



UX Bootcamp Testing

David Duke

Test Lead at BookingGo
(Pricing & Supply and Frictionless)



Learning objectives

By the end of this session:

You will understand how to interact with testers to optimise the user interface.

You will have experience of constructing, interpreting and executing test plans with a focus on accessibility and usability.



Agenda

1. Why do we test?
2. How can we test?
3. How can we structure our findings?



BookingGo

Why do we test?

Why do you think we test anything at all?

Lots of types of testing:

- User Interface testing : manual and automation

- Unit testing

- Accessibility testing

- API testing

Split you into groups of four or five.

Come up with other types of testing you think people might do. Use post-its

Why do you think we test anything at all?

Lots of types of testing:

- User Interface testing : manual and automation

- Unit testing

- Contract testing

- Accessibility testing

- API testing

Why do we test any of these things?

Why do we test?

To discover information in order to answer two questions.

- Are we building the right thing?
- Are we building the thing right?



Why do we test?

- Are we building the right thing?
- Are we building the thing right?

What do I mean by this?

What does this system do?

Is it the right thing to be building?

Does it do what it's supposed to do?

Example

I want to travel across a town on my own faster than I can walk.

Two questions:

- Are we building the right thing?
- Are we building the thing right?

- 1: Wrong thing
- 2: Right thing but won't work properly
- 3: Right thing and built the right way



Example

I want to travel across a town on my own faster than I can walk.

Two questions:

- Are we building the right thing?
- Are we building the thing right?

- 1: Wrong thing
- 2: Right thing but won't work properly
- 3: Right thing and built the right way



Example

I want to travel across a town on my own faster than I can walk.

Two questions:

- Are we building the right thing?
- Are we building the thing right?

- 1: Wrong thing
- 2: Right thing but won't work properly
- 3: Right thing and built the right way



Example

I want to travel across a town on my own faster than I can walk.

Two questions:

- Are we building the right thing?
- Are we building the thing right?

- 1: Wrong thing
- 2: Right thing but won't work properly
- 3: Right thing and built the right way



Example

I want to travel across a town on my own faster than I can walk.

Two questions:

- Are we building the right thing?
- Are we building the thing right?

- 1: Wrong thing
- 2: Right thing but won't work properly
- 3: Right thing and built the right way



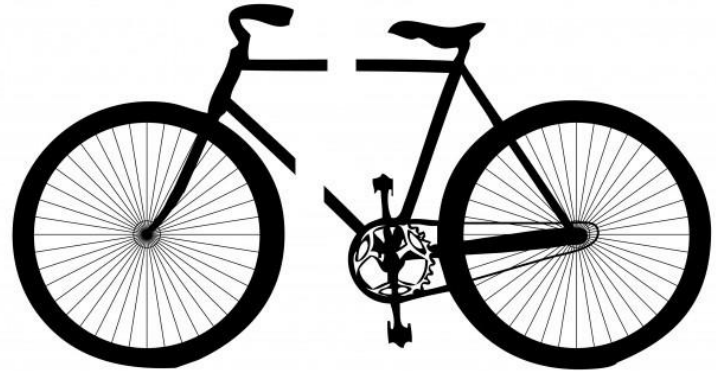
Example

I want to travel across a town on my own faster than I can walk.

Two questions:

- Are we building the right thing?
- Are we building the thing right?

- 1: Wrong thing
- 2: Right thing but won't work properly
- 3: Right thing and built the right way



Example

I want to travel across a town on my own faster than I can walk.

Two questions:

- Are we building the right thing?
- Are we building the thing right?

- 1: Wrong thing
- 2: Right thing but won't work properly
- 3: Right thing and built the right way



Example

I want to travel across a town on my own faster than I can walk.

Two questions:

- Are we building the right thing?
- Are we building the thing right?

- 1: Wrong thing
- 2: Right thing but won't work properly
- 3: Right thing and built the right way



- Are we building the right thing?
- Are we building the thing right?

Can we always answer these questions easily?

Basically you're trying to answer the question, does it work?

Functional requirements can be easy.

- Button should be green*
- Data should be saved/ updated/ deleted
- If you enter X, do Y

Non Functional Requirements can be much harder.

*(#006400 dark green)



Non functional requirements (NFR)

Usability and Accessibility are Non Functional requirements?

Do you know any other Non Functional requirements?



Why are NFRs difficult

These can be difficult to say whether things work according to these requirements because some of them are subjective.

