

Paca
Grupp 5

Projektplan
V. 1.6
2018-03-23

Dokumenthistorik

Datum	Version	Beskrivning	Författare
180307	1.0	Första versionen	Victor, Hanna
180316	1.1	Risikanalys, Process	Hanna, Milo
180316	1.2	Grovplan bearbetning	Lisa
180320	1.3	Produktbeskrivning, målgrupp, planering + mindre omformuleringar	Lisa
180320	1.4	Riskdiagram	Hanna
180322	1.5	Utvecklingsprocess, Planering + uppdaterat ansvarsområden	Lisa
180323	1.6	Färdigställt ordlista och referenslista, rättstavning, och ändringar så att projektplanen och kravdokumentet liknar varandra.	Milo

Innehåll

Dokumenthistorik	2
1. Inledning	5
1.1 Syfte	5
1.2 Ordlista	5
1.3 Referenser	7
2. Projekt	8
2.1 Syfte	8
2.2 Omfattning	8
2.3 Mål	8
3. Produkt	9
3.1 Produktbeskrivning	9
3.2 Målgrupp	9
4. Process	10
4.1 Utvecklingsprocess	10
4.2 Bemanning och ansvarsområden	11
5. Planering	12
5.1 Grovplan	12
5.1.1 Sprint 1	12
5.1.2 Sprint 2	13
5.1.3 Sprint 3	15
5.1.4 Sprint 4	16
5.2 Gantt-schema	19
6. Riskanalys	20
6.1 Identifierade risker	20
6.2 Riskdiagram	23

1. Inledning

1.1 Syfte

I detta dokument presenterar vi syfte och mål med projektet Paca samt förmedlar en övergripande bild av projektet. Dokumentet innehåller även produktbeskrivning och målgrupp. Syftet är även att beskriva utvecklingsprocess, ansvarsområde för respektive gruppmedlemmar, översiktlig planering samt analys av eventuella risker för projektet.

1.2 Ordlista

Ord	Förklaring
användargränssnitt	Ibland förkortat GUI (eng. graphical user interface). Är en metod för att underlätta interaktionen mellan människa och dator, vanligtvis mellan grafiska metaforer och bilder.
API	<i>Eng. Application programming interface.</i> En uppsättning av definitioner, protokoll och verktyg för att bygga applikationsprogram som kan kommunicera med en specifik programvara.
arkitektur	I datavetenskapliga sammanhang berör det arkitekturen, det vill säga organiseringen, av resurserna och komponenterna i ett datorsystem.
Could-krav	Från MoSCoW-modellen (se kravdokumentation). De krav som skulle kunna implementeras i produkten, men som inte är nödvändiga eller direkt förväntade ("nice to have").
Gantt-schema	Ett flödesschema som visar projektets olika faser och hur mycket av arbetet som är avklarat.
Kanban	En metod inom mjukvaruutveckling för att visualisera framsteg och process, från början till slut, på en bräda. Uppgifter placeras exempelvis under kategorin "Ska göras", "På gång", "Ska verifieras" eller "Färdigt".
klient/server-baserad arkitektur	En applikationsstruktur som delar upp uppgifter eller

