**Bethune-Cookman University**

**College of Science, Engineering and Mathematics**

**CIS 498 Senior Design II**

**CodeSpace**

**Team:**

**Kadena Banner**

**Tavon Goddard**

**Keshawn Fleming**

**Mentors:**

**Dr. David Bethelmy – Computer Science**

**Fall 2018**

**Executive Summary**

The goal of the CodeSpace project is to provide a space in which users worldwide can interact, share, and learn. Whether your skill level is of that of a beginner or expert, CodeSpace can surely assist individuals farther along in their career. The idea behind the design is to present a website that is simple for both tech and non-tech savvy users. A very simple design that doesn’t overwhelm our users or cause them to have anxiety.

Sometimes even the best of the best can get stuck trying to figure out how to fix a piece of code. As perhaps they have been working tireless, but still can’t seem to resolve the issues by themselves. Or suppose they have the solution to help resolve the issue for someone else’s code. Well now, with that of CodeSpace students and/ or nonstudents can utilize our site to do exactly that. CodeSpace is intended to essentially make students majoring in CIS and related fields life a bit easier. CodeSpace will be a well-structured website that allows users to not only communicate through posted comments, but also upload documents of code for assistance, and seek internship opportunities as well. CodeSpace utilizes tools such as Node.js, Robo 3T, and a Mongo database to create a solid back-end framework. While uses HTML, CSS, JavaScript, jQuery and Bootstrap to construct a neat and beautiful front-end framework.

# Table of Contents

# 

[Executive Summary 1](#_Toc498604150)

[Table of Contents 2](#_Toc498604151)-3

[1. Introduction 4](#_Toc498604152)-6

[1.1 Objective 4](#_Toc498604153)

[1.2 Motivation 4](#_Toc498604154)

[1.3 Background 5](#_Toc498604155)

[2. Project Description and Goals 7](#_Toc498604156)-10

[3. Constraints 11](#_Toc498604157)

[4. Alternative design 12](#_Toc498604159)

[5. Results & Analysis 13-14](#_Toc498604160)

[6. Conclusion 15](#_Toc498604161)

[7. Reference 16](#_Toc498604162)

[8. Gantt Chart 17](#_Toc498604163)

[(A)Timeline 17](#_Toc498604164)

[(B) Task View and timeline View 18](#_Toc498604165)

[9. Acknowledgements 19](#_Toc498604166)

[Appendix A 20-22](#_Toc498604167)

[Appendix B 23-25](#_Toc498604168)

1. **Introduction**
   1. **Objective**

The utilization of CodeSpace is intended to give individuals within STEM related fields an edge, by providing effective strategies for them to succeed in their fields. It will essentially bring together many students that think and work in similar ways for them to discover and share more information with each other pertaining to their major and updates in the technological world.

* 1. **Motivation**

Tutoring resources for students within STEM related fields, are usually slim to none. In fact, majority of individuals within said fields usually struggle in their core classes. Because unfortunately, tutoring for such core courses usually isn’t offered physically on campus. Which means students are usually left having to scowl the internet in order to acquire help with their issues. And although there’s nothing wrong with applying a bit of elbow grease to get the job done. Scowling the internet can be quite time consuming. So much so, one could perhaps waste an entire day attempting to fix their code, oppose to proceeding to the next task. So, we decided to come up with a place in which students and/ or nonstudents can have a place to interact. A place that coders can help each other, in whether it be web development or networking. This community will allow users to create a personal account, edit account info, interact with other members, search specific users, as well as seek advice from potential hiring companies.

* 1. **Background**

We have always needed information to be saved somewhere, we started using the practice to save data in many forms since ancient times. And since then, we dramatically evolved from stone age to modern age. During stone age we used to write on stones and caves. But with the advancement of technology, we now use computers to save our precious information. There are many different database tools in the world that one can choose from. However, one of the top database platforms aside from SQL would have to be MongoDB. [5]

MongoDB was developed in 2007 by a New York based organization 10gen which is now called MongoDB Inc. It was developed as a PAAS (Platform as a service) initially. But later one in the year of 2009, it was then introduced in the market as an open source database server that was maintained and supported by the MongoDB Inc. Many large and medium scale organizations such as Source Forge, Foursquare, craigslist and eBay are now using MongoDB in development of their database applications. [5]

MongoDB is an open source database that uses a document-oriented data model and a non-structured query language. It is one of the most powerful NoSQL systems and databases around today. Being a **NoSQL tool** means it does not use the usual rows and columns that we so much associate with the relational database management. It is an architecture that is built on collections and documents. The basic unit of data in this database consists of a set of key-value pairs.

It allows documents to have different fields and structures. This database uses a document storage format called BSON which is a binary style of JSON style documents. [5]

It follows is a highly elastic one that lets you combine and store data of multivariate types without having to compromise on the powerful indexing options, data access and validation rules. There is no downtime when you want to dynamically modify the schemas. So, what it means that you can concentrate more on making your data work harder rather than spending more time on preparing the data for the database. [5]

# Project Description and Goals

The Login page allows a user to sign up as well as login to their personal account. Displayed below is the functionality process:

1. The user must register to become a new member if they haven’t already done so. If said username and email is already in the database. The user will be prompted that the information is already on file.

