School of Information Technologies
Dr. Ying Zhou

COMP5347: Web Application Development **1.Sem./2017**

Week 9 Part Two: Mongoose and the model layer

09.05.2017

Learning Objectives

- Understand basic mongoose concepts and objects
- Integrate mongoose with Expressjs

Question 1: Add `mongoose` module

Start Eclipse and open the project you have worked in week 6 and week 8. Open file "package.json" and add the following line at the end of the "dependencies"

```
"mongoose": "^4.3.0",
```

Save the file and right click it on the project explorer panel to `Run As` then `npm update`. Alternatively, you can run the command `npm install mongooes --save` from your project's root directory in a command line window to install the `mongooes` package.

Question 2: Simple mongoose query

Download `week9-src.zip` from ELearning and extract the content in a directory. The zip file contains a `models` folder, three JS files,two PUG files and one CSS file. Import `mongoose.revision.js` from `models` folder to `app/modules` in your project folder. The script contains a few examples of running query, update and aggregation with mongoose. Inspect the code in Eclipse. Assume that the Mongodb server is started. You can run the code by right click the file name on project explorer panel and run as `node.js application`.

You need to uncomment the lines representing the update statement to try the update example. You may notice that writing queries,updates and aggregation statements in mongoose follow the same syntax as mongodb shell commands.

Try to write your own code to find the user who made the most non-minor revisions on article "CNN" and run `mongoose.revisions.js` again to test that you have obtained the correct results. You can also compare the results you get from mongoose to the results you get from respective shell command.

Question 3: Full MVC Express Application

This part will built a small application with mongodb as backend database server. It allows an end user to type in an article's title to find its latest revision data from the `revisions` collection. `week9-src.zip` contains majority of the source code of this application.

The view layer contains two views: one for displaying the form for entering title. The other for displaying revision data. Both views are provided as PUG files: `titleForm.pug` and `revision.pug`. Import them to your projects' `app/views` folder.

The model layer is supposed to handle data of the application. Database related code should be put in the model layer. Our simple application only needs to handle revision data, it has a module represent revision mode called `revision.js`. It also has a module called `db.js` which contains database initialization code. You will find the two files from `week9-src.zip` extracted content under folder `models`. Import them to `app/models` in your project and inspect the content.

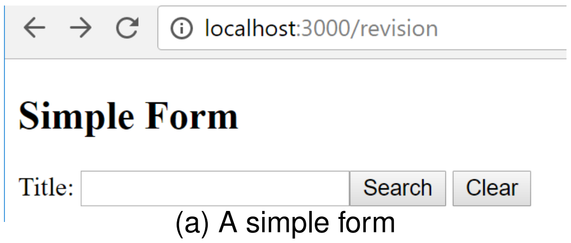
The `revision.js` script provides a static method `findTitleLatestRev`. This method will find an article's latest revision given a title. It does not define its own callback function.

The application also contains a controller `revision.server.controller.js`, which calls `findTitleLatestRev` and provides a callback function when the result is back. Import the controller filer to `app/controllers`.

The `revserver.js` contains code to creat and start the application. Import it to the project's root.

The only part missing is a route file that will bind controller methods to urls. Inspect all relevant code to determine the content of the route file and implement it with proper name under `app/routes`.

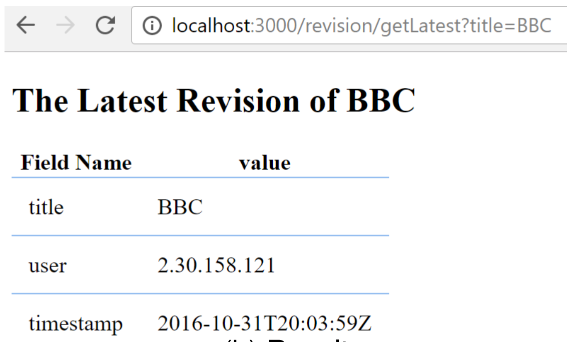
Run the start file as node.js application after you have implemented the route file. Once started, points your browser to `localhost:3000/revision`. It should display a form. When you type in a title say "BBC", the request will be sent and the server will query mongodb to find the latest revision on article BBC and send it back. See Figure 1 for sample output.



localhost:3000/revision

Simple Form

(a) A simple form



localhost:3000/revision/getLatest?title=BBC

The Latest Revision of BBC

Field Name	value
title	BBC
user	2.30.158.121
timestamp	2016-10-31T20:03:59Z

(b) Result

Figure 1: The application output