

# Team Par-King

*DAT256 Software Engineering Project, Autumn 2018*

---



**Albin Segestam, [segestam@student.chalmers.se](mailto:segestam@student.chalmers.se)**  
**Erik Tallbacka, [erikta@student.chalmers.se](mailto:erikta@student.chalmers.se)**  
**Markus Saarijärvi, [marsaa@student.chalmers.se](mailto:marsaa@student.chalmers.se)**  
**Theo Haugen, [htheo@student.chalmers.se](mailto:htheo@student.chalmers.se)**  
**Åke Svensson, [sake@student.chalmers.se](mailto:sake@student.chalmers.se)**

## Innehållsförteckning

<b>Team Par-King</b> .....	1
<b>DAT256 Software Engineering Project, Autumn 2018</b> .....	1
Introduktion .....	1
1. Customer Value and Scope.....	2
1.1 The chosen scope of the application under development including priority of features and for whom you are creating value .....	2
1.2 The success criteria for the team in terms of what you want to achieve with your application.....	3
1.3 Your user stories in terms of using a standard pattern, acceptance criteria, task breakdown and effort estimation .....	4
1.4 Your acceptance tests, such as how they were performed and with whom .....	4
1.5 The three KPIs you use for monitoring your progress and how you use them .....	5
2. Social Contract and Effort .....	6
2.1 Your social contract, i.e., the rules that define how you work together as a team.....	6
2.2 The time you have spent on the course (so keep track of your hours so you can describe the current situation) .....	7
3. Design decisions and product structure.....	8
3.1 How your design decisions (e.g., choice of APIs, architecture patterns, behaviour) support customer value .....	8
3.2 What you document and why, by using e.g. use cases, interaction diagrams, class diagrams, domain models or component diagrams, text documents etc. ....	9
3.3 How you use and update your documentation throughout the sprints .....	10
3.4 How you ensure code quality, enforce coding standards etc.....	10
4. Application of Scrum.....	11
4.1 The roles you have used within the team.....	11
4.2 The agile practices you have used for the current sprint .....	12
4.3 The sprint review (either in terms of outcome of the current week's exercise or meeting the product owner).....	13
4.4 Best practices for using new tools and technologies (IDEs, version control, scrum boards etc.).....	14
4.5 Relation to literature and guest lectures (how do your reflections relate to what others have to say?) .....	15

# Introduktion

Kursen DAT 256 *software engineering project* som vi gruppmedlemmar har deltagit i har varit uppdelad i två faser. Första delen av kursen var tillägnad för att få en introduktion till agila processer, "workshops" samt övriga ämnen som berörde kursen. Under den första fasen så skulle det skapas så kallade scrum development teams. Efter de första veckorna som den första fasen bestod av så gick det vidare till fas två som var att arbeta på projektet där en applikation skulle utvecklas. Inriktningen i årets kurs var att skapa en android applikation inriktad mot att underlätta vardagen i bilen.

Under den andra fasen av kursen så påbörjades arbetet på att utveckla applikationen som projektgruppen hade enats om. Genom bl.a. Gästföreläsare men även tillgång till att kunna medverka på gemensamma möten där man ytterligare kunde reflektera på utvecklingen av applikationen men framför allt tillvägagångssättet med att använda sig av agila processer, Scrum. Hela fas två i kursen skulle gå ut på att gruppmedlemmarna skulle reflektera över hur det är att arbeta på ett professionellt tillvägagångssätt som en ingenjör skall kunna göra i arbetslivet samt att kunna applicera det i den större bilden då projekt kan innebära att man samarbetar med ytterligare personer/grupper.

Vårt scrum development team kom att kallas "Team Par-King" och visionen med applikationen skulle bli att utveckla något som skulle underlätta och lokalisera parkeringar i sin närmiljö. Under de kommande sprintarna som kom att finnas i fas två av kursen så har det utvecklas både på applikationen men även på huruvida arbetsprocessen har gått till under projektets gång.

# 1. Customer Value and Scope

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

A:

Vi hade redan från början en klar vision på hur vi ville att vår applikation skulle fungera när den var klar. Vi gick med detta till produktägaren för att, i samråd med dessa, jobba ut hur en applikation som fungerar bra för båda parterna skulle se ut. Produktägaren föreslog att vi borde ha funktioner som underlättar för användaren när de ska göra ärenden, som exempelvis att parkeringar visas bara beroende på hur länge användaren ska vara parkerad.

Vi diskuterade även andra saker som skulle vara bra att prioritera som vi ansåg inte var bra representerat i andra parkeringsappar, som exempelvis göra det lättare för de med ett handikapp att hitta en bra parkering.

Andra funktioner som vi valde att ha med i vårt "scope" är kunna se vilka parkeringar som är gratis och vilka som kostar pengar och då också hur mycket det kostar att stå där. Man kan se avståndet från din nuvarande position, vilka som äger parkeringen, hur länge man får stå parkerad och vilka tider man kan stå där. Vi har även en favorit markerings funktion som tillåter dig att markera en parkering som du ofta använder för att läggas till i en lista i menyn som du senare har tillgång till.

Det finns andra mindre funktioner som vi implementerade för att höja standarden på appen. Som exempelvis att kunna ändra temat på kartan till; retro, night mode eller standard. Det går även att ändra vyn på kartan mellan; satellit, terräng, hybrid eller standard. När du har klickat på en parkering går det att använda en knapp i appen som tar dig till Google maps app som visar dig vilken rutt som är snabbast till den valda parkeringen.

B:

Ytterligare implementeringar av funktioner i applikationen har diskuterats. Det som hade velats göras ytterligare hade bl.a. Kunnat vara:

- Sökfunktion - Användaren skall kunna söka på gatuadresser eller platser för att kunna söka av området enklare för att lokalisera områdets parkeringar.
- Vägbeskrivning till markerad parkering internt i applikationen.
- Lägga till filtrering av ytterligare sorters av parkeringar, t.ex. el bilar, motorcyklar.
- Gruppering av markeringar på kartan med s.k "cluster" så att användaren inte ser alla markeringar utan bara dom som är nära.

På presentationen så fick vi feedback från de Volvoanställda på funktioner som vi hade kunnat implementera i applikationen som vi inte hade tänkt på men som var väldigt användbara. Exempelvis att ikonerna på kartan indikerar om det är en gratis-, handikapp- eller betalparkering för att slippa klicka på ikonerna. Ytterligare en funktion som hade kunnat

vara användbar är att kunna sätta en timer på appen så du får en notifikation strax innan din parkeringsbiljett löper ur.

