Series 40 UI design usability testing webinar - companion article

This article is companion for the *Debug your design for Series 40 Full Touch* webinars held in December 2012. It covers both sessions: 11th and 12th December 2012.

30 Dec 2012

Introduction

You know it's important to test your app before you publish it, but if you're testing only your code, you're testing only part of your app. It's important to find bugs in your design as well. Otherwise, the app may behave exactly the way your code intends, but users may still find it frustrating and give it low ratings in Nokia Store. Join UX expert Jan Krebber of Digia as he presents simple methods to find UX bugs and shows you how to correct them. These techniques can help you increase your app quality whether you work with a large organisation or completely on your own. As usual for Nokia Developer UX webinars, the presentation will feature exercises that will receive follow-up treatment in this Nokia Developer Wiki article.

The webinar and the wiki article are a great starting point if you're not a UI design expert. Check out Series 40 UI design library for more advanced information and resources.

This wiki page is the "companion" to the webinar. It includes:

- Webinar exercises and proposals for how to solve them.
- Checklist example from the webinar.
- Time plan example from the webinar.
- Task examples (pending).
- Open issues from Q/A.
- Further links and references.

Links related to this webinar:

- Slide deck link
- Recording link : Series 40 webinars: Debug your design for Series 40 full touch
- Webinar announcement

Proposals and solutions to webinar problems

This section contains problems raised in the webinar exercises. In addition, it contains proposals and possible solutions to solve the problems.

Watch the videos below. What went wrong? How do you fix it?

Shopping list

Video

The media player is loading...

Findings

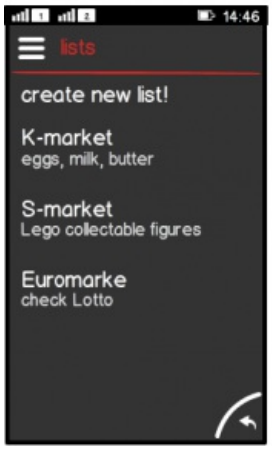
- She does not find the intended way of creating a new item via the list item.
  - She sees the text.
  - But she is reluctant pressing it.
- She states that she would have expected an "add new" button in Action Button 1 and not as list item (0:50).
- She goes via the Options menu to create a new list, which is more work than intended.
- As long as "create new list" does not have its own button, people may miss the functionality.

Proposals

- Change the list item "Create new list" to Action button #1.

(C) Copyright Nokia 2014. All rights reserved.

Current implementation



Update proposal

Purchase map

Video

The media player is loading...

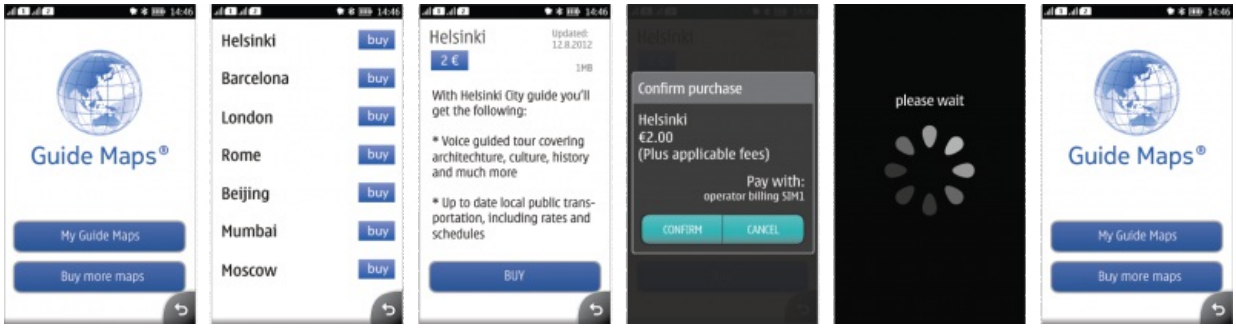
Findings

1. More information is not anticipated.
   - She is wondering from where to get more information before she buys the item.
   - Not evident from the layout that there is more information after the next step.
2. Expects to buy the item after the 2nd screen, but she does not know how much it will cost.
   - It says "buy" even though the next step still shows some information.
   - The price information is missing.
3. Buying process broken at the end.
   - She thinks the process got broken since she did not see/know where her item went.
   - Wants to start the purchase process again.

