

UPPSALA UNIVERSITET CAMPUS GOTLAND
Institutionen för informatik och media

Examinator: Jonas Sjöström, jonas.sjostrom@im.uu.se
Handledare: David Haskiya, Riksantikvarieämbetet, david.haskiya@raa.se
Handledare: Jakob Bandelin, Uppsala universitet, jakob.bandelin@im.uu.se
Författarnas e-postadresser: maxcollin@gmail.com, alfred.bjersander@gmail.com
Utbildningsprogram: Kandidatprogram i systemvetenskap Programvaruteknik, 180 HP
Omfattning: 5680 ord inklusive bilagor
Datum: 2018-05-23



white
clothing
sleeve
dress shirt
shirt
shoulder
standing
blouse
top
fashion
black and white
textile
outerwear
gentleman
vintage clothing
neck
collar
formal wear
jacket
fashion design

Fotograf: Hilding Mickelsson

Programvaruteknik B, 15 hp

Innehållsbaserad bildsök

Förbättrad informationskvalitet för det fotografiska
kulturarvet

Alfred Bjersander
Max Collin

Sammanfattning

Rapporten presenterar utveckling och utvärdering knutet till uppdraget "Innehållsbaserad bildsök" som utfördes hos Riksantikvarieämbetet (RAÄ) i Visby. Uppdragets mål var att utveckla en prototyp som kunde visa på eventuella styrkor eller svagheter med att tillföra maskin-extraherade attribut till bilder som tillgängliggörs via RAÄ:s application programming interface (API). I projektet användes Google Visions tjänst tillsammans med RAÄ:s K-samsök API för att sammanfoga bildinformation. Resultatet presenterades via en webbapplikation, som ger möjlighet att söka i bildmaterialet genom att dels kombinera etiketter samt söka på kombinationer av färger.

Prototypen visade på att den tillförda informationen ökar möjligheten att söka i bildmaterialet. Utvärderingen visade också på förekomsten av falska negativ, vilket bryter mot användarens intuitiva förväntningar. Kombinationer av valda etiketter kan reducera resultatrymden där så inte borde vara fallet. Exempelvis minskar antalet träffar om användaren kombinerar etiketterna "människa" och "porträtt".

Förmågan att söka bland bilder utifrån färginformation är i K-samöks befintliga metadata begränsad till manuellt annoterade textfält. Prototypen tillför ett nytt sätt att utforska bildsamlingarna genom att kombinera färger. Funktionaliteten rimmar väl med verksamhetens tillgänglighörandemål.

Nyckelord: Bildsök, Riksantikvarieämbetet, K-Samsök, Google Vision.

Abstract

This report presents development and evaluation of a prototype, commissioned by Riksantikvarieämbetet (RAÄ) in Visby. The prototype's purpose is to show the potential strengths or weaknesses of adding machine extracted attributes to images made available through RAÄ's Application Programming Interface (API). Google Vision's service was used in conjunction with the RAÄ K-Samsök API to merge image information. The result is presented via a web application that allows a user to search images by combining labels as well as combinations of colors.

The evaluation shows that added information increases the ability to search the image collections and reinforces the user experience. Examinations of individual labels shows that machine-extracted information increases searchability by providing information missing in existing metadata. The examination also shows the presence of false negatives, which violates the user's intuitive expectations. Combinations of selected labels can reduce the result space, which should not be the case. For example, the number of hits decreases if the user combines the labels "human" and "portrait".

The ability to search images based on color is restricted to manually annotated text fields in K-Samsök's existing metadata. The Prototype presents a new approach to exploring the image collections by combining colors. This functionality rhymes well with the organizational goals of RAÄ.

Keywords: Image search, Riksantikvarieämbetet, K-Samsök, Google Vision.

Förord

Vi vill tacka de personer som har varit behjälpliga under projektiden, David Haskya som utformade projektförslaget samt gav avgörande förslag på teknik och inspirationsunderlag, Albin Larsson som vid ett flertal tillfällen har hjälpt oss att lösa tekniska problem, hela utvecklingsavdelningen där vi bereddes arbetsplats och en välkomnande miljö att arbeta i, Jakob Bandelin för stöd och idéer vid författandet av denna rapport.

Innehåll

1	Uppdrag.....	7
1.1	Bakgrund	7
1.2	Projektmål.....	7
1.3	Avgränsningar.....	8
1.4	Förväntningar	8
1.5	Kunskapsinventering.....	9
1.6	Delförfattarnas bidrag.....	9
1.7	Disposition.....	9
2	Kunskapsbas.....	10
2.1	Definition av högnivåproblem.....	10
2.2	Liknande system	11
2.3	Relaterad forskning / litteratur	11
2.4	Implikationer för uppdraget	12
3	Programvara.....	13
3.1	Övergripande beskrivning	13
3.2	Programbibliotek och teknik.....	15
3.2.1	K-samsök API	15
3.2.2	Google Vision API	15
3.2.3	Elasticsearch	16
3.2.4	React / Reactivesearch	16
3.2.5	Node-script	17
4	Utvärdering.....	18
4.1	Urvalsmetod	18
4.2	Tolkande	18
4.3	Demonstrativ	20
4.4	Metodologisk.....	21
5	Diskussion	23
5.1	Slutsatser.....	23
5.2	Rekommendationer.....	23
5.3	Reflektioner.....	24
6	Referenser	26
	Bilaga 1 – Kravlista	28
	Bilaga 2 - Elasticsearch mapping av bildobjekt.....	29
	Bilaga 3 – Webapplikation dokumentation	30

Bilaga 4 – Node-modul dokumentation.....	32
---	-----------

Tabellförteckning

Tabell 1. Kunskapsinventering.....	9
Tabell 2. Etikett-utvärderingsresultat	19

Figurförteckning

Figur 1. Arkitektur.....	14
Figur 2. Skärmbild, sökexempel	15
Figur 3. React-komponenter	16

1 Uppdrag

Följande kapitel redogör för uppdragets bakgrund samt de mål, avgränsningar, och förväntningar som förelåg vid projektets inledning. Vidare redovisas behov av utökade kunskaper som krävdes för genomförandet under rubriken kunskapsinventering. Avslutningsvis redovisas delförfattarnas bidrag samt en disposition.

