Graz University of Technology

Master's Thesis

Usability Studies with Japanese Children for Mobile Applications: Conducting a Meta Analysis

Author:
Bernd Hollerit

Supervisor:
Wolfgang Slany

A thesis submitted in fulfilment of the requirements for the degree of Master of Science

in the

Catrobat Usability Team
Institute for Software Technology

October 2013

Declaration of Authorship

I, Bernd Hollerit, declare that this thesis titled, 'Usability Studies with Japanese Children for Mobile Applications: Conducting a Meta Analysis' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: ____________________________________

Date: ____________________________________

“Creativity is contagious, pass it on.”

-Albert Einstein

GRAZ UNIVERSITY OF TECHNOLOGY

Abstract

Faculty of Computer Science
Institute for Software Technology

Master of Science

Usability Studies with Japanese Children for Mobile Applications: Conducting a Meta Analysis

by Bernd Hollerit

We examine the importance of computer programming, the pervasion of smart phones and the merits of creativity in this day and age in our society. In order to combine these aspects in a pleasant way, we propose the Catrobat toolkit to teach children the basics of programming in a playful fashion. We discuss the merits of usability and creativity extensively and decide to focus on these aspects. As the software still has room for improvement, we investigate various usability evaluation techniques before performing a Meta-Analysis consisting of Questionnaires/Surveys, Activity Analysis, Cognitive Walkthrough, Thinking Aloud, Task Analysis and Consistency Inspection. Japanese children aged 13 - 18 got familiar with our software, edited previously installed programs and created their own applications. When we observed flaws in the design and impairment of the user experience, we suggested appropriate measures for adjustment: a unified redesign of the user interface and icons, improvement of drag and drop functionality and the introduction of a "save & exit" feature. Ultimately, the Catrobat project won the Austrian national award for innovation for multimedia and e-business on March 7th, 2013. Future endeavors include a tutorial, reevaluation of our design and a global release of the Catrobat software.

Keywords
Usability, User Experience, Usability Engineering, Catrobat, Catroid, Pocket Code, Pocket Paint, Creativity, User Interface, Graphical User Interface, Design, Mobile, Mobile Application, App

ACM Classification

Software and its engineering

– Software organization and properties

– – Extra-functional properties

– – – Software usability

Acknowledgements

I would like to express my gratitude to Wolfgang Slany from the Graz University of Technology (TUG) for creating, supervising and managing the Catrobat project, introducing me to the program and working together across different countries. Kenji Tanaka and Hideaki Miyata earned my appreciation for their guidance through my year abroad at the University of Tokyo (UT) by providing me with all the support that I needed and more. Hiroko Kobayashi from the Tokyo Institute of Technology has been so kind to aid me with the translation to Japanese of the surveys used during my usability tests. My thanks also goes out to Keith Andrews (TUG) for his lectures on usability, Andreas Holzinger (TUG) for lecturing me on human-computer-interaction, Rebecca Fukuzawa (UT) for shaping my academic writing skills and Farhan Sahito (TUG) for proofreading.

Finally, I am thankful for everyone else who worked on the Catrobat team to make this project a success.

Contents

List of Figures

List of Tables

Acronyms & Abbreviations

ACM	**A**ssociation for **C**omputing **M**achinery
app	**app**lication
CEO	**C**hief **E**xecutive **O**fficer
DP	**D**ivergent **P**roduction
FFG	**F**orschungs**f**örderungs**g**esellschaft[1]
GOMS	**G**oals, **O**perator, **M**ethods, and **S**election Rules
HTML	**H**yper **T**ext **M**arkup **L**anguage
HTTP	**H**yper **T**ext **T**ransfer **P**rotocol
IDE	**I**ntegrated **D**evelopment **E**nvironment
IQ	**I**ntelligence **Q**uotient
ISO	**I**nternational **O**rganisation for **S**tandardisation[2]
IT	**I**nformation **T**echnology
LLC	**L**imited **L**iability **C**ompany
MHP	**M**odel **H**uman **P**rocessor
MIT	**M**assachusetts **I**nstitute of **T**echnology
NPO	**N**on**p**rofit **O**rganization
OS	**O**perating **S**ystem
PHP	**P**HP: **H**ypertext **P**reprocessor[3]
RITE	**R**apid **I**terative **T**esting and **E**valuation
TAP	**T**hinking **A**loud **P**rotocol
TED	**T**echnology, **E**ducation, **D**esign

[1] German: Research Promotion Agency

[2] "Because 'International Organization for Standardization' would have different acronyms in different languages (IOS in English, OIN in French for Organisation internationale de normalisation), [the] founders decided to give it the short form ISO. ISO is derived from the Greek isos, meaning equal."[International Organization for Standardization, 2013]

[3] Recursive acronym

TTCT	**T**orrance **T**ests of **C**reative **T**hinking
TUG	**T**echnische **U**niversität **G**raz[4]
URL	**U**niform **R**esource **L**ocator
UT	**U**niversity of **T**okyo
UX	**U**ser E**x**perience
XML	E**x**tensible **M**arkup **L**anguage

[4] German: Graz University of Technology

Dedicated to my parents Brigitte & Horst

Chapter 1

Introduction

"The best time to plant a tree was 20 years ago. The second best time is now."

-Chinese proverb

1.1 Introduction to Catrobat

Catrobat is a visual programming language and set of creativity tools for smartphones, tablets, and mobile browsers. Catrobat and the software developed by the Catrobat team at Graz University of Technology are inspired by the Scratch programming system developed by the Lifelong Kindergarten Group at the MIT Media Lab. The usage of the Catrobat software aims to introduce children and teenagers to programming in a playful way. This approach is endorsed by [Fisher and Williams, 2004], who recommend using Information and Communications Technology in order to boost a child's creative output.

1.2 Motivation for Computer Science

Recent research has shown that information technology (IT) is of uttermost importance to most businesses and it is not only the IT staff that needs to be familiar with it, but the majority of employees working in an office. [Marshall, 2009] Furthermore, surveys among IT executives concerning their employment strategy made it apparent that the most sought after IT skills nowadays are programming and application development. [Greenspan, 2011] This leads to the conclusion that since programming skills are one of the most desirable skills in the business world these days, children should be introduced to application development at an early age, albeit in a playful way. As cell phone usage among children is on the rise, with over one third of children aged 10-11 years old owning one [phonearena.com, 2010], mobile phones make for an ideal platform for this task. Consequently, "Catrobat" has been developed in order to spark children's interest for one of the most important business skills in recent times.

1.2.1 Importance of Information Technology

"Information technology is related to studying, designing and developing information related to computers. This field is growing at a very fast pace over the last few years and according to successful and well-known people in the IT sector, this growth is expected to remain stable. Due to the robust growth, millions of jobs have been created in this field. However, it is very essential for us to understand what the importance of information technology is." [buzzle.com, 2011]

1.2.2 Programming & Application Development

According to [Greenspan, 2011], 61% of the 351 IT executives surveyed announced plans to employ web and application developers during the next year, making programming and application development the largest year-over-year growth rate of a single IT skill set. It is expected that record numbers of mobile phone application developers will be hired over the course of this year in order to accommodate to the needs of the tremendously growing mobile device market and its users. These potential customers constantly demand innovative applications.

1.2.3 Cell Phone Usage among Children

In 2009, 1,435 people including 546 children between the ages of seven and fifteen were questioned in an online survey by Populus, with the expected result that 35% of eight year olds own a mobile phone. Moreover, three quarters of the surveyed children from the age of seven to 15 reported to own one mobile phone or more. [Adams, 2009]

1.3 Related Work

Related work includes *Scratch*[1], a programming language developed by the Massachusetts Institute of Technology (Cambridge, USA). Scratch is targeted at children, who want

[1] Scratch: http://scratch.mit.edu/

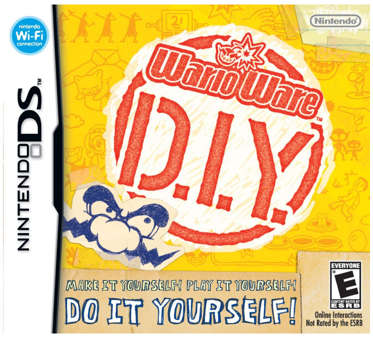
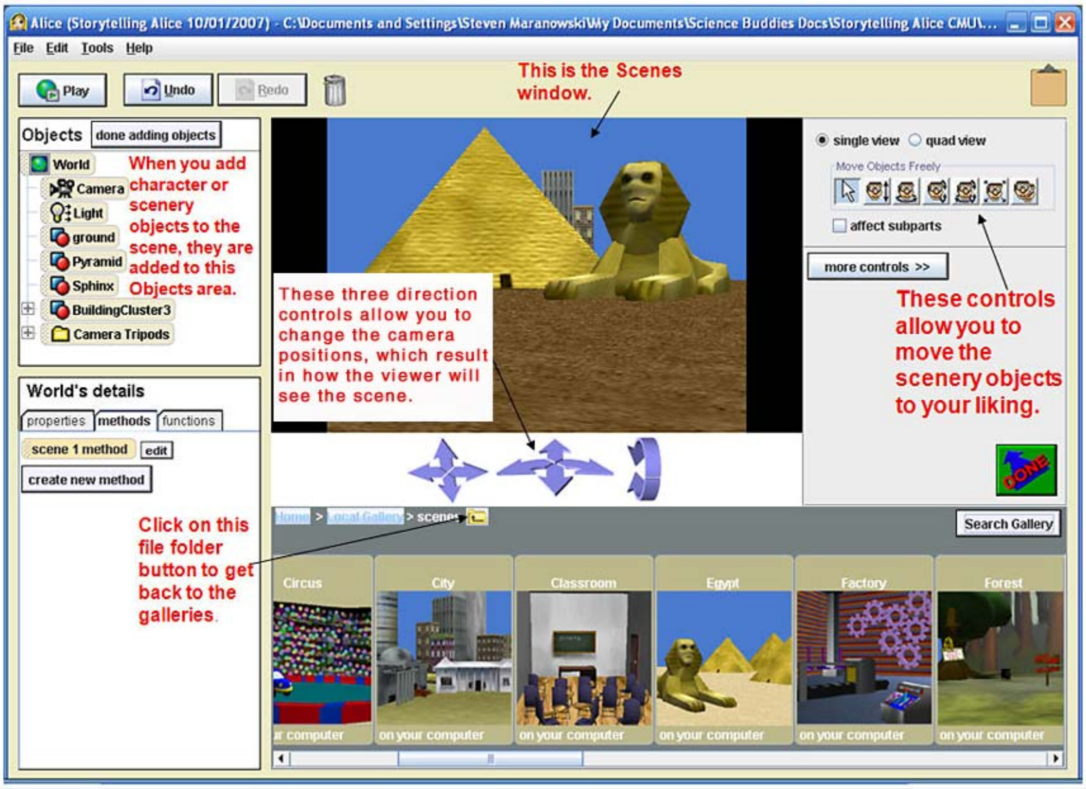

Nintendo
NINTENDO DS
Wi-Fi
WarioWare
D.I.Y.
MAKE IT YOURSELF! PLAY IT YOURSELF!
DO IT YOURSELF!
EVERYONE
E
ESRB
Online Interactions Not Rated by the ESRB

FIGURE 1.1: WarioWare D.I.Y., image taken from amazon.com

to get into programming from an early age. In an attempt to motivate girls especially, Caitlin Kelleher created *Storytelling Alice*[2], which can be used to tell interactive short stories. We can see the application in action in Figure 1.2. A playful commercial approach has been undergone by Nintendo's *WarioWare D.I.Y.*[3], where the player enjoys creating microgames with graphics and music (Figure 1.1). The upcoming *Code & Conquer*[4] encourages players to code a strategy for their army. The key to success lies in the mastery of programming and knowledge of algorithms.



This is the Scenes window.

When you add character or scenery objects to the scene, they are added to this Objects area.

These three direction controls allow you to change the camera positions, which result in how the viewer will see the scene.

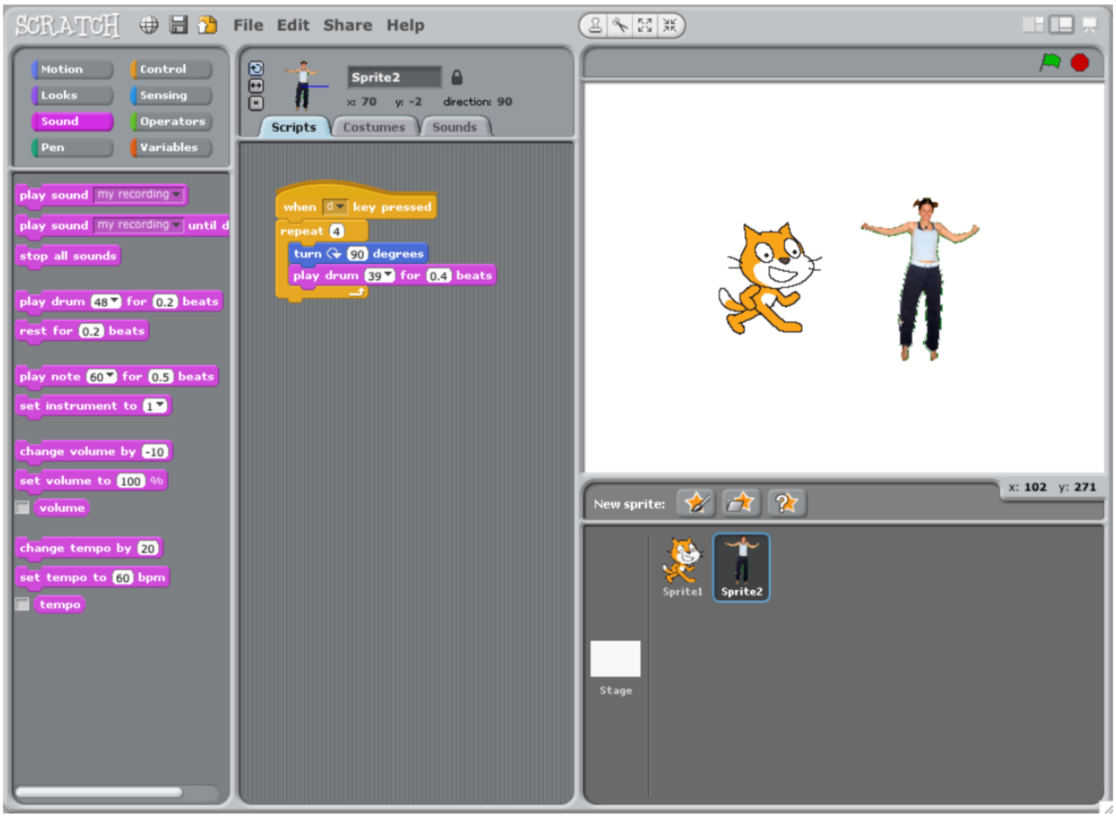
These controls allow you to move the scenery objects to your liking.

Click on this file folder button to get back to the galleries.

FIGURE 1.2: Storytelling Alice, image taken from sciencebuddies.org

[2] Storytelling Alice: http://www.alice.org/kelleher/storytelling/
[3] WarioWare D.I.Y. American website: http://www.wariowarediy.com/
[4] Code & Conquer: http://codeandconquer.co/

1.3.1 Scratch



```
when d key pressed
repeat 4
    turn 90 degrees
    play drum 39 for 0.4 beats
```

FIGURE 1.3: Scratch, image taken from info.scratch.mit.edu

Scratch (Figure 1.3) deserves special mention, because it was the original inspiration for Catrobat and the two project are closely linked together. The description on the Scratch homepage reads as follows:

"Scratch is a programming language that makes it easy to create your own interactive stories, animations, games, music, and art—and share your creations on the web. As young people create and share Scratch projects, they learn important mathematical and computational ideas, while also learning to think creatively, reason systematically, and work collaboratively. Scratch is developed by the Lifelong Kindergarten Group at the MIT Media Lab, with financial support from the National Science Foundation, Microsoft, Intel Foundation, MacArthur Foundation, Google, Iomega and MIT Media Lab research consortia" [Massachusetts Institute of Technology, 2013]

1.3.2 Related Work on the Catrobat Project

Five fellow students who worked on the Catrobat project have written their respective Master's thesis about their research. Tobias Gritschacher worked on the community website for Catrobat [Gritschacher, 2011], Ferdinand Knapitsch performed usability tests with children in Austria [Knapitsch-Scarpatetti-Unterwegen, 2012], Daniel Burtscher overlooked the continuous integration process in the open source project [Burtscher, 2012], Stefan Mayer tested the mobile HTML5 application [Mayer, 2012], and Peter Treitler researched smart self-rendering mobile user interfaces [Treitler, 2012].

1.4 Conclusion

We established that information technology and application development in this day and age are essential. Furthermore, we learned that children get in contact with mobile devices when they are still young, making them the ideal platform to playfully introduce them to the skills that will be very valuable in their lives. Motivated by these facts, Catrobat has been developed by the Graz University of Technology.

