

MiniDJ – utveckling
av en interaktiv ljudapplikation
för iPhone ur ett MDI-perspektiv

MARTIN HWASSER



**KTH Datavetenskap
och kommunikation**

MiniDJ – utveckling av en interaktiv ljudapplikation för iPhone ur ett MDI-perspektiv

M A R T I N H W A S S E R

Examensarbete i datalogi om 15 högskolepoäng
vid Programmet för datateknik
Kungliga Tekniska Högskolan år 2010
Handledare på CSC var Lars Kjelldahl
Examinator var Mads Dam

URL: www.csc.kth.se/utbildning/kandidatexjobb/datateknik/2010/hwasser_martin_K10061.pdf

Kungliga tekniska högskolan
Skolan för datavetenskap och kommunikation

KTH CSC
100 44 Stockholm

URL: www.kth.se/csc

Sammanfattning

Den här rapporten dokumenterar utvecklingen av en avancerad musikspelare till iPhone som erbjuder de nödvändiga funktioner som behövs för att dja. Applikationen är byggd och designad särskilt för pekskärmsformatet och rapporten diskuterar hur man utnyttjar dess egenskaper för att skapa ett dj-verktyg med ett sofistikerat gränssnitt som är intuitivt, elegant och enkelt att använda.

MINIDJ - DEVELOPMENT OF AN INTERACTIVE AUDIO APPLICATION FOR THE IPHONE FROM AN HCI PERSPECTIVE

Abstract

This report documents the development of an advanced music player for the iPhone that offers the necessary functions for dj:ing. The application is built and designed specifically for the touchscreen format and the paper discusses how to utilize its properties to create a dj-tool with a sophisticated interface that is intuitive, elegant and easy to use.

I INTRODUKTION.....	1
1.1 Syfte.....	1
1.2 Mål.....	2
1.3 Problemformulering.....	2
1.4 Omvärldsanalys.....	3
1.4.1 Unikhet.....	3
1.5 Metod och designprocess.....	3
1.5.1 Prototypdesign.....	3
1.5.2 Litteratursammanfattning.....	4
2 PROTOTYP.....	5
2.1 Den mentala modellen.....	5
2.1.1 Funktionsassociering.....	5
2.2 Tillstånd.....	5
2.3 Prototypskisser.....	6
2.3.1 Tillstånd: stoppad.....	6
2.3.2 Tillstånd: spelande.....	7
2.4 Användbarhet.....	8
2.4.1 Användbarhetstester.....	9
2.5 Framtida utveckling.....	10
3 RESULTAT.....	11
3.1 Funktionalitet.....	11
3.3 Design och struktur.....	11
3.2 Interaktion med användaren.....	11
4 SLUTSATS.....	12
KÄLLOR.....	13
BILAGA A: Teknisk specifikation.....	14
BILAGA B: Intervjufrågor.....	16

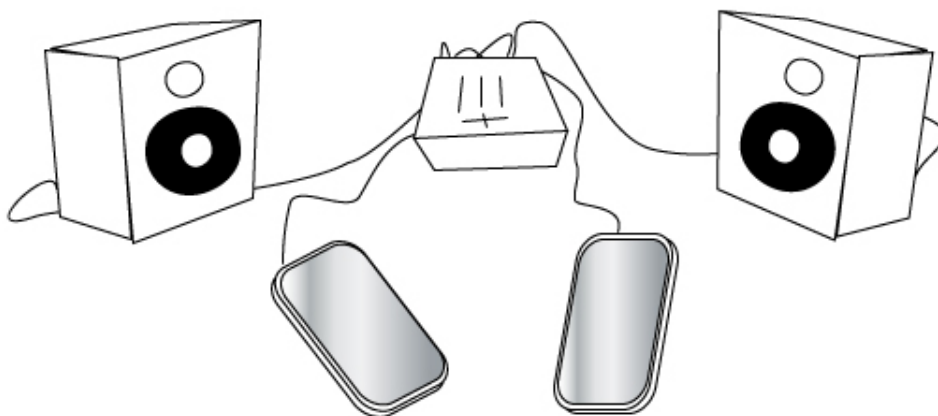
I INTRODUKTION

Den här rapporten dokumenterar utvecklingen av en applikation till iPhone som erbjuder ett mer avancerat gränssnitt för uppspelning av ljud än vad den inbyggda musikspelaren gör. Idén till applikationen föddes när jag deltog i Music Hack Day, ett evenemang där musikintresserade utvecklare får chansen att möta likasinnade människor och under 24 timmar skapa någonting musikrelaterat. Flera stora företag både deltog och sponsrade evenemanget, däribland SonyEricsson, Sveriges Radio, Spotify, Soundcloud och Last.fm. Mitt bidrag, Buddy, vann ett trådlöst högtalarsystem från företaget Sonos, något som motiverade mig att vidareutveckla applikationen, både från ett tekniskt och teoretiskt perspektiv.

