

H6 Svende Forløb 2023 - gruppe 2

# E-Scrap Solutions

## Procesrapport

---



Skrevet af Casper Andersen og Kevin Frost

---



DataTekniker med  
speciale i programmering  
<https://www.zbc.dk/>  
ZBC Ringsted

**Title:** E-Scrap Solutions

**Tema:** Elektronikaffald

**Rapport type:** Procesrapport

**Projekt start:** 11. apr. 2023

**Antal gruppemedlemmer:** 2

**Vurdering.** Fælles bedømmelse

**Medlemmer:**

Casper Andersen

Kevin Frost

**Github Profiler:**

Casper Andersen (@[Casp7654](#), @[Ndersorn](#))

Kevin Frost (@[Keviner123](#))

**Sider:** 38

**Projekt afslutning:** 8. maj 2023

# 1 Definitioner, akronymer og forkortelser

**Sorteringssystem:** De samlede enheder der står for at differentiere og sortere elektroniske komponenter fysisk.

**Registreringssystem:** De samlede applikationer der står for at kunne registrere og håndtere elektroniske komponenter logisk.

**YOLO/YOLO V8:** Kunstig intelligens-bibliotek der bruger en forud trænet model til objektgenkendelse.

**Control Center:** Fysisk og logisk enhed der er mellem punkt / kontrol enhed for sorteringsssystemet.

**Rest Api:** Server applikations interface til datahåndtering for elektroniske komponenter

**Web interface:** Web applikations til visuel og statistisk repræsentation af elektroniske komponenter

**ORM / TypeORM:** Object relations manager bibliotek der bruges til at forbinde Rest Api og Database.

**MariaDB:** Database applikation

**CET:** Central European Time Zone

**GPIO:** General purpose Input Output

**Machine Learning:** Fordring til en model med relevant data, så den bliver i stand til at komme med forudsigtelser og "tænke selv".

## **2 Indholdsfortegnelse**

<b>1 Definitioner, akronymer og forkortelser</b>	<b>3</b>
<b>2 Indholdsfortegnelse</b>	<b>4</b>
<b>3 Læsevejledning</b>	<b>6</b>
<b>4 Indledning</b>	<b>7</b>
<b>5 Problemformulering</b>	<b>8</b>
<b>6 Afgrænsning</b>	<b>8</b>
<b>7 Projektplanlægning</b>	<b>9</b>
7.1 Ugeplan	9
7.2 Tidsplan	10
<b>8 Forundersøgelse</b>	<b>11</b>
8.1 Målgruppe	11
8.2 Rige billede	12
8.3 Elektroniske komponenter	13
8.3.1 Modstande	14
8.3.1.1 Hvordan er de opbygget?	14
8.3.1.2 Kan den blive genbrugt	14
8.3.1.3 Kan den blive genanvendt	14
8.3.2 Kondensator	15
8.3.2.1 Kan den blive genanvendt	15
8.3.3 Transistorer	16
8.3.3.1 Hvordan er de opbygget?	16
8.3.3.2 Kan den blive genanvendt	16
8.3.3.3 Kan den blive genbrugt	16
8.4 Genanvendelse af komponenter	17
8.5 Risikoanalyse	18
8.6 Projekt strategier	18
<b>9 Foranalyse</b>	<b>19</b>
9.1 Objektgenkendelse	20
9.1.1 Huskylens	20
9.1.2 YOLO	21
9.1.2.1 Træning af egen model	23
9.1.2.2 Finjustering af datasæt	24
10 Versionering	25
11 Tekniske løsninger	25
11.1 Registreringssystemet	25
11.1.1 Database	25

11.1.2 API	26
11.1.2.1 ExpressJs	26
11.1.2.2 TypeORM	26
11.1.3 Web Interface	27
11.2 Sortering systemet	28
11.2.1 Kontrolcenter	28
11.2.2 Transportbånd	28
11.2.3 Opsamlingskasse	29
11.3 Salgs systemet	29
12 Sikkerhed	30
12.0.1 HTTPS	30
12.0.2 Token based auth	30
13 Teststrategi	31
13.1 Datasæt test	31
13.2 Unit test	32
13.3 User Acceptance Test	32
<b>14 Konklusion</b>	<b>33</b>
<b>15 Logbog</b>	<b>34</b>

### **3 Læsevejledning**

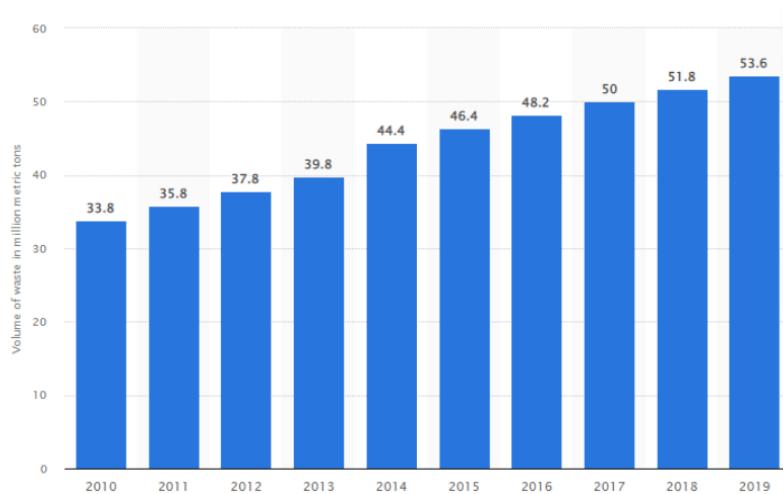
Denne rapport er udarbejdet til den studerende dатatekniker med speciale i programmering. Formålet med rapporten er at undersøge og analysere projektprocessen i dybden samt at vurdere det potentielle marked i forbindelse med projektets fuldførelse. Rapporten vil give en detaljeret beskrivelse af projektets forskellige faser og de tekniske aspekter af udviklingen af softwaren. Derudover omfatter rapporten også en markedsundersøgelse, der vil identificere potentielle målgrupper og konkurrenter, samt vurdere mulighederne for succes på markedet.

Den første del af rapporten er indledningen til projektet, hvori der bliver beskrevet problemstillingen, afgrænsningen og selve formålet med projektet. I anden del er der beskrevet vores undersøgelse af elektroniske komponenter og analyser af udbyttet ved genanvendelse af disse. Den tredje del af rapporten vil vise og forklare vores idéer i diagrammer og tanker til implementering af systemdesignet. I den sidste del af rapporten vil vi konkludere på hvorvidt vi har nået det mål vi gerne ville have ud af dette projekt, samt refleksioner over processen og eventuelle bud på forbedringer til et sådanne projekt

For den bedste indsigt i vores mål råder vi til at se vores præsentationsvideo som findes på [youtube](#), samt tage et kig på afsnittet omkring vores rige billede i section [8.2 Rige billede](#).

## 4 Indledning

I året 2019 blev der ifølge [statista.com](https://www.statista.com/statistics/499891/projection-ewaste-generation-worldwide/) - som kan ses på *Figur 1* - smidt hele 53.6 mio tons elektronikaffald ud. Dette tal stiger drastisk årligt, hvilket ifølge et estimat fra FN vil nå et højdepunkt på ca. 74. mio tons i 2030. Denne mængde af elektronikaffald, er et kæmpe problem på verdensplan, da det på nuværende tidspunkt kun er cirka 17% der bliver genanvendt.



*Figur 1<sup>1</sup> - Affaldsstatisistik*

Med denne estimerede stigning fra FN, samt tanken om at al dette elektronikaffald kommer fra værdifulde råstoffer, som hentes op af jorden, søger det danske miljøministeriet efter en løsning der vil få denne 17% genanvendelse til at stige.

Vores projekt vil fokusere på et produkt som vi formoder er en mulig løsning på den lave 17% genanvendelse af elektronikaffald, samt kigge ind i hvilket marked der er for genbrug af elektroniske komponenter der stadig har en god levetid.

Vores mål med denne rapport er at beskrive og dokumentere arbejdssprocessen, som vi har fulgt i udviklingen af metoder til genbrug af elektriske komponenter.

---

<sup>1</sup> <https://www.statista.com/statistics/499891/projection-ewaste-generation-worldwide/>

## **5 Problemformulering**

Vi vil undersøge, hvad man kan gøre der ville hjælpe ministeriet, virksomheder, samt privatpersoner og hobbyister, med at opdele elektronikaffald i enkelte elektroniske komponenttyper og måle mulig genanvendelsesværdi.

Vi forestiller os at det er muligt at teste/aflæse brugte elektriske komponenters levetid og slitage, samt danne en indsigt i hvilke muligheder der er for genbrug af disse i en direkte forstand.

