

# DAT256 Weedle report

Alsadi, Teif	Carlsson, Eric	Haux, Adam	Johnsson, Erik
Lindblom, Pontus	Magnusson, Vidar	Östberg, Oscar	

June 2019

# Innehåll

<b>1</b>	<b>Customer Value and Scope</b>	<b>1</b>
1.1	The chosen scope of the application under development including the priority of features and for whom you are creating value . . . . .	1
1.1.1	Sammanfattning av projektet (A) . . . . .	1
1.1.2	Vad vill vi göra i framtida projekt? (B) . . . . .	4
1.1.3	Vad krävs för att vi ska nå dit? (A → B) . . . . .	5
1.2	The success criteria for the team in terms of what you want to achieve within the project . . . . .	5
1.2.1	Sammanfattning av projektet (A) . . . . .	5
1.2.2	Vad vill vi göra i framtida projekt? (B) . . . . .	6
1.2.3	Vad krävs för att vi ska nå dit? (A → B) . . . . .	7
1.3	Your user stories in terms of using a standard pattern, acceptance criteria, task breakdown and effort estimation and how this influenced the way you worked and created value . . . . .	7
1.3.1	Sammanfattning av projektet (A) . . . . .	7
1.3.2	Vad vill vi göra i framtida projekt? (B) . . . . .	9
1.3.3	Vad krävs för att vi ska nå dit? (A → B) . . . . .	9
1.4	Your acceptance tests, such as how they were performed, with whom, and which value they provided for you and the other stakeholders . . . . .	11
1.4.1	Sammanfattning av projektet (A) . . . . .	11
1.4.2	Vad vill vi göra i framtida projekt? (B) . . . . .	11
1.4.3	Vad krävs för att vi ska nå dit? (A → B) . . . . .	12
1.5	The three KPIs you use for monitoring your progress and how you use them to improve your process . . . . .	12
1.5.1	Sammanfattning av projektet (A) . . . . .	12
1.5.2	Vad vill vi göra i framtida projekt? (B) . . . . .	14
1.5.3	Vad krävs för att vi ska nå dit? (A → B) . . . . .	15
<b>2</b>	<b>Social Contract and Effort</b>	<b>16</b>
2.1	Your social contract, i.e., the rules that define how you work together as a team, how it influenced your work, and how it evolved during the project . . . . .	16
2.1.1	Sammanfattning av projektet (A) . . . . .	16
2.1.2	Vad vill vi göra i framtida projekt? (B) . . . . .	17
2.1.3	Vad krävs för att vi ska nå dit? (A → B) . . . . .	17
2.2	The time you have spent on the course and how it relates to what you delivered . .	17
2.2.1	Sammanfattning av projektet (A) . . . . .	17

2.2.2	Vad vill vi göra i framtida projekt? (B)	18
2.2.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )	19
<b>3</b>	<b>Design decisions and product structure</b>	<b>20</b>
3.1	How your design decisions support customer value	20
3.1.1	Sammanfattning av projektet (A)	20
3.1.2	Vad vill vi göra i framtida projekt? (B)	21
3.1.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )	22
3.2	Which technical documentation you use and why	22
3.2.1	Sammanfattning av projektet (A)	22
3.2.2	Vad vill vi göra i framtida projekt? (B)	23
3.2.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )	23
3.3	How you use and update your documentation throughout the sprints	24
3.3.1	Sammanfattning av projektet (A)	24
3.3.2	Vad vill vi göra i framtida projekt? (B)	24
3.3.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )	24
3.4	How you ensure code quality and enforce coding standards	25
3.4.1	Sammanfattning av projektet (A)	25
3.4.2	Vad vill vi göra i framtida projekt? (B)	26
3.4.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )	26
<b>4</b>	<b>Application of Scrum</b>	<b>28</b>
4.1	The roles you have used within the team and their impact on your work	28
4.1.1	Sammanfattning av projektet (A)	28
4.1.2	Vad vill vi göra i framtida projekt? (B)	30
4.1.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )	30
4.2	The agile practices you have used and their impact on your work	31
4.2.1	Sammanfattning av projektet (A)	31
4.2.2	Vad vill vi göra i framtida projekt? (B)	34
4.2.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )	34
4.3	The sprint review and how it relates to your scope and customer value	35
4.3.1	Sammanfattning av projektet (A)	35
4.3.2	Vad vill vi göra i framtida projekt? (B)	36
4.3.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )	36
4.4	Best practices for learning and using new tools and technologies	37
4.4.1	Sammanfattning av projektet (A)	37
4.4.2	Vad vill vi göra i framtida projekt? (B)	38
4.4.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )	38

4.5	Relation to literature and guest lectures . . . . .	39
4.5.1	Sammanfattning av projektet (A) . . . . .	39
4.5.2	Vad vill vi göra i framtida projekt? (B) . . . . .	39
4.5.3	Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ ) . . . . .	40
<b>5</b>	<b>Referenslista</b>	<b>41</b>

# 1 Customer Value and Scope

## 1.1 The chosen scope of the application under development including the priority of features and for whom you are creating value

### 1.1.1 Sammanfattning av projektet (A)

Projektet inleddes med ett gruppmöte där projektets scope diskuterades och bestämdes baserat på vad som ville uppnås samt tillgängliga resurser. Gruppen identifierade då ett behov av en mobilapplikation som samlar parkeringsplatser från flertalet företagsspecifika parkeringsapplikationer som finns att tillgå i göteborgsområdet. Vidare skulle det vara möjligt att boka en plats hos en av dessa leverantörer direkt i applikationen. Denna lösning skulle medföra värde till alla som är i behov av parkering i Göteborg genom att alla platser samlas på ett ställe. Den skulle även medföra värde för parkeringsägarna genom att deras parkeringar når en större kundbas samt genom den marknadsföring som medförs när de är synliga i applikationen. Till sist finns det även ett värde för produktägaren som äger plattformen genom intäktsströmmar då en del av varje parkeringskostnad skulle tillfalla plattformsägaren.

För att säkerställa att värdet för användarna av applikationen kontinuerligt representerades fördes en diskussion inom gruppen som resulterade i en personifiering av den typiska användaren vilken användes för vägleda fortgående beslut. Vid detta stadie togs en *mockup* fram med alla funktioner som under tidpunkten ansågs kunna medföra värde för någon aktör på plattformen.

Nedan följer en genomgång av hur *scopet* såg ut för varje sprint och där de huvudsakliga funktionerna från varje *sprint board* presenteras i tabeller. Under projektet har ingen *user story* tilldelats ett absolut värde för att representera dess prioritet. Istället har de i tabellen lagts i den ordningen enligt vilken de är prioriterade. Sammantaget kan det påpekas att då sprintarna endast varit en vecka långa har det scope som undertagits vid varje sprint varit mycket smalt.

Efter de två inledande veckorna påbörjades projektet och den första sprinten. Första sprintens scope innefattade dels att se till att gruppmedlemmarna blev bekväma med utvecklingsmiljön. Det gjordes genom att först skapa och introducera alla till den *tech stack*, samling av tekniker, som skulle användas för att skapa produkten. Främst gav detta värde till utvecklarna då de blev bekväma i den utvecklingsmiljö som skulle användas. Enklare utveckling av applikationen påbörjades även, där de *User Stories* som finns i Tabell 1 utvecklades. Hänsyn togs nästan uteslutande till lärandet för projektgruppen där slutvärdet för användarna och produktägaren var sekundär. Detta sågs som en nödvändig investering för framtiden då många av utvecklarna hade bristande erfarenheter från

att programmera i *React Native* med *TypeScript*.

Till första sprinten hade projektet ingen officiell produktägare då projektgruppen fortfarande var i upplärningsfasen kring hur processen borde skötas och inte hade insett vikten av att tillsätta en. Avsaknaden av produktägare under *sprint planning* gjorde att det inte fanns någon *product backlog*. Istället togs endast tillräckligt många *user stories* fram för att fylla *sprint boarden* och uppnå de mål som tidigare nämnts om lärande. Därmed fanns inte heller någon prioritet utifrån kundvärde kring vad som utvecklades under första sprinten. Istället sattes prioritering baserat på vad som var bäst lämpat för utvecklarna att utöka sina förmågor. En produktägare och vice produktägare kom att tillsättas från gruppens medlemmar under sprintens gång vilket utvecklas vidare om i avsnittet *Application of scrum*. Hädanefter sattes prioriteringen av produktägaren i samråd med vice produktägare utifrån vad som uppskattats vara mest värdefullt för applikationens målgrupper samt vad som är mest värdefullt för produktägarna själva.

Tabell 1. De huvudsakliga *features* som utvecklades under *sprint* ett i fallande prioritet.

User Story	Estimat (Story points)
Listelement med mockdata	15
Lista över parkeringsplatser	10
En bokningsknapp	20

Till andra sprinten hade jämfört med första, en *product backlog* tagits fram. Dock, på grund av brist på struktur och vidare tidspress då denna togs fram så innehöll den inte *user stories* för alla de funktioner som var del av *mockupen*. Därav så även om produktägaren nu kunde prioritera *user stories* för att avgöra vad som skulle vara del av *sprint boarden* och utvecklas i andra sprinten, så kunde de inte göra det med det fulla scopet i åtanke. Vi ansåg dock att de viktigaste funktionerna, de mest värdebringande, fanns med i *product backlog*.

I tabell 2 redogörs för de huvudsakliga funktionerna under andra sprinten. Verklig data på parkeringsplatser samlades in vilket gav värde gav stort värde för kunderna genom att informationen som visades var verklig. Detta gavs därmed högst prioritet. En karta där användaren kan se sin position och minst en parkeringsplats fick andra prioritet då det ansågs ge stort värde till användarna i form av att de kan hitta parkeringsmöjligheter med funktionen. Till sist prioriterades även en användarprofil som går att modifiera så att man som användare kan se och korrigera information om sig själv.

Tabell 2. De huvudsakliga *features* som utvecklades under *sprint* två i fallande prioritet.

User Story	Estimat (Story points)
Insamling av verklig data	21
En karta med parkeringsplatsmarkörer samt användarposition	21
En användarprofil med e-postadress och möjlighet att lägga till bilar	20

Under tredje sprinten utökades arbetet med *product backlog* genom att ett separat tillfälle för *backlog refinement* infördes. Detta gjorde att *user stories* för samtliga delar av *mockupen* kunde tillföras *product backlog*. Därefter ordnade produktägaren en prioritet mellan alla *user stories* i *product backlog* vilket resulterade i de valda uppgifterna under sprinten. Högsta prioritet under denna sprint var en funktion som möjliggjorde för användaren att boka en parkeringsplats då det var grundfunktionen för applikationen. Näst högsta prioritet var en sökfunktion där användaren kan söka efter parkeringsplats då den typiska användaren är *on-the-go* och därmed behöver en smidig funktion för att snabbt hitta det som eftersöks. Prioriteten därefter var en menyrad som skapar värde för användaren genom att möjliggöra navigering mellan de olika vyerna. En sammanställning återges i tabell 3. Alla funktioner under det här stadiet har sin grund i att skapa värde för användaren som är i behov av en parkeringsplats. Detta gör att i ett efterkommande skede, genom att kunden använder applikationen, generaras intäktsströmmar åt produktägare och parkeringsägare.

Tabell 3. De huvudsakliga *features* som utvecklades under *sprint* tre i fallande prioritet.

User Story	Estimat (Story points)
Bokning av parkeringsplatser	13
Sökfunktion med platssökning för relevanta resultat	16
Menyrad för navigering i appen	15

I den fjärde sprinten utvecklades bland annat de *user stories* som finns i Tabell 4. Under denna sprint lades huvudfokus vid att ta fram stödfunktioner åt användaren. Detta innefattar att kunna se sin historik, möjlighet att komma i kontakt med företaget och att data sparas mellan användar-sessioner. Produktägaren prioriterade dessa då de bidrog till en stärkt helhetsupplevelse och enklare användning av applikationen

Tabell 4. De huvudsakliga *features* som utvecklades under *sprint* fyra i fallande prioritet.

