

Final Reflection DAT257 Team 4

Nedan följer vår slutliga team reflection där vi reflekterar över hela processen med utvecklingen av vår applikation för Chalmers Women Association. Strukturen liknar våra tidigare team reflections, men vi tar i denna med samtliga punkter som nämns i kurslitteraturen. Detta gör denna reflektion mer omfattande än tidigare reflektioner. Då vi i denna reflektion kan koppla hela arbetets utveckling till vår nu färdiga produkt, samt feedbacken vi fått på denna så drar vi även mer djupgående slutsatser gällande vårt projekt med avstamp i A,B,A->B loopen.

Customer Value and Scope

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

(A)

Vi har under arbetets gång lagt ner mycket tid på att prioritera funktioner till vår applikation i samråd med vår PO. Tidigt i projektet förklarade vår PO, Chalmers Women Association (CWA), att fokus vid utveckling av applikationen skall vara design och snarare minimalistiska funktioner än ett alltför stort fokus på att erbjuda komplexa funktioner. För att skapa värde för vår PO samt uppfylla vår MVP har vi prioriterat att utveckla hållbara lösningar i Wix-hemsidans backend. Designen av applikationen har utvecklats utefter arbetet med att behandla datan i CWA:s dataregister som Wix tillhandahåller. Tidigt förmedlade vår PO att medlemskaps-kort var en högprioriterad del av applikationen. Detta har vi dock, efter diskussioner med vår PO, valt att inte ta med och istället valt att använda oss av biljetter med QR-koder, då detta skapar mer värde för vår PO och appens framtida användare.

(B)

Vi har varit noggranna med att ständigt låta vår PO granska vår utveckling samt ta in och analysera dennes feedback, dock har vi egentligen inte fått någon kritik. Alltså; under de möten vi haft så har vår PO, i princip, genomgående varit positiv till vår nya funktionalitet och våra val av design, det är möjligt att detta har att göra med att det tekniska är svårt att förmedla och att vår PO då kanske tänker att vårt arbete är svårt att förändra och liknande. Med hänsyn till detta är det möjligt att vi skulle ha kunnat få mer konstruktiv feedback om vi på något sätt hade förändrat vår PO:s roll och deltagande i utvecklingen.

(A→B)

Det är möjligt att vår PO kanske skulle integreras mer i vårt arbete och därmed skulle denna kunna få mer djupgående förståelse av vår utveckling för att då kunna komma med lite mer

feedback. Detta skulle kunna ske genom att ytterligare boka in möten med PO:n, denne har i stor utsträckning varit tillgänglig för oss, men vi kanske skulle ha bokat in möten med denna i något större utsträckning. Utöver det så har vi jobbat emot en person från CWA, här kanske vi borde ha försökt att få med fler aktörer från CWA för att ytterligare kunna få mer feedback. Dock så har vår PO regelbundet fått med demon att visa upp för hela sin organisation, vilket även detta har resulterat i, i stort sett, enbart positiv feedback.

The success criteria for the team in terms of what you want to achieve within the project (this can include the application, but also your learning outcomes, your teamwork, or your effort).

(A)

Vi har haft en arbetsprocess där vi har prioriterat funktioner att utveckla och sedan har vi delat upp oss i subgrupper där vi har utvecklat dessa. I subgrupperna har vi ofta arbetat i par och vi har skiftat grupper och arbetsområde, så att team-medlemmarna har fått arbeta med olika medlemmar samt på olika områden. Detta har gjort att vi både har fått ta del av varandras kompetenser, men också lära oss om de olika arbetsområdena, programvaror, designverktyg och kodspråk, t.ex. Http, dart, flutter, android studio, Wix, Databaser och objektorienterad programmering.

(B)

Vårt ursprungliga mål var att lära oss om apputveckling samt användning av Scrum i mjukvaruprojekt och att hantera samt kommunicera med våra stakeholders. Vi vill även få godkänt i kursen och leverera en produkt som vår produktägare blir nöjd med. Under projektets gång har vi stött på nya utmaningar och funnit nya verktyg för att klara dessa, då har vi försökt lägga till dessa i våra interna läromål för kursen, t.ex. lära sig http, skapa animationer i Rive, lära sig en del saker i Wix m.m. Det är såklart möjligt att vissa teammedlemmar har lärt sig mer än andra inom specifika områden, vilket kan förklaras med att då någon har lagt mer tid på en del av utvecklingen och därmed blivit bättre på denna skapar det incitament att låta den fortsätta utveckla inom detta område eftersom att det då ökar utvecklingens effektivitet.

(A→B)

Vi ska vara öppna för nya metoder och lösningar på hur vi effektiviserar vårt arbete såväl som lärande. Vi bör återkoppla efter sprints och under projektets gång för att se om team-medlemmarna känner sig nöjda med vårt sätt att nå de mål vi satt upp samt att man fått lära sig det man ville. Genom team-reflections har vi tillsammans kunnat utvärdera och kvantifiera våra framsteg och vårt samarbete. En viktig sak att tänka på är hur vi delar upp de mindre grupperna som jobbar i breakout rooms. Det är svårt med vår begränsade erfarenhet att avgöra vad som är mest effektivt för inläring. Vi har försökt placera en som är duktig

