

URD-1

The Meta Family

Version 0.0

den 1 mars 2014

Daniel Bergström

Paulina Hensman

Marcus Hertz

Simon Karlsson

David Masko

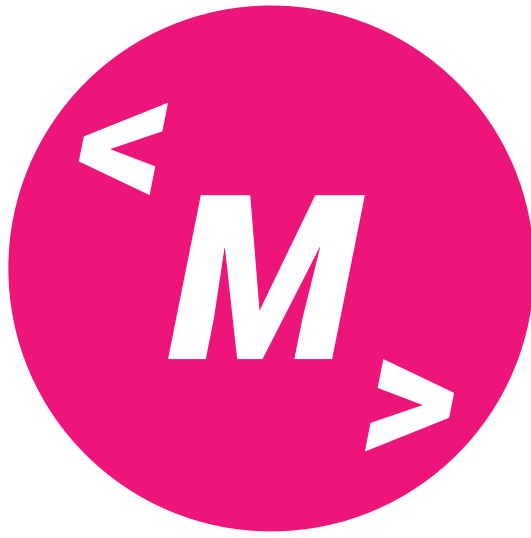
Marcus Nordberg

André Nyström

William Perkola

Axel Riese

Kerry Zhang



*The
Meta
Family*

Sammanfattning

Innehåll

Versionshistorik	1
1 Introduktion	2
1.1 Syfte	2
1.2 Produktens omfång	2
1.3 Begreppsdefinitioner	2
1.4 Referenser	2
1.5 Överblick över dokumentet	2
2 Generell Beskrivning	2
2.1 Produktperspektiv	2
2.2 General Capabilities (?)	2
2.3 Begränsningar	2
2.3.1 Antalet rörelsehändelser	2
2.3.2 Rörelsehändelser som inte tillhör applikationen	3
2.4 Användarbeskrivning	3
2.5 Antaganden och beroenden	3
2.6 Operativ miljö	3
3 Specifika Krav	4
3.1 Funktionsmässiga krav	4
3.1.1 Användaren har själv kontroll över vad som spelas in	4
3.2 Tekniska krav och begränsningar	4
3.2.1 Snabb implementation av SDK:n	4
Appendix	5

Versionshistorik

1 Introduktion

1.1 Syfte

1.2 Produktens omfång

1.3 Begreppsdefinitioner

Applikation Körbart program i smartphones.

Bitmap Grupp lagringsformat för digitala bilder där ingen data går förlorad. Lämpad för exempelvis skärmdumpar.

GUI Graphical User Interface är det gränssnitt som en användare av ett program ser och interagerar med.

iOS Namnet på det operativsystem som används på Apples smartphones.

Laravel Ett ramverk byggt främst i PHP för att förenkla utvecklandet av skalbara webbapplikationer(**laravel**).

PHP PHP: Hypertext Preprocessor (*rekursiv akronym*) är ett programmeringsspråk som körs på webbservern för att hantera dynamiskt innehåll på webbplatser.

Protokoll Samordnad överenskommelse mellan utvecklare av separata system om hur data skall skickas för att möjliggöra korrekt mottagande av data.

SDK Software Development Kit är en mängd verktyg som gör det möjligt för en utvecklare att bygga en viss mjukvara. Det kan röra sig om allt från ett enskilt bibliotek som tillhandahåller en mängd funktioner till en komplett utvecklingsmiljö.

TBF The Beta Family, vår projektbeställare.

Zurb Ett ramverk byggt i HTML, CSS och JavaScript för att underlätta front-end-utveckling hos webbapplikationer(**zurb**).

1.4 Referenser

1.5 Överblick över dokumentet

2 Generell Beskrivning

2.1 Produktperspektiv

The Beta Family har som affärsidé med sin produkt Super Recorder att bistå med testare när utvecklare önskar att betatesta sin iOS-applikation genom att sammanföra dessa genom en hemsida. Företaget betalar testarna en belöning och The Beta Family tar en andel av denna. Det

Samfund The Beta Family byggt upp gör att de kan erbjuda en mer sömlös betatestning jämfört med konkurrenter.

