

Creating Test Stability to achieve Continuous Delivery

Trisha Chetani
@trisha_1212

ME

- Introduction
- Networking - Conferences, Meetups
- Recently moved to Spain
- 125 awesome software testers by Agile Testing Days
- 1st International talk
- Twitter: @trisha_1212



WHAT IS THIS TALK ALL ABOUT

- How automated tests were slowing us down
- Increase in flaky tests
- A long journey of investigation and tackling different problems
- Lessons learned
- How I transformed myself from being a bottleneck to an enabler
- Maintained to avoid the fall back.

THAT'S ME @ VISENZE

Years : 2016

- **About Visenze**
- **My Role**



JOURNEY AT VISENZE - THE PROBLEM

FLAKY TESTS

*Definition: sometimes passes or fails**

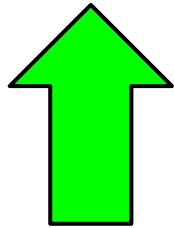
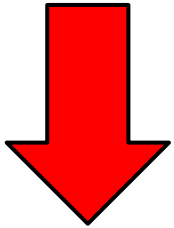
**SERVICE LEVEL
AGREEMENT NOT
MAINTAINED**

**TEAM WERE PUTTING
EXTRA EFFORT**



THE LONG JOURNEY AHEAD OF THE TEAM

FIX FLAKINESS
COLLABORATE
STABILITY
AND MUCH MORE...



**30:00
MIN**

**10:00
MIN**

CURRENT TIME
TAKEN FOR
EXECUTION(TEST
FAILED MULTIPLY
THE TIME)

GOAL FOR
EXECUTION

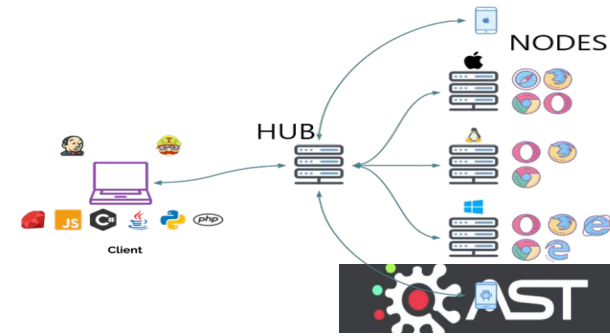
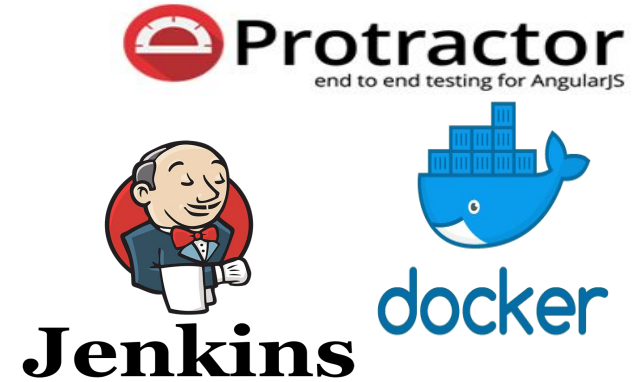
JOURNEY OF HOW I MADE TEST STABLE

ADAPTING TO CURRENT SYSTEM

- Exploring about system and application
- Exploring existing tests, at that time being maintained by developers
- Writing very first test
- Deciding the tests flows with the team

LEARNING TOOLS/TECHNOLOGY

- **Existing tool**
 - Protractor
 - Docker
 - Jenkins
 - Internal tools and many more
- **Tools I thought could help**
 - Selenium Grid



DIGGED DEEPER

DEAD CODE

**DUPLICATE
TEST**

**OUTDATED
TEST**

**MISSING
TESTS**

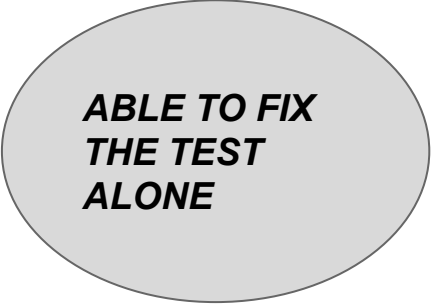
CURRENT STATUS - AFTER 5 MONTHS



**ADAPT TO
SINGAPORE**



**WORKING ON
MULTIPLE
PROJECTS**



**ABLE TO FIX
THE TEST
ALONE**



**ADD SOME
REGRESSION-
CHECKS**

BUILD TIME IS...

