Chengpeng Li. Developing Fitness Group Class Recommendations Web-Application for UNC Campus Recreation. A Master's Paper for the M.S. in I.S degree. April, 2018. 36 pages. Advisor: Arcot Rajasekar

UNC Chapel-Hill has great amount of gym facility and fitness group class services which are free to all students. However, students do have trouble in choosing the most suitable fitness group class due to many options and lacking in knowledge. For example, students who are new to fitness aren't very familiar with it. They are lacking in information about what kind of activity the class is doing as well as the best suitable class for them. Some classes are targeted to Intermediate fitness enthusiasts and may not be suitable for beginners, while some classes are entry-level courses for beginners and experienced Fitness lovers may find them boring. So it is definitely helpful if the website has detailed description of the class info, including the teacher, targeted people and brief information about the activity. For certain students, they love some teacher who have their Unique teaching methods.

Headings:

Information Systems

Requirement Analyst

Interface Design

System Implementation

DEVELOPING FITNESS GROUP CLASS RECOMMENDATIONS WEB-APPLICATION FOR UNC CAMPUS RECREATION

by
Chengpeng Li

A Master's paper submitted to the faculty
of the School of Information and Library Science
of the University of North Carolina at Chapel Hill
in partial fulfillment of the requirements
for the degree of Master of Science in
Information Science.

Chapel Hill, North Carolina

April 2018

Approved by

____

Arcot Rajasekar

Table of Contents

1 Introduction

1.1 Rationale for the Study

UNC Chapel-Hill has many great gym facilities and fitness group classes which are free to all students. They are popular with students and in great demand. However, students do have trouble in choosing the most suitable fitness group class due to many options. For example, students who are new to fitness are not very familiar with it. They are lacking in information about what kind of activities the class is doing as well as the best suitable class for them. Some classes are targeted to Intermediate fitness enthusiasts and may not be suitable for beginners, while some classes are entry-level courses for beginners thus experienced Fitness lovers may find them boring. So it is helpful if the website has detailed description of the class information, including the teacher, targeted people and brief information about the activity. For certain students, they love some teachers who have their Unique teaching methods. Also, getting feedback comments about the class from students is beneficial for both students and Campus Recreation. For students, that kind of information can help them to make a decision to choose the class. While for Campus Recreation, it is of great importance to help them improve the fitness group class to meet everyone's need.

This paper describes the process of developing the web-application from scratch. This paper starts from literature review of past related papers and then requirement analysis, sysstem design, and system implementation of the proposed web application. Finally, the conclusion section will summarize the whole developing process for this web-application and suggested improvements in the future.

1.2 Research Questions.

This research aims at implementing a web-application fitness class recommendation system using Front-end Framework Materialize CSS, back-end Framework Express and MongoDB database. The research is conducted based on the following questions:

1. The usability of the proposed system, does it meet the user's needs?
2. What is the advantage and disadvantage of NoSQL database over relational database?
3. What features do users expect, what information do users need when interact with the system?

2 Literature Review

2.1 Develop web-application

Most papers related to web-application have the same outline: first find out the users' need. Second, find out the best technologies to implement the system. Due to fast development of both internet and web development, technologies change pretty fast. Today, there are many different front end frameworks developed for more interactive and creative pages. As pointed out by Curtis, C. L.[1], he attested that he was given the unique opportunity to develop his own course curriculum using his expertise of blended and project-based learning while integrating the Bethel Public School District FIT Framework and simultaneously incorporating core-subject standards into the course curriculum he created. Also, Garrett, J. J.[8] came up with a new technology to web-application---Ajax. The definition for Ajax in his paper incorporates: standards-based presentation using XHTML and CSS; dynamic display and interaction using the Document Object Model; data interchange and manipulation using XML and XSLT; asynchronous data retrieval using XMLHttpRequest; and JavaScript binding everything together. In addition, Ducasse, S., Lienhard, A., & Renggli, L.[10] employed Seaside. Seaside is a framework which combines an object- oriented approach with a continuation-based one. A Seaside application is built out of components (i.e., objects) and the logic of the application benefits from the continuation-based program flow infrastructure. Seaside offers a unique way to have multiple control flows on a page, one for each component. This enables the developer to write components that are highly reusable and that can be used to compose complex web applications with higher quality in less time.

2.2 Recommendations System

Papers related to recommendations are filled with every field of life McDonald, D. W., & Ackerman, M. S.[3] ‘s work presents an architecture and implementation of the Expertise Recommender system (ER). ER offers several advances over previously existing systems: The architecture is open and flexible enough to address different organizational environments. Also, Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., ... & Sampath, D [2] point out in their present form, their recommendation system is a top-N recommender rather than a predictor. They also review how they evaluate the success of the recommendation system. An additional primary goal for YouTube recommendations is to maintain user privacy and provide explicit control over personalized user data that our back- end systems expose.

2.3 Fitness group class

Papers related to this subject show that many every college gym provides free fitness group class to students and discuss the effect of fitness classes. Frost H, Lamb S E, Moffett J A K, et al[20] aims to assess the long-term effect of a supervised fitness program on patients with chronic low back pain. In addition, some papers discuss whether people are willing to attend this free services. C. Brown, T., Volberding, J., Baghurst, T., & Sellers, J[19] conducted research to determine the reasons for staff and students(N=657; 35 percent males; M_{age}=45.20) at a large Southern university, for either using or not using the free fitness facilities on campus.

3 System Design and Implementation

3.1 Requirement Analysis

Requirement analysis has been conducted to know about user needs for fitness group class. Comments and thoughts from UNC Students were gathered through short interviews. The analysis points out the main functionalities for the whole web-application which satisfy user needs.

Here are some of the important requirements:

1. *Every user who goes to the web-application url address should be able to see the content of fitness group information which is public to everyone.*

Most people prefer more detailed and valuable information about classes. Not only just when and where the class takes place, but also the detailed introduction of the exercise, targeted people and other peoples' comments.

2. *On the main pages, users can only see brief and basic description of different fitness group because of the layout design. If they are willing to see and view more detailed information, they can click the button to another url address to get more information.*

   Today, people are tired of receiving a large volume of information at one time. They prefer to view information that interests them most. So this design and layout is effective as well as efficient.

3. *If the detailed description of class is allowed to leave comments, users have to log in to do make any comments. But they are always allowed to view other people's comments but don't have have right to edit them.*

   It turns out people think others' comments about the class are helpful. So it is of great importance to have this functionality. However, for security reasons, people only have rights to edit their own comments. Without this restriction, there's no point of comments section.

4. *Security features. For normal user, they can only view the content and leave the comment when they are logged in. But there should be a user with privileges that can add/edit/delete the content and information of fitness group information. And when adding new group class information entry, authorized user can choose to allow comments or not.*

People are willing to see updated and latest information. It's not helpful if the class is no longer provided but class information still shows up on the page.

There are other requirements proposed by users but not listed above. For example, it's helpful if users can search classes according to different qualifiers. Or users can leave image or video comments, not just text comments. However, due to limited time and energy these functionalities only be considered for future improvement of the proposed system.

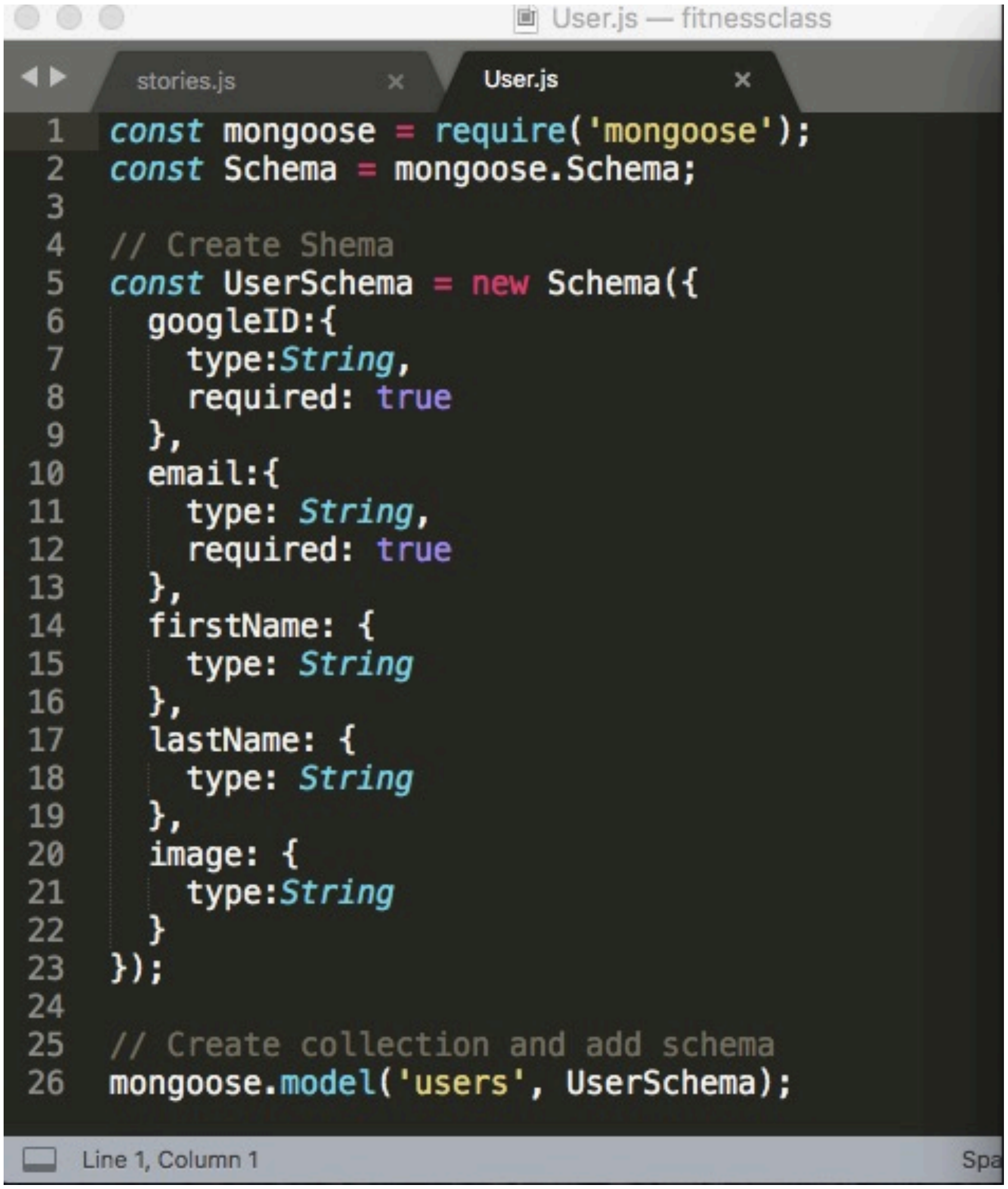
3.2 Architecture

3.2.1 Database Design

Database stores related data for the whole project and user information and fitness group are the two models required in the project.

There are two data model to store data.