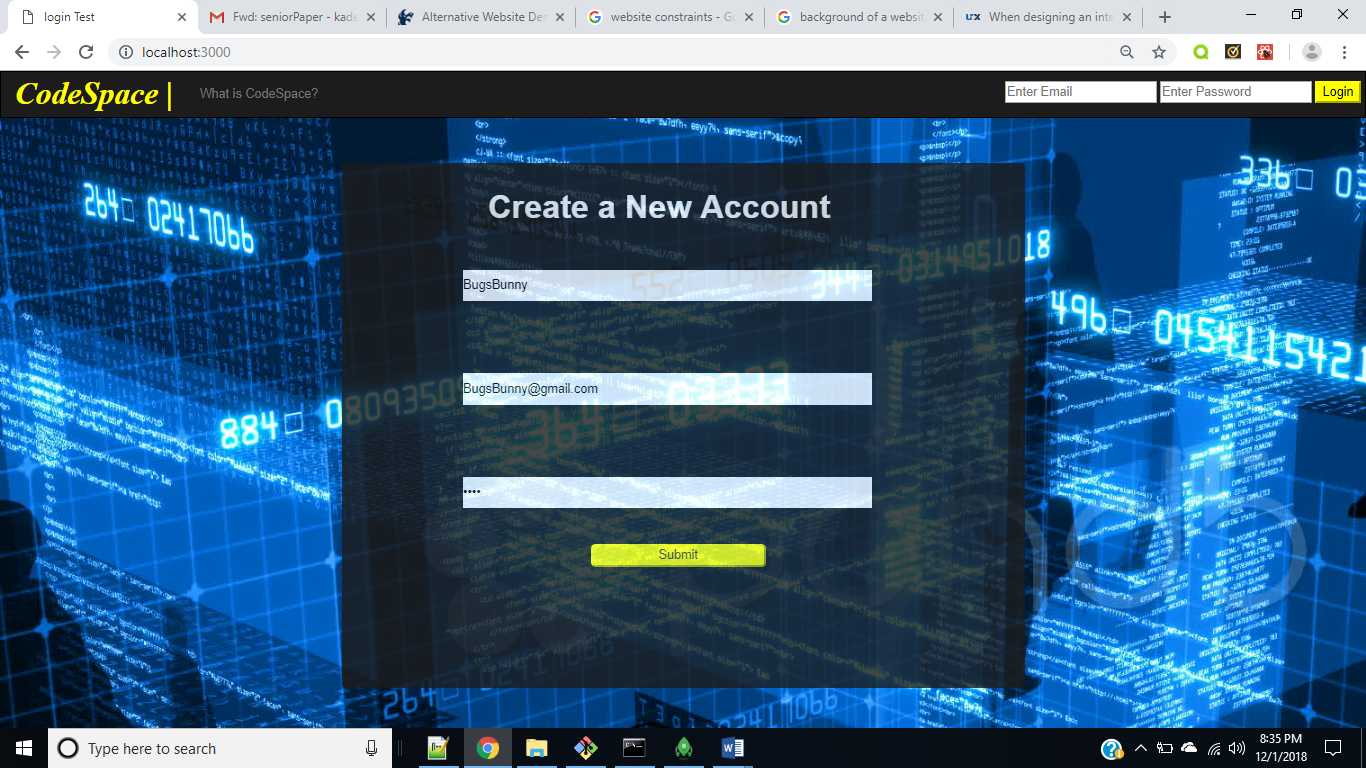
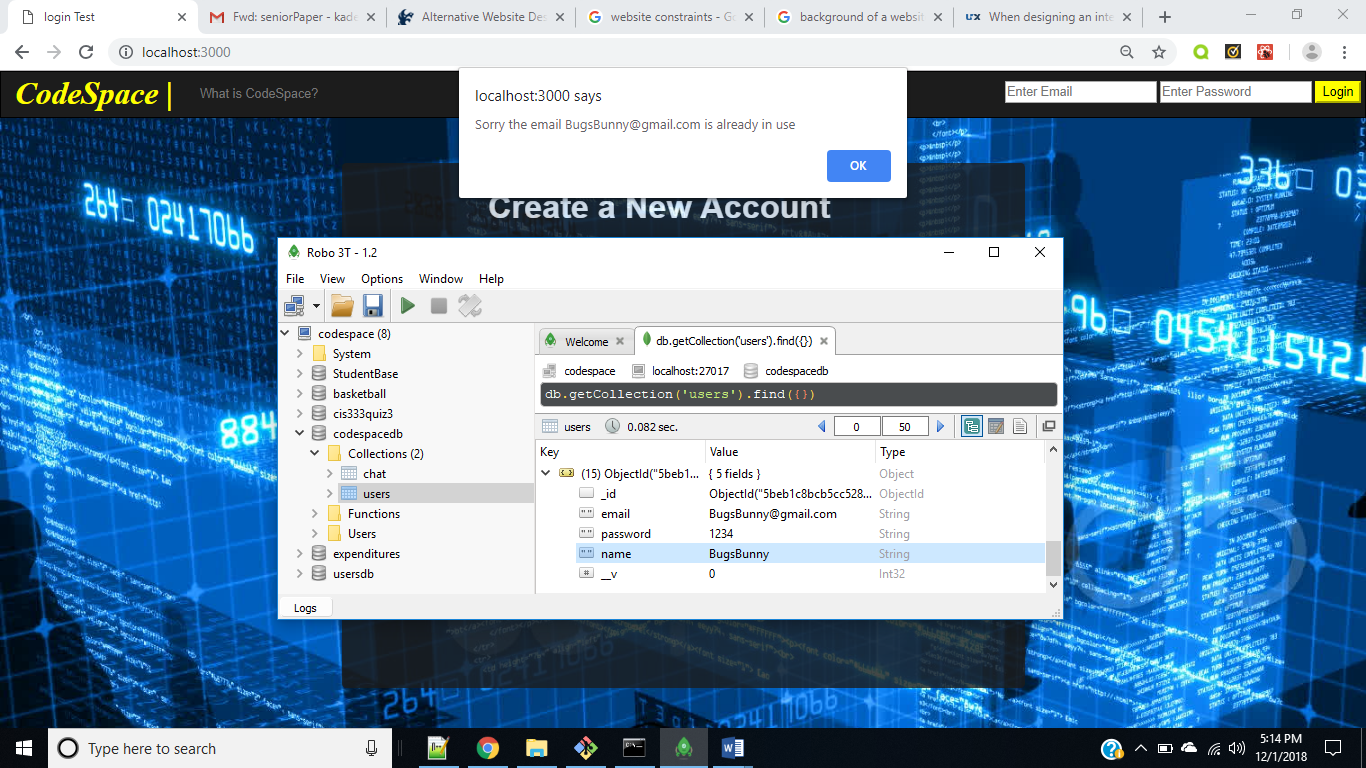
