

# CHALMERS



## Developing Mobile Applications For Children

MIKAEL WÄNGBERG

*Examensarbete*

*Högskoleingenjörsprogrammet för datateknik*

CHALMERS TEKNISKA HÖGSKOLA

Institutionen för data- och informationsteknik

Göteborg 2012

Innehållet i detta häfte är skyddat enligt Lagen om upphovsrätt, 1960:729, och får inte reproduceras eller spridas i någon form utan medgivande av författaren. Förbudet gäller hela verket såväl som delar av verket och inkluderar lagring i elektroniska och magnetiska media, visning på bildskärm samt bandupptagning.

✂ Mikael Wängberg, Göteborg 2012

### **Abstract**

*The challenge of designing software applications is an enormous area for discussion. Designing an application for kids is one part of this area and it in itself raise many questions. It is this area which will be the focus of this project. I will discuss several aspects which need to be taken into consideration when designing something for small children; including deciding on the graphical layout of an application, how to maneuver within the application, how to preempt a child's unpredictability, how to break away from gender stereotypes and how to commercialize a product aimed at children. Other authors have written discussion about what should be taken into consideration when designing for children and I have combined their results together with my own reasoning to create a sort of guide over what to think about. In conjunction with these discussions I have created an Android game which incorporates the results of the discussions in order to make an easy to use product which is both entertaining and educational. The game is a learning tool intended to help children connect words with objects, with the ability to allow for customization should the parent or any other authoritarian individual wish to specify the area of learning. I in the discussions relate to how results are incorporated in the application and what affect they have on design choices.*

### **Sammanfattning**

*Utmaningen att designa en mjukvaruapplikation är ett enormt område för diskussion. Att utveckla en applikation för barn är en del av detta område som i sig självt väcker många frågor. Det är inom detta område som fokus för detta projekt kommer ligga. Jag kommer diskutera flertalet aspekter som måste tas i åtanke då man utvecklar något som riktar sig mot små barn. Detta inkluderar val av hur den grafiska layouten skall formges; hur man manövrerar applikationen; hur man förebygger mot barns oförutsägbarhet; hur man skall motverka könsstereotyper; samt hur kommersialiserar man en produkt riktad mot små barn. Andra författare har diskuterat dessa frågor tidigare och jag har kombinerat deras kunskap med mitt eget resonerande för skapa en slags guide över vad man bör tänka på. I samhörighet med dessa diskussionerna har jag tillverkat ett Androidspel som inkorporerar diskussionsresultaten för att göra ett spel som är enkelt för barn att använda men samtidigt är roligt och lärorikt. Spelet är ett läroverktyg ämnat att hjälpa barn koppla samman ord med objekt. Spelet är även anpassningsbart så att man kan specificera vad barnet skall lära sig. I diskussionerna relaterar jag till hur resultaten använts i applikationen och vilken påverkan de haft vid olika designval.*

*Keywords: Android, learning, software application, design, children*

## Table of Contents

1. Introduction.....	1
1.1. Problem.....	2
2. Purpose.....	3
3. Method.....	4
4. Theory.....	5
4.1. Colors.....	5
4.1.1. Colors' effect on children.....	6
4.1.2. Combining colors.....	7
4.1.3. Colors and Gender.....	8
4.2. Text.....	10
4.3. Shapes And Symbols.....	13
4.4. Utilizing Screens/Views.....	15
4.4.1. The splash screen.....	15
4.4.2. The game menu.....	16
4.4.3. The settings screen.....	17
4.4.4. The game screen.....	18
4.4.5. Sizing interactive elements .....	19
4.4.6. Positioning interactive elements.....	20
4.4.7. Childproofing the UI.....	20
4.5. Sound.....	22
4.6. Advanced Technical Features.....	23
5. Creating an application.....	24
5.1. Application Introduction.....	24
5.2. Requirement Specification.....	26
5.2.1. Core requirements.....	26
5.2.2. If-possible requirements.....	27
5.2.3. Wish list requirements.....	27
5.3. Design goals.....	29

5.4. Method.....	30
5.5. Choosing the color scheme.....	31
5.6. Implementation of screens/views.....	33
5.6.1. The splash screen.....	33
5.6.2. The game menu.....	33
5.6.3. The settings screen.....	34
5.6.4. The game screen.....	34
5.6.5. Traversing the application.....	36
5.7. Commercialization.....	38
5.8. User Acceptance Testing.....	40
6. Conclusion.....	41
References.....	42

# 1. Introduction

Each day hundreds, perhaps thousands of new applications are released for cell phones and tablets around the globe. Some of these applications are created by companies to use at the company in order to increase work performance, others are made to advertise the company to others, the majority is of course applications made to be sold on different markets to other companies or to private parties. No matter what the application is for, a lot of thought has to go into its creation and a great many decisions have to be made. What is the purpose of the application? What does it do? Who is it for? These are perhaps the first questions asked.

When the basic idea has been hammered out new questions arise. Does the application have a graphical interface? If so, what does it look like? Creating a good user interface, hereafter referred to as UI, is integral to making a good application. The UI is what the user sees, what the user interacts with. If the UI is not good enough it often does not matter how sophisticated or thought out the rest of the application is.

When creating an application aimed for children it becomes even more imperative that a lot of thinking goes into the design of the UI. A child has limitations, such as limited motor skills, which the application has to be built around. This is important to have in mind when deciding on how to design for children.

It is because of this the intention of this report is to discuss what one as a software developer should be aware of when designing an application for small children. Furthermore if the application is to be a learning tool, thought must not only go into making the application fun for the user but also educational.

## **1.1. Problem**

The problem faced when designing for children is how to design to overcome a child's limitations. While an adult may be able to relate to past experience and form an idea of what would happen in correspondence to an action a child will often take the UI at face value. A child is also not yet completely physically developed, something which may affect how they use the application.

## **2. Purpose**

The purpose of this report is to discuss the challenges which come into effect when designing for children. These can be anything from deciding on what colors to use, to what fonts to use for text if text should be used at all, to deciding on whether to have a start screen or not. In essence what choices make interaction with the application fun, easy and educational, while at the same time lessen the need for an adult to offer constant assistance.

In conjunction with these discussions I will also attempt to create an Android application in order to incorporate theory with practical application. This application will take the form of a learning tool intended to teach small children the connection between a word and an object.

The first part of this report will discuss different topics on what to consider when creating an application for children in general while the second half will focus on the application being made and show how the theory is implemented.

