Lab 3: Building your own RESTful API

50.012 Networks

Hand-out: September 29
Hand-in: October 6

1 Objectives

- Learn HTTP request/response handling on more abstract objects than sockets
- Familiarize yourself with one of the REST API frameworks
- Decide on a simple data scenario to represent (*which nouns*?)
- Define a couple of simple RESTful API calls (which verb for which noun?)
- Implement and test the API on your own server running in the lab

2 Notes

- You can collaborate with another student, please hand in code individually with both authors noted in the header

3 Set up of your machine

- Connect to $\text{SUTD}_{\text{student}}$ over wireless, test that you have Internet access with `ping` or similar
  - You should also be connected to the wired network to contact other users
- Install flask with `pip install Flask`
- To interact with the API, we suggest either `curl` or Python with `requests` library
  - For a nice `requests` tutorial, see here: `http://www.python-requests.org/en/latest/user/quickstart/`

4 Introduction

- In this assignment, you will implement a simple RESTful API
- You can use a language and library of your choice, we recommend Python+Flask
  - Easy framework suitable for this assignment

- Flask tutorial can be found here: `http://blog.luisrei.com/articles/flaskrest.html`
  - In your API, don't just copy the nouns/verbs from the tutorial, this will lose you points!
  - No need to implement everything in the tutorial. We will point you towards specific content in the following.

5 Basic API

- Decide on a simple data scenario to represent (*which nouns*?)
  - Example: SUTD room information service
    - Which floor is the room on?
    - Is this an office, and who is using it?
    - What is the seating capacity?
    - Any other resources in this room (projector)?
    - Maybe even a reservation system for rooms (we don't have one!)
  - Simple message board with users and messages (or something like twitter?)
  - Maybe an API that aggregates information from other student's APIs? Get room number for user on message board!
- Define a couple of simple RESTful API calls (which verb for which noun?)
  - You should have at least 5 different API calls/ways to interact with your API
    - At least two nouns, and different verbs (GET/PUT/DELETE)
- Send/receive data in JSON datastructures
  - No need to have something super complicated, a dictionary with a couple of keys should be enough for most things
  - JSON tutorial at `http://www.tutorialspoint.com/json/json_python_example.htm`
  - Implement one of the API calls to support two different mimetypes (as discussed in tutorial and lecture)
    - For example, provide a natural text or JSON version of the same data
- Implement and test the API on your own machine running in the lab
  - You won't be able to access other student machines via the `SUTD_Student` network

6 User Authentication

- You should also implement some simple HTTP authentication with a username and a password
  - No need to implement any HTTPS/TLS yet
- Choose one or more useraccounts and passwords, then implement the authentication for at least one API resource as suggested in the tutorial.

7 What to Hand in

7.1 eDimension submission:

- You will submit in the complete API code via eDimension. You can collaborate with a friend on the code, in that case please state both your names in the comments at the start of the file. Both students will then submit the file individually.
- Please add a header at the beginning that comments on the implementation details.
- Please also provide a set of example calls for curl, or a Python script to test your API

7.2 Checkoff:

- Demo your Python code to the Prof/TA and explain it. In particular:
  - What is the general API setting?
  - Which resources did you implement?
  - Showcase some of the calls you made