Daitan GROUP

Orchestrating Automated Tests In Multiple Operating Systems And Browsers

About me

Kelvin Ronny Marques da Silva

- 9 years experience in Software Engineering, and 2 in Electronics
- BSc in Computer Science – IC-Unicamp
- Electronics Technician – Cotuca-Unicamp
- Currently working on Data Engineering, Performance Engineering and Test Automation at Daitan Group
- I love developing and testing!

- kelvsar@gmail.com

CHALLENGE

Problem 1: Ensure the product runs well on all supported operating systems

- Windows 7 x64
- Windows 7 x86
- Windows 8 x64
- Windows 8 x86
- Windows 8.1 x64
- Windows 8.1 x64
- Windows 10 x64
- Windows 10 x86

- Mac OS 10.11
- Mac OS 10.12
- Mac OS 10.13

- 13 Environments

Problem 2: Ensure the product runs well on all supported operating systems and browsers

- Windows 7 x64
- Windows 7 x86
- Windows 8 x64
- Windows 8 x86
- Windows 8.1 x64
- Windows 8.1 x86
- Windows 10 x64
- Windows 10 x86

- Mac OS 10.11
- Mac OS 10.12
- Mac OS 10.13

- Chrome
- Firefox
- Internet Explorer
- Safari

13 * 3 = 39 setups!

Problem 3: Product only runs one instance per machine
Problem 4: Test Suites contain scenarios using more than one instance of the product

Ex.:

- 1 machine is the host and 2 machines are guests of a video conferencing meeting
- 2 or more machines are playing a Game
- 1 machine is the presenter and 2 are the participants on a webinar

Problem 5: Some complex test suites cannot run in parallel in different clients

- Parallel Plans:
  Installation Tests, Memory Tests, and Test Cases using different sessions.

- Sequentially Plans:
  Account Limitations, Database restriction, and Sessions Tested individually

Problem 6: Orchestrate

- In order to solve all those problems, we need a mechanism to automate and orchestrate everything

- Due to the high quantity of environment combinations, the complexity of organizing and executing tests increases and can lead to an ineffective pipeline.

Challenge

- How to coordinate and schedule tests in an easy way?
- How to continue delivering the tests in an agile way, despite those problems?
- How creating and working with a solution for those problems could be simple?

Daitan GROUP

Constructing the Solution

Jmeter

- The most famous free tool for performance testing against API
- It could be used for regression testing against API as well
- We could create test plan, steps from a test and Threads to collect the result of the API Calls
- We can also integrate Jmeter with external libs, since we can use Java or Javascript inside JMeter
- **It is possible to send requests to more than one machine or server**

Instead of using JUnit or TestNG…

... We just need to create a server with the libraries used for test/task automation (Selenium, Sikuli, White, etc)

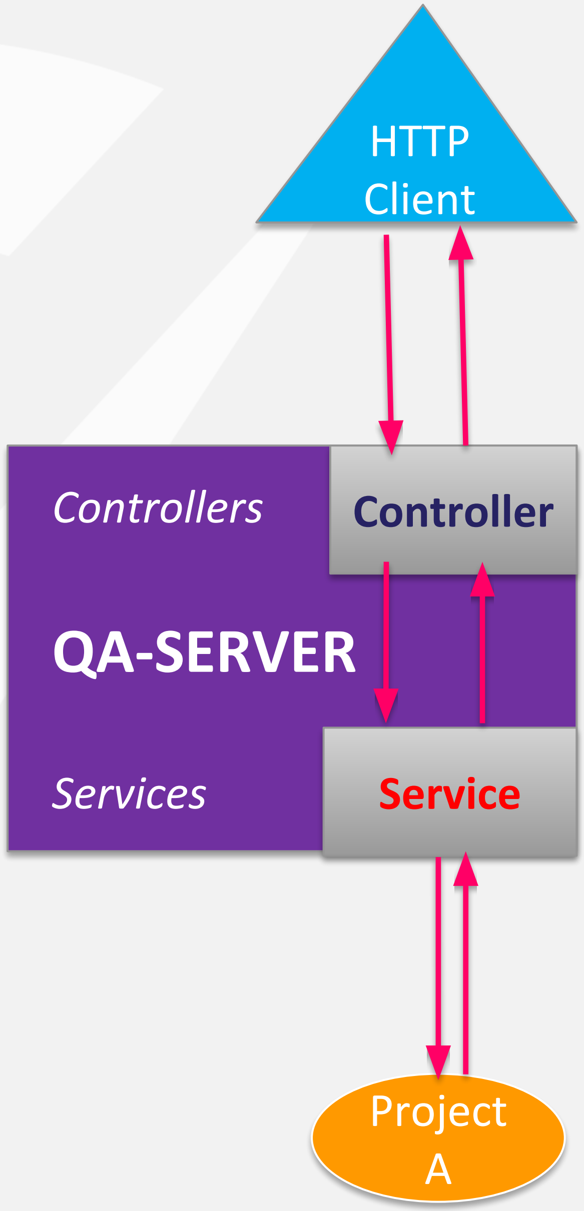
It does not require a specific language (we use Java for this server, but a server built with the libraries for automation could be created using Python, C#, etc)

We created our QA-Server!

- Java + Maven
- Spring Framework (dependency injection)
- Apache CXF (our base for the server, btw)
- External Libraries

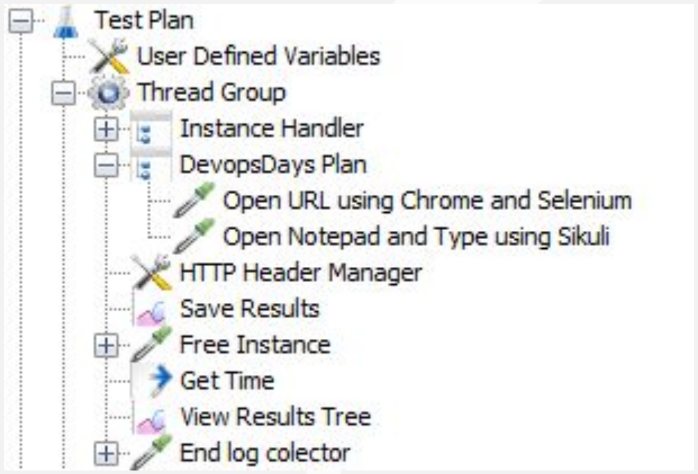
How QA-Server works

0. HTTP Client send a HTTP request to QA-Server;
1. Controller receives the request HTTP to execute some action or to check something;
2. Controller send the data to the proper Service, based on the request received;
3. Service chooses the proper Project to initiate some action or verification;
4. Project does the action or verification and returns the result to the Service;
5. Service returns the Result to the Controller;
6. Controller returns the Result to HTTP Client