En brist i tjänsten är att den endast riktar sig till användare som har en smart telefon med iOS och utvecklare som utvecklar produkter för iOS. I dagsläget produceras omkring 80

Projektet som fångas in av denna rapport har som uppgift att nå de kunder som använder Android. Detta genom att utveckla en SDK för Android som efterliknar den som The Beta Family tillhandahåller iOS så nära som möjligt utefter möjligheterna för Android. Detta möjliggör en bredare testning från utvecklare samt öppnar möjligheterna för testare som har Android i sina smarta telefoner. Projektet utgör därmed en möjlighet att bredda den befintliga tjänsten. En breddning av tjänsten ökar kvantiteten av testare som kan erbjudas, vilket kan vara inbjudande för fler utvecklare – en progression som i slutändan ökar intäkterna från tjänsten.

Eftersom projektet handlar om en efterlikning har inga behov för SDKn i sig behövs sökas då de har tillhandahållits direkt av The Beta Family genom deras SDK för iOS. Däremot har undersökningar i Androids möjligheter och dess begränsningar gjorts för att kunna säkra vad som är genomförbart. Dessa undersökningar gjordes genom att läsa igenom officiella dokument för Android samt liknande problemställningar andra utvecklare stött på och hur dessa lösningar diskuterats.

Eftersom efterlikning är av hög prioritet är följande funktionalitet hos tjänsten ytterst relevant eftersom stöd för Android ska kunnas implementeras på tjänsten utan att systemet ska behöva förändras till en komplicerad och tudelad process för användare. Den nuvarande processen för kunder beskrivs nedan.

Att som utvecklaren anpassa sin applikation för möjliggörandet av nödvändig inspelning av betatestingsessioner är snabb och enkel. Detta moment inkluderar endast att kopiera in några bibliotek till applikationen samt lägga till ett fåtal kodrader som möjliggör visningen av GUI;¹. Detta är en viktig del i tjänsten ty detta leder till att mycket låg arbetsbörda läggs på utvecklaren. Utöver detta behövs inget beaktande från utvecklaren i hur själva videon produceras eller sedan skickas iväg till The Beta Familys servrar då detta sköts av de medföljande biblioteken.

För testare av en given applikation behövs inga särskilda förberedelser göras. En testare laddar ner applikationen från Super Recorder-hemsidan, startar applikationen, startar inspelningen och följer sedan instruktionerna som utvecklaren har givit testarna.

Utöver att försöka att återskapa samma funktionalitet i SDKn för Android behövs även integration med The Beta Familys servrar göras. Eftersom det med stor sannolikhet är omöjligt att erhålla identisk data från Android likt den från iOS på grund av deras skillnader kommer inte det nuvarande protokollet mellan serverna och SDKn att kunna återvinnas i sin helhet hos SDKn för Android. SDKn projektet avser att utveckla är dock inte beroende av något bestämt protokoll vilket möjliggör att projektgruppen själva kan välja det protokoll som ska användas mellan SDKn och serverna.

Behovet av denna SDK är i stor betydelse för framtida vinster och en ökad mängd kunder. Då smarta telefoner med Android som tidigare nämnt står för en stor del av tillverkade produkter är relevansen att kunna nå ut till en sådan stor mängd potentiella kunder en starkt rekommenderad, möjligtvis även nödvändig, för tjänstens Super Recorders framgång.

2.2 Generella förmågor

För att uppskatta och fastställa vad denna plattform bör kunna göra delas slutanvändarna in i två grupper: apputvecklare och betatestare. Dessa är två vitt skilda målgrupper med olika förutsättningar och mål, men båda är centrala för plattformens framgång. Plattformens förmågor bör därför utformas efter båda dessa gruppers behov.

