

DIGITAL QA TRANSFORMATION

BUILDING OUT A WORLD CLASS QA CAPABILITY

CAMERON BRADLEY

MAY 16/2018

OVERVIEW

1

LEADERSHIP

7

AUTOMATION STATISTICS

2

TEAM FOCUS

8

AUTOMATION FRAMEWORK

3

PROJECT FOCUS

9

DIGITAL MOCK SERVERS

4

DIGITAL QA

10

QA ACADEMY

5

QA TRANSFORMATION

11

TECHNOLOGIES

6

STRATEGIC GOALS

12

SWOT

LEADERSHIP



Other Interests

Fin\$hare



The Bug Exchange



WHO AM I?

Cameron, Head of QA for Digital Melbourne

I joined Tabcorp as a consultant lead automation engineer in March 2016 on the abacus (terminal operator system replacement) project which sits in the retail digital space.

In the role I drove significant transformation of the wider teams approach to QA across Digital. Due to our teams improved performance and growth I was offered the newly established position at Tabcorp to Head up the QA team across all Melbourne Digital projects.

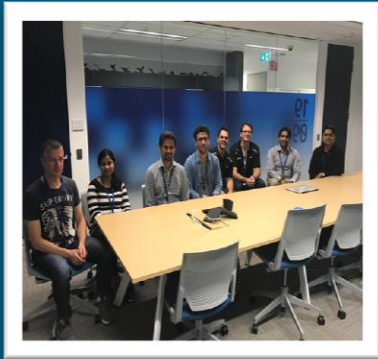
Prior to Tabcorp, I spent numerous years working across various companies (startup and finance) as both a software engineer and an automation engineer.

My passion for QA and automation really grew from experiencing first hand the incredible business value and quality that can be delivered by running a technical, continuously improving and highly skilled digital QA practice with automation at the forefront.

I believe my key strengths are strong technical and coding skills, the outcome driven approach and highly visible culture I cultivate within the team, my passion for team learning and continuous improvements as well nurturing a visionary team approach to QA that ties the best of both manual and automation testing approaches to ultimately delivery quality systems for our customers.

Tabcorp

TEAM FOCUS



WHO ARE WE?

DIGITAL MELBOURNE – QA TEAM



Cameron Bradley

Role: Head of Digital QA

Project: Active across all Digital & Digital Retail Projects.

Anthony Vertuli

Role: Senior QA

Project: Greenmoon, Form & Abacus.

Devinder Singh

Role: Senior QA

Project: Greenmoon & AML-SSC.

Leon Vesel

Role: Senior QA

Project: Keno Web & Keno Android

Raffi Chabdjian

Role: QA

Project: Longitude & EBT.

Sanghamithra Sujith

Role: QA

Project: Greenmoon, Information Screens, Information Center & EBT.

Aswin Golla

Role: Senior QA

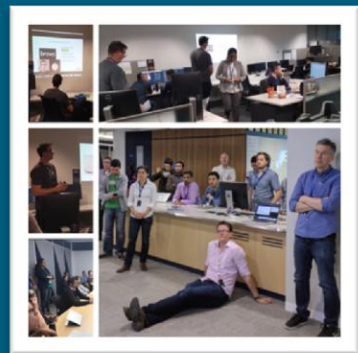
Project: Keno Digital API's

Prasanna Kumar

Role: QA

Project: Keno IOS

PROJECT FOCUS



OUR PROJECTS?

DIGITAL MELBOURNE

GREENMOON

INFO-SCREENS

ABACUS

EBT

Form

AML-SSC

Control Center

PRAGMATEAM



DIGITAL QA TRANSFORMATION

The Digital Melbourne QA practice was established in Melbourne on January 8th – 2017

What did Digital Melbourne look like on day 1?