	arbetsbelastningar mellan leverantören av en resurs (server) och mottagaren av resursen (klient).
Must-krav	Från MoSCoW-modellen (se kravdokumentation). De krav som systemet måste uppfylla för att bli en fullt fungerande produkt.
parprogrammering	En programutvecklingsteknik där en programmerare skriver kod medan den andra granskar det som skrivs. Rollerna växlas ofta mellan programmerarna.
retrospektmöte	I kursen “Systemutveckling och projekt I” är det ett möte där flera grupper samlas, tillsammans med deras gemensamma handledare, för att diskutera gruppernas individuella prestation under den avslutade sprinten.
scope creep	<i>Eng.</i> Ihållande eller okontrollerad tillväxt i projektets omfång, som ett resultat av dåligt definierade, dokumenterade eller kontrollerade ramar kring projektet.
Scrum	En metodik för systemutveckling som omfattar ett antal roller och ett antal beståndsdelar i form av obligatoriska möten och dokument.
Scrum Master	<i>Eng.</i> Coachar teamet och säkerställer efterlevnad av processen. Synkroniserar mellan aktörer samt avlägsnar hinder för projektgruppen.
Should-krav	Från MoSCoW-modellen (se kravdokumentation). De krav som produkten bör uppfylla, men inte nödvändigtvis.
Sprint Review	<i>Eng.</i> Ett informellt möte där den avslutade sprinten sammanfattas och bedöms, för att kunna göra nödvändiga ändringar i planeringen.
sprint	En viss tidsperiod, i systemutvecklingsmetoden Scrum, under vilken specifikt arbete måste slutföras och göras redo för granskning.
white-box testing	<i>Eng.</i> Testning av kod genom exekvering
wireframes	<i>Eng.</i> En enklare skiss av en webbsida, gjord för hand eller i ett bildbehandlingsprogram som Sketch.
UX	<i>Eng.</i> User experience. Användarupplevelse.

1.3 Referenser

[1] K. Schwaber, *Agile Project Management With Scrum*. Redmond, WA: Microsoft Press, 2004

[2] “Six Thinking Hats”, *The de Bono Group*, [online]. Tillgänglig:
http://www.debonogroup.com/six_thinking_hats.php [Hämtad: 23-03-2018]

2. Projekt

2.1 Syfte

Syftet med projektet Paca är att varje gruppmedlem såväl som hela gruppen ska få insikt, erfarenhet och förståelse för de relevanta och dominerande delarna inom systemutveckling; att tillämpa kunskaperna som värvats under teoridelen i kursen “Systemutveckling och Projekt I” samt parallella och tidigare kurser i programmet som alla gruppmedlemmar läser. Dessa kurser innefattar bland annat programmering, webbutveckling och databasteknik. De områden som är relevanta för detta projekt är gruppprocesser, processhantering, planering, kodning och programmering, testning, databashantering samt granskning, validering och verifiering av alla dokument som kommer förekommande under projektets gång.

2.2 Omfattning

Produkten som framställs kommer vara ett schemaläggningssystem för svenska företag som vill ha ett visuellt tilltalande och användarvänligt verktyg för sin planering. Projektets omfattning grundas på kraven på *sprintarna* som beskrivna i kursplanen för projektet.

Projektet kommer ej att omfatta personaladministration utöver grundläggande kontoadministration. Med kontoadministration menas sådan funktionalitet som rimligen krävs för att identifiera personer för att kunna koppla en individ till ett pass, och hantera kontaktinformation till personen.

2.3 Mål

Målet med detta projekt är att skapa en fungerande webbapplikation för att schemalägga arbetstimmar i ett företag. Målet är applikationen ska ha god *UX*-design, utifrån vedertagna standarder och teorier, i den grad att ingen utbildning ska behövas för att effektivt använda applikationen.

3. Produkt

3.1 Produktbeskrivning

Gruppens mål är att skapa ett webbaserat bokningssystem för företag med timanställda. Bokningssystemet ska vara tillgängligt för alla anställda, där de kan se scheman, boka tillgängliga pass, eller utföra passbyten med kollegor. Det ska även finnas åtkomst för chefer och administratörer, som ska kunna lägga till och korrigera de anställdas profiler, samt ha möjligheten att lägga ut lediga pass. Vid bokningar eller passbyten där alla krav inte uppfylls kommer godkännande från chef att krävas, i övrigt ska passbyten kunna ske självständigt.

3.2 Målgrupp

Paca riktar sig främst till företag med timanställda som jobbar i pass eller med flextider. Webbapplikationen kommer att vara på svenska, men en vilja att erbjuda även en version på engelska finns för framtiden. Systemet kräver en vilja hos företaget att implementera ett nytt system, samt anställda som är villiga att använda ett nytt system.