Dertil vil vi undersøge, hvad værdi det har for hobbyister at tilbyde salg af brugte elektroniske komponenter, og om vi kan forhøje udnyttelsesgraden af brugte elektroniske komponenter ved gensalg af disse

Er der sammenhæng mellem hvad det ville koste at tilbyde en løsning der fokuserer på genbrug/gensalg af elektroniske komponenter og hvad det vil kræve ressourcemæssigt at tilbyde dette.

Ydermere vil vi undersøge muligheden for at måle/kortlægge hvilke råstoffer i elektronikaffaldet der ville kunne udtrækkes/nedsmeltes til genanvendelse i nye produkter, samt hvilke komponenter der ikke ville kunne genbruges direkte i samme fag, men kunne få nyt liv i andet fag - som f.eks. i kunstverdenen hvor man kan lave smykker ud af modstande eller bruge sammenlimede komponenter som døråbninger mm..

## **6 Afgrænsning**

Før vi går i gang med at beskrive processen, vil vi gerne beskrive hvilke områder af det færdige produkt vi har afgrænset os fra i dette projekt.

Vi har valgt at afgrænse os fra alt salg af de elektroniske komponenter og dertil værende webshops, da vi mener at vores fokus og tid bedst bliver brugt på at kunne registrere og vurdere komponenternes genanvendelsesgrad. Derudover afgrænser vi os også fra at færdigudvikle brugerhåndtering samt login i vores systemer, af samme årsag.

# **7 Projektplanlægning**

Da vi er et 2-mands hold der skal få mest ud af den tid vi har til rådighed for dette projekt, har vi valgt at arbejde ud efter en meget simpel projektplan, med et fladt projektgruppe hierarki. Med det mener vi, at projektgruppens struktur er således at alle arbejder sammen på lige niveau. Dette betyder, at der ikke er klare hierarkiske niveauer eller forskelle i magt og autoritet mellem medlemmerne, og at vi har mere lige stillinger og roller i gruppen. I stedet for at have en enkelt leder, kan et fladt hierarki have flere teamledere eller koordinatorer, der samarbejder og kommunikerer tæt med hinanden og med de andre medlemmer af gruppen. Dette hierarki kan bidrage til en mere fleksibel og samarbejdsorienteret tilgang til projektledelse, hvor alle medlemmer af gruppen har en stemme og et bidrag til projektet og dets succes.

Projektets længde er 4 uger og vi har delt vores arbejdssage op i 4 et-uges forløb, hvor vi hver uge vil skifte vores fokusområde, for at kunne opnå mest muligt inden for givet tid og samtidig nå i mål med mest muligt. Vi vil udover den beskrevet projektplan holde korte daglige statusmøder, for at sikre at vi ikke overskider planen.

## **7.1 Ugeplan**

Første uge vil vi bruge til at undersøge elektroniske komponenter og deres genanvendelsesværdi. Ydermere analyserer og afprøve teknologier og biblioteker vi ville kunne bruge til udvikling af delsystemerne - mere om dem senere.

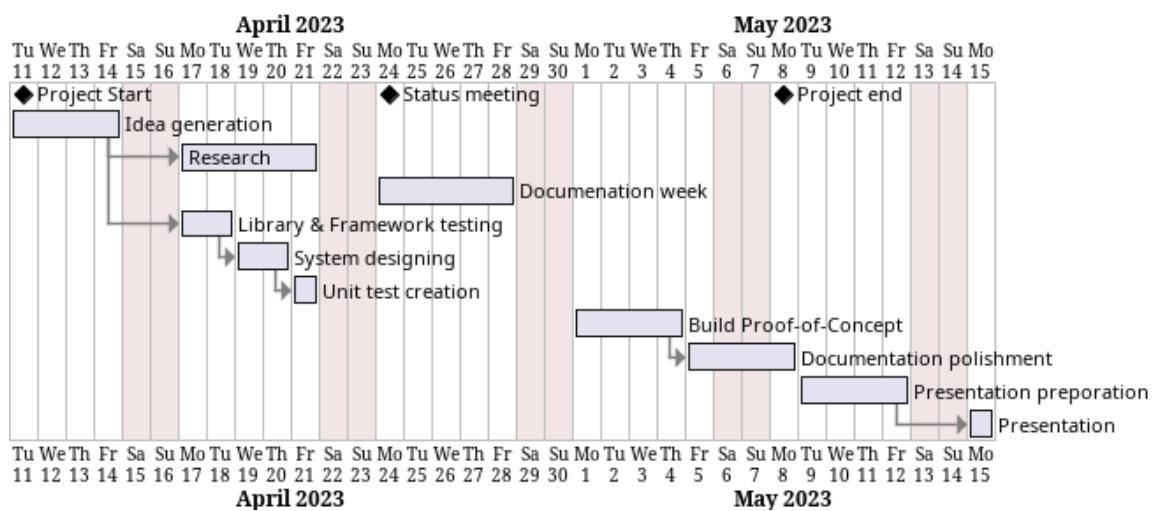
Anden uge vil vi fokusere på at få afprøvet vores idéer samt lave et komplet system design som vi kan bruge som værende vores proof-of-concept.

I tredje uge vil vi bruge på størstedelen af vores dokumentation, da vi mener dette er et godt tidspunkt at holde et statusmøde på det overordnede projekt. Her vil vi fokusere på størstedelen af vores dokumentation af produktet, samt få de sidste dele af det komplette system og designet til matche med det resultat vi på dette tidspunkt er nået.

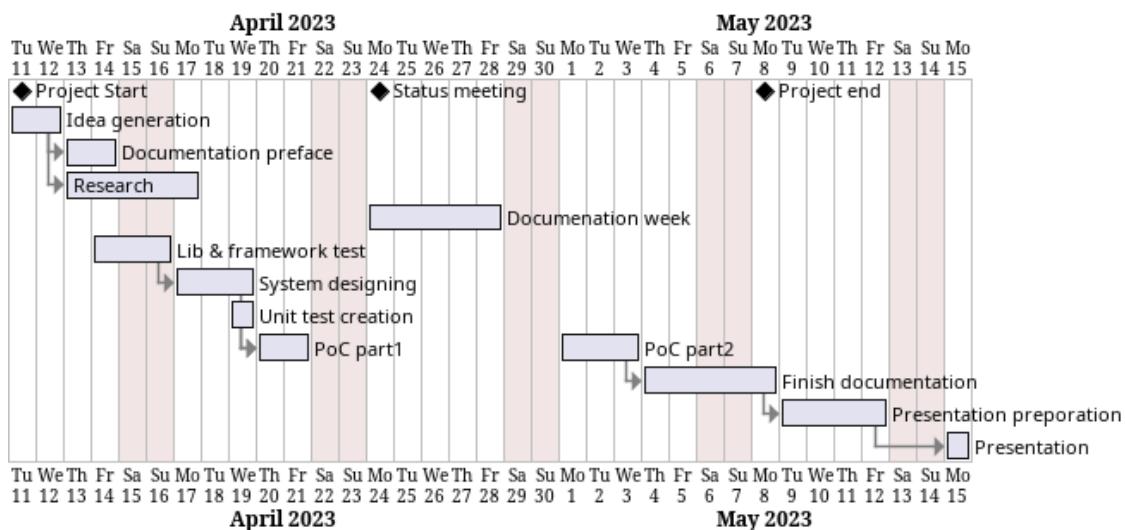
I fjerde uge vil fokusområdet være at få finpudset vores proof-of-concept, få færdiggjort dokumentationen og få konkluderet på, hvorvidt vi er nået i mål.

## 7.2 Tidsplan

Som det kan ses på vores Gantt diagrammer, formåede vi at overholde tidsplanen til en vis grænse. Den største forskel er på opdelingen af vores proof-of-concept, som vi brugte mere tid på end hvad forventet var, samt at vi har brugt mere tid på system designet og flyttet vores research tættere på startdatoen, da det gav mening at få noget for hånden efter en kort men konstruktiv ide generering af produktet.



Figur 2 - Gantt Diagram: Estimeret tidsplan



Figur 3 - Gantt Diagram: Realiseret tidsplan

## **8 Forundersøgelse**

Som start af vores research, startede vi ud med at undersøge den mest hensigtsmæssige tilgang til projektet samt undersøgte krav og ønsker fra potentielle målgrupper. I dette afsnit vil vi også beskrive de udfordringer, som kan opstå i sådanne projekter og deres eventuelle betydning for projektet.