¹ <https://thebetafamily.com/superrecorder/installation>

2.2.1 Applikationsutvecklarnas perspektiv

Arbetsgivarna till detta projekt fick idén om att utveckla en plattform för betatestning när de själva skulle lansera en mobilapplikation och insåg att det inte fanns någon bra tillgång till riktigare användare som testar applikationen i sina egna miljöer. Detta är ett ännu allvarigare problem för Androidutvecklare jämfört med iOS-utvecklare på grund av den fragmentering som uppstått då Androids operativsystem körs på hundratals olika hårdvaruspecifikationer med olika skärmstorlekar och versioner på operativsystemet. För att lösa detta problem har vissa företag köpt in alla mobiltelefoner som deras användare har (**buyAllPhones**), men detta är uppenbarligen inte en möjlighet som alla utvecklare har. En förmåga som plattformen bör ha är alltså att kunna **nå ut till riktiga användare**. Detta har arbetsgivarna löst genom att skapa ett community på nätet med testare och utvecklare som i skrivandets stund har över 10 000 testare(**betafamily**).

När de riktiga användare väl har nåtts bör **data om hur applikationen används** också kunna samlas in. Den tekniska informationen såsom kraschloggor, belastning av telefonens processorer och så vidare är av intresse för utvecklare, men den centrala förmågan som är högsta prioritet för detta projekt är videoinspelning av användningen, vilket inkluderar skärminspelning, inspelning av tryck på skärmen, samt inspelning av användarens reaktioner genom både videokameran och mikrofonen.

Ibland är det endast en viss funktion i applikationen som utvecklarna vill testa. Utvecklarna har därför möjligheten att på hemsidan specificera uppgifter som de vill att testarna ska utföra.

Testfunktionerna bör enkelt kunna integreras i den applikationen som ska testas. Arbetsgivarnas iOS-version importeras i delar till de olika utvecklingsmapparna som applikationen består av vilket beräknas ta ungefär två minuter, medan konkurrenten TestFairly erbjuder utvecklarna möjligheten att ladda upp deras färdigkompileerade kod på deras hemsida så bakas testfunktionerna in automatiskt.

Till sist bör all den insamlade informationen kunna presenteras på ett överskådligt sätt för utvecklarna, vilket görs på hemsidan där utvecklarna kan spela upp inspelningen av testerna samt följa de loggor som förts av exempelvis prestandan av enheten som applikationen körs på.

Allt detta är tänkt att ge applikationsutvecklarna mer information om vad som fungerar bra, samt mindre bra med deras applikation.

2.2.2 Betatestarnas perspektiv

Betatestare har olika motivationer för att testa nya applikationer. Vissa gör det av välvilja för att hjälpa till i utvecklingsprocessen; vissa gör det för att få testa nya applikationer innan de når allmänheten; och vissa gör det av monetära syften. Allt detta löser arbetsgivarna genom deras community där nya applikationer som kan testas annonseras ut och utvecklarna har möjligheten att erbjuda testare monetär kompensation för varje utförd test.

När det kommer till själva testandet så bör den applikationen som ska testas vara i fokus och allt omkringliggande ta så lite plats som möjligt. Lösningen på detta i iOS-versionen som arbetsgivarna har utvecklat är att testaren med ett dubbeltryck med två fingrar får ner en meny där de kan starta och stoppa inspelningen. Denna meny är det enda som finns utöver själva applikationen som ska testas. Testarens integritet måste också respekteras och därför finns möjligheten att stänga av inspelning genom videokameran och mikrofon även i menyn. Androidversionen ska försöka emulera detta beteende i så stor grad som möjligt.

Installationen bör vara så smidig som möjligt vilket möjliggörs av att endast en installationsfil behöver laddas ner genom hemsidan.

2.3 Begränsningar

2.3.1 Antalet rörelsehändelser

Android har ett bibliotek vid namn *android.view.View.OnTouchListener* (**touchlistener**) som sköter all inmatning från fingrar. En begränsning skulle kunna ligga i den mängd händelser Android kan läsa.