1.1 Bakgrund

Riksantikvarieämbetet (RAÄ) är en statlig myndighet som lyder under Kulturdepartementet, vars uppgift är att bevara, använda och utveckla kulturarvet så de nationella kulturmiljömålen uppnås. Myndigheten har huvudsakligen verksamhet i Stockholm, Visby och Tumba. RAÄ har också som uppgift att samordna svenska museers digitala samlingar i systemet K-samsök som tillgängliggörs via samsöktjänsten Kringla.

Uppdraget var kopplat till verksamheten digital förmedling som ingår i informationsavdelningen. Enligt avdelningens webbplats (Riksantikvarieämbetet 2018) har de som uppgift att; "Göra information om vår kulturmiljö användbar och tillgänglig för alla. Avdelningen ansvarar för utveckling och förvaltning av digitala informationssystem samt Riksantikvarieämbetets externa och interna kommunikation". Kontaktperson på RAÄ har under projektiden varit David Haskiya, enhetschef för digital förmedling.

RAÄ ansvarar bland annat för att tillgängliggöra miljontals bilder innehållande historiska föremål, konst med mera. Dessa bilder har i dagsläget metadata som inte följer samma standard och terminologi. Detta gör att nyckelordsbaserad bildsök blir utmanande. En tjänst för att söka bilder baserat på nyckelord tillgängliggörs av RAÄ (Kringla 2018). En liknande tjänst skulle i samverkan med ett ytterligare lager av extraherad bildinformation kunna förbättra sökning.

1.2 Projektmål

Projektets övergripande mål var att utreda om tillgängliga tjänster för extraktion av bildattribut kan användas för att förbättra tillgängliggörandet av det fotografiska kulturarvet. Som en del av utredningen utvecklades en prototyp som möjliggör sökning av bilder baserat på

bildinnehåll. Därtill levereras källkod och dokumentation till uppdragsgivaren. RAÄ bidrar med tillgång till bildmaterial via K-samsök, arbetsutrymmen på utvecklingsavdelningen, teknisk kompetens samt assistans i en kvalitativ urvalsprocess av lämpliga bildunderlag.

Projektets leverabler var en prototyp konstruerad i React med tillhörande databasimplementation i Elasticsearch. Prototypen kördes inledningsvis lokalt i utvecklingsmiljön och hämtade bilder från de källor som K-samsök levererade till databasen.

1.3 Avgränsningar

Prototypen avgränsades till att öka informationskvalité genom söktjänst med tillgång till extraherade data från en extern tjänst. Datamängden har begränsats till cirka tusen bilder som slumpas för att uppnå ett heterogent urval. Datasetet avgränsas till K-samsöks presentationsformat vilket innebär att all metadata för bilderna inte inkluderas i prototypens databas. Detta för att presentationsformatet är ett mer lätthanterligt format givet tidsramen för projektet. Därtill kan all bildrelaterade metadata nås via inkluderade länkar till Kringla.

Uppdraget har avgränsats från större implementation i RAÄ:s system, driftsättning av prototyp samt utveckling av gränssnitt som skiljer sig från befintliga system såsom Kringla. Prototypen avgränsades till att endast vara anpassad för en fönsterbredd över 1200 pixlar. Därtill har ingen anpassning till skärmläsare eller andra tillgänglighetsåtgärder beaktats i prototypens konstruktion.

1.4 Förväntningar

Förväntningarna var att projektet skulle leda till en prototyp som kunde uppvisa en förstärkt sökupplevelse för en delmängd av samlingarna.

Uppdragsgivaren bidrog med teknisk kunskap och tillhandahöll det material som låg till grund för utveckling av prototypen. I gengäld förväntade sig uppdragsgivaren källkod, undersökningar och dokumentation som återkoppling. Inga förväntningar fanns att projektet skulle vara driftklart vid uppdragets avslut.

Förväntningarna var att vi efter genomförandet uppmärksammas på komplexiteten i de olika delarna av ett programvaruprojekt samt samband mellan krav och egenskaper hos produkten. Därtill en utökad förståelse för inkrementell och iterativ prototyputveckling.

1.5 Kunskapsinventering

Under projektets första veckor genomfördes en kunskapsinventering för att identifiera kunskapsbehov. Tabellen nedan redogör för ingångsläget samt fördjupningsbehov.

	Teknisk kunskap	Tematisk kunskap	Procedurell kunskap
Ingångsläget	React, JavaScript, HTML, CSS, Dokumentdatabaser	Uppdragsgivarens har kunskap om tillgängliga bildunderlag.	Övergripande kunskap om systemutvecklingsmetoder, exempelvis Scrum och Continuous integration
Fördjupningsbehov	Elasticsearch, K-samsök API Google Vision API	Fördjupning i tillgängligt bildunderlag	Rapid prototyping

Tabell 1. Kunskapsinventering

1.6 Delförfattarnas bidrag

Författarna har till lika delar bidragit till alla rapportens delar. Gällande prototypens konstruktion har arbetet fördelats i små inkrementella moment där begränsad funktionalitet har implementerats individuellt. Parprogrammering har tillämpats för vissa delar av projektet

1.7 Disposition

Återstoden av rapporten är strukturerad enligt följande. Kapitel 2 redogör för högnivåproblemet och relaterad forskning samt dess implikationer för projektet. Kapitel 3 redogör för resultatet av uppdraget, vald teknik och prototyp. Resultatet utvärderas under kapitel 4 med utgångspunkt från högnivåproblemet och relaterad forskning i kapitel 2. Avslutningsvis redovisas slutsatser och rekommendationer för hur ett fortsatt arbete inom området kan genomföras under kapitel 5. Rapportens bilagor innehåller dokumentation för prototypens olika delar samt en kravlista. De är; 1 - Elasticsearch mappings, 2 – Kravlista, 3 – webapplikation dokumentation och 4 - Node-modul dokumentation.

2 Kunskapsbas

Eftersom projektets utgångspunkt är att sökningar i de bildsamlingar som RAÄ tillgängliggör kan stärkas genom att tillföra bildinformation, har högnivåproblemet konstaterats vara varierande informationskvalitet. För att kunna konstruera en prototyp som kunde erbjuda en starkt sökfunktion genom starkt informationskvalitet har två liknande system som har tillgång till bildsamlingarna studerats. En av tjänsterna har tillgång till metadata som erbjuds via K-Samsök och den andra en kombination av densamma samt den information som har tillförts prototypen.

