Assignment 2.3 – API Gateway Part II

Instructions
Complete the following
1. Create a User model and schema with three fields, username, password, and email
   a. models/user.js

   Example
   ```
   /**
   Fields username, password, and email
   */
   var mongoose = require('mongoose');
   var userSchema = new mongoose.Schema({
      username: String,
      password: String,
      email: String
   });
   module.exports = mongoose.model('User', userSchema);
   /**
   Database queries
   */
   ```

2. Add a public key to the config.js file
   a. config.web.secret = 'topsecret'
      i. Hopefully it goes without saying, but in a real production application you would not use a plain text string key. Instead, you would set an environment variable
3. Add a new file to the controller's directory and name it authController.js
4. Add the following lines of code

   ```
   var User = require('../models/user');

   // Register a new user on POST
   exports.user_register = function(req, res) {
      res.send('NOT IMPLEMENTED: User registration POST');
   };

   // Verify token on GET
   exports.user_token = function(req, res) {
      res.send('NOT IMPLEMENTED: User token lookup GET');
   };
   ```

5. Add a new file to the routes directory and name it api-catalog
6. Add the following lines of code

   ```
   /**
   ```

```
* API Routes
*/
var express = require('express');
var router = express.Router();
var auth_controller = require('../controllers/authController');

// POST request for registering a user
router.post('/auth/register', auth_controller.user_register);

// GET request for verifying user tokens
router.get('/auth/token', auth_controller.user_token);

module.exports = router;
```

7. Add a require statement to the app.js file for the api-catalog routes
    a. var apiCatalog = require('./routes/api-catalog');
8. Register the API Catalog's routes
    a. App.use('/api', apiCatalog);
9. Build, run, and test the application
10. Using cURL, create the following requests
    a. curl -X GET localhost:3000/api/auth/register
        i. This request should fail
    b. curl -X POST localhost:3000/api/auth/register
    c. curl -X GET localhost:3000/api/auth/token