User Story	Points
Lista av historik på betalda parkeringar	13
<i>About us page</i> för att ge användaren information och kontaktuppgifter till skaparna av appen	5
<i>Local storage</i> , lokalt sparande av användardata mellan användningssessioner av appen.	8
Enhetlig stil av applikationen	20

Den sista sprinten hade fokus på att åtgärda buggar och ta fram en enhetlig design istället för att utveckla nya funktioner till applikationen. Detta för att säkerställa att användarupplevelsen av applikationen nådde upp till utsagd skiss. Detta var mycket värdefullt för produktägaren då sprinten avslutades med en produktpresentation för möjliga investerare och användare, och därmed var det essentiellt att de befintliga funktionerna i applikationen fungerade som tänkt och att applikationen var estetiskt tilltalande samt användarvänlig.

### 1.1.2 Vad vill vi göra i framtida projekt? (B)

Målet för framtida projekt är att bättre använda de möjligheter som agil utveckling jämfört med vattensfallsmetoden erbjuder i termer av det scope som i början definieras. I ett framtida projekt är det eftersträvarvärt att i början ha ett mindre *scope* för hela projektet för att lättare kunna säkerställa att alla förväntningar uppfylls samt för att motivera utvecklarna med konkreta mål. Därefter kan *scopet* utökas successivt allt eftersom för att generera mer värde till slutkunden.

Något annat som upptäcktes som essentiellt för att maximera värdet var att införliva produktägaren i arbetet från start. Produktägaren har en viktig roll i att definiera varje sprints *scope* genom att prioritera det som ska utvecklas så att funktioner i applikationen tas fram i den ordning som levererar snabbast och största möjliga värde för kund.

Vidare vill vi även använda oss av längre sprinter för att på så sätt kunna utveckla mer värde under varje sådan. Detta då längden av en vecka upplevdes starkt begränsa scopet av värdet hos varje sprints inkrement.



### 1.1.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att ta fram ett mindre scope till arbetet är det viktigt att inledande i projektet diskutera med produktägare vad som är produkten i sitt minsta utförande. Detta för att därefter utgå från denna i början av utvecklingsarbetet. Viktigt är dock för att göra produktägaren tillmötesgående är att tydliggöra att ytterligare funktionalitet är möjlig att läggas till i ett senare skede för att uppnå den slutvision man har om produkten. Att komma överens om ett mindre scope, en grundprodukt, kan vidare uppnås genom påvisa att båda står att gagna av att börja med att ta fram en sådan. Genom att eftersträva att identifiera vad som är grundprodukten kommer man lättare se vad som ger mest värde. Detta kan då snabbare komma ut på marknaden och således kan bland annat viktig feedback om produkten tidigt erhållas. Denna feedback kan då direkt återföras till projektet och införlivas i vad som fortsatt ska utvecklas. Centralt är därför att ha en öppen och genomgående dialog med produktägaren i början så att man påbörjar projekt från ett smalt scope.

För att se till att det scope som man tar sig an med projekt bidrar med värde ska vi mer djupgående ta reda på och definiera detta innan man går vidare. Detta antingen genom en produktägare som är mer insatt i marknaden eller att man inledande lägger resurser på att genomföra en marknadsundersökning.

Vidare bör man under dessa dialoger med produktägaren även lägga fram vikten av denne är kontinuerligt delaktig under arbetet. Detta för att säkerställa att produktägaren är aktiv i arbetet med att prioritera funktionerna hos det inkrement som ska utvecklas varje sprint. För att vidare se till att man lyckas övertala produktägaren till att närvara under *sprint planning* kan även några av de strategier som Kniberg (2015) nämner användas.

## 1.2 The success criteria for the team in terms of what you want to achieve within the project

### 1.2.1 Sammanfattning av projektet (A)

I projektet har vi delvis haft tre stycken övergripande mål och dels även *sprint goals* för varje sprint. Kriteriet för att projektet skulle anses vara framgångsrikt var att uppnå dessa tre övergripande målen tillsammans med att i största möjliga utsträckning uppnå de för varje ny sprint, nya *sprint goal*et.

Det första övergripande målet har varit att skaffa oss erfarenhet av Scrum medan det andra var att

leverera en slutprodukt som efterliknar den uppsatta *mockupen* och som skapar värde för produktägaren och användarna. Det tredje och sista övergripande målet var att sett till gruppsamarbetet ska medlemmarna lära känna varandra och uppleva en god arbetsmiljö under projektet.

*Sprint goals* har formulerats till att för varje sprint belysa vad som var huvudmålet med det arbetet som togs till. Gällande den första sprinten var det att alla skulle komma igång med och få en inblick i de verktyg och programspråk som skulle användas under projektet. För andra sprinten var istället *the sprint goal* att vi ville ta tillvara på investeringen av att under första sprinten ha fokuserat på att lära oss och därav blev målet att vi ville klara av fler *story points* än föregående sprint. För sprint tre och fyra syftade *the sprint goal* till vilka funktioner som var av stor vikt att utveckla. Under den tredje var det att kunna boka en parkering medan under fjärde att ta fram stödfunktioner för användarna. *The sprint goal* för sista sprinten var att åtgärda alla buggar.

Efter sista sprinten kunde vi konstatera att projektet varit framgångsrikt då vi dels uppnått våra *sprint goals* under projektets gång men även de tre övergripande målen. Genom att ha lagt mycket resurser vid att tillämpa och reflektera över Scrum ansåg vi oss ha fått gedigen erfarenhet av arbetssättet. Vidare levererades en produkt som vi och produktägaren ansåg leva upp till vad som definierats. Vidare avslutades projektet med en gemensam avslutningsmiddag vilket tydde på att arbetsmiljön genom projektet varit bra.

### 1.2.2 Vad vill vi göra i framtida projekt? (B)

I ett framtida projekt vill vi definiera vad framgång innebär för att använda det som vägledning i arbetet. De *success criterias* som då används bör vara mer mätbara och med fördel då bära en (starkare) relation till de KPIer som används. Detta för att kunna mäta framsteg mot att uppnå det som definierats som ett framgångsrikt projekt men även för att tydligare i slutet kunna bevisa att målen uppnåtts. Att så var fallet att de uppnåddes i detta projekt gjordes på ett mer godtycklig vis genom att det vid slutet *ansågs* vara uppfyllda.

Vidare vill vi i fortsatta projekt använda oss av *sprint goals* då det var ett tydligt sätt att sammanställa vad man ville uppnå med varje sprint och samtidigt rikta resurser mot vad som var viktigast.

### 1.2.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att kunna definiera framgång ska vi inledande i projektet hålla en diskussion för att bilda en gemensam för gruppen syn på vad det innebär. Genom att förklara innebörden och framhäva fördelarna med att ha väl definierade *success criteria* så kan det få en ökad prioritet i planeringen och därmed tas i beaktning. Detta för att sedan kollektivt arbeta mot att uppnå dem. För att kunna ha tydliga och mätbara *success criteria* bör tid avsättas för att ta fram dessa i början av projektet. Dokumentation av de definierade kriterierna kan även vidare bidra till att underlätta att de uppnås. Vad som definierats som framgång ska sedan tas med till diskussionerna om vilka KPIer som skall användas för projektet.

För att ta fram *sprint goals* ska vi vid *sprint planning*-mötena sammanställa eller lyfta ut det viktigast av varje *sprint board* för att därigenom kunna formulera ett mål.

## 1.3 Your user stories in terms of using a standard pattern, acceptance criteria, task breakdown and effort estimation and how this influenced the way you worked and created value

### 1.3.1 Sammanfattning av projektet (A)

Under projektets gång har en enorm utveckling skett kring skapandet av *user stories*. Även om erfarenheten gjorde att utvecklingskurvan pekade uppåt så fanns det aspekter som var svåra att ändra på under projektets gång. Våra *user stories* var i början till stor del horisontella vilket satte ramar för möjligheterna att ta fram vertikala i resterande delar av projektet. Det vill säga då delar av applikationen var färdigutvecklade horisontellt var övergången till vertikal utveckling mycket svår. Dock tilläts det i detta projektet ett till viss del horisontellt utvecklande då de korta ramarna som sprintarnas längd satte, en vecka, gjorde det svårt att ta fram vertikala.

Varje *user story* bestod först av en förklaring kring vad den ämnade att uppnå vilket i termer av värde, för vem och varför. Därefter var de nedbruten i en lista av *tasks* som man skulle genomföra för att ta fram *user storyn*. Efter det följde en specifikation av acceptance criteria för *user storyn*. Vidare hade även varje user story ett estimat och en eller flera tilldelade utvecklare. Till sist fanns även en referering till vår *definition of done* som återfanns i det sociala kontraktet. Inom det flesta av dessa områden kom det under projektet att ske kontinuerliga förbättringar. Det enda undatagen var vår *definition of done* och den inledande värdeförklaringen som vi redan från början upplevde vara tydlig i att fånga värde, för vem och varför den utvecklades.

Nedan följer tillsammans med varje sprint en förklaring till hur väl, eller inte, vårt arbete resulterade i att uppnå någon eller några av INVEST-kriterierna. Tyvärr tillämpade vi dessa endast i en väldigt liten utsträckning under projektets gång. Dock presenterar vi nedan hur vårt arbete förhöll sig till dessa för att kunna reflektera över resultatet och sedan föra vidare det till ett framtida projekt, vilket vidare förklaras nedan vara ett mål.

Under första sprinten så skapades det *user stories* som var horisontella och vagt definierade och som vidare tjänade det främsta syftet av göra gruppen bekväm i utvecklingsmiljön. Deras *tasks* var vagt specificerade samtidigt som *acceptance criteria* var definierade som att alla *tasks* skulle vara genomförda. Av *INVEST-kriterierna* uppfylldes enbart *Independent*, *Estimable* och *Small*. De ansågs inte vara *Negotiable* fastän de kunde diskuteras med produktägaren då *acceptance criteria* var för otydligt definierade för att föra en dialog kring. Estimeringen av varje *user story* bestämdes med hjälp av en *planning poker*-applikation. Alla gruppmedlemmar fick sätta ett eget estimat varvid de två extremvärdena togs bort innan ett medelvärde räknades ut. Estimeringen var dock utmanande då delar av gruppen hade bristande erfarenhet från utvecklingsmiljön samt språket som användes och hade därmed svårt att göra rimliga estimat. Metoden ansågs ändå tillräckligt valid och användes under hela projektet. Estimaten kom även att sakta förbättras med varje sprint. Detta genom dels erfarenhet men även genom att *user storyn* mer systematiskt bröts ner i sina delar och från det byggdes det totala estimatet upp.

I den andra sprinten gjordes en ansats för att bättre definiera alla *user stories*. Deras *tasks* specificerades mer men visades under sprinten fortfarande vara på en något för hög abstraktionsnivå. Deras *acceptance criteria* var fortsatt att alla *tasks* skulle genomföras. En förbättring skedde däremot sett till värdeskapandet då alla *user stories* nu enbart hade ett fokus av att leverera värde till produktägaren. Av *INVEST-kriterierna* uppfylldes nu *Independent*, *Valuable*, *Estimable* och *Small*.

Slutligen för de resterande tre sprintarna hade samtliga delar av formatet för våra *user stories* goda resultatet, om än en möjlighet för mindre inkrementella förbättringar. Dock levde fortsatt det tidigare problemet om att det var svårt att ta fram vertikala *user stories* kvar. *Tasks* var från tredje sprinten mer specifika vilket underlättade estimeringen av arbetsbördan för varje *user story*. Vidare hade alla *tasks* en checkbox som utvecklaren kunde fylla i när den var genomförd. Varje *user story* hade även tydligt specificerade *acceptance criterias* som nu var helt separerade från *tasksen*. Sammantaget underlättade dessa förbättringar för utvecklarna att arbeta med sina uppgifter då de framkom tydligt vad de skulle göra samt vad som förväntades av uppgiften. Till tredje sprinten lyckades vi även utveckla mer vertikala *user stories* efter att vi efter sprint 2 läst på mer hur man kunde ta fram dessa samt bollat förslag med handledarna. Därmed hade vi från tredje sprinten

tagit fram en välfungerande process för att skapa konsekventa *user stories*. Därmed kunde alla av *INVEST*-kriterier uppfyllas genom att de nu var självständiga, förhandlingsbara, gav värde, var estimerbara och små men där det även fanns tydliga kriterier för när de ansågs vara klara genom *acceptance criterias* och *definition of done*.

