Lauri Lankinen

Animation guidelines for hybrid mobile applications

Metropolia University of Applied Sciences

Bachelor of Engineering

Media Technology

Thesis

25th April 2017

Author Title Number of Pages Date	Lauri Lankinen Animation guidelines for hybrid mobile applications 34 pages + 4 appendices 25 April 2017
Degree	Bachelor of Engineering
Degree Programme	Information and Communication Technology
Specialisation	Media Technology
Instructor	Merja Bauters, Researching Lecturer

This thesis focuses on the creation of the animation guidelines for flat design hybrid mobile applications (HMA). The purpose of this study is to create a guideline that can be used to create HMAs for both the iOS and Android platforms without compromising user experience (UX). The objective of this thesis was set because no guidelines for HMA animation exists.

The need for the guidelines is researched through a questionnaire on the end users' perception of the user interface (UI) animations in mobile applications. The results demonstrate that the user understanding of animations in mobile applications is insufficient. Furthermore, the results shed light on how difficult it is to find a mobile application that does not use any animations.

A comprehensive study is conducted into the major mobile platforms and their respective best practices for implementing animation. Based on the results of this thesis, a guideline document is formulated to aid the developers of HMAs. Furthermore, the guideline is tested with a simulated application that collects data enabling a comparison between the simulated HMA using animation according to the guideline and an application without animation.

In addition, based on the results of this thesis, a universal animation guideline in flat design HMAs is proposed. A usability study indicates that adhering to the guidelines results in an application that is 161% more user-friendly than an application that does not use animation. Moreover, the usability test in this thesis reveal that the application adhering to the guidelines is used 1.3 times longer. The results of this study demonstrate a significant correlation between better UX, the use of UI animation and the time spent on using an application. Therefore, the UX of the HMA can be enhanced by adhering to the guidelines devised in this thesis.

Keywords	Hybrid mobile application, HMA, animation, flat design

Tekijä Otsikko Sivumäärä Aika	Lauri Lankinen Animaatio-ohjeistus hybridisovelluksille 34 sivua + 4 liitettä 24.4.2017
Tutkinto-ohjelma	Insinööri (AMK)
Koulutusohjelma	Tieto- ja viestintätekniikka
Pääaine	Mediatekniikka
Ohjaaja	Tutkijaopettaja Merja Bauters

Insinöörityössä oli tavoitteena tuottaa ohjeistus animaatiokäytännöistä hybridisovelluksissa mobiilikäyttöjärjestelmillä. Mobiilisovelluskehittäjän kannalta tavoitteena oli luoda viitekehys, jonka pohjalta tuotetuilla animaatioilla voidaan kehittää hybridisovellus niin Android- kuin iOS-käyttöjärjestelmälle siten, että käytettävyys ei kärsi kummallakaan alustalla. Projektin tavoitteenasettelu pohjautui vallitsevaan tilanteeseen, jossa vastaavaa dokumentaatiota ei ollut saatavilla.

Insinöörityön tarvetta arvioitiin kyselytutkimuksella. Tutkimuksen tulokset osoittivat, että loppukäyttäjien ymmärrys animaatioiden yleisyydestä mobiilisovelluksissa on puutteellista. Vastaavasti kyselytutkimuksen tulokset osoittivat, että suurin osa mobiilisovelluksista hyödyntää animaatioita käyttökokemuksen parantamiseksi.

Insinöörityössä tutkittiin kahden merkittävimmän mobiilikäyttöjärjestelmän käytäntöjä animaatioiden suhteen, ja havaintojen pohjalta työstettiin yleisohjeistus hybridisovellusten kehittämistä varten. Tuotettu ohjeistus koestettiin luomalla simuloitu mobiilisovellus, jonka koekäytöstä kerätyn aineiston avulla analysoitiin animoidun sovellusversion käytettävyyttä verrattuna sovellukseen, joka ei hyödyntänyt animaatioita.

Insinöörityön lopputuloksena syntyi yleistasoinen ohjeistus animaatiokäytännöistä hybridimobiilisovelluksille. Ohjeistuksen mukaisesti tuotetun sovelluksen käytettävyysmittaus osoitti käyttökokemuksen olevan 161% parempi verrattuna vastaavaan sovellukseen ilman animaatioita. Erityisen kiinnostavaa on tulos, jonka perusteella animaatioita hyödyntävän sovelluksen parissa vietettiin 1.3 kertaa enemmän aikaa. Tulokset osoittavat, että sovelluksen parissa vietetyn ajan ja animaatioiden kautta parannetun käytettävyyden välillä on merkittävä korrelaatio. Toisin sanoen noudattamalla tuotettua ohjeistusta, voidaan animaatioilla parantaa hybridisovelluksen käyttökokemusta.

Avainsanat	Hybridimobiilisovellus, animointi, flat design

Sisällys

List of Abbreviations

1 Introduction 1

2 Background of hybrid mobile applications 2

2.1 Flat design 3
2.2 User experience guidelines on mobile device UI animations 4
2.3 Transition timing 5
2.4 Transition easing 6
2.5 Visual cues and error handling 10
2.6 First launch experience 11

3 Methods and materials 12

3.1 Experimental setup 15
3.2 Data collection and basis for analysis 22

4 Results 24

4.1 Created assets 24
4.2 Evaluation of the functionality of the HMA animation guidelines 26
4.3 Considerations 30

5 Conclusions 32

References 33

Appendices

Appendix 1: Questionnaire for animation awareness in mobile applications
Appendix 2: Results for Questionnaire in animation awareness in mobile applications
Appendix 3: Results for HMA animation guideline validation
Appendix 4: HMA animation guideline for flat design applications

List of Abbreviations

HMA	Hybrid Mobile Application. Mobile application developed in HTML + JavaScript packaged into a native application.
UI	User Interface. Graphical layout of elements for user to interact with application logic.
UX	User experience. Perceived experience when interacting with a device.
HTML	Hypertext Markup Language. Markup language for creating online content.
CSS	Cascading stylesheets. Stylesheet language for describing the presentation of an online document.
JS	JavaScript. Untyped, interpreted runtime language. Used for the creation of dynamic behaviour in online content.
PHP	PHP: Hypertext Pre-processor. Server side scripting language for processing data.
SQL	Structured Query Language. Language designed for communication against a relational database.

1 Introduction

Hybrid mobile application (HMA) is an efficient way of catering to multiple mobile platforms with the same code base and overall visual design. Packaging a JavaScript framework and necessary static assets means that a full web service can be run on a mobile device without the latency that is usually present when accessing a similar service online through a web browser. The packaged application can be distributed via application stores directly to user devices without the user ever knowing that their app is in reality a web browser loading local assets in an app-like manner.

HMAs have contributed to the development of flat design to a great degree. As hybrid implementations are meant to scale between different devices and multiple resolutions, many conventional approaches for creating user interface (UI) elements have shifted from fixed raster assets to better scaling solid colour fills and vector graphics. Gradients and rasterised textures have been flattened and shifted away from the skeuomorphic familiarity of everyday objects in order to implement better legibility and usability.

One of the biggest selling points for flat design is that it is platform agnostic. This is an imperative for the development of HMAs aimed for multiple devices where catering to just one platform specific UI guideline is not enough. While shared elements can be found between guidelines across different platforms, the developer of a HMA is often left without clear guidance on how to best implement flat design UI through animation. Furthermore, the developer often lacks sufficient guidance on how to handle state changes across a HMA to enable best possible user experience (UX).

The purpose of this thesis is to study and discuss the animation strategies on top of flat design in HMAs. The second chapter is a literature review providing theoretical background on existing guidelines and best practices. The chapter three describes the experimental setup for evaluating approaches for UI animations in flat design HMAs. The fourth chapter focuses in the gathered data and data analysis discussing results. Finally, the fifth chapter presents the conclusion on the topic of animation guidelines for HMA development.

2 Background of hybrid mobile applications

A HMA is a self-contained web site wrapped in a platform specific native code implementing a full screen web view. In contrast to a traditional hosted web site, HMA stores all static content locally minimising latency and providing application with access to device-specific interfaces, such as geolocation. In order to create a HMA, the static website content is packaged through an external framework, such as Apache Cordova, and compiled into a platform-specific distributable format [1,50].

Transitions between application states can be handled as different physical web pages on the device but this leads to undesired effects, such as screen flashing white when using a HMA [2,2876]. Often a JavaScript framework is implemented to drive a single page application so that a single HTML file can serve as a basis for multiple views and transitions between application states. There are many frameworks available but Angular from Google and React from Facebook are currently used in most of the applications.

The application interfaces between JavaScript and mobile device are provided through plugins that are written in native code for each platform [1,50]. For a HMA developer it means that it is sufficient to use the abstracted methods provided by plugins for each device-specific functionality from camera to geolocation.

