



iiCaptain

Team Wale Wale - Projektrapport (DAT255)

Fredrik Ingelsson

Max Edman

Julia Gustafsson

Mattias Lundell

Victor Persson

Eric Josefsson

TABLE OF CONTENTS

1 INTRODUKTION	3
2 APPLICATION OF SCRUM.....	4
2.1 ROLES, TEAMWORK AND THE SOCIAL CONTRACT.....	4
2.2 USED PRACTICES.....	5
2.2.1 <i>Pair-programming</i>	5
2.2.2 <i>Daily Scrum</i>	5
2.2.3 <i>Sprint Planning, Reviews and Retrospectives</i>	6
2.2.4 <i>Product Backlog</i>	7
2.2.5 <i>Sprint Backlog</i>	7
2.2.6 <i>Scrum of Scrums</i>	8
2.3 TIME DISTRIBUTION.....	8
2.4 EFFORT AND TASK BREAKDOWN	9
3 REFLECTION AND EVALUATION	10
3.1 REFLECTION ON THE SPRINT RETROSPECTIVES.....	10
3.2 REFLECTION ON THE SPRINT REVIEWS.....	11
3.3 REFLECTION ON THE RELATIONSHIP BETWEEN PROTOTYPE, PROCESS AND STAKEHOLDER VALUE.....	11
3.4 EVALUATION OF D1-D4	13
3.4.1 <i>D1 - Scrum Strategies</i>	13
3.4.2 <i>D2 - KPIs</i>	14
3.4.3 <i>D3 - Initial Product Backlog & Business Model Canvas</i>	16
3.4.4 <i>D4 - Half-time Evaluation</i>	17
4 BEST PRACTICES FOR USING NEW TOOLS AND TECHNOLOGIES	18
5 RELATING YOUR OWN PROCESS TO LITERATURE AND GUEST LECTURES.....	20
BILAGA 1 KPI:ER	21
BILAGA 2 SPRINT RETROSPECTIVE DOCUMENTATION.....	24

1 INTRODUCTION

Denna reflektionsrapport har upprättats av utvecklingsteamet Wale Wale inom ramen för kursen *DAT255 - Software Engineering Project*. Utvecklingsteamet, vidare enbart benämnt teamet, har utvecklat en mobilapplikation med syftet att underlätta och effektivisera informationsutbytet mellan olika aktörer vilka är inblandade i hamnanlöp till Göteborgs hamn. teamet ämnar genom denna rapport påvisa att vissa insikter under arbetet har nåtts om den komplexitet som är förknippad med design och utveckling av mjukvara, och därigenom uppfylla ett av kursens lärmål.

Teamet består av Julia Gustafsson, Eric Josefsson, Max Edman, Mattias Lundell, Fredrik Ingelsson och Victor Persson. Utöver teamet benämns även produktägaren, fortsättningsvis kallad PO, vid ett flertal tillfällen i rapporten och då åsyftas i första hand Sandra, Mathias och Anders från PortCDM.

Efter denna inledning beskrivs i kapitel 2 *Application of Scrum* hur teamet tillämpat och använt Scrum-ramverket som hjälpmedel för att genomföra ett mjukvaruprojekt präglad av ett agilt förhållningssätt. Därefter, i kapitel 3 *Reflection and Evaluation*, görs reflektioner med utgångspunkt i processen i allmänhet och kursens inlämningsuppgifter i synnerhet.

I rapportens kapitel 4 *Best practices for using new tools and technologies* redogörs för de teknikval utvecklingsteamet har gjort och därefter avslutas rapporten med kapitel 5 *Relating your own process to literature and guest lectures* där utvecklingsteamet relaterar den egna processen till erfarenheter som kommunicerats från Spotify samt 8 Dudes in a Garage under kursens gästföreläsningar samt Knibergs bok *Scrum and XP from the Trenches*.

2 APPLICATION OF SCRUM

I följande kapitel redogörs för hur teamet applicerat den agila utvecklingsmetodiken Scrum under projektets genomförandefas. Inledningsvis redogörs för rollfördelning, grupparbetet och det sociala kontraktet. Därefter behandlas de Scrum-koncept som tillämpats under projektet. Slutligen redogörs för hur tiden har fördelats mellan olika personer, roller och sysslor samt hur uppgifter brutits ner och estimerats.

2.1 ROLES, TEAMWORK AND THE SOCIAL CONTRACT

Under projektets uppstartsfas gjordes en initial rollfördelning för att säkerställa ett effektivt grupparbete. Genom rollfördelningen fick enskilda teammedlemmar väl avgränsade arbetsuppgifter vilket hindrar såväl "dubbelarbete" som att vissa arbetsuppgifter "faller mellan stolarna" och lägger grunden för ett effektivt arbete. Vid rollfördelningen utsågs en Scrum Master med det huvudsakliga ansvaret för att säkerställa att Scrum-metodiken tillämpas och efterlevs under hela projektet. Därutöver tilldelades Scrum Mastern rollen som representant och primär kommunikationsväg för teamet gentemot andra teams och var följaktligen teamets representant vid Scrum of Scrums-möten vilka behandlas mer ingående under kapitel 2.2.6 *Scrum of Scrums*.

Vidare gjordes rollfördelningen utefter gruppmedlemmarnas enskilda kompetenser och intressen. Därigenom säkerställdes, så långt det var möjligt, att samtliga gruppmedlemmar ålades arbetsuppgifter i linje med deras respektive kompetens och intresse. Därmed ökade också förutsättningarna för effektiva arbetsinsatser. Rollfördelningen ledde till en hög grad av autonomi och egenansvar på individnivå. Tre av teamets medlemmar tilldelades huvudansvar för programmeringen där en medlem arbetade särskilt med backendfrågor, en annan med front end och en tredje med logiken däremellan. Teamets tre andra medlemmar ansvarade för dokumentation och övriga administrativa arbetsuppgifter. Dessa tre var följaktligen mindre delaktiga i programmeringen, men var med som "navigators" när parprogrammering tillämpades. Parprogrammeringen beskrivs närmare i kapitel 2.2.1 *Pair programming*.