Buddy var den första prototypen och skapades på endast 24 timmar. Den andra prototypen, som ligger till grund för den här rapporten, går under namnet MiniDJ.

I.1 Syfte

Applikationen ska göra iPhone ett alternativ till dyra och tunga CD-spelare för en dj. Eftersom en iPhoneägare bär med sig telefonen överallt finns alltid möjligheten att dja utan skivor eller dator; man ska kunna utföra de mest väsentliga dj-metoderna med hjälp av en iPhone kopplad till ett mixerbord och högtalare. Det går dock alldeles utmärkt att koppla sin iPhone direkt till ett par högtalare och använda MiniDJ men programmet förlorar då lite av sitt syfte, då det är tänkt att programmet ska underlätta taktmixning mellan två källor av musik.



Figur I.1.1 - En typisk uppsättning för MiniDJ med två iPhones kopplade till en mixer.

1.2 Mål

För att försäkra att applikationen skulle uppnå sitt mål som ett bärbart dj-verktyg bestämdes några grundläggande mål. Applikationen skall erbjuda följande möjligheter:

- *Ändra pitch*¹. För att enklare kunna taktmixa ska man kunna ändra på hastigheten på en låt.
- *Snabbt kunna söka i ljudfilen*. Att kunna snabbspola fram eller bak, hoppa till ett visst taktslag, eller skippa ett visst parti i en låt ger full kontroll över musiken och är således en oerhört användbar funktion. Applikationen ska också ha en cuefunktion².
- *Visualisera ljudfilen som en vågform*. Förutom att vågformen är estetisk tilltalande så underlättar den för orientering i låten då man med dess hjälp väldigt tydligt kan urskilja partier och enskilda taktslag.
- *Få ut information om bpm*³. En taktmix mellan två låtar med ungefär lika många taktslag i minuten är mycket enklare att utföra och därför bör den informationen vara lätt tillgänglig.
- *Anpassad för iPhone*. Applikationen ska till fullo utnyttja telefonens kapacitet och egenskaper, samt vara formad med fokus på användbarhet och elegant design.

1.3 Problemformulering

En av de största utmaningarna, och ett ämne som ligger i fokus för rapporten, är att skärmen på en iPhone är mycket liten. Gränssnittet måste därför vara optimerat och väl genomtänkt. *Hur kan man utnyttja iPhonens egenskaper för att maximera antalet funktioner, utan att kompromissa med ett intuitivt och tydligt gränssnitt?*

Ett problem av mer teknisk natur är att musikbiblioteket på en iPhone endast är tillgängligt för den inbyggda musikspelaren. Applikationen måste således utnyttja andra sätt att importera musik. En icke implementerad idé är att erbjuda synkronisering mot lagliga musiktjänster på nätet och på så sätt få musik till applikationen. Fokus i denna rapport ligger dock på användbarheten och gränssnittet, varför möjligheten att importera musik inte ansågs avgörande vid tillfället för användbarhetstesterna.

¹ Pitch är tonen av ett ljud. Att öka ett ljuds pitch, och därmed dess frekvens, leder till ett ljusare ljud. Att sänka pitchen leder till ett mörkare ljud. Inom dj'ing används dock pitch mest för att öka eller sänka tempo på en låt, eftersom en bieffekt av att ändra på frekvensen leder till ändrad läshastighet.

² En cuepunkt kan sättas var som helst i en låt. Om man sedan går in i cue-mode börjar uppspelningen direkt från cuepunkten.

³ *Beats Per Minute*: Antalet taktslag per minut.

1.4 Omvärldsanalys

På marknaden finns redan en handfull dj-applikationer vilka dock ofta brister på flertalet punkter. En av de största är det mycket avancerade programmet TouchDJ 2 från Amidio Inc⁴. Problemet med TouchDJ 2, liksom flertalet av dess gelikar, är att den innehåller för mycket funktionalitet. Skärmen är fylld med knappar och komplicerade funktioner och gränssnittet känns därför otydligt. Programmet är dessutom relativt dyrt⁵. Marknaden ligger därför öppen för en gratis, alternativt billig, och mer slimmad applikation.

1.4.1 Unikhet

Det som gör MiniDJ unik är dess enkelhet, kreativa design och (i framtiden) möjligheten för användarna att synkronisera sin musik mot Internettjänster. Applikationen är nischad i den meningen att den är specifikt anpassad för iPhone och försöker inte förbehållningslöst imitera funktioner som kanske fungerar bra på en skivspelare eller dator, men mindre bra på en liten skärm.