Chapter 2

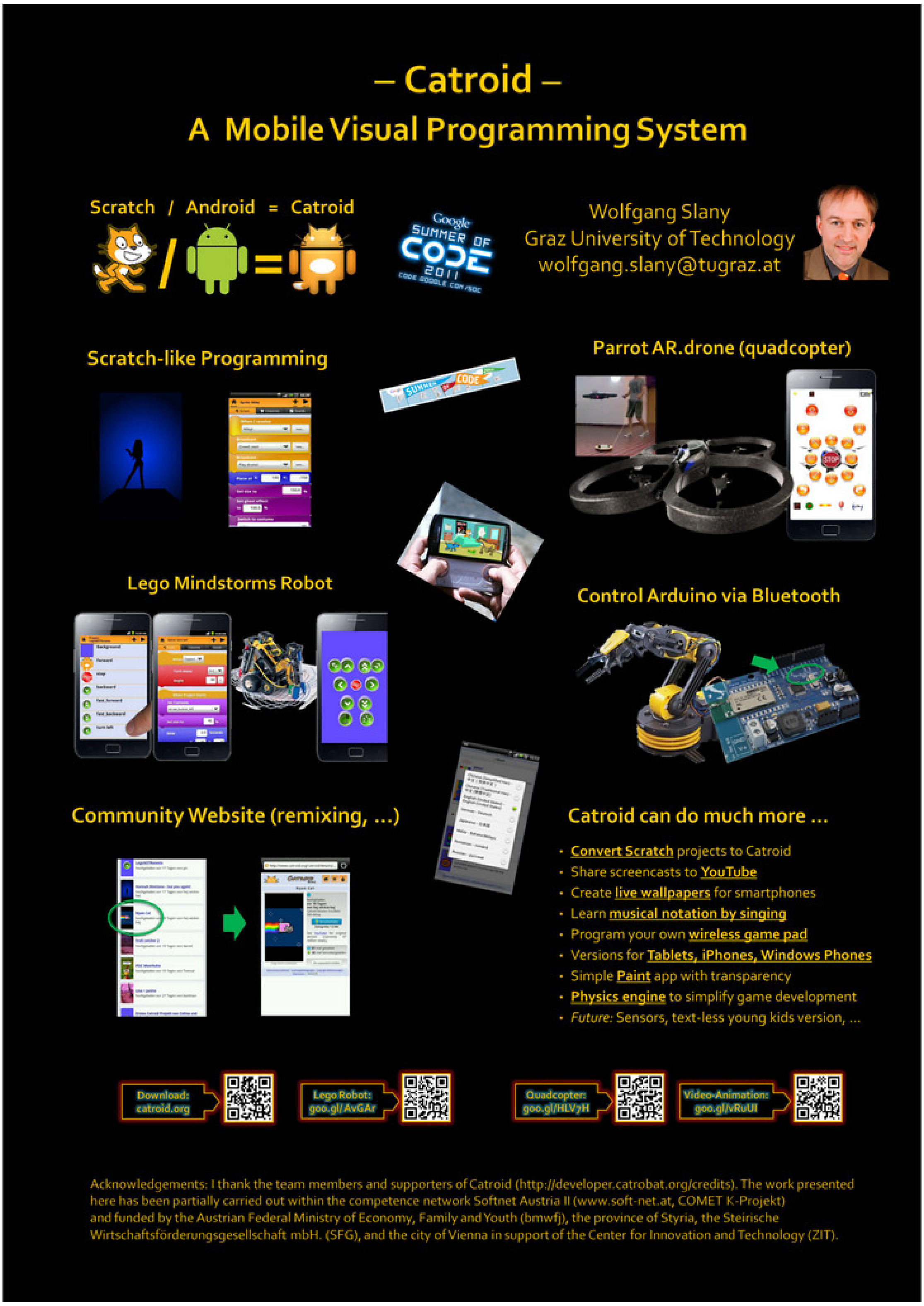
The Catrobat Project

"It is not what you do for your children, but what you have taught them to do for themselves, that will make them successful human beings."

-Ann Landers

2.1 Catrobat Overview

Catrobat is the brainchild of Professor Wolfgang Slany from the Graz University of Technology. The project aims to provide a visual programming language and set of creativity tools for children, in order to get them interested in software development, algorithms and the IT industry. Catrobat was inspired by the Scratch-environment, which was developed by the Lifelong Kindergarten Group at the Massachusetts Institute of Technology (Cambridge, USA) with a similar goal in mind: As young people create and share Scratch projects, they learn important mathematical and computational ideas, while also learning to think creatively, reason systematically, and work collaboratively. In contrast to Scratch, which is distributed for home computers, Catrobat aims to reach children who use their cell phones or tablets and different versions for Android, iOS and Windows phones are available. An impression of Catrobat's capabilities can be seen in Figure 2.1. Over one hundred students are involved in the development process of Catrobat and its subprojects.



– Catroid –

A Mobile Visual Programming System

Scratch / Android = Catroid

Wolfgang Slany
Graz University of Technology
wolfgang.slany@tugraz.at

Scratch-like Programming

Parrot AR.drone (quadcopter)

Lego Mindstorms Robot

Control Arduino via Bluetooth

Community Website (remixing, …)

Catroid can do much more …

- Convert Scratch projects to Catroid
- Share screencasts to YouTube
- Create live wallpapers for smartphones
- Learn musical notation by singing
- Program your own wireless game pad
- Versions for Tablets, iPhones, Windows Phones
- Simple Paint app with transparency
- Physics engine to simplify game development
- *Future:* Sensors, text-less young kids version, …

Download: catroid.org

Lego Robot: goo.gl/ AvGAr

Quadcopter: goo.gl/HLV7H

Video-Animation: goo.gl/vRuUI

Acknowledgements: I thank the team members and supporters of Catroid (http://developer.catrobat.org/credits). The work presented here has been partially carried out within the competence network Softnet Austria II (www.soft-net.at, COMET K-Projekt) and funded by the Austrian Federal Ministry of Economy, Family and Youth (bmwfj), the province of Styria, the Steirische Wirtschaftsförderungsgesellschaft mbH. (SFG), and the city of Vienna in support of the Center for Innovation and Technology (ZIT).

FIGURE 2.1: Catrobat Overview

2.2 Catrobat Terminology

In order to avoid confusion, we will give a brief explanation of the Catrobat subprojects and their terminology as used in this thesis in Table 2.1. Disclaimer: Recent developments led to the renaming of the Catrobat project. It will be addressed as "Pocket Code" in the future and this name will be used platform-independently. As the usability evaluation was conducted with the old terminology (the software that was installed on the cell phone was named Catroid, the questionnaires and surveys were titled "Pre Catrobat Use Survey" and "After Catrobat Usage Questionnaire" respectively), we will keep these names for the purpose of this work.

For future studies, we recommend to use the updated terms "Pocket Code" and "Pocket Paint".

Table 2.1: Catrobat Terminology

Term	**Explanation**
Catrobat	The whole project
Catroid	The IDE for the Android platform, which also includes the interpreter
Catroid Project	A Catroid program
Catroid Code	Visual code written with Catroid by a user
A Catrobat Project	A Catrobat project = a Catrobat program = a Catrobat app = a Catrobat wallpaper = Catrobat code: A project/program/app/wallpaper or visual-code written in the Catrobat language by a user (independent of Android or not)
The Catrobat Language	The visual programming language called Catrobat
Catty	iOS version of Catrobat
Paintroid	Paint editor for Catroid

2.3 Catrobat Teams

The Catrobat development is characterized by its classification into subprojects. Note that one person can be part of more than one subproject. The current composition of teams as of December 2012 can be seen in Table 2.2.

Table 2.2: Catrobat Teams

Subproject	**Number of students**
Coordinators	20 current, 8 former
Catroid (Android Version of Catrobat)	19 developers, 20 supporters
Paintroid (Integrated Image Editor)	4
Web Presentation	6
Design, Aesthetics & Skins	4 active developers and 6 supporters
Tutorial	6
Catrobat HTML 5 Edition	5
Stand Alone Catrobat Project Player and App Maker	7
Catroid Live Wallpapers	9
Arduino	6
Drone Quadcopter	3
Catroid Translation Support Community Site	1
Lego Robot	3
Usability & User Experience	5
Physics Engine	4
Sony Ericsson Xperia Play Gamepad-Keys Support	2
YouTube Recorder	3
Windows Phone Catrobat Edition	6
iOS Catrobat Edition	6
Musicdroid	4
Transcode Scratch Programs into Catrobat Programs	2 students & Professor Slany
Multilingual Media Wiki and Forum for Catrobat users for Small Screens	1
Tablet Integration	4
XML Serializer for Catroid	2
Catrobat as a Typeless Programming Language	1 student & Professor Slany
Drag & Drop in Pre-Stage	2
URL-Shortening-Service for Catrobat	1
Young Kids Version (Ages 3 to 7): Story-Telling Only Version	1

Continued on next page

Table 2.2 – continued from previous page

Subproject	Number of students
CatrobatEd: Online Resources for Educators	1
NPO Marketing for Catrobat	2 professors

2.3.1 Team Collaboration

While teams are assigned to specific tasks, their overall goal is collaboration. Teams exchange ideas in weekly meetings, give feedback to their respective team leaders and work on new solutions as required. Wolfgang Slany is the pivotal contact person when it comes to reassigning tasks and setting priorities.

2.3.2 Catrobat Usability & User Experience Team

The author is part of the Catrobat Usability & User Experience Team. While his colleagues Vesna Krnjic, Ferdinand Knapitsch, Peter Kapfer and Anja Petri focused on usability tests with Austrian children, he devoted his year abroad at the University of Tokyo to arrange and conduct usability studies with Japanese teenagers.

2.4 Internationalization

Catrobat is designed to be an international project. We believe that children are most comfortable using their native language, especially when it comes to abstract concepts like computer coding. If we consider programming languages like C++, Java or PHP, we think about for-loops, classes and functions—the syntax is in English. As we wish to reach as many children as possible, we want to confirm that our software is attractive, comprehensible and operates in any language, country and cultural area. Japan is a prime candidate for complementary field tests and usability studies, as its society notably differs in language, culture and curriculum from Catrobat's native land, Austria. Consequently, we seized the opportunity to test the waters in a foreign area.

2.5 Conclusion

The Catrobat project is an ambitious undertaking to encourage children to grapple with concepts of computer programming. About 200 people worked or are working on the software in various teams. This thesis focuses on the advancements in usability testing with Japanese children, conducive to establishing Catrobat as a successful international application.

Chapter 3

Usability Evaluation

"It is far better to adapt the technology to the user than to force the user to adapt to the technology."

-Larry Marine

3.1 Usability Overview

Usability, as defined by ISO 9241-11, is "the extent to which a product can be used by specified users to achieve specified goals with **effectiveness** (task completion by users), **efficiency** (task in time) and **satisfaction** (responded by user in term of experience) in a specified context of use (users, tasks, equipments & environments)." [International Organization for Standardization, 1998]

In his updated "Introduction to Usability" [Nielsen, 2012a], Jakob Nielsen, who has been described as "the guru of Web page usability" by the New York Times [Richtel, 1998], the "top leader in the usability field" by Webdesigner Depot [Apai, 2009] and who was ranked number 17 of the most influential people in the Telecom industry by Global Telecom Business [Global Telecoms Business, 2008], describes usability as a quality attribute, consisting of 5 quality components. These components are as follows:

- **Learnability**: How easy is it for users to accomplish basic tasks the first time they encounter the design?
- **Efficiency**: Once users have learned the design, how quickly can they perform tasks?

- **Memorability**: When users return to the design after a period of not using it, how easily can they re-establish proficiency?
- **Errors**: How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
- **Satisfaction**: How pleasant is it to use the design? [Nielsen, 2012a]

3.2 Six Usability Attributes

Keith Andrews, professor at Graz University of Technology, combines these overlapping definitions of the International Organization for Standardization and Jakob Nielsen from Section 3.1 to form the Six Usability Attributes.

1. **Effectiveness**: Completeness with which users achieve their goal.
2. **Learnability**: Ease of learning for novice users.
3. **Efficiency**: Steady-state performance of expert users.
4. **Memorability**: Ease of using system intermittently for casual users.
5. **Errors**: Error rate for minor and catastrophic errors.
6. **Satisfaction**: How satisfying a system is to use, from user's point of view. [Andrews, 2013]

3.3 Importance of Usability

Based on Jakob Nielsens Alertbox from January 4, 2012:
"If an application is difficult to use, people will stop using it. If the app fails to clearly convey what it has to offer and what users can do with it, people will stop using it. If users get lost while navigating through the app, they will stop using it. If the purpose of an app is hard to figure out or it doesn't fulfill a users' expectations, they will stop using it. Usability is a necessary condition for survival. There are plenty of other applications available; not using an app anymore is the first line of defense when users encounter a difficulty." [Nielsen, 2012a]
[Nielsen was describing websites, the author edited it to fit the medium in question, mobile applications.]

3.4 How Usability can be Improved

Improvement of usability can be achieved during the design phase and by evaluating the application. Early usability work focused mainly on human computer interaction and website design. For the purpose of this work, the author applied the established principles to mobile applications.

3.4.1 Designing for Usability

In their Communications for ACM article “Designing for usability: key principles and what designers think”, Gould and Lewis recommend three principles for designing for usability.

3.4.1.1 Three Principles of Design

We identify three principles of design that should be followed in order to reach great usability.

Early Focus on Users and Tasks
Designers need a good understanding about the target group. For that matter, they have to directly examine their anthropometric, attitudinal, behavioral and cognitive characteristics. Moreover, the expected activities while using an application have to be taken into consideration. E.g. a user of a mobile application might not always work with perfect lighting conditions, they might not be able to activate the sound of an application or they might otherwise be inconvenienced by their surroundings.

Empirical Measurement
Focus groups should test the prototypes in an early stage of the development cycle. They should perform the intended tasks and they should be observed regarding the execution of the task, their reaction and performance. These findings can be taped and analyzed.

Iterative Design
As soon as problems are made obvious during these tests, they have to be fixed. This results in an iterative process, where design, test, measure and redesign have to be repeated until the usability is satisfactory. [Gould and Lewis, 1985]

3.4.2 Evaluating Usability

We can evaluate usability with cognitive modeling methods, inspection methods, inquiry methods, testing methods and other methods.

3.4.2.1 Cognitive Modeling Methods

Parallel Design

For *Parallel Design*, multiple designer create drafts separately from each other. The results are compared, merged and if necessary, iteratively redesigned. This method can speed up time-to-market, however, resources are wasted when multiple people work on the same task, especially when most of each designer's individual effort will be discarded, reworked and merged. [Nielsen, 1996]

GOMS

The *GOMS* (**G**oals, **O**perator, **M**ethods, and **S**election rules) evaluation is concerned about the goals that a user wants to achieve and the operators (actions), which the user performs. Sequential operators that lead to a goal are called methods, and selection rules explain why a user chose a particular method rather than another. [Card, 1983]

Human Processor Model

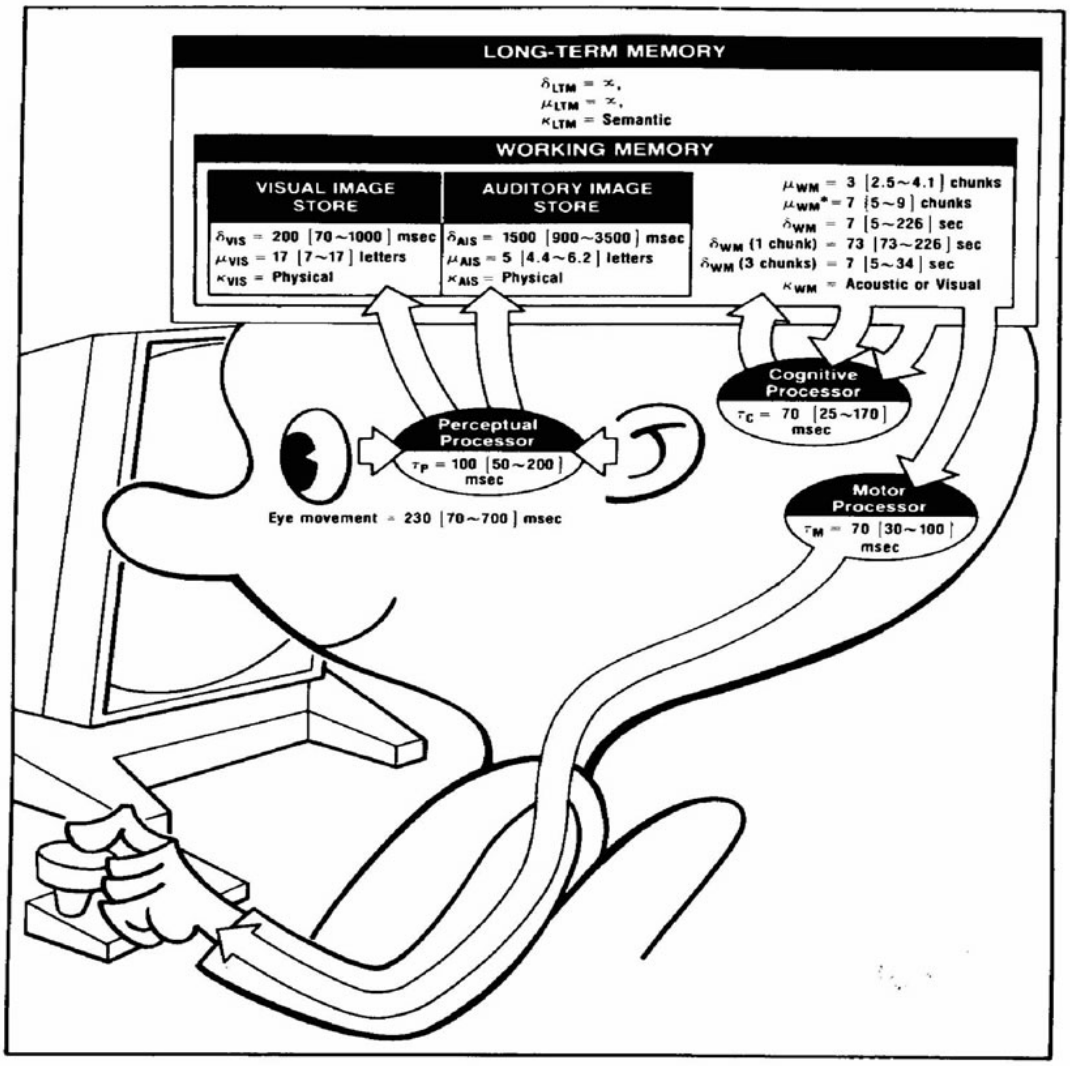
With the *Human Processor Model* or MHP (Model Human Processor), we can calculate how long a person requires to perform a certain task. In order to do this, we break down the whole activity into basic process level steps, e.g. eye movement time, memory decay time, storage capacity (pure working memory), effective working memory capacity and various cycle times (response time to brief pulse of light, time needed to differentiate fast clicks, time needed to perform a discrete micromovement). An example can be seen in Figure 3.1. If we look at the individual steps, we can identify the areas that can be improved. [Card, Moran, and Newell, 1986]

