## ACKNOWLEDGEMENT

I am incredibly grateful to my internship supervisor, Anirudh Bhalala, for their invaluable guidance and support throughout my internship. From the moment I started, Anirudh Bhalala took the time to get to know me and understand my goals for the internship. They provided me with clear direction and expectations, and were always available to answer my questions and provide valuable feedback.

Throughout the internship, Anirudh Bhalala provided me with invaluable insights and advice that helped me to grow as a professional. Their constructive feedback helped me to improve my skills and approach to my tasks, and their encouragement kept me motivated and focused. I am deeply thankful for Anirudh Bhalala’s time and effort, and for their commitment to my success.

## ABSTRACT

In the 30 days’ summer training I had studied about WEB DEVELOPMENT Front and Backend. I chose this training because it helps to develop dynamic web pages, and it is useful for my career in Information technology Industry. Under the I learned various new techniques of building websites from basics to advanced which is the very foundation of big problems solved at various levels in the Industry. Apart from that I learned to change the functionalities of complex web pages & make them more efficient by managing lines of code & using the latest technologies. at the end I developed a project called YelpCamp A node.js application which helped me solidify my understanding of all the latest tools and techniques learnt .

Table of Contents

[ACKNOWLEDGEMENT I](#_Toc142474366)

[ABSTRACT II](#_Toc142474367)

[1 INTRODUCTION 1](#_Toc142474368)

[2: TOOLS AND TECHNOLOGIES USED 3](#_Toc142474369)

[2.1: HARDWARE REQUIREMENTS 3](#_Toc142474370)

[2.2 : SOFTWARE REQUIREMENTS 3](#_Toc142474371)

[2.3 : JAVASCRIPT 3](#_Toc142474372)

[2.4 :VS CODE 4](#_Toc142474373)

[3 TECHNICAL CONTENTS 5](#_Toc142474374)

[4 INTERNSHIP CONTENTS 7](#_Toc142474375)

[6 SNAPSHOTS 17](#_Toc142474376)

[7 SUMMARY 20](#_Toc142474377)

[CONCLUSIONS & FUTURE SCOPE 21](#_Toc142474378)

1 INTRODUCTION

Writing a thousand lines of code and turning that into a website is one of the creative and complicated things for web developers. If you get excited seeing a lot of beautiful websites and thinking to try your hands on it then we need to open your eyes telling some important things that you should know as a web developer. Creating a website that gets a lot of users’ attention is not just about learning various programming languages, you also need to learn some other concepts like DevTools, data formats, testing, APIs, authentication and a lot of stuff like that once you will dig yourself into this field.

The most important skill or knowledge every developer should learn first is these three basic building blocks i.e. HTML, CSS, and JavaScript. We will be using HTML and CSS in frontend for interfaces. Just right click on our web browser and then selecting view page source option does it. We can find the structure of your website where a lot of HTML tags are used for different purposes.

CSS is also used in the frontend that decides the style, design, layout and how HTML elements need to be displayed on the screen.

Javascript is high in demand nowadays and it is basically responsible for making our HTML pages dynamic and interactive. Javascript also comes with a variety of languages like to make our website more interactive. If we’re gonna specialize in javascript like MEAN Stack or MERN stack then we’re gonna deep dive into this language because this one will be our frontend as well as backend language.

You can do a lot of stuff using browser DevTools like debugging, editing HTML elements, editing CSS properties, checking device, tracking javascript error, etc. Every developer should be aware of using different tabs (elements, console, network, etc.) in DevTools to make their work easier and faster. Depends on your browser you can use any DevTools or whatever browser you are using. People generally prefer using Chrome DevTools to develop, test and debug the web application but again it’s the choice of the developer which browser he/she is using to develop the website.

**1.1 Internship Objectives**

This 4 week programme aims to headstart our foray in to the exciting world of web development both front and back end. This course helped me master my basics as well as implement those basics at advanced level throught various projects throughout the course, i could hone my visualisation better and push the boundaries of development mindset. The course covered initial front end development like HTML,CSS,BOOTSTRAP,JS,DOM,Jquery as well as back end in depth including databases and advanced topics like RESTful Routing, data association & authentication . This course is for all the developers who are looking forward to enhance their development journey, and learn skills & technologies which will help them to be a full stack developer faster .

## 

## 2: TOOLS AND TECHNOLOGIES USED

## 2.1: HARDWARE REQUIREMENTS

• Pentium-IV(Processor) Or any processor.

• 256 MB Ram OR Above

• 512 KB Cache Memory

• Hard disk 10 GB or above

• Microsoft Compatible 101 or more Key Board

## 2.2 : SOFTWARE REQUIREMENTS

• Operating System : Windows 95/98/XP with MS-office or above.

• Programming language : JavaScript

• IDE : VS Code

## 2.3 : JAVASCRIPT

JavaScript often abbreviated as JS, is a programming language that conforms to the ECMA Script specification. JavaScript is high-level, often just-in-time compiled, and multi- paradigm. It has curly-bracket syntax, dynamic typing, prototype based, object-orientation, and first-class functions.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behavior, and all major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the document object model (DOM). However, the language itself does not include any input/output(I/O), such as networking, storage, or graphics facilities, as the host environment (usually a web browser) provides those APIs.