Det sociala kontrakt som utarbetades innan projektets genomförandefas, fungerade som ett övergripande ramverk under projektets gång. I kontraktet specificerades teamets målsättning för att säkerställa och förankra att samtliga gruppmedlemmar hade samma syn på vad som skulle presteras inom ramen för projektet. Därutöver specificerades förväntad arbetsinsats, i tid mätt, på veckobasis. Slutligen innehöll kontraktet allmänna förhållningsregler och vilka konsekvenser som följer då dessa ej beaktas. Det sociala kontraktet och dess utformning möjliggjorde att gruppmedlemmarna under genomförandefasen kunde fokusera på det som var väsentligt, det vill säga arbetet i sig och inte ägna tid åt eventuella intresse- eller målkonflikter. Bara det faktum att teamet i början av kursen ägnade tid åt att tillsammans sitta ner och prata om förväntningar och förhållningssätt gentemot varandra lade en god grund för det fortsatta arbetet, oavsett vad som formulerats i kontraktet.

2.2 USED PRACTICES

Inom ramen för detta delkapitel redogörs för vilka specifika delar av Scrum som har tillämpats, på vilket sätt de har tillämpats samt vad nyttan varit med respektive tillämpning.

2.2.1 PAIR-PROGRAMMING

Parprogrammering har tillämpats i varierande omfattning genom hela projektet. Som tidigare nämnts tilldelades tre medlemmar huvudansvar för programmeringsarbetet och dessa hade rollen som "driver" genom hela projektet. De övriga tre, vilka enligt rollfördelningen ansvarade för dokumentation m.m., användes följaktligen som "navigators". Kodning och synkning gentemot Github gjordes i det närmsta uteslutande från "drivers" datorer.

Denna typ av parprogrammering föll ut väldigt väl då "drivers" snabbt blev mycket insatta i detaljer medan "navigators" hade en mer övergripande bild, vilket underlättade vid felsökning och dokumentation. Alla teammedlemmar hade samma roller, det vill säga "driver" respektive "navigator" genom hela projektet, vilket gjorde att spetskompetens inom sitt specifika område utvecklades. Det bidrog till ett effektivt arbete men på bekostnad av att teammedlemmarna fick en något "smalare" kunskap och teamet insåg snart att det förelåg en trade-off mellan kunskapsutbyte och effektivitet. Teamet var dock överens om att effektivitet var nödvändigt för att säkerställa att arbetet gick framåt i önskad takt. Om kursmålen var utformade på ett annat sätt, exempelvis om en viss nivå av programmering var tvungen att uppnås, hade sannolikt ett annat tillvägagångssätt tillämpats.

2.2.2 DAILY SCRUM

Eftersom samtliga gruppmedlemmar under projektets första hälft befann sig i slutfasen på kandidatarbetet med olika kalendrar uppstod svårigheter med att arrangera fysiska möten på daglig basis. Samtidigt insåg teamet vikten av denna typ av lägesrapportering eftersom grupparbetet, som tidigare nämnts, präglats av hög grad av autonomi vilket leder till ett stort behov av koordinering. Teamet enades därför om att tillämpa en variant av Daily Scrum. Genom att använda Slack och den privata gruppen *dailyscrums*, besvarades varje kväll nedanstående frågor av samtliga teammedlemmar:

- i. Vad har jag gjort idag?
- ii. Vad har jag tänkt att göra imorgon?
- iii. Vad behöver jag hjälp med?

Tack vare dessa dagliga uppdateringar hade teamet hela tiden en gemensam lägesbild och såg hur arbetet fortskred. Genom denna lösning erhöles fördelarna med Daily Scrum-konceptet trots att fysiska möten inte kunde arrangeras på daglig basis. En nackdel med denna variant av Daily Scrum var att saker som togs upp av enskilda teammedlemmar under fråga *iii* inte lika naturligt hanteras och följas upp i direkt anslutning till mötet, något som sannolikt hade underlättats om fysiska möten hölls.

2.2.3 SPRINT PLANNING, REVIEWS AND RETROSPECTIVES

I enlighet med Scrum-metodiken har teamet tillämpat koncepten för Sprint Planning, Sprint Reviews och Sprint Retrospectives genom hela projektet. Genom tillämpning av dessa koncept har projektet genomgående följt en tydlig struktur samtidigt som teamet levt upp till lärmålet angående planering och genomförande av ett mindre, teambaserat mjukvaruprojekt.

Eftersom projektets genomförandefas bestod av fem veckor delades den också in i fem sprintar. Varje sprint inleddes respektive avslutades på onsdagar i samband med handledningen på Lindholmen Open Arena. Nedan redogörs kortfattat hur koncepten Sprint Planning, Sprint Reviews och Sprint Retrospectives under projektet konkret har tillämpats.

2.2.3.1 Sprint Planning

Inför varje sprint gjordes en planering med utgångspunkt i den befintliga produktbackloggen, det vill säga utifrån de User Stories och Epics som ännu ej var avklarade. Utifrån produktbackloggen prioriterades sedan vilka User Stories som skulle flyttas till veckans Sprint Backlog för åtgärd i den kommande sprinten. Teamet strävade efter att göra prioriteringen och sprintplaneringen tillsammans med produktägaren i så stor utsträckning som möjligt, men vid vissa tillfällen prioriterade teamet själva utifrån det givna scenariot samt de minimikrav som specificerats.

Såhär i efterhand kan teamet konstatera att en tydligare involvering av PO vid sprintplaneringen hade varit önskvärt, då teamet under några sprintar upplevde vissa svårigheter att avgöra vilka User Stories som var att prioritera. Den ibland låga graden av involvering av PO i sprintplaneringen berodde dock snarare på teamet än på PO. Teamet borde explicit bitt PO att välja ut de User Stories som var att prioritera relativt de andra. Samtidigt uppstår då en del svårigheter när dessa sedan skall ställas mot de User Stories som upprättats internt för att skapa värde för teamet. Oavsett så har teamet i efterhand insett vikten av att göra ordentliga och tydliga sprintplaneringar, i syfte att etablera en tydlig och gemensam målbild inför varje sprint och därigenom lägga grunden för en effektiv arbetsprocess.