Jastrzembski et al. found significant gaps in human performance when using mobile phones between younger (median age: 18.7) and older (median age: 69.1) people using this method. [Jastrzembski and Charness, 2007]

Keystroke Level Modeling

With *Keystroke Level Modeling*, again proposed by Card et al., the keystrokes of a user who wants to perform a task are registered and the time elapsed is summed up. Keystroke level includes pushing buttons on or moving a mouse and pressing keys on a

FIGURE 3.1: *Human Processor Model*, image taken from Card's *Handbook of Perception and Human Performance* [Card, Moran, and Newell, 1986]


LONG-TERM MEMORY
δ_{LTM} = ∞,
μ_{LTM} = ∞,
κ_{LTM} = Semantic
WORKING MEMORY
VISUAL IMAGE STORE
δ_{VIS} = 200 [70~1000] msec
μ_{VIS} = 17 [7~17] letters
κ_{VIS} = Physical
AUDITORY IMAGE STORE
δ_{AIS} = 1500 [900~3500] msec
μ_{AIS} = 5 [4.4~6.2] letters
κ_{AIS} = Physical
μ_{WM} = 3 [2.5~4.1] chunks
μ_{WM}^{*} = 7 [5~9] chunks
δ_{WM} = 7 [5~226] sec
δ_{WM} (1 chunk) = 73 [73~226] sec
δ_{WM} (3 chunks) = 7 [5~34] sec
κ_{WM} = Acoustic or Visual
Cognitive Processor
τ_C = 70 [25~170] msec
Perceptual Processor
τ_P = 100 [50~200] msec
Eye movement = 230 [70~700] msec
Motor Processor
τ_M = 70 [30~100] msec

keyboard, however, more abstract activities like logging into the system are excluded. [Card, 1983] [Kieras, 1993]

3.4.2.2 Inspection methods

Card Sorts

Card Sorts can be used to determine a hierarchy and groupings of categories, usually for a website. Participants receive cards that have exactly one item written on them. Their task is to group these items into sensible categories, which are either predetermined or arbitrary. [Spencer and Garrett, 2009] E.g. the items "apple", "beef", "carrot", "orange", "pear", "pumpkin", "venison" might be grouped into the categories "fruits", "vegetables" and "meat", however, another participant could contrive the categories "main dishes" and "side dishes". The facilitator has to make a call of judgement to structure the information in a way to make it intuitively accessible for the majority of users.

Tree Tests

Also known as "Reverse Card Sorting", *Tree Tests* aid in evaluating the design of a website's categories and subcategories. A participant should find an item, e.g. "jeans under €50". We can analyze whether the user found the specific item, whether they found it directly or with backtracking, where they went astray if they couldn't find the item, how long it took them to navigate through the structure and, in conclusion, which elements of the hierarchy were intuitive and which were not. [Spencer, 2003]

Ethnography

Usability expert Tim Fidgeon describes *Ethnography* as "a qualitative research method where researchers observe and/or interact with a studys participants in their real-life environment. [...] *Ethnography* is used to support a designers deeper understanding of the design problem including the relevant domain, audience(s), processes, goals and context(s) of use". [Fidgeon, 2011]

Heuristic Evaluation

Acknowledged usability principles, also known as heuristics, are used as a "gold standard" to evaluate an interface. A group of evaluators are in charge of thoroughly probing for usability issues and, if necessary, suggest a new iteration of the design. The larger the number of evaluators, the more usability problems will be found, however, Nielsen suggests three to five usability experts, because the additional pay-off is marginal.

Similar to *Parallel Design*, each evaluator should use the system on his own before discussing and summarizing the results. [Nielsen, 1995]

Usability Inspection

Much like during *Heuristic Evaluation*, usability experts perform a *Usability Inspection.* The difference is that users have already tested the system beforehand and the problematic areas are known. The evaluators focus on these areas and deliver suggestions for a redesign.

Pluralistic Inspection

As the name implies, *Pluralistic Inspection* requires a broader sample of inspectors, ranging from the developers to the designers, from the target audience to usability experts.

This time, interaction is encouraged. The more interaction and the more pluralistic the evaluators, the more useful *Pluralistic Inspection* will prove to be.

Consistency Inspection
Consistency Inspection is concerned with multiple products, projects or platforms to ensure a consistency in the design or user interface. E.g. if the start button is located on the bottom right of the screen in the sounds menu, it should also be located on the bottom right on the looks menu and the scripts menu.

Activity Analysis
Also known as "User Observation", during an *Activity Analysis* a user uninterruptedly works with a program. An observer studies their tasks and suggests improvements to the interface. This method should be used in the earliest stages of a development cycle, when the designers are still unsure about what a user may want to do with their application.

3.4.2.3 Inquiry Methods

Inquiry methods constitute a solid method to probe users on their subjective experience of an application.

Task Analysis
Task Analysis can be linked to ethnography in the sense that we learn about the surroundings and goals. We answer the question: In order to complete a task, what does the user want or need?

Focus Groups
During *Focus Group* testing, a *Focus Group* facilitator moderates a discussion between usually five to ten participants, who are selected based on specific traits like age, gender, education, ethnicity, experience or occupation. It is a useful market research technique, but might miss out on the multifaceted attitudes and desires of the actual users.

Questionnaires/Surveys
Questionnaires and *Surveys* are an inexpensive tool to collect the user's opinions and attitudes. They need to be put together in a sensible way and given to participants who actually used the product. Open questions ensure that the user can deliver their subjective praise and criticism, which can expose the best areas of the design and the parts that still need some work.

3.4.2.4 Testing methods

Participants are probed on quantitative data and usually taped with a video camera.

Remote Usability Testing

Webservices like `http://userfeel.com` or `http://www.loop11.com` offer *Remote Usability Testing* as a means to provide profound qualitative research or collect large sample sizes of an application, usually a website. Advantages include that the participants work in their real-life environment instead of labs and that they can again be targeted, selected or filtered by demographics, e.g. age, gender or knowledge of languages.

Remote Usability Testing for Mobile Devices

We can perform *Remote Usability Testing for Mobile Devices* by providing the participants with cameras and special fixtures to keep them in place to record activities on a mobile device. This usability evaluation method is especially useful for cell phones, since screen capturing software does not provide the information how a participant handles their device and where they tap the touch screen. On the flip side, their fingers might block the view of the screen. A device integrated camera can record eye movement.

Thinking Aloud

Labeled the "single most valuable usability engineering method" and the "first tool in your UX toolbox" by Nielsen, *Thinking Aloud* follows a simple concept: Participants are presented with an application and encouraged to verbalize their thoughts while they use it. This method has the benefits of being cheap, robust, flexible, convincing and easy to learn. The administrator of a *Thinking Aloud* test doesn't need any special equipment, they just take notes as the user progresses through the tasks. Nielsen recommends up to five users and working with them should take less than a day. When participants talk during their activities, reasonable good suggestions will be found. The method can be used from early in the development cycle to fully functioning systems, its findings are convincing to even the most stubborn of designers, developers and CEOs, because the customer's thoughts are exposed without rationalizations. Finally, *Thinking Aloud* is easy to learn and doesn't require any special knowledge.

Downsides include that this method doesn't provide detailed statistics, the users are put in an artificial situation, they might impose self-censorship on their thoughts or the administrator might bias their behavior. *Thinking Aloud* is best accompanied by other usability methods. [Nielsen, 2012b]

RITE Method

Rapid Iterative Testing and Evaluation (RITE) was proposed by Medlock et al. from Microsoft Research. "Traditional" usability evaluation methods like *Thinking Aloud* or *Focus Groups* are performed with emphasis on rapid fixes of the problems that are discovered. Ideally, an issue is fixed after the first user encountered it and the next user will already be tested on the improved system. [Medlock, Wixon, Terrano, Romero, and Fulton, 2002]

Subjects-in-Tandem or Co-Discovery

If two participants explore a system together, we call it *Subjects-in-Tandem* or *Co-Discovery*. The users should communicate their actions, engage in discussions and co-operate. This can either happen naturally by encouraging them to do so, or by giving them different responsibilities and auxiliary means, e.g. one person could be in charge of a computer mouse while the other works with the keyboard.

Component-Based Usability Testing

During *Component-Based Usability Testing*, not the whole system is considered but only parts of it. This method could target specific areas of a website or a subtask performed by an application.

3.4.2.5 Other methods

In this section, we list other notable usability inspection methods that do not fall under the categories of cognitive modelling, inspection, inquiry or testing.

Rapid Prototyping

Early in the development cycle, *Rapid Prototyping* can be used as a quick and cost-effective method to design and evaluate a user-interface. Commonly, paper mock-ups are used to convey the general idea of an application. The participants are encouraged to interact with these designs. If they "click" on an option of a mock-up, they are presented with the results this action would produce: The facilitator might exchange the current paper for a design of a subcategory of the site, if this would be the result of clicking the aforementioned link.

Cognitive Walkthrough

Another method that can gather results quickly and with low cost is the *Cognitive Walkthrough*. As opposed to *Heuristic Evaluation*, where the whole system is tested, a *Cognitive Walkthrough* focuses on a specific task. It is important that the evaluator is familiar with this task to analyze how much a real user goes astray while trying to carry it out.

Benchmarking

Benchmarking compares your application to established industry standards. It works with raw values of performance, e.g. time to perform a task, time to amend errors, time spent to get familiar with an application and the scope of operations.

Persona

A *Persona* is a fictional character, who should represent the target group of an application. With the implied goals, needs and wishes of a Persona, desired user-interface elements, features or content of an application can be derived.

Meta-Analysis

Multiple combined usability evaluation methods constitute a *Meta-Analysis.* If time and resources are non-issues, a *Meta-Analysis* delivers accurate and comprehensive results.

3.5 Our Approach

For our purpose of improving the usability of a mobile phone application for children, we can outright dismiss a few of the aforementioned methods. As the Catrobat project is already an ongoing one, we focus on the evaluation of usability rather than the design for it. We can dismiss *Parallel Design* and *Rapid Prototyping*, as we are already working with a fully implemented version of an application. The *Human Processor Model* is not only too time-consuming and complex, but also addresses aspects that are irrelevant to us, e.g. eye movement time or memory decay time. *Keystroke Level Modeling* is impossible without a keyboard and applying it to a touch-screen doesn't make much sense. *Card Sorts* face the same fate as *Parallel Design*, the method is less useful after we already have a working prototype. Besides, our user interface works with only one top level menu on the start screen and the elements "scripts", "looks" and "sounds" deeper in the application, none of which require a hierarchy or grouping. The same holds true for *Tree Tests*, which are more useful to evaluate a hierarchy, which we don't have or determine the ease to find content, which the user should create on their own with Catrobat. While *Ethnography* seems tempting, a lab environment where the facilitator can monitor and document the participant's actions is preferred over the insights a real-life usage in a school or subway could deliver, in order not to strain the privacy and comfort of the teenage participants. As we don't have any heuristics or benchmarks, we can't perform a *Heuristic Evaluation* or *Benchmarking*. *Usability Inspections* don't concern us, as we don't already know any problematic areas. Time constraints and limited resources speak against *Pluralistic Inspection* and *Remote Usability Testing (for Mobile Devices)*. The *RITE Method* has to be discarded for a lack of geographical proximity, as the developers are located in Austria and the usability evaluation will be performed in Japan, making rapid fixes and re-tests impossible. *Co-Discovery* doesn't seem to make much sense on a small screen of a cellphone. *Component-Based Usability Testing* is less important to us than a holistic evaluation of the application.

This leaves us with *GOMS*, *Consistency Inspection*, *Activity Analysis*, *Task Analysis*, *Focus Groups*, *Questionnaires/Surveys*, *Thinking Aloud*, *Cognitive Walkthrough*, *Personas* and a *Meta Analysis*. To get the most out of our usability evaluation, we want to incorporate the best practices for our specific case. We are concerned about *Focus Groups* with regard to our target audience, children between the age of eight and eighteen. As for gender, ethnicity or education, we have no preferences, but as the author was studying in Japan during the time when usability tests were desired, it seemed obvious to evaluate the Japanese version of Catrobat with Japanese children. Our *Persona* in that case is a Japanese child with access to a mobile phone. For *GOMS*, we don't want to focus too much on the methods and selection rules, but we are interested in the goals a participant has and the operations they perform, the same holds true for *Task Analysis*. *Consistency Inspection* can be done by the usability professionals themselves and shouldn't be imposed on the *Focus Group*.

On the bottom line, we choose a *Meta Analysis* of a *Focus Group*, we will work with *Questionnaires/Surveys*, *Activity Analysis*, *Thinking Aloud*, *Cognitive Walkthrough* and, to a lesser extent, *GOMS* and *Task Analysis*.

3.6 How many users should you test?

According to [Krug, 2000], "the ideal number of users for each round of testing is three, or at most four. The first three users are very likely to encounter all of the most significant problems [. . .]" [Nielsen, 2000] comes to the same conclusion, proposing at most five usability tests. We followed these suggestions and compromised on performing four tests with Japanese children.

3.7 Conclusion

Usability is important and needs to be addressed in software applications. We learned about a plethora of usability evaluation methods and chose a sensible hybrid for our own purposes. As suggested by [Krug, 2000], we will test four users. These participants will be probed with *Questionnaires*, before performing a *Cognitive Walkthrough*, *Thinking Aloud* or *Task Analysis*. Afterwards, the children are endorsed to give feedback. Finally, we will perform a *Meta Analysis* and make suggestions for improvements of the application.

Chapter 4

Creativity

"My contention is that creativity now is as important in education as literacy and we should treat it with the same status."

-Ken Robinson [TED Conferences, LLC, 2006]

4.1 Creativity Overview

In this chapter, we would like to argue that creativity is an underrated, yet crucial skill—[Sternberg, 1999] even makes the statement that creativity is more important for CEOs than their personality, memory or learning abilities—and should be nurtured in an early stage of the development of an individual. We will define creativity, explain the stages of the creative process, show that creativity can be measured and present the latest findings in creativity research.

4.2 Definition of Creativity

Creativity has been defined as "the tendency to generate or recognize ideas, alternatives, or possibilities that may be useful in solving problems, communicating with others, and entertaining ourselves and others." [Franken, 2006] Another definition states that "Creativity results when the individual somehow combines [. . .] existing elements and generates some new combination." [Sawyer, 2006] Further, "all who study creativity agree that for something to be creative, it is not enough for it to be novel: it must have value, or be appropriate to the cognitive demands of the situation." [Weisberg, 1993]

According to research, traits of a creative individual cover: autonomy, flexibility, preference for complexity, openness to experience, sensitivity and playfulness, tolerance of ambiguity, risk-taking and risk tolerance, intrinsic motivation, self-efficacy and wide interest and curiosity. [Sternberg, 1999] [Villalba, 2009] [Runco, 2010] Above all, creativity deals with originality and adequacy, which begs the question whether something ought to be new-to-the-person, new-to-the-field or new-to-the-domain.