Team	1 Automation Engineer (me) and 8 Manual Testers.
Project Growth	6 Projects across Digital Melbourne.
Agile QA (embedded QA's)	2 QA's embedded in agile sprint on projects.
Automation Tests	Minimal Unit Testing - 16 UI Automation Tests on a single project.
Mock Servers	1 mock server on 1 project – Not used for testing.
Bugs discovered	Not measured and minimal.
Automation Coverage	Automation Coverage was not measured and minimal.
Automation Tests Executed	Around 100 tests executed daily in CI
Manual Testing Effort saved	Around 20 minutes a day.
Duration of TRC	Not measured but was high due to manual test effort.
Automation Executing in CI	No automation executing correctly in CI (pull request)

DIGITAL QA TRANSFORMATION

Since then.....

What does Digital Melbourne look like 15 months later?	
Team Growth	8 Digital Quality Assurance Experts (average of 1 QA for every 4 devs)
Project Growth	12 Projects across Digital Melbourne.
Agile QA (embedded QA's)	All QA's embedded in agile sprints on projects.
Automation Tests	10,000s of Unit Tests - 3401 e2e - automation tests across twelve projects.
Mock Servers	5 mock servers - all 12 projects using mocking.
Bugs Discovered	We have over 750+ documented defects discovered. (record low prod defects raised)
Automation Coverage	All projects are trending towards 80-90% coverage.
Automation Tests Executed	17,405 automation tests executed every day.
Manual Testing Effort saved	Estimates are around 854 hours saved daily
Duration of TRC	Measured and trending down (MOM)
Automation Executing in CI	11 (of 12) projects executing correctly in CI

DIGITAL QA TRANSFORMATION

Focus on the fundamentals

Define what Digital Quality Assurance means to us as a team.

Develop a consistent definition of done across digital.

Standardise automation (unit, service layer and UI) technology stack.

Learn and follow the test pyramid approach to testing – Understand value of automation testing.

Create a strong QA community within Tabcorp.

Understand and convey the benefits of a strong automation practice.

Development and learning opportunities for all team members.

QA updates and visibility to the wider technology team.

Builds and Automation executing correctly in CI.

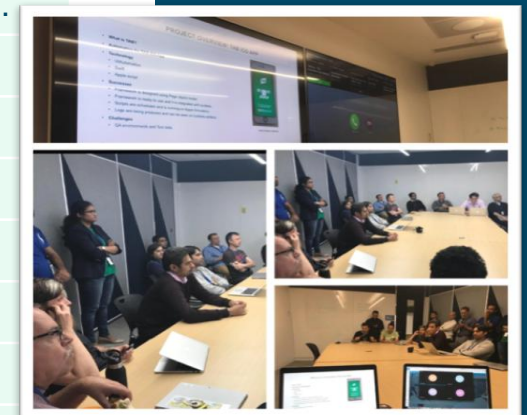
Mocking across all Digital Systems.

Working hand in hand with Delivery, BA, Dev-ops, Development and QA.

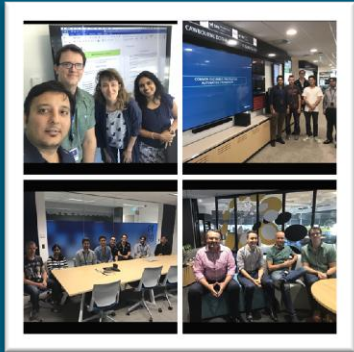
Test reporting generated in CI and automation performance metrics.

Discovering defects before our customers.

Constantly innovate and look to leaders in the industry.



DIGITAL QUALITY ASSURANCE



WHAT IS DIGITAL QUALITY ASSURANCE?

We believe QA should be “owned” by the whole project team. That is to say that all team members (delivery, BA, dev-ops, developers and QA) should have a collective and collaborative approach to ensuring quality in our systems being delivered to customers.

Our team members devote around 80% of their time to working as embedded QA’s in their agile project teams doing a mix as necessary of manual, exploratory testing and automation testing as part of the definition of the done.

The other 20% is around driving continuous improvements, participating in social forums, attending and supporting learning opportunities as well innovation and automation architecture growth.

Automation Testing across the test pyramid is at the forefront of how we operate and approach quality in our teams and for our systems.