### 1.3.2 Vad vill vi göra i framtida projekt? (B)

För och främst vill vi att framtida *user stories* ska vara mer förankrade i kundvärde jämfört med detta projekt då värde baserades på en magkänsla och en personifiering av den typiska kunden. Vill vi ha en mer tillförlitlig uppfattning av vad som är av värde för att säkerställa att det varje *user story* ämnar att tillföra är svar på en verklig uppfattning. Detta istället för, en av någon annan, uppfattad bild av vad som är av värde. Därmed kommer inte resurser att ödslas på aktiviteter som inte är värdeskapande.

Vidare är målet i ett framtida projekt att varje *user story* ska genomgående uppfylla alla *INVEST*-kriterier. Vidare att alla *user stories* är vertikala och har *tasks* som uppfyller *SMART*-kriterierna. Alla *user stories* ska vidare ha ett fortsatt standardiserat format och process för att ta fram dem så att utvecklare och produktägare lätt kan förstå innebörden och att arbetet är konsekvent vilket går att förbättringsmöjligheter lättare kan identifieras. Detta kommer då (förhoppningsvis) leda till mer precisa estimat, att varje utvecklare kan jobba effektivare med sina uppgifter under sprintarna, samt att det kommer skapas mer värde under varje sprint.

Vi vill även i framtida *user stories* lägga till en sektion som beskriver exakt hur de ska testas. Detta för att agera som ett komplement till *acceptance criteria* samt underlätta vid *sprint review*. Detta är något som Kniberg (2015) lyfter fram men som inte tillämpades i detta projekt.

### 1.3.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att se till att alla *user stories* tas fram för att fylla en funktion som är av värde ska antingen en produktägare som är bättre insatt och mer representativ för marknaden föras in i projektet. Alternativt att resurser inledande ska läggas på att kartlägga marknadens behov.

För att uppnå bra *user stories* ska det på förhand tydligt definieras en process för hur dessa ska tas fram samt i vilket format. Det vill säga vilka komponenter som ska ingå i varje *user story*. Däribland vill vi då utöver de som användes i detta projekt lyfta in en kortare beskrivning av

exakt hur de ska testas.

För att uppnå att varje *user story* är vertikal krävs fortsatt erfarenhet men även vidare hjälp genom exempelvis teori och stöd från mer seniora utvecklare. Dock kommer sprintarnas längd att utökas för att göra mer plats inom ramen för vilken man ska utveckla en vertikal *user story*.

För att uppnå alla *SMART*-kriterierna kommer varje *task* som *user storyn* brutits ner till att utvärderas mot dessa. Nedan följer vilka de innefattar:

- Specific - alla kan förstå vad som ingår i dem.
- Measurable - specificerat *Definition of Done* där det är tydligt när de är avklarade.
- Achievable - de är utförbara.
- Relevant - Bidrar till *user storyn*.
- Time-Boxed - Definera förväntad tid för utveckling dem.

Med tid kommer varje *task* att mer naturligt att utvärderas utifrån dessa utan att man explicit gör plats för det vilket kommer göra arbetet mer effektivt. Men för att se till att vi skaffar oss fortsatt bra erfarenhet från grunden kommer vi till en början vara mer metodiska i hur vi utvärderar alla *tasks*.

På liknande vis kommer varje *user story* att utvärderas utifrån *INVEST*-kriterierna, vilka följer nedan:

- Självständighet kommer uppnås om de utformas vertikalt.
- De kommer vara förhandlingsbara om det finns tydliga *acceptance criteria*.
- De kommer generera värde om det specificerats tydligt hur, till vem och varför.
- Estimeringen kommer underlättas med tydliga *tasks*.
- Small kan säkerställas genom *task breakdown*.
- Med korrekt definierade *acceptance criteria* kommer de kunna testas utifrån kraven.

Även här för att till en början fortsätta vår utveckling i att ta fram bättre *user stories* och det på ett sätt som vägleds av ett par *enkla* kriterier vilka man ämnar att uppfylla. Detta för att i ett senare skede kunna ta fram *user stories* som uppfyller dessa utan att lika systematiskt behöva lägga resurser på att utvärdera dem.

## 1.4 Your acceptance tests, such as how they were performed, with whom, and which value they provided for you and the other stakeholders

### 1.4.1 Sammanfattning av projektet (A)

Acceptanstesterna har i projektet genomförts i samband med *sprint review* vilket har gjort att produktägaren även varit närvarande vid dessa tester. Det förföll naturligt att integrera acceptanstesterna i denna del av sprinten då acceptanskriterierna ämnar att definiera värdet hos en story och det är vidare produktägaren som prioriterar och har huvudansvar för att driva värdeadderande utveckling. Därmed ansågs produktägaren behöva närvara vid acceptanstesterna. Vidare syftar även *sprint reviewn* till att utvärdera inkrementet vilket likande görs på en *user story*-nivå under acceptanstesterna. Det gjorde det ytterligare mer naturligt att integrera dessa båda processer.

Acceptanstesterna gav värde till utvecklingsteamet genom att på *user story*-nivå få feedback baserat på om vad som utvecklats uppfyllde specifikation. Vidare får acceptanskriterierna ytterligare vikt och vägledning för utvecklaren om dessa sedan även används för att utvärdera *user storyn*. Acceptanstesterna gav även värde till produktägaren genom att likväl på *user story*-nivå erbjuda möjligheten till uppföljning av det som lyfts in i *sprint boarden* baserat på den prioritering som angivits i början av sprinten och att utvärdera att allt producerats enligt specifikation. Efter acceptanstesterna fördes även feedback med tillbaka för att bättre utforma *acceptance criteria* vilket ytterligare kunde stärka värdet av dem.

Under projektet har vi alltså för samtliga *user stories* tagit fram acceptanskriterier vilka de skulle uppnå. Likt tidigare nämnt var dessa till en början löst definierade, ej vidtäckande och tjänade även syftet av att agera som *tasks*. För att förbättra arbetet skedde därför inkrementella förbättringar under varje sprint och efter avstämning tillsammans med handledare under tredje *sprint planningen* hade vi nått en god nivå.

### 1.4.2 Vad vill vi göra i framtida projekt? (B)

I framtida projekt vill vi fortsatt använda oss av acceptanstester då enligt vad som angivits ovan skapar de värde för flera intressenter till projektet. Acceptanstesterna ska även genomföras systematiskt för att säkerställa kvalitet samt användar- och produktägarvärde. Målet är vidare att göra acceptanstesterna till en naturlig del av sprintcykeln för att fortsatt kunna anbringa största möjliga värde.

Vidare som grund för acceptanstesterna ska välformulerade och vidtäckande *acceptance criteria* tas

fram. Stukturerar för att dra med lärdomar från resultatet av acceptanstesterna ska även upprättas för att fortsatt föra med dessa till utformningen av *acceptance criteria*. Detta för att förhindra att brister i definitionen av kriterierna vid något skede orsakar problem.

#### 1.4.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att genomföra acceptanstester bör först och främst dessa naturligt och systematiskt integreras i sprintcykeln. För att hitta ett sådant tillfälle ska den lösning som använts i den här kursen av att göra det i samband med *sprint review* bäras med. Dock bör möjligheten för det utvärderas utifrån omständigheterna av projektet. Dock identifierades många fördelar med detta vilka kan delges deltagarna för *sprint review* för att lyfta värdet av av ett sådant upplägg.

För att kunna genomföra acceptanstester som säkerställer värdeskapande för användare och produktägare är det viktigt att ha tydligt specificerade *acceptance criteria* så att utvecklingsteamet och produktägare är överens om vad det är som ska levereras. Detta även så att man ska kunna utgå från dessa när man avgör om det utförda arbetet ska godkännas eller ej. Vid testerna bär också någon åläggas att ansvara för att dokumentera hur *acceptance criteria* påverkade testningen för att kunna identifiera förbättringsmöjligheter i hur dessa tas fram.

### 1.5 The three KPIs you use for monitoring your progress and how you use them to improve your process

#### 1.5.1 Sammanfattning av projektet (A)

De tre KPIer vi använt oss av under kursen har varit; stress, test coverage samt att ställa antalet avklarade *story points* i relation till vår estimerade *velocity*. Dessa tre valdes då det först täckte in interna gruppaspekt genom att mäta stressnivån. Genom att följa upp test coverage kunde kvalitetsrelaterade aspekt belysas vilket är av intresse för både interna och externa intressenter. Till sist jämförde vi även antalet avklarade *story points* med vår estimerade *velocity* för att se om det verkliga och planerade arbetet var i linje med varandra. Detta för att kunna planera bättre vilket är av värde för både utvecklare och produktägare. Resultatet återges i tabell 5 nedanför.



Tabell 5. KPI - Resultat

Sprint	Stress	Test coverage	Klarade story points / Velocity
1	2.42	0%	59/59 (100%)
2	2.43	96%	49/78 (63%)
3	2.57	64%	44/57 (77%)
4	2.57	71%	56/75 (75%)
5	2.90	62%	50/44 (114%)

**Stress:** Stress var en KPI vi valde då mängden stress gruppmedlemmarna upplever ofta är starkt sammankopplat till hur mycket värde de kan leverera. En stressad utvecklare till exempel, kan tyda på att man tilldelats mer arbete än vad man klarar av medan för lite stress kan tyda på att man har för lite framför sig. Vidare har det visat sig att måttliga stressnivåer kan öka produktiviteten.

Stress är något som ofta är svårt att mäta men sättet vi valde att göra det på var genom att alla i gruppen fick säga en siffra mellan 1 och 5, varefter KPI:n var gruppens genomsnittliga värde. För att eftersträva en måttlig stressnivå sattes målet till att ligga mellan 2 och 3.

Som visas i tabell 5 så uppnåddes målet i samtliga sprinter vilket tyder på att gruppen som helhet hade lagom arbetsbörda genom hela kursen. Därav skedde det inga åtgärder under projektets gång för att uppnå målet.

**Test coverage:** Test coverage valdes som KPI då det är ett sätt att mäta kvalitén på koden vilket både ger värde för utvecklarna genom att bättre kod är enklare att underhålla och utveckla vidare men även för kund då testning av kod leder till färre buggar vilket i sin tur leder till en bättre användarupplevelse. Målet med KPI:n var att minst 70% av alla kodrader skulle täckas av testerna. I resultatet som återges i tabell 5 så framgår det att detta mål dock endast uppnåddes i två av våra sprinter. Anledningen till att KPI:n inte uppnåddes i fler sprintar beror delvis på att mycket av koden som skrevs endast tjänade ett visuellt syfte vilket är något som upptäcktes vara väldigt svårt att testa på ett meningsfullt sätt. Vidare visade sig även testbiblioteket vara svårt att ta fram tester för.

**Klarade story points / Velocity:** Den tredje KPI:n som valdes var att jämföra estimerad *velocity* med gruppens avklarade *story points* då det är något som visar hur bra gruppen har varit på att uppskatta arbetsbördan. KPI:n ger värde till både utvecklingsteamet, genom feedback på ens estimat, samt åt produktägaren som genom att estimaten följs upp och successivt blir bättre kan

enklare planera och prioritera framtida arbetsuppgifter.

Målet med denna KPI var att ligga så nära 100% som möjligt då värden under 100% kan tyda på att gruppen underestimerar arbetsbördan medans värden över 100% kan tyda på att gruppen överskattar arbetsbördan. Vilket framgår i tabell 5 så nådde vi detta mål under endast första *sprinten*. Till skillnad från efterkommande sprintar fanns det dock till första sprinten ingen stack med ytterligare *user stories* från *product backlog* som man kunde ta sig an i mån av tillgänglig tid. Denna stack är vidare anledningen till varför kvoten efter femte sprinten är större än ett.