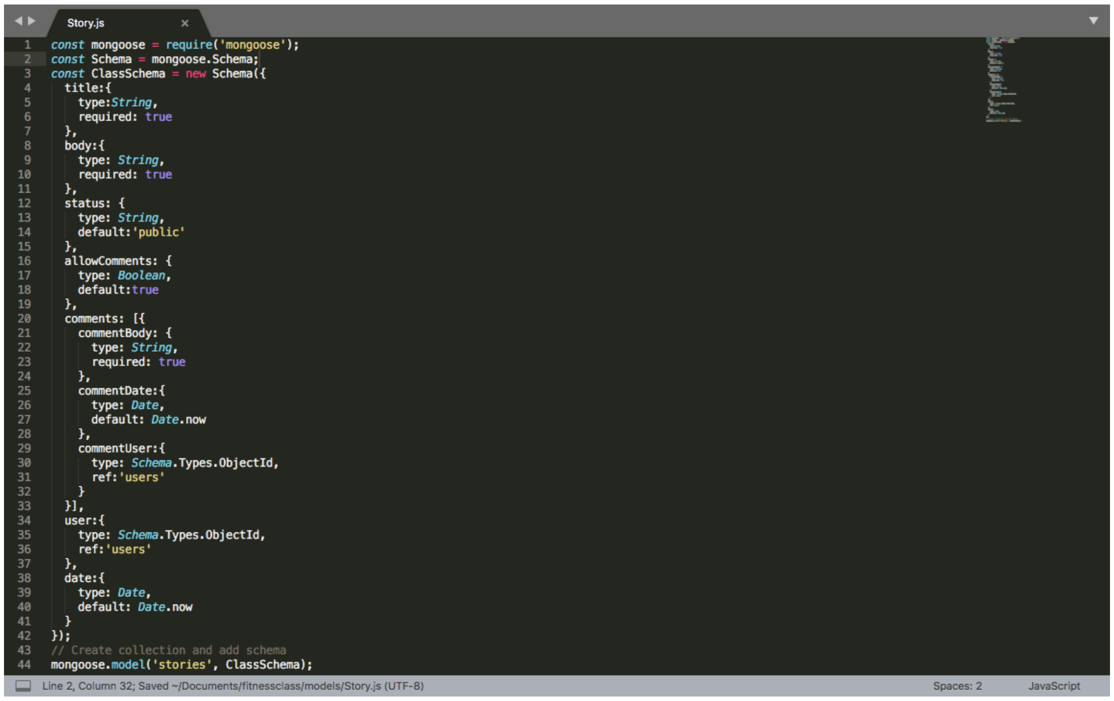
1. User Model: to store Google account information about login user and check where it is authorized user chengpengli88@gmail.com



```
const mongoose = require('mongoose');
const Schema = mongoose.Schema;

// Create Shema
const UserSchema = new Schema({
  googleID:{
    type:String,
    required: true
  },
  email:{
    type: String,
    required: true
  },
  firstName: {
    type: String
  },
  lastName: {
    type: String
  },
  image: {
    type:String
  }
});

// Create collection and add schema
mongoose.model('users', UserSchema);
```

Figure 1

2 fitness group class model: to store information about fitness class group that users will see.



```
const mongoose = require('mongoose');
const Schema = mongoose.Schema;
const ClassSchema = new Schema({
  title:{
    type:String,
    required: true
  },
  body:{
    type: String,
    required: true
  },
  status: {
    type: String,
    default:'public'
  },
  allowComments: {
    type: Boolean,
    default:true
  },
  comments: [{
    commentBody: {
      type: String,
      required: true
    },
    commentDate:{
      type: Date,
      default: Date.now
    },
    commentUser:{
      type: Schema.Types.ObjectId,
      ref:'users'
    }
  }],
  user:{
    type: Schema.Types.ObjectId,
    ref:'users'
  },
  date:{
    type: Date,
    default: Date.now
  }
});
// Create collection and add schema
mongoose.model('stories', ClassSchema);
```

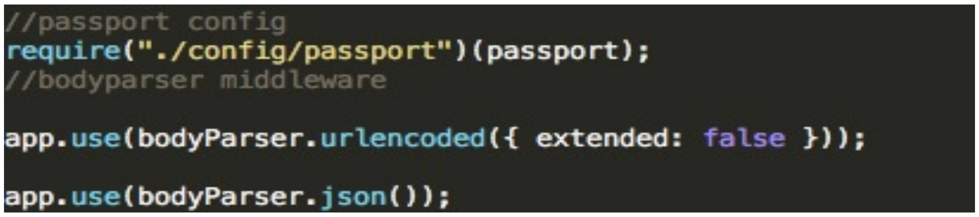
```
Line 2, Column 32; Saved ~/Documents/fitnessclass/models/Story.js (UTF-8)        Spaces: 2        JavaScript
```

Figure 2

3.2.2 Set up

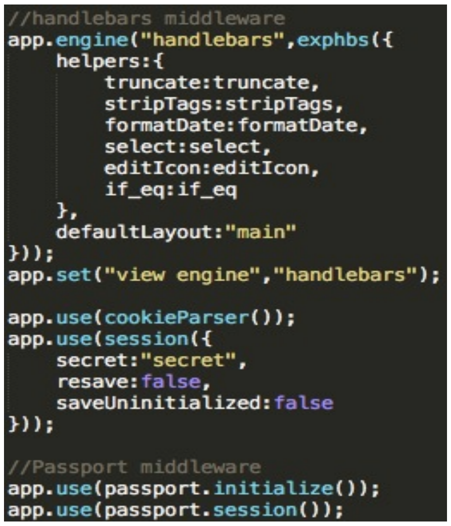
For the implementation of the project, some processes like middleware implementations or environment implementations should be set up before the main programming process. This section describes how to implement these functionalities which make the main functionality programming easier.

1) Add third-party middleware to add functionality to app.js (index file)



```
//passport config
require("./config/passport")(passport);
//bodyparser middleware

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

app.use(bodyParser.json());
```

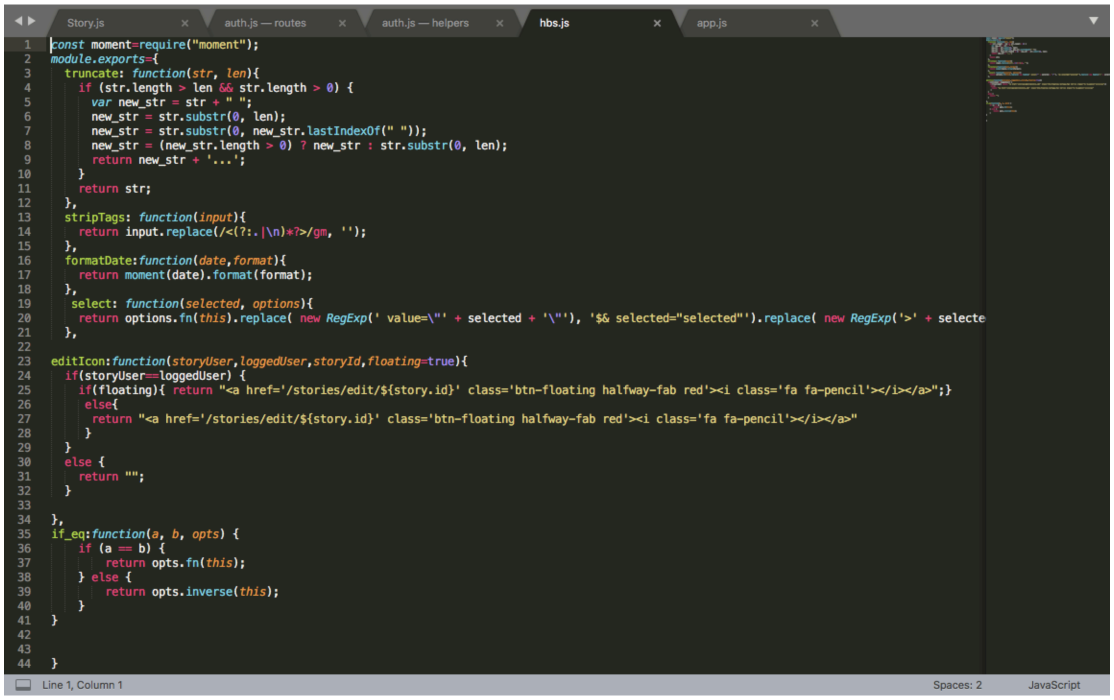
Figure 3



```
//handlebars middleware
app.engine("handlebars",exphbs({
    helpers:{
        truncate:truncate,
        stripTags:stripTags,
        formatDate:formatDate,
        select:select,
        editIcon:editIcon,
        if_eq:if_eq
    },
    defaultLayout:"main"
}));
app.set("view engine","handlebars");

app.use(cookieParser());
app.use(session({
    secret:"secret",
    resave:false,
    saveUninitialized:false
}));

//Passport middleware
app.use(passport.initialize());
app.use(passport.session());
```

Figure 4

2) Add helper function for handlebars engine

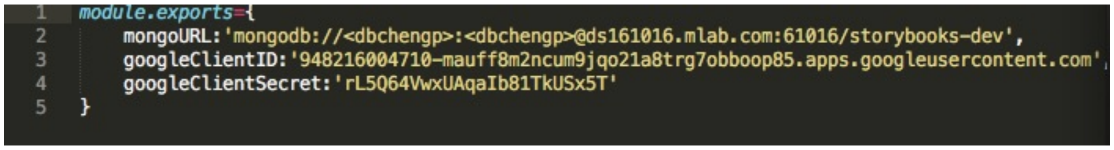


```
const moment=require("moment");
module.exports={
  truncate: function(str, len){
    if (str.length > len && str.length > 0) {
      var new_str = str + " ";
      new_str = str.substr(0, len);
      new_str = str.substr(0, new_str.lastIndexOf(" "));
      new_str = (new_str.length > 0) ? new_str : str.substr(0, len);
      return new_str + '...';
    }
    return str;
  },
  stripTags: function(input){
    return input.replace(/<(?:.|\n)*?>/gm, '');
  },
  formatDate:function(date,format){
    return moment(date).format(format);
  },
   select: function(selected, options){
    return options.fn(this).replace( new RegExp(' value=\"' + selected + '\"'), '$& selected="selected"').replace( new RegExp('>' + selecte
  },

editIcon:function(storyUser,loggedUser,storyId,floating=true){
  if(storyUser==loggedUser) {
    if(floating){ return "<a href='/stories/edit/${story.id}' class='btn-floating halfway-fab red'><i class='fa fa-pencil'></i></a>";}
     else{
      return "<a href='/stories/edit/${story.id}' class='btn-floating halfway-fab red'><i class='fa fa-pencil'></i></a>"
     }
  }
  else {
    return "";
  }

},
if_eq:function(a, b, opts) {
    if (a == b) {
        return opts.fn(this);
    } else {
        return opts.inverse(this);
    }
}


}
```

Figure 5

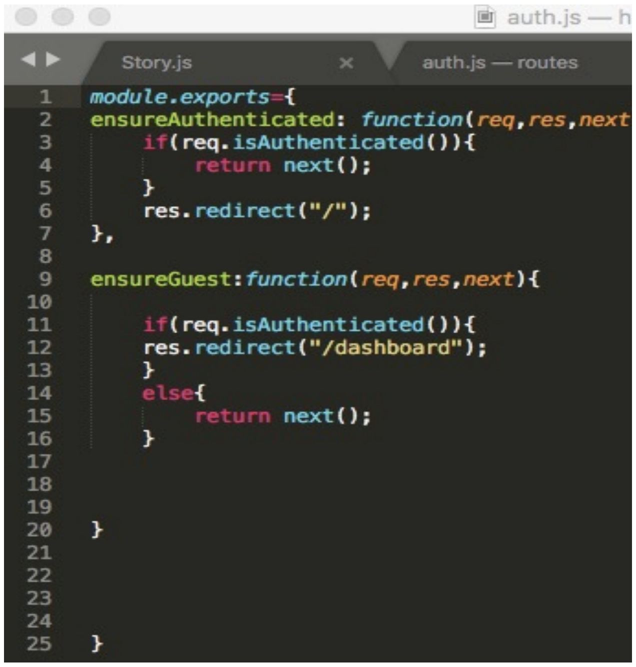
3.2.3 Log in implementation:

This section describes the implementation of log in functionality. Google OAth2 was employed to implement log in functionality and Use Google officially supported Node.js client library for accessing Google APIs.