Many of the topics discussed in this report are large enough to warrant reports on their own but I will attempt to cut them down to shorter sections where I put forward some problematic areas, how to work around them and in turn how they influence the creation of an application.



### **3. Method**

The discussions which make a large part of this report are built around what I have learned about software development during my studies at Chalmers University of Technology, my experience with different software applications, and what others have already written about the subjects. During the course of the project a multitude of search engines were used, such as: Chans (to search Chalmers library), Summon, ProQuest – Computer And Information Systems Abstracts, Merriam-Webster Dictionary, Google. Keywords frequently used were such as design, software, children, kids, colors, fonts, shapes, symbols, mobile applications, Android, iOS, mobile technology, tablets, education, learning.

## **4. Theory**

This chapter is divided into a number of sub-chapters where each sub-chapter discusses one subject of interest when designing software. Beginning with colors we discuss graphical aspects which have to be considered during design, the other subjects are text and shapes and forms. After this a section is devoted to discussing how to use the mobile device screen, as well as how and when to use different in-application views. The theory chapter concludes with two subjects which do not necessarily affect the applications graphical design, sound and utilizing device features.

### **4.1. Colors**

Colors play an essential part as to how we as humans perceive the world. They can invoke powerful emotions and memories and through our lives we are taught a correlation between colors and many different things. For example red is a color often used to represent love, blue is used to induce a feeling peace and tranquility, and green brings our thoughts to nature. These are examples how older children and adults may react to some colors. The reason we react to colors in different ways is that throughout our upbringing we are programmed to. As we grow we create connections, some we make from our own individual experiences and some are cultural. As an example the color white is often used to represent purity and innocence in Western countries where as in many Eastern countries it stands for mourning. (Cherry) Below is a list which exemplifies what different colors may represent.

Red	Passion, Love, Anger
Orange	Energy, Happiness, Vitality
Yellow	Happiness, Hope, Deceit
Green	New Beginnings, Nature
Blue	Calm, Responsible, Sadness
Purple	Creativity, Royalty, Wealth
Black	Mystery, Elegance, Evil
White	Purity, Cleanliness, Virtue
Gray	Moody, Formality
Brown	Nature, Dependability
Beige	Conservative, Piety, Dull
Ivory	Calm, Purity

*Tabell 1: Color Theory for Designers, Part 1: The Meaning of Color by Cameron Chapman*

#### **4.1.1. Colors' effect on children**

Younger children have not yet established these abstract connections. Does this mean colors do not affect children? No, children may not immediately think of danger when they see a red light but it is not only through these connections colors affect us. Colors can cause deeper reactions, having an impact on how we feel, how we react, and even on how well we perform. Rose Kivi writes that “the colors red, orange and yellow stimulate and increase brain activity” and “the colors green, blue and violet induce relaxation”. (Kivi) In her article *Do Colors Affect Children's Learning* Dana Tuffelmire states that in order to foster a learning environment for children in elementary classrooms one should choose to paint the room in primary colors such as red, yellow and orange. (Tuffelmire) In classrooms meant for older students Tuffelmire instead recommends cool colors such as blue or green because this will help the students focus. Kivi goes on to state that a comfortable feeling can be induced by the use of warm colors while bright colors create a feeling of excitement. While these are not definitives as to how a color will affect a specific child it provides a generalization which is very

helpful when deciding on what colors to use in a software.

Books or toys are often painted with bright colors to attract children. The reason why children respond so well to bright colors is that their eyes are not yet fully developed. This causes them to not be able to differentiate between shades in a color. (Evans) When bright colors are used and put in contrast with each other children can see the difference between them and this creates a more vivid world to explore.



*Illustration 1: Curious George on PBS Kids Play*

If the software is meant to be a fun game one could choose to use a happy color such as yellow to induce this feeling, or if it is a learning tool such as the application created in conjunction with this report one might strive to use either a warm red color scheme or a bright blue one. The idea behind the warm red colors would be to create a comfortable and stimulating learning experience for longer learning sessions whereas bright blue colors would be aimed at shorter but exciting and focused sessions.

#### **4.1.2. Combining colors**

While the theory behind how colors affect us is intriguing it is not the only thing which is important when deciding upon what colors to use in a software application. It is equally important to

consider how colors affect each other. If similar colors are used it becomes harder to distinguish between different parts of the screen. For a child this is even more so. While an adult may distinguish between shades of colors a child will not do so with the same ease. Therefore software for children should be created with contrast in mind. By having a significant contrast between different objects on the screen it becomes much easier to direct the user's attention to what is important.

Contrast can also be used to group objects together or create familiarity. As an example the button to return to the main screen, and only that button, could always be red. After a few tries the child will associate red buttons with this function and learn to seek it out if this is what they wish to do. But if multiple buttons look similar while having different functions it will be harder for the child to recognize what happens and this might make the child more reluctant to use the software.

It is also important to separate the objects which can be pressed from the background. When doing this with colors the most effective would likely be to use few colors for the background so that it does not become too cluttered and also to use a very different color for the object to press or at least for the border of the object.

By having a broad spectrum of contrasting colors you can use it to the software's advantage by creating an environment which is exciting and which invite to exploration. (Naranjo-Bock, 2011a) It can often be difficult to capture a child's attention for long because they are always looking for new things to experience. If a software application can offer something different wherever the child directs its eyes it is much more likely to keep the child interested.

### **4.1.3. Colors and Gender**

When discussing colors there is one topic which is almost always brought to the table, gender. Gender is discussed everywhere in society and countless of reports have been written on the subject. In this report I will in short discuss this subject in relation to colors and leave a more in-depth discussion for others to write.

There are some colors which are for one or another reason considered to be male or female colors. Whether the origin for these groupings originate from culture or biology has been the subject of many studies. No definitive conclusion has been made but many agree that it is a combination of both. (Clark)

Colors we promptly categorize as belonging to one gender ahead of the other are most recognizably blue for boys and pink for girls. A prominent example of this can be seen if entering virtually any toy store. One side of the store is marked with pink shelves and boxes containing toys the store want to market towards girls while the other side of the store with toys marketed for boys is most prominently blue. The same pattern can be found in software applications. Games marketed towards girls are often more likely to feature the color pink than a game marketed towards a boy would. This is not to say that there is something wrong with basing the color palette in a game aimed at young girls on the color pink. Decisions like those for what color to use normally takes multiple factors into account. If market studies show that a decision to use pink is the best one then there is no reason for why you should not be able to do so. Neither are there any reasons for why it should not be possible to make a pink game for boys or a blue one for girls.