In the creation of a HMA, a packaging framework, such as Apache Cordova, is responsible for software packaging and the interfacing of device capabilities [3]. Application logic implemented with the use of a JavaScript framework such as Angular builds on top of Cordova providing UI. It is worth noting that while Angular provides single page app functionality, it is not interested in overall UX [2,2876]. Scaffolding a HMA is easy from a technological standpoint but leaves multiple open questions when implemented for actual usage. The UI needs to be designed from the ground up and transitions and state management have to be handled in a manner that is pleasing for the end user.

2.1 Flat design

Flat design is a movement originating from the necessity for scalable applications and shared UIs between varying platforms. Implementing UI elements in solid colours means that slow-to-load raster images can be converted into HTML elements with the help of CSS fill colours that render instantaneously and scale across different resolutions. At the same time, flat design provides a design mentality to create interfaces with the aim for the best UX regardless of the platform [4,38]. Thus, the very same interface can be published on both Android and iOS devices as a HMA.

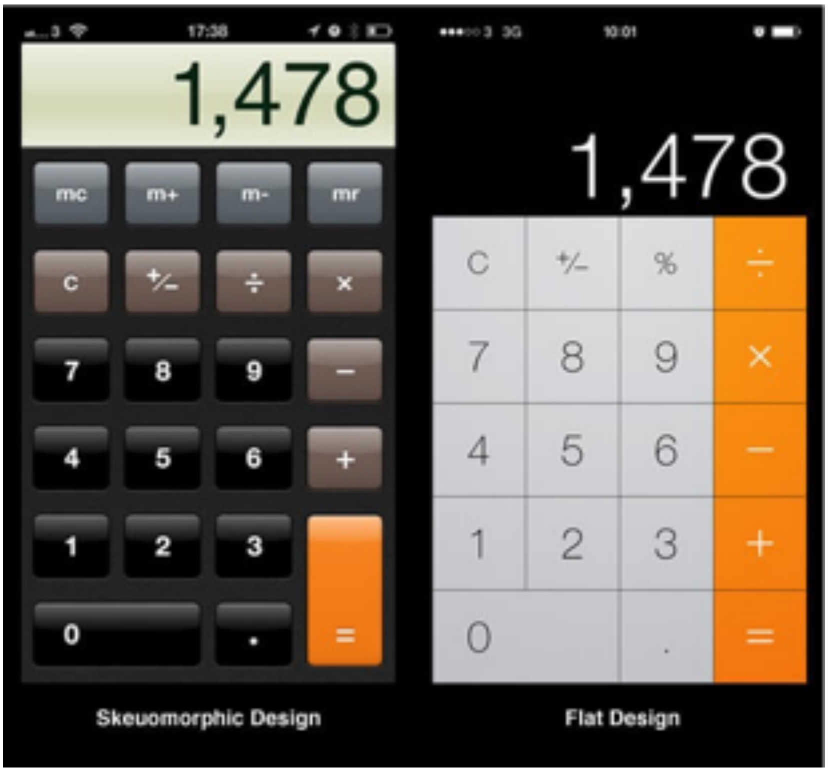

Figure 1. SKEUMORPHIC design in iOS 6 calculator app and flat design in iOS 7 calculator app reprinted from materialdesignblog.com [5]

Figure 1 shows on the left side how the calculator app in iOS 6 implements skeuomorphic design mimicking real life calculator with buttons and lcd screen. On the right side of figure 1 is the same application implemented in flat design inn iOS 7. This illustrates how minimalism is often associated with flat design: flat design aims for the emphasis of information through typography and flat colours [4,29]. Minimalistic elements are used when needed for functionality driving better UX due to greater legibility. Moreover, the minimalistic elements often produce a greater contrast in comparison to the interfaces with rasterized imagery and pseudo 3D elements [4,38].

It is worth noting that flat design does not provide guidelines for HMA transitions or UX. Flat design is a design language focusing on the purpose of creating highly scalable and functional UI elements with minimalistic approach. As such, flat design helps the HMA developer to create consistent UIs between different mobile platforms without touching the subject of UX on application transitions and animations.

2.2 User experience guidelines on mobile device UI animations

There are two main players in the mobile device platforms: Android by Google and iOS by Apple [6]. Both mobile operating systems have a strong emphasis on UX on their respective platforms. For this purpose, the platforms provide guidelines for UI elements and UX. While the HMA developer can and should choose the best practices from both, in many cases it is necessary to combine methodology from both in order to find a convenient middle course for catering same HMA on both platforms.

The Android and iOS platforms have both adopted the flat design system: first, Apple starting with iOS 7 in 2013 [7] and Google following with material design in 2014 [8]. While both of the platforms have their peculiarities in colour palettes and UI elements, they share many concepts closely. Both platforms have invested heavily in UI transitions and communication through animation. The chosen best practices for developer guidelines are mostly similar between Android and iOS.

Yet, there are no clear guidelines for implementing UI animations on HMAs that would target both platforms. The lack of clear guidelines is partly down to technological implementation which on the HMA side uses either JavaScript driven animation frames and transitions or transitions implemented through CSS styles [1,50]. While both of these methods provide granular control over animation, neither has any fixed default values that would ensure a great UX. Transition timing and easing need to be implemented per animation and, unlike native application, do not benefit from default practices set by platform.

The Basis for HMA animation and transitioning guidelines can be established by considering the standards set by the two main platforms and their respective documentation. The aim for this research is to highlight the current best practices for the creation of universal basis for HMA animation guidelines for HMAs that target both the iOS and Android mobile devices.

2.3 Transition timing

Animations are used in mobile devices to communicate application status and to provide feedback for user input. The animation of UI elements enhances UX by making the application more responsive compared to static UI implementation [9,3153]. Furthermore, animation allows more time for the application logic to run, thereby minimising the perceived waiting time for changes to take place.

Material design and iOS both recommend keeping animations subtle and fast enough. Animations, that run too slowly, create an overall feeling of an application running sluggishly and can easily deteriorate the UX. On the other hand, animations that run too fast are likely to be missed and, therefore, do not provide any improvement to the UX [10].

Correctly timed animations tie in with application flow and create sense of connection between the user and device [11]. If implemented correctly, subtle animations can create depth to flat design UI as demonstrated by iOS UI guideline's parallax scroll [11]. When foreground and background are scrolled in parallax style, different movement speeds between the flat design UI elements create a feeling of multiple layers sliding at varying depths. Same effect is implemented in material design by growing and shrinking drop shadow that is dependent on elements position on z-axis [12].

iOS UI guidelines provides creative freedom in animation timing allowing the developer to time transitions freely with any millisecond values. If omitted, iOS defaults to 350ms timing for full screen transitions or so called segues when navigating to and from between different views. Material design is more specific about timing: it provides different time limits for unique animation types to achieve the best UX as shown in table 1.

Table 1. Motion occurrence completion times. Data gathered from material.io [10]

Time for material design animation completion on mobile devices	
Large full screen transitions	> 375ms
Elements entering screen	> 225ms
Elements leaving screen	> 195ms
Motion should complete	< 400ms

Table 1 shows material design guidelines for timing motion occurrence completion. Full screen transitions should take at least 375 milliseconds and complete under 400 milliseconds. The latter 400ms limit being point where UI starts to feel unresponsive. Different UI elements have varying times for entering and leaving screen designed to catch and guide attention sufficiently. Elements that enter the screen should in general take more time compared to the elements that leave the screen. The difference in time shifts focus sufficiently for new items that should gain users' attention for relevant actions in UI. [10]

Material design guidelines also point out that motion on larger screens, such as tablets, should be extended by 30% in order to avoid excessively fast motion and the loss of user focus. Moving from tablets to desktop material design recommends using simpler animations with occurrence durations from 150ms to 200ms. On larger screens the abrupt animations and possible rendering issues will diminish UX. [10]

2.4 Transition easing

As outlined before, the animation of UI elements can enhance UX when implemented correctly. Lifelike and natural animations provide application with a flow that would be impossible to achieve with a static flat design UI [13]. Credibility and realism are keys in successful animation strategy on both platforms as dictated by guidelines. Animations should obey natural laws and provide flowing experience for user engagement. HMAs should implement physics based transitions wherever possible. Choosing correct easing strategies and mimicking physics-based behaviour provide best user engagement [11]. Transition easing methods for Material design are demonstrated in table 2.

Table 2. Material design easing methods. Data gathered from material.io [10]

Material design easing method and use		Easing type
Standard curve	**Growing and shrinking**	**ease in out**
Deceleration curve	**Elements entering screen**	**ease out**
Acceleration curve	**Elements leaving screen**	**ease in**
Sharp curve	**Elements that may return leaving screen**	**ease in out**

Both the iOS and Android platforms recommend the use of easing. Easing ensures that movement is natural and without mechanical characteristics ensuring that elements have tactile and lifelike properties [10]. Different variants for easing are used depending of use case. Table 2 illustrates easing methods and their most typical use case.

