**A PROJECT REPORT**

**ON**

**YELP CAMP**

**For the partial fulfilment for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**Submitted By**

**Shivam Baghel (1901920100269)**

**Tarun Bhardwaj (1901920100297)**

**Under the Supervision of**

**Ms. Asha Rani Mishra**





**G.L. BAJAJ INSTITUTE OF TECHNOLOGY & MANAGEMENT, GREATER NOIDA**

**Affiliated to**

**DR. APJ ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW**

**2021-22**

**Declaration**



We hereby declare that the project work presented in this report entitled “**YELP CAMP**”, in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Computer Science & Engineering, submitted to A.P.J. Abdul Kalam Technical University, Lucknow, is based on my own work carried out at Department of Computer Science & Engineering, G.L. Bajaj Institute of Technology & Management, Greater Noida. The work contained in the report is original and project work reported in this report has not been submitted by me/us for award of any other degree or diploma.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signature: |  |  | Signature: |  |  |
| Name:  TARUN BHARDWAJ |  |  | Name:  SHIVAM  BAGHEL |  |  |
| Roll No:  1901920100297 |  |  | Roll No:  1901920100297 |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Date:**

**Place: Greater Noida**

**Certificate**



This is to certify that the Project report entitled YELP CAMP” done by TARUN BHARDWAJ (1901920100297) and SHIVAM BAGHEL (1901920100297) is an original work carried out by them in Department of Computer Science & Engineering, G.L Bajaj Institute of Technology & Management, Greater Noida under my guidance. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

Date:

Ms. Asha Rani Mishra Dr. Sanjeev Kumar Pippal

Signature of the Supervisor Head of the Department

**Acknowledgement**

The merciful guidance bestowed to us by the almighty made us stick out this project to a successful end. We humbly pray with sincere heart for his guidance to continue forever.

We pay thanks to our project guide Ms. Asha Rani Mishra who has given guidance and light to us during this project. Her versatile knowledge has caused us in the critical times during the span of this project.

We pay special thanks to our Head of Department Dr. Sanjeev Kumar Pippal who has been always present as a support and help us in all possible way during this project.

We also take this opportunity to express our gratitude to all those people who have been directly and indirectly with us during the completion of the project.

We want to thanks our friends who have always encouraged us during this project.

At the last but not least thanks to all the faculty of CSE department who provided valuable suggestions during the period of project.

**TABLE OF CONTENT**



Declaration........................................................................................................ (ii)

Certificate..........................................................................................................(iii)

Acknowledgement............................................................................................(iv)

Abstract........................................................................................................... (v)

List of Figures ………………………………………………………………(vi)

**Chapter 1. Introduction**

* 1. **Problem Definition**
  2. **Project Overview/Specification**

**Chapter 2. System Analysis & Design**

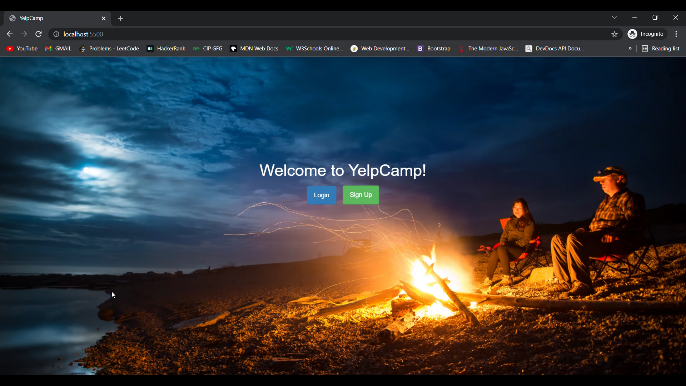
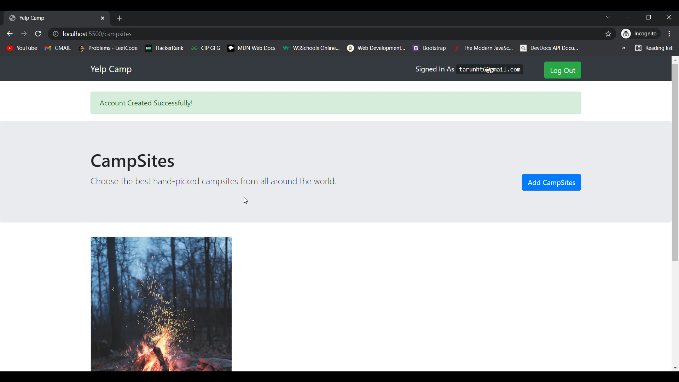
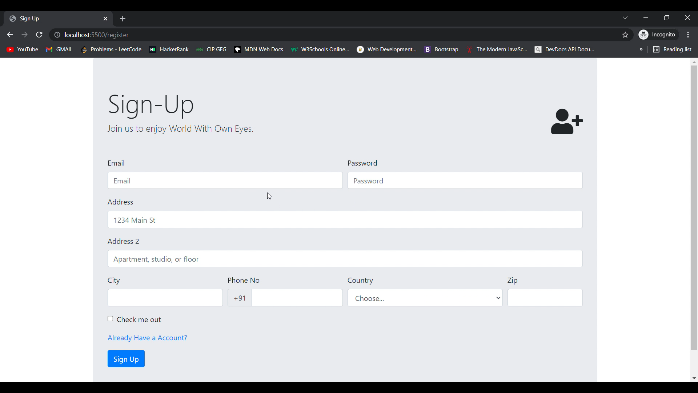
**Chapter 3. Implementation**