When creating a software meant to be attractive to both boys and girls it might be a good idea to look at colors which tend to be considered gender neutral. These colors include yellow, green, orange, white and brown. (Anon, n.d.-a) Children become aware of the fact that there are two genders and that they belong to one of them by the age of two. In order to find a secure place within its gender it is not uncommon for a child to adopt the colors society has decided defines that gender while rejecting colors which do not. (Clark) In order to reach as large an audience as possible it is important not to alienate consumers and one should therefore strive to use colors which everyone can find attractive. By choosing from the neutral colors to use as the base palette one avoid the problems which may otherwise arise and there is nothing which says that colors considered gender specific can not be used to complement the base palette.

## 4.2. Text

Everyday we humans are subject to large amounts of text. We see it on signs, we read it in books and newspapers, we use it to communicate with the people around us via e-mail, SMS and instant messaging. It can safely be said that without the written language human society would be radically different. Using text we can deliver a message exactly the way we meant it. When designing an application text is an important tool. Through text it can easily be explained that the function of one button is to quit the program while that of another is to open the settings menu.

When deciding on how and where to use text in an application one should as always look at the intended users. The ability to read varies greatly based upon the age of the users and whether they are experienced readers or not.

*“Your use of text should be moderate unless you are designing specific areas of an application that need to accommodate large amounts of text ... for children who have experience reading.”* (Naranjo-Bock, 2011b)

When you're designing for children you therefore have to factor in their limited capability to read. Most often designs aimed for children are to a large extent graphical and use visual cues to guide the child through the application. Sometimes though it can be appropriate to use text and if so there are certain things to be considered.

Children most often learn to read letter by letter until they are able to decipher the entire word. Because of this it is important to use a typeface where the letters are easily distinguished from each other. Naranjo-Bock suggests that a typeface is used where there is a bit of space between the letters and the contours are well-defined because this emits a warm and inviting feeling. (Naranjo-Bock, 2011b) One should avoid using cursive letters or a typeface which is deformed since it then becomes much harder for children to read.

It is important that it is as effortless to read the texts used as possible and it is therefore a good idea to use sans serif fonts because of their simpler forms. The exception to using sans serif fonts is if the application is meant to mimic a normal book. In that case we want the child to have a similar experience to the physical book the child normally reads.

Below is an example where the first illustration uses a font with serif and which is cursive. The second illustration is sans serif.

*PLAY* **PLAY**

In a software application for children the most likely places to find texts are on buttons or headings. These texts are often styled to have a sharp contrast with their surroundings to make it clear that the user can interact with these objects. If it becomes evident that users have trouble finding the interactive elements one should perhaps reconsider the typeface used and go for a simpler one.

Another aspect to consider is the usage of upper and lower case letters. Depending on the font used some lower case letters such as “a” or “q” might look radically different than the ones most children are familiar with, because of this one should strive to use fonts which are more akin to the ones commonly used. Using upper case letters might in some cases solve these issues. Upper case letters are also prudent to use since they are easy to separate from one another and it is often upper case letters which children begin to learn. Using upper case letters on buttons and other elements is also an effective way to catch the users attention and indicate that these elements can be interacted with.

It is important that the text is in balance with the graphics which surrounds it. It is also important to be consistent when choosing a font so to make the user experience more coherent. It is very well to use more than one font if it serves a certain purpose. For example one font could be used to attract the child user's



attention while another one could be aimed at an adult. But it is important that it does not become cluttered or confusing.

It is important to remember that children are not always used to reading and are not likely to read large chunks of text but rather are used to single words or perhaps short sentences. Because of this, before anything else, when it comes to text the most crucial aspect is for it to be clearly legible.

### **4.3. Shapes And Symbols**

Small children do not know how to read and therefore rely on symbols to explain what something on the screen means. How well symbols work depends on the knowledge of the child user as to what the symbol stands for. If the child have a limited knowledge then their understanding of what the symbol means will also be limited. The difficulty of understanding a symbol is not to recognize the symbol itself but to understand what it represents. (Madigan, 2005) Therefore it is also important that icons within a program are distinct from each other so that confusion can be avoided.

Using square buttons with the text “Next” or “Back” to move between pages is easily understandable for adults, but for children this concept would be harder to understand. Using, for example, arrows to indicate directions is something which the child can grasp. The same goes for other options within the application. The button which plays a song can be an icon of a musical note and to play a video the television icon should be pressed.



In games or picture books for children we can often encounter shapes which are round rather than edged. There are several reasons for why one would choose to design this way. One reason is that rounded shapes come of as less imposing. The game or book should welcome the child in and not alienate it. Another reason why it can be a good idea to avoid sharp edges or strong geometric patterns is that these shapes can cause heavy discomfort to people with neurological disorders. (Anon, n.d.-b)

Shapes are can be found all around us in everyday life, wherever

we look there are circles and squares, round corners and edged corners. How these affect how we experience an application or if shapes have an impact on our ability to adopt information is something which should be investigated further to provide a better understanding on how to design.

## **4.4. Utilizing Screens/Views**

Traversing the many different screens of an application is generally not something which constitutes a problem for adult users. They have the experience and capability to recognize the difference between a splash screen, a game menu or a settings screen. Children do not always know how to react when confronted with one of these views within an application. It is therefore of great importance to carefully weigh the necessity of a view before implementing it. When designing an application for children between one and three years old it is best if the application were to start at once. (Ibarra, 2011) The reason for this being that since children that young cannot read, nor are they able to evaluate which is the best option when presented with multiple choices. A menu screen would thus serve no legitimate purpose. It is better that should the child by accident close the application it will immediately be playable again when restarted. According to Ibarra they also discovered during tests that if something unexpected happens on the screen, such as a pause or opening a settings menu, the child will often not press the back button but instead press the “Home button” causing the application to close and then look for the icon to start the application again.

### **4.4.1. The splash screen**

If the application requires to be loaded before it can actually be used there is often a splash screen to hide this loading time. For an older child this might be fine, they understand that the application is loading. A small child may not be as understanding. They often do not have the patience to wait so if the application does not start within a few seconds they will believe it does not work or at least become frustrated. A way to circumvent this if the splash screen lingers is to make something fun of it. Instead of displaying a company logo we could play some music or have a short animation. It could even be possible to have a puzzle or short mini-game. In short anything which makes the waiting feel less like a wait.



*Illustration 2: The splash screen for The Fantastic Flying Books of Mr. Morris Lessmore*

#### **4.4.2. The game menu**

As mentioned previously the game menu should be avoided if possible when the game is intended for very small children. For children above the age of three it should be alright to use a game menu as long as the options are kept to less than a handful and it is made very clear what on the screen are buttons. They should for instance be large and made to feel like clicking on a physical button.



