



CHALMERS

Ett system för mobil administration av egna välgörenhetsinsamlingar

Examensarbete inom Data- och Informationsteknik

MICAELA CARLSSON
RICHARD DAHLGREN

EXAMENSARBETE

Ett system för mobil administration av egna välgörehetsinsamlingar

Examensarbete inom Data- och Informationsteknik

MICAELA CARLSSON
RICHARD DAHLGREN

Institutionen för Data- och Informationsteknik
CHALMERS TEKNISKA HÖGSKOLA
GÖTEBORGS UNIVERSITET

Göteborg 2017

Ett system för mobil administration av egna välgörehetsinsamlingar

Examensarbete inom Data- och Informationsteknik

MICAELA CARLSSON

RICHARD DAHLGREN

© MICAELA CARLSSON, RICHARD DAHLGREN, 2017

Examinator: Peter Lundin

Institutionen för Data- och Informationsteknik
Chalmers Tekniska Högskola / Göteborgs Universitet
412 96 Göteborg
Telefon: 031-772 1000

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet.

The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Institutionen för Data- och Informationsteknik
Göteborg 2016

SAMMANFATTNING

Välgörenhetsorganisationer har traditionellt bedrivit insamlingar genom att samla in pengar från privatpersoner med insamlingsbössor och genom annonsering. Det traditionella tillvägagångssättet har över tid förändrats till insamlingar via internet. Star for Life är en av de välgörenhetsorganisationer som börjat använda internet för att öka sina intäkter. Privatpersoner kan numera donera via deras webbplats. Detta gör det enklare och snabbare för bidragsgivare att bidra till insamlingar och därmed öka Star for Lifes intäkter. Implementationen av betalningslösningen för donationer har tagits fram av företaget Sigma IT Consulting. Star for Life har utvecklat en strategi för att öka antalet donationer från privatpersoner. Denna strategi syftar till att erbjuda privatpersoner möjligheten att skapa insamlingar i organisationens namn men med ett specifikt ändamål angivet av personen. Problemet i nuvarande läge är att insamlarna, d.v.s. de som skapat en insamling, inte själva kan administrera insamlingen genom exempelvis redigering av insamlingens beskrivning eller möjligheten att få en lista på donationer som inkommit. För att lösa detta problem vill Star for Life skapa en mobilapplikation och därigenom erbjuda verktyg för att administrera insamlingen. En mobilapplikation gör det möjligt för privatpersoner som startat insamlingar att enkelt administrera sina insamlingar var de än befinner sig. Examensarbetet har resulterat i en mobilapplikation för Android som gör det möjligt för insamlarna att administrera sina insamlingar. Begränsningar i WordPress REST-API har inneburit att ett eget API dessutom utvecklats.

Nyckelord: Android, Mobilapplikation, Välgörenhet, Java, WordPress, API, MVP

ABSTRACT

Charity organizations have traditionally been collecting funds by collecting money from individuals with donation tins and through advertising. The traditional method has over time transformed into the use of fundraisers on the internet. Star for Life is one of the charity organizations that have begun using the internet to increase their revenue. Individuals can now donate through their website. This solution makes it easier and quicker for the individuals to contribute to fundraisers and thereby increasing Star for Life's revenue. The implementation of the payment system for donations was created by Sigma IT Consulting. Star for Life has created a strategy to increase the amount of donations from individuals. This strategy aims to provide the ability for individuals to create fundraisers in the organization's name, where the fundraiser has a specific aim specified by the creator of the fundraiser. The problem in the current version is that the fundraiser creators can't administer their own fundraisers by, for example, editing the fundraiser's description or by having the ability to list all donations towards the fundraiser. To solve this problem, Star for Life wants to create a mobile application and through that offer tools to administer the fundraisers. A mobile application makes it possible for individuals who have created a fundraiser to easily administer their own fundraiser on the go. The thesis has resulted in a mobile application for Android that makes it possible for fundraiser creators to administer their fundraisers. Limits of the WordPress REST-API have resulted in a new API being developed alongside the Android application.

Keywords: Android, Mobile Application, Charity, Java, WordPress, API, MVP

FÖRORD

Denna rapport är ett examensarbete inom mjukvaruutveckling på datatekniksprogrammet vid Chalmers Tekniska Högskola (CTH) i Göteborg. Examensarbetet utfördes våren 2017 och beskriver utvecklingen av en applikation i Android.

Särskilda uppskattningar riktas till:

Joachim von Hacht, handledare på CTH, för vägledning under examensarbetets genomförande.

Fredrik Lunde, handledare på Sigma IT Consulting, för vägledning under examensarbetets genomförande.

Stina Arnby, representant för Star for Life, för löpande återkoppling under examensarbetets genomförande.

INNEHÅLLSFÖRTECKNING

1. INLEDNING.....	1
1.1 Bakgrund	1
1.2 Syfte.....	2
1.3 Mål.....	2
1.4 Avgränsningar	3
2. TEKNISK BAKGRUND.....	4
2.1 Scrum.....	4
2.2 MVP-mönstret	4
2.3 Android.....	5
2.3.1 Arkitektur	5
2.3.2 Activity	5
2.3.3 Fragment	5
2.3.4 Intent	8
2.3.5 Volley.....	8
2.3.6 Typiskt användningsfall för en Androidapplikation	9
2.4 JSON.....	9
2.5 WordPress.....	10
2.5.1 WordPress databas	11
2.5.2 Tilläggsmoduler	11
2.5.3 WordPress REST-API	11
2.6 Easy Fundraisers.....	12
3. GENOMFÖRANDE	14
3.1 Prototyp	14
3.2 Funktionella krav	14
3.3 Icke funktionella krav	15
3.4 Exempel på användningsfall.....	15
3.4.1 Administration av en insamling	15
4. ARKITEKTUR	17
4.1 WordPress.....	18
4.2 Eget API för webbplatsen.....	18
4.3 Databas	18
4.4 Design av Androidapplikationen.....	18

4.4.1	Listning av insamlingar.....	20
4.4.2	Redigering av insamlingar	20
4.4.3	Listning av donationer	20
4.4.4	View	20
4.4.5	Model	21
4.4.6	Presenter.....	21
4.5	Användargränssnitt.....	21
4.6	Externa beroenden	22
5.	IMPLEMENTATION	23
5.1	Inledande iterationer	23
5.2	Utvärdering.....	23
5.3	Avslutande iterationer.....	23
6.	RESULTAT	25
6.1	Startvyn och sidan ”Mina insamlingar”.....	25
6.2	Sidan ”Alla insamlingar.....	25
6.3	Detaljsidan för en insamling.....	25
6.4	Redigeringssidan för en insamling	25
6.5	Sidan med donationslistan för en insamling.....	26
6.6	Sidan ”Om oss”	26
6.7	Sidan ”Om appen”	26
7.	SLUTSATS.....	27
7.1	Kritisk diskussion	27
7.2	Generaliseringar.....	27
7.3	Framtida projekt	27
	REFERENSER.....	29
	BILAGOR	

TERMINOLOGI

API	Application Programming Interface. En specifikation för hur annan mjukvara kan kommunicera med en specifik mjukvara.
MVP	Minimum Viable Product. En produkt med tillräckligt många funktioner för att tillfredsställa kunder.
OAuth	Öppet protokoll för säker autentisering.
SDK	Software Development Kit. En uppsättning utvecklingsverktyg.
SMS	Short Message Service. En tjänst för korta textmeddelanden.
SQL	Structured Query Language. Används för att kommunicera med databaser.
Widget	Mindre grafisk komponent på en webbplats som kan placeras på förutbestämda platser.
XML	Extensible Markup Language. Ett märkspråk för att utväxla data mellan olika informationssystem.

1. INLEDNING

I detta kapitel kommer examensarbetets syfte och mål att specificeras. Kapitlet inleds med arbetets bakgrund som ligger till grund för rapportens syfte och mål. Avslutningsvis specificeras arbetets avgränsningar.

1.1 Bakgrund

Välgörenhetsorganisationer har traditionellt bedrivit insamlingar genom att samla in pengar från privatpersoner med insamlingsbössor och genom annonsering. I fallet med annonsering har det krävts att givaren gjort en insättning via bank eller PlusGiro (tidigare PostGiro).

Det traditionella tillvägagångssättet har över tid förändrats till insamlingar via internet. Insamlingar via internet ger organisationer en betydligt bättre möjlighet att verka över större områden eftersom de inte begränsas av transportkostnader och tid i samma utsträckning. Organisationer kan dessutom lättare nå ut till privatpersoner som får en större inblick i organisationerna. Detta medför att privatpersonerna känner sig mer delaktiga och att de kan påverka organisationernas verksamhet.

Star for Life är en av de välgörenhetsorganisationer som börjat använda internet för att öka sina intäkter. Privatpersoner kan numera donera via webbplatsen. Detta gör det enklare och snabbare för bidragsgivare att bidra till insamlingar och därmed öka Star for Lifes intäkter.

Organisationens webbplats är utvecklad m.h.a. plattformen WordPress. WordPress gör det enkelt för icke-specialister att utveckla relativt avancerade webbplatser. Plattformen ger dessutom möjligheter till att implementera tilläggsmoduler för ökad funktionalitet för webbplatsen. Möjligheten att donera på Star for Lifes webbplats är ett exempel på en sådan modul.

Implementationen av betalningslösningen för donationer har tagits fram av företaget Sigma IT Consulting. Donationerna sker via tilläggsmodulen med betalningslösningen PayEx istället för insättningar via bank eller PlusGiro.



Scholarships

Like many young people with ambition and dreams, after having completed their undergraduate studies our young people in South Africa and Namibia want to go on to higher education. But to be able to apply for a college they need to pay a fee called application fee, and once admitted to a program each student each year need to pay another fee called registration fee. For many of our learners coming from rural areas, these fees are tough, if not impossible, to pay. This is why Star for Life has a scholarship fund from which our students can apply for funds to pay the fees.

We want to help as many young people as possible to be able to continue their studies. Thank you for supporting us in this goal, your contribution makes a big difference.

Make a donation

First name: Last name:

Firstname or Anonymous Lastname or Anonymous

E-mail: Amount (USD)

my@email.com 100

Payment Method Terms of Service

☒ Creditcard ☐ I accept

PayEx
PayEx is used for payments.