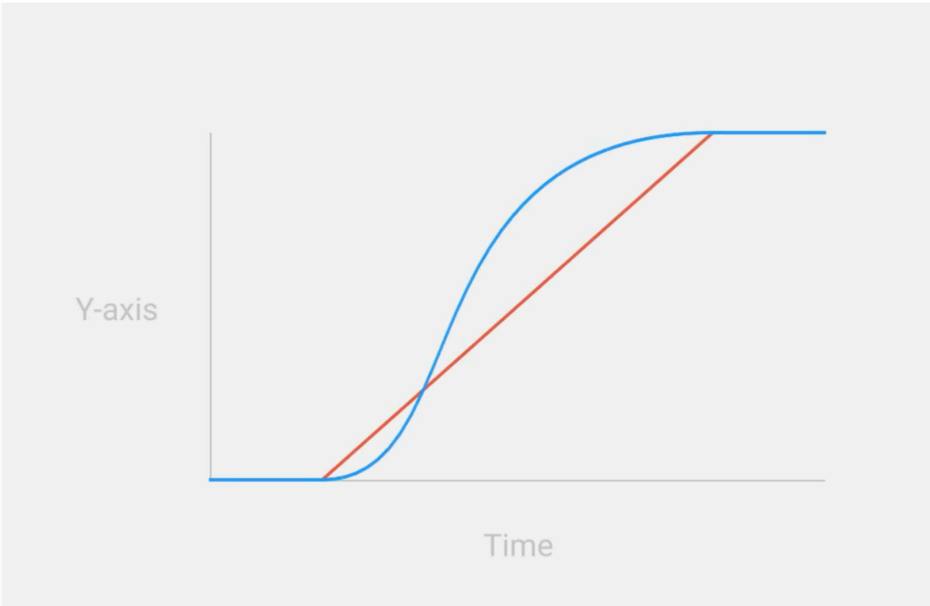

Figure 2. Standard curve easing and transition speed plotted in blue compared to flat animation speed plotted in red. Reprinted from material.io [10]

The standard curve with easing in and out as demonstrated in figure 2 is a typical method for scaling elements between different states. Quick acceleration followed by quick deceleration gives animated element more natural feeling in comparison to element with fixed transition speed [10]. Animating with standard curve brings element in to view in a fast pace and manages to slow down movement speed at the end of animation providing pleasing softness to the transition. The main point of the easing curve is to avoid sudden and unexpected stop of motion.

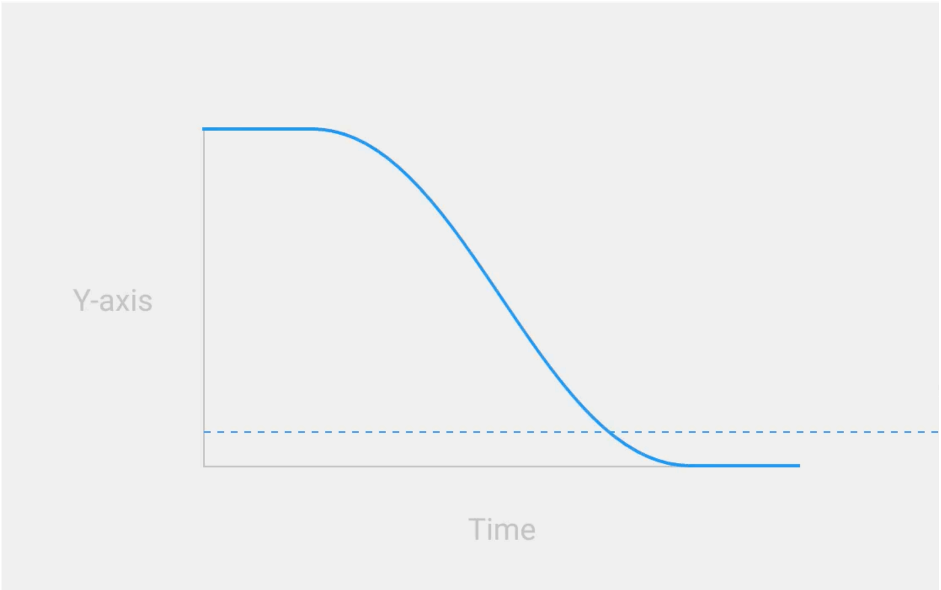

Figure 3. Sharp curve easing and y-axis position plot. Reprinted from material.io [10]

The sharp curve demonstrated in figure 3 is a variant of the standard curve with quicker animation completion time providing sharper feel [10]. The easing method works especially for elements that leave the screen of a device. Animating with sharp curve gives user a hint that the element could be brought back to screen with additional input. Therefore, sharp curve distinguishes itself from acceleration curve.

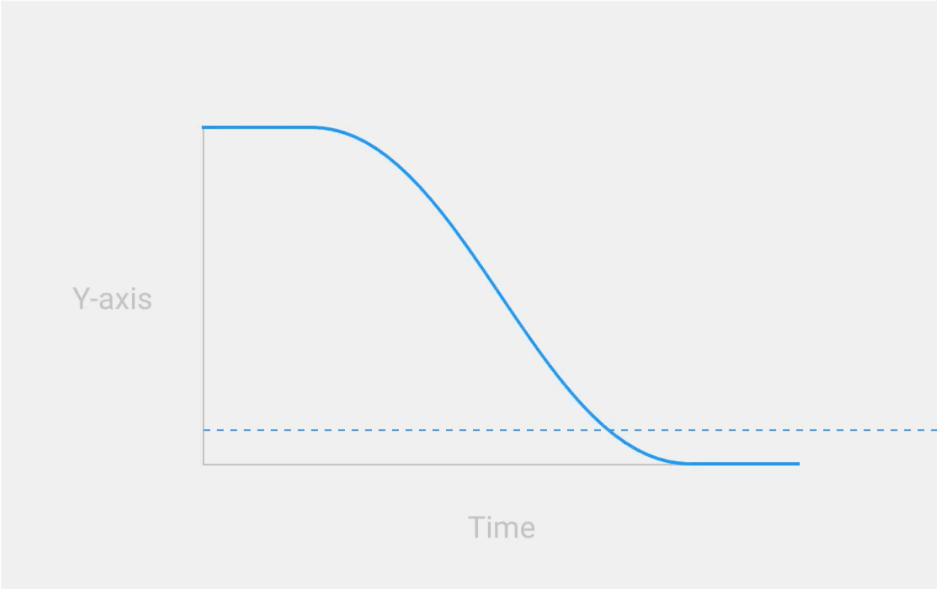

Figure 4. Acceleration curve easing and y-axis position plot. Reprinted from material.io [10]

The acceleration curve shown in figure 4 is useful for animating elements that are leaving the screen. Easing in provides a feeling of a more natural transition and helps avoid sudden movement speed at the start of animation [10]. Easing in can be considered as gravity taking hold of an object and accelerating it throughout the animation cycle.

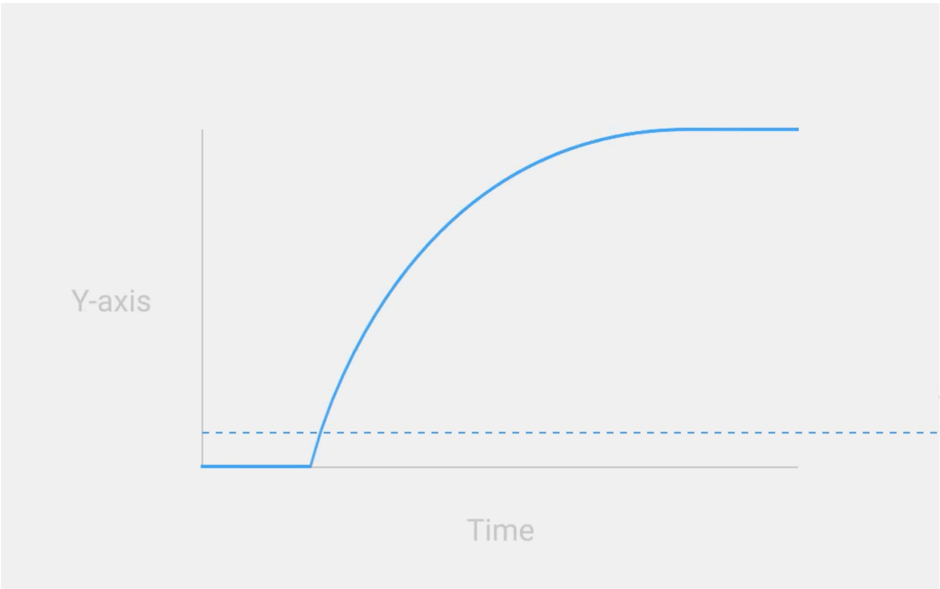

Figure 5. Deceleration curve easing and y-axis position plot. Reprinted from material.io [10]

Figure 5 demonstrates how the deceleration curve or easing out is the opposite implementation of the acceleration curve. The easing method helps element to come to a smooth halt at the end of animation. As such, the deceleration curve is perfectly suited for elements entering the mobile device screen [10].