Under de flesta sprintar låg värdet under målet. Det beror delvis på att det fanns en skillnad mellan gruppmedlemmarna i form av tidigare kunskaper vilket ledde till att samma *user story* egentligen borde haft estimerat som reflekterade vem som skulle utveckla den. Därför började vi efter sprint 2 beakta individuella kapaciteter mer, dock endast genom schablonmässiga överslag. Vidare upplevdes det i allmänhet vara svårt att estimerar arbetsbördan av varje *user story*

Som åtgärd för att nå vårt KPI reflekterade vi efter varje sprint över skillnaden mellan i hur stor arbetsbörda vi estimerade det till att vara med vad den visade sig vara. Detta för att mer aktivt skapa erfarenhet och därmed kunna förbättra oss och nå vårt KPI snabbare.

### 1.5.2 Vad vill vi göra i framtida projekt? (B)

Målet är att i framtida projekt använda KPIer som ett verktyg för att stärka arbetet. Detta genom att först påvisa positiva trender och därmed lättare kunna identifiera best practises och för interna och externa intressenter påvisa framsteg. Vi vill även använda KPIerna för att identifiera områden som är i behov av förbättringar och även påvisa trendavbrott. Det är därför viktigt att välja KPIer som är entydigt och enkelt mätbara.

Viktigt är att dock att de KPIer som vi använder i framtida projekt sätts i kontext av vad som ska produceras. Något som vi dock redan märkte i detta projekt och borde införas i ett nästakommande är något KPI för att mäta customer satisfaction. Vidare vill vi fortsatt använda KPIer för gruppens välmående men då något alternativ till stress. Detta då stress ofta skiljer sig från person till person och därav istället borde mätas individuellt. Det är dessutom ofta svårt att skilja på om stressen som upplevs av en individ kommer från projektet eller från andra faktorer vilket kan göra det till ett för sammanhanget delvis irrelevant mätvärde. Vi vill även i framtida projekt använda oss av verktyg för att följa upp kvalitén hos koden. Att mäta avklarade *story points* i relation estimerad *velocity* anser vi lämpar sig bäst som del av *sprint retrospective* och inte som ett övergripande KPI

att följa då det istället anses passa bättre att upp som del av det mötet.

Om det behövs något KPI för ytterligare områden utöver dessa tre bör det utvärderas utifrån projektet. Viktigt är dock att inte använda sig av för många KPIer utan att istället identifiera de viktigaste områdena och lägga resurser på att endast följa upp dessa.

### **1.5.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )**

För att nå målet bör man inledande diskutera vilka KPIer som kan tänkas vara relevanta att använda. Även i denna diskussion bära med sig om något speciellt område behöver följas upp för att till exempel tillgodose krav från någon intressent. Detta kan antingen ske internt eller i samråd med produktägaren beroende på hur arbetsdynamiken ser ut.

I dessa diskussioner kommer vi dock lyfta fram värdet av att följa upp customer satisfaction, gruppens välmående och kodens kvalitet då vi upplever dessa som viktiga områden. Dock bör det i jämförelse med detta projekt läggas ner mer resurser för att hitta vilket sätt som bäst lämpar sig för att följa upp dessa. Detta genom att identifiera alternativa lösningar och sedan ställa deras för- och nackdelar mot varandra för att hitta den som ger bäst resultat utan att beslagta för mycket resurser för att ta fram och mäta.

## 2 Social Contract and Effort

### 2.1 Your social contract, i.e., the rules that define how you work together as a team, how it influenced your work, and how it evolved during the project

#### 2.1.1 Sammanfattning av projektet (A)

Under projektets första vecka samlades hela gruppen för att skapa ett socialt kontrakt. Innehållet utgick från ett kontrakt som en gruppmedlem använt i ett tidigare projekt och som hade fungerat bra. Detta anpassades och utökades sedan utefter den nya gruppen och projektet genom att varje gruppmedlem lyfte fram tankar och idéer kring vad som borde ingå och inte varefter det diskuterades och specificerades i kontraktet.

Det sociala kontraktet innefattade ett flertal delar som bland annat gruppstruktur i termer av roller, mötesschema, förväntningar och gemensam målsättning. Kontraktet bestod även av flera aspekter relaterade till hur utvecklingsarbetet bör struktureras.

I och med att projektet kom igång upptäcktes flera aspekter som borde täckas in av det sociala kontraktet. Dessa förslag handlade huvudsakligen om att ytterligare förtydliga arbetsstrukturen. För att förhindra att dessa glömdes bort beslutades det om att införa en process för tillägg till det sociala kontraktet. Processen var som följande att en gruppmedlem kunde vid någon mötes-sammankomst lägga fram ett förslag om ändring eller tillägg varvid de närvarande röstade. Om en majoritet godkände förslaget ålades personen som lagt fram det att göra nödvändiga förändringar i det sociala kontraktet.

Då ovannämnda process framförallt riktade sig mot tillägg upptäcktes det under en revidering i mitten av projektet att vissa delar av kontraktet inte längre var relevanta. Därför bestämdes det att det sociala kontraktet skulle gås igenom varannan vecka av hela projektgruppen för att säkerställa att alla punkter var aktuella men även för att påminna medlemmarna om det sociala kontraktets innehåll.

Det sociala kontraktet har gett struktur till arbetet genom grundläggande ramar samt medfört att vi haft tydliga krav och förväntningar på varandra vilket gjort att arbetet lättare kunnat fortskrida. Det har vidare genom tydligt definierade förhållningsregler gjort att konflikter inom gruppen kunnat förhindras.

### **2.1.2 Vad vill vi göra i framtida projekt? (B)**

I framtida projekt vill vi ha en forstätt tydlig struktur på arbetet tillsammans med gemensamma målsättningar och vad som i sin tur förväntas av alla i gruppen. Ett sätt att göra det, vilket fungerade väl i detta projekt är via ett socialt kontrakt.

Målet är då att det kontraktet ska vara byggt på en gemensam förståelse och vara tydligt skrivet så att det inte uppstår olika tolkningar. Vidare att det även direkt från start ska finnas angivelser för hur ändringar till det sociala kontraktet ska genomföras. Även att det ska finnas strukturer som ser till att alla gruppmedlemmar hela tiden är införstådda men vad som står i kontraktet.

### **2.1.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )**

För att ta fram ett socialt kontrakt kommer den process som användes i det här projektet att tas tillvara på. Det vill säga att utgå från ett tidigare använt kontrakt för att ge struktur till mötesdiskussionen om vad som bör ingå men att denna endast används som en stomme varifrån projektspecifika anpassningar och tillägg görs. Vidare viktigt är att se till att alla gruppens medlemmar är närvarande vid detta möte för att låta dem föra sin talan. Från detta möte är det viktigt att alla beslut dokumenteras väl för att skapa tydliga förhållningsregler. I det sociala kontraktet är det viktigt att även direkt ange hur förändringar till det görs. Sedan även att efter att det sociala kontraktet färdigtställts, se till att alla läser igenom och godkänner det.

För att skapa fortsatt struktur som ser till att alla än införstådda med det sociala kontraktet hade vi i detta projekt en gemensam genomgång varann vecka. Detta ansågs som rimligt för detta projekt då var av kort karaktär och det skedde flera förändringar i början. Dock är det en mycket resurskrävande lösning varav i ett längre projekt hade en sådan lösning inte varit eftersträvensvärd. För att avgöra vad som är en lämplig struktur bör man därför sätta det i kontexten hos det framtida projektet men det viktiga är att bära med sig värdet av att införa en sådan.

## **2.2 The time you have spent on the course and how it relates to what you delivered**

### **2.2.1 Sammanfattning av projektet (A)**

Som del av det sociala kontraktet utformades gemensamma förväntningar på varje gruppmedlem gällande den tid man förväntades lägga ner för att bidra till gruppens leverabler. Detta uppgick till ca 20 timmar per vecka vilket även är vad som förväntas för de högskolepoäng kursen ger. För att

följa upp detta har vi vid start av varje ny vecka haft en kort avstämning om ungefär hur mycket tid man lagt på kursen under föregående veckan. Vid dessa tillfällen har det visat sig att samtliga medlemmar har i snitt lagt ner ca 20 timmar per vecka. När någon har haft mycket annat att göra en vecka så har det kompenserats den efterföljande vecka.

Vi valde dock att inte göra den tid man lagt ner till en formell KPI då det är mer intressant att mäta *output* snarare än *input*. Vidare frångick vi en tidslogg för att uppmuntra till eget ansvar. Under de tre första sprintarna lade vi mycket tid på möten, som *sprint planning* och gruppreflektioner. Detta valet grundades i det tidigare nämnda huvudmålet med kursen om att lära oss och skaffa så mycket erfarenhet som möjligt av *Scrum*. Det gjorde dock att vi således hade en låg produktivitet i form av vilket värde som skapades gällande applikationen.

Produktiviteten i kursen såg dock i och med fjärde sprinten en kraftig ökning. I början krävdes det som sagt mycket tid för att skapa en liten *output* men allteftersom vi lärde oss mer, och blev mer bekväma i arbetssättet och utvecklingsmiljön kunde vi generera betydligt mycket mer värde genom den tid vi lade ner. Det gjorde att vi under de två avslutande sprintarna kunde tillföra mycket funktionalitet till applikationen utan att behöva utöka den tid vi lade ner samtidigt som vi fortsatt lärde oss mer om *Scrum* genom att tillämpa mer mogen kunskap. På så sätt omsattes tiden mot slutet till att generera högt värde för externa intressen, applikationen, och interna intresse, erfarenhet av *Scrum*. Hade projektet dock fortsatt vidare skulle denna funktion avta då de features som har största värde utvecklas först och att det sedan är en avtagande effekt hos det värde som skapas.

### 2.2.2 Vad vill vi göra i framtida projekt? (B)

Viktigt i framtida projekt är att i början av projektet tydligt definiera vilken tid som förväntas läggas av alla gruppmedlemmar. Först och främst för att det bidrar till att tydliggöra hur man delar arbetsbördan. Även vidare att den tid som identifieras finns tillgänglig sätter ramar för vilket scope man kan ta sig an i projektet. Tillgänglig tid är en av de viktigaste resurserna att redogöra för. Den tid som man formellt förväntas lägga är ofta definierad i kurser eller arbetskontraktet men det är viktigt att se till att den tid även förväntas omsättas.

Lika viktigt, om än viktigare, som att ta fram den input i form av tid som finns tillgänglig är att effektivt omsätta den till att skapa värde. I ett framtida projekt ser vi därför gärna att man redan vid start lägger stor kraft vid att effektivisera processer och kontinuerligt identifierar förbättringsområden.

Vidare även lägga mindre tid på möten. Dessa fyller ofta en viktig funktion men det är viktigt att se till att de används som ett verktyg för att öka produktiviteten och inte har motsatt effekt som ta anspråk på mycket resurser.

### 2.2.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att uppnå målet ska vi i likhet med detta projekt inledande inom gruppen diskutera den tid som finns disponibel för projektet baserat på vad alla förväntas lägga ner. Sedan att bära med sig detta till andra aspekter relaterade till projektet som diskussioner kring scope.

För att få mer värde ur den tid som läggs ner till framtida projekt ska den erfarenhet vi fått av det här projektet tas med. Till exempel om hur att ha separata möten för *backlog refinement* och *sprint planning* leder till att alla har mer energi vid båda dessa och produktiviteten således kunde öka.

För att se till att det förekommer ett kontinuerligt arbete med att identifiera förbättringsområden bör värdet av detta lyftas fram. Detta för att alla gemensamt ska ha en förståelse för hur det gagnar projektet och man således kan skapa motivation för alla att bidra.

En teknik som förespråkas av Kniberg (2015) för att öka produktiviteten och som tyvärr inte användes till fullo i detta projekt är att *time-boxa* aktiviteter. Till exempel att på förhand bestämma att efter en timmas *sprint planning* så avbryts mötet. Det kan göra att produktiviten ökar då man har en fix mängd input och således endast kan fokusera på det mest värdeskapande arbetet för att hinna med.

## 3 Design decisions and product structure

### 3.1 How your design decisions support customer value

#### 3.1.1 Sammanfattning av projektet (A)

Målsättningen genom hela projektet har varit att alla beslut som fattas ska återspegla att ett positivt kundvärde skapas (när applicerbart). Detta gjordes genom att innan ett beslut fattades så diskuterades om och hur det kan påverka värdet för kund. Om det hade en påverkan på kundvärde, skulle det beslut som skapar mest värde väljas.