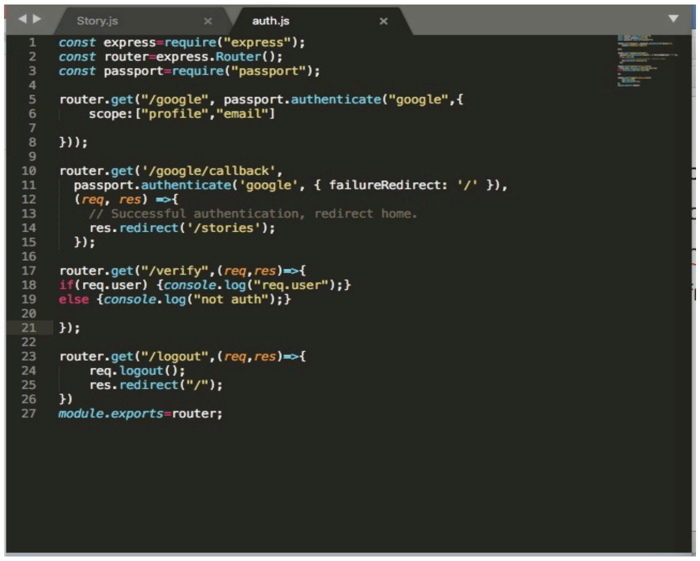
```
module.exports={
    mongoURL:'mongodb://<dbchengp>:<dbchengp>@ds161016.mlab.com:61016/storybooks-dev',
    googleClientID:'948216004710-mauff8m2ncum9jqo21a8trg7obboop85.apps.googleusercontent.com'
    googleClientSecret:'rL5Q64VwxUAqaIb81TkUSx5T'
}
```

Figure 6



```
module.exports={
ensureAuthenticated: function(req,res,next
    if(req.isAuthenticated()){
        return next();
    }
    res.redirect("/");
},

ensureGuest:function(req,res,next){

    if(req.isAuthenticated()){
    res.redirect("/dashboard");
    }
    else{
        return next();
    }



}




}
```

Figure 7



```
const express=require("express");
const router=express.Router();
const passport=require("passport");

router.get("/google", passport.authenticate("google",{
    scope:["profile","email"]

}));

router.get('/google/callback',
  passport.authenticate('google', { failureRedirect: '/' }),
  (req, res) =>{
    // Successful authentication, redirect home.
    res.redirect('/stories');
  });

router.get("/verify",(req,res)=>{
if(req.user) {console.log("req.user");}
else {console.log("not auth");}

});

router.get("/logout",(req,res)=>{
    req.logout();
    res.redirect("/");
})
module.exports=router;
```

Figure 8

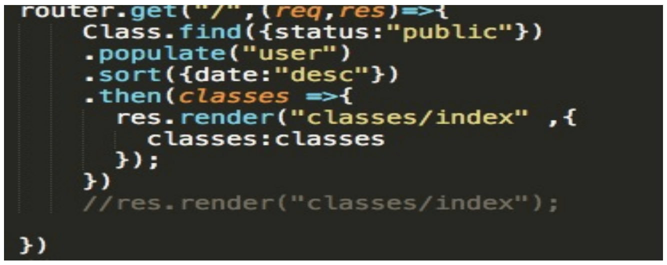
3.2.4 Show/update information

The main functionality of the project is to provide fitness class information for users.

This section shows how to implement show/update information functionality.

1) Index all class Information from database:

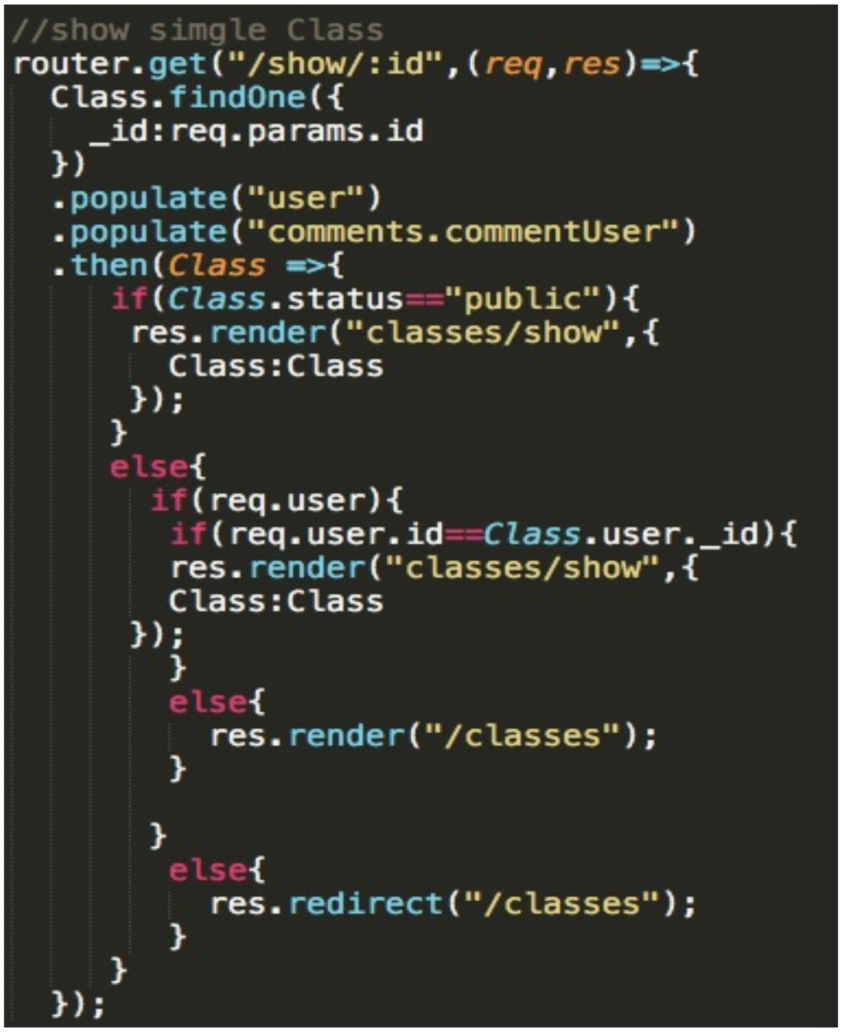
In class Router, ‘/’is the RESTAPI to retrieve all data.



```
router.get("/",(req,res)=>{
    Class.find({status:"public"})
    .populate("user")
    .sort({date:"desc"})
    .then(classes =>{
        res.render("classes/index" ,{
            classes:classes
        });
    })
    //res.render("classes/index");

})
```

Figure 9

In class Router, ‘/index/:id’ is the RESTAPI to retrieve specific data

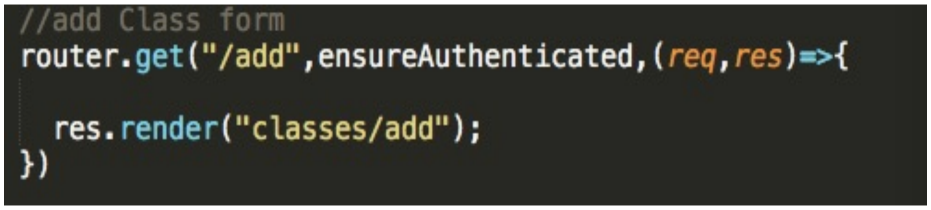


```
//show simgle Class
router.get("/show/:id",(req,res)=>{
  Class.findOne({
    _id:req.params.id
  })
  .populate("user")
  .populate("comments.commentUser")
  .then(Class =>{
     if(Class.status=="public"){
      res.render("classes/show",{
        Class:Class
      });
     }
     else{
       if(req.user){
        if(req.user.id==Class.user._id){
        res.render("classes/show",{
        Class:Class
      });
        }
        else{
          res.render("/classes");
        }

       }
        else{
          res.redirect("/classes");
        }
     }
  });
```

Figure 10

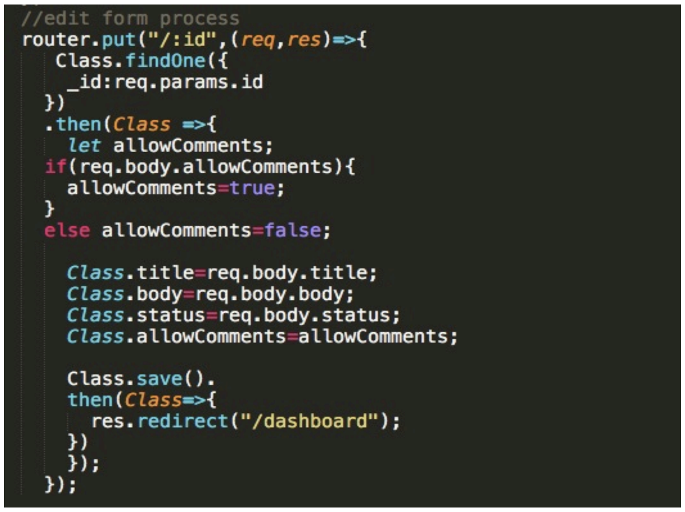
2) Add new class Information:

In class Router, '/add' is the RESTAPI for adding new class info. First retrieve the add form and then make POST request to save the data to database.



```
//add Class form
router.get("/add",ensureAuthenticated,(req,res)=>{

  res.render("classes/add");
})
```

Figure 11

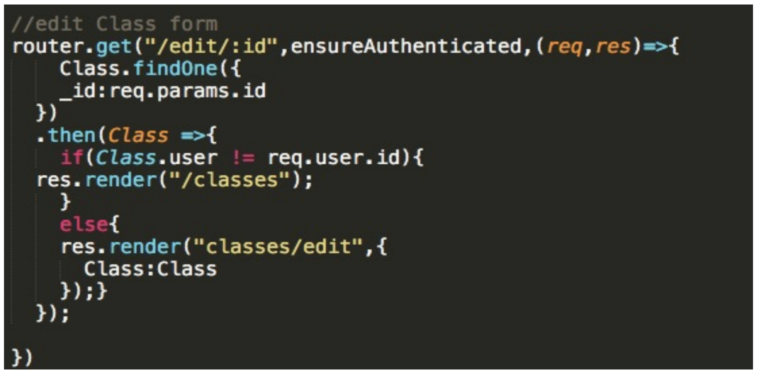


```
//edit form process
router.put("/:id",(req,res)=>{
   Class.findOne({
    _id:req.params.id
  })
  .then(Class =>{
    let allowComments;
  if(req.body.allowComments){
    allowComments=true;
  }
  else allowComments=false;

    Class.title=req.body.title;
    Class.body=req.body.body;
    Class.status=req.body.status;
    Class.allowComments=allowComments;

    Class.save().
    then(Class=>{
      res.redirect("/dashboard");
    })
    });
  });
```