Därför utvecklades en liten simpel applikation för att ta reda på vilken uppdateringsfrekvens som kan antas under projektutvecklingen. Det kan vara viktigt att veta då The Beta Family kan ha testare med äldre mobiler och skulle slutprodukten begränsas till någon enstaka bild per sekund kan det bli problematiskt att synkronisera rörelsehändelser med videon.

Enhetens klocka sparas när man vidrör skärmen och för varje rörelsehändelse (*TouchEvent*) som registreras ökas ett värde med ett och samtidigt jämförs den aktuella tiden med den sparade tiden. Genom att dividera antalet händelser med hur lång tid det gått kan ett värde fås fram. Detta värde kallas uppdateringsfrekvensen och blir alltså antalet händelser enheten registrerar per sekund.

I tabellen nedan hittas ett antal mobiltelefoner, året de släpptes samt vilken uppdateringsfrekvens som uppmätts med denna applikation.

Modell	Årtal	Uppdateringsfrekvens (Hz)
Google Nexus 5	2013	60
Samsung Galaxy Nexus	2011	57
Sony Xperia Z	2013	60
LG Optimus 2X	2011	60
HTC Incredible S	2011	63
HTC Sensation	2011	60
HTC Desire HD	2010	70

Uppdateringsfrekvens för skärmen hos vissa mobiltelefoner

Det visar sig att uppdateringsfrekvensen inte blir några problem alls. HTC Desire HD är den äldsta mobil som testades och den har inga problem att klara 70 händelser per sekund. Denna uppdateringsfrekvens är med säkerhet tillräcklig för det projektet syftar åstadkomma.

2.3.2 Rörelsehändelser som inte tillhör applikationen

Det finns en begränsning i vilka rörelsehändelser en applikation får läsa av. Vyer som tillhör andra applikationer och API går inte att hämta rörelsehändelser från. Det kan röra sig om Google Maps, tangentbordet, etc. Moduler av applikationen som inte är gjorda av utvecklaren kan alltså inte läsas från. Detta gör att den ScreenRecorder som ämnas implementeras kommer ha en begränsning i att använder sig applikationen av t.ex. Google Maps kommer inte resultatet (videon av testet) kunna visa hur användaren rörde sig inom denna del.

För att testa detta utvecklades en mycket simpel exempelapplikation. Applikationen skrev ut nuvarande X- och Y-position för fingret i övre vänstra hörnet. Sedan lades en Google Maps-karta till över halva skärmytan. När fingret rörde sig utanför denna yta uppdaterades X- och Y-position ständigt men såfort fingret hamnade inom ramen för kartan slutades värdena att uppdateras.

I presentationen skulle man kunna lägga in information om att testaren befinner sig i en vy som inte tillhör applikationen när detta inträffar. Till exempel genom en liten ruta med texten "Användaren befinner sig just nu i Google Maps."

2.4 Användarbeskrivning

2.4.1 Persona - Patrik Berglind



Patrik Berglind

Ålder	26
Relationsstatus	Sambo
Arbete	Applikationsutvecklare
Utbildning	Yrkesutbildning applikationsprogrammering
Boende	Kungsholmen

Patrik Berglind är 26 år gammal och uppvuxen i Kramfors. Efter att Patrik tagit examen från Yrkehögskolan Höga Kusten för tre år sedan där han läste till applikationsprogrammerare flyttade han till Stockholm och har sedan dess arbetat på App AB. Patrik ingår i en liten projektgrupp där han fingerar som programmerare och utvecklar mobilapplikationer till både iPhone och Android. Patrik Berglinds stora passion är programmering och han har programmerat sedan han var 15 år. När Patrik skulle välja utbildning var yrkesutbildningen inom applikationsprogrammering ett självklart val eftersom han ville ut i arbetslivet snabbt med eftertraktad kompetens inom den expansiva appmarknaden och Patrik fick arbetet som applikationsutvecklare på App AB en vecka efter sin examen.

Patrik har på senare år utvecklats till en riktigt frilufsmänniska och brukar åka till norra Norge och vandra i bergen tillsammans med sin sambo Lisa med jämna mellanrum.

