

TDP003 Projekt: Egna datormiljön

Projektplan

Författare

Dennis Abrikossov, denab905@student.liu.se
Philip Bromander, phibr608@student.liu.se

1 Revisionshistorik

Ver.	Revisionsbeskrivning	Datum
1.4	Tillade milstolpar	02/10/23
1.3	Korrigerade versionsfel och övriga tillägg	27/09/23
1.2	Ändrade introduktion för att följa den nya LOFI:n	21/09/23
1.1	Modifierade dokumentet för att hålla tabeller istället	05/09/23
1.0	Fyllt i dokument med deadlines & veckor	01/09/23

2 Introduktion

Under detta projekt ska en hemsida som fungerar som ett portfolio skapas. Startsidan för portfolioet ska ha information om två individer speglad på vardera sida av hemsidan med kontaktuppgifter. Det ska finnas flera undersidor för projekt och tekniker, som används i projekten.

En undersida ska vara en söksida för projekt där man kan söka efter alla projekt med filter och fritext. Sökresultaten visar en liten ikon, namnet på projektet, och en kort beskrivning om projektet. När man klickar in sig på ett projekt skall man komma till en projektsida som visar grundläggande data om projektet och beskriver projektet i sin helhet.

Det skall också finnas en undersida för tekniker som används i projekten. På teknikundersidan listas alla olika tekniker med en liten ikon och namnet på tekniken, samt en mycket kortfattad beskrivning om den. Det ska finnas också en räknare för hur många projekt som använder tekniken.

Bakom presentationslagret används en JSON-fil för att spara all information om projekten och teknikerna. Presentationslagret frågar ett datalager för att hämta datan från JSON filerna. Presentationslagret använder Jinja och Flask för att skapa slutliga HTML-dokumentet som visas i webbläsaren.

3 Arbetsmetodik

Gruppen måste samarbeta för att nå de uppsatta deadlines. Det är flera delmoment som ingår i projektet och vissa delmoment kräver olika arbetsmetoder för gruppens samarbetsförmåga. Se projektets gruppkontrakt för hur gruppen ska samarbeta. Om en uppgift inte blir färdig på den specificerade dagen ska arbetet fortsätta dagen efter. Tills den uppgiften är uppfylld i tidsplanen, skjuts tidsplanen en dag framåt.

3.1 Planering/ arkitektur:

I flera veckor kommer projektet att planeras och flera dokument kommer att skrivas som till exempel projektplan. Under denna period kommer samarbete vara extra viktigt för att gruppen ska vara överens om vad som ska göras och hur slutprodukten skall se ut. Det är också viktigt att kunden är nöjd med hur prototypen både ser ut och fungerar och är överens med dokumentationen innan arbete på produkten kan påbörjas. En del av detta är korrekturläsning vilket ska utföras av samtliga gruppmedlemmar. Alla är skyldiga att läsa igenom alla projektrelaterade dokument och påpeka ändringar för att förbättra dokumenten.

3.2 Programmering

När arbete på produkten börjar måste arbetsuppgifter bli tilldelade. Detta görs genom veckomöten som skall hållas två gånger i veckan. Veckomöten ska hållas under labbpass då dessa är redan ordinerade. Man rapporterar in till gruppen om man gör några förändringar till kodbasen som avviker från planeringen och varje dag kollar gruppen av med ett kort möte om alla ligger i fas samt om någon ändring till planeringen måste göras.

Under veckomöten presenterar gruppmedlemmar den kod de har skrivit och övriga medlemmar kritiserar koden i målet av att öka kvalitén av koden. Övriga pull-requests går gruppen igenom och man avslutar med att tilldela nya uppgifter. Skulle någon i gruppen behöva hjälp av gruppen för ett problem koordinerar man samarbetstid under veckomötet. Git är versionshanteringsprogrammet för projektet och när en gruppmedlem är färdig med en implementation skapar den en pull-request för deras branch.

3.2.1 Prioriteringar

Den grundläggande funktionaliteten i projektet ska prioriteras. Med grundläggande menas den funktionaliteten som är specificerad av kunden och/ eller kravspecifikationen. Att stila sidorna och lägga till extra funktionalitet samt "finputs" kod är mindre viktigt. Hemsidorna ska fungera och inte se "fula ut" innan extrafunktioner eller snyggare formattering skrivs.

