Hi,

In this article I am going to explain about Reportprotal.io and why you should migrate your reporting mechanism to it.

I head about Reportportal.io couple of months ago while searching ‘best reporting tools for selenium’. Everyone is talking about Exten, Klov, Allure, etc., But there is one small podcast that got my attention.

The moment I saw reportportal.io I was amazed with its features like Machine Learning, Pattern Analysis, etc., and I can’t stop myself from tasting it. And Now I am convinced that Reportportal is the Best reporting mechanism.

Benefits of Reportportal.io and Why you should migrate your reporting mechanism to it:

* Machine Learning Capability
* Live Reporting Mechanism
* Centralized reporting for your different test automation frameworks [selenium, Python, SOAP UI Etc.]
* Auto Analysis of issues based on log history
* Make automation results analysis actionable & collaborative
* Integration with defect management tools like Jira and Rally
* Build custom dashboards and metrics to learn from the past and predict trends
* Multi Language/Framework support

Setup:

There are two ways to setup reportportal.io

1. The Hard way: Install bunch of software and integrate them
2. The Easy way: build using docker-compose.

I am going to build using docker-compose as it is **very easy** now a days to build anything with dockers. And A Big thanks to **Karthik KK** and his ‘ [ABC of docker](https://www.youtube.com/watch?v=90Y6YhmwFVs&list=PL6tu16kXT9PqdhOZk4MNVtQDJp6xFrotg) ‘ YouTube series helped me learning dockers an year ago. Since then I became a fan of Karthik, Docker and its capabilities.

System Requirements and Involved Softwares:

The official integration steps from reportportal.io are based on deploying in Linux environment, But in my project I am using it in windows.

**Operating System:** Windows 10

**Docker version:** Any recent stable version [ I am using community edition]

**Docker-compose:** default compose version as my OS is windows

**RAM:** At least 5 GB of Dedicated RAM to Dockers

**ReportPortal version**: v4.3.0

**Database:** MongoDB

**SearchEngine**: ElasticSearch

**Gateway:** Fabio/use tarefik if deploying in Linux

**Defect Management Tool:** Jira/Rally

**PORT** : 8080 [if any of your service is already pointing to 8080 I suggest moving it to another port]

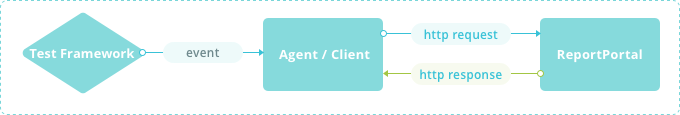
Volumes in use:

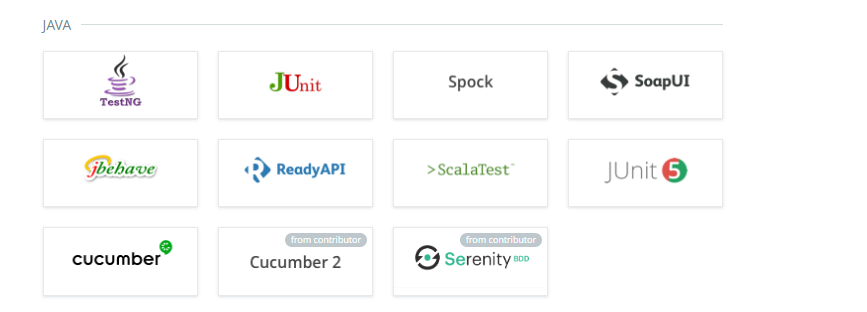
1. mongo
2. Consul
3. ElasticSearch

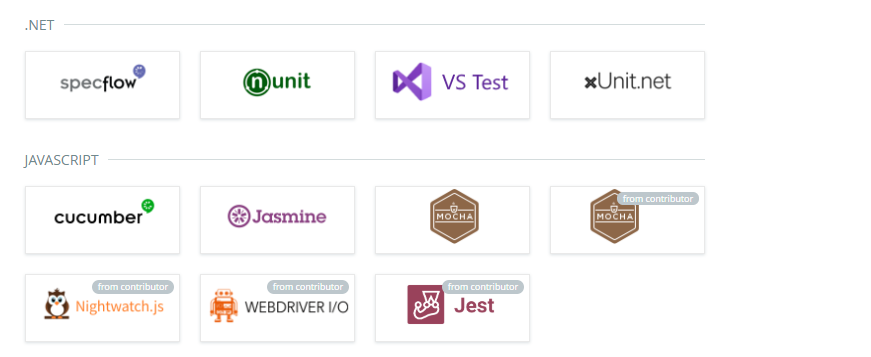
Docker Commands in use:

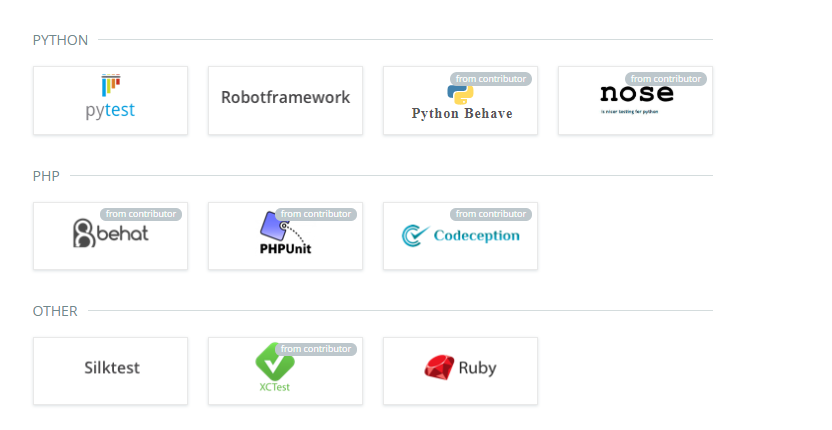
* Docker-compose up -d
* Docker ps
* Docker volume ls
* Docker container\_name logs
* Docker-compose down

Supported Frameworks Integration:





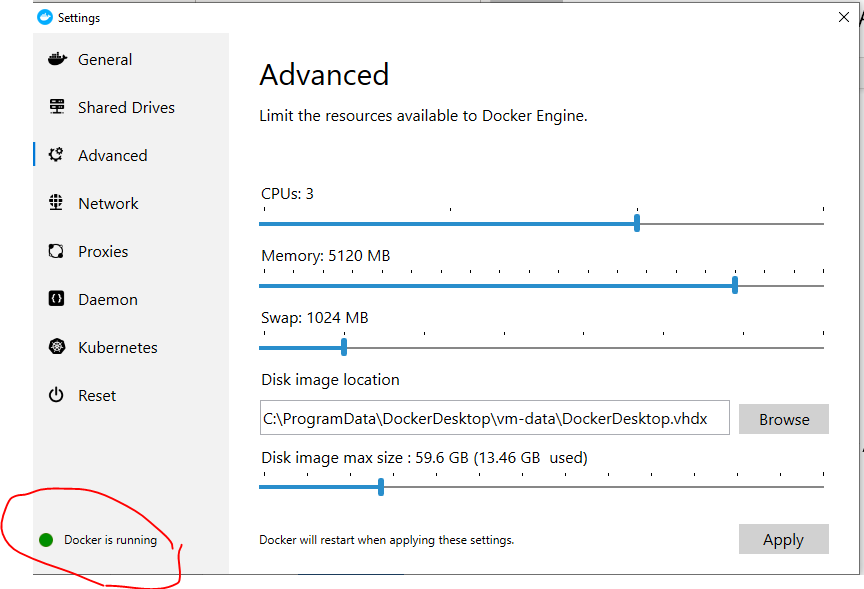




If your Tool/Framework is not listed above, you can create your own integration using its REST APIs.

**Installation Steps:**

1. [Download Docker](https://www.docker.com/products/docker-desktop) from official site and install.
2. Once install is complete make sure your Docker is UP and Running and Set RAM 5 GB.