Figure 12

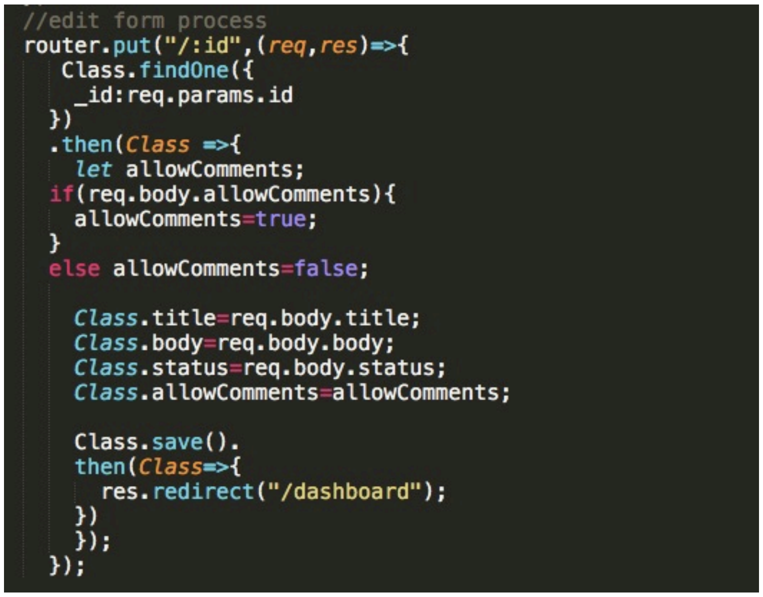
3) Edit class information:

In class Router, '/edit/:id' is the RESTAPI for adding new class info. First retrieve the edit form with relevant data and after editing, make POST request save the new data to database.



```
//edit Class form
router.get("/edit/:id",ensureAuthenticated,(req,res)=>{
    Class.findOne({
    _id:req.params.id
  })
  .then(Class =>{
    if(Class.user != req.user.id){
  res.render("/classes");
    }
    else{
    res.render("classes/edit",{
      Class:Class
    });}
  });

})
```

Figure 13

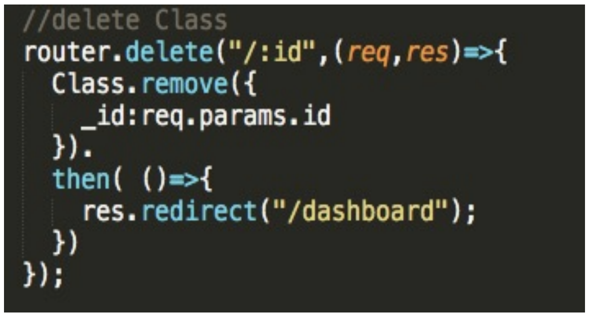


```
//edit form process
router.put("/:id",(req,res)=>{
   Class.findOne({
    _id:req.params.id
  })
  .then(Class =>{
    let allowComments;
  if(req.body.allowComments){
    allowComments=true;
  }
  else allowComments=false;

    Class.title=req.body.title;
    Class.body=req.body.body;
    Class.status=req.body.status;
    Class.allowComments=allowComments;

    Class.save().
    then(Class=>{
      res.redirect("/dashboard");
    })
    });
  });
```

Figure 14

4) Delete class information:

In class Router, '/:id' is the RESTAPI for deleting new class info.

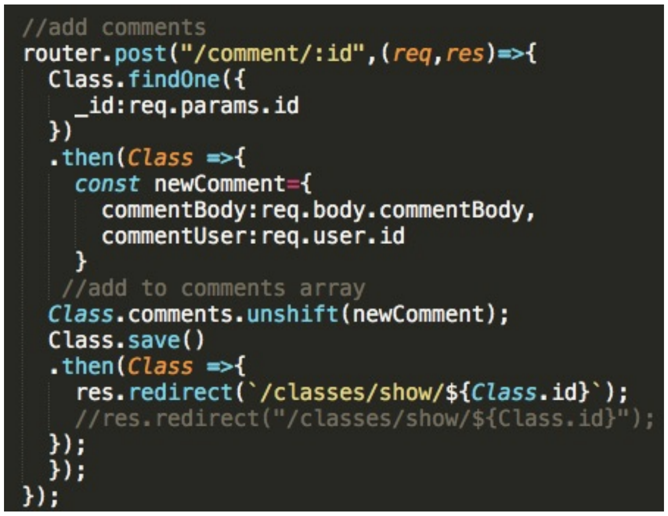


```
//delete Class
router.delete("/:id",(req,res)=>{
  Class.remove({
    _id:req.params.id
  }).
  then( ()=>{
    res.redirect("/dashboard");
  })
});
```

Figure 15

3.2.5 Leave comments

Logged in users are allowed to make comments and this section describes how to implement this functionality.
In class Router, '/comment/:id' is the RESTAPI for adding comments.



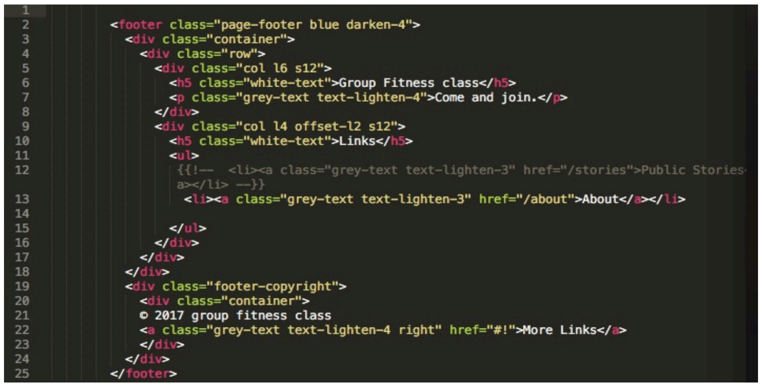
```
//add comments
router.post("/comment/:id",(req,res)=>{
  Class.findOne({
    _id:req.params.id
  })
  .then(Class =>{
    const newComment={
      commentBody:req.body.commentBody,
      commentUser:req.user.id
    }
   //add to comments array
  Class.comments.unshift(newComment);
  Class.save()
  .then(Class =>{
    res.redirect(`/classes/show/${Class.id}`);
    //res.redirect("/classes/show/${Class.id}");
  });
  });
});
```

Figure 16

3.2.6 Header and Footer section code

Every page has the same header and footer partial and thus it makes sense to separate them apart from body content.

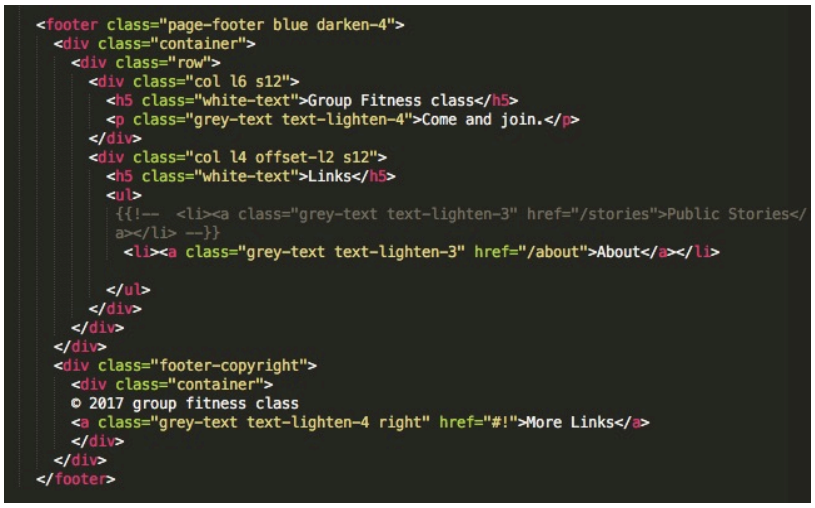
Header:



```

<footer class="page-footer blue darken-4">
  <div class="container">
    <div class="row">
      <div class="col l6 s12">
        <h5 class="white-text">Group Fitness class</h5>
        <p class="grey-text text-lighten-4">Come and join.</p>
      </div>
      <div class="col l4 offset-l2 s12">
        <h5 class="white-text">Links</h5>
        <ul>
         {{!--  <li><a class="grey-text text-lighten-3" href="/stories">Public Stories
         a></li> --}}
          <li><a class="grey-text text-lighten-3" href="/about">About</a></li>

        </ul>
      </div>
    </div>
  </div>
  <div class="footer-copyright">
    <div class="container">
    © 2017 group fitness class
    <a class="grey-text text-lighten-4 right" href="#!">More Links</a>
    </div>
  </div>
</footer>
```

Figure 17

Footer:



```
<footer class="page-footer blue darken-4">
  <div class="container">
    <div class="row">
      <div class="col l6 s12">
        <h5 class="white-text">Group Fitness class</h5>
        <p class="grey-text text-lighten-4">Come and join.</p>
      </div>
      <div class="col l4 offset-l2 s12">
        <h5 class="white-text">Links</h5>
        <ul>
          {{!--  <li><a class="grey-text text-lighten-3" href="/stories">Public Stories</
          a></li> --}}
           <li><a class="grey-text text-lighten-3" href="/about">About</a></li>

        </ul>
      </div>
    </div>
  </div>
  <div class="footer-copyright">
    <div class="container">
    © 2017 group fitness class
    <a class="grey-text text-lighten-4 right" href="#!">More Links</a>
    </div>
  </div>
</footer>
```

Figure 18

3.3 Interface

As mentioned above, the web-application implements basically 4 main functionalities. This section describes how interface was designed and implemented to help to implement the main functionality. As for technology, HTML, CSS, JavaScript as well as Materialize CSS are employed to decorate the components and arrange the main layouts for each page.

a) Login in page:

Login-in page is the first page when users come to the site. Users have the option of logging to see detailed information and additional functionality or just viewing the public class information. Other components like symbols provided by Materialize CSS in header and footer section reside in every page

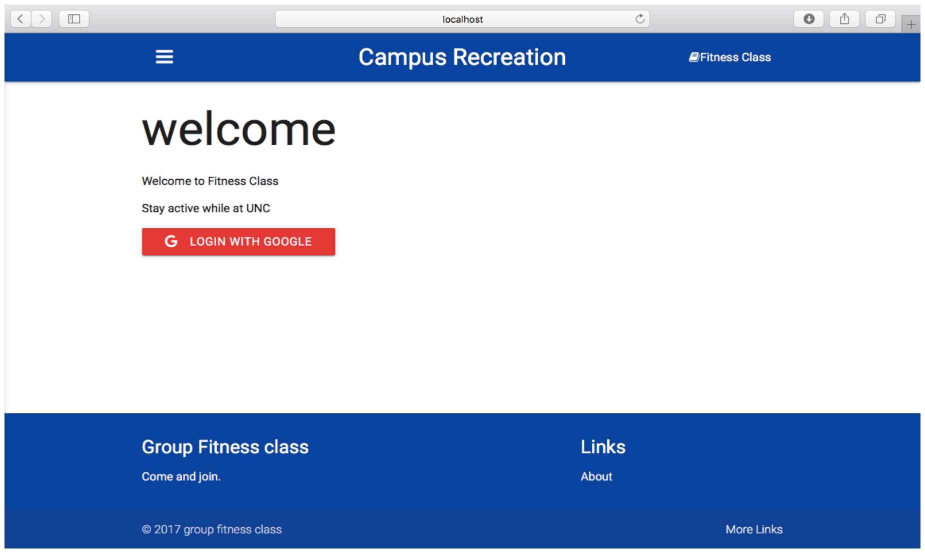


Figure 19

b) Left Section

Its responsive design and beautiful symbols are provided by Materialize CSS. For authorized user chengpengli88@gmail, dashboard is the place to update the contents. For common users, it is just plain page. Fitness Class is open to every user to view public contents of fitness group class content.