*Illustration 3: Kids Song Machine. An iOS application*

### 4.4.3. The settings screen

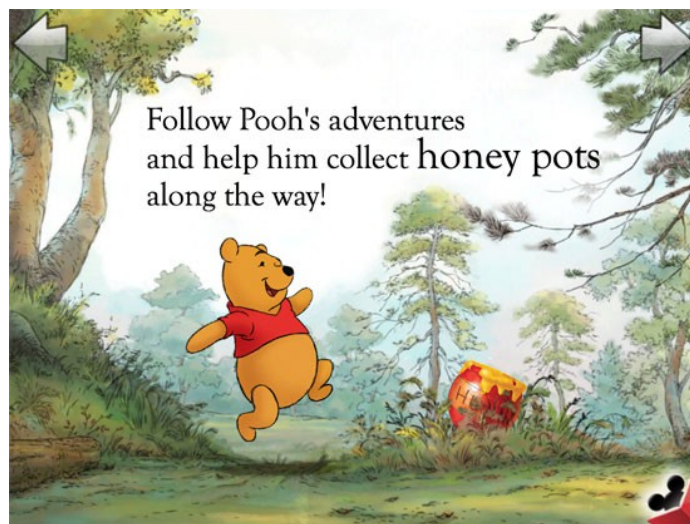
The settings option is another view featured in many applications. Sometimes it is a quick menu that pops up and then goes away, while at other times it has its own screen. When designing for children if the application does not absolutely need settings options, then it should be avoided. But if settings is a must, they have to be looked upon from a perspective of either to be used by children or by adults. If the settings are intended for children it is good if they are easily understandable, but the most important thing is that they do not have too large an impact on the application. A child may accidentally access the settings numerous times and they should therefore be as non-intrusive as possible. (Ibarra, 2011) If the settings are meant for an adult and have a large impact upon how the app plays they should be harder to access by mistake and if it is possible it could even be a good idea to keep them out of the application itself and place them in the section of the device where you reach all installed applications.



Illustration 4: iOS application Wheel on the Bus settings screen

#### 4.4.4. The game screen

We have now discussed when and how to implement splash screens, game menus and settings options. What is left to discuss is the heart of the application, the actual game screen, the one screen that cannot be left out of the program. The game screen can come in many different variants and it all depends on the type of application. A virtual book would require there to be multiple pages and an easy to use method to flip between the pages. One solution could be to use swiping motions but they are often difficult for younger children to use and even for those a little bit older it might not be intuitive. Also, if the application is sensitive to swiping movements it might mistake an ungainly press for a swipe. It is in most cases probably better to go with some kind of forward and back buttons. Perhaps in the shape of arrows as suggested in the chapter about symbols. But even though symbols are good there is something better, if it is possible and especially if the application is intended for toddlers it would make it easier if the application used one view. Suggestions for the other types of screens also apply to the game screen and in the next few sections more ideas which are applicable everywhere in the application will be presented.



*Illustration 5: Winnie the Pooh: What's a Bear to do? iOS puzzle book application*

#### 4.4.5. Sizing interactive elements

Apart from being easy to find and realize their function there is another important reason for making interactive elements large. Young children have yet to develop full motor skills. This makes it hard for them to perform delicate movements or target small objects. (Hourcade, 2007) Screens used for mobile devices are generally quite small. Looking at cell phones, screens are often less than four inches large, this cause them to be a poor platform when creating games for small children. The size of the interactive elements to make them easily pressed would severely limit the number of elements which can be on the screen. Tablets tend to be a little bit larger. Usually ranging from seven to ten inches in size making them in many cases more suitable.



*Illustration 6: iPad and iPhone comparison*

Small buttons are hard to hit, perhaps even hard to find, and if it is hard to press a button or the result of the action is not presented fast enough children often resort to rapidly pressing multiple times looking for the interactive object. This may result in them pressing a different button than the one they intended and in turn cause frustration.



#### **4.4.6. Positioning interactive elements**

In order to avoid the wrong element from being pressed it is important where objects are situated on the screen. The older the child using the application is the more of the screen can be utilized, but for young children one has to think twice before placing a button. To a toddler a tablet is quite large. They do not have the physical capability to use a table effectively. The tablet will have to be located on the floor or on the table in front of the child. This is true also for children a few years older. With the machine before them the top of the screen is the part furthest away and in order to reach this part of the screen it is not uncommon for the child to touch the lower region of the screen by mistake. Therefore interactive elements should as a rule be kept away from the bottom area of the screen. (van der Merwe, 2012) This is especially true if the accidental press has a large effect on what is happening on the screen. If it is a button to produce a sound effect it does not really affect what is on the screen so it does not really matter, but if it is a button to return to a previous screen or a settings menu the accidental press can be disastrous. Positioning buttons which may take the user away from the application or are not meant to be used often at the top of the screen and in the corners make them much less likely to be pressed by accident.

#### **4.4.7. Childproofing the UI**

No matter where on the screen an interactive element is positioned there is no guarantee that the button wont be pressed by a child, intentionally or by mistake. If the buttons are not meant for the child user but perhaps for an adult supervisor there are tricks to hide them. If they are not meant for children it is not necessary for the objects to be as large, so an easy first step can be to make them smaller and color them so they do not stand out as much from the surrounding. But even if the buttons are small there is a chance for them to be pressed. In the article *Designing Experiences for Young Kids: Child Proofing you Application* (Itzkovitch, 2012) we are presented with ways to prevent the wrong buttons from being activated. One possible technique to use would be a “double tap”-method. In this case the first interaction

with the button would cause it to change color or make it clearly visible. After the first click there is a short time window in which the button can be pressed again to actually perform the desired action. If nothing really awarding happens it is likely that the child will move on to other objects on the screen. A similar solution is to have the button move after the first click. If the cause of the first click is that an arm has been placed on the screen it does not help much that the button switches color. The arm will still be there to press it again. But if the button moves from the left side of the screen to the right or from the top to the bottom this situation might be avoided.

Another method which hides the game menu could be to make it something which has to be dragged on to the screen. Perhaps by pulling a rope or a banner. Children often have a harder time performing swiping movements than point and click and if swiping gestures is not part of the general application design it is even less likely that they should think of it. To make it apparent to the adult user that the object can be interacted with there could for example be a sign attached to the rope with the word “pull” on it.

