

Examensarbete, Högskolan på Åland, Utbildningsprogrammet för Informationsteknik

# DIGITALISERING AV KOMMUNIKATION

## - Elentreprenörens kommunikation med Mariehamns Energi

Dick Westerback



2022:19

Datum för godkännande: 18.05.2022  
Handledare: Björn-Erik Zetterman

# EXAMENSARBETE

## Högskolan på Åland

<b>Utbildningsprogram:</b>	Utbildningsprogrammet för informationsteknik
<b>Författare:</b>	Dick Westerback
<b>Arbetets namn:</b>	Digitalisering av kommunikation Elentreprenörens kommunikation med Mariehamns Energi
<b>Handledare:</b>	Björn-Erik Zetterman
<b>Uppdragsgivare:</b>	Mariehamns Energi

<b>Abstrakt</b>
<p>I detta examensarbete beskriver jag den web-portal för entreprenörer(samarbetspartners) som jag utvecklat i en L/WAMP stack. Bakgrunden till systemet var ett behov från uppdragsgivarens sida att digitalisera, samt ett nära samarbete mellan Mariehamns Energi och Högskolan på Åland. Arbetet berör säker identifikation samt lagring av data med två olika kategorier av användare, entreprenörerna och administratör/handläggare.</p> <p>Resultatet av detta examensarbete resulterade i en fungerande version där säker identifikation och lagring av data sker.</p>

<b>Nyckelord (sökord)</b>
LAMP, WAMP, PHP, MySQL, OAuth, OIDC, Apache, Javascript, JQuery, XAMPP, JWT, MVC

<b>Högskolans serienummer:</b>	<b>ISSN:</b>	<b>Språk:</b>	<b>Sidantal:</b>
2022:19	1458-1531	Svenska	32 sidor

<b>Inlämningsdatum:</b>	<b>Presentationsdatum:</b>	<b>Datum för godkännande:</b>
24.4.2022	13.5.2022	18.05.2022

# DEGREE THESIS

## Åland University of Applied Sciences

<b>Degree Programme:</b>	Education program for information technology
<b>Author:</b>	Dick Westerback
<b>Title:</b>	Digitalization of communication el-entrepreneurs communication with Mariehamns Energi
<b>Academic Supervisor:</b>	Björn-Erik Zetterman
<b>Commissioned by:</b>	Mariehamns Energi

<b>Abstract</b>
<p>In this thesis I write about the entrepreneur webportal, developed according to a L/WAMP stack. The background for this thesis was a need from the commissioner and an already working relationship with them. The webportal handles secure authentication and saving of data with two different user categories; entrepreneurs and administrators.</p> <p>The result of the webportal was a working version that handles secure authentication for entrepreneurs and storage of the data.</p>

<b>Keywords</b>
LAMP, WAMP, PHP, MySQL, OAuth , OIDC, Apache, Javascript, JQuery, XAMPP, JWT, MVC.

<b>Serial number:</b>	<b>ISSN:</b>	<b>Language:</b>	<b>Number of pages:</b>
2022:19	1458-1531	Swedish	32 pages

<b>Handed in:</b>	<b>Date of presentation:</b>	<b>Approved:</b>
24.4.2022	13.5.2022	18.05.2022

# INNEHÅLLSFÖRTECKNING/TABLE OF CONTENTS

<b>1. INLEDNING</b>	<b>6</b>
<b>1.1 Bakgrund</b>	<b>6</b>
<b>1.2 Syfte</b>	<b>6</b>
1.3 Metod	7
<b>2. VERKTYG OCH RAMVERK</b>	<b>8</b>
2.1 XAMPP	8
2.1 Linux	8
2.2 Apache	9
2.3 MySQL	9
2.4 PHP	9
2.5 JavaScript	9
2.6 jQuery	10
2.7 Windows - WAMP	10
2.8 JWT	10
2.8 MVC-ramverk	12
2.8.1 Model	12
2.8.2 View	12
2.8.3 Controller	13
2.9 OAuth 2.0	13
<b>3. OIDC - OPENID CONNECT</b>	<b>14</b>
<b>4. KRAVSPECIFIKATION</b>	<b>16</b>
4.1 Skall krav - initial	16
4.2 Skall krav - tillägg	16
4.3 Funktionalitetskrav som tillför funktionalitet	16
<b>5. IMPLEMENTERING</b>	<b>18</b>
5.1.1 MVC	18
5.1.2 Javascript	18
5.1.3 jQuery	19
5.2 Användargränssnitt - UI	20
5.2.1 Login	20
5.2.2 Formulären	21
5.2.3 Huvudmenyn	23
5.2.4 Bilagor	25

5.3 Backend	26
5.3.1 PHPMailer	28
5.3.2 LAMP till WAMP	28
5.4 Databasen	29
<b>6. SLUTSATSER</b>	<b>30</b>
<b>KÄLLFÖRTECKNING</b>	<b>31</b>
<b>BILAGOR</b>	<b>33</b>
<b>PROTOKOLL IBRUKTAGNINGSBESIKTNING FÖR ELINSTALLATION</b>	<b>33</b>

# 1. INLEDNING

## 1.1 Bakgrund

Bakgrunden till detta examensarbete började som ett kurs-projekt inom kursen programkonstruktion och projekthandledning (PoP) som pågick under våren 2021. Under denna kurs hade Mariehamns Energi AB ett beställningsarbete där deras befintliga pappersblanketter behöver digitaliseras.

Här är ett sammandrag från Mariehamns Energis IT-ansvariga över varför de valde att digitalisera sina besiktningsskyltar samt varför de valde att rikta sig till Högskolan på Åland och PoP-kursen.

*”Mariehamns Energi Ab såg ett behov av att digitalisera ärendehantering av el-entreprenörernas besiktningsskyltar i en sk entreprenörs portal som även skulle underlätta inlämnande av besiktningsskyltar, handlingar som rör objekten samt ritningar. Behovet av återkoppling till entreprenörerna för ärendehanterings gången hade länge efterfrågats och kunde med denna digitala portal även det lösas och tillgodoses. För att lyckas med detta att skapa en smidig digital plattform så krävdes det då en väletablerad autentiseringsmetod för brukarna.*

*Mariehamns Energi Ab har under många år samarbetet med HA kring examensarbeten och projektkurser, inom både energi- och IT-området. Därför föll det sig naturligt att vända oss till HA för att utreda möjligheterna, utvecklingen och färdigställandet av denna entreprenörsportal.”*

## 1.2 Syfte

Syftet med digitaliseringen är att entreprenörerna fylla i blanketter som behöver granskas av tjänsteman på Mariehamns Energi, för att ansluta en fastighet till elnätet. Genom att digitalisera processen kan entreprenören föra in sina formulär/dokument samt information för att få återkoppling, och att Mariehamns Energi har informationen i digital form för att möjliggöra digital transformation i nästa steg i processen. Även kommunikation mellan entreprenör och tjänsteman blir mer strukturerad och på så vis lättare att kvalitetssäkra.