DEFINITION OF DONE

Stages of the definition of done:

1. Conception of the card
2. Development
3. Code review
4. In QA
5. Business Testing
6. Pre-release
7. Release verification

DEFINITION OF DONE

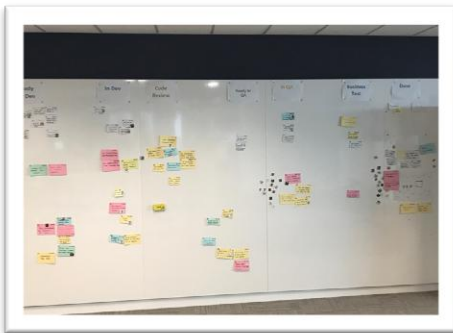
	Conception	Development	In QA	Business Testing	Pre-release	Prod Verification
DESCRIPTION	Conception is the earliest phase of the product development lifecycle. It involves business stakeholders coming together to define business requirements and new features for product development.	Development begins when the requirements are defined and prioritised and ready for development. It involves the developer understanding the feature to be developed in order to begin development of the feature.	When a developer believes a card is developed it is then ready for QA to begin testing. This involves the QA testing the acceptance criteria and ensuring the card is automated across the test pyramid.	Business testing is about getting your business stakeholders across the newly developed feature to ensure it looks as expected. At this point the card has passed "QA" and should be as expected. This is to ensure that this is still the case.	Pre-release is when a sprint of work, feature or set of features is ready for release and final pre-release testing measures can take place. This is a great opportunity to use exploratory testing methods to try and find outlier defects or issues.	The newly developed features have been released into production. Now it is important to verify that production is working just as expected in reality. Prod verification is purely a last step – peace of mind task.
PHASES	Defining Acceptance Criteria Business stakeholders discussing and defining business requirements with delivery and BA. Sprint Planning between team to allocate story points on cards required to be developed.	Kick-off Dev, BA, and QA come together to officially kick off a new card together. Development Development begins on the card in development that has passed kick-off.	Hand-over Dev, BA and QA come together to receive a demo of the development completed functionality. Unit Testing Demo: Dev takes the QA through the unit testing in order to increase the QA's understanding of coverage. Acceptance criteria (QA) testing: QA & Automation begins on the cards acceptance criteria.	Demo Business Stakeholder / QA come together to share a demo of the feature that has passed QA. Business Stakeholder Analysis Business stakeholder analyses the newly developed feature to see if it is working as expected.	Bug bash The whole team comes together to bug bash the release functionality and wider application. Environment testing Preprod environment testing exploratory testing takes place here. Regression testing Manual and automation regression testing should be executing and passing. Regulatory: BMM testing	Production only scenarios testing Verifying production only scenarios like deposit and withdraw with real accounts. Sanity Check A simple set of key scenarios executed manually or via automation to ensure production is working as expected.



Stages of the definition of done



	1	2	3	4	5	6
QUALITY ASSURANCE MEASURES	BA: Understandable requirements provided for sprint planning story cards. DEV/QA: Mocking feature requirements defined by the team. QA: Participation in sprint planning estimates.	BA: Ensure kick-off takes place between Dev / BA and QA. DEV/QA: Identify any potential missing gaps in requirements. DEV/QA: Identify any further mocking that might be required for optimal testing. DEV Ensure newly developed code is unit tested.	DEV/QA Ensure hand-over takes place between DEV / BA and QA. DEV/QA Complete a walkthrough of the unit testing completed for the newly developed feature. QA Test acceptance criteria with a mix of manual and automation testing as necessary and required.	Business Stakeholder / BA Ensure a demo is completed on the newly developed feature. QA Once a card has been successfully signed off by the business stakeholder present it is considered done and ready for deployment.	QA Champion the bug bash and organise the team, features to be tested, devices, browsers to be tested on by team members. QA System specific – ensure Regression testing is done as required. QA System specific – ensure required BMM testing is completed as required.	QA Ensure production only testable scenarios are executed as expected. QA Run a peace of mind sanity check through the system to ensure it is working as expected



Automation testing is expected as part of our definition of done!