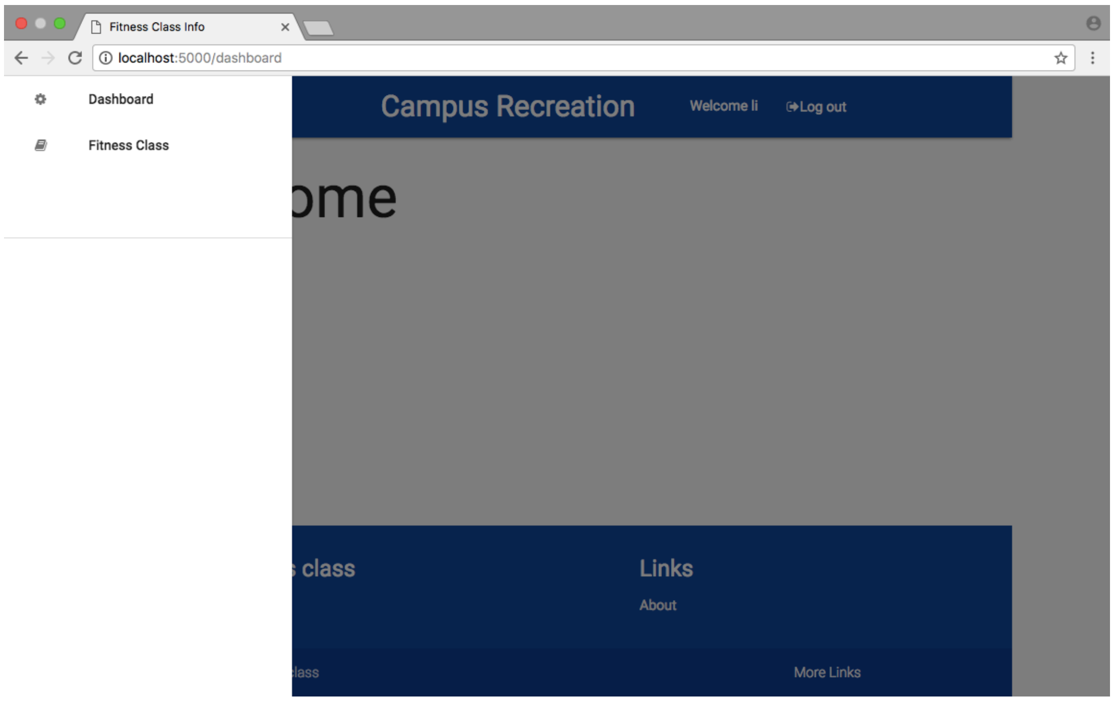
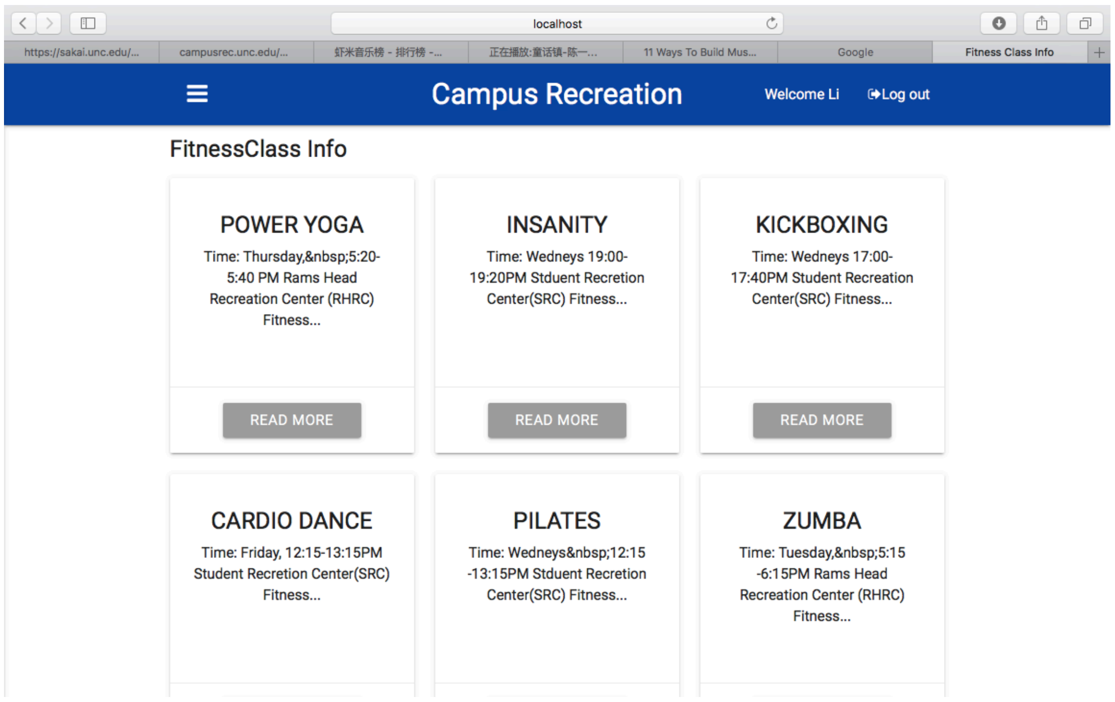


Figure 20

c) Public seen page:

Whether users are logged in or not, they can see preview part of different class information. The layout of the page and each card section design are implemented by Materialize CSS to make it clear and fit. To see detailed information, click the read more button and more information about the class and comments from other users can be seen.



FitnessClass Info

POWER YOGA
Time: Thursday, 5:20-5:40 PM Rams Head Recreation Center (RHRC) Fitness...

INSANITY
Time: Wedneys 19:00-19:20PM Stduent Recreation Center(SRC) Fitness...

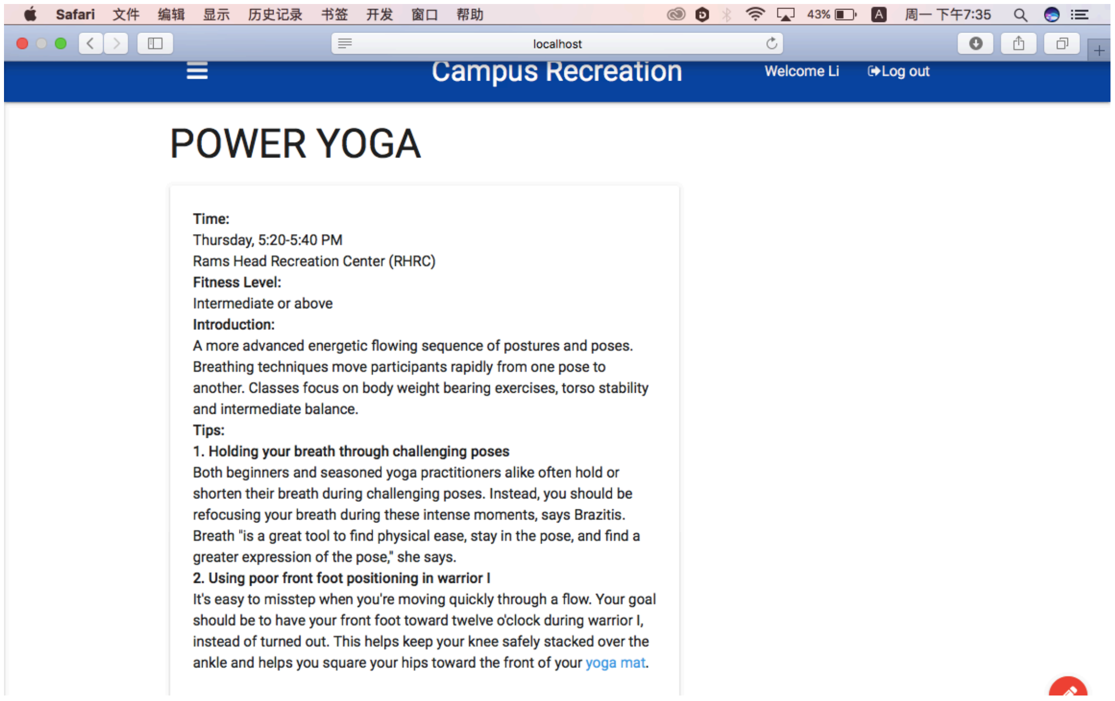
KICKBOXING
Time: Wedneys 17:00-17:40PM Student Recreation Center(SRC) Fitness...

CARDIO DANCE
Time: Friday, 12:15-13:15PM Student Recretion Center(SRC) Fitness...

PILATES
Time: Wedneys 12:15 -13:15PM Stduent Recretion Center(SRC) Fitness...

ZUMBA
Time: Tuesday, 5:15 -6:15PM Rams Head Recreation Center (RHRC) Fitness...

Figure 21



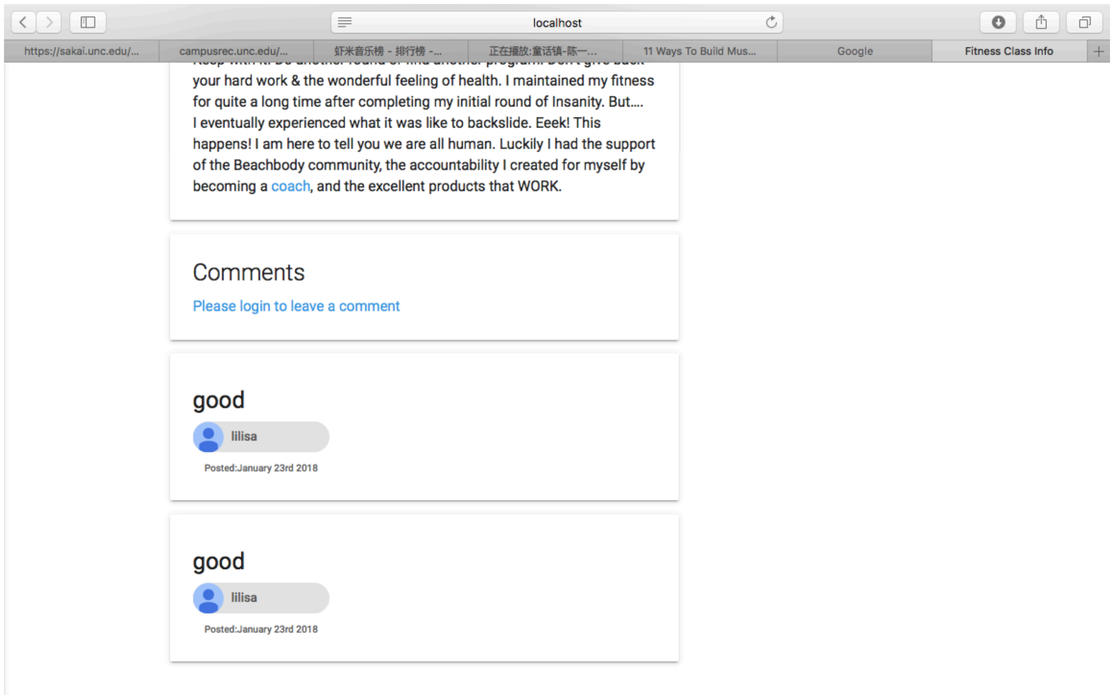
POWER YOGA

Time:
Thursday, 5:20-5:40 PM
Rams Head Recreation Center (RHRC)
Fitness Level:
Intermediate or above
Introduction:
A more advanced energetic flowing sequence of postures and poses. Breathing techniques move participants rapidly from one pose to another. Classes focus on body weight bearing exercises, torso stability and intermediate balance.
Tips:
1. Holding your breath through challenging poses
Both beginners and seasoned yoga practitioners alike often hold or shorten their breath during challenging poses. Instead, you should be refocusing your breath during these intense moments, says Brazitis. Breath "is a great tool to find physical ease, stay in the pose, and find a greater expression of the pose," she says.
2. Using poor front foot positioning in warrior I
It's easy to misstep when you're moving quickly through a flow. Your goal should be to have your front foot toward twelve o'clock during warrior I, instead of turned out. This helps keep your knee safely stacked over the ankle and helps you square your hips toward the front of your yoga mat.