Traditionellt har elentreprenören behövt ha med sig papper och penna för att sedan leverera det i pappersform eller elektroniskt via e-post. Detta projekt blev ett större arbete och kursen tog slut innan en komplett leverans kunde ges till kunden och det fortsatta arbetet blev mitt examensarbete samt praktik som utfördes under sommaren och hösten 2021.

### **1.3 Metod**

Projektet började med ett inledande möte med kunden (Mariehamns Energi AB:s IT-ansvariga samt besiktning-ledaren) där projektets ramar ingick och vad som behövdes digitaliseras, i huvudsak ibruktagning-besiktningarna som finns som PDF eller Word-dokument i dagsläget samt att det ska ingå en typ av bank-identifikation i projektet för säkerheten.

Efter det första inledande mötet med kunden tog gruppen fram olika versioner av hur formulären skulle tänkas se ut plus att hitta en lämplig aktör för hanteringen av identifikation av aktörer. Identifieringen behövde hålla juridiskt, som en signering av ett dokument, det var därför bank-identifiering undersöktes. Efter att vi hade en preliminär demo för formuläret i den föregående kursen, har jag i detta examensarbete förverkligat bank-identifikationen. Sedan gick gruppen inklusive besiktning-ledaren, IT-ansvariga samt utvecklarna genom formulären/processen med beställaren för att se ifall det behövdes ändringar på formulären samt ifall den valda aktören för bank identifikation var acceptabel för kundens behov. Eftersom kunden accepterar demonstrationen för både formulären och identifikationen skedde arbetet löpande där kunden blev informerad över projektets framfart samt när beställaren behövde agera på företagets vägnar vid t ex. signeringar av avtal.

## 2. VERKTYG OCH RAMVERK

I detta kapitel går jag igenom vilka verktyg som valdes för utveckling och varför just de verktyg användes i projektet.

Projektet är gjort i en *LAMP (Linux, Apache, MySQL, PHP) stack* från början av projektet eftersom det är en av teknikerna som undervisas på högskolan i t ex. serverteknik-kursen.

### 2.1 XAMPP

XAMPP som står för Cross Platform, Apache, MySQL Php Pearl är en *open-source*, (*open-source* eller öppen källkod tillåter användare att gratis använd, redigera och ändra i programvara inom license ramar) -plattform som låter användaren skriva, testa och köra sin kod på en lokal webbserver, skapad av Apache Friends.

Denna miljö användes i detta examensarbete för att kunna testa lösningar på diverse problem för att sedan sätta dem i en produktionsmiljö och för att bygga databasen för ibruktagnings-besiktningarna. (*What Is XAMPP?*, 2021)

### 2.1 Linux

Linux är ett *open-source*-operativsystem som först släpptes under 1990 talet. Eftersom Linux finns som *open-source* existerar många olika versioner av operativsystemet, vilka kallas för “distributioner”.

Några av dem är bl.a. Linux Mint, Manjaro, Debian och Ubuntu utöver att Linux är *open-source* är det även gratis att använda gentemot Windows eller iOS (*What Is Linux?*, 2018).



## 2.2 Apache

Apache HTTP-server är den mest använda webbservern i Linux, där den hanterar förfrågningar som användaren av webbsidor begär. Apache webserver används oftast i kombination med MySQL och PHP, Perl eller Python som är de mest valda alternativen i en LAMP server stack (Fielding & Kaiser, 1997).

## 2.3 MySQL

Databasen som valdes är MySQL som utvecklas av Oracle. MySQL tillhör LAMP stacken, då M i "LAMP" står för MySQL. MySQL är relationsdatabas, där olika databaser kan innehålla olika tabeller. En tabell kan ha fält av olika typer. Utöver det finns även ett rättighets/behörighetshantering-system inbyggt i databasservern. Det finns olika lagringssystem såsom Performance Schema, InnoDB, MRG MYISAM, BLACKHOLE och MyISAM (*MySQL :: MySQL 8.0 Reference Manual :: 16 Alternative Storage Engines*, n.d.) (*What Is MySQL? Everything You Need to Know*, n.d.).

## 2.4 PHP

PHP som står för Hypertext Preprocessor är ett *open source server side scripting* språk som används inom webbutveckling och stöder inbäddad HTML PHP tillåter injektion av t ex. HTML i PHP filer eller injektion av PHP i HTML, (*What Is PHP?*, n.d.) P ett i LAMP står även för PHP.

## 2.5 JavaScript

JavaScript är ett scripting- eller programmeringsspråk som låter dig använda komplexa funktioner, som att visa kartor, tids-uppdateringar, 2D- eller 3D-grafik, tillåter animeringar m.m. JavaScript används oftast i samband med andra standard-webbutvecklings-teknologier nämligen HTML och CSS (Harder, 2018) JavaScript har hundratals bibliotek för olika användarområden såsom AngularJS, Express och Vue.

## 2.6 jQuery

jQuery är ett snabbt, litet och funktionsrikt JavaScript-bibliotek. jQuery tillåter användaren att manipulera HTML dokument, eventhantering, animationer och ajax-anrop med ett API anrop som kan användas på de mest använda webbläsarna. Utöver jQuery existerar även jQueryUI och jQuery Mobile (JS Foundation-js. foundation, n.d.).

## 2.7 Windows - WAMP

WAMP är motsvarigheten till LAMP men den kör på Windows istället för Linux. Precis som LAMP så står AMP för Apache, MySQL och PHP. Där alla fyra av dessa verktyg används tillsammans så kallas det för en *WAMP stack* vilket tillåter grundläggande tjänster för webbtjänster, med en relationsdatabas (se 2.3) och en kod-motor (se 2.4) (Wang, 2011).