Test Automation Benefits / ROI

Early bug detection

Reusability

Platform and resolution testing

Test the impossible

Reliability

Reporting & Screenshots

Team morale

Automation running 24/7

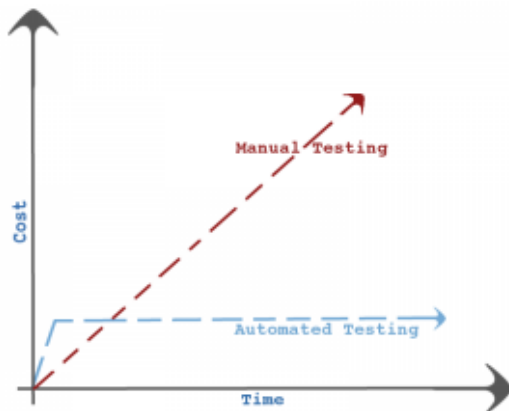
Simultaneity

Less manual effort

Team collaboration

Saves time for the team..

Lowers regression periods



TEST STRATEGY ACROSS PROJECTS

“Consistent Team defined test strategy across projects is essential.”

Defined test approach:

1. General QA approach
2. Regulatory requirements
3. QA Environments
4. Environment / External dependencies
5. Continuous integration
6. Communication channels
7. Repositories (GIT)
8. The team (External teams)
9. Compatibility
10. Project Tooling
11. Important to know

Casino Project - Test Strategy



Cameron Bradley

Last modified Jan 16, 2018



- Purpose:
- General Test Approach:
- Test Automation Approach: (The Definition of Done)
- Automation Coverage:
- Automation Reporting:
- Test Automation Statistics Initiative
- Production Health Check
- Automation Testing Gaps:
- Regulatory:
- Environments:
- Environment Based Test Limitations
- CMS:
- Continuous Integration:
- Team Communication channels:
- Dependencies:
- Repositories:
- Team:
- Compatibility:
- Project tooling:
- Important to know:
 - Progressive Jackpots - DO NOT PLAY

COMMON AUTOMATION FRAMEWORK

The common automation framework has been developed collaboratively by both developers and QA's across Tabcorp.

It provides a comprehensive suite of “out of the box” cucumber steps on top of protractor/JavaScript and selenium that will work with any angular or react application

The common framework has empowered all our QA's and many developers to be able to add automation as part of the definition of done.

Generic Behavior Driven Development

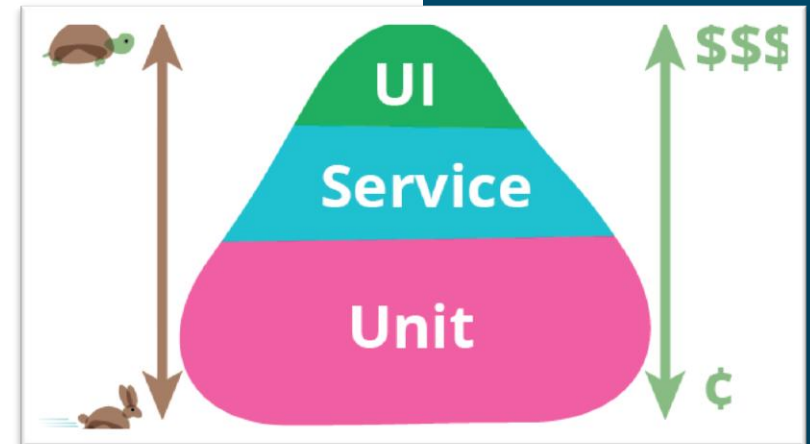
The screenshot shows the npm package page for 'generic-cucumber-protractor-how?'. It includes the package name, a 'learn more' link, the publisher 'cameron.bradley' from a month ago, and version information stating '1.7.0 is the latest of 70 releases'. A GitHub repository link is provided. Below this, there is a section for 'Collaborators list' with several profile icons. The 'Stats' section shows download counts: 95 in the last day, 492 in the last week, and 1,738 in the last month.

