

## FIT3047

### Testing - all sorts

Week 6 -



## Housekeeping



## On the menu this week

Watch:

Read:

Participation quiz

Extra Readings (on moodle)



## Roadmap: Where we are

### Plan for project



#### Project Kickoff

Discovery workshop

Team understanding and rules

Business understanding

System Overview/ Business Vision

#### Project Initiation

Trello with backlog

Project Governance Portfolio

Tech stack

Iteration 1 kick off - Top priority backlog items

Iteration 2 kick off - Next top priority backlog items

User Docs

Tech Docs

Handover Package (leverage PGP)

#### Iterate



MONASH  
University

8

## Test Driven Development at an Acceptance Level

You begin with what is needed  
**You design your tests before you start building**  
(You do not design your tests after you have built)

- Created when you analyse what you want to develop
- Based on clear and well-defined use cases/ functional requirements
- Bottled in user stories and acceptance criteria
- Start there and then build to those tests.
- Should be developed collaboratively.



MONASH  
University

9

### Approach to Testing

- For each test case devise comprehensive sets of test data that contain
  - Correct and incorrect data
  - Incorrect data
  - Weird data
- Think like an inexperienced user or mischief-makers when devising test cases and test data
- Thoroughly document test cases and test data before commencing testing

IE Seminar Series

MONASH  
University

Testing.10



### Approach to Testing prep Question

**Field - Customer name**

**what will you test for?**

**Examples of instances of test data**

**Field - Date of booking required, entered? drop down?**

**what will you test for?**

**Examples of instances of test data**

## Functional testing

Validates the system against business requirements. Evaluates *what* the system does.

Any time your user has to enter information or do anything

- No data
- Valid data set
- Invalid data set
- Inc. dates -
- Incorrect data format
- Boundary condition data set
- Duplicate data
- Weird data
- Deletes (cascading deletes?)



## Non Functional Testing

Tends to qualities that reflect the quality of the product in the context of the suitability perspective of its end users. Evaluates *the way* the system does it.

- E.g Usability Testing
- mobile responsive AND (perhaps not in this instance)
- Load Testing
- Availability Testing
- Performance Testing
- Compatibility Testing



## Test Case - Example

Test Case: Login Page					
Functionality	Test ID	Description	Test Date	Tester	Test Type
Login as user	1	Users must use username and password to login when accessing the system	14/04/2012	BTF	Unit Test
	<b>Test Criteria</b>	<b>Anticipated Result</b>	<b>Actual Result</b>	<b>Status</b>	
	Load the login page	Login page loaded completely	Login page loaded	Passed	
	Leave both username and password fields blank and attempt to log into the system	User cannot login to the system. Error message displayed on the bottom of login page	Warning message appears and access denied	Passed	
	Entering the wrong username with a password to log in	User cannot login to the system. Error message displayed on the bottom of login page	Warning message appears and access denied	Passed	
	Entering a correct username with a wrong password to login	User cannot login to the system. Error message displayed on the bottom of login page	Warning message appears and access denied	Passed	
	Enter a correct username with a correct password to login	User successfully logged into the system	User login granted	Passed	

## Test Case - Example

Test Case: Create Client					
Functionality	Test ID	Description	Test Date	Tester	Test Type
Create Client/case/session	2	Testing the function of Create Client, all fields must contain correct input for the system to create a new client.	14/04/2012	BTF	Unit Test
	<b>Test Criteria</b>	<b>Anticipated Result</b>	<b>Actual Result</b>	<b>Status</b>	
	0000450	1. Fill in correct information 2. Test with blank 3. Test with numerical data 4. Test with symbols 5. Test with letters 6. Select using drop down box 7. "Other" text box with attached with "other" option	Client cannot be created with invalid data type Warning message if any field is blank or invalid data type Telephone number change to "contact number" Confirmation message show to user if client's age is under 18 Language spoken (including 'Other' Checkbox) Rent Amount change to 0 if NA if homeless Case Time set as current time Case time manually adjusted "other" textbox is able to type when choose "other" option	Client cannot be created with invalid data type Warning message if any field is blank or invalid data type Telephone number changed to "contact number" Confirmation message show to user if client's age is under 18 Language spoken (including 'Other' Checkbox) Rent Amount change to 0 if NA if homeless Case Time set as current time Case time manually adjusted "other" textbox is able to type when choose "other" option	Passed Passed Passed Passed Passed Passed Passed Passed
	0000460				
	0000451				
	0000528				

## Do not need 400 pages of test cases

Rather have 400 actual tests

### Reality of testing

Many alternatives and combinations in this case

- Too many to effectively test every error combination with every type of test data
- Use your judgement
- Do all single error alternatives
- Do a reasonable number of combinations
- Use a good range of types of test data
- Always test with weird or dangerous data
- Consider automated tools or suites
- Many free open source and trial versions available

