

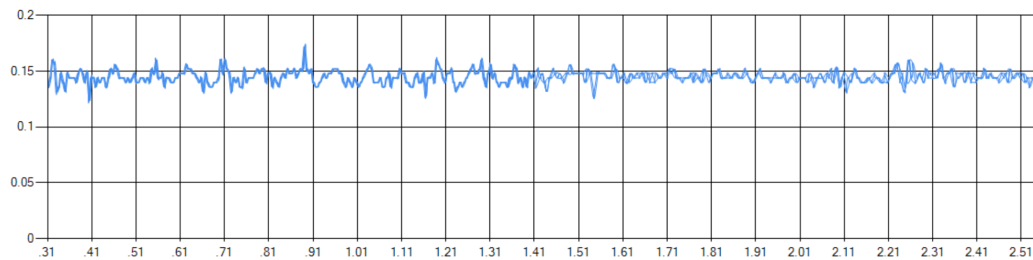


## Time Series

Samples: 2000

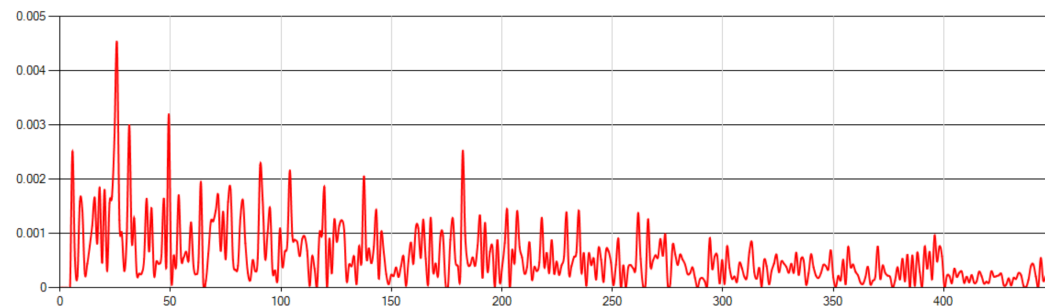
Time: 2.23624992370605

Sample Rate: 0.00111812496185303



## Frequency Analysis

Top Frequency: 26



# Bestämna och visualisera maskinstatus med hjälp av enkla sensorer

Kandidatarbete inom Produkt- och produktionsutveckling

RICKARD ERIKSSON

KRISTOFFER GERLE

HENRIK HALLBERG

ELIAS SONNSJÖ LÖNEGREN

DAVID STAVMYR

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet. The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

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

Determine and visualize machine status using simple sensors

RICKARD ERIKSSON  
KRISTOFFER GERLE  
HENRIK HALLBERG  
ELIAS SONNSJÖ LÖNEGREN  
DAVID STAVMYR

© RICKARD ERIKSSON, 2016.  
© KRISTOFFER GERLE, 2016.  
© HENRIK HALLBERG, 2016.  
© ELIAS SONNSJÖ LÖNEGREN, 2016.  
© DAVID STAVMYR, 2016.

Supervisor: Jon Laborn  
Examiner: Torbjörn Ytterle

Chalmers University of Technology  
University of Gothenburg  
Department of Product and production Development  
SE-412 96 Göteborg  
Sweden  
Telephone +46 (0)31-772 1000

Department of Product and production Development

# Sammandrag

Underhåll definieras som den process som används där ett objekt återställs till det tillstånd där det kan utföra sina utsedda funktioner. De senaste decenniernas utveckling av teknik har medfört en drastisk utveckling av underhållsmetoder. Underhåll kan delas in i tre huvudkategorier: Avhjälpande underhåll, Förebyggande underhåll, vilka är traditionella underhållsmetoder, samt Konditionsbaserat underhåll.

Konditionsbaserat underhåll är en underhållsmetod som, i förhållande till de traditionella underhållsformerna ökar resursutnyttjandet i en process. I dagsläget finns det dock en nackdel med konditionsbaserat underhåll, nämligen de stora uppstarts-kostnaderna jämfört med de andra underhållsmetoderna.

Projektet syftade till att undersöka möjligheterna att, med hjälp av enkla och billiga medel, skapa ett system som baserat på IOT kunde förenkla det konditionsbaserade underhållet. Projektets grundades genom ett kandidatarbete som utfördes på Chalmers Tekniska Högskola våren 2015, vilket undersökte möjligheterna att använda enkel hårdvara vid vibrationsmätning för att ge underlag till bedömning av roterande maskiners kondition.

Utförandet byggde på programmering av hårdvara, där de viktigaste komponenterna var enkortsdatorn Raspberry Pi samt acceleremometermodulen ADXL345. Resultatet visade på att anordningen som utvecklades i detta projekt både mätte snabbare, var mindre samt även var mer lättanvänt jämfört med anordningen som byggdes i förra årets projekt.

**NYCKELORD:** underhåll, programmering, Raspberry Pi, accelerometer, ADXL345, IoT, konditionsbaserat



# Abstract

Maintenance is defined as the process where an object is restored to the state in which it is able to perform its designated functions. The development of technology over the latest decades has brought a drastic development also in the methods of maintenance. Maintenance can be divided into three main categories; corrective maintenance, preventive maintenance, which are traditional methods of maintenance, and condition-based maintenance.

Condition-based maintenance is a maintenance method which, compared to the traditional maintenance methods, increases resource utilization in a process. There is however a great disadvantage with condition-based maintenance in the large start-up costs in comparison to other maintenance methods.

The project aims to explore the possibilities, with the use of simple and inexpensive means, of creating a system based on IOT which could simplify the condition-based maintenance. This project is sprung out of a bachelor report, carried out at Chalmers University in spring 2015, which explored the possibility of using simple hardware for vibration measurements in order to provide a basis for determine the condition of rotating machines.

The execution of the project was based on the programming of the hardware, in which the main components were the single-board computer Raspberry Pi and an accelerometer module called ADXL345. The results showed that the device developed in this project, both measured faster, was lighter and was also easier to use in comparison to the device that was built in the last year's project.

**KEYWORDS:** maintenance, programming, Raspberry Pi, accelerometer, ADXL345, IoT, condition-based



# Tack till

Vi vill tacka vår handledare Jon Larborn för hans inspirerande engagemang samt  
medarbetarna på PPU för deras assistans.



# Ordlista

**FFT** - Fast Fourier Transform - Snabb algoritm för beräkning av begränsad, diskret Fouriertransform

**Fouriertransform** - Generalisering av Fourierserie

**MATLAB** - Programvara för bland annat matematiska och tekniska beräkningar

**Sampeltid** - I denna rapport tid mellan två mätningar

**Samplingsfrekvens** - Frekvensen mätningen sker i, beräknas  $1/\text{Sampeltid}$

**Klockhastighet** - Processorns klockfrekvens

**IoT** - Internet of Things

**PWM** - Pulse-width modulation, teknik för att styra genomsnittlig utspänning från en digital utgång.

**Hall-effektsensor** - Sensor som registrerar magnetfält. Används ofta för att mäta varvtal.

**Flashminne** - Transistorbaserat lagringsminne

**Enkortsdator** - Dator byggd på ett enda kretskort, inkluderar CPU, I/O, ethernet-port mm.

**Mikrokontroller** - Enklare typ av enkortsdator, innehåller ofta en långsammare processor än en dator, programmeras ofta i lågnivåspråk som C eller basic.

**IP-klass** - Klassifiering för motstånd mot inträngning av objekt, såsom exempelvis smuts eller vätska

**ARM** - Advanced RISC Machine, en processorarkitektur



# Innehållsförteckning

|          |                                                                               |           |
|----------|-------------------------------------------------------------------------------|-----------|
| <b>1</b> | <b>Inledning</b>                                                              | <b>1</b>  |
| 1.1      | Syfte . . . . .                                                               | 2         |
| 1.2      | Förutsättningar . . . . .                                                     | 2         |
| 1.3      | Avgränsningar och begränsningar . . . . .                                     | 2         |
| 1.3.1    | Hård- och Mjukvarubegränsningar . . . . .                                     | 2         |
| 1.4      | Problemformulering . . . . .                                                  | 3         |
| 1.5      | Rapportens disposition . . . . .                                              | 3         |
| <b>2</b> | <b>Teori</b>                                                                  | <b>5</b>  |
| 2.1      | Konditionsbaserat underhåll . . . . .                                         | 5         |
| 2.1.1    | Det konditionsbaserade underhållets historia . . . . .                        | 5         |
| 2.1.2    | Utförande av konditionsbaserat underhåll . . . . .                            | 6         |
| 2.1.2.1  | Vibrationsmätning inom konditionsbaserat underhåll                            | 6         |
| 2.1.2.2  | Vibrationsmätningens historik . . . . .                                       | 7         |
| 2.1.3    | Nackdelar och framtid för konditionsbaserat underhåll . . . . .               | 7         |
| 2.2      | IoT - Internet of things . . . . .                                            | 8         |
| 2.2.1    | IoT inom det konditionsbaserade underhållet . . . . .                         | 9         |
| 2.3      | Hård-och mjukvara . . . . .                                                   | 9         |
| 2.3.1    | Sensortyper för vibrationsmätning . . . . .                                   | 9         |
| 2.3.2    | Lagring av vibrationsdata . . . . .                                           | 10        |
| 2.3.3    | Raspberry Pi . . . . .                                                        | 10        |
| 2.3.4    | Microsoft Visual Studio . . . . .                                             | 12        |
| 2.3.5    | .NET Gadgeteer . . . . .                                                      | 12        |
| 2.3.6    | Arduino UNO ® . . . . .                                                       | 13        |
| <b>3</b> | <b>Metod</b>                                                                  | <b>15</b> |
| 3.1      | Inledningsfas . . . . .                                                       | 15        |
| 3.1.1    | Marknadsundersökning . . . . .                                                | 16        |
| 3.1.2    | Brainstorming . . . . .                                                       | 17        |
| 3.1.3    | Referenslösning . . . . .                                                     | 17        |
| 3.1.4    | Kundvärdesmatriser . . . . .                                                  | 17        |
| 3.1.5    | Funktionella modeller . . . . .                                               | 18        |
| 3.1.6    | Kravspecifikationer . . . . .                                                 | 18        |
| 3.2      | Utvecklingsfas . . . . .                                                      | 19        |
| 3.2.1    | Utveckling av mjukvara för datainsamling, visualisering samt analys . . . . . | 19        |

|          |                                                               |           |
|----------|---------------------------------------------------------------|-----------|
| 3.2.2    | Utveckling av design för inkapsling av moderkort . . . . .    | 19        |
| 3.3      | Utvärderingsfas . . . . .                                     | 20        |
| 3.3.1    | Utvärdering av mätsystemen . . . . .                          | 20        |
| <b>4</b> | <b>Inledningsfas</b>                                          | <b>21</b> |
| 4.1      | Inventering av kända koncept . . . . .                        | 21        |
| 4.1.1    | Marknadsundersökning . . . . .                                | 21        |
| 4.1.2    | Summering av marknadsundersökningen . . . . .                 | 23        |
| 4.1.3    | Referenslösning för det nya systemet . . . . .                | 23        |
| 4.2      | Vidareutveckling av tidigare system . . . . .                 | 24        |
| 4.2.1    | Utveckling mot kontakt med server . . . . .                   | 24        |
| 4.2.2    | Utveckling av inkapslande chassi . . . . .                    | 26        |
| 4.3      | Utveckling av ett nytt system . . . . .                       | 28        |
| 4.3.1    | Brainstorming . . . . .                                       | 28        |
| 4.3.2    | Kundvärdesmatris samt kravspecifikation . . . . .             | 29        |
| 4.3.3    | Val av hårdvara för datainsamling . . . . .                   | 31        |
| 4.3.3.1  | Val av moderkort . . . . .                                    | 31        |
| 4.3.3.2  | Val av accelerometer . . . . .                                | 31        |
| 4.3.4    | Funktionell modell . . . . .                                  | 32        |
| 4.4      | Utveckling av webbsida för visualisering av mätdata . . . . . | 33        |
| 4.5      | Reflektion av inledningsfasen . . . . .                       | 34        |
| <b>5</b> | <b>Utvecklingsfas</b>                                         | <b>35</b> |
| 5.1      | Utveckling av det tidigare systemet . . . . .                 | 35        |
| 5.1.1    | Mjukvara . . . . .                                            | 35        |
| 5.1.1.1  | Internetanslutning för mätsystemet . . . . .                  | 35        |
| 5.1.1.2  | Kommunikation mellan mätsystem och databas/server . . . . .   | 36        |
| 5.1.1.3  | Reflektion av webservice/kommunikation . . . . .              | 38        |
| 5.1.2    | Utveckling av inkapslande chassi . . . . .                    | 38        |
| 5.2      | Utveckling av nytt mätsystem . . . . .                        | 39        |
| 5.2.1    | Installation av vibrationssensor . . . . .                    | 39        |
| 5.2.1.1  | Installation av Accel G248 Module . . . . .                   | 39        |
| 5.2.1.2  | Installation av ADXL345 . . . . .                             | 40        |
| 5.2.2    | Utveckling av mjukvara . . . . .                              | 40        |
| 5.2.3    | Utveckling av hårdvara . . . . .                              | 42        |
| 5.3      | Utveckling av webbsida för visualisering av mätdata . . . . . | 44        |
| 5.4      | Reflektion på utvecklingsfasen . . . . .                      | 45        |
| 5.4.1    | Reflektion utveckling av mjukvara . . . . .                   | 45        |
| <b>6</b> | <b>Utvärderingsfas</b>                                        | <b>47</b> |
| 6.1      | Utveckling av nytt testsystem . . . . .                       | 47        |
| 6.1.1    | Referenslösning . . . . .                                     | 47        |
| 6.1.2    | Konceptgenerering . . . . .                                   | 47        |
| 6.1.3    | Funktion och CAD-modell . . . . .                             | 48        |
| 6.1.4    | Motorstyrning . . . . .                                       | 49        |
| 6.1.5    | Reflektion utveckling av nytt testsystem . . . . .            | 50        |
| 6.2      | Validering av mätsystem . . . . .                             | 50        |

|          |                                                       |            |
|----------|-------------------------------------------------------|------------|
| 6.2.1    | Inledande analyser . . . . .                          | 50         |
| 6.2.1.1  | Reflektion inledande analyser . . . . .               | 50         |
| 6.2.2    | Fortsatta analyser . . . . .                          | 51         |
| 6.2.2.1  | Tester på högtalare . . . . .                         | 51         |
| 6.2.2.2  | Tester på roterande maskiner . . . . .                | 52         |
| 6.2.2.3  | Reflektion fortsatta analyser . . . . .               | 53         |
| 6.3      | Reflektion på utvärderingsfasen . . . . .             | 53         |
| <b>7</b> | <b>Diskussion och slutsats</b>                        | <b>55</b>  |
| 7.1      | Framtid . . . . .                                     | 56         |
|          | <b>Referenser</b>                                     | <b>57</b>  |
| <b>A</b> | <b>CAD-ritning bottenplatta, tidigare system</b>      | <b>I</b>   |
| <b>B</b> | <b>CAD-ritning lock, tidigare system</b>              | <b>III</b> |
| <b>C</b> | <b>CAD-ritning inkapsling, nytt system</b>            | <b>V</b>   |
| <b>D</b> | <b>CAD-ritning inkapsling för sensor, nytt system</b> | <b>VII</b> |



# 1

## Inledning

Maskinhaverier kan skapa mycket stora kostnader för industriföretag i form av utebliven produktionstid. Underhåll av maskiner är därför mycket viktigt för att hålla maskinerna i fungerande skick. I dagsläget är underhållsarbetet uppdelat i tre metoder: avhjälpande, förebyggande samt konditionsbaserat underhåll. Den förstnämnda metoden, avhjälpande underhåll, är den typ där maskinen återställs till fungerande status först då den har fallerat. Förebyggande underhåll är den typ där maskinen underhålls i förebyggande syfte, ofta genom erfarenhet eller riktlinjer. Den sistnämnda, konditionsbaserat underhåll, är av typen där maskinens status övervakas, för att sedan utföra underhåll just när det behövs.

Förebyggande underhåll, frekvent använd i industrin, har ett flertal brister. Underhållsarbete upplagt rutinmässigt på vecko- eller månadsbasis leder till många onödiga stopp i produktionen, samtidigt som intervallerna är baserade på analyser och erfarenhet. Det framtagna underhållsintervallet är dessutom ofta enbart en kvalificerad gissning [1]. Underhåll utifrån gissningar eller erfarenheter leder ofta till onödigt underhåll, vilket kan kosta industrier miljontals kronor [2].

När exempelvis kullagren i en maskin börjar bli slitna kommer maskinens vibrationer intensifieras och samtidigt börjar den ofta låta annorlunda [3]. Att via empiriska studier samla in samt analysera exempelvis vibrationsdata kan det vibrationsmönster som uppträder i samband med maskinhaverier fastställas. När det fastställda mönstret uppträder byts det aktuella lagret ut och drifttiden optimeras utan haveristopp. Den här typen av stopp kan kosta bilindustrin upp till 180 000 SEK <sup>1</sup> per minut och är därför högst åtråvärda att minska [4].

I kandidatarbetet *Prototyp för konditionsbaserat underhåll med enkla sensorer* [22] undersöker författarna hur man med hjälp av billiga sensorer kan fastställa den här typen av problem i produktion, och vid rätt tidpunkt. Att med hjälp av dessa sensorer skapa en grund för konditionsbaserat underhåll kan spara enorma pengar. I vissa fall kan så mycket som 70 procent av den totala produktionskostnaden vara underhållskostnad, och så mycket som en tredjedel av den här kostanden är ett resultat av felaktigt underhåll [5]. Att använda billiga och enkla sensorer är åtråvärt för, stora som små, industriföretag som vill undvika stopp och slippa köpa in den dyra utrustningen som finns på marknaden idag.

---

<sup>1</sup>\$22 000/minut 2006, omräknat och justerat för inflation

### 1.1 Syfte

Projektet syftar till att undersöka huruvida enkla standardsensorer<sup>2</sup>, i kombination med enkortsdatorn Raspberry Pi är en bra lösning på industrins problem gällande underhåll. Projektgruppen ska skapa ett nytt system baserat på IoT, där fokus ligger på visualisering och analysering av datan för att ge operatören tydlig indikation om aktuell maskinstatus.

Det gamla systemet, FEZ Raptor som togs fram i *Prototyp för konditionsbaserat underhåll med enkla sensorer*, ska utvecklas och jämföras med Raspberry Pi.

Dessa enheter ska med hjälp av en server kommunicera med en webbaserad hemsida, där den insamlade och analyserade informationen presenteras på ett överskådligt sätt.

### 1.2 Förutsättningar

Förra årets projekt[22] väckte idén att med hjälp av enkla och billiga medel föra det konditionsbaserade underhållet mot IOT. Vid starten av projektet gavs projektgruppen plattformen som skulle utvecklas.

De förutsättningar som finns för detta projektet är förra årets kandidatarbete [22] där ett moderkort av typen FEZ-raptor med tillhörande mjukvara finns tillgänglig, samt en låda som fungerar som skydd för moderkortet. Mjukvaran till FEZ-raptorn ska utvecklas för att bli optimerat mot kommunikation med en server.