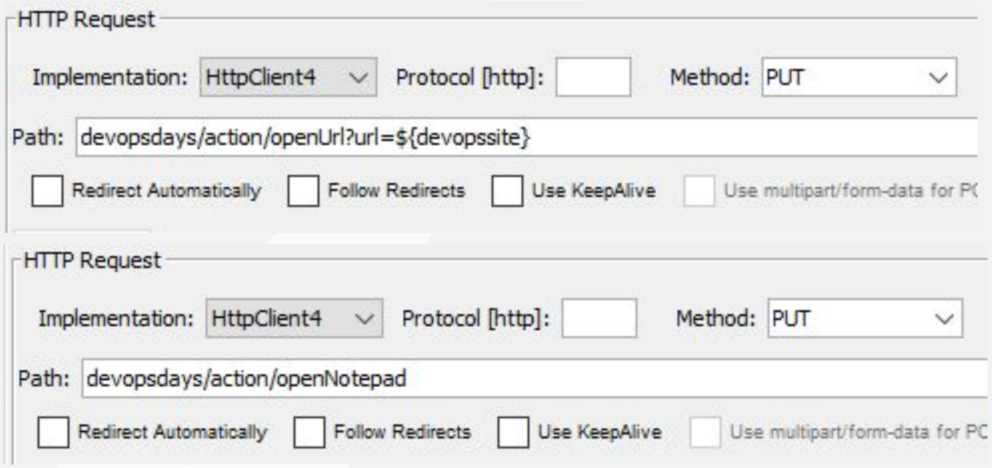
Example: JMeter Side

Jmeter Plan



- Test Plan
  - User Defined Variables
  - Thread Group
    - Instance Handler
    - DevopsDays Plan
      - Open URL using Chrome and Selenium
      - Open Notepad and Type using Sikuli
    - HTTP Header Manager
    - Save Results
    - Free Instance
    - Get Time
    - View Results Tree
    - End log colector

HTTP Requests



HTTP Request

Implementation: HttpClient4 Protocol [http]: Method: PUT

Path: devopsdays/action/openUrl?url=${devopssite}

☐ Redirect Automatically ☐ Follow Redirects ☐ Use KeepAlive ☐ Use multipart/form-data for PC

HTTP Request

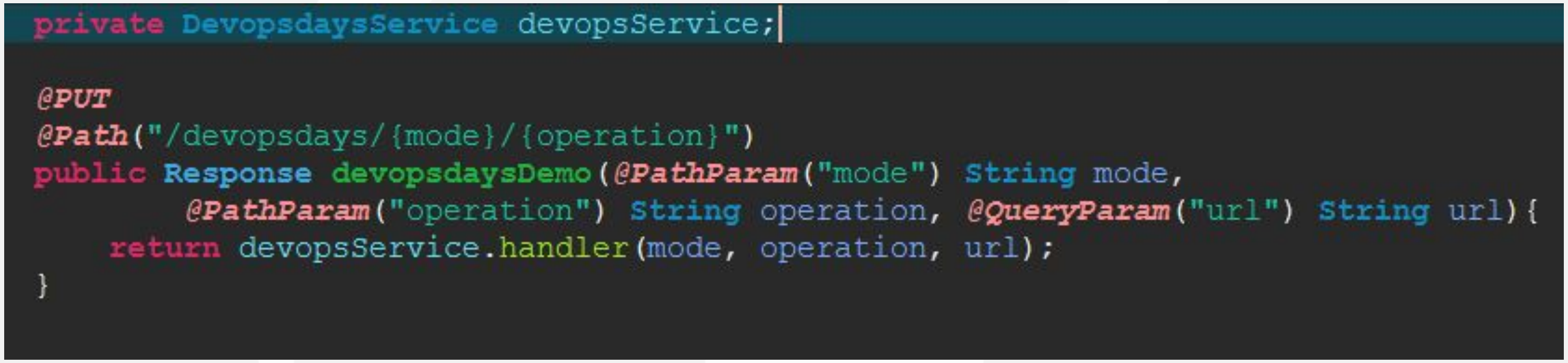
Implementation: HttpClient4 Protocol [http]: Method: PUT

Path: devopsdays/action/openNotepad

☐ Redirect Automatically ☐ Follow Redirects ☐ Use KeepAlive ☐ Use multipart/form-data for PC

Example: QA-Server

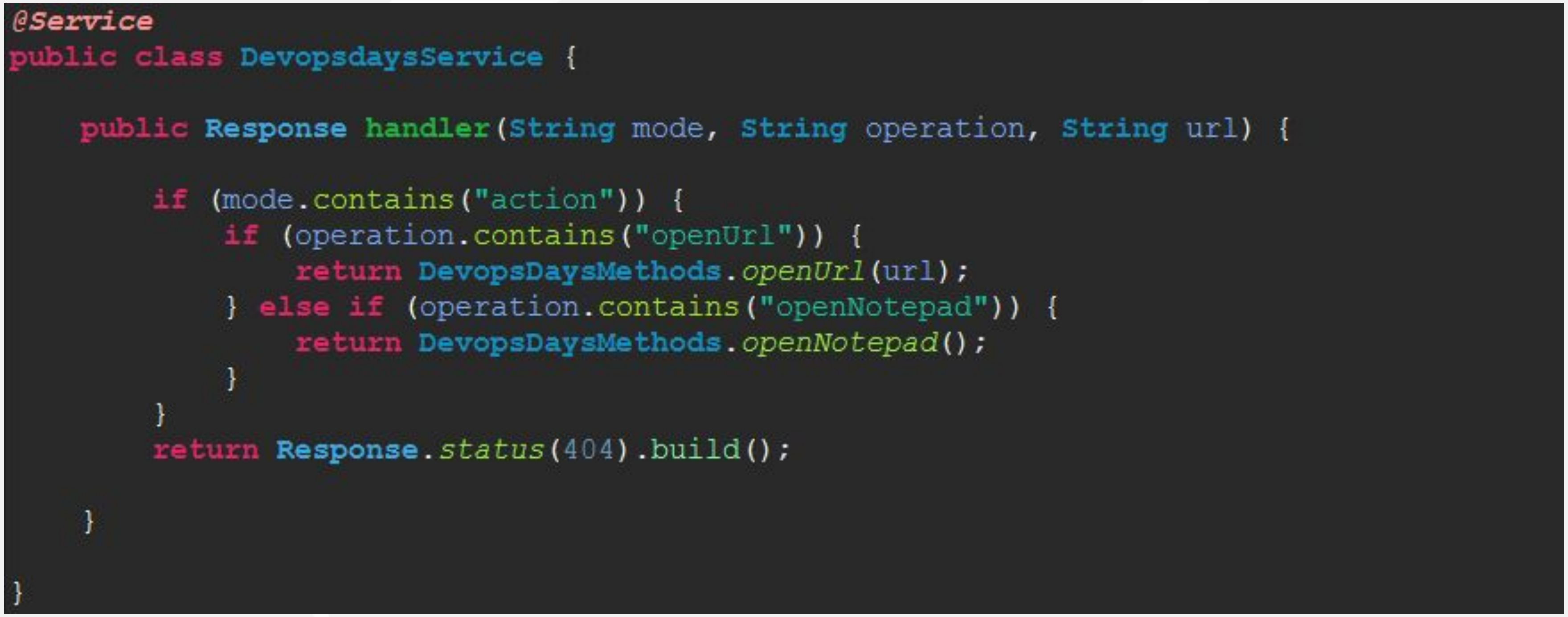
Controller



```
private DevopsdaysService devopsService;

@PUT
@Path("/devopsdays/{mode}/{operation}")
public Response devopsdaysDemo(@PathParam("mode") String mode,
        @PathParam("operation") String operation, @QueryParam("url") String url){
    return devopsService.handler(mode, operation, url);
}
```