STILL 30 MINUTES

PROBLEMS

ENDLESS EFFORT

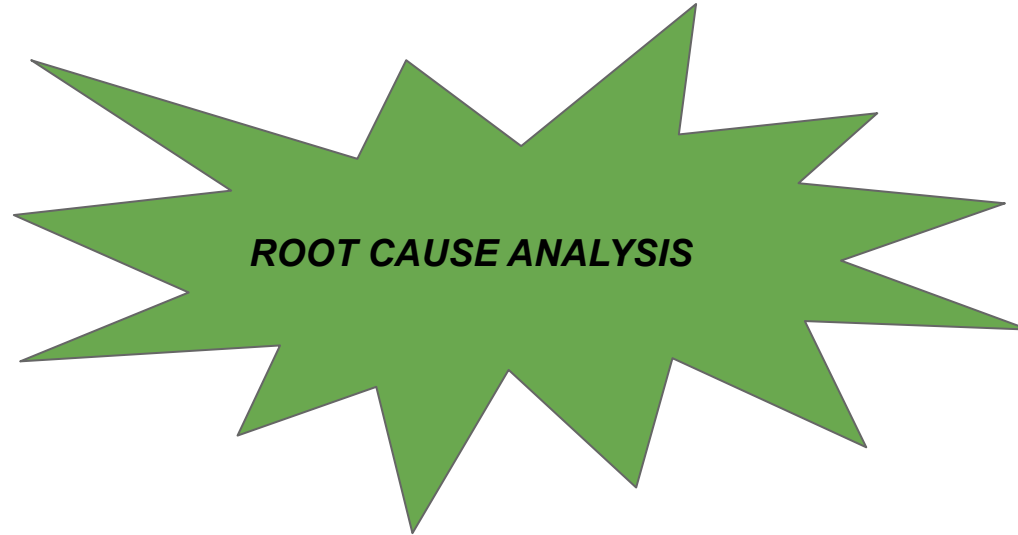
**WORK LIFE
BALANCE**

ONGOING CHANGES

Average stage times:

	build	test[unit&quality]	deploy[development]	deploy[test]	test[functional]
RED AND GREEN VICIOUS CYCLE	8s	15s	5min 24s	7min 53s	16s
RG1 Jun 23 13:58 No Changes	4s	15s	6min 17s	2min 7s <small>(skipped for 19s)</small>	17s
RG2 Jun 23 13:41 1 commits	5s	13s	5min 45s	2min 7s <small>(skipped for 29s - 16s)</small>	15s
RG3 Jun 23 13:35 1 commits	5s	15s	30s	2min 7s <small>(skipped for 19s)</small>	15s
RG4 Jun 23 13:09 No Changes	21s	13s	5min 56s	2min 15s <small>(skipped for 19s - 20s)</small>	17s

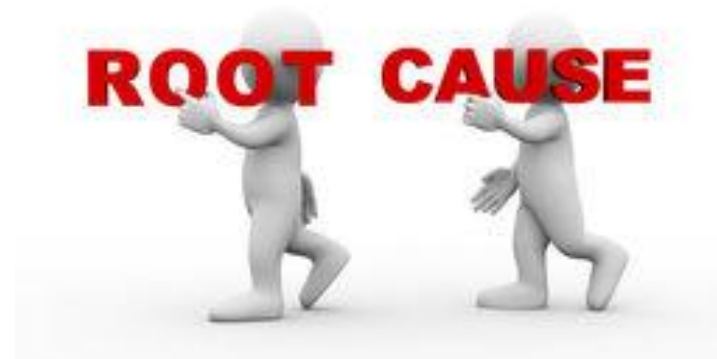
STEP TOWARDS THE SOLUTION



UNDERLYING ISSUES FOUND IN ROOT CAUSE ANALYSIS

- Locators were changing
- Basic flows were changing
- Unexpected changes
- Application slowness
- Automate all tests
- No time to make significant improvement for tests