Her er en række spørgsmål, som vi vil undersøgt:

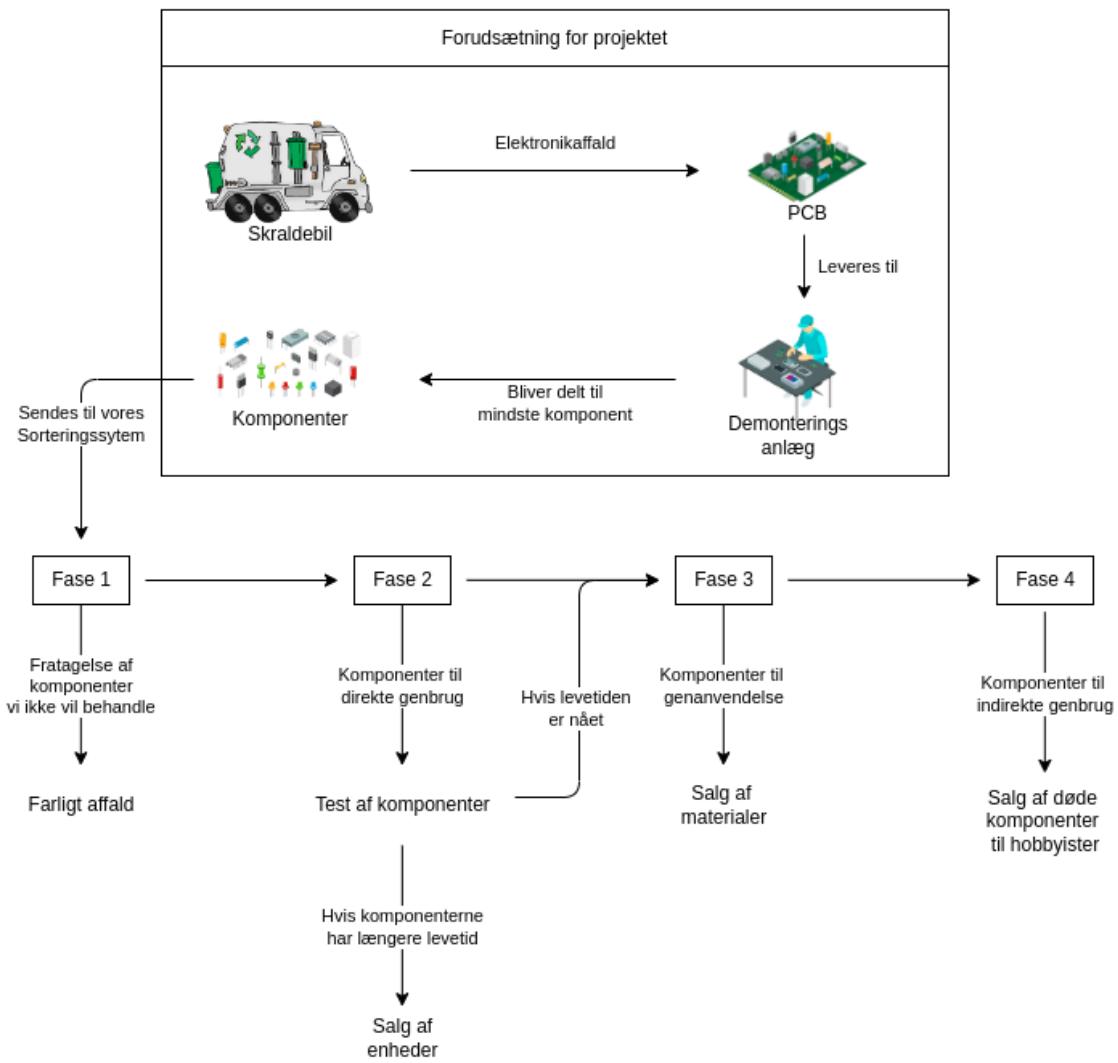
- Hvordan er disse komponenter opbygget, og hvad ville vi kunne udnytte?
- Hvilke komponenter kan vi genbruge direkte?
- Hvilke komponenter kan vi genanvende?
- Hvilken værdi har det at genanvende komponenternes råmaterialer?
- Hvilke komponenter står til at kunne "få nyt liv" ved gensalg/genbrug i andet fag ?

### **8.1 Målgruppe**

Projektets målgruppe er delt op i flere delmålgrupper, da vi både tilbyder at produktet kan anvendes til salg af komponenter til privatpersoner og virksomheder, samtidig tilbyder vi en løsning til virksomheder der ønsker at tage ansvar for miljøet og reducere deres affaldsproduktion.

## 8.2 Rige billede

Dette rige billede er en visual representation af vores samlede system, som er udarbejdet for at opnå en fælles forståelse for produktet gennem projektet.



Figur 4 - Rigt billede

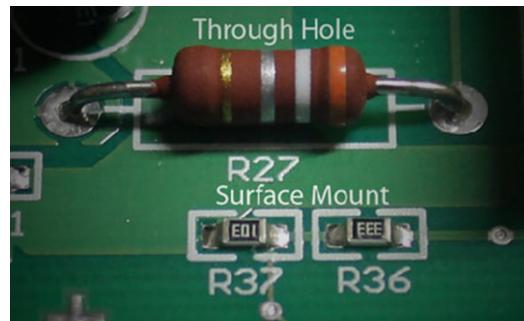
Som det kan ses på Figur 4 , så har vi valgt at lave et 3 delt system for at isolerer/illustrerer de forskellige processer vi mener der skal til for at få mest ud af disse indleverede komponenter. Disse dele kalder vi for faser og vil blive beskrevet mere detaljeret senere i vores rapport. Vi har også en forudsætning for at vi får disse elektroniske komponenter fra et demonterings anlæg, som også er beskrevet senere.

## 8.3 Elektroniske komponenter

I vores løsning vil der blive modtaget en række forskellige elektriske komponenter, som vil inkludere både SMD-komponenter (Surface Mount Device) og THT-komponenter (Through-Hole Technology). Disse komponenter skal sorteres for at kunne behandles korrekt.

SMD-komponenter (Surface Mount Device) er mindre og fladere end through hole-komponenter, og de monteres direkte på overfladen af printkortet i stedet for at blive monteret gennem huller i printkortet. Dette gør det muligt at producere mindre og mere kompakte elektroniske produkter.

Through hole-komponenter er større og mere robuste end SMD-komponenter, og de monteres ved at indsætte benene i huller i printkortet og derefter lodde dem fast. Dette gør det nemmere at udskifte og reparere komponenter, men også mere besværligt og tidskrævende at montere dem på printkortet.



Figur 5<sup>2</sup> - Forskellen på SMD og Through hole-komponenter

Generelt kan man sige, at SMD-komponenter er mere miljøvenlige end THT-komponenter, da de kræver mindre plads på printkortet og derfor reducerer mængden af materiale, der anvendes til produktionen af printkortet. SMD-komponenter kræver også mindre lodning og mindre energi til produktion og transport, da de er mindre og lettere.

---

<sup>2</sup>

<https://www.mokotechnology.com/through-hole-pcb-surface-mount-pcb-which-one-do-we-recommend/>

### **8.3.1 Modstande**

Modstande anvendes i en bred vifte af elektroniske kredsløb med det formål at regulere strøm, spænding eller andre dele i kredsløbets ydeevne. For eksempel kan en modstand anvendes til at begrænse strømmen, der løber gennem en LED, så LED'en ikke bliver beskadiget under drift.

#### 8.3.1.1 Hvordan er de opbygget?

En udbredt og økonomisk form for modstand er metalfilm modstande. Disse modstande er sammensat af et metal (såsom en nikkel legering) eller metaloxid, der er viklet rundt omkring et keramisk fyldstof og forbundet til forbindelsesledninger, som ofte er loddet på modstandens krop. Modstanden afhænger af tykkelsen af metalfilm belægningen, jo tykkere belægningen er, jo mindre modstanden.

#### 8.3.1.2 Kan den blive genbrugt

Forbindelsesledningerne, som er fremstillet af stål med kobberbelægning og tin, kan genanvende, mens modstanderens krop typisk kun kan bortskaffes

Metalfilm-modstande indeholder ruthenium, som kan opløses og derefter tørres og nedsmeltes til at opnå rent metal. Faktisk bliver op mod 90% af verdens forbrug af ruthenium<sup>3</sup> brugt til fremstilling af modstande. Desværre er genbrug af ruthenium fra modstande sjældent økonomisk rentabelt på grund af omkostningerne ved udvindingen.