inom ett arbetsområde med en som vill lära sig det för att underlätta arbetet men då finns det en risk att personen som redan är duktig inom arbetsområdet inte lär sig något nytt. Samtidigt kan det vara svårt att placera två stycken som inte har någon erfarenhet inom ett arbetsområde i samma grupp då det kan sluta med att de inte gör några framsteg och behöver då ta hjälp från andra medlemmar. Vi har ytterligare försökt att överbrygga problematiken med att det finns större incitament att låta någon/några med mer erfarenhet inom ett visst område fortsätta jobba på det genom att diskutera vad som faktiskt är våra mål med kursen och därmed vad som innebär att effektivisera projektet. Här menar vi att det också innebär en ökad utveckling av projektet om medlemmar i största möjliga mån får lära sig så mycket som möjligt inom varje område av utvecklingen, men här har ibland tid varit en faktor som har gjort det svårt att utnyttja detta. Det går att diskutera kring att vårt projekt då kanske är för omfattande eller liknande, men samtidigt är vi kollektivt nöjda med våran respektive inlärnin g på de olika områdena för projektet.

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.

(A)

I "Team reflection 24/4" skrevs det att vårt arbetssätt kring user stories började med att definiera produktägarens MVP. Utifrån det utformades två epics som speglar MVP. Mindre beståndsdelar bröts ut för att utforma varje enskild user story. *"När vi formulerade våra user stories använde vi den kunskap som vi fått från lektionerna och från Kinbergs bok. Våra task breakdowns har vi tagit fram genom att försöka förstå vilka steg som behöver göras för att en user story ska uppfyllas."* detta skrev vi i vår "Team reflection 24/4". När det gäller estimering av tid så använde teamet sig av poker planning, där man för varje user story röstar och/eller argumenterar för varje valt tidsestimat. Diskussion uppkom när man hade en stor varians mellan rösterna. I dessa fallen uppnåddes konsensus genom diskussion och sedan en ny iteration av poker planning, därav bildas teamets tidsestimat för varje user story. Lagets scrumboard för user stories innefattar product backlog -> sprint backlog -> Done. Detta är en livscykel för en user story som har definierats och fått ett tidsestimat. User stories är utgångspunkten för vårt arbete och det är med hjälp av user stories som vi kan strukturera värdet vi ger produktägaren. Vi har bjudit in vår produktägare till vårt scrumboard. Därmed kan vi diskutera och stämma av ifall det vi utgår ifrån är rimligt och rätt. En sak som vi blivit bättre på sedan "Team reflection 24/4" är hur vi reflekterar på user stories efter varje sprint. Tack vare reflektionerna har vi fått nya insikter om hur vi ska definiera user stories, göra dem "vertikala" och estimerar tiden på bästa sätt.

(B)

Det som vi ej har behandlat i våra user stories är definition av acceptanskriterier. Detta har vi istället diskuterat muntligt och haft icke dokumenterat, vilket kanske inte är det mest effektiva sättet att jobba med dessa kriterier.

(A→B)

I framtida projekt bör vi fokusera på att utforma varje aspekt av user stories. Från att definiera tasks, estimerar tiden och, framförallt då, lägga till acceptanskriterier. Vi anser att vi hade följt processen som vi haft i detta projekt men utöver det även definiera acceptanskriterier för varje user story. Med hjälp av acceptanskriterier skulle vi även kunna lägga till tester som kollar ifall dessa är uppnådda innan man får markera en user story som klar. Detta hade kunnat inkluderas i definition of done.

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

(A)

Vi använder oss av tre olika KPI:s, två för att kontrollera vår utveckling och en för att kontrollera vårt välmående över sprintsen. De som representerar vår utveckling är i formen av s.k. "burndown charts", här representerar den ena hur mycket tid vi har planerat att lägga på projektet och jämför det med vår tidsestimering för projektet. Den andra jämför antalet genomförda stories och jämför dessa med antalet planerade stories, jämt fördelade över våra fem sprints. I välmående-grafen fyller team-medlemmarna i sitt mående under respektive sprint. Ett tydligt resultat är att vi underestimerade tidsåtgången för våra sprints och att vi konsekvent behövt lägga mer tid än planerat på våra stories. Det har dock skett en medveten underestimering för att ha tid för eventuella missar. Detta syns i resultatet då vi har några man hours över i slutet av sprint 5. Vi klarade nästan av samtliga av våra 20 stories, men lämnade en av dem då den kändes meningslös i termer av att öka kundvärdet. Vi såg också en dipp i välmåendet på grund av hög arbetsbelastning under sprint fyra, som reparerade sig något när vi såg "ljuset i tunneln" under sprint 5. Det går eventuellt att se en korrelation mellan hög avvikelse mellan elapsed man hours och estimated task hours och lågt välmående under en sprint, då detta kan ha ökat stressnivån och överarbetstiden.

(B)

Under det här projektet har vi väl egentligen inte analyserat och använt våra kpi:er i särskilt stor utsträckning, det är möjligt att vi hade kunnat öka exempelvis det genomsnittliga välmåendet genom att på något sätt granska vår välmående-kpi. Vi borde kanske även ha använt tidsestimering och storyestimerings-KPI:erna för att planera våra sprints utifrån hur många timmar vi behöver lägga samt hur många stories vi behöver genomföra.