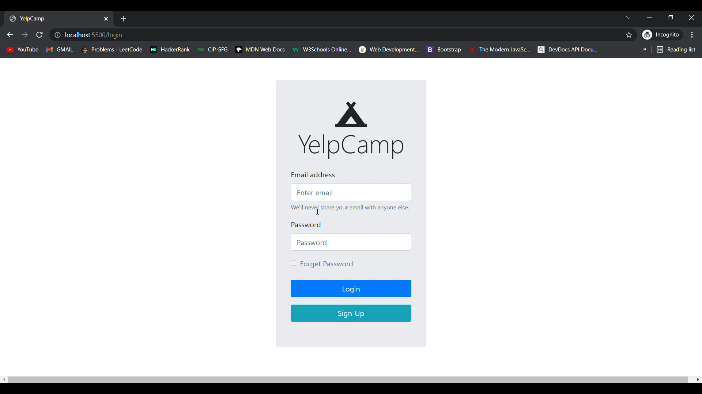
**Chapter 4. Result & Discussion**

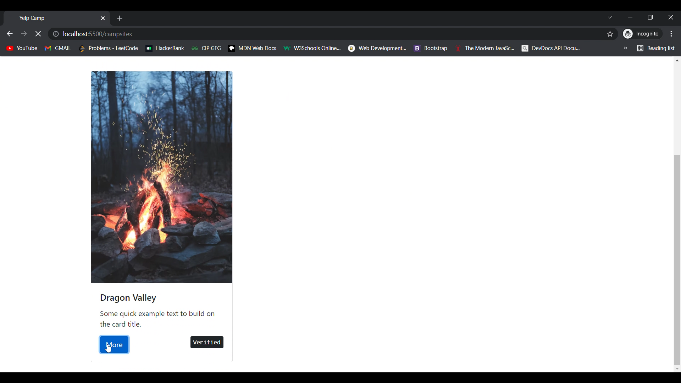
**Chapter 5. Conclusion, Limitation & Future Scope**