Sedan diskuterade vi diverse tekniska saker såsom hur ofta vi uppdaterar informationen och hur vi hämtar informationen. Utifrån det så skulle vi vilja uppdatera informationen oftare för att användaren skall ha så mycket relevant information som möjligt. Dessutom borde vi dela upp hämtningen av information så att vi inte hämtar allt data från Göteborgs Stad på samma gång utan hämtar bara den data som användaren ser just nu.

A → B:

Den största skillnaden om vi skulle gjort om projektet är att vi har mycket större kunskap om hur android fungerar och hur man implementerar diverse funktioner. Vi i gruppen har också fått en mycket bättre förståelse för hur lång tid de diverse funktionerna tar att förverkliga, vilket skulle vara till stor hjälp under planeringsstadiet.

## 1.2 The success criteria for the team in terms of what you want to achieve with your application

A:

Vi lyckades att uppnå i princip det mesta som vi satte ut för att göra i början av projektet. Vi gjorde en "Epic" i vår första sprint som vi delade upp i mindre "user stories". Kriterier sattes på varje uppgift för att de skulle återspegla vad vi förväntade oss med samråd med projektägaren.

Med projektets gång kom vi på fler funktioner som skulle kunna implementeras vilket skapades samt återspeglades i user stories som sedan blev tillagda i backloggen. Det finns fortfarande funktioner som vi skulle vilja utveckla men i helhet så slutfördes allting som fanns med i början av projektet och vad som vi hade lagt in i vår epic.

Det fanns även user stories som kom att diskuteras med vår produktägare som vi inte hade med i vår epic från början. Efter ytterligare diskussioner med produktägaren så kunde ytterligare en user story bli konstruerad och tillagd i backloggen.

B:

Om vidare utveckling skulle ske av applikationen för att sedan kunna lanseras på google play store så skulle det vara ett steg i rätt riktning tycker samtliga gruppmedlemmar. Det som skulle kunna gå att utveckla ytterligare funktioner samt att förenkla upplevelsen av applikationen för användaren innan lansering. Under demon av applikationen hos volvo så blev det mycket tydligt att snabb återkoppling av användarupplevelsen gjorde det tydligt för vad som bör göras bättre.

A → B:

Detta kan uppnås genom att från början planera i detalj hur applikationens funktioner skall fungera och se ut utifrån vad vi skulle förväntas av en applikation på Google Play Store. Vi borde också satt striktare och mer utförliga kriterier på varje user story samt tasken som

tillhörde dem för att lättare nå den vision som vi förväntade oss av user storyn. Ytterligare en sak som skulle kunna göras bättre för att uppnå ett bättre resultat är att låta utvalda utomstående individer testa och reflektera över applikationen och dess funktionalitet.

### 1.3 Your user stories in terms of using a standard pattern, acceptance criteria, task breakdown and effort estimation

A:

Gruppen kollade övergripande på måndagar över tillgängliga user stories som skulle möjligtvis utvecklas under kommande sprint. Gruppen kom överens om vilka user stories som skulle bearbetas genom föregående veckors kunskap, diskussioner mellan alla gruppmedlemmar samt med produktägaren om prioritet och hur mycket arbete det hade estimerats att en user story skulle ta.

Tilldelningen av tasks gjordes samtidigt vilket påverkade hur mycket en task behövdes brytas för att göra den hanterbar under en veckas tid. Ytterligare en uppgift som var viktig var att se till att sträva efter att göra tasks oberoende från andra tasks, vilket kan vara svårt ibland med en grupp med flertal medlemmar. För det mesta fungerade det bra men vid ett tillfälle bildades en flaskhals där i princip hela gruppen fick hjälpas åt med samma uppgift. Breakdown av en task som vi hanterade var att om det inte fanns någon kompetens sedan innan på det som skulle utföras så blev det lättare att hantera och estimeras om man bröt ner tasken i smådelar.

B:

Det finns alltid utrymme för att kunna vara ännu mer tydliga och specifika i våra acceptance criteria. Vi stötte på en stor flaskhals i form av hämtning av data från en databas, något som vi skulle vilja undvika fortsättningsvis då det satte större delen av projektet stilla.

A → B:

Det här projektet har gett mer kunskap av vad som är möjligt att göra vilket gör det lättare att bedöma om en user story är lätt att möjliggöra. T.ex. vet vi att det tar längre tid än vi trodde att jobba med databaser då allt i dem inte alltid är lika perfekt som man vill. För att både göra det enklare för oss och göra det tydligare vad vi ville ha ut av varje user story så borde vi ha använt oss av mer task break down. Det kunde hjälpa oss att undvika vissa problem att bryta ner dem mer så att det blir lättare att lösa ett mindre problem i taget istället för ett stort på en gång.

### 1.4 Your acceptance tests, such as how they were performed and with whom

A:

I gruppen testade vi alltid vår kod innan vi skickade det till vår main branch. Vid använde oss utav en testbranch som fungerade som en mellanhand gentemot vår master branch och vår personliga kod. Efter att någon hade kodat klart en user story eller task som skulle läggas in till resten av koden så skickades den till vår test avdelning. Efter det

meddelade personen resten av gruppen, då minst två andra medlemmar testade koden genom att använda appen så att det inte hade skett någon kvalitetsförsämring samt att den menade funktionen fungerade enligt våra acceptance criterias. Sedan kontrollerar man även att koden som blev tillagd att den var kommenterad rätt så att det var lättförståeligt och inte onödigt komplicerat. Efter att två personer har meddelat att koden är upp till standard och att appen fungerade bra så slog vi ihop koden med vår master branch.

B:

En förbättring som kan göras här är att ha system som testar logiken i olika situationer för den kod som skrivs och den kod som vi redan har fastställt, detta skulle se till att ingen logik ändras eller förstörs med varje tillägg av ny kod. Vi hade också velat ha ett bättre och mer fastställt system för de test som vi gjorde på varandras kod med tydliga beskrivningar om hur det skall fungera och hur det skall se ut.