Figure 22

d) Comment page:

If the class information is allowed to make comments, users have to log in to write a comment. But all users are eligible to see comments from others. User name, Gmail user image and the date they made the comment can also be seen.



your hard work & the wonderful feeling of health. I maintained my fitness for quite a long time after completing my initial round of Insanity. But.... I eventually experienced what it was like to backslide. Eeek! This happens! I am here to tell you we are all human. Luckily I had the support of the Beachbody community, the accountability I created for myself by becoming a coach, and the excellent products that WORK.

Comments

Please login to leave a comment

good

lilisa

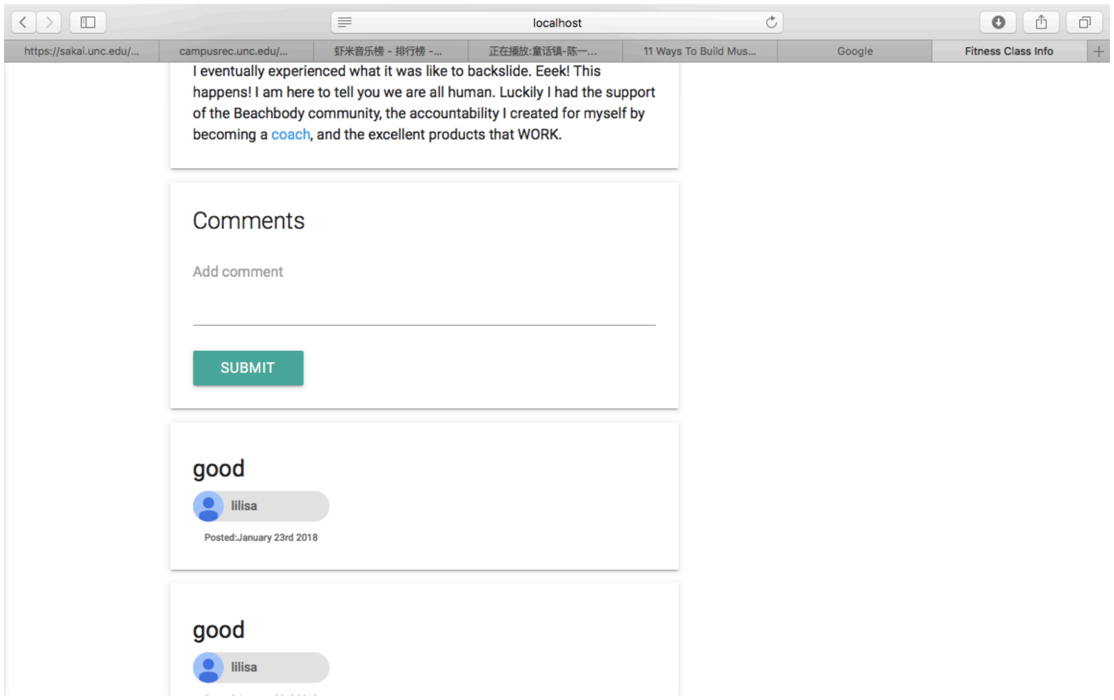
Posted:January 23rd 2018

good

lilisa

Posted:January 23rd 2018

Figure 23



I eventually experienced what it was like to backslide. Eeek! This happens! I am here to tell you we are all human. Luckily I had the support of the Beachbody community, the accountability I created for myself by becoming a coach, and the excellent products that WORK.

Comments

Add comment

SUBMIT

good

lilisa

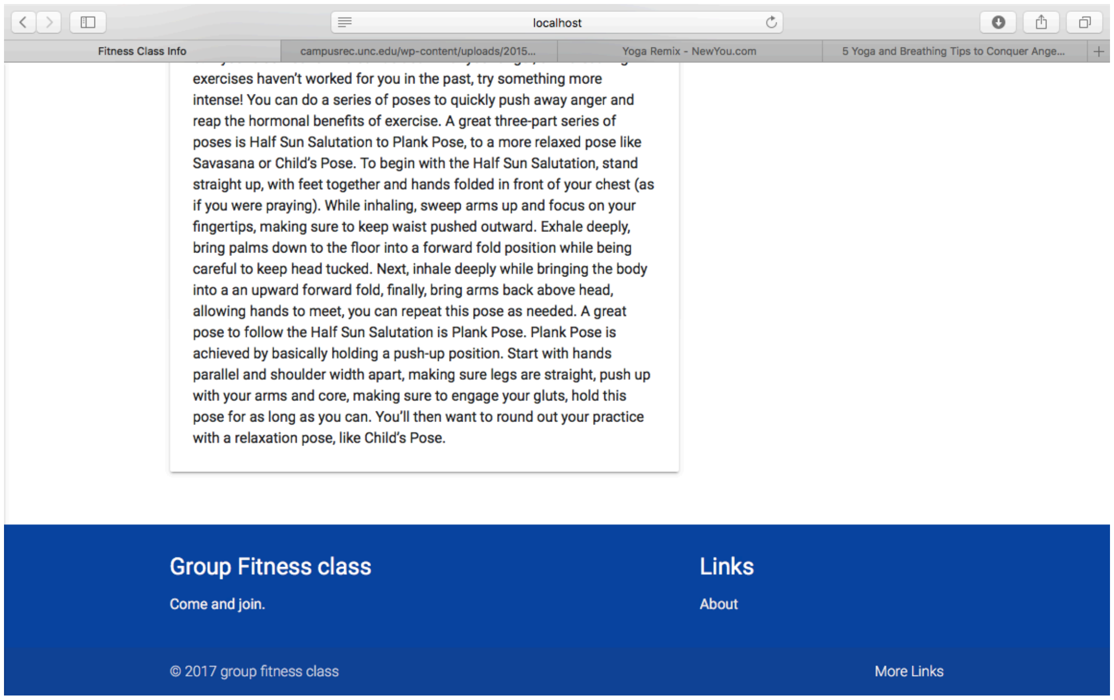
Posted:January 23rd 2018

good

lilisa

Figure 24

If this specify class is not allowed for comments, users cannot find place to make their comments.


localhost
Fitness Class Info
campusrec.unc.edu/wp-content/uploads/2015...
Yoga Remix - NewYou.com
5 Yoga and Breathing Tips to Conquer Ange...

exercises haven't worked for you in the past, try something more intense! You can do a series of poses to quickly push away anger and reap the hormonal benefits of exercise. A great three-part series of poses is Half Sun Salutation to Plank Pose, to a more relaxed pose like Savasana or Child's Pose. To begin with the Half Sun Salutation, stand straight up, with feet together and hands folded in front of your chest (as if you were praying). While inhaling, sweep arms up and focus on your fingertips, making sure to keep waist pushed outward. Exhale deeply, bring palms down to the floor into a forward fold position while being careful to keep head tucked. Next, inhale deeply while bringing the body into a an upward forward fold, finally, bring arms back above head, allowing hands to meet, you can repeat this pose as needed. A great pose to follow the Half Sun Salutation is Plank Pose. Plank Pose is achieved by basically holding a push-up position. Start with hands parallel and shoulder width apart, making sure legs are straight, push up with your arms and core, making sure to engage your gluts, hold this pose for as long as you can. You'll then want to round out your practice with a relaxation pose, like Child's Pose.

Group Fitness class

Come and join.

Links

About

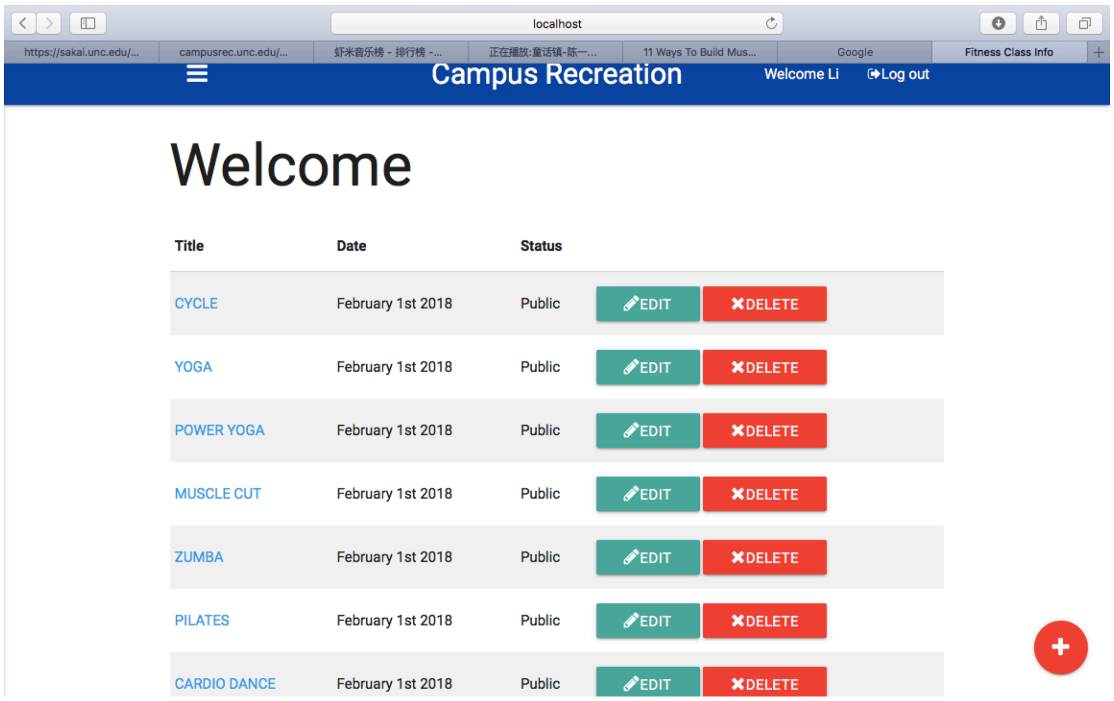
© 2017 group fitness class

More Links

Figure 25

e) Add new fitness class page:

If logged in user is authorized like user chengpengli88@gmail.com, from the dashboard, there is floating plus circle button at the bottom of the page for adding new fitness class entry. It is designed to always float at fixed position and its design is provided by Materialize CSS. After clicking the circle button, authorized user come to the add page. Then users can provide title and description body information for new class information. Authorized users can choose to make comments allowed or not.