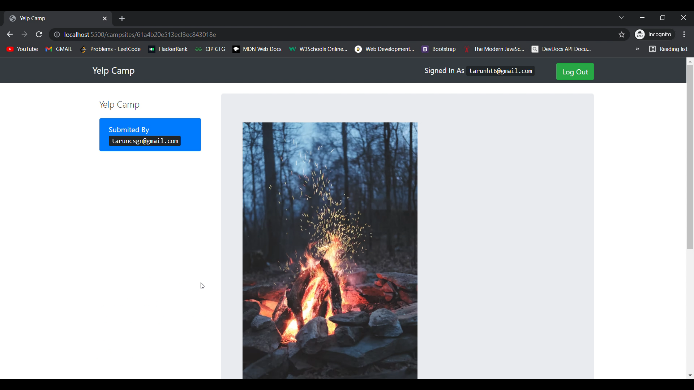
**Reference**

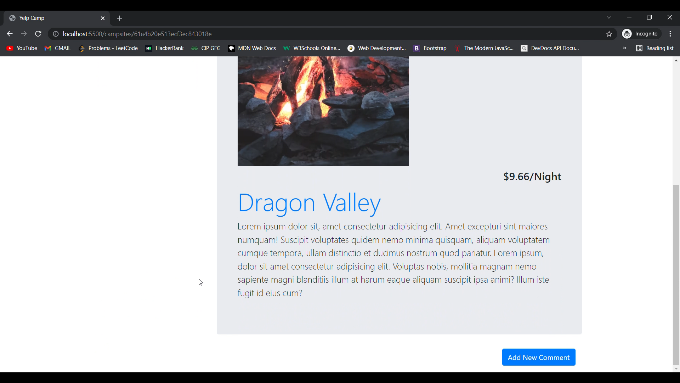
**LIST OF FIGURES**

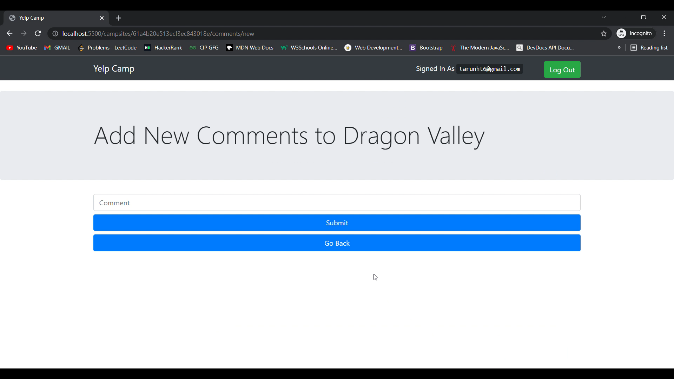
****

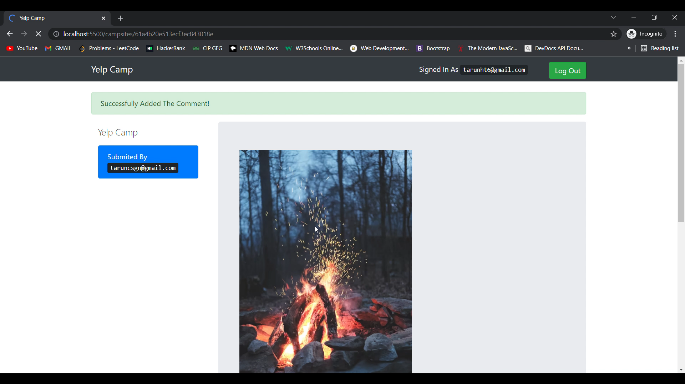


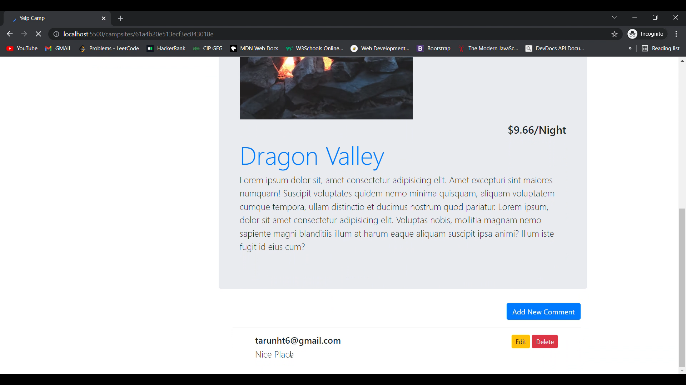


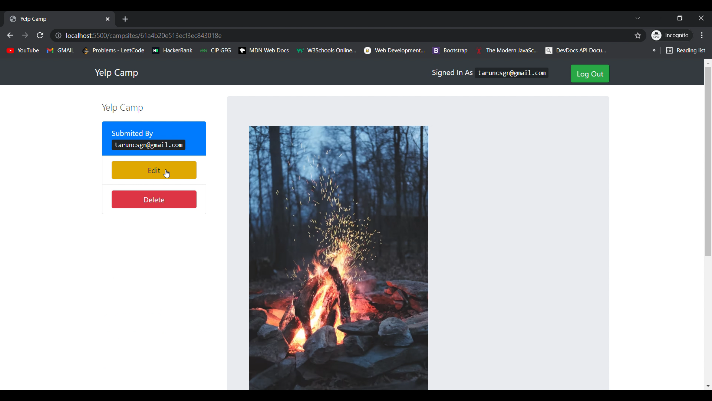












**Chapter 1**

**Introduction**

1. **Problem Definition**

This application allows users to share their campsite experiences and collaborate with one another through comments. The application uses the seven types of RESTFUL routes in order to display a list of campsites and comments, create a new campsite or comment, show more info about a particular campsite, edit campsite information or comment content, and delete campsites as well as comments. In order to collaborate on the application you need to be logged in, or sign up, and be authenticated. Only the user who posts a campsite can edit and delete it. Same goes for comments that users post. The application makes use of Express.js to build the backend server and mongoDB as the database to store campsite, comment, and user data.