A → B:

Vi borde ha satt upp mer tydliga regler om vad som gäller när man ska testa någon annans kod för att tydliggöra vad det är som behöver kontrolleras för att säkerställa kvaliteten. Vi kan implementera ett test system med exempelvis JUnit där man skriver test för den logik man vill testa för att säkerställa att allt fungerar som tänkt. Test måste avklaras vid varje ny funktions implementering. På detta vis kan personer som testar fokusera på utseendet och beteendet av funktionerna och inte på logiken i bakgrunden.

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

A:

Vi använde oss av tre olika KPI:er, det första vi har mäter hur mycket poäng vi hade planerat att göra per vecka gentemot hur mycket vi levererade i slutet av sprinten (se bilaga 1). KPI nummer två i form av en graf som visar hur mycket varje individuell medlem kände att han hade lärt sig på en skala 1-10 per sprint (se bilaga 2). Vårt sista KPI är en graf över vår projekterade arbetskurva och vår verkliga arbetskurva (se bilaga 3).

Innan dessa slutgiltiga KPI:er så hade vi en graf som skulle representera hur mycket kundvärde vi hade levererat per vecka men efter en diskussion med Volvo så insåg vi att det inte var en bra idé då deras representant sa att han hade svårt att säga hur mycket kundvärde vi hade gjort, därför blev vårt tredje KPI (bilaga 3) inte så informativt då den ungefär ger samma information som den första (bilaga 1).

Om man analyserar graferna kan man tydligt se att under den andra och tredje sprinten så stötte vi på en stor flaskhals och inte levererade så mycket som vi planerade. Detta delvis på grund av att vi hade en dålig estimation på hur mycket poäng varje user story var den veckan, men även också på grund av vår brist på kunskap inom området. Dock så kan man också se att under den sprinten så lärde vi oss väldigt mycket, så mycket av tiden under de sprintarna gick till att lära sig hur vi skulle gå tillväga att göra uppgifterna som vi hade framför oss.

B:

En klar förbättring med våra KPI som kunde göras är att ha en tredje variabel att mäta som särskiljer sig från de nuvarande, det hade gett oss en extra överblick som vi hade kunnat använda i våra analyser och reflektioner över projektet. Med en extra mätning på våra reflektioner hade det kunnat gynna oss under projektets gång och se ifall det blir en förbättring alternativt märka om det är något som inte går som det ska.

A → B:

Vi borde ha gett mer tid till våra KPI:er och tagit mer tid till att komma fram med bra måtvärden. I början av projektet så var det svårt att veta vad man kunde mäta på som gav någon nytta då synen på vad KPI:erna skulle åstadkomma var vagt. Nu när man har jobbat med scrum och på detta sättet så har man andra perspektiv på hur man kan se på projektet och kan inse att det finns många variabler som kretsar kring ett grupprojeckt. I processen så kom det att vara viktigt att kunna se vart man hade haft problematik, reflektera över det och förbättra arbetsprocessen. Det som vi hade kunnat göra bättre var att försöka att i ett tidigare stadie inse vad som hade kunnat gå snett eller vad vi hade velat mäta för att få en effektiv process. Vi diskuterade när vi kände att vi hade problematik med personer med erfarenhet så som våra handledare i kursen samt med Volvo. Efter givande diskussioner så kom vi fram till vad vi skulle förbättra samt att vi fick en tydligare bild på hur KPI:er kunde användas i en större bild.

## 2. Social Contract and Effort

### 2.1 Your social contract, i.e., the rules that define how you work together as a team

A:

Vi satte upp regler under projektet huruvida vi skulle arbeta med varandra samt uppföra oss under samtliga sprintar. Under så står de övergripande regler som vi i projektgruppen kom överens om.

Regler:

- 80% närvaro vid fysiska möten
- Lämna återbud minst 24 timmar innan planerat möte
- Respektera arbetsplatsen
- Visa hänsyn för att gynna varandras arbetsro
- Samarbetsvillig och kompromissvillig
- Kommunicera väl med kollegorna
- Aldrig krav att jobba på helger

Det kan kännas lite abstrakt att bara ha en simpel lista för förhållningsregler inom gruppen men i vår syn så krävdes det inte så mycket mer. Samtliga gruppmedlemmar har arbetat tillsammans sedan innan så känslan för projektet var att bara ha övergripande förhållningsregler i allmänhet. Då vi redan hade jobbat innan med varandra och visste att det inte var några problem sedan innan så behövdes det inte ytterligare specifikationer eller förtydliganden för hur man skall uppföra sig genom arbetets gång. Hade vi inte känt varandra



sedan innan men framför allt inte arbetat tillsammans med bra resultat så hade det varit en helt annan sak när det gäller hur social kontraktet hade tagits fram. Att säkerställa effektivitet och ha en specifik specifikation om man inte hade jobbat med varandra innan hade varit väsentligt att ha något att luta sig mot när det skulle uppkomma komplikationer eller svårigheter.

B:

Att kunna samarbeta och lösa konflikter i en arbetsgrupp är väsentligt och eftertraktat. Andra egenskaper att kunna hålla tider, respektera arbetsplatsen och kunna kompromissa är även det viktigt och eftertraktat. Allt för att kunna uppnå den potentiella effektivitet som åtrås i ett projekt så är sammanhållningen i gruppen är extremt viktig. Inte att man måste hålla med om allt osv inom gruppen men att ha en process som fungerar väl med avseende på vad som man levererar.

A → B:

Vad som skulle kunna behövas att beskriva för att uppnå en bra förhållning i gruppen så att effort och dylikt faller på plats är den stora frågan. Vi som har arbetat tillsammans sedan innan hade redan en känsla för vad som fungerade sedan innan i tidigare projekt. Det som vi gruppmedlemmar känner är ytterst viktigt är just kommunikationen. För att kunna uppnå en hög effektivitet och jämnfördelat arbete krävs det kommunikation. I alla våra regler som valdes i vårt social contract så är det kommunikationen som kommer leda oss framåt i projektet. Om det hade varit en grupp som sedan innan inte hade arbetat tillsammans så hade det förmodligen behövts tydligare specifikationer och rutiner för att hantera diverse olika situationer för att kunna överkomma problem vilket skulle öka effektiviteten i projektet.