Welcome

Title	Date	Status		
CYCLE	February 1st 2018	Public	EDIT	DELETE
YOGA	February 1st 2018	Public	EDIT	DELETE
POWER YOGA	February 1st 2018	Public	EDIT	DELETE
MUSCLE CUT	February 1st 2018	Public	EDIT	DELETE
ZUMBA	February 1st 2018	Public	EDIT	DELETE
PILATES	February 1st 2018	Public	EDIT	DELETE
CARDIO DANCE	February 1st 2018	Public	EDIT	DELETE

Figure 26

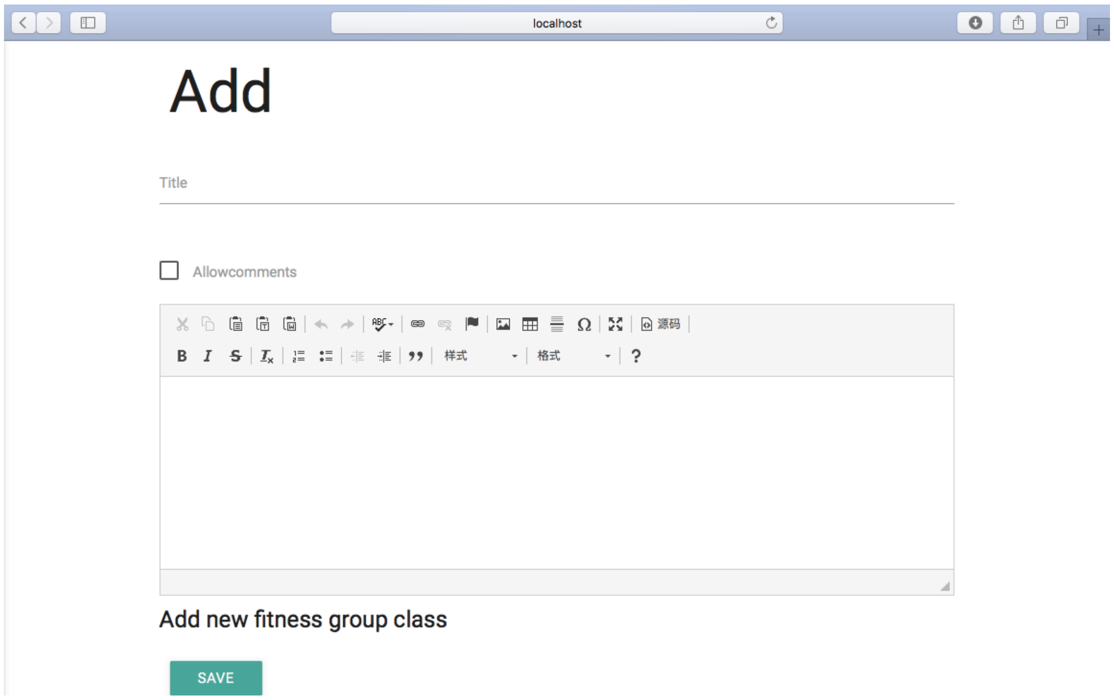
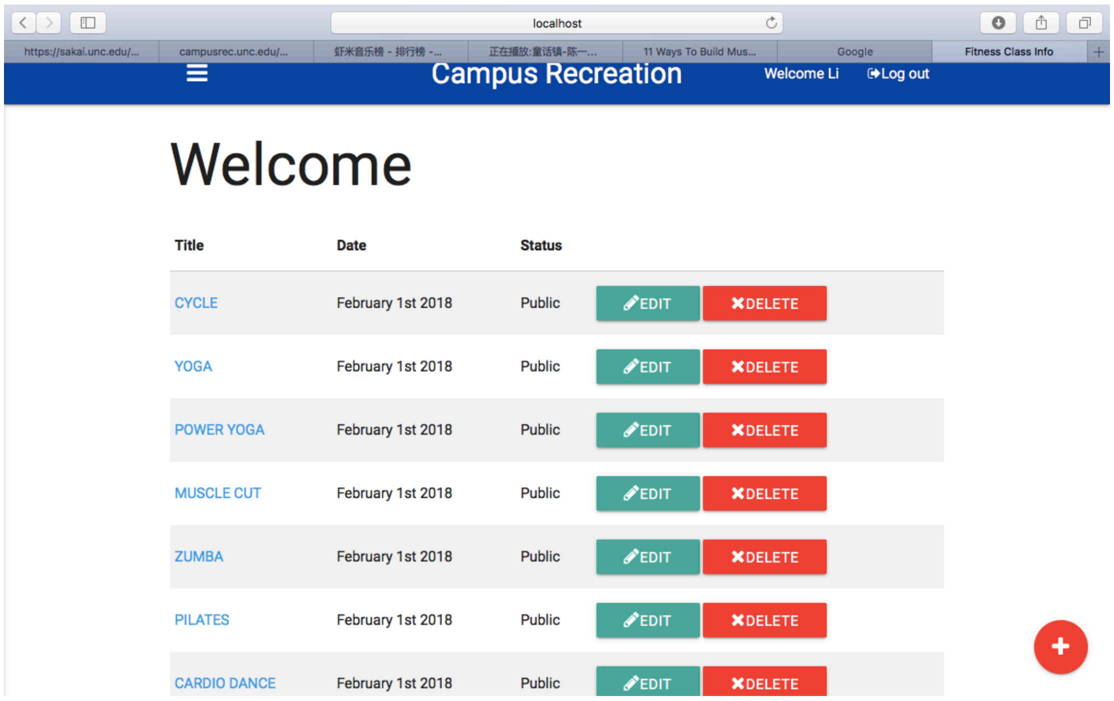


Figure 27

f) Edit fitness page:

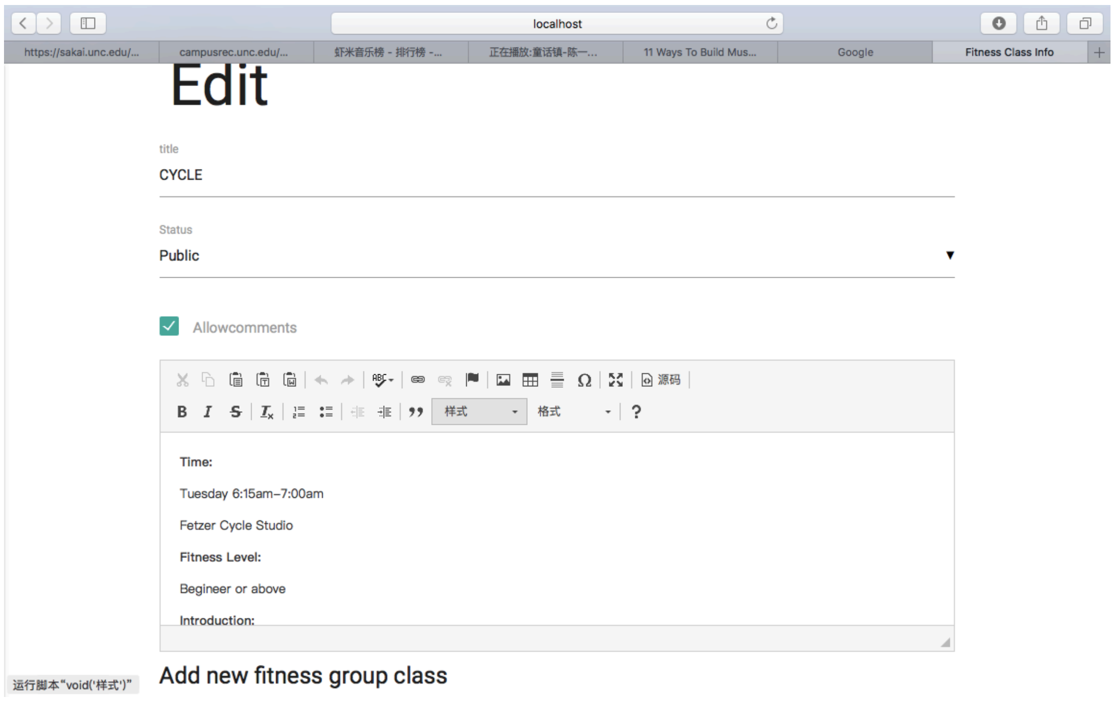
If logged in user is authorized like user chengpengli88@gmail.com, from the dashboard, there is an option of editing the class information which link to the actual page of editing,



Welcome

Title	Date	Status	
CYCLE	February 1st 2018	Public	EDIT DELETE
YOGA	February 1st 2018	Public	EDIT DELETE
POWER YOGA	February 1st 2018	Public	EDIT DELETE
MUSCLE CUT	February 1st 2018	Public	EDIT DELETE
ZUMBA	February 1st 2018	Public	EDIT DELETE
PILATES	February 1st 2018	Public	EDIT DELETE
CARDIO DANCE	February 1st 2018	Public	EDIT DELETE

Figure 28



Edit

title

CYCLE

Status

Public ▼

✓ Allowcomments

Time:

Tuesday 6:15am–7:00am

Fetzer Cycle Studio

Fitness Level:

Begineer or above

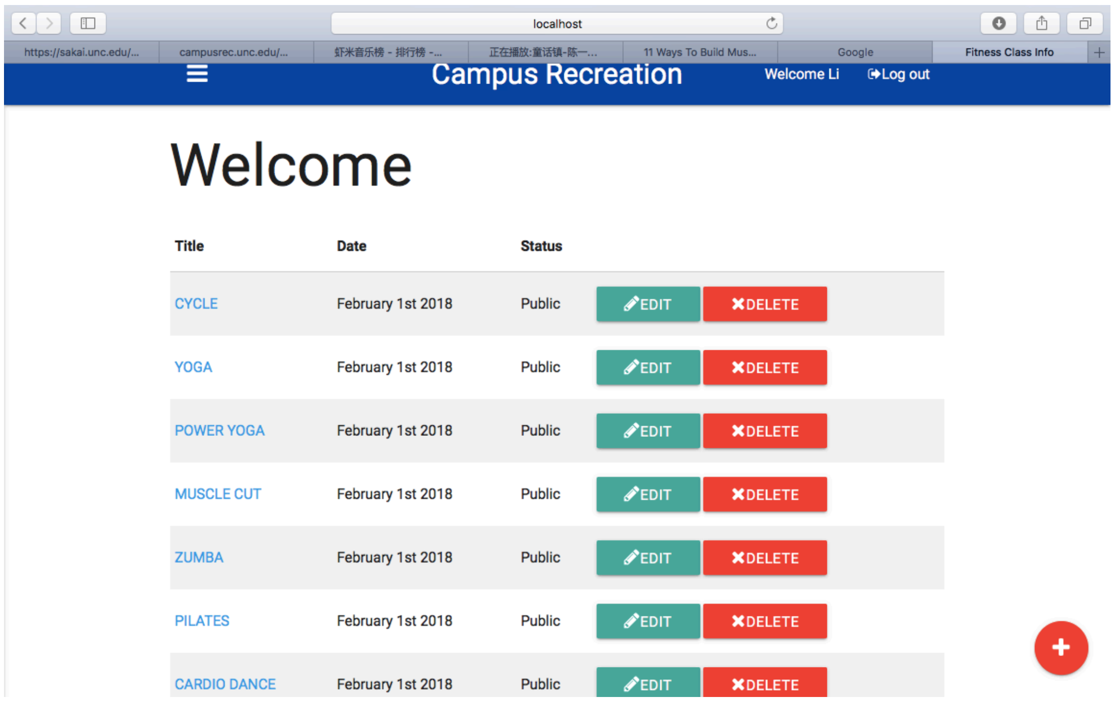
Introduction:

Add new fitness group class

Figure 29

g) Delete fitness class page:

If logged in user is authorized user like chengpengli88@gmail.com, from dashboard page, there are options of deleting class entries. Deletion button design is provided by Materialize CSS.