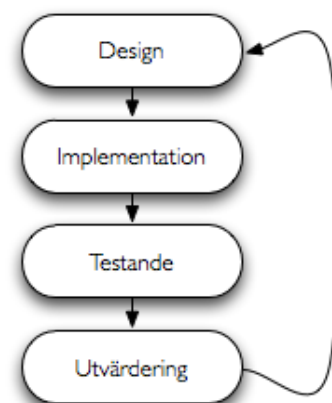
1.5 Metod och designprocess

"One of the most fundamental things to know about design is that an understanding of the design situation is established in parallel with the first design proposals."
(Löwgren & Stolterman, 2007)

Parallellt med design och utveckling utforskades teoretiska områden kring användbarhet och interaktiva gränssnitt. Praktisk design och teoretiskt arbete var på så sätt sammanvävda och påverkade varandra under hela processen. När prototyputvecklingen avancerade, utforskades de teoretiska koncepten djupare och vice versa.

1.5.1 Prototypdesign

Den första prototypen, BuddyJ, designades med fokus på funktionalitet. Det var en snabb, iterativ process för att utforska de tekniska aspekterna (se figur 1.5.1). Målet var att komma så långt som möjligt på kort tid. Detta underlättade identifiering av både konceptuella och tekniska problem och banade väg för en mer teoretisk inställning till designen. Att arbeta iterativt på detta sätt är dock inte så enkelt som det ser ut eftersom designprocessen är inte linjär utan skiftar mellan abstrakta idéer och konkret arbete på specifika detaljer (Löwgren & Stolterman, 2007). Medveten om detta faktum tog utvecklandet av den andra prototypen en mer utforskande



Figur 1.5.1 - Iterativ utveckling

⁴ <http://amidio.com/touch-dj-2>

⁵ TouchDJ kostar 149.00kr (1/5 - 2010)

karaktär som gick utanför ramarna och ifrågasatte hur en musikspelare egentligen ska se ut och hur man som användare kan interagera med den.

1.5.2 Litteratursammanfattning

En bred litteratursammanfattning sammanställdes för att täcka det spektrum av ämnen som rapporten handlar om. Huvudämnet i rapporten är människa-datorinteraktion, men en stor del av litteraturen var av teknisk natur och handlar om ljudhantering, ramverket Cocoa Touch⁶ och dess API:er⁷. I bilaga A finns viss teknisk information om hur applikationen utvecklades för den intresserade.

⁶ Cocoa Touch är ett ramverk för att bygga applikationer till iPhone.

⁷ Application programming interface

2 PROTOTYP

En stor del av arbetet vid sidan om rapporten gick åt att utveckla av en prototyp som användare kunde utforska och testa. Prototypen, MiniDJ, som beskrivs i följande del är en uppföljning på den allra första prototypen, BuddyJ. MiniDJ är mer grafisk utformad och fokus ligger på användbarheten.

2.1 Den mentala modellen

“Users often form a schema when learning a device and subsequent interactions are compared to the mental model formed during the initial learning phase.” (Mach m.fl., 2010)

I artikeln *Neurophysiological correlates in interface design: An HCI perspective* fastslår Mach m.fl. att användare av en ny produkt oftast redan har en mental modell som beskriver produktens uppgift och syfte. En mental modell har sin härkomst från en kombination av det verkliga livet, erfarenhet med annan mjukvara och med datorer i allmänhet (Apple, 2010). För att upptäcka mentala och konceptuella modeller som användare associerar med funktioner i applikationen undersöktes bland annat hur liknande funktioner på en dator eller CD-spelare används. Vidare utforskades om dessa funktioner kunde återskapas eller till och med förbättras för en mer strömlinjeformad applikation.

2.1.1 Funktionsassociering

I mediaspelare på en dator finns normalt en *slider*, en grafisk vågrät regel, som representerar var i låten man befinner sig. Man kan också söka i låten genom att klicka eller dra i regeln. En liknande, dock mer avancerad, funktionalitet finns i MiniDJ men representeras istället grafiskt av låtens vågform. Användartesterna visar att användaren genast inser möjligheten att förflytta sig i låten genom att dra med fingret på skärmen och på så sätt ändra position i vågformen.

Att spela och pausa är en musikspelares två absolut viktigaste funktioner, men relativt snabbt insågs att knappar för dessa två funktioner är redundanta. Istället klickar användaren på skärmen: enkelclick för att spela, dubbelclick för att pausa (användaren tvingas dubbelklicka för att pausa för att undvika att musiken stannar om man råkar komma åt skärmen). Förutom att starta och stoppa uppspelning försätter dessa funktioner även applikationen i olika tillstånd som beskrivs i följande del.