(A→B)

I framtiden bör vi inkludera en värdering av våra kpi:er halvvägs genom varje sprint och använda dessa för att kontrollera och kvantifiera vår progression samt teamets kollektiva och individuella välmående. En tanke vi diskuterat i efterhand är ifall vi skulle fylla i välmående KPI:et mitt i sprinten och anpassa velocity beroende på det. När vi har en tuff vecka kan vi eventuellt dra ner på velocityn och när vi har en bra vecka kan vi öka den. Genom att kolla på välmående grafen kan vi få ett snitt på hur teamet mår och bör då diskutera detta efter varje sprint för att se om något i vår arbetsprocess bör förändras eller liknande men det begränsar oss från att påverka den pågående sprinten. I framtiden hade det varit mer optimalt att alla medlemmar fyller i välmående KPI:et halvvägs in i sprinten.

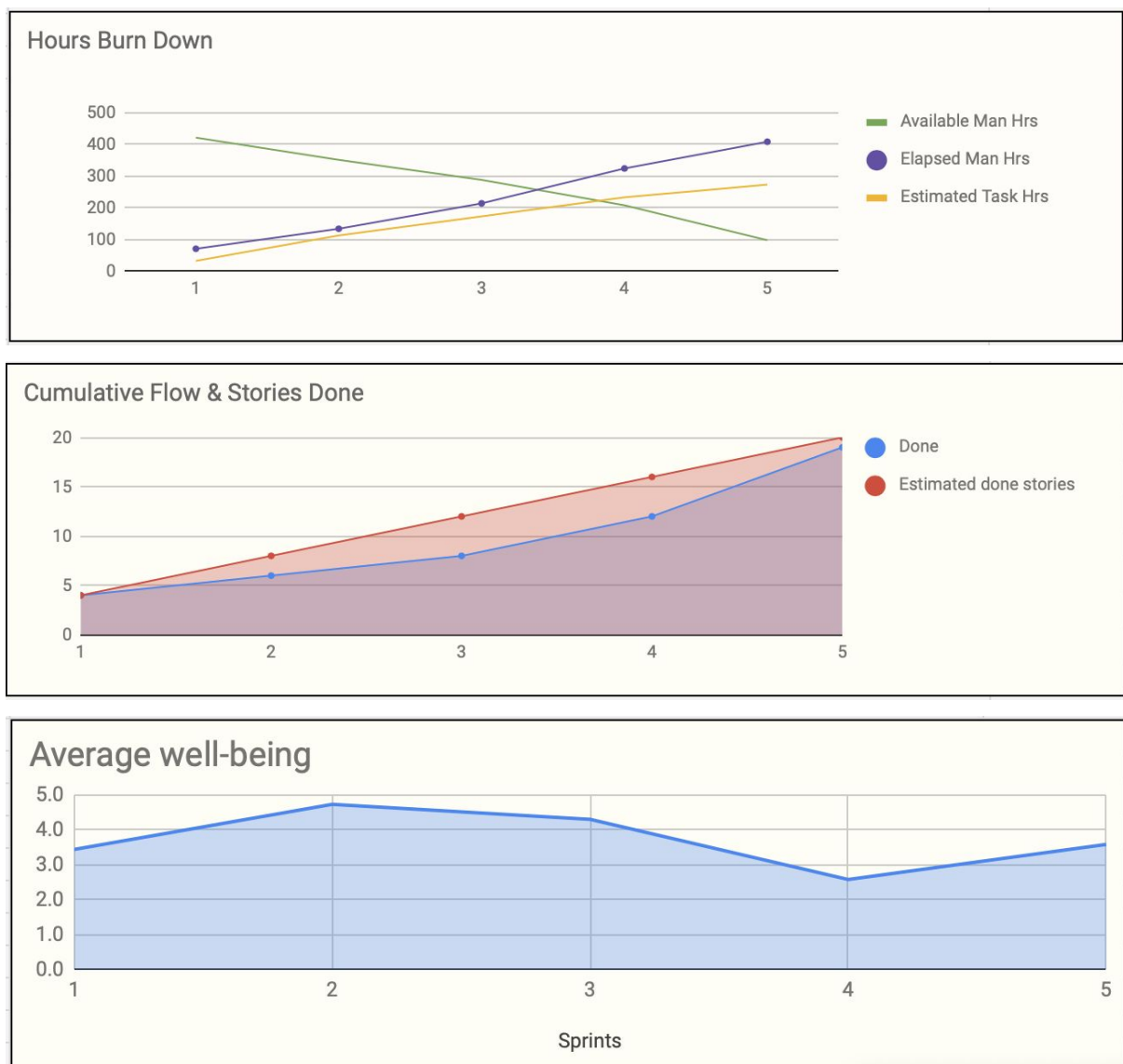


Fig. 1. Våra tre KPI:er

Social Contract and Effort

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 (this means, of course, you should create one in the first week and continuously update it when the need arrives.

(A)