Proposals

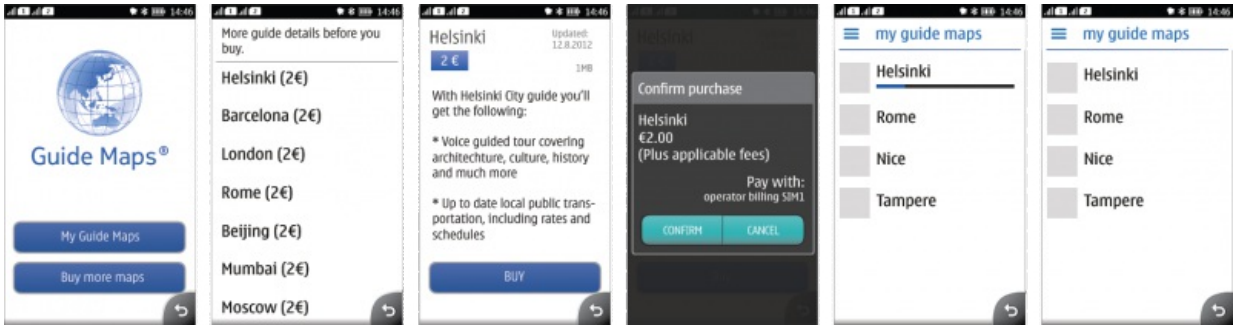
1. Inform people that there is more information after this view before they actually buy.
2. Do not show "buy" here, but mention the price at this point.
3. Buy should be reserved for the last screen before the shop UI takes over.
4. Change the purchase flow so that the user sees the map download. This could be even an "artificial download indicator" displayed just long enough for the user to see where it went, that it got downloaded.

current implementation



Update proposal

(C) Copyright Nokia 2014. All rights reserved.



When are 2-3 participants sufficient for usability testing? Why?

- If you test multiple times, e.g. in an 8 week mobile implementation phase with 4 sprints you could have 5 tests (one beforehand and one after each sprint) and 2,3,2,2,3 participants, that yields to 12 participants.
- You will not find all usability problems in the first test round, but with 12 participants you will spot the majority of your problems.
- Frequent tests will allow you to re-test some of your redesigns or tweaks.
- With more participants (4-6) you will not learn much more during one round.

You are not sure what a participant is thinking. What do you say?

- What are you thinking? :-)
- What are you doing right now?
- Is this what you expect to happen?

A participant mentions that she is not giving you what you expect from her. What do you say?

- Even if you think this is not the right information you were looking for, it can be valuable information. You must collect it and analyse later, after you have a clear mind.
- It is important to keep people talking and not to scare them away but make them feel comfortable.
- Even if the current information is completely useless for you, you must keep the participant in a comfortable mood. Maybe she will will point out some important stuff later on.
- Value participants' input, value their help, value that they give you their time.
- You must have a positive attitude about all the information you are getting from a test. The information is there to improve your application and to help you in the end to improve the Return-of-Investment (no matter if this is real money or credits).
- You say:
  - No worries, this is exactly what I am looking for.
  - I haven't looked at it from that point of view before, please tell me more about it.

A participant thinks she is doing everything wrong. What do you say to her?

- You are testing the application, not the participant, therefore the participant is not doing anything wrong.
- If you still think she is doing something wrong, you must question your preparation first - especially if this happening more often to you.
- You say:
  - No you are doing everything correct, the application is not working correctly.
  - You are doing a great job. It shows where we have to improve our application.
  - You are not the first one running into troubles here, we have to change that.
- However, it is very important that she continues so that you check your app as much as possible and improve it wherever possible.

Why not give the participant all the tasks on paper

- It is unlikely that you skip one task if you have a pile of cards and take one card after another from the pile.
- On paper you may skip a task more easy.
- Do not let the participant sneak into the next task.
  - She should concentrate on the current task.
  - The next task might reveal the answer to the current task.

How would you classify the errors you find. Which errors do you fix first?

(C) Copyright Nokia 2014. All rights reserved.

One possibility is to classify into 3 classes:

1. Immediate (must fix, if you do not fix it people will give you bad ratings and it might happen that people won't open your app a second time because of bad user experience).
2. Scheduled (should fix, many will benefit and it still affects the rating of your app).
3. At leisure (minor).

- The most urgent error must be fixed as early as possible and it must get all your attention.
- There is no use in continuing with anything else before, because if you do not fix it your app will fail.
- The biggest issue is if people get stuck in your app in a way that you (as an app author/publisher):
  - do not make money,
  - waste people's time,
  - waste people's money.

Here would be an additional list for classifying errors:

- If the majority cannot reach a goal at all, it is a must fix.
- If the minority cannot reach a goal at all, it is a should fix.
- If the majority cannot reach a primary goal conveniently, it is a must fix.
- If the minority cannot reach a primary goal conveniently, it is a should fix.
- If the majority cannot reach a secondary goal conveniently, it is a should fix.
- If the minority cannot reach a secondary goal conveniently it is a minor error.
- Typos, broken layout (in a way that you still can read everything correctly), this is a minor error.

However, this requires some knowledge of whether the mistake shown was really an exception or if it might happen with most of the people. Since you are testing with only few people in each round, this may not always be clear.

Checklist example

- Get participants (for 28.11.).
- Make schedule.
- Get helper to bring the next participant in.
- Task list & scenarios on paper.
- Prototype for all test cases.
- Pencil, eraser, paper, scissors, post-its.
- Rooms.
- Refreshments.
- Rewards.
- Agreements printed.
- Task list & scenarios on index cards.
- Call participants day before and remind (27.11.).
- Run pilot.

Time plan example

Minutes	Tasks
00	Welcome
01	Introduction ▪ My name is... ▪ The project is about booking tickets from your mobile phone for 1st and 2nd league soccer events.
02	Read and sign the agreement about the test and NDA if necessary.
05	Warm-up questions ▪ What are you doing for living? ▪ What are your hobbies? ▪ How old are you?