## 2.8 JWT

JWT eller *Json Web Tokens* innehåller krypterad information om användaren efter en godkänd inloggning. Med hjälp av [jwt.io](http://jwt.io) kan man dekryptera information eller hitta bibliotek för olika programmeringsspråk som hjälper med att dekrypterar en JWT med hjälp av en JWK (Json Web Keys). En JWT ser ut enligt figur 1:

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IjE6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MzI1MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c
```

Figur 1, En JWTsträng

Algorithm RS256

### Encoded

PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoiNjUwMDEyMzQ1NiJ9.NHVaYe26Mbt0YhSKkoKYdFVomg4i8ZJd8_-RU8VNbftc4TSMb4bXP313Y1NWAcwyXPGffz5aXhc61ty1Y2t4SWRqGteragsVdZufDn5B1nJ19pdR_kdVfUsra2rWKEofkZeIC4yWytE58sMIihvo9H1ScmmVwBcQp6XETqYd0aShp1g0a9RdUPDvoXQ5oqygTqVtxaDr6wUfKrKIgtGBMzWIdNZ6y709E0DhEPTbE9rfBo6KTFsHAZnMg4k68CDp2woYIaXbmYTWcvbzIuH07_37GT79Xdiwkm95QJ7hYC9RiwrV7mesbY4PAahERJawntho0my942XheVLmGwLMBkQ
```

### Decoded

EDIT THE PAYLOAD AND SECRET

```
HEADER: ALGORITHM & TOKEN TYPE
{
  "alg": "RS256",
  "typ": "JWT"
}

PAYLOAD: DATA
{
  "sub": "1234567890",
  "name": "John Doe",
  "admin": true,
  "iat": 1516239022
}

VERIFY SIGNATURE
RSASHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  -----BEGIN PUBLIC KEY-----
  MIIBIjANBgkqhkiG9w0BAQEFAAOC
  AQ8AMIIBCgKCAQEAu1SU1LFVLPHC
  ozMxH2Mo
  -----BEGIN PRIVATE KEY-----
  MIIEvwIBADANBgkqhkiG9w0BAQEF
  AASCBAQkkggS1AgEAAoIBAQC7VJTU
  t9Us8cKj
  MzEFYyjiWA4R4/M2bS1GB4t7Nxp9
)
```

Signature Verified

SHARE JWT

Figur 2. en JWT encoded och decoded (Jwt.Io, n.d.)

JWT:n ovan är krypterad och uppdelad i tre olika delar, en **header**, en **payload** och en **verify signature**, med en punkt som separerar dem. **Header** innehåller en algoritm och en typ, **payload** innehåller attribut så som sub, name, iat. Verify signature som används för att försäkra att användarens JWT inte har blivit ändrad i något skede (Hjlp: Doc: RFC 7523: JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants, n.d.).

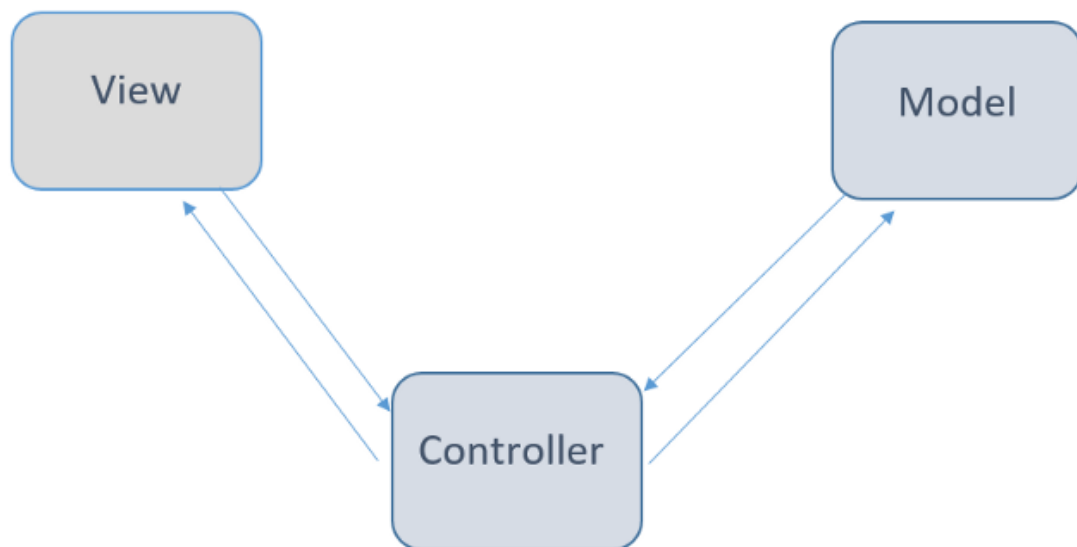
JWT:n verifieras mot följande element, där den måste bli godkänd i följande moment (Auth, n.d.).

1. iss, sso.example.com
2. sub, subject
3. aud, audience
4. exp, expiration

5. nbf, not before
6. iat, issued at

## 2.8 MVC-ramverk

MVC-ramverket är ett känt arkitektur mönster, som består av tre olika delar, Model, View och Controller. Det kan också ses som tre olika logiska delar. En av orsakerna till varför MVC ramverket används är för att det tillåter en standard-struktur för hur det är möjligt att strukturera en webbapplikation (Svirca, 2020). I figur 3 är en bild på hur MVC-ramverket fungerar.



Figur 3. MVCArkitektur och hur det kommunicerar.

### 2.8.1 Model

Model-delen av MVC-ramverket är den lägsta nivån av ramverket vilket betyder att Model-delen har som uppgift att underhålla informationen i databasen. Addera, uppdatera och raderingen av data är även det gjort från Model-delen. Model svarar på Controllerns förfrågningar eftersom Controller inte kommunicerar med databasen eller med Model (Svirca, 2020).

### 2.8.2 View

Data representeras i View-delen av ramverket, View genererar ett UI eller användar-gränssnitt för användaren. En View-komponent är alltså likt HTML/CSS. View är

skapad med data som hämtas från Model-komponenten, som visas i en View-komponent genom Controller eftersom informationen som Model har hämtat inte kommunicerar med View utan endast med Controllern (Svirca, 2020).

### **2.8.3 Controller**

Controllern är huvudet och hjärnan i ramverket, Controllern tillåter kommunikation mellan Model och View komponenten, då information måste hämtas från Model komponenten så pekar Controllern på Model, och den returnerade datan blir skickad till View komponenten, en Model och View kan således inte kommunicera med varandra utan de behöver en Controller som dirigerar dem (Svirca, 2020).