Systemet som skulle utvecklas skulle kunna mäta vibrationer på roterande maskiner, skicka mätdatan till en server och till sist kunna på ett grafiskt sätt visa upp datan.

### 1.3 Avgränsningar och begränsningar

Projektet kommer exkluderade begränsas till vibrationsmätning på roterande maskiner, då vibrationer på dessa är vanligt förekommande. De sensorer som används behöver, per definition av syftet, vara relativt billiga och enkla<sup>3</sup>.

Projektet är även tidsbegränsat då det löper över en termin och cirka 20 timmars jobb per person och vecka är avsatt.

#### 1.3.1 Hård- och Mjukvarubegränsningar

Utvecklingen av mätsystemet, kommunikation från server till hemsida kommer att begränsas till enkortsdatorn Raspberry Pi.

---

<sup>2</sup>Den typen av sensorer som kan förvärvas av privatpersoner för en liten summa pengar

<sup>3</sup>Se fotnot 2

Projektgruppen kommer att ta fram en prototyp duglig för att presentera storlek och användning av produkten. Dock kommer inget fokus läggas på att göra produkten godkänd för industrin i form av exempelvis IP-klassning, då produkten enbart syftar till gruppens egen förståelse.

## 1.4 Problemformulering

För att uppnå projektets syfte har fyra delsyften tagits fram. Dessa delsyften är mer detaljerade och bygger tillsammans projektets huvudsyfte. Projektgruppen skall visa att det går att ta fram ett industridugligt koncept, med enkla och billiga sensorer, som kan förebygga haverier utifrån konditionsbaserat underhåll. För att uppnå detta delas syftet in i fyra delar:

- 1) Projektgruppen ska utveckla ett nytt mätsystem som analyserar vibrationer, baserat på Raspberry Pi. Systemet skall vara funktionsdugligt och därmed kunna hantera vibrationsdata som mäts av en accelerometer.
- 2) Projektgruppen skall utveckla den mätanordning vilken analyserade vibrationer och togs fram i "Prototyp för konditionsbaserat underhåll med enkla sensorer" [22], för att även den skall uppfylla projektets syfte.
- 3) De båda systemen skall utvecklas för att med hjälp av internet visualisera maskinstatus på en hemsida.
- 4) Inkapslingschassi-prototyper ska tas fram för de båda systemen.

I sin helhet ska gruppen utveckla en helhetslösning som samlar in, analyserar samt grafiskt visar resultatet av vibrationsdata.

## 1.5 Rapportens disposition

Rapporten är upplagd på ett sätt för att läsaren först ska få en bakgrund av projektet, förståelse kring tekniken och de metoder som senare används och slutligen vilka resultat som tagits fram.

Rapportens första kapitel består av en teoretisk referensram där de tekniska förkunskapskrav som krävs samt tidigare arbeten och allmän teori på området presenteras. I metodkapitlet förklaras vilka metod- och teoriverktyg som används i utvecklingen av konceptet, såsom konceptgenerering, produktutveckling samt prototypframtagning. Vidare är resultatkapitlet upplagt enligt tre faser, där varje fas motsvarar en fastställd tid och presenterar projektets framfart på olika fronter under fasen. Till sist kan läsaren i diskussion och slutsats läsa projektgruppens generella slutsats och utvärdering av projektet, hur projektet har uppfyllt diverse mål samt en rekommendation kring hur projektet bör föras vidare.



# 2

## Teori

I detta kapitel skall läsaren få en grundläggande bild över vad konditionsbaserat underhåll är, hur det utförs samt vilka tekniska verktyg som kan användas vid utveckling av ett system för konditionsbaserat underhåll.

### 2.1 Konditionsbaserat underhåll

Konditionsbaserat underhåll (engelska: Condition based maintenance) är en underhållsstrategi som övervakar det aktuella tillståndet hos en utrustning eller maskin för att avgöra när eventuell service eller reservdelsbyte behöver genomföras. Principen utgår från att endast underhålla en utrustning när den indikerar risk för minskad prestanda eller eventuellt ett framtida haveri. Till skillnad från det klassiska planerade underhållet kan intervallen mellan underhåll förlängas med konditionsbaserat underhåll, då underhåll endast genomförs när det finns ett behov, i precis rätt tid. [1]

#### 2.1.1 Det konditionsbaserade underhållets historia

Det traditionella underhållet har funnits i flera varianter: allt från Run-to-fail-principen [23] till planerat underhåll [24]. Run-to-fail är den simplaste av dessa strategier och bygger på att avsiktligt tillåta komponenter verka till den punkt att de går sönder, och därefter genomförs nödvändigt, avhjälpande underhåll.

Inom vissa områden eller på särskilda maskiner, där ett eventuellt stopp inte är betydande, används fortfarande Run-to-fail-principen, men i övrigt är strategin utdaterad. Istället använder sig de flesta industrier av olika varianter av planerat, eller förebyggande, underhåll. För varje enskild komponent och maskin finns fastställda cykler för service, som resultat av dokumentering och beräkningar för att genomföra nödvändiga byten innan ett haveri inträffar. Generellt är detta en mycket mer effektiv strategi gentemot Run-to-fail då onödiga stopp ofta kan undvikas. Samtidigt utnyttjas sällan komponenter och maskindelar till fullo, då byten sker enligt tabell oberoende av dess skick.

Av naturliga skäl har konditionsbaserat underhåll därför växt fram som en kraftfull konkurrent till de mer klassiska strategierna. Konditionsbaserat underhåll var till en början enbart baserat på visuella kontroller [25], där man lät vissa indikationer påvisa när service var tvungen att genomföras. Successivt, allteftersom

att tekniken förbättrats, har dessa visuella kontroller blivit ersatta med exempelvis vibrationsmätning- och analyser [26].

Haverierna som kan förväntas av Run-to-fail-metoden undviks helt och det slöseri som uppstår vid planerat underhåll, med icke utnyttjade komponenter, elimineras och företag kan därmed lägga fokus på värdeadderande aktiviteter. Dessutom drar företaget ner på rena underhållskostnader[27] vilka tidigare påvisats uppgå till 70 procent av den totala produktionskostnaden.

### 2.1.2 Utförande av konditionsbaserat underhåll

Målet med konditionsbaserat underhåll är att ta reda på maskinens status samtidigt som maskinen är i drift [28], vilket kan utföras på flera olika sätt. Tre exempel är oljeanalys, analys av ljud samt vibrationsanalys [28]. Oljan i en maskin kan berätta mycket om hur maskinen mår: en analys kan visa på exempelvis förslitning av drev eller läckage [28]. Även ljud och vibrationer från en maskin berättar mycket om hur den mår, abnormala ljud eller vibrationer kan tyda på exempelvis slitna lager, eller fel arbetstakt [15] [28].

Konditionsbaserat underhåll med hjälp av oljeanalys eller analys av ljud kommer inte förklaras närmare då detta projekt är inriktat på analys av vibrationer.

Konditionsbaserat underhåll består av antingen ett utförande av permanent övervakning eller tillfällig övervakning av maskinstatus [28]. Vid permanent övervakning sitter en mätutrustning på maskinen hela tiden. Den typen av mätutrustning samlar in stora mängder data, vilket sammanställs i vibrationsprofiler som sparas undan. Vid permanent övervakning skapas ofta en mycket säker vibrationsprofil över maskinen. Eftersom det konditionsbaserade underhållet ser hur en maskin mår direkt [28] [15] så kan även en tillfällig övervakning av en maskin användas för att bestämma status. Vid denna typ av övervakning sitter en mätutrustning på maskinen en kortare tid, och samlar därmed in mindre mängd data [28]. Fördelen med en tillfällig övervakning är att kostnaderna blir lägre [15].

#### 2.1.2.1 Vibrationsmätning inom konditionsbaserat underhåll

Att diagnostisera felet och dess omfattning genom att samla upp dessa vibrationer och tolka dem är idag det vanligaste sättet att utföra konditionsbaserat underhåll hos roterande maskiner [16]. Vibrationsanalysen utförs genom att sensorer av olika slag mäter hur exempelvis en maskin eller ett lager vibrerar under en viss arbetstakt. Maskiner vibrerar alltid under drift, exempelvis på grund av att lager snurrar, olja flödar, axlar rör sig och likande [15] [16]. Dessa vibrationer kan därmed kopplas till specifika händelser, och en vibrationsprofil kan då skapas för en maskin [15]. Vibrationsanalysen bygger alltså på att ta fram de vibrationer som inte passar in i vibrationsprofilen [28], och att förstå vad de abnormala vibrationerna kommer ifrån. Kan dessa vibrationer upptäckas, analyseras samt begripas ger det en möjlighet till att orsaken kan åtgärdas innan det är för sent [15] [28].

### 2.1.2.2 Vibrationsmätningens historik

Idén om att kunna utröna en maskins status bör ha sitt ursprung redan med framtagningen av de första roterande maskinerna. Dock var det först senare, vid år 1939, som de första riktlinjerna för hur maskinstatus skulle utrönas ur vibrationer togs fram. Dessa publicerades i en artikel kallad "Vibration Tolerance" i tidskriften *Power Plant Engineering*[17]. Överlag kan det sägas att tekniken att använda sig av vibrationer som verktyg för att utröna maskinstatus genomgått fem stora utvecklingssteg. Fastställandet av att vibrationer hos en maskin har en koppling till statusen kan anses vara det första steget. Därefter togs ett verktyg fram för att kunna analysera vibrationerna. Vår tids teknik grundar sig sannolikt i det resultat Art Grawford lyckades åstadkomma när han antog utmaningen att lista ut hur balansering av höghastighetsspindlar ska ske, vilket senare resulterade i *International Research and Development* (hädanefter förkortat IRD). IRD ansågs därefter ledande inom frekvensanalysområdet i de kommande 30 åren. Då det hade utvecklats enklare verktyg för att kunna mäta och tolka data från maskinerna startade steg tre i utvecklingen. Det började då utvecklas teknik för att kunna utföra kontinuerlig mätning. Vid den här tiden började även accelerometrar användas för att samla in data istället för det tidigare hastighetsmätare. Detta gjorde det möjligt att mäta ett större frekvensomfång än tidigare.

De analysmetoder som var tillgängliga vid detta laget var inte särskilt effektiva och utvecklingen av att bättre kunna tolka och separera signalerna som nu kunde samlas in kan anses vara det fjärde stora steget i utvecklingen. Nu började användningen av Fouriertransformer för att bättre kunna analysera datan tillämpas. Det sista steget i utvecklingen var utvecklingen av handhållna vibrationsmätningssinstrument, där det första kommersiella produkten, AVM-1, togs fram år 1982 [17].

### 2.1.3 Nackdelar och framtid för konditionsbaserat underhåll

Det konditionsbaserade underhållet är just nu sett som den underhållsstrategi som är mest lönsam, effektiv samt säker [28]. Dock har även denna underhållsstrategi nackdelar där den största nackdelen, för många ses, som uppstartningskostnaderna.

Enligt kapitel 2.1.2 utförs konditionsbaserat underhåll tillfälligt eller permanent, där den permanenta övervakningen ger en bättre säkerhet [28]. Att införa permanent övervakning av en maskins status är, främst för mindre företag, en stor investering som kan vara svår att bekosta, vilket leder till att vissa företag istället satsar på någon annan underhållsstrategi, såsom klassiska planerat underhåll.

Framtiden för konditionsbaserat underhåll ser mycket ljus ut. Att de elektriska komponenterna blir billigare leder till att enheterna för vibrationsanalys blir billigare att framställa. I slutändan resulterar detta i att kunden behöver göra en mindre investering. Därmed kan fler företag använda sig av konditionsbaserat underhåll. En annan framtidsmöjlighet är att minska osäkerheten som finns vid exempelvis vibrationsmätning.

### 2.2 IoT - Internet of things

Internet of Things kan definieras som den utveckling som innebär att maskiner, fordon, hushållsapparater med mera, kopplas samman med hjälp av inbyggda sensorer och processorer för att få dessa att kommunicera med sin omvärld. IoT ger maskinerna möjligheten att rapportera sin status till omvärlden och skapa ett situationsanpassat beteende[12]. IoT är inte ett vedertaget begrepp utan ska mer ses som ett koncept där maskiner är anslutna till en server via internet och uppdaterar sin status så att den kan följas.

Begreppet Internet of Things är relativt nytt då det myntades först år 1999 av Kevin Ashton. Det har dock tagit fram till en bit in på 2000-talet för att få IoT att slå igenom. Att det tog en bra bit in på 2000-talet är till stor del på grund av den mängd data som hanteras, vilket på 90-talet var en mycket kostsam verksamhet. Först när lagringen på nätet (molnlagring) blev en utvecklad tjänst kunde utvecklingen och utnyttjandet ta fart. IoT är helt enkelt ett samlingsbegrepp för när olika enheter samverkar och kommunicerar med varandra genom internet[13][14].

Konceptet med maskiner anslutna till en server via internet har implementerats som lösning i många olika scenarier, såsom hälso-och sjukvård, säkerhet, industri med flera. Det finns dock ingen tydlig och bestämd metod för att utforma lösningar baserade på konceptet IoT, istället utvärderas ofta processen efter det att systemet redan är implementerat, vilket kan vara riskfullt och dyrt.[47] IoT kan implementeras inom många olika områden, på många olika sätt. För att ge läsaren förståelse för begreppets bredd nämns några exempel som involverar IoT:

#### **IoT: Transport/Logistik**

Inom transportlogistik förbättrar inte bara IoT materialflöden och processer, dessutom kan GPS ge möjlighet till spårning av paket och leveranser, därmed kan en automatisk identifiering av gods öka effektiviteten i ett logistiksystem. Sammanfattningsvis kan det med hjälp av ett system där olika komponenter ständigt uppdaterar sin information förväntas medföra stora förändringar inom global logistik genom en kontinuerlig synkronisering av olika transportsystem och ett mer visuellt flöde för fraktgods och objekt.[45]

#### **IoT: Det smarta hemmet**

Framtida hem kan komma att utnyttja elektronik som är uppkopplade mot internet, främst inom tre olika aspekter; resursanvändning, komfort och säkerhet. Genom att övervaka resursanvändning som till exempel vattenförbrukning och temperatur kan komforten förbättras samtidigt som de totala utgifterna för energianvändning minskas. Olika säkerhetssystem för exempelvis brand och stöld kan vara uppkopplade och räddningstjänst och vaktbolag kan direkt ta del av och bli informerade av aktuell status i hemmet.[45]

### 2.2.1 IoT inom det konditionsbaserade underhållet

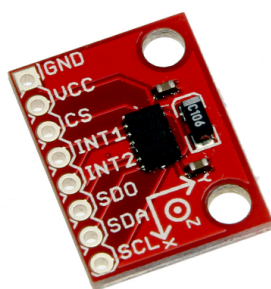
I och med att det konditionsbaserade underhållet blir viktigare inom industrin ökar behovet av maskiner som kan kommunicera sin status med operatörer. Detta har lett till att IoT spelar en avgörande roll för fortskridandet av maskiners "röst". Tidigare, då maskinerna inte kunnat förmedla underhålls krav, krävde underhållet bestämda intervall vilka är kostsamma och ineffektiva. Dessutom är risken för att den "mänskliga faktorn" leder till haverier då IoT inte automatiserar genomförandet. Om maskinen kan förmedla när en del är sliten och behöver underhållas, alternativt bytas, kan ett haveri undvikas i större utsträckning[20].

## 2.3 Hård-och mjukvara

I detta delkapitel kommer de tekniska verktyg som använts genom projektets gång presenteras för att ge läsaren en förståelse för de tekniska detaljerna kring projektet.

### 2.3.1 Sensortyper för vibrationsmätning

Mätning av vibrationer görs till största del av två olika sensortyper, piezo eller mikroelektromekaniska (MEMS). Piezo-accelerometrar använder sambandet mellan ett materials resistansförändring kontra kraftutsättning för att beräkna acceleration [31], medans MEMS-accelerometrar i grunden är maskinelement i en förminskad skala [33]. MEMS-accelerometrar har ofta låg produktionskostnad och används i allt från mobiler till krockkuddar [31].



**Figur 2.1:** ADXL345 monterad på ett kretskort från Sparkfun. Bild licensierad under CCBY-NC-SA3.0

I detta projekt användes en mikroelektromekanisk accelerometer lödd på ett kretskort från Sparkfun [34], detta för att förenkla användningen av accelerometern, vilken benämns ADXL345 och är tillverkad av Analog Devices [32].

### 2.3.2 Lagring av vibrationsdata

Vibrationssensorns data måste sparas undan någonstans för att kunna analyseras [32]. Detta går att lösa på flera sätt, ett exempel är hur de gjorde i kandidatarbetet *Prototyp för konditionsbaserat underhåll med enkla sensorer* [22]. Där sparas datan från sensorn på ett minneskort som sitter monterat på apparaturen. Detta leder till ett extra moment där minneskortet måste plockas ut ur apparaturen, för att sedan sättas in i en annan dator där datan sedan analyseras. Man kan slippa detta moment, samtidigt som användaren får tillgång till datan snabbare, genom att apparaturen skickar datan till en server istället.

### 2.3.3 Raspberry Pi

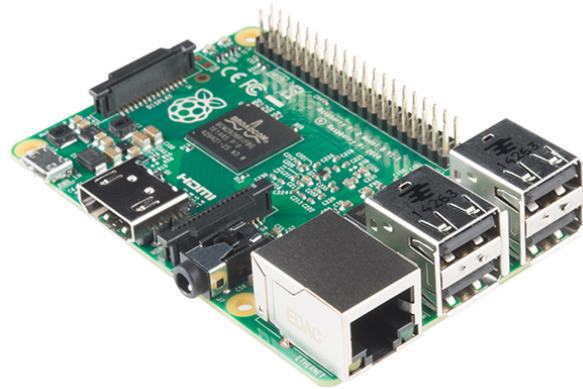
Raspberry Pi är en enkortsdator i storleken av ett kreditkort som utvecklas av det brittiska företaget Raspberry Pi Foundation. Raspberry:n har en ARM-baserad processor, vilket betyder att processorn bygger på liknande arkitektur som processorer i exempelvis mobiltelefoner eller surfplattor [54]. Detta innebär att inga "vanliga" operativsystem, exempelvis Windows, inte fungerar på en Raspberry Pi. Istället måste ett operativsystem som är byggt för att fungera enligt processorns arkitektur användas. För Raspberry Pi är det populäraste operativsystemet Raspbian/Linux [54].

Datorn har många användningsområden, exempelvis fungerar det till stor del som en "vanlig"<sup>1</sup> dator, vilket leder till att användaren kan använda den till att spela upp film, skriva dokument, programmera, lyssna på musik och liknande [7] [54]. Datorn har även möjlighet till anslutning av exempelvis sensorer, lampor och dylikt, därför kan datorn användas inom många olika projekt och områden. [7]

Fördelarna med en Raspberry Pi är många: pris, storlek, processorkraft, utvecklingsmöjligheter etc. Raspberry Pi kommer i olika versioner och storlekar, där skillnaderna på de olika versionerna är antal in-/utgångar samt hur kraftfull processorn är. Den mest högkvalitativa versionen heter Raspberry Pi modell 3, och har en processor med en klockfrekvens på, ungefär, 1 GHz [7], vilket kan jämföras med nutidens smartphones. Denna version har även en inbyggd WiFi-modul, vilket versionerna innan inte hade.