## 2.2 The time you have spent on the course (so keep track of your hours so you can describe the current situation)

A:

Vi har inte dokumenterat tiden på något sätt under kursens gång. Förutom varje måndag då vi var vi på handledningen som var på ca. 2 timmar och våra möten varje fredag som snittade cirka 1 ½ timme. Jobbet under veckan så satt alla på sin egna tid och vi räknade med att vi skulle lägga 20 timmar per vecka på projektet men det varierar självklart då vi läste en annan kurs samtidigt och vissa veckor så fick prioriteten ligga på den andra kursen och tvärtom.

B:

Optimalt i våra ögon är inte att räkna med hur mycket tid vi lägger per vecka utan snarare att vi hinner med det vi har planerat att göra. Kursen ska vara på 20 timmar per vecka men vi finner ingen anledning till att försöka lägga en timme extra på något som inte är nödvändigt om vi hade blivit klara med våra mål för sprinten efter exempelvis 19 timmar.

A → B:

Om vi hade funnit tidsperspektivet som en viktig del i projektet så kunde det vara enklare att hålla koll på den om vi hade haft fler schemalagda tider då vi skulle mötas eller jobba

tillsammans, då hade man enklare kunnat räkna ihop alla möten genom att kolla på schemat och få en ganska exakt tid på hur mycket jobb som hade gjorts under veckan.

Ett bättre sätt än att mäta tidsperspektiv hade varit att mäta velocity och kunna mäta hur mycket man kan ta på sig varje sprint, för att fortfarande få en bra mätning på hur mycket man har jobbat utan att mäta det i tid. I ett projekt med fler sprintar kan man då se varje sprint om man tar på sig för lite jobb och inte håller den velocityn som man hade kunnat göra.

## 3. Design decisions and product structure

### 3.1 How your design decisions (e.g., choice of APIs, architecture patterns, behaviour) support customer value

A:

Första design valet gjordes mycket tidigt av projektet då vi skulle välja vilket kart API som vi skulle använda för vår applikation. Det naturliga valet var Google Maps API då man i Android Studio enkelt kan skapa ett applikations skelett med Google Maps implementationen.

Förutom att det var enkelt att komma igång med så är det ett väldigt bra dokumenterat API med mycket resurser online vilket gjorde att vi snabbt kunde komma igång och därmed leverera kundvärde. Dessutom så är Google Maps den mest använda och igenkännande kart applikationen i världen vilket ger vår applikation ett bekant och bekvämt utseende för våra användare.

Vi hade dock problem att köra vår applikation på Android CAR emulatorn men trodde att det skulle lösa sig då vi inväntade svar från Semcon angående problemet, så vi fortsatte med att utveckla med Google Maps och testade i en vanlig telefon-emulator. Det var inte förrän andra handledningstillfället då vi fick reda på att Google Maps inte skulle fungera i Android CAR emulatorn men då hade vi redan en hyggligt bra bas till vår applikation så vi valde att inte refaktorera om allt vi hade gjort för att byta till ett annat API så vi stannade med Google maps.

Nästa steg var att välja rätt API för att hämta information om parkeringsplatser. Vi började med att försöka använda Google Places för detta men hade problem med att få det att fungera korrekt och det var även brist på information om parkeringar på Google Places, oftast kunde man bara hämta adressen. Vi fick sedan ett tips om Göteborgs Stads egna öppna data API där man kan hämta detaljerad information om nästan alla parkeringar i Göteborg. När vi fick detta att fungera så gav detta mycket värde för användarna då de nu kan se position, adress, antal platser, maximal parkeringstid, pris och diverse övrig information för varje parkering.

En del som vi såg var viktigt för vår applikation var att göra det lättare för handikappade att hitta parkeringsplatser. Vi fokuserade på detta istället för att till exempel implementera parkeringsplatser för exempelvis motorcyklar men vi implementerade det på ett sådant sätt så att det är enklare att nu i efterhand gå in och lägga till mer filtrering som kan göra det enklare att hitta specifika parkeringsplatser.

B:

Vi hade inte någon direkt plan på utseende på appen från början eller någon struktur på layouten utan den utvecklades under projektets gång. Det finns även en del förbättringar i

designen som hade kunnat göra appen tydligare och lättare att använda. Detta är t.ex. att ändra ikonerna för parkeringar av olika sorter såsom gratis, betal och handikappsparkeringar. Det hade även kunnat finnas utrymme att dela upp och göra våra menyer tydligare att navigera.

A → B:

För att kunna nå våra mål med att ha högt kundvärde på våra design decisions vore att ha en tydlig designplan från början och låta någon utomstående ge en recension på designen för att kunna ge tips och förslag på hur det hade kunnat vara bättre. Vi fick många bra tips från de Volvoanställda om hur vi hade kunnat designa applikationen för att gynna användaren mer som vi helt enkelt inte tänkte på.

Många företag tar nog just design decisions från recensioner om deras applikationer så att skicka ut en betaversion innan den är klar men då hade kanske problemet varit att få in dessa ändringar utöver våra user stories och tasks som vi redan hade i vår planering. Längre tid på ett projekt ger mer möjligheter att kunna förbättra designen för att det ska ge så mycket kundvärde som möjligt.

### 3.2 What you document and why, by using e.g. use cases, interaction diagrams, class diagrams, domain models or component diagrams, text documents etc.

A:

Varje fredag hade vi ett online möte då vi diskuterade varje gång hur veckans sprint hade varit och hur det hade gått i form utav en reflektion. Vi följde de rubriker som fanns på kurs-hemsidans Git och skrev tillsammans hur veckans arbete och dylikt hade gått. Dokumentation om hur långt vi kommit i projektet gjordes främst genom trello som uppdaterade varje gång en task eller user story uppfylldes.

Eftersom vi aldrig har arbetat med Android och Android Studio så visste vi inte hur vi skulle gå tillväga i början, vilket gjorde det svårt för oss att planera allt i detalj i form av klassdiagram eller liknande. Vi kände att vi inte bör lägga vårt fokus på detta i början av projektet då vi inte hade ett grepp om hur man arbetar med Android. Då vi inte hade så mycket tid åt planering och förarbete så fokuserade vi på att generera något som ger värde till kund och PO varje vecka så planering och dokumentation hamnade på sidolinjen. Vilket ledde till att vi istället diskuterade applikationsstruktur, klass relationer och metoder muntligt och sedan implementerades det vi diskuterade.