#### 8.3.1.3 Kan den blive genanvendt

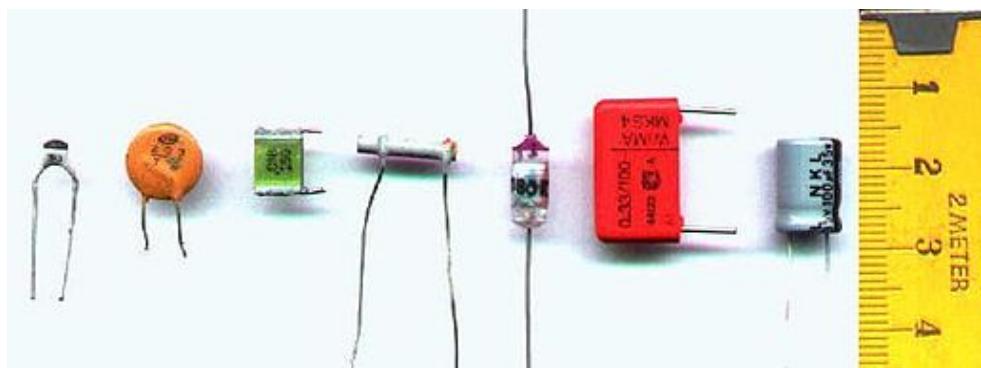
Modstande er sjældent fejlbehæftede og kan i teorien genanvendes mange gange uden at miste deres funktionalitet

---

<sup>3</sup> <https://passive-components.eu/resistor-markets-tighten-as-ruthenium-prices-skyrocket/>

### 8.3.2 Kondensator

En kondensator er en elektronisk komponent, der kan lagre elektrisk ladning. Den består af to ledende plader adskilt af isolerende materialer. Når en spændingsforskell påføres kondensatoren, opbygges ladningen på pladerne, og der opstår et elektrisk felt mellem pladerne. Jo større spændingsforskell og jo større overfladearealet på pladerne, desto mere ladning kan kondensatoren lagre. Kondensatorer anvendes i en række elektroniske kredsløb og systemer til at filtrere og opbevare elektrisk energi.



Figur 6<sup>4</sup> - Eksempel på forksellige kondensatorer

#### 8.3.2.1 Kan den blive genanvendt

Kondensatorer vil med tiden udvikle en højere indre modstand, hvilket medfører tab af deres evne til at lagre energi over tid. En kondensator har normalt en brugbar levetid på omkring 700 timer. Efter et stykke tid vil lækstrømmen mellem metalpladerne stige, og dette gør det muligt for strømmen at løbe gennem isolatoren m og forringe den tilsigtede drift af kondensatoren. Det er ikke godt, at små kondensatorer bliver bortskaffet som almindeligt affald. Kondensatorer indeholder ofte farlige stoffer som bly og kviksølv, der kan have alvorlige negative konsekvenser for miljøet og menneskers sundhed, hvis de ikke bortskaffes korrekt. Derudover kan kondensatorer indeholde materialer som aluminium og kobber, som er værdifulde ressourcer og kan genanvendes til fremstilling af nye produkter. Det er derfor vigtigt at bortskaffe kondensatorer på en miljøvenlig måde, f.eks. ved at sende dem til genanvendelse eller affaldsbehandling på en specialiseret facilitet.

<sup>4</sup> [https://da.wikipedia.org/wiki/Elektrisk\\_kondensator](https://da.wikipedia.org/wiki/Elektrisk_kondensator)

### **8.3.3 Transistorer**

En transistor er en elektronisk komponent, der kan fungere som en forstærker eller en switch i et elektronisk kredsløb.

#### 8.3.3.1 Hvordan er de opbygget?

Den består af tre lag af halvledermateriale, typisk dopet silicium. Disse lag er henholdsvis emitteren, basen og kollektoren. Når en strøm påføres basen, kan den kontrollere strømmen mellem emitteren og kollektoren, og dermed fungere som en forstærker eller en switch. Transistorer anvendes i en bred vifte af elektroniske kredsløb og systemer, herunder forstærkere, oscillatorer, signalbehandling og strømstyring.

#### 8.3.3.2 Kan den blive genanvendt

Det er muligt at genanvende en transistor, hvis den er blevet fjernet fra en kredsløbsplade og derefter testet i henhold til dens specifikationer i dens datasheet.

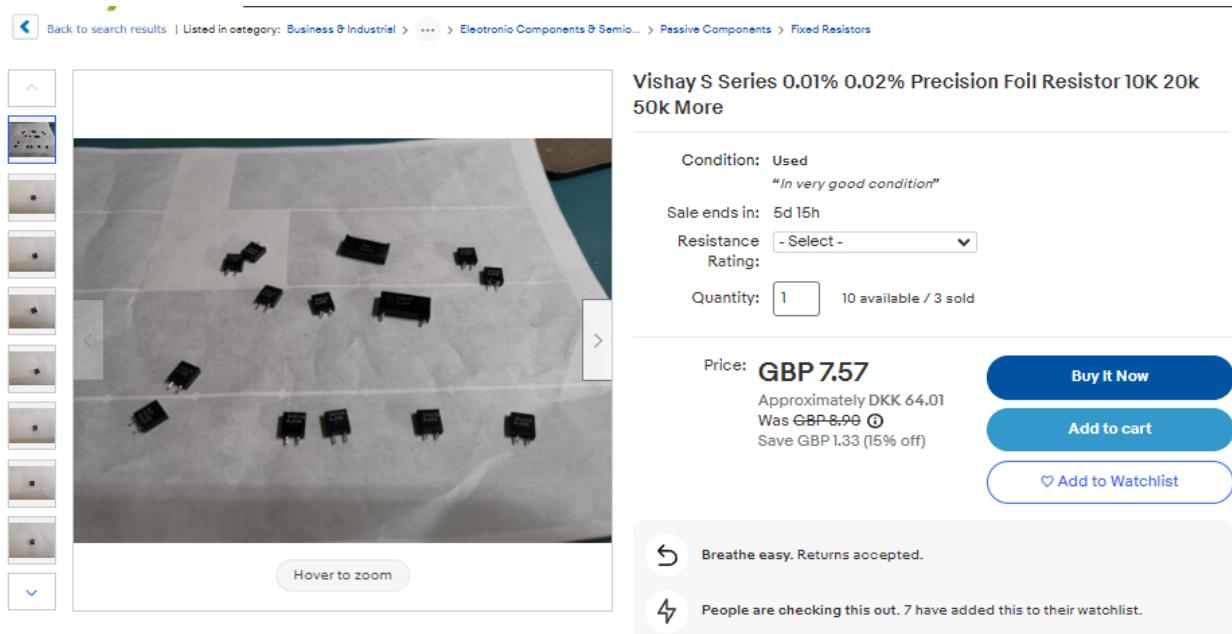
Ved at teste transistoren vil man kunne afgøre dens funktionalitet og om den stadig opfylder dens specificerede krav. Hvis den opfylder kravene, kan det genanvendes i andre kredsløb. Det er vigtigt at sikre, at transistoren er i god stand og fungerer korrekt, før den bruges igen, da en defekt transistor kan påvirke hele kredsløbet og medføre fejl eller skade på andre komponenter.

#### 8.3.3.3 Kan den blive genbrugt

De metalben eller ledninger, der sidder på separate transistorer, kan skilles ad og genanvendes som skrotmetal. Desværre bliver resten af en transistor normalt ikke genanvendt og kasseret som affald.

## 8.4 Genanvendelse af komponenter

For at få en idé om, hvordan vi bedst kunne vurdere genanvendelsesværdien samt værdien af genbruget af komponenter i vores projekt, undersøgte vi markedet for at finde ud af, hvilke virksomheder der i øjeblikket tilbyder brugte komponenter.



Figur 7 - Eksempel på komponenter fra eBay

Efter vores undersøgelse konkluderede vi, at eBay var den primære kilde til brugte komponenter. Der var en betydelig mængde testede elektronikkomponenter, der var blevet loddet af forskellige printplader. Fælles for alle disse komponenter var, at de havde en imponerende præcision på 2 decimaler, hvilket gjorde dem yderst attraktive at købe.

## **8.5 Risikoanalyse**

Vi har med vores undersøgelser dannet os et overblik over, hvad risici der er involveret i dette projekt.

Vi er som start, afhængige af at de elektroniske komponenter vi håndterer i vores system, kommer til os fra Vasac<sup>5</sup> firmaet som er beskrevet i vores case. Hvis dette - eller andet - samarbejde ikke er tilrådeligt, vil dette projekt ikke være aktuelt og bør derfor ikke fortsættes. Derudover har vi stor afhængighed af at projektet bliver finansieret tilstrækkeligt, da der ikke vil - efter vores egen vurdering - være positiv økonomi i dette projekt.

I vores udførelse af projektet har vi en stor risiko for at nogle af projektets dele vil blive udskudt, på grund af at bemandingen har været lavt fra start, hvilket giver store spring i effektivitetsprocenten. Med det mener vi at hvis et gruppemedlem bliver syg, falder effektivitetsniveauet med et minimum på 25% - da vi regner med at man ville kunne arbejde en lille smule under sygdom.

Vi har også indset fra starten af, at der vil være mange små systemer der ville skulle snakke sammen, på kryds og tværs af fysiske og logiske enheder. Dette giver en stor risiko for at det komplette system ikke vil være operationelt til alle tider, da selve produktet er afhængig af at alle delsystemer er oppe og kører.

## **8.6 Projekt strategier**

I projektet har vi prøvet at holde os til at arbejde ud efter et Docker udviklings- og produktionsmiljø, hvilket giver os mulighed for at kunne opsætte ny server / udarbejde systemer der vil virke cross-platforme, så længe Docker Engine er installeret og virker.

