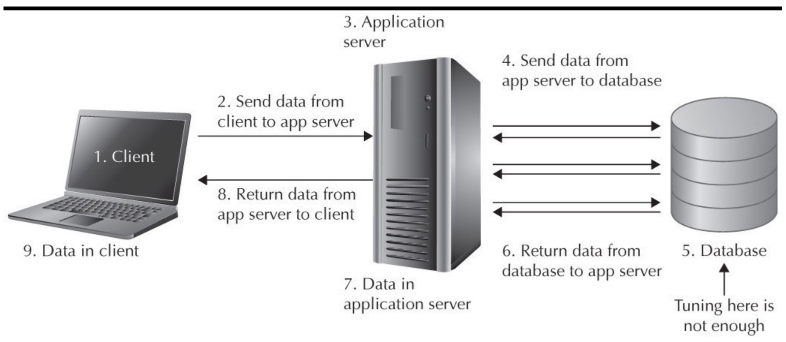

3. Application server
4. Send data from app server to database
2. Send data from client to app server
1. Client
8. Return data from app server to client
9. Data in client
6. Return data from database to app server
5. Database
7. Data in application server
Tuning here is not enough

Figure 2.6: Application Server

2.3 Browser's Event Loop

We have heard several terms like V8, single-threaded, callbacks and all when we ponder around the high-level object oriented language named JavaScript. The new common terms that we like surely get to are single-threaded, concurrent, non-blocking, asynchronous these are four key terms that every developer would compile JavaScript in, but these four terms have lot of abstraction layers to undiscover to know the actual working of the JavaScript. Things don't end up here when we dig a level we will get introduced to the terms like call stack, event loops, callback queues, APIs, heap, makes one feel like it is highly complex to understand, but actually it is very interesting to do through these abstraction layers and unserstand how it works what is callback, what is callback and all other terms. The answer of all these whats are clearly related to each other and hence is really interesting for the one who wants to understand the technology from the roots and be efficient in it.

Starting with V8, so what is V8? V8 is a JavaScript engine, it is an open source webAssemply engine, manages by Google and is written in C++ often referred as Chrome's V8. As the name JavaScript engine, it is the program that helps to execute JS code and runs in coordination with the DOM (Document Object model) and rendering engine. JavaScript runtime environment consists of heap for the purpose of memory allocation and callstack for execution context.[8]

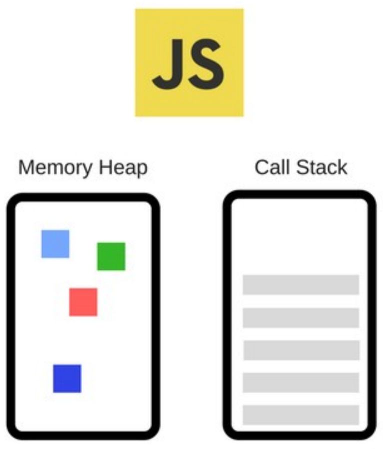


Figure 2.7: JavaScript runtime environment

When a developers writes a code in JavaScript then many times in order to make the code flow sequential they make use of functions like setTimeout(), also make use of document object model (DOM) and HTTP requests (GET, POST, PUT etc) and browser do understand the meaning of all these technical instructions, that to like it makes things work really cool and effective. So that means even in the broswer's JavaScript engine somewhere these terms must be defined so that the browser can understand that is the user asks for setTimeout() that means the process has to wait for say xyz milliseconds and if the such an event occures in which HTTP request is made then so and so function should be executed. Like genuine and basic concept of programming that if we have to call a function then we have to define it somewhere and then we can work as we want. But here is the surprise that came out of the research, that if we consider one of the most efficient JavaScript engine say V8 and we look at it's codebase, there is nothing like setTimeout() or HTTP requests or DOM. The very next thought, what is this going on, it is some sort of magic or what, not really as mentioned earlier it is very interesting and beautiful concept to understand how the flow works. Here is how it works.

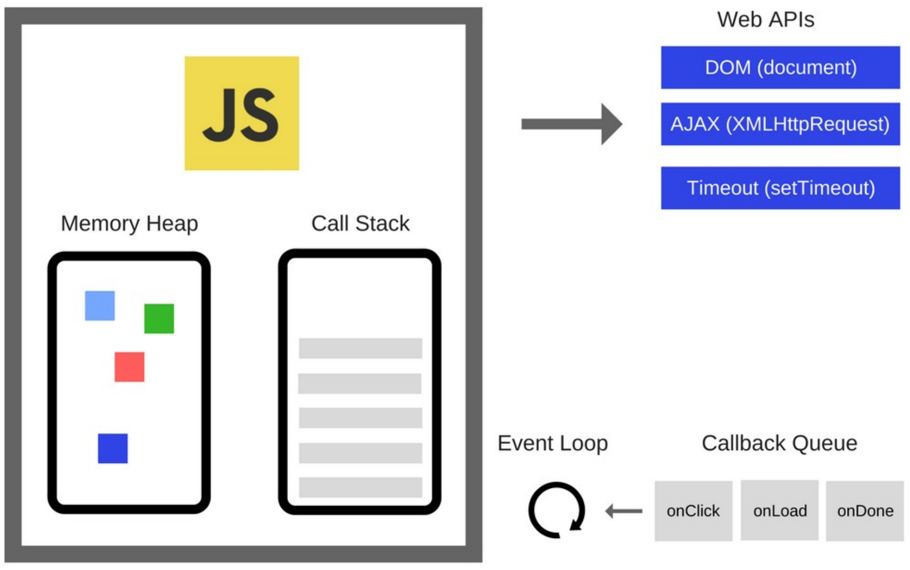


Figure 2.8: Event Loop

In V8 we have something like Web APIs these are some of the extensions that a browser provides and all this includes DOM, the XMLHttpRequests and timeouts. All this runs with the concept of event loop which is again a concept, and callbacks. As we know JavaScript is a single threaded high-level programming language, this means it has a single call stack and it can do only one thing at a time, the normal defination of single-threaded stands here. Call stack as the term consists of the word 'stack' which is a very popular data structure (with follows last in first out thing) and here is do the same work. It keeps records of the calls to be made in a program by keeping track of what is the next return function (or call) is to be called. As the engine goes through all the JS code it puts all the functions being called in the stack and the pops them out when the function is executed. So calling a function means pushing in the stack and return is poping off the stack, main is the first function that is pushed onto the stack.

When the function is called recursively with no end return statement or no terminating case then the call stack limit exceeds and the browser reports a bug in the console. Call Stack helps in sequential running the code, but what if we have a while loop that makes network requests, say we have while loop running say 10 times and each time the network request takes 1 sec (on say 3G internet connection), that means that the user will see the loader for 10 seconds and then the entire process below it will continue and if those processes are alerts of console or some other hardcoded code then it will be comparatively much more faster then this while loop and hence the user experience will be extremely bad.

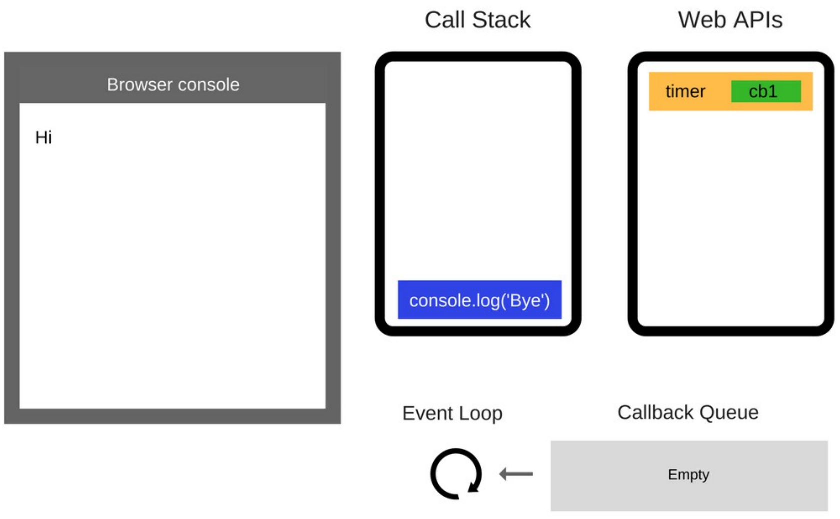


Figure 2.9: WebAPIs working

Then came the concept of asynchronous callbacks which works as, if some process is taking like longer than certain milliseconds then the browser will not wait for the process to finish rather send those processes to the webapis the pops that off the call stack and pushes other inline functions in the stack, remember the running process is still on in the webapis. Say the process was a setTimeout with a timer of 5 seconds and after 1 sec it was moves to the webapis, now after 5 seconds the process ends so what do you think the webapis will let the process enter again in the call stack or what, obviously if it does so then we don't know what is there in callstack that can be disturbed. For this purpose when the timeout finishes then the callback is sent to the callback queue or task queue and then sequentially sends back to the callback stack with the help of the event loop. So what is event loop now, it is kind of a manager between the task queue and the call stack, which keeps check on the call stack and if it is empty then it pushes the first element in the task queue to the callstack.

Yes that is true that the JavaScript is single threaded and runs only one thing at a time, but this is all beacause of the browser that provide us with the extensions that work more like threads and allow JavaScript to work like multiple threaded despite of the fact that it is single-threaded. When we go through this entire process of the event loop we will see that setTimeout() is not the garunteed time of execution rather it is the minimun possible time of execution of the code. Also when all the above process goes on then the JavaScript blocks the rendering on the screen that is when the data is in the callstack at that time the rendering queues runs with the screen refresh rate like 16 milliseconds, this way render is given high priority then the callbacks. This is the reason why all the other UI elements are blocked when the JavaScript engine is in process. The concept of blocking came in picture after this rendering and callstack thing interacted.

In order to avoid this blocking and maintiang the good UI/UX experience on the website it is always advisable not to put any slow code in flow beacuse it make the callstack busy and all the callback quere and event loop have to manage all the things, leading to slow rendering on the screen. The image below compiles the entire process of the Javascript processing to browser and JIT compliation.