B:

Vi skulle vilja ha ännu mer förarbete för att göra projektet mer överskådligt från start, detta i form av klassdiagram och diverse dokument som listar de funktioner som kan vara intressanta att implementera. Även om dessa hade ändrats under projektets gång hade de verkat som en slags mall för att få hela gruppen på samma bana. Även ha en mer överskådlig blick om vad som är möjligt att göra i Android Studio, så att man kan göra mer design beslut i ett tidigt stadie.

A → B:

Genom att framförallt lägga mer tid under den första veckan att skapa ett första utkast av diagram och dokument vi skulle vilja använda för dokumentering. Sedan uppdatera dessa som man ser nödvändigt under projektets gång. När det kommer till att kunna ha en bättre förståelse för vad som går att skapa i Android Studio så är det svårt att veta på rak arm. Det som krävs för att kunna estimerar vad som går att göra samt hur i Android Studio kommer att handla om erfarenhet, learn by doing. Det som hade kunnat göras bättre för att få en större inblick i olika skeden om vad man hade kunnat skapa är om man hade utrett olika metoder samt teknologier som redan finns till hands. Sedan hade även dokumentationen kunnat bli bättre genom att ha eftersökt olika metoder/teknologier så hade man enklare kunnat påvisa varför man hade valt att göra på ett visst sätt osv.

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

A:

Det mesta av dokumentationen var i form av hur föregående sprintar hade gått vilket uppdaterades på fredagar. Den användes sedan på måndagar där vi fördelade user stories, tasks och generellt planerade kommande sprint. Vi använde Trello för att dokumentera och visualisera vilka tasks som var aktuella denna veckan samt vilket stadie de var i.

B:

Vi känner att förbättringar ligger mer på att dokumentera mer istället för hur vi använde och uppdaterade dokumentationen. I fallet där vi hade haft mer dokumentation hade vi velat lägga mer tid under mötena på både måndagar och fredagar för att uppdatera och analysera.

A → B:

Först och främst vara bättre på att dokumentera och göra t.ex. klassdiagram och interaktionsdiagram och färdigställa ett system tidigt i processen. Gör ett exempelvis ett klassdiagram första veckan innan någon kodning har startat och använd ett par timmar i veckan efter varje sprint till att diskutera inom gruppen hur projektet efterliknar diagrammet. Har man gjort några ändringar i t.ex. strukturen av programmet kan diagrammet behövas att uppdateras.

### 3.4 How you ensure code quality, enforce coding standards etc.

A:

Gällande arbetet i gruppen och själva kvaliteten på koden så gjorde vi avvägningar i projektet. Vi valde i gruppen att införa vilket kan tyckas vara självklart men att kod som blivit utvecklad skall även vara kommenterad. Andra förhållningar som valdes att hållas var att försöka i största möjliga mån att använda hjälpmetoder för att simplificera och för att kunna lättare följa med i processerna vid andra tillfällen koden överblickas. Gällande övergripiga implementationer i programmet fördes diskussioner samt parprogrammering i gruppen för att säkerställa kvaliteten på koden men framför allt logiken som skulle kunna gynna projektet på lång sikt. Det fanns ytterligare ett sätt som vi säkerställer kvaliteten med och det var genom att vi testade och recenserade varandras kod. Genom att skicka sin kod till test branchen så hade vi som krav att två andra gruppmedlemmar skulle analysera koden så att

den uppfyllde rätt funktion, var optimerad och korrekt gjord. Efter detta så förde vi diskussioner om man behövde vidta åtgärder med koden eller om den var tillräckligt klar till att skickas upp till master branchen.

B:

Rutiner och förhållningar till hur arbetet går till under ett projekt för att säkerställa kvaliteten som krävs för att kunna leverera en produkt kan vara väsentlig för en kund. Framför allt så är det viktigt att införa standarder för hur processen skall gå till i utvecklingen om projektet skall fortgå under lång tid samt vidareutvecklas. Allt för att enklare kunna identifiera processer och logik i programvaran så är det nödvändigt att ha dokumentation som förtydligar den framtagna koden som skapats för att lättare kunna hantera det i längden. Kodkvalitet kan gynna på olika väsentliga sätt. Om det skall börjas med att om nya lagmedlemmar introduceras till projektet skall det vara enkelt/enklare att kunna bli insatt i logiken som programmet är uppbyggt på om kvaliteten på koden är hög. Vid tillfällen där optimering är en viktig faktor blir det tydligare att inse vart programmet har sina begränsningar för att simplare kunna lösa optimerings frågor.

A → B:

Säkerställa struktur och en god kodkvalité går att åstadkomma på olika sätt men som i helhet leder till en bättre struktur långsiktigt. Man kan som sagt standardisera processen kring hur man arbetar för att säkerställa kodens kvalité osv. Standardisering kan ske på olika plan men exempel på det kan ju vara hur man testar koden eller hur man tar fram den. Det kan även ingå i processen i skapandet av projektet där kvalitetskontroller på olika plan säkerställer en effektiv men framför allt en kvalitativ utveckling. Parprogrammering kan vara ett alternativ så att idéer diskuteras innan de implementeras i applikationen. Det kan även vara lättare att identifiera problematik som kan uppkomma under utvecklingen som hade missats om arbetet hanterats av en enskild individ. Just erfarenheten samt kompetensen i laget är inte alltid jämnt fördelat på alla områden så med diskussioner och liknande arbetsstrategier så kan arbetet bli mer solitt överlag. Det kan införas olika standarder för allt som indentering, hur koden skall vara strukturerad och kommenterad som även det kan höja kvaliteten på koden.

## 4. Application of Scrum

### 4.1 The roles you have used within the team

A:

Vi valde att rotera scrum master varje vecka så att alla skulle testa på rollen som det. Scrum masters roll var följande punkter:

- Ta tag i veckomöten
- Diskutera med Volvo vid behov
- Hur går projektet

Varje person i gruppen fick göra olika mycket när han var scrum master då vi ibland inte behövde prata med Volvo lika mycket som andra veckor, men vi tyckte det var bra att kunna rotera så alla fick känna på det inför framtiden. Det som gick bra med scrum master rollen

var att hålla kommunikationen igång och bestämma möten. Kraven var inte så stora i rollen som scrum master då vi ansåg att det är delvis allas ansvar att se till så att kommunikationen i gruppen flyter på.

Vi hade också en Git master vars roll var följande punkter.