---

<sup>1</sup>Den typen av dator som oftast används av privatpersoner som bruksdator



**Figur 2.2:** Raspberry Pi 2. Bild licensierad under CCBY-NC-SA3.0[38]

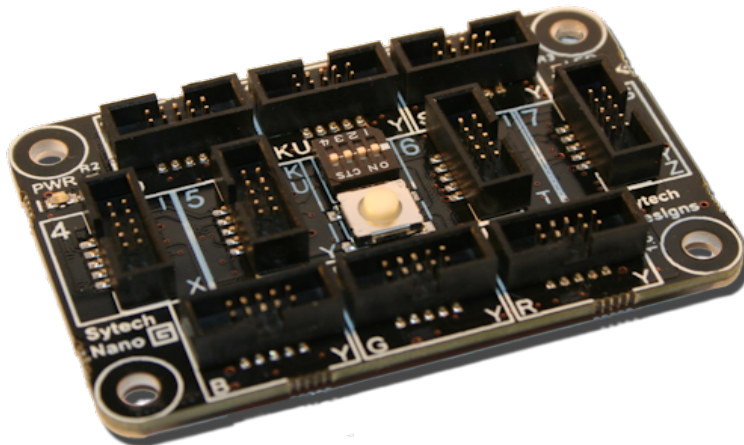
### 2.3.4 Microsoft Visual Studio

Microsoft Visual Studio är en utvecklingsplattform för ASP.NET webbapplikationer, XML Web Services, skrivbordsapplikationer, samt även mobilapplikationer med mera. [5]

I Visual Studio kan användare med olika programmeringskunskaper programmera tillsammans, tack vare att Visual Studio använder samma integrerade utvecklingsmiljö för alla programmeringsspråk. Programmeringsspråk som fungerar i Visual Studio är exempelvis JavaScript, C#, C++ samt Visual Basic. Vidare kan användaren även programmera med olika språk vid utveckling av ett projekt, vilket kan vara till stor fördel då utvecklingen av vissa applikationer kan underlättas med hjälp av flera språk. Dessutom binder Visual Studio dessa programmeringsspråk till .NET, vilket leder till att utveckling av ASP-webbapplikationer förenklas, detta på grund av att funktioner från .NET blir integrerat i språket. [5]

### 2.3.5 .NET Gadgeteer

Microsofts .NET Gadgeteer är ett open-source verktyg för att utveckla små elektriska anordningar [29], utvecklat av Microsoft [30]. Fördelarna med .NET Gadgeteer är bland annat att objektorienterad programmering kan användas, vilket förenklar vid flera typer av programmerings-lösningar. En annan stor fördel är att komponenterna som fungerar med Gadgeteer även är byggda på så vis att lödning inte är nödvändig, vilket många gånger förenklar samt gör utvecklingsprocessen snabbare [29].



**Figur 2.3:** Huvudkort från GHI Electronics, modell NANO [35]

Plattformen är uppbyggd av moduler som kan kopplas ihop med ett huvudkort (se figur 2.1) med hjälp av sladdar. Dessa moduler kan vara allt från skärmar och högtalare till accelerometrar eller internet-moduler [29] [30]. Enkel inkoppling av moduler, samt mjukvara som tillåter programmering i Visual Studio, där mjukvaran genererar hjälpande kod [30] leder till att .NET Gadgeteer med fördel kan användas vid utveckling av småskaliga elektroniska projekt.

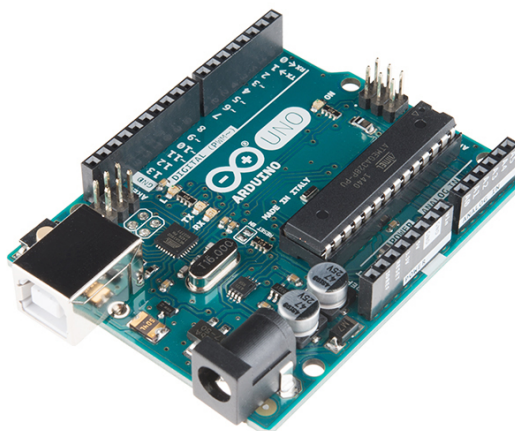
### 2.3.6 Arduino UNO ®

Arduino är en open-source prototypplattform baserad på vedertaget lättanvändlig hård- och mjukvara. En av fördelarna med Arduino jämfört med .NET Gadgeteer är att den är lättare att få tag på, då den säljs på många elektronikaffärer runt om i Sverige. Arduino programmeras i språket Arduino programming language, vilket bygger på Wiring i utvecklingsmiljön the Arduino Software (IDE). The Arduino Software (IDE) körs på Windows, OSX och Linux [6]. Arduino ses ofta som en bättre nybörjarplattform än .NET Gadgeteer och Raspberry Pi på grund av enkelheten i dess programmeringsmiljö. Arduino UNO® är en komplett mikrokontroller baserad på ATmega328P och har följande tekniska specifikationer:

**Tabell 2.1:** Tekniska specifikationer Arduino UNO®. [8]

|                            |                            |
|----------------------------|----------------------------|
| Microcontroller            | ATmega328P                 |
| Driftspänning              | 5V                         |
| Inspänning (rekommenderad) | 7-12V                      |
| Inspänning (gräns)         | 6-20V                      |
| Digitala I/O anslutningar  | 14 (Varav 6st stödjer PWM) |
| Analoga ingångar           | 6                          |
| DC per I/O anslutning      | 20 mA                      |
| DC för 3.3V anslutning     | 50 mA                      |
| Flashminne                 | 32 KB                      |
| SRAM                       | 2 KB                       |
| EEPROM                     | 1 KB                       |
| Klockfrekvens              | 16 MHz                     |
| Längd x Bredd              | 68.6 mm x 53.4 mm          |
| Vikt                       | 25 g                       |

På bilden nedan syns en Arduino Uno®, de svarta stiftlisterna som syns underlättar anslutning av komponenter då lödning ej krävs vid koppling av kablar eller moduler.



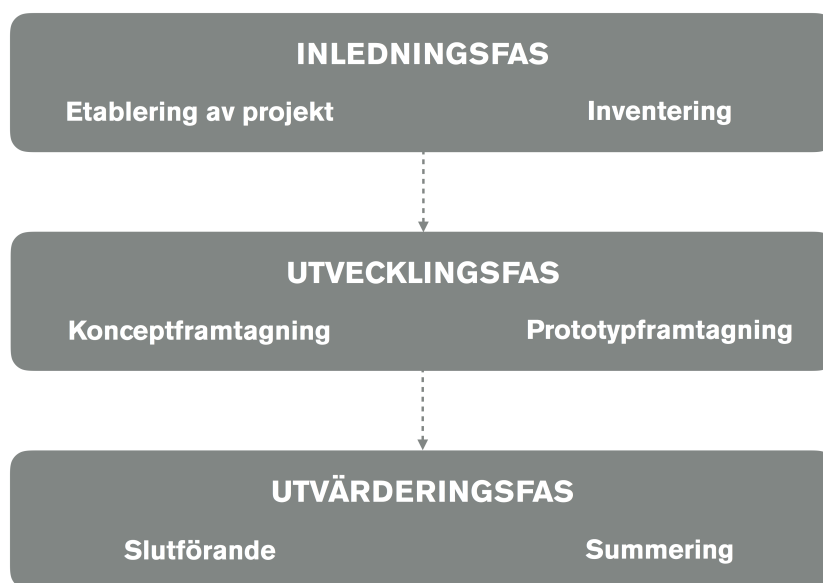
**Figur 2.4:** Arduino UNO®. Bild licensierad under CCBY-NC-SA3.0 [36]



# 3

## Metod

Projektet har genomförts i tre tidsbaserade faser där de olika områdena parallellt har bearbetats. Den första fasen var inledningsfasen där etablering av projektet genom generell struktur, datainsamling, utvärderingar och inventeringar genomfördes. Utvecklingsfasen, den andra fasen i ordning, syftade till att ta fram det generella konceptet. Den tredje och sista fasen, utvärderingsfasen, användes för att slutföra projektet och sammanställa resultat.



**Figur 3.1:** Bild över projektets faser

### 3.1 Inledningsfas

I inledningsfasen strukturerades projektet upp och genom en marknadsundersökning fastställdes vilken marknad konceptet kommer husera på, och vilka huvudsakliga konkurrenter som existerar. Sedan inleddes en klassisk produktutvecklingsfas som syftar till att ta fram en principiell arkitektur för projektet.

Vid klassisk produktutveckling genomförs arbetet i flertalet olika faser för att uppnå ett så tillfredsställande resultat som möjligt. Varje fas adderar ett värde i produktutvecklingsprocessen för att det slutgiltiga konceptet får högsta möjliga måluppfyllnad[21].

Produktutvecklingen följer ofta ett antal förutbestämda steg, med viss variation i användning av delmetoder och dylikt. Generellt sett inleds ett projekt med marknadsanalys och utvärdering av eventuellt kundbehov, för att därefter inleda en brainstorming-session[39]. Syftet med brainstorming-sessionen är att inleda konceptgenereringsarbetet så brett som möjligt för att inte utesluta några lösningar eller angränsande samt närbesläktade områden. Brainstorming är en process där nya idéer tas fram för att sedan kombineras med andra idéer utifrån vad kunden begär. Därefter ställs alla koncept upp i diverse olika matriser, bland annat Pugh-matriser, morfologiska matriser, funktionella modeller och jämförs med framtagna kravspecifikation för att kontrollera måluppfyllnad. Det är sällan en produkt utvecklas helt utan tidigare förankring på marknaden, utan oftast finns många liknande produkter som produktutvecklingsprocessen bygger vidare på i syfte att göra en befintlig produkt desto bättre.

Produktutvecklingen genomfördes till stor del i enlighet med produktutvecklingsmetoden The Value model[21], från boken med samma namn. Boken har använts som en handbok och beskrev utvecklingsprocessen, steg för steg, av en produkt eller tjänst. Metoden syftade till att skapa oöverträffat kundvärde och drev projektgruppen mot att ständigt optimera projektets uppfyllnad. Enligt boken bör fokuset för produktutveckling av en befintlig produkt alltid ligga på att skapa ett så högt, oöverträffbart, kundvärde som möjligt [21][39].

#### 3.1.1 Marknadsundersökning

För att utveckla ett så bra och konkurrenskraftigt koncept som möjligt krävdes en kartläggning och inventering av marknaden och befintliga koncept. Genomförandet av diverse marknadsundersökningar gav en tydlig bild av potentiella konkurrenter och en insikt om vilka utforskade innovativa lösningar som existerade. Patentundersökning och efterforskning av befintliga lösningar är båda delar som har genomförts i inventeringen.

Vidare var det av yttersta vikt att kartlägga vad som förväntades av intressenter, samt hur ett nytt koncept kunde tillfredsställa detta. För att få fram en tydlig bild av vad som ökar det potentiella kundvärdet och förväntningar på produkten krävdes en analys av vad en intressent ansåg vara bas- och kritiska krav, för att på så sätt styra projektet i rätt riktning[21].

### 3.1.2 Brainstorming

Brainstorming användes som ett verktyg för att generera nya idéer och koncept. Brainstormingen skulle vara ett öppet forum där idéer och koncept skapades utan någon analys av idéer då detta skulle genomföras senare [51] [50].

Vid brainstorming är det lämpligt med en förut bestämd tid som brainstorming-passet ska hållas under. Detta för att deltagarna inte ska låsas fast i sina idéer utan på ett sätt tvingas till att gå vidare och genererar nya idéer [51].

Det behövs en ledare för passet, ledaren fungerar som en moderator som samlar in alla idéer och låter alla komma till tals. En bra metod att använda sig av är lappar där idéer och koncept skrivs ner. Lapparna gör det lätt att senare analysera insamlad data och matcha idéer med varandra [51].

En av de viktigaste reglerna vid brainstorming är att kritik inte ska förekomma även om en idé kan verka helt galen är det kanske en version av just den som leder till det vinnande konceptet i slutändan. Att kombinera koncept med varandra görs efter avslutat pass då analyseras all data tillsammans för att få fram olika koncept som kan arbetas vidare med [51].

### 3.1.3 Referenslösning

Från en väl genomförd förundersökning har ett antal alternativa koncept radats upp, från vilka projektet tar avstamp. De olika koncepten graderas internt utifrån fördelar och nackdelar på viktiga funktioner innan ett koncept slutligen välades som referenslösning. Den valda lösningen blir sedan en kontinuerlig referens för projektets framfart, som användes för att visa framgångar och fördelar med ett nytt koncept.

I vissa fall väljs en färdig produkt redan från start som referens, då det kan vara ett visst företags koncept som skall förbättras.

### 3.1.4 Kundvärdesmatriser

Kundvärdesmatriser ställdes upp utifrån referenslösningarna i en tydlig struktur för att belysa kundbehov i en kolumn och dess prioritetsordning i en annan. I kundbehovskolumnen fanns uttalade basbehov och värdehöjare samt uttalade prestandakrav. I den andra kolumnen rankades alla behov i prioritetsordning baserat på marknadsundersökningen. Kundvärdesmatrisernas primära syfte var att göra en enkel inventering av de behov som fanns och behövde inte nödvändigtvis ha mätvärden på de olika kraven. Kundvärdesmatriserna låg som grund för de funktionella modellerna samt för kravspecifikationerna.

### 3.1.5 Funktionella modeller

När referenslösningarna och kundvärdesmatriserna var fastställda krävdes en identifiering av de primära, och därmed viktigaste, funktionerna. Dessa ställdes upp i en funktionell modell som i sin enkelhet demonstrerade vad som skedde i processen, från in- till utsignal. Vitsen med de funktionella modellerna var att de skalar ner de komplexa lösningarna till enklare modeller som illustrerar vilka förväntningar som fanns på systemet.

### 3.1.6 Kravspecifikationer

För att på ett så strukturerat och tydligt sätt som möjligt lyfta fram vilken prestanda som var av största vikt för konceptet togs kravspecifikationer fram. Kravspecifikationerna erbjöd projektet en tydlig ram att hållas inom, och byggdes på följande funktioner:

**Huvudfunktioner**, vilka syftade till vad koncepten i huvudsak skulle göra, exempelvis att en bil syftar till att transportera människor.

**Tilläggsfunktioner**, vilka syftade till de funktioner som höjde kundvärdet och särskilde koncepten från andra liknande koncept.

**Stödfunktioner**, vilka syftade till de funktioner i koncepten som krävdes och förväntades av en kund, men som inte höjde kundvärdet.

**Oönskade funktioner**, vilka syftade till de funktioner som inte gick att eliminera helt ur konceptet, och som istället behövde regleras med vissa krav. Från exemplet med bilen kategoriseras utsläpp av avgaser här.

Specifikationerna förhåller sig till de funktionella modellerna över systemen, vilka vid lyckad nedskalning, enbart belyste de krav som ställts på konceptens helhetslösningar. Kravspecifikationerna byggdes upp utifrån krav och diverse viktade önskemål på koncepten eller produkterna, där allt skulle vara mätbart för att på så sätt säkerställa uppfyllnaden av dessa.

## 3.2 Utvecklingsfas

Den andra fasen i projektet benämndes utvecklingsfasen, i vilken de planerade koncepten med bestämda krav från inledningsfasen skulle utvecklas.

### 3.2.1 Utveckling av mjukvara för datainsamling, visualisering samt analys

En metod som frekvent används vid programmering inom områden där informationen är bristfällig eller inte tillgodoser det specifika projektets ändamål är trial and error-metoden. Metoden baseras på att användaren testar olika lösningar och teorier för att uppnå ett tillfredsställande resultat [57]. Metoden saknar struktur av definitionsskäl, men kan leverera bra resultat om användaren besitter tillräcklig kunskap samt har tillgång till teori om liknande eller närbesläktade områden.

Då projektet till stor del byggde på diverse tester och ständiga praktiska förändringar i konceptet var trial and error-metoden mycket användbar eftersom att den tillät gruppen att göra förändringar i olika delar samtidigt på en testbasis. Projektet syftade inte till att hitta den bästa lösningen utan snarare till att hitta *en* lösning, vilket gjorde trial and error, som bygger på itereringar för att uppnå ett resultat, stämde överens med syftet.

Ett annat välanvänt verktyg vid projektets programmering var funktionella modeller, se kapitel 3.1.5. De funktionella modellerna gav en översiktsbild över hur koden skulle fungera, vilket underlättade vid utvecklandet av den.

### 3.2.2 Utveckling av design för inkapsling av moderkort

För att nå de krav och önskemål som ställdes på konceptens design i inledningsfasen utvecklades modeller av koncepten i ett CAD-program. När modeller på koncepten var framtagna tillverkades prototyper med hjälp av 3D-skrivare och enkla verktyg.

### 3.3 Utvärderingsfas

Utvärderingsfasen syftade till att validera huruvida de utvecklade systemen kunde betraktas som klara genom att de utvärderades mot de krav som ställts på dem i kravspecifikationerna. För att utvärderingen skulle genomföras så korrekt som möjligt testades de båda systemen mot två typer av vibrationsbänkar istället för enbart en, där målet var att jämföra hur det utvecklade systemet ställde sig mot den utvecklade referenslösningen. Att samla in vibrationsdata från fler källor leder till att en analys av högre säkerhet kan genomföras.

Vid utvärdering och jämförelse av resultat för olika system som ska samla in liknande data krävs ofta mer än en valideringsmetod. Detta för att säkerställa att de uppmätta resultaten inte slumpartat blir de samma, utan att de är konstanta oavsett valideringsmetod. Därför var utvärderingsfasen samlad kring framtagandet av ytterligare en testsystem med möjlighet att genomföra andra tester, i form av mer verklighetstroga test av kullager.

#### 3.3.1 Utvärdering av mätsystemen

För utvärderingen av de två mätsystemens funktion skulle ett nytt testsystem utvecklas för att fler tester och jämförelser skulle kunna genomföras. Som nämnt i stycket ovan leder fler tester till en mer trovärdig analys, vilket är bakgrunden till att det nya testsystemet skulle utvecklas. Med hjälp av de två testsystemen skulle empiriska studier kunna genomföras för att samla in tillräcklig data för att ge ett bra underlag för att jämföra och analysera de två mätsystemen. I utvärderingsfasen var det tänkt att analyser på den insamlade datan skulle utföras för att få en bild över vilket system som presterar bäst i form av mätsäkerhet, mät hastighet samt användarvänlighet. Det nya testsystemet skulle möjliggöra tester på kullager för att få fram trovärdiga vibrationer, till skillnad från det tidigare testsystemet vilket endast genererade vibrationer med hjälp av en vibrerande platta. Vibrationssensorerna testades även på högtalare och roterande maskiner för att samla in data och jämföra de olika systemen.