## Testing does not mean

- “Yeah I’ll fix the screen layout later, but the function works!”
- “Don’t enter incorrect data, as we have not done validation yet”
- “Wait! don’t hit that button that’s in the next iteration...”
- “We will make everything ‘look good’ later !”
- “No this is the order you have to enter the information it was designed this way”



## Testing involves:

- Design and documentation of test cases
- Recording and reporting results of testing
- Test management
- Test process, techniques, test measurement
- Test coordinating, planning and reporting

All driven by business/ project/ user/ audience requirements

1. Plan your testing regime thoroughly
2. Record any bugs found in Trello
3. Allocate sufficient time for tests and fixing



## Automation of testing

- For applications that you have developed (not necessarily these ones)
- Scripting languages (Java, Python, JS etc)
- UI-automation (Selenium/ Selenide, locators, html, css selectors)
- API testing (Rest)
- Bulk data
- Comparing results



## Lots of Tools for Testing

- Load Impact
  - BoomQ.io
- Website rating test:
  - Gtmetrix.com
- Accessibility Testing:
  - Nibbler: 'Tastes' your site, Tests for colour blindness, accessibility, detailed and useful results report



MONASH  
University

20

## More Tools for Testing

- Browsershots:
  - Web browser compatibility testing
  - Google "browser testing" for many options
- Performance testing
  - Webpagetest.org
- Accessibility testing
  - Wave.webaim.org
- Responsiveness testing
  - Responsivedesignchecker.com



MONASH  
University

21

You will need a test environment...

MONASH  
University

22

This is not a test environment



But Your dev folder is!

MONASH  
University

23

## Testing is important.

If we see that you haven't tested, (and we will know) your build out of your iteration will fail.

We will test your build against the report that you have that it has been built from. It has to match – again, if not  
That build will fail!

Your Mentors



MONASH  
University

2

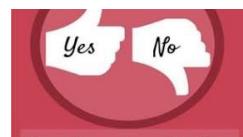
# Acceptance Testing

5.30

MONASH  
University

## ACCEPTANCE TESTING prep questions

- When?
- Who should be there?
- Where?
- How long will it take?
- Any resources required?



MONASH  
University

26

## for 3048 and a client ACCEPTANCE TESTING

- You will give them a form/list containing all those high level user stories (in client terms)  
(that you said you would deliver at the beginning)
- You will give them user instructions
- You will upload to the client's ISP - hidden
- They will sign off or otherwise
- You will fix - They will retest
- *When completed*



THIS CAN BE YOUR FORMAL SIGN OFF  
OF SYSTEMS ACCEPTANCE of that iteration

MONASH  
University

27

## ACCEPTANCE TESTING

- The goal of Acceptance Testing is to assess if the system can support **day-to-day business**
- To verify the system matches the requirements, the business needs
- The final stage of testing before the client / user accepts that new system.

## Integrity Testing

## Client acceptance testing

Can the user use the software?  
Is it really what they asked for?  
Do they have trouble using it?  
Does it behave exactly as anticipated?



- It is not your client's role to test your system for bugs.
- They are looking to see if the system fits their business needs that they have described to you (and you promised them!)

A question about acceptance testing.

### Participation quiz question

Final User Acceptance testing, when we have a client is carried out where?

## INTEGRITY TESTING



- This is where your studio mentor will test your work.
- It is acceptance testing but on steroids.
- Your mentor will not skip over things because they are OK (but not quite right) – as your client might.
- Your mentor will not think, “*that's OK I can write the number down on a piece of paper and then enter it again on the next screen*”
- Your mentor will not think “*Oh it crashed when I used that link, I will use a different link that works.*”
- This will go towards your iteration 1 marks.

## Integrity testing

You will not be there.

Your mentors will use your trello board and your acceptance testing part of your iteration

Did it change?

The reviewing will be recorded and the video will be available to you during week 7



## Integrity test of iteration 1

We should know why we are there, the call to action.

Are we happy that it will work for your personas (from your report)?

Is it professional, images good, is it DONE

The work should be potentially shippable.

Ready to hand to a customer, should they desire to showcase it to potential investors / customers / staff / etc.

DONE!

## Pre - acceptance testing

Integrity Testing - what's happening now

To be completed and fixed before client gets to acceptance test

This will be carried out by your mentors

(and they will look at ALL your system, they will check that it really works).

Any errors here you will fix BEFORE you let your user acceptance test!



## Also

(maybe not this time.....)

We will - Check Code / Comments /GIT,

### But we will check

Sys/Maintenance doc - In there,  
Data scripts, Database, relational model architecture

Finally does it work on my phone.

Is it reactive? Mobile / Phone Etc

## Iteration 1

- After integrity test fixes made -

Normally,

prepare for acceptance testing by your product owner. ( a refined new acceptance doc.)  
then

allow client to acceptance test / feedback.

But this week, the product owner will view what you have and may or may not ask for changes. (Clients always do!)

- Oh and don't forget AGILE = CHANGE!!!!

**On our way but with your team!**