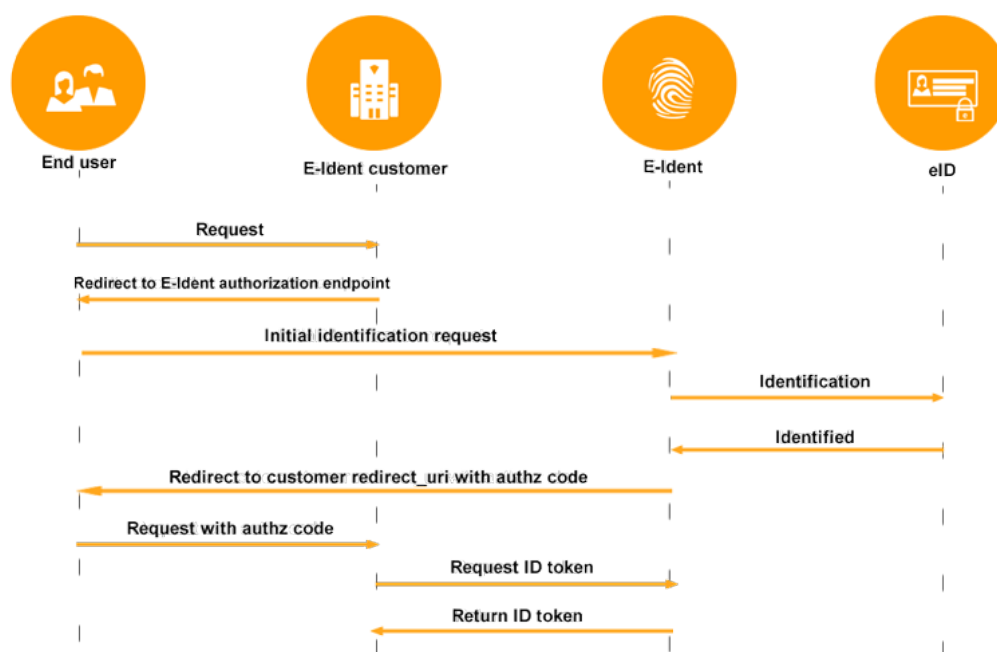
## **2.9 OAuth 2.0**

OAuth 2.0 är ett ramverk för identitet och åtkomst, som tillåter en tredje parts tjänst att nå begränsad åtkomst till en HTTP tjänst, antingen på resurs ägarens (,resursägaren är en entitet som kan ge åtkomst till en skyddad resurs, ifall resursägaren är en person kallas den för end-user). Eller en resursbegäran genom att en interaktion med resursägaren och den HTTP tjänst som anropas eller genom att tillåta en tredjepartsapplikation att själv få åtkomst till resursen (*RFC 6749 - the OAuth 2.0 Authorization Framework*, n.d.).

### 3. OIDC - OPENID CONNECT

OIDC eller OpenID Connect är ett enkelt lager ovanpå OAuth 2.0 protokollet, vad är då OAuth 2.0, OAuth 2.0 som står för *Open Authorization* är en standard som är designad för att nå resurser som är existerar på andra webbsidor på användarens begäran med hjälp av diverse *url endpoint* anrop (*What Is OAuth 2.0 and What Does It Do for You?*, n.d.). OIDC tillåter användaren att verifiera information om slut-användaren dvs. den fysiska person som ska identifiera sig. I detta fall är det entreprenören som ska verifiera att de är vem de säger att de är genom att använda sig av sitt bank ID.

Figur 4 förtydligar hur identifikationen sker mellan användaren, servern och aktören.



Figur 4. Hur identifikationen sker (Siriwardena, 2014).

- **End user**, el-entreprenör.
- **E-Ident customer**, Mariehamns energi's entreprenörs portal
- **E-Ident**, den tjänst som identifierar slut användaren.
- **eID**, en bank t ex. Ålandsbanken.

När entreprenören har loggat in på sin valda bank från login sidan blir de skickad till kundsidan, authz-koden fångas upp och med hjälp av den kan servern nå ID token eller

JWT:n som är krypterad, med hjälp av JWK (Json Web Keys) från den givna aktör kan man dekryptera och verifiera din JWT och ifall verifikationen inte lyckas så returneras entreprenören till login sidan med ett felmeddelande, om inloggning lyckas skickas entreprenören till huvudsidan.

## 4. KRAVSPECIFIKATION

I detta kapitel så sammanställs kravspecifikationen. Vilka krav som sattes som bas-krav i början av projektet samt hur kravspecifikationen har växt i och med projektets bredd har blivit större samt att nya idéer har tillkommit via testare och genom användning av test portalen.

### 4.1 Skallkrav - ursprungliga

Dessa krav var de ursprungliga som gavs vid projektkursen.

1. Ibruktagnings-besiktningensformulär i digital form
2. Fastighets besiktningensformulär i digital form
3. Användarverifiering skall ske med säker identifiering (bankkoder)
4. Admin/back office funktionalitet ska finnas som en konsekvens av krav 1
  - a. Verifiering av tjänsteman
  - b. Bedömning av ibruktagningen
  - c. Redigering för slutbehandling
5. Kommunikation/Notifiering till tjänsteman då ärende anländer, via e-post.

### 4.2 Skallkrav - tillägg

Dessa krav är för arbetet som gjordes för examensarbetet.

6. Ladda upp/lägg till bilagor till en blankett/anmälan
7. Fastighets besiktningens formuläret valdes bort, krav 2 utgick
8. Blankett och funktionalitet för register-besiktningar lades till
9. Automatisk utloggning vid inaktivitet längre än 15 minuter
10. Radering av bilagor under arbete(entreprenören).
11. Navigering i systemet enligt skild specifikation

### 4.3 Funktionalitetskrav som tillför funktionalitet

1. Hantering av dubbeletta bilagor dvs. att två bilder får samma namn.
2. Hashade lösenord för admin-användaren
3. Dynamiska sökfält.
4. Validering av inputdata.



5. Möjlighet att välja flera alternativ i blanketter/formulär samt att spara dem i databasen i samma kolumn.
6. E-post fält i alla formulär för att kunna använda sig av PHPMailer för att skicka.
7. En test portal för att testa implementeringar före de används i produktion.

## 5. IMPLEMENTERING

Implementeringen gjordes i huvudsakligen i två steg, det första steget involverade en lokal XAMPP instans som används före servern där ibruktagnings besiktningens formuläret byggdes upp sektion för sektion, med en nära feedback från både IT-ansvarig, besiktning- och drift-ledarna. I den processen uppstod kravet att ladda upp filer och lades till i kravspecifikationen, då det visade sig vara ett uppenbar del av projektet att det ska kunna vara möjligt att ladda upp bilagor.

De första veckorna arbetades det mest med *backend* alltså det som sker i portalen som användaren inte kan se samt med databasen för att få allt att fungera i tid med höstens preliminära *deadline*. Har arbeta mycket med UI (Användargränssnitt) och testade lite olika idéer för hur det grafiska gränssnittet ska se ut.

### 5.1.1 MVC

Då kodandet började så valdes det att bygga enligt ett MVC ramverk fast hemmabyggt. Detta för att kunna dela upp koden i tre olika delar så det blir lättare i längden att underhålla. Projektets undermappar blev då bland annat dessa, **Model**, **View** och **Controller**, en `index.php`-fil kallar på *controller*-filen som sedan renderar view-modellen med ett POST-anrop. När man skickar in formuläret så anropas Controllern igen fast då pekar den mot model-filen och där skrivs formuläret in i MySQL-databasen.

### 5.1.2 JavaScript

JavaScript används även i projektet för att kunna skapa interaktivitet, förenkla koden t ex. Istället för en lista i HTML format skapades en *array* i JavaScript som innehåll, alla av Ålands kommuner som sedan kunde skickas in till t ex. `besiktning.php`. JavaScript används också för en lösning där de standardiserade svaren “-”, “+” och “0” visades i varje kryssruta. Med hjälp av JavaScript görs en *array* av det så då behövdes inte samma del av koden upprepas om och om igen.