2.1 Definition av högnivåproblem

Varierande metadata kopplat till de samlingar som RAÄ tillgängliggör medför att det kan vara svårt att navigera innehållet. Följaktligen kan problemet definieras som varierande informationskvalitet. Delone och McLean (2003) redogör för tre olika kvaliteter som påverkar nettonyttan av ett informationssystem, de är informationskvalitet, systemkvalitet och servicekvalitet. Sammantaget påverkar dessa användartillfredsställelsen och därmed användarens avsikt att använda systemet. Eftersom projektets huvudmål är att leverera en prototyp är en ökning av servicekvaliteten utesluten. Systemkvalitet påverkas av de verktyg som används för att presentera informationen för användaren. Prototypens gränssnitt och informationslagring sker genom erkänt effektiva verktyg såsom Elasticsearch och Reactivesearch. Således skiljer sig inte prototypen i många avseenden från redan tillgängliga lösningar som exempelvis Kringla. Prototypen kan dock inte förväntas uppnå systemkvalitet som motsvarar Kringla då det är ett driftsatt system.

En ökning av informationskvalitet har enligt Delone och McLean (2003) visat sig ha en signifikant påverkan på den upplevda systemkvaliteten. Informationskvalitet kan enligt författarna delas in i följande underkategorier.

- Fullständighet
- Lättförståelig
- Situationsanpassad / Personanpassad
- Relevans
- Säkerhet

RAÄ:s befintliga system levererar sökresultat utifrån tillgängliga metadata i ett lättförståeligt format anpassat till de sökresultat som användaren anger. Ett bidrag till hur komplett (fullständighet) informationen är kan öka nyttan genom att presentera ett större omfång av samband mellan sökta objekt för användaren. En förutsättning för detta är dock att en extern tjänst för att extrahera bildinnehåll kan leverera relevant semantik gällande bildernas innehåll.

2.2 Liknande system

RAÄ tillhandahåller söktjänsten Kringla som i interaktion med API: et K-samsök tillgängliggör bildmaterial. Prototypen liknar i sin utformning och funktionalitet Kringla. Den avgörande skillnaden är att prototypen tillhandahåller utökade möjligheter för sökningar genom att integrera information som har extraherats ur bilderna via extern tjänst. Många av de fält som i Kringla kan modifiera sökningar via fasettering eller nyckelordssökning erbjuds inte heller via prototypen.

Google erbjuder via sin söktjänst möjligheter till att avgränsa sökningar till en sida, som till exempelvis Kringla. Därtill går det att söka på bildattribut och ett litet urval av färger. Prototypen utökar denna funktionalitet genom att redovisa relaterade bildattribut där resultatet kan reduceras genom att välja kombinationer av ord inom olika intervall av säkerhet. Därtill erbjuder prototypen möjlighet att söka på kombinationer av färger i motsats till Googles bildsök där endast en färg kan väljas.

2.3 Relaterad forskning / litteratur

Sökning har genomförts efter relaterad forskning inom ämnet bildkategorisering av kulturarv, artiklar gällande effektivisering av bildsök samt forskning inom informationskvalitet.

Artikeln A hybrid ontology and visual-based retrieval model for cultural heritage multimedia collections (Vrochidis et al. 2008) introducerar en hybridmodell för bildsökning där sökningar via metadata förstärks med bildinnehåll för att hitta liknande bilder. Genom en transparent presentation av hur de två sökmetoderna samverkar presenteras rekommendationer för användaren.

The DeLone and McLean model of information systems success: a ten-year update (Delone och McLean 2003), redogör för tre olika kvalitetskategorier som påverkar netto nyttan av ett informationssystem.

Continuous Prototyping: Unified Application Delivery from Early Design to Code (Alperowitz et al. 2017) Tar upp konceptet "Continuous Prototyping" som en metod för att alltid ha en prototyp klar för utvärdering. Enligt Alperowitz underlättar detta designval och kommunikation mellan utvecklare och uppdragsgivare.

Continuous integration (Fowler och Matthew 2006), redogör för arbetssättet "continuous integration" som påtalar vikten av att utveckling sker i små steg för att tidigt identifiera fel och brister i programvara.

2.4 Implikationer för uppdraget

Vrochidis et al. (2008) redogör för ett angreppssätt där sökningen är transparent för användaren, samt hur kombinationen av manuellt annoterade metadata och bildinnehåll kan stärka sökningar i bildsamlingar. För prototypen innebär detta att vi har valt att göra sökningar så transparenta och tydliga som möjligt. Främst genom att presentera relaterade bildattribut samt färger och tillgängliggöra reglage för avgränsningar.

Delone och McLean (2003) visar på att förbättrad informationskvalitet kan stärka användarupplevelsen och den upplevda nyttan. Tillförd maskin-genererad information förväntas förbättra sökningar genom ökad fullständighet. Frågan är om den tillförda informationen är semantiskt relevant samt om sökningar där attribut kombineras är lättförståeliga. Artikeln bidrar med en begreppsmodell kring informationskvalitet som har använts för att utvärdera lösningen.

Continuous prototyping (Alperowitz et al. 2017) förespråkar kontinuerlig uppdatering och tillhandahållande av alltid fungerande prototyp. Detta har använts i projektet för säkra framtida krav tillsammans med uppdragsgivaren då en fungerande prototyp alltid funnits på testserver för utvärdering.

Continuous integration (Fowler et al. 2006) har använts under utveckling av prototypen för att förenkla arbetet med implementation och felhantering. Många små implementationer har medfört att fel tidigare kunnat upptäckas och åtgärdas.