*COUPLE OF WEEKS LATER
WHILE FOLLOWING RED
GREEN CYCLE*



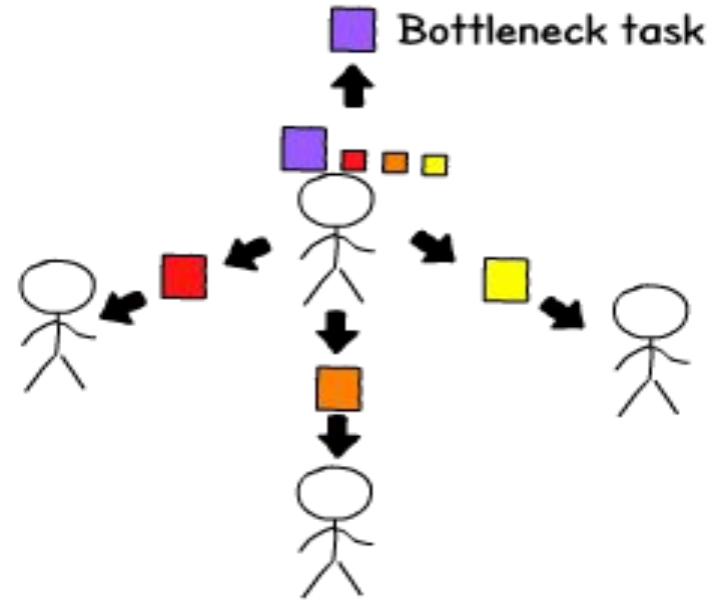
FEW WEEKS LATER - I WENT ON LEAVE



TEAM



ME - NOT SUSTAINABLE



RETROSPECTIVE DISCUSSION



RETROSPECTIVE ACTIONS ITEMS

- Stop developing new feature
- Focus on test improvement
- Team decides that developer increase unit and integration tests
- Automate test cases based on risk, priority, business impact*
- Focused on operational excellence tasks

MY ACTION ITEM - 7 MONTHS

- Digging deeper - focused not only on testing but improve on overall quality aspects.
- Open task - support on operational excellence task
- Not automating all test cases - focus on end to end user flow
- Fixing the existing flaky test cases.

CODING

- Locators(id> name> css> xpath)
- Assertion
- Waits
- Small & Independent tests

DEBUGGING

- Logs
- Screenshots
- Retry before failing test
- JS console errors
- Integrated with library - test reports

FRAMEWORK

- Ease of configuration to test
- Test data management
- Community support

ENVIRONMENT

- Docker
- Selenium grid
- Clean test environment
- Test environment and production environment

IMPROVED PRODUCT TESTABILITY

- Developer help to optimize page load, reduce downtime
- Reduce uncertainty inside application and infrastructure issues
- Reduce the technical debts and refactoring of code
- Separate the backend and frontend code base and pipeline

IMPROVED PROCESS TESTABILITY

- Root Cause Analysis for the actual failure of automated code
- Rotation task in maintaining the test code among team member
- Code review of test code
- Priority in maintaining and writing new test code

IMPROVED PEOPLE TESTABILITY

- Everyone in team started contributing to automated checks
- Increase tech stack understanding
- Improved AWS/Google cloud understanding
- Developed our knowledge on containers

FINALLY BUILD TIME IS...

5 MINUTES

MAINTAIN THE PEOPLE TESTABILITY

- Increased cross team system knowledge by evolving more people in system deployments
- Team knowledge asking for feedback about tests and product.
- Pairing with new joiners and making them up to the speed

MAINTAIN THE PROCESS TESTABILITY

- Update our automated tests continuously
- Maintain pipeline always green
- Having operational day in a sprints

MAINTAIN THE PRODUCT TESTABILITY

- Monitoring the APIs,
- Continuous Refactoring of code.
- Infrastructure improvement
- Added additional logs and implemented Kibana

SUMMARY

IF YOU WANT TO STABILISE YOUR TESTS YOUR NEED TO...

- *IMPLEMENT GOOD CODING PRACTICES*
- *ENCOURAGE TEAM RESPONSIBILITY FOR TEST IMPROVEMENT*
- *FOCUS ON HAVING STABLE TEST ENVIRONMENTS*
- *HAVE PEOPLE WITH THE RIGHT SKILLS AND KNOWLEDGE*
- *HAVE GOOD PRODUCT TESTABILITY*
- *CREATE AN ENVIRONMENT OF CONTINUOUS IMPROVEMENT AND LEARNING*

THANK YOU



carousell

9TH TAQELAH MEETUP:

- Creating test stability to achieve continuous delivery
Trisha Chetani, QA Evangelist
- Let's know mobile security
Anandh Kumar, Senior Associate, Cognizant Technology Solutions



Thur, 2 May



06:30pm - 08:30pm



GovTech Hive
1 Fusionopolis View, #08-01 Sandcrawler, Singapore 138577



TRISHA



ANANDH

V I S E N Z E



Everyone.



@ajay184f



@friendlytester



@aahunsberger



@mariakedemo



@kriscorbus

@trisha_1212



Questions?

@trisha_1212