**Features:**

* **Add Campgrounds**
  + The ultimate goal of the app is to add campgrounds, be they real or fictitious, I’m not much of a camper myself so I usually add fictional ones like for example, I made a campground called Lake Hylia Lodge, which is a reference to the Zelda series, however if you want to add real campground information feel free to, I actually implore anyone that visits the site to put up one of their favorite camping sports.
* **User Authentication**
  + You can sign up, sign in, and its a necessary process to go through if you want to contribute to the site, why? because when a user is signed in it allows them to add campgrounds and comments, to really add a sense of community and interactivity. It’s pretty barebones at the minute but hopefully in future updates I can implement more features that require User Auth that make it feel like a full fledged app, maybe even OAuth so you can sign in using existing media platforms. There are also measures in place to help you reset your password should you need to, pretty cool right?
* **Comments**
  + As said above, you can add comments to campgrounds to express all kinds of things, maybe you’ve visited the campsite and want to give a little review (unless of course you’re viewing one on mine, which are highly fictitious, but feel free to drop a few words anyway), or whatever else you feel the need to.

1. **Project Overview/Specification:**

**2.1 HTML:**

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

* 1. **CSS:**

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

* 1. **Node.js:**

Node.js is an open-source, cross-platform JavaScript run-time environment that executes JavaScript code outside of a browser, Node.js lets developers use JavaScript to write Command Line tools and for server-side scripting—running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. As an asynchronous event driven JavaScript runtime, Node is designed to build scalable network applications. In the following "hello world" example, many connections can be handled concurrently. Upon each connection the callback is fired, but if there is no work to be done, Node will sleep.

* 1. **MongoDB:**

MongoDB is a free and open-source cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with schemata. MongoDB is developed by MongoDB Inc., and is published under a combination of the GNU Affero General Public License and the Apache License.

**2.5 Express Framework:**

Express.js, or simply Express, is a web application framework for Node.js, released as free and open-source software under the MIT License. It is designed for building web applications and APIs.[3] It has been called the de facto standard server framework for Node.js.[4] The original author, TJ Holowaychuk, described it as a Sinatra-inspired server, meaning that it is relatively minimal with many features available as plugins. Express is the backend part of the MEAN stack, together with MongoDB database and AngularJS frontend framework

**Chapter 2**

**System Analysis & Design**

**INTRODUCTION:**

The YELP Camp project is all about the camping activities done at the various tourist places.

**REQUIREMENT GATHERING:**

This section highlights the list of steps involved in gathering the requirements.

**Functional Requirements:**

The application provides various camping sites around the world.

**Data requirements:**

This app uses the GPS functionality of devices to gather the longitude and latitude of the palace and then passes it to the YelpCamp to give camp data of that information.

**Environmental requirements:**

a. Used for exploring camp sites

b. Simple usage and interactive UI

c. It will run on Node.js & MongoDB

**User Requirements:**

a. Campers can post the camp site they recently visited.

b. People can explore various camp sites and get to know more about them.

**Usability Requirement:**

Web Application requires constant Internet connection and a Mobile Phone or PC.

**FEASIBILTY STUDY**

A feasibility analysis usually involves a thorough assessment of the operation, financial and technical aspects of a proposal. Feasibility study is the test of the system proposal made to identify whether the user needs may be satisfied using the current software and hardware technologies.

**FIVE AREAS OF PROJECT FEASIBILITY STUDY:**

**Technical Feasibility:**

This project is the web application so it requires only web browser environment to run on. The language which is used in development is Node.js and we take help of Express Framework to implement the predefined methods so it is understandable by developer therefore we can state that our project is technically feasible.

**Economic Feasibility:**

As in this project the all resources are open source and user participation is spontaneous, so we can say that this project is economically feasible.

**Operational Feasibility:**

As this project is is all about the recreational camping activities done at the various tourist places so it fulfills all the qualities of operational feasibility.

**Legal Feasibility:**

My project fulfills all the legal documentation so it is legally feasible in nature.

**Scheduling Feasibility:**

This project is completed within my training period so it is completed in scheduled time so it is feasible on time.

**REQUIREMENT DEFINITION:**

This section includes the requirements:-

**Functional Requirements:**

This section includes the requirements that specify all the direct or fundamental requirements of the software system.

**Admin Case Study**:

As in this application we have a provision of creating an admin account, so the admin can take control of all the data about location and give outputs to the user.