Ett visuellt val som gjorts i detta examensarbete är att använda JavaScript för att göra det möjligt att använda checkboxar i stället för radioknappar men att de uppfyller samma resultat. Med hjälp av JavaScript är det nu möjligt att välja en checkbox och endast en checkbox då ett alternativ ska väljas i en ifyllning av ett formulär.

### 5.1.3 jQuery

För att göra det enklare för användarna, både med att underlätta med sortering när formulären blir väldigt många, ex. 50-100 st alternativt ifall el entreprenörerna behöver hitta ett specifikt formulär så kan de söka på dess adress för att få ut alla formulär med adressen "Ålandsvägen 1", andra användningar av jQuery är bland annat plugin (för galleri av bilagor och *tool tipster*). Ett annat exempel på effektiviserad utveckling med jQuery är den automatiska utloggningen som kan ses i figur 5.



Figur 5 Inaktiv användare

## 5.2 Användargränssnitt - UI

När det kommer till UI hade jag ganska fria händer med designen men valde ett lättförståeligt gränssnitt med få knappar för att minska förvirring. Till stor del används färgerna svart, vitt

och samt nyanser av grått med inslag av rött efter Mariehamns Energis logo. UI i sig består av en login sida, olika formulär och ett galleri.

### 5.2.1 Login

*Login* består av två olika sidor en för el-entreprenörerna och en annan *login*-sida för administratörerna. El-entreprenörernas *loginsida* innehåller endast en beskrivande text över hur de ska göra för att kunna logga in med sina bankkoder. Sedan sköter aktören om identifikationen och skickar kunden till huvudsidan, se figur 6.

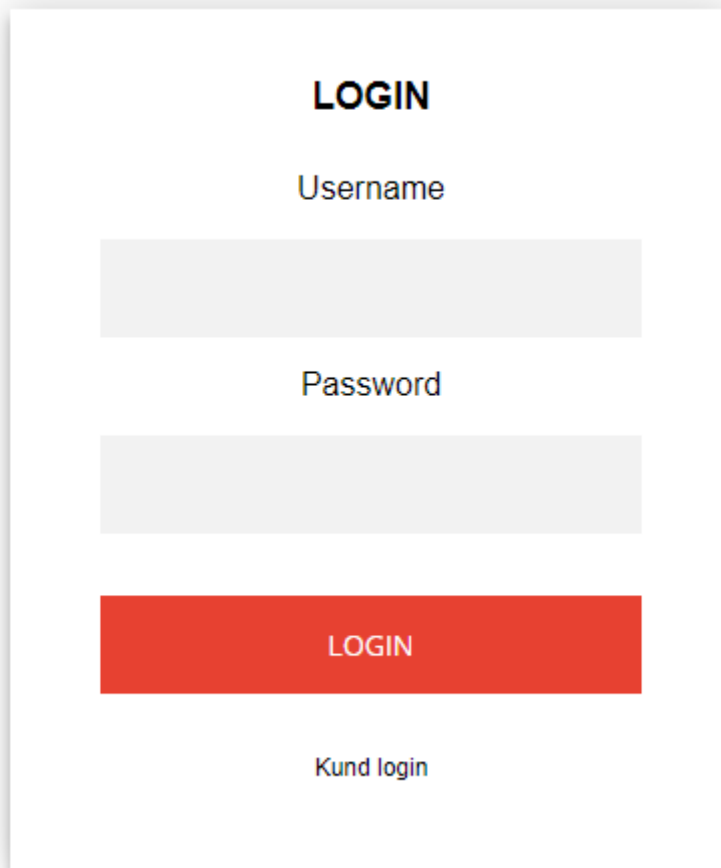
---

Ni kan logga in med era bankkoder och göra era pappers ärenden på kundportalen genom att klicka på den röda knappen



*Figur 6, Login*

Administratörernas inloggning är ett standard *login*formulär som skickar ett användarnamn och ett lösenord till databasen för att jämföra ifall de matchar. Ifall det är en match så tas de vidare till administratörernas huvudsida. Från administratörernas inloggningssida är det även möjligt att gå till el-entreprenörernas inloggningssida, se figur 7.

A login form titled "LOGIN" in bold black text. Below the title are two input fields: "Username" and "Password", each with a light gray rectangular placeholder. Below these fields is a red rectangular button with the text "LOGIN" in white. At the bottom of the form is a link labeled "Kund login" in blue text.

*Figur 7, Adminlogin*

### **5.2.2 Formulären**

Formulären förblev den största grafiska utmaningen. De utgår från de tidigare blanketterna, se BILAGA 1. Att visa formulären på ett lättförståeligt och skalbart sätt var inte helt enkelt då det gällde att testa sig fram med olika tekniker. Efter ett tips att använda *tableview* samt att jämföra resultatet på min första idé mot en *tableview* så ansågs det att *tableviews* var det bästa sättet att representera data i då det gick att dela upp formuläret i celler så undviks problemet med att det blir för rörigt då det var svårt att få in något som separerade de olika fälten samt fält texten i den första lösningen, se figur 8.

Figur 8, Ibruktagningsbesiktning Formulär

När ett formulär har fyllts i så körs ett jQuery-script som validerar att det finns information i de fält som är markerade som nödvändiga, namn på el entreprenörs firma, e-post, arbetets adress m.m. Då det gäller e-posten behöver den även vara i ett giltigt format. Ett formulär som inte är ifyllt korrekt skickas inte in till databasen och därav så lagras ingen data förrän skriptet godkänner formuläret, se figur 9.

Figur 9, ibruktagnings besiktningformulärvalidation

Efter en intern diskussion med personalen så valdes fastighetsformuläret bort som ett skalkrav och lades på så sätt åt sidan för att göra rum för registerformuläret som ansågs vara mer behövligt att få digitaliserat samt att det kom förfrågning på att ha med ett *tooltip* för att förklara vilken information som formuläret hänvisar till, se figur 10.

## 1. ARBETSOBJEKT

Egnahemshus, kontor etc.

Typ:

Figur 10, Tooltipster beskriver vilken information som krävs för fältet

### 5.2.3 Huvudmenyn

Huvudmenyn består av 3 huvuddelar där “headern” innehåller en variabel med namnet på den inloggade el-entreprenören, en *footer* som endast innehåller en knapp för att logga ut. Då returneras man till login sidan igen. Emellan dessa två är själva portalen som består av två stycken *accordions* (en per formulär) som går att öppna och stänga. Själva dragspelen har ett sökfält direkt under titeln. Detta används för att enkelt kunna hitta specifika formulär t.ex. de som berör en specifik gata (figur 11), person eller ifall bara de som inte är godkända ska visas.

När dragspelet är öppet så syns en beskrivning på formuläret samt alternativet att skicka in en ny besiktning. Av innehållet som skickas in till databasen så används namnet på kunden, adressen där arbetet utförs, resultatet för besiktningen samt det datum som formuläret skickades in på som alternativ för att söka bland de inskickade objekten.