- Sköta merging mellan test branchen och huvud branchen.
- Ha koll på vad som senast blev inlagt på test branchen och ej låta andra i gruppen lägga in nytt innan det som finns där är testat och klart.
- I allmänhet hjälpa och lösa problem som uppstår med Github repositoryn.

B:

Vi tycker att scrum masters roll i detta projekt var ganska komplett, i större skala så hade scrum masters roll varit viktigare i att organisera möten och ha koll på hur projektet går men på en grupp av fem personer så är rollen betydligt enklare då det inte är särskilt jobbigt att hålla koll på projektets gång och kunna organisera upp mötena.

A → B:

En förbättring som hade kunnat ske i rollen som scrum master hade varit att ha en bättre och mer kontinuerlig kontakt med Volvo om olika beslut om appen samt höra med deras representant om diverse idéer, designer eller funktioner hade varit gynnande.

## 4.2 The agile practices you have used for the current sprint

A:

Arbetsmetodiken som kom att bli i sista sprinten blev tillslut indelad i olika stadier som då växte fram under projektet i helhet. Processen som blev tydligare allt eftersom man kunde diskutera sig fram, läsa om men samt kändes naturlig kan ungefär delas upp i sex olika stadier som vi känner det.

Första stadiet i sprinten var alltid att reflektera över vart vi var i projektet. Reflektionerna gjordes internt i teamet samt med produktägaren för att få en klar bild på vart man befann sig i arbetet/projektet. Det kunde ju vara att olika delar av teamet hade jobbat på olika lösningar eller saker som inte hade uppkommit än så att ha en gemensam bas för vart man står är en viktig punkt för varje sprint. Det kan även vara så att man hade saker kvar sedan förra sprinten och att man måste inse vart man är och vad man hade kvar sedan förra sprinten som ett exempel.

Andra stadiet kan man säga är att när man visste klart och tydligt både för teamet samt för produktägaren vart man befann sig så kunde man här börja diskutera vad man skall fortgå med att göra. Diskussioner angående prioriteringar i backloggen och varför man ska välja specifika user stories var den stora delen av denna fasen i sprinten.

Tredje stadiet för sprinten var att när man visste vad man skulle arbeta på var att undersöka vilka olika alternativ på metoder samt teknologier som kunde användas. Det kunde vara under detta stadie som design decisions osv diskuterades samt att om produktägaren hade preferenser på olika infallsvinklar. Under detta stadie samtalades det inte avsevärt mycket med produktägaren då Volvo hade inte så mycket kriterier angående hur det skulle funka utan kunde ha mer att säga om vad man ville skulle funka.

Fjärde stadiet var när allt skulle implementeras så det var här den större delen av programmering skedde under sprinten. När vi hade valt specifika metoder och teknologier att arbeta kring så var det tydligare och effektivare i sista sprinten med vad som skulle åstadkommas. Efter lösningar har utvecklats så skulle de testas osv.

Femte och sista stadiet för sprinten blev att reflektera över sprinten i sig själv. Det dokumenterades vad man hade levererat samt diverse olika diskussioner som föreföll varje sprint angående effort, vad man hade kunnat göra bättre osv. Man skapar sig en bra grund för att kunna få en bra utveckling i laget samt se efter vad som kan göras bättre.

B:

Det som hade kunnat vara bättre i sprinten hade varit om feedback om hur applikationen kändes att hanteras och dylikt framkom bättre. Det hade varit bra att ha en input från individer angående hur applikationen är att använda samt vad som skulle kunna förbättras.

A → B:

Det som man hade kunnat göra och borde gjort kontinuerligt var att i det fjärde stadiet i sprinten så skulle man haft utomstående individer som kunde testa applikationen och funktionaliteten som har utvecklats för att kunna ta det i åtanke under utvecklandet. Det hade kunnat göras enkelt genom att låta produktägaren prova men under vårt projekt var det inte alltid så enkelt att påvisa vad man utvecklat hela tiden. Det som man hade kunnat göra om det hade accepterats i teamet samt från produktägaren är att låta utomstående individer att testa appen och lämna feedback.

## 4.3 The sprint review (either in terms of outcome of the current week's exercise or meeting the product owner)

A:

Vi hade reflektion tillsammans varje fredag. Scrum mastern bestämde en tid i samråd med teamet, sedan satte vi upp ett onlinemöte där vi diskuterade allt som hänt under veckans sprint. Vi gick igenom vår scrum board och reflekterade över sprintens mål och om dem inte uppnåts så diskuterades varför och hur det skall lösas bättre i framtiden. Det reflekteras även om punkterna i materialet som tillfördes av kursen i helhet som sedan dokumenterades i vårt gemensamma dokument för teamet. Under reflektionen utvärderade vi även våra KPI:er samt uppdaterade de med veckans resultat så att de var i fas. Under denna period så diskuterades det även gällande social contract så att samtliga medlemmar höll sig inom ramen för vad som förväntades av varandra.

B:

Det som hade kunnat vara bättre hade varit om man hade formulerat ett antal frågor i förväg som skulle vägleda och effektivisera dokumenteringen av diskussionerna som framkom under varje enskilt möte. Något som även hade gynnats av det hade varit en mycket tydligare och sammanhängande struktur överlag som skulle vara lättare att följa över en längre tid eller för en utomstående individ utanför teamet.