## 2.4 : VS CODE

Visual Studio Code is a source-code editor developed by Microsoft for Windows, Linux and macOS.It includes support for debugging, embedded Git control and GitHub, syntax highlighting, intelligent code completion, snippets, and code refactoring. It is highly customizable, allowing users to change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. The source code is free and open source and released under the permissive MIT License. The compiled binaries are freeware and free for private or commercial use.

Visual Studio Code is based on Electron, a framework which is used to deploy Node.js applications for the desktop running on the Blink layout engine. Although it uses the Electron framework, the software does not use Atom and instead employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services).

In the Stack Overflow 2019 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool, with 50.7% of 87,317 respondents claiming to use it.

## 3 TECHNICAL CONTENTS

1) Getting hold on development basics

The course covered various basic Topics like how one should and can follow the path of being a full stack developer from basics to advanced level . A lot of problems were solved based on basic knowledge & it’s implementation rather than just memorizing stuff .

2) HTML & CSS

This includes covering commonly used HTML tags. We can refer to MDN Docs as well quite early in the course as a reference material. Overall this is a small section which gets completed soon. We got skimmed through CSS section. There is lot of ground to cover in CSS. We got to learn about Specificity and Cascade, Selectors, Fonts etc. There were 2 projects to give a practical experience in writing CSS.

3) Bootstrap

This is a huge section where we learn about various Bootstrap(latest version) components such as navs, grids, forms. There is a image gallery project using Bootstrap which was cool.

4) JavaScript

JS lessons were very good. Syntax, Control Flow, Functions, Arrays, Object Basics were taught in great detail. (Almost 8 hrs content in 40 hrs course). This is mainly because whole backend section rests on JS foundations taught in these sections. There are various tiny coding exercises and quiz to hammer home the JS concepts.

5) DOM Manipulation

This was very good and most interesting part of the course. DOM manipulation was taught using vannila JS with methods such as querySelector, querySelectorALL, addEventListner etc.

6) NODE

Introduction to REPL, quick exercises, npm, installing packages etc. npm faker exercise was fun.

7) Express

This begins with a simple express app, package json file, route params, basic exercise. Colt uses 'ejs' or embedded JavaScript to render view templates.

8) Database (MongoDB and Mongoose)

We got a brief introduction to MongoDB, Mongo Shell and Mongoose connection etc. Further about how to integrate mongdb with yelpcamp (Data Persistence) to create various campgrounds, users, comments etc.