3 Programvara

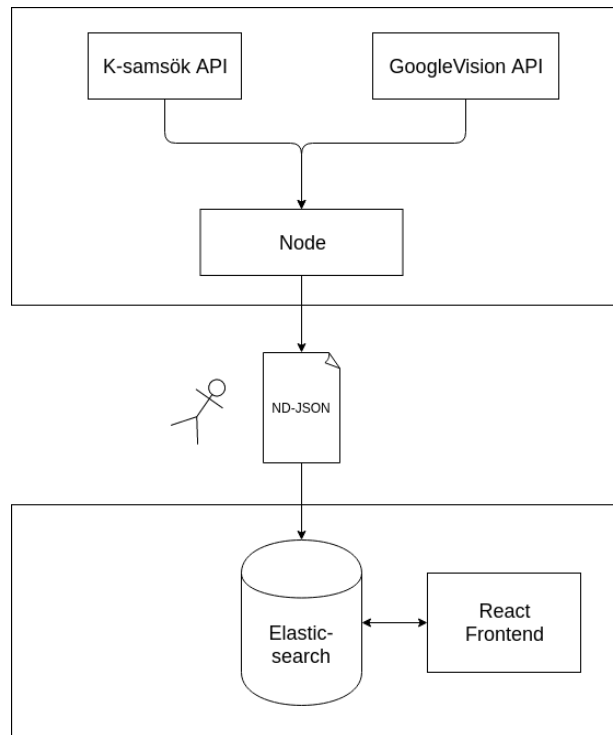
Under arbetet med prototypen har olika iterativa utvecklingsmodeller används, bland annat Continuous Integration (Fowler, et al. 2006) och Continuous Prototyping (Alperowitz, et al. 2017). Dessa har bidragit till kontinuerlig förbättring och motverkat brister i design av applikationen. Det inledande arbetet bestod i att definiera vad som kan anses vara en prototyp samt att ta fram en initial kravlista som stöd för det fortsatta arbetet. Under utvecklingsprocessen har nya designbeslut kommit till och vissa förändringar har gjorts. Kravlistan (bilaga 1) som presenteras i rapporten är den slutgiltiga kravlistan.

En stor del av projektiden har använts för att bearbeta den slutgiltiga datastruktur som krävts för prototypen samt att utveckla ett användargränssnitt som underlag för utvärdering. Eftersom författarna inte var bekanta med delar av de teknologier som används så har utvecklingen tillika delar varit ett utforskande av möjligheter och begränsningar som implementation av funktionalitet. Avsnittets underrubriker redogör för den konstruktion och teknikval som var aktuella när utvecklingsarbetet avslutades. En kombination av iterativ utveckling och utvärdering har uteslutit teknikval till förmån för andra under arbetets gång. De teknikval som under hela projektiden varit konstanta är Elasticsearch och React för webbapplikationen samt Node.js för sammanställning av bilddata från API:er. Det som varierat har främst varit olika React-bibliotek och komponenter. Inledningsvis användes biblioteket Searchkit (Searchkit 2018) för att interagera med Elasticsearch. Brister i dokumentationen medförde dock svårigheter att anpassa och utöka de komponenter som Searchkit erbjöd. Därför påbörjades en ny version av prototypen där ReactiveSearch utvärderades som ersättare. Den nya prototypen hade inom loppet av en dag mer funktionalitet implementerad, varvid det beslutades om att byta React-bibliotek. Dokumentation av prototypens delar finns i bilaga 3 – webbapplikation dokumentation och bilaga 4 – Node-modul dokumentation.

3.1 Övergripande beskrivning

Prototypen består av två separata delar (Figur 1). En av dem kombinerar information från K-samsök och Google Vision som sedan skriver dem till en fil i formatet ND-JSON (New Line Delimited JSON). Den kombinerade informationen används för att fylla Elasticsearch som nyttjas av

prototypens presentationsdel. Processen genomförs manuellt för varje nytt dataset som tillförs till presentationsdelen. Datainsamlingsmodulen är skriven i JavaScript och körs i Node.

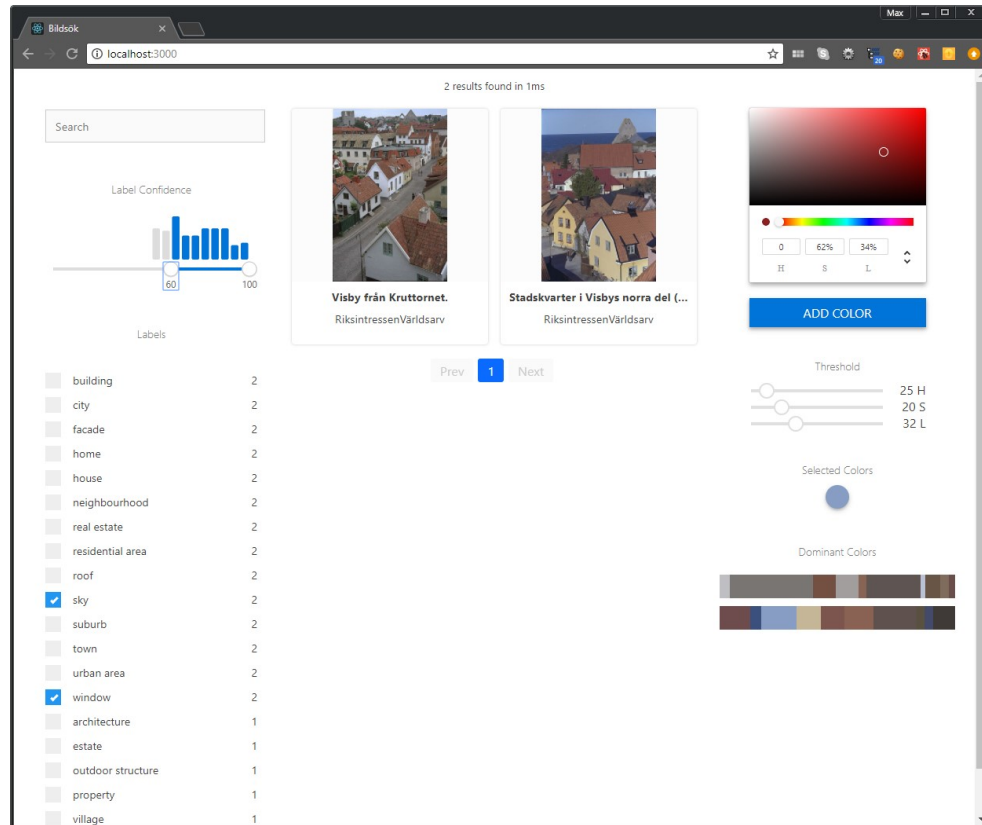


Figur 1. Arkitektur

Presentationsdelen består av en webbapplikation skriven i React med följande beroenden.

- Reactivesearch, 2.5.0
- Histoslides, 2.0.14
- React-color (ChromePicker), 2.14.1
- React-modal (Modal), 3.4.4

Reactivsearch bidrar med huvuddelen av funktionaliteten i applikationen genom att agera mellanhand för Elasticsearch och egenskrivna React-komponenter. Användaren har möjlighet att filtrera sökresultat baserat på kombinationer av bildinnehåll och färger. I figur 1 har användaren valt att avgränsa sin sökning till etiketterna "fönster" och "himmel" samt avgränsat till en specifik blå färg. Därtill har sökningen avgränsats till att endast visa etiketter i säkerhetsintervallet 60–100 för hur hög säkerhetsgrad de har tilldelats av Google Vision, samt toleranser i färgavgränsningen.