Patrik och Lisa väntar sitt första barn och går i tankarna att köpa ett radhus i någon förort till Stockholm. De har tittat extra mycket på ett radhus i Boo alldeles i närheten av Järlasjön och ett annat vid Sticklinge Udde.

Mål

Även fast Patrik älskar att programmera har han på sistone känt att det blir tråkigt att bara programmera hela dagarna och har därför som mål att inom kort försöka bli projektledare och således själv få styra en projektgrupp.

2.4.2 Scenario 1 - Patrik Berglind

2.4.3 Persona - Emma Henriksson

Emma Henriksson

Ålder	18
Relationsstatus	Singel
Sysselsättning	Gymnasiestudent
Boende	Täby

Emma Henriksson är 18 år gammal och pluggar teknik på Åva Gymnasium i Täby där hon går i tvåan. Emmas favoritämnen är matematik och fysik så det var aldrig någon tvekan om vilket gymnasieprogram hon skulle välja. Emma är sugen på att studera vidare direkt efter gymnasiet och hon står inför valet mellan KTH och Chalmers där hon vill plugga till Civilingenjör i Maskinteknik.

En av Emmas största intressen är film och hennes absoluta favoritfilm är *La vita é bella*. Om hon inte tittar på film lyssnar hon gärna på musik och då är det The National eller Sufjan Stevens som gäller. I övrigt lever hon ett vanligt tonårsliv där hon gärna umgås med vänner, både på riktigt och på Facebook.

Mål

Emmas mål är att komma in på ett civilingenjörsprogram på antingen KTH eller Chalmers. Inom den närmsta framtiden vill hon gärna hitta ett eget boende och hon drömmer om att få testa på att leva och bo utomlands.

2.4.4 Scenario 1 - Emma Henriksson

Klockan är 7.30 och Emma har precis hoppat på bussen mot skolan. Eftersom hållplatsen ligger ganska tidigt på busslinjen är det inte många personer på bussen så hon väljer en plats lång bak. Det dröjer dock inte länge innan bussen är full och Emmas vän Saga stigit på bussen och slagit sig ner på platsen bredvid Emma. Emma berättar att hon känner sig orolig inför morgondagens prov i Kemi som hon inte börjat plugga inför i tillräckligt god tid.

Emma kommer plötsligt på att hon planerat att testa en ny app som samlar information om världens alla band med hjälp av SuperRecorder. Hon drar upp sin HTC Sensation ur sin handväska, plockar upp sitt headset, ber Saga om ursäkt, och stoppar in dem i öronen. Hon sätter igång appen hon ska testa och startar SuperRecorder. Eftersom hon sitter på bussen väljer hon att inte filma sitt ansikte eller spela in ljud eftersom det är ganska mycket oväsen på bussen. Emma söker genast upp sitt favoritband The National och kollar runt på deras bandsida. Hon ser bland annat att de kommer till Stockholm nästa höst och påminner sig själv att köpa biljett.

När bussen närmar sig skolan avslutar Emma testet och stänger ner appen innan hon lägger ner mobilen i handväskan och kliver av bussen tillsammans med Saga.

2.4.5 Scenario 2 - Emma Henriksson

2.5 Antaganden och beroenden

2.5.1 Plattformsomfattning

TBF har idag inga krav på vilka versioner av iOS eller Android testbeställare får rikta sig till (**betafaq**). iOS-versionen av SuperRecorder, som gruppens produkt strävar efter att likna, stödjer också alla versioner av iOS (**superrec**). På grund av denna information har gruppen gjort antagandet att ett brett versionsstöd är en viktig punkt för TBF och dess kunder. Testbeställare som använder sig av SuperRecorder kommer att vänta sig att tjänsten fungerar på nya såväl som gamla versioner av Android, eftersom övriga delar av TBF:s testtjänster redan gör det. Versionsoberoende lösningar bör därför alltid föredras framför tillämpningar som baserar sig på ny systemfunktionalitet.