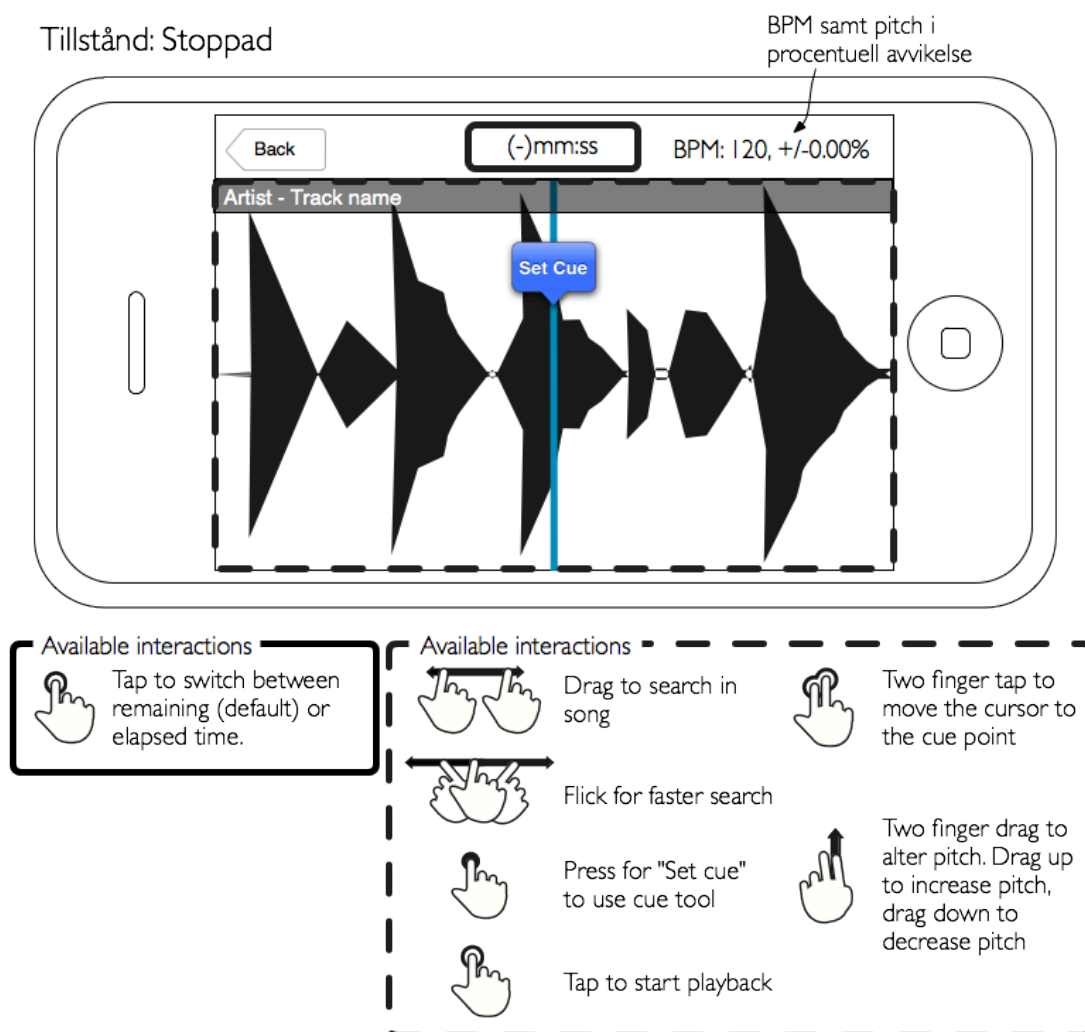
2.2 Tillstånd

Vi definierar två tillstånd: *spelande* och *stoppad*. Beroende på vilket tillstånd spelaren befinner sig påverkar användarens gester olika funktioner. Om man till exempel drar i vågformen då spelaren är *stoppad* så söker man i ljudfilen, men i *spelande* tillstånd så ökar man tillfälligt pitchen, något som kan liknas vid att snurra en vinylskiva lite snabbare.

2.3 Prototypskisser

Att konstruera en prototypskiss tvingar designern att ta itu med detaljerade frågor om interaktionsteknik och layout. Allteftersom skissen av MiniDJ tog form så definierades funktioner och tillstånd samt spatial strukturering. I följande del visas prototypskissen för de två tillstånden.