Under andra veckan beslutades det först om att ta fram en mobilapplikation och för det andra att utveckla den i en miljö som tillät att slutprodukten kunde användas på både *iOS* och *Android* baserade telefoner. Beslutet om att utveckla en mobilbaserad applikation byggde dels på att konkurrerande lösningar alla använder en mobilbaserad applikation och dels för att kunder ansågs värdesätta denna typ av lösning mest då den skapar den enklaste användarupplevelsen för en användare on-the-go. Att detta var fallet kom från en diskussion där den typiska användare personifierats och olika lösningar diskuterades därefter. Efter att en mobilapplikation hade fastställts vara den mest lämpliga lösningen beslutades det även om att utveckla mot både *iOS* och *Android*. Detta val grundade sig i två anledningar. Först att utvecklingsteamet hade telefoner från bägge plattformar och att det därav fanns ett internt intresse av att lösningen skulle kunna användas av alla. Vidare, genom att skapa en lösning mot bägge plattformarna kunde en större kundbas nås vilket ytterligare stärkte värdet av applikationen för plattformens ena kundbas, de som erbjuder parkeringar. Det fanns självklart motargument mot båda dessa beslut såsom att det lade till plattformsspecifika problem som krävdes att ta hänsyn till. Däremot ansågs fördelarna med besluten väga upp nackdelarna.

Under andra veckan beslutades det också att applikationen skulle utvecklas med *React Native*. Valet gjordes främst baserat på att det möjliggjorde att köra samma kod på båda mobilplattformarna. Därmed ökade hastigheten med vilken applikationen kunde utvecklas och därav värdet för produktägaren genom *economies of speed*. En annan anledning till att *React Native* valdes var att flera av gruppmedlemmarna redan hade kunskaper i det snarlika *React*. Dessa gruppmedlemmar kunde därför bidra till att lära upp resten av gruppen vilket återigen ökade hastigheten av utvecklingen och värdeskapandet för produktägaren.

Utöver *React Native* valdes också utvecklingsplattformen *Expo*. Detta eftersom den gjorde det enklare att testa applikationen direkt på en riktig telefon istället för på en emulerad telefon i datorn.



Utvecklarna kunde därför få en bättre överblick över hur applikationen presterade och såg ut i målmiljön och användarupplevelsen kunde därmed kontinuerligt stämmas av mot.

Gällande valet av APIer för att hämta parkeringsplatsinformation fanns det flera att tillgå. Först letades olika alternativ upp tillsammans med beskrivningar för hur de kunde användas. Till femte och projektets sista sprint hade två av dessa implementerats. Dessa båda var Parkering Göteborg, som hade en stor mängd parkeringsplatser att hämta data ifrån (något över 500 parkeringsplatser) samt Q-Park. Valet att börja med dessa två grundades dels i att de hade enklare APIer att arbeta med vilket ökade produktiviteten och dels i att de resulterade i ett stort antal parkeringar att välja mellan för kund. Under projektets slutskede ändrades däremot något med Q-Parks API och den datakällan slutade fungera vilket gjorde att vid prestationstillfället fungerande endast en datakälla för parkeringsplatser hos slutprodukten.

Utöver APIerna för att hämta parkeringsdata krävdes det även ett kartbibliotek då en stor del av produkten byggde på användningen av kartor. För detta valdes biblioteket *React Native-Maps*. Valet gjordes med bakgrund av det ansågs kunna tillhandahålla all nödvändig funktionalitet samt att det var det mest mogna kartbiblioteket för *React Native*, bland annat för att den var skapad och upprätthållen av ett stort och seriöst företag, *Airbnb*. Vidare att den även stödjer både *Android* och *iOS*. Det var dess möjlighet att erbjuda nödvändig funktionalitet tillsammans med stabilitet och stöd för bägge plattformar som ansågs skapa det positiva kundvärde som därmed ledde till valet av API.

Under projektets gång ansågs det även skapa stort kundvärde att tillhandahålla en sökfunktion som ger relevanta sökresultat för användare som inte vet exakt vad en parkeringsplats heter men som har en måldestination. Det ställde krav på API som kunde ge sådana funktionalitet. För detta valdes *Open Street Maps* då den erbjöd funktionaliteten gratis samtidigt som det var relativt lätt att implementera.

### **3.1.2 Vad vill vi göra i framtida projekt? (B)**

Målsättningen för ett nästkommande projekt är låta olika val styras av vad som skapar positivt kundvärde. En stor fördel med detta, utöver att det ökar värdet av slutprodukten, är att det kan guidea olika val. Genom att hela tiden ha kund i åtanke vid olika designval ödslas inte resurser på något som inte skapar värde.

I övrigt är det olika val som fattas unika för de omständigheter som finns kring varje projekt varvid inga specifika val i dagsläget eftersträvas att bäras med. Men den erfarenhet som skapats av till exempel olika APIer och utvecklingsmiljöer kan ligga grund för hur kundvärde kan stödjas i framtida projekt.

### 3.1.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att bättre säkerställa att alla val är i enlighet med vad som skapar kundvärde bör man, i likhet med vad som tidigare påpekats, mer genomgående analysera kundbehov. Värdedefinitionen i detta utvecklingsprojekt byggde mycket på en magkänsla och uppfattning om vad som gav värde baserat på en personifiering av slutkunden. Designvalen hade i en större utsträckning kunnat vara värdeadderande om kundvärdet hade varit mer specificerat. För att specificera detta bättre hade man antingen eftersträvat en mer väl insatt produktägare eller att någon typ av marknadsundersökning hade genomförts.

För att i efterföljande steg fånga och skapa något av värde för kund bör olika alternativ utvärderas för att hitta den eller de som är mest lämpliga. Detta genom att diskutera de olika alternativens för- och nackdelar i deras förmåga i att skapa ett värde för kund.

## 3.2 Which technical documentation you use and why

### 3.2.1 Sammanfattning av projektet (A)

Under hela projektets gång använde vi oss endast av en *mockup* som dokumentation. *Mockupen* ansågs ge stort värde till projektet genom att stödja kommunikation mellan utvecklingsteamet och produktägaren men även inom utvecklingsteamet. Först såg *mockupen* till att det fanns en gemensam förståelse mellan produktägare och utvecklingsteamet om vad som skulle utvecklas. Det gjorde att redan från start kunde förväntningar läggas i linje med vad som kunde produceras baserat på tillgängliga resurser. Vidare såg även *mockupen* till att det fanns en gemensam syn på applikationen inom teamet vilket starkt förenklade att koordinera arbetet.

Det fanns en målsättning om att skapa ett komponentdiagram men värdet av detta, enligt den prioritering som givits av produktägaren, var inte tillräckligt hög i relation till övriga utvecklingsprojekt för att skapa utrymme för att ta fram en. Sett till detta projekt ansågs den dokumentation

som använts vara tillräcklig.

Valet att frångå formell dokumentation i detta projekt byggde på premissen om att gruppmedlemmarna var införstådda med de olika komponenterna. Vidare att de resurser som skulle krävas för att ta fram dokumentation inte vägdes upp av fördelarna i relation till detta arbetets scope. Detta beslut visade sig även hålla vid projektets slut då det under projektet inte uppstått någon problematik gällande produktens struktur. De få frågetecken som har uppkommit har istället kunnat lösas genom frekvent dialog mellan utvecklarna och en allmän förståelse hos alla i teamet för produktens olika komponenter och hur de hängde ihop.

### **3.2.2 Vad vill vi göra i framtida projekt? (B)**

Målet i framtida arbeten är att använda sig av olika tekniska modeller och dokumentation på ett sätt som skapar värde för både utvecklare och produktägare. För utvecklare genom att det hjälper till att ha en enad bild om vad som ska utvecklas och vad som skapats hittills i projektet. För produktägare som får en bättre inblick i vad denna kommer få levererat och hur det som utvecklats hänger ihop. Det senare vilket starkt kan underlätta fortgående underhållningsarbete och vidareutveckling. Dokumentationen kan således användas för att skapa ytterligare värde. Vidare kan dokumentation vara användbart för produktägaren genom att denne tidigt och enkelt kan få en överblick och därmed erbjuds möjligheten att ge feedback till utvecklarna.

Värdet av dokumentation bör dock sättas i kontext hos det framtida projektet. Om upprättandet och underhållet av dokumentation anses ta fokus och resurser från vad som är mer värdeskapande bör det dock åsidosättas men detta kan ibland vara något som är svårt att avgöra. På liknande vis, om projektet exempelvis är av en större, mer komplex och långsiktig karaktär där utvecklings-teamet även är mer flytande bör värdet av dokumentation tydligt lyftas fram.

### **3.2.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )**

För att nå målet bör man utvärdera syftet av projektet för att i relation till detta ställa värdet av dokumentation. I denna utvärdering är det också viktigt att beakta alla olika stakeholders synpunkter. Produktägaren kan argumentera för att dokumentation blir ett sidospår som inte anbringar mycket värde. Det är då viktigt att ge utvecklarna utrymme för att påvisa hur dokumentationen kan stödja intern och extern kvalité och att det vidare skänker struktur till arbetet vilket kan förhindra att framtida resurser behöver läggas på att, som Kniberg (2015) beskriver, betala tillbaka

“technical debt”. Men även hur det vidare, likt tidigare nämnt, kan ge värde åt produktägare genom att underlätta fortgående underhållningsarbete och vidareutveckling av produkten.

### 3.3 How you use and update your documentation throughout the sprints

#### 3.3.1 Sammanfattning av projektet (A)

Dokumentation är något som har funnits med genomgående sedan starten av projektet. Det sattes tidigt upp muntliga mål i gruppen om att koden skulle vara tillräckligt dokumenterad för att alla i gruppen skulle förstå den skrivna koden. Detta för att underlätta för alla att underhålla och utöka koden samtidigt som vi ville undvika krav på onödig dokumentation som inte bidrog till detta mål utan snarare bara gjorde koden mer svåröverskådlig. För att säkerhetsställa att dokumentationsstandarden uppnåddes bestämdes det att gruppmedlemmar skulle kontrollera att detta uppfylldes vid *code review*. Detta gjorde att koden var förståelig för minst 4 gruppmedlemmar, den som skrev koden samt de tre som gjorde *code review*, vilket var en majoritet av gruppen.

#### 3.3.2 Vad vill vi göra i framtida projekt? (B)

Målet i framtida projekt är att fortsatt använda sig av dokumentation för koden. Dokumentation är något som alltid är viktigt i olika programmeringsprojekt. Delvis för den som skrev koden då man ofta snabbt glömmer bort hur koden fungerar när man inte aktivt arbetar med den. Men ännu viktigare är däremot dokumentation för andra som skall jobba med eller använda sig av koden då det kan spara mycket tid och undvika onödig förvirring. Samtidigt ska inte koden överdokumenteras med onödig eller intetsägende information. Det är en balansgång som kan vara svår att gå men svårigheten framtalar inte vikten av att i fortsatta projekt se till att koden dokumenteras.

#### 3.3.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att se till att dokumentationen håller sig rätt under balansgången mellan för lite och för mycket ska vi, likt i detta projekt, granska varandras kod. Detta för att avgöra om koden är tillräckligt väl dokumenterad för att vara förståelig utan att vara överflödigt vilket därigenom skapar en god utgångspunkt för att ta fram hållbar kod.

För att se till att denna granskning genomförs anser vi det lämpligt att göra det som del av en formell process. Fördelaktigen, likt under detta projekt, vid *code review*. Det är dock viktigt att

gemensamt se till utvecklingsteamets uppfattning av värdet av dokumentation för koden för att inte låta denna process skyndas igenom. För att uppnå målet gällande god dokumentation ska uppnås är det därav viktigt att utöver att upprätta processer för att säkerställa granskning av dokumentation även se till att motivation till att från början skapa den och sedan även kritiskt granska den finns i gruppen. Detta kan lämpligen åstadkommas genom att i början av projektet diskutera inom gruppen, den gemensamma synen på dokumentation och hur det kan agera som ett verktyg, dels för att nå slutprodukten men även för att underlätta vägen dit och underhållet därefter.