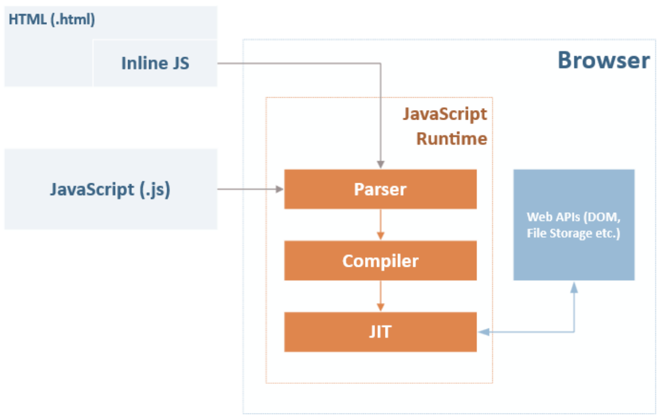


Figure 2.10: Just-in-time compilation (JIT)

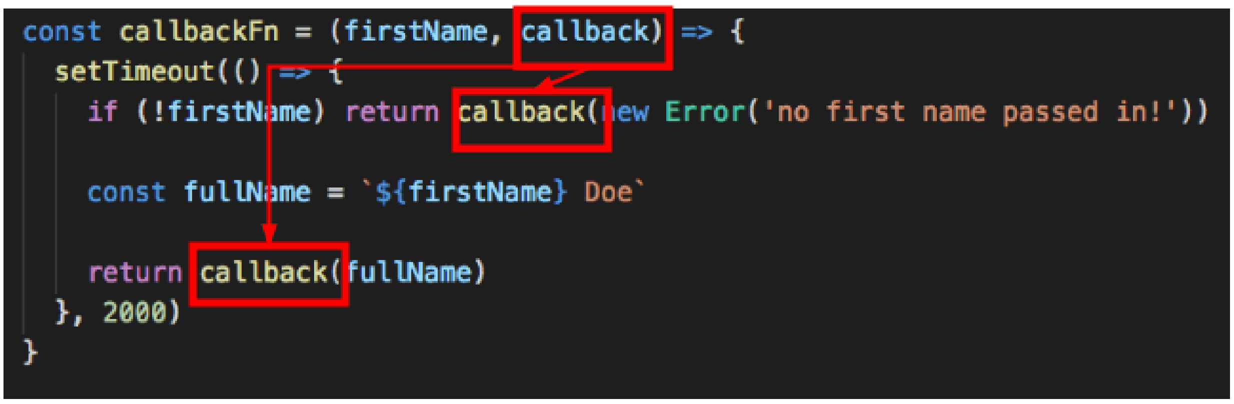
2.4 Callback, Promises and Observables

Callbacks, promises and observables are most commenly used and heard word when onw wants to go for asynchronous function call in order to make better UI/UX experience.[8] All the three are used for increasing the performance of the code and maintain a better user experience as the user don't have to wait for the network request to end which can both be a success or failure, that is known once the response is received. So what are these terms and why we have so much blogs and research done on this topic and even now after so many years of web development, this topis is still a hot topic discussed in the community forums. The reasons why these topics is till a hot topic is:

1. These topics directly connect to the user experience amd user experience directly link to the review if the website is good or bad. Obviously, a good web developer always want his/her efforts to pay and he gets good reviews and his work is appreciated.

2. Asynchronicity can directly manage the speed of the website, the performance can degrade if few core freatures that can run parallelly are set synchronous.

These two are the basic reasons, many other reasons link them when the platform grows. Coming to callback functions these are also called high order functions and are most common when we work with JavaScript. Any JavaScript developer uses these functions in their daily routine.



```
const callbackFn = (firstName, callback) => {
  setTimeout(() => {
    if (!firstName) return callback(new Error('no first name passed in!'))

    const fullName = `${firstName} Doe`

    return callback(fullName)
  }, 2000)
}
```

Figure 2.11: Callbacks

The best way to explain callbacks is the code snapshot in the above image. We can see that there are three callback functions, each function called within another function or passed as an argument in the previous function, all done to complete the entire action or process. The purpose of these functions as the name 'CALL+BACK' is to keep the code flow asynchronous after keeping themselves synchronous. It says the function has to call itself back after it has finished executing itself. It is usually done when the developer has to make multiple API hits or multiple network requests in a sequential order. Beacuse if we consider a code with say (starting with the small amount of data) three API hits and as we know the time when these requests are send the data the code runs in sequence but when we make 3 API hits and get the data in an array and display the array on the screen, we will notice that every time the hit is made the order of the data as per the page is different and not in the sequential order, not in the order in which the requests are made. The reason being it is in our hand to send the network request but the response and network latency is all that matters when the response is returned. The reason why, everytime the order of the data receive is changes. Now this is the reason why callbacks being used for asynchronous code flow still cannot satisfy it, So here came the concept of promises. But, before moving on to promises we should know more about callbacks especially callback hell.

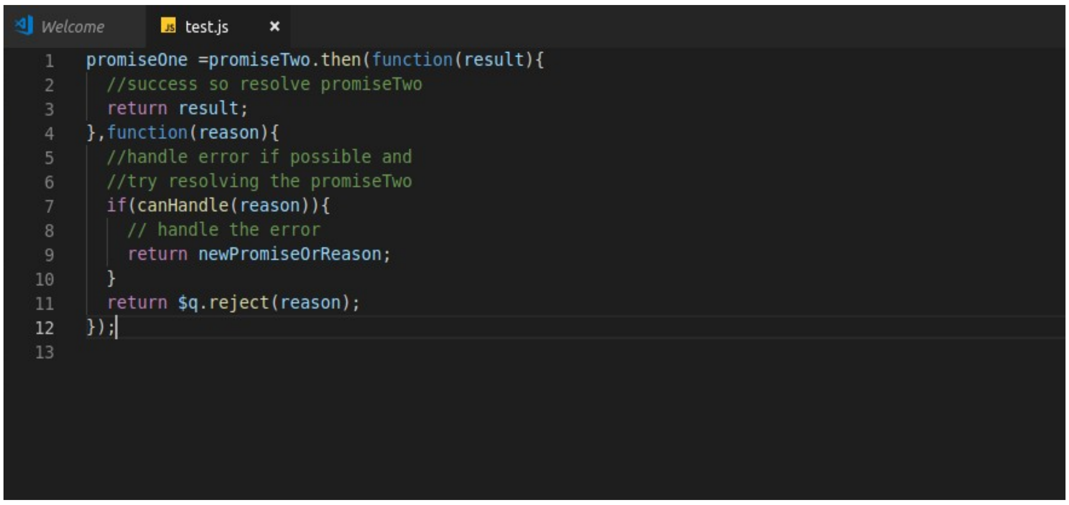
What are callback hells? Callback hell is when the code written by the developer is a callback function but it is presented in such a way that is quite difficult to understand the function and highly difficult to dryrun it, like it feels someone is stuck in deep hell of callbacks. Yes, in JavaScript or rather in all languages the termiologies are kept keeping in mind the way the thing works, so one can clearly relate and underrstand the concept of everything. The major reason of this callback hell is the mindset of the developer that yes, as other languages JavaScript also works in a sequential manner, but actually this is the case with languages like C, python ruby etc., in JavaScript the execution happens does not happen from top-to-bottom. So callbacks are asynchronous which itself means will take some time to return the result. To avoid the callback hells one must use proper coding standards and naming conventions and should use error handlers where there are chance of error.

Another reason why we moves to promises as a good practice is that in callbacks we don’t have any option , that is suppose that the developer has written a callback function for sequential call of 3 API hits, what will happen if one of them fails say the first callback fails that means we won’t get the response of the other two APIs. But promises do make it easy to handle such a case.You won’t believe it but it is actually so, promises are something (basically functions) that handles all the asynchronous operations or maybe even multiple asynchronous operations without making it a callback hell. The way promises are written is really manageable as compared to callbacks.

Therefore the reasons why promises came into picture are more realiable are:

1. These make the code more managed and do not create a callback hell
2. These always keep the handling in picture that is there are always two arguments resolve and reject, if the status is resolved then it calls the next function in the hierarchy, otherwise it checks for what is these if the function returns the reject response and hence catched the exception. That is always prepared with the exception or error handling.
3. With promises the flow of the code works as it visually appears.

These three major benefits of promises covers all the drawbacks of the callbacks and hence angularJS make use of promises for handling the asynchronous calls. Even in AngularJS we have a **$q** constructor which always has a resolver function as it arguments, resolver function further has two arguments namely **resolve** and **reject.** So $q is the ES6 way of creating a promise and make the flow asynchronous. $q makes use of a deferred object and construct new instance of defered using '$q.defer()' . The purpose of this object is to reveal other promise instances as well as other APIs to signal the status of task, whether successfully completed or not. The promises can also be chained using **then** method. Below is the example of promises, that clearly describes the syntax and the concept of promises, the then method and the try-catch way of handling the error of the things, also the $q.reject function the way it takes argument as the reject or reason of the rejection.



```
promiseOne =promiseTwo.then(function(result){
  //success so resolve promiseTwo
  return result;
},function(reason){
  //handle error if possible and
  //try resolving the promiseTwo
  if(canHandle(reason)){
    // handle the error
    return newPromiseOrReason;
  }
  return $q.reject(reason);
});
```

Figure 2.12: Promises

The problem with promises was that the native promises that is the pure promises in the JavaScript can not be cancelled, though with the other third-part libraries it can be cancelled, but no in-built support is being given. Also in case of promises it takes one event at a time, so if we consider like we have to make a search bar which shows the suggestions as soon as we type the key is pressed on the keyboard then say the user types the word in the search bar which says 'Jack' then for each character of the word a promise will be triggered and one can easily imagine if the word length is say ten or fifteen characters, keeping in mind that promises work on single event a time, this will lead to a traffic in to the service there by hitting ten to fifteen requests, being queued up, degrading the performance of the search. All this showed that the promises do not have a use case and hence are only meant for the cases when the event is to be triggered and not cancelled and we have a single event and have time laps between the two consecutive events. For this purpose the observables come into picture.

With the launch of Angular 2.0 came the concept of observables and with that came the confusion and lot of questions, what are these observables, how are they used, okay, if these are used for asynchronous calling, but what is the difference between promises and observables, untill all this was made clear with the help of the tutorial examples in the offical documentation and various community forums. It covered all the setbacks of the promises like promises handle only a single event at a time whereas observables are capable enough to handle multiple or say stream of events parallelly, promises cannot be cancelled untill we make use of any third-party libraries but the subscription, yes the observables are to be subscribed, can be cancelled.

Also in case of observables if we consider the same example of the searchbar, as taken for promises earlier, then the observable is not triggered as the number of characters in the string rather the new in-built functions like debounce time and checking distinct, reduced the triggering of observables to the count of words in a sentence. More or less Observables are quite relatable to the way a user subscribes to any of the favourite youtube channel, when they like to get any kind of notification after the channel uploads any new video (it is like someone else is observing for you to update you about the changes being made and hence named as observables) and when the user don’t want to get any further updates they can easily cancel the subscription of the channel. Now, one could relate and see that yes, these two are relatable. The observables can even be converted to promises if the developer to use in any specifuc case. So observables have various use-cases, all kept in mind when they come in Angular 2. For more on observables one can research on RxJS libraries.