2.2.3.2 Sprint Review

Varje sprint avslutades med att teamet presenterade de framsteg, eller inkrement, som producerats under sprinten för PO. Denne avgjorde därefter huruvida inkrementen var att betrakta som godkända eller om det behövde göras korrigeringar för att nå en godkänd nivå. Utfallet, det vill säga antalet av produktägaren godkända inkrement, dokumenterades därefter tillsammans med det totala antalet producerade inkrement i Excel och ligger till grund för en av teamets KPI:er. Nedan tabell illustrerar schematiskt hur denna dokumentation såg ut, för fullständig dokumentation se bilaga 1 *KPI:er*. För vidare reflektion av Sprint Reviews, se kapitel 3.2 *Reflection on the Sprint Reviews*.

sprint	producerade	godkända	utfall
x	y	z	z / y

Tabell 1 Schematisk illustration över andelen godkända inkrement per sprint. Talet y erhålls genom att summera story points för alla under sprinten producerade User Stories. Talet z erhålls genom att summera story points för alla godkända vid Sprint Review'n godkända User Stories.

2.2.3.3 Sprint Retrospective

Efter det att veckans Sprint Review var avslutad genomfördes en utvärdering av den precis avslutade sprinten, en Sprint Retrospective. Tanken var att vid denna utvärdering fokusera på processen och reflektera kring vad som gått bra och mindre bra. Utifrån detta enades teamet om ett antal förbättringsåtgärder att ta med sig in i nästa sprint. Under denna utvärdering följdes också teamet KPI:er avseende psykosocial miljö och upplevd arbetsbelastning upp. Samtliga Sprint Retrospectives har dokumenterats, enligt formatet i tabell 1 nedan, och återfinns i sin helhet i bilaga 2 *Sprint Retrospective Documentation*. För vidare reflektion av Sprint Retrospectives, se kapitel 3.1 *Reflection on the Sprint Retrospectives*.

Sprint	Vad gick bra? (Keep doing)	Vad gick mindre bra? (Stop doing)	Förslag till förbättring (Try doing)
#X	Synpunkt / Reflektion	Synpunkt / Reflektion	Förslag till åtgärd

Tabell 2 Schematisk illustration över hur Sprint Retrospectives har dokumenterats.

2.2.4 PRODUCT BACKLOG

Den initiala produktbackloggen utarbetades av teamet utifrån det scenario som publicerades på kurssidan den 3 april. Genom att kartlägga de aktiviteter i vilka kaptenen var inblandad, antingen i en aktiv eller passiv roll, skapades en grundläggande förståelse för hur aktörens behov såg ut. Därigenom fick teamet också övergripande bild av vilken funktionalitet som behövde finnas för att skapa värde för aktören. Under denna process förbättrade teammedlemmarna förmågan att identifiera och klarlägga vilka behov och krav som förelåg utifrån den verklighetsbaserade domänbeskrivningen samt vilken design som krävdes för att säkerställa att identifierade krav och behov tillgodosågs i så stor utsträckning som möjligt. Teamets arbete med att utforma den initiala produktbackloggen anknyter därför tydligt till ett av kursens lärmål, nämligen det att klarlägga krav och där utefter designa en lösning givet ett verkligt problem.

Produktbackloggen var dynamisk genom hela projektet och förändringar gjordes på veckobasis då teamet träffade produktägaren. Vid den första sprintplaneringen klargjordes en del saker vilka inte gick att utläsa från scenariot, nämligen att kaptenen behöver en mobil lösning och helst med notisfunktion. Därmed ändrades teamets ursprungliga tanke som var en webbapplikation, till att istället utveckla en mobilapplikation. Vidare kom produktbackloggen att förändras vid varje sprintplanering eftersom produktägaren såg nya möjligheter att få med allt mer funktionalitet i takt med att applikationen utvecklades.

När teamet avslutade projektet fanns en del User Stories kvar i produktbackloggen som inte implementerats, det beror i första hand på att denna funktionalitet har prioriterats lägre relativt den funktionalitet som faktiskt har implementerats. Mer om dessa User Stories går att läsa i kapitel 4 *Reflection on the relationship between prototype, process and stakeholder value*.

2.2.5 SPRINT BACKLOG

Till skillnad mot produktbackloggen bör sprintbackloggen vara statisk, åtminstone under sprintarna. Detta visade sig vara problematiskt och något som teamet inte alltid lyckades med. Sprintbackloggens innehåll kom ibland att förändras under sprintarna då vissa User Stories byttes ut mot andra till exempel. Anledningarna till detta är sannolikt flera. Svårigheter att tidsestimera enskilda User Stories, oklar prioriteringsordning och en underskattning av vikten

av att ha "Tech Stories", alltså för teamet värdeskapande User Stories, i projektets inledande fas är några sådana anledningar.

2.2.6 SCRUM OF SCRUMS

Eftersom teamets projekt endast var en del av ett större projekt fanns ett behov av koordinering gentemot de andra teamen. Genom att tillämpa Scrum of Scrums kunde denna koordinering åstadkommas på ett praktiskt och effektivt sätt.

Genom Scrum of Scrums kunde teamet som helhet få en lägesbild över hur projektet som helhet framskred och ta del av problem och insikter från andra grupper. Därigenom kunde olika team hjälpa varandra för att undvika att "uppfinna hjulet på nytt". Såhär i efterhand kan teamet konstatera att den största nyttan med koordineringen gentemot de andra teamen faktiskt var att få höra hur de gjorde framsteg, eftersom detta faktum gjorde att det arbete teamet självt utförde kom att kännas allt mer värdefullt.

2.3 TIME DISTRIBUTION

Eftersom teamet, i det sociala kontraktet, enats om hur mycket tid som på individnivå skall spenderas på projektet rådde inga oklarheter kring det och följaktligen har teamets medlemmar spenderat ungefär lika mycket tid på projektet. Däremot skiljde det sig i hur mycket enskilda teammedlemmar lyckades producera per tidsenhet vilket till stor del berodde på att vi alla hade olika förkunskaper och tidigare erfarenheter. Oaktat det, lade samtliga gruppmedlemmar ner den tid som ansågs nödvändig för att uppfylla teamets högt ställda målsättning. Teamet anser även att tiden distribuerades väl över samtliga medlemmar under projektets gång. De teammedlemmar vilka tilldelats huvudansvaret för dokumentation och administration var som tidigare nämnts behjälpliga som "navigators" när arbetsuppgifter inom deras primära ansvarsområde inte förelåg.