TEST PYRAMID APPROACH

The Digital Melbourne team are big believers in the agile test pyramid approach to testing.

We are aligned in our belief that the majority of automation tests should be in the unit and service layer followed by the user interface.

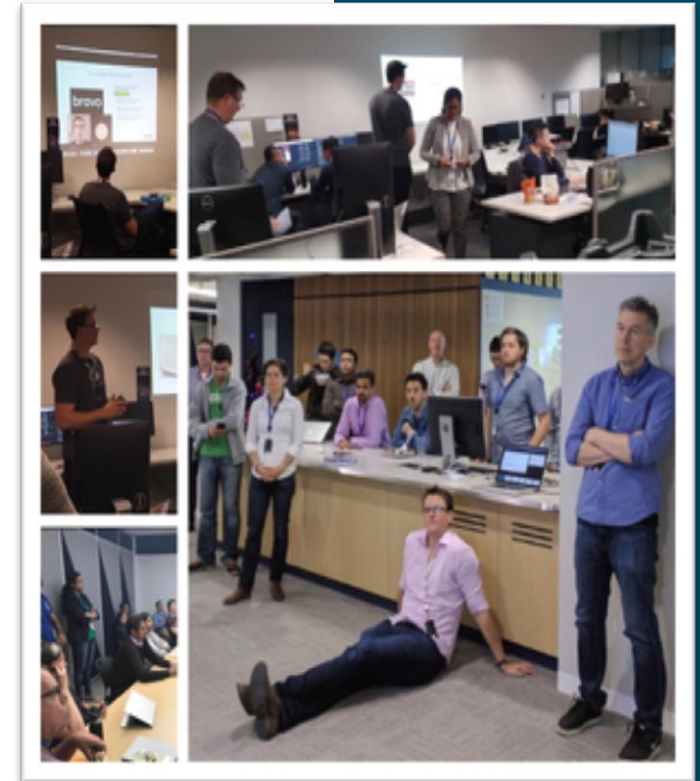
We empower, trust and encourage our QA's to be experts in their systems they are testing and accordingly endeavor to work hand in hand with developers to understand the level of UI automation required to provide confidence in our system releases.



CREATING A STRONG QA COMMUNITY

The Digital Melbourne QA team are leaders in the social scene at Tabcorp as judged by our peers. We operate quarterly (brownbags, automation-syncs, showcases, team catch-ups) and participate in as many end of sprint showcases as possible.

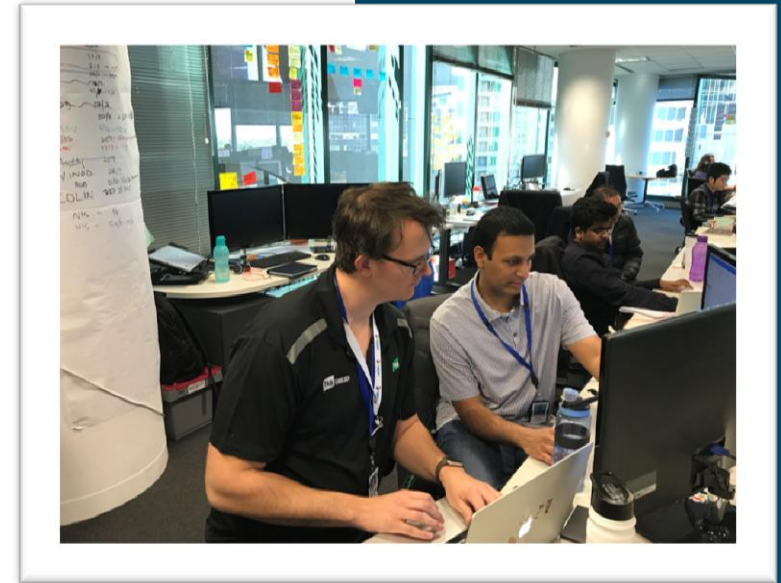
We believe it is essential to be as visible as possible and take everyone on the journey with us to achieve our best outcomes. We also believe these social forums hold us accountable to our peers.