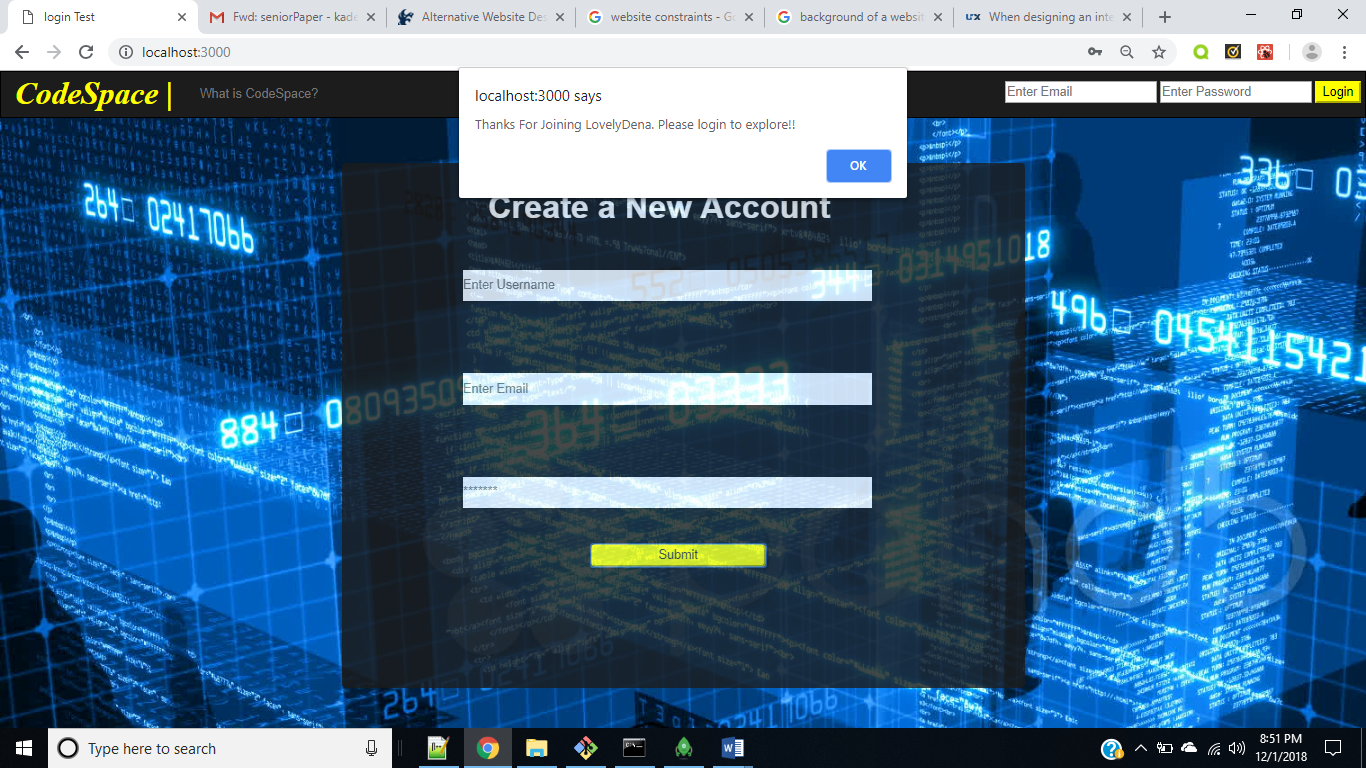


