

# TDP003 Projekt: Egna datormiljön

## Projektplan

Författare

Oliver Andersson, [olian419@student.liu.se](mailto:olian419@student.liu.se)  
Daniel Avila Roman, [danav696@student.liu.se](mailto:danav696@student.liu.se)

## 1 Revisionshistorik

Ver.	Revisionsbeskrivning	Datum
1.5	Lagt till diskussion av risker, utförnings datum av uppgifter, veckovisa prioriteringar av aktiviteter.	23-10-05
1.4	Lagt till tidsplanering i en tabell, med moment och dess delmoment samt en tidsestimering.	23-09-27
1.3	Fixat rubrik, risker, uppdaterat milstolpar och ökat innehållet på nästintill alla rubriker	23-09-27
1.2	Skapat tabeller för Milstolpar samt Delmoment och deadlines	23-09-18
1.1	Skrivit ut Arbetsfördelning samt rutiner kring arbetet.	23-09-18
1.0	Skapad för skrivning av projektplan	23-09-17

## 2 Projektintroduktion

Det kommer skapas en portfolio där det presenteras projekt som arbetats med upp tills denna punkt. Det kommer finnas information kring vilken kurs, vilka datum, vem/vilka som jobbat på projektet, vad projektet använder sig av för tekniker, vad projektet är för något och någon bild på hur projektet ser ut eller vad det innefattar.

### 2.1 Datalager

Datalagret är en databas där det sparas flertal kategorier av data för att senare kunna visa upp datan för användaren via presentationslagret. Datalagret har även funktionaliteterna för att kunna söka och filtrera data efter teknik och användar sökning.

### 2.2 Presentationslager

Presentationslagret kommer kommunicera med datalagret och ta emot data för att senare visa upp den. Presentationslagret kommer att utvecklas främst av HTML, CSS, Jinja och Flask och kommer bestå av fyra huvudsidor och ett bastemplate. De fyra huvudsidorna kommer vara index (Home), Projektsidan (Projects), Tekniksidan (Techniques) och de enskilda projektens sidor (Projectpage). Bastempletet kommer innehålla navigationen, bakgrund, footer samt titel för sidan.

- **Home** - Startsidan där det kommer finnas en kort 'om mig' sektion och visas upp några av de senast uppladade projekten.
- **Projects** - En lista av alla projekt med hover effekter för att visa namn och datum när de projekten skapats.
- **Techniques** - En lista av alla projekt, med möjlighet att sortera efter teknik eller söka efter projekt med en search box.
- **Projectpage** - En sida för vardera projekt som beskriver dem och visar djupare information kring deras kurser och tekniker som använts.

## 3 Arbetsmetodik

Det kommer utvecklas genom att bolla idéer med varandra, därifrån sätts vad som behöver göras upp och arbetet delas på. Efter det bestämts vem som ska göra vad, ta individuellt reda på hur man löser uppgiften. Arbetet delas vidare under projektets gång utefter vem som hinner göra mer jobb, så att det hinner bli klart till deadlines.

## 4 Arbetsfördelning och rutiner

**Gemensamt:** Tidsplan, Loft-prototyp, Implementation av datalagret mot presentationslagret, Publicering på servermiljön, Systemdokumentationen kollas på gemensamt.

**Oliver:** Första utkast av Tidsplan, första utkast av Projektplan, första utkast av Installationsmanualen, Komplettering av Tidsplan, Komplettering av Projektplan, Komplettering av Installationsmanualen, Bygga Datalagret, Halva Presentationslagret.

**Daniel:** Halva Presentationslagret, Kolla och eventuellt arbeta med Datalagret, Skapa Latex fil för Systemdokumentationen.

**Enskilt:** Reflektionsdokument

## 4.1 Rutiner

- Arbeta med de uppgifter som fördelats, om man inte kan lösa ett problem fråga om hjälp.
- Hinner man inte med uppgiften, säg till **I TID** så att det hinner lösas.
- Jobba aktivt i några timmar om dagen, behöver inte vara länge, men få någonting gjort.