I ett tidigt skede av utvecklingsprojektet formulerade vi ett social contract. Kontraktet innehåller de överenskommelser som gruppen tillsammans upplevde som nödvändiga för att uppnå ett gott resultat, både för kunden/produktägaren och oss själva. I kontraktet framgår vilken tid medlemmarna bör och vill lägga ner varje vecka, vilka roller de olika medlemmarna tar, hur våra veckomöten ska se ut, hur sprintsen ska vara utformade, vad våra success factors är för både tasks och projektet som helhet och en hel del annat. Vårt fokus har varit att skapa en produkt som vi och vår kund blir nöjda med samt att lära oss av varandras kompetenser. Planen var initialt att kontraktet skulle följas upp vid varje veckomöte, men så har det inte sett ut, det sociala kontraktet och mötet som hölls då har snarare lagt grund för att team-medlemmarna skulle kunna förstå varandras olika ambitioner och mål med kursen, vilket definitivt har förbättrat vårt samarbete samt kommunikationen inom gruppen. Det går att se att många av de formella riktlinjer och roller vi formulerade i vårt kontrakt har följts noggrant, t.ex. så har vi jobbat i ett mötesprotokoll, vi har haft två möteskoordinatorer, en mötesledare(vilket dock har tagits över av veckans scrummaster), en konfliktledare(vilken dock inte har behövt gripa in än), samt en sekreterare. Men vår kanske allra viktigaste punkt under mötet om det sociala kontraktet handlade om att samtliga medlemmarna har haft som mål att lära sig så mycket som möjligt under kursen, av varandra och under projektets gång. Detta är något vi regelbundet tagit upp under våra sprinter och möten, och som vi har jobbat med genom att demonstrera och förklara vad respektive medlem har utvecklat samt låtit alla medlemmar få ta del av olika delar av utvecklingen, t.ex. http-kopplingar mellan Wix och appen, samt ren utveckling av appen och dokumentation, trots att det har gjort att arbetet kan ha blivit lite mindre tidseffektivt. Även om kontraktet inte följdes upp och uppdaterades varje vecka så har vi förändrat delar av det, t.ex. vår DoD, sedan har vi även under sprint retrospective och liknande möten tagit upp saker som är formade från vårt sociala kontrakt, e.g. om alla känner att de har fått lära sig det de vill och liknande.

(B)

Vi insåg tidigt att andra saker prioriteras högre än att hålla oss till punkt och pricka till det sociala kontraktet, bland annat utformning av möten, kontakt med kund, skapa en kodmiljö, lära oss nya kodspråk och arbetssätt(dvs. dart och scrum). Även om vi kanske borde ha följt upp det sociala kontraktet oftare har det känts som att kontraktet är viktigt men att kursen är tidskrävande och att våra prioriteringar behöver riktas åt andra håll för att hinna med det hela. Tidsaspekten är därmed kanske en faktor som har påverkat vårt användande av det sociala kontraktet, hade vi haft mer tid under respektive sprint skulle vi ha kunnat iterera tillbaka till det sociala kontraktet i större utsträckning. En annan faktor är att om vi hade stött på sociala

motgångar och möjligtvis konflikter hade det sociala kontraktet kunnat spela en viktig roll. Då hade det varit viktigt att någon lyfter det och det är möjligt att det blir svårare att ta upp det sociala kontraktet om man under tidigare veckor inte alls har kollat tillbaka på det. Därmed är det möjligt att vi borde ha itererat tillbaka mer till vårt sociala kontrakt, även om det trots detta har påverkat vårt arbete på ett bra sätt och att vi till stor del har följt det.

(A→B)

I en tidig team reflection nämnde vi att vi skulle försöka ta fram det sociala kontraktet oftare, detta gjordes dock inte. Vi skulle ha kunnat ta fram det sociala kontraktet kanske en gång per sprint och snabbt se till så att alla känner att det som sagts fortfarande uppfylls. Det hade inte varit särskilt tidskrävande, men hade kunnat underlättat ifall att vi skulle ha stött på motgångar eller hamnat i interna konflikter. Det är dock möjligt att arbetsbördan i relation till tiden gjorde det svårt att hinna med, då är det möjligt att vi tidigt skulle se till att skapa oss tid för att hålla ett möte där vi tog upp det sociala kontraktet, det hade inte behövt ta särskilt lång tid. Alternativt så skulle vi kunna ha något möte med återkoppling till det sociala kontraktet varje kvartal eller till och med ett möte halvvägs in i projektet.

The time you have spent on the course and how it relates to what you delivered (so keep track of your hours so you can describe the current situation).

(A)

Under projektets uppstart fick team och individual reflections vara med i varje veckas tidsplanering, vilket vi däremot inte hade råd med under projektets slutliga veckor. Som vi skriver i team reflection 24/4 har vi *“estimerat tidsåtgången för varje task genom Poker Planning, vilket har varit svårt till en början då vi inte vet hur mycket tid som kommer läggas på att komma in i ett nytt kodspråk och arbeta efter Scrum, som är ett arbetssätt som inte använts av någon i teamet tidigare. Vi har därför valt att lägga tiden i överkant för att räkna in det eventuella extrajobb som det kan orsaka.”*. Vi har regelbundet efter varje sprint i projektet sett till att ha en färdig tidsplan för kommande vecka i form av estimerade user stories, tillsammans med sina tillhörande tasks.

Det som går att se utifrån ett utvecklingsperspektiv i vår arbetsprocess är att vi progressivt slutade planera in tid åt dokumentationen i projektet. Dokumentationen innefattar då allting från estimeringar av user stories till skrivandet av team reflections. Detta var inte direkt något aktivt val vi gjorde utan vi blev bättre och bättre på det tills vi tillslut kände att vi inte behövde lägga undan tid för det. Som man kan se på en av våra KPI:er som omfattar tidsberäkning, ser man att laget tenderar att jobba fler timmar än estimerat (Se Fig. 1). Detta skulle kunna bero på att vi slutade planera in tid för dokumentering, men samtidigt skulle det kunna bero på att appen vi utvecklade begärde progressivt mer och mer GUI. Vidare är det

viktigt att ta i åtanke att produktägaren prioriterar ett välgjort grafiskt gränssnitt högt. Detta resulterade i att vi la extra tid på att förfinas GUI:t med exempelvis animationer, utan att det var ett kriterium i tasks:en/user story:n.