Figur 2. Skärmbild, sökexempel

3.2 Programbibliotek och teknik

3.2.1 K-samsök API

K-samsök (K-samsök 2018) tillhandahåller digitalt material från svenska museer och andra minnesinstitutioner. Dessa görs tillgängliga via ett öppet API. API:et har använts i projektet för att hämta och sammanföra data med Google Visions tjänster.

3.2.2 Google Vision API

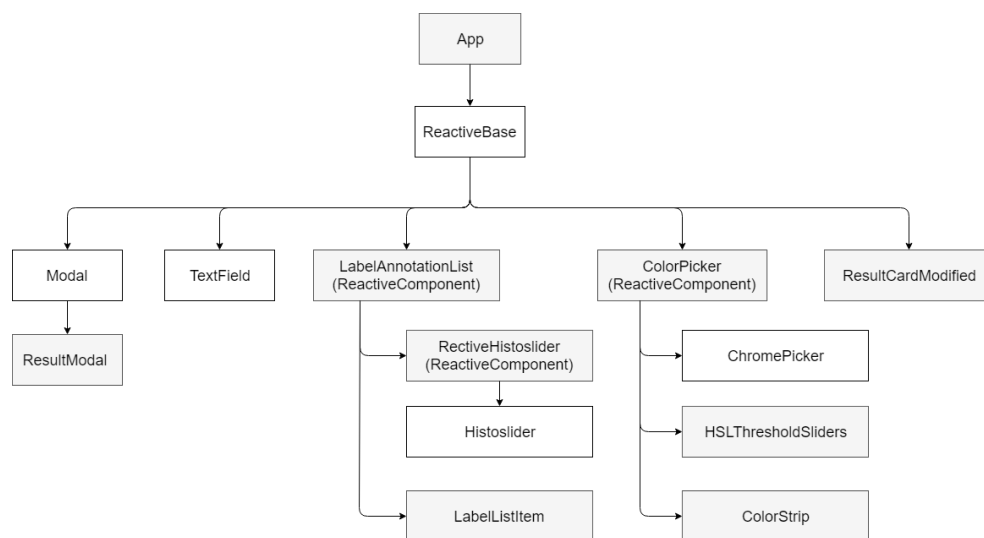
Google Vision (Google 2018) är ett API som tillåter utvecklare att extrahera bildinformation ur bilder, bland annat etiketter, dominanta färger, platsinformation och beskärningsförslag. Projektet använder endast funktionerna för etiketter och färger.

3.2.3 Elasticsearch

Elasticsearch (Elasticsearch 2018) är en dokumentdatabas avsedd för sökning som är skalbar via sharding, vilket betyder att databasens innehåll kan spridas på flera servrar för att på ett effektivare vis tillhandahålla data till många användare. Eftersom prototypen endast tillhandahåller sökningar i ett avgränsat innehåll samt ett fåtal användare, tillämpas inte sharding i implementationen. Elasticsearch plattar till dokumentinnehåll när data indexeras vilket innebär att relationssamband kan förloras vid sökningar. För att motverka detta har fälten för bildattribut och färger tilldelats en särskild mappning för nästlade strukturer (Bilaga Elastic Mapping). Följaktligen måste sökningar på de berörda fälten anpassas i presentationslagret.

3.2.4 React / Reactivesearch

React (reactjs.org 2018) är ett kraftfullt bibliotek för webbgränssnitt som tillsammans med Reactivesearch (appbase.io 2018) erbjuder en bra grund för att konstruera sökgränssnitt som interagerar med Elasticsearch. Figur 3 redogör för de komponenter som används. Egenutvecklade komponenter är gråmarkerade. Resterande är externa beroenden där användningen är begränsad till props alternativt databasfrågor.



Figur 3. React-komponenter

Många av komponenterna är beroende av varandra i det avseendet att de ändras beroende på val användaren gör i en annan komponent. Val av etiketter påverkar vilka färger som visas och vice versa. Möjligheten att utforska bildsamlingen är följaktligen inte begränsad till ett visuellt

resultatområde. Områdena består av tre kolumner (figur 2) från vänster till höger är de: LabelAnotationlist som visar en lista med etiketter samt ett reglage för att ställa vilket intervall av säkerhet som ska visas, ResultcardModified visar sökresultat i ett rutnät, ColorPicker visar valbara dominanta färger för ett givet sökresultat samt möjlighet att välja färger. Därtill kan tröskelvärden för sökningen av valda färger ställas.

3.2.5 Node-script

Scriptet hämtar och sammanfogar JSON från Google Vision och K-Samsök, samt översätter RGB-värden till HSL-format för att de ska vara sökbara via webbapplikationen. Objektet skrivs till fil som ND-JSON för att sedan infogas i databasen via CURL.

4 Utvärdering

Projektet har utvärderats med följande valideringstyper, demonstrativ utvärdering där olika medarbetare i organisationen har bidragit med åsikter, tolkande diskussion där tillförda bildattribut utvärderas samt ett metodologiskt avsnitt som redogör för utvecklingsprocessen. Avsnittet inleds med en redogörelse av urvalsmetoden för de bilder som prototypen innehåller.

4.1 Urvalsmetod

Utvärderingen av prototypen bygger på att innehållet är representativt för de bilder som RAÄ tillgängliggör. Därför har urvalet genomförts slumpmässigt. Därtill har urvalet begränsats till olika Creative Commons licenser (se nedanstående lista) samt objekt som innehöll bilder från medel till hög upplösning. Dessutom uteslöts bilder från Statens historiska museer då samlingarna innehåller objekt som felaktigt märkts som foto. Om ett foto av en spik från järnåldern kan kategoriseras som foto eller inte, är en tolkningsfråga. Vi har valt att i möjligaste mån utesluta den typen av bilder. Under ovanstående restriktioner finns det 1 128 269 bilder att tillgå via K-samsök, av dessa har 1047 bilder hämtats slumpmässigt för att uppnå ett heterogent urval.