Functional Requirement:

- I want this button to send a specific message when you click on it.

Non Functional requirement:

- I want my page to be performant: what does performant even mean, how long is okay for it to take to send a message, how many people are you expecting, do you even know?



How can we Test?

Oracles: usability/ accessibility
oracles

Test cases/ Exploratory charters

Oracles

How do we know whether things are working?

What does working even mean?

It's time for Oracles.



What is an Oracle

ISTQB (International Software Testing Qualification Board):

“Test Oracle: A source to determine expected results to compare with the actual result of the system under test.”



Why do I need Oracles if I have a spec?

- No specification
- Specification isn't all encompassing
- Conflicts with common sense
- Self Conflicting
- Conflicts with regulations
- Confusing or vague specification



Oracles - group exercise

If you were testing a website or other system, with a focus on the user interface, including how usable / accessible it is, what kind of Oracles would you potentially use?

Split into teams of 4 or 5. You have 10 minutes to come up with as many as possible oracles as you can. Write each one up on a post-it. Afterwards we will discuss what you've come up with.



Oracles - group exercise share

Okay, let's discuss the oracles you came up with.

**How do we
structure
our testing?**

Test Cases/
Exploratory Charters

What are test Cases?

ISTQB: A set of preconditions, inputs, actions, expected results and postconditions, developed based on test conditions.

These are things like

1. Go in to the <https://www.theguardian.com/uk> home page and click on the Sport heading,
Expected: it should take you to the sport url:
<https://www.theguardian.com/uk/sport>
2. Go in to the <https://www.theguardian.com/uk> home page and click on the Culture heading,
Expected: it should take you to the sport url:
<https://www.theguardian.com/uk/culture>
3. etc...

Test Cases Example

You have 15 minutes to test using this set of test cases.

<http://bit.ly/testcase1604>

Websites to use

<https://www.lonelyplanet.com/>

<https://www.viator.com/>

<https://www.musement.com>

<https://www.roughguides.com>



Test Cases - discuss

Okay, discuss the following things in groups.

1. Did you find any issues in the test cases, were they wrong in any way or confusing in any way?
2. Did you think of any more interesting things to test that weren't covered in the test cases so you weren't allowed to do them?
3. How do you think these would be maintained? Are there any problems in that?
4. What do you think might be the problems of always running the same tests every time?

What are Exploratory Charters?

Exploratory testing != Ad hoc.

Exploratory Testing is a type of testing where you design and execute the testing simultaneously.

One way of doing this is using a Charter.

Explore (a place or thing)

With (some tools)

To discover (how something works)



Sections of Charter

Goal

Oracles

Scope

Discoveries

Questions / Follow-up / Bugs /Successes

Personas

Examples of things you may have in a persona:

Age	21	40	68
Job/role	Wants to purchase the cheapest ticket.	Wants to purchase a ticket for a four person family.	Wants to purchase tickets for three different things.
Accessibility	Colourblind	ADHD	A little nearsighted
Geo Location	Germany	UK	US
Behaviour	Doesn't fill any fields unless they really have to. So will probably click submit with as little on forms before it's complete.	Lots of tabs open	Impatient. So will click buttons multiple times if they aren't given feedback that the system is responding

Exploratory Charter example

You have 30 mins to run the following Exploratory charter.

<http://bit.ly/goexplore1604>

Explore a travel website search area

With manual testing and/or plugins

To discover how accessible or usable the functionality is

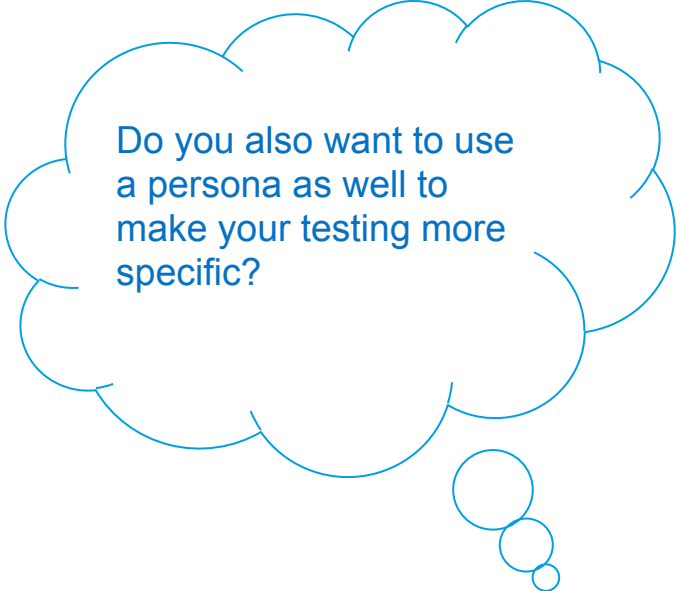
Websites to use:

<https://www.lonelyplanet.com/>

<https://www.viator.com/>

<https://www.musement.com>

<https://www.roughguides.com>



Do you also want to use a persona as well to make your testing more specific?



Share

Have a 15 minute session in groups of 4 or 5. Talk about the following things.

What did you prefer about test cases?

What did you prefer about exploratory charters?

What did you find easy about the testing?

What's the most interesting issue you found?

What was the most challenging thing about the testing?

What was the most surprising thing about doing the testing?

Can you see reasons for why you would do exploratory over test cases and the other way round?

How is testing NFR
different to functional
requirements.

More unlikely to have easy Oracles to point at
More unlikely that what you are testing for will be
present in the specification.



BookingGo

How do we structure our findings?

How should findings be structured?

Storytelling is key

Types of ways you may convey testing:

Information on what you've tested

Information on what you haven't tested

Information on what you've found

Bug reports

PEOPLE WORKing

This is from Michael Bolton's version of the course material for RST



Problem

What you've seen

I don't have cake in front of me

PEOPLE WORKing

Example

To illustrate the problem



PEOPLE WORKing



Oracle

Means by which we recognize
the problem.

What tells us it's a problem

I expected there to be cake because
you advertised there would be cake in
your advert.

PE **O**PLE WORKing



Polite

No one really likes bad news, so don't use slang or be accusatory

You said you'd bring me cake and you haven't, you incompetent loser, I hate you.

PEOP**P**LE WORKing

BookingGo



Literate

It's a story of what had
happened.

Write it like a story so that
people can understand it.

Remember that someone else
needs to understand what you
are talking about.

dont have cake expected cake. In
window saw thing said cake. Sad.

PEOPLE WORKing

Extrapolation

How general is the problem?

Could the problem be more general or less general?

Could it be more extreme or less extreme than what you have found?

Is it just on one browser or all of them?

Is it potentially indicative of a bigger issue?

PEOPLE WORKING

The person next to me has cake.

In fact everyone else has cake.

So the problem isn't wider, it's just me.

In fact I have a cup of tea so it's not affecting drinks maybe, just food?



Workaround

Is there a way to reduce the severity?

Can we just take out the feature?

Me and the person next to me could share our cake for the moment?

Can the user still do other things, it just doesn't totally work for them?

PEOPLE **WORK**ing



PEOPLE WORKing with NFRs

Is it not the same?

Well, yes and no.

Your oracle is unlikely to be your spec

It's unlikely to be black and white as to whether it matches expectations,
and more of a grey issue

You may need to use more persuasive writing techniques



You try the PEOPLE WORKing

Problem

Example

Oracle

Polite

Literate

Extrapolate

WORKaround

Using this technique

Write a bug report for a **usability** or **accessibility** issue you have found today.

Show your report to one of the mentors

You should be able to just use a text editor to type your report.



Discussion

Problem

Example

Oracle

Polite

Literate

Extrapolate

WORKaround

- Do people ever find themselves raising issues or adding to existing reports?
- Are you used to considering these points when you raise issues?
- Does anyone have anything that they think should be reported when you raise a bug which isn't listed here?



BookingGo

Wrap up



BookingGo

You will understand how to
interact with testers to optimise
the user interface.



BookingGo

You will have experience of constructing, interpreting and executing test plans with a focus on accessibility and usability.



More Sources

<https://www.istqb.org/> :International Software Testing Qualification Board

<https://glossary.istqb.org/search/oracle> : if ever testers are using words you haven't come accross, the glossary can be useful.

<https://www.developsense.com/> : Rapid Software Testing Course

Perfect Software: And Other Illusions about Testing by Gerald Weinberg

<https://www.developsense.com/blog/2012/07/few-hiccups/> <- good source of an idea about how we build up expectations.

<https://crossbrowstesting.com/exploratory-testing/personas> <- though it's advertising it's interesting

davidduke@rentalcars.com

@allcapstester