Now Copy paste my below optimized YAML file in your local drive and go to that path in powershell



Execute command **'docker-compose -f reportportal.yml -p reportportal up -d --force-recreate --build'**

Where:

'-p reportportal' -- specifies container's prefix (project name)

'-d' -- enables daemon mode

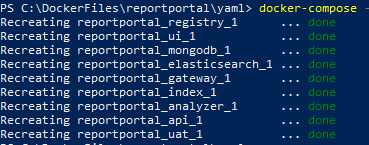
'--force-recreate' -- forces re-recreating of all containers

‘-f’ – specify compose file name

‘—build’ –build images properly before creating containers



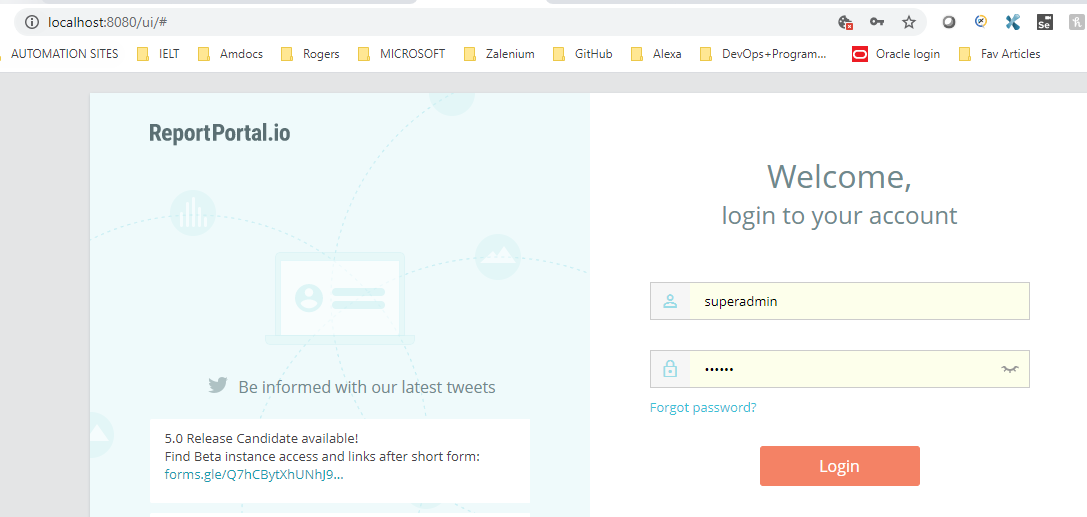
Once you execute above command you will see something like this.



Now execute command ‘docker ps’ and make sure all containers are stable.