## 

## 4 INTERNSHIP CONTENTS

1) Basics & Logic building

• Path to be a full stack developer

• Installation & basics

• VS Code local environment set up

2) Front End

• HTML

• CSS

• JavaScript + jQuery

• Bootstrap

3) Advanced Front End

• SemanticUI

• Advanced DOM Manipulation

4) Back end

• NodeJS

• NPM

• Express JS

• MongoDB

5) Advanced Back end

• REST API

• Database Associations

• Authentication

• Authorization

6) Main Project concepts

• YelpCamp

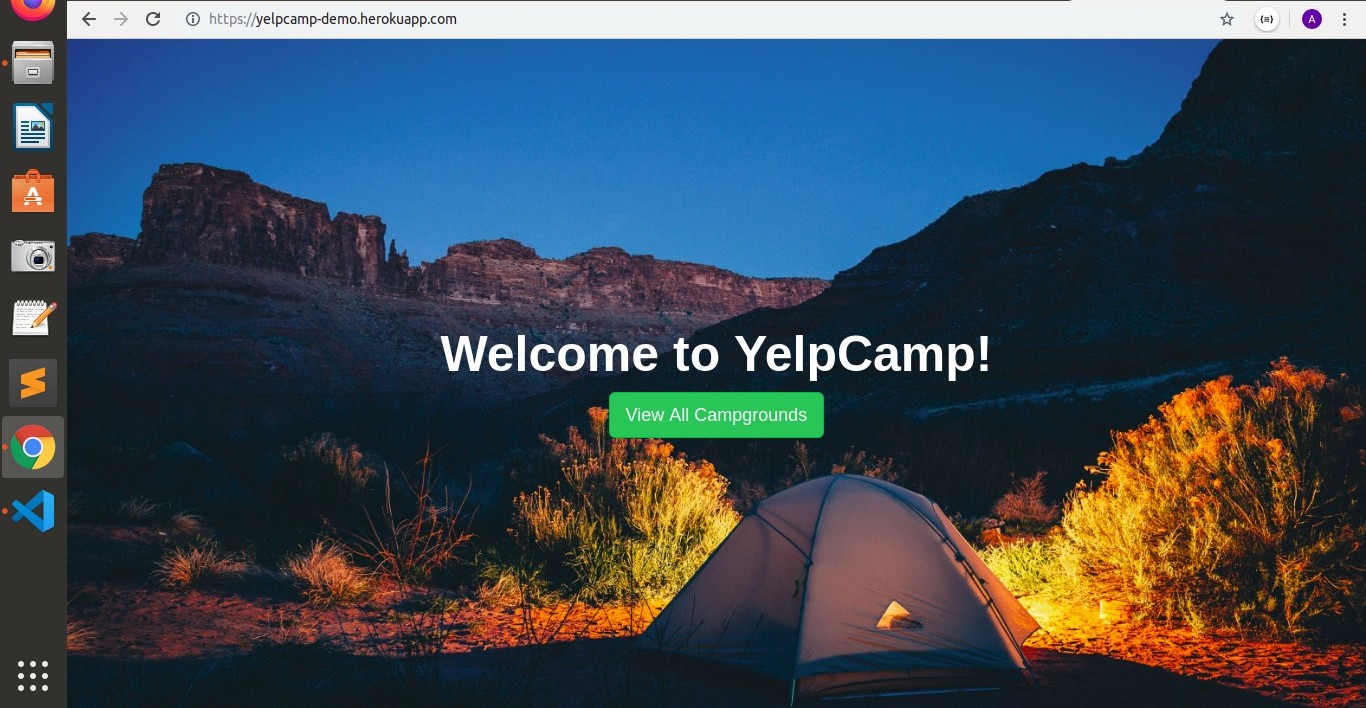
• RESTful Routing

• Maintenance

**YELPCAMP Implementation Using Web Development**

ABSTRACT

Yelpcamp is an application for a fictional start up that allows us to add campgrounds for other users to comment and rate, as view campgrounds added by other users. The project uses various techniques which includes both front end and back end concepts like Node.js, Express.js,and MongoDB . Following is what the home page looks like of our final webpage -



INTRODUCTION

Yelp Camp is a fun application that allows you to view campgrounds with a short description. Once you login or signup you can begin to create your own campgrounds that includes a title, image address and short description.

YelpCamp is a more complex application built from scratch using the following technologies:

On the front-end I have used HTML5, CSS3, JavaScript, Bootstrap for responsive layout, and jQuery. On the back-end I used NodeJS, NPM, ExpressJS, REST, Authentication, Authorization and PassportJS. For datastore I used non-sql MongoDB. The application was developed on Cloud9 IDE and source versioned with GIT. The application is hosted on Heroku servers and MongoLab.

**SCOPES OF WEB DEVELOPMENT**

The scope of web development is finding new meanings of implementation in various fields. As a web developer, We tend to create websites that are more responsive and less compex. That is, we are responsible for behind-the-scenes coding and programming of websites.

On the other hand, we have to maintain a website’s speed, capacity in terms of traffic and overall performance.

The swelling wave of e-Commerce dominance fueling the need for organizations to establish their online presence has led to the emergence of Web Designing as an important organizational function. More and more organizations are recruiting internal Web Designing teams or are employing external agencies to dish out professional and functional websites for them. The basic role of a web designer entails one to employ several skill sets in the production and maintenance of Web Sites. To deliver appropriate quality, they would need to harness knowledge in the following areas:

• Graphic design

• Interface design

• Authoring using standardized code and proprietary software

• User experience design

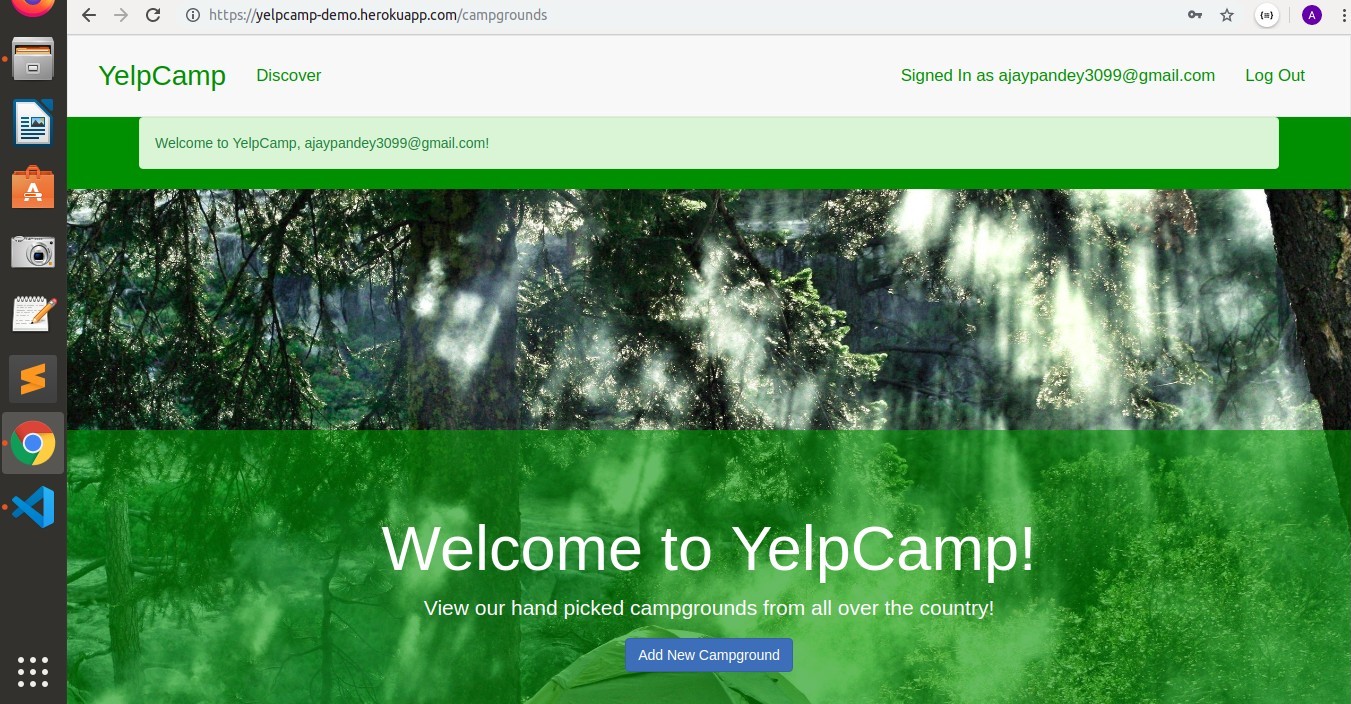
• Search engine optimization

No matter how advanced we go in the field of computer science and technology, Web development will travel with us. Programming languages/platforms etc may change. But the way they are treated by a computer remains . It might take a different way of representation/ideology. But it stays on.

ABOUT THIS PROJECT

In this project that is an implementation of all the concepts learnt in the developer bootcamp, I will be making a node.js application called yelpCamp where a user can visit & discover the campgrounds, check in all the details, make comments about their experience and even they can add their own campground after a simple sign up .

YelpCamp is based on the popular Yelp.com , the difference being it focuses on the campgrounds that any host/owner can post for the users across the globe to check and review. The application is hosted on heroku . The final result we will be trying to obtain here will look like this –



FUNCTIONALITIES

The above final result will consist of a lot of coding including both front and back end tools and technologies and their combined effect, however at the end of the project we will finally be able to achieve following functionalities -

• Everyone can view the camps and reviews without signing up or logging in.

• The user will have to login to edit the campground details or any comments.

• The user can only edit/delete the campgrounds and comments that they have added.

TECHNOLOGIES USED (WITH DESCRIPTION)

• HTML5 - markup language for creating web pages and web applications

• CSS3 - used for describing the presentation of a document written in a markup language

• Bootstrap - free and open-source front-end web framework for designing websites and web applications quickly

• DOM Manipulation - is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document

• Node.js - pen-source, cross-platform JavaScript run-time environment for executing JavaScript code server-side

• Express.js - for building web applications and APIs and connecting middleware

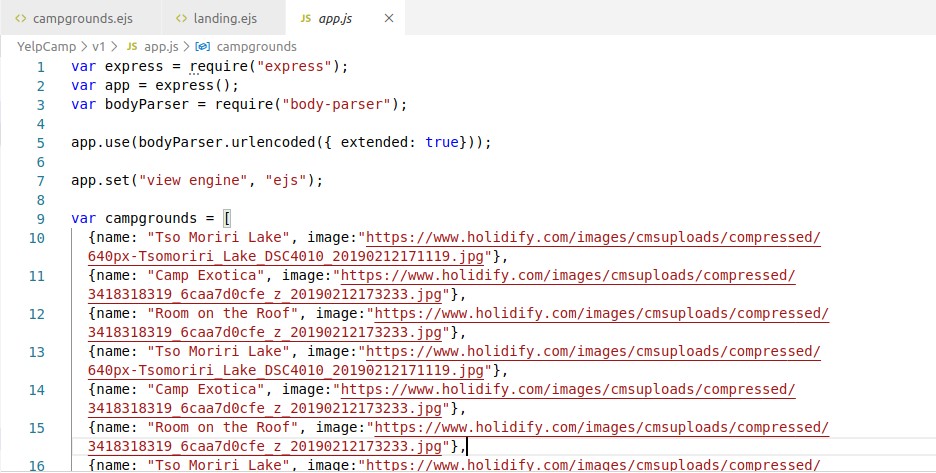
• REST - REST (REpresentational State Transfer) is an architectural style for developing web services

• MongoDB - open-source cross-platform document-oriented NoSQL database program to store details like users info, campgrounds info and comments

• Data Associations - associating user data with the respective campgrounds and comments using reference method