# 4

## Inledningsfas

Under projektets första fas inleddes arbetet med att dela in arbetet i fem övergripande delar: Inventering av kända koncept, utveckling av det tidigare systemet, utveckling av ett nytt system, utveckling av en testsystem samt utveckling av en webbsida. Dessa har systematiskt bearbetats parallellt under projektets fortskridning. En stor del av arbetet under inledningsfasen har varit att kartlägga det tidigare konceptet, fastställa referenslösningar och sammanställa kravspecifikationer för prototyper såväl som för de olika koncepten i sin helhet.

### 4.1 Inventering av kända koncept

#### 4.1.1 Marknadsundersökning

Innan en potentiell generering av möjliga koncept och helhetslösningar kunde göras genomfördes en marknadsundersökning för att konstatera hur dagens situation ser ut: vilka funktioner som ökar kundvärdet och vad som förväntas av ett helhetskoncept. De produkter som idag finns på marknaden kan delas in i två undergrupper: Handhållna instrument samt instrument som placeras permanent. De handhållna instrumenten används i ett omfång där de med givna, förutbestämda, intervaller ger indikation av maskinens hälsa. De är mer mobila och enkla, men en stor nackdel ligger i att operatören själv måste avsätta tid för att genomföra en statuskontroll. Typen av underhåll placeras någonstans mellan planerat och konditionsbaserat underhåll.

Den andra undergruppen, permanent placerade instrument, är aktiva hela tiden för att mer kontinuerligt informera operatören om maskinens aktuella status och underhållstypen det gäller är alltså konditionsbaserat underhåll. Det är särskilt de instrument som mäter kontinuerligt som är av intresse för projektet, även om mobiliteten i den handhållna instrumenten kan medföra inspiration. Nedan följer sammanfattande information om några av de största konkurrenterna av liknande produkter och system som finns på marknaden.

### **FAG Smartcheck**

FAG Smartcheck är ett system som innefattar många av de finesser som finns i visionen om detta projekt. Systemet är för det första billigt i relation till de andra konkurrerande systemen som finns på marknaden idag [49]. FAG Smartcheck är ett litet <sup>1</sup> system som monteras på maskiner med roterande lager. Den insamlade och analyserade datan kan sedan visas på exempelvis webbläsaren i en telefon. Användaren får då en enkel översiktsbild över hur maskinen som analyseras mår [49]. Det som gör att FAG Smartcheck sticker ut från de andra systemen på marknaden är att systemet lär sig medan det mäter, med andra ord så förstår systemet hur maskinen vibrerar under normala förhållanden, och därför kan systemet larma när vibrationerna blir onormala [49].

### **SPM Instrument Intelnova**

Ett online-system för tillståndskontroll och diagnostisering kapabelt till att handhålla upp till 32 stycken sensorer som utför kontinuerlig mätning som sedan skickas och analyseras i en kontrollenhet. Det färdigtolkade resultatet ger sedan en god bas att ta beslut angående om en maskin kräver underhåll eller ej. Systemet finns i olika utförande beroende på vilka krav som ställs på det [41].

### **SPM Instrument MG-4**

MG-4 är till för övervakning av maskiner av roterande typ. Den består i grunden av en mätenhet och tillhörande givare och monteras för kontinuerlig bruk. Det finns fyra olika programmerbara mätenheter som kan användas beroende på behov. Där skillnaden ligger i hur många och av vilken typ kanal enheten tillhandahåller avseende teknikerna vibrationstal och stötpulsmätning. Systemets inkapsling har IP-klassning 65 [42].

### **Delphins ExpertVibro**

ExpertVibro är en enhet för att transistenter och vibrationer. Den är kapabel till att mäta upp till 16 separata inputs och tillhandahålla en samplingfrekvens på 50 kHz. Mätdata övervakas sedan i realtid och skulle vissa givna värden överskridas har systemet sedan upp till åtta digitala utgångar som kan aktiveras [43].

### **Kimo Bearing Checker**

Bearing Checker är ett handhållet mätinstrument i fickformat. Det är kapabelt till att ge information om maskinstatus för flera olika typer av roterande maskiner. Instrumentet mäter både värme och stötpulssignaler för att ge en uppskattning om maskinens aktuella lagerstatus [44].

Marknadsundersökningen gav fem tydliga exempel på färdiga produkter, med olika styrkor och svagheter. Kvalitéerna hos koncepten varierar inom olika områden, varför projektet valde att lyfta fram just dessa. För fastställande av referenslösning och för att formulera så bra krav och önskemål som möjligt i kravspecifikationen är dessa en källa för inspiration.

---

<sup>1</sup>Mätanordningen är 44 x 57 x 55 mm

### 4.1.2 Summering av marknadsundersökningen

Efter att marknadsundersökningen hade genomförts hade gruppen en tydlig bild över hur ett välfungerande system borde specificeras, både utifrån prestanda och design. Marknadsundersökningen gav även gruppen tydliga indikationer av vilka krav som borde sättas på mät hastighet, frekvensregister och användarvänlighet. De krav som togs fram i marknadsundersökningen kommer att presenteras tydligare i form av kravspecifikationen i senare kapitel.

### 4.1.3 Referenslösning för det nya systemet

Projektet syftar till att skapa ett nytt mätsystem. Därav krävs en referenslösning som systemet kontinuerligt kan utvärderas mot.

Eftersom att projektet tydligt angränsar till, och även bygger på, förra årets kandidatarbete [22] fanns redan vid projektstart ett naturligt val av referenslösning. Den mätplattform kandidatarbetet "Prototyp för konditionsbaserat underhåll med enkla sensorer" tog fram kunde, enligt rapportens diskussion, framgångsrikt användas för att samla in nödvändig accelerationsdata för analys. Mätplattformen har rent principiellt kunnat analysera inhämtad data av tillräckligt hög kvalitet för att genomföra analyser, men har begränsats av en relativt låg sampelhastighet. Befintliga produkter på marknaden arbetar med ett frekvensregister upp till 10000 Hz, men den begränsade sampelhastigheten medför att det inte går att mäta frekvenser över 420 Hz med systemet som förra projektgruppen utvecklade.

Då konceptet principiellt har lyckats, samtidigt som en del förbättringspotential finns, har projektgruppen valt att sätta detta som referenslösning när det kommer till mätanordningen. Dock kommer alla system som undersöktes i marknadsundersökningen, särskilt FAG Smartcheck, ligga till grund för utvecklingen. Att FAG Smartcheck-systemet särskilt lyfts fram är på grund av dess pris, vilket är lågt i relation till övriga system från marknadsundersökningen. För att det tidigare systemet skulle fungera som referenslösning krävs utveckling mot samma syfte som det nya systemet. På så sätt blev det en referenslösning som ständigt är giltig.

## 4.2 Vidareutveckling av tidigare system

Som sagt i kapitel 4.1.2 var utveckling av det tidigare mätsystemet nödvändigt för att det skulle kunna användas som en giltig referenslösning till det nya system som senare skall utvecklas. Det tidigare mätsystemet som utvecklades i kandidatarbetet ”Prototyp för konditionsbaserat underhåll med enkla sensorer” bestod av ett huvudkort av typ FEZ Raptor från GHI Electronics[22]. Systemet var redan ett färdigt principkoncept vilket innebar att en kundvärdesmatris och en kravspecifikation blev överflödigt då ramarna för det redan var satta. Alltså fortgick utvecklingen av konceptet vidare och en ny inkapsling skulle tas fram samt möjliggörandes av kommunikation mellan systemet och en databas.

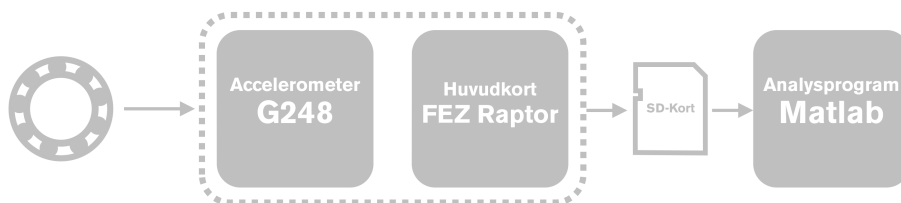
### 4.2.1 Utveckling mot kontakt med server

Syftet med detta projekt var att skapa en lösning som på ett billigt och enkelt sätt avläser samt visualiserar vibrationsdatan för användaren. Som tidigare konstaterats var systemet som utvecklades av förra årets projektgrupp inte anpassat för att möjliggöra kontakt med en server för hantering av avläst vibrationsdata [22]. Att systemet skulle ha kontakt med en server såg projektgruppen som ett krav för att syftet i projektet skulle uppfyllas.

Tack vare att systemet som skulle utvecklas byggde på grundplattformen FEZ Raptor valdes Visual Studio som utvecklingsmiljö och C# som programmeringsspråk. Dessa val stöttades av flera faktorer, där de viktigaste för projektet var:

- Möjlighet till användandet av gammal kod
- Förkunskaper i C#
- Handledarens kunskap i Visual Studio

För att förstå hur det tidigare systemet utförde sin datainsamling samt analys byggdes ett flödesschema upp. Ett flödesschema ger en överskådlig bild på funktionen av koden, utan att kunskap angående exekveringsordningen är essentiell [52]. Flödesschemat kommer vid senare utveckling även användas för att analysera och förbättra mjukvaran om det blir nödvändigt.



**Figur 4.1:** Flödesschema över det tidigare systemets funktion.

För att möjliggöra kontakt mellan mätsystemet och en webbsida/server krävs främst två specifika tekniska lösningar:

- Internetanslutning för mätsystemet.
- Kommunikation mellan mätsystem och databas

### 4.2.2 Utveckling av inkapslande chassi

Vid utvecklandet av den låda som ska innesluta moderkortet till FEZ-raptorn skapades en kravspecifikation. Detta gjordes för att på ett tydligt sätt sätta ett värde på de krav och önskemål en slutgiltig produkt i industriell skala kommer behöva uppfylla för att kunna tillfredsställa den potentiella kundens behov.

| Kravspecifikation inkapslande låda                   |                            |               |                              |
|------------------------------------------------------|----------------------------|---------------|------------------------------|
|                                                      | Målvärde                   | Krav/Önskemål | Verifiering                  |
| <b>1. Huvudfunktioner</b>                            |                            |               |                              |
| 1.1 Innesluta moderkort från smuts och fukt          | IP-Klass 54                | K             | Test och design mot IP-klass |
| <b>2. Tilläggfunktioner</b>                          |                            |               |                              |
| 2.1 Möjlighet att se värden från inkopplade enheter  | Ha synlig display          | K             | Visuellt test                |
| 2.2 Avtagbart lock                                   | < 3 minuter                | K             | Test                         |
| 2.3 Klara av att befinna sig i industrimiljö         | IP-Klass 54                | Ö4            | Design mot IP-klass          |
| 2.4 Skydda mot stötar                                | Klara fall från <1.5 meter | Ö3            | Test                         |
| <b>3. Stödfunktioner</b>                             |                            |               |                              |
| 3.1 Möjlighet att koppla in n antal enheter          | Ha n antal hål             | K             | Design                       |
| 3.2 Möjlighet till rengöring/uppdatering av hårdvara | Avtagbart lock             | K             | Design                       |

**Figur 4.2:** Kravspecifikation av moderkortshölje

Som syns i kravspecifikationen är huvudfunktionen av ett hölje runt moderkortet att avvisa damm och fukt. Detta för att ett moderkort är en elektrisk krets som kan ta skada av att just sådana substanser kommer åt kretsen. Vidare för att ha ett tydligt mätvärde på detta valdes IP-klassificering 54. 5 betyder att den skall vara skyddande mot damm i skadliga mängder för att det inte skall kunna ta sig in på elektriska komponenter så att isolationsfel kan uppstå. Armaturen måste vara tät och packningar är därför nödvändigt [48]. 4 står för vattenavstötande funktion, den anger att komponenten skall avstöta rinnande vatten från alla vinklar. Verifiering av IP-klassificeringen görs genom tester och strävan efter att uppfylla kraven redan i designen av produkten. När IP-klassificeringen gjorts har andra liknande produkter tagits i beaktande för att få en så bra referens som möjligt.

Det är ett krav att moderkortet skall kunna tas ut ur höljet för eventuellt underhåll eller uppdatering av hårdvaran. Därför följer kravet av att locket skall vara avtagbart. Det leder till att konstruktionen mot IP-klassificeringen blir mer krävande men det skall inte förbi ses då fokus ligger på att skapa en så bra produkt för kunden som möjligt. Det mätbara kravet på denna funktion ligger på att det skall gå att demontera locket och ta ut moderkortet på under tre minuter för att inte slösa värdefull tid på en icke värdeskapande process.

För att kunna få en uppdaterad bild av vad som händer är det ett krav att den display som är kopplad till moderkortet kan visa sina värden under pågående process. Därför tas detta med som ett designkrav på produkten.

För att ha en flexibel och funktionell ”låda” är det viktigt att det finns möjlighet att koppla in de sensorer och Ethernet-kablar som behövs för att moderkortet skall kunna utföra sitt jobb. Därför är kravet att tre stycket enheter skall genom att håll måste tas i med i designen av produkten.

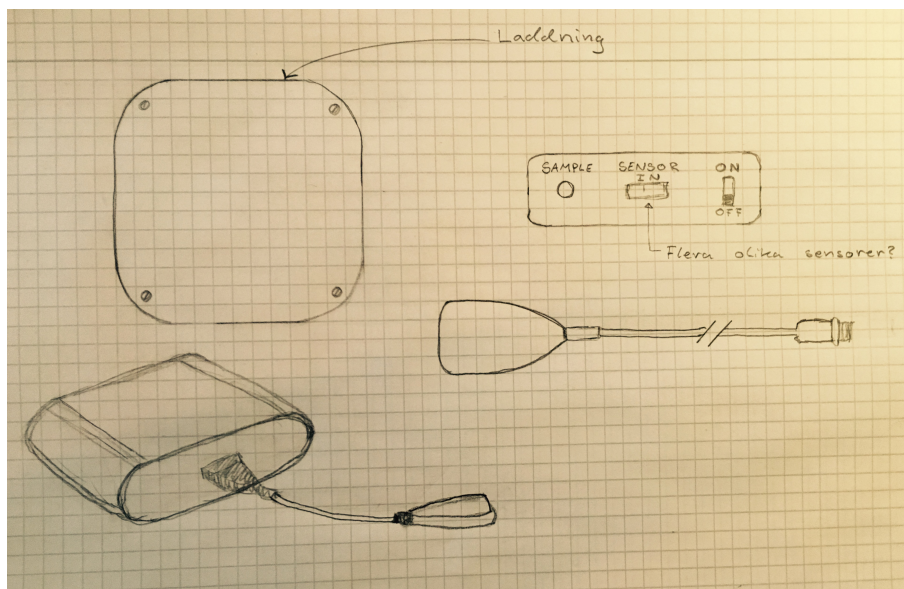
Ingen oönskad funktion uppstår i systemet över lådan, då prototypens huvudsakliga funktioner, att skydda moderkortet, inte medför någon negativ inverkan för lösningen.

### 4.3 Utveckling av ett nytt system

För att utvärdera hur väl systemet som bygger grundplattformen FEZ-Raptor presterar, bestämde gruppen att utveckla ett nytt system, med en annan typ av hårdvara. För projektets syfte var det inte enbart systemets prestanda och pris som var intressant, även enkelheten vid programmeringen var intressant att analysera, då en av projektets grundpelare är att det skall vara enkelt.

#### 4.3.1 Brainstorming

Under Brainstormingen kom gruppens medlemmar med flera olika idéer på lösningar. Den konceptlösning som hade störst potential att uppfylla de krav som senare skulle ställas på systemet lyftes fram för vidare utveckling. En skiss på konceptet som skulle vidareutvecklas kan ses nedan:



**Figur 4.3:** Exempelskiss vid brainstorming

På skissen syns ett koncept där vikten har lagts vid att systemet skall vara portabelt och lättanvänt, något som gruppen såg som viktigt efter marknadsundersökningen i kapitel 4.1.1.

### 4.3.2 Kundvärdesmatris samt kravspecifikation

En kundvärdesmatris med uttalade krav samt önskemål ställdes upp utifrån den funktion som var eftertraktad ur projektsyftets-synvinkel. Kraven och önskemålen är formulerade ur projektets definition samt från marknadsundersökningen och referenslösningen som har tagits fram.

| Kundbehov                        | Prioritetsordning |
|----------------------------------|-------------------|
| <b>1. Uttalade basbehov</b>      |                   |
| 1.1 Mäta vibrationer             | 1                 |
| 1.2 Analysera vibrationsdata     | 1                 |
| <b>2. Uttalade värdehöjare</b>   |                   |
| 2.1 Portabel                     | 3                 |
| 2.2 Modulär                      | 4                 |
| <b>3. Uttalade prestandakrav</b> |                   |
| 3.1 Visualisera vibrationsdata   | 1                 |

**Figur 4.4:** Kundvärdesmatrisen

De funktioner som kategoriserats och synliggjorts i kundvärdesmatrisen har inga målvärden för bedömning, och därmed var det nödvändigt att en kravspecifikation utformades. Vilka målvärden som ställs på de olika huvud-, tilläggs-, stöd- samt önskade funktionerna följer ur marknadsundersökningen som utfördes i kapitel 4.1.1 och från referenslösningen som bestämdes i kapitel 4.1.2.

För det nya mätsystemet sammanställdes åtta stycken krav och önskemål och bröts ner i de fyra ovan nämnda kategorierna. Det blev snabbt tydligt att principlösningen inte skulle medföra några önskade funktioner som behövde elimineras, utan snarare att mängden tilläggfunktioner som potentiellt kunde öka kundvärdet var många.

| Kravspecifikation för det nya mätsystemet |                                           |               |               |
|-------------------------------------------|-------------------------------------------|---------------|---------------|
|                                           | Målvärde                                  | Krav/Önskemål | Verifiering   |
| <b>1. Huvudfunktioner</b>                 |                                           |               |               |
| 1.1 Mäta vibrationer                      | 50-240[Hz]                                | K             | Mätning       |
| 1.2 Visualisera vibrationsdata            | Visa upp datan inom 2 sekunder [s]        | K             | Visuellt test |
| <b>2. Tilläggfunktioner</b>               |                                           |               |               |
| 2.1 Portabel                              | Ingen yttre permanent strömkälla.         | K             | Konstruktion  |
| 2.2 Kommunikation                         | 400 datapunkter per sekund                | K             | Mätning       |
| 2.3 Modulär                               | Möjlighet att byta sensor till en likadan | Ö5            | Konstruktion  |
| 2.4 Storlek                               | Max 200x200 [mm]                          | Ö4            | Konstruktion  |
| 2.5 Batteritid                            | 2 timmar normal driftanvändning           | Ö4            | Mätning/Test  |
| <b>3. Stödfunktioner</b>                  |                                           |               |               |
| 3.1 Analysera vibrationsdata              | 400 datapunkter per sekund                | K             | Mätning       |
| <b>4. Önskade funktioner</b>              |                                           |               |               |

**Figur 4.5:** Kravspecifikation för det nya mätsystemet.

Kategorin huvudfunktioner innehöll två kriterier som var de principiellt viktigaste gällande valt koncept. Själva vibrationsmätningen och visualiseringen av insamlad data ansågs vara de två funktioner som projektet syftade till att genomföra. De målvärden som satts på vibrationsmätningen, att mäta frekvenser som ligger mellan

50-420 Hz, är ett absolut lägsta krav som baseras på de värden förra årets prototyp [22] lyckades mäta. Gruppen ansåg att det lägsta acceptabla resultatet var att komma upp i samma frekvensområde som projektet förra året kom upp i, men att målet var att överträffa det resultatet.