Now go to http://localhost:8080

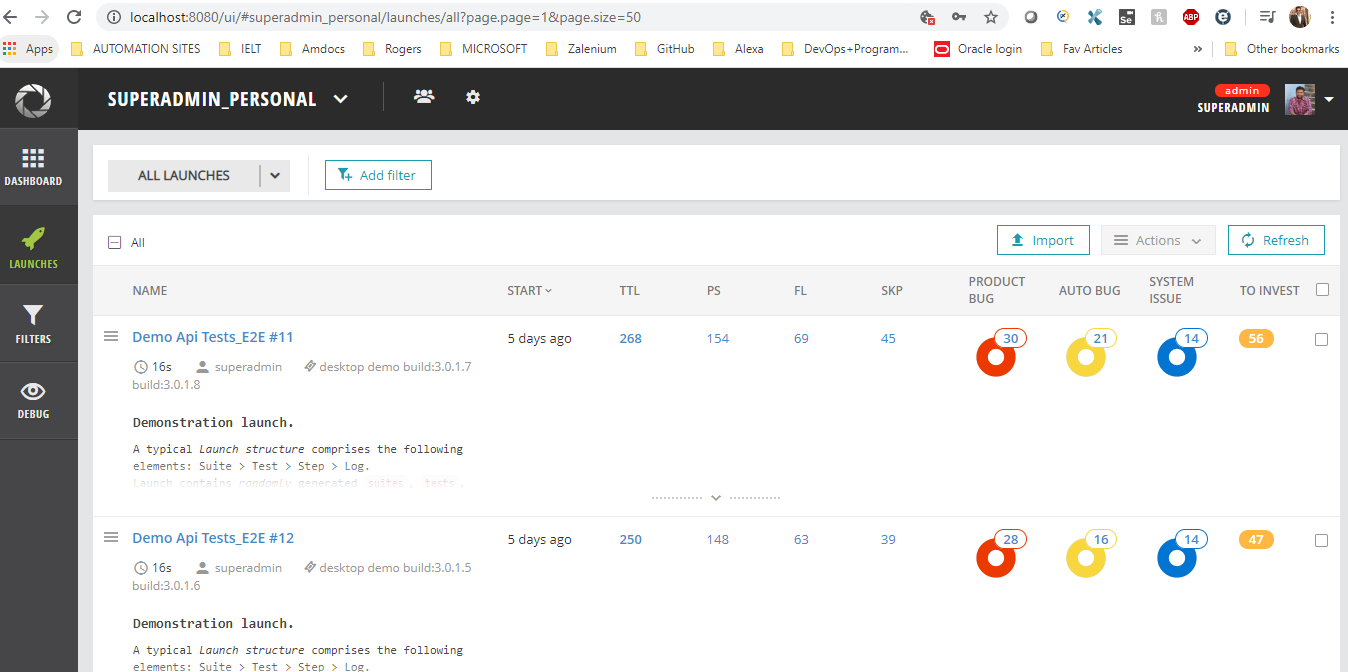


Username: superadmin

Password: erebus

For security reasons I suggest changing the admin password after login.

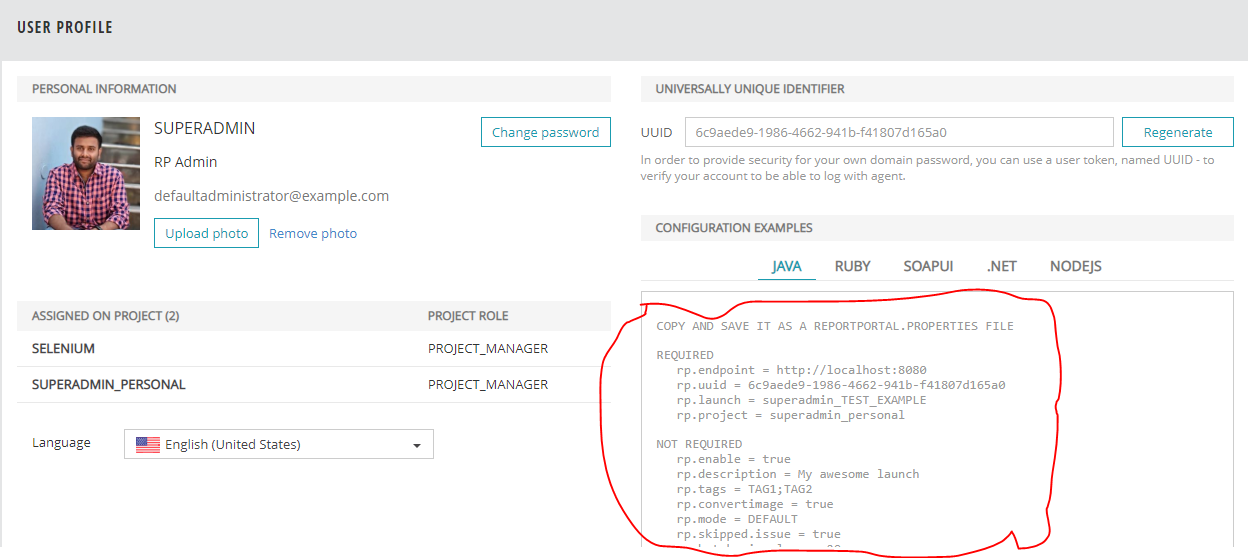
Now explore the portal with all available options there.



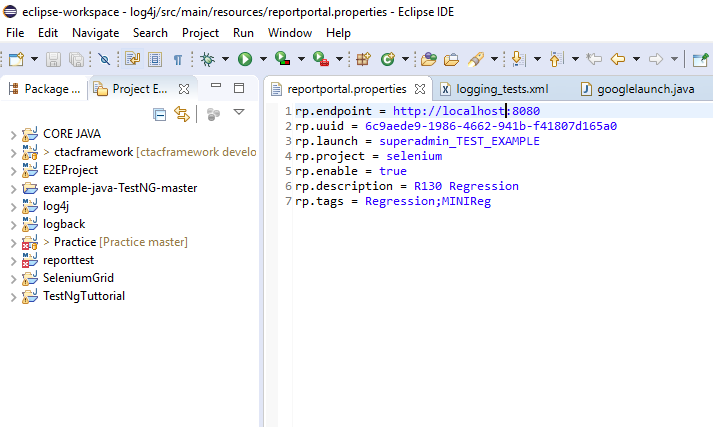
How to Integrate your test automation framework:

I am using a Selenium TestNG framework to test this. Source Code Github [link](https://github.com/reportportal/agent-java-testNG)

Follow the steps mentioned in above and change the reportportal.properties file from portal menu -> profile

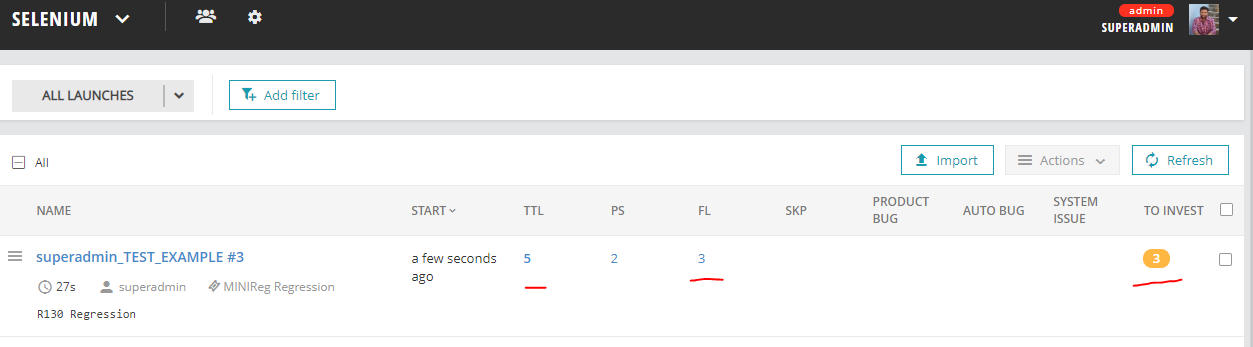


And put this on your framework