DEVELOPMENT AND LEARNING

It is our belief that to retain and excite our team we must always be improving our skills and learning new things.

Across Digital Melbourne we have been doing fortnightly catch-ups and pair programming sessions. Many of our team members began with no cucumber, protractor or JavaScript experience and are now competent, effective and efficient in their field.



VISIBILITY AND UPDATES

The ‘Monthly Digital QA report’ and the “Monthly Digital QA Newsletter” are just two examples of how the team strive to share all our work across key business stakeholders.

We understand the power of taking these stakeholders on the journey with us. We share examples of collaborative initiatives and slide packs for all social activities.

This culture of sharing not only provides visibility but holds us accountable for what we are setting out to achieve together as a team.



CONTINUOUS INTEGRATION (CI)

“If the automation is not running in CI, It does not count.”

We believe that our builds and automation testing across the test pyramid should be executing in CI. A manually triggered automation suite defeats the key benefits of automation testing.

All code commits should trigger the build (& tests) and block the team from merging until the tests are passing. This mentality has led to 11 of our system being correctly hooked up in CI so far as well as build monitors and reporting across projects.

- ☞ Sun Casino
- ☞ Keno Web
- ☞ abacus-dev-seed
- ☞ abacus-dev-seed-acceptance
- ☞ api-service-retail-management
- ☞ api-service-retail-transaction
- ☞ FORM
- ☞ Keno Android
- ☞ offer-management-tool
- ☞ terminal-parts
- ☞ www-abacus
- ☞ www-on-course
- ☞ www-ssc-aml
- ☞ www-sunbets
- ☞ www-sunbets-casino

BUGS

Automation

- Running on every code commit.
- Executing otherwise impossible scenarios.
- Reliability in testing all scenarios.
- Running 24/7 – environments.

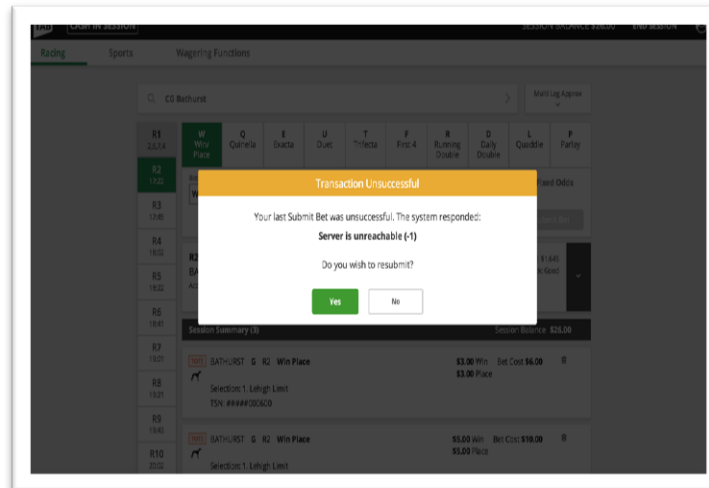
Exploratory / Manual

- Random user experience like a customer.
- Great for finding UI issues.
- Business stakeholder input.
- Sometimes it has to be manually tested.



Just one example:

Betting API going down during place bet on Greenmoon:



Power of **mocking** and **automation**

AUTOMATION STATISTICS

Digital Melbourne – End to End (E2E) - Test Automation Statistics

	Total Automation Tests		Automation Coverage [including unit + service layer + UI]	Automation tests executed	Manual testing effort saved	Duration of Total Regression Cycle	Defects discovered in Development	Defects discovered in Production	Automation Executing in CI
	04/02	04/03		Total Daily	Hours Daily	Hours / Days	For The Month	For The Month	buildkite
Sun Vegas	145	165	88.5%	1,130	57 hours	2h:10mins	5	1	Yes
Sun Casino	34	40							
Keno Web	239	239	90.0%	1,259	48 hours	2h:30mins	10	1	Yes
Keno Android	24	25	20.0%	130	16 hours	8h:00mins	8	2	Yes
Keno IOS	28	31	44.2%	112	5 hours 24 minutes	2h:47mins	10	1	Yes
Keno API	181	238	6.03%	362	12 hours	40h:4mins	13	0	Yes
Greenmoon	1303	1346	82.05%	13,460	652 hours	3h:22mins	5	0	Yes
Abacus	1255	1255	90.0%	1255	62 hours	6h:30mins	3	0	Coming Soon
Form	20	20	10.0%	36	2 hours	12hours:30mins	3	0	Yes
SSC	38	38	90.0%	190	9 hours 30 minutes	6hours:10mins	2	0	Yes
EBT	63	63	15.0%	63	5 hours	5 days / 5 people	5	1	Coming Soon
Total	3,330	3,460		17,405	854 hours		64	6	

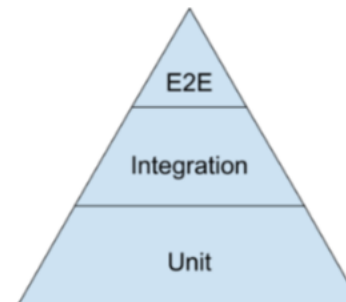
AUTOMATION PERFORMANCE METRICS

“We must measure the value received from automation testing.”

Sun Casino – End to End (E2E) - Test Automation Statistics								
	Automation Tests	Automation Coverage [unit + service layer + ui]	Time taken to execute suite.	Automation [UI] tests executed	Manual testing effort saved	Duration of Total Regression Cycle	Defects discovered in Development	Defects discovered in Production
	Automation Executing in CI			Total Daily	Hours Daily	Hours / Days	For The Month	For The Month
Sun Casino Angular Application	@build - 18 ↑	62.0% ↑	6min – 24 secs	1,140	58 hours	2h:05mins	Coming Soon	3
	@smoke - 45 ↑	71.0% ↑	13min – 51 secs					
	@solid - 77 ↑	81.0% ↑	20min – 3 secs					

Sun Casino – Unit & Integration Testing - Statistics			
	Total Unit & Integration Tests	Test Coverage	Time taken to execute test suite
Sun Casino Angular Application	42	25.0%	35 secs
Sunbets casino React Application	138 ↑	26.0% ↑	7 mins – 31 secs
Web parts	1000s ↑	Extensive ↑	10 mins – 41 secs

Test Pyramid Approach



COMMON AUTOMATION FRAMEWORK

Common Automation Framework Repository

COMMON REPOSITORY FOR THE GENERIC CUCUMBER PROTRACTOR FRAMEWORK
<https://www.npmjs.com/package/generic-cucumber-protractor-framework>

Project – Automation running using the common repository

Project	28/02/2018	02/04/2018
Sun Vegas (Melbourne Digital)	Completed	Completed
Sun Casino (Melbourne Digital)	Completed	Completed
Keno Web (Melbourne Digital)	Completed	Completed
Greenmoon (Melbourne Digital)	Coming Soon	Coming Soon
Abacus (Melbourne Digital)	In Progress	Coming Soon
Form (Melbourne Digital)	Completed	Completed
AML-SSC (Melbourne Digital)	Completed*	Completed
EBT (Melbourne Digital)	Coming Soon	Coming Soon
Rubix Sports (Wagering Technology)	Completed	Completed
Sun Bets (Sydney Digital)	Completed	Completed
Sky DVS (Wagering Technology)	Completed	Completed

APPROVED BY TABCORP SECURITY FOR OPEN SOURCE!

- Single automation framework that can be installed in each (angular/react) project.
- Common repository collaborated on by developers and QA's across Tab.
- Changes and improvements to the framework hit all projects at once.
- Increased collaboration across Tabcorp in achieving quality outcomes with automation.

DIGITAL MOCK SERVERS

Digital Mock Servers

DIGITAL MOCK SERVERS REPRESENT A COLLABORTIVE EFFORT ACROSS THE WHOLE DIGITAL TEAM TO IMPROVE THE WAY WE DEVELOP AND AUTOAMTION TEST OUR SYSTEMS.