Figure 2-1 Login Page Figure 2-2 Login Page

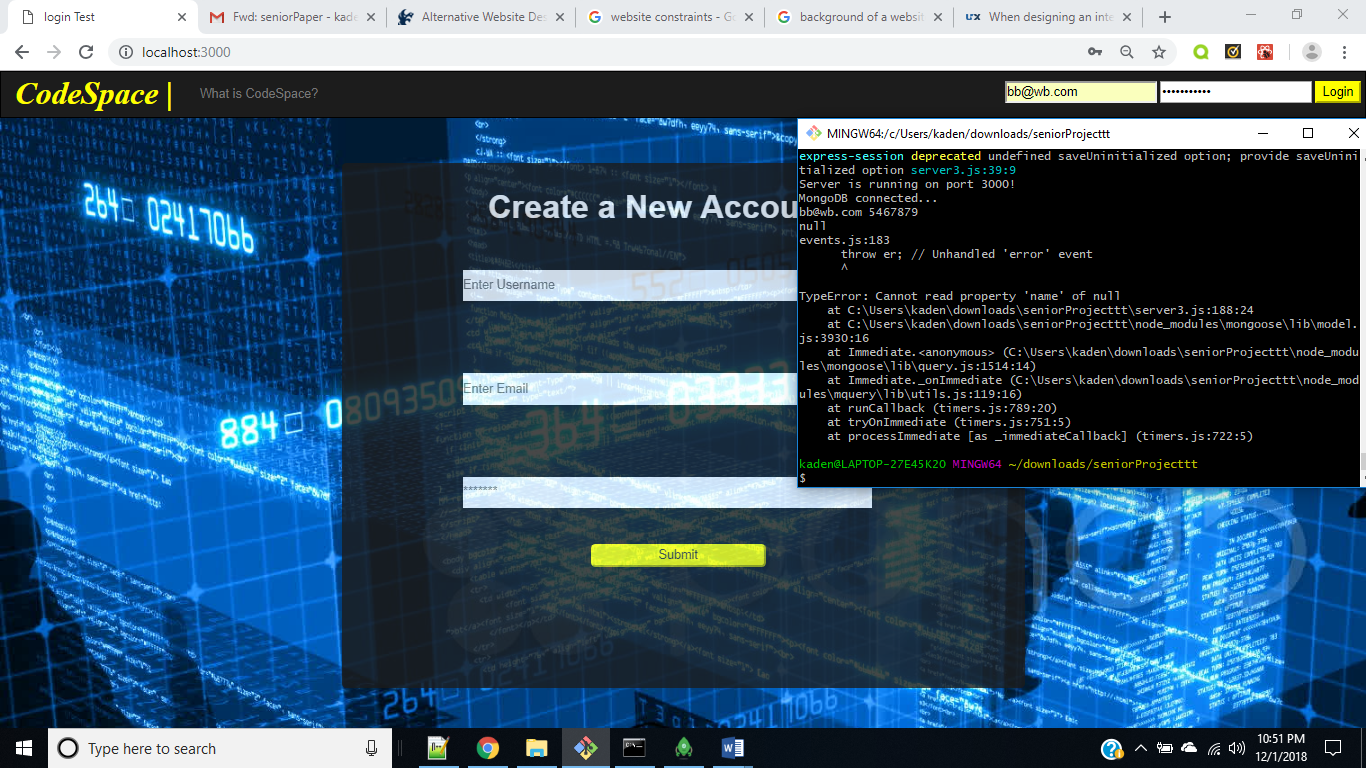
1. Once the user has successfully registered as a new member, they are then prompted to login.

Input desired user info

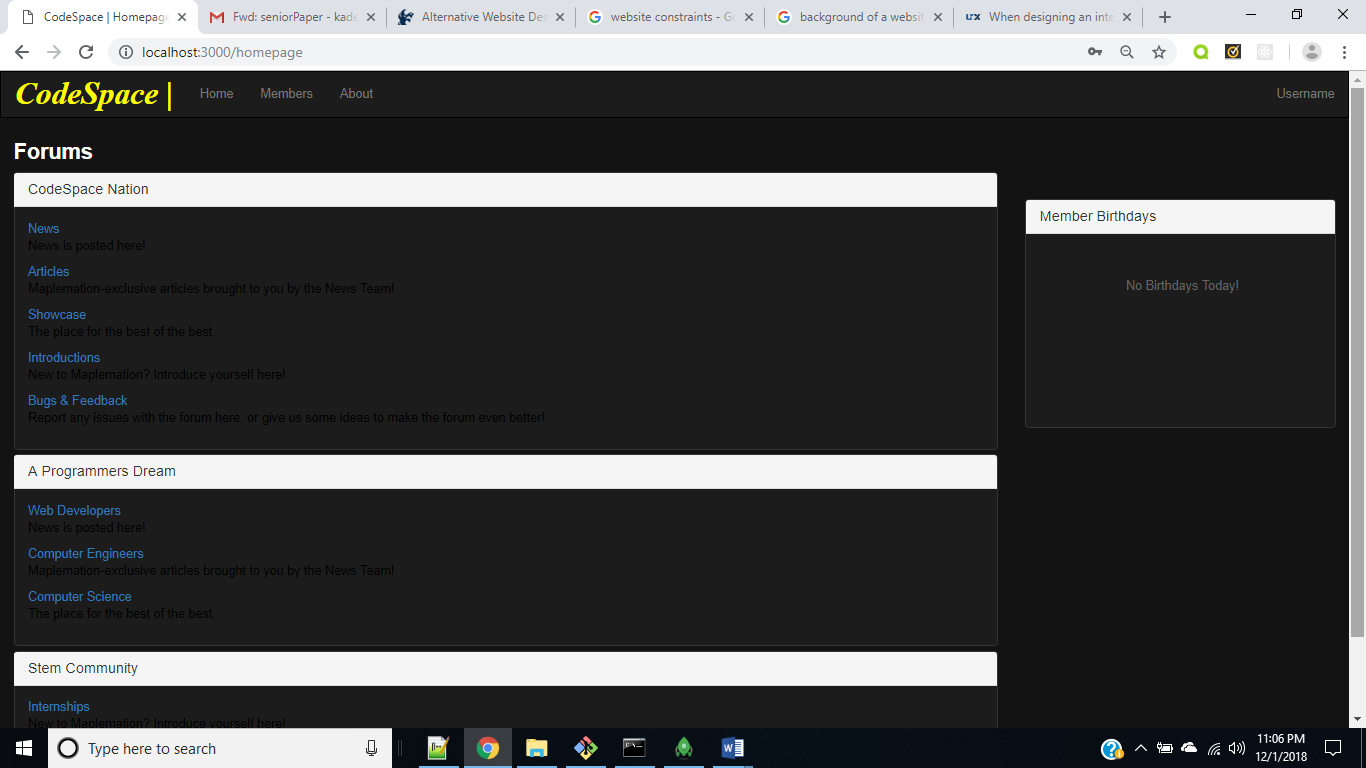
User has successfully joined, system asks them to proceed and login