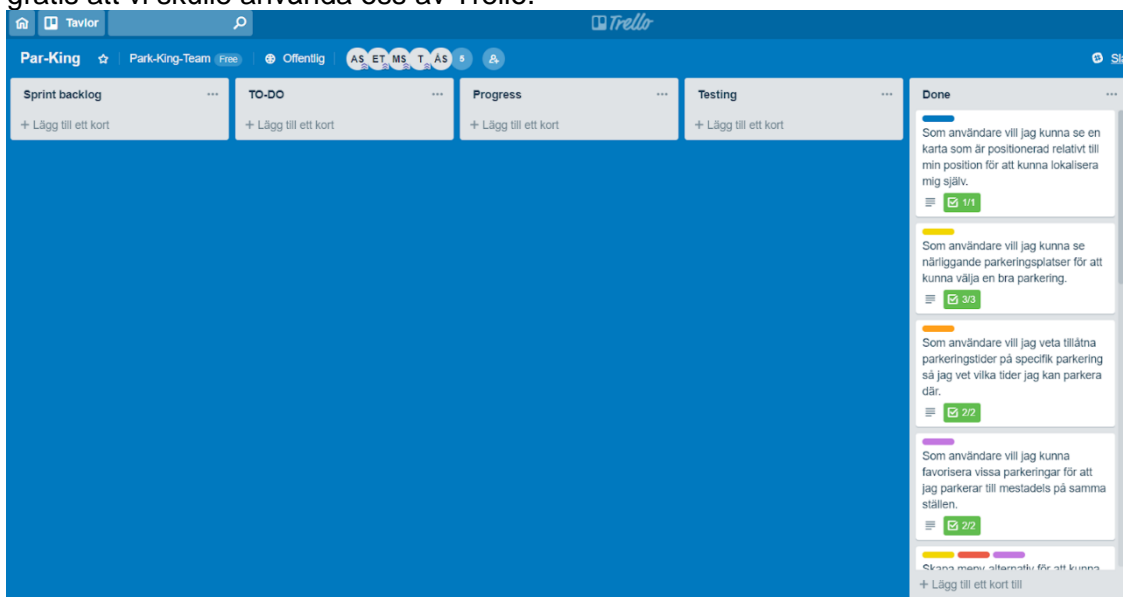
A → B:

Det som skulle behöva göras för att få en effektivare och tydligare struktur i dokumenteringen av reflektionerna hade just varit att ha en ännu tydligare definition och standardisering i hur man och vad man skriver under reflektionerna. Hade det funnits tydliga frågor hade det kunnat åstadkomma en mer riktad och effektivare reflektion angående projektet så hade det även kunnat bli mindre chans att väsentlig information missas i dokumenteringen. Det som man även kunnat göra med en sådan definition eller standard hade varit om man reflekterade om det täckte allt som skulle täckas och annars även den bli kompletterad med tiden.

## 4.4 Best practices for using new tools and technologies (IDEs, version control, scrum boards etc.)

A:

Vi ansåg att det bästa sättet för oss att hålla koll på de nya verktygen var att använda oss utav ett scrum board. Vi valde efter lite sökning vad som fanns som var lätt att använda samt gratis att vi skulle använda oss av Trello.



### Trellos struktur:

- Under Backlog placerades stories som vi planerar att göra men som inte är aktuella i nuvarande sprint.
- Under Sprint backlog placerades alla stories som skulle göras under den aktuella sprinten.
- Under TO-DO placerades alla nedbrutna stories som tasks som skulle göras under aktuella sprinten.
- Under Progress placerades tasks som någon har börjat arbeta på.
- Under Testing placerades tasks som var klara och redo för testning av andra i gruppen.
- Under Done placerades tasks och stories som mötte vår Definition of Done efter de har testats.

Vi integrerade även Slack i vår Trello för att få notifikationer när kort flyttades så att man snabbt fick reda på om något var klart eller liknande.



För att kunna så lätt som möjligt dela kod samt att slå samman det använde vi oss utav Github eftersom vi har redan har jobbat med det i tidigare kurser så var vi redan bekanta med det. Github används också av andra företag och i de flesta kod relaterade projekt. För att programmera vår applikation använde vi oss av Android Studio då ett kriterium var att den skulle vara baserad i Android. Detta gjorde det lätt att testa hur applikationen presterade i emulatorer samt lätt att överföra till våra Android telefoner för att testa "på riktigt". Android studio har också koppling till Github som gjorde det enklare att synkronisera med våra olika arbeten.

B:

Det hade varit optimalt att vara mer bekant med programmen vi använder oss av. Trots att vi kände till Github sedan innan så hade det sparat oss mycket tid och gynnat oss om vi vore mer bekanta med det. Detsamma gällde med Android Studio, förkunskap om dessa program hade gett oss en mycket smidigare början av projektet och vi hade sparat många timmar av onödigt krångel.

A → B:

För att kunna vara mer förberedd och bekant till framtiden så kan man ta första sprinten (som vi förvisso gjorde) och bekanta sig med programmen mer. Github är ett väldigt simpelt och smidigt program att använda men tar nog väldigt lång tid att bemästra så detta kan egentligen bara bli bättre med tiden du använder det, så det är väldigt svårt att förbereda sig mer med detta än de grunderna som alla hade när vi började med projektet. När det kommer till Android Studio och kodningen av applikationen så hade det gynnat oss om vi tog de flesta design decisions i början, för då kan man ta den första sprinten till att läsa om hur man exempelvis lägger till en appdrawer i en Android applikation eller hur man hanterar xml data.

## 4.5 Relation to literature and guest lectures (how do your reflections relate to what others have to say?)

A:

Tyvärr så läste vi som grupp inte mycket av den angivna litteraturen utan det vi som grupp fick mest nytta av i relation till föreläsningar och litteratur var de tre övningarna som gjordes i början av kursen. Där fick vi inblick om hur man skall tänka för att jobba agilt och med scrum. Kata övningen gjorde så att vi förstod vikten av att testa sig fram, ändra vad som gick fel och snabbt göra förbättringar men även att man snabbt kan bli sämre om en ändring inte var optimal. Lego övningen visade hur viktigt det är att jobba effektivt som ett lag då vi snabbt delade upp roller inom laget för att optimera den korta tiden i hade, men också hur viktigt det är att ha bra kommunikation med produktägaren. Elefant carpaccio övningen lärde oss hur man bör tänka inför att bryta ner stories för att alltid kunna leverera värde, detta görs genom att göra tasks vertikala samt att göra de självständiga och separerade från varandra. Det var under denna föreläsning vi lärde oss mycket om INVEST kriterierna vilket var något som vi tänkte på mycket då vi skapade stories och tasks.

Det vi tog med oss från Semcon föreläsningen var att dokumentera det vi gör och inte ta den korta vägen bara för att det är bekvämare.

Föreläsningen med Michael från Trine gav oss en inblick om hur stora företag jobbar agilt vilket var väldigt givande då vi kunde relatera hur vi arbetar i vårt lilla projekt och hur det görs ute i branschen.

B:

Nu i efterhand så känner vi att det hade varit bättre om vi läste oss vad litteraturen hade givit oss i början av kursen så vi hade fått en större överblick för vad som bör beröras under projektet. Vi i gruppen använde oss av litteratur som var utanför den som fanns på kurshemsidan, t.ex. Youtube och andra sidor som vi kände gav mycket information. Det som vi däremot hade kunnat göra i början är just att planlägga vilka delar av processen som vi skulle kunna inför varje sprint. Om man hade haft en bättre överblick över vad som skulle tas in så hade det varit enklare att ha en effektiv progression i lärandet av scrum.