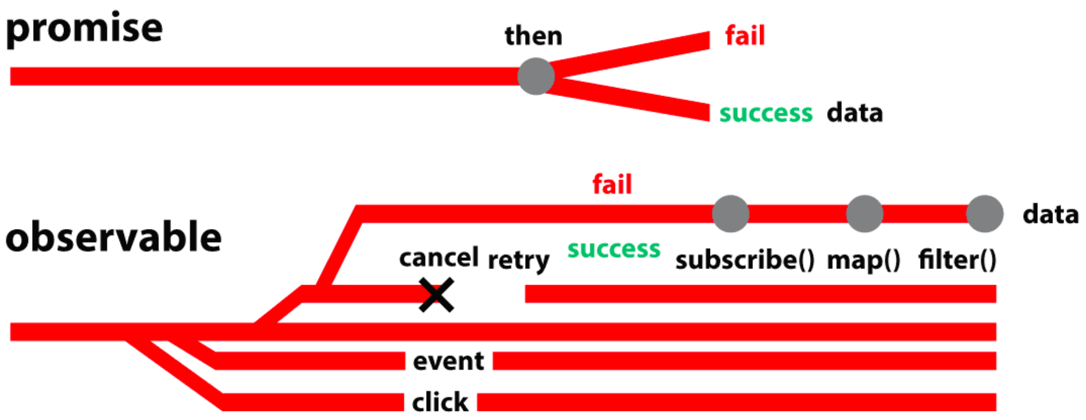


Figure 2.13: Promises vs. Observables

Chapter-3

SYSTEM DEVELOPMENT

3.1 Analysis and Design

Analysis and design are two important and key processes that need to be done before any organisation or any individual need to work of any minor or major task. One need to analyse the environment, the enitre background study is to be made before taking any step towards creation or innovation because innovations means new ideas and anything is termed as ‘new’ if it does not existed earlier, and this term ‘new’ is really relative. One we hear in our daily life that this is a ‘new product’ that may mean the product with new features that noone have ever launched or maybe the same product made easy for the user or the same quality at a low price. This is how the market works everyday thousands of new products are launched, thousands of integrations and like hundreds of services from these product, many fail and very few shine like market role-models, the reason being these products are launched in market after proper market analysis and the team working being it has brilliant minds to make that thought to execution. Also it is important to have competition in the market in order to have better market analysis. Analysis is the root of every creation and every company do analyse the market, the market statistics before investing on any kind of product or service, because in business everything works on return on investment (ROI). Then comes designing, if we know about the software engineering models namely waterfall model, agile, spiral models etc. These models are followed in every industry for the purpose of managing the entire work. Generally in waterfall model we have following few steps as shown in he image all these steps occur in a way that the consecutive new step will not take place untill the previous step is finished that works really well for the industry that don’t have to deal with customer requirements again and again and the first requirements are the final one. After each phase a document is being made to be sure about the decisions made.

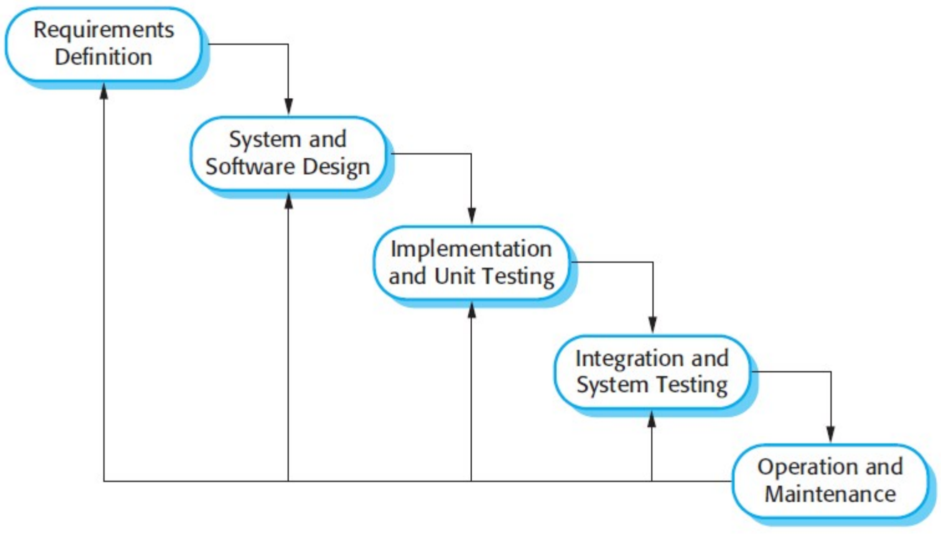

Requirements Definition
System and Software Design
Implementation and Unit Testing
Integration and System Testing
Operation and Maintenance

Figure 3.1: Waterfall model

But the company I worked for is a startup and follows agile model. Agile is a flexible model that can adapt as per the circumstances and usually followed when the nature of the company is more dynamic and less strict. Agile model works considering all software development models, the iterative and incremental approach. Here the increments are not that large as in case of other incremental models and with time small and significant features are added to the product or system based on the customer feedback and then this developed feature undergoes series of steps in the SDL, software developement life cycle model and then are made avaliable for the customer every few weeks or so. This best suits the application where the requirement or the updates are more frequent during the entire development process. Scrum is one of the agile methods, also have agile modeling etc. So agile is a combination of iterative and incremental approach in which small portion of the features is decided and developed then testing and then gradually adding more and more features to it and incrementing the completness of the product.

The startup Jungleworks did all the analysis and come up with the idea to present the entrepreneurs with the entire operating system for the business, which involves everything required on a single platform with integration of products for tacking, instant and actionable communication, online storefront, business analytics, most effective route planners and intelligent CRM dashboard. All these on one platform names Jungleworks. The intelligent CRM dashboard is named as bulbul. After all the analysis the product team starts to design the product. As any product development and launch requires enough planning, discussions, investment, newness and a group of people who can actually execute the thoughts to action. The team of technicians, developers, designer work together day and night to develop this product. So a team is made with a proper hierarchy CEO to CTO to manager to team lead to team members (developers and testers).

When the development starts the very first step is system designing which is the base of converting the product entire theory (generally SRS document) to actual developed product, and the answer to the question ‘How to implement’. System designing and all those models we ever studied in our courses like use case diagram, entity-relation model all those things are done to divide such a multiplex task of system development into sub tasks and activities, which can be performed parallelly to increase the production rate and also the way these parallelly running tasks coordinate with each other properly. While system designing a proper analysis is required for the current situation, the requirements, all the metadata of the proposed system.

This results to the following output:

1. The relationship or say the data schema of the proposed system.
2. The pseudocode for all the program’s module.

3. A prototype of the proposed system in terms of the dummy designs made by the designers, who keep in mind all the UI/UX standards in their mind while designing.

4. The proper hierarchy model in the form of diagrams or graphical representation of the program structure.

That means system design is vast enough, it not only works for the frontend designing but also covers the entire concept of the backend server handling by planning the management of the load on the servers, horizaontal or vertical scaling which one will work the best as per the system requirement. It is all about physical, logical, architectural and detailed designing with the help of data models (such as ER models). A proper documentation (documentation of the system, the code standards, the users and the operations) is done beforehand and the entire idea is to be passed by the authority and then the development begins. For BulBul the server management is done by the devOps team they take care of all the load balancing, scaling, redundancy and other scripting tasks. These is a team of designers that make designs for the front end to develop those designs, the designs can be made interactive and shared online, for better understanding of the developer, via the online applications like zeplin, freemium, InVision, Marvel etc. There is a database adminstration team (the backend team) that takes care of the database and provide backend APIs to the frontend.

3.2 Product Design and Architecture

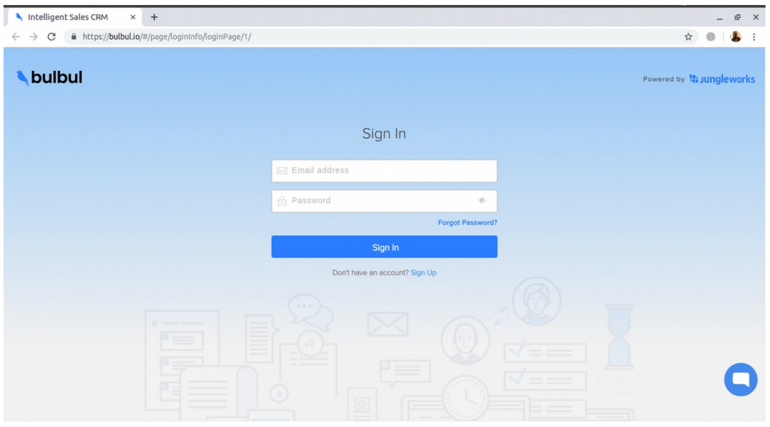


Figure 3.2: CRM login screen

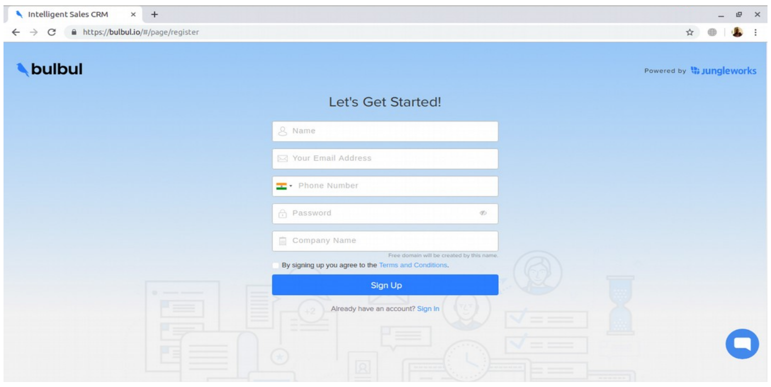
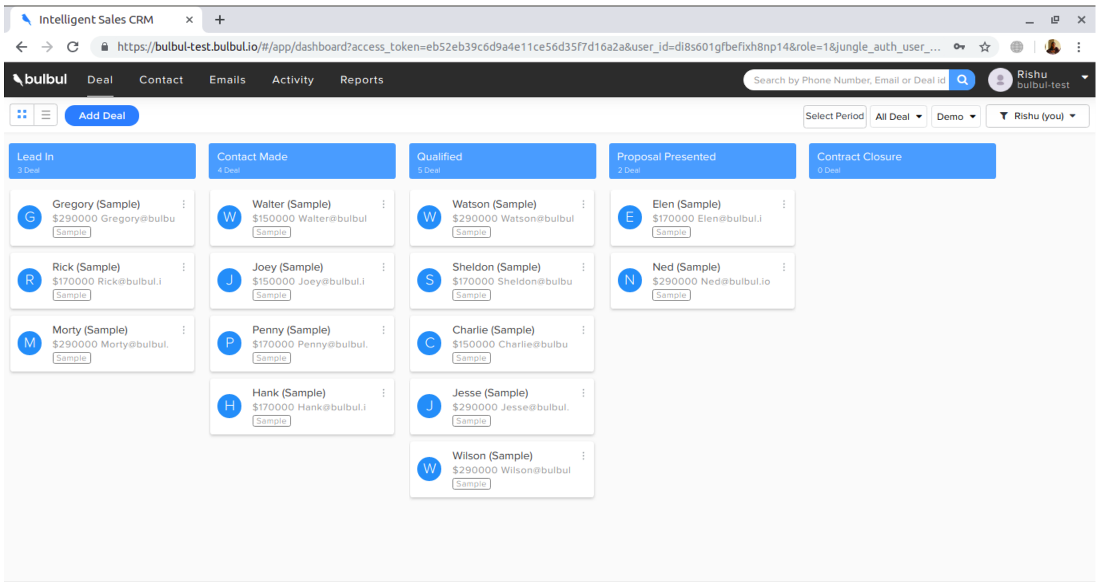

Intelligent Sales CRM
https://bulbul.io/#/page/register
bulbul
Powered by jungleworks
Let's Get Started!
Name
Your Email Address
Phone Number
Password
Company Name
Free domain will be created by this name.
By signing up you agree to the Terms and Conditions.
Sign Up
Already have an account? Sign In

Figure 3.3: CRM Signup screen

Above image are the snapshot of the login ang sign-up screen of bulbul, as all other application products it is also authenticated. The user first has to sign-up to the website in order to use it, the normal procedure that all authenticated websites follow. We can see the round message icon in the snapshot, that is the chat support, bulbul-hippo integration. This chat support is another product of Jungleworks, which is kindof a messaging bot that assists the user in case the user want to talk to any of the customer care executive regarding the product demo then the user can message this via Hippo and the request is forwarded to the sales team and they contact the user via mail or any call, any feasible method. After the user logs in the CRM system then the user will be able so see following screen (as in the following snapshot).