Vidare har tiden distribuerats olika över sprintarna. Team Velocity sattes till 60 och följaktligen förväntades totalt 300 story points produceras under fem sprintar. Dock var samtliga teammedlemmar medvetna om att antalet producerade story points under sprint 2 och 3 sannolikt skulle understiga 60, eftersom samtliga teammedlemmar under dessa sprintar befann sig i en intensiv fas med sina respektive kandidatarbeten. Under sprint 4 och 5 däremot, kunde fler story points produceras eftersom väsentligt mindre tid behövde ägnas åt kandidatarbetet och teamet kunde fokusera på projektet i mycket högre utsträckning.

Ett annat alternativ teamet stod inför var att dela upp projektet i fyra sprintar istället för fem. I ett sådant fall hade teammedlemmarna helhjärtat kunnat fokusera på kandidatarbetet under en vecka och sprint 3 hade således utgått. Teamet såg dock risker med att fullständigt släppa projektet under en hel vecka. Dessutom såg teamet värdet i att hela tiden presentera ny funktionalitet för PO och på så sätt få ny viktig input från densamme för att föra processen vidare.

Slutligen har projekttriangeln och dess innebörd varit vägledande när teamet, vid vissa tillfällen, upplevt att tiden varit knapp. Teamet har då insett att tidsfaktorn är bortom vår möjlighet att påverka och när denna upplevdes som knapp gjordes istället kompromisser avseende scope, kvalitet eller en kombination av de båda.

2.4 EFFORT AND TASK BREAKDOWN

Teamet hade, framförallt inledningsvis i projektet, relativt svårt att bryta ner arbetet i User Stories av rimlig omfattning. Problemet var framförallt att de var alldeles för stora och omfattande. Detta ledde i sin tur till att teamet fann det svårt att estimerar effort eller story points för dessa User Stories. Dessutom krävdes därefter ett betydande arbete med att bryta ned omfattande User Stories i mindre tasks för att faktiskt kunna bearbeta dem. Vid detta laget var User Stories snarare en belastning än tillgång för teamet och betraktades närmast som onödigt arbete.

Teamets förmåga att utarbeta mindre och mer lätthanterliga User Stories blev dock bättre ju längre projektet fortgick och därmed ökade också träffsäkerheten i estimeringen av effort/story points. Detta uppnåddes främst genom att återgå till INVEST Criteria-modellen vilken gav en tydlig schematisk bild över vilka egenskaper som User Stories bör ha. Därigenom nådde teamet en punkt där utarbetade User Stories faktiskt fungerade som ett hjälpmedel.

En av anledningarna till att teamet initialt hade svårigheter att utforma användbara User Stories låg i svårigheterna med att kombinera vertikala User Stories med specialiserade programmeringsuppgifter. Eftersom teamet hade utarbetat en arbetsfördelning där medlemmarna i väldigt hög utsträckning arbetade horisontellt krävdes allt som oftast att flera teammedlemmar arbetade med samma User Story, vilket inledningsvis ledde till problem på grund av svårigheterna med att arrangera fysiska möten. I projektets slutskede däremot, hade samtliga teammedlemmar ett mer tillåtande schema vilket möjliggjorde att gemensamma programmerings-sessioner kunde hållas vilket underlättade tvärfunktionellt arbete.

Såhär i efterhand kan teamet konstatera att INVEST Criteria-modellen skulle tillämpats från start då den utgjorde en god grund för utformning av User Stories. Hur mycket bättre våra User Stories då hade blivit är naturligtvis omöjligt att svara på men förutsättningarna hade i ett sådant fall varit bättre. Samtidigt är det förståeligt att teamet i början av projektet upplevde svårigheter med att utarbeta User Stories eftersom tidigare erfarenheter av detta saknades, vilket ytterligare stärker tesen om att teamet borde fallit tillbaka till INVEST Criteria-modellen i ett tidigt skede. Detta för att underlätta nedbrytning av uppgifter och uppskatta effort av dessa.

3 REFLECTION AND EVALUATION

I detta kapitel görs en tillbakablick på projektet i allmänhet och processen och inlämningsuppgifterna i synnerhet. Mer specifikt görs reflektioner med utgångspunkt i ett resonemang om hur utfallet blev i relation till hur det kunde ha blivit om andra tillvägagångssätt tillämpats. Därigenom ämnar teamet genom detta kapitel uppfylla lärmålet om att reflektera över de, i projektet, använda metoderna för mjukvaruutveckling.

3.1 REFLECTION ON THE SPRINT RETROSPECTIVES

Som nämnts tidigare har Sprint Retrospectives genomförts varje onsdag, efter innevarande sprints Sprint Review och innan nästkommande sprints Sprint Planning. Syftet var att låta teammedlemmarna reflektera över vad som har gått bra respektive mindre bra under den gångna sprinten. Därefter utformades sedan ett antal lösningsförslag för att överbrygga gapet mellan den faktiska situationen och målsituationen.

Teamet gjorde ett aktivt val att inte engagera PO i genomförandet av retrospectives för att skapa ett tydligare fokus på processen framför prototypen. Trots detta fanns ibland vissa svårigheter att fokusera på rätt saker och teamet hamnade inte sällan i diskussioner om programmeringstekniska detaljfrågor. Om projektet fortsatt hade vi försökt att byta fysisk plats för genomförandet av retrospectives och lagt datorerna åt sidan för att skapa ett tydligare fokus.

Trots vissa svårigheter att fokusera på "rätt saker" fungerade retrospectives som ett utmärkt sätt att ventilerat synpunkter såväl positiva som negativa. Inledningsvis kände en del teammedlemmar att vissa synpunkter var för triviala att ta upp under retrospectives. Detta visade sig dock vara en helt obefogad känsla då samma synpunkter eller tankar ofta delades av andra teammedlemmar och kunde följaktligen åtgärdas av teamet. Sådär i efterhand kan teamet verkligen se värdet av genomföra retrospectives. Genom denna stund för reflektion identifierades ofördelaktiga beteenden och tillvägagångssätt vilka sedan kunde korrigeras för att säkerställa en effektiv arbetsprocess och för att reducera eventuella gap mellan utfall och målbild.

Erfarenheterna från teamets retrospectives har tydliga likheter med den Kata/Pussel-övning som gjordes under andra läsveckan och då specifikt reflektionsdelarna. Genom att utgå ifrån frågorna om vad som har hänt, vilka lärdomar som dragits, vilka som hinder som finns på vägen kunde teamet på ett naturligt sätt identifiera och testa nya inkrementellt förbättrade arbetssätt.

