

---

# Rapport

---

EXAMENSARBETE ALTEN SVERIGE AB

MAX JOURDANIS  
max.jourdanis@gmail.com  
# 0730474197

HENNING URRUTIA MÅLQUIST  
malquisthenning@gmail.com  
# 0730474197

# Innehåll

<b>1</b>	<b>Inledning</b>	<b>1</b>
1.1	Bakgrund . . . . .	1
1.2	Mål . . . . .	1
1.2.1	Projektmål . . . . .	1
1.2.2	Slutmål . . . . .	2
1.3	Metod . . . . .	2
1.3.1	Verktyg . . . . .	2
1.3.2	Utvecklingsverktyg . . . . .	2
1.3.3	Programmeringsverktyg . . . . .	2
1.4	Kod . . . . .	2
1.4.1	Säkerhet . . . . .	2

# 1 Inledning

## 1.1 Bakgrund

Alten har en avdelning, Training, som arbetar med att sälja kurser främst till andra företag men även internt samt öppna kurser. I dagsläget sker planeringen av dessa kurser i Excel-dokument som skickas fram och tillbaka mellan Training och Invoice-avdelningen. Detta arbetssätt har varit rörigt och lätt att det blivit fel.

Training har då önskat ett system som de ska kunna använda för att planera kurser med alla kostnader och intäkter, kommunicera med Invoice samt göra budgetar.

Projektet påbörjades av lia-studenter 2017. Henning och Max blev tilldelad projektet under båda våra lia-perioder.

Efter 10 veckor av vår andra lia-period blev vi dock tilldelade ett annat projekt så vi har totalt arbetat med detta i 20 veckor.

Arbetet skedde agilt med kundmöten nästan veckovis. Under dessa möten visade vi upp vad som gjorts och diskuterade vad vi skulle fokusera på här näst för att så snabbt som möjligt nå en första version som skulle kunna användas.

## 1.2 Mål

### 1.2.1 Projektmål

Målet vi hade var alltså att få ut en första version av systemet som skulle kunna användas. För detta krävdes Roller - De olika användarna av systemet ska kunna göra olika saker. Invoice ska t.ex inte kunna komma åt eller kunna ändra något i kurserna förutom att lägga till åorkorder number". Totalt ska det vara fem olika roller som alla har olika behörigheter. Säkerhet - För att implementera roller ska även säkerhets biten ses över och delvis bytas ut. Vi ska användas oss av Json Web Tokens. Hämta ut kunder och leverantörer från internt system. I dagsläget görs skrivs allt detta manuellt och problem har uppstått. Budget-vy - För att systemet ska vara användbart behövdes en ny budget-vy. I denna vyn ska budgetar kunna sättas för de olika säljarna, för öppna kurser samt en total budget. Här ska det också gå att se hur mycket av budgeten som sålts, hur mycket som är kvar samt gross margin. Testmiljö - Sätta upp en testmiljö där alla har tillgång att testa systemet och se att det fungerar som tänkt. Buggar - t.ex så sparas datum fel och fel kurser visas i kalendern. Se till att uträkningarna räknar på rätt saker. Namngivning - Reda ut vad allt ska kallas. T.ex ska "utgifter" kallas "kostnader" Fakturerbara kurser - Ska vara checkboxes som visar ifall en utgift eller kostnad är fakturerad Pingfunktion - Det ska gå att pinga Invoice när ett kurstillfälle är redo för fakturering samt ifall ett fakturerbart kurstillfälle behöver förändringar.

Kunden hade även önskningar för att arbetet i systemet skulle gå smidigare som vi skulle göra i mån av tid. Sökning på kurser. För att arbetsflödet ska gå smidigt behövs även en sökfunktion där det ska gå att söka på kursnummer, plats, kursnamn samt datumintervall. Bifoga dokument. Detta ska gå att bifoga och ladda ner dokument för varje kurstillfälle. Stilsättning - Utefter en UX-designers rekommendationer. Varning i fall en med rollen invoice försöker lämna sidan där den fyller i workordernummer och inte har meddelat training.

### 1.2.2 Slutmål

## 1.3 Metod

### 1.3.1 Verktyg

### 1.3.2 Utvecklingsverktyg

### 1.3.3 Programmeringsverktyg

## 1.4 Kod

### 1.4.1 Säkerhet

```
1 @Override
2 protected void configure(HttpSecurity http) throws Exception {
3     http.sessionManagement()
4         .sessionCreationPolicy(SessionCreationPolicy.STATELESS);
5     http.cors().and().csrf().disable().authorizeRequests()
6         .antMatchers(HttpMethod.OPTIONS, "**").permitAll()
7         .antMatchers("/**/*.*", "/assets/**", "/", "index.html", "
8             *.js").permitAll()
9         .antMatchers(HttpMethod.POST, "/api/admin/login").
10             permitAll()
11         // .anyRequest().permitAll();
12         .anyRequest().authenticated();
13     http.apply(new JwtTokenFilterConfigurer(jwtTokenProvider
14         ));
15 }
```