2.3.1 Tillstånd: stoppad



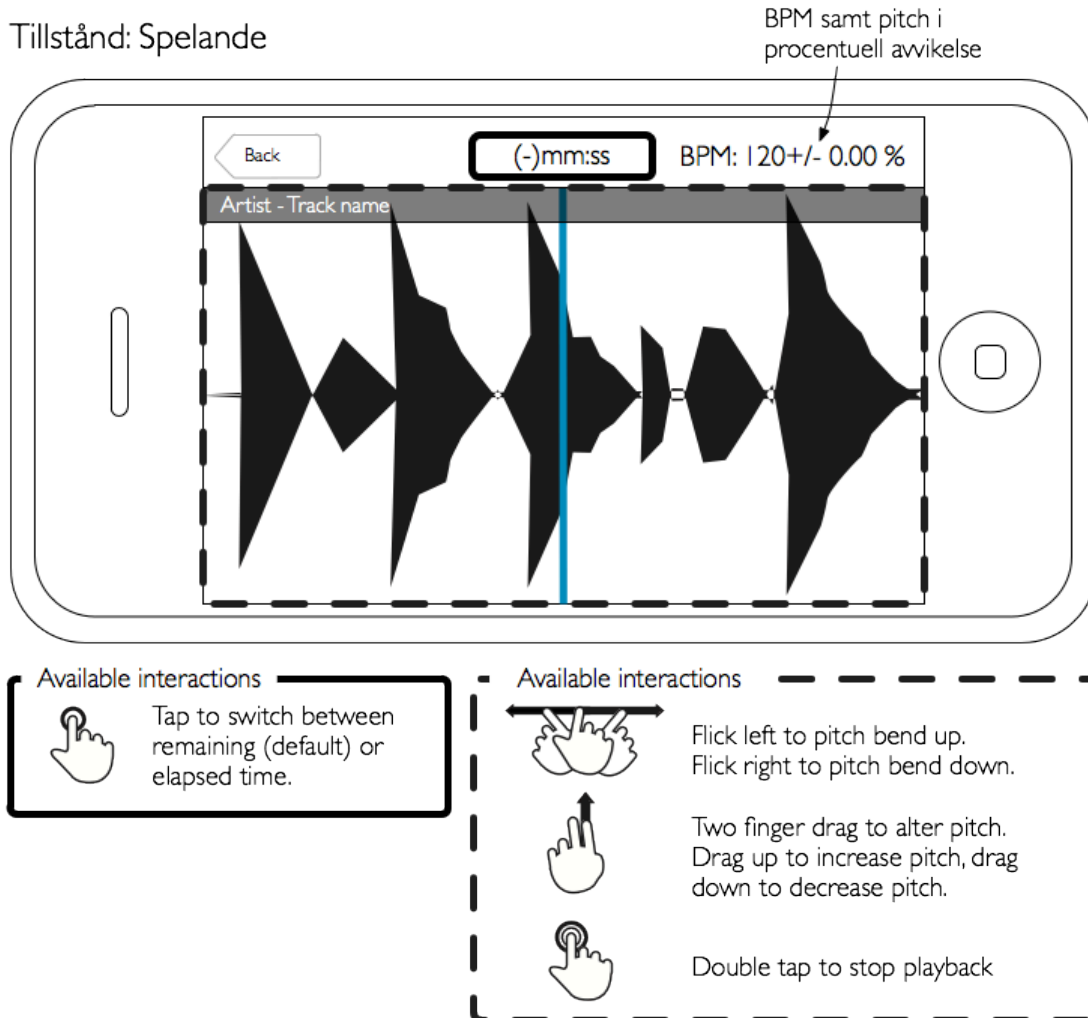
Figur 2.3.1 - Prototypskiss i tillstånd: stoppad.

Illustrerat i Figur 2.3.1 är MiniDJ i tillståndet *stoppad*, det vill säga att den inte spelar upp musik. I rutorna med heldragen eller streckad ram under iPhonen syns hur användaren kan interagera med applikationen genom att utföra olika handgester.

Möjlig interaktion i stoppat tillstånd:

- *Alternera mellan tid spelad och tid kvar:* Tryck på rutan som visar tiden.
- *Söka i ljudfilen:* Dra vågformen åt höger eller vänster vågrätt för att söka bakåt respektive framåt i ljudfilen. Snärta vågformen för att söka snabbare.
- *Sätta cue:* Tryck på "Set cue" för att sätta en cuepunkt.
- *Spela:* Klicka på vågformen en gång för att spela.
- *Gå till cuepunkten:* Dubbelklicka på vågformen för att gå till cuepunkten.
- *Pitcha:* Dra två fingrar upp eller ned för att höja respektive sänka pitchen. Den nya pitchen uppdateras i vyn och antalet bpm justeras därefter.

2.3.2 Tillstånd: spelande



Figur 2.3.2 - Prototypskiss i tillstånd: spelande.

Möjlig interaktion i spelande tillstånd:

- *Alternera mellan tid spelad och tid kvar*: tryck på rutan som visar tiden.
- *Pitchbenda*: Snärta vågformen åt höger eller vänster för att temporärt höja respektive sänka pitchen.
- *Pitcha*: Dra två fingrar upp eller ned för att höja respektive sänka pitchen. Den nya pitchen uppdateras i vyn och antalet bpm justeras därefter.
- *Pausa*: Dubbelklicka på vågformen för att pausa.