*Figure 2-3 Login Page Figure 2-4 Login Page*

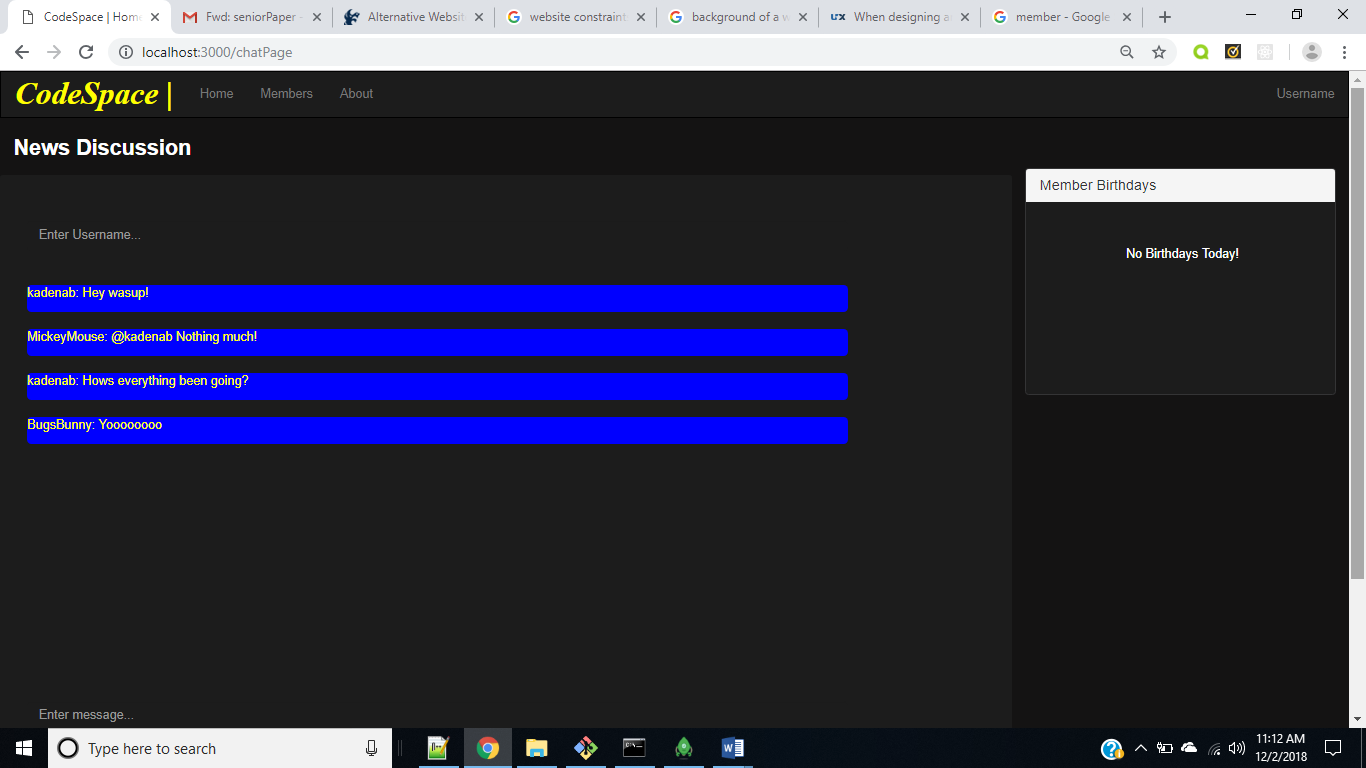
1. If said member isn’t found, access will be denied until the correct information is submitted.