Samtliga Sprint Retrospectives har dokumenterats och återfinns i sin helhet i bilaga 2 *Sprint Retrospective Documentation*.

3.2 REFLECTION ON THE SPRINT REVIEWS

Precis som för retrospectives hölls varje onsdag Sprint Reviews där teamet fick chansen att demonstrera den funktionalitet som utvecklats under den gångna sprinten. PO var vid samtliga tillfällen med under dessa tillfällen och bedömde ny funktionalitet.

Det faktum att teamet fick möjligheten att träffa PO vid varje Sprint Review gjorde dessa väldigt betydelsefulla ur ett processperspektiv. Ett tillfälle var särskilt värdefullt då PO representerades av den aktör som teamet faktiskt utvecklade mot, nämligen kaptenen. Teamet fick därigenom viktig bekräftelse om att fortsätta på samma spår eller korrigera vissa saker som implementerats under den gångna veckan. Dock upplevde teamet att PO oftast hade svårt att ta ställning till huruvida producerade User Stories var godkända eller inte, vilket skulle kunna bero på att det där och då är svårt för PO att sätta sig in i en mängd olika User Stories och dess acceptanskriterier. Oavsett anledning godkändes, allt som oftast, de User Stories som teamet producerat under sprinten.

Interaktionen med PO under Sprint Reviews var även ibland svår att särskilja från en sprintplanering. Dessa måste naturligtvis inte göras sekventiellt men om de genomförs "samtidigt" krävs en mer strukturerad approach än den teamet tillämpade. Mer precist borde teamet upprättat en agenda att gå igenom tillsammans med PO vid dessa tillfällen. Därigenom elimineras risken att missa något väsentligt, vilket inträffade vid några tillfällen. Inför sprint 4 exempelvis, glömdes det av att göra en tydlig prioriteringsordning vilket vållade en del problem under kommande sprint.

Demonstrationen av ny funktionalitet gentemot PO resulterade i feedback. Precis som i Kata/Pussel-övningen var erhållen feedback grundläggande för fortsatt förbättring och utveckling. Genom Sprint Reviews fick teamet i praktiken erfara hur extern part kan uppleva, och utifrån sitt perspektiv värdera, gapet mellan utfallet och målbild, vilket har utgjort essentiell feedback i teamets arbete med att reducera detta gap.

3.3 REFLECTION ON THE RELATIONSHIP BETWEEN PROTOTYPE, PROCESS AND STAKEHOLDER VALUE

Under arbetets gång har många prioriteringar behövts göras, den funktionalitet som PO har värderat högst är den som teamet har lagt mest energi på. Då detta projekt är tidsbegränsat har tyvärr inte teamet hunnit implementera all önskvärd funktionalitet. Nedan följer en ingående redogörelse för teamets prototyp och hur den förhåller sig till de önskemål som PO har presenterat. Även funktionalitet samt designelement som teamet själva har önskat implementera finns redovisat nedan.

Innan detaljer presenteras bör teamets bild av projektet redogöras för, detta för att lättare kunna relatera till varför teamet lagt fokus på vissa saker och varför teamet har prioriterat på ett visst sätt. Den applikation som teamet har tagit fram en prototyp till ämnar underlätta kommunikationen vid ett hamnanlöp, denna kommunikation sker i dagsläget via mail. Teamets framtagna lösning bör alltså kunna genomföra denna typ av kommunikation på ett smidigare och mer intuitivt sätt än mail för att addera värde till intressenten. Om vår prototyp enbart fungerar som ett kommunikationsmedel till aktörer kan en kapten lika väl fortsätta använda mail. Med denna insikt reflekterade teamet mycket i projektets inledande fas kring hur applikationen bör vara utformad för att addera värde. Denna förståelse för intressenten och dess krav knyter även an till ett av kursens lärmål.

Kommunikationsfunktionaliteten har i detta projekt varit en så kallad Must-have. När sedan möjligheten till kommunikation var implementerad riktades all fokus på att strömlinjeforma upplevelsen för att addera värde. I just detta projekt har detta värde skapats genom att göra kommunikationen smidig och intuitiv samtidigt som information om anlöpet presenterats för kaptenen på ett tillfredställande sätt. Detta är viktigt för att förstå var teamet har identifierat gapen mellan sin prototyp och de önskemål produktägaren har presenterat. Hur informationen presenteras i applikationen har teamet lagt stor vikt vid. Teamets värdeerbjudande kretsar kring att leverera ett kommunikationsmedel som kan konkurrera med mail. Det föll sig då naturligt att presentera information på ett överskådligt sätt, något som mail misslyckas att göra. Teamet anser att prototypen uppfyller den eftersökta kommunikationsfunktionaliteten och är väldigt nöjda med det prototypen erbjuder. Teamet har däremot identifierat en rad gap mellan prototypen samt intressentens önskemål. Dessa gap berör mest användarvänlighet men även viss ytterligare funktion hade varit fördelaktig. Hade tiden tillåtit hade dessa gap glädjigen byggts av teamet.

Det första gapet är ett smidigare sätt att bekräftade föreslagna tider skickade av andra aktörer. I dagsläget behöver kapten skicka en ny ETA för att på så sätt bekräfta den föreslagna tiden, det hade istället varit önskvärt om det fanns en funktion som gjorde att en föreslagen tid kunde bekräftas med ett knapptryck. Vidare ville teamet implementera en funktion i applikationen som påminner kaptenen när vissa tidsstämplar rimligtvis bör skickas in, dessa påminnelser ska öka i intensitet ju närmare destinationen fartyget befinner sig. Denna funktionalitet prioriterades inte lika högt och fick tyvärr prioriteras bort i projektets slutskede.

För att en kapten skall hålla sig uppdaterad rörande relevant information ville teamet implementera notiser i applikationen. Något som teamet även lyckades med men tyvärr aldrig hann få fullt funktionella. Notiser dyker upp i notiscentret men är ej klickbara, detta beror på hur teamet i dagsläget initierar applikationens tråd. Då tråden startas i samband med att applikationens main-klass exekveras leder det till att parallella trådar startas om en användare klickar på en notis ur notiscentret. Detta upptäckte teamet för sent in i projektet och det var då för sent att ändra. Vidare på samma spår ville teamet implementera notisikoner vid de olika menyerna inuti applikationen så att användaren snabbt ser var det skett uppdateringar. Något som i dagsläget inte existerar i applikationen.