Derudover vil vi prøve at udvikle delsystemerne til at have deres egne "tilstand" (state), hvilket gør at selv hvis en enhed går ned, vil systemet blot stå i venteposition og vil kunne fortsætte ved udsættelse af fejl på nedbrudt enhed. - kort sagt, vil vi prøve at få systemerne til at holde deres state, indtil de får svar om at fortsætte.

---

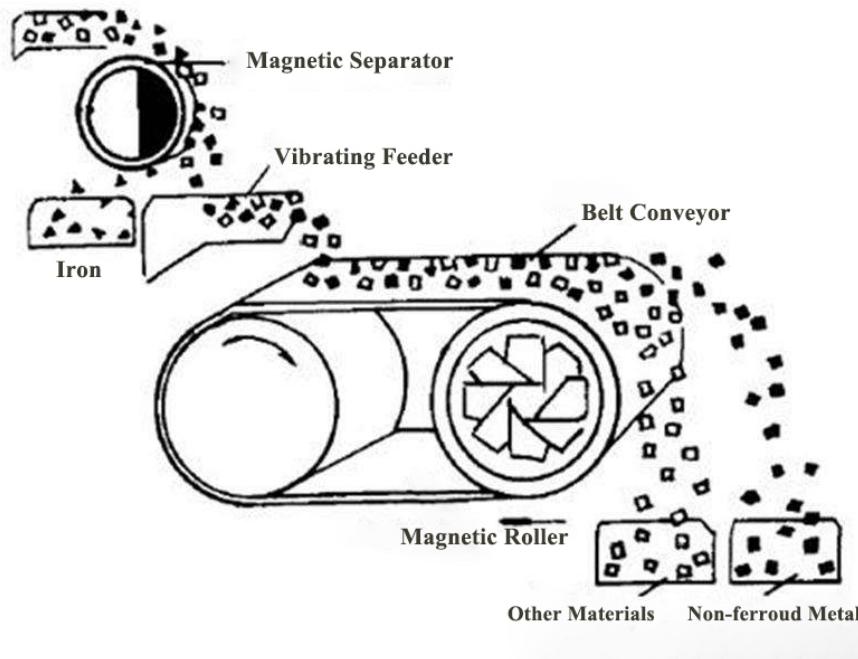
<sup>5</sup> <https://vasac.slagelse.dk/>

## 9 Foranalyse

Efter undersøgelsesfasen af vores projekt, har vi dannet os en idé om hvordan vi kan bygge et system der vil løse den lave 17% genanvendelsesprocent af elektronikaffald.

Normalt vil elektronik blive skåret i mindre stykker og ført ned på et transportbånd. Efterfølgende kan en stor elektromagnet, der er placeret over båndet, fange alle de magnetiske metaller.

Herefter kører du forbi en Eddy Current Separator, som er en type maskine, der bruger magnetiske felter til at adskille metaller fra ikke-metalliske materialer. Maskinen fungerer ved at generere et magnetfelt, som inducerer elektriske strøm i metallerne. Disse strømme skaber et andet magnetfelt, som modsætter sig det oprindelige felt, og dette fører til at metallerne bliver skubbet væk fra de ikke-metalliske materialer og adskilt fra dem. Eddy Current Separators bruges ofte til at adskille aluminium fra andre materialer, såsom plast eller glas, og de kan anvendes i en række industrier.



Figur 8<sup>6</sup> - Eddy Current Separator diagram

<sup>6</sup> [https://www.bangorecycling.com/product/eddy\\_current\\_separator.html](https://www.bangorecycling.com/product/eddy_current_separator.html)

## 9.1 Objektgenkendelse

Objektgenkendelse er en teknologi, der udnytter algoritmer og kunstig intelligens til at opdage og kategorisere forskellige objekter.

### 9.1.1 Huskylens



Figur 9<sup>7</sup> - Huskylens printplade

Efter at have undersøgt muligheden for at anvende en huskylens til vores projekt, måtte vi desværre fravælge den, da den ikke opfyldte vores krav. Vores ønske var at kunne implementere en pre trænet model, men desværre var det kun muligt at generere en model ved at tage billeder direkte på enheden. Derudover var det en udfordring, at vi ikke kunne vise hvad huskylensen så på vores Raspberry Pi, hvilket også bidrog til vores beslutning om at fravælge den.

---

<sup>7</sup> <https://github.com/dtex/huskylens>

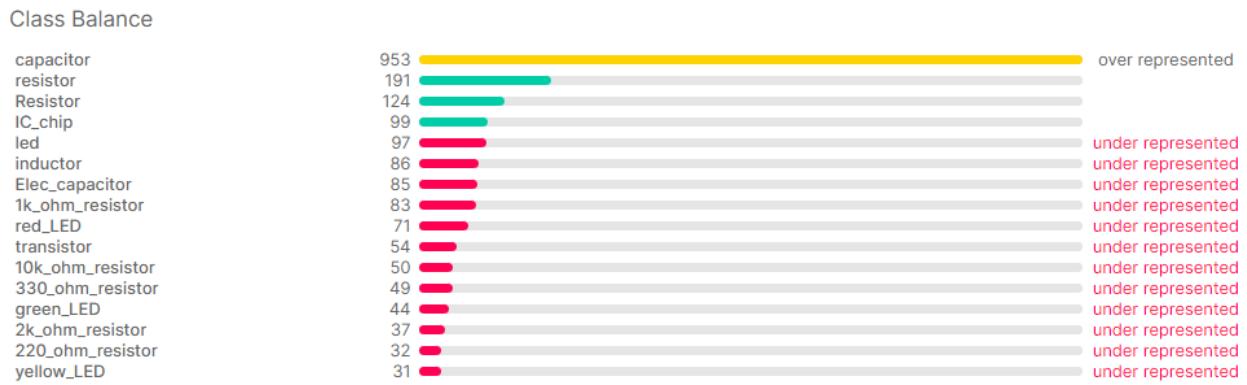
## 9.1.2 YOLO

Efter at have søgt på nettet for at undersøge de eksisterende muligheder for objektgenkendelse, stødte vi på YOLO algoritmen som den mest populære. YOLO, også kendt som You Only Look Once, er en sofistikeret algoritme til objektgenkendelse, der er udviklet af Ultralytics. Denne algoritme er optimeret til at kunne køre på både CPU og GPU, hvilket gør det muligt at anvende den i realtid på en bred vifte af enheder, herunder computere, mobiltelefoner og droner. Derudover er algoritmen blevet forbedret til at kunne fungere effektivt under vanskelige lysforhold, samt at kunne håndtere både støj og bevægelse i et givent billede. Den er hurtig til at genkende objekter i realtid, og anvendes bl.a. i overvågning, bilindustrien og medicinsk forskning.

Vi undersøgte om der var nogle der havde lavet eksisterende datasæts til genkendelse af elektroniske komponenter og fandt frem til følgende hjemmeside "[roboflow.com](https://roboflow.com)" hvor folk kan dele deres datasæts og pre trænede modeller, vi fandt så frem til følgende 3 datasæts, som vi ville forsøge at bruge i vores projekt.

Navn	Skaber	Antal bilereder
<a href="#">Electronic components recognition</a>	CPEN291G8	611
<a href="#">smdComponents</a>	Dainius	1950
<a href="#">Batteries</a>	cw	913

Efter at have prøvet de eksisterende datasæt, indså vi hurtigt, at de ikke var særligt godt balancede.



Figur 10 - Klasse balancen af Electronic components recognition datasættet

Dette kunne vi også validere, ved at teste den af på et live webcam feed, hvor nærmest alt blev markeret som at være capacitors (modstande)

Dette var ikke et stort problem i datasættet smdComponents, men da vi valgte at fravælge kategorisering af SMD-komponenter i vores proof of concept, besluttede vi i stedet at oprette vores eget datasæt.

### 9.1.2.1 Træning af egen model

For at træne dit YOLO-datasæt, skal du starte med at indsamle en stor mængde billeder af de objekter, du ønsker at genkende. Derefter skal du opdele disse billeder i tre grupper.

**Train:** Bruges til at træne YOLO-modellen ved hjælp af et datasæt med billeder og etiketter.

**Valid:** Bruges til at vurdere YOLO-modellens præstation under træning og justere modellens hyper parametre.

**Test:** Bruges til at evaluere YOLO-modellens ydeevne på nye og ukendte billeder efter træning og validering.

Dette skal gøres samme med best practice regelsættet skrevet af ultralytics.

- **Images per class.**  $\geq 1500$  images per class recommended
- **Instances per class.**  $\geq 10000$  instances (labeled objects) per class recommended
- **Image variety.** Must be representative of deployed environment. For real-world use cases we recommend images from different times of day, different seasons, different weather, different lighting, different angles, different sources (scraped online, collected locally, different cameras) etc.
- **Label consistency.** All instances of all classes in all images must be labelled. Partial labelling will not work.
- **Label accuracy.** Labels must closely enclose each object. No space should exist between an object and its bounding box. No objects should be missing a label.
- **Label verification.** View `train_batch*.jpg` on train start to verify your labels appear correct, i.e. see `example` mosaic.
- **Background images.** Background images are images with no objects that are added to a dataset to reduce False Positives (FP). We recommend about 0-10% background images to help reduce FPs (COCO has 1000 background images for reference, 1% of the total). No labels are required for background images.