Make Donation

Started by:	Stina Arnby
Target amount:	12000 USD
Gathered amount:	0 USD
Ending at:	2018-1-1

Figur 1: Insamling på Star for Life:s webbplats

Star for Life har utvecklat en strategi för att öka antalet donationer från privatpersoner. Denna strategi syftar till att erbjuda privatpersoner möjligheten att skapa insamlingar i organisationens namn men med ett specifikt ändamål angivet av personen. Efter att en sådan insamling godkänts av Star for Life visas den på deras webbplats. Insamlingarna fungerar som riktlinjer för vad pengarna som samlas in ska gå till inom organisationen.

Problemet i nuvarande läge är att insamlarna, d.v.s. de som skapat en insamling, inte själva kan administrera insamlingen genom exempelvis redigering av insamlingens beskrivning eller möjligheten att få en lista på donationer som inkommit. Sådana verktyg skulle ge extra incitament till en fortsatt utveckling av en insamling. För att lösa detta problem vill Star for Life skapa en mobilapplikation och därigenom erbjuda verktyg för att administrera insamlingen. En mobilapplikation gör det möjligt för privatpersoner som startat insamlingar att enkelt administrera sina insamlingar var de än befinner sig.

1.2 Syfte

Syftet med examensarbetet är att undersöka huruvida en mobil applikation kan öka antalet donationer till insamlingar på Star for Lifes webbplats. Applikationen ska ge insamlarna möjligheten att administrera sina insamlingar och följa dess utveckling över tid. Detta för att ge ytterligare incitament till ökad användning av insamlingsfunktionaliteten.

1.3 Mål

Målet är att utveckla en applikation för mobila enheter för att privatpersoner ska kunna administrera sina insamlingar som skapats på Star for Lifes webbplats.

1.4 Avgränsningar

Mobilapplikationen som utvecklas i projektet kommer att utvecklas för operativsystemet Android eftersom detta är en av de största plattformarna.

2. TEKNISK BAKGRUND

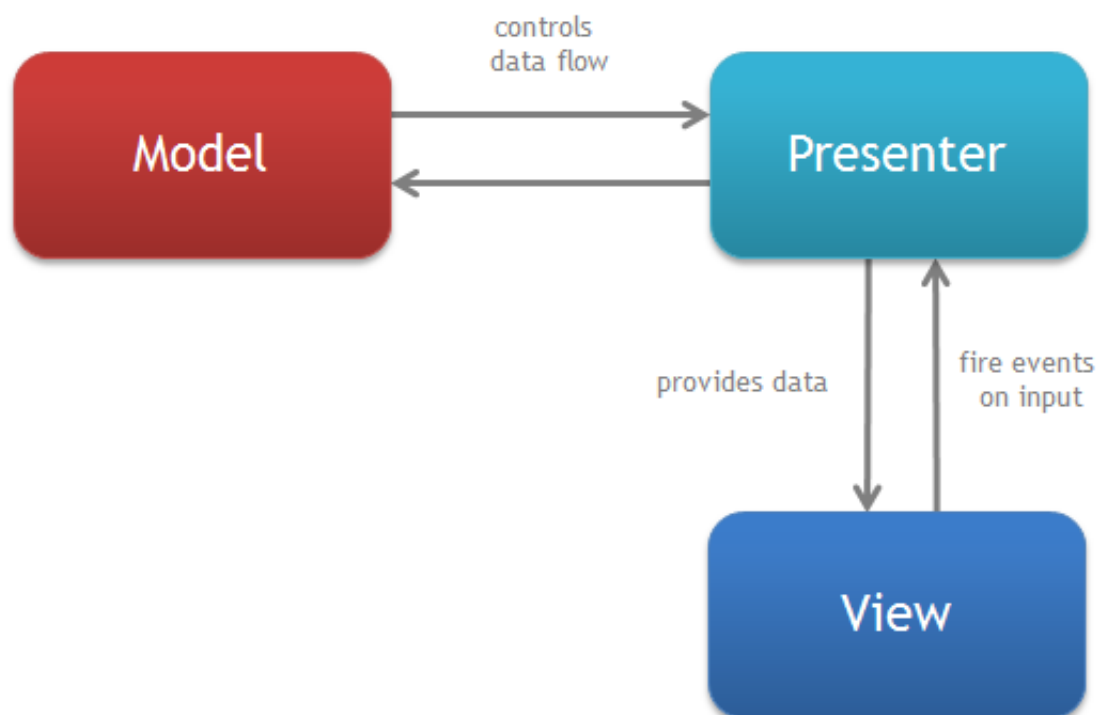
Flera olika ramverk, komponenter och verktyg har använts under projektet. Dessa beskrivs övergripande i detta kapitel.

2.1 Scrum

Scrum är en agil utvecklingsprocess som används för mjukvaruutveckling. Processen utgår ifrån att en produktägare specificerar en prioriterad lista med punkter över allt som ska göras. Utvecklingen består sedan av en iterativ arbetscykel på ofta 2–4 veckor som kallas för sprint. En sprint börjar med planering av arbetet genom val och uppdelning av uppgifter från den prioriterade listan. Under sprinten utvecklas mjukvaran genom utförande av de uppgifter som valts, i ordning av prioritet. Sprinten avslutas med en granskning av sprinten för att ge återkoppling på saker som kan förbättras inför kommande sprint (Scrum Alliance, 2017).

2.2 MVP-mönstret

MVP står för Model-View-Presenter (MVP). MVP är ett designmönster för programstruktur. Mönstret beskriver hur programmet är uppbyggt och hur de olika delarna i programmet kommunicerar med varandra. Designmönstret gör det enkelt att urskilja applikationens olika delar, vilket gör applikationen enklare att modifiera vid behov. Model är den del som hanterar datan i programmet, view hanterar allt som visas för användaren och presenter hanterar uppdateringen mellan model och view (Leiva, 2014).



Figur 2: Översikt över MVP-mönster (Sánchez, 2016).

2.3 Android

Det Linux-baserade operativsystemet Android lanserades år 2007 i samband med att Open Handset Alliance skapades. Open Handset Alliance är en sammanslutning av 86 hårdvaru-, mjukvaru- och telekommunikationsföretag med syfte att utveckla öppna standarder för mobila enheter (Android (operating system), 2017). Applikationer i Android programmeras framförallt i Java, men även andra programmeringsspråk kan användas, exempelvis C# eller LUA (Sims, 2017).

2.3.1 Arkitektur

Android består av ett antal komponenter. De applikationskomponenter en utvecklare av Android-applikationer behöver känna till är framför allt: activities, intents, services och content providers (Burnette, 2010).



Figur 3: Viktiga applikationskomponenter i Android (Java T Point, u.å.).

2.3.2 Activity

En applikation kan bestå av en eller flera activities. Dessa skapar applikationens användargränssnitt och hanterar användarens interaktion med applikationen. Alla activities har ett eget tillstånd. Varje activity ansvarar för att spara sitt tillstånd för att kunna återuppta det senare om användaren navigerar tillbaka till activityn (Burnette, 2010). I figur 4 syns livscykeln för en activity. Där syns vilka metoder som kallas vid olika tillståndsändringar.

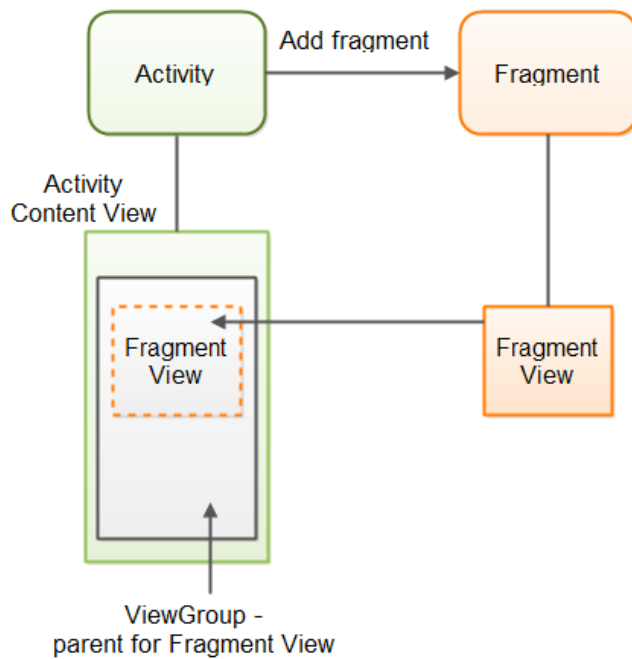
En activity använder visuella element som t.ex. views och fragments för att skapa användargränssnittet och lyssnare för att hantera kommandon från användaren. De visuella elementen är oftast definierade i en layoutfil i XML (Vogel, u.å.).

2.3.3 Fragment

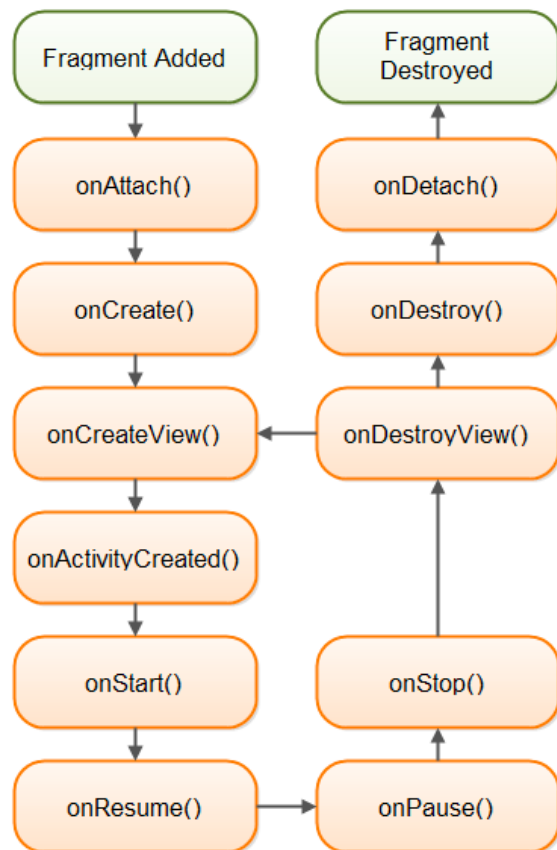
Ett fragment används för att bygga dynamiska användargränssnitt inom en activity. En activity kan använda en stor mängd fragment. Den största fördelen med fragment är att fragment förenklar uppbyggandet av användargränssnitt för olika skärmstorlekar (Chugh, 2016), exempelvis genom möjligheten att visa flera fragment samtidigt på en skärm i de fall då skärmen är tillräckligt stor.