Intelligent Sales CRM
https://bulbul-test.bulbul.io/#/app/dashboard?access_token=eb52eb39c6d9a4e11ce56d35f7d16a2a&user_id=di8s601gfbefixh8np14&role=1&jungle_auth_user_...
bulbul
Deal
Contact
Emails
Activity
Reports
Search by Phone Number, Email or Deal id
Rishu
bulbul-test
Add Deal
Select Period
All Deal
Demo
Rishu (you)

Lead In 3 Deal
- Gregory (Sample) $290000 Gregory@bulbu Sample
- Rick (Sample) $170000 Rick@bulbul.i Sample
- Morty (Sample) $290000 Morty@bulbul. Sample

Contact Made 4 Deal
- Walter (Sample) $150000 Walter@bulbul Sample
- Joey (Sample) $150000 Joey@bulbul.i Sample
- Penny (Sample) $170000 Penny@bulbul. Sample
- Hank (Sample) $170000 Hank@bulbul.i Sample

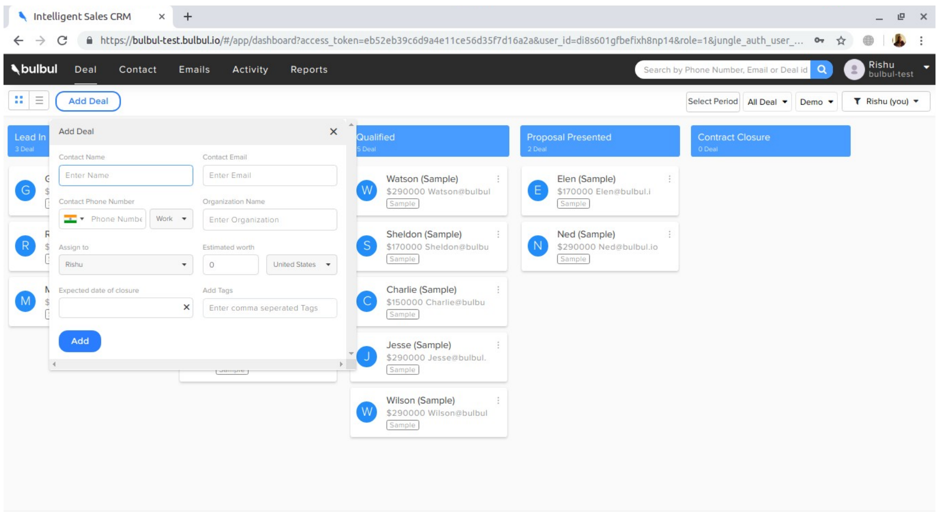
Qualified 5 Deal
- Watson (Sample) $290000 Watson@bulbul Sample
- Sheldon (Sample) $170000 Sheldon@bulbu Sample
- Charlie (Sample) $150000 Charlie@bulbu Sample
- Jesse (Sample) $290000 Jesse@bulbul. Sample
- Wilson (Sample) $290000 Wilson@bulbul Sample

Proposal Presented 2 Deal
- Elen (Sample) $170000 Elen@bulbul.i Sample
- Ned (Sample) $290000 Ned@bulbul.io Sample

Contract Closure 0 Deal

Figure 3.4: Canven View

This screen is basically the deals page where the user can see all the deals he creates, but initally the system provides you with the sample deals so that is becomes easy for the user to understand how the system works.


Intelligent Sales CRM
https://bulbul-test.bulbul.io/#/app/dashboard?access_token=eb52eb39c6d9a4e11ce56d35f7d16a2a&user_id=di8s601gfbefixh8np14&role=1&jungle_auth_user_...
bulbul
Deal
Contact
Emails
Activity
Reports
Search by Phone Number, Email or Deal id
Rishu
bulbul-test
Add Deal
Select Period
All Deal
Demo
Rishu (you)

Add Deal

Contact Name: Enter Name	Contact Email: Enter Email
Contact Phone Number: Phone Number, Work	Organization Name: Enter Organization
Assign to: Rishu	Estimated worth: 0, United States
Expected date of closure	Add Tags: Enter comma seperated Tags

Add

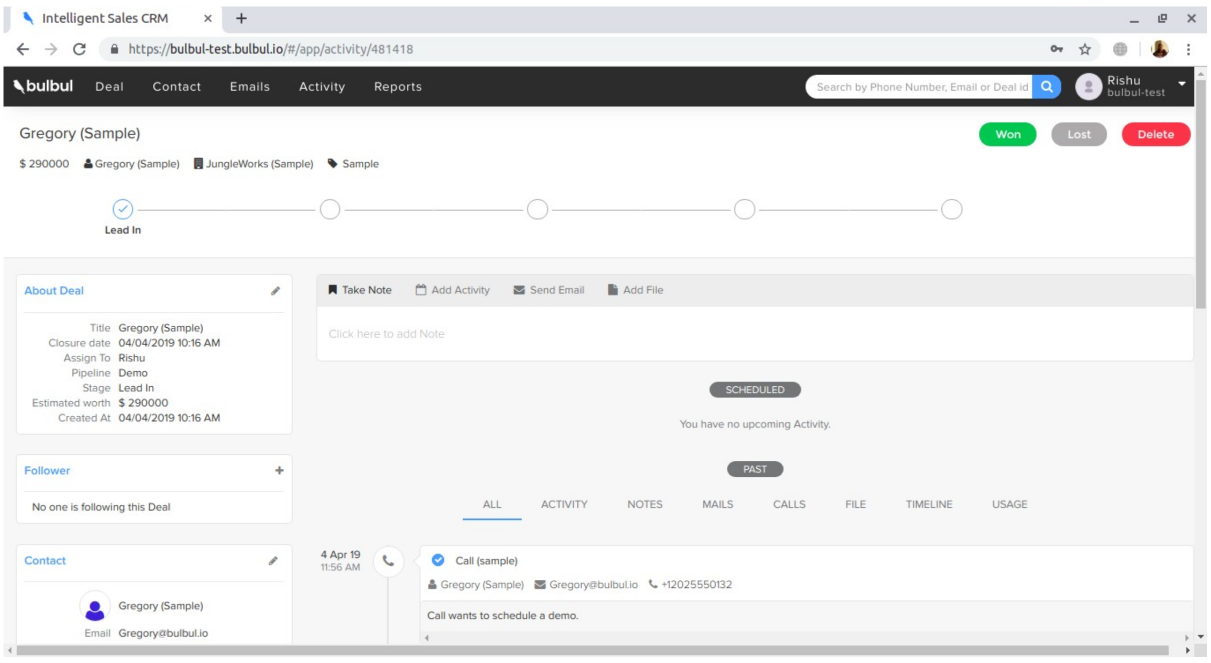
Qualified 5 Deal: Watson (Sample) $290000 Watson@bulbul; Sheldon (Sample) $170000 Sheldon@bulbu; Charlie (Sample) $150000 Charlie@bulbu; Jesse (Sample) $290000 Jesse@bulbul.; Wilson (Sample) $290000 Wilson@bulbul

Proposal Presented 2 Deal: Elen (Sample) $170000 Elen@bulbul.i; Ned (Sample) $290000 Ned@bulbul.io

Contract Closure 0 Deal

Figure 3.5: Add deals

The user can create a new deal by either clicking on ‘Add deal’ option or by importing any csv file (for uploading multiple deals at a time), the deal page has two views namely list view and canven view. The snapshot above was the canven view. After user makes the deal the user can see the deal in both the views but the list view give the detailed view of the deal. Similarly we have contact page that lists all the contacts of the created deals.


Intelligent Sales CRM

https://bulbul-test.bulbul.io/#/app/activity/481418

Gregory (Sample)

$ 290000 Gregory (Sample) JungleWorks (Sample) Sample

Lead In

About Deal

Title	Gregory (Sample)
Closure date	04/04/2019 10:16 AM
Assign To	Rishu
Pipeline	Demo
Stage	Lead In
Estimated worth	$ 290000
Created At	04/04/2019 10:16 AM

Follower

No one is following this Deal

Contact

Gregory (Sample)

Email Gregory@bulbul.io

You have no upcoming Activity.

4 Apr 19
11:56 AM

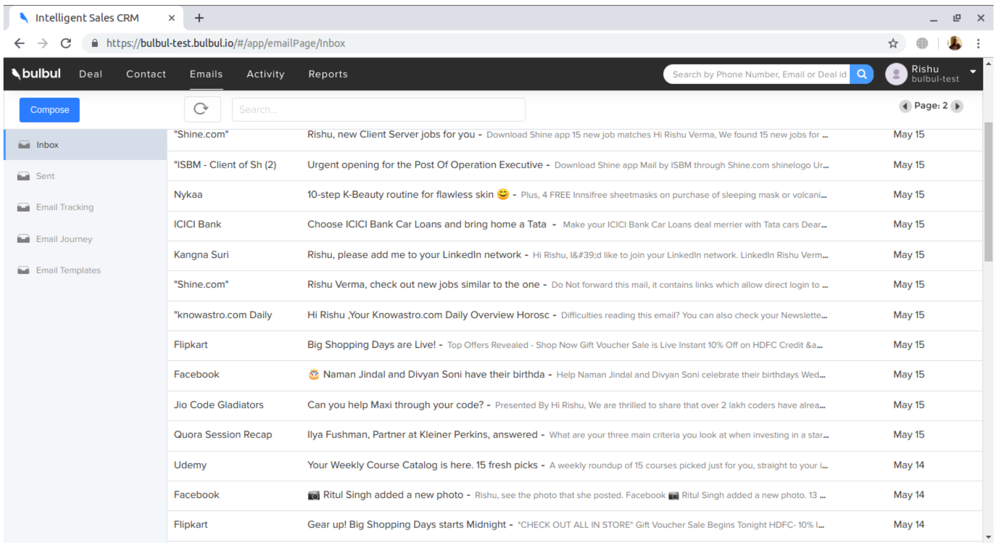
Call (sample)

Gregory (Sample) Gregory@bulbul.io +12025550132

Call wants to schedule a demo.