Digital Melbourne Projects	Mock Server	Description
Sunbets SunCasino SunVegas SunCasinoLive	sunbets-mock-server inbox-mock-server	Mock servers to allow end-to-end testing across casino. Some features include authentication, deposit, offers & notifications.
Keno Web Keno Android Keno IOS	keno-mock-server	Mock server to allow end-to-end testing across all keno platforms. Some features include play keno, rewards, deposit & statements.
Greenmoon Abacus Form EBT ssc_aml	terminal-sts-stub terminal-api-service-information-stub service-cdef-mock	Mock servers to allow end-to-end testing across greenmoon, abacus, form, ebt and ssc-aml. Some features include betting, statements, login, race navigation, closed races & sports.

ONE OF OUR PROUDEST COLLABORATIVE ACHIVEMENTS

- Developed collaboratively as part of the sprint effort between dev and QA.
- Team code commits, and merges in to the main code base execute automation in CI that runs on the mock servers. This ensures stable automation suite runs across systems as it reduces reliance on sometimes unstable test (QA) environments.
- Allows our developers and QA's to automation test what might otherwise be impossible

QA ACADEMY

QA Academy

A POWERFUL COLLABORATIVE ACADEMY TO DRIVE QUALITY ASSURANCE IMPROVEMENTS AND OUTCOMES TOGETHER AS A TEAM.

Action Item	Status	Update
Redefine and refresh the definition of done	60%	The team has defined the various stages of the definition of done. A official template has been created now to create our definition of done.
Improved documentation for digital mock servers	25%	The team has worked together to agree on a format for how we want to document our digital mock servers. Our sunbets mock server is the first of 6 to be documented as part of this initiative.
Pair automation programming sessions	80%	Pair automation programming sessions have now been kicked off across Digital Melbourne with team members coming together to skill up and share knowledge.
POC Mocking for Tab Android	10%	Information service mock server identified as a good candidate for our first mock server to share. We have now introduced a change to the allow the mock serer to run standalone in preparation for this task.
Future QA Tech Sessoin with team	0%	We are looking to introduce a QA technology session to discuss and look at interesting new technologies, discuss our automation framework.
Automation suite analysis for Tab (devices)	0%	A simple analysis is to be completed over the TAB (IOS) and (Android) test suites to analyse a pathway forward to improved automation outcomes.

DRIVING IMPROVEMENT OUTCOMES TOGETHER

- The key purpose of QA academy is to come together as a team and define items of continuous improvement.
- We have a framework for supporting these action items to achieve real and visible outcomes for quality.

TECHNOLOGIES (CONSISTENCY)

PLATFORM

WEB

IOS

Android



TECH STACK



AUTOMATION



GENERIC CUCUMBER PROTRACTOR
FRAMEWORK



JavaScript



Tabcorp

STRATEGIC GOALS

Destination 2019

Build Digital Mock Servers together to support lotteries and Keno.

Explore & implement a common automation framework approach to automation testing.

Bigger picture – improving QA beyond keno and lotteries.

Refreshed definition of done together as a team.

Improved accountability of our work as a team.

Work hand in hand with the keno host team.

QA Leadership meetings across tech streams.

Leverage the best of both Lotteries and Keno QA.

Defined consistent test strategy documentation together.

Ensure our CI Pipeline's are reaching full potential.

Increase automation coverage across projects.

Exploratory testing as a team.

Technology Leadership Meetings.

Reduced duration of total regression cycle.

Follow the Test Pyramid approach to automation testing.

Reduced production defects.

Increased manual test effort saved.

Lotteries and Keno team attending QA Academy.

Fortnightly catchups with team members to drive outcomes.

Pair automation Sessions

Social scene Improved visibility.

Build monitors

Learning and development opportunities for team.

QUESTIONS?

Tabcorp



Luxbet



INTECQ
Limited

TO ADVANCE THE WAY WE PLAY

Tabcorp



Luxbet



INTECQ
Limited