Mariehamn	Entreprenör	Adress
	EI AB	Mariehamnvägen 22
	EI AB	Mariehamnvägen 22
	EI AB	Mariehamnvägen 22
	EI AB	Mariehamnvägen 22

Figur 11, Specifik sökning

När ett formulär har skickats in men inte ännu blivit godkänt av en administratör eller tjänsteman har el-entreprenörerna möjlighet att ladda upp ytterligare bilagor samt möjlighet att uppdatera formuläret. Genom att trycka på resultatet för ett individuellt formulär kan man få en mer beskrivande förklaring över varför besiktningen inte godkändes, t ex. att en bilaga saknades, se figur 12.

The screenshot shows the Mariehamns Energi customer portal. At the top, there is a navigation bar with the text "Välkommen Väinö Tunnistus" and the Mariehamns Energi logo. Below this, a message states: "Det här är Mariehamns Energis kundportal där du som kund hos oss ska kunna sköta allt pappers jobb via webben".

The main content area is titled "Dina Ibruktagningsbesiktningar" and contains a search bar and a list of inspection results. A warning message reads: "Här kan du se dina tidigare besiktningar från nyaste till äldsta, ifall du trycker på redigera så kan du redigera din Ibruktagningsbesiktning tills Mariehamns Energi godkänt den. OBS! När en besiktning är godkänd så kan ni inte längre redigera eller ladda upp bilagor. Genom att klicka på resultatet av besiktningen så kan ni få en detaljerad birst beskrivning".

Below the warning is a table with the following columns: Kund namn, Address, Resultat, Anlände, Bilagor, and Redigera. The table contains several rows of data, including entries for Ulla-Brit and Dick Westerback.

At the bottom of the page, there is a "Logga ut" button.

Figur 12, Huvudmenyn

Eftersom tanken med portalen är att det ska möjliggöra att entreprenörerna kan skicka in en besiktning på fältet, dvs. medan de utför besiktning i real-tid och eftersom upplägg med en lång rad av information inte är passande för mobilskärmar så sätts allt på höjden enligt figur 13 och givetvis fungerar sökfältet på samma vis.



Ibruktagningsbesiktningar

Sök på entreprenör

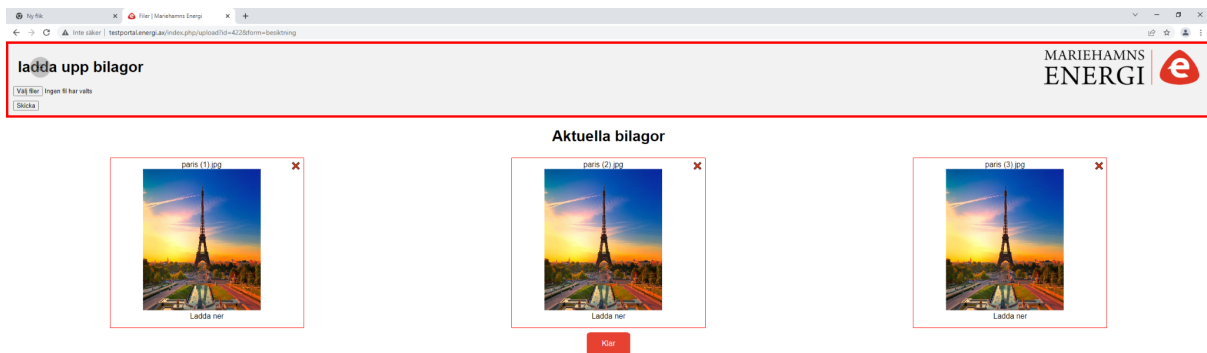
<b>Entreprenör</b>	EI AB
<b>Address</b>	Kumlinge 223
<b>Resultat</b>	uppnådd
<b>Anlände</b>	2021-12-21
<b>Bilagor</b>	Visa besiktning
<b>Redigera</b>	Visa filer
<b>Entreprenör</b>	EI AB
<b>Address</b>	Mariehamnvägen 22
<b>Resultat</b>	ouppnådd (bristerna i bilaga)
<b>Anlände</b>	2021-12-20
<b>Bilagor</b>	Visa besiktning
<b>Redigera</b>	Visa filer
<b>Entreprenör</b>	EI AB

Figur 13, Huvudmeny i mobilläge

#### 5.2.4 Bilagor

Bilagorna laddas upp i steg två av ifyllningen av ett formulär vilket betyder att formuläret skapas med ett visst ID samt med ett ID för användaren som fyllde i formuläret, med hjälp av dessa två variabler så är det möjligt att öppna en tom bilaga där användaren har möjlighet att ladda upp bilagor i olika bildformat, PDF- format eller som DXF-filer vilket är ett autocad-format vilket är ett av filformaten som programmet ska stödja.

Ifall användaren har laddat upp fel fil så är det möjligt att radera den från bilagorna genom att trycka på det röda krysset som finns i det övre högra hörnet av varje uppladdat fil. Då filen tas bort från bolagssidan används funktionen `free()` för att radera den valda filen från servern, filen tas bort permanent.



figur 14, Uppladdning av bilagor.

### 5.3 Backend

Backend delen av programmet är byggt med PHP med användning av funktionen **\$SESSION** vilket tillåter lagring av information för användare och deras ID för diverse formulär och bilagor för att sedan kunna tömma sessionen vid behov eller ge den ett nytt värde. Mappstrukturen är byggt enligt ett **Model View Controller** ramverk, som ett eget implementerat ramverk och inte något som redan existerade.

För *login* så fungerar det som följande; en el-entreprenör har gått genom inloggningen från loginsidan och auktoriserat sig med sina bankkoder. Aktören för bankinloggningen returnerar data som sedan med hjälp av variabler som *state*, *noun*, *client ID* och *client secret* går att ta ut en *ID token* som sedan måste avkrypteras. Ifall *state* och *noun* värdena inte är de exakt samma som de som angavs före inloggningen vid inloggningsidan så skickas entreprenören tillbaka till inloggningssidan med ett felmeddelande.