### 3.4 How you ensure code quality and enforce coding standards

#### 3.4.1 Sammanfattning av projektet (A)

Till början av första sprinten hade *code conventions* fastställts som del av det sociala kontraktet. Dessa syftade till att upprätta gemensamma konventioner för utvecklingsarbetet vilket därigenom underlättade att uppnå en intern kvalitet hos projekt. Till detta infördes det även för all kod som skulle mergas in i utvecklings-branchen att genomföra en *code review*. *Code review* ämnade att säkerställa funktionaliteten hos det som utvecklats samt att det uppfyllde alla konventioner och, som tidigare nämnt, var väl dokumenterad. För att kunna merge in det i utvecklings-branchen beslutades det att tre andra personer i utvecklingsteamet behövde granska och godkänna det som skapats. Därigenom blev den interna kvalitén ett kollektivt ansvar hos gruppen. Vidare, även att koden var tvungen att godkännas av en majoritet (fyra utav sju utvecklare) av gruppen. Om koden fick avslag under *Code review* ålades utvecklaren att åtgärda det som orsakat att den inte godkänkts. För att merge in något i master-branchen krävdes det att alla i gruppen godkände koden, detta var däremot något som endast gjordes i slutet av kursen då produkten för första gången “produktionssattes” i och med presentationen av applikationen.

Således användes för projektet ett git-arbetsätt kallat *Gitflow*. Det går ut på att man alltid har en utvecklings (*develop*) branch där allt ska vara i ett relativt fungerande stadie. Från denna skapas olika typer av branches för vidareutveckling av produkten. Till exempel en *feature*-branch för att lägga till ny funktionalitet, en *fix*-branch för att lösa en bugg eller liknande. Utöver detta finns även en *master*-branch som representerar vad som ska vara i produktion, det vill säga det som levereras till användaren. Denna användes endast i slutet av projektet inför slutpresentationen då applikationen “släpptes” för första gången. Genom att använda detta arbetsätt kunde vi upprätta checkpoints, in-mergning till utvecklings- och masterbranchen, som kontrollpunkter där kvalitén kunde säkerställas vidare.

Utöver detta användes Travis CI vilket är ett *Continuous Integration* verktyg som varje gång något pushas till GitHub automatiskt exekverar alla tester samt *tslint*, en stilcheck för *Typescript*. Om någon av dessa checkar fallerar kommer kodförslaget blockeras tills problemet blivit åtgärdat. De tester som kördes av Travis skrevs huvudsakligen för att säkerställa att alla komponenter som användes skulle fungera. Dock, på grund av ett svårbemästrat testbibliotek och att flera av gruppmedlemmarna inte hade erfarenhet av att ta fram tester begränsades arbetet med att ta fram sådana. Detta låg dock fortfarande i linje med vad som satts upp i *definition of done*. Det vill säga att det i *DoD* fanns ett krav om att all kod skulle vara testad till utvecklarens bästa förmåga.

Både hur Travis användes samt vilka uppgifter Travis genomförde ändrades under projektets gång. I början sattes Travis upp för att köra testerna men det dröjde till tredje *sprinten* innan stilcheckning av koden lades till. Detta gjordes efter att det visat sig förekomma viss skillnad i hur de olika gruppmedlemmarna strukturerad sin kod.

### 3.4.2 Vad vill vi göra i framtida projekt? (B)

Kodkvalitet är något som särskilt i större projekt kan bli avgörande för ett projekts hållbarhet och livslängd. Utan krav på standarder hos en kodbas kan den snabbt bli rörig och svårutvecklad. Det kan dessutom leda till betydande utökning av utvecklingstiden vilket i sin tur leder till ökade utvecklingskostnader samt lägre produktkvalitet. Av dessa anledningar är det viktigt att tidigt i, och om möjligt före, utvecklingsfasen sätta upp gemensamma kodstandarder och stilramar. Vidare vill vi för att säkerställa att dessa efterföljs upprätta både automatiska och manuella kontroller för att garantera att all kod följer de standarder som satts upp. I framtida projekt vill vi även skapa mer utrymme för att ta fram bättre tester för koden.

### 3.4.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

Inför framtida projekt vill vi ta med oss de processer vi upprättade och använde under detta projekt. Till detta faller *Gitflow* arbetssättet, *code reviews* samt *code conventions*. Men för att i större utsträckning nå målet bör användningen av dessa verktyg utökas ytterligare. Exempelvis kan tydligare angivelser för *code reviews* tas fram och på samma sätt kan *code conventions* tydliggöras och utökas. Vidare bör man innan projektets start samla utvecklingsteamet för att upprätta dessa. Detta för att förhindra att stor del av arbetsbördan senare i projektet riktas mot att uppdatera tidigare producerad kod. Inledande i projektet kommer även de verktyg och rutiner som behövs

för att genomföra automatiska och manuella kontroller tas fram.

För att i framtida projekt se till att det genomförs fler och bättre tester kommer värdet av att ge utrymme för att ta fram dessa understrykas mer. Som tidigare nämnt orsakade resursbegräningar, i form av kompetens och i en vidare bemärkelse tid, att det inte genomfördes i detta projekt. I framtida projekt behöver man därför i en större utsträckning frigöra resurser för att nå målet om bättre testning av koden.

## 4 Application of Scrum

### 4.1 The roles you have used within the team and their impact on your work

#### 4.1.1 Sammanfattning av projektet (A)

Under projektet har vi använt oss av följande roller som även sedan beskrivs mer ingående nedan:

- Scrum Master
- Sekreterare
- Produktägare och vice produktägare
- Utvecklingsteam

Under första arbetsveckan och därmed innan utvecklingsarbetet påbörjades tillsattes en av gruppens medlemmar till att vara *Scrum Master*. Vi gjorde detta för att så tidigt som möjligt i projektet skapa struktur i arbetet. *Scrum Mastern* tilldelades att förbereda och ansvara för att hålla alla sammanträden under projektets gång. Vidare ansvarade även *Scrum Mastern* för att se till att alla deadlines efterlevdes av gruppmedlemmarna.

Genom att *Scrum Mastern* tog över administrativa uppgifter skapades en tydlig arbetsfördelning och därigenom utrymme för teamet att fokusera på de uppgifter som låg framför dem. Under hela projektets gång var det en och samma person i projektgruppen som höll rollen och det skedde inga förändringar gällande ansvar och uppgifter. Fördelen med att samma person hade rollen var att denna hann då mogna i rollen och störningar till projektet, på grund av ansvars-rotationer, kunde förhindras.

Första veckan tillsattes även en sekreterare för att föra protokoll vid alla sammanträddanden. Detta för att säkerhetsställa att det som beslutades vid dessa möten även sparades.

Till första *sprint planningen* fanns ingen förutbestämd produktägare. Det gjorde att värdefokuset istället blev ett kollektivt ansvar inom gruppen. I sin tur gjorde detta att processen var ostrukturerad eftersom ingen förde en tydlig värdelinje och vidare blev det en ansvarsfördelning att föra denna linje vilket kan ha fräntagit motivation. Efter att ha insett detta, tillsattes under första sprinten två av gruppmedlemmarna att agera produktägare och vice produktägare. Den vice produktägaren tillsattes dels för att stödja produktägaren i fokuset på värdeskapande och dels för att agera beställare på det som utvecklats av produktägaren eftersom denne även var del av utveck-



lingsteamet. I och med att de båda produktägarna tillsattes kunde värdeskapandet hos varje *user story* få ett större fokus. Produktägarna såg även till att utvärdera värdet av det som hade skapats under sprinten vid *sprint review* och hur detta inkrement påverkade arbetet under nästa *sprint*. Att utvärdera och sätta varje inkrement i sin kontext hos slutprodukten gjorde att vi lättare kunde röra oss mot en mer agil process eftersom det uppmuntrade till vidta nödvändiga förändringar. Som exempel kan nämnas att det efter *sprint reviewen* skedde förändringar gällande prioritet i *product backlogen* samt att nya *user stories* togs fram då man genom att visa upp och diskutera applikationen kunde identifiera nödvändiga funktioner.

Att två av gruppmedlemmarna tillsattes till att agera produktägare kan dock ha burit med sig komplikationer. Detta i form av att då produktägarna även var del av utvecklingsteamets kan de ha blivit starkt påverkade av teamets syn på värde istället för vad som var kund- och ägarvärde. Vidare förespråkas det i litteratur och av de agila principerna att produktägaren inte ska detaljstyra utan lämna ifrån sig ansvar till utvecklingsteamet. Detta är något som starkt försvårades av att produktägaren var del av utvecklingsteamet eftersom denna då hade kännedom om detaljer vilket lättare bjöd in till att ta upp detaljaspekt i diskussioner. Dock ansågs lösningen vara rimlig baserad på de förutsättningar som fanns. Detta då vi eftersträvade schemamässig flexibilitet, korta iterationer och att produktägaren, enligt vad Kniberg (2015) starkt förespråkar, är del av *sprint planning*. Vidare bedömdes även produktägarna inneha tillräcklig kännedom om de båda kundgrupperna för att kunna framhäva vad som var av värde. Detta är även en anledning till varför vi lät samma personer vara produktägare och vice produktägare genom hela projektet. De ansågs som sagt ha tillräcklig kännedom om kund och vidare ville vi inte ha återkommande omprioriteringar i *product backlog* baserat på skiljande uppfattningar om vad som ansågs skapa mest värde.

Alla gruppens medlemmar var även del av projektets utvecklingsteam, *the scrum team*. Utöver det direkta värdeskapandet genom att utveckla applikationen bidrog teamet även genom att ta fram *user stories*, bryta ner dessa i *tasks* och estimerade dem.

Även om alla de roller vi kom att använda redan tillsattes före och under första sprinten, var det för samtliga av dessa ytliga förståelser av vad de innebar. Till andra sprinten ålades därför alla att läsa på om den roll man hade. I och med denna teoretiska ansats kunde värdeskapandet stärkas genom att rollinnehavarna då bättre förstod sina uppgifter och befogenheter och således kunde utnyttja dem. Ett exempel är den teknik Kniberg (2015) skriver om där produktägaren ser till att en *user story* delas upp för att skapa utrymme och därmed kunna lyfta in den i *sprint backlogen*. Under efterföljande sprinter skedde en fortsatt inkrementell förfining av hur man hanterade rollerna. I och med att man blev mer bekväm i rollen kunde sammanträden som *sprint planning* och *sprint*

*review* effektiviseras. Detta märktes tydligt mot de två sista sprintarna då alla vid det laget visste vad de skulle göra och sammanträdena kunde således genomföras under kortare tid och med bättre resultat än under de inledande sprintarna.

#### 4.1.2 Vad vill vi göra i framtida projekt? (B)

Ett mål för framtida projekt är en tydlig roll- och ansvarsfördelning från start samt se till att gruppen har en gemensam förståelse för dessa. Detta då det leder till ett effektivare arbete där var och en vet vad de själva och andra ska göra. Ett exempel där tydligare ansvarsfördelning bör införas är gällande de strukturella aspekten hos *the product backlog*.

Även att produktägaren, med bakgrund av de komplikationer som nämnts ovan, inte bör vara del av utvecklingsteamet och vidare att denna bör ha ännu större förståelse för kund. I det fall av att produktägaren är extern till utvecklingsteamet, att även då bara ha en sådan, det vill säga ingen vice. Detta för att behovet i det fallet fräntas och därmed fokus flyttas till att bevara resurser.

#### 4.1.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att uppnå en tydlig roll- och ansvarsfördelning som alla är införstådda med ska rollerna introduceras väl och diskuteras om i gruppen innan projektets start. Att vid detta möte då bära med sig de insikter man erhållit från detta projekt för att dela med sig av sin egen erfarenhet samt sätta andras uppfattningar i kontext av hur man själv ser på rollerna.

Viktigt att bära med sig i ansvarsfördelningen är dock avvägningen mot att eftersträva *self organized teams*. Vid upprättande av de olika rollerna och ansvaren bör dessa växa fram inom teamet och inte styras uppifrån. Då det senare skulle bryta mot den agila principen om att ha *self organized teams*.