De tilläggsfunktioner som projektgruppen har tagit fram, utifrån marknadsanalysen i kapitel 4.1.1, är generellt sett fokuserade kring området att göra konceptet mobilt och portabelt. Detta ska uppnås genom att kräva att ingen yttre strömkälla behövs, begränsa storleken och maximera batteritiden för enheten. I övrigt kräver systemet en kommunikation på 400 datapunkter per sekund för den kontinuerliga bevakningen, och att genom möjliggöra sensorbyte göra konceptet modulärt.

För att möjliggöra den krävda visualiseringen identifieras stödfunktionen analys av 400 datapunkter i sekunden. Stödfunktionen är inget som tillför något som helst värde för konceptet men som krävs, och förväntas, för att huvudfunktionerna skall kunna genomföras.

### 4.3.3 Val av hårdvara för datainsamling

#### 4.3.3.1 Val av moderkort

Bestämt enligt beskrivningen av projektet var det redan från början bestämt att det moderkort som skulle användas var enkortsdatorn Raspberry Pi. Raspberry Pi finns i fyra olika modeller, där det mest särskiljande var storleken på dessa. Vilken modell som används kommer inte att påverka projektets syfte, och därav specificeras inte heller det för att tillåta gruppen flexibilitet i utvecklingsarbetet.

#### 4.3.3.2 Val av accelerometer

Accelerometern är den enhet som kommer vara i direkt kontakt med det objekt som analyseras och det är denna enhet som känner av vibrationerna. Vid valet av accelerometer så var det fyra egenskaper som efter marknadsundersökningen i kapitel 4.1.1 var av största vikt, nämligen:

- Mäthastighet, antal mätvärden per sekund [Hz]
- Mätområde [G]
- Kostnad [SEK]
- Monteringsmöjligheter

Av dessa egenskaper sågs mäthastigheten samt kostnaden som de egenskaper som var av högsta vikt. En kravspecifikation sattes upp för att utvärdering av accelerometer skulle vara möjligt.

| Kravspecifikation accelerometer                       |                   |               |                             |
|-------------------------------------------------------|-------------------|---------------|-----------------------------|
|                                                       | Målvärde          | Krav/Önskemål | Verifiering                 |
| <b>1. Huvudfunktioner</b>                             |                   |               |                             |
| 1.1 Avkänna vibrationer genom mätning av acceleration | Upp till minst 8G | K             | Enligt datablad             |
| <b>2. Tilläggfunktioner</b>                           |                   |               |                             |
| 2.1 Frekvensområde                                    | > 800Hz           | K             | Enligt datablad             |
| 2.2 Upplösning                                        | > 10bit           | K             | Enligt datablad             |
| 2.3 Drivspänning                                      | Stöd för 3.3V     | K             | Enligt datablad             |
| <b>3. Stödfunktioner</b>                              |                   |               |                             |
| 3.1 I2C bus                                           | Ja                | K             | Enligt datablad             |
| 3.2 Kostnad                                           | < 300SEK          | Ö             | Uppgift från återförsäljare |

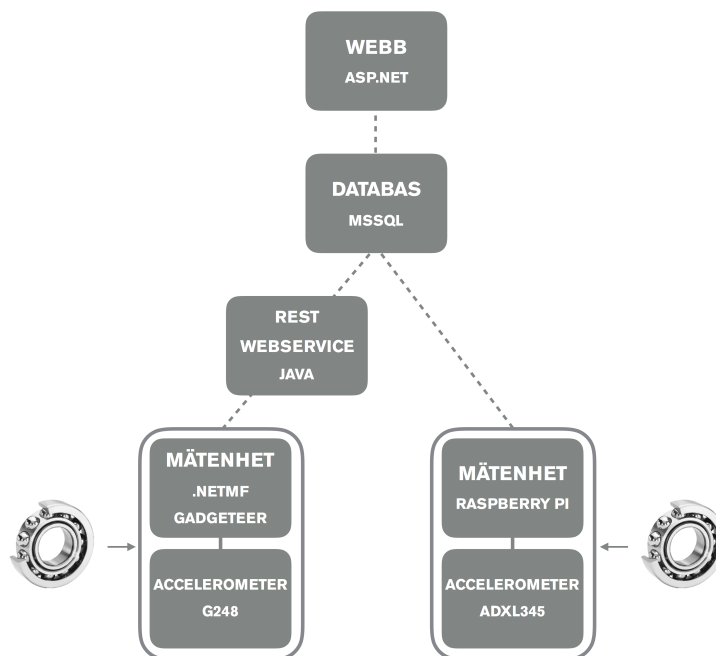
**Figur 4.6:** Kravspecifikation för accelerometer.

Efter litteraturstudier i datablad för accelerometrar och noga övervägande togs ett beslut att samma accelerometer de använde i "Prototyp för konditionsbaserat underhåll med enkla sensorer" [22] året skulle användas, om möjligt. Accelerometern i fråga, Accel G248 Module, används till att börja med, förutsatt att den uppfyller visas grundläggande krav. De krav som ställdes på mätutrustningen i kravspecifikationen ansågs vara ett bra mått. Accelerometern uppfyllde dessa och projektgruppen ansåg att användning av samma accelerometer på båda mätsystemen skulle leda till en

mer rättvis bedömning av systemen. Om det skulle visa sig att installation av Accel G248 inte var möjlig skulle istället en accelerometer med benämningen ADXL345 användas, detta på grund av att denna accelerometer är, enligt utvärdering med hjälp av kravspecifikationen den mest lämpliga för ändamålet.

### 4.3.4 Funktionell modell

För att få en bra bild över funktionen av systemet sattes en funktionell modell upp. Denna modell ger en bild över vilka funktioner som skulle behövas för att systemet skulle uppfylla projektets syfte. Det nya systemet består av Raspberry Pi-datorn.



**Figur 4.7:** Funktionell modell över systemet

Den funktionella modellen ger en övergripande bild över både det nya och tidigare systemet, se Figur 4.7. Modellen visualiserar hur vibrationsdata samlas in från två olika mätenheter (FEZ-rapor och Raspberry Pi), oberoende av varandra, och samlar denna i ett första steg. För enheten FEZ-rapor skickas sedan datan genom ett mellansteg i form av en REST-server, något som raspberry:n inte gör, mer om detta senare. Till slut skickas samt sammanställs datan i en databas, för att sedan kunna visualiseras på en webbsida.

## 4.4 Utveckling av webbsida för visualisering av mätdata

För att kunna visualisera mätdata över internet behövde en webbsida utvecklas. Utvecklingsmiljön Visual Studio har ett bra och enkelt stöd för integration med Microsoft SQL Databas och denna utvecklingsmiljön valdes för utveckling av webbsidan. Med hjälp av ASP.NET är det relativt enkelt att integrera en anslutning till Microsoft SQL Databas för att visa data med en hög uppdateringsfrekvens, utvecklingsmiljön möjliggör även enkla sätt att skapa grafer och diagram över datan. En kravspecifikation med de viktigaste funktionerna listade sattes upp för att underlätta utvecklandet av webbsidan.

| Kravspecifikation hemsida                            |                                                      |               |                                 |
|------------------------------------------------------|------------------------------------------------------|---------------|---------------------------------|
|                                                      | Målvärde                                             | Krav/Önskemål | Verifiering                     |
| <b>1. Huvudfunktioner</b>                            |                                                      |               |                                 |
| 1.1 Enkelt visualisera mättata                       | Lättillgängliga grafer                               | K             | Visuellt test                   |
| <b>2. Tilläggsfunktioner</b>                         |                                                      |               |                                 |
| 2.1 Enkel navigation mellan olika mätserier          | Drop-down meny eller liknande                        | K             | Visuellt- och funktionellt test |
| 2.2 Automatisk uppdatering av graf för nya mätserier | Automatisk refresh av hemsida.                       | Ö3            | Visuellt- och funktionellt test |
| 2.3 Möjlighet att exportera mätserier                | Export av serier av typen excel, CSV eller liknande. | Ö1            | Visuellt- och funktionellt test |

**Figur 4.8:** Kravspecifikation för webbsidan.

### 4.5 Reflektion av inledningsfasen

Upplägget att från projektets start utforma väldefinierade och genomförbara mål visade sig genom projektets gång ha hjälpt fortskridandet av arbetet avsevärt. Vid projektets avslut visade det sig dock att mer tid skulle behöva vara avlagt för utvärderingsfasen, då mer analyser och tester av konceptet skulle vara nödvändigt för att projektet skulle lyckas fullt ut. Om en mer omfattande marknadsundersökning, där det inte enbart fokuserats på olika varianter av konditionsbaserat underhåll, utan även på produkter och industrier som angränsar eller helt enkelt har bra lösningar på handhållna system genomförts, hade möjligheten funnits att utveckla ett ännu bättre system. En potentiellt bredare marknadsundersökning bör även ha toppats med företagsbesök för att på så sätt få en inblick i industrier och kartlägga problem som annars lätt kan bli förbisedda.

De kravspecifikationer och mål som satts upp i inledningsfasen har varit baserade till stor del på föregående kandidatarbete [22], och en potentiellt mer nytänkande lösning kunde möjligtvis ha tagits fram vid fastställande av annan referenslösning, exempelvis en färdig produkt såsom FAG Smartcheck [49]. Detta hade ställt högre krav på konceptet, men även erbjudit möjligheten för gruppen att utveckla det längre.

Kravspecifikationen över webbsidan hade kunnat utformas mer ingenjörsmässigt genom att verifieringsmetoderna för de olika krav och önskemålen inte enbart skulle vara visuella samt funktionella test. Anledningen till att införa andra verifieringsmetoder är att visuella och funktionella test till stor del är subjektiva bedömningar vilka är svåra att ställa mätbara krav på.

# 5

## Utvecklingsfas

Inledningsfasen sammanställde de krav som ställdes på de olika systemen och synliggjorde dessa i tydliga kravspecifikationer. Det välstrukturerade inledningsarbetet ledde till att utvecklingsarbetet med enkelhet kunde fortgå, och utvecklingen av det nya såväl som det tidigare systemet hade tydliga mål.

### 5.1 Utveckling av det tidigare systemet

Vid utveckling av det tidigare systemet var målet att förbättra det system som fanns från föregående kandidatarbete [22] till det bättre. De punkter som valdes att utveckla var framförallt mjukvaran som driver systemet och designen på den låda som är till för att skydda moderkortet.

#### 5.1.1 Mjukvara

De två tekniska punkter som nämnts i 4.2.1, möjliggöra internetanslutning för mät-systemet samt möjliggöra kommunikation mellan mätsystemet och databas är de viktigaste och mest centrala delarna för utvecklingen av systemet.

##### 5.1.1.1 Internetanslutning för mätsystemet

Då FEZ Raptor är baserat på .NET Gadgeteer/.NETMF som är ett open-source system så finns många färdiga moduler att köpa från tillverkaren GHI Electronics. I detta projektet används Ethernet-modulen 'EthernetENC28J60', till modulerna finns även färdiga klasser i .NET Gadgeteer/.NETMF innehållande metoder som underlättar programmeringen. Följande kod initierar nätverksanslutning för systemet:

```
private void initNetwork(){
    ethernet.NetworkInterface.EnableDhcp();
    ethernet.NetworkInterface.Open();
}
```

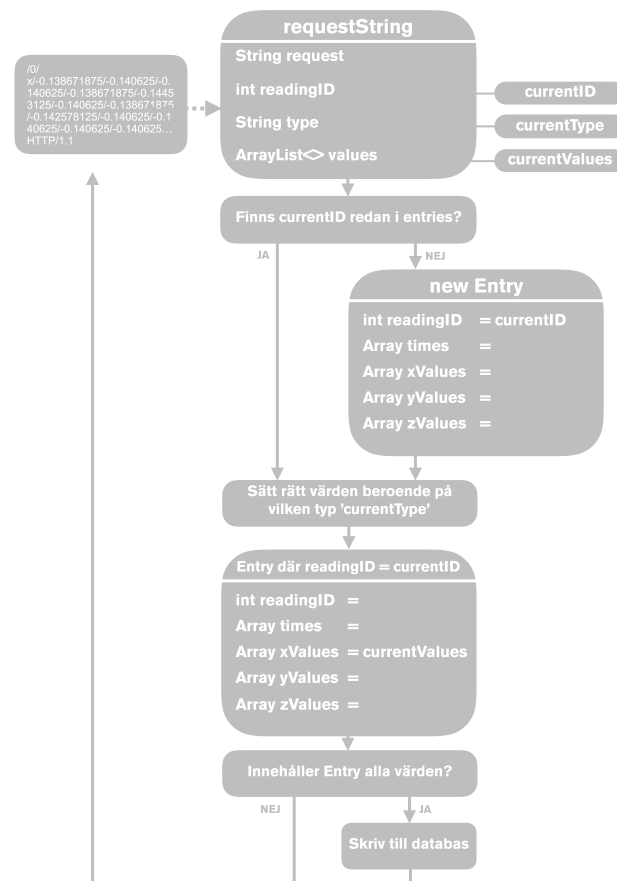
### 5.1.1.2 Kommunikation mellan mätsystem och databas/server

Då det inte finns färdiga klasser i .NET Gadgeteer/.NETMF för att kommunicera direkt mellan Microsoft SQL och FEZ Raptor behövs någon form av länk mellan systemen.



**Figur 5.1:** Modell över kommunikation för systemet och databas/server

Länken mellan mätsystemet och databasen är en HTTP-server skriven i Java, nedan är ett flödesdiagram som visar serverns funktion. Detta kallas även för webservice.



**Figur 5.2:** Funktionell modell över HTTP-servers/webservicens funktion.

Förutom att fungera som en koppling mellan mätsystem och databas ger en web-service ytterligare fördelar jämfört med en direkt koppling, bland annat[56]:

- Klienter behöver inte lagra lösenord och annan känslig information till databasen.
- Ger enklare möjlighet att byta ut databas utan att programmera om klienter.
- Ger enklare möjlighet att kombinera många datakällor i samma service.
- Möjlighet att använda samma format oavsett datakälla.

FEZ Raptorn skickar en HTTP-request i form av en url, exempel på en sådan URL kan se ut som nedan:

```
/0/x/-0.138671875/-0.140625/-0.140625/-0.138671875/-0.14453125/-0.140625... HTTP/1.1
```

En HTTP-request består av ett ID, en typ (tid, X-värde, Y-värde, Z-värde) samt hundra datavärden. Dessa olika värden och parametrar skapar ett objekt av klassen *RequestString*. Ur detta objekt sätts ett antal variabler i programmet som används för att skapa objekt av klassen *Entry* som kommer att innehålla all information från en mätpunkt. När ett objekt av klassen *Entry* innehåller all data skriver programmet denna datan till en Microsoft SQL Databas för att senare kunna hämtas och analyseras av en webbapplikation eller liknande.

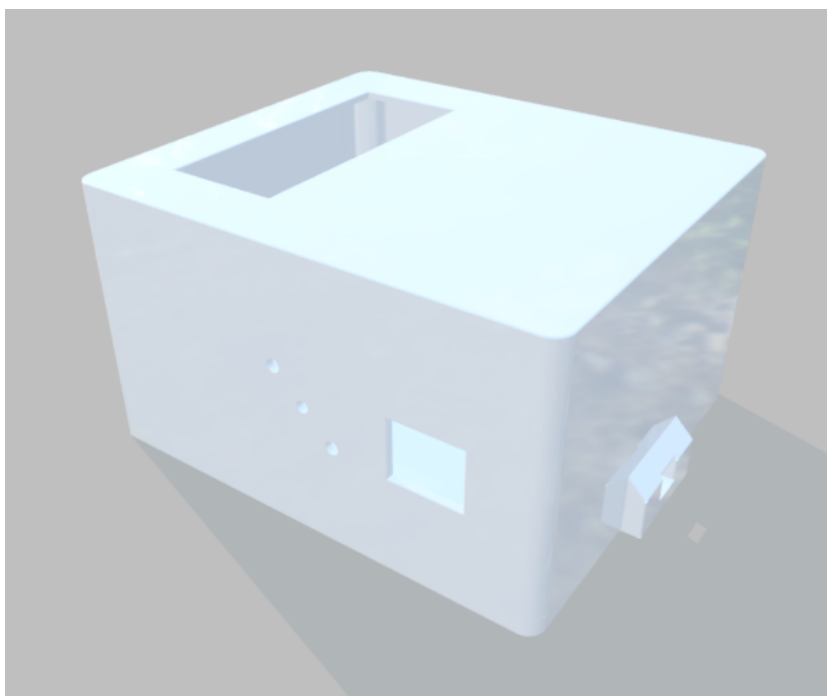
### 5.1.1.3 Reflektion av webservice/kommunikation

Kommunikationen mellan mätsystemet och servern hade kunnat utvecklas till ett enklare system genom att använda till exempel JSON-klasser för att skicka datan. Användning av JSON förenklar konverteringen mellan objekt i C# och de HTTP-requests och HTTP-response som skickas mellan enheterna. Tidsåtgången för utvecklingen av systemet hade minskats.

### 5.1.2 Utveckling av inkapslande chassi

Vid början av projektet gjordes en kravspecifikation för lådan som ska omsluta moderkortet till FEZ-raptorn. Kraven i kravspecifikation togs i beaktande när designen skulle utformas, då moderkortet redan har standardiserad mått fanns det inte en så stor marginal att spela på vad det gäller mått med mera för lådan. Målet med designen var att ta fram en mer praktisk låda till moderkortet då den existerande var något otymplig och stor. Detta målet kom från marknadsundersökningen i kapitel 4.1.1 där det konstaterades att ett portabelt system var önskvärt.

Utvecklingen gick till på så sätt att en första skiss över designen gjordes för hand på papper där de grova design dragen skissades upp för att skapa ett underlag för att ta fram en datorgenererad 3D-modell. Därefter skapades en 3D-modell i Catia V5 med de design drag som ansågs vara mest nödvändiga för att ha ett funktionellt skydd för moderkortet. Chassits design låter användaren få en ökad upplevelse för att få ut de önskade funktionerna. De viktigaste funktionerna är att mäta vibrationer med sensor, se data på display samt att kunna ladda upp data på en server med hjälp av enkla kontroller.



**Figur 5.3:** 3D-modell av inkapsling

När lådan var modellerad i Catia tillverkades lådan. Detta gjordes med hjälp av 3D-skrivning. Kravspecifikationen som är gjord för lådan är till för en produkt som ska fungera i en industriellmiljö, den prototyp som projektet har resulterat i är inte en fullvärdig produkt för en industriellmiljö. Det resulterade i en prototyp som är till för att visualisera hur en riktig produkt skulle kunna se ut. Den prototyp som tagits fram uppfyller inte den IP-klass som är satt utefter kravspecifikationen då det använda produktionssättet att tillverka lådan i använder ett material som inte är vattentätt och poröst.