(B)

Läser man i team reflection 24/4 där vi skriver om den aktuella frågan ser man att vi önskade oss en mer detaljerad tidsplanering, till exempel att vi borde tidsestimera varje task för att bättre kunna fördela arbete och effektivisera oss. *“Vi kommer tydligare behöva definiera tidsåtgången för varje task för att jämnt fördela arbetsbelastningen för gruppmedlemmarna, men också arbetsfördelningen mellan varje sprint. Eftersom det kändes som att våra första tasks gick snabbare än beräknat behöver vi nu fundera ett extra varv kring tidsåtgången för de resterande tasken, så att vi i kommande sprints slipper arbeta övertid.”*. Detta var tydligt något vi inte applicerade i vår projektplanering eftersom vi i team reflection 22/5, drygt en månad efter att vi skrev det som citeras ovan, återigen skrev nästan samma sak som förbättringsförslag. Det är svårt att veta exakt vilken anledningen är till att vi inte började estimera individuella tasks, men vi tror så här i efterhand att det helt kändes onödigt i stunden eftersom estimeringar kostade så mycket tid i början av projektet.

(A→B)

Vi som grupp tror att om vi hade implementerat en beprövad metod för att estimera tasks så hade det kanske faktiskt gått att realisera och tillföra något värde i form av effektivisering av vårt arbete med appen. Men eftersom vi inte hade något beprövad metod att gå på resulterade det i att vi inte estimerade enskilda tasks, utan enbart user stories. Vi tror att om vi hade utfört ett nytt projekt liknande detta någon gång i framtiden, då hade vi inte heller valt att estimera enskilda tasks. Vi anser som grupp nu i slutändan att det är överflödigt arbete trots att vi både är bättre och snabbare på att estimera uppgifter. Har man en så stor task att den behöver estimeras går det nästan alltid att bryta upp denna i flera mindre tasks som inte kräver en estimering.

Design decisions and product structure

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

(A)

Eftersom att en av de viktigaste kriterierna för produktägaren var att ha en snygg och konsekvent design rent grafiskt, har vi tidigare resonerat mycket kring API:er och bibliotek som hjälper till med detta. I “Team reflection 30/4” skrevs det att vi lagt till beroende till ett bibliotek kallat “feather icons” i flutter. Detta möjliggjorde en konsekvent design gällande

ikoner i appen. I samma team reflection skrev vi att vi utnyttjar material-design som en av våra grunder för designval och dessutom utnyttjar deras bibliotek för widgets som är starkt förknippade med flutters egna ramverk. Gällande design av kod så hade vi i dåläget (30/4) inte kommit jättelångt på källkoden och det nämndes att designmönster för en modulär design var önskvärt. Däremot nämns det inget vidare om det i samma punkt som framkommer i “Team reflection 15/5”. I nuläget har vi inte implementerat specifika designmönster för att få en mer modulär design. Detta grundar sig i att dart med flutter som ramverk inte är så likt ren java-programmering. Eftersom allting grundar sig på widgets och inte öppnar upp för designmönster som exempelvis MVC-mönster (och vi ej är lika insatta i dart som i java) så har vi ansett att vi inte kommer hinna att läsa på om olika designmönster och sedan implementera eller refaktorera koden. Ifall vi hade försökt implementera designmönster, så hade vi kanske behövt lägga mindre tid på andra saker och därmed minskat det som produktägaren anser vara av värde. Sammanfattningsvis så har det varit en prioritetsfråga. En bra och modulär design är alltid bra, men på 6 veckor där produktägaren anser att funktionerna av appen och det grafiska gränssnittet är det som är värdefullt, så har vi självfallet försökt uppnå det som produktägaren vill ha i första hand.

(B)

Det bästa möjliga hade varit att kunna uppnå de kriterier som produktägaren blir nöjd över samt att ha en källkod som är modulär med bra utvalda designmönster som följer SOLID-principerna. Trots att vi har utvecklat en app under 6 veckor så är tanken att appen ska användas i praktiken också. Det innebär att med tiden så kan det behövas uppdateringar eller nya implementationer. Med en modulär design underlättar man för framtida utvecklare med nya implementationer eller ändringar.

(A→B)

I ett framtida projekt är vi alla eniga om att det hade varit bra att läsa på om programmeringsspråket i fråga och dess olika standardiserade designmönster. Detta för att redan från början få en inblick i vad konventionerna är och dessutom minska risken för att behöva refaktorera koden i efterhand. Dessutom skulle man kunna diskutera dessa frågor med produktägaren för att förklara vad värdet av att ha en modulär design innebär. Då kanske man inte skulle behöva prioritera bort en sådan viktig aspekt av utvecklingen och samtidigt “framtidssäkra” ens kod. Det hade även frigjort mer tid till att fokusera på stakeholdern, snarare än att lära sig nya kodspråk och liknande, dock så är det värt att tillägga att vår stakeholder är “otroligt nöjd med produkten”.

Which technical documentation you use and why (e.g. use cases, interaction diagrams, class diagrams, domain models or component diagrams, text documents)

(A)

Det är viktigt för vår externa aktör att designen var konsekvent till deras hemsida och image. Vi använder oss därför av ett designdokument där vi delar ideer och tankar kring det grafiska gränssnittet för applikationen. Vi har använt det för att enkelt kunna kommunicera om olika design-ideer innan dessa implementerats. I dokumentet ingår även färgkoderna vi ska använda, vilket gör att färgkoderna är lättillgängliga till alla medlemmar. Detta försäkrar att alla färger i applikationen är koncisa.