Figur 11 - Klasse balancen af Electronic components recognition datasættet

Efterfølgende tog vi 10 billeder af hver af de følgende komponenter: en modstand, en kondensator og en induktor. Vi opdelte disse billeder i træning (80%), validering (10%) og test (10%).

Efter at have trænet vores model med datasættet og testet den, konkluderede vi, at modellen var effektiv til at detektere vores komponenter. Vi opdagede dog hurtigt, at modellen kun var trænet på billeder taget med det samme kamera og under samme lysforhold. Hvis billederne blev taget i et rum med et andet lysniveau, kunne modellen ikke med sikkerhed genkende komponenterne.

### 9.1.2.2 Finjustering af datasæt

Vi stod over for et større problem, da vi forsøgte at placere komponenterne på vores rullebane. Det viste sig, at modellen så en masse komponenter, som ikke var til stede i virkeligheden. Vi nåede derfor til den konklusion, at vores datasæt var for begrænset, og at det ville kræve en udvidelse for at opnå bedre resultater.

Vi havde derfor 2 muligheder for at forbedre vores model

1. Tag en masse billeder fra forskellige tidspunkter af dagen, forskellige årstider, forskelligt vejr, forskellig belysning, forskellige vinkler og forskellige kameraer.
2. Anvend billeder fra nettet til hurtigere at opbygge en model.

Af de muligheder, vi havde, virkede mulighed nummer 2 mest attraktiv, fordi den gav os en reel chance for at nå frem til et fungerende resultat indenfor den tidsramme, der var blevet tildelt vores projekt.

For at skaffe billeder hurtigt til vores datasæt, besluttede vi at benytte biblioteket bing-image-downloader til Python. Dette bibliotek kan tage en forespørgsel og downloade et valgfrit antal billeder. Dette gjorde processen nemmere, da det blot krævede en let oprydning af billederne til sidst.

Vi fik fat i et datasæt, som generelt havde en høj præcision, men da vores kamera var monteret parallelt med rullebåndet, og de fleste billeder fra nettet var produktbilleder, der var taget med en skrå vinkel, var vi nødt til at supplere med nogle ekstra billeder, som vi selv tog med vores webcam. Dette forbedrede modellens nøjagtighed betydeligt.

## 10 Versionering

Vi har valgt at køre versionsstyring ind over hele projektet, og dertil har vi valgt at bruge de platforme vi var kendte i forvejen.

Til source-kode har vi valgt at bruge Github, til dokumenter og eventuelle regneark, har vi valgt at bruge Google Docs og til hurtig deling af projektressourcer - så som billeder eller research .pdf filer - har vi valgt at køre kommunikationen over Discord.

## 11 Tekniske løsninger

Vi har prøvet at holde os til at køre både nutidige, stabile og afprøvede teknologier, da vi i vores projekt har mange del-systemer at skulle håndterer og derved også en masse kodenormer at skulle overholde.

Af det komplette produkt har vi taget udgangspunkt i at opdele systemet i 3 dele, et registreringssystem, et sorteringssystem og et salgssystem (webshops).

### 11.1 Registreringssystemet

Registreringssystemet er delt op i en logisk 3-lags model:

- En database til at holde vores dataobjekter
- Et REST API til at udfører vores domain logik
- Et WebInterface til oversigt over dataen

#### 11.1.1 Database

Vi har valgt at anvende MariaDB. MariaDB er et af de mest populære open-source relationelle databasesystemer i verden. Den kan nemt integreres med forskellige programmeringssprog og frameworks, hvilket gør det nemt at integrere i vores projekt.

## 11.1.2 API

Vi har valgt at skrive vores kildekode i Typescript da det er et superset til Javascript, der tilføjer typer som giver os bedre mulighed for at skrive objektorienteret programmering i Javascript. Samtidig vil vi skrive vores modeller i et delt bibliotek så vi ikke kommer ud for unødvendige komplikationer ved fremtidige ændringer. Vi har også valgt 2 populære og gennemtestede frameworks som grundsten til vores API.

### 11.1.2.1 ExpressJs

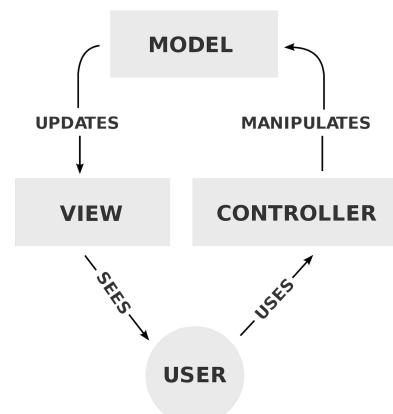
ExpressJs er et framework til håndtering af HTTP/HTTPS

Request til vores api, hvilket er valgt på grund af dets hændelsesbaseret arkitektur. ExpressJs udnytter Javascripts *on Event* (hændelser) funktionalitet på højeste plan, hvilket godt kan give en udfordring i 3-lags MVC strukturen vi gerne ville overholde.

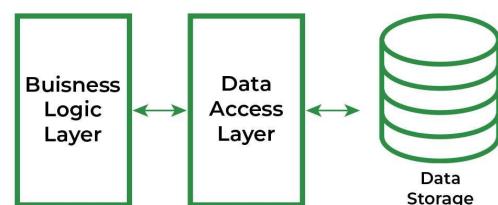
I vores produkt har vi fundet en løsning på denne udfordring, ved at udnytte MVC strukturen ved opstart af programmet og dermed gøre læsbarheden af source-koden lettere at navigere. Når en hændelse så bliver affyret - ved kald til Rest Api'et - vil programmet være i et anonymt stade, og dermed ikke have en *direkte state*.

### 11.1.2.2 TypeORM

TypeORM er et framework som vi har brugt til håndtering af vores Data Access Layer. TypeORM er et Entity Framework, der bruger Typescripts tilføjede funktionalitet ved navn dekorationer til at oprette tabeller i en database ved at bruge de data klasser i kildekode som er defineret til formålet.



Figur 12<sup>8</sup> - Software Design Model: MVC



Figur 13<sup>9</sup> - Data Access Layer

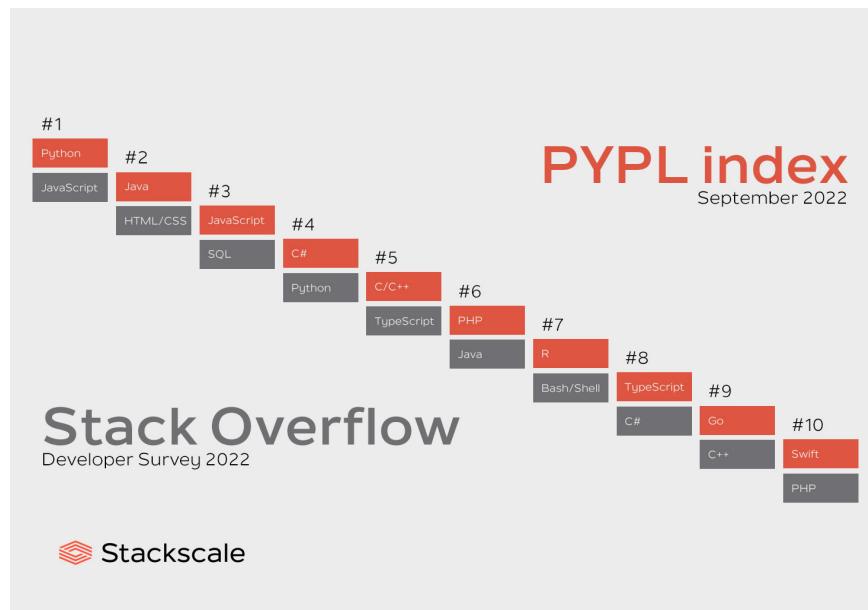
<sup>8</sup> <https://hxann.com/t3-architecture/>

<sup>9</sup> <https://www.geeksforgeeks.org/data-access-layer/>

Disse dekorationer i kildekoden beskriver hvilke felter, attributter samt andre parametre og samtidig står for at synkronisere ændringer automatisk. Dette framework er valgt på baggrund af at der er en rigtig god integration til MariaDB.

### 11.1.3 Web Interface

Vi har valgt at bygge vores web interface i Javascript, da det er det mest benyttede hjemmeside og frontend sprog der bruges på nuværende tidspunkt ifølge Stack Overflow, kan ses på *Figur 14* - dog top 3 i følge PYPL index, som også er en fokuseret hjemmeside til python.