2.4 Användbarhet

Användbarhet definieras enligt International Organization for Standardization i *ISO 9241-11* som kraftfullheten, effektiviteten och tillfredsställelsen med vilken en specificerad användare uppnår specifika mål i en särskild omgivning (Chen, 2006). Chen beskriver de tre egenskaperna enligt följande (översatt från engelska):

Kraftfullheten är det mått på ansträngning som behövs för att avklara ett mål - ju mindre ansträngning, desto kraftfullare. Med effektiviteten menas i vilken utsträckning som ett mål eller uppgift blir avklarad. Med tillfredsställelsen menas den nivå av belåtenhet som användaren känner för produkten.

Med hänsyn till detta undersöktes användbarheten för applikationen med hjälp av fyra olika punkter:

- Användarens förmåga att utföra olika uppgifter, mätt genom att bevaka frekvensen avklarade uppgifter.
- Flexibilitet i designen, mätt genom förhållandet av användare som klarar att använda applikationen och behärska dess funktioner.
- Hur lätt applikationen är att lära sig, mätt genom ändring i frekvensen av avklarade uppgifter under en viss tid.
- Användarnas nivå av belåtenhet med produkten, mätt med hjälp av intervjuer.

Användartester utfördes med både mer och mindre erfarna discjockeys, då målgruppen för applikationen är vem som helst med en iPhone och ett intresse för att dja.

2.4.1 Användbarhetstester

En liten grupp valda användare fick chansen att testa MiniDJ under en kortare period. Målet var att få kritik och idéer för framtida utveckling. Intervjuer genomfördes direkt och var av mindre formell natur. Frågorna som ställdes var öppna men inte ledande (se bilaga B). Från responsen drogs flera slutsatser.

I överlag fann användarna att MiniDJ var spännande samt fångade deras intresse. Flera användare ansåg att applikationen var interaktiv eftersom gesterna de utförde genast reflekterades grafiskt i vågformen, något som ledde till en lekfull och utforskande användning. Detta märktes tydligt i hur de testade olika sätt att klicka och dra i vågformen. Erfarna iPhoneanvändare var som väntat de som snabbast lärde sig de olika gesterna. Efter bara någon minut hade dock alla användare lärt sig alla gester och klarade att utföra dem utan hjälp.

En intressant iakttagelse var att användarna ofta blev väldigt engagerade och kom med kreativa idéer om hur man skulle kunna vidareutveckla applikationen. En användare ville t.ex. ha möjlighet att spela in det utgående ljudet till fil för senare lyssning. En annan användare ville kunna använda applikationen för att styra musik på datorn.

Den vanligaste negativa reaktionen var över att biblioteket med tillgängliga låtar var så sparsam, något som står direkt i relation till att Apple inte ger tillgång till ljudfilerna i användarens musikkatalog.

En av de intervjuade tyckte att färgen på vågformen borde vara ljusare för att underlätta spelande i mörker. Det kan tyckas vara en liten småsak men påvisar att användbarheten även innefattar detaljer som färgkombinationer och ljusstyrka.



Figur 2.3.1 - Prototypen som användarna fick testa. Den lodräta linjen i mitten markerar var i låten spelaren befinner sig. Ovanför syns tiden spelad och till höger information om BPM och pitch. Knappen till vänster används för att byta låt.

2.5 Framtida utveckling

Ett antal tekniska förbättringar skulle kunna utföras som skulle göra applikationen intressantare och lättare att använda. Vissa av dessa förbättringar var redan planerade, andra är ett direkt resultat av användbarhetstesterna. En påtagligt viktig fråga är den kring överföring av musik till applikationen. Detta måste vara lätt och smidigt och är något som definitivt kommer ligga i fokus i vidareutvecklingen av MiniDJ.

En slutsats som kan dras från testerna är att erbjuda möjlighet att zooma in och ut på vågformen för att på så sätt ge mer översikt eller en mer detaljerad vy av låten. En annan funktion som föreslogs under intervjuerna var att omedelbart kunna återgå till normal spelningstakt.

Vidare kommer det undersökas om det ska finnas en avancerad vy för de som vill offra enkelheten för att utföra mer komplexa funktioner, t.ex. som att spela två ljudfiler samtidigt, eller att ändra i frekvensregistret.

3 RESULTAT

Den här delen behandlar resultaten av användbarhetstesterna och diskuterar hur iPhone:n fungerar som ett portabelt verktyg för en dj. Som påminnelse var målet att applikationen skulle erbjuda de mest väsentliga funktionerna för att kunna dj:a. Detta diskuteras från tre synvinklar:

- *Funktionalitet.* Fyller applikationen sitt syfte som ett portabelt dj-verktyg?
- *Design och struktur.* Lyckas applikationen anpassa sig till formatet?
- *Interaktion med användaren.* Är gesterna som används för att utföra funktioner intuitiva och lätta att lära sig?