I. Understanding and Implementation

Step 1: Creating JS file & including express



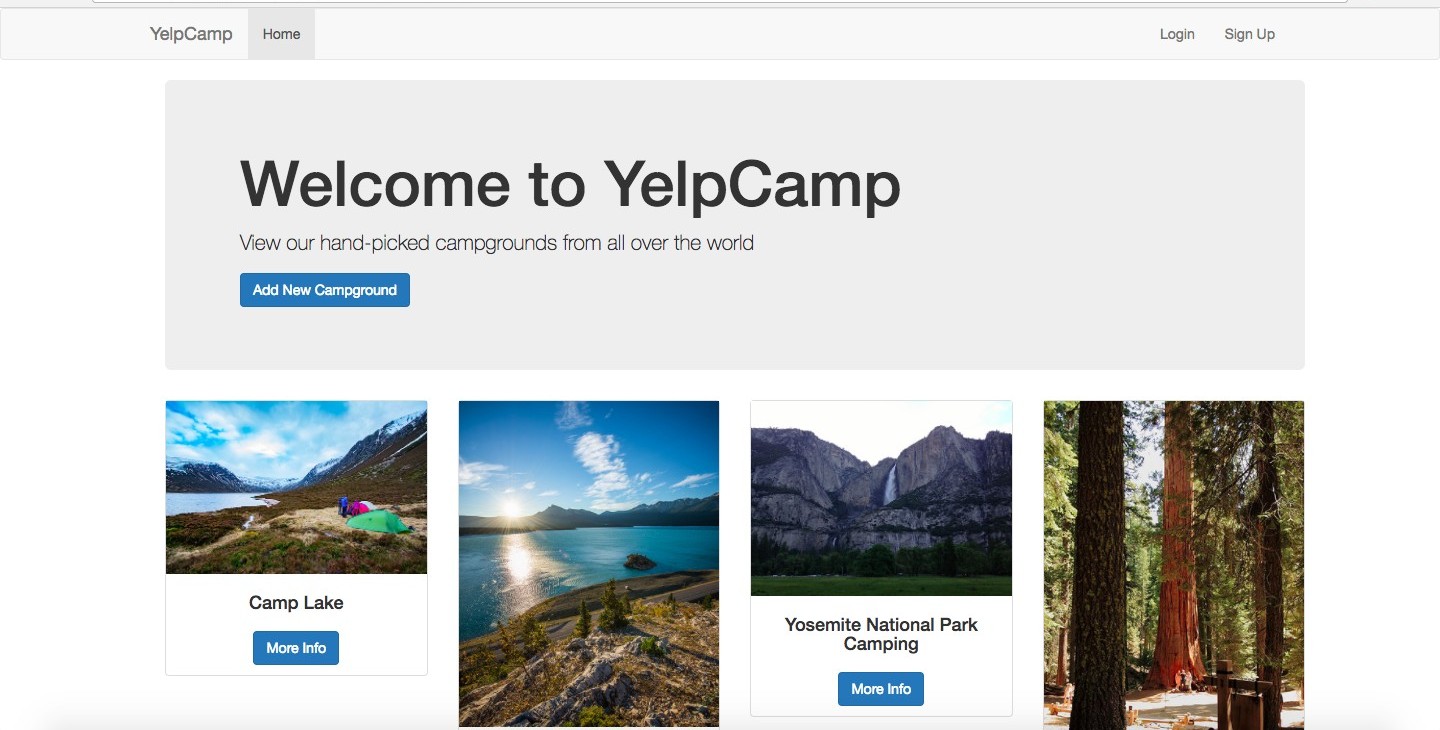
After express is acquired for further using in our project, we use a few methods so that we can avoid our code being less cleaned .

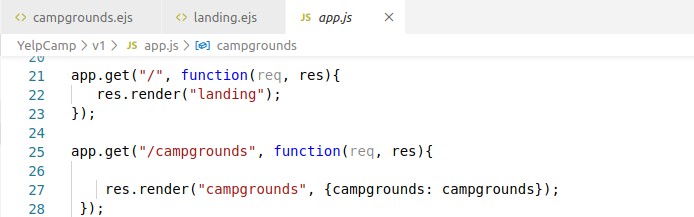
App.set() method lets us use ejs files without mentioning the .ejs extension everytime, the parser is used to access body content & then finally we can add campground images for the basic structure right now, although we are gonna replace them later through our database but for now, this completes our first step . But our app.js file is not completed just yet, we are going to come back everytime and edit this file because this is important for most of the functionalities .

STEP 2: DEFINING PATHS OF REQUESTS

This is the most important and & difficult part of our project, which includes GET & POST requests and affects the functionality of our project, we need to send appropriate response based on the type of request made, analyse the request and finally revert back with an appropriate response the user is looking for . We want to achieve this :

This is going to our very first step that will include setting up our basic structure of the yelpCamp application, & for that we will be adding app.js file that allows us to require express so that we can use it , let me list the app.js file :