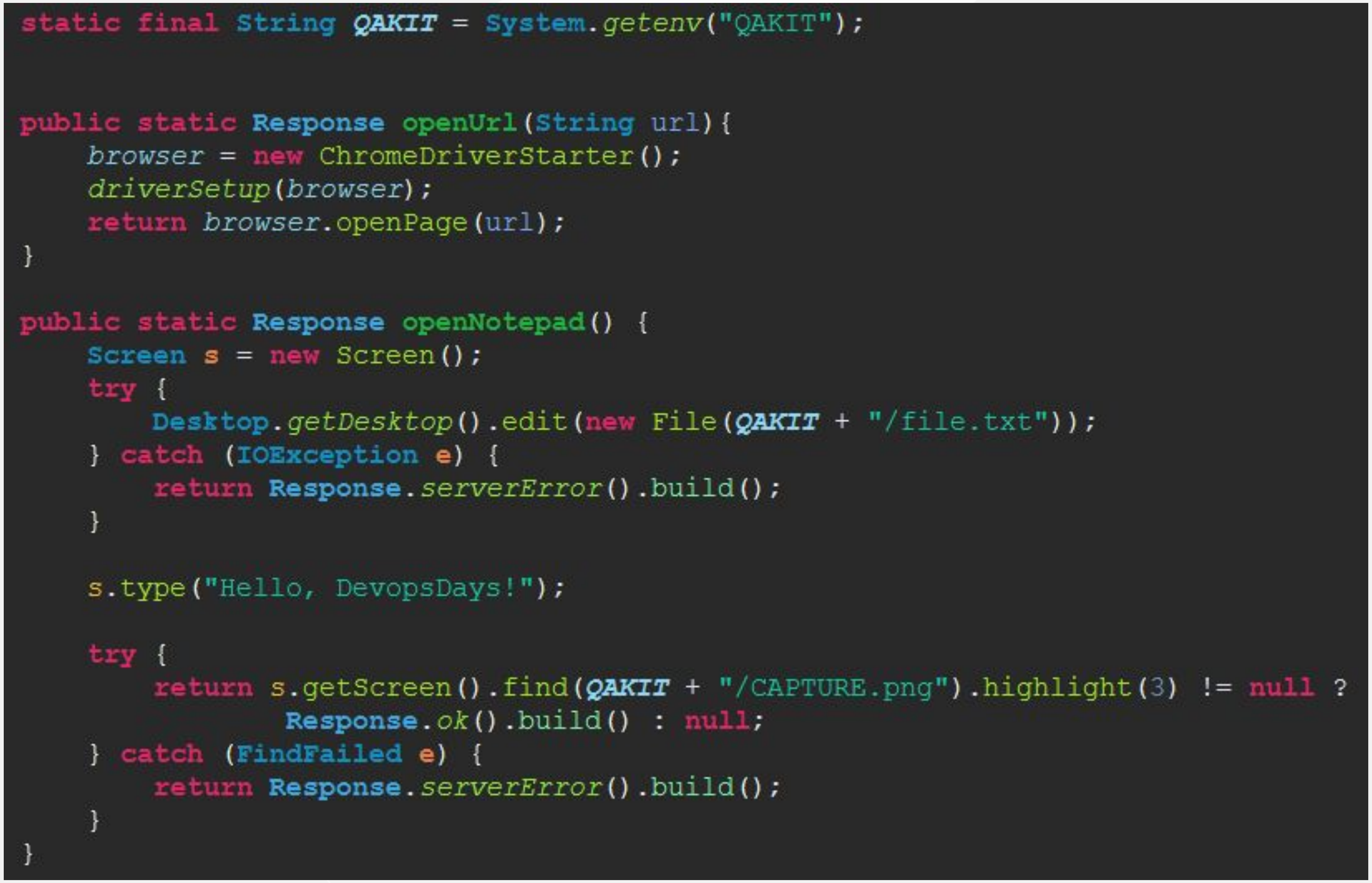
Example: QA-Server

Service



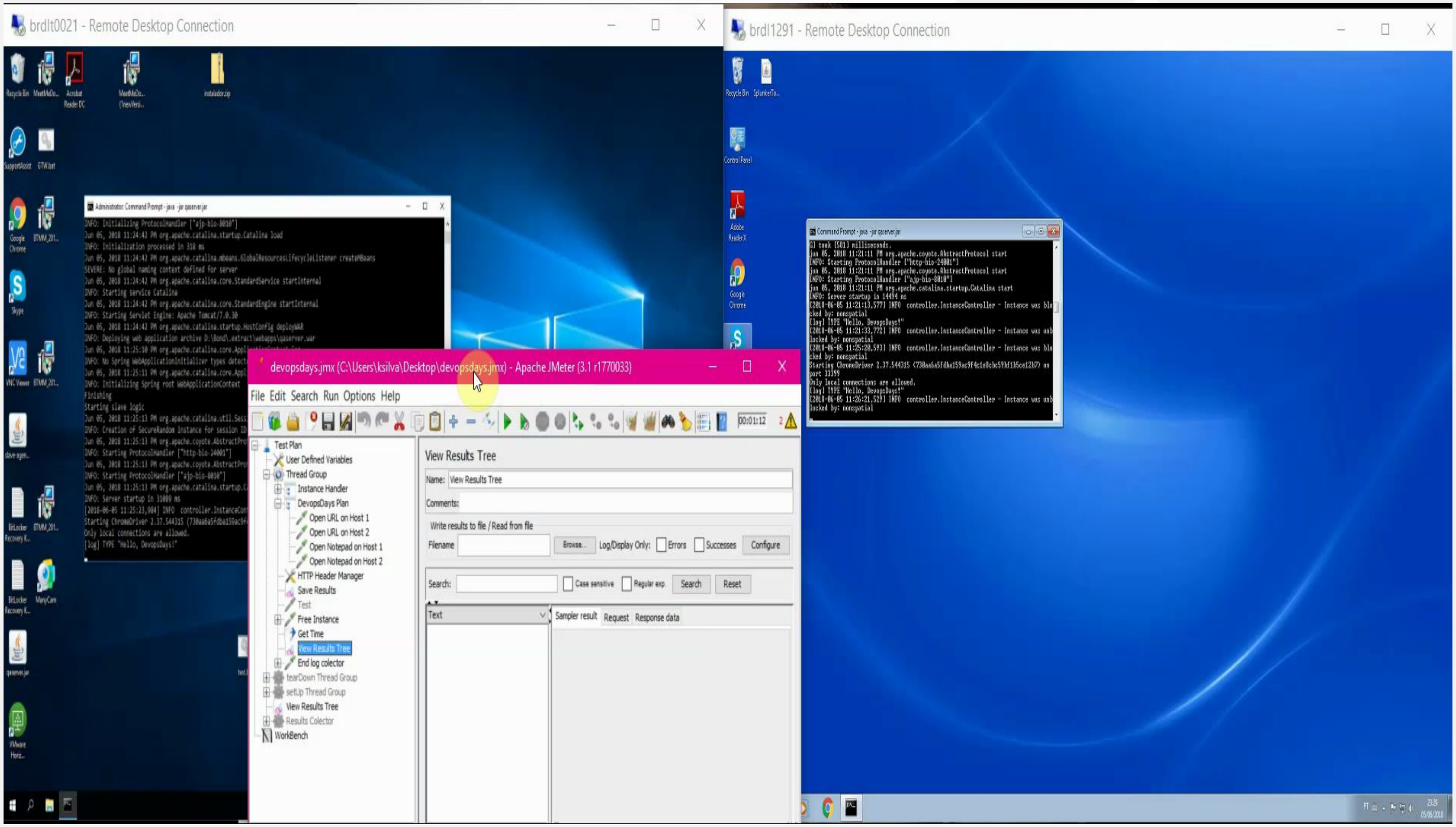
```
@Service
public class DevopsdaysService {

    public Response handler(String mode, String operation, String url) {

        if (mode.contains("action")) {
            if (operation.contains("openUrl")) {
                return DevopsDaysMethods.openUrl(url);
            } else if (operation.contains("openNotepad")) {
                return DevopsDayMethods.openNotepad();
            }
        }
        return Response.status(404).build();

    }

}
```