Figure 3.6: CRM Deal info page

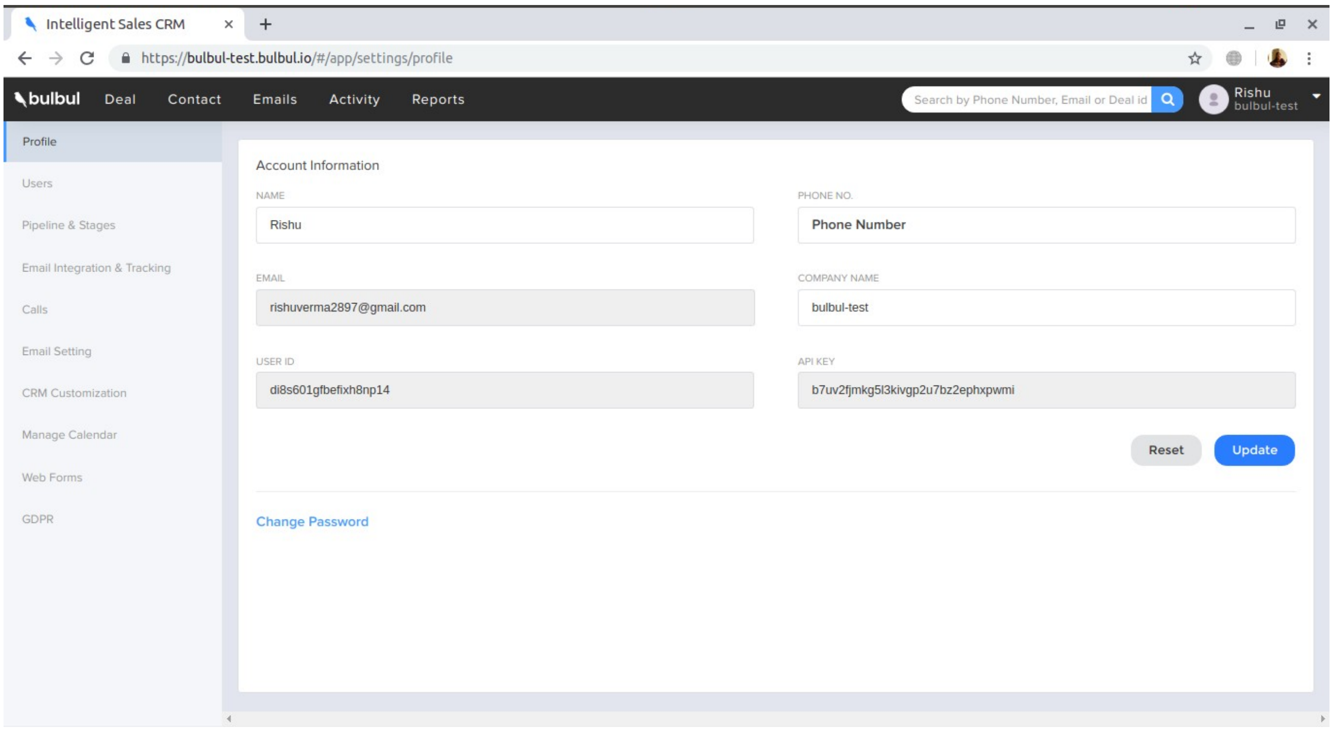
This is the deal info page which gives all the related information about a particular deal, all the contacts, scheduled activities sent mails and even if any kind of document of pdf is used during the deal that can be kept here in the deal info page. At the top right corner of the screen we can three deal status that the user can set about the deal. Below the name we can see the pipeline which shows the stage at which the deal is currently (will be updates by the user) the user creates the pipeline which has various stages (which the user defines while creating the pipeline). One of the best feature that makes Bulbul different from any other market CRM system is the integration with google. The user can directly sync the gmail account by going to the email option in the navigation bar and then can send the mails from here itself, this also allows the user to sync the gmail account, also this sync is a 2-way sync as, when the user sends the email then an activity is created on the deal info page and also it allows the user to sync the google calendar as well, which is again a two way sync. The another interesting feature is the email journey, if the user has synced the email account then the user can create an mail journey for the deal and then the emails are automically send to the respective user.



"Shine.com"	Rishu, new Client Server jobs for you - Download Shine app 15 new job matches Hi Rishu Verma, We found 15 new jobs for ...	May 15
"ISBM - Client of Sh (2)	Urgent opening for the Post Of Operation Executive - Download Shine app Mail by ISBM through Shine.com shinelogo Ur...	May 15
Nykaa	10-step K-Beauty routine for flawless skin 😊 - Plus, 4 FREE Innsifree sheetmasks on purchase of sleeping mask or volcani...	May 15
ICICI Bank	Choose ICICI Bank Car Loans and bring home a Tata - Make your ICICI Bank Car Loans deal merrier with Tata cars Dear...	May 15
Kangna Suri	Rishu, please add me to your LinkedIn network - Hi Rishu, I&#39;d like to join your LinkedIn network. LinkedIn Rishu Verm...	May 15
"Shine.com"	Rishu Verma, check out new jobs similar to the one - Do Not forward this mail, it contains links which allow direct login to ...	May 15
"knowastro.com Daily	Hi Rishu ,Your Knowastro.com Daily Overview Horosc - Difficulties reading this email? You can also check your Newslette...	May 15
Flipkart	Big Shopping Days are Live! - Top Offers Revealed - Shop Now Gift Voucher Sale is Live Instant 10% Off on HDFC Credit &a...	May 15
Facebook	🎂 Naman Jindal and Divyan Soni have their birthda - Help Naman Jindal and Divyan Soni celebrate their birthdays Wed...	May 15
Jio Code Gladiators	Can you help Maxi through your code? - Presented By Hi Rishu, We are thrilled to share that over 2 lakh coders have alrea...	May 15
Quora Session Recap	Ilya Fushman, Partner at Kleiner Perkins, answered - What are your three main criteria you look at when investing in a star...	May 15
Udemy	Your Weekly Course Catalog is here. 15 fresh picks - A weekly roundup of 15 courses picked just for you, straight to your i...	May 14
Facebook	📷 Ritul Singh added a new photo - Rishu, see the photo that she posted. Facebook 📷 Ritul Singh added a new photo. 13 ...	May 14
Flipkart	Gear up! Big Shopping Days starts Midnight - *CHECK OUT ALL IN STORE* Gift Voucher Sale Begins Tonight HDFC- 10% I...	May 14

Figure 3.7: CRM email synced

In the above snapshot it can be seen that the user has synced the mail and now can see the following options as Inbox, sent mails, email tracking, email journey, email template. Email tracking allows the user to keep the track of the mails the user has sent to any of the client or anyone from this portal, it tells when the other person has read the mail (tracked by, if the mail is opened or not) and how many have replied. In email template the user can create the template of their own choice and requirement, template of the mail which is send for the follow-ups by email journey.



Account Information

NAME	PHONE NO.
Rishu	Phone Number
EMAIL	COMPANY NAME
rishuverma2897@gmail.com	bulbul-test
USER ID	API KEY
di8s601gfbefixh8np14	b7uv2fjmkg5l3kivgp2u7bz2ephxpwmi

Figure 3.8: CRM settings

Next we have the settings page that shows all the other options in the sidebar. The profile option shows the profile of the user and all the other basic informations. The next option is users in the CRM provides the current user to add more users so as to work together as a team, in the very beginning there are two roles namely the admin and the member. When any user creates the account by registering on the bulbul's site and then login then the user is named as business owner by the development team. This business owner (if the product is sold and white labelled) has the admin rights, and admin has all the permissions and rights (which we will see shortly). When the business owner invites any other user via a mail invitation then they specify the role of the user being invited and hence the permissions which can be changed only by the business owner.

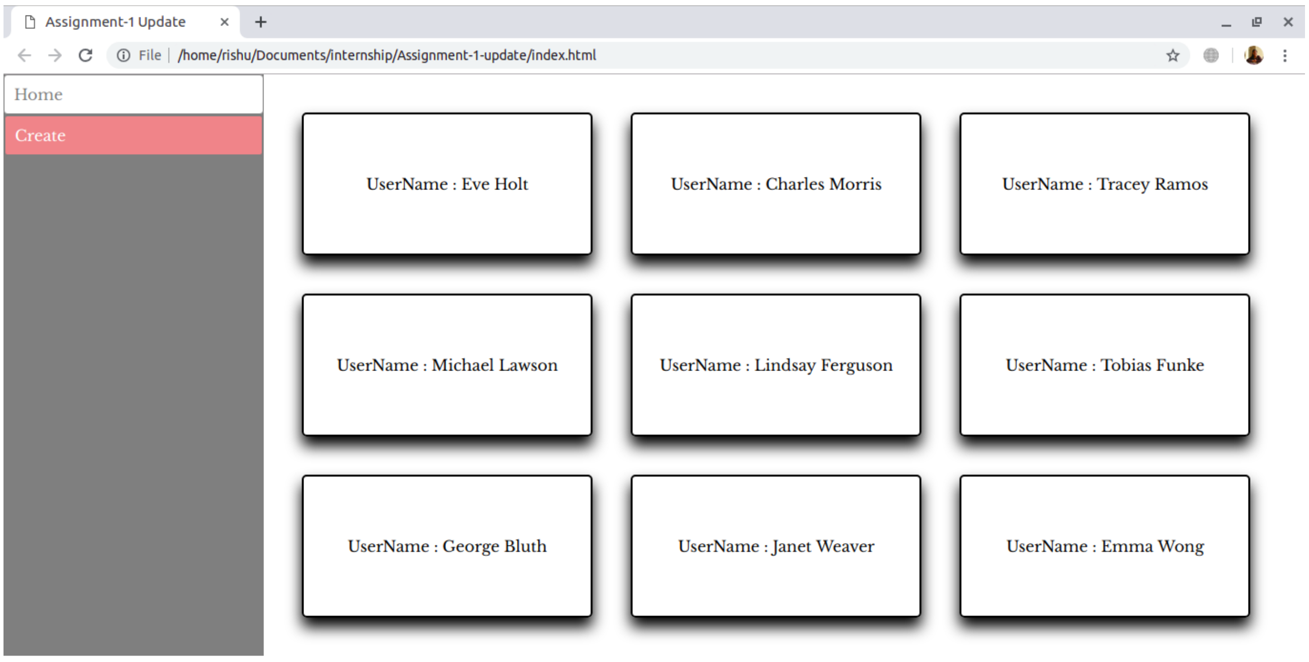
In pipeline and stages the user can create of delete the pipeline as per the permissions. Email integration and testing is for connecting the email to the system and to connect to whistle, which is an extension for the gmail. Calls allow the user to make the calls to the person the user is dealing with via two third parties namely plivo and ring central. In email setting we can set the number of emails that can be send. CRM customization again the most amazing feature in this product. The user can customize the entire product as per their requirement, these customizations are like changing the name of deals to leads and that will be reflected everywhere in the system. Manage calandar allows the user to allow Bulbul to sync the calendar in order to get the follow-up schedule on our google calendar. Web forms allow the user to get the deals from directly from the comment box or form in the website. In this feature the CRM provides it's own APIs to user to use.