Inkluderande Licenser:

- BY-NC, Attribution-NonCommercial
- BY-SA, Attribution-ShareAlike
- BY, Attribution
- PDMARK, Public Domain Mark (No Copyright)

4.2 Tolkande

Projektets utgångspunkt är att ett söksystem kan stärkas genom en höjning av informationskvalitet (Delone och McLean 2003), samt att tillgängliggörande av underliggande information i applikationens gränssnitt innebär en förbättrad sökupplevelse (Vrochidis et al. 2008). För att kunna stärka att så är fallet har vi genomfört nedslag på enskilda etiketter och jämfört informationen som har tillförts med befintliga metadata från K-samsök.

Vi har valt att fokusera på etiketten porträtt. Av prototypens innehåll på 1043 bilder klassificeras 91 stycken som porträtt i säkerhetsintervallet 0–100. De extraherade attributen överklassificerar porträtt. Enligt definitionen som anges av nationalencyklopedin (2018) är ett porträtt en;

”avbildning av en person om skulptur, målning, grafisk bild el. foto; vanl. av (nästan) enbart ansiktet el. med tonvikt på ansiktet”.

Prototypen innehåller porträtt-etiketterade foton där flera personer förekommer och där tonvikten inte ligger på ansiktet. Därför har vi anpassat säkerhetsintervallet till 80 - 100 vilket leder till ett resultat på 33 foton som vi bedömer falla inom nationalencyklopedins definition av porträtt. Av dessa 33 bilder fann vi endast att 7 stycken hade ”porträtt” någonstans i metadata från k-samsök. Ett komplement av maskin-extraherade etiketter från Google vision ökar följaktligen resultatet för sökningar på porträtt med 22 träffar. Det innebär en ökning av informationskvalité inom kategorin fullständighet. Därtill påvisas nyttan av att tillgängliggöra reglage för information såsom etikett-säkerhet. I vissa fall förekommer det falska positiva resultat med höga säkerhetspoäng, vilket innebär att användaren ställs inför en ”allt eller inget” situation. En avvägning mellan etikett-säkerhet och inkludering har genomförts för resterande etikett-nedslag som redovisas i tabell 2.

Etikett	Antal (SI* 0 -100)	Antal efter justerat SI*	SI*	Förekomst i K-Samsök	Falska positiva inom justerat SI*
Portrait	91	33	80-100	7	0
Dress	21	21	0-100	0	3
Sitting	54	54	0-100	3	5
Road	44	29	65-100	4	1
Sculpture	31	31	0-100	7	1
Summa	241	168		21	10

Tabell 2. Etikett-utvärderingsresultat

*Säkerhetsintervall

Resultatet av nedslaget är dock inte enbart positivt. Av 91 porträttbilder i säkerhetsintervallet 0 - 100 fanns endast 21 resultat med etiketten ”human”. Andra relaterade etiketter förekom i samlingen om 91 bilder som ”person”, ”gentleman”, ”woman”, ”man”. Här anser vi att etiketten ”human” borde förekommit i kombination. Resultatet strider mot vad användaren intuitivt förväntar sig och innebär en sänkning av informationskvalité inom kategorin lättförståelighet.

Färger annoteras sparsamt i k-Samsöks data, och i de fall där färg förekommer är det manuellt annoterat av en person som först betraktat färgen och skrivit en generell beskrivning i text. För sökningar tillför den typen av annotation begränsad nytta. Google Visions tillförda data medför att bilder kan sökas genom att ange en eller flera färger. Därtill redovisas sökresultatens dominanta färger. Det innebär en stärkning av informationskvalité i avseende på färger. Sökförmågan skulle kunna förbättras genom ytterligare funktionalitet. I nuvarande utformning medger prototypen endast förmåga att ställa tröskelvärden för hela sökningen. Tröskelvärden för vardera av de sökta färgerna skulle kunna göra sökningar mer specifika i avseende på sammansättning.

4.3 Demonstrativ

Prototypens funktionalitet demonstreras och åsikter inhämtas angående lösningens kvalitéer. En första demonstration genomfördes för ansvarig på enheten för digital förmedling David Haskiya samt verksamhetsutvecklare Albin Larsson. Här framkom potentiella förbättringar och eventuell verksamhetsnytta som prototypens funktionalitet skulle kunna tillföra.

De framförde att prototypen visar på hur liknande lösningar kan leverera verksamhetsnytta. Dels som ett verktyg för att analysera och förbättra den manuella annoteringen av samlingarna, och som ett informationslager vid presentation för en allmänhet. Det påpekades dock att den tillförda informationen bör märkas som maskin-genererad för användaren, med motiveringen att användaren ska förstå varför eventuella fel och märklig etikettering kan förekomma.

Under mötet uppmärksammades också att cirka 80% av de slumpvis utvalda bilderna i prototypens databas är svartvita. Ett antagande om att det återspeglar fördelningen mellan färgfoton och svartvita foton i samlingarna är följaktligen rimligt. Att kunna identifiera svartvita foton och vice versa är med befintliga metadata inte möjligt. Varvid prototypen påvisar att det kan genomföras med tillförda maskin-genererade etiketter. I prototypens nuvarande utformning är det dock endast möjligt att visa svartvita bilder. Det går inte att baserat på etiketter välja bort bilder för att till exempelvis filtrera bort svartvita motiv. Avsaknaden av denna typ av funktionalitet är ett område för förbättring.

Demonstration genomfördes även för anställda på utvecklingsavdelningen. Under denna framkom liknande synpunkter som tidigare framförts i mötet med uppdragsgivaren. Om projektet riktas mot en

allmänhet ska det tydligt framgå att data som tillförts är maskin-genererad och att fel kan förekomma. Prototypen mottogs positivt av avdelningen som kunde relatera till nyttan som extraherade bildattribut medför.

4.4 Metodologisk

Tidigt i projektet ställdes frågan: hur definieras den tänkta prototypen? En prototyp kan variera mellan en icke funktionell uppvisning av tilltänkt applikation till en prototyp med fullt fungerande implementation som är begränsad till viss funktionalitet. Definitionen av prototypen ansågs viktig då denna är ett stöd för att utvecklare och uppdragsgivare tidigt ska kunna identifiera framtida behov i applikationen (Alperowitz et al. 2017). Valet gjordes att prototypen skulle innehålla begränsad men färdig implementation av funktionalitet utifrån en första kravlista (se: bilaga 1 - Kravlista).