Welcome

Title	Date	Status	
CYCLE	February 1st 2018	Public	EDIT DELETE
YOGA	February 1st 2018	Public	EDIT DELETE
POWER YOGA	February 1st 2018	Public	EDIT DELETE
MUSCLE CUT	February 1st 2018	Public	EDIT DELETE
ZUMBA	February 1st 2018	Public	EDIT DELETE
PILATES	February 1st 2018	Public	EDIT DELETE
CARDIO DANCE	February 1st 2018	Public	EDIT DELETE

Figure 30

h) If the logged in user is not chengpengli88@gmail, there are no options of adding/editing/deleting on dashboard page.

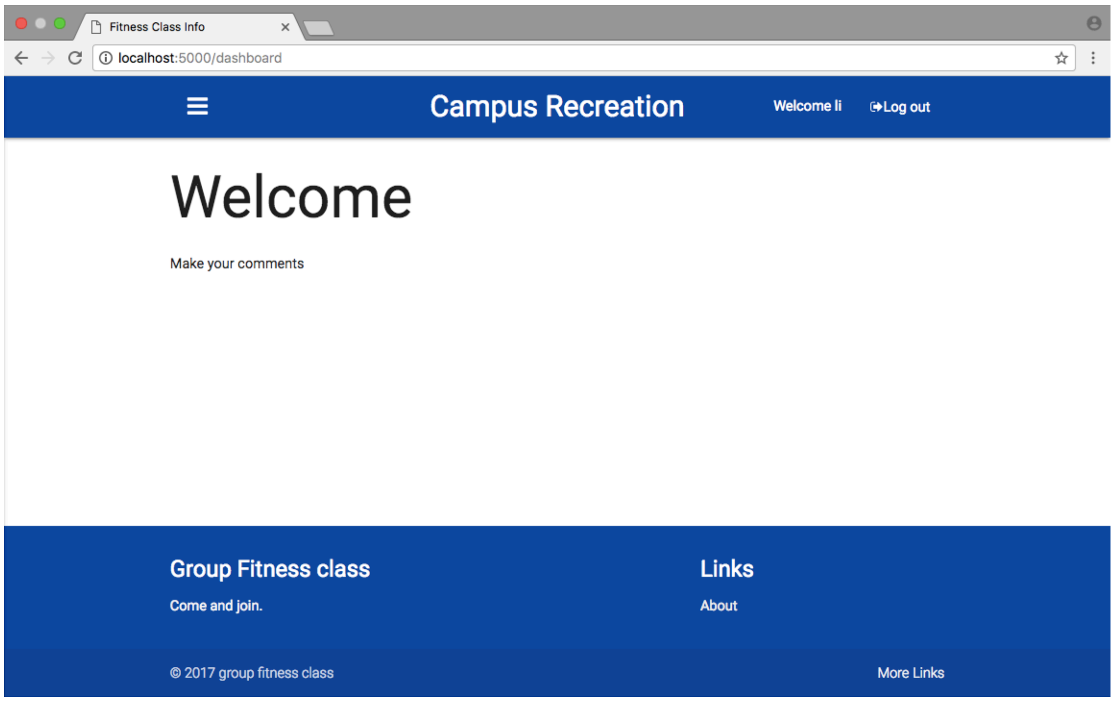


Figure 31

4 Conclusion

4.1 Answers to Research Questions

1. The usability of the proposed system, does it meet the user's needs?

The experiment showed that, in general, the usability of the system is good. Users are happy with the information proposed system provides. In addition, the results showed the system is easy to use and users are willing to do exercises at UNC Campus Recreation more.

2 What is the advantage and disadvantage of NoSQL database over relational database?

NoSQL databases are gaining importance and it is considered as an alternative model for traditional databases. First of all, due to its elastic scaling property, it's easy to expand new functionalities for future improvement and make any changes to the system. Also, the data volume managed by RDBMS is impossible for future improvement. Although there’s limited data for current system, a large volume of data need to be stored for future use. But this drawback can be overcome by NoSQL as NoSQL system helps to handle large volume data such as Hadoop.

3 What features do users expect, what information do users need when interact with the system?

In most cases, the information needed by users is: finding detailed information about fitness group classes, leaving comments and reviewing comments. For authorized users, they can edit and update the class information. The exiting information provided by UNC Campus Recreation Center does have basic information like timeline. However, it cannot provide more additional and crucial information. In addition, some test subjects showed their willingness of going to fitness group class more often because of the detailed information provided by proposed system.

4.2 Problems encountered in the project

Three main problems were encountered during the project. They are requirement analysis, software implementation and writing the master paper. How to gather thoughts and ideas from users and summarize it to main functionality for the proposed system is the first major problem. However, what I have learnt form INLS582 guided me through the whole process and finally helped me to outline expected functionalities for this proposed system. For the second step, courses from Computer Science were helpful for the implementation of the whole system, which were the major part of the system. Finally, INLS 781 course was helpful for writing the whole master paper. It guided me through how to write a proposal, how to define research questions, how to conduct interviews and how to organize a good master paper.

4.3 Suggested future development

The main future work for this system could be additional functionality extension. This application could provide search options for users to seek the most popular class according to how many times each class information is clicked to view detailed information. In addition, users could be allowed to provide additional and detailed comments like uploading pictures or videos of each fitness class group. Then the comments section will be more useful for other people.

References:

[1] Curtis, C. L. (2014). Implementing Technological Change: Effects on Student Learning Through Implementation of a Learning Management System for Enhanced Two-Way Communication between School, Students and Parents.
[2] Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., ... & Sampath, D. (2010, September). The YouTube video recommendation system. In *Proceedings of the fourth ACM conference on Recommender systems* (pp. 293-296). ACM.
[3] McDonald, D. W., & Ackerman, M. S. (2000, December). Expertise recommender: a flexible recommendation system and architecture. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work* (pp. 231-240). ACM.
[4]Gąsiorowski, R. (2006). Utilizing. Semantic. Web. and. Software. Agents. in. a. Travel. Support. System. *Semantic Web Technologies and E-Business: Toward the Integrated Virtual Organization and Business Process Automation: Toward the Integrated Virtual Organization and Business Process Automation*, 325.
[5] Markey, K., Swanson, F., Jenkins, A., Jennings, B., St Jean, B., Rosenberg, V., ... & Frost, R. L. (2008). *Engaging undergraduates in research through a storytelling and gaming strategy*. School of Information.
[6] Liu, H. H. (2011). *Software performance and scalability: a quantitative approach* (Vol. 7). John Wiley & Sons.
[7] Conallen, J. (2002). *Building Web applications with UML*. Addison-Wesley Longman Publishing Co., Inc..
[8] Garrett, J. J. (2005). Ajax: A new approach to web applications.
[9] Colton, P., & Sarid, U. (2009). *U.S. Patent No. 7,596,620*. Washington, DC: U.S. Patent and Trademark Office.
[10] Ducasse, S., Lienhard, A., & Renggli, L. (2004). Seaside—a multiple control flow web application framework.
[11] Huang, Y. W., Huang, S. K., Lin, T. P., & Tsai, C. H. (2003, May). Web application security assessment by fault injection and behavior monitoring. In *Proceedings of the 12th international conference on World Wide Web* (pp. 148-159). ACM.
[12] Wing, C. H. (2014). The evolution of group fitness: Shaping the history of fitness. *ACSM's Health & Fitness Journal*, *18*(6), 5-7.
[13] Leslie, E., Fotheringham, M., Owen, N., & Veitch, J. (2000). A university campus physical activity promotion program. *Health Promotion Journal of Australia: Official Journal of Australian Association of Health Promotion Professionals*, *10*(1), 51.
[14] Huang, Y. W., Yu, F., Hang, C., Tsai, C. H., Lee, D. T., & Kuo, S. Y. (2004, May). Securing web application code by static analysis and runtime protection. In *Proceedings of the 13th international conference on World Wide Web* (pp. 40-52). ACM.
[15] LIANG, M., & WANG, W. (2006). Developing Web Application Based on AJAX [J]. *Computer Knowledge and Technology (Academic Exchange)*, *5*, 063.
[16] Russo, N. L., & Graham, B. R. (1999). A first step in developing a Web application design methodology: understanding the environment. In *Methodologies for Developing and Managing Emerging Technology Based Information Systems* (pp. 24-33). Springer, London.
[17] Shan, T. C., & Hua, W. W. (2006, October). Taxonomy of java web application frameworks. In *E-Business Engineering, 2006. ICEBE'06. IEEE International Conference on* (pp. 378-385). IEEE.

[18] Knight, S. A., & Burn, J. (2005). Developing a framework for assessing information quality on the World Wide Web. *Informing Science*, *8*.

[19] C. Brown, T., Volberding, J., Baghurst, T., & Sellers, J. (2014). Faculty/staff perceptions of a free campus fitness facility. *International Journal of Workplace Health Management*, *7*(3), 156-170.

[20] Frost, H., Lamb, S. E., Moffett, J. K., Fairbank, J. C. T., & Moser, J. S. (1998). A fitness programme for patients with chronic low back pain: 2-year follow-up of a randomised controlled trial. *Pain*, *75*(2-3), 273-279.

APPENDIX I POSTTEST QUESTIONNAIRE

1. It is easy to get information about fitness group class using the application:

   ____Strongly Agree
   ____Somewhat Agee
   ____Neutral
   ____Somewhat Disagree
   ____Strongly Disagree

2. It is easy to leave a comment for fitness group class:

   ____Strongly Agree
   ____Somewhat Agee
   ____Neutral
   ____Somewhat Disagree
   ____Strongly Disagree

3. In general, the information provided in this website is helpful:

   ____Strongly Agree
   ____Somewhat Agee
   ____Neutral
   ____Somewhat Disagree
   ____Strongly Disagree

4. I am comfortable to use this system and has no problem viewing the information it provides:

   ____Strongly Agree
   ____Somewhat Agee
   ____Neutral
   ____Somewhat Disagree
   ____Strongly Disagree

5. I encountered problems while using the system and could not solve it:

   ____Strongly Agree
   ____Somewhat Agee
   ____Neutral
   ____Somewhat Disagree

____Strongly Disagree
*Please provide more information:

____
____
____

6. What other features would you like to have in this web-application:

____
____
____
____