Example: Devopsdays Project



```java
static final String QAKIT = System.getenv("QAKIT");


public static Response openUrl(String url){
    browser = new ChromeDriverStarter();
    driverSetup(browser);
    return browser.openPage(url);
}

public static Response openNotepad() {
    Screen s = new Screen();
    try {
        Desktop.getDesktop().edit(new File(QAKIT + "/file.txt"));
    } catch (IOException e) {
        return Response.serverError().build();
    }

    s.type("Hello, DevopsDays!");

    try {
        return s.getScreen().find(QAKIT + "/CAPTURE.png").highlight(3) != null ?
                Response.ok().build() : null;
    } catch (FindFailed e) {
        return Response.serverError().build();
    }
}
```

QA-Server + Jmeter: Demo



```
INFO: Initializing ProtocolHandler ["ajp-bio-8010"]
Jun 05, 2018 11:24:42 PM org.apache.catalina.startup.Catalina load
INFO: Initialization processed in 318 ms
Jun 05, 2018 11:24:42 PM org.apache.catalina.mbeans.GlobalResourcesLifecycleListener createMBeans
SEVERE: No global naming context defined for server
Jun 05, 2018 11:24:42 PM org.apache.catalina.core.StandardService startInternal
INFO: Starting service Catalina
Jun 05, 2018 11:24:42 PM org.apache.catalina.core.StandardEngine startInternal
INFO: Starting Servlet Engine: Apache Tomcat/7.0.30
Jun 05, 2018 11:24:42 PM org.apache.catalina.startup.HostConfig deployWAR
INFO: Deploying web application archive D:\Bond\.extract\webapps\qaserver.war
Jun 05, 2018 11:25:10 PM org.apache.catalina.core.Appl
INFO: No Spring WebApplicationInitializer types detect
Jun 05, 2018 11:25:11 PM org.apache.catalina.core.Appl
INFO: Initializing Spring root WebApplicationContext
Finishing
Starting slave logic
Jun 05, 2018 11:25:13 PM org.apache.catalina.util.Sess
INFO: Creation of SecureRandom instance for session ID
Jun 05, 2018 11:25:13 PM org.apache.coyote.AbstractPro
INFO: Starting ProtocolHandler ["http-bio-24001"]
Jun 05, 2018 11:25:13 PM org.apache.coyote.AbstractPro
INFO: Starting ProtocolHandler ["ajp-bio-8010"]
Jun 05, 2018 11:25:13 PM org.apache.catalina.startup.C
INFO: Server startup in 31089 ms
[2018-06-05 11:25:23,984] INFO  controller.InstanceCon
Starting ChromeDriver 2.37.544315 (730aa6a5fdba159ac9f
Only local connections are allowed.
[log] TYPE "Hello, DevopsDays!"
```

```
G] took [501] milliseconds.
jun 05, 2018 11:21:11 PM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-bio-24001"]
jun 05, 2018 11:21:11 PM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["ajp-bio-8010"]
jun 05, 2018 11:21:11 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 14494 ms
[2018-06-05 11:21:13,577] INFO  controller.InstanceController - Instance was blo
cked by: nonspatial
[log] TYPE "Hello, DevopsDays!"
[2018-06-05 11:21:33,772] INFO  controller.InstanceController - Instance was unb
locked by: nonspatial
[2018-06-05 11:25:20,593] INFO  controller.InstanceController - Instance was blo
cked by: nonspatial
Starting ChromeDriver 2.37.544315 (730aa6a5fdba159ac9f4c1e8cbc59bf1b5ce12b7) on
port 33399
Only local connections are allowed.
[log] TYPE "Hello, DevopsDays!"
[2018-06-05 11:26:21,529] INFO  controller.InstanceController - Instance was unb
locked by: nonspatial
```

Partial Solution

Problem 1: Support all OS.
Problem 2: Support all browsers.

QA-Server on each OS, pointing to the project, libraries and frameworks.

Problem 3: Several Instances of the Client
Problem 4: One Instance per Machine

JMeter coordinates what QA-Server will do on each machine being used by the test plan.

Remaining Problems:

Problem 5: Support Sequential Plans
Problem 6: Orchestrate to create an effective pipeline

Solving the 2 problems left

- We have created an automated orchestration of our complex test suites that delivers the effectiveness needed in Devops culture

- Now, we can coordinate and schedule tests in an easy way, implement changes in parameters, and cover the maximum number of machines

QA-Planner

We have developed a tool called QA-Planner.

- FrontEnd: AngularJS + Javascript
- BackEnd: QA-Server / Java
- DB: MongoDB

What is QA-Planner?

The idea came from the following concepts:

- Coordinate and schedule tests in an easy way, in different machines, respecting the dependency of each test set (Sequential Plans, for instance)
- Verify if the automation tasks are being done properly on each machine
- Return an automated effective pipeline of the Test Suite

QA-Planner: Scheduler Algorithm

The Algorithm needs to deal with these restrictions:

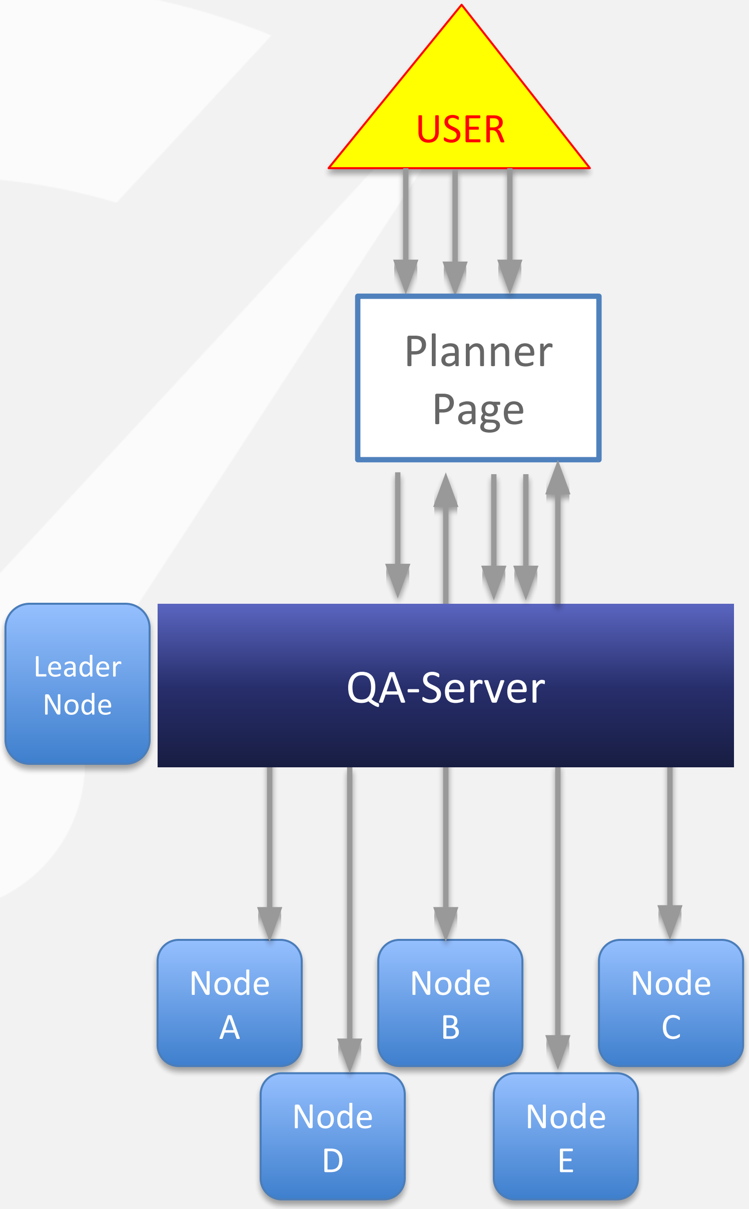
- Sequential Test Plans must not run in parallel
- We need to be able to choose an order for each Test Plan
- It must be accessible/runnable by external tools, such as Jenkins
- Everyone on the Devops team must be able to create or access the Tool in realtime

QA-Planner: JMeter Plans Orchestrator

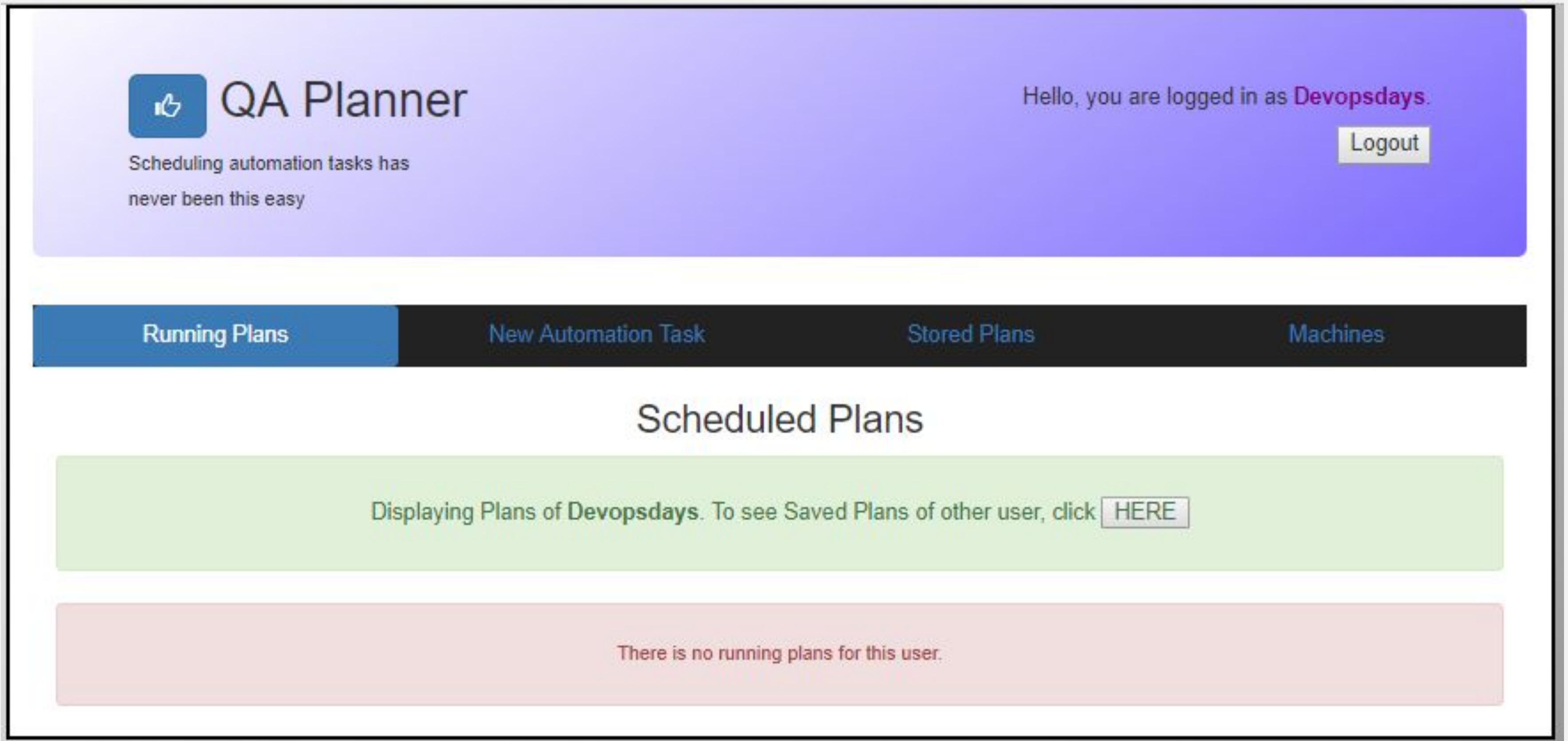
- QA-Planner controls which plans will run and when
- Once that pipeline starts to run, the orchestrator will be processing all plans and it will keep monitoring JMeter processes
- It also changes the variables inserted into JMeter on execution time

How QA-Planner works

0. User selects the plans and machines
1. User requests a pipeline
2. BackEnd uses the Scheduler Algorithm to create the pipeline
3. User receives a pipeline candidate
4.
   - A) User accepts the pipeline and the tests begin.
   - B) User accepts the pipeline and take the id to put on other tool
   - C) User rejects the pipeline. User can change the plans or machines selected before to create a new pipeline.