Projektet har till viss del följt metoden Continuous integration (Fowler et al. 2006) bortsett från automatiserade tester av kod. Dock har projektet konstant byggts och manuellt testats innan bidrag gjorts till kodbasen i Git. Efter implementation har prototypen byggts och lagts på testserver där applikationen funnits tillgänglig för alla intresserade. All data som används, såsom data för databasen och instruktioner för manuell körning av byggprocess har sparats tillsammans med prototypen i kodbasen. Projektet har i så stor utsträckning som möjligt gjorts automatiserad vad gäller builds och uppladdning till server, men på grund av projektets storlek har inte mycket tid lagts vid detta.

Continuous integration (Fowler et al. 2006) innefattar också testdriven utveckling (TDD), valet gjordes dock att frånga denna metod då stora delar av området för prototypen var okänt för utvecklarna. Att använda TDD skulle medföra ytterligare komplexitet till ett projekt som var under utforskning samt av begränsad omfattning.

Under projekttiden har metoden rapid prototyping tillämpats då en fungerande version av prototypen byggts efter lyckad implementation och gjorts tillgänglig via testserver. Detta har medfört att utvecklare och uppdragsgivare kunnat visualisera och diskutera prototypens funktionalitet (Alperowitz et al. 2017).

Under utvecklingsperioden har varje implementation testats och muntligen utvärderats av författarna. I regel genomfördes dessa informellt under kortare tidsintervall på tio till femton minuter.

Detta har sedan legat till grund för fortsatt utveckling. Då projektet inte varit så omfattande har tillfällen uppstått då extreme programming (agilealliance.org 2018) tillämpats i form av parprogrammering. Detta för att underlätta arbetet vid mindre och koncentrerade implementationer.

Att arbeta enligt dessa metoder har medfört att fel kunnat identifieras i tidigt skede och därför tagit kortare tid att åtgärda. Manuella tester har gjorts vid varje mindre implementation och en fungerande prototyp har alltid funnits på testservern vilket underlättat utvärdering av kommande funktionalitet.

5 Diskussion

Med utgångspunkt från uppdraget och följande utvärdering har ett antal slutsatser och rekommendationer formulerats av författarna. En reflektion över projektet och utvecklingsprocessen avslutar därefter rapporten.

5.1 Slutsatser

Inledningsvis kan konstateras att etiketterna, vid de nedslag vi genomfört tillför information som saknas i befintliga metadata. Förekomsten av falska negativ (saknad etikettering) visar dock att etiketterna försvårar fasettsökning genom att presentera inkonsekventa resultat för användaren. Varvid lämpligheten vid en implementation av den typen riktad mot slutanvändare kan ifrågasättas. Andra tjänster är möjligen mer konsekventa i detta avseende. Varför fler tjänster behöver utredas, alternativt en annorlunda presentation av etiketter.

Gällande sökning på färger kan konstateras att implementationen låter användaren utforska samlingarna på ett nytt vis. Funktionalitet rimmar väl med uppdragsgivarens verksamhetsmål. Kombinationen av fasetterad sökning och färgurval utgör ett kraftfullt verktyg och bidrar till nya möjligheter för att genomföra urval.

Prototypen visar på de möjligheter som maskin-extraherad information kan bidra med till verksamheten. Fler tjänster och genomförandealternativ behöver utredas för att utröna hur verksamheten kan nyttja denna typ av tjänst.

5.2 Rekommendationer

De fält som använts i projektet är sammansatt av k-samsöks presentationsformat, detta innebär att det saknas fält som bör vara med i en slutlig prototyp för att stärka precisionen i sökresultaten. Därför rekommenderas att ett komplett dataset från K-samsök integreras i framtida prototyper. En programmatisk jämförelse kan då genomföras för att statistiskt fastställa etiketternas bidrag till sökbarheten. En förutsättning är då att etiketterna översätts till svenska.

Facetterad sökning av etiketter kan medföra att oönskad reducering sker i sökresultatet. Vår rekommendation är att detta bör utvärderas ytterligare för att hitta en sökfunktion som kan presentera resultatet på ett alternativt vis. Den demonstrativa utvärderingen visar dessutom vikten av att tydliggöra etiketternas ursprung för användaren, för att skapa förståelse för förekomsten av eventuella fel. Detta skulle innebära ökad nytta för verksamheten och slutanvändaren. Färgsök bör också inkluderas i en färdig prototyp för att bidra till RAÄ:s verksamhetsmål, som är att tillgängliggöra kulturarvet.

Den sammantagna utvärderingen av projektet pekar på en förbättring av sökupplevelsen för användaren, om existerande bilddata kompletteras med ytterligare information från externa tjänster som Google Vision. Med förbehåll för att endast enstaka nedslag har genomförts för att kontrollera etiketternas precision. Våra rekommendationer till uppdragsgivaren är därför att ytterligare utvärdera vad detta kan bidra till i verksamheten utifrån det underlag som presenterats i rapporten och de svagheter som nämnts i slutsatsen.

5.3 Reflektioner

Projekttiden har ur många aspekter varit lärorik. Allt från att planera utvecklingsarbete till att säkra krav som ställts på prototypen, har bidragit till ökad förståelse för såväl programmering som utvecklingsprocesser. Projektet har utformats enligt en iterativ process vilket bidragit till att designbeslut tidigt har kunnat omvärderas.

Att förutse komplexitet i byggprocessen samt tidsestimering har varit en utmaning. Sammanslagningen av data från två separata API:er tog längre tid än beräknat. Det initiala datasetet som användes för att utveckla webbapplikationen var homogent i sin struktur, vilket inte var fallet när det slutgiltiga slumpade datasetet sammanställdes. Det fanns initialt förhoppningar om att fler tjänster för bildinnehålls-extraktion skulle kunna utvärderas under projekttiden. Detta har inte varit möjligt då mycket av tiden ägnades åt att bekanta oss med Elasticsearch och Reactivesearch-biblioteket. Om projektet skulle göras om igen med den utökade kunskap som vi har förvärvat under projekttiden skulle mer tid ha lagts på förarbeten.

Ämnet är intressant och det genomförda arbetet är långt ifrån att ge fullständiga svar på alla områden där denna typ av teknik skulle kunna tillföra verksamhetsnytta. Projektet bidrar till att identifiera områden för ytterligare fördjupning och utvärdering där skarpare avgränsningar

skulle kunna tillämpas. Till exempelvis, hur relaterade etiketter presenteras för att bäst kunna navigeras av användare mot bakgrund av att kombinationen av dem inte alltid förbättrar resultatet? Är det lämpligt att presentera reglage för säkerhetsintervall i en implementation för slutanvändare, med tanke på att det förekommer feletiketterade bilder vid höga säkerhetsvärden?