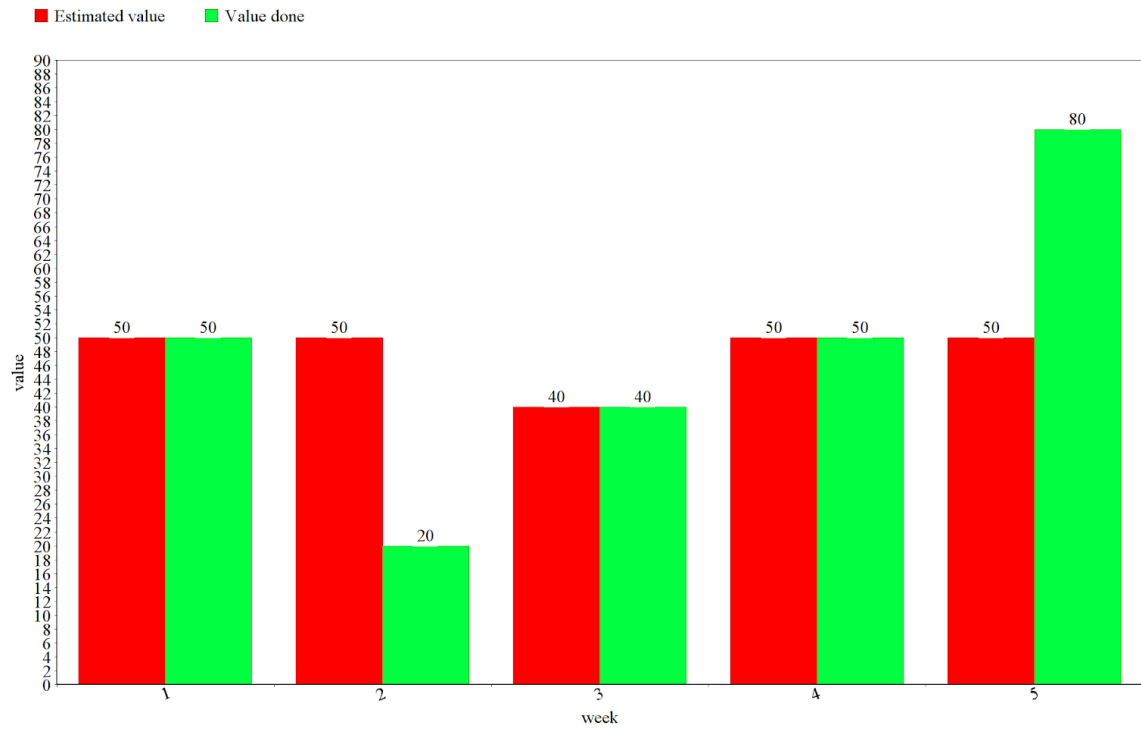
A → B:

Vi kan känna att problematiken som var riktad till gästföreläsningarna var att de skulle vara tydligare beskrivna sedan innan vad de skulle bidra med för projektet. Det som hade kunnat vara annorlunda hade kunnat vara att man hade kunnat hitta information angående vad föreläsningarna handlade mer om och sedan kunna göra en bättre bedömning om man skall närvara på föreläsningen eller om man ska läsa in material på eget håll. En bättre kommunikation mellan handlarna i kursen med gästföreläsarna hade uppskattas också då handledarna målade upp en viss bild på vad en av föreläsarna skulle diskutera men sedan när vi närvarade på föreläsningen så representerades inte det som handledarna hade tytt på sedan innan.

# 5. Bilagor

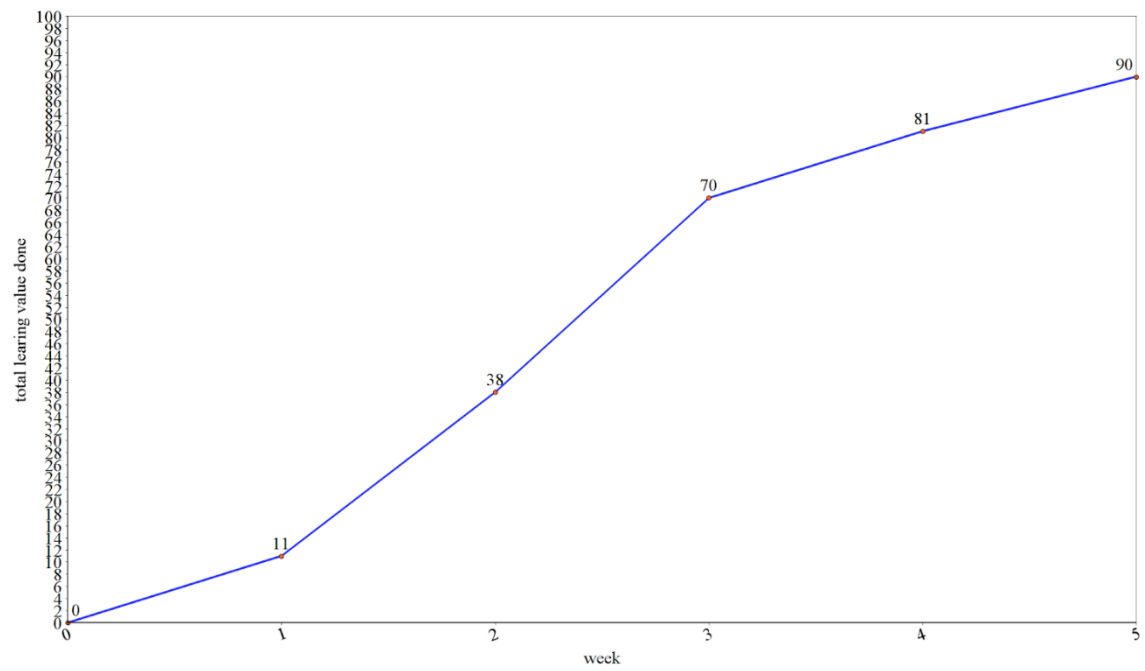
## Bilaga 1

KPI



## Bilaga 2

KPI



## Bilaga 3

KPI