STEP 3: DEFINING EJS & INTERACTING WITH JS

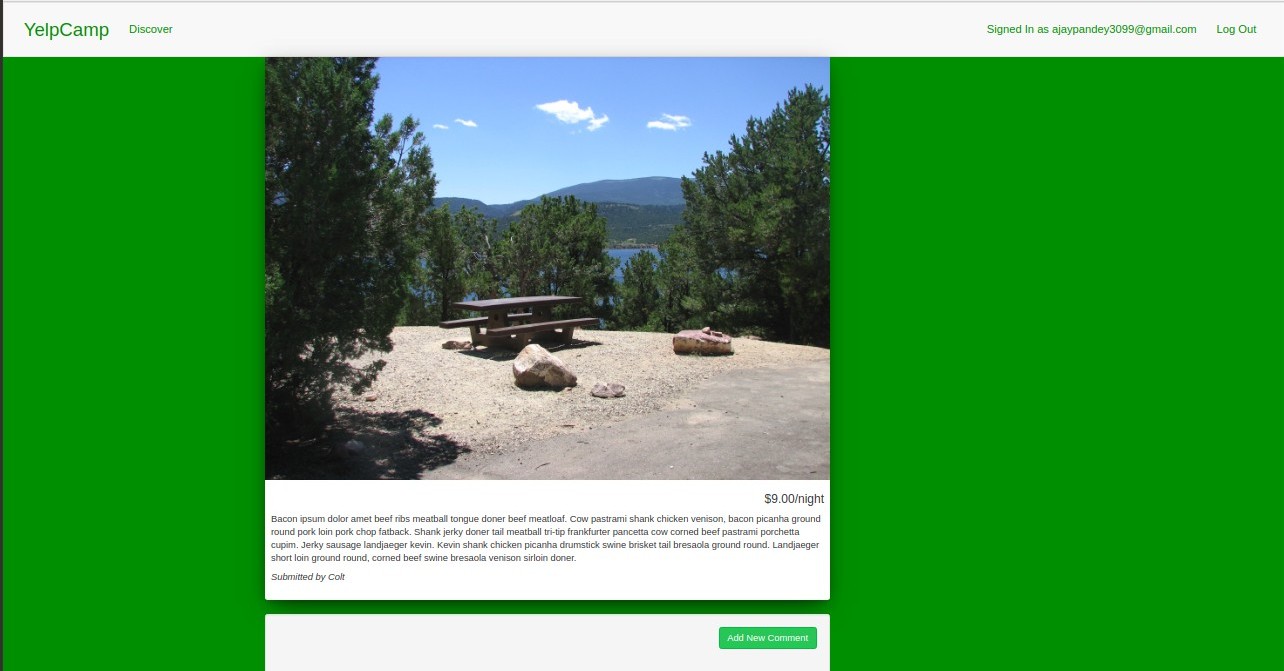
Although simple html files are good in most of the cases but sometimes if we want to make our website look more responsive while keeping our code as clean as possible we need EJS (embedded javascript files), these files help us in interacting between javascript & html & css files .

Till now we have defined paths for our GET requests that we are going to get from our user and now we need to define what functionality these ejs files will achieve when their respective path will be called by user . We are going to look for the request in our ejs file, & then actually render the data present in the ejs through res.render() method .

These headers and footers are used to achieve cleaning up of our code so that we don’t have to write same html again and again what we do here is write it once and then link all the files we want it in . These are also called EJS partials .

STEP 4: DEFINING CAMPGROUNDS + STYLING

After defining the basic paths we now need to define those paths and most important one is the campground.ejs that is when the user visits /campground then we need to manage the request and send response, that response will be looking like this :



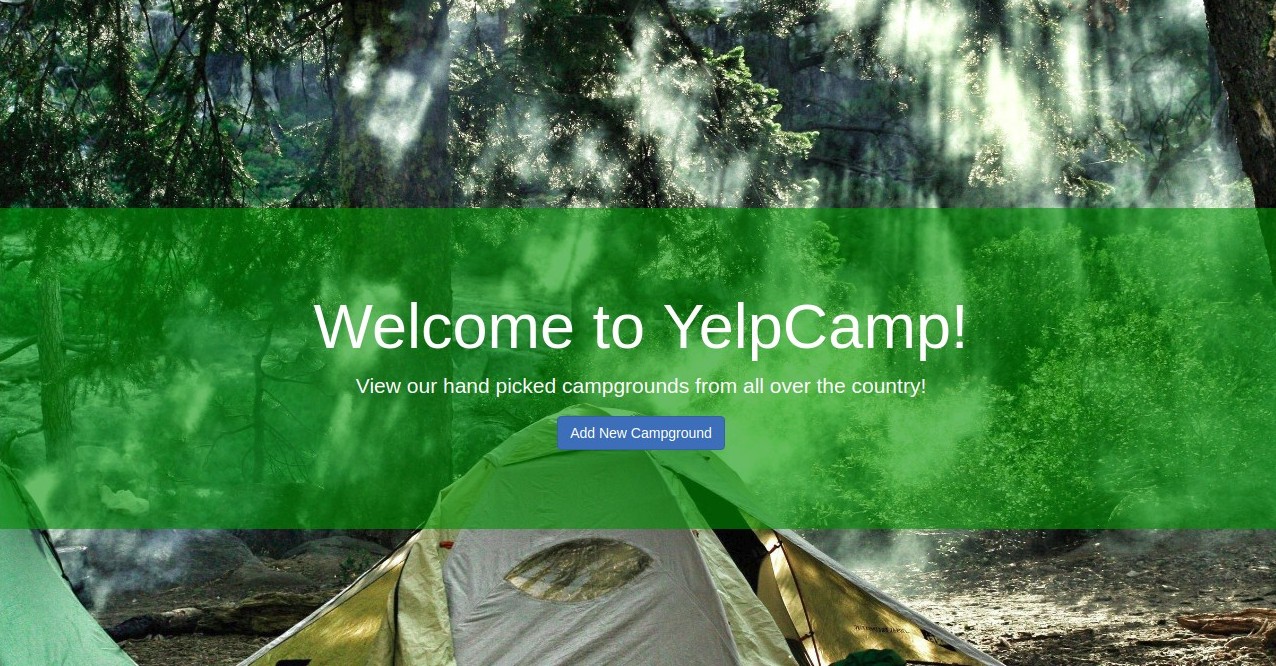
After setting our basic campground.ejs, we need to use navbars, jumbotrons & grid system for making it responsive in mobile or any other device & not only our computer screen, here is the code