As mentioned earlier Jungleworks is more of a business operating system so the system is designed in such a way that if a user registers itself in any of the product of the jungleworks then an entry is made on the main authentication server and for all other products the user don't have to register again rather just log-in to the product portal and use, which is the normal use case of any SAS platform. This is what makes integrations of other products possible in Junglworks, bulbul is integrated to all other jungleworks product, when any user signs up on any of the product of jungleworks then a database entry is made in the main authentication server, and when any other purchase is made on that product say when we talk about yelo-bulbul integration then when an order is made on the yelo's user's profile then a deal is created in Bulbul on the deal info page, when that order is being prepared then that is all shown in the deal timeline and so on. This way the integration of various other products in done.

Chapter-4

RESULT AND PERFORMANCE ANALYSIS:

In this chapter, I have discussed about the features I worked on along with the team and the project I did during the training period in the company. During the training period we were asked to learn few technologies and frameworks from the industry point of view and know about the problems that are usually encountered during the production. So this time the learning was done keeping in mind not that we have to learn the syntax rather we have to focus on the concepts. During training period we were given few assignments which were designed by the mentor in such a way that we can get an idea of the problems that one generally face in production. The first assignment was based on the understanding of HTML, CSS, DOM and JavaScript. We were asked to make a simple page using flex and grid (in CSS, native) and make a asynchronous API hit to reqres for dummy backend and display the blocks on the screen dynamically (keeping in mind that the entire display should be responsive), as in the number of records we get as the response from the API. Following is the snapshot of the very first training assignment.


Assignment-1 Update
File | /home/rishu/Documents/internship/Assignment-1-update/index.html
Home
Create

UserName : Eve Holt

UserName : Charles Morris

UserName : Tracey Ramos

UserName : Michael Lawson

UserName : Lindsay Ferguson

UserName : Tobias Funke

UserName : George Bluth

UserName : Janet Weaver

UserName : Emma Wong

Figure 4.1: Assignment-1 home page

The above snapshot shows the home page where the data is being displayed dynamically after page load. Another purpose of this assignment is to make us familiar with the HTTP requests (get and post) so the ‘create’ link on the sidebar is for the purpose of post request, which is a form that asks name and job post and hits a post request to the reqres.

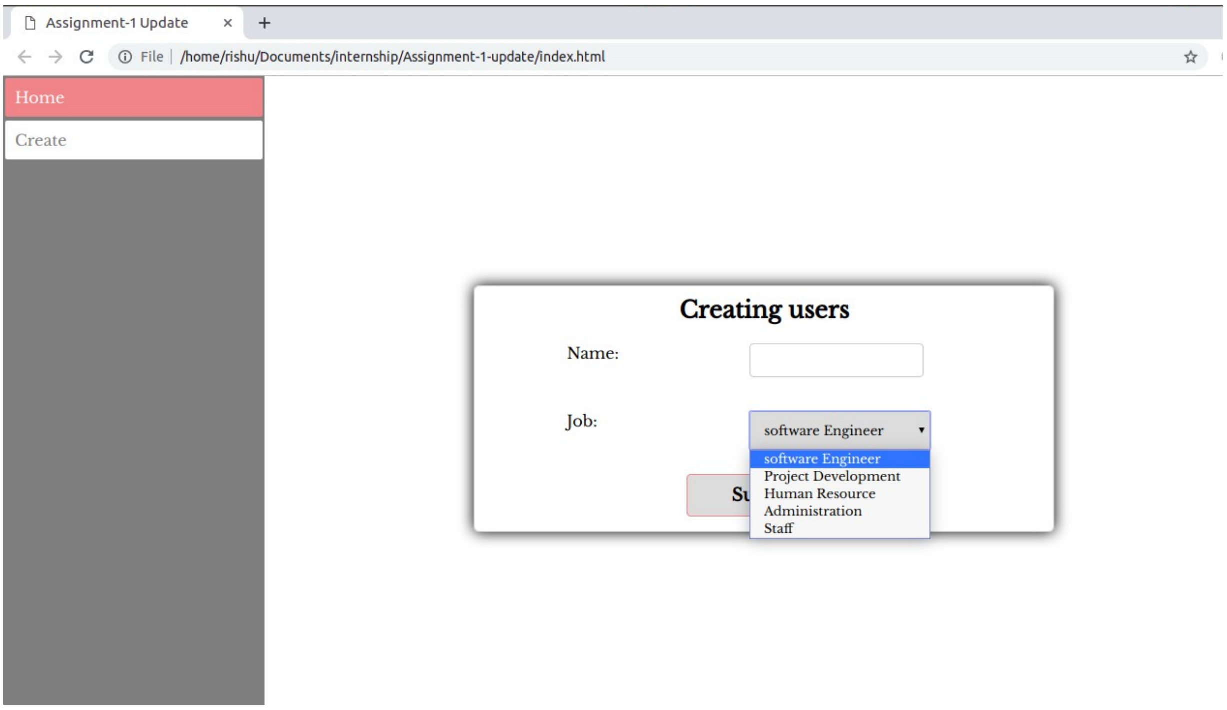


Figure 4.2: Assignment-1 create page

Then the second assignment was given for the understanding of the AngularJS framework. We were asked to implement the tour of heroes which is the most popular tutorial of angular, in angularJS. This was a challenging task as we have to make every this perfect the tour of heroes gif was a sort of InVision for us. We have to understand the concept of $scope, $rootscope, all about how watchers work, what are components, services, factories, the concept of watchers, interpolation, data binding. The problem statement also asked to maintain the file structure with proper commenting and indentation, routing concepts were judged, form validation, directives all these were judged.

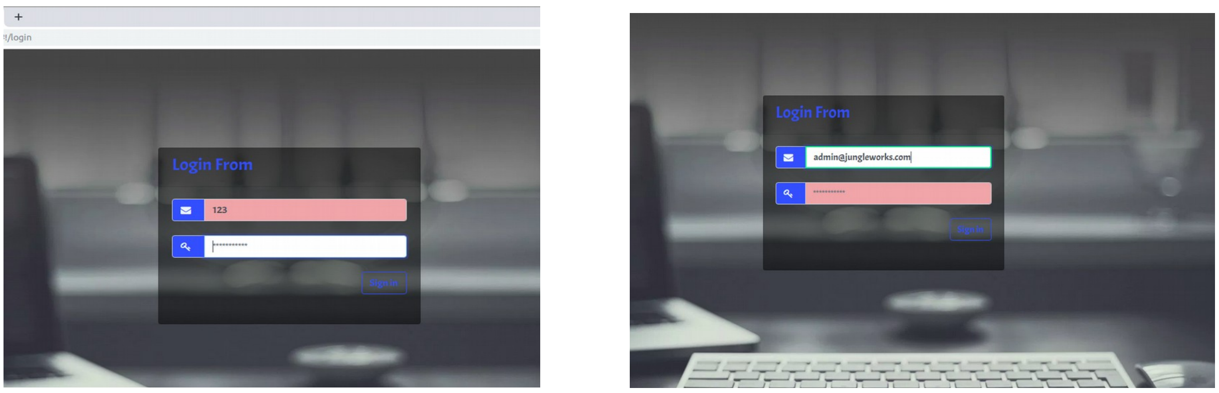
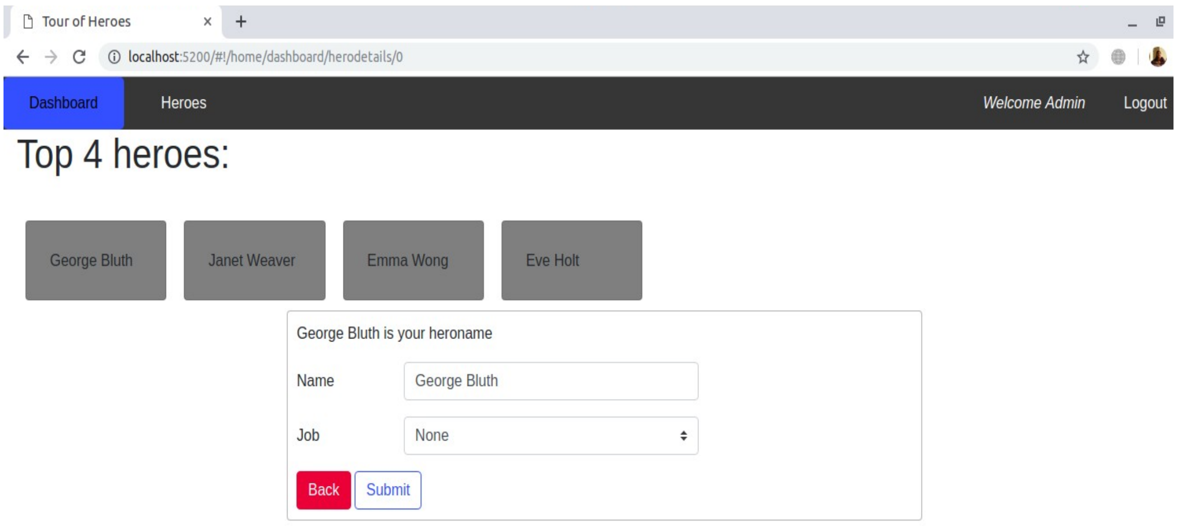


Figure 4.3: Assignment-2 login page with validations

The above snapshots shows the implementation of the Login page and form validations using pristine, dirty, field focus and all other directives.