2.5 Visual cues and error handling

Animation is a useful tool for connecting user with application UI providing feedback for user input. Animating UI elements is also an efficient way of getting users' attention to the desired parts of application highlighting currently relevant information [14,999]. This can be seen on both Material design and iOS platforms when calling for user input, such as answering a call. Transitioning UI to a dedicated view with animated UI elements provides the end user with understanding of current task being different from general use of the device. Gaining user attention can be implemented beneficially in those parts of the application where user input is taken.

UI animations and visual cues help to make an application feel more responsive. The visual cues are used on both platforms across UI at places where user input takes place. The main idea behind visual cues is to communicate erroneous and missing input data to guide the user throughout the process of providing required information [15]. The visual cues act as a transparent validation process right from the beginning of user input.

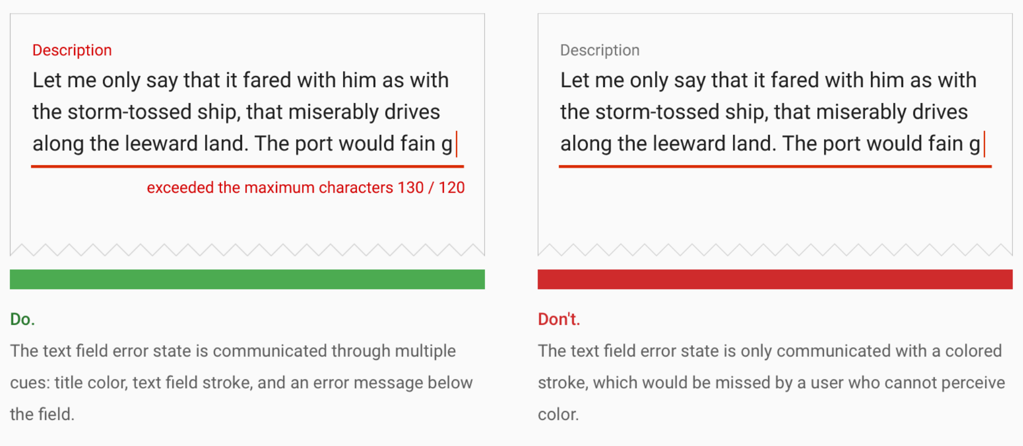

Figure 6. Usage of the visual cues on material design. Reprinted from material.io [15]

The left side of Figure 6 represents the correct usage of the visual cues on Material design to visualise input state where data exceeds the maximum allowed character count. On the right side of Figure 6, a similar case is displayed without the visual cues. The latter input is harder to troubleshoot and correct from the end user perspective.

The visual cues function as an animation method themselves updating the status of current input versus the state desired by application logic [15]. Visual cues show how small UI animations can provide valuable enhancements to UX when interacting with an application. Highlighting input sets that require user attention assist the user to correct input data faster compared to the fields without the visual cues [16,479]. Combining the visual cues with animation further enhance UX by guiding user attention to the right part of the flat design application.

2.6 First launch experience

Guidelines for iOS highlight first launch experience as a special case for user interaction. For the application to respond as quick as possible, the iOS UI guidelines advice the use of a start-up screen [17]. The start-up screen is a static image file that can be served immediately while waiting the application to load in the background. Providing instant feedback about application launch enhances UX making the application appear faster [18,153]. The feeling of immediate application responsiveness can be elaborated further by guiding the application to launch in correct orientation compared to the devices orientation. [17]

iOS UI guidelines also discourage the use of modals, menus and instructions for the sake of getting the end user to the application itself as fast as possible [17]. At the same time the applications are encouraged to anticipate the need for help and provide sufficient mechanisms to advice in situations where the end user is stuck. The application should implement the use of animations as a means to interactively guide the user through the application context [17].

Showing a start-up screen at launch gives the application more time to load in assets without the need to sacrifice perceived responsiveness from the user's perspective

[18,153]. The use of animated start-up screen is also becoming increasingly common in mobile devices to provide additional feeling of responsiveness.

3 Methods and materials

This chapter describes the setup and methods for measuring UX on flat design HMA transitions with and without animation. The aim of the experimental setup is to test the effect of UI animation on UX in flat design HMA. The chosen animation strategy for the test application builds on the top of the best practices found on both the iOS UI guidelines and Material design to validate a universal guideline for animations for future flat design HMA development.

Questionnaire

Motivation for HMA animation guideline creation was measured in the form of a questionnaire. The questionnaire (Appendix 1) was created to quantify user perception and understanding of the state of animation in mobile applications. The presumption for the creation of the HMA animation guidelines is that animations in mobile applications are so common that most users do not recognise the presence of the animated UI elements.

To further study and evaluate the presumption, the questionnaire consists of two main questions: the first question asks a participant to name an application on their mobile device that does not use animation. The second question asks for an application that does use animation.

The results are judged by comparing the questionnaire responses and whether applications brought up by the participants leverage animation as a part of UI. Multiple mobile applications use platform provided native state changes that are animated and, thus, it is difficult to find applications without any animated transitions.

Web application for data gathering and analysis

Validating the HMA animation guidelines requires testing an animated application against a control sample. The differences between the applications can be measured and quantified to validate the value of animation on overall UX. The HMA guidelines can be considered successful if subjective UX can be shown to be statistically better compared to application without animation.

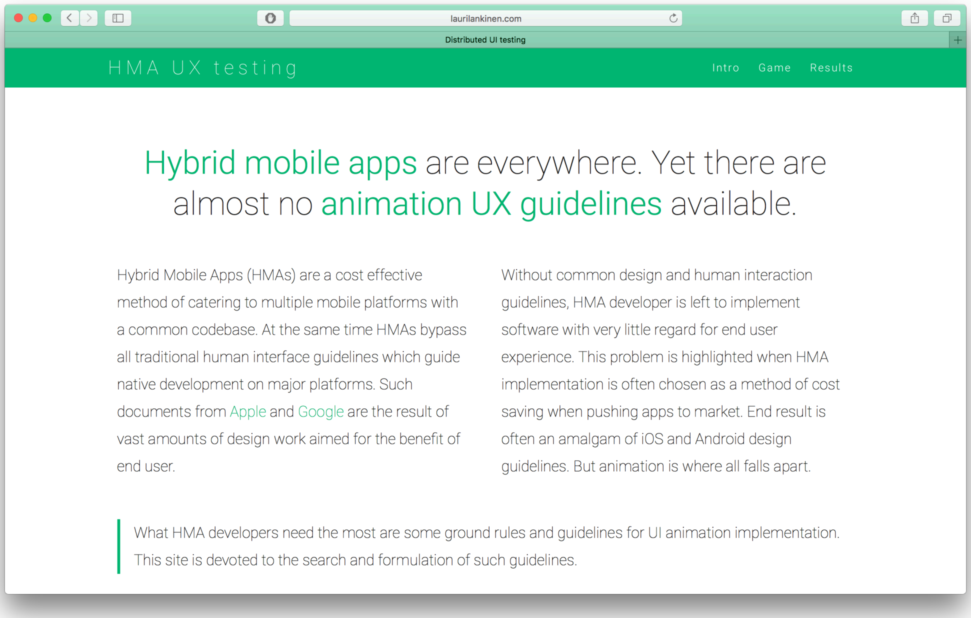

Figure 7. Website created for the HMA animation guidelines distribution and validation

The website shown in Figure 7 was created both to function as the source of the HMA animation guidelines as well as a data gathering tool. The website itself can be considered to consist of three unique parts: the first part of the website is an introduction to the state of the HMA animation guidelines. The second part of the website is a simulated HMA application used for collecting measurement data to validate the HMA animation guidelines. The third part of the website presents the results gathered from the simulated HMA and the final HMA animation guidelines. To validate the HMA animation guidelines, the following metrics were conducted: time, surviving erroneous input, user interaction and user satisfaction and feel of responsiveness.

Metrics

Measuring UX is a complicated task as the subjective experience with mobile devices and applications varies between users. Comparing the overall effect of the transitions should therefore be measured in concrete and comparable ways to ensure that correct conclusions can be drawn from the collected data. To minimise the influence of mobile platform on the test data, the measurements should be taken on a neutral platform. Furthermore, the studies show that the difference between laboratory and field testing are negligible when focusing on usability [19,14]. This motivates the creation of the criteria that can be evaluated programmatically in an online environment.

Time

Time is an efficient differentiating factor when measuring the overall capability of a user accomplishing a given task. Time is measured for two purposes: first, the interval between the start of an application state and the first user input and, second, the cumulative time user spent completing the task.

Comparing time between animated and non-animated flat design UI helps distinguish whether user attention can be guided sufficiently to minimise the accumulated time used to complete the task. While animations take time, it is assumed that implementing UI animations will help user to complete tasks faster than a non-animated application. Therefore, the overall UX is expected to be better for the HMA that use animations.

Surviving erroneous input

Most mobile applications rely on taking user input and processing it further. The input is usually required to meet certain criteria, such as length, format and styling. Handling user input validation is critical and, therefore, the UI needs to be able to distinguish correctly formatted input from erroneous input.