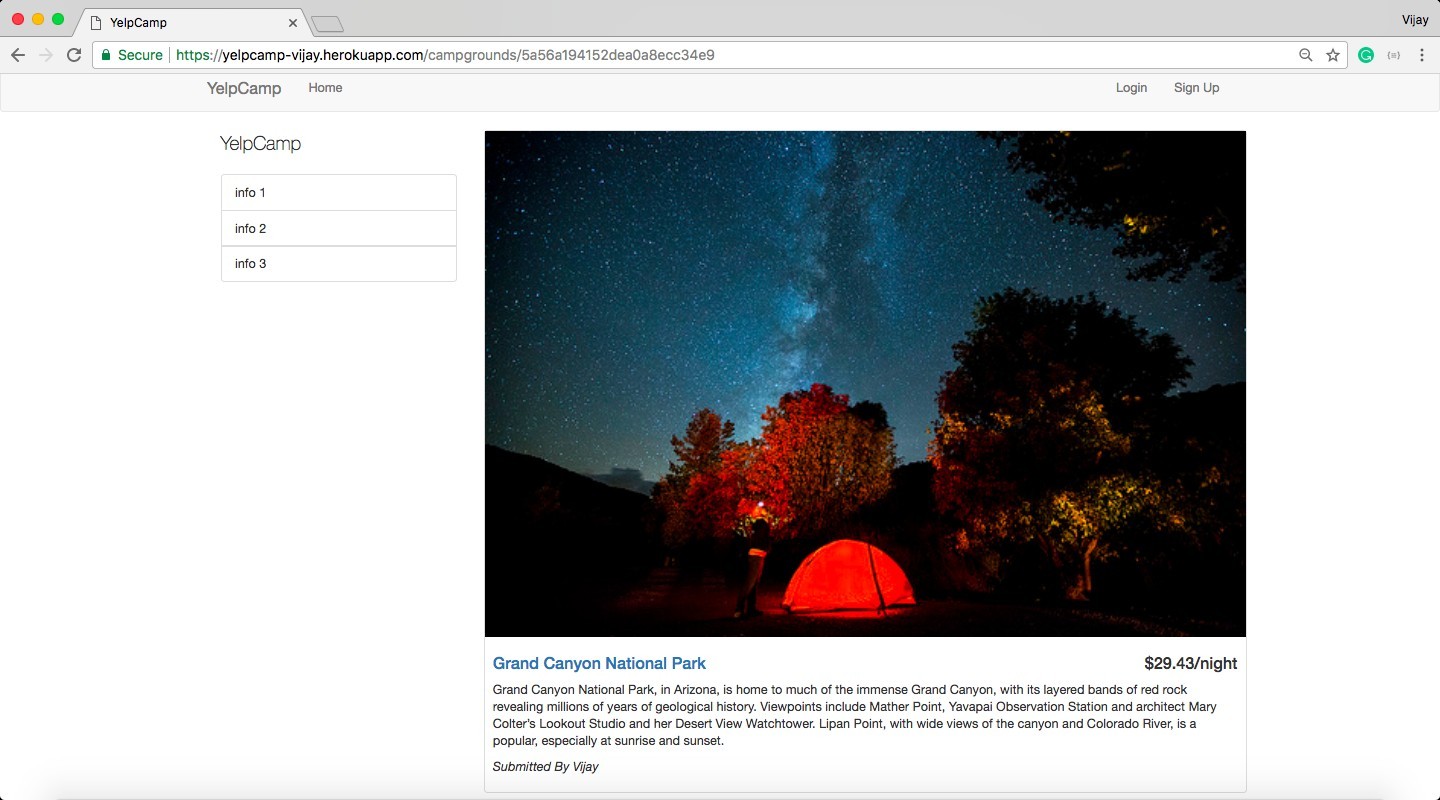
**STEP 5: ADDING A NEW CAMPGROUND**

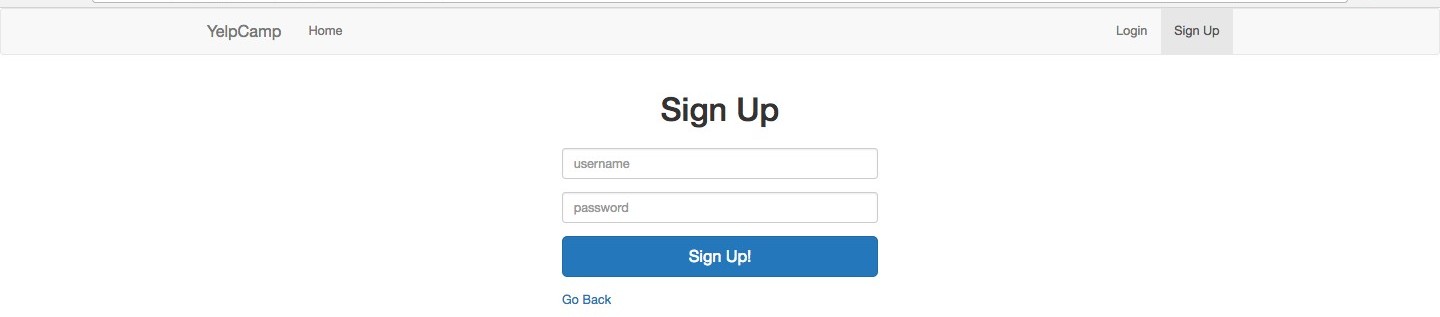
By allowing the user to signup or login we give them access to actually add a campground on their own which however should reach all the fields presented in the earlier app. By the end of this step we want to reach this final result:

## 

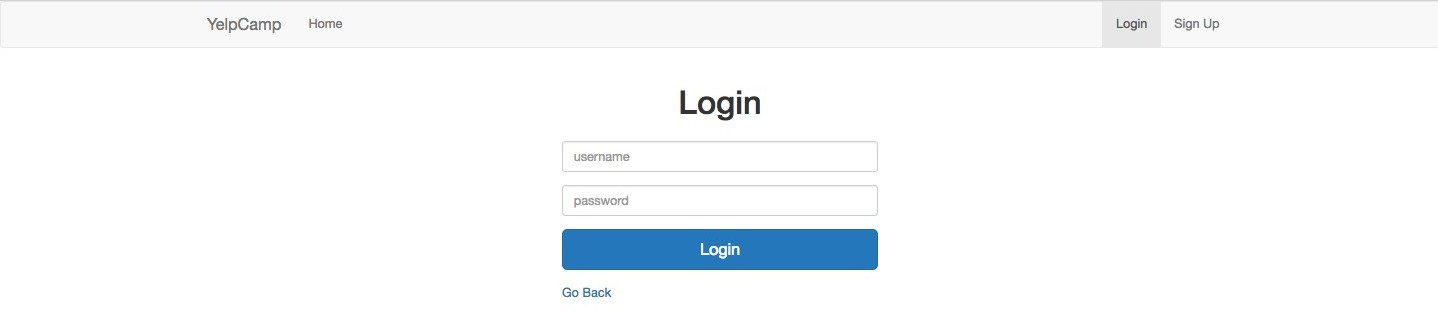
## 6 SNAPSHOTS



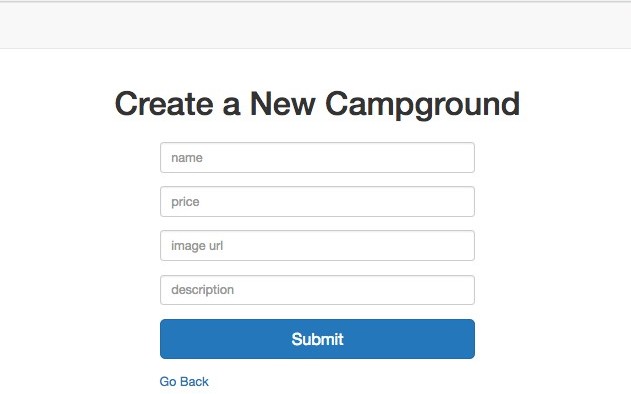




*sign up screen*



*login screen*





## 7 SUMMARY

Overall the training helped me to understand web development & its practical usage in a better way and the project I undertook helped me to solidify these concepts. I learned a lot from the bootcamp developer Anirudh Bhalala, who himself is an ace developer He has been in this thing since past several years & helped various beginners achieve their dream of being a full stack developer I wish to continue working on concepts learnt & futher solidifying them in future too .

## CONCLUSIONS & FUTURE SCOPE

Future scope of Web Development :

Due to the high technologies evolve rapidly, the requirements to the developers change as well. New trends require new solutions along with approaches. If you crave to get insights into what changes in the industry are yet to come in the near future, then observe what the programmers should know and consider in 2020 to remain competitive and be the pros in the field.

1.The need to know more languages and technologies

There was a time not so long ago when it was enough to learn a single programming language like C++, JavaScript, PHP, or Python for a good salary. However, web development is a quite shifting industry and programmers need to have a broad scope of knowledge today to do the work in the best possible manner. Thus, it’s strongly recommended to learn more languages and techniques for the fast and easy switch in case it’s needed. Keep in mind that Python, JavaScript along with WebAssembly will be on the top in 2020.

2.Adaptability

According to the statistics, 51.98 % of web page views worldwide are mobile internet traffic in February 2020. It makes the experts believe that mobile-friendly sites are a must-have for the year and beyond. Of course, in case you’re going to create a trendy website. Today, people don’t want to use the site that is limited in functionality, hard-to-navigate, and is just unappealing to the mobile audience. That’s why web development in the direction will be on- demand. So, pay attention to the adaptability of the site to enhance the customer experience and stay competitive.

3.Less client work more personal projects

Today more and more developers start to think over making their own apps and starting their own projects than working for a client. First and foremost, there is a hope to design the next Facebook, Tweeter or something like those services.

Also working on your own project makes you always learn and learn more, discover new ways and approaches. Thus, you’re updated on the latest trends and newest technologies.

4.The Bottomline

The future is here. As you could notice not all the trends 2020 mentioned above are new. Some of them just keep on dominating in the field that requires lots of tech-savvy pros.

The list of hits is much longer and includes Cybersecurity, Motion UI, Frameworks, Hyper- automation, etc. The range can be enhanced depending on the course of the business.

The market is evolving constantly making the web developers be aware of the trends, changes and new solutions appeared. It’s a challenge, but along with that, it’s a driver for progress.