I figur 5 visas hur ett fragment läggs till i en activity. En activity har en view som visas på skärmen. I view finns en content view där ett eller flera fragments view kan läggas.

I figur 6 visas livscykeln för ett fragment. Den innehåller de metoder som anropas under ett fragments livstid.



Figur 4: Fragment läggs till i en activity (Chugh, 2016).



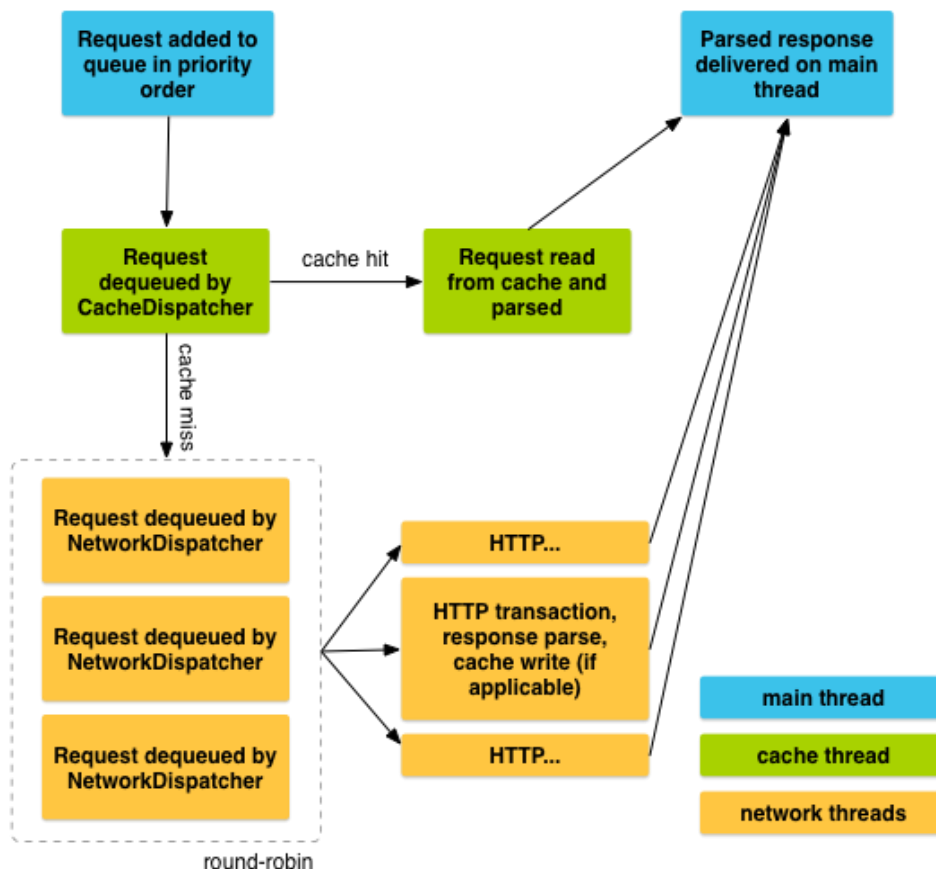
Figur 5: Livscykel för ett fragment (Chugh, 2016).

2.3.4 Intent

En intent används av applikationen för att begära en handling av en annan applikationskomponent (t.ex. activity, fragment). En sådan handling kan vara att t.ex. starta en ny activity. En intent kan skapas för att ersätta redan existerande funktionalitet, t.ex. standardapplikationen för e-post. När användaren försöker skicka ett mejl presenteras förfrågan om att använda den alternativa applikationen för e-post (Burnette, 2010).

2.3.5 Volley

Volley är ett standard HTTP-bibliotek i Android. Volley används för att göra det möjligt för en applikation att skicka HTTP-anrop. Applikationer som använder sig av volley skapar en anropskö (requestqueue) till vilken ett anropsobjekt (request) skickas. Kön använder arbetstrådar, vilka ansvarar för att utföra nätverksoperationer och läsa inkommande data. Request-objektet läser sedan inkommande data och volley skickar tillbaka det till applikationens huvudtråd (blå i bilden nedan).

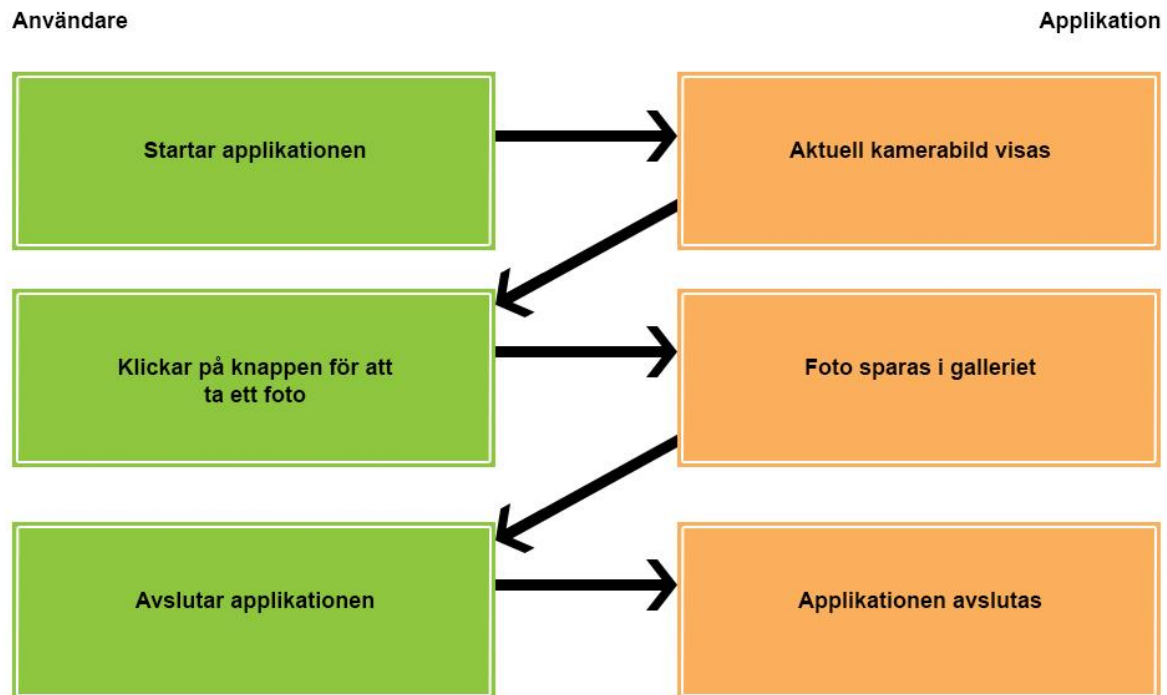


Figur 6: Anrop med Volley (Android, u.å.).

När ett objekt läggs i kön plockas det upp av en speciell tråd, cache-tråden (grön i bilden), och en bedömning görs hur operationen ska utföras. Om operationen kan utföras från cache-tråden hanteras responsen på cache-tråden och levereras sedan till huvudtråden. Om request-objektet inte kan utföras från cache-tråden placeras det i nätverkskön (gul-orange i bilden). Den först tillgängliga nätverkstråden tar sedan request-objektet, utför en HTTP-transaktion och returnerar svaret till en arbetartråd som i sin tur skriver svaret på cache och skickar tillbaka den till huvudtråden (Android Developers, u.å.).

2.3.6 Typiskt användningsfall för en Androidapplikation

Följande användningsfall beskriver en kameraapplikation i Android.



Figur 7: Typiskt användningsfall för en Android-applikation.

Användningsfallet inleds av att applikationen startas, varpå en sida visas med aktuell kamerabild. På sidan visas knappar för att ta foto, spela in video, samt en meny med kamerainställningar. När användaren klickar på knappen för att ta ett foto sparas den aktuella kamerabilden som ett foto i galleriet. Applikationen återgår sedan till startsidan och visar den aktuella kamerabilden. Väljer användaren sedan att avsluta applikationen så avslutas den.

2.4 JSON

JSON är ett textformat för lagring och utbyte av data. Det består av text skrivet med JavaScript-notation för objekt (W3schools, u.å.). Textformatet är helt oberoende av programmeringsspråket som används. JSON bygger på två strukturer: en samling av par med namn och värde, samt en lista med värden (JSON, u.å.).

Enkelt exempel på en JSON-fil syns i bild nedanför.

```

{
  "publish": {
    "name": "Published",
    "public": true,
    "queryable": true,
    "slug": "publish",
    "_links": {
      "archives": [
        {
          "href": "http://dev.starforlife.org/wp-json/wp/v2/posts"
        }
      ]
    }
  }
}

```

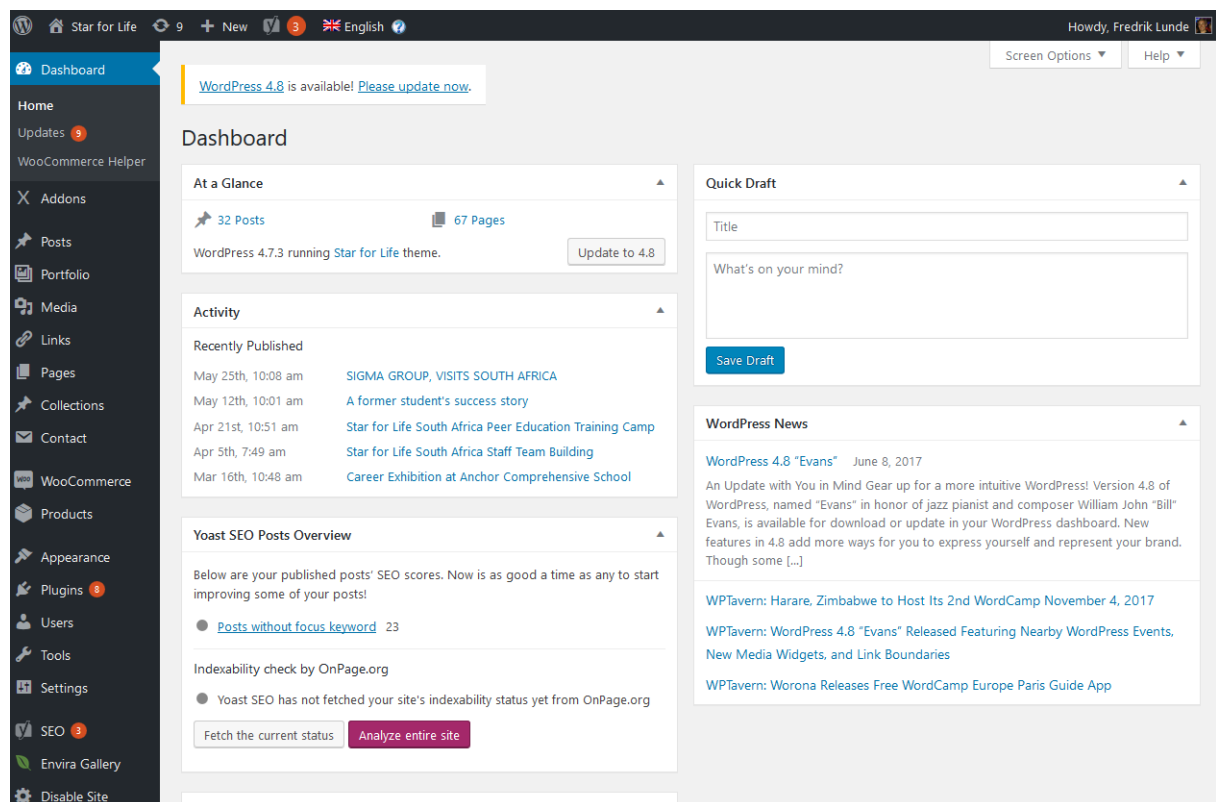
Figur 8: Exempel på returdata i från Wordpress i JSON.