## 5.2 Utveckling av nytt mätsystem

Vid projektets start blev gruppen tilldelad en Raspberry Pi 2, som är en enkortsdator med stiftanslutningar för externa komponenter (se 2.3.3). I projektets slutfas tilldelades gruppen en Raspberry Pi modell 3, vilket möjliggör uppkoppling via WiFi. En Raspberry Pi har till skillnad från en vanlig dator ingen hårddisk som operativsystemet installeras på, istället monteras ett micro SD-kort som i sin tur operativsystemet är installerat på. På grund av att Raspberryn har en ARM-baserade processor så finns det ingen möjlighet att installera Windows [54], och istället måste ett operativsystem som stödjer ARM-arkitekturen installeras, exempelvis någon Linux-distribution.

Efter litteraturstudier av operativsystem kompatibla med Raspberry pi togs beslutet att installera operativsystemet Raspbian. Detta operativsystem är en Linuxdistribution som är optimerat för användning på just en Raspberry Pi. Raspbian ger en känsla av att arbeta i ett Windows-system, då skrivbord, mappsystem, filhanterare och annat som brukar finnas i en vanlig dator finns [54]. Valet av Raspbian byggde på att gruppen ville hålla gränssnittet så enkelt som möjligt, då ingen av gruppmedlemmarna var vana att arbeta i ett Linux-system.

### 5.2.1 Installation av vibrationssensor

När operativsystemet var installerat var installation av en accelerometer möjlig. I projektets inledningsfas sattes ett mål upp där samma accelerometer som används i det tidigare systemet (Accel G248) skulle installeras på Raspberry:n om det var möjligt.

#### 5.2.1.1 Installation av Accel G248 Module

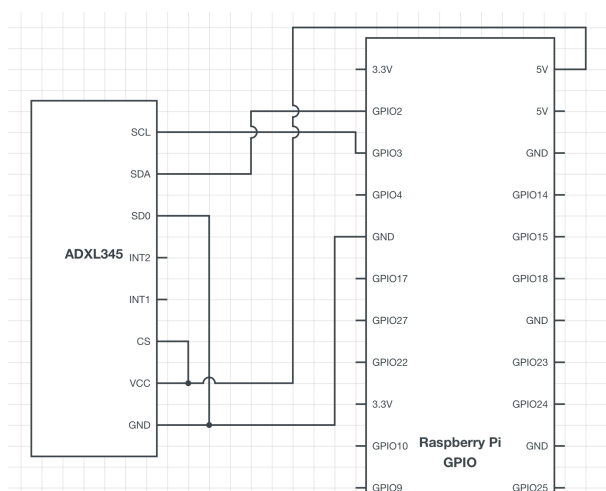
Accelerationsmodulen G248 bygger på en MEMS-accelerometer (se 2.3.1) med benämningen MMA8453Q från Freescale Semiconductor, Inc. [22]. Denna modul är från början utvecklad för att fungera med huvudkort från GHI Electronics, exempelvis FEZ-Raptor som används i det tidigare systemet. Detta ledde till vissa komplikationer, i form av fysiska problem vid inkoppling samt kommunikationsproblem mellan Raspberry:n och accelerometern.

Vid inledande försök att få kommunikationen mellan sensorn och huvudenheten

att fungera upptäcktes att kodning på lågnivå var nödvändigt för att möjliggöra avläsning av sensorvärdena trots att sensorn kommunicerar via I2C-bus som Raspberry Pi stödjer. Vid litteraturstudier i databladet för sensorn [55] visade det sig att de register som innehöll de olika datavärden som behövdes inte var specificerade. Anledningen till detta var att sensorn inte var anpassad för att fungera med något annat system än .NET Gadgeteer/.NETMF. På grund av detta togs beslutet att istället installera accelerometern ADXL345.

### 5.2.1.2 Installation av ADXL345

Installationen av ADXL345 blev, på grund av att den var anpassad för att fungera med en mängd olika system, mycket mindre tidskrävande än Accel G248 Module. Kommunikationen mellan Raspberry:n och sensorn var möjlig med både I2C-bus och SPI-bus, dock valdes I2C-bussen på grund av att det var beprövat tidigare av förra projektgruppen [22] med lyckat resultat.

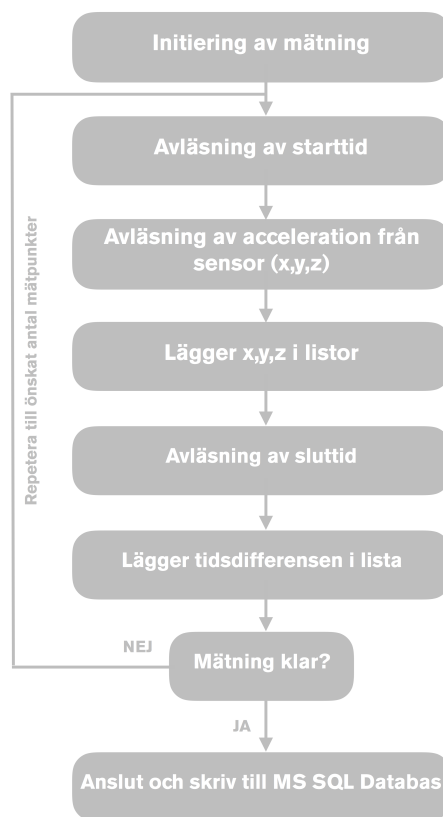


**Figur 5.4:** Kopplingsschema för Raspberry - ADXL345

Figuren ovan visar en principbild på kopplingen mellan Raspberry:n och sensorn. Som synes i kopplingsschemat är inkopplingen av sensorn mycket enkel på grund av att få kablar behöver kopplas in. Inkopplingen byggde från början på stiftlistor och kablar avsedda för inkoppling i dessa listor. Senare i projektet kom kablarna att lödas för att förhindra störningar och även minimera risken för att kablar lossnar.

### 5.2.2 Utveckling av mjukvara

För att uppfylla de krav som ställdes på funktionen av systemet både i syftet och i inledningsfasen av arbetet ställdes en funktionell modell över hur designen av mjukvaran skulle se ut. Modellen visar övergripande de funktioner som mjukvaran skall ha i stort. Programmet är skrivit i programmeringsspråket Python och körs direkt i operativsystemet Raspbians terminal.



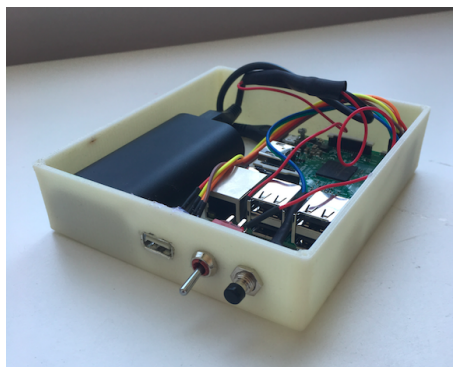
**Figur 5.5:** Funktionell modell över mjukvaran för datainsamling i Raspberry:n.

Mjukvaran utvecklades med mål att följa den funktionella modellen. Tack vare av att utveckling av Raspberry Pi-enheter är vanligt så fanns mycket värdefull information att hitta på internet. Exempelvis så fanns det mycket information angående hur uppkoppling mot en databas går till. Med hjälp av studerande av tidigare kod samt trial and error-metoden utvecklades mjukvaran med enkelhet för att utföra de uppgifter som ställdes upp i den funktionella modellen.

### 5.2.3 Utveckling av hårdvara

Liknande hur utvecklandet av det inkapslande chassit för det tidigare systemet gick till så utnyttjades även här CAD samt 3D-skrivare för utvecklingen. Med hjälp av de krav som ställts på systemet i form av att systemet skall vara portabelt, i form av att ingen yttre permanent strömkälla skulle behövas, samt att systemet skulle vara modulärt i form av att möjlighet för byte av sensor skulle finnas utvecklades en modell i en CAD-miljö.

När CAD-modellen var klar användes en 3D-skrivare för att ta fram en prototyp. Prototypen bearbetades därefter med enkla verktyg för att möjliggöra montering av de komponenter som behövdes. Resultaten blev en inkapsling med ett batteri-pack på 2200 mAh som strömförsörjning, en kontakt som möjliggjorde inkoppling av vibrationssensorn, en strömbrytare för att starta systemet och till sist en knapp för att utföra vibrationsmätning.



**Figur 5.6:** Det nya mätsystemet utan monterat lock.

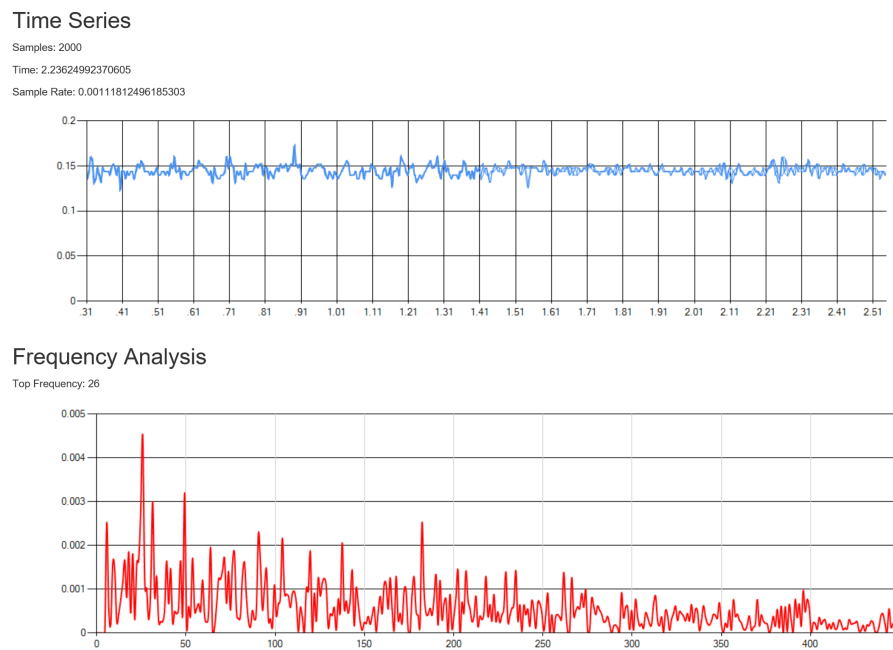
Även vibrationssensorn kapslades in. Även denna inkapsling utvecklades i CAD och framtogs med hjälp av en 3D-skrivare. Detta utfördes på grund av visionen att prototypen skulle fungera i en industrimiljö där smuts och fukt finns, vilket sensorn behövdes skyddas mot. För att inkapslingen skulle påverka vibrationsmätningen på ett så litet sätt som möjligt limmades sensorn och inkapslingen ihop för att bli en hel enhet. Hur inkapslingen påverkade mätresultatet undersöktes ej på grund av tidsbrist. En USB-kabel monterades på sensorn för att uppnå kravet på ett modulärt system, i enlighet med kravspecifikationen i kapitel 4.3.2.



**Figur 5.7:** Inkapslad vibrationssenor med USB-kabel monterad.

### 5.3 Utveckling av webbsida för visualisering av mätdata

Webbsidan utvecklades som en webbapplikation i Visual Studio med ASP.NET. Med hjälp av integrerade verktyg för databasanslutning och kontroller skapades en enkel webbsida där datan visualiseras. Fouriertransformen som krävs för att visa frekvensspektrumet genomfördes också av webbapplikationen i form av en FFT och baseras på resultanten av accelerationen i x-,y- och z-led från den insamlade datan.



**Figur 5.8:** Skärmdump av webbsida för visualisering av mätdata

## 5.4 Reflektion på utvecklingsfasen

Användning av olika matriser för att se vad det är som efterfrågas och för att kunna ställa krav på en kommande produkt har gjort att resultatet har blivit på ett sådant sätt att det tillfredsställer en potentiell kunds största behov. Det har gjort att utvecklingsprocessen har lett till uppfyllandet av de krav som ställts i de olika kravspecifikationerna i kapitel 4.

Det område där mer tid och energi hade kunnat läggas är vid utvecklingen av kravspecifikationerna. Fler krav och mål, och ännu mer nedbrutna specifikationer, hade förmodligen kunnat styra projektet i en tydligare riktning med mindre utsvängningar. Exempelvis hade kravspecifikationen för inkapslingschassit, vilken ursprungligen syftade till en industriduglig produkt, kunnat vara mer inriktad mot framtagandet av en prototyp. Principformen hade varit densamma för både prototyp som färdig produkt, men prototypframtagningsmetoden har varit 3D-skrivning vilket inte är en perfekt metod. Bland annat är vissa vinklar som krävs för hållframställning svåra att få till ordentligt vid 3D-skrivning. Hänsyn till detta fanns inte taget i kravspecifikationen, så ändringar i ritningarna fick göras allt eftersom.

### 5.4.1 Reflektion utveckling av mjukvara

Mjukvaran i det nya mätsystemet skickar data till databasen utan mellansteget med webservern. Anledningen till detta var att projektgruppen ansåg att syftet med projektet uppfyllades utan att implementera detta. Vikt måste dock läggas vid att uppkoppling utan detta mellansteg gör systemet samt databasen inflexibelt och sårbart ur vissa synpunkter. Känslig information lagras på klienten (mätsystemet) vilket är negativt ur säkerhetssynpunkt. Vid byte av databas måste klienter programmeras om och svårigheter kan uppstå om olika enheter skall kopplas till samma databas och de är programmerade på olika sätt.



# 6

## Utvärderingsfas

För att utvecklingen av systemen skulle kunna betraktas som klara inleddes arbetet med att utvärdera dessa mot deras respektive kravspecifikationer. I metodkapitlet konstaterades att utvärderingens precision och säkerhet kan höjas med hjälp av fler tester med mer data från fler källor, detta ledde till utvecklingen av ett nytt testsystem.

### 6.1 Utveckling av nytt testsystem

Syftet med det nya systemet som skulle utvecklas var att samla in mer verklig-hetsbetonad data i form av data från faktiska kullager. Genom att montera lager i olika skick i testsystemet kan mätsystemets förmåga att identifiera och analysera lagerskador utvärderas.

#### 6.1.1 Referenslösning

Likt mätanordningen lyckades kandidatarbetet *“Prototyp för konditionsbaserat underhåll med enkla sensorer”* ta fram en enkel men bra prototyp för testmiljö. Testmiljöprototypen ställdes in för att vibrera och tillåta användaren att testa bestämda signaler och därigenom genomföra analyser.

Den valda referenslösningen är det framtagna testsystemet i kandidatarbetet *“Prototyp för konditionsbaserat underhåll med enkla sensorer”*. Referenslösningen består i grunden av en huvudplatta där en mindre vibrationsplatta som ger ifrån sig vibrationer upp till 200 Hz monterats. För att uppfylla kravet på mätning upp till 240 Hz som ställts i kravspecifikationen för det nya systemet, se kapitel 4.3.2, kommer referensen sättas därefter. Plattan drivs av DC-motorer som styrs med PWM teknik via en Arduino. Genom att reglera motorernas varvtal kan den frekvens vibrationsplattan vibrerar med också regleras till önskad frekvens.

Det nya testsystemet liksom det gamla utgår från mikrokontrollern Arduino.

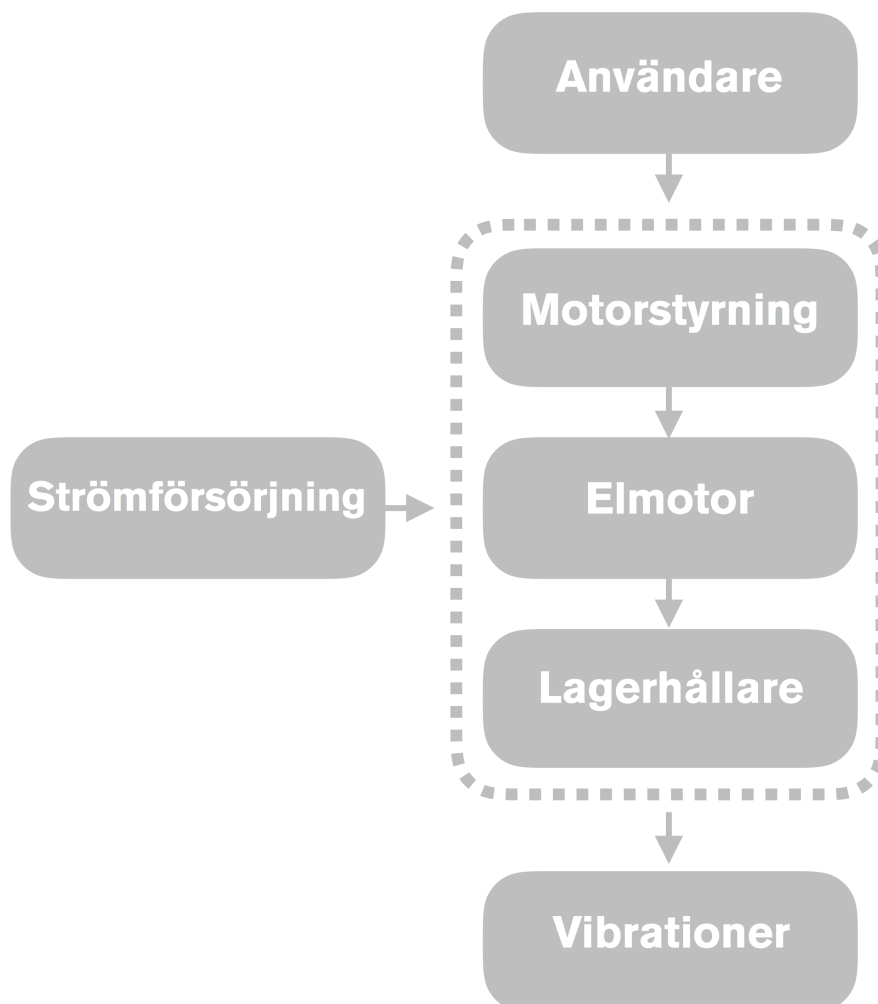
#### 6.1.2 Konzeptgenerering

För att underlätta framtagningen ställdes en kravspecifikation upp för den nya testmiljön. Därefter utfördes en brainstorming med både specifikationen och referenslösningen i åtanke för att ta fram fram koncept som skulle kunna nå kraven. Mycket

inspiration vid konceptgenereringen hämtades även från koncept 6 från tidigare års kandidatarbete *“Prototyp för konditionsbaserat underhåll med enkla sensorer”*[22]. Ett slutligt koncept som verkade lovande togs tillslut fram.