**Customer Use Case:**

The user only has control on their internet and GPS connection. Users just need to enter the camp data for posting it.

**Non-Functional Requirements:**

A user must have the web browser and internet connection.

**Performance**: It performed well Google Chrome as it requires low amount of storage and ram. Its performance is mainly based on the internet connection. If there is fast internet connection then it gives the output fast otherwise depends on the speed of internet. It gives the currently listed camps on the home.

**Reliability**:

This web app is reliable in nature and it only goes down whenever there is no stable internet connection.

**Availability:**

This web application is easily available for access on web browsers. It is user friendly in appearance.

**Security:**

As this web app is developed with keeping various security parameters in mind therefore it secures the data of the user.

**Maintainability:**

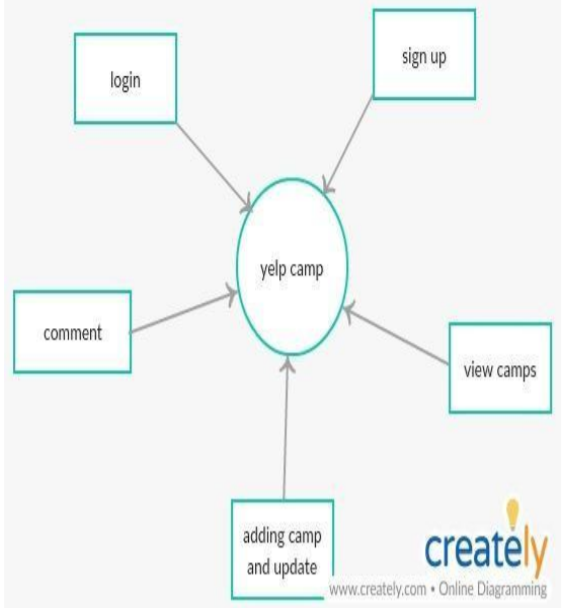
This web app is maintained easily by the single person itself.

**Portability:**

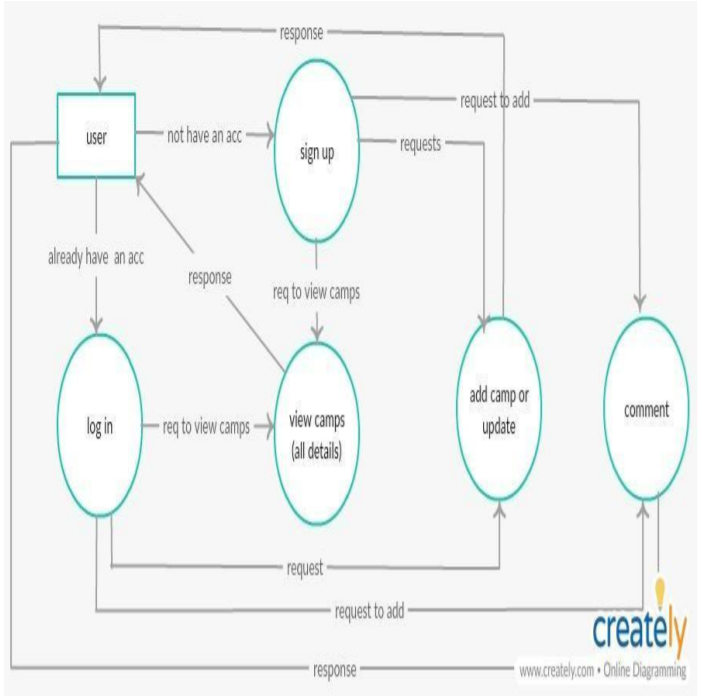
This web app is easily moved to another browser because of the cross browser compatibility code used, therefore the app is portable.