3.1 Funktionalitet

I tidigt skede togs ett beslut om att endast implementera ytterst oundgängliga funktioner, med inställningen att för mycket funktionalitet i ett litet format snarare stjälper än hjälper. De mest essentiella funktionerna för att underlätta taktmixande implementerades och fungerade som planerat.

Ett smärre hinder för applikationen är att tredjepartsprogram inte ges tillgång till iPhones musikbibliotek. Användaren måste alltså överföra musik till iPhone:n manuellt, något som gör användarmönstret längre och mer komplicerat. Ett faktum som mottogs med viss missnöje hos de intervjuade.

3.3 Design och struktur

Användargränssnittet är enkelt, intuitivt och väl anpassat för en pekskärm. För att maximera integrationen med pekskärmen utnyttjades med fördel gester framför vanliga knappar. En knapp definierar oftast sitt syfte explicit genom en bild eller titel, till skillnad från en gest som användaren måste lära sig, även om gesten är implementerad på ett sätt som implicit imiterar en fysisk funktion. En nackdel med att inte använda knappar är alltså att användaren måste förstå vilka gester som korresponderar mot vilka funktioner. När denna gräns dock väl är passerad och användaren kopplar rörelser med funktioner framstår knappar som överflödiga och utnyttjandet av gester känns naturligt och självklart.

För att underlätta inläringen finns en infoknapp som visar instruktioner för hur man kontrollerar musikspelaren.

3.2 Interaktion med användaren

Användaren kontrollerar uppspelningen genom att utföra olika gester och rörelser på pekskärmen. Gesterna valdes noga ut med hänsyn till hur en mental modell av en viss funktion kan se ut. På de flesta skivspelare med förmåga att påverka pitch

finns oftast en regel som man drar upp eller ned för att höja respektive sänka pitch, varför en vertikal rörelse också motsvarar denna funktion i applikationen.

På liknande sätt utnyttjar MiniDJ den mentala modellen för hur man överhuvudtaget kan interagera med just en iPhone. Erfarna iPhoneanvändare har lärt sig grundläggande gester såsom att snärta med fingret för att t.ex. scrola snabbare genom en lista, eller att hålla fingret på ett objekt under en kort tid för att kunna utföra funktioner på det. En användare av MiniDJ lägger kanske inte märke till dessa detaljer utan tycker helt enkelt att det är naturligt för en iPhoneapplikation att stödja dessa riktlinjer för gestikulering, trots att det faktiskt är en implementationsfråga och en konceptuell anpassning till Apples riktlinjer från designerns sida.

4 SLUTSATS

En iPhone har alla egenskaper för att tjäna som ett portabelt dj-verktyg. Utvecklingen av MiniDJ har varit en balansgång mellan att erbjuda ett bra utbud av funktioner och samtidigt ha ett genomtänkt gränssnitt riktat mot en tydlig målgrupp. Målgruppen i detta fall är ganska snäv, då användaren både måste ha en iPhone och dessutom ett intresse för att dj:a. Användare som fyller dessa två kriterier har nästan uteslutande haft en positiv inställning till MiniDJ, även om det kvarstår en del att implementera.

Processen att dokumentera utvecklingen av MiniDJ har varit väldigt inspirerande och har tillfört nya tankar och resonemang kring både teoretiska och praktiska aspekter. Att motivera de designmässiga val som gjorts har inte bara lett till genomtänkta lösningar utan också banat väg för nya idéer och tankesätt som absolut kommer utforskas vidare.

En intressant aspekt med mobilapplikationer som MiniDJ är de sällan fyller en viktig funktion utan snarare är en bonus för dem som äger mobiltelefonen. Ingen köper nog en iPhone för en viss applikation, utan för det stora utbudet av olika applikationer. Råkar man äga en iPhone och dessutom är dj-intresserad är MiniDJ en bonus, men kanske inte en vital del i ens dj:ande. Det sagt betyder det inte att mobilapplikationer är onödiga - tvärtom - de fyller ett tomrum som tidigare inte existerade.

KÄLLOR

Apple, 2010. Human-computer interaction. Besökt April 2010. Hämtad från:
<http://developer.apple.com/Mac/library/documentation/UserExperience/Conceptual/AppleHIGuidelines/OSXHIGuidelines.pdf>

Audacity, 2009. Digital Audio. Besökt Mars 2010. Hämtat från:
http://manual.audacityteam.org/index.php?title=Digital_Audio

Chen, F. (2006). *Designing Human Interface in Speech Technology*. Springer.