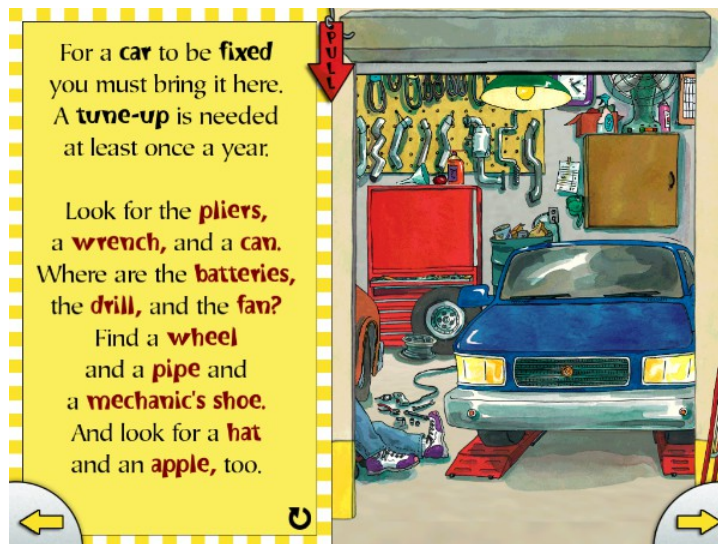


Illustration 7: Roxie's Doors for iPad

## **4.5. Sound**

There have not been many studies made focusing on what effect sound has when used in children applications. (Hourcade, 2007) Therefore this is a difficult subject to discuss. But one thing we can be sure of is that sound attracts attention. Using sound one can direct what the user should press. For example a button which the child is meant to use might play something fun while a settings button which the child is not supposed to use does not give any audio feed back at all. The sound adds another dimension to the application which the user can explore. Perhaps different buttons play different tones.

Music in the background is another way to use sound and different melodies can produce different reactions from the user. A quick song, a calm song, a happy song, they each affect the user differently. But how exactly this work and how sound can be combined to create an environment which is educational, fun, and/or exciting is something which requires further studying and will not be further explored in this report.

## 4.6. Advanced Technical Features

Mobile devices often come with many features which can be fun to use. Taking a photo with the camera, recording sound with the microphone, moving a ball on the screen with a gyro, and vibrating the device when the player is hit are only a few examples of what some machines can do. Depending on how old the user is these functions can be used to varied success. A button which allows the user to record a few seconds of sound and then plays it back can be a fun extra feature usable by all ages and it does not intrude too much on the general function of the application. Most mobile devices come with a built in gyro which can be used to create entertaining games where the user tilt the machine to move the character from left to right or shake it to make the snow fall inside a virtual snow globe. While functions like these might in fact make a fun game it might not be suitable for younger children since they often lack the strength and physical maturity to hold on to the device while performing the required manoeuvres. The end result might be the destruction of an expensive piece of hardware and an infuriated adult not willing to download other applications from the same creator. Therefore it is best to thoroughly analyze what features the device provides, who the user is, and whether the application actually become better using these features.



*Illustration 8: A vibrating Samsung Galaxy s2*

## 5. Creating an application

### 5.1. Application Introduction

There are many applications on Google Play which can be downloaded to an Android tablet. The nature of these applications vary greatly. Some are advanced role-playing games while others are simpler puzzle games, and others yet are tools to learn new languages. Whatever program one might search for there is a big chance that someone has created it and it is available for download.

Due to how easy tablets can be to interact with they have become more and more common in all parts of society. They are used at home, at companies, in school, even at day cares are they used. Since all you need to do to interact with a tablet is to touch the screen it is not difficult to use them even for children. With a growing number of educational applications tablets have become even more useful. Children can use the tablet to learn the alphabet or to interact with a song book.

When a family member of mine approached me about creating an application for a friend of her's to help that friend's child to learn the names of objects or family members. Her friend had been looking for such an application but had not found one that fulfilled her requirements. That was when I hatched the idea for *What's That!?*.

*What's That!?* is a game developed for tablets with the intention to be a tool for small children, around 2 years old, to learn the connection between a word and an object. The main idea for the application was that of a number of pictures arranged in the center of the screen. An audio clip would then play, giving the name of the object one of the pictures depict. It is then the child's task to find the correct picture and press it. If the correct picture is pressed a new set of pictures would appear. The order in which the pictures appear and the decision of which picture is the one to be pressed are randomized so that there wont be a pattern as to

which is the correct picture.

The pictures are retrieved from a local database and the application comes prepared with a number of premade pictures and audio clips, but it should also be possible for the user to add their own material to the database. To customize the experience further it should be possible for the user to arrange lists with pictures of their choice. This way the user could specify what words the child should learn. For example the user could construct a list containing vehicles, or if the user has added pictures of family members the child can learn to connect the names with the persons.

The application had to be designed to be easily used by children and special consideration should be taken to help children with learning disorders.

## **5.2. Requirement Specification**

This section states the requirements which *What's that!?* has to meet. They are divided into three levels of importance. Core requirements are requirements which have to be fulfilled for the application to be considered to live up to its purpose. If-possible requirements are requirements which should be fulfilled if there is time or possibility to do so. Wish list requirements are just that, requirements which we wish to have in the application but only if there is time after all other requirements are fulfilled. Within each level of importance there is no further prioritizing.

### **5.2.1. Core requirements**

#### Functional requirements

- ◆ User shall be able to press pictures representing the answer and get feed back
- ◆ Four pictures shall be displayed on the screen
- ◆ Choosing the right answer shall be rewarded
- ◆ Choosing the wrong one should be reprimanded
- ◆ The word chosen as the answer shall be read aloud
- ◆ It shall be possible for the user to replay the word
- ◆ It shall be possible for the user to add new words to their database
- ◆ The user shall be able to remove words inserted by the user.
- ◆ The application shall come with premade pictures and audio files stored in the database
- ◆ The user should no be able to traverse to a previous view using the Android back button

#### Non-functional requirements

- ◆ The same word should have the possibility to appear multiple times
- ◆ The application shall work with Android 4.0
- ◆ The application shall be displayed proportionally correct on 10.1 inch screens
- ◆ Images, audio, etc. shall be stored in a local database
- ◆ Starting the application should require a maximum of ten seconds
- ◆ Word audio files should be a maximum of three seconds long

- ◆ The application should be locked to a horizontal display mode

### **5.2.2. If-possible requirements**

#### Functional requirements

- ◆ It should be possible for the user to create lists with words to be used in the game
- ◆ The user should be able to remove user created lists
- ◆ There should be a screen lock to prevent the child user from leaving the game
- ◆ The device should stay awake for a while even if there are no user inputs