För att skapa struktur i *product backlogen* och se till att formateringen är konsekvent bör någon åläggas detta ansvar. Detta kan antingen tilldelas produktägaren, som redan har det yttersta ansvaret för denna (Schwaber & Sutherland, 2017), eller att produktägaren överlåter det strukturella ansvaret till någon i utvecklingsteamet. Vilket alternativ som är att föredra bör sättas i kontext till det projekt som undertas och därmed de omständigheter som då finns.

För att vidare se till att produktägaren inte är del av utvecklingsteamet bör de komplikationer som de gav upphov till bäras med. Hur man i sin tur uppnår att produktägaren hålls extern till utvecklingsteamet bör utvärderas utifrån de omständigheter som uppkommer gällande projektet. Men för att vidare uppnå målet om att produktägaren har en djupare förståelse för vad som är av värde bör denne antingen komma från ett håll av att ha mer kunskap om användarna eller resurser för att skaffa denna förståelsen.

## 4.2 The agile practices you have used and their impact on your work

### 4.2.1 Sammanfattning av projektet (A)

Under projektet har samtliga av de 12 *Principles of the Agile Manifesto* använts men i varierande utsträckning. Dessa har uppnåtts genom att utnyttja olika roller samt genom olika aktiviteter/processer. Ovan beskrevs de roller som använts och nedan följer en beskrivning av de olika aktiviteter/processer som använts och hur de har stött någon av de agila principerna och vårt arbete.

Först och främst har allt utvecklingsarbetet ämnat att vara svar på något som är av värde för kund. Att ha ett kundorienterat tillvägagångssätt är något som lyfts fram en grunddel inom agil utveckling. För att uppnå detta har vi därför genomgående reflekterat och utvärderat vad vi ämnar att skapa utifrån vad som anses ge värde.

Vidare har utvecklingsarbetet varit uppdelat i *sprinter*. Under projektet hann fem *sprinter* genomföras som alla med undantaget av en, hade en veckas längd. Dessa löpte från måndag till måndag. På grund av omtentavecka och röda dagar utökades vår andra *sprint* till att vara 3 kalenderveckor lång. I boken *Scrum and XP from the Trenches* skriver Kniberg (2015) att en veckas sprint ofta är för kort vilket märktes under projektet. Den korta sprintlängden gjorde att det blev svårare att skapa vertikala *user stories* då just vertikala, värdebringande, användarupplevelser var svåra att hinna utveckla under den tidsperioden. Värdeskapandet anpassades således till utvecklingsprocessen då de ramar de korta sprinters längd satte gjorde att vi tillät horisontella *user stories*. Däremot erbjöd korta sprintlängder korta feedback-cykler vilket passande ändamålet av att lära oss mer om utvecklingsprocessen. Kursens tidsbegränsning gjorde vidare att för att uppnå målet om att ha genomfört flera sprinter krävdes det att dessa var av en kortare längd. Upplägget tillät även för att uppnå principerna för frekvent leverans av produkt och att vid regelbundna intervaller reflektera över arbetet.

Till första sprinten hade vi en löst strukturerad *sprint planning*. Vidare fanns det vid detta möte

ingen *product backlog* utan istället hanterades både att ta fram denna och sedan en *sprint backlog* under benämningen av *sprint planning*. Bristen på struktur och den bredd av uppgifter som skulle undertas gjorde att vi visserligen tog fram *user stories* för första sprinten men hos dessa följde det med en del strukturella problem. Utöver en ofta undermålig definition av det värde som *user storyn* skapade så var de löst specificerade. Vidare var *the tasks* skrivna på en hög abstraktionsnivå och användes även som *acceptance criteria*. På grund av de problem som uppkom gällande *user stories*; som horisontellt värdeskapande, otydliga *tasks* och estimat samt tvetydigheter under *sprint review* på grund av ospecificerade *acceptance criteria*, så lades stort fokus vid att förbättra *sprint planning* processen. Till andra sprinten gjordes detta genom att bära med oss de misstag som uppkommit från sprint ett. En direkt förbättring var att vi redan till detta möte hade en *product backlog*, om än i mindre utförligt skick. Vidare förbättringar var att våra *user stories* specificerades något mer utförligt och *acceptance criteria* skiljdes från *tasks* men i övrigt kvarstod problem, som horisontella *user stories* och en ostrukturerad planeringsprocess. Till starten av sprint 3 beslutades det därmed att alla gruppmedlemmar skulle repetera teorin i ämnet för att ta en teoretisk ansats i hur vi genomförde processen. Repetitionen av teori tillsammans med den erfarenhet vi skaffat gjorde att det började lossna. I den mån där det var möjligt skapades vertikala stories, tidigare horisontellt skapande satte vissa ramar, vidare användes en strukturerad process och *user stories* ansågs väl specificerad även efter närmare utvärdering med handledare.

Under tredje sprinten infördes ett separat tillfälle för *backlog refinement*. Detta gjordes då vi efter sprint ett och två upplevt att *sprint planning* mötena tagit väldigt lång tid. Kniberg (2015) förespråkar att *sprint planning* möten, som tumregel, är en timma per sprintlängdsvecka. Då vi fann att dessa möten blev uppemot tre timmar långa ansåg vi att det behövdes göras något åt saken. Gruppen beslutade därför att på torsdagar, dvs på en annan dag än då *sprint planning* genomfördes, hålla ett möte på en timma dedikerat åt *backlog refinement*. Sammantaget gjorde detta att vi hade med oss en bättre *product backlog* in i *sprint planning*-mötena och kunde därmed producera en bättre *sprint backlog*.

Under sprint fyra och fem skedde inga strukturella förändringar till *sprint planning* processen. Istället fokuserade vi på inkrementella förändringar som gjorde att arbetet blev effektivare.

Genom *sprint planning* har flera agila principer kunnat följas. Vi såg till att högsta prioritet var att leverera något av kundvärde. *Sprint planning* erbjöd vidare en mötesplats för flera stakeholders till projektet att samarbeta och detta genom ett fysiskt möte.

Vid första sprinten infördes även *daily scrum*. Detta genomfördes genom att en bot i *Slack* varje

dag bad alla i gruppen att senast kl 10 svara på följande frågor:

- Vad har du gjort igår?
- Har du stött på problem du behöver hjälp med?
- Vad ska du göra idag?

Att gruppmedlemmarna genom *daily scrum* varje arbetsdag frågades vad de skulle bidra med till projektet kan ha bidragit till att många varje dag fann sig arbeta med något av värde till projektet. Vidare erbjöd *daily scrum* till att det vid vissa tillfällen startades diskussioner kring olika hinder man upplevde. Ett problem som uppkom var dock att gruppmedlemmarna sällan gick in och kollade vad personer efter dem hade skrivit vilket motarbetade syftet. Vidare uppstod även ett problem i att frågorna ofta besvarades med kortfattade och intetsägande formuleringar. Dessa var båda troligen konsekvenser av att vi valde att hålla *daily scrum* över ett digitalt medium. Fördelarna med att använda en sådan lösning ansågs dock väga mer än den problematik som hade uppkommit av att anordna fysiska träffar varje dag. Detta då projektet genomfördes på halvfart och gruppmedlemmarna således hade andra åtaganden som hade gjort att närvaron vid dessa hade varit mycket låg. Vi valde dock efter sprint 3, för att stärka värdet av *daily scrum*, att uppmana alla till att ta vara på aktiviteten mer genom att skriva mer utförliga svar och läsa vad andra hade skrivit vid *daily scrum* deadline kl 10. Genom *daily scrum* kunde vi följa principer såsom dagliga möten och att stötta medlemmarna i teamet.

Vidare användes även *sprint reviews*, för mer om hur det har påverkat arbetet se nästa avsnitt. Det gav möjlighet för projektet att följa flera agila principer som kontinuerlig leverans av värde till kund och en fysisk mötesplats för stakeholders till projektet.

Efter varje *sprint* genomfördes utöver den individuella och gruppreflektionen ett *sprint retrospective*-möte. Dessa möten genomfördes baserat på de riktlinjer som Kniberg (2015) återberättar och innefattade de tre punkterna:

- Vad var bra med förra sprinten?
- Vad kunde gjorts bättre?
- Sammanställning av tre förbättringsförslag baserat på det som diskuterats.

Genom dessa möten kunde vi bland annat, efter *sprint* två, inse att kommunikationen som skedde över Slack behövde förbättras. Efter att ha tagit upp detta kunde kommunikationen förbättras och fortsatt i arbetet upplevdes det inte vara något problem. Vidare, efter sprint tre lyftes under *sprint retrospective*-mötet värdet av grupprogrammering. Detta då vi vid dessa tillfällen ofta kunde lösa problem och frågetecken väldigt snabbt. Det gjorde att vi i efterföljande sprintar försökte boka in

tillfällen för att programmera allihopa i grupp.

#### 4.2.2 Vad vill vi göra i framtida projekt? (B)

I ett framtida projekt vill vi ta med oss det arbetssätt vi använt oss av under det här arbetet. Detta för att kunna komma igång direkt från start vid framtida arbeten utan att orsaka störningar på utvecklandet och att det vidare upplevdes som ett effektivt arbetssätt när man väl kommit in i det. Den del av det ovan nämnda som vi främst vill utveckla arbetet inom är att ha en bättre förståelse för kund för att därigenom mer väl kunna utvärdera att det som skapas fyller en verklig efterfrågan.

De förändringar vi däremot vill se jämfört detta projekt är dels längre sprinter. Sprintens längd kommer återigen behövas sättas i kontext av projektet men vi anser det eftersträvansvärt att framöver utöka längden till mer än en vecka. Även vidare att börja med *backlog refinement* redan vid projektstart för att effektivisera och fokusera arbetet under *sprint planning*. Men också att genomföra fysiska *daily scrum* möten istället för att hålla dem över ett digitalt medium. Vi vill även vara effektiva i vårt utförande av alla aktiviteter.

#### 4.2.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att i framtiden använda sig av de agila aktiviteter som använts i detta projekt ska värdet samt hur de påverkade arbetet tas med. Erfarenhet kring den problematik korta sprinter kan orsaka ska föras med för att i högsta möjliga mån argumentera för att dessas längd bör vara mer än en vecka. På liknande vis lyfta för alla relevanta stakeholders den problematik som kan uppkomma av att inte ha ett *backlog refinement*-möte för att minska arbetsbördan under *sprint planning*. Vidare även tydliggöra att hålla *daily scrum* över ett digitalt medium kan fränta stora delar av dess syfte och det därför i allra största mån bör eftersträvas att genomföra dem under fysiska sammankomster. Om det vid alla arbetsveckans alla dagar ändå fortfarande inte är möjligt bör det experimenteras med att istället då hålla *daily scrum*-möten vid tre eller fyra dagar per vecka. För att uppnå målet om effektivitet ska vi även time-boxa alla aktiviteter för att minska utdragna möten.

För att se till att man har en god förståelse för vad som är av värde för kund ska vi inledande i projektet lägga resurser på att sammanställa detta. Antingen genom att införliva en produktägare som har en godare förståelse för marknaden eller genom marknadsundersökningar.

Sammantaget handlar vägen om att nå vårt mål om att bära med sig de insikter som kommit av

detta arbete för att påvisa värdet av de förändringar vi anser eftersträvansvärda i ett kommande projekt. Även på liknande vis för att kunna lyfta värdet av aktiviteter vi vill behålla.

### 4.3 The sprint review and how it relates to your scope and customer value

#### 4.3.1 Sammanfattning av projektet (A)

För att något ska uppfylla alla de kriterier som sattes upp i *Definition of Done* krävs det, bland annat, att bidraget godkänns av produktägaren. Detta genomfördes under *sprint reviewn*.