Stages of the Creative Process

The creative process can be divided into the following stages:

1. Preparation
2. Incubation
3. Insight
4. Verification
    - 4.1 Evaluation
    - 4.2 Elaboration

- *Preparation* is the initial phase of preliminary work: collecting data and information, searching for related ideas, listening to suggestions.
- *Incubation* is the delay between preparation and the moment of insight; during this time the prepared material is internally elaborated and organized.
- *Insight* is the subjective experience of having the idea--the aha or eureka moment
- *Verification* includes two substages: the evaluation of the worth of the insight, and elaboration into its complete form. [Sawyer, 2006]

4.3 Relevance of Creativity

Creativity is a valuable skill on an individual level for solving problems in daily life, as well as on a societal level, where it can lead to novel scientific findings, refreshing pieces of art, revolutionary inventions and new-found social programs. These movements in turn boost the economy, creating jobs in the service industry or new products to market and create value for a company. As such, creativity should be a much sought-after skill for CEOs. Moreover, the academic performance of students can be improved by recognizing and valuing their creativity. [Sternberg, 1999] But not only the most successful and distinguished members of society profit from creativity. Employers value flexibility, adaptability, innovation, and problem-solving-skills, all of which can be improved by a creative mind. Every activity and aspect of human endeavor can profit from creative thinking. We can change the world we live in with our own creative thinking. For this reason, educators should focus on developing children's creativity from an early age, for they will face novel problems and challenges. [Fisher and Williams, 2004]

On another note, creativity can also be fulfilling on an emotional level. Creative people love what they do. Creative engineers, chemists, writers, musicians, businesspeople, social reformers, historians, architects, sociologists and physicians concur: They do their job because it is fun in the first place. [Csikszentmihalyi, 2009]

The Flow

Psychology professor Mihaly Csikszentmihalyi describes the mental state of a person who is completely immersed in their activity as "flow". [Csikszentmihalyi, 2009] The flow is characterized by concentration, involvement, focus and indulgence of the action they perform. He identifies nine key elements for the flow to occur:

1. There are clear goals every step of the way
2. There is immediate feedback to one's actions
3. There is a balance between challenges and skills
4. Action and awareness are merged
5. Distractions are excluded from consciousness
6. There is no worry of failure
7. Self-consciousness disappears
8. The sense of time becomes distorted
9. The activity becomes an end in itself

Colloquially, the flow is also known as being in the zone, on a roll, on fire, wired in, in the moment, present, in tune, centered, singularly focused or in the groove. It is a state of mind in which many people experience their greatest joy. [Csikszentmihalyi, 2009] [Fisher and Williams, 2004] Accordingly, it can be argued from an utilitarian point of view that nurturing creativity is a means to an end in itself.

4.4 Measurement of Creativity

Recent undertakings in order to measure creativity on a large scale have been a scoreboard approach, which, due to lacking quantitative indicators, measured countries' performance in creativity and design on an array of proxy indicators. [Hollanders and Van Cruysen, 2009] The Hong Kong Creativity Index is computed in a similar way, by the outcomes of the *a*) creativity index, *b*) structural/institutional capital index, *c*) human capital index, *d*) social capital index and *e*) capital index. [Bacon-Shone and Hui, 2009] During their efforts to measure creativity, Johan Stierna and Ernesto Villalba concede that gauging scientific creativity is problematic in concept as well as empirically. Nonetheless, they propose a framework, which aids in the measurement of the environments nurturing creativity. [Stierna and Villalba, 2009]

4.4.1 Practicality of Measuring Creativity

[Cropley, 2000] compares various creativity tests and their merits. Most of these tests focus on a specific aspect of creativity, e.g. a product and its originality, relevance and usefulness; the process of creativity itself including the fluency of ideas, recognizing solutions and seeing implications; the motivation of the creative process with its desire to go beyond the conventional, risk-taking and willingness to ask (unusual) questions; or, questioning personality and/or abilities like flexibility, curiosity and independence. Real life validity of these tests is considerably lower than the soundness of IQ tests. Still, creativity tests reach high scores in their reliability, in particular inter-rater agreements of 90% and more. Internal consistencies rank around 80%, test-retest reliabilities are being measured between 60% and 80%. This leads the author to conclude that creativity tests are worth using. Cropley especially endorses the TTCT-DP[1], as he deems it suitable for people over a wide age range.

4.4.2 Implications for the Catrobat Project

These findings are relevant for the Catrobat project for mainly two reasons. Most importantly, it suggests that creativity can be measured and therefore compared. This means that we can test the creativity of children before they start to use the application as well as afterwards and evaluate if Catrobat can be seen as a creativity enabler. Further, creativity of age-groups, nationality-groups and gender-groups can be compared. Moreover, if the creativity of a person can be determined at an early age, finding and suggesting enjoyable and sensible career paths might become easier. This would hold

[1] Torrance Test of Creative Thinking - Divergent Production

true for Catrobat in particular, as children probably don't realize their own potential. Most children would deem programming languages boring and uninteresting, until they are exposed to the results that can be achieved with them (see Mitch Resnick's TEDx[2] talk: Let's teach kids to code [Resnick, 2013]). As they playfully learn the concept of programming, they might get interested. In conclusion, it would be of high value to demonstrate a link between the usage of Catrobat and an increase in creativity of the users as well as their willingness to take up programming. [Sawyer, 2006]

4.5 Fostering Creativity

Creative acts are a part of everyday life. Whether we take a short-cut on our daily commute, add details when retelling a story or perform a manoeuvre to outsmart an opponent in a video-game, we are acting creatively. We want to foster this everyday creativity and create creative capital.

Creative Capital

Three circumstances that need to work together if a person ought to perform well on a project are identified as creative capital by [Fisher and Williams, 2004].

1. The creative self: skills, commitment and talent brought to the task
2. The creative environment: the creative resources needed
3. The creative partnerships: learning partners that support one another

Every person brings their own strengths and weaknesses to the table. Some people might be more creative engaging in conversation, others are whizzes with numbers and a third group might have outstanding musical talent. The creative individual realizes that he cannot outperform everyone in every activity and rather focuses on their own fortitudes, enhancing them even further.

For the creative self to bloom, a creative environment is needed. A bookworm needs books, a mathematical genius needs a computer and formulas, a Mozart to be needs a musical instrument and sheets of music.

Lastly, creative partnerships are important. Our bookworm might want to join a book

[2] Created in the spirit of TED's mission, ideas worth spreading, the TEDx program is designed to give communities, organizations and individuals the opportunity to stimulate dialogue through TED-like experiences at the local level. TEDx events are fully planned and coordinated independently, on a community-by-community basis. [TED Conferences, LLC, 2013]

club, our arithmetic prodigy might thrive in an algebra class and someone playing a musical instrument could shine in an ensemble.

If we relate all of this to Catrobat, we find that if we expose children to computer programming at an early age, some of them are bound to develop an interest in IT. We provide the creative environment with the Catrobat software, which includes pre-set images as well as a program to draw new pictures. Sounds can be recorded and pre-existing programs are up for modification. Creative partnerships can be formed through the online community. Everyone is able to upload their programs and share them with the world. Other users might want to modify the available applications and re-upload the result. All in all, Catrobat is designed to be a re-enforcing learning environment to bolster creativity in IT.

4.6 Conclusion

We established that creativity is useful and significant skill, explained methods of how creativity can be measured and outlined how Catrobat can help to nurture the creativity in IT. In the future, we can look at implementing some of these creativity tests to demonstrate a causal link between the usage of Catrobat and the development of infant creativity in certain areas.

Chapter 5

Catrobat Usability Evaluation with Japanese Children

"If theres a trick to it, the UI is broken."

-Douglas Anderson

5.1 Catrobat Usability Evaluation with Japanese Children Overview

In order to guarantee useful and successful software, the Catrobat project employs a usability team that works closely together with the developers. While the Catrobat usability team in Graz, Austria conducted usability tests with Austrian children, the author performed the evaluation with children in Tokyo, Japan. The tests were of similar nature, with a few adjustments, e.g. the surveys were composed in English or Japanese respectively and the Catrobat software was set to English/Japanese (Figure 5.1) as well.

5.2 Catrobat Usability Evaluation Setup

The Catrobat usability evaluation consisted of three parts:

1. A *Survey* before conducting the actual test

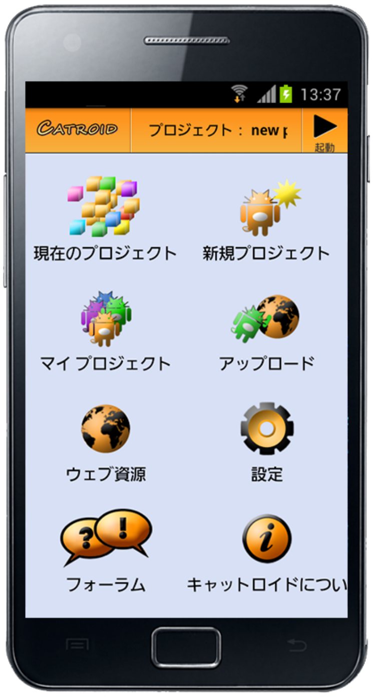


FIGURE 5.1: Japanese Version of Catrobat

2. A hands-on experience of the Catrobat software, typically lasting one hour. It consisted of a mix of *Cognitive Walkthrough*, *Thinking Aloud* and *Task Analysis*
3. A *Questionnaire* after using Catrobat

A separate *Consistency Inspection* was performed by our usability experts.

5.2.1 Pre Catrobat Use Survey

The *Survey* that can be seen in Appendix A was inspired by [Kelleher, 2006] and adapted to Catrobat. It has been translated to Japanese by Hiroko Kobayashi.

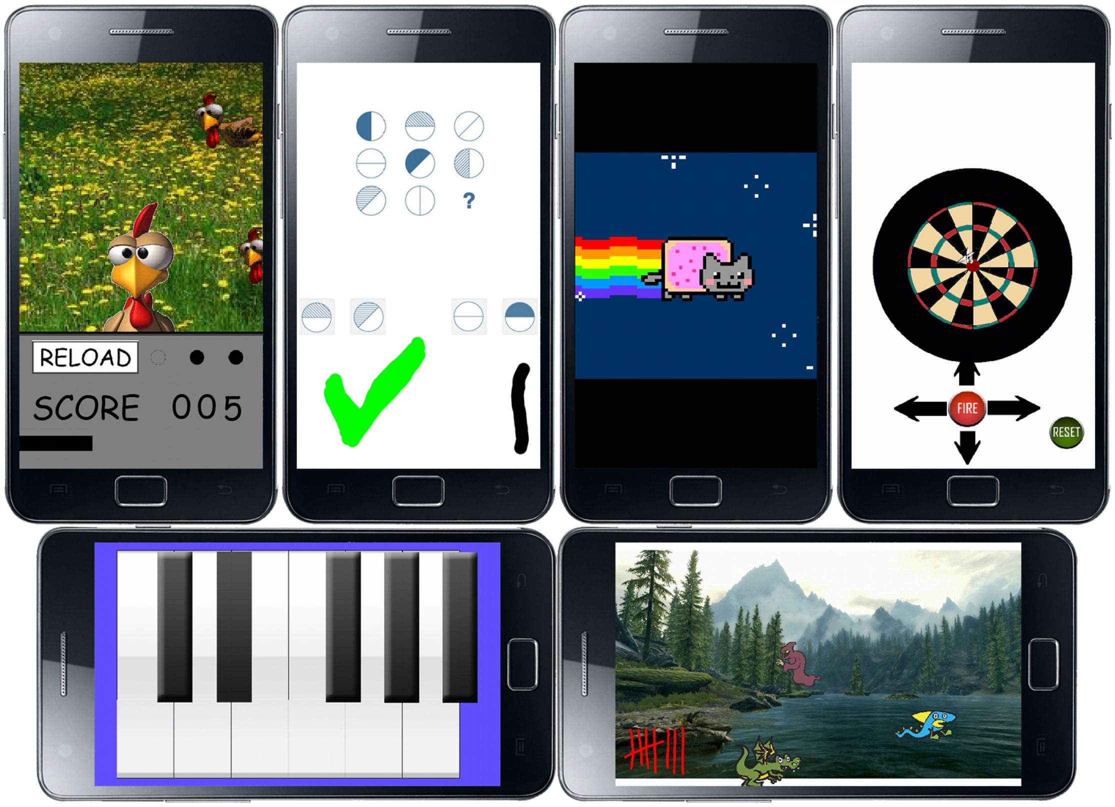
5.2.2 Catrobat Usability Evaluation

The children were instructed to perform the tasks as described in Appendix A. As the most apparent usability flaws get detected by candid usage of the program, the emphasis was on the first task, which instructed the children to use Catrobat to their hearts' content. Some usability tests just consisted of this step without any interference. Only if the child didn't know how to proceed or got bored, further specific tasks were presented.

5.2.3 After Catrobat Usage Questionnaire

The participants were presented with several statements regarding the usage of the Catrobat software. They were encouraged to give us a piece of their mind regarding their experience with Catrobat. The *Questionnaire* can be found in Appendix A.

FIGURE 5.2: Selected Pre-Installed Catrobat Apps



5.3 Catrobat Usability Evaluation Protocols

We handed the children the *Survey*. After filling out the forms, we gave them an Android device (Samsung Galaxy S2, Model SC-02C) with the Catrobat plus Paintroid software pre-installed and encouraged them to try out the program without any restrictions. Only when they were hesitant, asked questions or didn't seem to know how to proceed, we gave them further instructions. Before usability testing, we have downloaded and added user-made Catrobat applications. Six selected ones can be seen in figure 5.2.

5.3.1 Catrobat Usability Evaluation 1

The first protocol of a usability evaluation with a Japanese child using Catrobat is found in Table 5.1.

TABLE 5.1: Catrobat Usability Evaluation 1

Time	**Comment**
10:06	Starts *Survey*
10:10	Starts application Taps icons on home screen Goes back to home screen with phone's back button
10:13	Starts mole game Plays around, taps moles Ends game Looks at sprites Starts mole game again Looks at bricks Starts for the third time
10:15	Creates new program Draws in Catroid Tries Paintroid Tools
10:17	Wants to add brick, taps number field Adds purple brick Adds blue brick Adds the background he drew Starts program Taps predefined cat Very quick at trying different bricks
10:20	Looks at other programs Tries darts Tries to add brick to darts, taps brick, can't enter it Tries again, succeeds Modifies numbers Again tries to insert brick by clicking number field Succeeds Changes numbers Talks about how "it's about programming, but it's difficult to use."

Continued on next page

Table 5.1 – continued from previous page

Time	Comment
	"I don't know what's going on" I present him with the "animal"-task
10:25	Immediately uses preset cat, adds bricks so that it moves around
10:28	Draws own animal by deleting preset costume Inserts his own animal instead of preset cat When tapped, changes to other preset cat Laughs when trying his own creations Tries many blue scripts
10:33	I tell him to use only *his* animal and not the predefined cats, which were still used He changes the costume immediately and correctly
10:35	Draws new costume in Paintroid Green foundation and tree with an apple Plays around with many settings Uses it as background
10:36	I tell him that his animal should make a sound when tapped He adds sound from Gallery (some preset action music)
10:38	I tell him he can also record his own sounds He immediately uses Catroid sound recorder Records a "meow" sound for his cat Adds background music, modifies a few scripts
10:41	Finish
10:42	After usage questionnaire (thinks a lot about open ended questions)
10:51	End

5.3.1.1 Comments

This was a very fast trial and error based approach. The participant seemed to know what he was doing or corrected his mistakes instantly. He apparently enjoyed playing with the games that had already been installed on the system and was quick at following instructions to create his own program, however, he didn't come up with any ideas on his own.

5.3.1.2 Usability Suggestions

The user's workflow was interrupted twice when he tried to enter bricks. First, when he tapped the number field of a brick while he entered it, which switches to input mode instead of allowing the user to grab the brick. Secondly, he attempted to add a brick by touching it once and clicking to the desired location after, instead of dragging and dropping. The first issue can be resolved by making the number field part of the regular brick with its draggable features, the second one is debatable. If drag and drop is assumed to be well understood, letting go of the brick can be the assumed action for dropping the brick. Clicking the brick and the location it should be situated at is also a viable option. In any case, a tutorial for drag and drop actions is recommended.

5.3.2 Catrobat Usability Evaluation 2

Table 5.2 shows the course of the second usability evaluation.