#### Non-functional requirements

- ◆ Lists should be a minimum of four words in size
- ◆ The user should be able to create a maximum of twenty lists
- ◆ Saving a list should take no more than three seconds
- ◆ Without user input the device should stay awake for a minimum of fifty second
- ◆ Without user input the device should stay awake for a maximum of ninety seconds

### **5.2.3. Wish list requirements**

#### Functional requirements

- ◆ The user should be able to select the language the word should be read aloud in
- ◆ The user should be able to download word packs which include multiple new words
- ◆ For certain pictures it should be possible to hear a sound affiliated with the picture instead of the word being read aloud. Example: A dog barking
- ◆ Users should be able to share lists with each other



- ◆ The user should be able to select in what language text inside the application should be presented
- ◆ The user should be able to create lists by category. Ex: vehicles

#### Non-functional requirements

- ◆ The application shall function on devices without camera or microphone
- ◆ The application shall be displayed proportionally correct on all screen sizes
- ◆ The application shall function with Android version previous to version 4.0

### **5.3. Design goals**

The aim of this part of the project was to create a learning tool which is easy to use by young children and which takes in to account the problematic decisions when designing for children. The user should be given a word and a choice of pictures from which the correct picture matching that word is to be chosen. The word should be taken from an extensive database with prepared words or from a user created list of words. These user created lists should persist even after the application has been restarted. The pictures and words that come with the application belong to different categories to provide variation. They are animals, vehicles, household appliances, nature, sports, etc. To create the database which will hold all this data the application will use the SQLite database management system which comes with Android. This will ensure that the data is only accessible by the application.

#### **5.4. Method**

The design choices were made based on the discussions in the first half of this report together with the target audience of this application in mind.

## 5.5. Choosing the color scheme

Deciding on what color scheme to use is a crucial decision. As mentioned in the previous discussion about colors, colors can invoke many different emotions and if not used correctly the application can come off as being cluttered. Because of a child's short attention span if something goes wrong the child will move on to other applications more to its liking. Since *What's That!?* is intended for small children and with no specific gender in mind I have decided to go with a neutral color for the applications background. Since the application is meant to be a learning tool to be used casually we want to use colors which stimulate the brain to be more susceptible to taking in information while at the same time make the user feel comfortable to use the application under a longer period of time. Since the primary focus of the user should be on the pictures and nothing else I believe it is best to not use an image as the background nor use multiple colors. While an image may be fun and give the user a chance to explore I feel that it would be to much of a distraction.

To fulfill these conditions a warm orange color should be used for the background. Since we do not want to take focus from the elements which can be interacted with it is important that the background is not too bright nor that it is of a color which aggressively demands attention, such as red. Orange being a color which is both energetic and alluring is then an acceptable choice. (Chapman, 2010) While orange is not a overly bright color it is not a dull one and it can be used to create good contrast with other colors.

The design for *What's That!?* is simple. There are not overly many objects for the user to interact with. Since the main focus are the pictures presented and they already come with their own colors it is important that the background is a color which can match most other colors while not overpower them. I believe orange to be such a color.

To further separate the pictures from the background as well as from each other and make it clearer that these can be interacted

with I have decided to put borders around the pictures. Before choosing the final color to use for the border I tested several others. I tried to use a red border to draw the users attention but when in combination with the pictures it drew too much attention to itself and did not work in combination. Other colors I tested faced the same problem. While they worked with some pictures they were less successful with others. Though I wanted to use striking colors I decided to try using black. Black is a color which works in combination with almost every other color. It is not as eye catching as red or yellow, but by using it as the border I can create the illusion that the pictures stand out from the screen.

## **5.6. Implementation of screens/views**

Early on I decided what screens I would use and which I would not. The main screen where the game takes place was of course a given, but what about the others. In this section I will discuss my reasoning for including or excluding screens, different features within these and how I chose to allow the user to move through the application.

### **5.6.1. The splash screen**

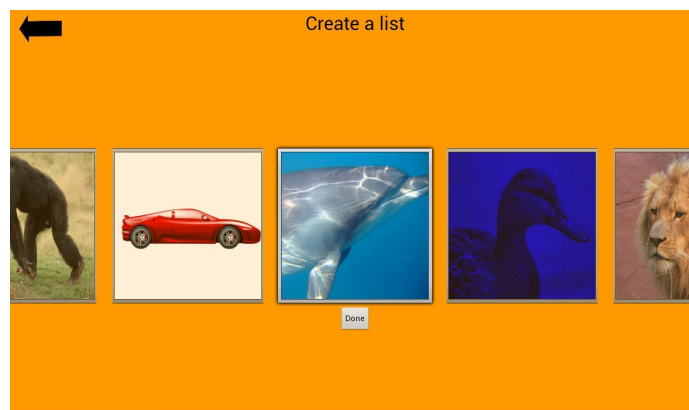
A splash screen is not just a way to cover up a loading time, it is also a good way to remind the user of who made the game. While this information is probably not of interest to the child using my application I opted to include the splash screen in hope that an adult supervisor might remember it when they go looking for new applications. Since my application does not take long to load it should not have to display the screen for more than a few seconds and I believe that should not inconvenience the child too much. Should the application in the future require longer loading time I would have to re-evaluate how this screen is implemented. Perhaps include some audio or make a short animation of the logo.

### **5.6.2. The game menu**

After the splash screen the application goes directly into the game, but my first design of the application included a menu before the game actually started. I had at first thought of using the menu as a way to reach the game's settings options since these can be used to regulate what images appear during the game and I wanted the adult user to set this up before starting the game. But since this game is intended primarily for toddlers and I want them to be able to use it without too much interference from an adult. I removed the game menu hoping that if the child mistakenly closes the application it might be able to start it anew and continue playing.

### 5.6.3. The settings screen

Leaving the settings options out of the game completely would make the need for adult intervention even less likely, but this is not an option since a large part of the game's functionality comes from the adult user's ability to add new words and images and the ability to decide exactly what words the child should focus on learning. The settings section contains multiple views and options intended for the adult user which the child user would not understand and it is therefore important to make it less likely to be accessible by mistake. In the event that the child opens the settings screen there is a button shaped like an arrow that will take the child back to the game. Hopefully the child will recognize the function of the arrow. The settings screen was made to only include options which would not change how the game works, should the child manage to change some of the settings before leaving. If the words shown are changed or if the child adds a image of itself it is still possible to continue playing the game.