I dagsläget presenteras information i listor där typ, tid och avsedd plats finns angiven. Dessa vyer går tyvärr ej att sortera, något som hade varit önskvärt för att snabbt nå den information som önskas. Här har teamet även identifierat värdet av en färgkodning, där olika typer av tidsstämplar får olika färger för att snabbt kunna identifieras av användaren. Vidare lagras samtliga tidsstämplar som tillhör ett PortCallID, när en ny tidsstämpel av samma typ inkommer så behålls även dem gamla i samma lista. Detta gör att listan kan innehålla tidsstämplar som inte längre är aktuella. De senaste tidsstämplarna finns längst upp i listan men för en ökad tydlighet anser teamet att detta borde presenteras på ett mer intuitivt sätt. Exempelvis genom att de utdaterade stämplarna tas bort eller att de går att nå genom att klicka på den aktuella tidsstämplens, som en form av historik.

Teamet ville även implementera en tidslinje på applikationens hemskärm som skulle fylla syftet att på ett överskådligt och enkelt sätt. Detta skulle addera enormt värde till intressenten och särskilja prototypen från nuvarande alternativ. Denna tidslinje skulle automatiskt nyttja rapporterade ETA & AT tidsstämplar för att presentera de tider som blivit bekräftade och belysa vilka som ännu saknas.

Avslutningsvis har teamet identifierat funktioner som borde implementerats men som ej har varit möjligt på grund av begränsningar i kursens upplägg. Dessa inkluderar: Möjligheten att logga in mot en annan hamn än Göteborg, lagra historik över tidigare anlöp till andra hamnar, nyttja den implementerade funktionen "Reported By" samt att göra applikationen mer säker. I dagsläget är hamnen Göteborg hårdkodad i applikationen, möjligheten att byta hamn är givetvis önskvärd och skulle även öppna upp möjligheten för att föra historik över tidigare hamnanlöp. Detta skulle underlätta för kaptener som är nya eller när de gör anlöp i en hamn de tidigare inte besökt. Att kunna nyttja "Reported By" skulle göra det möjligt för en kapten att härleda vem som skickat ett meddelande för att kunna följa upp vid eventuella oklarheter. Avslutningsvis är säkerhetsaspekten av yttersta vikt men något som kursen har avgränsat sig ifrån. I dagsläget loggar en användare in enbart med IMO, lösenord krävs alltså ej.

Förutom de fyra senaste nämnda gapen mellan prototyp och den lösning intressenten önskar så anser teamet att de har nyttjat en process som stänger detta gapet. Det som har identifierats som begränsande faktorn har varit tiden och som tidigare nämnt har prioritering behövs göras i stor utsträckning för att leverera bästa möjliga funktionalitet efter förutsättningarna. Det som teamet har identifierat som nyckeln i sin process är en närhet till intressenterna för att identifiera de problem som bör lösas istället för de som presenteras i ett första skede. När väl rätt problem har identifierats har goda prioriteringar varit nyckeln till att teamets tid och energi läggs på det som genererar mest värde.

Något som har uppmärksammats under projektets gång är att kontinuerligt uppdatera varandra om sina förehavanden bidrar mycket till projektets framfart. Detta då det är lätt att snöa in sig på en speciell funktion som kanske inte genererar ett värde som motsvarar arbetsinsatsen. Med kontinuerliga avstämningar blir det lättare för teamet att identifiera arbetsuppgifter som är av lägre prioritet genom att tillsammans diskutera dess signifikans för intressenten. Utöver detta har par-programmering visat sig vara gynnsamt då även det medför en godare helhetsbild genom hela teamet. Avslutningsvis bör understrykas att teamet är nöjd över den process som anammats och skulle inte göra några drastiska ändringar om mer tid skulle skänkas till ändamålet.

3.4 EVALUATION OF D1-D4

Nedan följer teamets erfarenheter och lärdomar som har dragits ur kursens fyra inlämningsuppgifter. Varje inlämning behandlas separat och teamet redogör för hur dessa har påverkat arbetet och vad som kunnat förbättras.

3.4.1 D1 - SCRUM STRATEGIES

Teamets tankar kring det sociala kontraktet går att återfinna i kapitel 2.1, nedan följer en redogörelse för teamets tankar kring Lego-övningen. Att få en introduktion till att arbeta agilt var något alla i teamet uppskattade då det var ett nytt arbetssätt för samtliga. Utöver en grundläggande förståelse för hur arbetet läggs upp inom detta arbetssätt nådde även teamet tre, något mer djupgående, insikter:

1. Det är av yttersta vikt att förstå grundproblemet, eftersom den efterfrågade lösningen inte nödvändigtvis alltid är den bästa.
2. Tydlig arbetsfördelning är en grundförutsättning för effektivt arbete.
3. Att vara konsekvent i sitt arbete och inte ändra för många parametrar åt gången.

Dessa insikter formulerades upp i 3 strategier som har legat till grund för teamets fortsatta arbete. Något som däremot blev tydligt under arbetets gång är att en Lego-övning inte fångar komplexiteten som grupparbetet senare innehöll. Teamet inser sin naivitet inför denna uppgift och har under arbetets gång fått ställa dessa insikter i relation till en mer komplex uppgift.

Den första av de tre var den viktigaste för teamet och har följt med oss under hela arbetet. Då arbetet lades upp på ett sådant sätt att teamet till en början jobbade utifrån en idé som bollades med PO hamnade teamet i lite av ett tunnelseende. Istället för att förstå hur POs situation kan underlättas av en digital lösning så var fokus på hur vår skiss och tanke kunde anpassas till PO. Denna strategi fick oss däremot att omvärdera vårt tankesätt och byta approach. Frågorna till PO blev mer av den karaktär att vi fick en god inblick i det sammanhang vår applikation skulle användas, det var sedan upp till oss som grupp att lösa de problem som PO i dagsläget erfarar.