Om det står "grundläggande funktionalitet" eller om något moment "påbörjas" betyder det att det momentet ska uppfylla de grundläggande kraven utlagda av specifikationen, om inget annat anges. Alla moment skall minst uppfylla högsta prioriteringarna och överblivande tid kan läggas på mindre viktiga delar.

3.2.2 Kodstil

När man skriver kod försöker man skriva Pythonic om möjligt; Pythonic är inte ett krav. Man använder snake_case för variabel- och funktionsnamn. Tab-storlek ska vara 4 mellanrum stort. Man skriver alltid mellanrum mellan operatörer (till exempel: $1 + 1 = 2$, inte $1+1=2$). Variabelnamn ska förklara vad variabeln är för någonting. Funktionsnamn ska förklara vad funktionen gör. Man ska undvika förkortningar i alla fall förutom i små och lokala användningsområden. Man ska aldrig använda förkortningar för globala element.

3.3 Dokumentering

Dokumentering av både projekt och kod är ett måste. Alla medlemmar har ansvar att skapa läsbara och förståbara kommentarer i sin egna kod. Varje funktion skall ha minst en kort sammanfattning och skulle koden vara stor eller invecklad måste dokumentationen reflektera kodens komplikation på ett relevant sätt.

En utförlig systemdokumentation skrivs efter produkten är färdig där projektets kärnstruktur och metoder ska utförligt noteras. Systemdokumentationen är till för bland annat framtida utbyggnad av projektet och ska beskriva hur projektet fungerar invändigt.

Efter avslutad projekt skriver varje medlem ett reflektionsdokument. Detta dokument ska vara assisterat av en personlig dagbok som följer hela projektet.

3.4 Risker

Under projektets gång finns det flera risker som kan sakta ner eller förstöra utvecklingen. Nedan är några listade tillsammans med möjliga åtgärder. Skulle det framgå mot all förmodan att projektet inte hinner bli färdigt på anordnad tid så behöver gruppen arbeta utanför ordinerad tid. Om riskerna respekteras och åtgärderna nedan följs kommer arbete inte kräva arbete utanför ordinerad tid.

3.4.1 Frånvaro

Det kommer med högst sannolikhet uppkomma frånvaro från någon gruppmedlem. Givetvis ska varje gruppmedlem minska frånvaro så mycket som möjligt men ibland är det omöjligt.

Skulle en medlem bli frånvarande men är fortfarande kontaktbar och kan arbeta (som till exempel mild sjuka) ska man fortsätta arbeta så gott som omständigheterna tillåter. Detta tillsammans med goda tidsmarginaler bör minska påverkan frånvaro orsakar avsevärt.

3.4.2 Missförstånd av arbetsuppgift

Skulle det upptäckas att utvecklingen av projektet inte följer kundens specifikation sent under utvecklingen måste skarpa ändringar göras. Projektet måste testas periodiskt och i rätt sorts körmiljö. Det är viktigt att fråga kunden om en ny implementation eller implementation under utveckling stämmer överens med specifikationerna.

3.4.3 Brist på kunskap

Skulle det bli uppenbart att gruppen inte är kvalificerad nog att genomföra projektet, måste gruppen förbättra kunskaperna inom området. Bra metoder för att göra detta är med hjälp från andra grupper, online-resurser, assistenter, eller projektansvarig.

3.5 Arbetsverktyg

För att arbeta på projektet används flera verktyg. Hur man installerar samtliga verktyg hittar man i projektets installationsmanual. Kod redigeringsprogram är fritt val men Visual Studio Code är listad. Bildredigeringsprogrammet används för att skapa illustrationer av slutprodukten till prototypen och behövs inte för att hantera eller utöka kodbasen. Python och de två tilläggen Flask och Jinja är huvudspråket produkten skrivs i. Python är därmed ett krav för arbetet.

VScode	Kodredigering
Gimp	Bildredigering
JSON	Databaslagringsmetod
Git	Versionshantering
Python	Huvudspråk
Flask	Webbframverk
Jinja2	'Template Engine'
HTML	Presentationsspråk
CSS	Presentationsspråk

4 Tidsplan

Denna tidsplan är tagen från det dedikerade dokumentet för tidsplan. Allting under denna sektion skall vara identisk till dokumentet för tidsplanen. Tillagd i projektplan för smidig åtkomst.

4.1 Milstolpar

4.1.1 Milstolpe 1

Den 2023-09-12 ska "Low-Fidelity Prototyp" vara färdig ritad samt ska den ha testats med andra grupper genom att köra en torr kör.

4.1.2 Milstolpe 2

Den 2023-09-18 funktionen load ska vara färdig och fungera enligt de givna specifikationerna.