Top 4 heroes:

George Bluth
Janet Weaver
Emma Wong
Eve Holt

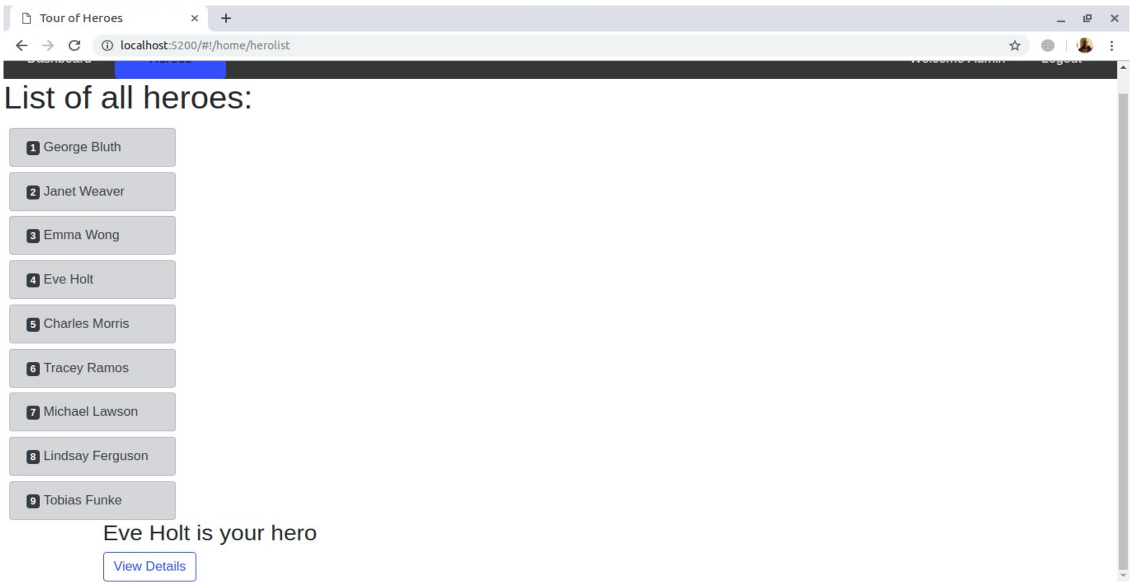
George Bluth is your heroname

Name: George Bluth

Job: None

Figure 4.4: Assignment-2 Admin dashboard

In the admin login we were asked to show the dashboard and home bar in the navbar and in the user login only the list of heroes was to be showed. The list of top 4 heroes displayed on the dashboard is displayed after API hit to reqres and getting the data to an array in a sequential order with the help of promises. The form below the list of heroes is displayed on the click of the specific heroes name with the help of routing (parent-child routing implemented). When the user changes the name in the name field and select submit then the name is updated name is reflected everywhere on the portal. The form validations are applied to this form as well.



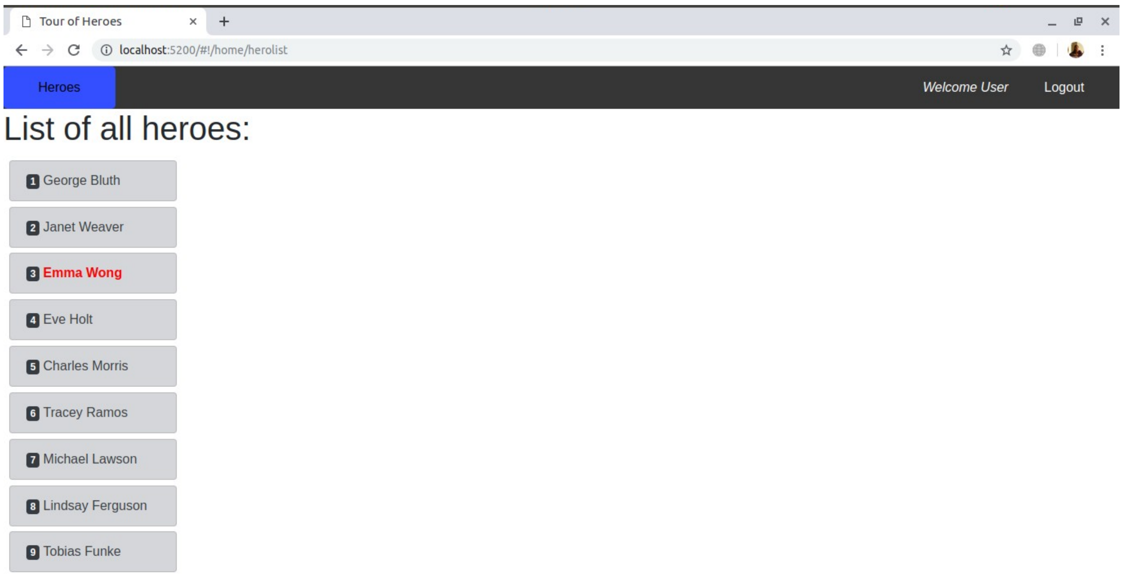
List of all heroes:

1 George Bluth
2 Janet Weaver
3 Emma Wong
4 Eve Holt
5 Charles Morris
6 Tracey Ramos
7 Michael Lawson
8 Lindsay Ferguson
9 Tobias Funke

Eve Holt is your hero

Figure 4.5: Assignment-2 Admin home

Another page is the hero list page where the list of heroes is displayed and when we click on the hero's name then the message is displayed on the screen with the button saying 'View Details' which again redirects to a page with form as displayed in the previous snapshot.


Tour of Heroes

localhost:5200/#!/home/herolist

Heroes

Welcome User

Logout

List of all heroes:

1 George Bluth

2 Janet Weaver

3 Emma Wong

4 Eve Holt

5 Charles Morris

6 Tracey Ramos

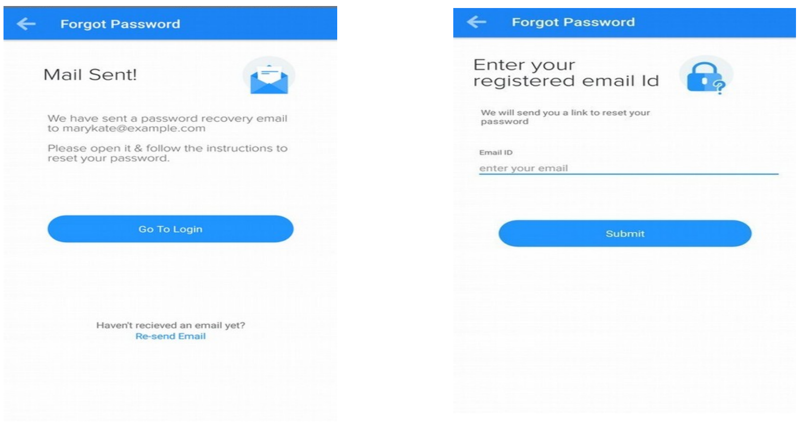
7 Michael Lawson

8 Lindsay Ferguson

9 Tobias Funke

Figure 4.6: Assignment-2 user login home page

In case of the user, only the non-clickable list of users is to be displayed, as per the problem statement given. Then after the training period was done and we were shifted on life product, so I got the opportunity to work on the Intelligent CRM of Jungleworks. I started with working on the hybrid application, of the NativeScript framework. I developed few pages of the application as per the designes provided. The major challenge on this framework for me was that, the framework required knowledge of angular and I was only familiar with AngularJS, so I got my hands dirty on angular and then on NativeScript framework for a week and started working on the hybrid application. Hybrid application as discussed earlier is an application were the codebase is shared and the developer can use one ts (typescript) code to work on web, android and iOS. The snapshot of few views of hybrid app can be sceen as below.


Forgot Password

Mail Sent!

We have sent a password recovery email to marykate@example.com

Please open it & follow the instructions to reset your password.

Go To Login

Haven't recieved an email yet?
Re-send Email

Forgot Password

Enter your registered email Id

We will send you a link to reset your password

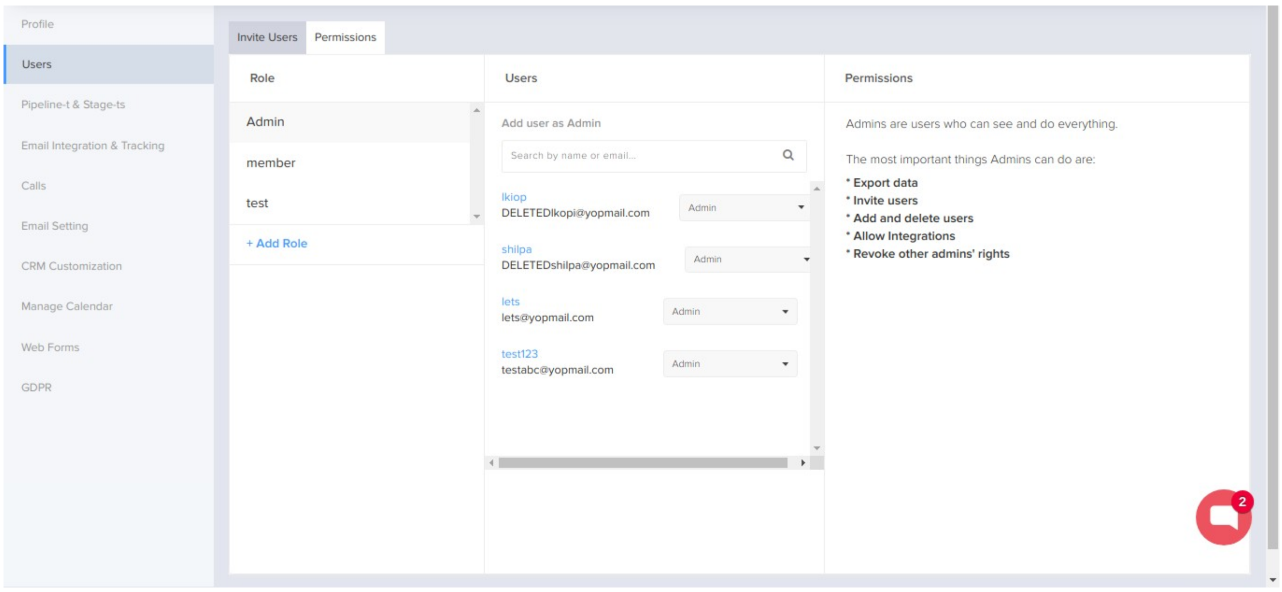
Email ID

enter your email

Submit

Figure 4.7: Hybrid app

Then I started working on the live web application of the CRM, the first feature that I worked on with the team was ACL (Access Control List). It is a new feature which is recently added to the product in which the business owner can create a new role and set the permissions to that particular role, also the user can now invite other users via invitation and assign them the role he created by himself. ACL was a very complex feature expecially it’s unit testing, as the feature is connected to the enite product and every single permission has to be checked for admin, business owner and member.



lkiop	DELETEDlkopi@yopmail.com	Admin
shilpa	DELETEDshilpa@yopmail.com	Admin
lets	lets@yopmail.com	Admin
test123	testabc@yopmail.com	Admin

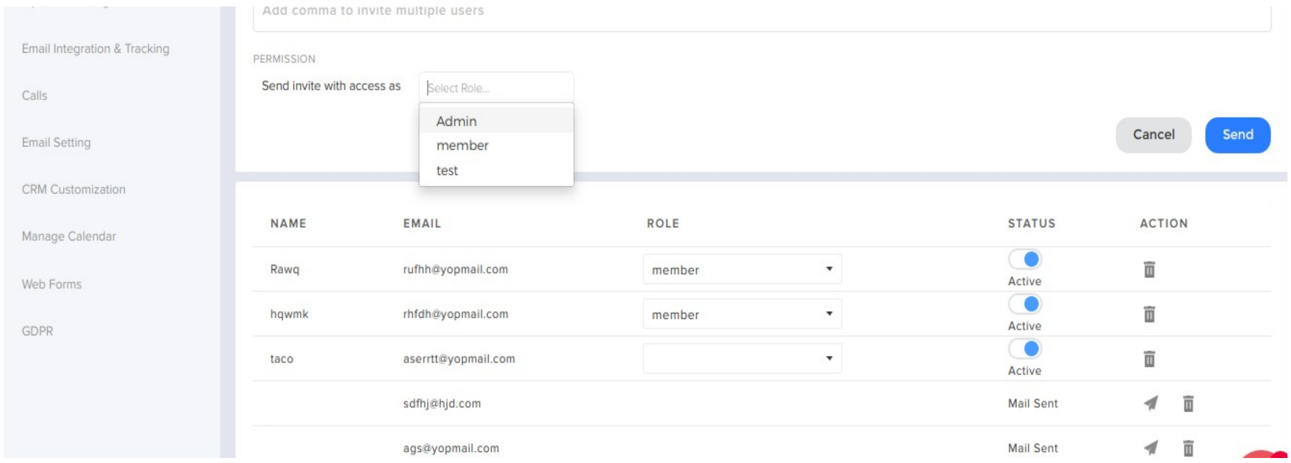
Admins are users who can see and do everything.