Figur 14<sup>10</sup> - Stackscale overview of most popular web programming languages

Vi har valgt at opbygge applikationen ved hjælp af frameworket Astro, da det er et simpelt men kraftfuldt nyt komponent baseret framework der giver mulighed for at bygge komponenter i forskellige frontend frameworks - såsom ReactJs, NextJs og Svelte - og implementerer dem direkte. Dette vil sige, at vi kan bygge nogle komponenter i f.eks. Svelte og nogle i Common Js uden yderligere integrations komplikationer, da Astro frameworket vil stå for integrationen. Dette mener vi er en god fremtidssikring, da vi ikke binder os fast i ét specifikt ui framework.

<sup>10</sup> <https://www.stackscale.com/blog/most-popular-programming-languages/>

## 11.2 Sortering systemet

Efter vores research fandt vi ud af, at et automatisk sorteringssystem var nødvendigt for vores projekt. Derfor valgte vi at konstruere en automatiseret transportbånds løsning for at undgå den besværlige manuelle sortering af komponenter. Denne løsning vil være i stand til at håndtere forskellige typer af komponenter og sortere dem i forskellige kasser afhængigt af, om de skal testes, genanvendes eller kasseres.

### 11.2.1 Kontrolcenter

Til vores prototype har vi valgt at bruge en Raspberry Pi, fordi den passer perfekt til vores krav. Udover dens lave omkostninger, har Raspberry Pi en række indbyggede I/O porte, som gør det muligt at tilføje eksterne enheder og sensorer, der kan integreres i vores prototype. Raspberry Pi passer desuden også perfekt til vores krav.

- En CPU der er hurtig nok til at køre vores objektgenkendelse model
- Analoge pins til at kontrollere eksterne enheder (f eks. relæ)
- Analoge pins til at afprøve komponenter (f eks modstande og kondensatorer)

### 11.2.2 Transportbånd

Vi benytter et transportbånd til at transportere komponenter ned til vores kamera, så vi kan detektere deres type.

Til vores proof-of-concept fik vi udleveret et fischertechnik 24v transportbånd, som vi styrer via et 5v relæ til vores Raspberry Pi.

Vi havde dog foretrukket et bånd med en glat overflade, da vores meget små SMD-komponenter blev fanget i revnerne i det nuværende bånd. Desuden har vores kamera sværere ved at genkende komponenter med det nuværende bånd som baggrund.

### 11.2.3 Opsamlingskasse

Da vi undersøgte mulighederne for at sortere vores komponenter i 3 grupper, opdagede vi mange forskellige løsningsforslag. Vi konkluderede, at den bedste løsning for os var at montere en kasse med 3 rum på enden af vores sorterbånd, og koble den til et transportbånd. Disse 3 rum kan køres frem og tilbage, så de står lige ud foran sorteringsbåndet, og på den måde kan komponenterne lande præcist i de rette rum.

Vi besluttede os for at bruge en Ender 3 3D-printer til dette formål, og fjernede Z- og Y-aksen, så vi kun havde X-aksen tilbage. En fordel ved at anvende denne metode er, at en 3D-printer har et indbygget endestop på hver akse, som gør det muligt for os at "home" vores 3D-printer og præcist vide, hvor startpositionen er.

### 11.3 Salgs systemet

Vores mål er at kunne sælge genbrugbare komponenter gennem flere webshops, delt op i samme fase struktur som sorteringsanlægget. Tanken var at vores webshop skulle have været på niveau med de store komponent-sider som f.eks. Mouser, RS Components og Digi-Key.

48256 produkter fundet for "modstand"  
Rettet fra "modstand"

**Udvalgte kategorier**

Chassismonterede modstande (3844)  
Hulmonterede modstande (10771)  
Justerbare effektmodstande (72)  
Modstand - monteringsbeslag (17)  
Modstands-arrays (701)  
SMD modstande (31199)

Diskrete halvledere (429)

Bipolare transistorer (423)  
MOSFET (4)

Hulmonterede modstande (10771)  
Modstandssortimenter (99)  
SMD modstande (31199)  
Chassismonterede modstande (3844)  
Reostater - variable modstande (94)  
Justerbare effektmodstande (72)  
Modstands-arrays (701)  
Modstand - monteringsbeslag (17)

Sorter efter Relevans Sammenlign 0/8 Side 1 af 2413

Figur 15 - RS Components Webshop

## **12 Sikkerhed**

I forhold til sikkerhed mellem vores del-systemer har vi valgt at al kommunikation skulle foregå over HTTPS protocollen med JWT token-based authentication implementering.

### **12.0.1 HTTPS**

HTTPS Protokollen i sig selv ses ikke som et sikkerhedsdrag, men det gør kommunikationen mellem enheder mindre modtagelige for man-in-the-middle angreb. Derudover sikrer vi også vores kunder, at de er på det rigtige domæne ved at have et opdateret SSL-certifikat. Dog har vi i vores Proof-of-concept afgrænset os fra dette.

### **12.0.2 Token based auth**

Vi har valgt at køre med Token based authentication, ved at bruge den standardiserede implementering der hedder JWT. Dog har vi i vores Proof-of-concept afgrænset os fra dette også.

# 13 Teststrategi

Vi har valgt at implementere 3 forskellige testformater til at vi fremtidigt nemmere kan vedligeholde vores produkt. I dette afsnit vil vi gå mere i dybden med hvilke testformater der er snak om.

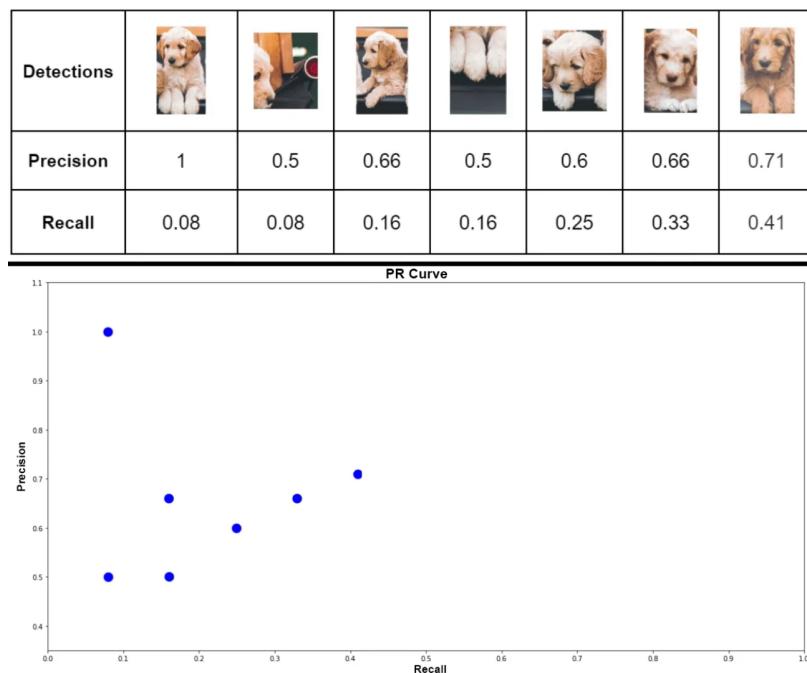
## 13.1 Datasæt test

For at evaluere kvaliteten af ens datasæt til at genkende objekter, kan man udføre en test. Denne test udføres efter modellen er blevet opbygget og kan give indsigt i, om datasættet mangler billeder for at forbedre resultaterne.

Vi har med vilje tildelt 10% af vores datasæt til at teste.

Disse billeder bliver prøvet af i modellen, hvor modellen tager disse billeder, og ser om de kan genkende emnerne og hvor præcist de gør det.

Disse tal bliver beregnet sammen til en mAP (Mean Average Precision) værdi



Figur 16<sup>11</sup> - PR kurve eksempel på genkendelse af hunde

Vi kan ud fra denne værdi se hvor præcis vores model er, og dermed sætte et threads hold for hvor mange procent den skal være for at kunne anvendes i produktion, vi har sat vores threshold på 90%.

Vi kan desuden også lave mAP over hver type komponent, som vil fortælle os om vores datasæt har brug for mere data.

## 13.2 Unit test

Unit tests er en form for test der bliver brugt i kildekode for at være sikker på at en såkaldt *unit* bliver ved med at udføre en opgave med samme resultat som forventet. Der er flere måder at lave og vedligeholde disse test, hvor vi har valgt i vores projekt at forholde os til at vi ikke får 100% dækning af vores kode med Unit tests, men at vi som minimum får lavet, vedligeholde og kørt vores unit tests på de mest nødvendige dele af vores komplette system.

## 13.3 User Acceptance Test

User Acceptance tests er en form for test der sørger for at de beskrevne produkt funktionaliteter faktisk er implementeret på et *accepteret* niveau. Disse skal til hver en version af systemet være udfyldt på ny, for at sikre at brugeroplevelsen ikke forværres. Udfyldte User Acceptance Tests kan findes i vores produktrapport.