2.5 WordPress

WordPress är ett publiceringsverktyg som används för att utveckla webbplatser. Det startades som ett verktyg för att skapa bloggar, men har byggts på med mer funktionalitet för att idag kunna användas för att skapa webbplatser med avancerad funktionalitet.

Utvecklingen har dragit fördel av att WordPress är byggd på öppen källkod till vilken enskilda utvecklare fritt kan bidra (Wordpress, u.å.).

Alla webbplatser som använder Wordpress har ett administratörskonto med utökade rättigheter. När WordPress har installerats på en webbplats kommer administratören åt en administrationspanel där hela webbplatsen kan administreras. Till exempel kan administratören lägga till sidor och widgets på webbplatsen (wpbeginner, u.å.).



Figur 9: Administrationspanelen i WordPress.

2.5.1 WordPress databas

Wordpress har en databas där all data lagras som webbplatser byggda med WordPress behöver för att fungera. Databasen hanterar även all data som tillhör tilläggsmoduler.

För att lägga till nytt dynamiskt innehåll på en webbplats används posttyper. Posttyperna definieras i sin tur i tabeller i databasen. Exempel på namn för posttyper som finns i en standard WordPress-installation är inlägg, sida och bilaga. Dynamiskt innehåll lagras i tabellen wp_post där en kolumn definierar posttypen för innehållet. Innehållet har ett unikt id som används för att identifiera innehållet. I tabellen wp_postmeta lagras eventuell extra data om innehåll vilket är kopplat till innehållets id (wpbeginner, u.å.).

2.5.2 Tilläggsmoduler

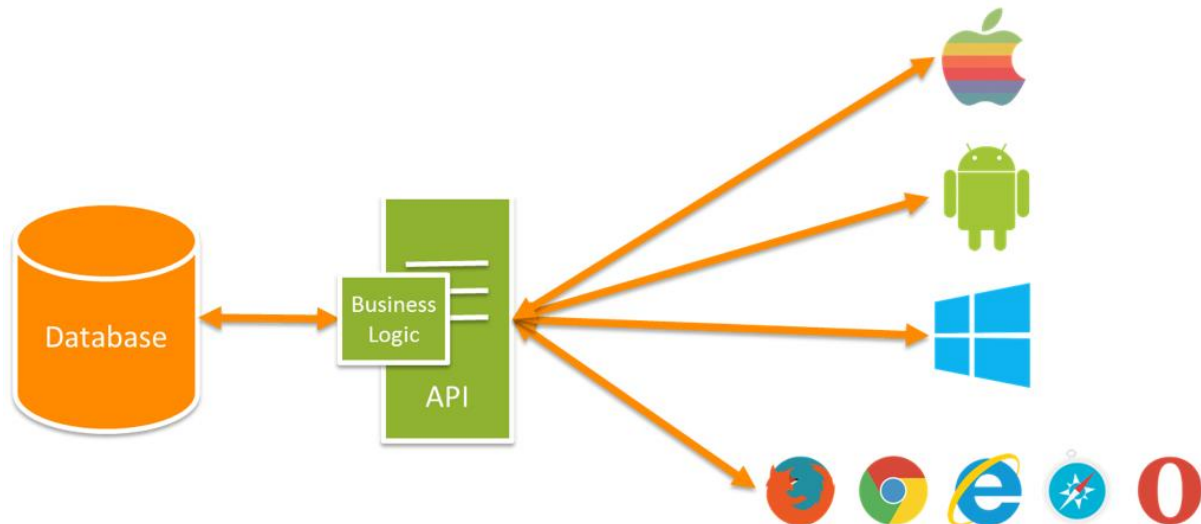
En viktig funktionalitet för administratörer är möjligheten att installera tilläggsmoduler. WordPress använder tilläggsmoduler för att öka eller ändra funktionaliteten på webbplatsen. Administratörer kan lägga till sidor och widgets på webbplatsen som finns i tilläggsmodulerna. Exempel på en tilläggsmodul är en modul som medför att administratören enkelt kan skapa en webbshop där besökare kan köpa produkter, eller en modul som hjälper administratören att lägga till annonsering på webbplatsen (Wordpress, u.å.).

Tilläggsmoduler kan skapa egna posttyper för att lagra en viss typ av innehåll i WordPress. Om man exempelvis skapar en tilläggsmodul för filmrecensioner, kan det vara en bra idé att skapa en egen posttyp för en filmrecension.

2.5.3 WordPress REST-API

Wordpress REST-API är ett gränssnitt som gör det möjligt att skicka och läsa data m.h.a. standard HTTP-metoder (t.ex. GET och POST). API:et gör det möjligt att få tillgång till data i

WordPress databas. Genom anrop på API:et kan utomstående applikationer få data i form av ett JSON-objekt. API:et erbjuder dessutom möjligheten till autentisering med kakor, OAuth, applikationslösenord samt grundläggande autentisering. Autentisering med kakor är den enda metoden som inte kräver installation av externa tilläggsmoduler (wp-api, u.å.).



Figur 10: Kommunikation med ett REST-API (Kearn, 2015).

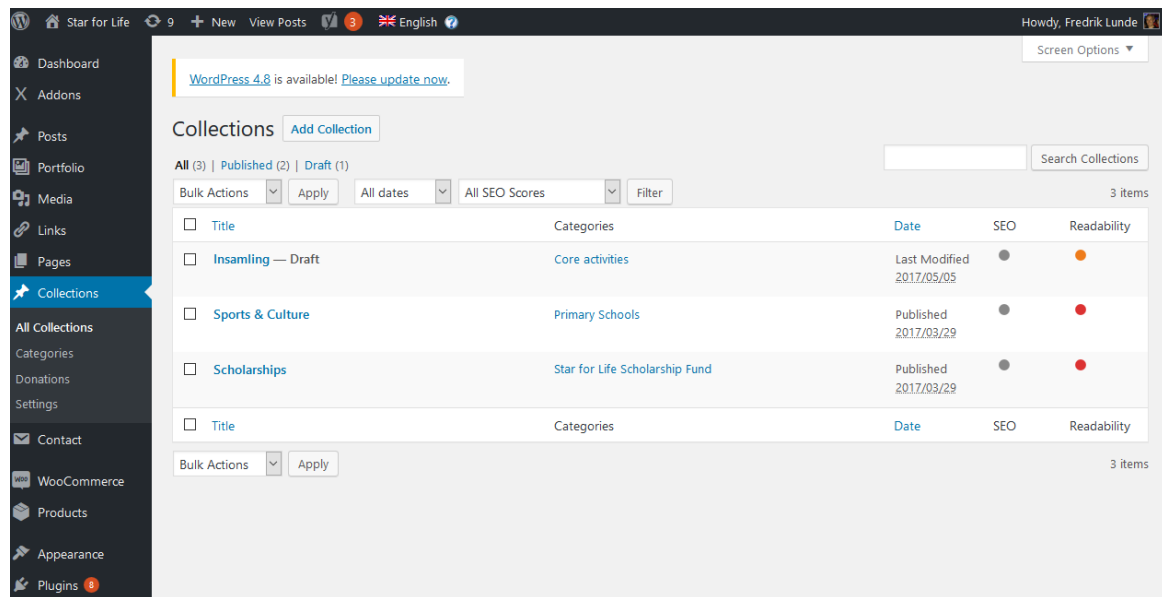
I API:et finns ingen funktionalitet för att skicka SQL-anrop av säkerhetsskäl.

2.6 Easy Fundraisers

Easy Fundraisers är en tilläggsmodul som finns installerad på Star for Lifes webbplats. Modulen erbjuder ett gränssnitt för att privatpersoner ska kunna skapa insamlingar för specifika ändamål inom organisationen. Modulen tillåter även att andra privatpersoner kan donera till insamlingen genom att fylla i ett formulär.

Easy Fundraisers skapar en egen posttyp för insamlingar. Insamlingarna lagras i tabellen wp_posts i WordPress databas. Extra information om insamlingar lagras i tabellen wp_postmeta, exempelvis vem som skapade insamlingen, när insamlingen går ut och hur mycket pengar som samlats in till insamlingen. Easy Fundraiser skapar en egen tabell, wp_donations, för lagring av donationsdata. Där lagras bl.a. datum för donation, transaktionsnummer, id för insamlingen som doneringen gick till och beloppet som donerades.

I figur 10 nedan visas administrationspanelen för insamlingarna. I denna kan administratörer granska insamlingar och acceptera dem innan de visas på webbplatsen.



Figur 11: Administrationspanelen för insamlingar för Easy Fundraisers i WordPress.

3. GENOMFÖRANDE

Examensarbetet inleds med en analys av problemområdet och den teknik som ska användas. Analysen kommer bl.a. att ligga till grund för applikationens användargränssnitt och programstruktur.