(B)

Vi har inte använt oss av dokumentation på kodens struktur såsom UML-diagram, detta hade dock i efterhand antagligen gjort det enklare för oss att skriva modulär kod. Det hade även underlättat för oss och andra människor i framtiden ifall vi dokumenterade mer i vår kod t.ex. förklara metoder och klasser i koden.

(C)

I framtiden får vi planera in tid för att rita/generera UML-diagram och använda dessa under arbetet som ett hjälpmedel för att designa vår kod. Vi behöver även reflektera över våra tekniska dokument och fundera över ifall vi behöver några nya för att hjälpa oss skapa den produkt som vår externa aktör vill ha.

How you use and update your documentation throughout the sprints

(A)

I "Team reflection 30/4" skrevs det att vi har ett mötesprotokoll för att dokumentera vad vi har gjort/diskuterat tidigare och vad som skall ske. Mötesprotokollet är en redogörelse där vi skriver ner vad vi har att göra, vad vi har kommit fram till, frågor som vi vill ställa till produktägaren (& produktägarens svar) och vem som är veckans scrummaster m.m. Dokumentet är informellt skrivet men är ett bra sätt att uppdatera och hålla reda på var i utvecklingsfasen laget befinner sig i. Utöver det så har vi även separata sprint-dokument där vi dokumenterar scrum-relaterade ämnen. I dessa skriver vi ner vår sprint-planning och sprint-retrospective. Dessa uppdateras ständigt i början och slutet av varje sprint.

(B)

Med dokument som uppdateras ständigt har man en bra struktur och dokumentation som man alltid kan falla tillbaka på ifall någonting är oklart. Däremot är det viktigt att hålla det måttligt så att man inte lägger allt för mycket tid på att dokumentera istället för andra viktiga aspekter i ett agilt projekt. För lagets del så har det varit givande att dokumentera och det har även varit i en måttlig mängd. Så slutsatsen är att man inte hade behövt dokumentera mer men istället jobbat med kvalitén på dokumentationen.

(A→B)

För att säkerställa att dokumentationen är av kvalitet, att vem som helst kan läsa och att man kan se en kontinuitet mellan tillfällena då man dokumenterar, så hade man i förhand kunnat utforma specifika mallar för samtliga typer av dokument. Med en stiftad mall underlättar man uppdateringar av dokument. Med tanke på hur informellt skrivet mötesprotokollet är så hade man kunnat skapa en mall för att öka kvalitén på både dokumentationen och t.o.m. strukturen på möten.

How you ensure code quality and enforce coding standards

(A)

Under projektet tänkte vi inte till en början på hur vi ska försäkra att vi skriver bra kod. Vi demonstrerade all kod för varandra redan under första sprinten och alla fick se över den och ge sina tankar på den. Detta system visade sig fungera ganska bra eftersom alla fick lära sig av varandra samtidigt som koden kollades över. Vi har senare uppdaterat vår DoD där vi har som krav att alla ska ha sett över och godkänt delar av kod som har lagts till. Denna process har följts, vi har arbetat under kortare sessioner där vi delar upp arbetet för att sedan demonstrera den nya koden för resten av laget, vilka sedan kan komma med förslag på förändringar eller frågor. Vi försöker såklart att skriva kod enligt best practice och eftersom att Dart (kodspråket vi skriver i) är ett objektorienterat språk kan vi implementera detta i vårt arbete. Vi testar även vår kod regelbundet, dock har vi inte gjort några unit tests under arbetet då det mesta i Flutter är grafiskt. Vi har även inte lagt särskilt mycket fokus på att följa en objektorienterad design. Det är något vi har haft i baktanken men på grund av vår begränsade tid har vi inte lagt något större fokus på det.

(B)

Det vi skulle vilja göra i framtiden är att lägga större fokus på att göra vår kod mer modular. Vi hade även kunnat använda oss av GitHubs code review-funktion. Dock såg vi mer värde i att göra våra code reviews som nämnts tidigare eftersom de sessionerna var tillfällen att lära sig nya saker av varandra.

(A→B)

För att åstadkomma det vi vill kan vi ändra hur vi utför våra code reviews så att varje gruppmedlem ser till att utvärdera de andras kod genom att t.ex. Använda GitHubs code review-funktion. Detta kan eventuellt spara oss tid vilket ger mer tid till att försöka designa koden på ett objektorienterat sätt. Med den erfarenhet vi fått under projektets gång kommer det i framtiden vara enklare för oss att jobba med bland annat Scrum, Flutter och https funktioner. Allt detta gör det enklare för oss att bedöma kod och det kommer spara oss värdefull tid som vi kan lägga på att designa koden bättre. Fortsatt erfarenhet och goda rutiner kring code review kommer leda till att vi skriver kvalitativ kod.