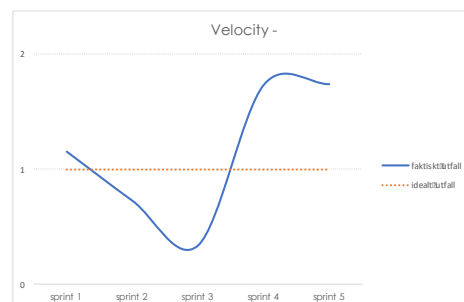
De övriga två strategierna formulerades utefter de relativt enkla förhållandena som Lego-uppgiften erbjöd, teamet insåg snabbt att projektet var av mer komplex karaktär. Trots detta mer komplexa sammanhang så har strategierna visat sig vara gynnsamma och den tydliga rollfördelningen som teamet anammat har lett till att projektet genomförts mer effektivt. Att arbeta konsekvent och enbart ändra några parametrar åt gången var dock en strategi som teamet fick justera en aning efter projektets mer komplexa förhållanden. Att inte göra några ändringar under en sprint som sträcker sig över en vecka gör att teamet har få tillfällen att revidera beslut. Då arbetet enbart sträckte sig över 5 sprintar medför att teamet enbart haft 4 tillfällen att justera sitt arbetssätt, något som tyvärr hade varit begränsande. Att däremot vara återhållsam med justeringar under sprintarna och noga dokumentera de ändringar som görs i arbetssätten var något teamet gynnades av.

3.4.2 D2 - KPIs

Teamet valde att mäta 5 stycken KPIer istället för de rekommenderade 3. Detta eftersom att teamet ansåg sig själv veta för lite om vilka parametrar som bör utvärderas för att på ett framgångsrikt sätt mäta teamets prestanda. Teamets tankar rörande valda KPI:er går att läsa under respektive underrubrik nedan, resultatet av undersökta KPI:er återfinns i bilaga 1 *KPI:er*.

3.4.2.1 Velocity

Teamet valde att sätta velocity till 60. Med tanke på teamets brist på tidigare erfarenhet av att jobba på detta sätt var det en uppskattning som efter arbetets gång var i behov av utvärdering. För att synkronisera med Release Burndown valde teamet att låta sin velocity vara 60 varje sprint under projektet. Däremot började teamets optimala velocity undersökas under projektets gång. Detta gjordes genom att nästkommande sprintens velocity beräknades genom att räkna ut medelvärdet av tidigare veckors godkända inkrement. Detta visualiseras nedan.



Graf 1 Illustrerar resultat av godkända inkrement dividerat med veckans velocity.

Om en siffra över 1 uppnås genom denna beräkning har teamet presterat mer än veckans velocity och den bör således höjas, om siffran å andra sidan blir under 1 är veckans velocity för hög. Den optimala siffran att uppnå genom denna beräkning är således 1, då överensstämmer godkända inkrement med veckans velocity. Denna metod kräver fler än 5 sprintar för att ge en god velocity för teamet att utgå ifrån, av den anledningen valde teamet att behålla 60 velocity under hela projektet.

Teamet har genomgående under arbetets gång haft svårt att estimera uppgifternas tidsåtgång. I arbetet att utvärdera om teamet arbetar i rätt takt under en vecka finns ett flertal okända variabler som påverkar utfallet och det har varit svårt att veta vilken som har varit boven i dramat då fel estimeringar har gjorts. Att vi som team uppskattat fel när det kommer till hur lång tid en uppgift tar är förståeligt då vi har utfört uppgifter vi aldrig tidigare gjort. Att estimera tidsåtgången har således blivit en chansning, en insikt som har lett teamet till att ifrågasätta om velocity hjälpt eller stjälpt teamet i sitt arbete. Teamet tror dock att velocity blir lättare med tiden och ser dess värde då alla blivit mer vana att arbeta med velocity och utför uppgifter som är bekanta.

3.4.2.2 Upplevd arbetsbelastning

Denna KPI har varit värdefull för teamet då det har blottlagt ojämn arbetsfördelning, detta har i sin tur lett till bättre fördelning av uppgifter och ansvar mellan teamets medlemmar. Teamet är dock ödmjuk inför det faktum att denna KPI använder sig av subjektiva värden som är svåra att jämföra sinsemellan.

3.4.2.3 Psykosocial miljö

Den psykosociala miljön har identifierats som högst relevant för teamets framgång. Med facit på hand har teamet dock upplevt det mer eller mindre överflödigt att mäta det i en KPI. Detta eftersom teamet nått en väldigt god förståelse för varandra och på ett ypperligt sätt lyckats samarbeta. Teamet inser dock dess värde i grupper som inte samarbetat med varandra tidigare, där den inre friktionen kan vara högre.

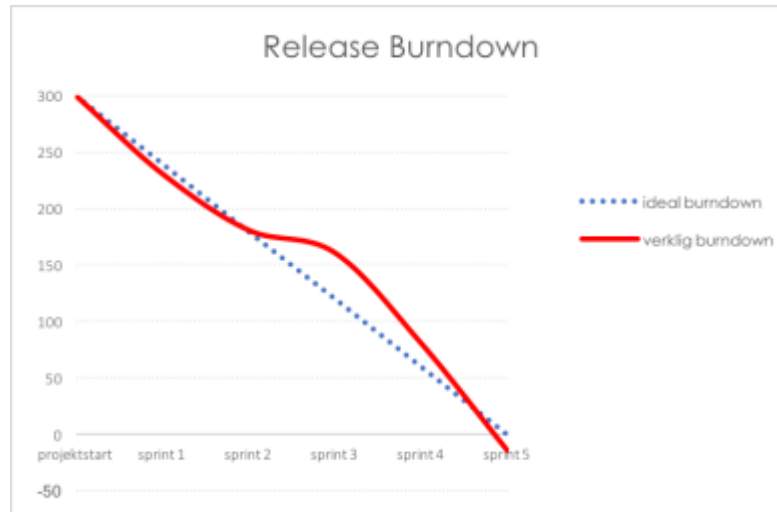
3.4.2.4 Godkända inkrement

Till en början ställde sig Teamet något skeptisk till att mäta enbart godkända inkrement. Detta eftersom att mycket av investerad tid kunde falla utanför vad som kunde redovisas och en vecka kunde återspeglas på ett sätt som inte motsvarade investerad tid. Under arbetets gång har dess värde infunnit sig hos teamet då detta leder till att arbetet fokuseras på PO och dess problembild, utan att lösa dess problem är arbetet mer eller mindre värdelöst oavsett hur mycket tid som har investerats. Att således lägga mycket tid på att förstå vilka kärnproblem PO har och som grupp göra sitt bästa för att lösa dessa har blivit otroligt viktigt. Detta är även något som återkopplar till den första strategin som formulerades efter den genomförda Lego-övningen.