De samma händer ifall entreprenören inte lyckas auktorisera sig pga problem med JWT, *idcode* eller andra värden som inloggningen kräver. Efter en lyckad inloggning så sparas den information som behövs såsom namn och den inloggade personens unika id.

```

if(is_null($_SESSION["state"])) {
    $error = "Inloggningen har avbrutits eftersom din session inte längre är giltig";
    header("location: login?error=$error");
}
if($_SESSION["state"] != $_GET['state']) {
    $error = "Inloggningen har avbrutits eftersom din session inte längre är giltig";
    header("location: login?error=$error");
}

if(isset($_GET['error'])) {
    $error = "Inloggningen har avbrutits med orsaken: " . htmlspecialchars($_GET['error']);
    header("location: login?error=$error");
}

```

Figur 15, Exempel på felmeddelande för state

I och med att administratörerna inte behöver identifiera sig med bankid så är deras inloggning endast ett loginformulär som skickas från *View* delen av MVC ramverket vidare till *Model* med hjälp av *Controller*. Där skickas användarnamnet in i databasen för att hitta ett matchande värde i databasen. Ifall ett matchande användarnamn hittas så returneras lösenordet och användarnamnet, men eftersom lösenordet är hashat i databasen pga. säkerhetsskäl så måste funktionen **password\_verify()** användas. Med hjälp av den funktionen så är det möjligt att jämföra ett lösenord som “apa” mot det hashade lösenordet som har det här formatet:

*\$2y\$10\$yqFvDGyv2Tz4d/A/yulbFe5ISH9oR3gvU7GQLMYRKR7XQJnGpQOau*

figur 16, ett hashat lösenord

Ifall användarnamnet inte finns i databasen returneras administratören till admin inloognings sidan med felmeddelande. Det samam händer även ifall lösenordet är fel men användarnamnet är rätt.

The image shows a login form with the following elements:

- LOGIN** (Section Header)
- Username label above a grey input field.
- Password label above a grey input field.
- Wrong password** (Error message in red text).
- A red **LOGIN** button.
- Kund login** (Link below the button).

*Figur 17, Felmeddelande vid login*

### **5.3.1 PHPMailer**

För att göra det möjligt för el-entreprenörerna att skicka in bilagor så måste de fylla i sin e-postadress. Med hjälp av det fältet är det möjligt att skicka ett ifyllt formulär till Mariehamns Energi och med hjälp av el-entreprenörernas e-post så är det möjligt för administratörerna att skicka ett svarsmejl till el-entreprenörerna när en bilaga är godkänd eller kräver komplettering.

### **5.3.2 LAMP till WAMP**

Eftersom projektet var byggt enligt en LAMPmodell så kördes servern på en Linuxmaskin hos kunden. Det blev senare valt i efterhand att övergå till en *WAMP*stack eftersom det är enklare att hantera en Windowsserver för kunden då det är de servrar som de använder internt.

## 5.4 Databasen

Projektet använder sig av PDO (*PHP Data Objects*), med en DB (*database*) *Handler* som initialiserar kopplingen när den behövs. Genom att kalla på en fil för databasens PDO objekt så undviks repetition av koden och ger en renare kod, ifall databaskopplingen inte lyckas så skrivs ett felmeddelande ut till användaren.

Databasen är som nämnt ovan en MySQL databas med tabeller för admin, besiktning, register och filer. För att ha en röd tråd att följa i ibruktagningsbesiktning under rubriken “Sensoriska besiktningar” (se bilaga 1) så används prefixet e, f, g, h etc. Detta för att få ett så tydligt som placering, konstruktion osv. upprepas på flera ställen i det fysiska formuläret.

Administratörskonton är även hashade genom PHP funktionen **password hash()** för att dölja det egentliga lösenordet då det inte går att hasha tillbaka lösenordet utan det är endast möjligt att kryptera lösenordet med denna funktion. Då det inte går att hasha tillbaka lösenordet är det nästintill omöjligt att lista ut lösenordet för inloggningen ifall databasen blir sårbar.

## 6. SLUTSATSER

Som slutsats för detta examensarbete kan jag konstatera att det har varit en utmaning att lära sig utveckla självständigt då nya tekniker krävts. Jag har arbetat med *full stack*, alltså både arbete med UI och *backend* samt att även arbetat lite med nätverk och serverar då *LAMP* och *WAMP* stacken skulle konfigureras för att stödja projektet. Jag har lärt mig mycket om auktorisering, JWT, identifikation och jQuery. Dessa är bara några av de verktyg som jag lärde mig använda.

Det kom även rikligt med feedback från de interna på Mariehamns Energi så där fick jag erfarenhet av utvecklingsledare även om jag var den enda utvecklaren under största delen av perioden. Erfarenhet av att möta upp och presentera samt att inse att alla projekt måste ha ett avslut och att inte implementera mer och mer vartefter. Slutresultatet blev en fungerande version 1.0. som uppfyllde de krav som fanns med i kravspecifikationen.

### 6.1.1. Vidare utveckling

Vidare utveckling på detta projekt kan vara bl.a. att överse databasen, koden samt att optimera koden ifall det finns möjlighet. Det finns ett bra upplägg för att introducera andra formulär till portalen ifall behovet uppkommer samt att ta in feedback från användarna av portalen för att få fram förbättringsförslag.

# KÄLLFÖRTECKNING

- Auth. (n.d.). *Validate JSON Web Tokens*. Auth0 Docs. Retrieved May 21, 2022, from <https://auth0.com/docs/secure/tokens/json-web-tokens/validate-json-web-tokens>
- Fielding, R. T., & Kaiser, G. (1997). The Apache HTTP Server Project. *IEEE Internet Computing*, *1*(4), 88–90.
- Harder, J. (2018). What Is JavaScript? In *Graphics and Multimedia for the Web with Adobe Creative Cloud* (pp. 881–898). Apress.
- Hjp: Doc: RFC 7523: JSON web token (JWT) profile for OAuth 2.0 client authentication and authorization grants. (n.d.). Retrieved April 26, 2022, from <https://www.hjp.at/doc/rfc/rfc7523.html>
- JS Foundation-js. foundation. (n.d.). *JQuery*. Retrieved April 19, 2022, from <https://jquery.com/>
- Jwt.Io. (n.d.). Retrieved May 21, 2022, from <https://jwt.io/>
- MySQL :: MySQL 8.0 reference manual :: 16 alternative storage engines. (n.d.). Retrieved April 19, 2022, from <https://dev.mysql.com/doc/refman/8.0/en/storage-engines.html>
- RFC 6749 - the OAuth 2.0 authorization framework. (n.d.). Retrieved April 26, 2022, from <https://datatracker.ietf.org/doc/html/rfc6749>
- Siriwardena, P. (2014). OpenID Connect. In *Advanced API Security* (pp. 181–200). Apress.
- Svirca, Z. (2020, May 29). *Everything you need to know about MVC architecture*. Towards Data Science. <https://towardsdatascience.com/everything-you-need-to-know-about-mvc-architecture-3c827930b4c1>
- Wang, N. (2011). *Building the WAMP Platform* [Mikkelin ammattikorkeakoulu]. <https://www.theseus.fi/handle/10024/32554>
- What is Linux?* (2018, August 20). Linux.com. <https://www.linux.com/what-is-linux/>

*What is MySQL? Everything You Need to Know.* (n.d.). Talend - A Leader in Data Integration & Data Integrity. Retrieved April 19, 2022, from <https://www.talend.com/resources/what-is-mysql/>

*What is OAuth 2.0 and what does it do for you?* (n.d.). auth0. Retrieved April 19, 2022, from <https://auth0.com/intro-to-iam/what-is-oauth-2/>

*What is PHP?* (n.d.). Retrieved February 18, 2022, from [https://www.php.net/manual/en/intro-whatis.php#:~:text=PHP%20\(recursive%20acronym%20for%20PHP,can%20be%20embedded%20into%20HTML.](https://www.php.net/manual/en/intro-whatis.php#:~:text=PHP%20(recursive%20acronym%20for%20PHP,can%20be%20embedded%20into%20HTML.)