4.1.3 Milstolpe 3

Den 2023-09-19 funktionen get_technique_stats ska vara färdig och fungera enligt de givna specifikationerna.

4.2 Deadlines

Alla deadlines är mjuka deadlines men projektet skall hållas till dem punktligt som ett minimum.

Datum	Uppgift	Tidsangivelse
04/ 09	Gruppkontrakt	8 timmar
12/ 09	Tidsplan	8 timmar
15/ 09	LoFi-Prototyp	8 timmar
21/ 09	Projektplan & Installations Manual	16 timmar & 12 timmar, respektivt
28/ 09	Projektplan & Installations Manual	Korrigerande ändringar: 2 timmar vardera
29/ 09	Datalager	16 timmar
29/ 09	Presentationslager	Cirka 40 timmar
12/ 10	Projektet Publicerat & Utkast till Systemdokumentation	Cirka 40 timmar
19/ 10	Individuell Reflektionsdokument	24 timmar

Tabell 1: Deadlines och förväntad arbetstid vid varje moment

4.3 Veckoplanering

Vi siktar på att arbeta 30 timmar per vecka.

Vecka 35	
Måndag	Dokumenterar deadlines och sammanfattar projektet.
Torsdag	Skriver på gruppkontraktet.
Fredag	Fortsätter skriva på gruppkontraktet.
Vecka 36	
Måndag	Färdigställer gruppkontraktet och lämnar in det.
Tisdag	Börjar arbete på tidsplanen.
Torsdag	Börjar på LoFi-Prototypen.
Fredag	Blir färdig med LoFi-Prototyp & tidsplan.

Vecka 37	
Måndag	Börjar skriva på installationsmanualen.
Tisdag	Vi provar LoFi-Prototypen med andra grupper i klassen. Färdigställer installationsmanual.
Onsdag	Blir helt färdig med installationsmanual & börjar på projektplanen.
Torsdag	Jobbar på projektplanen.
Fredag	Blir klar med projektplanen.
Vecka 38	
Måndag	Börjar med datalagret. Funktionerna load, get_project_count, och get_project ska vara färdiga.
Tisdag	Håller veckomöte och skriver färdigt datalagret. Funktionerna search, get_techniques, och get_technique_stats ska vara färdiga.
Onsdag	Skriver noteringar till systemdokumentation för datalagret.
Torsdag	Håller veckomöte & börjar med presentationslagret. Skapar all grundläggande funktionalitet för 'index.html' med Flask och Jinja.
Fredag	Skriver på presentationslagret. Grundläggande funktionalitet för tekniksidan ska vara färdigställd.
Vecka 39	
Måndag	Skriver presentationslagret. Grundläggande funktionalitet för projektsidan och söksidan ska vara färdigställd.
Tisdag	Håller veckomöte & skriver presentations lagret. Lämnar in datalagret. Fullbordar sökfunktion på söksidan.
Onsdag	Kompletterar projektplan & installationsmanual. Stylar 'index.html' och projektsidan.
Torsdag	Håller veckomöte & skriver presentationslagret. Stylar tekniksidan och söksidan.
Fredag	Blir färdig med presentationslagret. Kompletterar styling på sidor som eventuellt tar längre tid.
Vecka 40	
Måndag	Skriver noteringar till systemdokumentation för presentationslager. Säkerställer att projektet publiceras.
Tisdag	Skriver systemdokumentationen. Ett flödesdiagram ska vara färdigt.
Onsdag	Skriver systemdokumentationen. För in tidigare noteringar och kompletterar.
Torsdag	Skriver systemdokumentationen. Skriver information som saknas i systemdokumentation.
Fredag	Blir färdig med systemdokumentationen. Komponerar ihop dokumentet på ett tillfredställande sätt.
Vecka 41	
Måndag	Oallokerad dag för arbete som inte hann färdigt i givna tidsramen.
Tisdag	Oallokerad dag för arbete som inte hann färdigt i givna tidsramen.
Onsdag	Oallokerad dag för arbete som inte hann färdigt i givna tidsramen.
Torsdag	Obligatorisk systemdemonstration.
Fredag	Påbörjar reflektionsdokumentet.
Vecka 42	
Måndag	Skriver egna reflektionsdokumentet.
Tisdag	Skriver egna reflektionsdokumentet.
Onsdag	Skriver egna reflektionsdokumentet.
Torsdag	Deadline för egna reflektionsdokumentet.