6 Referenser

Alperowitz, Lukas, et al. "Continuous Prototyping: Unified Application Delivery from Early Design to Code." *CSE@ SE*. 2017.

Delone, William H., and Ephraim R. McLean. "The DeLone and McLean model of information systems success: a ten-year update." *Journal of management information systems* 19.4 (2003): 9-30.

Elasticsearch. elastic.co <https://www.elastic.co/products/elasticsearch> (Hämtat: 2018-05-14).

Extreme programming. agilealliance.org <https://www.agilealliance.org/> (Hämtat: 2018-05-11).

Fowler, Martin, and Matthew Foemmel. "Continuous integration." *Thought-Works*) <http://www.thoughtworks.com/Continuous Integration.pdf> 122 (2006): 14.

Google. cloud.google.com <https://cloud.google.com/vision/docs/> (Hämtat: 2018-05-14).

Informationsavdelningen. Riksantikvarieämbetet.se. <https://www.raa.se/kontakta-oss/informationsavdelningen/#digital> (Hämtat: 2018-03-28).

Kringla. <http://www.kringla.nu/kringla/> (Hämtat: 2018-03-28)

K-samsök. Riksantikvarieämbetet. ksamsok.se <http://www.ksamsok.se/> (Hämtat: 2018-05-14).

Lew, Michael S., et al. "Content-based multimedia information retrieval: State of the art and challenges." *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 2.1(2006):1-19.

Reactjs. React.org. <https://reactjs.org/> (Hämtad: 2018-05-09).

Reactive Manual, appbase.io <https://opensource.appbase.io/reactive-manual/> (Hämtat: 2018-05-09).

Searchkit. Searchkit.co <http://www.searchkit.co/> (Hämtat: 2018).

Vrochidis, Stefanos, et al. "A hybrid ontology and visual-based retrieval model for cultural heritage multimedia collections." *International Journal of Metadata, Semantics and Ontologies* 3.3 (2008): 167-182.

Bilaga 1 – Kravlista

- Användaren ska kunna söka bland k-samsöks beskrivningar i bildbiblioteket
- Visning av bildattribut baseras på Google Visions extrahering av attribut.
- Genom att välja bland bildattribut ska användaren kunna reducera sökresultatet till kombinationer av attribut.
- Visningen av dominanta färger i bildsamlingen baseras på Google Visions extrahering av färger.
- Användaren ska kunna söka bland bildinnehållet baserat på bildernas färginnehåll.
- Alla avgränsningarna i sökningen kombineras till ett resultat.
- Användaren ska kunna klicka på bilderna för att få utökad information.
- Användaren ska kunna välja hur många bildattribut som visas.
- Användaren ska kunna begränsa sökresultatet till bildattribut inom ett relevansintervall.
- Användaren ska kunna sätta tröskelvärde för färgavgränsningar för varje kanal (hue, saturation, lightness)
- Användaren ska via färgväljare kunna lägga till egna färger.

Bilaga 2 - Elasticsearch mapping av bildobjekt

(Visare endast mapping av de två nestade objekten i databasen)

```
{
  "mappings": {
    "googleVision": {
      "dynamic_templates": [
        {
          "colors_field_as_nested": {
            "match": "colors",
            "mapping": {
              "type": "nested"
            }
          }
        }
      ],
      {
        "labelAnnotations_field_as_nested": {
          "match": "labelAnnotations",
          "mapping": {
            "type": "nested"
          }
        }
      }
    ]
  }
}
```

Bilaga 3 – Webapplikation dokumentation

Content based image search student project

This is a prototype commissioned by Riksantikvarieämbetet (RAÄ) in Visby. The prototype's purpose is to show the potential strengths or weaknesses of adding machine extracted attributes to images made available through RAÄ's API. The Google Vision service was used in conjunction with the RAÄ K-Samsök API to merge image information of 1000 photos. The result is presented via a React web application that allows a user to search images by combining labels as well as combinations of colors.

Getting Started

Prerequisites

- Nodejs needs to be installed on your system. Refer to Nodejs documentation for your Operating system at: <https://nodejs.org/en/>
- Elasticsearch is needed as database and searchengine. Check documentation at: <https://www.elastic.co/>

Installing

- Install Nodejs
- Install Elasticsearch
- Clone project to local folder

To install required dependencies run:

```
npm install
```

Deployment

Add data to Elasticsearch database.

First, add mapping included in './ElasticsearchConfigs/elasticsearchMappings.txt'-file:

```
curl -XPUT 'localhost:9200/test_data?pretty' -H 'Content-Type: application/json' -d <Content of elasticsearchMappings>
```

Then add test data from './ElasticsearchConfigs/test_data.json'-file:

```
curl -XPOST 'localhost:9200/test_data/googleVision/_bulk?pretty' -  
-data-binary "@/path-to-test-data-file/test_data.json" -H  
'Content-Type: application/json'
```

The default Elasticsearch-server is set to 'localhost:9200' and indice 'test_data'. If a change is needed, edit props for ReactiveBase-component in './src/App.js'

Run project

run:

```
npm start
```

Built With

- [React.JS](#) - The web framework used
- [ReactiveSearch](#) - React.js library for searchfunctions
- [Histoslidder](#) - A D3 based histogram slider component for React.JS
- [react-color](#) - A Collection of Color Pickers for React.JS
- [react-modal](#) - Accessible modal dialog component for React.JS

Authors

- [Alfred Bjersander](#)
- [Max Collin](#)

License

This project is licensed under the MIT License - see the [LICENSE.md](#) file for details

Bilaga 4 – Node-modul dokumentation

Google Vision and K-Samsök Merge

Merges data from Google Vision and K-Samsök. The result is stored in a ND-JSON file suitable for Elasticsearch bulk uploads.

Getting Started

Prerequisites

- Google Vision API-key need to be submitted in index.js
- Nodejs needs to be installed on your system. Check documentation for your Operating System at <https://nodejs.org/en/>

Installing

- Install Nodejs

To install required dependencies run:

```
npm install
```

Deployment

Run project

```
npm start
```

Built With

- [Node.JS](#) - The framework used.
- [RGBtoHSL](#) - Node.JS library for converting RGB to HSL
- [Node-Fetch](#) - A light-weight module that brings window.fetch to Node.js

Authors

- [Alfred Bjersander](#) [Max Collin](#)

License

This project is licensed under the MIT License - see the [LICENSE.md](#) file for details