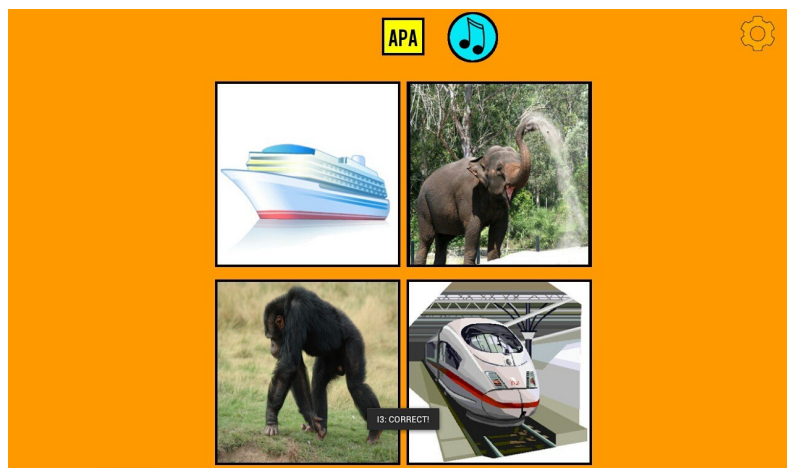


*Illustration 9: Create a new list in What's That!?*

### 5.6.4. The game screen

Since the purpose of this application is to learn words this should be the natural focus point. As each word has four pictures representing possible answers I decided to make these pictures large enough to cover most of the game screen, but also centered to maintain a relationship between them. I want the user to instinctively know that it is one of these which should be pressed.

There is space between each image to avoid the wrong image being pressed or multiple pictures pressed at once. This gives room to make them look more like buttons. Aside from the pictures the button the user should want to use is the button to replay the word audio file. This button is situated in the top center of the screen and is a round icon of a musical note. Next to the note is the word written in capital letters. I include the word in text even though most users of this application won't be able to read because it gives the user a third way to connect to the word. Having the word there also makes it possible for an adult to spell the word to the child should they be using the application together. Since this is only an aid for spelling the word the font is sans serif and it is in upper case since that is the form children are usually taught first. I use the musical note to represent sound since it is a simple icon and it is a connection many children learn while young.



*Illustration 10: What's That!? game screen example*



### **5.6.5. Traversing the application**

Access to the settings comes through the game screen and it should also be possible to exit the application. Reaching the settings is not something intended for children so I therefore looked at the methods mentioned in an earlier section on how to prevent this. While a swipe gesture to pull the settings on to the screen would have been impressive I did not feel it matched the otherwise down scaled tone of the application. I instead decided to use a double tap function. Originally the button is made to not stand out from the background, after the first tap it becomes clearly visible and only after the second tap does it take the user to the settings. Since the button looks similar to the background the child user will have a harder time noticing it. Especially since the other objects intended for the child take so much focus. The button itself is small and located in the top right corner of the screen. This keep it far away from the general area which the child will utilize. It has the shape of a cogwheel since this is something adults may connect to settings.

To exit the application one could opt to include a button to perform this function, I though have decided not to include this even though it was my first instinct. A exit button is an intuitive way to let the user know that this will close the application, but the device home button performs the same function. I assume that the user of this application knows this and I can therefore exclude an option which could otherwise mistakenly take the child out of the game. While I could use a technique like the one used for the settings button to make it less likely for the child to use the button it is still a possibility and if I can leave a possible object of irritation out of the game I opt to do so.

Otherwise moving between the different views is exclusively done using buttons inside the application. While it could be possible to return to a previous view using the Android implemented back button I have decided to disable it to make the application more stream lined. A button is used to enter a view and an application implemented icon of an arrow returns the user to the previous one. Another reason for disabling the Android back button is that this menu is often located on the bottom of the screen and I want

to avoid miss-clicks as well as this button being used to return to an unintended view. Located near this back button is the previously mentioned home button. One could question my decision not to disable this button since it could be mistakenly pressed just as easy as the back button. The reason I do not disable the home button is that this is something which should as a rule not be done. The home button is meant to function the same no matter where it is used and if applications began to to change this it could in turn damage the Android platform. The same would be true if I changed the back button to instead move forward into a view. This is not the intended function of that button and would cause confusion.

## **5.7. Commercialization**

This section differ somewhat from previous sections in that earlier the discussion has primarily focused on graphical elements. While this topic can have an effect on the UI it is not necessarily so.

There are generally three ways to make a profit from an application. Either there is an initial cost when the application is downloaded from the market place or there are adverts inside the application or there is an option to buy virtual items in the game. A fourth way would be to go outside the market and sell the application directly to users, but this is not always possible since some platforms require installation of applications to go through a market place.

The second method to use advertisements inside the application is favored in many applications found on Google Play, Google's online market for Android applications. The reason for the second method being highly favored by developers is a history of Android users being less accustomed to paying for software. There has been a consensus that Android is an open and free space and therefore applications should be so as well. The addition of advertising inside applications kept the appearance of the applications being free while still generating profit for the developers. This is a view which is slowly changing as more and more applications become available to a cost.

When creating software with the intention of somewhere down the road make a profit from it one has to think about the possible ways this can be achieved. As always the decision should be made with the primary audience in mind. In the case of this report we look at it from a child user's perspective. Using adverts in an application for children has the problem that children often do not know the difference between clicking on an advert and what is actually part of the game. The inclusion of adverts could be said to have the intention to lure children in to doing something they do not want to do and reach a place they do not know how to return from. This may cause an adult to be forced to intervene and thus make it less attractive in their eyes since they might like

the child to be able to get along by itself.

The other way to earn money inside the game by adding the possibility to buy items inside the game is another method which should be avoided when making a game for children. A child does not understand the consequences of its actions and the addition of this element serves no other purpose than to trick the child. Applications with these functions will most likely be deleted right away. As Rian van der Merwe writes, "If You Try to Trick My Kid Into Buying Stuff, You're Dead To Me". (van der Merwe, 2012)

The most appreciated method from a parents view would likely be to have a reasonable initial price and then leave other methods out of the application. Perhaps include a short segment at the splash screen with information about other games from the same developer. The problem with initial costs is that they cannot be too high since that will make users reluctant to purchase them, especially when they do not know how the application actually works. Partially because of this we can see when browsing any application market that applications usually cost one or a few dollars. To interest more buyers it might be a good idea to release a trial version where users can test some of the applications functionality and from that decide if they want to buy the full version.

To make a full assessment on how to make an application commercial as well as how to price it would require a far deeper investigation into the topic than what I do in this report but I leave this for future students to research.