Löwgren & Stolterman. (2007). *Thoughtful interaction design*. Cambridge, Massachusetts: The MIT Press.

Mach, Hunter & Grewal. (2010). Neurophysiological correlates in interface design: An HCI perspective. *COMPUTERS IN HUMAN BEHAVIOR*, 26 (3): 371-376 MAY 2010.

BILAGA A: TEKNISK SPECIFIKATION

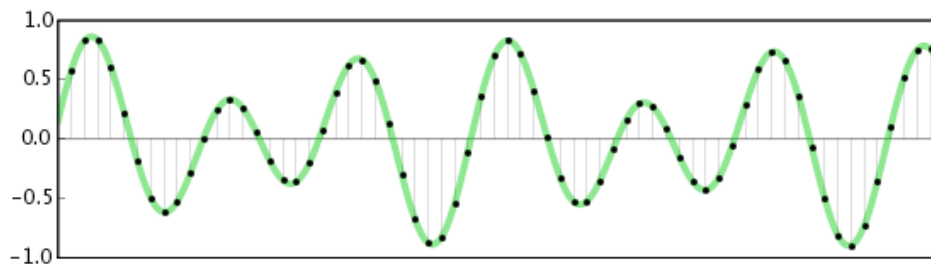
Programmeringsmiljö

Applikationen är skriven i programmeringsspråken C, Objective-C samt C++ i Cocoa-Touch. För att få full kontroll över ljuduppspelningen användes Core Audio, som är ett stort och komplext API på låg nivå, samt ramverket AudioToolbox vilket erbjuder ett gränssnitt för att hantera läsning och skrivning av ljudfiler. Applikationen använder också Audio Units som är API på systemnivå för att rendera ljudströmmar.

Programmet läser in en ljudfil till en floatbuffer och ljuduppspelningen sker sedan genom regelbundna callbacks till en renderingsmetod som itererar genom bytebuffern och spelar upp varje enskild sample. Detta gör det möjligt att påverka ljudströmmen in i detalj.

Digitalt ljud

Digital ljud konverteras från en analog källa genom att ta ögonblicksbilder, s.k. samples, med en viss frekvens. En CD har vanligtvis en samplingsfrekvens på 44100 Hz. Dessa samples har också en viss precision. Precisionen mäts i bitar, och för en vanlig CD är detta 16 bitar för varje sample. När ljudfilen sedan spelas upp konverteras den tillbaka till en analog källa med samma samplingsfrekvens och bitdjup.



*Figur 1.3.1 - Den heldragna linjen är en analog ljudvåg. Varje prick representerar en sample. Ju högre samplingsfrekvens, desto fler prickar och ju större bitdjup, desto bättre precision i varje sample.
(Audacity, 2009)*

Digitalt ljud på iPhone

Digitalt ljud komprimeras ofta för att storleken på musikfilerna ska bli mindre. En iPhone klarar dock bara att spela upp okomprimerat ljud. Detta format kallas *LPCM*, Linear Pulse Code Modulation. Därför måste alla komprimerade ljudfiler konverteras till *LPCM* innan de spelas upp.

Pitch och interpolering

När man pitchar ett ljud uppåt så ökar man steget med vilken man itererar genom ljudbuffern, vilket resulterar i att ljudet spelas upp snabbare. Detta kan leda till att ljudet blir hackigt och otydligt på grund av att man hoppar över samples. För att undvika detta används en algoritm för linjär interpolering som anpassar sig till vågkurvan och bestämmer vilken nästa sample för uppspelning ska bli.

Vågform

Att rita en vågform går ut på att iterera genom ljudbuffern och mäta styrkan i enskilda samples, och plotta dem proportionerligt i y-led. Men med tanke på att en vanlig låt har en samplingsfrekvens på 44100 Hz skulle det krävas 92 iPhone-skärmar för att plotta en sekund om varje sample representerades av en pixel i x-led. Således bör man alltså reducera ljuddata och beräkna maximum respektive minimum över ett visst intervall. Dessa värden kan man med fördel sedan interpolera över för att få en jämnare och vackrare ljudvåg.

Analysering av BPM

Jag har valt att inte dokumentera analyseringen av BPM då denna än inte är helt stabil och den tekniska aspekten, som bl.a. inkluderar en snabb fouriertransform, är utanför spektrumet för min rapport.

BILAGA B: INTERVJUFRÅGOR

Intervjuerna var mindre formella och tog ofta annorlunda vägar än vad som reflekteras i dessa frågor:

- Har du en iPhone?
- Har du erfarenhet av att dja?
- Vad tyckte du om MiniDJ? Varför?
- Var gränssnittet intuitivt och lätt att lära sig?
- När du inte visste hur man skulle utföra nånting, vad gjorde du då?
- Hur kan MiniDJ förbättras?