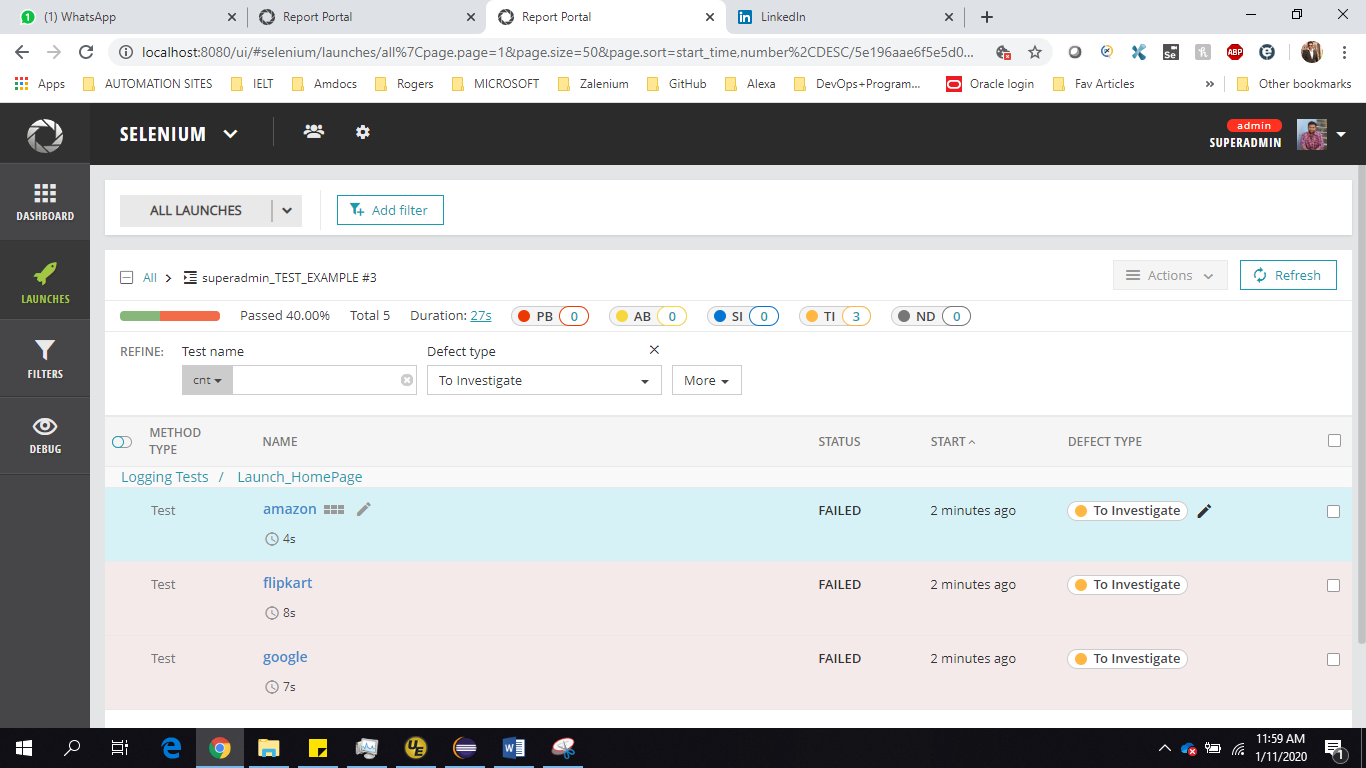


Run the Project as TestNG and see the data going inside reportportal live.

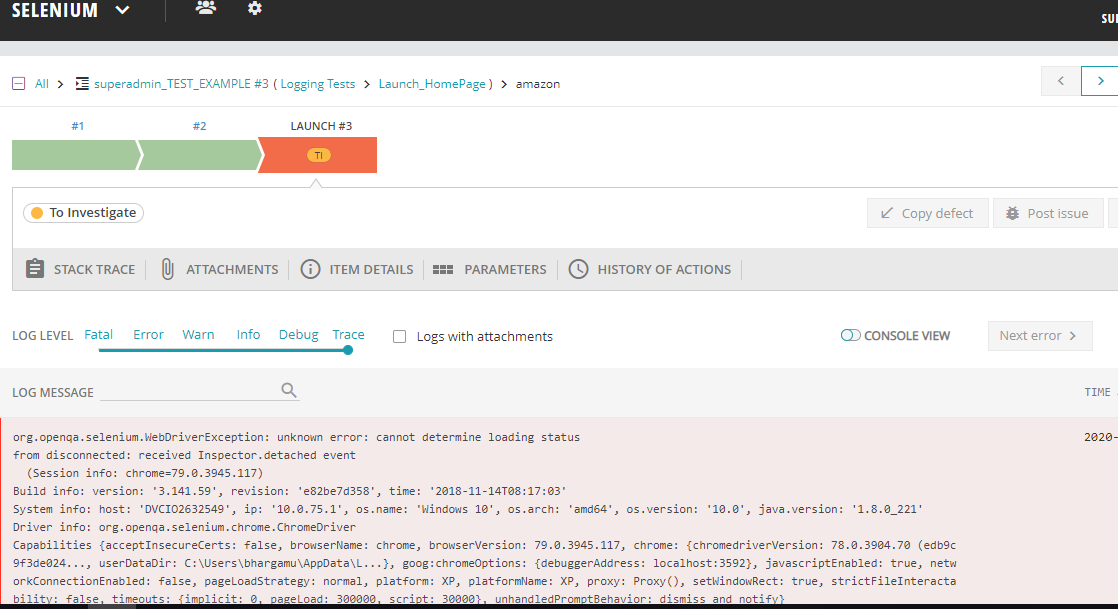
Now it is time to analyze the failures and Train your ML Engine. By Default all failures go under TO\_INVESTIGATE section.



Now click on failures and start feeding the analysis to ML Engine.