*Figure 2-5 Login Page*

1. Once the information is correct the user will then have access to the homepage.

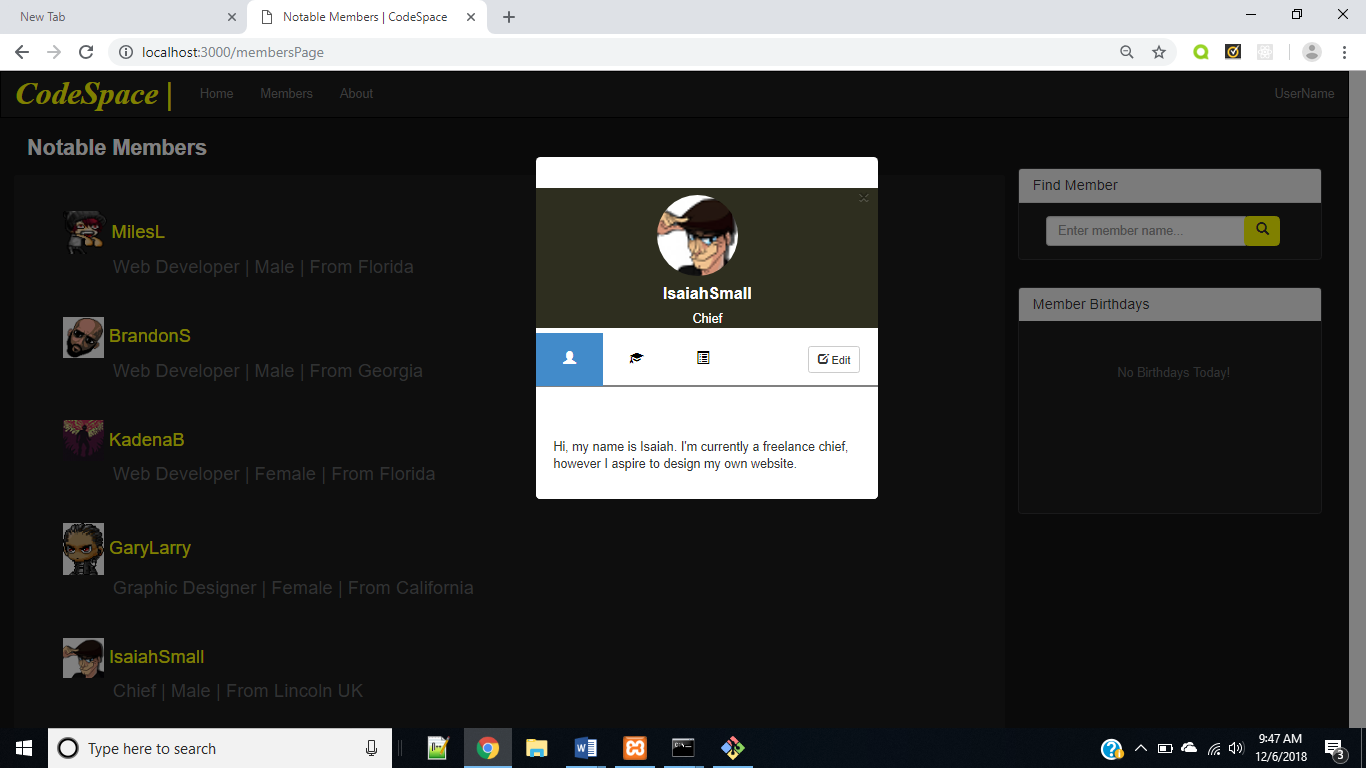
This page displays an accordion with that of links going to the users desired chat discussion.

*Figure 2-6 Discussion Forum Page*

1. Once the user selects their desired category, they will then be redirected to that specific chat discussion. *Figure 2-7 Chat Page*

News link was selected, as a result the News discussion is displayed.

1. All notable members are displayed on the members page. When specific image is clicked it display a modal of said user profile.



*Figure 2-8 Members Page*

# Constraints

The design of this project will use MongoDB as the database tool, we originally used a JSON file to display the data onto the page. But being that MongoDB is already a JSON- like database, the transfer was quite simple. As it only required us to import our JSON file within MongoDB, to be then saved as a Mongo database. All the data is saved within the database, then brought in from the server as well as an ajax call to display. [5]

**Social constraints**

A multitude of professionals rely on websites such are LinkedIn, Slack, Codecademy and etc to assist them within their career. And the social constraints we have developed within this project is to provide said professionals with a platform to interact with one another to do exactly that. Our design is like that of a teched out Facebook.

**Ethical constraints**

The ethical constraints that apply to our project is not for nefarious reason and/ or exploiting the information for personal gain. Meaning that the information that is collected from our members will not be released to that of demography collectors.

**Manufacturing constraints**

From a manufacturing stand point our design required a login system page. Which a user must follow through in order to be permitted access to any of the other pages. Otherwise there will be no access granted to anyone that is not registered as a member.

# Alternative design

Our alternative design was to create an entire user profile page. This page was intended to display all the user personal account information such as their career, skill set, education and etc. However, we had to come up with another strategy in order to display each user profile. As it would have deemed, to be extremely difficult to handle such a task. So, instead of our original design idea, we ended up having to come up with a simpler strategy, to do the exact same thing without generating an entire page. Another design change we had to bring about was with the chat page. As we would have liked for this page to get access to the user session, without them having to make a username input. But unfortunately, we stumbled across issues while trying to accomplish making this connection. Which is why we asked that the user must input their username, as we wanted to identify the member each chat was from.

# Results & Analysis

Our Design consist of 5 pages, such as the login, home, forum link, chat and members page. Code A program will be utilized to allow each user to register as well as login. The information will then be saved within the database. However, if the input of the user is already stored within the database the user will then be prompted with an issue informing them that said information is already in use. While Code B acts to initiate the functionality of the chat page. As this code retrieves the input of both the username and message value for the member, then append the username along with the message. This allows us to identify who the message is coming from, instead of it being a just a blank message. Below is a working copy of our server that preforms a post call to routes on the login page, to access the data from the user input.