Under hela utvecklingsprocessen kommer en nära dialog att hållas med Sigma IT Consulting och Star for Life för att säkerställa att applikationen utvecklas enligt kravspecifikationen. Efter utvecklingsprocessen kommer en vecka att avsättas för avstämning, ifall justeringar i applikationen behöver göras.

3.1 Prototyp

En enklare prototyp kommer att implementeras för att ligga till grund för projektets vidareutveckling. Applikationens olika funktioner kommer att delas upp i grundläggande och mer avancerade funktioner. För att uppfylla projektets Minimum Viable Product (MVP) krävs de grundläggande funktionerna. De avancerade funktionerna krävs inte för att uppfylla projektets MVP och är därmed lågt prioriterade och görs i mån av tid.

3.2 Funktionella krav

Följande funktionella krav har framkommit:

1. Applikationen ska kunna användas i Android.
2. Det ska finnas en startsida med en meny som användaren kan använda för att navigera genom applikationen.
3. Det ska gå att skapa en ny insamling från applikationen.
4. Det ska finnas en lista över alla insamlingar.
5. Det ska finnas en lista med gamla insamlingar för insamlingar med passerat slutdatum.
6. Det ska finnas en detaljsida för varje insamling.
7. Det ska finnas en bild på insamlingens detaljsida.
8. Information om insamlingen ska finnas på insamlingens detaljsida. Detta inkluderar titel, beskrivning, hur mycket som samlats in av målet, slutdatum och namn på personen som startade insamlingen.
9. På detaljsidan ska det finnas en graf som visar hur mycket av målet som uppnåtts.
10. En användare ska kunna logga in för att se en lista på alla sina insamlingar, oavsett om de är publicerade eller ej.
11. En användare ska kunna redigera en del av informationen om sin insamling och då ska både informationen i applikationen och på webbplatsen ändras.
12. En användare ska kunna ändra bilden för sin insamling.
13. En användare ska kunna gömma och visa sin insamling på webbplatsen och i applikationen.
14. En användare ska kunna utmana en annan användare.
15. Notifikationer ska skickas till insamlaren när insamlingen nått vissa mål.
16. En notifikation ska skickas till administratören när en insamling ändrats och när den gömts eller visats på webbplatsen.
17. Laddningssymbol med text ska visas medan laddning pågår och användaren väntar.
18. Vid användarinput ska en text finnas som förklarar vad användaren ska skriva i fältet.
19. Knappar ska vara tydligt namngivna för att användaren inte ska behöva gissa vad de gör.

20. Typsnitt och färg i applikationen ska matcha Star for Lifes webbplats för att få de att se sammanhängande ut.
21. Applikationen ska använda sig av standardkomponenter från Android för att navigering genom applikationen ska ske naturligt för nya användare som är vana vid Android.
22. En lista på insamlingar ska sparas lokalt för att laddning ska gå snabbare. Detta eftersom listan inte behöver uppdateras vid varje navigering.

3.3 Icke funktionella krav

Följande icke funktionella krav har framkommit:

1. Data ska kunna lagras lokalt.

3.4 Exempel på användningsfall

Användningsfallen beskriver i detalj de funktionella kraven. Se bilaga 2 för fler användningsfall.

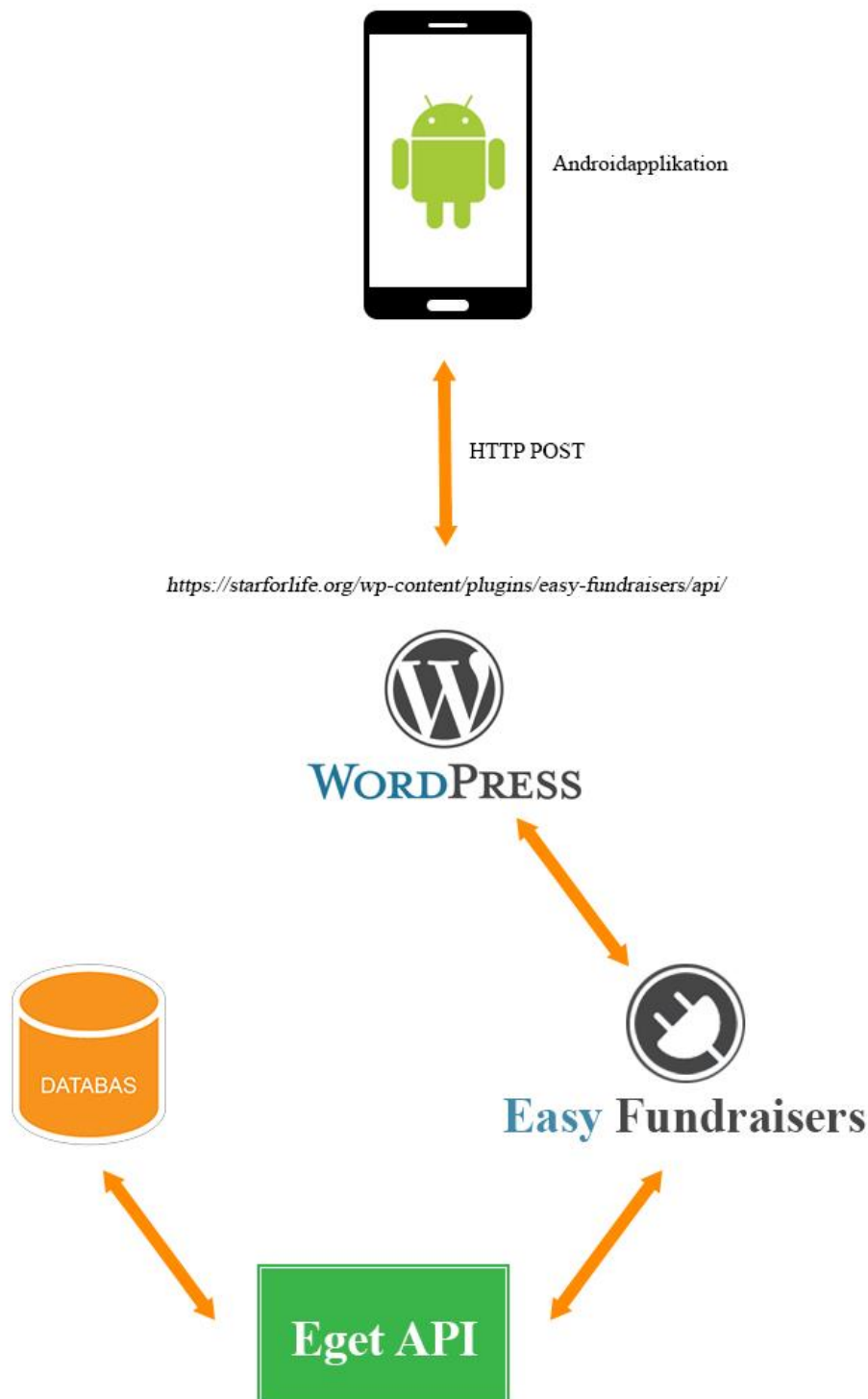
3.4.1 Administration av en insamling

S/N	Användare (handling)		Applikation (respons)
1	Startar applikationen.		
		2	Inloggningssida visas för sidan "Mina insamlingar". Användaren måste ange e-post och lösenord.
3	Anger e-post och lösenord.		
		4	Godkänner inloggning och visar sidan "Mina insamlingar" om e-posten finns och lösenordet är korrekt, annars avbryts inloggningen.
5	Trycker på en insamling i listan om inloggningen godkändes.		
		6	Detaljsidan för insamlingen visas.
7	Trycker på knappen för att lista inkomna donationer.		
		8	En lista över inkomna donationer visas.
9	Trycker på tillbakaknappen för att sedan trycka på knappen för att redigera en insamling.		
		10	Detaljsidan för en insamling visas. Därefter öppnas en sida med fält för att redigera en insamling.

11	Ändrar samtliga fält, trycker på knappen för att spara ändringarna. Öppnar därefter "options-menyn" och trycker på knappen för att gömma eller visa insamlingen.		
		12	Insamlingens titel, beskrivning och tackmejl vid donation ändras. Därefter göms insamlingen om den är synlig, eller visas om den är gömd från sidan "Alla insamlingar".
13	Öppnar "options-menyn" och trycker på knappen för att logga ut.		
		14	Loggar ut användaren och visar inloggningssidan.

4. ARKITEKTUR

I detta kapitel kommer den övergripande strukturen av hela systemet specificeras, likaså hur kommunikationen gentemot tilläggsmodulen Easy Fundraisers kommer att fungera. Systemet använder en klient-server-modell, där Androidapplikationen agerar klient och WordPress agerar server (se figur 13).



Figur 12: Arkitektur för systemet.

4.1 WordPress

WordPress agerar server för Androidapplikationen. Easy Fundraiser är installerad i WordPress och har ett eget API som nås via HTTP POST-anrop mot URL:en <http://starforlife.org/wp-content/plugins/easy-fundraisers/api/>.

4.2 Eget API för webbplatsen

Vid autentisering av användare via WordPress REST-API uppkom svårigheter eftersom det inte var en önskvärd lösning att installera externa tilläggsmoduler. Detta var inte önskvärt eftersom projektet skulle resultera i en enhetlig och fristående lösning. Autentisering med kakor är den enda metoden som inte kräver installation av externa tilläggsmoduler. Dessvärre så uppstod svårigheter eftersom denna metod kräver att API:et används inom WordPress och att användaren redan är inloggad (Wordpress, u.å.). Ett annat problem var att API:ets standardutförande inte erbjöd möjlighet att få tillgång till all data i databasen. Exempel på sådan data är data som lagras i tabeller som inte ingår i standardinstallationen av WordPress, t.ex. donationsdata i en separat tabell skapad av Easy Fundraisers. Ytterligare utveckling på webbplatsen skulle krävas för att göra detta möjligt. Eftersom applikationen kräver säker autentisering av användare och åtkomst till data som lagras i tabeller som inte ingår i standardinstallationen av WordPress fattades beslutet att utveckla ett eget API som en del av tilläggsmodulen Easy Fundraisers.

Genom att anropa det egna API:et med kommandon returneras efterfrågad data i XML-format. Viss data (t.ex. en lista över användarens egna insamlingar) kräver att en användare loggar in genom att skicka ett login-anrop eller genom att ange ytterligare parametrar.