UI animation strategies are typically implemented to guide user attention towards erroneous input and to aid recovery from errors. Surviving errors is a great differentiating criterion between the animated and non-animated flat design application. Two aspects are measured: the number of errors the users made and how long it took for the users to input correctly formatted data to advance in the application.

User interaction

The UI animations should aid the user sufficiently through the first usage of the application to highlight the elements that provide user interaction. Thus, it is meaningful to track user interactions against the UI. Interactions over the UI can be listened for user inputs, such as clicks across the UI.

The differences in guidance generated by animation can be matched with the areas and number of clicks for each view generated by users when navigating in the application. The data from user interaction enables the comparison between possible different usage patterns in the software with and without animation implementation. The version of flat design HMA that can be navigated with the least user interactions should provide the best UX.

User satisfaction and feel of responsiveness

The usability and the UX of mobile applications are highly subjective. Therefore, it makes sense to measure the subjective satisfaction of the UI to enable comparisons between animated transitions and their non-animated static counterpart. The user satisfaction can be collected after application usage by asking the end user for feedback. The same can be done for the responsiveness of a simulated HMA. The answers can be used to deduct the differences between animation strategies to understand how the responsiveness varies between the animated and non-animated version.

3.1 Experimental setup

The setup for validation of the best practices between the animated and non-animated flat design HMA interfaces is done by creating a web application. Moving the test away from the mobile platforms minimises the difference caused by some users possibly running more capable hardware. In addition, the setup hides possible variances in how certain applications and design strategies are viewed across the different mobile platforms.

The web application environment also enables tracking of user inputs over the UI. With the collected data, the visualisation of usage flow can be recreated and compared between the animated and non-animated UIs.

Simulating a mobile device

The experimental setup needs to tie the context to a mobile device. Thus, a web page is designed with a discreet mobile phone visualisation taking the centre place of the screen. The virtual mobile device is then scaled according to users screen to ensure that the end user does not ever need to scroll vertically when participating in the test.

The virtual mobile device also functions as a wrapper for the simulated HMA. The transitions between the application states are limited to the virtual device screen and appear as on an actual device. The chosen implementation enables to model the animation for elements entering and leaving the screen.

Figure 8. Simulation of a mobile device with flat design UI

Figure 8 shows an example of the simulated mobile device running a flat design UI. Limiting the content and transitions within the designated screen area, the mobile device usage can be simulated in a normal web browser environment. The simulated phone is scaled to fit different resolutions but limited to the maximum size to reflect the physical dimensions of a generic mobile device.

Simulating a HMA

The software running within the simulated mobile device is implemented in HTML5 for the UI elements and additional styling is provided by CSS3. The application logic and user tracking are constructed in JavaScript with additional DOM manipulation leveraged by the jQuery library. jQuery is also used to initiate the animations with built-in functionality for animating 2D transitions in the browser as well as manipulating the element styling classes for the CSS based animations. An external library called Materialize.css is used in the test setup to implement the visual cues as a part of the animation strategy. The library aids the generation of input related guidance mimicking the functionality found in Material design.

The tracking logic is written in JavaScript and listens to user input building a structured JSON object from user interaction with the application for each different view. The coordinates for each click across the UI are recorded and pushed into the JSON object for further analysis. The input events are tracked and the view transitions initiated when sufficient structure and format for the desired input elements are provided.

Structure of the simulated application

The simulated application is aimed for the first-time interaction with the end user. The first-time interaction provides opportunity to measure the value of the UI animations and compare their meaningfulness in the context of UX. To ensure the most relevant differentiation in the UX between the animated and non-animated counterparts, the application focuses on the most tedious processes of the application adoption, such as the register and login functionalities. At launch, the application randomly sets animation either to a default described in the HMA animation guidelines, slow animation or off for the whole duration of the application usage. The data is saved as a part of the usage data within the client side session.

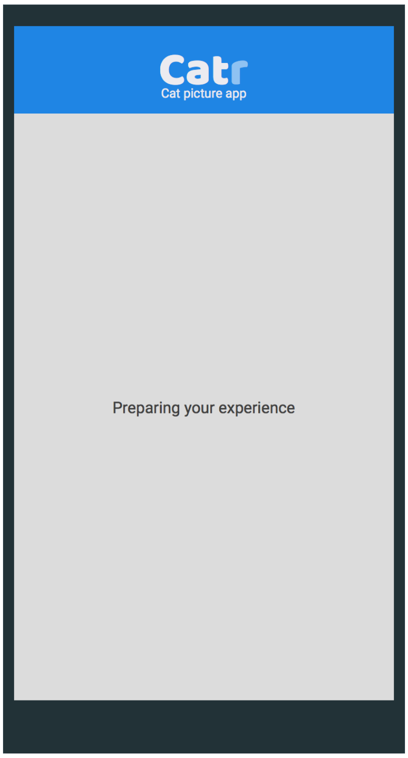

Figure 9. Launch image of the simulated HMA

Figure 9 shows the first view that the user encounters when interacting with the simulated HMA. The view is timed to last for 2500ms before automatically moving the application state to the register view. The main purpose of the view in Figure 9 is to simulate loading of the mobile application to create a notion of loading complex application logic and building up user expectations. Furthermore, the effects of the animation on the end user experience are set at this point by providing an animated loading indicator when animations are used in the simulated HMA and displaying a static text content when animations are not used.

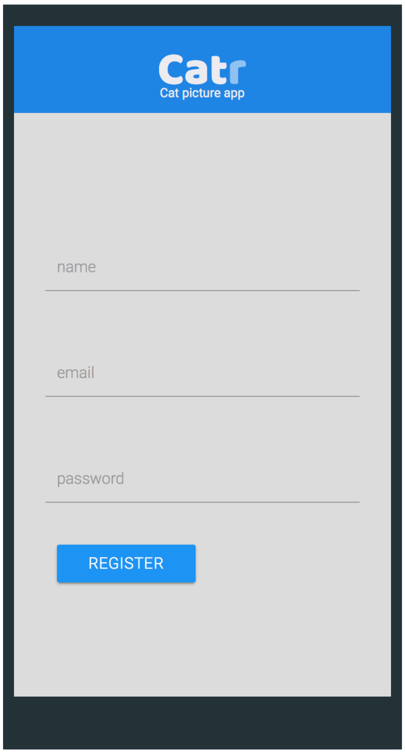

Figure 10. Register view of the simulated HMA

Starting the application with the register process is not the best practise according to the iOS guidelines [17]. However, many services still require the user to create an account before starting full usage of the application. Creating an account means that the end user is required to provide multiple textual inputs for data, such as name, email and password. With the required fields the first actual test cycle is the register view as shown in Figure 10. The view tests how well the user can recover from an erroneous input when incorrect data is inserted. The animation strategy is to highlight the erroneous data as well as to guide to the next input field when correct data is entered. The difference between the animated and non-animated UI is distinguished by the accumulated time and the number of errors on the UI.

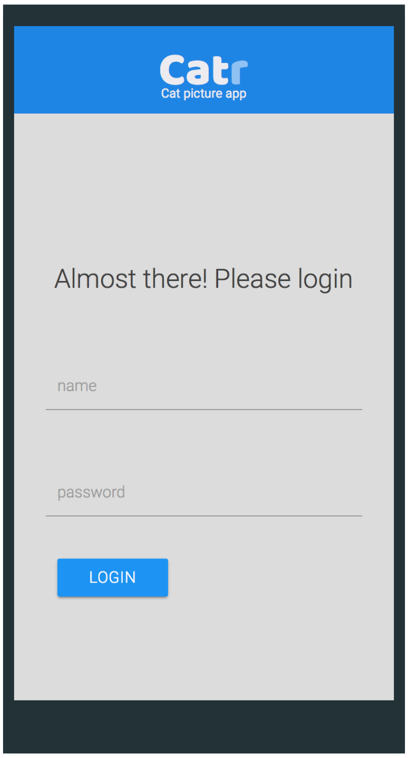

Figure 11. Login view of the simulated HMA

Figure 11 shows the login view of the second test cycle for the application logic. The login view resembles the register view by asking username and password. Only subtle differences can be seen between these first views that enable the measurement of state awareness in UX. The transitions that happen instantaneously might be missed if the user is not paying enough attention to the UI elements. The transition animations are supposed to highlight the changes in application state, thereby, lowering the time required for completing the login. The variation between the animated and non-animated UI is measured in terms of time waited for the first user input.