2.5.2 Testbeställarens målgrupp

Apputvecklare söker sig till TBF för att testa sina produkter på ett så antal scenarion som möjligt. Förutom att nå ut till så många plattformar som möjligt, se ??, vill testbeställare givetvis nå ut till så många olika användare som möjligt. Detta innebär idealt att testarna som TBF förmedlar ska ha en brett kompetensspann, med både ovana och avancerade användare inkluderat. För att kunna nå ut till en så varierande skara användare bör SuperRecorder ställa så få krav på användarna som möjligt. Tjänsten bör likt SuperRecorder för iOS kunna användas utan speciella förkunskaper, och utan speciella krav på befintlig hård- och mjukvara, så länge testarens mobil är baserad på Android.

2.5.3 Testarens nätanslutning

För att SuperRecorder ska kunna ge en högupplöst gestaltning av ett användarscenario måste tjänsten skicka relativt mycket data mellan testarens mobil och TBF:s servrar. Detta är i synnerhet sant om behandlingen av rådatan sker på server-sidan, vilket kan bli ett krav för att kunna nå ut till testare med äldre mobiltelefoner. För att kunna skapa en produkt som motsvarar TBF:s förväntningar måste vi därför anta att testaren har en robust och snabb nätanslutning. Allra helst ska testaren vara WiFi-ansluten vid testtillfället.

2.5.4 Testbeställarens tillgång till källkod

Integreringen av SuperRecorder för iOS sker genom att importera TBF:s bibliotek till applikationen, och sedan lägga till ett par rader i källkoden (**superrec**). Detta är en hyfsat trivial handling i utvecklingsverktyget för iOS. Gruppens efterforskningar har visat att motsvarande integrering i Android troligen inte är lika smärtfri. Ändringar kan komma att bli nödvändiga på ett flertal ställen i testbeställarens källkod för att SuperRecorder ska få önskad funktionalitet. Denna process har goda möjligheter för att kunna automatiseras med program, men kräver dock att testbeställaren har full tillgång till hela den testade applikationens källkod. Använder sig den testade applikationen av redan kompillerade tredjeparts-bibliotek, kommer dessa moduler inte att kunna loggas med SuperRecorder.

2.5.5 Serversidans miljö

Det finns ett flertal öppna och effektiva program och bibliotek för behandling av bitmaps-bilder och videoskopande av dessa. För att kunna nyttja dessa är det dock nödvändigt att servermiljön som SuperRecorder anropar har möjlighet att använda sig av godtycklig programvara. Gruppen antar därför att TBF har tillgång till en servermiljö som de kan modifiera efter eget tycke, och som alltså inte är låst till en viss mängd fördefinierade verktyg.

2.6 Operativ miljö

Den bakomliggande servern och databasen tar emot data från testaren. Det som tas emot är inspelningen av användarens upplevelser samt diverse teknisk data. Denna data ska sedan presenteras via ett webbgränssnitt som apputvecklaren kommer åt. Webbgränssnittet skapas endast i mån av tid och ska i sådana fall integreras med The Beta Familys webbplats som just nu är under omarbeting. Webbplatsen ramverket Laravel (PHP och MySQL) samt ramverket Zurb (HTML, CSS och JavaScript).

2.6.1 Skapa video med FFmpeg

De skärmdumpar som tagits ska sättas ihop tillsammans med ljud och touch för att bilda en video. Detta sker på serversidan genom att först synkronisera och slå ihop de inspelade touch-händelserna med skärmdumparna. Detta görs i PHP med modulen MagickWand(**magickwand**). Sedan används (kallas på via PHP) programmet FFmpeg(**ffmpeg**) för att sammanfoga de behandlade bitmaps-bilderna och ljudinspelningen för att skapa en video som sparas på servern.