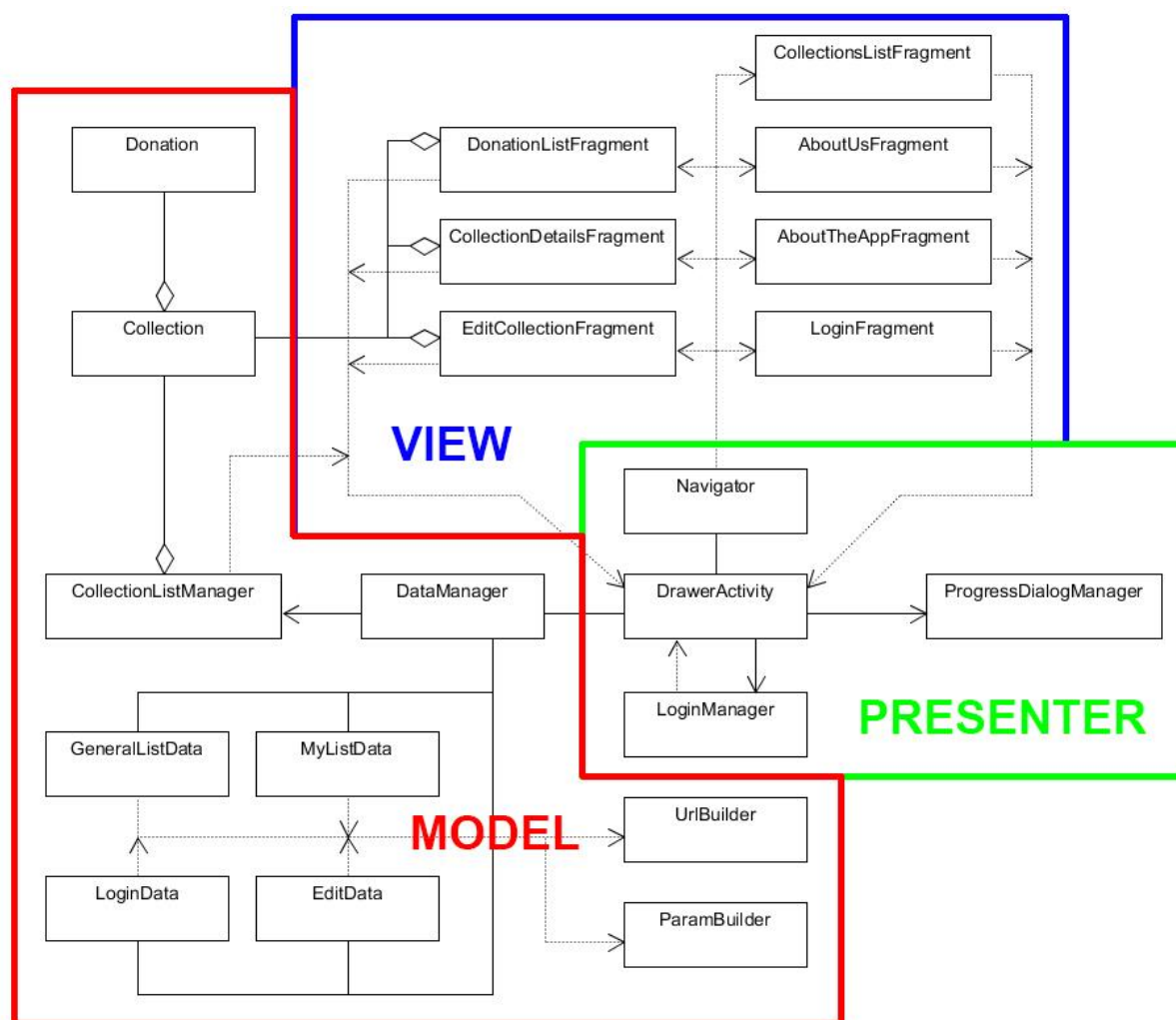
4.3 Databas

Databasen på webbplatsen innehåller data som används av WordPress och av tilläggsmodulen Easy Fundraisers. Databasen lagrar information om insamlingarna som används av Easy Fundraisers. Easy Fundraisers skapar dessutom en tabell för donationer som tillhör insamlingarna, samt en tabell för nycklarna som används i API:et.

4.4 Design av Androidapplikationen

Androidapplikationen använder API:et på webbplatsen för att autentisera användare genom att skicka HTTP-anrop för att generera en nyckel (unik och identifierbar textsträng). Nyckeln lagras både i databasen för API:et och lokalt i applikationen och används för att identifiera en inloggad användare. Nyckeln lagras med en utgångstid då den blir inaktiv som bestäms genom att lägga på 24 timmar till den tid som nyckeln aktiverades vid.

Applikationen har strukturerats enligt designmönstret MVP. Anledningen är för att tydligt urskilja applikationens olika delar (se figur 14). Ett problem med Android är att view och presenter ofta är tätt sammankopplade. Det beror på att en activity eller ett fragment inte bara hanterar det visuella, utan även interaktionen med användaren. Applikationen sköter därför viss logik i delarna för view, men merparten i presenter.



Figur 13: Klassdiagram över klientapplikationen.

Nedan följer en beskrivning av de viktigaste klasserna i klientapplikationen.

Klassnamn	Beskrivning
DrawerActivity	Activity-instansen för hela klientapplikationen. Sammankopplar de olika klasserna i applikationen.
Navigator	Hanterar navigeringen mellan fragment i klientapplikationen.
Collection	Representerar en insamling. Innehåller information om en insamling.
Donation	Representerar en donation till en insamling. Innehåller data om donationen.
DataManager	Hanterar anrop mellan klientapplikationen och API:et på webbplatsen.
CollectionListManager	Hanterar den temporära lagringen av insamlingar när klientapplikationen körs.

LoginManager	Hanterar inloggningen av en användare.
ProgressDialogManager	Hanterar visningen av en laddningsruta.

4.4.1 Listning av insamlingar

Processen inleds av att klassen Navigator byter fragment till fragmentet CollectionsListFragment som används för att visa insamlingar i en lista. En laddningsruta visas samtidigt som klassen CollectionListFragment (genom klassen DataManager) hämtar data om insamlingarna från API:et på webbplatsen. CollectionListFragment läser in returdatan och byter därefter innehållet i listan för insamlingarna till insamlingarna som precis hämtats. Nu göms laddningsrutan och användaren kan interagera med listan över de nyligen hämtade insamlingarna.

4.4.2 Redigering av insamlingar

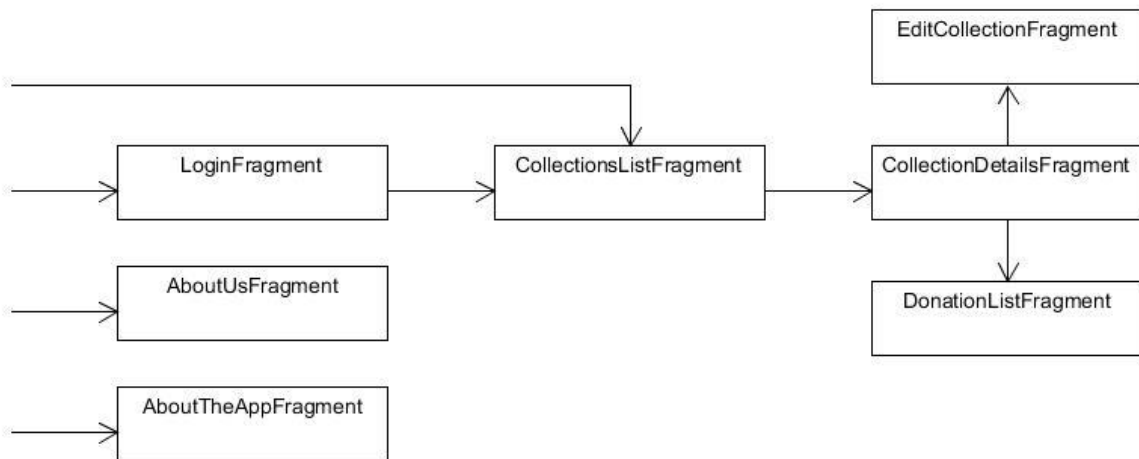
Processen förutsätter att fragmentet EditCollectionFragment är aktivt. När en användare initierat en redigering av en insamling anropas klassen EditData (genom DataManager) som skickar en förfrågan om redigering till API:et på webbplatsen. EditData läser in returdatan och returnerar huruvida redigeringen lyckats eller ej.

4.4.3 Listning av donationer

Processen inleds av att klassen Navigator byter fragment till fragmentet DonationListFragment som används för att visa donationer i en lista. En laddningsruta visas samtidigt som klassen CollectionListFragment (genom klassen DataManager) hämtar data om insamlingarna från API:et på webbplatsen. CollectionListFragment läser in returdatan och byter därefter innehållet i listan för donationerna till donationerna som precis hämtats. Nu göms laddningsrutan och användaren kan interagera med listan över de nyligen hämtade donationerna.

4.4.4 View

View hanterar fragmenten. Den har hand om logiken för vad som visas på det specifika fragmentet. Fragmentet för detaljsidan hanterar t.ex. input från användaren vid redigering och listfragmentet skickar vidare användare till detaljsidan för insamlingen vid val av insamling i listan över insamlingar.



Figur 14: Navigation mellan användargränssnitt. CollectionListFragment, LoginFragment, AboutUsFragment, AboutTheAppFragment kan åtkommas från applikationens alla tillstånd.

4.4.5 Model

Model hanterar datan samt nätverksdelen av applikationen som hämtar och skickar data till och från API:et och lagrar det lokalt.

4.4.6 Presenter

Presenter är grunden i applikationen. Den innehåller den aktivitet som hanterar navigering mellan fragment, laddningsrutan som visas när användare väntar på övergångar mellan olika tillstånd i applikationen samt status för inloggning.

4.5 Användargränssnitt

Användargränssnittet har utvecklats i nära samverkan med en representant för Star for Life för att det i möjligaste mån ska anpassas för deras önskemål. Det har varit prioritet att anpassa applikationens användargränssnitt utefter deras webbplats, som följer ett enkelt och modernt utseende med solida färger. En grå färgton används generellt som basfärg över samtliga sidor på webbplatsen och denna färgton har applicerats på applikationen för att erhålla ett enhetligt utseende mellan de två plattformarna.