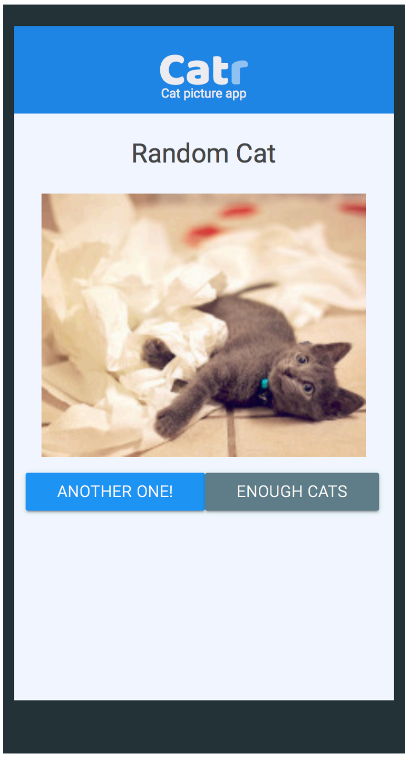

Figure 12. Main application view of the simulated HMA

The Figure 12 shows the main application view. The third test cycle for the application consists of a state in which UX is measured against the overall time spent and the overall amount of media consumed. The assumption is that a better UX leads to a heavier usage of the main application logic, hence, further enhancing UX in a closed loop. The main application view also demonstrates the visualisation of asynchronous loading with an animated loader when the simulated HMA uses animation. The animated loader is expected to provide better more responsive usage experience when additional media content is loaded.

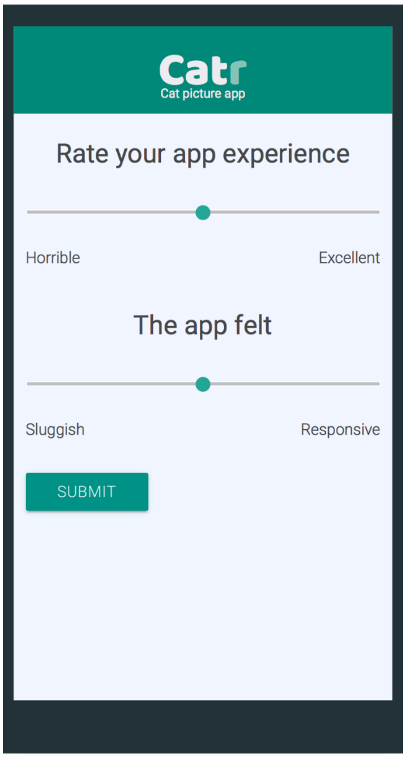

Figure 13. Rating view of the simulated HMA

The last state for the simulated HMA usage is the rating view displayed in Figure 13. The rating view asks the end user to rate subjectively the whole experience of the app usage on a scale from 0 to 7. Furthermore, the same scale is used when prompting the end user for the feeling of responsiveness throughout the application state transitions and asynchronous media loading. Upon submitting the ratings, the end user is provided with an opportunity to see the accumulated results of all the tests.

3.2 Data collection and basis for analysis

The user specific usage data is collected in JSON format and posted to the backend server implemented in PHP and SQL. The JSON data is posted to the backend after a successful test run of the simulated HMA. Each test cycle of the simulated HMA updates locally stored JSON object for gradual modelling the UX and application flow. Progressive updating of the data model was evaluated but dismissed as an update strategy in favour for generating a full dataset that provides with comparable results for user feedback and UX evaluation. The PHP backend implements functionality for querying the aggregated results to show a comparison between the animated and non-animated use cases.

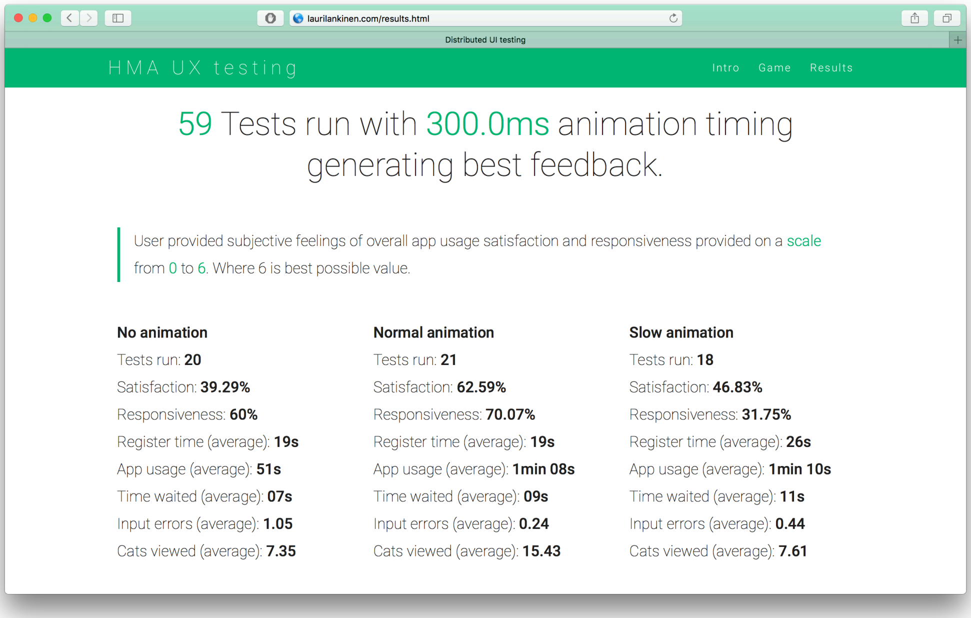

Figure 14. Result page for tests run to validate the HMA animation guidelines

The created web service provides a result page as shown in Figure 14 for the comparison of the accumulated test results. The result page is shown after completing the experimental application. The webpage was implemented to motivate each test user to further study animation possibilities in HMAs. Comparing the user interactions and media consumption between the animated and non-animated test application functions as a stark reminder on the importance of UX.

Each view for the test application is shown side by side on the results page comparing user interaction on a heat map that visualises clicks on the UI for the variants of the simulated HMA with and without animation. The visualisation helps differentiate the usage pattern characteristics on the HMA UI with and without the animation provided guidance and transitions.

Each view is available for comparison with data, such as, the total number of test cycles, average completion time, number of user input errors and amount of media consumed.

Thus, comparing the versions of the HMA show in a glance the distinction generated by the UI animations on flat design application.

4 Results

This chapter investigates the motivation for the creation of the HMA animation guideline by analysing the user questionnaire found in Appendix 1. Furthermore, this chapter evaluates the data gathered from the user testing and discusses the benefits of animation driven UX. The UX of the simulated HMA that adheres to the HMA UI animation guideline is compared to the application without animation. The results of the UX and the perceived responsiveness are used to validate the HMA animation guidelines for flat design applications both for the iOS and Android platforms.

4.1 Created assets

Multiple assets were created to validate the benefits of the UI animation guidelines on the flat design HMAs. First, a study was conducted into the best prevailing practices on the existing major platforms to establish a basis for the HMA animation guidelines. Second, a questionnaire was conducted to measure user awareness of animation in mobile applications. Finally, a website was created to inform the users on the importance of UI animation. The website also gathered and analysed data to differentiate the UX between the HMA with and without the HMA animation guidelines.

User awareness of animation in mobile applications

The volunteers participating in the questionnaire were asked to list an application on their mobile device that does not use animation. The question was abstract on purpose so that the participants would clearly define what is perceived as a UI animation. The results of the question are presented in Table 3.

Table 3. The results of the question that asked to name an application that does not use animation.

One app on my phone that does not use animation	Animated
Easyark	Yes
Nordea bank app	Yes
Facebook	Yes
Untappd	Yes
Calculator	Yes
Calculator	Yes
N/a	-
WhatsApp	Yes
WhatsApp	Yes

Table 3 presents the answers provided by the participants when asked about applications that do not use animation. However, a further study on the results indicates that all the applications do implement animations to a varying degree. Even the calculator applications take advantage of animations to denote interaction on user input. Also, the messaging and the payment applications shown in Table 3 benefit from animated transitions between application states.

Table 4. The applications that use animation according to the questionnaire respondents.

One app on my phone that does not use animation	Animated
Facebook	Yes
Facebook	Yes
Puniverse	Yes
New star soccer	Yes
Kindle	Yes
iBooks	Yes
Facebook messenger	Yes
S-mobiili	Yes
Timely	Yes

Table 4 presents the applications that use animation according to the respondents. Most of the applications named by the respondents use animation to emphasize interaction with the UI. The applications, such as iBooks and Kindle, simulate flip of a page mimicking the activity of reading a book. The games, such as New star soccer, and the media applications, such as Puniverse, depend on the animation as their core logic. These results show how the respondents discern animation only when it truly stands out from the platform provided defaults.

In conclusion, the results gathered from the questionnaire indicate that the participants take animations in the mobile applications for granted and have grown accustomed with the animated transitions between the application states. The results further motivate the creation of the HMA animation guidelines to ensure that hybrid applications cater to the end user expectations.

4.2 Evaluation of the functionality of the HMA animation guidelines

The simulated HMA tests received 59 responses. The test navigated through three different versions of the simulated application: the first version without animation, the second with the HMA guideline animations and the third with extended animation completion time. The results gathered from the user tests indicate that the average satisfaction and responsiveness was highest when adhering to the HMA animation guidelines as illustrated in Table 5.