In my application I have decided not to use advertising since it is intended for very small children and this inclusion would stand out clearly from the rest of the application making it very likely that children will press it, distracting them from the true intention of the game. Instead, should this application become available on the market, I would opt to have an initial cost for the product while also releasing a trial version where the settings options would be limited to encourage the users to buy the full version if interested.

### **5.8. User Acceptance Testing**

To evaluate how well the application serves the needs of its users is important for the development of any application. In the case of *What's That!?* it would aid the development greatly to have a selection of children test the application to see how well they respond to it. Furthermore, if possible, it would be interesting to see what happens if the children at a day care were allowed to test the software for a period of time. One would then not only be able to see how the children react to the game but also evaluate how much of an impact it has on their learning. To perform such a test one would have to establish how much the children know before they start to use the application and then see if any progress has been made after the trial period.

During this project there was no time to perform these tests, but further development would require it.

## 6. Conclusion

The purpose of this thesis was to write down some guidelines for when designing applications for children. It was never intended to be a strict guide to adhere to but instead a tool to help developers when they contemplate the direction for which they should take their application. There is a need to discuss and research this subject in order to develop better applications which deal with the difficulties which arise from children's limitations. After researching topics of interest to design as well as applying the theory in an attempt to create a learning application, the conclusion which was reached was that there is no set of rules which apply to all projects. Each project has to be approached from its own angle and adjustments have to be made, among other things, based on the age of the user, the form of the application, and the purpose of the application. All applications for children should strive to teach something, but the application will look radically different if it is a tool to learn words or if it is a puzzle. Several ideas proposed in this thesis work for some application while other ideas work for another.

The Android application, *What's That!?*, created in conjunction with this thesis is one example of how an application can be designed. To conclude if the design chosen was good or if it needs to be adjusted to better serve the needs of its users require extensive testing, something which was not achievable during the scope of this project.

The discussions in this thesis barely touch the surface of each subject to give a quick overview. To gain a deeper understanding of how each subject should be approached further research would be required.

## References

- ANON. n.d.-a. *Gender Neutral Baby Colors* [Online]. [www.ehow.com](http://www.ehow.com): eHow. Available: [http://www.ehow.com/list\\_6661953\\_gender-neutral-baby-colors.html#page=0](http://www.ehow.com/list_6661953_gender-neutral-baby-colors.html#page=0) [Accessed May 3 2012].
- ANON. n.d.-b. *Organic And Inorganic Design* [Online]. <http://www.basiccomposition.com/>: BasicComposition.com. Available: <http://www.basiccomposition.com/resources/organicandinorganicdesign.pdf> [Accessed June 3 2012].
- CHAPMAN, C. 2010. *Color Theory for Designers, Part 1: The Meaning of Color* [Online]. <http://smashingmagazine.com>: Smashing Magazine. Available: <http://www.smashingmagazine.com/2010/01/28/color-theory-for-designers-part-1-the-meaning-of-color/> [Accessed April 10 2012].
- CHERRY, K. *Color Psychology: How Colors Impact Moods, Feelings, and Behaviors* [Online]. [www.about.com](http://www.about.com): About.com. Available: <http://psychology.about.com/od/sensationandperception/a/colorpsych.htm> [Accessed April 15 2012].
- CLARK, J. *Why do girls wear pink and boys wear blue?* [Online]. [www.howstuffworks.com](http://www.howstuffworks.com): HowStuffWorks, Inc. Available: <http://people.howstuffworks.com/gender-color.htm> [Accessed April 23 2012].
- EVANS, K. *How Do Bright Colors Appeal to Kids* [Online]. [www.ehow.com](http://www.ehow.com): eHow. Available: [http://www.ehow.com/how-does\\_5476948\\_do-bright-colors-appeal-kids.html](http://www.ehow.com/how-does_5476948_do-bright-colors-appeal-kids.html) [Accessed May 3 2012].
- HOURCADE, J. P. 2007. *Interaction Design and Children. Foundations and Trends in Human-Computer Interaction*. [www.divms.uiowa.edu](http://www.divms.uiowa.edu): University of Iowa.
- IBARRA, K. 2011. *Designing Apps for Kids* [Online]. [www.uxmag.com](http://www.uxmag.com): UX Magazine. Available: <http://uxmag.com/articles/designing-apps-for-kids> [Accessed April 10 2012].
- ITZKOVITCH, A. 2012. *Designing Experiences for Young Kids : Child Proofing your Application* [Online]. <http://uxmag.com>: UX Magazine. Available: <http://uxmag.com/articles/designing-experiences-for-young-kids> [Accessed April 10 2012].

- KIVI, R. *How Does Color Affect Learning?* [Online]. [www.ehow.com](http://www.ehow.com): eHow. Available: [http://www.ehow.com/how-does\\_4570805\\_color-affect-learning.html](http://www.ehow.com/how-does_4570805_color-affect-learning.html) [Accessed April 10 2012].
- MADIGAN, S. 2005. *Using Symbol Communication to support Pre-school Children with English as an Additional Language*. BA, The University of Warwick.
- NARANJO-BOCK, C. 2011a. *Effective Use of Color and Graphics in Applications for Children, Part I: Toddlers and Preschoolers* [Online]. [www.uxmatters.com](http://www.uxmatters.com): UX Matters. Available: <http://uxmatters.com/mt/archives/2011/10/effective-use-of-color-and-graphics-in-applications-for-children-part-i-toddlers-and-preschoolers.php> [Accessed April 4 2012].
- NARANJO-BOCK, C. 2011b. *Effective Use of Typography in Applications for Children* [Online]. [www.uxmatters.com](http://www.uxmatters.com): UX Matters. Available: <http://www.uxmatters.com/mt/archives/2011/06/effective-use-of-typography-in-applications-for-children-3.php> [Accessed April 10 2012].
- TUFFELMIRE, D. *Do Colors Affect Children's Learning?* [Online]. [www.eHow.com](http://www.ehow.com): eHow. Available: [http://www.ehow.com/facts\\_7358881\\_do-colors-affect-children\\_s-learning\\_.html](http://www.ehow.com/facts_7358881_do-colors-affect-children_s-learning_.html) [Accessed April 10 2012].
- VAN DER MERWE, R. 2012. *A Dad's Plea To Developers Of iPad Apps For Children* [Online]. [uxdesign.smashingmagazine.com](http://uxdesign.smashingmagazine.com): Smashing Magazine. Available: <http://uxdesign.smashingmagazine.com/2012/03/12/dads-plea-developers-ipad-apps-children/> [Accessed April 10 2012].