Application of Scrum

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

(A)

I början av projektet så hade vi delat in medlemmar i olika roller, t.ex. mötesledare och konflikthanterare, men under projektets gång så försvann dessa och ansvaret föll istället på gruppen som helhet. Detta var inget som vi tänkte på, utan skedde naturligt under utvecklingen av appen. Detta är något som vi anser fungerat bra, då alla har fått testa på olika arbetsroller och tillsammans gjort alla beslut. Men det skulle även kunna ha varit givande ifall vi hade hållit oss till rollerna för att få en mer strukturerad arbetsprocess och alla medlemmar hade fått lära sig mer om en roll. De enda rollerna som vi hade under hela projektet var Scrummaster och PO. Vi hade en roterande Scrummaster, så att alla skulle få känna på den rollen och få testa på en mer ledande roll. Detta har gjort att alla medlemmar har fått en djupare förståelse för Scrum och kan använda det i framtida projekt. PO, som var vår externa aktör, har också haft en viktig roll i vår utveckling. Med hjälp av vår PO kunde vi definiera MVP och ta fram de viktigaste aspekterna av applikationen först. PO har även haft uppgiften att godkänna våra sprints, så att vi inte lagt till något som var oönskat i appen.

(B)

Som nämnt ovan har vi lyckats fördela scrum-master-ansvaret relativt bra. Vår initiala tanke var dock att alla skulle få chansen att vara scrum-master en vecka var. Tyvärr hann vi bara med fem hela veckor där vi hade en förbestämd scrum-master vilket ledde till att två personer inte hann prova på. En tydlig förbättringspunkt är därför att fördela tiden så att alla hade fått prova på att vara scrum-master. Ett annat tydligt mål som vi pratade om tidigt i processen var att fler roller skulle fördelas inom gruppen, så att medlemmarna får så mycket erfarenhet som möjligt. Utöver detta hade det också varit bra om vi stannade upp för att reflektera över hur gruppen kände kring sina arbetsroller.

(A→B)

För att ge alla medlemmar möjligheten att vara scrum-master så skulle vi direkt räknat ut hur många arbetsdagar vi hade totalt, och sedan fördela ansvaret rättvist inom gruppen. Under projektets gång så kunde den som varit scrum master leva sig in i det lite mer för att på så sätt skapa en verkligare arbetssituation, vilket hade kunnat leda till att man skapar värdefulla erfarenheter kring hur scrum implementeras. Detta har dock varit svårt då vi endast är sju medlemmar totalt, och att allokera en person till att enbart fokusera på scrum skulle vara tidsineffektivt. I vårt fall är scrum-mastern även en del av developer-teamet vilket gav oss extra resurser till att jobba på appen, men det bidrog därför till att scrum-metoder ibland kom i andra hand. Ett ytterligare alternativ hade kunnat vara att bestämma en person som genom hela projektet agerade scrum-master. Då hade denna personen successivt lärt sig mer och mer, vilket hade kunnat göra gruppen effektivare. Dock var ett av våra mål att alla skulle få testa på nya områden, vilket gjorde att vi ändå valde att rotera ansvaret inom gruppen. Vi hade också kunnat vara hårdare på att behålla våra andra ansvarsområden, trots att det inte kändes som att de behövdes i början. Ett exempel är att ingen enskild individ har haft ansvar över vårt mötesprotokoll. Detta fungerade bra i början, men med tiden så rann det ut i sanden. Därför hade det varit bättre att hålla sig till de förbestämda rollerna och dessutom sett till att alla haft en roll.

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

(A)

Vi har försökt följa scrum-ramverket så mycket som möjligt och med det kommer agila metoder, primärt sprints. Sprintsen har tvingat oss att kontinuerligt reflektera över vad som har och inte har fungerat. Genom våra sprints har vi kontinuerligt reflekterat över vårt arbetssätt för att kunna utveckla detta ytterligare. Detta har lett till att vi över tid blivit mer agila. I början av projektet hade vi svårt att definiera bra user stories. När det fungerade hade vi istället svårt att tidsestimera dem. Med tiden har vi dock överkommit båda dessa problem, vilket har lett till att vi lättare kan slutföra våra sprints i tid, samtidigt som vi inte heller har blivit klara långt innan planerat. Till skillnad från ovanstående punkt har vi hållit oss till de rekommendationer som presenterats i Knibergs bok, vilket har gynnat oss. I början var det svårt att se någon poäng i att väldigt ofta stanna upp i processen för att reflektera, planera

eller estimerar, men med tiden klarnade det och bidrog till teamets ökade effektivitet. För att visualisera våra sprints har vi använt trello som sprint board och estimerat stories vilket har gett oss en tydlig överblick. Genom hela arbetet har vi haft tät kontakt med PO och har därför aldrig behövt ändra några stora funktioner i appen. Internt i gruppen har vi återkopplat med varandra flera gånger om dagen när vi jobbar på projektet för att se till att alla är samstämmiga.

(B)

Vad vi kunde ha varit bättre på för att jobba ännu mer agilt skulle vara att använda oss av kpi:er redan från start. Detta hade gjort att vi snabbare kunnat åtgärda problem som kommit till ytan vid utvärdering av våra kpi:er, såsom välmående inom grupper eller hur väl vi ligger i fas. Det hade också hjälpt om vi fick återkoppling på hur vårt agila arbete fungerar från någon utanför gruppen.

(A→B)

För att kunna uppfylla det som nämndes i B hade vi kunna involvera handledaren mer, dels för att få återkoppling på projektets fortskridande i allmänhet, men även på hur väl vår agila process fungerade. Vi hade dessutom kunnat vara bättre på att *alltid* ha daily scrum, även när det känns överflödigt.