3 Specifika Krav

3.1 Funktionsmässiga krav

3.1.1 Skärminspelning

Identifikator	Skärminspelning
Beskrivning	Allt som användaren ser ska dokumenteras genom att ta så många bilder vi kan i sådan hög frekvens som möjligt.
Motivering	Applikationsutvecklaren måste få möjlighet att se hur användaren navigerar i applikationen för att se hur användarvänligt gränssnittet är.
Behov	Minimum
Prioritet	Skärminspelningen har hög prioritet. Det är en av de viktigaste funktionerna för att få reda på hur användaren använder appen. En snabb prototyp behövs men skärminspelningsarbetet är tidsödande och behöver arbetas med konstant för att få ett så bra resultat som möjligt.
Källa	Dokument med krav från The Beta Family.
Verifiering	Kravet får verifieras genom att testa produkten på ett antal olika applikationer för att se att det fungerar att spela in skärmen oavsett vilken app som använder den.

3.1.2 Röstinspelning

Identifikator	Röstinspelning
Beskrivning	Vad användaren säger under testningen av applikationen ska spelas in.
Motivering	Att höra användarens tankegångar under testning utav användargränssnitt är högst önskvärt av apputvecklare.
Behov	Minimum
Prioritet	Röstinspelningen har hög prioritet. Det är en av huvudfunktionerna och en av de funktioner som har tidigast deadline.
Källa	Dokument med krav från The Beta Family.
Verifiering	Kravet får verifieras genom att testa produkten på ett antal olika applikationer för att se att det fungerar att spela in ljud oavsett vilken app som använder den.

3.1.3 Användaren har själv kontroll över vad som spelas in

Identifikator	Användarkontroll
Beskrivning	När en person betatestar en app med hjälp utav The Beta Familys ScreenRecorder ska de kunna välja själv vad för information de vill ge utvecklarna. Testaren ska själv kunna bestämma om frontkameran och mikrofonen ska spela in under testet.
Motivering	Testaren bör kunna stänga av vissa funktioner då man inte alltid befinner sig i situationer då man är bekväm att t.ex. tala högljutt om vad som händer i appen. Det kan handla om att man gör ett test på tunnelbanan på vägen hem och bara vill visa hur man navigerar runt i applikationen.
Behov	Standard
Prioritet	Att användaren själv kan stänga av vissa funktioner är önskvärt men inte av högsta prioritet.
Källa	Dokument med krav från The Beta Family
Verifiering	En oberoende part får testa att genomföra ett test med olika funktioner avstängda. Lyckas personen utan problem stänga av önskade funktioner kan kravet anses uppfyllt och dessutom är det ett tecken på bra interaktionsdesign.

3.2 Tekniska krav och begränsningar**3.2.1 Snabb implementation av SDK:n**

Identifikator	Snabb SDK-implementation.
Beskrivning	Enligt The Beta Family kan deras SpeedRecorder för iOS implementeras på under två minuter av utvecklaren. Det är önskvärt att Androidversionen är lika simpel att implementera.
Motivering	En del av The Beta Familys affärsidé är att utvecklarna själva implementerar deras ScreenRecorder. Detta ställer krav på utvecklaren att de kan implementera bibliotek och skriva några rader kod. Därför vill The Beta Family att implementationen är så simpel som möjligt. Krävs en mängd invecklade operationer och beslut kan detta skrämma bort eventuella kunder som då kanske väljer att inte betatesta applikationen.
Behov	Standard
Prioritet	Att SDK:n är simpel att genomföra har ganska hög prioritet. Det viktigaste är självklart att den funktionsmässigt fungerar, är den krånglig att implementera kan den fortfarande användas av mer tekniska utvecklare. En SDK som är simpel
Källa	Uppstartsmöte med The Beta Family där VD:n Axel Nordenström gick igenom krav.
Verifiering	När projektet är färdigställt kan en oberoende part låtas implementera SDK:n på tid. Lyckas personen implementera SDK:n på under två minuter kan detta krav ses som uppfyllt.

Identifikator	
Beskrivning	
Motivering	
Behov	
Prioritet	
Källa	
Verifiering	

Appendix