(C) Copyright Nokia 2014. All rights reserved.

	▪ What is your current mobile phone? ▪ Which phones did you use before that? ▪ Have you ever bought any goods (real hardware) or event tickets from your mobile phone?
10	Tasks 1. Give story 1 (start). 2. Book two tickets for the game "FC Dudes vs. Dudes United" in March. 3. Give story 2 (more info). 4. Find information about the event place. 5. Find out more about the FC Dudes team. 6. How well have Dudes United performed this year in comparison to FC Dudes? 7. Give story 3 (you go there). 8. Drive to the event place. 9. Give story 4 (you are there). 10. Show the e-tickets at the entrance. 11. Find your location in the stadium.

40	Closure
	▪ Open issues.
	  ▪ What was unclear to me?
	  ▪ Any task which should be done again?
	  ▪ Any new task I would like to see performed?
	▪ Any questions from the participant?
	▪ I wrap up.
	  ▪ What went well 1-2-3.
	  ▪ What needs improvement so that it becomes easier for the participant 1-2-3.
	▪ Give reward.
	▪ Thank you & Good bye.

50	Transition ▪ Take additional notes. ▪ Reset prototype. ▪ Refill coffee. ▪ Toilet break.

Task examples

Precondition: Test a prototype with real hardware.

No	(not for the test participant)	Task description or story given to the test participant	Notes
1.	Give story 1 (start).	You are heading home from down town and the next bus leaves in 15 minutes. This seems to be a good chance to try your new “Soccerbooker” application.	Try to be positive or neutral.
2.	Book two tickets for the game "FC Dudes vs. Dudes United" in March.	Open your phone, start the application and book 2 tickets for the game “FC Dudes vs. Dudes United” on March 14.	Give detailed information in the beginning so that people do not have to do guesswork right at the start.

(C) Copyright Nokia 2014. All rights reserved.

3.	Give story 2 (more info).	There are still more than 10 minutes left before the bus comes. You want to find out more about the additional features of “Soccerbooker”.	10 minutes are an indicator that there is still enough time left to start playing with the app.
4.	Find information about the event place.	FC Dudes built a new arena which you do not know so far. You want to know more about it.	Try to avoid wording which hints to the actual terms being used in the application. In this case it would be “arena” or “stadium” and “information”.
5.	Find out more about the FC Dudes team.	You want to know more about the FC Dudes team, e.g. how well they have performed so far and who is their current top scorer.	Try to avoid using the term “information” in task description. Place information that should be looked for on an info page. Make sure that the information is detailed enough and written clearly, even in a paper prototype.
6.	How well have Dudes United performed this year in comparison to FC Dudes?	Compare performance of the 2 teams, Dudes United and FC Dudes. Whose defense is working better?	Here the task is a bit hidden, again trying to avoid wording which is used in the UI. Defense usually relates directly to the amount of goals against one team. So they should find again certain information without knowing the exact term.
7.	Give story 3 (you go there).	Some days have passed. Today is the day of the event. You remember from the app description that it also should help you getting to the event place by car.	Clearly indicate that some time has elapsed between the last task and the next task. “Stadium” is still found from the UI so it is substituted. Even the term was revealed before, it might be a good idea not to use it. It could be that you had to skip the previous task which revealed “Stadium”.
8.	Drive to the event place.	You do not know how to get to the event place. Find the information from “Soccerbooker” app.	Now it is OK to use “information” since it will not be used later on in the UI anymore. This is one example why giving all the tasks in advance might reveal terms. In this case try to avoid “stadium” and “navigate”. If people do not know what you mean by “event place”, you have to explain that you are talking about the place where the game actually is going to take place. Still try to avoid the terms in the UI when you give additional oral descriptions.
9.	Give story 4 (you are there).	You arrived at stadium, left your car at the parking lot and walked to the entrance. On the way you bought something to eat, so you have just one hand free to operate the phone.	This description tries to take care of some context effects, like having just one hand available to operate the phone. In such a situation it is important to see if your application can be used smoothly with “thumb-only” operation; use the next two tasks to test this.
10.	Show the e-tickets at the entrance.	At the entrance you are asked to show your ticket. Show the QR code found from the application.	QR code is a hint of what they should find from the application. Again the term is not found in the UI, but in this case it is a description of the actual image they are looking for.
11.	Find your location in the stadium.	Once inside the stadium, use “Soccerbooker” to find your seat.	The task description is again a bit vague to avoid any terms of the UI.

Open issues from Q/A

No open issues left from Q/A sessions.

Further links and references

(C) Copyright Nokia 2014. All rights reserved.

- UI guidelines for Series 40 full-touch
- UI components' demo app
- Test example video by Steve Krug
- Jacob Nielsen's blog
- Rocket Surgery Made Easy: The Do-it-yourself Guide to Finding and Fixing Usability Problems. Steve Krug (2009). ISBN-13: 978-0321657299
- Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces. Carolyn Snyder (2003). ISBN-13: 978-1558608702
- Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests. Jeffrey Rubin and Dana Chisnell (2008). ISBN-13: 978-0470185483

(C) Copyright Nokia 2014. All rights reserved.