COMP 2406 Assignment 4

Muhammad Mustafa

100823576

9th February 2015

……………………………………………………………………………………………....

Questions:

1. [4] In the listings below the following files were not included. What is the high-level purpose of each file, and when is each accessed? (Note that you do not need to refer to these files to answer the following questions.)

   - app.js
   - bin/www
   - views/error.jade
   - package.json

2. [2] Can a user embed a hyperlink in the title of a note? What about in the content portion of the note? Why?

3. [2] The /edit/[note number] page redirects to /notes when it cannot find the requested note. When it does so, it passes an error message of "Unknown Note to Edit" to the page; however, this error is not displayed anywhere except for the URL bar. How could you make /notes display errors in the body of the page?

4. [2] You want to change the font used to render page headers (h1 tags) to in notes-demo to be the same font as the main text, but in 50 point bold italic letters. How would you change it?

5. [2] You want to start playing with jQuery in your web application. Before you do this, however, you need to add code to include jQuery in your app. How would you add client-side jQuery support to notes-demo?

6. [2] How would you add a Cancel button to the Note editor page? It should be a button (like the other buttons in the application) that takes the user back to the notes listing page.

7. [6] How would you add a Date field to each note? The Date field should be shown beside the title in the main notes listing page. The Date should also be modifiable from the edit page.

……………………………………………………………………………………………………

Answers:

1. - App.js is to configure middleware and dispatch routes. It also creates several javascript variables used to tie with different packages. It initializes express and assigns application variables to it. Its principal use is to direct routes and also hold the control logic of the app.

- Bin/www directory is used to specify a location where you can place various scripts that are called at the start-up.

- Views/error.jade logs errors when there is one. It called when we try to access an invalid request

- Package.jason is simply a document that keeps track of your project and any module that your project needs. It gets access each tie your app needs to import a module.

2. A hyperlink cannot be added in the title but it can be added in the content portion. This is because in the note we extent the layout which already has a hyperlink existing in the layout.

3. We add the following lines of code that are in **bold**:

In index.js :

```
router.get('/notes', function(req, res) {
        var username = req.session.username;
        if (username) {
  res.render("notes.jade", {username: username,
                                title: username +"'s Notes",
                                userNotes: userNotes[username],
                                error: req.query.error});
```

Also in notes.jade:

```
block content
 h1 #{username} Account
 - if(error)
        div #{error}
 p Welcome #{username}
```

4. We add the following to CSS:
```
h1 {
font: 50px "Lucida Grande", Helvetica, Arial, sans-serif; font-style: italic; font-weight: bold;
}
```

5. To use the jQuery just reference the file directly from http://code.jquery.com in the script tag in your index.jade

block header

```
   <script src="//code.jquery.com/jquery-1.11.2.min.js"></script>
   <script src="//code.jquery.com/jquery-migrate-1.2.1.min.js"></script>
```

6. add this to the end of edit.jade:

   button(type="button", onclick = " window.location.href='/notes' ") Cancel

7. **Do the following:**

In index.js:

1. Inside the userNotes array for the predefined object "alice" add the field like this - "date": "9th February 2015"

2. Inside- router.post('/update',..), add this - note.date = req.body.date

3. Inside- router.get('/new',..), add this – date: "insert date here"

In edit.jade:

1. add the following after input#title(…) :
div
        input#date(type="text", name="date", value=note.date)

In notes.jade edit the following

1. b= userNotes[i].title + "   " + "(" + userNote[i].date + ")" + " : "

        -var note_url = "/edit/" + i

        a(href=note_url) [edit]

        p!= userNotes[i].content

…………………………………………....- END -……………………………………………