## 5 Tidsplan

Nr.	Aktivitet	Tidsestimering	Vecka
3	Lofi-prototyp		
3.1	Rita klart prototyp skisser	30 min	v.36-37
3.2	Påbörja text dokumentet som förklarar tankar kring skissen	2 timmar	v.36-37
3.3	Gör i ordning textfil för vad sidorna ska innehålla	2 timmar	v.36-37
3.4	Kolla igenom så skiss och förklaring stämmer överens	1 timme	v.36-37
4	Installationsmanual		
4.1	Lista upp alla språk, program och steg som kommer användas under projektet	2 timmar	v.37-38
4.2	Beskriva hur man installerar språk, program med mera	2 timmar	v.37-38
4.3	Vanliga fel eller problem som kan uppstå, samt hur man kan lösa dem	1-2 timmar	v.37-38
5	Projektplan		
5.1	Skriva vad det är för projekt, vilka som gör det, vem är huvudansvarig	2 timmar	v.38-39
5.2	Skriva planering av rutiner, möten, läsa igenom allt, programmerings uppgift uppdelning, kodgranskning	8+ timmar	v.38-39
5.3	Tidsplanering kring när olika delar av projektet ska vara klart, första prototypens deadline, vad som ska ändras, läggas till, tas bort	8+ timmar	v.38-39
5.4	Riskbedömning och hur vi kan minska risker och ta itu med eventuella problem	2-3 timmar	v.38-39
6	Datalager		
6.1	Sätta upp datatyperna, projektnamn, projekt-id, startdatum, slutdatum, kurskod, kursnamn, kurspoäng, använda tekniker, kort beskrivning, lång beskrivning, liten och stor bild, gruppmedlemmar och en länk till projektsidan	8+ timmar	v.38-39
6.2	Påbörja att fylla i datatyperna. Skriva in projektnamn, projekt-id till en början	2+ timmar	v.38-39
6.3	Testa datalager mot presentationslagret (Hela tiden under projektets gång)		
7	Presentationslager		
7.1	Alla dagar (Ingen specifik tid): Kommer jobba med presentationslagret under hela projektets gång, då vi kommer kolla hur det ser ut och fungerar varje gång vi lägger till något till datalagret		
8	Publicering av portfolio		
8.1	Se till att portfolion fungerar i servermiljön	2+ timmar	v.39-41
8.2	Sätta upp delat repository genom GitLab	1 timme	v.36
8.3	Förbereda det lokala repositoryt, förbereda det individuella publiceringsrepositoryt	2 timmar	v.39-41
8.4	Koppla samman repository med publiceringsrepositoryt och publicera portfolion	1-2 timmar	v.39-41
9	Systemdokumentationen		
9.1	Börja med sekvensdiagram	2+ timmar	v.41
9.2	En översiktsbild med förklarande text ska läggas till	2+ timmar	v.41
9.3	Dokumentera hur presentationslagrets funktioner agerar mot datalagret	2 timmar	v.41
9.4	Beskriva hur fel loggas och hanteras samt vart man kan hitta portfoliologgen, beskriva vilka program och metoder som används vid felsökning. Samt beskriva eventuella enhetstester och hur de fungerar	2+ timmar	v.41
10	Reflektionsdokument		
10.1	Påbörja skrivning av reflektionsdokumentet	8 timmar	v.41
10.2	Fortsätta skrivning av reflektionsdokumentet, bli klar med ett första utkast	8 timmar	v.41-42
10.3	Fortsätta skrivning av reflektionsdokumentet, bli klar med ett andra utkast	8 timmar	v.41-42
10.4	Läsa igenom och avsluta skrivningen av reflektionsdokumentet	8+ timmar	v.41-42

## 6 Prioriteringar

Prioritering	Vecka
Rita klart och beskriva Lofi-Prototypen	v.36-37
Se till att det finns en grundlig installationsmanual att följa för att sätta upp projektet	v.37-38
Skriva projektplanen och sätta upp tidsplanering kring hur lång tid vi tror allt tar	v.38-39
Byggande av presentationslagret	v.40-41
Implementation av datalager mot presentationslagret	v.40-41
Jobba vidare med presentationslagret så det blir klart	v.40-41
Sätta upp servermiljön och ladda upp portfolion	v.41
Se till så att projektet fungerar i servermiljön	v.41

## 7 Milstolpar

Milstolpe	Datum
Implementation av datalager mot presentationslagret	30/9
Kolla igenom hur långt projektet kommit och jobba ikapp så det kan nå deadlines om det skulle behövas	2/10
Satt upp allting inför publicering och lärt oss hur man ska publicera	8/10
Köra projektet på annan dator för att se att allting funkar som det ska	11/10

## 8 Delmoment och Deadlines

Delmoment	Deadline	Beräknad tid	Faktisk tid
Inlämning av tidsplan	12/9	5,5 timmar	4 timmar
Första version av installationsmanualen	21/9	8 timmar	3 timmar
Första utkast av projektplan	21/9	22 timmar	12 timmar
Datalager godkänt av assistent	29/9	10 timmar	20 timmar
Presentationslager	29/9	10 timmar	
Publicering av portfolio	12/10	7 timmar	
Första version av systemdokumentationen	12/10	8 timmar	
Reflektionsdokument inlämnat	19/10	32 timmar	
Brister i systemdokumentationen korrigerade	19/10	4 timmar	

## 9 Risker

Risker finns i flertal steg av processen, det skulle kunna vara sjukdom, avhopp eller kunskapsbrist delar av projektet. Huvudriskerna som finns skulle kunna vara datalagret samt parallella uppgifter som behöver läggas uppmärksamhet på. TDP002 samt TDP003 har båda veckovisa deadlines som ska träffas och TDP002 specifikt har tunga uppgifter att hinna klart med i tid. Det gör att TDP003 arbetet fördröjs ytterligare utöver att man måste lära sig hur alla tekniker fungerar. Då både parter i gruppen är nya till de mesta som används måste vi läsa på om hur exempelvis Jinjia, Flask, HTML, CSS och Python funkar. Vilka syntaxer som finns för att göra saker, eventuella imports, osv för att göra jobbet smidigare.