Eftersom syftet med applikationen är att privatpersoner som startat insamlingar (s.k. insamlare) på webbplatsen ska kunna administrera sina insamlingar har användargränssnittet utvecklats för att på ett användarvänligt sätt tillgodose detta behov. Detta innebär alltså att funktioner som främst kommer att användas av insamlare, t.ex. åtkomst av egna insamlingar och redigering av dessa, prioriteras högre än exempelvis listning av insamlingar skapade av alla insamlare.

4.6 Externa beroenden

Applikationen använder sig av ett externt grafbibliotek som kallas MPAndroidChart för att rita upp grafen på detaljsidan. Applikationen använder sig av HTTP-biblioteket Volley för alla HTTP-anrop.

5. IMPLEMENTATION

I detta kapitel beskrivs utvecklingen av applikationen. Utvecklingsprocessen bygger på arbetsmetodiken Scrum med en vecka per sprint.

5.1 Inledande iterationer

Första iterationen bestod av att utveckla en prototyp där Androidapplikationens utseende beskrevs i bilder. Under samma iteration utvecklades en enkel applikation bestående av en tom sida för att utforska och göra oss bekanta med miljön.

I andra iterationen testades anrop till WordPress REST-API. Vid slutet av iterationen bestod applikationen av en activity som visade en lista med alla insamlingar genom anrop till API:et.

Början på ett autentiseringssystem utvecklades i tredje iterationen. I samma iteration påbörjades arbetet med applikationens utseende och användargränssnitt. Under iterationen undersöktes möjligheterna för API:et ytterligare, vilket resulterade i att problem upptäcktes. Lösningen blev att utveckla ett eget API till Easy Fundraisers.

I fjärde iterationen utvecklades ett eget API som implementerades som en del av Easy Fundraisers för att användas av applikationen. Återkoppling från en kontaktperson på Star for Life gällande användargränssnittet i applikationen resulterade i ändringar av utseendet till ett mer sammanhängande användargränssnitt med Star for Lifes webbplats. Detta skedde genom att ersätta loggan och färgerna som användes i applikationen.

5.2 Utvärdering

Efter varje iteration gjordes en kort granskning av hur arbetet under iterationen fungerat och vad som kunde förbättras. Det största problemet som upptäcktes var att API:et inte fungerade som tänkt vilket ledde till mer arbete än vad som var planerat för. Detta kunde upptäckts tidigare om en ordentlig granskning gjorts tidigare över hur API:et fungerade.

Det som fungerade bra i alla iterationerna var att hålla en regelbunden kontakt med Star for Life. Detta resulterade i en tydlig prioritering över vilka funktioner som skulle utvecklas och gav snabb återkoppling på applikationens utseende.

5.3 Avslutande iterationer

I iteration fem skapades detaljsidor för insamlingarna i applikationen. Detaljsidorna består av ett fragment som visas när användaren navigerar dit från sidor med listor över insamlingar ("Alla insamlingar", "Mina insamlingar" samt "Gamla insamlingar"). Detaljsidorna skiljer sig åt om användaren är inloggad och har skapat insamlingen eller om användaren inte är inloggad eller inte har skapat insamlingen. Exempelvis ska en insamling som användaren skapat innehålla en knapp för att redigera insamlingen. Samma knapp ska gömmas om användaren inte är inloggad eller om användaren inte skapade insamlingen.

Ett problem som upptäcktes var att Star for Lifes webbplats är långsam att hämta data ifrån vilket medförde långa responstider vid hämtning av data. Detta löstes genom att lagra datan lokalt på telefonen istället för att alltid hämta den. Denna lösningen ledde till att applikationen inte behövde hämta en lista med alla insamlingar varje gång användaren navigerar till listan. Lösningen gjorde att applikationen blev mycket snabbare vid uppstart och återöppning av

“Alla insamlingar”-sidan. Användaren kan nu istället manuellt tvinga applikationen att uppdatera listan på alla insamlingar.

Det diskuterades även ifall listan med en inloggad användares insamlingar skulle lagras lokalt, eftersom användare enligt kravspecifikationen ska förbli inloggade även efter att applikationen avslutas. Problemet som uppstod när detta implementerades var att nyckeln som autentiserar användaren kunde bli ogiltig under tiden som applikationen varit stängd. Det beslutades därav att användare måste logga in på nytt efter att applikationen avslutats.

Slutligen implementerades mejlfunktionalitet för att meddela administratören för Star for Life via mejl när en insamling ändrats. På detaljsidan implementerades en graf som tydliggör procentuellt hur mycket av målet som insamlingen uppnått. Slutligen prioriterades finjustering av applikationens utseende samt testning för att säkerställa att all funktionalitet fungerade korrekt innan lanseringen av applikationen.

6. RESULTAT

I detta kapitel beskrivs det slutliga resultatet. Det extra arbetet som krävdes för att skapa ett eget API resulterade i att inte alla punkter i kravspecifikationen uppnåddes. De punkterna är:

1. Det ska gå att skapa en ny insamling från applikationen.
2. En användare ska kunna ändra bilden för sin insamling.
3. En användare ska kunna utmana en annan användare.
4. Notifikationer ska skickas till insamlaren när insamlingen nått vissa mål.

6.1 Startvyn och sidan "Mina insamlingar"

Sidan "Mina insamlingar" visas om en användare är inloggad och därmed har en giltig autentiseringsnyckel. Sidan listar användarens insamlingar, som kan väljas för att visa en detaljsida för insamlingen. Insamlingar visas för användaren även medan den väntar på att bli godkänd för publicering eller när en insamling blivit dold från webbplatsen. Om användaren inte är inloggad visas applikationens inloggningssida. Se bilaga 2 för bildrepresentation.

6.2 Sidan "Alla insamlingar"

En lista över alla publicerade insamlingar visas på sidan. Till höger i listan visas procentuellt hur mycket av målet för insamlingen som är uppnått. Målet för insamlingen är en summa pengar som bestäms vid skapelsen av insamlingen. Menyn som öppnas när användaren klickar högst uppe till höger på sidan (de tre vertikala prickarna) erbjuder möjligheten att visa utgångna insamlingar. Slutdatumet på en insamling i förhållande till dagens datum avgör om den är utgången. Se bilaga 3 för bildrepresentation.

6.3 Detaljsidan för en insamling

Detaljsidan (se bilaga 4) för en insamling visas när användaren väljer en insamling i någon av sidorna med listor över insamlingar. På detaljsidan visar en cirkelgraf hur långt det procentuellt är kvar till att nå målet för insamlingen. Målet för insamlingen är en tidigare bestämd summa pengar. Inuti grafen visas bilden som skaparen av insamlingen tilldelat insamlingen. Längre ner på sidan finns titel och beskrivning för insamlingen. Under beskrivningen finns information om hur mycket pengar som samlats in till insamlingen i förhållande till insamlingens mål. Det finns också information om slutdatum och vem som skapat insamlingen.

Användaren kan trycka på en knapp för att dela insamlingen via t.ex. sociala medier, SMS eller e-post. Det finns också en knapp för att donera till insamlingen. Syftet med examensarbetet var att utveckla en lösning för att privatpersoner som startat insamlingar på webbplatsen ska kunna administrera sina insamlingar. Det var därav inte prioriterat att implementera möjlighet för donering via Androidapplikationen. Knappen för att donera till insamlingen öppnar istället insamlingen i webbläsaren eftersom donering kan ske via webbplatsen.

Om insamlingen skapats av användaren som är inloggad så visas ytterligare två knappar på detaljsidan för insamlingen (se bilaga 5). En knapp öppnar redigeringssidan för insamlingen, varpå den andra öppnar sidan med donationslistan för en insamling.

6.4 Redigeringssidan för en insamling

Redigeringssidan erbjuder en inloggad användare som skapat en insamling möjligheten att redigera insamlingens titel, beskrivning och tackmejlet som skickas till de som donerar till

insamlingen. Menyn som öppnas när användaren klickar högst upp till höger på sidan (de tre vertikala prickarna) erbjuder möjlighet att gömma eller visa en insamling på webbplatsen. Samtliga redigeringar medför att administratören på Star for Life får ett mejl som meddelar att en ändring utförts. Se bilaga 6 för bildrepresentation.

6.5 Sidan med donationslistan för en insamling

Sidan med donationslistan ger inloggad användare som skapat en insamling möjligheten att se när och hur mycket pengar personer donerat till användarens insamling. Donationerna är anonyma för att det inte ska gå att se om en specifik person har donerat. Se bilaga 7 för en bildrepresentation.

6.6 Sidan "Om oss"

Sidan "Om oss" innehåller en kort introduktion för Star for Life och en länk som användare kan använda för att hitta mer information om organisationen. Se bilaga 8 för bildrepresentation.

6.7 Sidan "Om appen"

Sidan "Om appen" innehåller information för användare som vill veta hur Androidapplikationen fungerar och vilka funktionaliteter som finns. Se bilaga 9 för bildrepresentation.

7. SLUTSATS

Utifrån redovisat resultat anser vi att examensarbetet har uppfyllt sitt mål. Examensarbetet har resulterat i en applikation där Star for Lifes insamlare kan administrera sina insamlingar och följa dess utveckling.

Det som inte kunde uppfyllas är arbetets syfte. Syftet var att undersöka huruvida framtagningen av en mobil applikation för insamlare kunde öka antalet donationer till insamlingarna. Dessvärre så har Star for Life inte publicerat applikationen inom en rimlig tid från arbetets avslutande, vilket medfört att ingen förändring i användningen av insamlingarna kunnat undersökas. Detta har i sin tur inneburit att ökningen av antalet donationer inte heller kunnat undersökas.

7.1 Kritisk diskussion

I kravspecifikationen har 19 av 23 punkter uppfyllts. Det finns två anledningar till att samtliga punkter inte uppfylldes. Den första anledningen var att flera punkter som hade låg prioritet och uppskattades ta mycket tid att färdigställa tillkom efter examensarbetets uppstart. Den andra anledningen var att Wordpress REST-API inte fungerade som förväntat och inte erbjöd den funktionalitet som eftersträvades vid utvecklingen av applikationen. Detta innebar oförutsett mycket mer arbete eftersom ett nytt API behövde utvecklas. Eftersom det nya API:et krävdes av de grundläggande funktionerna i applikationen behövde det prioriteras innan utvecklingen av nya funktioner kunde ta vid.