TABLE 5.2: Catrobat Usability Evaluation 2

Time	**Comment**
13:01	Starts *Survey*
13:08	Starts Catroid Looks at icons Taps current project, which is preset to Moorhuhn Looks at bricks Back Again to current project Looks at scripts Back Looks at bricks again Wonders if she can touch bricks Touches brick and looks at one script Looks at costume
13:11	Opens Paintroid to draw on preset Moorhuhn background Chooses new color Chooses another color Chooses thickness of brush Doesn't know what "brush" means

Continued on next page

Table 5.2 – continued from previous page

Time	Comment
	Draws outside of boundaries and wonders why the color disappears Looks at other drawing tools Chooses cursor Again draws outside of image boundaries I tell her that she can only draw inside the image “Naruhodo!” (*“I see!”*)
13:17	“What is magic?” I explain that it fills the boundaries of a picture Uses stamp tool Cuts out half of the image Paintroid message for leaving paintroid still in english
13:18	Creates new project I set input to Japanese (phone preset is English) Ponders about the name of the project and enters it Tries to change background with paintroid Goes back Duplicates background Doesn’t know how to use the stamp Picks new color Chooses many tools: stamp, pipette, changes color, changes thickness of brush Goes back without changing anything Tries to add sound Goes back Starts program Goes back Looks at cat costumes Smiles Changes scripts for cat sprites (renames one of them) Draws on cat with paintroid Undoes change Chooses other color Taps cat
13:25	Laughs when it changes facial expression Changes order of cat costumes

Continued on next page

Table 5.2 – continued from previous page

Time	Comment
	Taps cat again and smiles Figures out that she can influence the costumes and what happens to the cat “Omoshiroi!” *(“Interesting!”)* Uses Paintroid to change cat costume Colors cat pink with magic Goes back without saving Again enters paintroid Now uses elaborate color chooser Takes long time to choose perfect color Apparantly tries to save but doesn’t seem to know how Uses new color Undo button Messages still in English, I ask to translate Uses her new picture Colors next cat costume with magic in another color Saves picture Colors third cat costume with magic in a third color
13:33	Clicks play Taps cat many times Renames her cat costume Starts her program again, taps cat Goes back, changes cat’s color again Taps add image Goes back Edits one of the cat costumes again Uses magic to color the whole image blue Tries her program Changes scripts, decreases wait time for cat to change appearance
13:38	Opens Paintroid on cat costume again, uses stamp Not sure how to use it and what to do Resizes stamps boundaries back and forth Draws on cat with brush Fills whole image with one color Uses this image

Continued on next page

Table 5.2 – continued from previous page

Time	Comment
13:40	Changes order of cat costumes in scripts Goes back to main menu Picks current project again Deletes one of the cats Starts program Goes to my programs Picks Moorhuhn Looks at scripts Looks at costumes Opens Moorhuhn costume in paintroid Uses magic on Moorhuhn image to alter background Looks at Moorhuhn sounds Starts Moorhuhn Plays game Shoots Moorhühner I need to explain to her how to reload Finishes game Looks at scripts Opens one very tiny Moorhuhn costume in Paintroid Plays game again Opens Moorhuhn costume in Paintroid Closes it, looks at scripts again Looks at background Goes to main menu Chooses Nyan Cat program Starts it
13:49	Taps cat, which meows Goes back Starts program again Pushes back button, restarts program Chooses darts Looks at scripts Starts program Throws dart Surprised that the dart doesn't land where she points it

Continued on next page

Table 5.2 – continued from previous page

Time	Comment
	Tries multiple times to hit bulls eye Succeeds Goes back Looks at all other programs Chooses whack a mole Starts Hits moles Tries to tap “Ende” (end) Nothing happens, exits program, looks at scripts Modifies parameters of scripts in whack a mole program Goes back
13:55	Chooses aquarium program Looks at scripts, then starts it Taps fish Goes back Taps fish
13:57	Looks at costume Looks at scripts Looks at background image Edits background image to another color and starts program again Goes back, chooses whack a mole program Looks at scripts Changes parameter of mole program to a shorter timer, starts program Browses through programs again
14:00	Chooses asteroids Shoots asteroids and laughs Looks at images of asteroids program, modifies laser to a blue color Starts again, laughs when her laser is blue now Browses through programs again Chooses piano Plays piano, goes back Browses through programs again
14:05	Chooses IQ test After looking at scripts, goes back Moorhuhn again

Continued on next page

Table 5.2 – continued from previous page

Time	Comment
	Looking at scripts Plays Moorhuhn
14:08	Chooses Skyrim Hunter Kills dragons Goes back Browses through programs again Chooses the one with the ghost Taps ghost, returns Chooses whack a mole again Examines scripts Changes parameter
14:15	Looks at costumes for timer Changes name of costumes Renaming them to their appropriate number from 1-10 Starts her modified version of whack a mole, which plays rather differently
14:19	Looks at mole scripts Browses through programs again Chooses egg timer Starts it, goes back Starts darts again Hits bull's eye Throws a few more darts
14:22	Browses through programs again Chooses aquarium program Goes back to main menu Browses through programs again
14:24	Chooses IQ test Starts Thinks hard about first question Gives correct answer Thinks hard about second question Correct answer Very fast correct answer for 3rd question Takes a long time for 4th question, tries to drag the question around
	Continued on next page

Table 5.2 – continued from previous page

Time	Comment
	Gives incorrect answer Thinks very hard about 5th question Doesn't seem to want to answer Answers correctly
14:30	Finishes program, looks at scripts Browses through programs again Chooses pythagoras Watches the program, exits Chooses hand tally counter Counts up, resets, exits program
14:39	Finish
14:40	After usage *Questionnaire* Finished *Questionnaire*

5.3.2.1 Comments

She took her time and enjoyed playing and modifying preset programs. She had problems with Paintroid and the save management system.

5.3.2.2 Usability Suggestions

This usability test revealed the issues young, inexperienced users could have. She was 13 years old, proclaimed to use her cell phone only for one hour on a weekday and two hours on a Saturday and Sunday. Apparently she was not familiar with usage of tools known from photo editing software, like the brush, magic and stamp. These are unintuitive concepts that should be explained in a tutorial. Moreover, parts of Catroid haven't been fully localized to Japanese, which caused misunderstandings. Especially the save management of Paintroid is tricky and a "Save & Exit-Button" is recommended. Another issue with Paintroid is that the images open on a very tiny zoom level and children try to draw outside the borders. The zoomlevel can be adjusted by the universally used pinch-to-zoom (Figure 5.3) method, however, young children might not be familiar with it yet. In addition, there is no reason not to automatically have Paintroid preset the zoom level to fill the screen.

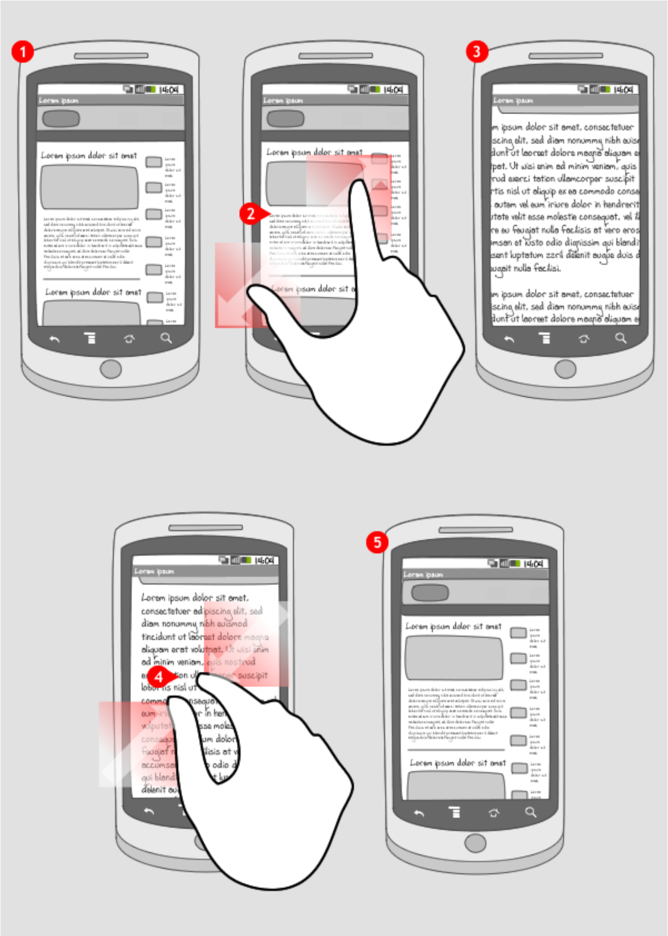


Figure 5.3: Pinch-to-zoom is used on Android-, Apple- and Windows-devices with touch interfaces, however, young children might not be familiar with it. Image taken from androidpatterns.com [UNITiD and contributors, 2013]

5.3.3 Catrobat Usability Evaluation 3

The progress of the third usability evaluation is shown in Table 5.3.

Table 5.3: Catrobat Usability Evaluation 3

Time	Comment
14:00	Start *Survey*
14:01	Start Japanese Catroid Creates new project Names it Looks at scripts Uses Paintroid to modify background Uses stamp, doesn’t know what to do with it

Continued on next page

Table 5.3 – continued from previous page

Time	Comment
	Uses magic to color everything black Draws with brush in yellow color Saves picture with save feature Names it Wants to open another picture Opens his own picture and inserts it into the existing drawing Figures out how to zoom in and out of pictures (pinch zoom) Adds more details Undoes many of his changes with undo button
14:06	Actually undoes ALL of his changes to arrive at the initial background Redoes some of the changes with redo button Draws some more Saves picture Seems not to know how to exit
14:08	Draws more Also outside of boundaries of picture Zooms in and out Undoes some changes, draws more Uses stamp tool to efficiently clone some of the stars he drew Gets rid of them again Clicks add image button Adds one of the images from my phone Saves changes with a new filename Looks at it like it's supposed to go to the main menu Draws some more on the picture Adds another picture and rotates it around
14:13	Resizes picture by pinching, sets it to the desired location Deletes some part of the picture by using transparent tool Uses stamp tool to copy some parts of picture Browses for more pictures to use
14:18	Uses some of the pictures that I used for the IQ test application Zooms in and out As he seems lost in Paintroid, I tell him how he can get out He adds a sound Adds a picture

Continued on next page

Table 5.3 – continued from previous page

Time	Comment
	Looks at blue bricks Adds a blue brick Adds a pink brick Deletes the sound he previously used Browses sounds and music Uses action hero music Goes to main menu Starts his program Taps cat Goes back Watches about screen Goes online Shortly looks at downloadable programs Goes back to his own program Changes costumes of preset cats Browses other programs
14:21	Chooses IQ test Starts it Quickly taps answers without thinking Gets 1 out of 5 correct starts it again Thinks about questions Gets first one correct Second one correct Third one correct Thinks longer about fourth question Gets it incorrect Thinks even longer about fifth question Gets it correct Restarts program Exits program Browses other programs Chooses asteroids Shoots asteroids Finishes, looks at sprites and scripts

Continued on next page

Table 5.3 – continued from previous page

Time	Comment
	Goes back Chooses Skyrim Hunter and plays it Back after 20 kills Looks at scripts Creates new project I set the input method to Japanese He gives his project a Japanese name
14:30	Changes background by adding one of his previously drawn images Uses cat picture that I have on my phone Adds sprite Undoes Starts program with his new background image Taps cat, goes back Creates new sprite, names it Adds costume Starts paintroid Draws a stick figure Uses it Adds blue script Yellow script automatically inserted Doesnt use blue script Chooses new blue script Uses it Uses purple script Starts program His picture is now in the program Looks at original cat scripts Tries to move blue script, but can't: tries to flip finger around I tell him that he has to longpress it He does it and deletes the blue script
14:35	Adds a few scripts and starts the program Draws new image with paintroid Now has two images that switch when tapped, starts to try it Inserts many new scripts When tapped, his image now gets bigger and rotates

Continued on next page

Table 5.3 – continued from previous page

Time	Comment
	Starts his program, taps his image multiple times Browses other programs Chooses the program from the 1st usability test Taps cat Exits, chooses Hannah Montana program Tries to tap it, nothing changes, goes back Plays Moorhuhn Tries shooting without reloading Figures out how to reload Ends score 15 Looks at Moorhuhn scripts Starts the application "Cogs" Tries to tap, nothing changes, exits Tries cat egg program taps cat multiple times, exits Uses hand tally counter Exits, shortly looks at scripts Plays darts Throws dart 6 times, exits, looks at script of the red button, exits Plays piano Tries hitting multiple keys at once Exits
14:45	Chooses aquarium Taps fish Taps plants Longpresses a program, rename option comes up, hits back button Opens Nyan cat app Opens Pythagoras app Looks at animation, tries to tap triangles Exits Plays asteroids again Plays Skyrim Hunter again Shortly looks at scripts
14:50	Opens binary counter Opens alarm clock

Continued on next page

Table 5.3 – continued from previous page

Time	Comment
	Opens IQ test Opens Moorhuhn Plays Moorhuhn Looks at Moorhuhn scripts Looks at Moorhuhn pictures
14:52	Uses home button to get to start screen Creates new project Modifies background to transparent Adds script Draws in Paintroid Sets brush to very thin Draws a knight as a stick figure Adds another picture
14:55	Draws limbs of knight around an explosion Adds script Adds his knight images to script Exits program, reenters it, deletes preset cat Starts program Modifies scripts by adding some more Deletes some script Moves scripts around
14:58	Starts program, his knight now explodes when tapped Adds blue script and modifies the location of the knight
15:00	Looks at asteroids scripts, starts playing it One hour of testing is up, I hand him the end *Questionnaire* Quick at filling it out, thinks a little longer about negative aspects of Catroid

5.3.3.1 Comments

He was the first to create his own program without instructions. It was a knight who exploded when tapped. He had problems with the stamp-tool, saving his images and long-press-actions, but did figure them out in the end. He played many games and examined their bricks carefully.

5.3.3.2 Usability Suggestions

New concepts like the stamp, magic, the boundaries of a picture and how to resize them need an introduction. While this participant figured them out on his own eventually, other users might become frustrated and stop using certain features or Catrobat altogether. Long-press-actions are becoming more of a standard in mobile applications, however, they still need to be explained to children. During this session, the participant tried to swipe scripts to another location instead of long-pressing them. Swiping should be used for scrolling the screen, which is why we do not suggest to redesign the interface.

5.3.4 Catrobat Usability Evaluation 4

The final participant of our usability studies also programmed a small app (Figure 5.4). The whole protocol can be seen in Table 5.4.

Table 5.4: Catrobat Usability Evaluation 4

Time	Comment
15:07	Explain what he is supposed to do, hand him *Survey* While filling out the *Survey*, he tells me (in Japanese) that he has written a small computer program in school Struggles a little bit with the English *Survey* (I was under the assumption that he speaks English rather well, unfortunately he just spoke a little bit)
15:16	Starts Catroid Opens current project (asteroids) Looks at scripts Creates new project Names it Goes back without saving, watches about screen Taps current project, still asteroids, goes to background image, goes to scripts
15:20	Looks at various sprites of asteroids Browses other programs Chooses asteroids again Clicks play Shoots asteroids Goes back

Continued on next page

Table 5.4 – continued from previous page

Time	Comment
	Browses other programs again Chooses the program his friend made Starts, taps knight, knight explodes Browses other programs
15:23	Chooses Moorhuhn Shoots Moorhühner, quickly realizes he has to reload Score 13 Browses Moorhuhn scripts Browses other programs Chooses Pythagoras, wants to tap triangles Watches animation Taps at end, when it starts again he quits the program
15:25	Chooses IQ test Looks at scripts, starts Thinks about first question Incorrect answer Thinks about second question Exits program Browsers other programs Creates new project Names it Goes to images tab, opens background image with paintroid Tries to go back, reads English message about USE/DISCARD picture Clicks add sound, catroid recorder, but goes back Draws on background image Deletes it
15:30	Opens new image and draws in light blue color Undoes changes, goes back Opens paintroid again Closes again Looks at cat scripts Looks at me questioning, I ask him if he's alright, he says “yes, it's just a little confusing” Opens other program, starts it and taps it Back to his own program

Continued on next page

Table 5.4 – continued from previous page

Time	Comment
	Draws on cat Saves picture Catroid displays "Error???"-message
15:35	Deletes all cat costumes Add picture Picks gallery first, goes back to paintroid Uses cursor Then magic to fill everything with white Changes color to black Changes thickness of cursor, but still has magic selected, so when he tries to draw the whole background changes to black Undoes Draws a face (Anpanman) Saves picture Goes back and clicks use picture Adds new costume Names it Goes back instead
15:40	Draws new picture with paintroid Sad Anpanman with bite mark Asks me if I know Anpanman, I say yes Laughs, draws another Anpanman face, now with 2 bite marks Next drawing with 3 bite marks Next drawing with 4 bite marks
15:45	Goes to scripts and uses his Anpanman pictures instead of the preset cats Looks at the scripts of his friend's program Back to his own program Adds scripts Tries his program When tapped, his Anpanman changes from 1 to 2 to 3 bite marks He tells me that he wants to change to one more image (referring to the one with 4 bite marks) I tell him that he needs a script
15:50	He adds more scripts

Continued on next page

Table 5.4 – continued from previous page

Time	Comment
	Starts program, it displays all faces in succession Changes some scripts, now the program only displays 2 faces Changes again, 4 faces again Browses through other programs
15:55	Tells me that his arm hurts from holding the cell phone I finish the usability test and hand him the *Questionnaire* I show him the Japanese translation of the *Questionnaire* to help him Takes long to answer open questions

5.3.4.1 Comments

He created the most advanced program during our usability studies, starring Anpanman[1]. If you tap the character, it gets eaten more and more every half a second, as seen in Figure 5.4. He seemed to enjoy this creative process a lot.