---

<sup>11</sup>

<https://learnopencv.com/mean-average-precision-map-object-detection-model-evaluation-metric/#Precision->

## **14 Konklusion**

I vores projekt har vi udviklet en maskine til at sortere elektriske komponenter, baseret på deres egnethed til genbrug eller genanvendelse. Selvom vi var klar over, at det økonomisk set ikke var meningsfuldt, hvis man køber en hel kasse med 200 modstande til kun 30 øre stykket fra f.eks. RS Components, har vi alligevel fundet en effektiv måde at sortere forskellige komponenter på.

Vi har udviklet en nem og skalerbar metode til at sortere store mængder af komponenter, som ville kunne forbedres yderligere, hvis produktet skulle tages i brug til reel produktion. For eksempel ville det være muligt at tilføje flere forskellige komponenttyper til objektgenkendelse datasæt og forbedre transportbåndene, hvilket ville gøre vores sorterings mere præcis og hurtig.

Desuden opdagede vi også at elektriske komponenter uden problemer kunne genanvendes, så længe der blev taget tid til at sortere og teste dem. Dette var en stor øjenåbner, da det ville have stor positiv betydning for miljøet.

Vi opdagede, at det var udfordrende at identificere de faktiske råmaterialer, der udgjorde de forskellige komponenter. Dette skyldes, at det krævede kemisk separering, hvilket gjorde det mere kompliceret at præcisere sammensætningen af komponenterne.

# **15 Logbog**

## **Dag 1 - 11/04**

I dag har vi gennemlæst det udleveret materiale, samt researchet hvad praksis for genbrug af elektroniske komponenter samt samlet generel viden omkring elektronikaffald. Vi har haft en lang konstruktiv diskussion omkring hvilket område vi skal have dækket med vores proof-of-concept / løsning / produkt. Vi har også udarbejdet et udkast til vores problemformulering for at sikre at vi får dækket vores udleveret case på bedst mulig vis samt opsat et kildekode repositories.

## **Dag 2 - 12/04**

I dag har vi reformuleret vores problemformulering, vi har fået startet på vores forundersøgelse. Derudover har vi gennemgået og beskrevet en masse elektroniske komponenter vi vil have fokus på i dette projekt. Samt researchet en masse omkring elektroniske komponenter, deres opbygning og hvilke materialer disse er af.

## **Dag 3 - 13/04**

På dag 3 har vi fået opsat et lokalt netværk til brug i vores proof-of-concept, samt sat en Raspberry Pi op, som vi også vil kunne teste eventuelle genkendelses / målingsudstyr. Derudover har vi arbejdet på vores rige billede samt dannet et overblik over hvilke faser der er nødvendige at opdele vores sorteringssystem i for at kunne lave løsningen.

## **Dag 4 - 14/04**

I dag har vi kigget på at få hentet nogle forud trænede modeller til genstandsgenkendelse til at teste om hvorvidt dette er en mulig fremgangsmåde. Vi har også fået lavet en god kodebase for et Rest API i typescript vha. ExpressJs frameworket og unit tests. Ydermere har vi fået kigget på mulige frontend frameworks til vores web interfaces

## **Dag 5 - 17/04**

I dag har vi fået lov til at låne et transportbånd fra automatikteknikerne på skolen, som vi vil bruge i vores proof-of-concept. Vi har ydermere researchet på Astro frameworket til Javascript, AdminLTE skabelonen og ChartsJs biblioteket til brug i vores web Interface, samt fået lavet et ER-diagram for vores database.

## **Dag 6 - 18/04**

I dag har vi fået startet vores komplette system design, som vi er gået i gang med at udvikle, samt fået researchet en del på Typescript, Javascript, NodeJs, Python og håndtering af anlæg til vores proof-of-concept.

## **Dag 6 - 19/04**

I dag har vi arbejdet med at finde den bedste måde at håndtere billedgenkendelse på objekter, der kører på tværs af vores transportbånd. Derudover har vi påbegyndt udarbejdelsen af vores diagrammer og use-cases til vores produktrapport.

## **Dag 7 - 20/04**

I dag har vi formået at finjustere vores model tilstrækkeligt, så den nu kan genkende objekter, der bevæger sig gennem transportbåndet. Derudover har vi konfigureret vores modificerede 3D-printer til at blive styret af vores Raspberry Pi.

Vi har desuden også fået vores web interface til at præsentere vores testdata visuelt i frontend.

## **Dag 8 - 21/04**

Vi har idag fået skrevet de sidste unit tests og arbejdet videre med det skriftlige i vores procesrapport

### **Dag 9 - 24/04**

I dag har vi brugt en stor del af min tid på at arbejde med mockup af UI'et til vores. Vi har undersøgt forskellige designmuligheder og eksperimenteret med forskellige farveskemaer og layout muligheder for at gøre vores sorteringssystem let at anvende  
Vi fik også skrevet videre på vores procesrapport

### **Dag 10 - 25/04**

I dag har vi brugt en stor del af min tid på at arbejde med at implementere UI på vores sorteringssystem. Det har været en udfordrende opgave, da vi skulle sørge for, at UI'et fungerer optimalt sammen med vores eksisterende system. Vi har brugt tid på at teste og fejfinde. Vi har også skrevet videre på vores procesrapport, hvor vi har beskrevet mere om vores arbejde med objektgenkendelse. Vi har undersøgt forskellige metoder og teknologier for objektgenkendelse og har skrevet om vores resultater og erfaringer med at implementere denne teknologi i vores system. Det var en spændende proces, og vi er glade for at have fundet en løsning, der fungerer godt for vores projekt.

### **Dag 11 - 26/04**

I dag har vi arbejdet med sikkerheden i vores system. Det har været en vigtig opgave at sørge for, at vores system er sikkert for vores brugere og beskytter deres data mod potentielle trusler. Vi har også arbejdet videre med strukturen af vores API. Det var en vigtig opgave at sikre, at API'et fungerer optimalt med en god struktur.

### **Dag 12 - 27/04**

I dag har vi arbejdet videre med sikkerheden i vores system. Vi har også skrevet videre på vores procesrapport, og strukturen af vores API

### **Dag 13 - 28/04**

I dag brugte vi hele dagen på at skrive videre på vores procesrapport og produktrapport

### **Dag 14 - 01/05**

I dag har vi skrevet videre på vores procesrapport. Vi har også arbejdet videre med vores projektrapport, hvor vi har arbejdet på at tilføje yderligere information om vores produkt og dets funktioner. Vi har også tilføjet et afsnit om FURPS - en model, vi bruger til at specificere kravene til vores produkt. Derudover har vi kortlagt, hvilke diagrammer der mangler i vores rapport, og har planlagt at arbejde på at tilføje disse i de kommende dage.

### **Dag 15 - 02/05**

I dag har vi arbejdet på at teste vores datasæt ift. objektgenkendelse. Vi har arbejdet på at træne vores algoritmer og finjustere vores modeller for at forbedre vores genkendelses resultater. Vi har også analyseret vores data for at identificere eventuelle svagheder og forbedringsmuligheder i vores objektgenkendelsessystem.

Derudover har vi arbejdet på vores frontend.

### **Dag 16 - 03/05**

I dag har vi færdiggjort vores test af datasættet ift. objektgenkendelse. Vi har sat en grænse for, hvornår vores datasæt er godt nok til at blive brugt i vores objekt genkendelsessystem. Vi har gennemgået vores testresultater og konkluderet, at vi er tilfredse med vores resultater og er klar til at implementere vores system.

Vi har også arbejdet videre på vores produktrapport. Vi har fokuseret på at beskrive vores produkt og dets funktioner mere detaljeret.

### **Dag 17 - 04/05**

I dag har vi haft en travl dag med flere forskellige opgaver. Først og fremmest har vi skrevet konklusionen af vores projekt. Vi har samlet vores resultater og reflekteret over vores arbejde, og vi er tilfredse med den endelige konklusion. Vi har også arbejdet på at udarbejde diagrammer til vores rapport. Vi har oprettet forskellige diagrammer. Endelig har vi udarbejdet vores testrapport. Vi har beskrevet vores testmetoder og resultaterne af vores tests.

### **Dag 18 - 05/05**

I dag har vi fokuseret på at færdiggøre vores frontend og machine learning. Vi har brugt tid på at fejlfinde og optimere vores kode for at sikre, at vores produkt fungerer korrekt og er brugervenligt.

Vi har også arbejdet videre med at skrive på vores procesrapport. og få kortlagt hvilke ting der mangler

### **Dag 19 - 08/05**

Vi er nået til den sidste dag og har færdiggjort vores rapport og vores kodebase for alle delsystemerne af vores produkt. Det sidste vi ikke får med her er vores præsentation af projektet som vil blive udarbejdet i dagene 09/05 - 12/05.