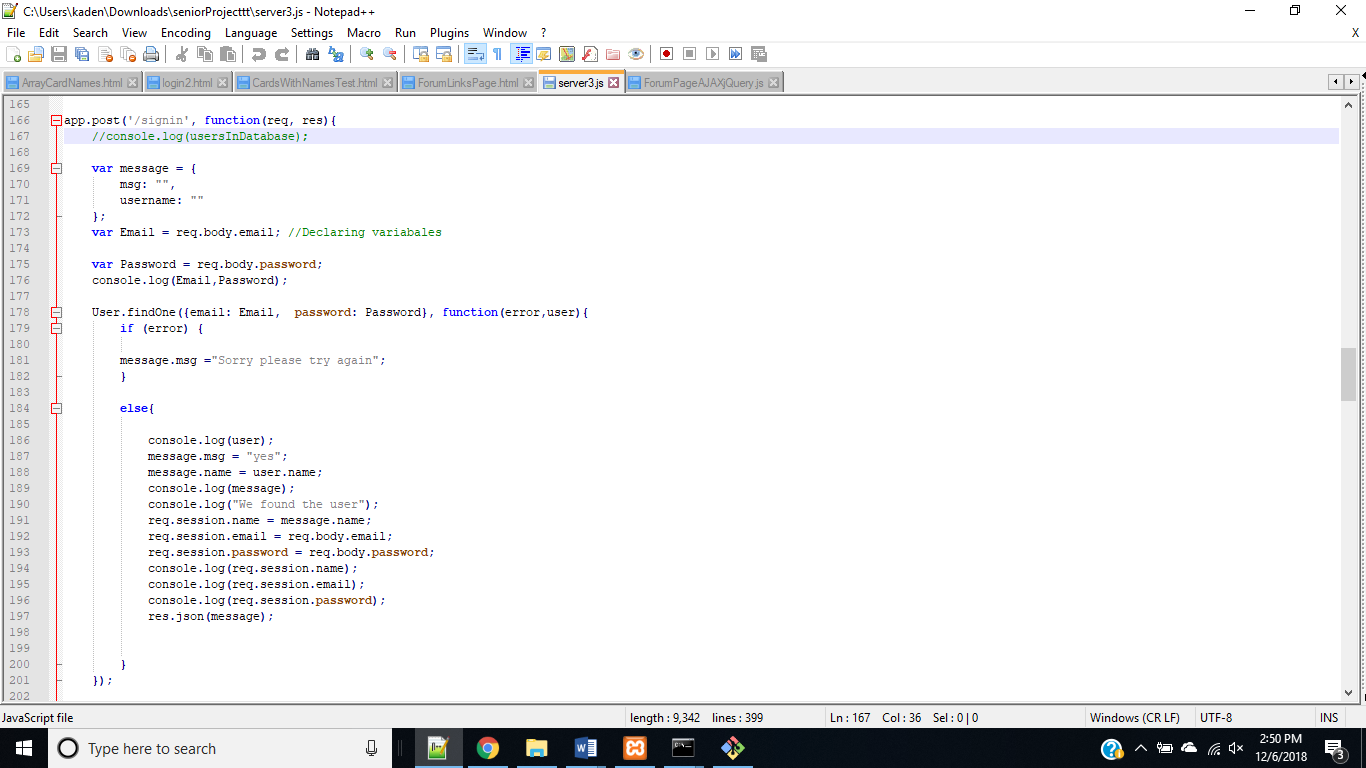


Figure 5-1 Sign-In route for server

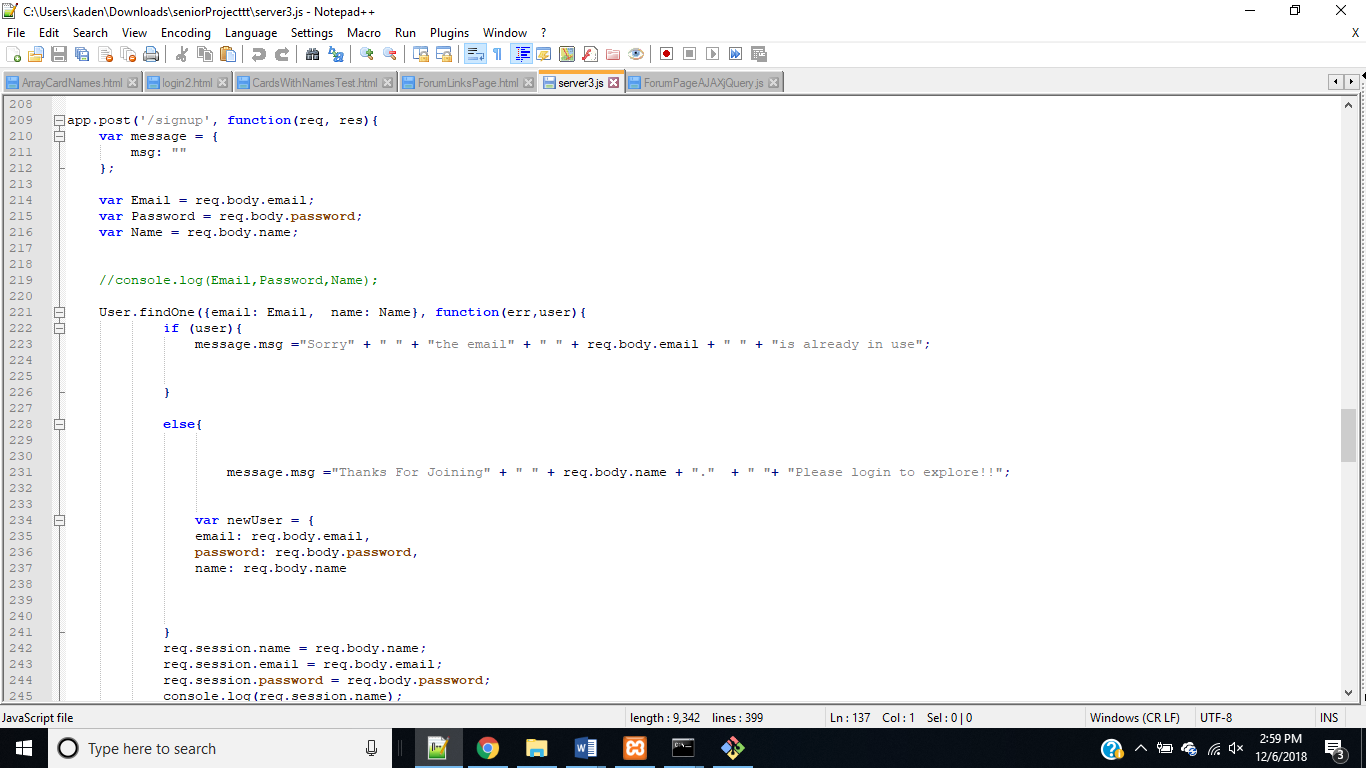


Figure 5-2 SignUp Route for Server

# Conclusion

As we started this project, we weren’t aware of the amount of research it would take for us to accomplish the tasks we aimed to complete within the timeframe. But along the way, we discovered we could do a lot more than we thought. This project not only challenged our preexisting skills, but also forced us to learn a multitude of different strategies that were unfamiliar to us. We found that to accomplish this project, required us to combine everything we learned through our duration of college.

Although we stumbled to assemble the pieces together, we were still able deliver a decent design. Something of which is simple, and user friendly to where all age groups can navigate our site. In conclusion, we have learned a lot from our project. And although our project is starting to come to an end, our journey of learning is long from. So, even with the conclusion of our project we will continue to learn and test our abilities.

# Reference

1. "About Node.js, and why you should add Node.js to your skill set?". Training.com. Training.com. Retrieved 5 May 2018.
2. "Netscape opens intranet attack". CNET. Retrieved 2018-05-5.
3. Ryan Dahl (2010-11-09). "Joyent and Node". Google Groups. Retrieved 2018-05-05.
4. What Is Mongodb? - Definition from Techopedia

https://www.techopedia.com/definition/30340/mongodb

1. What Is Mongodb - Introduction to Mongodb Architecture & Features

https://intellipaat.com/blog/what-is-mongodb/

# Gantt Chart

# (A)Timeline

# (B) Task View and timeline View



# Acknowledgements

This project would not have been possible without the assistance and guidance of the entire Science Department. However, we would like to give special thanks to Dr. Bethelmy for his endless support and mentoring us during the development of our project. We would also like to thank Dr. Adams for teaching us the foundation of developing a database and how to go about managing a project. Lastly, we would like to thank Dr. Obeng for his insight, patience and mentoring during our senior year.

# Appendix A

**Code A**

$(document).ready(function()

{

$("#loginForm").submit(searchByLogin);

$("#signupForm").submit(searchBySignup);

});

function searchByLogin(e)

{

e.preventDefault();

//alert('Login');

//You need to check here to see if the correct type of data was entered.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

var data =

{ 'name': $('#signupUsername').val().trim(),

'email':$('#loginEmail').val().trim(),

'password':$('#loginPassword').val().trim()

};

//clear the input areas

$('#loginEmail').val("");

$('#loginPassword').val("");

// show the email and password assigned to data

console.log(data);

// post "/signin" route in server using the data captured fromm the input

$.post( '/signin', data, function(result){

if(result.msg == "yes")

{

window.location.href = "/homepage";

}

else

{

alert("There is an error");

}

});

}

function searchBySignup(e)

{

e.preventDefault();

//alert('Signup');

//You need to check here to see if the correct type of data was entered.

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

var data =

{

'name': $('#signupUsername').val().trim(),

'email':$('#signupEmail').val().trim(),

'password':$('#signupPassword').val().trim()

};

//clear the input areas

$('#signupUsername').val("");

$('#signupEmail').val("");

$('#signupPassword').val("");

$.post( '/signup', data, function(result){

if(result.msg == "yes")

{

window.location.href = "/homepage";

}

else

{

alert(result.msg);

}

});

}

# Appendix B

**Code B**

(function(){

var element = function(id){

return document.getElementById(id);

}

// Get Elements

var status = element('status');

var messages = element('messages');

var textarea = element('textarea');

var username = element('username');

// Set default status

var statusDefault = status.textContent;

var setStatus = function(s){

// Set status

status.textContent = s;

if(s !== statusDefault){

var delay = setTimeout(function(){

setStatus(statusDefault);

}, 4000);

}

}

// Connect to socket.io

var socket = io.connect('http://127.0.0.1:4000');

// Check for connection

if(socket !== undefined){

console.log('Connected to socket...');

// Handle Output

socket.on('output', function(data){

//console.log(data);

if(data.length){

for(var x = 0;x < data.length;x++){

// Build out message div

var message = document.createElement('div');

var breakline =document.createElement('br');

message.setAttribute('class', 'chat-message');

message.textContent = data[x].name+": "+data[x].message;

messages.appendChild(message);

messages.appendChild(breakline);

messages.insertBefore(breakline,message, messages.firstChild);

}

}

});

// Get Status From Server

socket.on('status', function(data){

// get message status

setStatus((typeof data === 'object')? data.message : data);

// If status is clear, clear text

if(data.clear){

textarea.value = '';

}

});

// Handle Input

textarea.addEventListener('keydown', function(event){

if(event.which === 13 && event.shiftKey == false){

// Emit to server input

socket.emit('input', {

name:username.value,

message:textarea.value

});

event.preventDefault();

}

})

// Handle Chat Clear

clearBtn.addEventListener('click', function(){

socket.emit('clear');

});

// Clear Message

socket.on('cleared', function(){

messages.textContent = '';

});

}

})();