5.3.4.2 Usability Suggestions

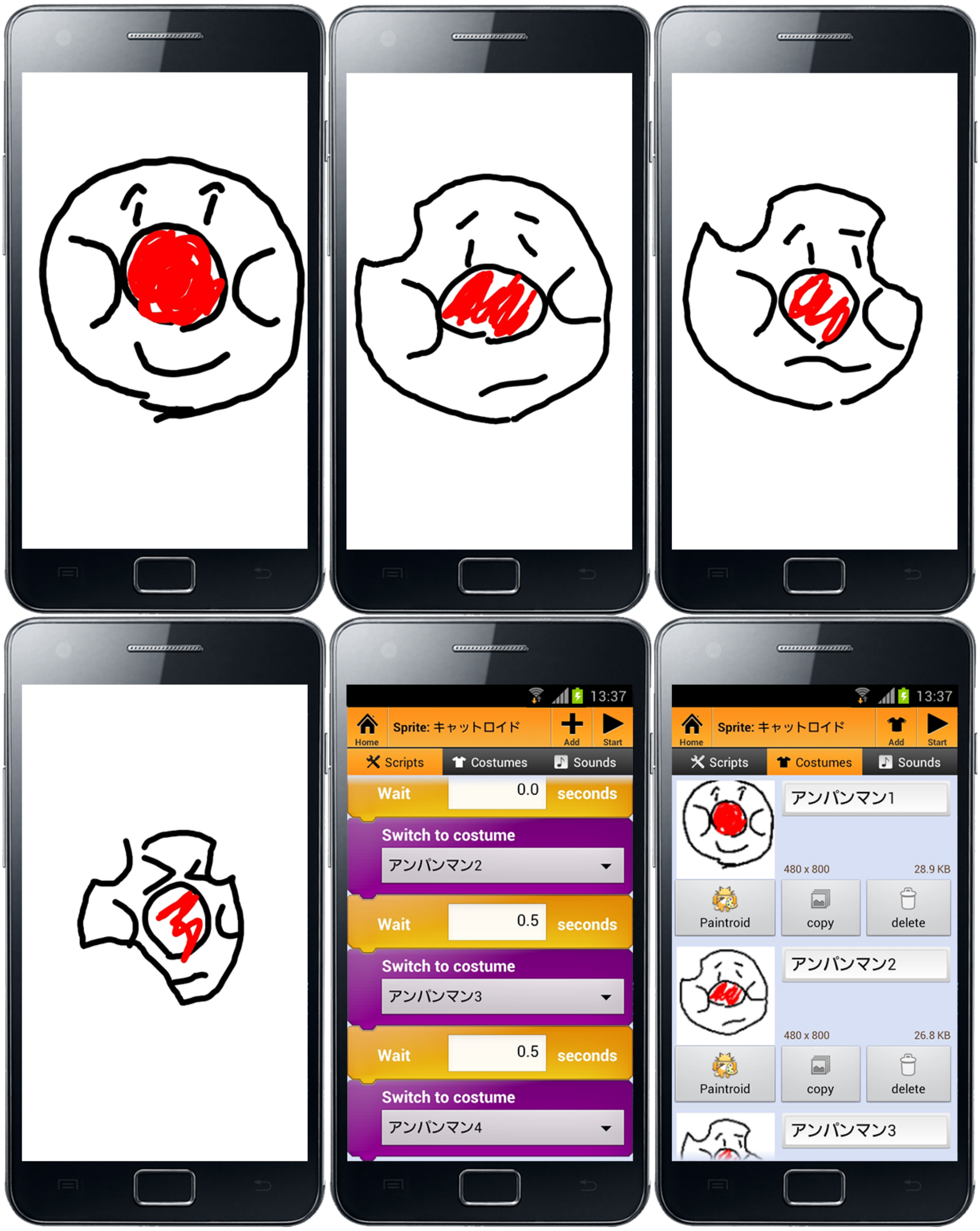
During this session, we didn't find any specific usability issues. The only persisting problem is that the overall user interface is "a little confusing". To alleviate this weakness, we suggest a uniform, consistent design.

5.4 Evaluation of Surveys

While the survey results are not statistically significant, we observe that Catrobat is perceived positively throughout. All the children agreed or strongly agreed that the usage of Catrobat is **fun**, **interesting**, **cool** and **entertaining**. They disagreed or strongly disagreed that Catrobat is **frustrating** and **boring**. One child was "not sure" about Catrobat being **annoying**, while the other children disagreed or disagreed strongly. Half of the children agreed that Catrobat was **confusing**, with the other half disagreeing. Half of the children agreed that Catrobat was **easy to learn**, with the other half being

[1] Anpanman is a popular Japanese picture book and anime series: http://anpanman.jp/index.html (Japanese)

FIGURE 5.4: Anpanman application, created during Usability Evaluation 4



Wait 0.0 seconds
Switch to costume アンパンマン2
Wait 0.5 seconds
Switch to costume アンパンマン3
Wait 0.5 seconds
Switch to costume アンパンマン4

アンパンマン1 480 x 800 28.9 KB
アンパンマン2 480 x 800 26.8 KB
アンパンマン3

not sure. In conclusion, we learn that the participants enjoyed Catrobat, however, it is a little confusing and not too easy to learn.

The other questions we were interested in the most, were how the children thought about programming languages and computer science classes. We asked the same two questions before and after the usage of Catrobat:

1. Do you think you could learn a computer language like Java or C++?
2. Would you be interested in taking a computer science class in high school?

The first question saw only one change from “probably not” to “maybe yes, maybe no”. However, the responses to the second question were just as desired. Almost uniformly the children went up one step of the ladder, from “maybe yes, maybe no” to “probably yes” and from “probably yes” to “definitely yes” (Table 5.5).

Table 5.5: Trend of childrens’ answers to the question: “Would you be interested in taking a computer science class in high school?” before and after using Catrobat

Before Catrobat usage	**After Catrobat usage**
Maybe yes, maybe no	Probably yes
Probably yes	Definitely yes
Probably yes	Probably yes
Maybe yes, maybe no	Probably yes

Finally, we take a look at the answers to the open questions about Catrobat. We asked about the best and worst things about Catrobat.

Best things about Catrobat

1. You can create your own background and character
2. You can program a motion
3. It is easy to set music and motion

1. Programming is unexpectedly easy
2. The characters that already existed are interesting
3. My sensation and interest is exciting

1. I can make whatever I want to make
2. I can show it to my friends
3. I can play with it easily

1. I could easily create something
2. I could draw freely
3. Someone else can play with what I have made

Worst things about Catrobat

1. It is a little bit confusing
2. The paint brush is thick

1. Making straight lines while drawing

1. It is difficult to get used to
2. If there were a little more explanations, it would be better

1. Easy things are easy to make, difficult things are difficult
2. When I played an easy game I gave up

The children confirm that programming with Catrobat is surprisingly easy, fun and rewarding. The social aspect of downloading other people's programs and uploading their own applications is a welcome concept. Nonetheless, Catrobat still lacks in its ease of use. An overhaul regarding the design is in order and specific functions for Paintroid were suggested.

5.5 Differences between Austrian and Japanese Children

While again not statistically significant with our sample group of four children, we noticed some differences between how Austrian and Japanese children handled Catrobat. Not only did we find different usability issues like insufficient localization or the problem that pinch-to-zoom seemed unfamiliar to one child; we also confirmed the Japanese children's captivation with animation and drawing. Japan's manga[2]-industry is estimated at $5 billion, whereas the American, European and Middle Eastern equivalent combined totals merely $370 million—a clear indicator of the pervasiveness of this subculture in the land of the rising sun. [Davidson, 2012] Our participants exhibited joy in drawing and animating their applications, leading us to the conclusion that we ought to focus on these aspects of our software for the sake of inspiring potential Japanese users.

[2] Japanese comic

5.6 Conclusion

We have performed extensive and comprehensive usability tests with Japanese children. The problems in Table 5.6 were identified. We will look into the suggested improvements and their implementation in Chapter 6.

Table 5.6: Summary of Problems

Problem
Children tap field for number input when they want to move a brick around
Children draw outside of boundaries of Paintroid image
Children don't know the Paintroid tool "magic"
Children don't know the Paintroid tool "stamp"
Children exit without saving
Images open on a very tiny zoom level
Localisation errors in Paintroid
Long-press actions aren't intuitive
Interface is generally confusing

Differences between Austrian and Japanese participants exist and we would be well-advised to focus on the drawing- and animation-features of Catrobat to attract Japanese users.

Chapter 6

Usability Suggestions and Implementations

"Questions about whether design is necessary or affordable are quite beside the point: design is inevitable. The alternative to good design is bad design, not no design at all. Everyone makes design decisions all the time without realizing it—good design is simply the result of making these decisions consciously, at the right stage, and in consultation with others as the need arise."

-Douglas Martin

6.1 Usability Suggestions Overview

The number one complaint about Catrobat was the confusing user interface, so we modernized it. Additionally, Paintroid caused a lot of confusion and was also refurbished. We expect the new interface to be more intuitive and consistent—so far, the initial reception was positive. It has yet to be evaluated if these redesigns as seen in Table 6.2 are effective.

6.2 Detailed Review of the New Design

The usability team passed their findings on to the design team. Working closely together, Catrobat has been redesigned and the progress is illustrated in Table 6.2. The top priority was to make the design consistent and less crowded to avoid confusion. The designers took care of that by giving the application a minimalistic uniform look in a distinct color.

6.2.1 Catrobat Redesign

We avoid visual overload by replacing the multicolored icons on the home screen with basic icons that are easier to process visually. The eight buttons that were placed in two vertical columns have been reduced to six options in a single column. The missing two submenus (*Settings* and *About Catroid*) and the new *Rate Us!*-option can now be accessed by clicking the cell phone's menu-button. We renamed the *Current Project*-button to *Continue*, *My Projects* to *Programs*, *Web Resources* to *Community* and dropped the *Project*-part of *New Project* and *Upload Project*. We believe that these changes make the home screen less crowded, less confusing, more intuitive and easier to process.

We slightly restructured the display of the layers. While we keep the original structure of displaying the background and the object(s), we separate them visually by introducing small headers. We are also keeping our design consistent by displaying the *Add*-button on the bottom left and the *Start*-button on the bottom right. We introduce new functionality with a *Delete*-button on the top right and we get rid of the "Project:"-description in the title.

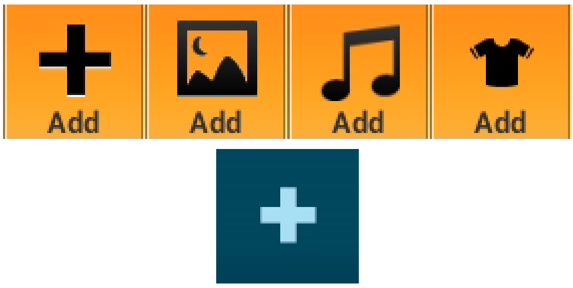
When the user is prompted for a name for their new program, we also give them the ability to write a short description of their project.

Scripts, backgrounds and sounds are now structured in a drop-down-menu instead of tabs. This leaves more place for the actual content. Again, we keep our *Add*-button on the bottom left and the *Start*-button on the bottom right of the screen for consistency. Additionally to the new *Delete*-button, we also introduce a *Copy*-button, which allows the user to copy scripts.

A look at the background-submenu immediately feels less crowded. The copy- and delete-functionality is now located on the top right, leaving extra space for more designs, scripts or sounds. Clicking the background will automatically open Paintroid (or prompt to install it), thus eliminating the need of a separate button. Again, we replaced the tabbed structure by a drop-down-menu to save space. These changes are even more obvious if we compare the *Costumes*-menu (now *Looks*-menu) with its three elements in them. Before the redesign, we could hardly fit the three elements on the screen. After the redesign, we have enough space for more than four elements, without compromising the functionality. The *Scripts*-menu loses a few pixels in height because of the introduction of the uniform *Add*- and *Start*-button, a small sacrifice we have to make in the name of consistency. However, we gain some pixels by switching from the tabbed view to the dropdown-menu and now feature the options to delete and copy.

In a *Consistency Inspection*, we found the *Add*-buttons to be lacking in uniformity. We replaced them with a single one (Table 6.1).

Table 6.1: Inconsistent *Add*-Buttons and their uniform redesign

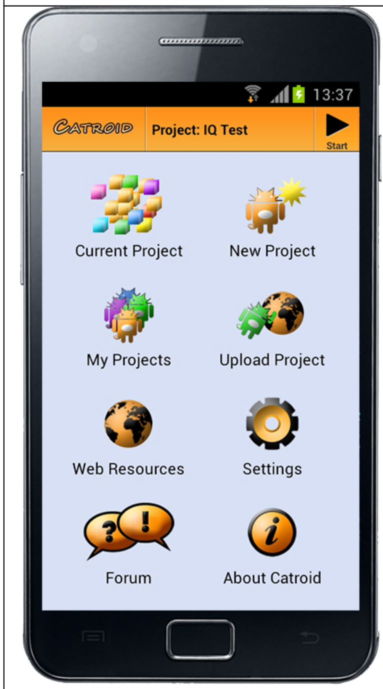


6.2.2 Paintroid Redesign

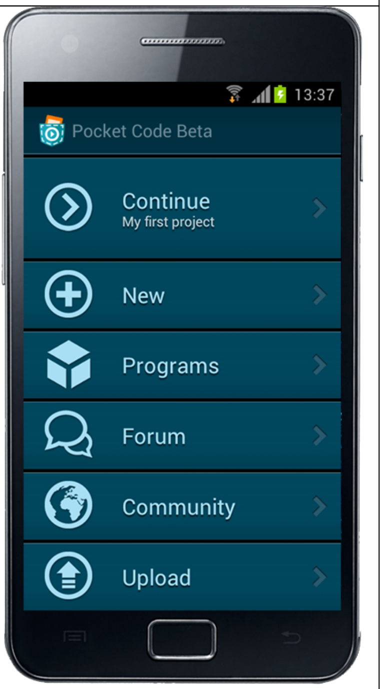
Paintroid also received a major redesign. The four prominent buttons on the bottom have been replaced by seven less intrusive buttons with four on the top and three on the bottom, respectively. We saw many children struggle with Paintroid during the usability tests and hope to alleviate the problems they had with a more intuitive design that is loosely based on existing image manipulation software. The *Undo*-button is now complemented by a *Redo*-button. This design decision has been made not because a *Redo*-button is a feature that is used very often, but because the combination is very intuitive and widely used. The third button on the top row displays the color used and can also be clicked to pick and choose a color, while the fourth button switches between *Brush*-mode and *Move*-mode. During *Brush*-mode, the user can draw with their fingers and after switching to *Move*-mode, the canvas can be dragged around and zoomed. The leftmost button on the bottom row is used to define the stroke width and shape of the tool used, the middle button on the bottom row also acts as the color chooser, but with a more recognizable design. The final button on the bottom right opens a submenu to choose from the following tools: *Brush*, *Pipette*, *Stamp*, *Import Image*, *Eraser*, *Move*, *Cursor*, *Fill*, *Rectangle*, *Crop Image*, *Flip* and *Zoom*.