På grund av avsaknaden av en riktig kund har vi tagit fram ett fiktivt företag att förhålla oss till när det kommer till utformningen av Paca. Företaget, en dagligvaruhandel, finns i Malmö och har 30 anställda varav 10 är heltidsanställda, 10 är deltidsanställda och 10 behovsanställda.

4. Process

4.1 Utvecklingsprocess

Gruppen kommer att följa en egendesignad process med agilt arbetssätt, med idéer från *Scrum* och *Kanban* [1]. Ingen kommer att tilldelas rollen som *Scrum Master*, men en projektledarroll tilldelas en av gruppmedlemmarna och dagliga check-ins görs via en funktion i samarbetsverktyget Slack, för att fånga upp eventuella problem i ett tidigt skede. Arbetet kommer att ske i sprintar å tre veckor med ett planeringsmöte i början av varje. Gruppen har ett längre veckomöte planerat i samband med handledningsmötet varje vecka, där vi går igenom förra veckan och planerar kommande arbete. Därtill kommer vi mötas i grupp, eller i par, för vidare arbete inom de olika områden det är fokus på under den veckan eller sprinten. Gruppen har en övergripande plan för varje arbetsvecka, och varje gruppmedlem har ett ansvarsområde och fördjupningsområde. För att främja personlig utveckling inom olika områden kommer vi tillämpa oss av *parprogrammering*, och till viss del jobba i grupp utöver självständigt arbete. Gruppen har vidare beslutat att huvudansvar och fördjupning inte kommer att var densamma för en person, eftersom ett fördjupningsområde i stor kontrast till ansvarsområde bidrar till fler läromoment och mer objektiv granskning.

Användargränssnittet kommer att arbetas fram med hjälp av Python, HTML och CSS, och då det är en viktig del kommer hela gruppen vara delaktiga i designbesluten. *Wireframes* har därför skapats i ett första utkast och kommer att utvecklas under tidens gång.

Kraven identifieras genom diskussion och med inspiration från liknande tjänster som finns i bruk, samt genom ostrukturerade intervjuer med personer som använder befintliga tjänster i samma syfte för att skapa en bättre bild av vilka behov som finns. Eftersom vi inte har någon specifik kund kommer ett fiktivt företag användas i utformningen av krav.

Testning och validering kommer att utföras dels av gruppen själv, men även av andra grupper som har samma handledare. *White-box testing* kommer att utföras, och vi kommer även att använda oss av andra testningsverktyg. *Arkitekturen* kommer att vara *klient/server-baserat*, möjligtvis med *API* för att hämta schema från andra system.

4.2 Bemanning och ansvarsområden

Indelningen av bemanning och ansvarsområde är baserat på kunskaper, önskemål och uteslutningsmetod.

Hanna Fellwing, Informationsarkitekt

Övergripande ansvarsområde: JavaScript

Involverad i: Front-end (HTML,CSS)

Individuell fördjupning: Användbarhetstest och användbarhetsanalys

Victor Dahl, Informationsarkitekt

Övergripande ansvarsområde: Python, databas

Involverad i: Dokumentation

Individuell fördjupning: Dokumentgranskning

Milo Bengtsson, Informationsarkitekt

Övergripande ansvarsområde: Projektdokumentation

Involverad i: Front-end (HTML,CSS)

Individuell fördjupning: Kodgranskning

Lisa Wasslöv, Informationsarkitekt

Övergripande ansvarsområde: Tidsplan och projektkoll

Involverad i: Python, databas

Individuell fördjupning: Användbarhetstest och användbarhetsanalys

5. Planering

Tidsbudget för projektet är 220 timmar per gruppmedlem, vilket resulterar i cirka 18 timmar per vecka per person. 30 minuters handledningsmöte är inplanerat varje vecka, och i samband det har gruppen ett veckomöte i två timmar för att diskutera föregående veckas prestationer, planera kommande vecka, och eventuellt dela upp uppgifter som tagit mer tid än planerat. Utöver detta har gruppen som mål att ha minst ett till möte på cirka två timmar under veckans gång, även om individuellt arbete kan komma ske under dessa timmar. Gruppen gör dagliga check-ins på Slack, där vi meddelar vad vi gjort dagen innan, vad vi ska göra under dagen och om det finns eventuella hinder i vägen. Eftersom uppgifter kommer att delas upp individuellt och/eller parvis kommer individerna själva ansvara för att ordna dessa möten sinsemellan.

5.1 Grovplan

Alla tider angivna i grovplaneringen är per gruppmedlem.

5.1.1 Sprint 1

5.1.1.1 Vecka 10

- Aktiviteter
 - Planering 4 timmar: Ta del av och planera runt tidsplan, deadlines, projektplan osv
 - Handledning 2 timmar: Handledning med Dipak och information samt planering
 - Verktygslabb 4 timmar: GitHub, GitHub Desktop, YouTrack, Slack
 - Föreläsning 2 timmar: Projekthantering
- Milstolpar
 - Verktyg, 8 mars
 - Resurser och verktyg, såsom YouTrack och Slack, är färdiga för användning
 - Projektplan, 8 mars
 - En första version färdig

5.1.1.2 Vecka 11

- Aktiviteter
 - Individuellt handledningsmöte Dipak 2 timmar: På banan & checkup
 - Drop-in tid 2 timmar: Webb och JavaScript labb
- Milstolpar
 - Dokument, 16 mars
 - Klara med projektplanen i sin helhet. Inga delar är tomma och en tydlig struktur. Samt en första version av kravdokument.
 - Leverabler:
 - Projektplan

5.1.1.3 Vecka 12

- Aktiviteter
 - Krav 4 timmar: Insamling och hantering av krav
 - Möte 3 timmar: Var vi är och är påväg. Vad vi har att göra inför retrospekt.
 - Retrospekt presentationsslides 1 timme: Förbereda inför retrospektmötet
 - Självgranskning Kravdokumentation: 2h
- Milstolpar
 - Inlämning, 23 mars
 - Inlämning av fullständig projektplan och kravdokument med reservation för ändringar
 - Leverabler:
 - Kravdokument
 - Projektplan
 - Demo, 23 mars
 - Ska finnas en första version eller prototyp av Paca
 - Leverabler:
 - HTML-kod, eller,
 - Python-kod

5.1.2 Sprint 2

5.1.2.1 Vecka 13

- Aktiviteter
 - Sprintmöte, 4 timmar: *Sprint Review* av förra sprinten, förberedelse inför *retrospektmötet*, och planering och initialisering av nya sprinten samt
 - Handledning, 30min: Handledning av Dipak
 - Retrospektmöte 4 timmar: möte med de andra grupperna och presentation
 - Dokumentation 8 timmar: VoV-dokument, designdokument
 - Test, 2 timmar: testning av kraven i kravdokument
 - Dokumentgranskning, 2 timmar: granska andra gruppers dokument
- Milstolpar
 - Dokument
 - En första version av Vov-dokument och designdokument
 - Testning av kravdokument
 - Testa kraven i krav dokumentet och utvärdera detta.
 - Leverabler:
 - Use Cases

5.1.2.2 Vecka 14

- Aktiviteter
 - Handledningsmöte, 30 minuter
 - Gruppmöte 2 timmar: catch-up på vad som händer i projektet och gruppen
 - Kodgranskning 2 timmar: Granskning av kod dokument med walkthrough
 - Självfördjupning <x> timmar: <beskrivning av aktivitet>
- Milstolpar
 - Kodgranskning <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>
 - <Namn på milstolpe> <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>

5.1.2.3 Vecka 15

- Aktiviteter
 - Retrospektmöte 4 timmar: möte med de andra grupperna och presentation
 - Dokumentgranskning <x> timmar: <beskrivning av aktivitet>
 - <Namn på aktivitet> <x> timmar: <beskrivning av aktivitet>
- Milstolpar
 - Inlämning inför retrospekt möte 12 april 12.00
 - Leverabler:
 - Det ska finnas en version av produkten med en delmängd av de specificerade kraven ur kravdokumentet implementerade
 - Det ska finnas en första version av verifierings- och valideringsdokumentet
 - Det ska ha genomförts kravbaserade tester enligt verifierings- och valideringsdokumentet och dessa ska ha dokumenterats i minst en testrapport
 - Minst en kodgranskning och en dokumentgranskning ska ha genomförts (delar av individuell fördjupning)

5.1.3 Sprint 3

5.1.3.1 Vecka 16

- Aktiviteter
 - Sprint 3, 4 timmar: Planering och initialisering av sprint 3
 - Möte <x> timmar: <beskrivning av aktivitet>
 - Kravanalys <x> timmar: <beskrivning av aktivitet>
 - Planera testning <x> timmar: <beskrivning av aktivitet>
- Milstolpar
 - *Must-krav* implementerade <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>
 - Testgrupp klar och införstått <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - <beskrivning av leverabel>

5.1.3.2 Vecka 17

- Aktiviteter
 - Affisch workshop 2 timmar: <beskrivning av aktivitet>
 - Affisch <x> timmar: <beskrivning av aktivitet>
 - Krav testning <x> timmar: <beskrivning av aktivitet>
 - Möte <x> timmar: <beskrivning av aktivitet>
- Milstolpar
 - Testning
 - Testning på alla Must-krav, är alla dessa implementerade?
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>
 - En annan grupp har testat vår produkt, white-box testning <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>

5.1.3.3 Vecka 18

- Aktiviteter
 - Användbarhetstestning <x> timmar: <beskrivning av aktivitet>
 - Inlämning retrospekt deadline torsdag 3 maj 12.00
 - Möte <x> timmar: <beskrivning av aktivitet>
 - Första produktversion <x> timmar: <beskrivning av aktivitet>
- Milstolpar
 - Färdig version <Datum för milstolpe>
 - En första produktversion ska vara färdig och fungerande
 - Leverabler:
 - Applikationen
 - Retrospektinlämning <Datum för milstolpe>
 - Powerpoint och punkter för retrospekt mötet är färdigt och förberett
 - Leverabler:
 - Affisch

5.1.4 Sprint 4

5.1.4.1 Vecka 19

- Aktiviteter
 - Sprint möte 4 timmar: Planering och initialisering av sprint 4
 - Mindre krav implementation <x> timmar: <beskrivning av aktivitet>
 - Självgranskning <x> timmar: <beskrivning av aktivitet>
 - Dokument check <x> timmar: <beskrivning av aktivitet>
- Milstolpar
 - *Should-krav* implementerade och testade <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>
 - Dokumentation är under kontroll och i fas <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>

5.1.4.2 Vecka 20

- Aktiviteter
 - Åtgärder <x> timmar: Gå igenom och fixa problem som noterats
 - Möte <x> timmar: <beskrivning av aktivitet>
 - Regressionstester <x> timmar: <beskrivning av aktivitet>
 - Implementationstest <x> timmar: <beskrivning av aktivitet>
- Milstolpar
 - Åtgärder är genomförda <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>
 - <Namn på milstolpe> <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>

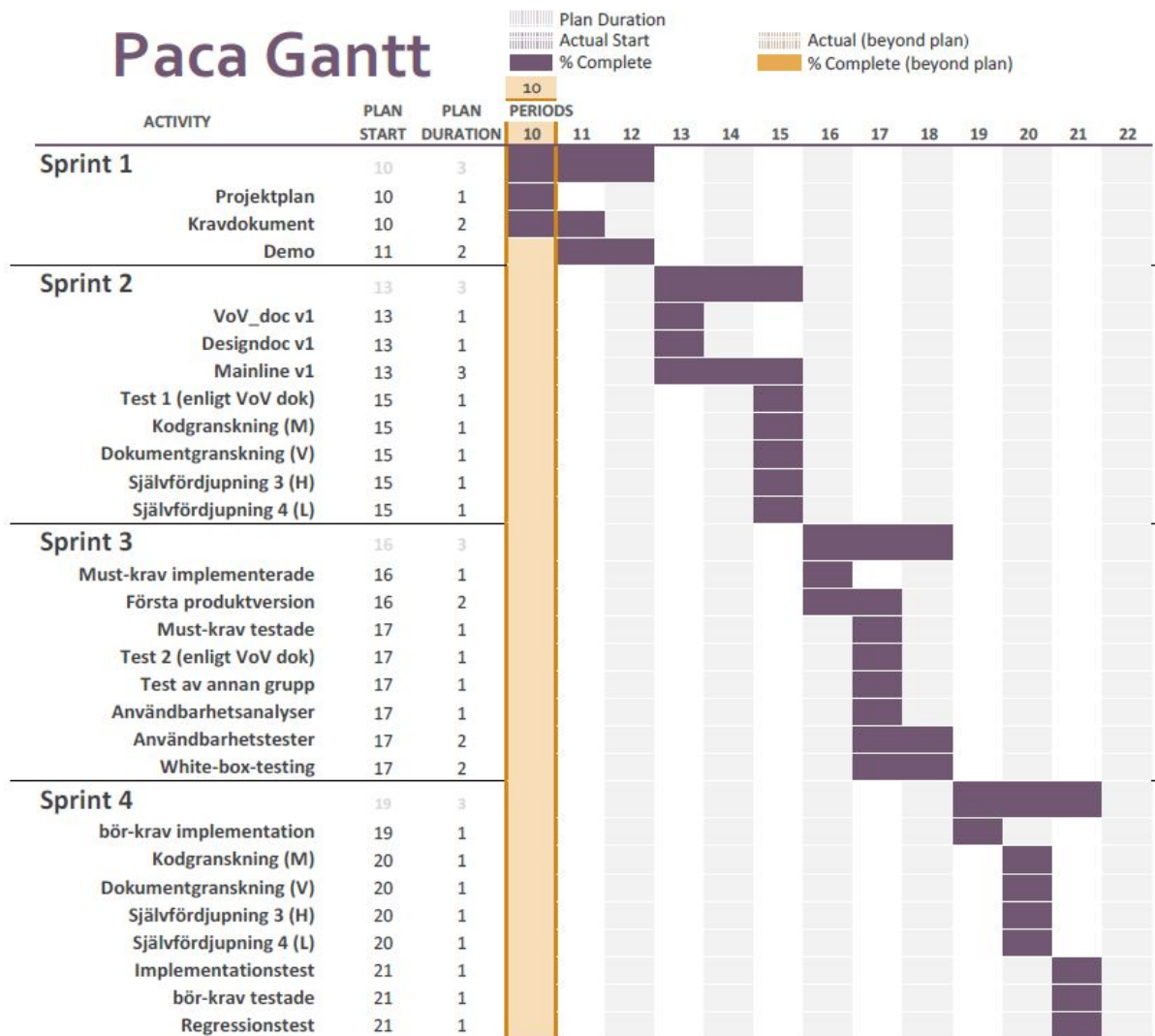
5.1.4.3 Vecka 21

- Aktiviteter
 - Möte <x> timmar: <beskrivning av aktivitet>
 - Utställningsdag <x> timmar: <beskrivning av aktivitet>
 - <Namn på aktivitet> <x> timmar: <beskrivning av aktivitet>
 - <Namn på aktivitet> <x> timmar: <beskrivning av aktivitet>
- Milstolpar
 - Utställningsdag <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>
 - <Namn på milstolpe> <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>

5.1.4.4 Vecka 22

- Aktiviteter
 - Utvärdering <x> timmar: <beskrivning av aktivitet>
 - <Namn på aktivitet> <x> timmar: <beskrivning av aktivitet>
 - <Namn på aktivitet> <x> timmar: <beskrivning av aktivitet>
 - <Namn på aktivitet> <x> timmar: <beskrivning av aktivitet>
- Milstolpar
 - <Namn på milstolpe> <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>
 - <Namn på milstolpe> <Datum för milstolpe>
 - <beskrivning av milstolpe, några meningar är tillräckligt>
 - Leverabler:
 - <beskrivning av leverabel>
 - <beskrivning av leverabel>

5.2 Gantt-schema



6. Riskanalys

Syftet med riskanalysen är att identifiera framtida risker som kan ske under projektets gång. Riskerna har tagits fram under gruppmöten där vi diskuterat dess sannolikhet, samt bedömt konsekvens i kategorierna låg, mellan eller hög. Även en handlingsplan för varje enskilt riskmoment har diskuterats för att ha något att falla tillbaka på om det skulle ske.

6.1 Identifierade risker

R1. Sjukdom hos projektmedlemmar

- Att en eller flera bortfaller från såväl formella som informella möte på grund av kort- eller långvarig sjukdom.
- **Sannolikhet:** Hög
- **Konsekvens:** Låg
- **Handlingsplan:** Eftersom vi har stora delar av vår kommunikation via olika kommunikationskanaler via internet, till exempel Slack, Facebook Messenger och YouTrack, kommer några dagars bortfall för en eller flera gruppmedlemmar inte påverka projektet särskilt omfattande då en kan arbeta och följa utvecklingen hemifrån.

R2. Bristande standardisering av namngivning

- Fel i kod beroende av missförstånd i namn. Att kodaren inte följt våra gemensamt uppsatta riktlinjer för kodnamn och kommentarer.
- **Sannolikhet:** Medel
- **Konsekvens:** Medel
- **Handlingsplan:** Skulle detta hända är alla gruppmedlemmar öppna för konstruktiv kritik och öppna diskussioner. Fel kan rättas till och ändras. Målet är att skapa så tydliga standarder för namngivning som möjligt, men riktlinjer är inte sällan öppna för tolkning och kan därför alltid öppna för risker.

R3. Studieavbrott

- Att någon i gruppen skulle besluta sig för att hoppa av studierna mitt i kursen
- **Sannolikhet:** Låg
- **Konsekvens:** Hög
- **Handlingsplan:** Skulle detta hända, vilket är väldigt osannolikt, får projektets omfång samt grupproller och ansvarsområde omvärderas och analyseras. Nedskärningar i *Want*- och eventuellt *Could*-krav kan bli nödvändigt.

R4. Implementerar inte must-krav

- Att vi inte skulle klara av eller ha tid att implementera våra Must-krav i vår webbapplikation när projektet är färdigt eller när tredje sprinten är klar.
- **Sannolikhet:** Låg
- **Konsekvens:** Hög
- **Handlingsplan:** Lyckas vi inte att implementera Must-kraven som vi själva definierat och utformat kommer Paca webbapplikationen ha låg nivå av funktionalitet och vi kan inte räkna vårt projekt som genomfört. Handlingsplanen är att inse detta i tid och be om hjälp inom gruppen samt gå på drop in-tider och hitta ursprunget till problemet.

R5. Teknisk hinder inom gruppen

- Att någon teknisk aspekt inte skulle fungera för en eller några gruppmedlemmar, såsom för hög svårighetsgrad av något moment i kod, granskning, testning eller annat avgörande delmoment i projektet.
- **Sannolikhet:** Medel
- **Konsekvens:** Medel
- **Handlingsplan:** Att öppet berätta om och lyfta sina personliga svårigheter inom gruppen för att få stöd och hjälp. Parprogrammering, vilket vi har kommit överens om inom gruppen, minskar denna risk. Skulle inte hjälp finnas att tillgå inom gruppen ansvarar såväl individen som gruppen för att hjälp införskaffas och åtgärder vidtas. Kan vara en konsekvens som långsiktigt ger upphov till R4.

R6. Negativ gruppdynamik

- Negativ energi inom gruppen som påverkar arbetsprocessen och projektets utveckling.
- **Sannolikhet:** Låg
- **Konsekvens:** Hög
- **Handlingsplan:** Dålig stämning och dynamik inom gruppen kan påverka motivationen för både individer och gruppen som helhet. Detta kan i sin tur påverka prestationen och leda till att sannolikheten för riskerna R2 och R4 höjs.

R7. Dålig testning

- Huvudsakliga testning ska genomföras av en annan grupp med white-box testing – vilket är svårt att påverka. Vi ansvarar dock för att testa vår webbapplikation själva också.
- **Sannolikhet:** Medel
- **Konsekvens:** Låg
- **Handlingsplan:** Vi ska själva testa vår webbapplikation med externt vedertaget testning verktyg samt en handfull tänkbara användare i olika åldrar och med olika teknisk kunskapsbank.

R8. Felformulerade krav

- Felformulerade krav som leder till missförstånd och tolkningssvårigheter och funktionella misstag.
- **Sannolikhet:** Låg
- **Konsekvens:** Medel
- **Handlingsplan:** Då det är vi i gruppen som tillsammans formulerar, skriver och implementerar kraven är sannolikheten för detta relativt låg. Det finns heller ingen kund att förhålla till, med undantag för de riktlinjer och mål som lärare och kursen stället på projektet. Skulle risken dock inträffa ska detta meddelas till resterande gruppmedlemmar och vi bör träffas för att diskutera om kravet ska stå kvar, redas ut, ändras eller tas bort.

R9. *Scope creep*

- Att vi hamnar utanför ramen och spinner vidare och fokuserar på icke relevanta detaljer och funktioner som inte är huvudsakliga eller avgörande för projektet och skjuter undan de stora kraven och delmoment som påverkar projektet i stort.
- **Sannolikhet:** Medel
- **Konsekvens:** Hög
- **Handlingsplan:** När utsvävningar händer och vi märker att fokus flyttas, spinner utom kontroll eller överskrider rimliga gränser i tid och resurser, ansvarar vi för att backa och reflektera över hur den bakomliggande idén egentligen påverkar projektet och webbapplikationens funktionalitet. Utse vid projektstart eller vid varje sprint en person som har den "svarta hatten" [2] och kritiskt samt rationellt tänkande.

R10. Oidentifierade krav

- I början är risken för att vi har oidentifierade krav en stor risk, men desto längre tid det går och projektet utvecklas blir detta en större risk och kostnad i såväl tid som resurser.
- **Sannolikhet:** Hög
- **Konsekvens:** Medel
- **Handlingsplan:** Vi i gruppen får träffas och diskutera det nya kravet som uppkommit och besluta ifall det ska införas som ett nytt krav eller kan uteslutas.

R11. Inga tillgängliga användare till användartester

- Att inga användare, som kan representera den tänkta målgruppen, är tillgängliga inför användartester.
- **Sannolikhet:** Låg
- **Konsekvens:** Hög

- **Handlingsplan:** Då vår målgrupp är ytterst bred är sannolikheten låg att inte hitta deltagare till testerna men om det skulle inträffa får vi försöka få tag i frivilliga på plats och fråga folk i närheten, exempelvis studenter på Niagara eller Orkanen. För att förhindra detta bör användare inför tester planeras och informeras i god tid innan satt datum.

6.2 Riskdiagram