Table 5. The results of the HMA simulation with and without animation.

	No animation	Guideline animation	Slow animation
Tests run	20	21	18
User satisfaction	39 %	63 %	47 %
Responsiveness	60 %	70 %	32 %
Images viewed (average)	7.35	15.43	7.61

The number of the responses is fairly even between the three versions of the simulated HMA as shown in Table 5. Based on the results, certain trends can be distinguished in the accumulated results. The user satisfaction is 1.6 times higher on the HMA version adhering to the HMA animation guidelines in comparison to the version without transition animations. Furthermore, the media consumption is more than twice the amount on the simulated application that was developed with the HMA animation guidelines when compared to both the non-animated and the slow-animated versions. The same findings are reflected in the perceived responsiveness. With fast enough animations, the HMA version following the animation guidelines results in the increase of 15% in the perceived responsiveness over non-animated application. The increase in responsiveness is experienced even though the users spend more time waiting as shown in Table 6.

Table 6. Results of the timing and errors in the HMA simulation with and without animation.

	No animation	Guideline animation	Slow animation
Register time (average)	19s	19s	26s
Usage time (average)	51s	1min 8s	1min 10s
Time waited (average)	7s	9s	11s
Input errors (average)	1.05	0.24	0.44

The average time for the user to successfully complete the register process takes 19 seconds on all versions except the slow animation version as displayed in Table 6. The non-animated version more than doubles the number of errors in input data in comparison to the slow version of the animated HMA. The application version following the animation guidelines was completed with the least number of errors on average, indicating a positive correlation of animation and visual cues in comparison to the simulated application without animation and visual cues.

The comparison of the results in Table 5 and Table 6 indicates how implementing animation according to the HMA animation guidelines has tangible benefits over applications without animation. Therefore, it can be concluded that with the accumulated statistical data, the HMA animation guidelines are validated to be functional and preferable over non-animated HMAs. Adhering to the HMA animation guidelines enhances application responsiveness and overall user satisfaction while at the same time minimising the number of input errors.

Comparing user interactions

Part of the simulated HMA tracking logic is to save the user input coordinates for further analysis. Presumption is that the usage patterns may differ between the animated and non-animated application versions. The end results of user input are visualized as heat maps (Figure 15) and show only minor usage pattern differences in the register view.

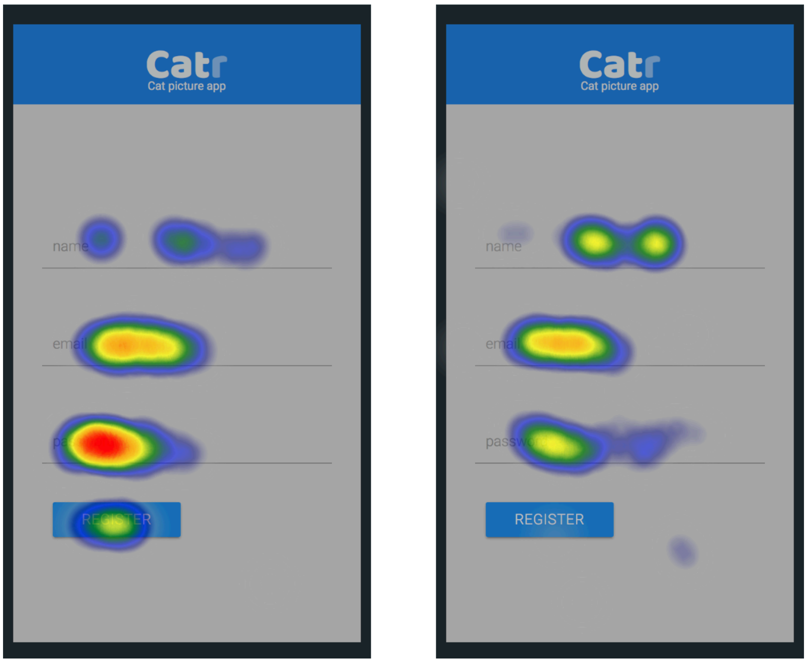

Figure 15. The register view showing user interactions as a heat map: the heat map on the left is without animation whereas the heat map on the right uses the HMA guideline specified animation.

The usage patterns visualized in the heat map format show clustering of the user interactions as displayed in Figure 15. The areas coloured red have the largest number of interactions with density of the interactions diminishing from yellow to green and finally to blue. While the HMA version with animation on the right-hand side in Figure 15 has more distribution amongst the user input, the interaction areas remain roughly the same. The same effect can be observed in the login view in Figure 16.

Figure 16. The login view showing user interactions as a heat map: the heat map on the left is without animation whereas the heat map on the right uses the HMA guideline specified animation.

The visualisation of the user interaction in the login view shows that similar interaction areas can be found on both the non-animated and animated versions in Figure 16. The animated version shows larger deviation from neat clustering with some interactions outside of the clusters when compared to the non-animated version. It seems that the user interaction in the password field is more divided on the animated version of the simulated HMA.

Figure 17. The main view showing user interactions as a heat map: the heat map on the left is without animation whereas the heat map on the right uses the HMA guideline specified animation.

The interactions in the main view are presented in heat map in Figure 17. Both the non-animated and the animated versions of the HMA show neat clustering of user interaction. The large cluster below the first UI button displays a development issue in tracking logic pertaining only for this view. Due to the positioning in CSS, the user input tracked on the second button on the right-hand side of the UI is registered as occurring below the first button. The issue has negligible effect on the results. Furthermore, the heat map on the right-hand side in Figure 17 shows how the animated version of the main view attracted nearly twice as many interactions for downloading additional media content than the non-animated version.

4.3 Considerations

The sample size of the HMA animation guideline validation test consists of 59 respondents. The sample size is decent considering the timeframe of one week of gathering user data online. It is worth noting that operating in an online environment creates a challenge in data collection. Designing sufficient tracking logic is tedious and, once deployed, difficult to change. In addition, the variations between the end users' devices and software should be taken into consideration beforehand.

No discernible source of errors can be distinguished in the devised test setup. A minor bug concerning click event coordinates was discovered in the tracking process of the simulated HMA experience. Some users participated with devices that support touch events and, therefore, as the simulation was developed for the click events, the tracking logic could not handle user interaction coordinates for these devices. The touch enabled devices reported the coordinates in a null format that does not affect the results.

An advantage in the online-based validation of the UI interaction strategies is that such data gathering can be extended. The sample size can be expanded to a further study: it is also possible to introduce new paradigms and change the emphasis of the measurements to focus on other test-specific aspects. For example, the current test framework allows further examination to study the effects of optimistic UI over plain animations in flat design HMA development.

5 Conclusions

The purpose of this thesis was to create and validate the best practices for UI animation on flat design HMAs targeting both the iOS and Android platforms. The guidelines for HMA animation were created through the study of the prevailing best practices on major mobile platforms that enable mobile developers to efficiently adopt animation strategies. Efficient adoption of animation is especially important because UI animations are taken for granted as shown in the questionnaire conducted in this thesis to measure the user perception for the state of animation in mobile applications.

The HMA animation guidelines were tested as a simulated mobile application on a website. The website was created for the analysis of the usage patterns and subjective preferences between an application adhering to the HMA animation guidelines and a non-animated version. The results show that user satisfaction was 1.6 times higher for the version adhering to the HMA animation guidelines in comparison to the non-animated version. Furthermore, the study found out that when adhering to the guidelines, the users consumed more than twice the amount of media. The results also show that following the HMA guidelines enhances the application responsiveness by 15% in this study

For HMA development, the implications of this study are distinct: developing HMAs without consideration for the UI animation will lead to a weaker overall satisfaction of application and diminished usage in comparison to the HMAs that adopt the best practices of the HMA animation guidelines. Therefore, the adoption of the HMA animation guidelines is recommended to enhance UX of mobile applications.

References

1 Andre Charland, Brian Leroux. Mobile application development: web vs. native. Communications of the ACM 2011;54(5):49-53.

2 Madhuri A. Jadhav, Balkrishna R. Sawant, Anushree Deshmukh. Single Page Application using AngularJS. International Journal of Computer Science and Information Technologies 2015,6(3):2876-2879

3 Stefan Bosnic, Ištvan Papp, Sebastian Novak. (2016, November 11th) The development of hybrid mobile applications with Apache Cordova. Paper presented at Telecommunications Forum (TELFOR), Belgrade, Serbia. doi:10.1109/TELFOR.2016.7818919

4 Sian Morson. iOS 7 and Flat Design. CA, USA, Apress; 2014.

5 Graeme Fulton. Google’s Material Design Language Can Inject Life into Your Boring Website [online]. Material Design Blog; 2015. URL: http://materialdesignblog.com/googles-material-design-language-can-inject-life-into-your-boring-website. Accessed 14 April 2017.

6 Anshul Gupta, Gartner Says Worldwide Sales of Smartphones Grew 7 Percent in the Fourth Quarter of 2016 [online]. Gartner; 2017. URL: http://www.gartner.com/newsroom/id/3609817. Accessed 14 April 2017.