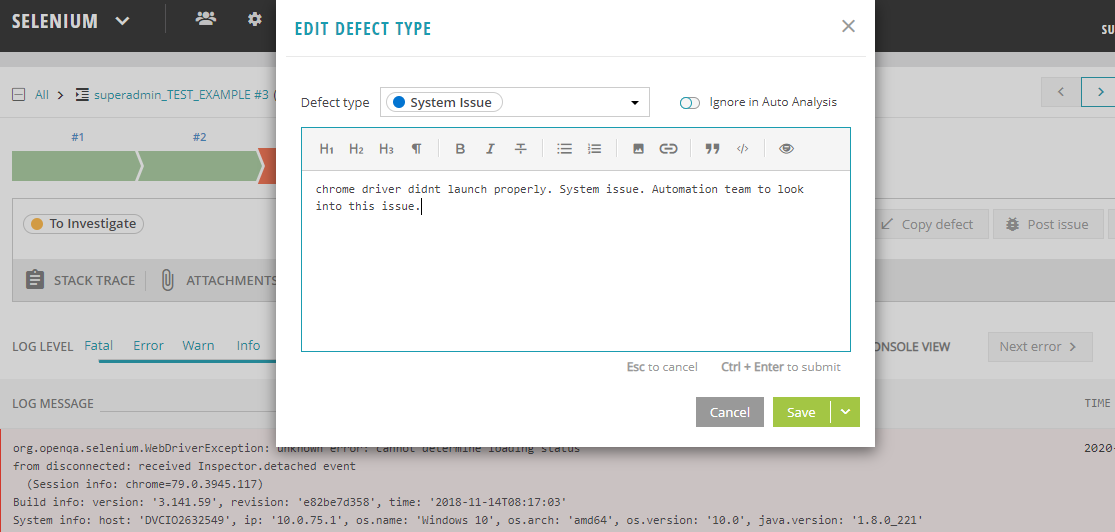


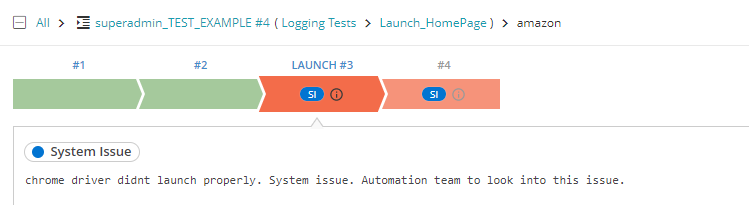
Go Inside your failed TC and check the Logs for failure



Based on Manual analysis change the mode from TO Investigate to related mode.

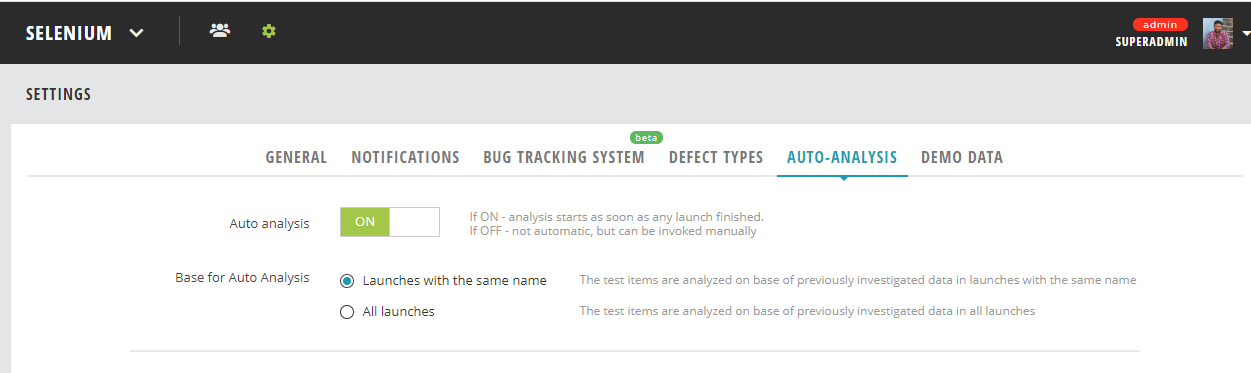
I am marking it as System issue as chrome driver didn’t launched properly.





Run all your Scenarios and make sure you feed the Engine with proper comments and analysis.

Repeat this process for couple of days and go to Settings -> turn Auto Analysis ON for Machine Learning.



Now Run your tests once again and see how issues are analyzed automatic based the training you gave to ML Engine.

And you will see the Mark AA (means Auto Analyzed) and comments from previous Run Histroy.