Trots att målet hade uppfyllts fanns inte tid innan examensarbetets avslutande för att undersöka huruvida antalet donationer ökade efter publiceringen av applikationen. Undersökningen hade kunnat visa på betydelsen av att ha en mobil applikation för att involvera användare i insamlingarnas utveckling och resten av organisationens arbete. Undersökningen hade även kunnat ge svar på om den ökande användningen av insamlingarna i sin tur kunde öka antalet donationer. Detta genom att få insamlare att känna sig mer involverade och mer manade att dela insamlingen i sin bekantskapskrets.

7.2 Generaliseringar

Syftet med examensarbetet har varit att undersöka huruvida en mobil applikation kan öka antalet donationer till insamlingar på Star for Lifes webbplats. Under arbetet har det konstaterats att denna typ av undersökning även kan vara intressant i andra sammanhang. Sådana sammanhang kan exempelvis vara huruvida en mobil applikation kan bidra till ökad användning av en webbshop på en webbsida. Dessvärre har arbetet inte kunnat ge några rekommendationer eller dra några slutsatser huruvida en mobil applikation kan öka användningen av en viss funktionalitet på en webbsida.

7.3 Framtida projekt

I detta avsnitt följer förslag till fortsatt arbete.

- För att erbjuda en enhetlig Androidapplikation för insamlingar för Star for Life så bör funktionalitet för att donera till en insamling implementeras i applikationen. En sådan implementation skulle göra det snabbare att donera till en insamling eftersom användaren inte behöver öppna webbplatsen.

- En av de punkter som inte hanns med i projektet med var att meddela användare när deras insamlingar når vissa delmål, t.ex. när 25%, 50% och 75% av målet uppnåts. Detta skulle kunna utvecklas genom notifikationer i Android, vilket fungerar som påminnelse och motivation till att fortsätta dela insamlingen.
- Android är bara ett av de större mobila operativsystemen. En IOS-applikation skulle öppna upp för fler användare, vilket i sin tur skulle öka användandet av insamlingsfunktionaliteten. I detta projektet fanns inte tid för utveckling till fler än ett operativsystem.
- Utförande av undersökningen som inte hanns med - huruvida applikationen ökade användningen av insamlingsfunktionaliteten och ökade antal donationer till insamlingarna.

REFERENSER

Android (u.å.). Anrop med Volley. [Elektronisk bild]

<https://developer.android.com/training/volley/simple.html> [Hämtad 23 Maj 2017].

Android Developers. (u.å.). *Sending a Simple Request*.

<https://developer.android.com/training/volley/simple.html> [Hämtad 19 Maj 2017].

Android (operating system). (2017).

<http://www.immagic.com/eLibrary/ARCHIVES/GENERAL/WIKIPEDI/W120716A.pdf>
[Hämtad 31 Mar. 2017].

Burnette, E. (2010). *Hello, Android*. 3rd ed. Pragmatic Bookshelf, pp.39-40.

Chugh, A. (2016). *Android Fragment Lifecycle - JournalDev*. JournalDev.

<http://www.journaldev.com/9266/android-fragment-lifecycle> [Hämtad 28 Maj 2017].

Chugh, A. (2016). Fragment läggs till i en activity. [Elektronisk bild]

<http://www.journaldev.com/9266/android-fragment-lifecycle> [Hämtad 28 Maj 2017].

Chugh, A. (2016). Livscykel för ett fragment. [Elektronisk bild]

<http://www.journaldev.com/9266/android-fragment-lifecycle> [Hämtad 28 Maj 2017].

Java T Point (u.å.). Viktiga applikationskomponenter i Android. [Elektronisk bild]

<https://www.javatpoint.com/android-core-building-blocks> [Hämtad 5 Jun. 2017].

JSON. (u.å.). *Introducing JSON*. <http://www.json.org/> [Hämtad 5 Jun. 2017].

Kearn, M. (2015). *Kommunikation med ett REST-API*. [Elektronisk bild]

<https://blogs.msdn.microsoft.com/martinkearn/2015/01/05/introduction-to-rest-and-net-web-api/> [Hämtad 16 Maj 2017].

Leiva, A. (2014). *MVP for Android: how to organize the presentation layer*. antonioleiva.

<https://antonioleiva.com/mvp-android/> [Hämtad 13 Jun. 2017].

Sánchez, P. (2016). Översikt över MVP-mönster. [Elektronisk bild]

<http://blog.karumi.com/interfaces-for-presenters-in-mvp-are-a-waste-of-time/> [Hämtad 23 Maj 2017].

Scrum Alliance. (2017). *Learn About Scrum*. <https://www.scrumalliance.org/why-scrum>

[Hämtad 28 Maj 2017].

Sims, G. (2017). *I want to develop Android apps – What languages should I learn?*. Android

Authority. <http://www.androidauthority.com/want-develop-android-apps-languages-learn-391008/> [Hämtad 13 Jun. 2017].

Vogel, L. (u.å.). *Getting started with Android development - Tutorial*. Vogella.com.

<http://www.vogella.com/tutorials/Android/article.html> [Hämtad 5 Apr. 2017].

Wordpress. (u.å.). *About WordPress*. Available at: <https://wordpress.org/about/> [Hämtad 5 Apr. 2017].

Wordpress. (u.å.). *Authentication*. <https://developer.wordpress.org/rest-api/using-the-rest-api/authentication/> [Hämtad 5 Jun. 2017].

wpbeginner. (u.å.). *What is: Admin Area*. <http://www.wpbeginner.com/glossary/admin-area/> [Hämtad 13 Jun. 2017].

wpbeginner. (u.å.). *What is: Database*. <http://www.wpbeginner.com/glossary/database/> [Hämtad 13 Jun. 2017].

wp-api. (u.å.). *About the WordPress REST API*. <http://v2.wp-api.org/> [Hämtad 13 Jun. 2017].

W3schools. (u.å.). *JSON Introduction*. https://www.w3schools.com/js/js_json_intro.asp [Hämtad 26 Maj 2017].

BILAGOR

BILAGA 1. Sid 1(10)

Visning, donering och delning av annans insamling:

S/N	Användare (handling)		Applikation (respons)
1	Startar applikationen.		
		2	Inloggningssida visas för sidan "Mina insamlingar". Användaren måste ange e-post och lösenord.
3	Öppnar "drawer-menyn" och klickar på "Alla insamlingar".		
		4	En lista över alla insamlingar visas.
5	Öppnar "options-menyn" och trycker på knappen för att visa utgångna insamlingar.		
		6	En lista över utgångna insamlingar visas.
7	Trycker på en insamling i listan.		
		8	Detaljsidan för insamlingen visas.
9	Trycker på knappen för att dela insamlingen.		
		10	Öppnar en intent för att dela insamlingen.
11	Trycker på knappen för att donera till insamlingen.		
		12	Öppnar intent för att donera till insamlingen via webbplatsen.

Visning av informationssidorna:

S/N	Användare (handling)		Applikation (respons)
1	Startar applikationen.		
		2	Inloggningssida visas för sidan "Mina insamlingar". Användaren måste ange e-post och lösenord.
3	Öppnar "drawer-menyn" och klickar på "Om oss".		
		4	Information om Star for Life visas på en sida.
5	Öppnar "drawer-menyn" och klickar på "Om appen".		
		6	Information om Star for Life visas på en sida.

Sidan “Mina insamlingar” innan inloggning (till vänster) och efter inloggning (till höger):

☰ Mina insamlingar	☰ Mina insamlingar ⋮
 E-mail: min@email.com Lösenord: lösenord LOGGA IN	 Insamling

Sidan “Alla insamlingar”:

☰	Alla insamlingar	⋮
	Scholarships	
	Sports & Culture	1 %

Detaljsidan för en insamling:

☰ Info om insamling	☰ Info om insamling ⋮
 72%	Sports & Culture Interest in sports and music is high among students in our schools in South Africa and Namibia. Unfortunately the schoolyards echo with the sound of lack of necessities such as balls, goals and jungle gyms, and music classes are conducted with very few instruments if any. In these remote villages, what can be regarded as small things make a big difference. Musical instruments and sports equipment are expensive and in time are worn out. By supporting this collection, you can play a big role in a child's first experience of the tones from a xylophone or guitar, or the feeling of scoring in a soccer game.
Sports & Culture Interest in sports and music is high among students in our schools in South Africa and Namibia. Unfortunately the schoolyards echo with the sound of lack of necessities such as balls, goals and jungle gyms, and music classes are conducted with very few instruments if any. In these remote villages, what can be regarded as small things make a big difference. Musical instruments and sports equipment are expensive and in time are worn out. By supporting this collection, you can play a big role in a child's first experience of the tones from a xylophone or guitar, or the feeling of scoring in a soccer game.	Gathered: 5400/7500 USD
	Slutdatum: 2018-4-3
	Startad av: Stina Arnby
	DELA
	DONERA

Detaljsidan för en insamling då användaren är skaparen av insamlingen:

☰

Info om insamling

⋮



Insamling

Den här insamlingen går till att...

Gathered: 0/10 USD

Slutdatum: 2017-1-1

Startad av: Micaela Carlsson

Status: Utkast

REDIGERA

DONATIONER

Redigeringssidan för en insamling:

 **Redigera insamling** 

Titel:
Om du ändrar den här kommer titeln på din insamling att ändras.

Insamling

Beskrivning:
Om du ändrar den här kommer din insamlings beskrivning att ändras.

Den här insamlingen går till att...

Tackmejl:
Om du ändrar den här kommer tackmejllet som skickas till folk som donerar att ändras.

Tack för ditt bidrag!

Kontakta oss på collections@starforlife.se om du vill
ändra slutdatum, mål o.s.v.

SPARA

Donationslistan för en insamling:

☰	Donationer	⋮
100.0 SEK	2017-03-22 16:52:00	
45.0 SEK	2017-03-22 16:52:00	
60.0 SEK	2017-03-22 17:04:41	
84.0 SEK	2017-03-22 17:04:41	
50.0 SEK	2017-03-22 17:08:36	
65.0 SEK	2017-03-22 17:08:36	

Sidan “Om oss”:

 Om oss


Star for Life
empowering young people

Star for Life har som vision att motivera och inspirera ungdomar till bättre liv och en ökad framtidstro. Ungdomar stimuleras att fokusera på sin utbildning och ta hand om sin hälsa. I Sydafrika och Namibia används programmet för att minska spridningen av hiv och aids. Organisationen bedrivs utan vinstsyfte och är politiskt och religiöst oberoende.

Mer information på:
www.starforlife.se

Sidan “Om appen”:

