

# Agile Web Development för SCA

JESPER LUNDIN



**KTH Datavetenskap  
och kommunikation**

Examensarbete  
Stockholm, Sverige 2010

# Agile Web Development för SCA

J E S P E R L U N D I N

Examensarbete i datalogi om 15 högskolepoäng  
vid Programmet för datateknik  
Kungliga Tekniska Högskolan år 2010  
Handledare på CSC var Lars Kjelldahl  
Examinator var Mads Dam

URL: [www.csc.kth.se/utbildning/kandidatexjobb/datateknik/2010/  
lundin\\_jesper\\_K10062.pdf](http://www.csc.kth.se/utbildning/kandidatexjobb/datateknik/2010/lundin_jesper_K10062.pdf)

Kungliga tekniska högskolan  
*Skolan för datavetenskap och kommunikation*

**KTH** CSC  
100 44 Stockholm

URL: [www.kth.se/csc](http://www.kth.se/csc)

## **Sammanfattning**

Denna uppsats syftar till att skapa en konceptuell modell för iterativ webbutveckling åt Svenska Cellulosa AB. Modellen bygger på grunderna inom agile development och en särskild metod inom detta kallad extreme programming. På grund av efterfrågan om en mer användarcentrerad utvecklingsmodell inkluderas även särskilda metoder för kravanalys, konstruktion av prototyper, utvärdering och testning. Uppsatsen presenterar de utvalda metoderna och går igenom allmänna riktlinjer och sedvänjor för att påvisa användbarheten inom webbutveckling.

## **Abstract**

This thesis' purpose is to present a conceptual model for iterative web development for Svenska Cellulosa AB. The model derives from the practises of agile development and especially a methodology known as extreme programming. Because of the initial concern for a more user centered development model the thesis also includes specific methodologies handling requirement analysis, prototyping, evaluation and testing. The thesis presents these methods including their common practises and guide lines for the purpose of validate the feasibility of application for web development.



# Innehåll

<b>1</b>	<b>Inledning</b>	<b>5</b>
<b>2</b>	<b>Bakgrund</b>	<b>5</b>
2.1	PRIME . . . . .	6
<b>3</b>	<b>Problemställning</b>	<b>8</b>
<b>4</b>	<b>Metod</b>	<b>9</b>
4.1	Teoretiska och praktiska referenser . . . . .	9
4.2	Upplägg . . . . .	10
<b>5</b>	<b>Användarcentrerad utveckling</b>	<b>10</b>
5.1	Identifiering av målgrupp . . . . .	10
5.2	Personas . . . . .	10
5.2.1	Design av personas . . . . .	11
5.3	Användbarhetstestning . . . . .	14
5.4	Heuristisk utvärdering . . . . .	14
5.4.1	Nielsens användbarhetsmodell . . . . .	15
5.5	Kognitiv genomgång . . . . .	16
<b>6</b>	<b>Agile development</b>	<b>17</b>
6.1	eXtreme Programming . . . . .	17
6.2	Värderingar inom XP . . . . .	17
6.3	Aktiviteter inom XP . . . . .	18
6.4	Typiska projektroller . . . . .	18
6.5	Arbetsmodell för XP . . . . .	19
6.5.1	Kravspecifikation . . . . .	19
6.5.2	Leveransplanering . . . . .	20
6.5.3	Iterationsplanering . . . . .	20
6.5.4	Utveckling och design . . . . .	20
6.5.5	Accepttest . . . . .	21
<b>7</b>	<b>Tidigare studier</b>	<b>22</b>
<b>8</b>	<b>Teoribildning</b>	<b>22</b>
8.1	Kravanalys och förstudier . . . . .	22
8.2	Iterationsförbättring . . . . .	23
8.3	Utvärdering och testning . . . . .	24
<b>9</b>	<b>Resultat</b>	<b>25</b>
9.1	Konceptuell modell . . . . .	25

<b>10 Slutsatser</b>	<b>27</b>
10.1 Fokus på kravfasen . . . . .	27
10.2 Hastighet är säkerhet . . . . .	27
10.3 Prototyper och utvärdering . . . . .	28
<b>11 Referenser</b>	<b>29</b>
<b>12 Appendix</b>	<b>30</b>

## 1 Inledning

Ramverk och modeller för mjukvarukonstruktion har utvecklats i många år. Trots det har företag och organisationer svårt att anpassa sin interna utveckling efter de senaste metoderna och modellerna. Just webbutveckling är en kategori som inom många organisationer inte anpassats optimalt. Traditionella utvecklingsmodeller för mjukvara har granskats i flera årtionden men webbutveckling har endast varit i fokus de senaste tio åren. De oklara kraven ifrån slutanvändaren bidrar till komplexiteten i utvecklingen och stärker behovet av en klar målgrupp.

De huvudsakliga problemen uppstår i designprocessen, bl.a att upprätthålla fokus på användaren till produkten. Webprojekt där beställande kund och användare särskiljs kräver en balans och struktur i kravanalysen för att inte orsaka fördröjningar. Att tillämpa en smidig metod för att anpassa utvecklingen för förändringar, inkludera användaren i processen och inom organisationen behålla standard och struktur är utmanande.

Denna uppsats presenterar de grundläggande principerna inom mjukvarukonstruktion och webbutveckling och presenterar, utefter kända beprövade metoder, en konceptuell modell för agile användaranpassad webbutveckling anpassad efter Svenska Cellulosa AB's nuvarande projektmodell och organisatoriska riktlinjer.

## 2 Bakgrund

Intresset för en ny utvecklingsmodell för webbutveckling väcktes under Svenska Cellulosa Aktiebolagets introduktion av TENAnet. Portalen är en tjänst som skall introducera och förstärka en ny målgrupp för varumärket TENA och erbjuda exklusiv tillgång till nyheter, produktprover och särskild litteratur. Portalen skall också fungera som en mötesplats för personer som arbetar inom hälsovård med specifik inriktning mot inkontinens.

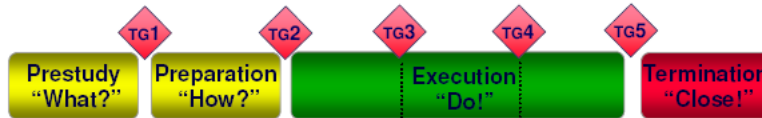
Svenska Cellulosa Aktiebolaget efterfrågade en utvecklingsmetod för webbutveckling som skulle anpassa den nuvarande projektmodellen efter modernare praxis. Den nuvarande modellen [se sektion 2.1] kan liknas vid en traditionell linjär vattenfallsmodell som ligger till grunden för utvecklingen av industriprodukter inom koncernen. Projektmodellen för webbutveckling idag är en mindre välanpassad adhoc-modell [se sektion 2.1] som konstruerades utefter grundprinciperna för den globala modellen och saknar fasta värderingar och förändringsanpassningar. För att effektivisera utvecklingen men samtidigt behålla struktur och standard bestämdes den nuvarande projektmodellen som utgångspunkt. Denna skulle därefter anpassas med huvudsaklig fokus på slutanvändaren och generell förändringsanpassning.

Fokus på slutanvändaren efterfrågades i form av inblandning av användarens kognition i utvecklingen samt en generell utvärderingsmetod för gränssnitt och interaktiva funktioner. Förändringsanpassning efterfrågades i form av iterationsförstärkning och/eller tidiga prototyper som en integrerad del av utvecklingsprocessen.

## 2.1 PRIME

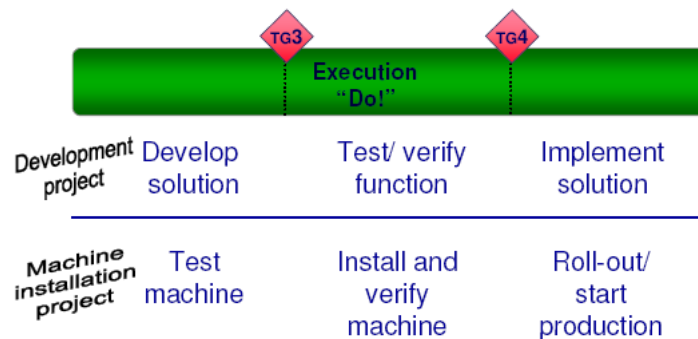
SCA använder sig globalt av en projektmodell som internt kallas PRIME. Modellen är inte anpassad efter specifika arbetsdomäner utan syftar till att globalt ge en strukturerad och standardiserad arbetsprocess som är nationellt oberoende. Problem uppstår inom utvecklingsprojekt inom IT. Den del av PRIME som är mest tillämpbar för alla utvecklingsprojekt är den globala modellen som används för produktutveckling. Även om denna sedan inofficiellt justeras för att passa ett pågående projekt så kvarstår grundstenarna i PRIME vilket bidrar till en mycket linjär arbetsprocess.

PRIME har fem stycken "tollgates" som representerar milstenar i projektet där sponsorerna har möjligheten att överväga genomförbarheten och se över kostnadseffektiviteten. Projektet är i planeringsfasen ända tills tollgate två är avklarad. Själva utvecklingen omfattas av två tollgates där den faktiska utvecklingen endast sker mellan tollgate tre och fyra. Modellen bygger på att ett fast mål och syfte och påvisar inga möjligheter till anpassningar för förändringar över tiden.



Figur 1: PRIME flöde med tollgates

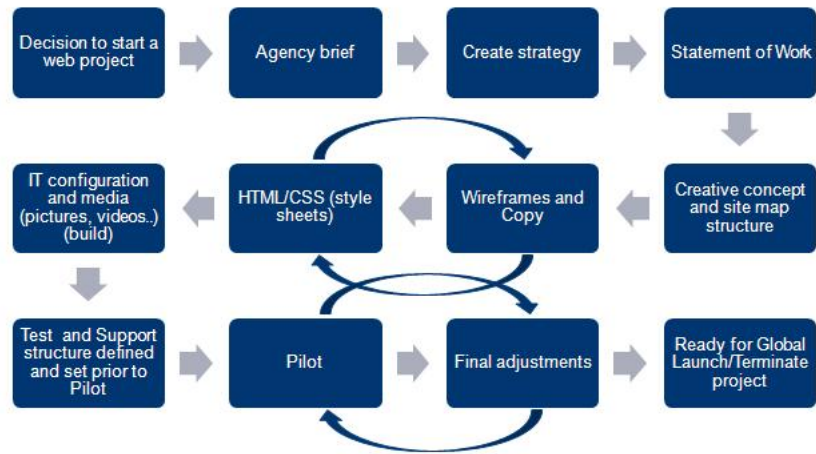
All utveckling sker innan tollgate tre enligt följande figur:



Figur 2: PRIME TG3



Vid varje tollgate samlas dokumentation för föregående process och ordres huruvida projektet skall fortgå. En anpassning av PRIME har gjorts för webutvecklingsprojekt, en form av adhoc-modell för att göra utvecklingen något mer iterativ:



Figur 3: SCAs adhoc-modell för webprojekt

### 3 Problemställning

De nuvarande modeller och riktlinjer som SCA följer tillämpas även inom web- och mjukvaruutveckling och är inte optimerad därefter. Även fastän en universell modell bidrar med standard, bra uppföljningsmöjligheter och struktur finns det risk för att viktiga delar inom webutveckling förkastas. De huvudsakliga delarna innefattar exempelvis identifiering av målgrupp, kravanalys med fokus på användaren och användbarhetstestning.

Problem uppstår också i kravbemötandet gentemot kund och användare. Ursprungliga funktionella krav och restriktioner levereras internt med organisationen som beställande kund vilket ofta utelämnar interaktiva kravställningar. Hänsyn måste även tas till den faktiska användaren av produkten när det kommer till design, funktionalitet och användbarhet.

Den nuvarande använda modellen är ej heller anpassad efter iterativ utveckling med tidiga levererbara prototyper. Inom webutveckling är en iterativ process med konceptuell design och utvärdering essentiell för att hålla fokus på kund och användare. Ett annat problem som uppstår med en mindre iterativ utvecklingsmodell är känsligheten för förändring. Långa och tidigt estimerade iterationsteg ökar risken för förseningar i levererbara delsystem och genererar ofta överflödiga dokumentation.

SCA efterfrågade några huvudelement som ville skulle förstärkas i utvecklingsmodellen:

- Djupare integrera användaren i utvecklingen genom välanvända, beprövade och erkända metoder.
- Förstärka iterationsutvecklingen för att minimera påverkan av förändringar.
- Stödja och föreslå lösningar för tidiga levererbara designprototyper.

## 4 Metod

Den vetenskapliga utgångspunkten i studien är användarcentrerad utveckling samt agile development. Särskilda områden inom utvecklingsmetoderna kommer att betonas och en bedömning görs i dialog med SCA huruvida dessa är tillämpbara sett till nuvarande praxis. Studien bedrivs delvis genom att analysera kända metoder inom mjukvarukonstruktion och webbutveckling utifrån litteratur. De metoder som anses vara mest tillämpbara i sammanhanget identifieras och möjligheterna kring rekonstruktion och integration av dessa följer i senare avsnitt.

Studien inkluderar även besök hos SCA i syfte att få en uppfattning om organisationens arbets sätt, personligheter och färdigheter samt att fastställa målbilden för den konceptuella modellen som presenteras i resultatet. En analys av SCA:s nuvarande projektmodell utförs även för att lokalisera de mest kritiska delarna i processen som bör behandlas. En kontinuerlig dialog med ansvariga personligheter upprätthålls under studien för att underlätta uppföljning av arbetet.

Motivet för val av metod refereras till tidigare studier inom människa-datorinteraktion samt mjukvarukonstruktion. Samtliga komponenter i den konceptuella modellen stärks av praktisk och teoretisk erfarenhet inom den berörda domänen.

### 4.1 Teoretiska och praktiska referenser

Tidigare studier inom mjukvarukonstruktion innefattas av teoretiska samt praktiska studier. De teoretiska studier inom mjukvaruutveckling som stärker validiteten av teoribildningen för den konceptuella modellen innefattar specifik fördjupning inom förändringprocesser och dess relation till inläring och organisatorisk påverkan.

De praktiska studier inom mjukvaruutveckling som utgör den största delen av erfarenhetsbasen innefattar ett mjukvaruprojekt i samarbete med Sweco AB. Projektet syftade till att konstruera en konceptuell lösning för spårning av farligt gods i Sverige. Projektet innefattade praxis gällande linjär mjukvarukonstruktion vilket stärker validiteten av kritiken gentemot den nuvarande använda adhoc-modellen.

Tidigare studier inom människa-datorinteraktion innefattas av praktiska och teoretiska studier. De teoretiska studier inom detta ämne innefattar fördjupning och undersökning inom exempelvis utvärderingsmetoder, användbarhet och kognition (exempel kan ses i appendix B). Detta exempel kvantifierar särskilda utvärderingsmetoder för hemsidor. Studierna skapar en grundläggande förståelse för användarcentrerad design / utveckling och stärker validiteten av den del av den konceptuella modellen som berör detta ämne. Det huvudsakliga underlaget till den delen av den konceptuella modellen som kan hänvisas till användarcentrerad design härstammar från praktiska erfarenheter inom utvecklandet av

personas.

Sett till nuvarande kunskaper inom områdena samt det litterära underlaget bedöms metoden som högst tillämpbar. Den konceptuella modellen konstrueras med ekvivalenta värderingar och praxis som utövas vid utveckling av mjukvara enligt de vetenskapliga utgångspunkterna.

## 4.2 Upplägg

Kommande två sektioner beskriver de metodiker som tillsammans med SCA har uppskattats som mest tillämpbara sett till projektnivån. Utförandet av diskussionen var svårillustrerat men resonemang, kritik och tankegång presenteras i sammanhanget och integreras i teoribildningen.

Alla grafiska representationer i uppsatsen är använda med tillstånd av SCA eller illustrerade av författaren.

# 5 Användarcentrerad utveckling

En av de vetenskapliga utgångspunkterna är användarcentrerad utveckling. Användarcentrerad utveckling är en välanvänd metod som håller användaren i fokus under hela utvecklingsprocessen. Till skillnad från traditionella modeller där produkten utvecklas efter en kravspecifikation involveras användaren i den iterativa designprocessen.

## 5.1 Identifiering av målgrupp

För webbutvecklingsprojekt av denna typ saknas det ofta konkreta krav från en betalande kund. Det finns interna krav och riktlinjer inom organisationen, men funktionella krav och designanpassningar från användaren saknas. I dessa fall stärks behovet av en klar målgrupp. Att identifiera målgruppen för systemet görs ofta genom enkäter, intervjuer och marknadsundersökningar.

Utvecklas systemet eller produkten inom en särskild arbetsdomän är även kunskap inom detta område relevant för utvecklingen. Grundläggande demografisk information kan exempelvis tillhandahållas av organisationer inom domänen och minskar då behovet av mer omfattande förstudier. Om domänen implicerar restriktioner inom terminologi eller övrig kunskap bör målgruppen vara mycket klar.

## 5.2 Personas

Personas är ett kraftfullt verktyg som fungerar som en potentiell användarmodell. De representerar specifika användare med unika karaktärsdrag utifrån den identifierade målgruppen. Personas förser oss med vetskapen om hur en användare tänker, uppför sig och vad de försöker uppnå [9: s 75]. Personas är inga "riktiga" användarprofiler utan är baserade på förstudier av målgruppen för produkten. Även fastän profilerna inte representerar riktiga personer genererar de grundkrav och förhindrar vanliga designproblem i utvecklingprocessen.

Personas kan hjälpa utvecklingen och designen genom att:

- Definiera precis vad en produkt skall göra och hur den skall uppträda.
- Skapa ett entydigt sätt att diskutera designfrågor; användarcentrering av design genom hela utvecklingsprocessen.
- Minskar behovet av flödesdiagram, tekniska modeller etc.
- Skapa ett mått på designens effektivitet; val som görs i designprocessen kan testas gentemot personas på samma sätt som mot verkliga användare (ersätter inte behovet av användbarhetstestning).
- Bidra till övriga delar i projektet; marknadsföring och säljplan.

De vanliga designproblem som brukar uppstå under produktutvecklingen är “den elastiska användaren”, själv-refererad design och design efter särfall [9: s 80].

- **Elastisk användare**

Uttrycket elastisk användare innebär att olika utvecklare har sin egen bild om vem användaren är och hur hon/han betar sig. När beslut om produkten skall fattas anpassas användaren för att passa enskilda individers åsikter. Detta orsakar en elasticitet i termen “användare” och utveckling tappar det verkliga användarfokus man från början strävat efter.

Personas är inte elastiska, liksom de riktiga användare som de representerar, och är därför ett ypperligt verktyg för att motverka detta.

- **Själv-refererad design:**

Själv-refererad design är fenomenet där utvecklare tenderar att projicera sina egna tankar, färdigheter och uppfattningar på produkten. Personas ger en klar bild om hur målgruppen ser ut och hur produkten bör utformas oavsett utvecklarnas egna mentala modeller.

- **Design efter särfall:**

Det är givetvis fördelaktigt att utveckla en produkt som kan anpassa sig efter alla tänkbara särfall men otänkbara användarscenarios bör aldrig ligga i fokus. Personas bidrar med en verklighetsanpassning för designen och

### 5.2.1 Design av personas

Alan Cooper har utvecklat en framgångsrik modell med särskilda riktlinjer för utveckling av personas [9: s 97]. Han påpekar även den viktiga kopplingen mellan design och användarens mål och definierar med hjälp av Norman (ses [9: s 89]) tre typer av målsättningar hos användaren. Dessa bygger i grunden på Normans uppfattning om att interaktiv design bör anpassas till tre nivåer hos användarens kognitiva bearbetning ; inre bearbetning, beteendebearbetning och reflektiv bearbetning. Den inre bearbetningen sker direkt vid visuell kontakt

eller andra sinnesaspekter som ger en möjlighet att skapa en direkt uppfattning utan omfattande interaktion. Beteendebearbetning sker under interaktionen och utgör huvuddelen av människans aktivitet. Reflektiv bearbetning sker "minst direkt" och innefattar medveten bedömning och reflektioner kring tidigare erfarenheter.

Utefter detta har Cooper konstruerat dessa tre målsättningar hos användaren:

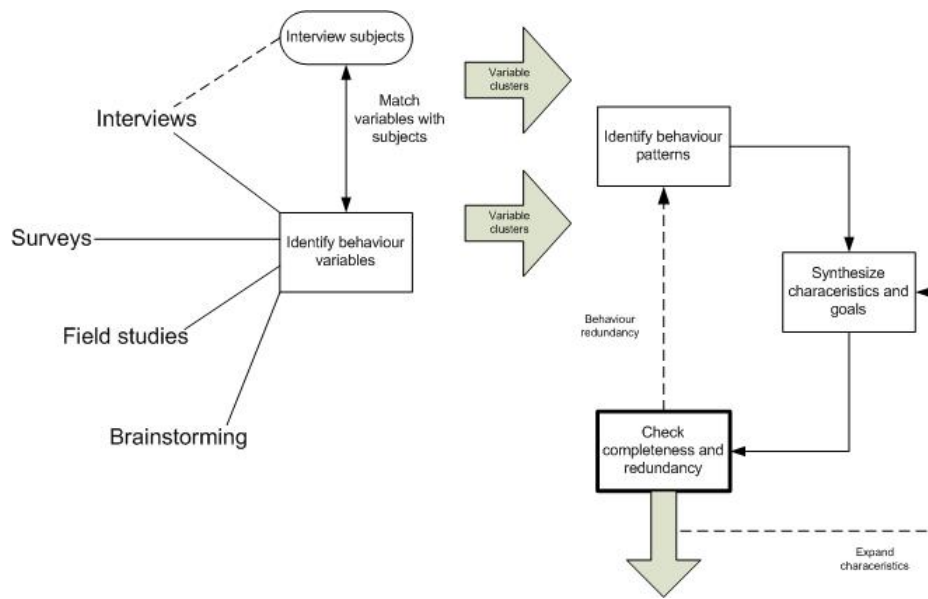
- Mål med upplevelsen; *utefter inre bearbetning*
  - representerar användarens mål med upplevelsen under användandet av produkten.
  - hur en användare vill känna sig.
- Slutgiltiga mål; *utefter beteendebearbetning*
  - representerar användarens mål med användandet av produkten.
  - vad en användare vill göra.
- Mål med livet; *utefter reflektiv bearbetning*
  - dessa representerar målen en användare har som han/hon vill uppnå genom sina slutgiltiga mål; exempelvis mål med karriären, bli en respekterad medarbetare etc.
  - vem en användare vill vara.

Dessa tre målsättningar hos användaren kommer att utgöra en betydande del i utvecklingen av personas. Detta eftersom användarens mål kan likställas med användarens motivation att använda produkten.

Konstruktionen av personas är en del i den konceptuella utvecklingsmodellen som denna uppsats syftar till att presentera men är även i sig en komplex process. Det finns en huvudsaklig modell [9: s 97] som är högst tillämpbar.

1. **Förstudier**; förutom demografisk information, inkludera användarens vanor, attityd mot produktområden, motivation inom produktområden, attityd mot tekniska faktorer och målsättningar. Anpassa efter Coopers tre målsättningar. Förstudier bedrivs lämpligtvis med:
  - (a) Enkäter
  - (b) Intervjuer
  - (c) Marknadundersökningar
  - (d) Brainstorming - kan inkludera personahypotes.
2. **Identifiera beteendevariabler**;
  - (a) Organisera insamlad data

- (b) Lista distinkta beteendevariabler. De variabler som oftast utmärker särskilda beteendemönster är;
  - i. Aktiviteter - vad användaren gör; frekvens och volym.
  - ii. Attityder - attityder gentemot produktdomän, teknologi etc.
  - iii. Aptituder (?) - vilken utbildning användaren har, förmåga att lära sig, övrig träning.
  - iv. Motivation - varför användaren är engagerad i produktdomänen.
  - v. Kunskaper och färdigheter - användarens färdigheter i relation till produktdomänen, kan relateras till (iii).
- 3. **Matcha intervjupersoner mot beteendevariabler;**
  - (a) Länka intervjuer mot de identifierade beteendevariablerna och skapa beteendemönster.
- 4. **Identifiera signifikanta beteendemönster;** dessa mönster utgör sedan grunden för personas.
  - (a) En mängd intervjuer bildar sannolikt kluster av ett antal beteendevariabler som i sin tur bildar beteendemönster.
  - (b) Beteendemönstrena måste bildas av trovärdiga kluster; logiska korrelationer mellan variabler.
- 5. **Syntetisera karaktärsdrag och målsättningar;** i detta steg konstrueras innehållet för personas.
  - (a) Fokus hålls på målsättningar
  - (b) Personarelationer kan konstrueras om det anses relevant
- 6. **Kontrollera kompletthet och redundans;**
  - (a) Kontrollera främst förekomsten av redundans inom beteendevariablerna. Om två personas endast skiljer sig demografiskt bör man överväga att utesluta en av dem.
  - (b) Kontrollera även att inga vitala mönster eller variabler saknas.
- 7. **Expandera och vidareutveckla karaktärsdrag;** detta är det avslutande steget efter iterationen. Detta är ett förbättringssteg som tillämpas i mån av tid och behov. Man säkerställer även att personen verkligen representerar en syntetisering av de viktigaste detaljerna som påträffats under förstudierna.



Figur 4: Utvecklingsflöde för personas

Det som personas slutligen genererar är det scenario som beskriver användningen av systemet under utveckling från en personas perspektiv inkluderat alla variabler som sätter samman dem. Dessa scenarios kommer utgöra grunden för den interaktiva kravställningen på produkten som utvecklas.

### 5.3 Användbarhetstestning

En av huvudelementen i användarcentrerad utveckling är användbarhetstestning. Inlärningprocessen från användare av en produkts användbarhet kallas användbarhetstestning och skiljer sig från övriga utvärderingsmetoder i och med nyttjandet av verkliga användare [1: s 9]. Fördelen med användbarhetstestning är representationen av verkliga användare, verkliga uppgifter att utföra och objektiv feedback. Dock är det en både kostsam och tidskrävande process som förutsätter en anpassad testmiljö.

Med avseende på kostnaderna av användbarhetstestning anser jag att det inte är en föredragen metod för denna konceptuella modell men kan tillämpas i mån av tid och övriga resurser.

### 5.4 Heuristisk utvärdering

För att verifiera ett systems användbarhet görs ofta ett flertal tester. Omfattande användbarhetstester kan bli tidskrävande och kostsamma och ersätts därför ofta av heuristisk utvärdering. Heuristisk utvärdering är den populäraste metoden för att verifiera användbarhet i interaktiva system [1; s 34] och går ut



på att varje testare arbetar enskilt och testar systemet mot en samling regler och principer. Utvärderingen blir en komponent i iterationsdelen av utvecklingsprocessen och blir effektiv när ett flertal utvärderingar görs med unika testare där resultaten överlappar varandra eftersom en enskild utvärderare eller testare i genomsnitt bara identifierar 35% av användbarhetsproblemen [1; s 36].

Det finns ett flertal samlingar av heuristiker och modeller för hur system bör konstrueras för att maximera användbarhet och anpassa design av gränssnitt. Exempel på sådana modeller är Nielsens tio tumregler [sektion 3.2.1] och Gerhardt-Powals modell för kognitiv genomgång.

#### 5.4.1 Nielsens användbarhetsmodell

En populär modell för heuristisk utvärdering utvecklades av Jakob Nielsen tillsammans med Rolf Molich år 1990 [Wikipedia; Heuristic evaluation]. Modellen innefattar ett tiotal regler och principer för utvärdering av användbarheten i interaktiva system:

1. Synbar systemstatus
  - (a) System bör alltid underrätta användaren om vad som händer
  - (b) Informationen ska ges genom passande feedback inom rimlig tid
2. Matchning mellan system och verklighet
  - (a) Systemet bör prata användarens språk; ord, fraser och koncept ska vara kända för användaren och inte systemspecifika.
  - (b) Systemet bör följa allmänna uppfattningar; information ska presenteras i en naturlig och logisk ordning.
3. Användarkontroll och frihet
  - (a) Användare tenderar att välja specifika funktioner av misstag och behöver en tydligt markerad väg ut för att lämna det oönskade tillståndet.
  - (b) Stöd för ångra och upprepa.
4. Följdriktighet och standard
  - (a) Användare ska inte behöva fundera över om olika ord, situationer eller handlingar är ekvivalenta.
5. Förhindrande/förebyggande av fel
  - (a) Prioritera förhindrande av fel framför välformulerade felmeddelanden
6. Minimera användarens minnesbörda
  - (a) Objekt, handlingar och valmöjligheter ska vara synliga.

- (b) Användaren ska inte behöva komma ihåg information ifrån en del av dialogen till en annan.
  - (c) Instruktioner för användandet av systemet ska alltid vara lättillgängliga och synbara i alla tillstånd.
7. Flexibilitet och effektivitet
- (a) Systemet ska erbjuda genvägar eller “acceleratorer” för erfarna användare men som inte hindrar mindre erfarna användare.
8. Estetisk och enkel design
- (a) Dialoger ska inte innehålla information som är överflödigt eller irrelevant i sammanhanget (varje extra informationenhet sätts i strid med de redan existerande relevanta enheterna).
9. Hjälpa användare att känna igen, diagnostisera och avhjälpa fel.
- (a) Felmeddelanden skall vara enkelt uttryckta, precist beskriva problemet och föreslå en lösning. Undvik felkoder.
10. Hjälp och dokumentation
- (a) Dokumentation och hjälp skall finnas tillgängligt och vara anpassat efter användaren; t.ex. i form av listor som presenteras stegvis.

## 5.5 Kognitiv genomgång

En av de primära frågeställningarna för webbutvecklingen i fråga är hur användaren upplever och utnyttjar det nuvarande systemet. För att kartlägga och anpassa produkten efter användaren måste man ha vetskap om hur systemet används. En kognitiv genomgång av systemet är en annan typ av användbarhetsinspektion som syftar till att utvärdera möjligheterna till inläring genom utforskning [1; s 38].

I en kognitiv genomgång utvärderas gränssnittet med testpersoner inom ramen av arbetsuppgifter en användare kan tänkas utföra. Ett utvecklingslag konstruerar en problemställning och ett antal premisser för varje testperson. Även fastän utvärderingens huvudsyfte har fokus kring inläring genom utforskning presenteras övriga problem i resultatet, såsom användarvänlighet och funktionalitet.

## 6 Agile development

Huvudsyftet med agile development är att skapa en utvecklingsmiljö som anpassar sig efter förändring under projektets gång. Man vill skapa stegvis växande planering för små iterationer och undvika utdragen långtidsplanering. Grundreglerna inom agile development kan sammanfattas i fyra prioriteringspunkter [5: s 18]:

- Individier och interaktioner framför processer och verktyg
- Utveckla mjukvara framför författande av dokumentation
- Kundkollaboration framför kontraktförhandling
- Svara till förändring efter en plan

Inom mjukvarukonstruktion finns det ett antal modeller som anses vara förändringsanpassade bl.a. Scrum och eXtreme Programming.

### 6.1 eXtreme Programming

Extreme programming (XP) utvecklades i slutet på 90-talet av Kent Beck. XP bygger på den universella uppfattningen om att kostnaden för att ändra en programvara under utveckling ökar exponentiellt med tiden [8: s 5]. Förändringskostnadskurvan är den främsta anledningen till att utvecklare lägger stor vikt vid kravanalysen i ett tidigt stadi. XP försöker vrida på denna uppfattning genom att kontinuerligt lägga till restriktioner och krav för utvecklingen under projektets livstid instället för att i en initial fas låsa kravanalysen.

XP definieras av fyra huvudsakliga byggstenar: värderingar, tillämpningar, aktiviteter och roller. Alla är utformade för att platta ut förändringskostnadskurvan.

### 6.2 Värderingar inom XP

De fyra värderingar inom XP är följande:

- Kommunikation
  - XP föreslår att den bästa formen av kommunikation är verbal eftersom det minskar risken för missuppfattningar.
- Enkelhet
  - gör alltid det enklast möjliga för att möta kraven. Även om det visar sig leda till ett komplext problem är det en bra utgångspunkt.
- Feedback
  - feedback är viktigt och uppmärksammar enkla problem som enstaka individer lätt missar. Det utgör en viktig del i iterationsutveckling.

- Mod
  - ha modet att ta egna initiativ, komma med förbättringsförslag och göra drastiska förändringar.

### 6.3 Aktiviteter inom XP

Inom XP definieras fyra olika aktiviteter. Dessa aktiviteter är: koda, testa, lyssna och designa.

- Koda
  - det är rimligt att påstå att all mjukvaruutveckling inkluderar kod. Inom XP är denna fas av störst betydelse för projektet.
- Testa
  - planera testet först, koda sen. Denna del kommer att behöva förändras för anpassningen till webbutveckling.
- Lyssna
  - kunden fastställer kraven för programvaran. Problemet som brukar uppstå är att varje utvecklare tolkar krav och restriktioner individuellt. Denna aktivitet påpekar vikten av att lyssna till ledningen för att skapa en gemensam syn.
- Designa
  - denna aktivitet syftar till systemdesign och beroenden mellan delar av systemet. XP föreslår en minimalistisk design före utveckling för att minska risken för uppkommandet av alltför kraftiga beroenden. Kraftiga funktionella beroenden

### 6.4 Typiska projektroller

Som alla andra projektm modeller har XP ett antal huvudsakliga roller [10]. Projektledaren, kunden, coachen, spårhunden, programmeraren, testaren och pesimisten. Fördelningen av dessa roller kan variera men framgår relativt självklar av beskrivningen.

- Projektledaren leder projektet. Planerar möten, följer upp projektets arbetsprocess, rapporterar till ledning, etc. Denna roll skiljer sig inte märkbart från övriga projektm modeller.
- Kunden skriver och prioriterar användarberättelser. Kunden har även ansvaret att skriva acceptanstesterna och under hela projektets livstid rätt att ta avgörande beslut. Kunden är alltid central inom XP.

- Coachen ser över projektet för att säkerställa att laget alltid följer de best-practices som XP förespråkar. Detta inkluderar de tidigare nämnda värderingarna och aktiviteterna.
- Spårhunden följer upp lagets framgång. Denna roll har en kompletterande funktion och syftar till att avlasta projektledaren och varna för uppkommande problem.
- Programmeraren estimerar de användarberättelser som kunden skrivit, delar upp dem i arbetsuppgifter och uppskattar arbetsbördan för dessa. Förutom övergripande programmering ansvarar programmeraren för att skriva egna enhetstester.
- Testaren hjälper kunden med att skriva användarberättelserna. Hon skriver (tillsammans med programmeraren) och kör även funktionella tester och rapporterar resultaten.
- Pessimisten eller nej-sägaren kan vara en godtycklig person i projektlaget. Dennes uppgift är att uppmärksamma hela laget om uppkommande problem när dessa påträffas. Denna roll är mer symbolisk än funktionell och påvisar endast vikten av kontinuerlig dialog mellan projektgruppen och kunden.

Detta är en sammanfattning av de unika roller som XP introducerar. Dessa roller, eller snarare dess ansvarområden, bör alltså tillämpas utöver den standard eller praxis som organisationen är van vid. Rollerna blir tydligare vid inspektion av den generella arbetsmodellen för XP.

## 6.5 Arbetsmodell för XP

Den grundläggande arbetsmodellen kan delas in i ett flertal processer. Samtliga processer präglas och de värderingar och aktiviteter som tidigare nämnts i [sektion 6.2] och [sektion 6.3]. För att lättare presentera processer och komponenter inom XP kommer vissa generaliseringar att ske i kommande beskrivning.

### 6.5.1 Kravspecifikation

Inom XP ersätts den traditionella kravspecifikationen med ett flertal komposerade s k användarberättelser. Dessa författas av den beställande kunden och innefattar en specifik funktion per berättelse. En användarberättelse är ett kort beskrivet scenario som inkluderar efterfrågad funktionalitet. Syftet är att ge en mångfasetterad syn på kompositionen av krav och presentera möjligheter för flexibilitet i den kommande utvecklingen. Samtliga användarberättelser utgör informationsbasen för den kompletta kravspecifikationen [7].

### 6.5.2 Leveransplanering

I detta stadié görs den s k leveransplaneringen som skall utgöra basen för hela utvecklingsprojektet. Samtliga berörda av projektet skall vara en del av leveransplaneringen och fullfölja sin roll genom hela fasen. Stor vikt läggs vid att rätt roll ska beröra rätt planeringsfrågor.

Utvecklarna har till uppgift att estimerar varje användarberättelse i form av ideala veckor, dvs veckor utan förhinder [11]. Kunden bestämmer därefter prioriteringsordningen för varje användarberättelse. Under estimeringen av användarberättelser uppstår ofta problem i form av osäkerhet. Ett sätt att lösa osäker estimering är att skapa en s k spike som är ett mycket simpelt program som syftar till att endast ge en uppfattning om hur berättelsen ska estimeras.

En viktig del i leveransplaneringen är att fastställa projektets hastighet i form av ett estimerat antal berättelser som kan tänkas implementeras. Inledningsvis måste även hastigheten estimeras, men kommer efter ett flertal iterationer att stabiliseras och försäkras.

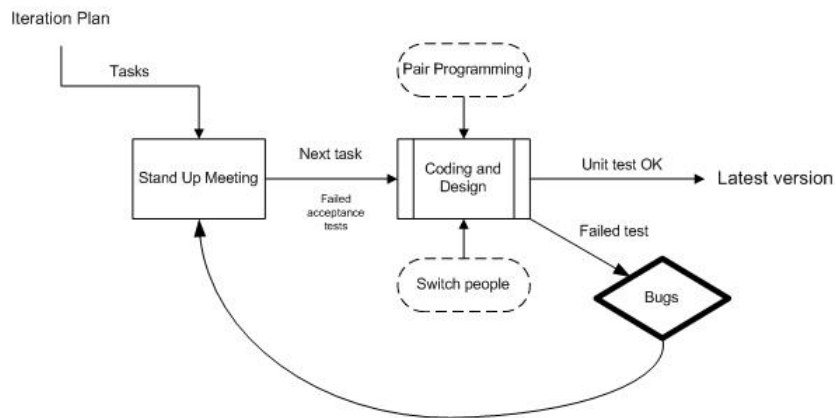
Ett av syftena med leveransplanering är att få förståelse för hur de fyra huvudsakliga variablerna i ett projekt påverkar varandra. Dessa variabler innefattar; projektets omfattning, tillgängliga resurser, tid och kvalitet. Grundinställningen är att det är omöjligt att kontrollera alla fyra och tillsammans med kunder bör projektledningen bestämma vilken prioritering projektet skall följa.

### 6.5.3 Iterationsplanering

Iterationsplaneringen är den fas som föregår själva utvecklingen. I leveransplaneringen fastställdes projektets hastighet som nu tillämpas i iterationsplaneringen i form av antal berättelser som ska implementeras. Dessa väljs av kunden utefter den bestämda prioriteringsordningen [11]. Utifrån berättelserna extraheras sedan uppgifterna åt utvecklarna. Exempelvis kan en berättelse resultera i ett flertal uppgifter. Utvecklarna väljer därefter uppgifter och estimerar dessa i form av utvecklingstid som varierar mellan en, två eller tre dagar. Återigen summeras dessa estimeringar ihop och kontrolleras mot den tidigare satta projekthastigheten. För att hålla utvecklingen konsekvent bör estimeringar aldrig ändras, varken för användarberättelser eller uppgifter.

### 6.5.4 Utveckling och design

Varje utvecklingsdag bör börja med ett kort utvecklingsmöte ("Stand Up Meeting") där samtliga involverade har möjlighet att uttrycka sig och presentera resultat. I övrigt följer den dagliga utvecklingen följande modell:



Figur 5: Den dagliga utvecklingen

### 6.5.5 Accepttest

I slutet av varje iteration genomförs ett s k accepttest som formulerats av kunder i samarbete med utvecklarna [11]. Detta test syftar till att testa just den funktionalitet som användarberättelserna krävde. Beroende på hastigheten och omfattningen kan testprocessen variera men bör alltid tidsmässigt stå i proportion till resterande processer. Notera att den är kunden som genomför det slutgiltiga godkännandet av leveransen och om så är fallet fortsätter utvecklingen till nästa iteration. Förhoppningsvis i detta skede ges även en bättre uppfattning om projektets faktiska hastighet och denna kan nu justeras för kommande iterationer.

## 7 Tidigare studier

Tidigare forskning och teorier har konstruerats för förändringsanpassad webbutveckling, exempelvis enligt [10]. I detta görs ett försök att applicera värderingarna och tillämpningarna inom XP på webbutveckling. Största delen av den ursprungliga teorin och grundprinciperna behålls och vidareutvecklas för att behålla kundfokus. Syftet var att generellt förbättra och effektivisera arbetsprocessen genom att implementera och tillämpa agile development.

Enligt [10] var det huvudsakliga problemet nischad fokus. Modern webbutveckling har enligt best practice extrem fokus på slutanvändaren och förkastar metoder kring smidig projektledning. Den multidisciplinerade IT-domänen glöms delvis bort och vissa viktiga variabler i utvecklingen nedprioriteras. Man återgår till den klassiska vattenfallsmodellen och försöker estimeras projektet i ett initialt skede vilket i sin tur leder till ett ofördelaktigt förhållande mellan kund och utvecklare där kunden väljer att göra en bred tolkning av överenskommelsen medan utvecklaren endast inkluderar komponenterna som utgör absolut minimum [10; kap 1].

Ytterligare en avgörande skillnad för webbutveckling är att varje sida som skall implementeras förutsätter en godkänd design [10: kap 7]. Detta stärker behovet av en tidiga designprototyper som bör utvecklas med avseende på användaren och inte kundens funktionella krav.

Studien konkluderade tre delar av XP-modellen som ansågs vara mest applicerbara och inkluderade projektestimering, kundrelationer och leveransplanering.

## 8 Teoribildning

Teoribildningen som ligger till grund för uppsatsen bygger på den tidigare forskningen kring XP för webprojekt. Teorin för den konceptuella modellen ärver huvudkomponenterna inom XP och parallelliserar arbetsflödet med användarcentrerad utveckling, huvusakligen sett till kravanalys och förstudier.

Teoribildningen följs enklast genom att samtidigt betrakta den konceptuella modellen i resultatet.

### 8.1 Kravanalys och förstudier

Kravanalys är ett viktigt element i alla typer av utvecklingsprojekt. I och med särskiljningen av kund och användare måste kravanalysen fördelas och följa unika delflöden. Antagandet görs att kunden och slutanvändaren tillhör olika arbetsdomäner och på så sätt kräver särskild behandling.

Teoribildningen för kravställningsprocessen följer den generella teorin om parallelliserad utveckling och avslutas i leveransplaneringsstadiet. Dock kräver flödet för kravställningen för design och interaktivitet särskilt utrymme om den ursprungliga idén som inkluderar personas ska appliceras.



Den funktionella kravställningen följer grundprinciperna för XP och borde teoretiskt kunna följa ett tidsmässigt ekvivalent flöde för utveckling av personascenarios som i sin tur utgör basen för den interaktiva kravställningen. Teorin bakom utveckling av personas följer Coopers modell enligt [sektion 5.2.1] och utgör ett sidospår med början parallellt med den initiala delen av kravfasen.

Integrationen av de separata kravställningarna som sedan utgör underlaget för leveransplaneringen måste beaktas i form av redundant funktionalitet. Den interaktiva kravställningen som genereras från personascenarios kan teoretiskt innehålla lika mycket funktionalitet som den rent funktionella kravställningen. Likväl kan den kravställningen som görs utefter användarberättelser också innefatta krav på design och utseende. Beroende på projektformen och omfattningen måste detta prioriteras enligt tidigare fastställd arbetsdomän och all redundant funktionalitet elimineras.

Tillämpningen av spikes för leveransplaneringen kan bidra till elimination av redundant kravställningar och osäkra estimeringar enligt [6.2.2].

I och med bortfallet av särskilda komponenter inom kravanalys förstärks teorin om en kontrollerande projektroll. Under utvecklingen och uppföljningen av projektet, särskilt gällande interaktiv design, bör en utvald individ kontrollera pågående arbetsprocess och verifiera att anpassningen till personas fullföljs. Teorin för minskat bortfall och nedprioritering av särskilda interaktiva designkrav föreslår denna specifika projektroll för detta syfte. Behovet verifieras i detta fall av bl.a [sektion 2] gällande tidigare praktiska erfarenheter samt av de tidigare studier som beskrivs i [sektion 7]. Projektrollen bör tilldelas i samband med övrig rollfördelning men är inte verksam förrän leveransplaneringen träder i kraft.

## 8.2 Iterationsförbättring

Teorin som utgör grunden för den iterativa delen i den konceptuella modellen utgår från principerna kring agile development m a förändringanpassad utveckling. Efter inspektion av den nuvarande använda adhoc-modellen som kan ses i [Figur 3] kan två separata iterationer identifieras. Dessa iterationer saknar tidsspann men bedöms vara flexibla i det avseendet. Modellen bör fortfarande inte anses vara iterativ och smidig då den saknar planeringsprocessen som en komponent i iterationen. Teorin föreslår tillämpningen av XPs iterationsplanering som en initial fas till den iterativa utvecklingen. Teoretiskt bör detta tillföra flexibilitet och erbjuda möjligheter att lägga till ytterligare krav under utvecklingen i form av antingen personascenarios eller användarberättelser [11]. Eftersom den nuvarande adhoc-modellen saknar en planeringsfas som del av den iterativa processen borde implementationen av iterationsplanering teoretiskt implicera förbättring. Detta skulle kräva att projektet återgick till leveransplaneringsfasen om nya kravställningar presenterades enligt principerna för XP [11].

Generellt saknas även fasta tillämpningar för IT-projekt och utrymme finns för implementation av XPs värderingar och tillämpningar för daglig iterativ utveckling för konstruktion av programkoden enligt [sektion 6.5.4]. Teoretiskt uppstår problem om prototyper skall konstrueras parallellt med den dagliga

utveckling eftersom det enligt [sektion 7] fastställdes att varje komponent som skall utvecklas inom webprojekt förutsätter en design. Därför borde skapandet av prototyper föregå den dagliga iterationen för kodning. Teorin föreslår därför skapandet av prototyper som en enskild process som sedan följs av [sektion 6.5.4].

### 8.3 Utvärdering och testning

En av huvudkomponenterna inom iterativ utveckling är testning. För att möta kraven för interaktiv design och förändringsanpassad utveckling krävs en integration av synsätten för de olika metoderna. Testningsprocessen borde kunna parallelliseras liksom huvuddelen av arbetsflödet utan att orsaka förseningar eller stagnation. Argumentet kan stärkas av inspektion av de olika metoderna. Funktionell testning görs utefter den ursprungliga kravställningen som följer de användarhistorier som kunden författat medan användbarhetstestning och testning av interaktiv funktionalitet görs enligt praxis för användarcentrerad utveckling, dvs användbarhetstestning för både funktionalitet och design. Båda tillvägagångssätten är linjära och strukturerade och har potential att konstrueras som en serie av uppgifter, exempelvis användbarhetstestning i form av heuristisk utvärdering [sektion 5.4]. Tidsrelaterade problem kan uppstå om för många metoder för användbarhetstestning tillämpas eftersom den totala testningen inte kan avslutas förrän både användbarhetstestningen och den funktionella testningen är genomförd. Dessutom uppskattas användbarhetstestningen som den mest tidskrävande processen, särskilt om någon annan metod än heuristisk utvärdering tillämpas.

Även fastän de båda testningsprocesserna teoretiskt sätt kan parallelliseras så är det, i och med risken för ineffektivitet och tidsförlust, inte lämpligt och processerna bör därför följa av varandra. Lämpligtvis föregår användbarhetstestningen den funktionella testningen i och med att viss funktionalitet kopplad till interaktiva komponenter testas redan i utvärderingen. Detta följer dessutom modellen för XP där den funktionella testningen alltid utgör den absolut sista delen av iterationen innan delsystemet med säkerhet kan levereras. Validiteten av teoribildningen som hänvisar till utvärdering och testning stärks av den väl beprövade metoden XP eftersom endast en ytterligare uppgift i flödet för den totala utvecklingen lagts till.

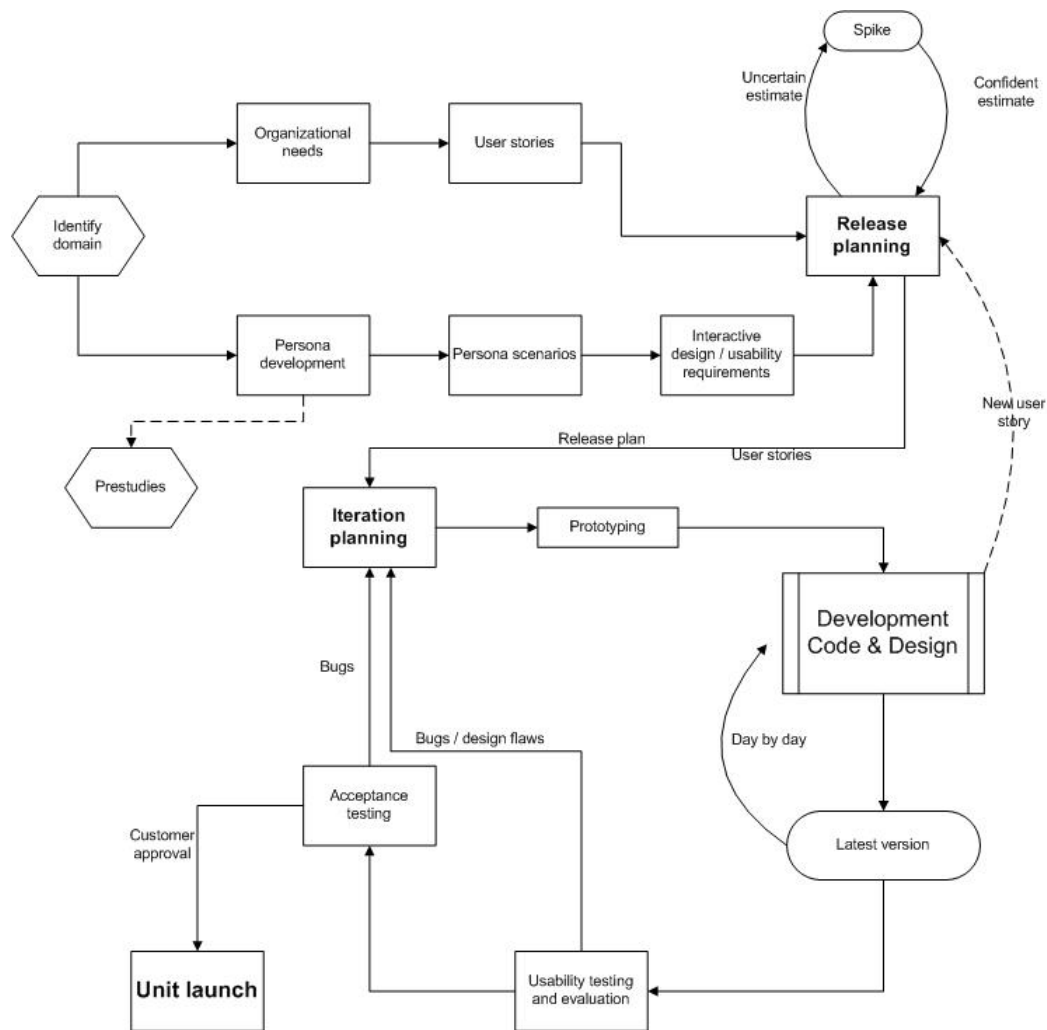
## 9 Resultat

Resultaten som presenteras är den deduktiva följderna av resonemanget i teoribildningen [sektion 8]. Syftet med studien var att presentera en konceptuell modell för iterativ användaranpassad webbutveckling.

Det generella resultatet följer teoribildningen och diskussionen i [sektion 8]. Förhoppningen att helt anpassa den nuvarande projektmodellen PRIME som presenterades in [sektion 2] nedprioriterades av delvis tidsmässiga skäl. Det ursprungliga informationsunderlaget för PRIME bedömdes inte som tillräckligt för att göra en fullskalig integration med teoribildningen. Den modell som presenteras i resultatet är därför för nuvarande inte helt integrerad med PRIME eller adhoc-anpassningen av denna metod men påvisar flexibilitet i rollfördelning och anpassningen av de inledande planeringsfaserna.

### 9.1 Konceptuell modell

Den konceptuella modellen är ett resultat ur teoribildningen i [sektion 8] och följer grundprinciperna för XP. Parallelliseringen av särskilda arbetsflöden ses tydligt i [Figur 5]. Rollfördelningen för projektet presenteras inte i resultatet utan följer med fördel praxis för XP med tillägget av en kontrollerande projektroll enligt [sektion 8.1] för stabilisering av kravuppföljning.



Figur 6: Konceptuell modell för webutveckling

Modellen är en integration av XP och användarcentrerad utveckling, främst sett till utvärdering, testning och kravanalys. Teoribildningen för resultaten påvisar en flexibilitet i den iterativa processen och modellen presenterar ett flöde för webutveckling som är anpassad efter modern förändringsanpassad utveckling och särskiljer slutanvändare och beställande kund.

Teoribildningen påvisar en användbarhet för modellen inom webutveckling och presenterar ett nytt, mer effektivt sätt att planera arbetsprocessen inom IT-projekt för SCA. Dock bör det tilläggas att omfattningen av studien är för liten för att med säkerhet och framgång implementera modellen i en global organisation. Men syftet och behoven uppfylls fortfarande eftersom efterfrågan

specificerade en konceptuell modell och inte en komplett färdig och beprövad utvecklingsmodell.

Uppsatsen och dess resultat [Figur 5] presenterar nya idéer och tillämpnings sätt för projekt inom webutveckling som i sin tur kan ge upphov eller lägga grunden för vidare studier.

## 10 Slutsatser

Den huvudsakliga slutsatsen dras efter jämförelse av resultat och SCAs nuvarande använda projektmodell. Den konceptuella modellens kapacitet kan inte med definitiv säkerhet verifieras men kan med sannolikhet anses vara mer användbar för webutvecklingsprojekt än PRIME. Den generella slutsatsen baseras på den universella uppfattningen om utvecklingsmodeller för mjukvarukonstruktion där agile development anses vara best practice. Antagandet görs även att en parallell kan dras till webutveckling då dessa två domäner är mycket närliggande.

När modeller utvecklas för särskilda arbetsprocesser nedprioriteras vissa ingående variabler och i vissa fall även förkastas helt. Exempelvis som de tidigare studierna i [sektion 7] föreslår exkluderas ofta principerna kring förändringsanpassad utveckling (agile development) när utvecklingen håller fokus på slutanvändaren. Iterativa utvecklingsprocesser tillämpas möjligtvis under själva utvecklingen men en av de viktigaste komponenterna, kravanalys och kravhantering, glöms bort.

### 10.1 Fokus på kravfasen

Som nämnt i tidigare stycke sker ofta bortfall av olika ingående variabler inom kravanalys och kravställning. Utifrån teoribildningen [sektion 8] och tidigare studier [sektion 7] kan slutsatsen dras om särskilt behov av fokus på kravställningen. Samtliga involverade bör ha förståelse för komplexiteten inom kravanalys, särskilt där kunden berörs. Därför bör kunden i alla avseenden alltid involveras i utvecklingen, inom allt ifrån leveransplanering till användbarhetstestning. Dessutom bör extra uppmärksamhet tilldelas integrationen av de olika kraven från personascenarios och användarberättelser. Om den funktionella kravställningen från kunden i något sammanhang blir redundant (nämnt i [sektion 8.1]) ska kundens version alltid prioriteras.

### 10.2 Hastighet är säkerhet

Enligt principerna för XP som presenteras i [sektion 6] sätts projektets hastighet utefter estimering av arbetsbördan. För att bättre kontrollera projektets tidsram bör hastigheten estimeras frekvent. Slutsatsen kan dras att projektets hastighet kontrollerar tidsramen, och kontroll över tidsramen ger generell säkerhet och verifierad genomförbarhet. Med hjälp av teoribildningen i [sektion 8] och med principerna

inom XP i [sektion 6] kan projektets hastighet sammanfattas som mycket relevant och t o m avgörande för att bibehålla en fördelaktig kundrelation.

### **10.3 Prototyper och utvärdering**

Förutom de separata processer under kravanalysfasen som syftar till att ge en mer användaranpassad produkt så är det två komponenter i varje iteration som utgör den huvudsakliga anpassningen till slutanvändaren. Dessa innefattar tidig prototypkonstruktion och användbarhetsutvärdering. Man kan dra slutsatsen att dessa kommer vara kritiska punkter i utvecklingen och kan potentiellt orsaka stagnation om planering inför dessa processer inte genomförs optimalt. För att undvika detta bör man enligt teoribildningen i [sektion 8.3] använda en strukturerad och linjär utvärderingsmetod som inte kräver en separat genomgående planering. Detsamma borde gälla för prototypkonstruktion, att en fast standard för detta sätts i ett tidigt skede och inte planeras för varje iteration. Metoden för prototypkonstruktion borde fastställas redan i leveransplaneringsfasen.

## 11 Referenser

### Referenser

- [1] Usability Testing and Research, Carol M. Barnum, 2002 Longman Publishers (Pearson Education, Inc), ISBN: 0-205-31519-4
- [2] Human Computer Interaction: Concepts, Methodologies, Tools and Applications, Panayiotis Zaphiris & Chee Siang Ang, 2009 IGI Global, ISBN: 978-1-60566-053-0
- [3] Customer Relationship Management, Kristin Anderson & Carol Kerr, 2002 McGraw-Hill, ISBN: 0-07-137954-1
- [4] Användarcentrerad Systemdesign, Jan Gulliksen & Bengt Göransson, 2002 Studenlitteratur, ISBN: 91-44-02029-5
- [5] Agile Web Development with Rails, Sam Ruby; Dave Thomas & David Heinemeier Hansson, 2008 The Pragmatic Programmers LLC, ISBN: 978-1-934356-16-6
- [6] Web Usability: A User-centered Design Approach, Jonathan Lazar, 2006 Pearson Education, ISBN: 0-321-32135-9
- [7] eXtreme Programming, Björn Eiderbäck, 2007 Studenlitteratur, ISBN: 978-91-44-04545-0
- [8] Extreme Programming Refactored: The Case Against XP, Stephens & Rosenberg, 2003 Springer-Verlag, ISBN: 1-59059-096-1
- [9] About Face 3: The Essentials of Interaction Design, Cooper & Reimann & Cronin, 2007 Wiley Publishing, ISBN: 978-0-470-08411-3
- [10] Extreme Programming for Web Projects, Wallace & Raggett & Aufgang, 2002 Addison Wesley, ISBN: 0-201-79427-6
- [11] <http://www.extremeprogramming.org>

## 12 Appendix

### Appendix A

#### Analys av utvärdering *med* respektive *utan* användare

*Heuristisk utvärdering går ut på att hitta användarproblem i gränssnittet så att de blir uppmärksammade och kan åtgärdas i en iterativ designprocess. Här följer en kort sammanfattad utvärdering av Vägverkets hemsida enligt tumregelsmetoden<sup>1</sup>.*

Hemsidan är lättförståelig, informativ och håller en design och struktur som att lätt att relatera till. Användaren kan även till viss del anpassa sidan efter egen smak, t.ex teckensnitt, radavstånd, teckenstorlek och kontrast. Informationsfördelningen hindrar inte användarens arbete på något sätt och det är lätt att navigera på sidan vilket ger frihet och kontroll. Dessutom visas minimalt med överflödigt information som skulle kunna förvirra användaren, dock något missvisande information i somliga fall. Vad gäller flexibilitet är det orienteringen på sidan som kan anpassas, bortsett från de tidigare funktionerna för att anpassa utseendet. Antingen kan en användare bläddra sig igenom huvudmenyn, söka på något specifikt eller välja att se en lista på alternativ sorterad i alfabetisk ordning. Det finns några funktioner som ökar tillgängligheten för vissa grupper av användare. Exempelvis användare med mindre erfarenhet av svenska språket och användare med teckenspråk som första språk. All information på hemsidan finns även att läsa på engelska.

#### Användartest

Ett användartest genomfördes med en rutinerad och kunnig användare. Användaren<sup>2</sup> som testades hade goda kunskaper inom programmering, webbutveckling och MDI.

*Uppgiftbeskrivning:* Hitta vägarbetet närmast din egen boplat

Intuivt tyckte användaren att det självklara valet var att leta under huvudmenyn som syns direkt till vänster på hemsidan. Fliken vägprojekt tycktes passa syftet och han letade vidare därifrån. Det presenterades en lista med län varpå användaren valde det län han var bostad i. Detta ledde dock inte till målet eftersom Vägverket särskiljer vägarbete och vägprojekt och användaren var tvungen att hitta nya utgångspunkter. Därefter testades sökfunktionen med nyckelordet "vägarbeten" vilket även det ledde in användaren på fel spår. Slutligen efter ca 8-9 minuters letande, nåddes målet och informationen presenterades på ett bra sätt. Anledningen till fördröjningen var att sökresultatet presenterades på ett, enligt användaren, märkligt sätt. Användaren filtrerade bort det "rätta" alternativet eftersom detta presenterades på ett helt annat sätt än de övriga sökträffarna.

<sup>1</sup><http://www.csc.kth.se/utbildning/kth/kurser/DH2620/mdik08/PDF/10punkter.pdf>

<sup>2</sup>Joakim Lundin, teknolog inom medieteknik, inriktning MDI



## Analys och reflektioner

Till synes är Vägverkets hemsida mycket välstrukturerad och informativ. I stort sett all information om Sveriges vägnät och trafikrelaterade händelser är inkluderade. Men trots detta krackelerar fasaden då den vid användartestning ger ett dåligt resultat. Tumregelsmetoden presenterade hemsidan som väldigt användbar då den uppfyllde majoriteten av punkterna. Om man ser till REAL-modellen (relevans, effektivitet, attityd och lättlärdhet) ger den endast positivt resultat på två av punkterna efter analysen av användartestet. Systemet/hemsidan utför nog de uppgifter användaren vill, men effektiviteten är dessvärre låg, mycket pga att sidan är svårorienterad. Attityden är initialt bra, men irritation uppstår när målet inte nås inom en rimlig tidsram. Däremot är hemsidan relativt lättlärd, eftersom när användaren väl har lärt sig systemets särskilda struktur kan han/hon bruka den betydligt mer effektivt än tidigare.

Jämför man tumregelsmetoden och användartestning tycker jag att användartestning ger ett mycket tydligare och ett mer autentiskt resultat. Dock stämmer de överens på vissa punkter. Bl.a att flexibiliteten är hög, användaren brukade både meny och sökfunktionen men däremot försedde dessa inte användaren med relevant information. Vid sökning valde användaren att filtrera bort rätt resultat vilket tyder på att relevant information anges på ett felaktigt sätt. Skillnaderna och likheterna mellan metoderna kan bero på flera saker. Resultaten av tumregelsmetoden är något missvisande, eftersom systemet framstår som välanpassat och användbart på de flesta punkterna. Tumregelsmetoden är en äldre metod, utvecklad i början av 90-talet, och därför inte särskilt välanpassad för hemsidor, eftersom internet inte var lika etablerat då som det är idag. Dessutom bör man ha i åtanke att metoden har fördelar som att den är billig, snabb och enkelt och därför kanske inte alltid är den mest optimala metoden. En stor nackdel med denna metod är att den inte är bunden till en specifik uppgift, vilket också kan påverka det slutgiltiga resultatet. Istället ser den till allmännyttan och kan därför förbise vissa mindre uppenbara brister i specifika funktioner. Personligen tycker jag att den utgör en typ av grundmall för hur ett system bör struktureras och inte längre är lika attraktiv vid optimering av användbarheten.

Användartestning ger alltid en bättre uppfattning om hur användbart ett system är, eftersom den metoden har mycket färre begränsningar. Det är en verklighetsanpassad metod som ger en möjlighet att optimera och anpassa interaktiva system oavsett typ. Resultaten av användartestning kan variera med vilket uppgift som har satts. En mycket specifik och nischad uppgift kan vara mer svårslöslig och generera ett tillsynes sämre resultat. Personligen anser jag att resultaten av användartestningen delvis även beror på vilka kognitiva processer som är involverade och förutsatt att man har en tillräckligt stor testgrupp borde resultaten sällan vara missvisande.

Den största skillnaden mellan resultaten av metoderna i denna utvärdering är troligtvis uppgiftsrelaterat. Hade man valt en någorlunda enkel uppgift hade troligtvis metodernas resultat varit mycket snarlika.