**DESIGN:**

**DATA FLOW DIAGRAM**

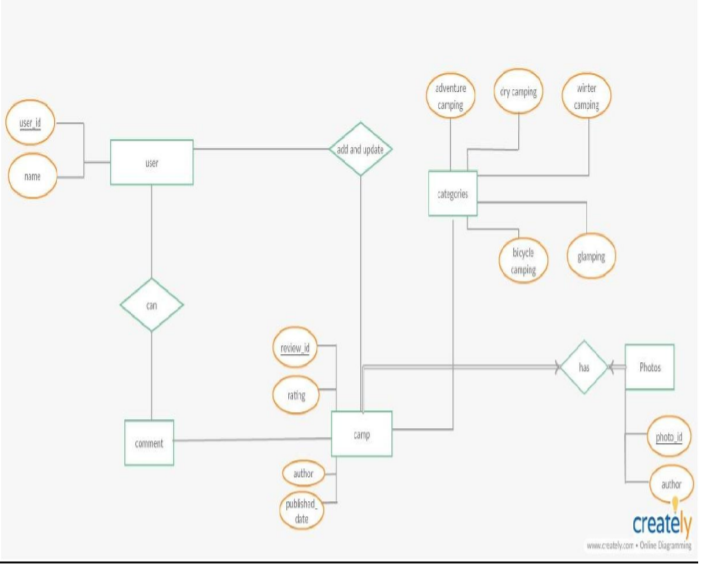
****

**Fig1. Zero Level DFD**

****

**Fig2. One Level DFD**

**ENTITY – RELATIONSHIP DIAGRAM**

****

**Fig. ER DIAGRAM**

**Chapter 3**

**Implementation**

var express = require("express");

var mongoose = require("mongoose");

var passport = require("passport");

var LocalStrategy = require("passport-local");

var passportLocalMongoose = require("passport-local-mongoose");

var bodyParser = require("body-parser");

var session = require("express-session");

var camp = require("./modules/campScehma");

var comment = require("./modules/commentSchema");

var seeder = require("./modules/seed");

var User = require("./modules/userSchema");

var methodOveride = require("method-override");

var flash = require("connect-flash");

var campsitesRoutes = require("./routes/campsites");

var commentsRoutes = require("./routes/comments");

var indexRoutes = require("./routes/index");

var DBURL = process.env.DBURL || "mongodb://localhost:27017/camps",

IP = process.env.IP || "127.0.0.1",

PORT = process.env.PORT || 5500;

//============================================================

// CONFIG

//============================================================

var app = express();

mongoose.connect(DBURL, function(err){

if (err){

console.log(err);

}

});

app.use(bodyParser.urlencoded({extended: true}));

app.use(methodOveride("\_method"));

app.set("view engine", "ejs");

app.use(express.static('public'));

app.use(flash());

//seeder();

//============================================================

// PASSPORT CONFIG

//============================================================

app.use(session({

secret: '\*&\*&()IOP{{PO((\*&^YTGBHHHI()\_+>>>',

resave: false,

saveUninitialized: false

}));

app.use(passport.initialize());

app.use(passport.session());

passport.use(new LocalStrategy(User.authenticate()));

passport.serializeUser(User.serializeUser());

passport.deserializeUser(User.deserializeUser());

app.use(function(req, res, next){

res.locals.currentUser = req.user;

res.locals.success = req.flash("success");

res.locals.error = req.flash("error");

next();

});

//===========================================================

app.use("/campsites", campsitesRoutes);

app.use("/campsites/:id/comments/", commentsRoutes);

app.use(indexRoutes);

app.listen(PORT, IP, function(){

console.log("server started successfully");

});

**Chapter 4**

**Conclusion, Limitation & Future Scope**

**Conclusion:**

App - ‘YELP CAMP’ is a really efficient application. The application has a professional look and feel, and it is also very easy to use as compared to the other applications. Web app development is a continuous process with many iterations. There is no such thing as a perfect design and the app has to dynamically be able to change in order to follow the trends, because the App World is a dynamic world. What is good today, may not be good tomorrow. Developers have to always keep this in mind if they want their app to succeed in this market, and although testing is a very important step of this process, so the app is not published with bugs, the best test they can make to it is to release it. It will reach millions of users throughout the world, all with different needs and tastes and only then will you have the results that allow you to develop the app in one or more directions.

**Things Learned:**

**Design:**

The design has to reflect the quality of those camps because the user has no idea of what the services can or cannot do. He only knows what he sees and interacts with.

**Node.js:**

Node is powerful resource helpful in creating web applications with great ease and efficiency. As an asynchronous event driven JavaScript runtime, Node is designed to build scalable network applications.

**Express:**

Light Weight Framework for node which boost the app development process and is very easy to use. 9.1.4 APIs: Through this project I get know about APIs and how we can use them in our application. I also learn how to get the information through APIs and how to link the APIs through its key.

**Limitations:**

**Future Scope:**

**REFERENCES**

<https://www.udemy.com/the-web-developer-bootcamp/>

<https://www.npmjs.com/>

<http://reduxblog.herokuapp.com/>

David Herron(2014). “Node Web Development, Second Edition”. Packt: O'Reilly.

Ethan Brown (2015). “Web Development with Node and Express: Leveraging the JavaScript Stack, Third Edition”. Packt: O'Reilly.