*What is XAMPP?* (2021, August 2). EDUCBA. <https://www.educba.com/what-is-xampp/>



# BILAGOR

## PROTOKOLL IBRUKTAGNINGSBESIKTNING FÖR ELINSTALLATION

1. ARBETSOBJEKT			
Typ	Kund Mobil tel.		
Mätarplats nr	Adress Telefon		
2. ELENTREPRENÖR			
Namn	Registreradledareavelarbete		
Adress	Mobil tel. Telefon		
3. ELNÄTETS ÄGARE ÅEA Mariehamns Energi			
4. NOMINELL MÄRKSPÄNNING 5. KORTSLUTNINGSTRÖM VID ANSLUTNINGEN (minsta/största)			
6. GRUNDEN FÖR BESIKTNINGEN			
Nyinstallation Ändrings- eller förstöringsarbete Reparationsarbete Förnyad besiktning Annat, vad:			
Arbete:			
7. SENSORISK BESIKTNING			
+ i skick - något att anmärka 0 hör inte fil besiktningen			
I besiktningen konstateras sensoriskt de i punkt a till m nämnda ärendena I tillämpliga delar. Konstaterade brister meddelas på separat bilaga.			
a. Anslutningsledning			
ändbox skydd			
typ: area:			
b. Genomföring			
c. Huvudsäkringar			
Säkring / element x A / x A			
d. Jordelektrod Enligt SFS 6000-5-54			
i fundament	40m slinga	1st 20m i matningskabelns dike	elnätets jordningssystem
nära fundament	lodräta elektroder	2st 20m i olika riktningar	
Potentialutjämning			
PE- eller PEN-skena			

1. ARBETSOBJEKT	
Typ	Kund Mobil tel.
Mätarplats nr	Adress Telefon
betongarmering	fjärrvärmesät vattenledningsnät ventilationskanaler antennsystem telefonsystem åskskydd
e. Huvudcentral placering konstruktion märkningar fränskiljningsmöjlighet rotationsriktning installation	
f. Fördelningscentraler placering konstruktion märkningar montering rotationsriktning	
g. Gruppledningar anslutning till centralen areor märkningar montering	
h. Uttag placering konstruktion rotationsriktning ledningarnas anslutningar	
i. Armaturer placering konstruktion ledningarnas anslutningar	
j. Värmeanläggningar radiatoreernas bastuugnens värmefolie värmekabel installationer placering och placering och installationer montering montering värmepump installationer	
k. Övriga anläggningar mikroproducent spis laddstation elbil	

l. övriga installationer telefoninstallationer antenninstallationer övriga teletekniska installationer	
m. Slutritningar centralschemor ledningsschemor bruksanvisningar och skötseldirektiv	
INSTALLATIONERNA I SKICK Bilagor st.	
8. MÄTNINGAR CENTRALVIS	
a. Kontinuiteten hos skyddsledare och potentialutjämningsledare Kontinuiteten konstaterats genom mätning	

b. Isolationsresistans Hela centralens isolationsresistans MΩ					
Gruppledningar som bör mätas separat					
Grupp nr Isolationsresistans			Grupp nr Isolationsresistans		
c. Konstateras uppfyllande av kraven för automatisk fränkoppling					
Konstaterats genom mätning Konstaterats vid planering					
Den lägsta kortslutningsströmmen i grupper som mäts separat					
Grupp nr Ikmin/A Säkringstyp/storlek			Grupp nr Ikmin/A Säkringstyp/storlek		
d. Funktionsprovning för jordfelsbrytare					
Fabrikat, Typ, Märkning/beteckning Nominella värden In/IΔn Uppmätt funktionsström IΔ Utlösningstid [ms]					
e. Bilagor som hänför sig till mätningarna st.					
f. Mätutrustningar som använts: Instrumentet skall uppfylla standarden EN 61 557					
Utrustning				Tillverkare Typ	

### 9. Elbilsladdstation Installationen innefattar flera laddstationer se särskild bilaga [ ]

Märke och typ

Max effekt kW Laddaren är installerad enligt SFS-6000-7-722 ja

Gruppens isolationsresistans MΩ

Laddningsmetod Laddaren kan användas som tilläggskraftkälla till nätet ja

Funktionsprovning för jordfelsbrytare

Fabrikat, Typ, Märkning/beteckning Nominella värden In/IΔn Uppmått funktionsström IΔ Utlösningstid [ms]

--	--	--	--

### 10. EMC-SKYDDET

Objektet är ett TN-S system

Jordningar och potentialutjämnningar har konstaterats uppfylla EMC-kraven

Valet, placeringen och installationen av kablar har konstaterats uppfylla EMC-kraven

Vid val av elutrustning (elapparater) har man beaktat installationsomgivningens krav

Anvisningar från elutrustningens tillverkare har iakttagits

Annan, vad?

Bilagor: \_\_\_\_\_  
Elanläggningen uppfyller kraven i elsäkerhetslagen 1135/2016 och statsrådets förordning om den elektriska utrustningens och elanläggningarnas elektromagnetiska kompatibilitet (1436/2016)

### 11. BEHOVET AV SERVICE- OCH UNDERHÅLLSPROGRAM (Enligt SFS-6000-6-6.5)

Underhållsprogram för objektet krävs krävs inte

Objektet har service- och underhållsprogram

Objektet har drifts-, service- och underhållsanvisningar

Objektet har utrymningsbelysning

Objektet har underhållsprogram för utrymningsbelysningen

### 12. FÖLJANDE BESIKTING

Periodiskbesiktning: krävs tidpunkten för besiktningen  
krävs inte

Certifieringsbesiktning krävs bestämd tidpunkten för besiktningen  
krävs inte ej beställd

Obs!

För kraven om certifieringsbesiktning se Landskapslag om tillämpning på Åland av rikets elsäkerhetslag §3

### 13. BESIKTNINGENS RESULTAT

Enligt säkerhetsnivån i SFS 6000 uppnådd uppnådd (bristerna i bilaga)

Korrigeringsuppmaning givits Datum då befintliga brister måste vara avhjälpda:

**14. UTFÖRARE AV BESIKTNINGEN**

Namn

Tid och plats Underskrift

**15. ELENTREPRENÖR**

Namn

Tid och plats Underskrift