Table 6.2: Catrobat Redesign after Usability Evaluation

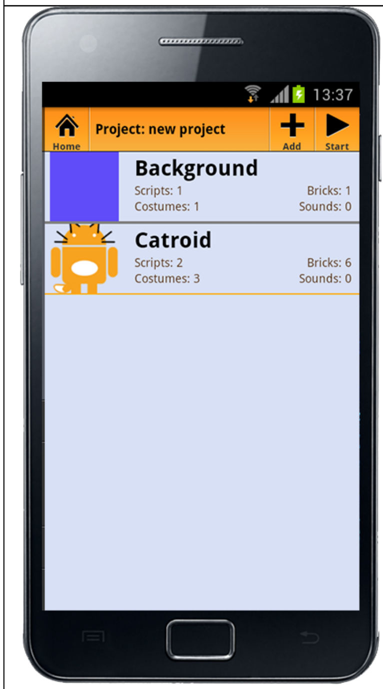
<table>
<tr><th>Before Redesign</th><th>After Redesign</th></tr>
<tr><td>



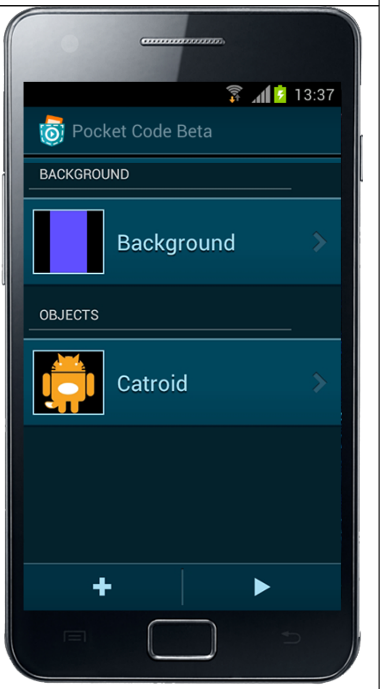
</td><td>



</td></tr>
<tr><td>



</td><td>



</td></tr>
</table>

Continued on next page

Table 6.2 – continued from previous page

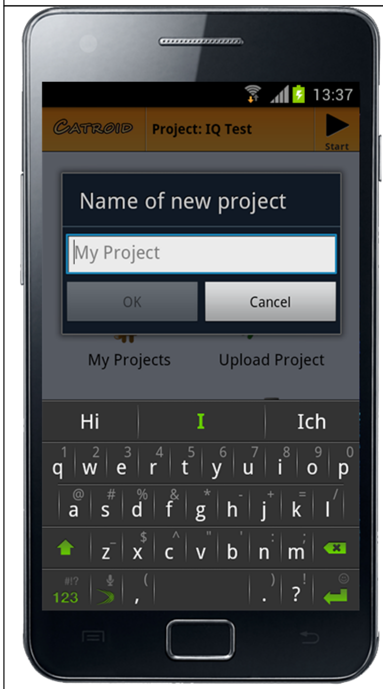
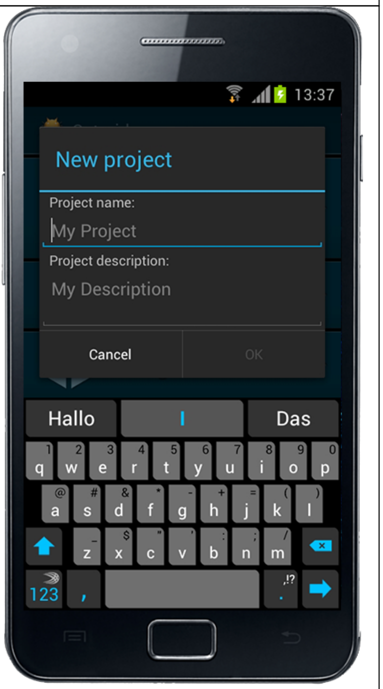
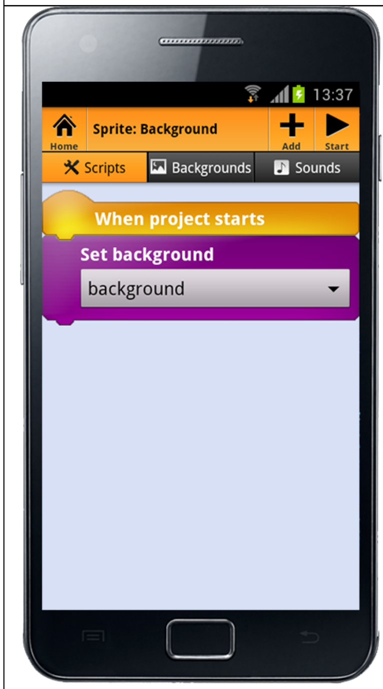
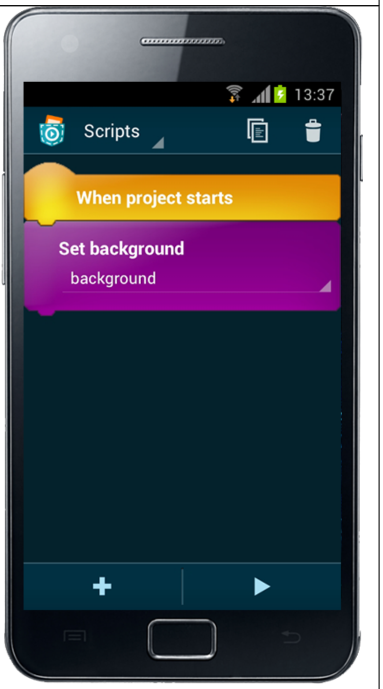
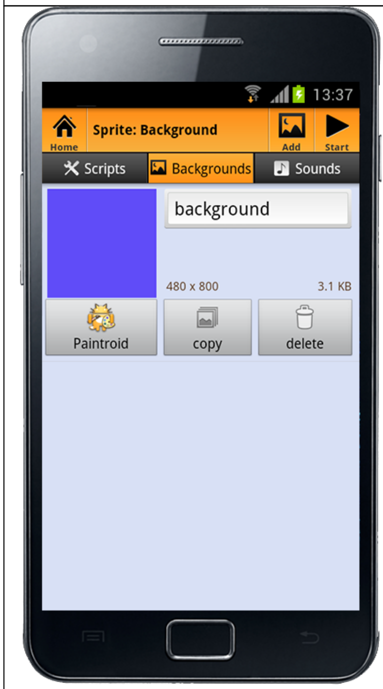
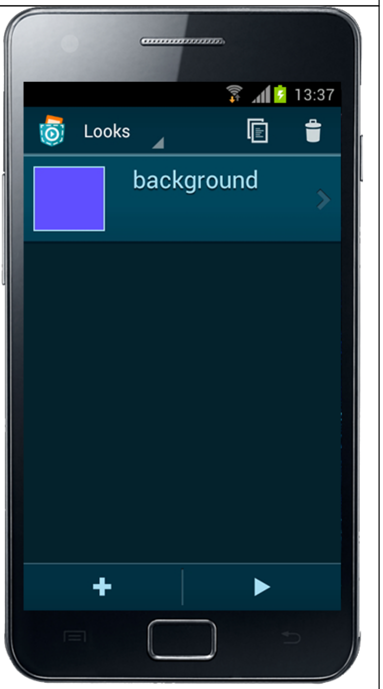
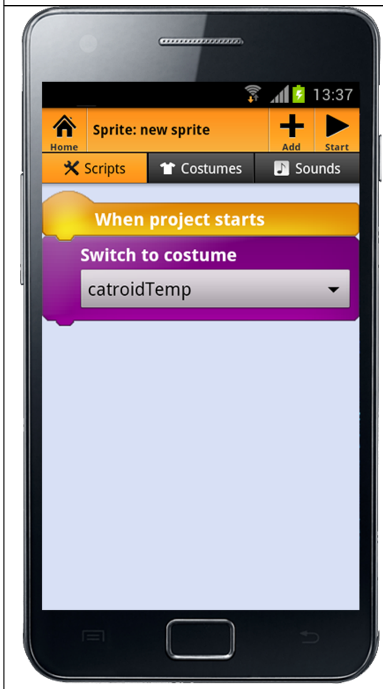
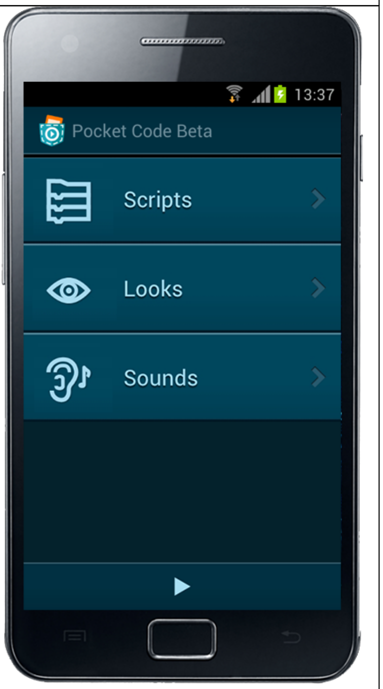
Before Redesign	After Redesign
 13:37 CATROID Project: IQ Test Start Name of new project My Project OK Cancel My Projects Upload Project Hi I Ich	 13:37 New project Project name: My Project Project description: My Description Cancel OK Hallo I Das
 13:37 Home Sprite: Background Add Start Scripts Backgrounds Sounds When project starts Set background background	 13:37 Scripts When project starts Set background background
	Continued on next page

Table 6.2 – continued from previous page

Before Redesign	After Redesign
 13:37 Home Sprite: Background Add Start Scripts Backgrounds Sounds background 480 x 800 3.1 KB Paintroid copy delete	
 13:37 Home Sprite: new sprite Add Start Scripts Costumes Sounds When project starts Switch to costume catroidTemp	

Continued on next page

Table 6.2 – continued from previous page

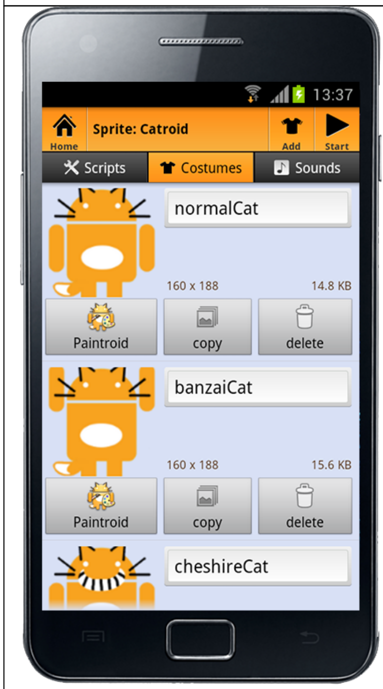
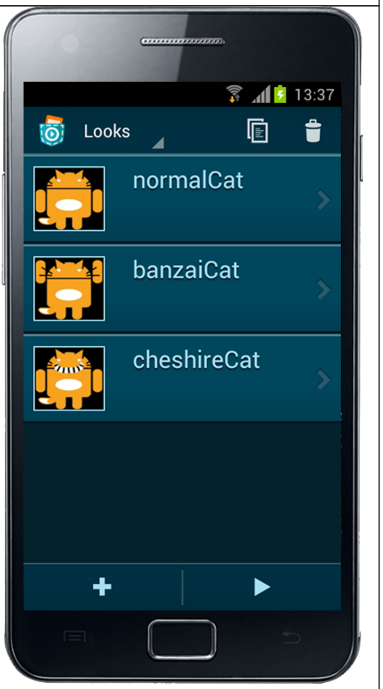
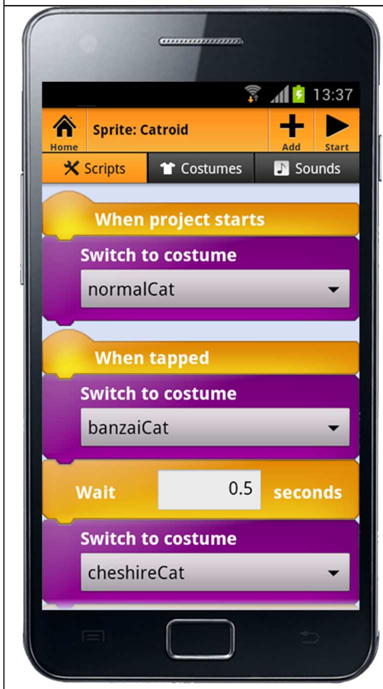
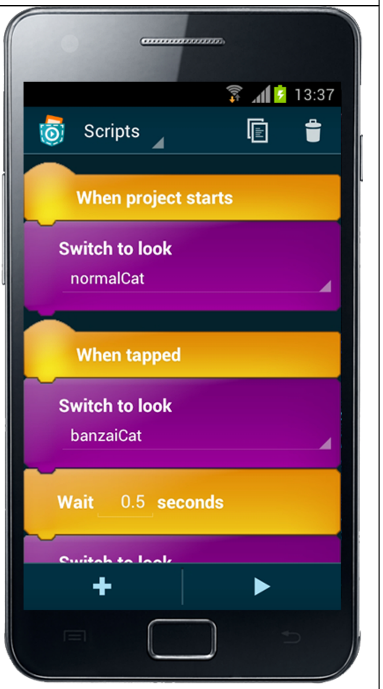
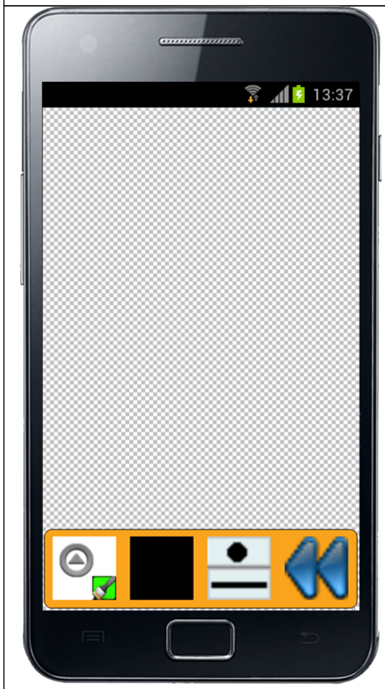
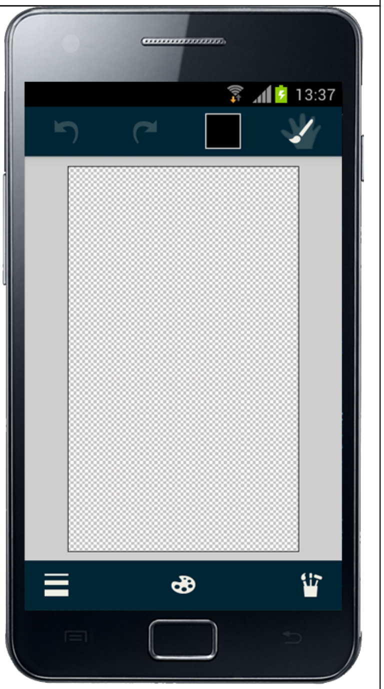
<table>
<tr><th>Before Redesign</th><th>After Redesign</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td colspan="2">Continued on next page</td></tr>
</table>

Table 6.2 – continued from previous page

Before Redesign	After Redesign
	

6.3 Suggestions for the Japanese Market

The overhaul of Catrobat's usability was a much needed necessity and the beginning of a longer journey. We need to reevaluate the program for usability, preferably in many iterations and with a plurality of participants. As for the Japanese market, we ought to focus on features regarding drawing and animation, as suggested in Chapter 5. This could be achieved by a multitude of preset icons, images and animations, or a layer-system as introduced in WarioWare D.I.Y.—a user might draw on a model or pattern. After saving, only the user-generated content remains.

6.4 Conclusion

We have performed usability tests, encountered problems, suggested a redesign and discussed said overhaul of the Catrobat software. We are confident that the new iterations constitute a vast improvement over the outdated versions of Catrobat and Paintroid. In the future, more usability testing will be needed to prove our intuitions and, where appropriate, perform an iterative redesign.

Chapter 7

Conclusion & Outlook

"However beautiful the strategy, you should occasionally look at the results."

-Winston Churchill

7.1 Conclusion

We have discussed, prepared, conducted and reviewed usability evaluation in order to improve the user experience of the Catrobat (now "Pocket Code") software. For our particular case, we deemed a *Meta-Analysis* consisting of *Questionnaires/Surveys*, *Activity Analysis*, *Cognitive Walkthrough*, *Thinking Aloud*, *Task Analysis* and a *Consistency Inspection* as the most useful. These usability evaluation methods were performed with Japanese children, compared with usability tests that were conducted with Austrian children and fine-tuned by our usability experts. Suggestions for improvements were made and applied. The usability team worked closely with the programmers to ensure a positive outcome. For our efforts, the Catrobat project received the Austrian national award for innovation for multimedia and e-business on March 7th, 2013 (Figure 7.1).

Drawbacks of our strategy included language barriers, e.g. only partially localized software or *Questionnaires* in English instead of the children's native language, Japanese. *Thinking Aloud* was limited by these constraints and we settled for a *Cognitive Walkthrough* or *Task Analysis.* Testing four participants was sufficient, however, multiple rounds of usability evaluation after a redesign are recommended.

Figure 7.1: Catrobat is the winner of the Austrian national award for innovation for multimedia and e-business. From left to right: Henrietta Egerth (FFG), Anton Rieder, Christof Stromberger, Bernd Hollerit, Wolfgang Slany, Jeton Arifi. Photographer: Matthias Silveri

7.2 Outlook

Further projects include the reevaluation of the redesign and the assessment of actual user opinions after the software hits the market. We would like to distribute the Catrobat software to schoolchildren worldwide and evaluate their experience. We are optimistic that our software will be received positively and contribute to a better education of children.

For further information, some useful websites monitor the progress of Catrobat, Paintroid and Pocket Code:

- `http://catrobat.org` – Developer website hosting the latest builds of Pocket Code and Pocket Paint
- `https://pocketcode.org` – Hosts user-created apps that can be downloaded and run by any Pocket Code version

- https://play.google.com/store/apps/details?id=org.catrobat.catroid – Beta version of Pocket Code, available for Android devices at the Google Play Store
- https://play.google.com/store/apps/details?id=org.catrobat.paintroid – Latest version of Pocket Paint, available for Android devices at the Google Play Store
- http://blog.catroid.org – Catrobat Blog
- https://plus.google.com/communities/116816272940643231129 – Catrobat Community on Google+
- https://github.com/Catrobat – Catrobat GitHub Project

Appendix A

Catrobat Questionnaires

"Who questions much, shall learn much, and retain much."

-Francis Bacon

A.1 Pre Catrobat Use Survey

Name: ____________________________

Age: _____

Grade in School: ____________________________

What kind of school do you go to? (Please circle one answer)

(a) a public school

(b) a private school

(c) I am home-schooled

What are your favorite subjects in school? (Please circle all that apply)

(a) English

(b) History

(c) Math

(d) Science

(e) Foreign Language

(f) Government

(g) Art

(h) Music

(i) Other: ____________________________

How many hours on a weekday (from Monday until Friday) do you use your mobile phone for any purpose?
Hours on a weekday _____

How many hours on a weekend (from Saturday and Sunday) do you use your mobile phone for any purpose?
Hours on the weekend _____

What do you use phones for?