7 Haris Bacic, Apple Is Embracing The Flat Design Trend - Are You? [online]. Forbes; 2013. URL: https://www.forbes.com/sites/allbusiness/2013/09/09/apple-is-embracing-the-flat-design-trend-are-you

8 Aaron Souppouris, Material Design is Google's new visual look for Android, Chrome OS, and more [online]. The Verge; 2014. URL: http://www.theverge.com/2014/6/25/5841044/material-design-new-google-interface-2014

9 Benedikt Merz, Klaus Opwis, Alexandre N. Tuch. (2016, May 12th) Perceived User Experience of Animated Transitions in Mobile User Interfaces. CHI EA '16 Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, California, USA. doi:10.1145/2851581.2892489

10 Material design guidelines. Duration & easing [online]. Material Design guidelines. URL: https://material.io/guidelines/motion/duration-easing.html. Accessed 14 April 2017.

11 iOS design guidelines. Visual design [online]. iOS human interface guidelines. URL: https://developer.apple.com/ios/human-interface-guidelines/visual-design/animation. Accessed 14 April 2017.

12 Material design guidelines. Elevation & shadows [online]. Material Design guidelines. URL: https://material.io/guidelines/material-design/elevation-shadows.html. Accessed 14 April 2017.

13 Material design guidelines. Material motion [online]. Material Design guidelines. URL: https://material.io/guidelines/motion/material-motion.html. Accessed 14 April 2017.

14 Steven L. Franconeri, Daniel J. Simons. Moving and looming stimuli capture attention. Perception & Psychophysics 2003;65(7):999-1010.

15 Material design guidelines. Usability - Accessibility [online]. Material Design guidelines. URL: https://material.io/guidelines/usability/accessibility.html. Accessed 15 April 2017.

16 Keiji Ogata, Yasuhiro Seya, Katsumi Watanabe, Tohru Ifukube. Effects of visual cues on the complicated search task. NordiCHI '12 Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design 2012:478-485.

17 iOS design guidelines. First Launch Experience [online]. iOS human interface guidelines. URL: https://developer.apple.com/ios/human-interface-guidelines/interaction/first-launch-experience. Accessed 15 April 2017.

18 Fatih Nayebi, Jean-Marc Desharnais, Alain Abran. (2013, Oct 26th) Perceived User Experience of Animated Transitions in Mobile User Interfaces. 2013 Joint Conference of the 23rd International Workshop on Software Measurement and the 8th International Conference on Software Process and Product Measurement, Ankara, Turkey. doi:10.1109/IWSM-Mensura.2013.30

19 Anne Kaikkonen, Aki Kekäläinen, Mihael Cankar, Titti Kallio, Anu Kankainen. Usability testing of mobile applications: a comparison between laboratory and field testing. Journal of Usability Studies 2005;1(1):4-16.

Appendix 1: Questionnaire for animation awareness in mobile applications

Animation in mobile apps

Questionnaire about animation awareness in mobile apps

*Required

1. **My mobile phone platform is** *
   Choose one
   - Android
   - iOS
   - Windows Phone
   - Muu: __________

2. **One app (name) on my phone that is not animated**

3. **One app (name) on my phone that uses animation**

4. **Gender** *
   Choose one
   - Man
   - Woman

5. **Age**

Appendix 2: Results for Questionnaire in animation awareness in mobile applications

My platform	App without animation	Animated	App that uses animation	Animated	Gender	Age
iOS	Easypark	Yes	Facebook	Yes	Man	58
Android	Nordea bank app	Yes	Facebook	Yes	Man	33
iOS	Facebook	Yes	Puniverse	Yes	Woman	59
Android	Untappd	Yes	New star soccer	Yes	Man	33
Android	Calculator	Yes	Kindle	Yes	Man	32
iOS	Calculator	Yes	iBooks	Yes	Man	46
Android	N/a	-	Facebook messenger	Yes	Woman	27
Android	WhatsApp	Yes	S-mobiili	Yes	Woman	24
Android	WhatsApp	Yes	Timely	Yes	Man	24

Appendix 3: Results for HMA animation guideline validation

id	anim-Length	responsive-ness	satisfac-tion	duration (ms)	wait (ms)
46	0.0	4.0	4.0	19434.0	4050.0
48	0.0	4.0	1.0	73763.0	6822.0
50	0.0	5.0	3.0	53710.0	4230.0
52	0.0	4.0	1.0	22722.0	4620.0
63	0.0	4.0	2.0	87375.0	19008.0
64	0.0	5.0	4.0	40007.0	3291.0
66	0.0	5.0	0.0	55541.0	3771.0
72	0.0	0.0	0.0	109033.0	7782.0
75	0.0	5.0	2.0	30148.0	5316.0
76	0.0	4.0	4.0	44539.0	3678.0
79	0.0	4.0	4.0	39093.0	3186.0
80	0.0	5.0	4.0	33343.0	4482.0
83	0.0	5.0	3.0	29932.0	4251.0
86	0.0	5.0	4.0	36686.0	9255.0
87	0.0	5.0	2.0	60067.0	10905.0
90	0.0	4.0	4.0	45675.0	9891.0
91	0.0	4.0	3.0	40530.0	6969.0
92	0.0	2.0	4.0	109834.0	6564.0
95	0.0	5.0	2.0	59111.0	13185.0
97	0.0	5.0	4.0	24788.0	3777.0
42	1250.0	1.0	5.0	192972.0	7578.0
44	1250.0	1.0	4.0	171786.0	50412.0
47	1250.0	2.0	3.0	109483.0	13710.0
54	1250.0	1.0	2.0	22768.0	4821.0
56	1250.0	1.0	2.0	32570.0	5202.0
57	1250.0	2.0	2.0	34394.0	6273.0
61	1250.0	3.0	3.0	46629.0	17232.0
62	1250.0	4.0	5.0	82354.0	6360.0
68	1250.0	2.0	1.0	39504.0	8847.0
69	1250.0	6.0	4.0	52131.0	8001.0
71	1250.0	5.0	3.0	149279.0	25434.0
73	1250.0	1.0	5.0	41528.0	8364.0
74	1250.0	1.0	4.0	47228.0	5826.0
77	1250.0	1.0	3.0	42306.0	5199.0
84	1250.0	1.0	3.0	32340.0	5658.0
88	1250.0	1.0	1.0	32809.0	5064.0
93	1250.0	5.0	6.0	72328.0	6222.0
94	1250.0	2.0	3.0	55256.0	7011.0

39	300.0	5.0	5.0	49251.0	4584.0
40	300.0	5.0	5.0	27536.0	4287.0
41	300.0	5.0	5.0	34503.0	5100.0
43	300.0	3.0	4.0	121846.0	7032.0
45	300.0	5.0	5.0	113161.0	6201.0
49	300.0	2.0	4.0	111024.0	7413.0
51	300.0	6.0	5.0	64313.0	56985.0
53	300.0	5.0	5.0	25585.0	4242.0
55	300.0	5.0	5.0	39308.0	5097.0
58	300.0	6.0	6.0	95151.0	15807.0
59	300.0	4.0	2.0	88655.0	6690.0
60	300.0	4.0	1.0	47041.0	7002.0
65	300.0	5.0	4.0	66575.0	9099.0
67	300.0	4.0	2.0	64528.0	4800.0
70	300.0	4.0	4.0	72484.0	8592.0
78	300.0	6.0	6.0	66498.0	4761.0
81	300.0	6.0	5.0	62333.0	3771.0
82	300.0	6.0	5.0	96578.0	6294.0
85	300.0	5.0	4.0	47918.0	5826.0
89	300.0	6.0	5.0	51728.0	4146.0
96	300.0	6.0	5.0	71541.0	5052.0

Appendix 4: HMA animation guideline for flat design applications

Animation Guidelines

for Hybrid Mobile Applications

Animation timing

Time in which animation occurrence should complete

Phone	Tablet	Laptop
350ms	450ms	150ms

Animation easing

Various animations benefit from easing whether animation is used for full screen transition or elements entering / leaving screen

Transition	Enter	Leave
ease in-out	ease out	ease in

Visual cues

User input should be validated proactively as input happens. Guiding user attention to misshaps creates better user experience.

Do	Do not
validate input data proactively and aid error correction	force user to wait for validation after full data input of a form

Asynchronous loading

Visualise process of asynchronous loading

Do	Do not
clearly visualise process of loading assets	force user to wait with inert UI while loading

Launch experience

Provide user with immediate feedback

Do	Do not
use launch image or preloader and animate content occurrence	force user to wait full load of application before showing UI

Easing in css

Ready to use transition styles

ease in-out

transition-timing-function: **cubic-bezier(0.4, 0.0, 0.2, 1);**

ease out

transition-timing-function: **cubic-bezier(0.0, 0.0, 0.2, 1);**

ease in

transition-timing-function: **cubic-bezier(0.4, 0.0, 1, 1);**

Metropolia