QA-Planner: Page

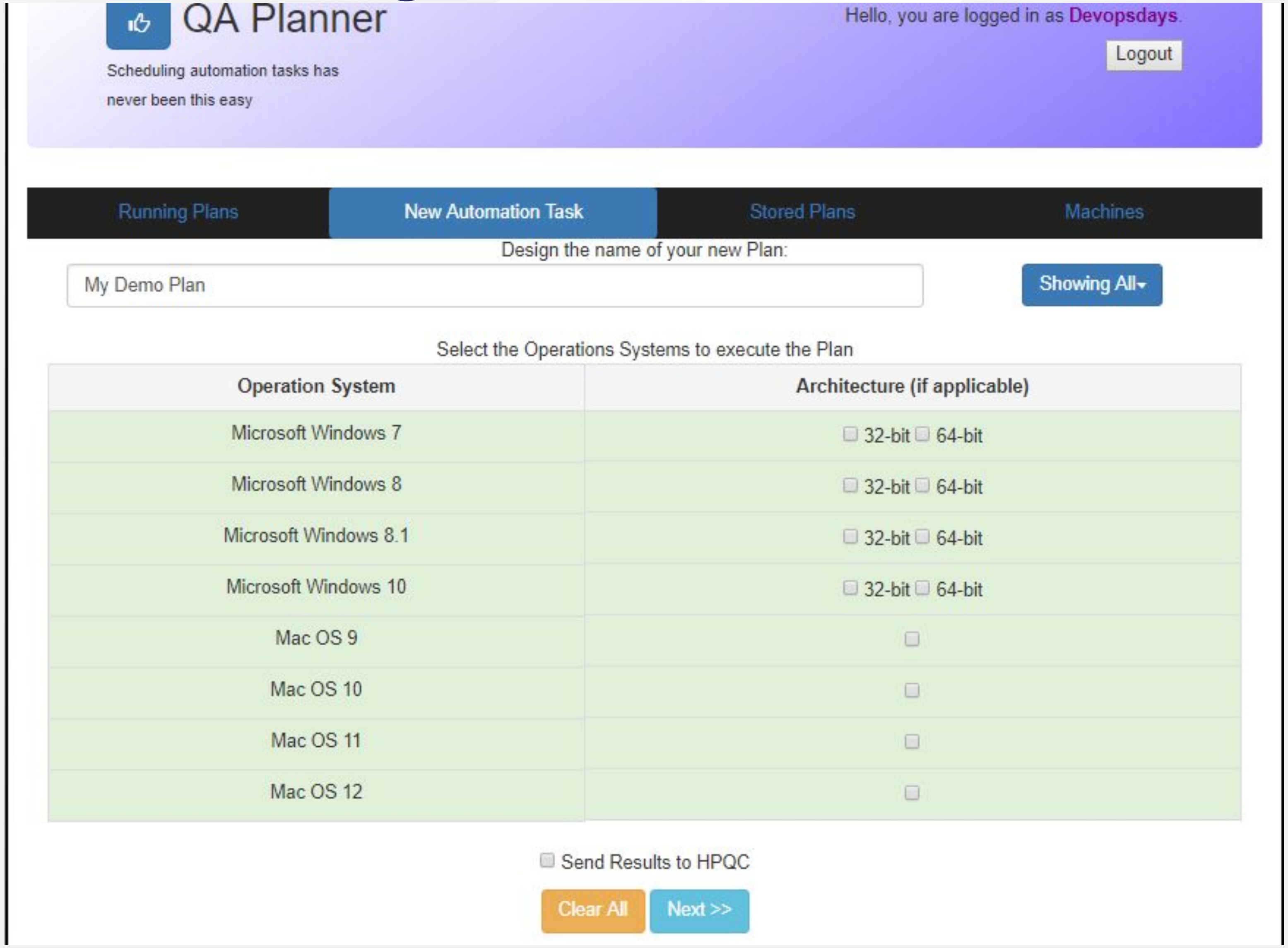

QA Planner
Scheduling automation tasks has never been this easy
Hello, you are logged in as Devopsdays.
Logout
Running Plans
New Automation Task
Stored Plans
Machines

Scheduled Plans

Displaying Plans of **Devopsdays**. To see Saved Plans of other user, click HERE

There is no running plans for this user.

QA-Planner: Page



Scheduling automation tasks has never been this easy

Design the name of your new Plan:

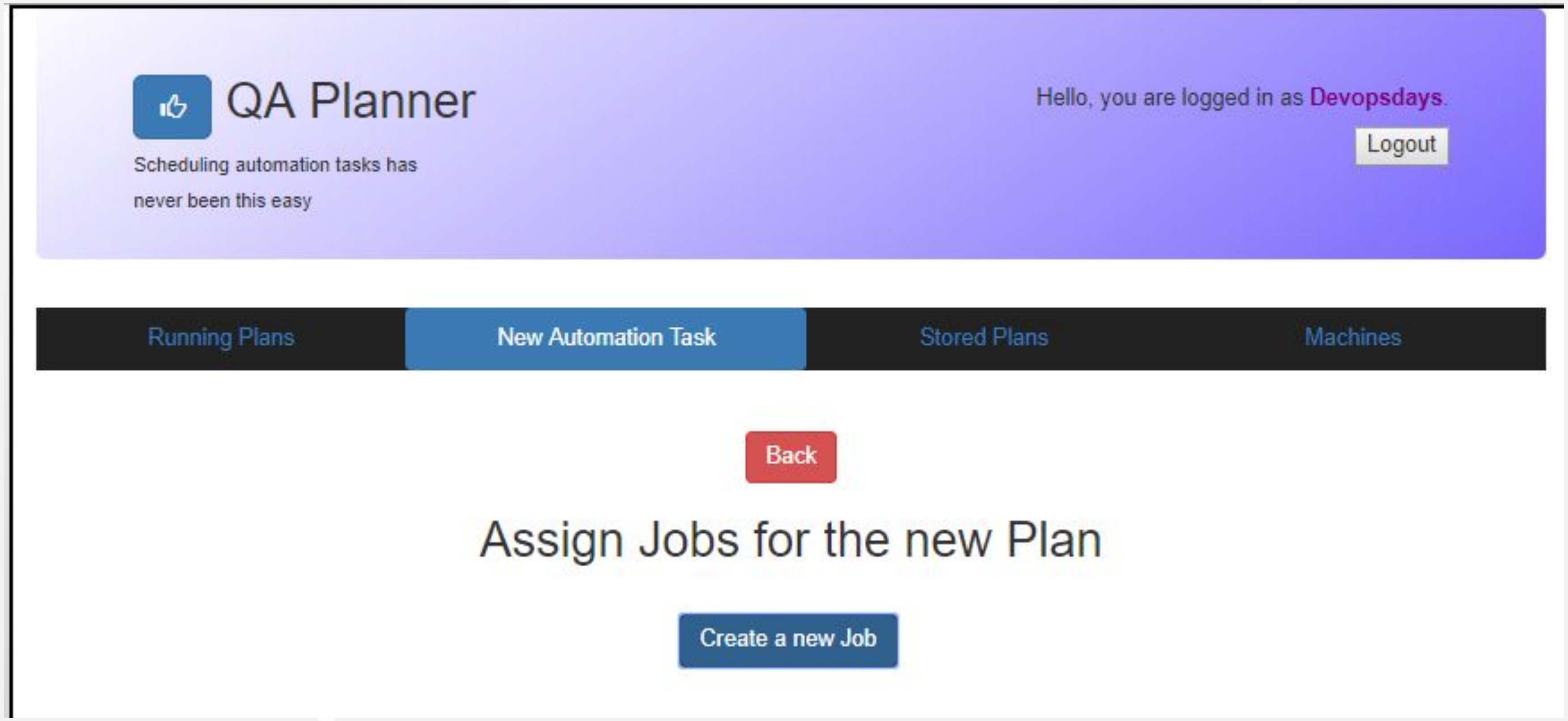
My Demo Plan

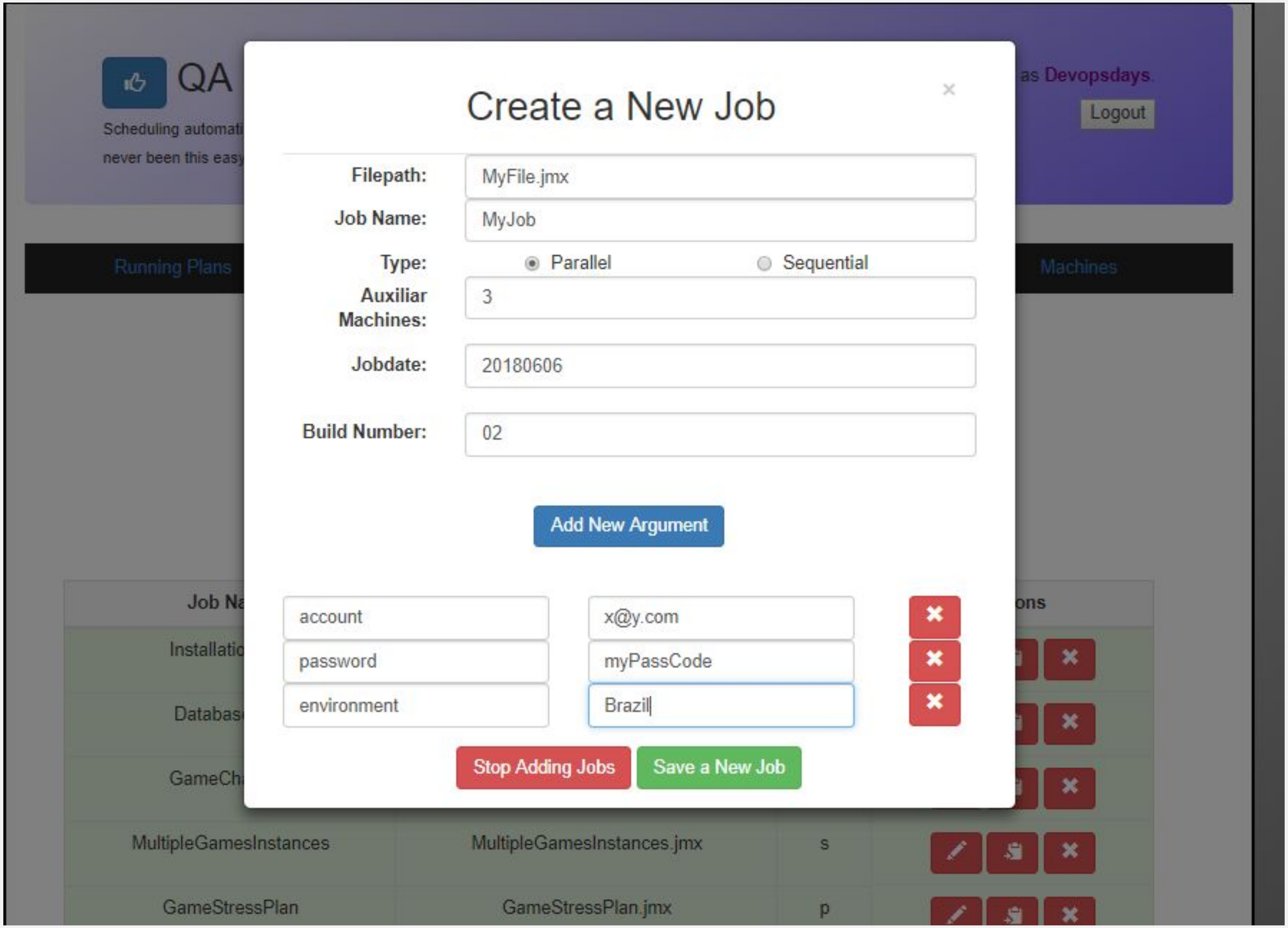
Select the Operations Systems to execute the Plan

Operation System	Architecture (if applicable)
Microsoft Windows 7	☐ 32-bit ☐ 64-bit
Microsoft Windows 8	☐ 32-bit ☐ 64-bit
Microsoft Windows 8.1	☐ 32-bit ☐ 64-bit
Microsoft Windows 10	☐ 32-bit ☐ 64-bit
Mac OS 9	☐
Mac OS 10	☐
Mac OS 11	☐
Mac OS 12	☐

☐ Send Results to HPQC

QA-Planner: Page


QA Planner
Scheduling automation tasks has never been this easy
Hello, you are logged in as Devopsdays.
Logout
Running Plans
New Automation Task
Stored Plans
Machines
Back
Assign Jobs for the new Plan
Create a new Job



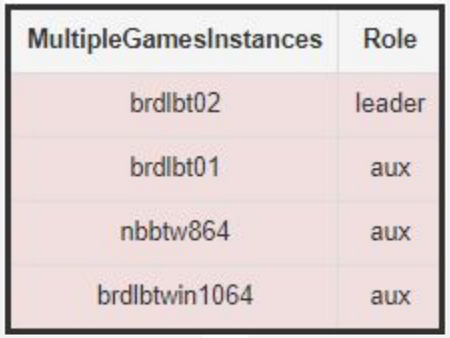
Create a New Job

Filepath:	MyFile.jmx
Job Name:	MyJob
Type:	[✓ Parallel] Sequential
Auxiliar Machines:	3
Jobdate:	20180606
Build Number:	02

account	x@y.com
password	myPassCode
environment	Brazil

MultipleGamesInstances	MultipleGamesInstances.jmx	s
GameStressPlan	GameStressPlan.jmx	p

QA-Planner: Roles



MultipleGamesInstances	Role
brdlbt02	leader
brdlbt01	aux
nbbtw864	aux
brdlbtwin1064	aux