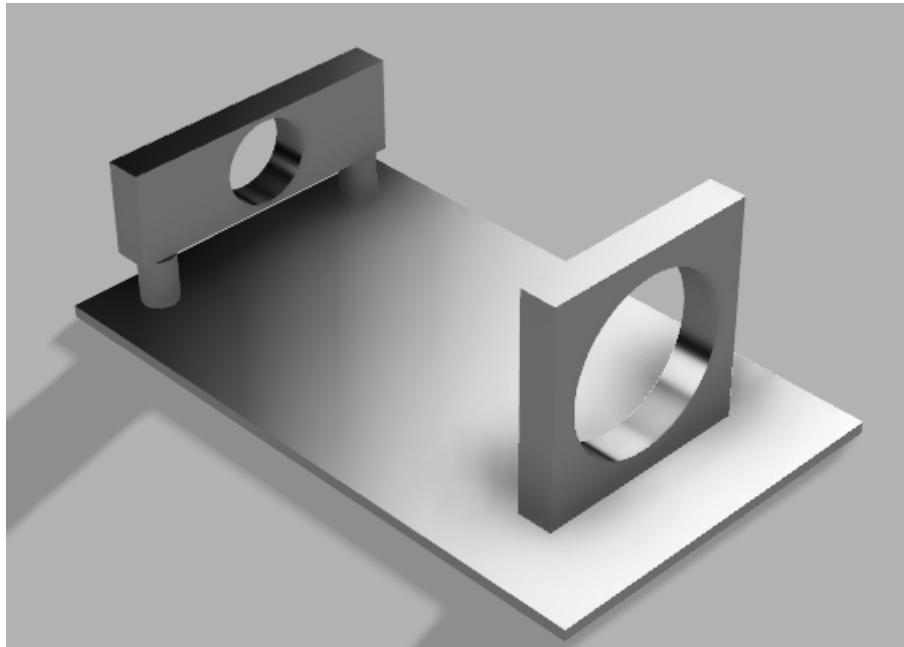
### 6.1.3 Funktion och CAD-modell

Konceptet består av en elmotor och en lagerhållare som är sammankopplade med en axel vilket möjliggör rotering av lagret som skall analyseras. Testsystemet utformas för att ge användaren möjlighet att på ett enkelt sätt, via enklare kodning, styra varvtalet på det roterande lagret. Då lagret som analyseras är i dåligt skick så utsätts drivaxeln för varierande krafter och moment. För att möjliggöra rotation med ett konstant varvantal behövs ett regelsystem, detta implementeras i samma program som registrerar önskat varvantal från användaren med hjälp av en mikrokontrollern Arduino. Nedan är en funktionell modell över testsystemets funktion.



**Figur 6.1:** Funktionell modell testsystemet.

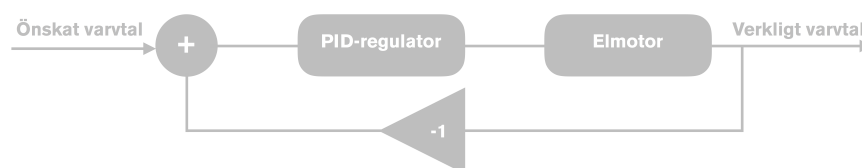
När konceptet var framtaget utvecklades en grov skiss av systemet i CAD. Syftet med skissen var att fungera som en principskiss för att underlätta den praktiska utvecklingen av systemet i verkstaden.



**Figur 6.2:** CAD-skiss på testsystemet.

#### 6.1.4 Motorstyrning

Till mikrokontrollern Arduino finns mycket tillgänglig kod från utvecklarens webbsida, det finns också mycket färdiga bibliotek som underlättar programmeringen av reglersystemet för elmotorn. Elmotorn och dess varvtal regleras med hjälp av en PID-regulator, nedan syns en skiss över systemet:



**Figur 6.3:** Principskiss över reglersystemet med PID-regulator.

Varvtalet för det roterande systemet mäts med hjälp av en hall-effektsensor som registrerar varje gång systemet roterar ett helvarv. För att få en klar signal till mikrokontrollern används en debounce krets bestående av en resistor, inverterad schmitt trigger samt kapacitator. Reglersystemet jämför sedan det aktuella varvtalet med ett av användaren angivet varvtal och reglerar strömmen till motorn med hjälp av en PWM-signal.

### 6.1.5 Reflektion utveckling av nytt testsystem

På grund av bristande tid, kunskap och planering lyckades det nya testsystemet inte bli färdigställt innan tiden för projektarbetet hade tagit slut. På grund av detta togs ett beslut att validera mätsystemen med hjälp av det testsystem som redan fanns. För att på ett så bra sätt som möjligt kunna validera sina resultat hade det varit gynnsamt för gruppen att ha ett testsystem som varit klart i tid. Det som ledde till att testsystemet inte blev klart i tid var att gruppen under planeringsfasen underskattat mängden tid som det krävdes för att färdigställa systemet. Hade mer vikt lagts åt att undersöka tidigare projekt där liknande system utvecklats för att få en bild av hur mycket tid som borde varit rimligt att lägga ner.

Trots att målet inte är fullständigt nått har dock gruppen fått bra kunskaper om hur en relativt enkel testsystem skulle konstrueras och fungera med en enkel motor som styr varvtal med en Arduino som bas. Det kommer vara till stor nytta i framtida projekt när ett testsystem inte finns för en produkt som ska utvecklas. Kunskapen att veta hur ett bra verktyg för testning ska se ut och fungera kommer alltid kunna utnyttjas.

## 6.2 Validering av mätsystem

För att få en uppfattning om hur väl det nya mätsystemet fungerade i förhållande till det gamla mätsystemet utfördes empiriska studier. Dessa studier byggde på analysering av mätdata hämtat från vibrationsbänken från förra årets projektgrupp [22], högtalare samt maskiner med roterande lager.

### 6.2.1 Inledande analyser

De inledande analyserna byggde på att försäkra funktionen av den nya accelerometern (ADXL345). Dessa analyser genomfördes utan hölje på accelerometern. Här utfördes tester med hjälp av vibrationsbänken, och den insamlade datan jämfördes med data insamlad från det tidigare systemet. Resultatet visade på att sensorernas data var lika inom  $\pm 2\%$ .

#### 6.2.1.1 Reflektion inledande analyser

Differensen mellan datan från de två sensorerna sågs som godtagbar på grund av att det kunde bero på mätosäkerhet, olika placeringar av sensorerna samt skillnader i montering av sensorerna. Datan från den nya vibrationssensorn antogs på grund av detta stämma.

## 6.2.2 Fortsatta analyser

För att försäkra funktionen av hela det nya systemet i form av sensor, moderkort, databas och webbsida utfördes empiriska studier på roterande maskiner samt högtalare. Målet med dessa studier var både att kalibrera hela systemet för att ett eftersökt resultat skulle visualiseras på webbsidan samt försäkra att datan visualiserades med den hastighet som var uppställd i kravspecifikationen.

### 6.2.2.1 Tester på högtalare

En högtalare ställdes in på att generera en sinussignal med bestämd frekvens. Systemet fick samla in, analysera och visualisera datan i form av en FFT på webbsidan.



**Figur 6.4:** Sensorn liggande på en högtalare.

De frekvenser som analyserades var 50 till 150 Hz, i steg av 10 Hz. Anledningen till att lägre eller högre frekvenser inte testades var högtalarens begränsning. Resultatet visade på att systemet gav ett mätvärde som låg 8 % över den frekvens som genererades, på alla testade frekvenser. Den större differensen har delvis sin grund i att de inledande analyserna jämförde rådatan mellan de två systemen, medan det vid högtalartesterna var en jämförelse av genererad sinusfrekvens mot FFT-resultat på webbsidan. Här får även vikt läggas vid att sensorn vid dessa tester var monterad i sitt hölje, vilket kan ha påverkat resultatet.

### 6.2.2.2 Tester på roterande maskiner

För att analysera frekvenser utanför det frekvensområdet vilket högtalaren kunde generera utfördes tester på roterande maskiner. Målet med dessa tester var både att analysera lägre frekvenser, samtidigt som dessa mätningar ger en bättre bild av systemets funktion, då det är analys av roterande maskiner som projektet syftar till. Testerna utfördes på en pelarborr i E-sektionens Teletekniska Avdelning: ETA.



**Figur 6.5:** Sensorn monterad på en pelarborr.

Sensorn monterades med hjälp av tejp så nära ett roterande lager som möjligt. För att beräkna förväntat resultat vid ett visst varvtal användes följande formel:

$$Frekvens = \frac{Varvtal}{60}$$

Tester utfördes med varvtal från 600 till 2400, vilket enligt ovanstående formel ger frekvenser på mellan 10 och 40 Hz. Även dessa tester visade att systemet visade ett resultat som var 8 % över det beräknade resultatet på alla testade frekvenser. Systemet visade vid alla tester, både på högtalare och roterande maskiner, samma fel. På grund av detta kalibrerades sensorn. Efter kalibreringen visade systemet det beräknade resultatet. Tiden för att datan skulle visualiseras på webbsidan uppskattades till 1.5 sekunder för 500 datapunkter, vilket uppfyller och överträffar de krav som ställts på systemet i kapitel 4.3.2.

### 6.2.2.3 Reflektion fortsatta analyser

För att försäkra resultatet borde fler tester utförts, detta för att kalibreringen av sensorn skulle bli mer exakt. Tester som hade kunnat utföras var exempelvis tester med inkapslad sensor på vibrationsbänk samt tester med icke inkapslad sensor på pelarborr och högtalare. Kalibreringen av systemet bygger efter dessa analyser på att pelarborren samt högtalaren fungerar perfekt. Det nya systemet hade innan kalibreringen 2 %-enheter större mätfel än vad det tidigare systemet hade innan kalibrering [22]. Detta kan bero på att sensorn till det nya systemet är inkapslad, vilket den gamla sensorn inte är.

## 6.3 Reflektion på utvärderingsfasen

Utvärderingsfasen visade på ett resultat där det nya mätsystemet hade ett större mätfel än det tidigare systemet innan kalibreringen. Som sagt tidigare beror detta förmodligen på inkapslingen av sensorn. För att få detta validerat borde fler tester utförts, något som tiden satte stopp för. Det nya mätsystemet uppfyller dock alla de krav som var uppställda i kravspecifikationen i kapitel 4.3.2.

För att utvecklandet av den nya testbänken skulle lyckas borde gruppen satsat på att utveckla ett testsystem utifrån ett fungerande koncept istället för att utveckla systemet med hjälp av egna idéer och erfarenheter. En lämplig metod att använda sig av hade varit att göra en marknadsundersökning eller eventuellt ett patentsök genom Chalmers bibliotek för att få en övergripande bild av hur andra system ser ut. Då det testsystem som har tagits fram verkar fungera i teorin borde den reflektera verkligheten. Om förstudier gjorts hade antagligen resultatet blivit till det bättre och mer pålitligt.



# 7

## Diskussion och slutsats

En prototyp för att visualisera roterande maskiners kondition har tagits fram. Prototypen mäter och analyserar vibrationer genom insamlad accelerationsdata för att sedan, i IoT-anda, visualisera det på en webbsida. Kombinationen av de valda sensorerna med enkortsdatorn Raspberry Pi har visat sig uppfylla projektets syfte där användaren på ett enkelt och tydligt sätt erhåller information om aktuell maskinstatus.

Det gamla systemet, baserat på FEZ Raptor, har utvecklats för att kunna jämföras med det nya mätsystemet. Tester har visat att systemen har mycket liknande prestanda, där den stora skillnaden är att det nya systemet är billigare och kan visualisera uppmätt data utan onödiga mellansteg i form av att användaren själv måste överföra datan från SD-kort till analysprogram. Raspberry Pi-systemet kan kommunicera med en webbsida och därigenom kontinuerligt uppvisa maskinstatus för operatören. På så sätt är det enkelt för operatören att avläsa skillnaden i maskinens beteende vilket, i en större utsträckning, leder till effektivare underhållsstrategier då underhållet kan hållas till ett minimum.

Det nya mätsystemet har lyckats samla in den accelerationsdata som krävs för att genomföra vibrationsanalyser och har även lyckats visualisera detta på en hemsida. Dock är insamlad data inte tillräcklig för att validera huruvida systemet klarar av att detektera lagerskador, då inga tester av den typen har utförts. Gruppen bestämde istället att om det nya systemets, Raspberry Pi, insamling och analys av datan ger liknande resultat som det gamla systemet, FEZ Raptorn, är det en tillräcklig validering av systemets funktioner. Till grund för detta låg det faktum att det tidigare visats att det gamla systemet har fungerat i praktiken.

Raspberry Pi-systemet har brister liknande det gamla systemet i form av att det inte kan mäta lika brett frekvensområde som befintliga produkter på marknaden. Dock mäter det nya systemet högre frekvenser än det tidigare systemet och datan kan överföras snabbare vilket gör att mer data kan samlas in under samma tidsperiod, vilket i slutändan ger en noggrannare analys av lagerskador. Raspberry Pi har möjlighet att mäta högre frekvenser än vad som gjorts i detta projekt då den största begränsningen har varit mjukvaran snarare än hårdvaran. Hade programmeringen optimerats för att läsa data från accelerometer snabbare så hade ett större frekvensområde uppnåtts.

Hade den planerade testbänken, i vilken lager kunde monteras, blivit fullständigt

framtagen hade testerna kunnat validera funktionen på ett verklighetsförankrat sätt. I projektets start var gruppen positivt inställda till mängden tid och hade god tro att testbänken skulle kunna slutföras på ett tillfredställande sätt. Det visade sig dock att tiden inte var tillräcklig då mer fokus behövde läggas på mjukvaruutvecklingen av de båda mätsystemen.

Sammanfattningsvis har projektet visat att mätsystemet baserat på Raspberry Pi är både lättare att utveckla samt att hårdvaran är billigare än för FEZ raptorn.

### 7.1 Framtid

Då det är en prototyp som utvecklats uppfyller inte inkapslingschassit de krav som ställs för att få den IP-klassificerad för industrin. För att produkten skall vara användbar på industrinivå krävs utveckling av dessa områden, och då främst inkapslingens täthet gällande att stänga ute damm och fukt.

För att göra systemet mer användarvänligt bör det utvecklas så att det känner igen lagerskador, utan att en kunnig person ska sitta och analysera datan som kommer in. Kan systemet själv känna igen skadliga vibrationer blir det lättare för användare att se när ett lager är skadat. På så sätt minskar behovet av att operatören har stor erfarenhet och kunskap kring hur lagernas beteende reflekterar dess hälsa. För att lyckas med detta behöver testbänken utvecklas så att olika lagerbeteenden kan byggas in i systemets databas och kännas igen vid vibrationsmätningar.

För att utveckla analysen av maskinernas kondition skulle även andra parametrar än enbart vibrationer vara intressanta att mäta och analysera. Olja, temperaturer och ljud är några av dessa och bidrar till en bättre bild över en maskins kondition. Systemet skulle kunna utvecklas till en helhetslösning där flera parametrar analyseras samtidigt och då ge en nästintill optimal bild över vilket skick maskinen befinner sig i.

Projektet har lyckats visa att insamlad data går att visualisera på ett enkelt sätt som ger en tydlig bild över de vibrationer en roterande maskin ger ifrån sig. Sett utifrån presenterade resultat har de krav och mål som satts upp i projektet uppfyllts och därmed även dess syfte, att samla in och visualisera data med hjälp av billig hårdvara.

# Referenser

- [1] "Condition Based Maintenance and Monitoring | Maintenance Assistant", Maintenance Assistant, 2016. [Online]. Available: <https://www.maintenanceassistant.com/condition-based-maintenance/>. [Accessed: 29- Apr- 2016].
- [2] Basim Al-Najjar. 2007. Med färre dyra stopp kan produktionen räddas kvar. [ONLINE] Available at: <http://www.e-maintenance.se/wp-content/uploads/2013/03/Dagens-industri-artikel-2.pdf>. [Accessed 20 May 2016].
- [3] Bearing failures and their causes, Product Information 401", SKF, 1994. [Online]. Available at: [http://www.mana-omerzu.si/files/Bearing\\_Failures\\_and\\_Their\\_Causes.pdf](http://www.mana-omerzu.si/files/Bearing_Failures_and_Their_Causes.pdf). [Accessed 16 May 2016]
- [4] Downtime Costs Auto Industry 22k/Minute – Survey. 2016. *Downtime Costs Auto Industry 22k/Minute - Survey*. [ONLINE] Available at: <http://news.thomasnet.com/companystory/downtime-costs-auto-industry-22k-minute-survey-481017>. [Accessed 16 May 2016]
- [5] msn.microsoft.com. 2014. Introducing Visual Studio. [ONLINE] Available at: [https://msdn.microsoft.com/en-us/library/fx6bk1f4\(v=vs.90\).aspx](https://msdn.microsoft.com/en-us/library/fx6bk1f4(v=vs.90).aspx). [Accessed 19 April 2016].
- [6] Arduino. 2014. What is Arduino?. [ONLINE] Available at: <https://www.arduino.cc/en/Guide/Introduction>. [Accessed 19 April 2016].
- [7] raspberry.org. 2015. FAQs. [ONLINE] Available at: <https://www.raspberrypi.org/help/faqs/#introWhatIs>. [Accessed 19 April 2016].
- [8] Arduino. 2016. Arduino UNO & Geuino UNO. [ONLINE] Available at: <https://www.arduino.cc/en/Main/ArduinoBoardUno>. [Accessed 19 April 2016].
- [9] Weisstein, Eric W. Fast Fourier Transform. From MathWorld—A Wolfram Web Resource. <http://mathworld.wolfram.com/FastFourierTransform.html>
- [10] Weisstein, Eric W. Fourier Transform. From MathWorld—A Wolfram Web Resource. <http://mathworld.wolfram.com/FourierTransform.html>
- [11] MathWorks. 2016. The Language of Technical Computing. [ONLINE] Available at: <http://se.mathworks.com/products/matlab/>. [Accessed 19 April 2016].
- [12] Internet Of Things Sverige. 2010. Om IoT. [ONLINE] Available at: <http://iotsverige.se/internet-things-2/>. [Accessed 19 April 2016].
- [13] the internet of things. 2016. IoT news. [ONLINE] Available at: <http://www.theinternetofthings.eu>. [Accessed 19 April 2016].