The sprint review and how it relates to your scope and customer value (Did you have a PO, if yes, who?, if no, how did you carry out the review? Did the review result in a re-prioritisation of user stories? How did the reviews relate to your DoD? Did the feedback change your way of working?)

(A)

Veckans scrum-master har sett till att en sprint review har genomförts. Det är scrum-mastern som har lett vår review. Vi har valt att hålla våra sprint reviews utan vår stakeholder som är vår PO, men vi har haft vår stakeholders respons på vår sprint i åtanke under dessa möten. Under dessa har vi kollat på vår trello-board (sprint-board) och diskuterat vår product backlog samt vad som har gjorts och vad som behöver göras. Dessa möten har lett till att vi har omprioriterat val av user stories med fokus på estimerad tid och prioritering från vår stakeholder. I våra möten med vår stakeholder får denna godkänna våra sprints. Detta är en del av vår DoD, vilket betyder att vår stakeholder indirekt medverkar i våra sprint reviews.

(B)

För att få mer användning av vår DoD ändrade vi den så att den skulle passa mer i vårt arbete. För att få en tydligare uppfattning av hur ett liknande projekt skulle se ut i arbetslivet så skulle vi ha kunnat ha våra möten mer formella. Eventuellt hade vi tjänat på att vår PO kom med egna definitioner av värde och vilka funktioner som känns viktiga att utveckla. I detta

utvecklingsprojekt har vi praktiskt taget frågat en stakeholder om de vill ha en app och sedan frågat vilka funktioner som behövs. Det har gjort att vi haft ett väldigt enkelt samarbete med vår PO, men vi hade kanske behövt utforma vår review session på ett sätt som hade utmanat oss och vårt arbete på ett bättre sätt. Vi har fram till detta stadiet t.ex. aldrig haft en user story som inte godkänts av vår PO.

(A→B)

Det är möjligt att vi som i en tidigare punkt sagt att vi borde ha involverat vår PO i högre utsträckning. I nuläget var vår PO:s roll till stor del att svara ja/nej på våra förslag av nya features, det är då möjligt att de kanske skulle snarare ge förslag på vad som var värdeskapande än att vi tog fram nya funktioner för godkännande, men här ser vi dock att kundnöjdheten dvs. "PO-nöjdheten" i vårt fall uppfylldes med vårt tillvägagångssätt, vilket är av främsta vikt.

Best practices for learning and using new tools and technologies (IDEs, version control, scrum boards etc.; do not only describe which tools you used but focus on how you developed the expertise to use them)

(A)

Vi har tagit in ny information om nya program och verktyg genom att kolla på tutorials och genom att läsa på nätet. Vi har lärt oss om verktygen genom att försöka implementera dem i vårt projekt, i.e. learning by doing. Vi har också försökt att förstå på vilket sätt ett visst verktyg kan skapa värde för gruppen för att på så sätt motivera oss själva till att lära oss mer om dem. Vi har fått fler personer att lära sig om tex scrum boarden genom att låta alla vara scrum-master någon gång genom projektets gång.

(B)

Vi skulle vilja att alla lär sig så mycket som möjligt genom att testa alla olika verktyg. På så sätt kan alla känna igen dem och kan lättare utveckla sin kunskap om dem i framtiden.

(A→B)

För att få så många som möjligt i gruppen att lära sig så mycket som möjligt om varje verktyg så bör varje person försöka utmana sig själv och försöka använda det verktyg eller teknik som den känner sig minst bekväm med. Problemet med detta är att det lätt kan bli tidsödande om alla medlemmar ska sätta sig in i varje process. Vi har hela tiden satt lärandet som högsta prioritet, men under arbetets gång har vi ibland tummat på detta, för att istället hinna med våra planerade user stories.

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

(A)

För alla scrum-relaterade saker har vi utgått ifrån Henrik Kinbergs bok “SCRUM AND XP FROM THE TRENCHES” och <https://www.scrum.org/>. Från boken och hemsidan har vi följt hur vi utför sprint planning, sprint review, sprint retrospective och daily scrum. Utöver detta har vi använt boken för att lära oss om vilken roll scrum mastern har i sprints.

(B)

Vi vill vara konsekventa i vårt sätt att använda Scrum. Det är möjligt att det hade varit intressant att ta in lite olika källor för vår användning av scrum, men eftersom att vi ville ha en konsekvent arbetsprocess och användning av scrum har vi valt att inte göra det.

Hursomhelst så hade det varit intressant och möjligtvis utvecklande för vår användning av scrum att använda oss av litteratur som mer fokuserar på att reflektera på hur scrum fungerar, nu har vi snarare bara strikt följt processerna som vår valda litteratur beskriver, snarare än att kritiskt granska och reflektera kring våra processer med hjälp av exempelvis gästföreläsningen.

(A→B)

Genom att kolla tillbaka på exempelvis gästföreläsningen och även ta hjälp från kursens resurser, såsom handledaren och kursansvariga skulle vi i större utsträckning kunna reflektera kring vår användning av scrum snarare än att endast implementera scrum utan att kritiskt granska vår användning. Dock så är vi eniga om att vårt arbete med scrum har varit ytterst effektivt, detta kan vara en anledning till att vi inte granskade vår användning av scrum då vi inte mötte några egentliga komplikationer under vårt projekt. Här skulle en lösning som sagt vara att använda oss av kursens resurser, då exempelvis handledaren eller kursansvarig kanske kan ge feedback som vi inte tänker på då vi som grupp får ett sorts kollektivt tunnelseende.