För att avsluta varje *sprint* under projektet genomfördes en *sprint review*. Denna process var mycket ostrukturerad till en början. I *Scrum and XP from the Trenches* beskrivs det att det är eftersträvansvärt att ha ett högt tempo och inte lägga mycket tid på förberedelser. Dessa båda aspekter mötte vi men på bekostnad av struktur och granskning av det värde som skapats. Detta var däremot något som successivt förbättrades för varje sprint och vid den tredje sprintens avslut hade vi etablerat en struktur som fungerade för teamet och alla stakeholders. Då började vi med att beskriva vilket värde som inkrementet ämnade att skapa, därefter fick produktägaren testa funktionaliteten. Som tidigare nämnt har produktägaren för att kunna avgöra vad som är av värde för kund, utgått från en personifiering av vad som är en typisk kund, och det genom magkänsla. Om det som utvärderades inte godkändes av produktägaren så uppfyllde det inte DoD. Då räknades inte *story points* från den *user storyn* till den sprintens *velocity*.

I *Scrum and XP from the trenches* skrivs det att man kan genomföra *sprint review* i två omgångar, en publik mot externa stakeholders och en intern för de mer tekniska aspekterna. Då de som äger produktägare även var del av teamet, genomfördes intern och extern *sprint review* samtidigt. Det gjorde att alla relevanta aspekter hos en feature kunde föras upp samtidigt men det gjorde även i vissa fall att målet om att hålla *sprint reviewn* på en affärsorienterad nivå frångicks.

*Sprint reviewn* erbjöd således möjligheten att utvärdera det scope och värde som utvecklats under sprinten och chansen att sätta det inkrementet i dess kontext hos hela projektets scope. Som tidigare nämnt gjorde detta att vi efter *sprint reviewn* gjorde förändringar gällande prioritet i *product backlogen* samt att nya *user stories* togs fram då man genom att visa upp och diskutera applikationen kunde identifiera nödvändiga funktioner. Ett konkret exempel där vi prioriterade om *sprint reviewn* var efter andra sprinten då vi märkte att lidande intern kvalitet började spridas. Därmed prioriterades det upp att åtgärda vissa strukturella aspekter under nästa sprint.

#### 4.3.2 Vad vill vi göra i framtida projekt? (B)

Till framtida projekt vill vi från start ha en bättre fastställd process för *sprint review*. Detta för att redan från början kritiskt granska värdet så att efterlöpande utvecklingsarbete inte behöver anpassa sig till problem som uppstått under de inledande sprintarna. Vidare upplevde vi att i *definition of done* ha ett kriterie om att produktägaren måste godkänna det som lyckat varvid vi vill behålla det i framtida projekt.

I tidigare delar av rapporten har det även lyfts fram att det i framtida projekt är eftersträvaransvärt att produktägaren inte bör vara del av utvecklingsteamet. Under sådana omständigheter anser vi förändringar till den process vi använde gällande *sprint review* vara eftersträvaransvärda. Detta genom att inkrementet som skapats under sprinten bör granskas innan mötet med externa stakeholders. Därmed att det i det fallet bör införas en intern och en extern *sprint review*. På så vis kan man därmed vidare uppnå målet om att hålla *sprint review* med externa stakeholders på en mer affärsorienterad nivå. Till sist vill vi även efter största möjliga förmåga se till att vid *sprint review* kunna utvärdera inkrementet baserat på vad som är av värde.

#### 4.3.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

För att nå ovan nämnda ideal bör man inledande i projektet diskutera och fastställa processen för hur sprinten kommer avslutas och detta tillsammans med relevanta stakeholders. I samband med detta då även fastställa att dela upp *sprint review* i en extern och en intern.

För att se till att hålla *sprint review* med externa stakeholders på en mer affärsorienterad nivå och säkerställa att det finns ett värde för produktägaren av inkrementet ska den interna *sprint review* hållas först. Detta kan jämföras med Kniberg (2015) som utan att gå in på detaljer skriver att en intern *sprint review* kan hållas efter den externa. Vi ser dock möjligheter i att använda den interna review för att samla alla bidrag till inkrementet, diskutera de tekniska aspekten och föra med endast det essentiella till det externa *sprint review*-mötet.

För att kunna utvärdera att inkrementet är av värde ska vi, likt tidigare argumenterat, se till att produktägaren har en väldigt god förståelse för vad som är av värde på marknaden. Alternativt att man inledande in projektet åsidosätter resurser för att genomföra en marknadsundersökning.



## 4.4 Best practices for learning and using new tools and technologies

### 4.4.1 Sammanfattning av projektet (A)

I projektet har vi använt ett antal olika verktyg och teknologier, flera av vilka var nya för flertalet gruppmedlemmar, de verktyg och teknologier som har använts är följande:

**Språk:** *React Native* med *Typescript*.

**IDE:** *Visual Studio Code*.

**Versionshantering:** *Git* / *GitHub*.

**Scrum board:** *Zenhub*-tillägget på *GitHub*.

**Kommunikation & Daily Scrum:** *Slack*.

**Arbetsätt:** *Scrum*.

**Övriga verktyg:** *Business Model Canvas*.

Under projektets gång har flera metoder använts för att lära oss de ovan nämnda verktygen och teknologierna. Under de två arbetsveckor som föranledde utvecklingsarbetet användes huvudsakligen tre metoder för lärandeprocessen. Först, föreläsningar samt LEGO-övningen som en introduktion till bland annat arbetsmetodiken som skulle användas. Denna lärandemetod bar med sig fördelen av att det essentiella i ämnet lyftes fram. Den andra metoden var ämnesintroduktioner av de olika gruppmedlemmarna. Till exempel hade medlemmarna från industriell ekonomi en genomgång om *Business Model Canvas* och IT-studenterna höll introduktioner till *GitHub* och *React Native* med *Typescript*. Denna lärandemetoden tillät för ett öppet format som effektiviserades genom att fokusera på kunskapsluckor då de som höll i genomgången stämde av mot och fyllde på i de områden det fanns osäkerheter.

Från och med andra arbetsveckan började även diverse online-resurser att användas, dessa var främst riktade mot att ytterligare introducera och fördjupa kunskaper om programmeringsspråket samt ge ytterligare insikt kring implementationsdetaljer. Denna lärandeprocess syftade även till att skapa effektivitet då den riktade sig till att fylla individuella kunskapsgap. Under andra arbetsveckan påbörjades även ett arbete om att komplettera föreläsningarna gällande arbetsmetodiken genom att gå igenom kurslitteraturen, *Scrum and XP from the trenches*. Detta i syfte att ytterligare introducera *Scrum* inför utvecklingsstarten. Att påpeka här är att det tidigare nämnts att vi stött på problem under de första *sprintarna* i hur man bör jobba. Efter att ha reflekterat över detta insåg vi att det inte i första hand rörde sig om bristande teoretisk kunskap utan snarare om att vi föll kort om att till fullo tillämpa den samt en svårighet i att översätta det till praktik. I takt med att vi började använda arbetsprocesserna ökade förståelsen för dessa och den teori som vi därefter började mer genomgående repetera efter *sprint* två kunde lättare appliceras för att

förbättra arbetet vilket påvisade vikten av praktik i kombination med teori för att lära sig.

Under första dagen till den första sprinten hölls workshops, då för Visual Studio Code och Zenhub. Gällande Visual Studio Code fanns det varitioner kring vilken erfarenhet alla gruppmedlemmar hade. Workshopen syftade då till att säkerställa att alla hade en tillräckligt god förståelse för att kunna arbeta individuellt. Gällande Zenhub var det endast två gruppmedlemmar som hade tidigare erfarenhet och som därmed introducerade hur vi skulle använda verktyget under projektet.

I och med första sprintens start påbörjades utvecklingsarbetet. Detta stärkte kraftigt inläringen av programmeringsspråket och versionshanteringsverktyget genom praktisk erfarenhet och att allt som man tidigare lärt sig kunde sättas i en verklig kontext. Denna process av praktisk inläring varierades även genom parprogrammering vilket kombinerade att lära av varann med praktisk tillämpning. Vidare roterades grupperna för parprogrammering mellan sprintveckorna för att ytterligare stärka värdet av att lära av varandra. Därmed spreds kunskap i gruppen lättare.

#### **4.4.2 Vad vill vi göra i framtida projekt? (B)**

I framtida arbeten vill vi utvärdera valet av verktyg efter vilka som är mest lämpliga och, likt i detta projekt, använda en kombination av metoder för att lära oss dem. Det vill säga en mix av till exempel självinläring, lära av varandra och praktisk tillämpning. Detta visade sig resultera i en brant inlärningskurva för ämnen som tidigare varit okända samtidigt som det var ett effektivt sätt att täcka kunskapsluckor inom mer bekanta områden.

En mer specifik målsättning är att fortsätta att utöka antalet tillfällen för par- och grupprogrammering då de upplevdes vara en mycket effektiv inlärningsmetod. Till de metoder vi använt oss av i detta projekt hade vi även velat testa att införa återkommande kunskapsavstämningar där nya begrepp och tekniker som man lärt sig och anses vara värdefulla för gruppen delas.

#### **4.4.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )**

För att välja verktyg kommer vi, likt i detta arbete, först identifiera behov varefter vi kommer söka upp olika lösningar som tillgodoser de behoven. Därefter kommer vi väga fördelarna mot nackdelarna med de olika alternativen. Detta kommer öppna upp för alla relevanta aspekter gällande verktygen och därmed vad vi kommer behöva lära oss.

För att uppnå målet vill vi vidare eftersträva att skapa en kultur i framtida grupper som tillåter och uppmuntrar till att lära av varandra samt en vilja att hjälpa till. Till detta även se till att ha en kultur där alla gruppmedlemmar är motiverade att ta eget ansvar över sitt lärande för att på så sätt leta upp information på egen hand.

Att etablera en kultur av hjälpsamt och vilja att dela med sig hade också bidragit till att uppfylla målet med att ha kunskapsavstämningar. Därmed hade det bara kvarstått att skapa utrymme för att med visst mellanrum anordna dessa.

För att se till att par- och grupprogrammering fortsätts använda ska det många fördelar som det ger bäras med för att motivera införandet av sådana tillfällen.

## **4.5 Relation to literature and guest lectures**

### **4.5.1 Sammanfattning av projektet (A)**

Gällande inläringen av *Scrum* har huvudsakligen tre resurser använts i kursen:

- Kursens föreläsningar och LEGO-övningen
- *Scrum Guide* (Schwaber & Sutherland)
- *Scrum and XP from the Trenches* (Kniberg)

I tidigare frågorna har det löpande presenterats hur de olika informationskällorna förhållit sig till våra upplevelser av Scrum. Gällande Kniberg (2015) kan det sammantaget sägas att flera av de rekommendationer och revideringar som han har i andra upplagan ligger i linje med våra upplevelser från detta projekt. Som exempel kan återigen nämnas sprintarnas längd, som för detta projekt var en vecka, upplevdes vara för kort i enlighet med den problematik som förts upp i litteraturen.

### **4.5.2 Vad vill vi göra i framtida projekt? (B)**

Målet för fortsatta projekt är att införliva teori i praktiken och reflektera över hur de jämför sig genom att verifiera och kontrastera olika uppfattningar. Detta för att i gränslandet mellan praktik och teori hitta den väg som bäst lämpar sig för ändamålet. För framtida projekt vill vi därmed ta lärdom av det som andra har kommit fram till samtidigt som vi fyller ut och jämför detta med egna erfarenheter.

#### 4.5.3 Vad krävs för att vi ska nå dit? ( $A \rightarrow B$ )

Vid framtida projekt ska vi ta med de erfarenheter vi fått från detta projekt. Även vidare fortsätta ta del av litteratur och andra informationskällor för att få mer information samt nya synvinklar som man annars kanske missat. För att uppnå det krävs det en motivation för att lägga ner de resurser som krävs. Hur man uppnår den motivationen är individuellt men likväl bör värdet av att göra det för att lättare uppnå gruppens leverabler framhävas.

## 5 Referenslista

Kniberg, H. (2015). *Scrum and XP from the trenches* (2. ed.). C4Media.

Schwaber, K., & Sutherland, J. (2017). *The Scrum Guide*. Hämtad från: <https://www.scrumguides.org/docs/scrumguide/Scrum-Guide-US.pdf#zoom=100>

(2001). *Manifesto for Agile Software Development*. Hämtad 2019-05-24 från <http://agilemanifesto.org/>