



REST Assured

REST Test Client

Team 31

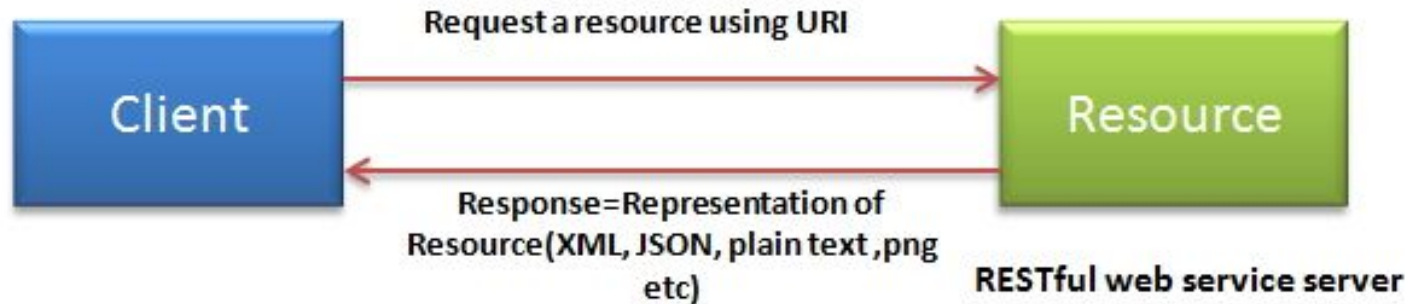
- » Dawson Myers
- » Yang Liu
- » Brandon Roberts

Project Objective

- » Create a test client for testing RESTful services
- » Based on the Sails Live Chrome App
- » Reimplemented as web app
- » Generalize feature set



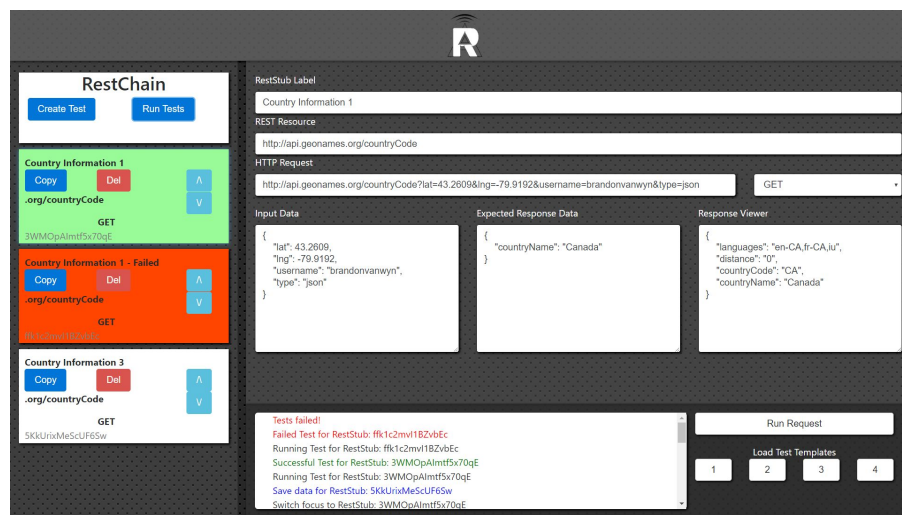
REST



- » Architectural style
- » Lightweight, scalable, maintainable
- » RESTful service
- » Uses HTTP
- » Client-server
- » URIs
- » Stateless

Intuitive Interface

- » User-friendly
- » Master-detail display
- » Test case saving and chaining
- » User-defined parameters for testing
- » Visualized test result
- » Ensure efficiency in workflow

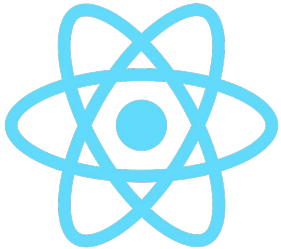




Portability, Scalability, Maintainability

- » Zero install web application
 - » High availability solution
 - » Modular architecture
 - » Fast and easy deployment for new features
 - » Potential features
- 

Technology





Testing

» Jasmin Behaviour-Driven JavaScript Testing Framework



Jasmine Testing - RestStub Example

This is a
representation of a
minimized RestStub.





Jasmine 2.8.0

.....

Ran 2 of 29 specs - run all

2 specs, 0 failures

RestStub

When creating a RestStub

it should be empty.

it should be a copy of data if entered.

Jasmine: Self-Documenting Tests

```
describe("Creating a RestStub", function() {  
  it("Should be empty", function() {  
    let restStub = new RestStub();  
    expect(restStub.identifier).toBeNull();  
  });  
  
  it("Should be a copy of data if entered", function() {  
    let restStub = new RestStub();  
    restStub.identifier = "Dogs";  
    let newRestStub = new RestStub(restStub.toJson());  
    expect(newRestStub.identifier).toEqual(restStub.identifier);  
  });  
});
```




Ran 2 of 29 specs - run all

2 specs, 0 failures


RestStub

When setting the request type
it should accept GET/PUSH/POST/DELETE.
it should not accept non GET/PUSH/POST/DELETE.



Jasmine: Self-Documenting Tests

```
describe("Setting the HTTP request method", function() {  
  it("Should accept GET/PUSH/POST/DELETE", function() {  
    let restStub = new RestStub();  
    expect(restStub.setRequestType("GET")).nothing();  
    expect(restStub.setRequestType("PUT")).nothing();  
    expect(restStub.setRequestType("POST")).nothing();  
    expect(restStub.setRequestType("DELETE")).nothing();  
  });  
  it("Should not accept non GET/PUSH/POST/DELETE.", function() {  
    let restStub = new RestStub();  
    expect(restStub.setRequestType("ERROR")).toThrowError();  
  });  
});
```



Implementation Demo

REST Assured <http://brandonroberts.me/rest/>





Questions?