The most important things Admins can do are:

* Export data
* Invite users
* Add and delete users
* Allow Integrations
* Revoke other admins' rights

Figure 4.8: CRM-ACL

In the above snapshot we can the the option for the user to add role, when the user clicks on the button then a dialogue box apprears on the screen and then the user enters the role name, the role is created after that the user can change the permissions as per the requirment from the permissions section on the same page (with the help of the toggle button).

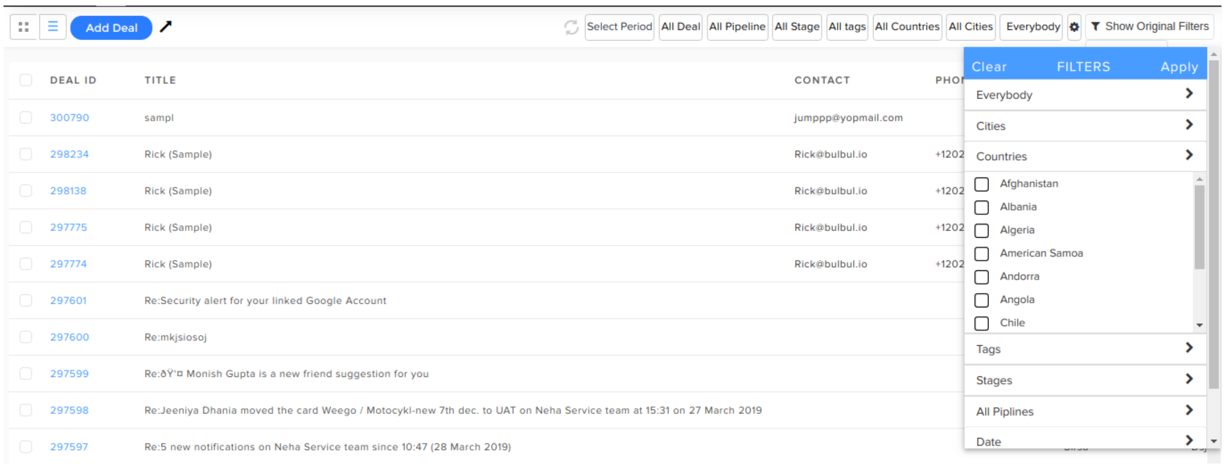


NAME	EMAIL	ROLE	STATUS	ACTION
Rawq	rufhh@yopmail.com	member	Active	
hqwmk	rhfdh@yopmail.com	member	Active	
taco	aserrtt@yopmail.com		Active	
	sdfhj@hjd.com		Mail Sent	
	ags@yopmail.com		Mail Sent	

Figure 4.9: CRM-invite user page

When a new role is added in the permissions section then the same role is also reflected in the list of roles while sending the invitation. One can also edit or delete the role except the admin role (which cannot be deleted) by hovering over the roles in the roles list. During this feature I learned a lot about the product, the concept of roles.The next feature was to make the password more secure by validating it with the pasleys validations, during that I learned how the regular expression works.

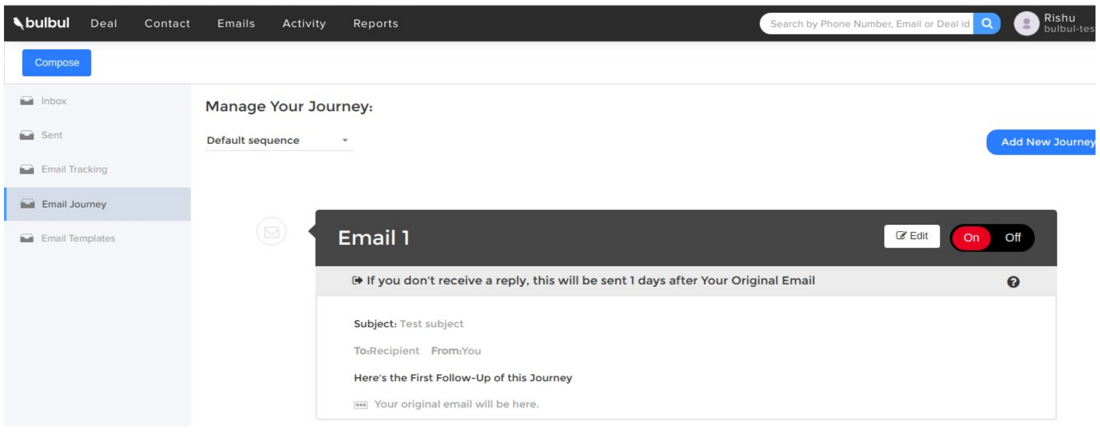
Next we worked on the optimization of the code for filters, filters in the CRM make the work really easy and we have filters on almost all the pages, so we decided to make a custom directive that can fulfill all the requirement of filter anywhere just with a single line of code. The usage of loacal storage was made clear in this feature.



DEAL ID	TITLE	CONTACT	PHO
300790	sampl	jumppp@yopmail.com	
298234	Rick (Sample)	Rick@bulbul.io	+1202
298138	Rick (Sample)	Rick@bulbul.io	+1202
297775	Rick (Sample)	Rick@bulbul.io	+1202
297774	Rick (Sample)	Rick@bulbul.io	+1202
297601	Re:Security alert for your linked Google Account		
297600	Re:mkjsiosoj		
297599	Re:ðŸ'¤ Monish Gupta is a new friend suggestion for you		
297598	Re:Jeeniya Dhania moved the card Weego / Motocykl-new 7th dec. to UAT on Neha Service team at 15:31 on 27 March 2019		
297597	Re:5 new notifications on Neha Service team since 10:47 (28 March 2019)		

Figure 4.10: Filter-directive

Next I worked on Inactivity modal, that is if the screen for few minutes then a modal should appear on the screen. This cleared the file hierarchy the way. Then I worked on the feature to clear multiple to all contacts on the contact info page. Both these work are still on test. Then I worked on adding subject field to the email journey.



Manage Your Journey:

Default sequence

Email 1

If you don't receive a reply, this will be sent 1 days after Your Original Email

Subject: Test subject

To:Recipient From:You

Here's the First Follow-Up of this Journey

Your original email will be here.

Figure 4.11: Email subject

The above snapshot shows the subject field which is seen only when the user wants to enter the subject manually to the follow-ups, else the mails are send with the default previous subject. Next I worked in a team for yelo-bulbul integration, which do not have much frontend work,but few major bugs related to the timeline in deal info page were fixed. Also if the ring central account is active for any user than a RC call setting tab is seen of the sidebar in settings and the user sees all the contact details of the client.

Chapter-5

CONCLUSION AND FUTURE WORK:

5.1 Conclusion

Working on this project in an industry so far has been a great learing experience. The hands on experience on the technology and dealing the real live issues made the learning even more interesting. This type of learning by getting one's hand dirty always counts to your own experiences and research in the matter. The product gave me a deep insight of many unknown requirments of the sales team and how this system of sales work.

The constant review from the sales employees and the market feedback is constantly helping in the project growth with various brilliant minds working on it. The market and the competition in the market is always a motivation for the product. Competition helps to find the gap where one lags and then we try to think more and this leads to innovative ideas leading to growth.

The work environment is quite motivating, the brillance of not so experienced employees motivates one to learn more and more every second and dig deep to concepts for every 'why'. This is the reason which motivates me to learn the technologies and frameworks, which were completely untouched by me and pushing this learning further to optimization and implementation.

It gives me immense proud to present this project report which says all about the hardwork and efforts I put up during my industrial training (internship). All the difficulties, hardships that I faced learning something and the implementing it as per the industrial standards were always a motivating to improve. We as students are really scared of exams and marks, but the actual test is when you develop and thousands of users have to user it and you have to keep all the thousand satisfied with your ideas, with your development. The team is still working passionately and in no ready to settle mode towards achieving the common goal of taking this product a level up with every new feature.

Last, I would like to conclude this report by thanking all the readers for being patient throughout the report and reading it till the end. Thank you so much for sparing your valuable and precious time and reading, evaluating and assessing the final year project report. All the feedbacks will be taken positively by me and will add on to improving me in terms of my performance and surely in imporving me as an individual.

Working in Jungleworks has been an wholly amazing and most passionate working experience. The things I learned from single line of code implementation to git to server deployment added up to my growth. If I compare myself with same me few months back I can feel the difference in my knowledge, skills and attitude.I will always try my best to work hard and learn more and invest my knowledge wherever required to become better.

5.2 Future Work

The product is working on a great pace and various new features are being added. The product manager is really encouraging and enthusiastic, he keeps bringing new features to the team after dealing with the clients and higher authorities. I am really thankful to all the backings involved being this that I got the chance to work in the team. Also, as already a lot of research work and market study has been done on this project so every upcoming feature is designed keeping in mind that the product should be user-friendly, extensible, understandable, reusable.

REFERENCES

[1] Grosskurth, Alan, and Michael W. Godfrey. "A reference architecture for web browsers."*21st IEEE International Conference on Software Maintenance (ICSM'05)*. IEEE, 2005.

[2] Aho, Alfred V., Ravi Sethi, and Jeffrey D. Ullman. "Compilers, principles, techniques."*Addison wesley* 7.8 (1986): 9.

[3] https://www.w3.org/TR/html401/

[4] http://w3c.github.io/html/

[5] https://developer.mozilla.org/en-US/docs/Mozilla/Developer_guide/Build_Instructions

[6] https://github.com/angular/material

[7]https://medium.com/@earthtojhuang/what-happens-when-you-type-a-url-in-a-browser-and-hit-enter-68942c052a6c

[8] https://github.com/getify/You-Dont-Know-JS

[9] https://angularjs.org/

[10] https://angular.io/

[11] https://docs.nativescript.org/angular/start/introduction

[12] https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web