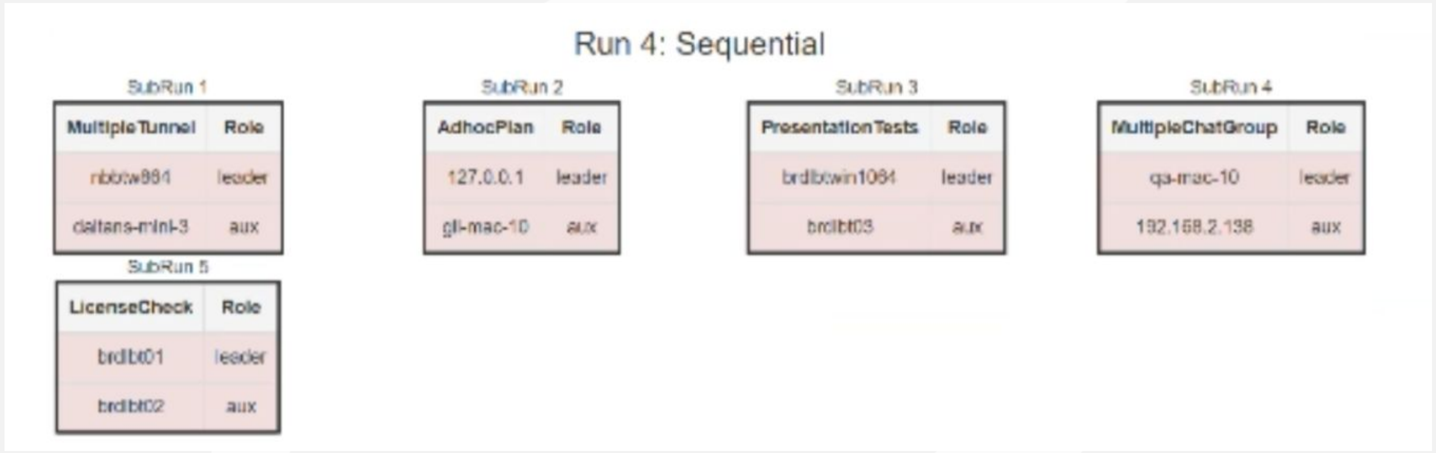
Leader: Machine under testing

Aux: One or more machines used to make all scenarios "testable" by automated tests

Ex.: Leader is the Game running on Windows 8, others 3 machines (don't care about OS) will join the Game as well, but the focus here is testing all features on Windows 8

QA-Planner: Demo – Full Test Suite Pipeline

QA-Planner: Demo



Run 4: Sequential

SubRun 1

MultipleTunnel	Role
nbbtw884	leader
daitans-mini-3	aux

SubRun 2

AdhocPlan	Role
127.0.0.1	leader
gl-mac-10	aux

SubRun 3

PresentationTests	Role
brdlbtwin1064	leader
brdlbt03	aux

SubRun 4

MultipleChatGroup	Role
qa-mac-10	leader
192.168.2.138	aux

SubRun 5

LicenseCheck	Role
brdlbt01	leader
brdlbt02	aux

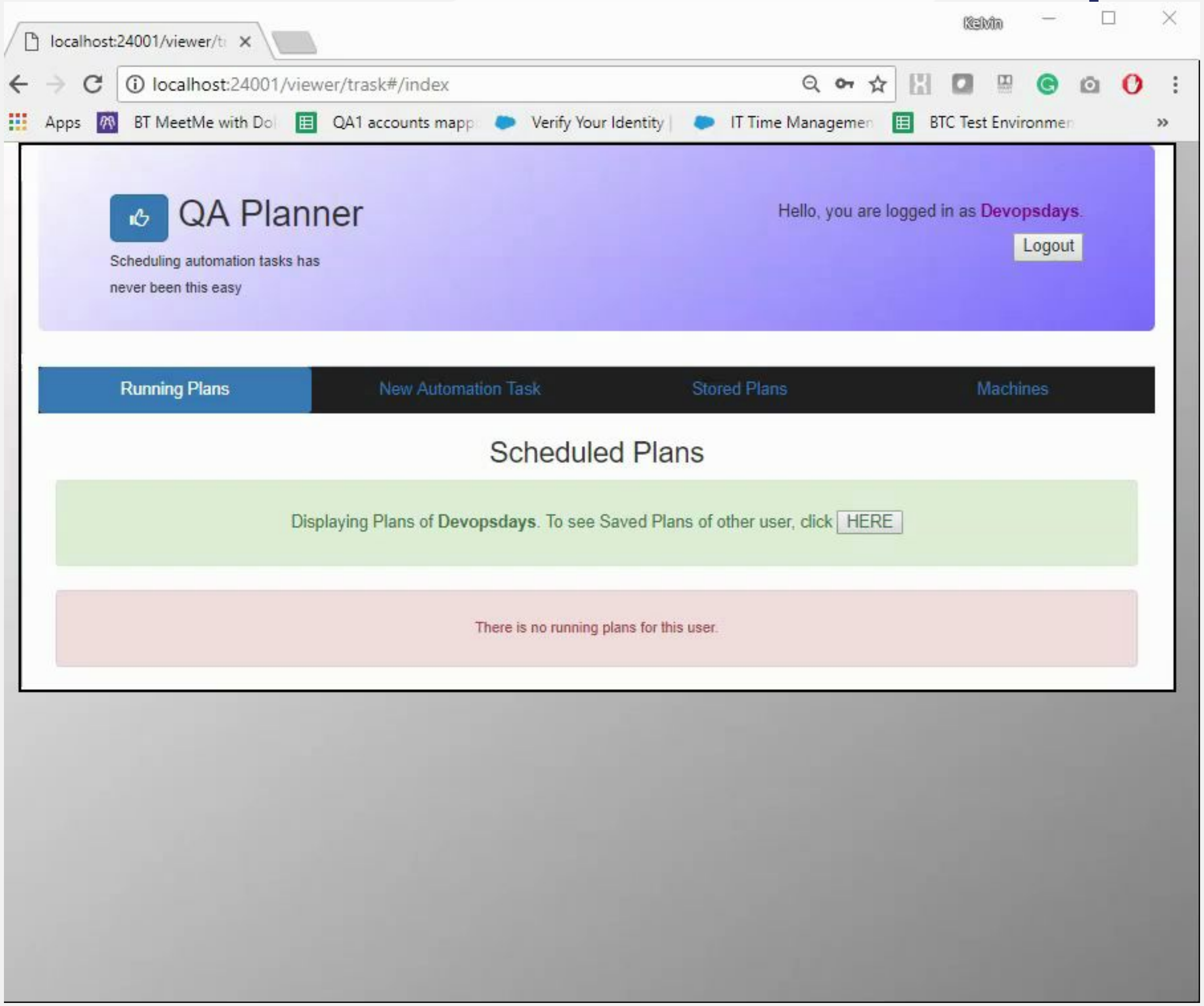
If each test plan takes 10 minutes to be done.

5 scenarios * 13 machines =

Traditional pipeline: 650 minutes!

QA-Planner: 650/5 = 130 minutes!

QA-Planner: Demo – Demo Test Suite Pipeline



Scheduling automation tasks has never been this easy

Hello, you are logged in as Devopsdays.

Scheduled Plans

Displaying Plans of **Devopsdays**. To see Saved Plans of other user, click HERE

There is no running plans for this user.

Solved!

Problem 1: Support all OS.
Problem 2: Support all browsers.
QA-Server on each OS, pointing to the project, libraries and frameworks.

Problem 3: Several Instances of the Client
Problem 4: One Instance per Machine
JMeter coordinates what QA-Server will do on each machine being used by the test plan.

Problem 5: Support Sequential Plans
Problem 6: Orchestrate to create an effective pipeline
QA-Planner coordinates and orchestrates everything!

Conclusion

- This solution saved a lot of time
- It’s really easy to create new tests and reuse the steps by using HTTP calls on JMeter
- It’s not just solving problems, it is a new pattern followed here

Daitan GROUP

The quality software development partner who ***significantly accelerates*** *your time-to-market.*

Thank You!

kelvsar@gmail.com