(a) Only for messaging and phone calls

(b) Mostly for messaging and phone calls, some for fun

(c) About equally for messaging and phone calls and fun

(d) Mostly for fun and some for messaging and phone calls

(e) Only for fun

What is your skill level at using mobile phones?

(a) Poor or nonexistent

(b) Fair

(c) Good

(d) Very good

(e) Excellent

Which kind of computer do you use?

(a) I have my own computer

(b) I use my family's computer

(c) I use a computer at school

(d) I use a public computer

(e) I don't use computers

What is your skill level at using computers?

(a) Poor or nonexistent

(b) Fair

(c) Good

(d) Very good

(e) Excellent

Have you ever written a computer program?

(a) Yes

(b) No

(c) Don't know

Have you ever made your own web page?

(a) Yes

(b) No

(c) Don't know

When something goes wrong with your computer, how frequently do you ask friends or family members for help fixing it?

(a) Very frequently

(b) Somewhat frequently

(c) Neither frequently nor infrequently

(d) Somewhat infrequently

(e) Very infrequently

When you want to install a new computer program, how frequently do you ask friends or family members to help you install it?

(a) Very frequently

(b) Somewhat frequently

(c) Neither frequently nor infrequently

(d) Somewhat infrequently

(e) Very infrequently

Do you think you could learn a computer language like Java or C++?

(a) Definitely not

(b) Probably not

(c) Maybe yes, maybe no

(d) Probably yes

(e) Definitely yes

Would you be interested in taking a computer science class in high school?

(a) Definitely not

(b) Probably not

(c) Maybe yes, maybe no

(d) Probably yes

(e) Definitely yes

A.2 Tasks

Program an animal (45-50 min) The animal should move around and make a sound when you tap it. A background image is inserted and assigned, a new object created or the existing one replaced. A sound is added. The images will be prepared and can be found in the gallery (maybe the children can draw the animal). The sound will be recorded with the Catroid Sound Recorder.

Unsupervised Test
Shortcut - Task (estimated time)

1. Project
   (a) CP Create project and name it (2 min.)
   (b) SP Start program (1 min.)
   (c) UP Upload program (2 min.)
2. Objects
   (a) CO Create object and name it (2 min.)
   (b) DO Delete object (2 min.)
   (c) RO Rename object (2 min.)
3. Bricks (scripts)
   (a) IB Insert brick (script) (5 min.)
      - New brick
      - Search for brick
      - Choose brick

   (b) DB Delete brick (script) (2 min.)
   (c) MB Move brick (script) (2 min.)
4. Costume
   (a) IC Insert costume (7 min.)
      - Start gallery and choose costume
      - Start Paintroid, draw, return to Paintroid

   (b) RC Rename Costume (2 min.)
   (c) CC Copy costume (1 min.)
   (d) DC Delete costume (1 min.)

   (e) AC Assign costume (3 min.)

5. Sound

   (a) IS Insert sound (5 min.)

      - New sound
      - Choose sound
      - Catroid Sound recorder
      - Music Player List

   (b) RS Rename sound (1 min.)

   (c) CS Copy sound (1 min.)

   (d) DS Delete sound (1 min.)

   (e) AS Assign sound (3 min.)

Supervised Test **(42 min.)**

1. Get to know the app and first impressions (10 min.)

   - You can try the app for a few minutes. Try everything and anything.

2. Create new project and start it (2 min.)

   (a) Create a new project

   (b) Name the project

   (c) Start the project

3. Create an animal (15 min.)

   - Find out how to create an animal

   The child has 2 possibilities to create the animal

   (1) Create it (15 min.)

      (a) Find out, how you can create an animal, give it a name and an appearance

      (b) BREAK If the child can't continue, ask: How would you insert a new costume?

      (c) BREAK If the child can't continue: Give more hints. Else help.

      (d) BREAK If the child can't continue: (i) You need a script. (ii) Which script could it be?

      (e) The child might insert the script at the wrong location

      (f) Find out how to move the script

(g) BREAK If the child can't continue:

- What would you do on your cell phone?

(h) Find out how to delete the object Cat. If you found it out, delete the object Cat

(i) BREAK If the child can't continue:

- How would you delete something?

(2) Replace existing item (10 min.)

(a) Find out how to change the appearance and make a new animal out of the existing one

(b) BREAK If the child can't continue, ask:

- How would you insert an image?

(c) BREAK If the child can't continue: (i) You need a script. (ii) Which script could it be?

4. The animal moves (10 min.)
   The animal should move if you tap it
   Find out how the animal can move

(a) BREAK If the child can't continue: (i) You need a script (ii) Which script could it be? (iii) What exactly do you want to do?

(b) The child might insert the script at the wrong location

(c) Find out how you can move the script

(d) BREAK If the child can't continue:

- What would you do on your cell phone?

(e) If it's sliding: 5 seconds, and enter X: 0 as well as Y: 800

5. The sound of the animal (10 min.)
   The animal should make a sound if it is tapped
   Find out how the animal can make a sound.

(a) BREAK If the child can't continue: (i) You need a sound. (ii) How would you insert a sound?

(b) Find out how the animal can make a sound if it is tapped

(c) BREAK If the child can't continue: (i) You need a script. (ii) Which script could it be?

(d) The child might insert the script at the wrong location

(e) Find out how you can move the script

(f) BREAK If the child can't continue:

- What would you do on your cell phone?

6. Change Background (5 min.)

   (a) Find out how you can insert a new background

   (b) BREAK If the child can't continue:

   - How would you insert a new image?

   (c) Search for a new image in the gallery

   (d) PAUSE

   (e) Find out how to display the new image.

   (f) PAUSE

   (g) Choose the new image as background

A.3 After Catrobat Usage Questionnaire

There are several statements about using Catroid today. Please put an "X" in one of the boxes to indicate whether you agree or disagree with each statement.

	Strongly Disagree	Disagree	Not Sure	Agree	Strongly Agree
1 Using Catroid was **fun**					
2 Using Catroid was **interesting**					
3 Using Catroid was **frustrating**					
4 Using Catroid was **boring**					
5 Using Catroid was **cool**					
6 Using Catroid was **confusing**					
7 Using Catroid was **annoying**					
8 Using Catroid was **easy to learn**					
9 Using Catroid was **entertaining**					

If you used Catroid again, how long do you think you could use it at one time without getting bored? (Please circle one answer)

(a) Less than 1 hour

(b) 1-2 hours

(c) 2-3 hours

(d) 3-4 hours

(e) More than 4 hours

Catroid is currently being used to teach high school and college students. In what kinds of classes is Catroid used most frequently?

(a) Science classes

(b) Computer science classes

(c) Art classes

(d) Communications classes

(e) Other: ____________________________

If you had the application you used today (Catroid) on your cell phone, how often during the next month do you think you would use it? (Please circle one answer)

(a) Never

(b) Once during the next month

(c) Twice or three times during the next month

(d) Once a week during the next month

(e) More than once a week during the next month

Would you be interested in taking another Catroid class?

(a) Definitely not

(b) Probably not

(c) Maybe yes, maybe no

(d) Probably yes

(e) Definitely yes

Do you think you could create a program in Catroid that you would be proud to show your friends?

(a) Definitely not

(b) Probably not

(c) Maybe yes, maybe no

(d) Probably yes

(e) Definitely yes

Do you think you could learn to use advanced features in the Catroid program?

(a) Definitely not

(b) Probably not

(c) Maybe yes, maybe no

(d) Probably yes

(e) Definitely yes

Do you think you could learn a computer language like Java or C++?

(a) Definitely not

(b) Probably not

(c) Maybe yes, maybe no

(d) Probably yes

(e) Definitely yes

Would you be interested in taking a computer science class in high school?

(a) Definitely not

(b) Probably not

(c) Maybe yes, maybe no

(d) Probably yes

(e) Definitely yes

Can you imagine growing up to be a computer scientist?

(a) Definitely not

(b) Probably not

(c) Maybe yes, maybe no

(d) Probably yes

(e) Definitely yes

How likely is it that you will tell anyone about your experience with Catroid today?

(a) Very unlikely

(b) Somewhat unlikely

(c) Neither unlikely nor likely

(d) Somewhat likely

(e) Very likely

If you do plan to talk about your experience today, what will you say? (Please write your answer below.)

What are the 3 best things about Catroid?

1.

2.

3.

What are the 3 worst things about Catroid?

1.

2.

3.

Bibliography

S. Adams. Children get first mobile phone at average age of eight, Feb. 2009. URL http://www.telegraph.co.uk/technology/news/4680507/Children-get-first-mobile-phone-at-average-age-of-eight.html.

K. Andrews. Human-computer interaction – course notes, May 2013. URL http://courses.iicm.tugraz.at/hci/hci.pdf.

W. Apai. Interview with web usability guru, Jakob Nielsen, Sept. 2009. URL http://www.webdesignerdepot.com/2009/09/interview-with-web-usability-guru-jakob-nielsen/.

J. Bacon-Shone and D. Hui. Design and construction of the Hong Kong Creativity Index. In *Measuring Creativity: Proceedings for the Conference "Can Creativity be Measured?". Brussels, May 28-29, 2009*, Report EUR, pages 91–102. Publications Office of the European Union, 2009. ISBN 9789279128622. URL http://ec.europa.eu/education/lifelong-learning-policy/doc/creativity/report/hongkong.pdf.

D. Burtscher. Introduction of a continuous integration process in an open source project. Master's thesis, Graz University of Technology, Oct. 2012.

buzzle.com. Importance of information technology, Sept. 2011. URL http://www.buzzle.com/articles/importance-of-information-technology.html.

S. Card. *The psychology of human-computer interaction.* L. Erlbaum Associates, Hillsdale, N.J, 1983. ISBN 0898598591.

S. Card, T. P. Moran, and A. Newell. *Handbook of Perception and Human Performance*, chapter The Model Human Processor: An Engineering Model of Human Performance, pages 1–35. John Wiley and sons, 1986.

A. Cropley. Defining and measuring creativity: Are creativity tests worth using? *Roeper Review*, 23(2):72–79, 2000. ISSN 02783193. doi: 10.1080/02783190009554069. URL http://www.informaworld.com/openurl?genre=article&doi=10.1080/02783190009554069&magic=crossref.

M. Csikszentmihalyi. *Creativity: Flow and the Psychology of Discovery and.* HarperCollins, 2009. ISBN 9780061844034. URL http://books.google.co.jp/books?id=aci_Ea4c6woC.

D. Davidson. Manga grows in the heart of europe, Jan. 2012. URL http://geekout.blogs.cnn.com/2012/01/26/manga-in-the-heart-of-europe/.

T. Fidgeon. Ethnography when and how to use, Nov. 2011. URL http://www.spotlessinteractive.com/articles/usability-research/ethnography-when-and-how-to-use.php.

R. Fisher and M. Williams. *Unlocking Creativity: A Teacher's Guide to Creativity Across the Curriculum*. Unlocking Series. Taylor & Francis, 2004. ISBN 9781843120926. URL http://books.google.co.jp/books?id=W6WOTKVVjCwC.

R. Franken. *Human Motivation*. Thomson/Wadsworth, 2006. ISBN 9780495090816. URL http://books.google.at/books?id=hvW6AAAACAAJ.

Global Telecoms Business. A century of influence, Oct. 2008. URL http://www.globaltelecomsbusiness.com/Article/2199338/Search/Results/A-century-of-influence.html.

J. D. Gould and C. Lewis. Designing for usability: Key principles and what designers think. 1985. URL http://dl.acm.org/citation.cfm?id=3170.

D. Greenspan. Top it skills in demand for 2012, Oct. 2011. URL http://www.itcareerfinder.com/brain-food/blog/entry/top-9-it-skills-in-demand-for-2012.html.

T. Gritschacher. A community website for interactive mobile content created by children and teenagers. Master's thesis, Graz University of Technology, 2011.

H. Hollanders and A. Van Cruysen. Design and construction of the Hong Kong Creativity Index. In *Measuring Creativity: Proceedings for the Conference "Can Creativity be Measured?". Brussels, May 28-29, 2009*, Report EUR, pages 41–81. Publications Office of the European Union, 2009. ISBN 9789279128622. URL http://ec.europa.eu/education/lifelong-learning-policy/doc/creativity/report/design.pdf.

International Organization for Standardization. Iso 9241-11:1998 ergonomic requirements for office work with visual display terminals (vdts) – part 11: Guidance on usability, 1998. URL http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=16883.

International Organization for Standardization. About ISO, May 2013. URL http://www.iso.org/iso/about/about.

T. S. Jastrzembski and N. Charness. The model human processor and the older adult: Parameter estimation and validation within a mobile phone task. 2007.

C. Kelleher. *Motivating Programming: using storytelling to make computer programming attractive to middle school girls.* PhD thesis, Carnegie Mellon University, 2006.

D. Kieras. Using the keystroke-level model to estimate execution times. 1993. URL http://www.pitt.edu/~cmlewis/KSM.pdf.

F. R. Knapitsch-Scarpatetti-Unterwegen. Usability testing of mobile applications for children. Master's thesis, Graz University of Technology, Oct. 2012.

S. Krug. Don't make me think: A common sense approach to web usability. 2000.

A. Marshall. Importance of IT in business, Nov. 2009. URL http://ezinearticles.com/?Importance-of-IT-in-Business&id=3221599.

Massachusetts Institute of Technology. Scratch, Feb. 2013. URL http://scratch.mit.edu/.

S. Mayer. Testen von mobilen html5 web-applikationen. Master's thesis, Graz University of Technology, Feb. 2012.

M. C. Medlock, D. Wixon, M. Terrano, R. Romero, and B. Fulton. Using the RITE method to improve products: A definition and a case study. *Usability Professionals Association*, 2002.

J. Nielsen. How to conduct a heuristic evaluation, Jan. 1995. URL http://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/.

J. Nielsen. Improving system usability through parallel design, Feb. 1996. URL http://www.nngroup.com/articles/parallel-design/.

J. Nielsen. Why you only need to test with 5 users, Mar. 2000. URL http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/.

J. Nielsen. Usability 101: Introduction to usability, Jan. 2012a. URL http://www.nngroup.com/articles/usability-101-introduction-to-usability/.

J. Nielsen. Thinking aloud: The #1 usability tool, Jan. 2012b. URL http://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/.

phonearena.com. No surprise—more children owning cell phones, Jan. 2010. URL http://www.phonearena.com/news/No-surprise---more-children-owning-cell-phones_id8622.

M. Resnick. Mitch Resnick: Let's teach kids to code, Jan. 2013. URL http://www.ted.com/talks/mitch_resnick_let_s_teach_kids_to_code.html.

M. Richtel. Making web sites more 'usable' is former Sun-engineer's goal, July 1998. URL http://www.nytimes.com/library/tech/98/07/cyber/articles/13usability.html.

M. Runco. *Creativity: Theories and Themes: Research, Development, and Practice*. Educational psychology. Elsevier Science, 2010. ISBN 9780080467832. URL http://books.google.at/books?id=nJlfKE-meqEC.

R. K. Sawyer. *Explaining creativity: The science of human innovation*. Oxford University Press, New York, 2006. URL http://ukcatalogue.oup.com/product/9780195304459.do?keyword=keith+sawyer&#38;sortby=bestMatches.

D. Spencer. Card-based classification evaluation, Apr. 2003. URL http://boxesandarrows.com/card-based-classification-evaluation/.

D. Spencer and J. Garrett. *Card Sorting: Designing Usable Categories*. Rosenfeld Media, 2009. ISBN 9781933820026. URL http://books.google.at/books?id=_h4D9gqi5tsC.

R. Sternberg. *Handbook of Creativity*. Cambridge University Press, 1999. ISBN 9780521576048. URL http://books.google.ca/books?id=d1KTEQpQ6vsC.

J. Stierna and E. Villalba. Design and construction of the Hong Kong Creativity Index. In *Measuring Creativity: Proceedings for the Conference "Can Creativity be Measured?". Brussels, May 28-29, 2009*, Report EUR, pages 103–111. Publications Office of the European Union, 2009. ISBN 9789279128622. URL http://ec.europa.eu/education/lifelong-learning-policy/doc/creativity/report/science.pdf.

TED Conferences, LLC. TED Talks: Ken Robinson says schools kill creativity, June 2006. URL http://www.ted.com/talks/lang/en/ken_robinson_says_schools_kill_creativity.html.

TED Conferences, LLC. TEDx Events, Feb. 2013. URL https://www.ted.com/tedx.

P. Treitler. A generic approach towards smart self-rendering mobile user interfaces. Master's thesis, Graz University of Technology, 2012.

UNITiD and contributors. Android patterns, Apr. 2013. URL http://www.androidpatterns.com/uap_pattern/pinch-spread.

E. Villalba. *Measuring Creativity: Proceedings for the Conference "Can Creativity be Measured?". Brussels, May 28-29, 2009*. Report EUR. Publications Office of the European Union, 2009. ISBN 9789279128622. URL http://books.google.at/books?id=DczbcQAACAAJ.

R. Weisberg. *Creativity: Beyond the Myth of Genius*. Books in psychology. Freeman, 1993. ISBN 9780716723677. URL http://books.google.at/books?id=_VNwQgAACAAJ.