- [14] the internet of things. 2016. The Internet of Things. [ONLINE] Available at: <http://www.theinternetofthings.eu/what-is-the-internet-of-things>. [Accessed 19 April 2016].
- [15] en-us.fluke.com. 2015. Understanding the benefits of vibration monitoring and analysis. [ONLINE] Available at: <http://en-us.fluke.com/community/fluke-news-plus/vibration/understanding-the-benefits-of-vibration-monitoring-and-analysis.html>. [Accessed 19 April 2016].
- [16] spminstrument.se. 2002. Vibrationsövervakning. [ONLINE] Available at: <http://www.spminstrument.se/Matteknik/Vibrationsovervakning/>. [Accessed 19 April 2016].
- [17] sandv.com. 2001. From Vibration Measurements to Condition Based Maintenance. [ONLINE] Available at: <http://www.sandv.com/downloads/0701mitc.pdf>. [Accessed 19 April 2016].
- [18] maintenanceassistant.com. 2006. Condition based maintenance (CBM). [ONLINE] Available at: <https://www.maintenanceassistant.com/condition-based-maintenance/>. [Accessed 19 April 2016].
- [19] mana-omerzu.si. 2011. Bearing failures and their causes. [ONLINE] Available at: [http://www.mana-omerzu.si/files/Bearing\\_Failures\\_and\\_Their\\_Causes.pdf](http://www.mana-omerzu.si/files/Bearing_Failures_and_Their_Causes.pdf). [Accessed 19 April 2016].
- [20] Paul Daugherty, Prith Banerjee, Walid Negm and Allan E. Alter. 2015. Driving Unconventional Growth through the Industrial Internet of Things. [ONLINE] Available at: [https://www.accenture.com/us-en/\\_acnmedia/Accenture/next-gen/reassembling-industry/pdf/Accenture-Driving-Unconventional-Growth-through-IIoT.pdf](https://www.accenture.com/us-en/_acnmedia/Accenture/next-gen/reassembling-industry/pdf/Accenture-Driving-Unconventional-Growth-through-IIoT.pdf). [Accessed 19 April 2016].
- [21] Per Lindstedt, 2003. The Value Model : How to Master Product Development and Create Unrivalled Customer Value. Edition. Nimba, NA.
- [22] Börjesson, S. and Holck, V. and Karlsson, K. and Skalberg, K. (2015) Prototyp för konditionsbaserat underhåll med simpla sensorer, Gothenburg: Institutionen för Produkt- och Produktionsutveckling.
- [23] maintenanceassistant.com. 2015. Run-to-failure maintenance. [ONLINE] Available at: <https://www.maintenanceassistant.com/run-to-failure-maintenance/>. [Accessed 22 April 2016].
- [24] maintenanceassistant.com. 2015. Planned maintenance. [ONLINE] Available at: <https://www.maintenanceassistant.com/planned-maintenance/>. [Accessed 22 April 2016].
- [25] omdec.com. 2002. A History Of CBM (Condition Based Maintenance). [ONLINE] Available at: <http://www.omdec.com/moxie/Technical/Reliability/a-history-of-cbm.shtml>. [Accessed 22 April 2016].
- [26] fluke.com. 2015. Understanding the benefits of vibration monitoring and analysis. [ONLINE] Available at: <http://en-us.fluke.com/community/fluke-news-plus/vibration/understanding-the-benefits-of-vibration-monitoring-and-analysis.html>. [Accessed 22 April 2016].

- 
- [27] renewablenrgsystems.com. 2013. WHAT IS YOUR RETURN ON CONDITION-BASED MAINTENANCE?. [ONLINE] Available at: <http://www.renewablenrgsystems.com/blog/what-is-your-return-on-condition-based-maintenance/>. [Accessed 22 April 2016].
  - [28] Randall, R (2010). Vibration-based Condition Monitoring : Industrial, Aerospace and Automotive Applications. University of New South Wales, Australia : Wiley. p3-20.
  - [29] netmf.com. 2007. What is Gadgeteer. [ONLINE] Available at:<http://www.netmf.com/gadgeteer/>. [Accessed 25 April 2016].
  - [30] ghielectronics.com. 2002. .NET Micro Framework. [ONLINE] Available at: <https://www.ghielectronics.com/technologies/netmf%23gadgeteer>. [Accessed 25 April 2016].
  - [31] P.L.Walter, Sound and Vibration Magazine,"01 2007. [ONLINE] Available at: <http://www.sandv.com/downloads/0701walt.pdf>. [Accessed 25 April 2016].
  - [32] ANALOG DEVICES. 2002. ADXL345 datasheet. [ONLINE] Available at:<https://www.sparkfun.com/datasheets/Sensors/Accelerometer/ADXL345.pdf>. [Accessed 25 April 2016].
  - [33] I, Poberaj. 1992. MEMS ACCELEROMETERS. [ONLINE] Available at:[http://mafija.fmf.uni-lj.si/seminar/files/2007\\_2008/MEMS\\_accelerometers-koncna.pdf](http://mafija.fmf.uni-lj.si/seminar/files/2007_2008/MEMS_accelerometers-koncna.pdf). [Accessed 25 April 2016].
  - [34] Sparkfun. 2002. SparkFun Triple Axis Accelerometer Breakout - ADXL345. [ONLINE] Available at: <https://www.sparkfun.com/products/9836>. [Accessed 25 April 2016].
  - [35] Gadgeteer, (2010), NANO Main Board [ONLINE]. Available at: <http://sytechdesigns.com/graphics/Sytech/Gadgeteer/NanoTop01Med.png> [Accessed 26 April 2016].
  - [36] Richest Elektronik, (2011), ARDUINO UNO [ONLINE]. Available at: <https://www.reichelt.com/Single-board-microcontroller/ARDUINO-UNO/3/index.html?ACTION=3&GROUPID=6667&ARTICLE=119045> [Accessed 26 April 2016].
  - [37] Sparkfun, (2010), SparkFun Triple Axis Accelerometer - ADXL345 [ONLINE]. Available at: <https://solarbotics.com/product/50656/> [Accessed 26 April 2016].
  - [38] Raspberry Pi, (2010), Raspberry Pi [ONLINE]. Available at: <http://www.pcworld.com/article/2886260/raspberry-pi-2-review-the-revolutionary-35-micro-pc-supercharged.html> [Accessed 26 April 2016].
  - [39] P. Lindstedt and J. Burenus, Skapa oöverträffat kundvärde. Ödeborg: Nimba, 1999.
  - [40] Per Lindstedt, 2003. The Value Model : How to Master Product Development and Create Unrivalled Customer Value. Edition. Nimba, NA.
  - [41] spminstrument.se. 2002. Intellinova. [ONLINE] Available at: <http://www.spminstrument.se/Produkter-och-tjanster/Kontinuerlig-overvakning/Intellinova/>. [Accessed 3 May 2016].
  - [42] spminstrument.se. 2002. MG-4. [ONLINE] Available at: <http://www.spminstrument.se/Produkter-och-tjanster/Kontinuerlig-overvakning/MG-4/>. [Accessed 3 May 2016].

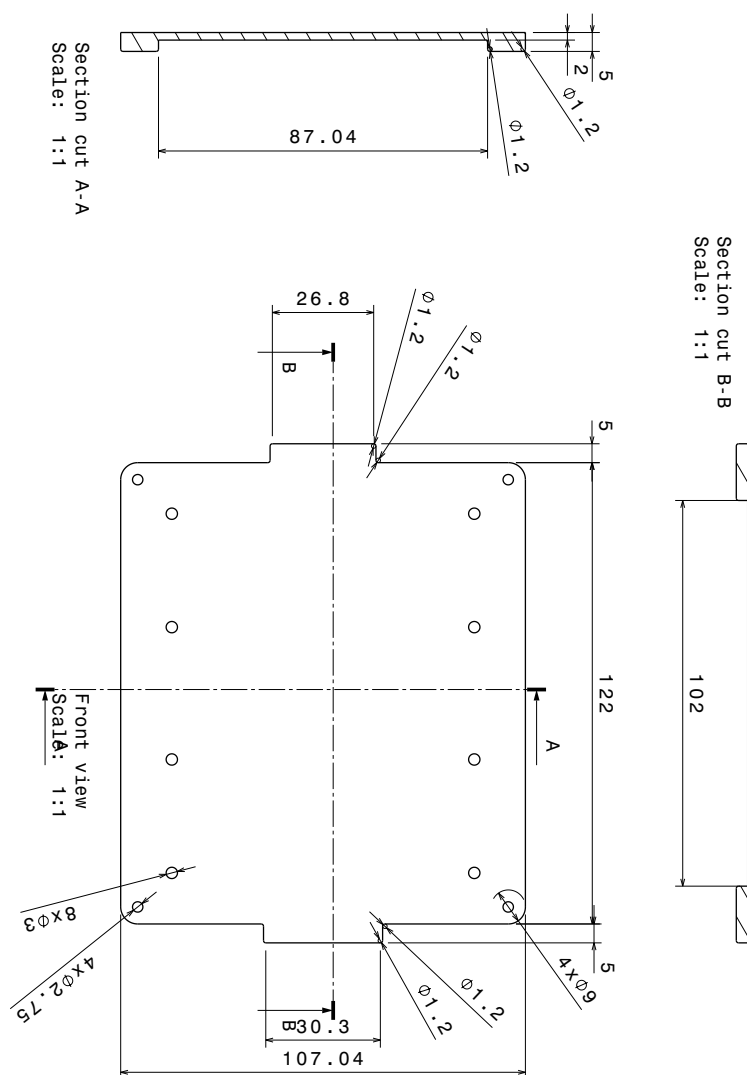
- [43] metix.se. 2010. Expert Vibro - Vibrationsövervakning. [ONLINE] Available at: [http://www.metix.se/produkter/datalogger/delphin/expertvibro?gclid=Cj0KEQjwl-e4BRCwqeWkv8TWq0oBEiQAMocbP2J6QF6NelAUT03o\\_1ndmqnNAH20\\_b1Vp5420mltYdcaAmSI8P8HAQ](http://www.metix.se/produkter/datalogger/delphin/expertvibro?gclid=Cj0KEQjwl-e4BRCwqeWkv8TWq0oBEiQAMocbP2J6QF6NelAUT03o_1ndmqnNAH20_b1Vp5420mltYdcaAmSI8P8HAQ). [Accessed 3 May 2016].
- [44] kimo.se. 2004. Bearing Checker. [ONLINE] Available at: [http://www.kimo.se/vibrationsmatare\\_Bearing\\_Checker\\_\(81001\).htm?meny=handinstrument](http://www.kimo.se/vibrationsmatare_Bearing_Checker_(81001).htm?meny=handinstrument). [Accessed 3 May 2016].
- [45] Lee, J., Seo, S., Choi, M., Hwang, Y., Lee, T. & Park, S. 2015, Design and implementation of IoT-based intelligent condition management system for the industrial facilities"
- [46] Internet of Things. IoT Infrastructures: First International Summit, IoT360 2014, Rome, Italy, October 27-28, 2014, Revised Selected Papers, Part II, 2015, 2015th edn, Springer International Publishing, Cham.
- [47] Monares, a., Ochoa, S.F., Santos, R., Orozco, J. & Meseguer, R. 2014, Modeling IoT-Based Solutions Using Human-Centric Wireless Sensor Networks", Sensors, vol. 14, no. 9, pp. 15687-15713.
- [48] glamox.com. 2015. IP klasser. [ONLINE] Available at: <http://glamox.com/se/ip-klasser>. [Accessed 6 May 2016].
- [49] schaeffler.com. 2015. FAG SmartCheck. [ONLINE] Available at: [http://www.schaeffler.com/remotemedien/media/\\_shared\\_media/08\\_media\\_library/01\\_publications/schaeffler\\_2/tpi/downloads\\_8/tpi\\_214\\_de\\_en.pdf](http://www.schaeffler.com/remotemedien/media/_shared_media/08_media_library/01_publications/schaeffler_2/tpi/downloads_8/tpi_214_de_en.pdf). [Accessed 6 May 2016].
- [50] Wilson, C., ScienceDirect (e-book collection), Books24x7 (e-book collection) & Books24x7, I. 2013, Brainstorming and beyond: a user-centered design method, Morgan Kaufmann, Amsterdam.
- [51] brainstorming.co.uk. 2012. Preparing for a successful brainstorming session. [ONLINE] Available at: <http://www.brainstorming.co.uk/tutorials/preparingforbrainstorming.html>. [Accessed 9 May 2016].
- [52] kursnavet.se. 2009. Flödesschema. [ONLINE] Available at: [http://www.kursnavet.se/kurser/Programmering%20A-C/htm/10\\_flodesscheman/10\\_flodesschema.html](http://www.kursnavet.se/kurser/Programmering%20A-C/htm/10_flodesscheman/10_flodesschema.html). [Accessed 8 May 2016].
- [53] Documents - Networking - GHI Electronics. 2016. Documents - Networking - GHI Electronics. [ONLINE] Available at: <https://www.ghielectronics.com/docs/30/networking>. [Accessed 09 May 2016].
- [54] kjell.com. 2015. Raspberry Pi. [ONLINE] Available at: <http://www.kjell.com/se/fraga-kjell/teman/raspberry-pi>. [Accessed 8 May 2016].
- [55] NXP. 2016. NXP Semiconductors. [ONLINE] Available at: [http://www.nxp.com/files/sensors/doc/data\\_sheet/MMA8453Q.pdf](http://www.nxp.com/files/sensors/doc/data_sheet/MMA8453Q.pdf). [Accessed 10 May 2016].
- [56] Shahram Dustdar and Wolfgang Schreiner. 2005. A survey on web services composition. [ONLINE] Available at: [http://www.infosys.tuwien.ac.at/staff/sd/papers/A%20survey%20on%20web%20services%20composition\\_Dustdar\\_Schreiner\\_inPress.pdf](http://www.infosys.tuwien.ac.at/staff/sd/papers/A%20survey%20on%20web%20services%20composition_Dustdar_Schreiner_inPress.pdf). [Accessed 12 May 2016].

- [57] Exforsys. 2016. The Use of Trial and Error To Solve Problems. [ONLINE] Available at: <http://www.exforsys.com/career-center/problem-solving/the-use-of-trial-and-error-to-solve-problems.html>. [Accessed 20 May 2016].



# A

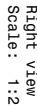
## CAD-ritning bottenplatta, tidigare system



**Figur A.1:** Ritning bottenplatta för inkapsling



## B

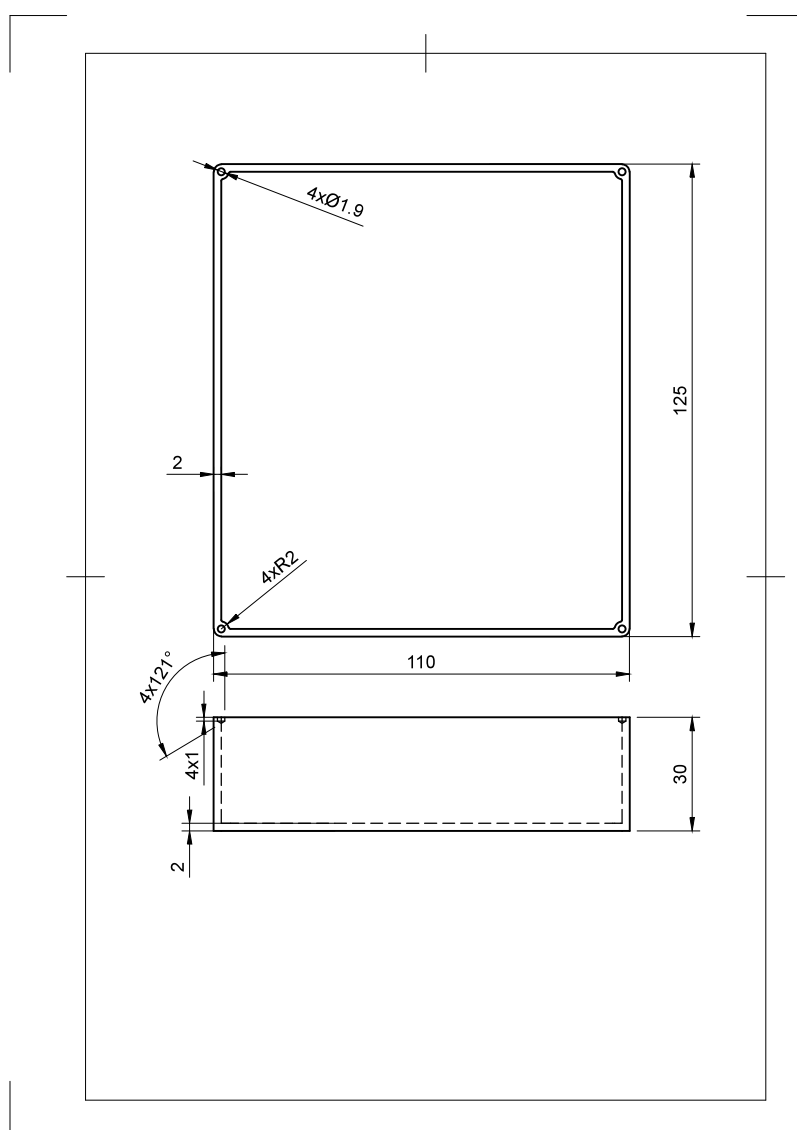


**Figur B.1:** Ritning bottenplatta för inkapsling



# C

## CAD-ritning inkapsling, nytt system

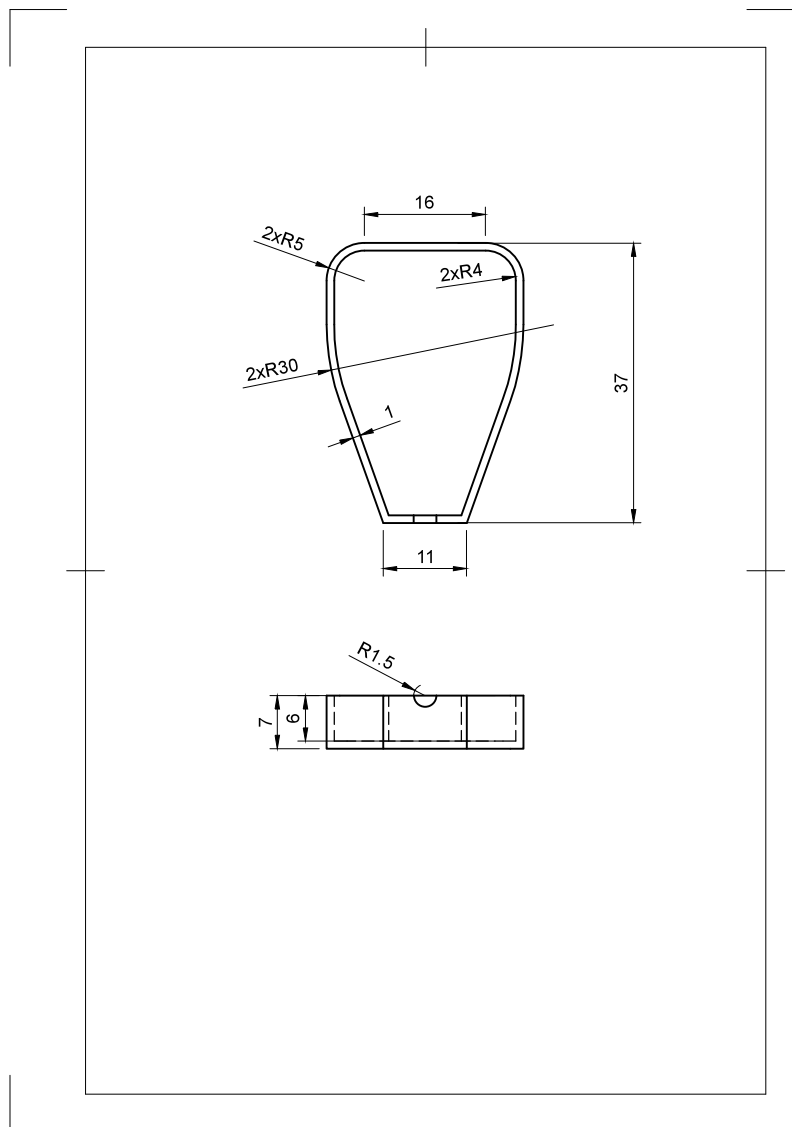


**Figur C.1:** Ritning över inkapsling för mätenheten.



# D

## CAD-ritning inkapsling för sensor, nytt system



**Figur D.1:** Ritning över inkapsling för sensor.