3.4.2.5 Release Burndown

Detta mättes för att alla i teamet skulle ha en god uppfattning om projektets framfart i relation till dess omfattning. För att implementera release burndown på ett framgångsrikt sätt insåg teamet däremot att en djup och god förståelse var essentiell, något som saknades i projektets uppstartsfas. Teamet uppskattar dock att fokus lagts på en högre abstraktionsnivå med hjälp av release burndown. Den övergripande bilden av projektet har under arbetets gång agerat som vägvisare trots att estimaten inte alltid reflekterat den verkliga arbetsbördan.

En annan aspekt som avsaknaden av en fullgod förståelse medfört är ett kraftigt scope creep som användandet av release burndown har belyst, något som varit svårare att upptäcka om denna KPI inte valts. Skulle teamet genomföra en liknande uppgift idag skulle denna lärdom tillsammans med den insikt att försöka förstå POs problembild bidra mycket till en bättre implementering av release burndown.



Graf 2 Illustrerar utvecklingen av antalet godkända story points per sprint, ställt i relation till det ursprungliga antalet och avspeglar därmed också hur mycket tid som ägnats åt projektet under respektive sprint.

3.4.3 D3 - INITIAL PRODUCT BACKLOG & BUSINESS MODEL CANVAS

I följande kapitel redogörs först teamets tankar kring dess product backlog och hur den kunde förbättras. Avslutningsvis behandlas den business model canvas som teamet producerade vid projektets början. De två inlämningsuppgifterna behandlas under varsin rubrik nedan.

3.4.3.1 Product Backlog

Som nämndes ovan i samband med release burndown var teamets förståelse för problemet inte tillräckligt god vid utformningen av produkt backloggen, det resulterade i en backlog som inte helt korrekt återspeglade produkten som teamet gemensamt strävade mot. I samband med samtal med PO utformades nästa sprint utifrån förra veckans godkända inkrement tillsammans med feedback från PO. Teamet gjorde ansträngingar för att product backloggen skulle återspegla de insikter som samtalen med PO medfört men inser i efterhand att detta kunde skötas bättre. Teamet la vikt vid samtalen med PO och utformade därefter sitt arbete, något som i sig är positivt då det innebär att chansen för att inkrementen blir godkända ökar samt att det teamet levererar är det som PO efterfrågar.

Vid en första utformning av produkt backloggen hade teamet svårt att utforma vertikala user stories. Här var elefantövningen till stor hjälp och var som en ögonöppnare för teamet, efter denna övning blev teamet bättre på att utforma något som gav värde till PO. Detta fortsatte dock att vara en utmaning under arbetets gång, något som kan relateras till att det är ett annorlunda sätt att tänk jämfört med vad teamet är vana vid.

3.4.3.2 Business Model Canvas

Att nyttja en business model canvas (BMC) i detta sammanhang var något som teamet inte var vana vid. Detta eftersom en BMC vanligtvis används för att presentera hur ett helt företag ska lyckas vara lönsamt med hjälp av tydliga och enkla beskrivningar av dess kunderbjudande, kostnader & intäkter etc. Det var dock en nyttig övning att teamet satte sig ned och reflekterade över vad den färdiga produkten ska erbjuda, vilka aktörer som ett samarbete kommer krävas med samt vilka aktiviteter som bör genomföras för att leverera produkten. Att ha diskuterat dessa punkter innan arbetet startar bidrog till en gemensam bild av projektet inom teamet samt en bättre förståelse för sammanhanget. Att använda det inom en BMC förvirrade dock teamet en aning.



Figur 1 Business Model Canvas

3.4.4 D4 - HALF-TIME EVALUATION

När teamet författade sin halvtidsutvärdering var stämningen optimistisk med avseende på både process och produkt. De två första sprintarna var de mest problematiska, problemen avtog sedan i takt med att teamet lärde av sina misstag samtidigt som en godare förståelse för slutprodukt samt tillvägagångssätt erhöles. Sprint tre förflöt tämligen smärtfritt och det återspeglades även i teamets utvärdering. En utvärdering som utförs efter sprint två skulle enligt teamets mening vara mer framgångsrik i att belysa de olika bekymren som dyker upp vid en första implementering av ett agilt arbetssätt. Skulle den förläggas vid ovan föreslagna tidpunkt skulle det krävas mer reflektion internt i teamet för att utvärdera sin situation och således resultera i en mer nyanserad utvärdering.

Teamet förutspådde alltså att de största svårigheterna redan hade skett och att arbetet skulle fortskrida på ett mer behagligt sätt tack vare de insikter teamet fått och de processer som anammats. Så här med facit på hand kan teamet med stolthet bekräfta att andra halvan av arbetet flöt på lika smärtfritt som förutspått.

4 BEST PRACTICES FOR USING NEW TOOLS AND TECHNOLOGIES

För att kunna genomföra detta projekt har teamet nyttjat en rad olika verktyg, vissa av dem har teamet varit bekanta med medan andra har varit helt nya för teamet. De som teamet kommit i kontakt med tidigare gick relativt smärtfritt att implementera medan de nya, föga förvånande, krävde större insatser.

För att kommunicera intern användes kommunikationsprogrammet **Slack**, en mjukvara som teamet nyttjat tidigare och som anammades utan större bekymmer. Som tidigare nämnt användes även detta till Daily Scrum, något som visade sig vara väldigt gynnsamt för teamet och fortsatte användas under hela projektet.

Teamet valde **Trello** som sin Scrum board technology, detta för att några ur teamet använt det i tidigare projekt och fått en god uppfattning av dess funktionalitet och användarvänlighet. Till skillnad från Slack fanns en liten inlärningskurva till Trello, när väl samtliga medlemmar ur teamet kunde orientera sig inom Trello och sköta alla brickor och listor stötte teamet på utmaningen att på ett enkelt och tydligt sätt presentera sina Epics, User Stories och dylikt. Till en början blev det ganska rörigt hur teamet än angrep Trello. För att komma till bukt med denna problematik tillsattes en Trelloansvarig efter sprint 2. Ansvaret innefattade att säkerställa att samtliga brickor i Trello följde de regler som teamet kommit överens om samt att uppdatera brädet efter varje sprint. Teamet valde att färgkoda brickor efter typ, tillsätta samtliga user stories till teammedlemmar samt nyttja deadlines. Samtliga av dessa arbetssätt var gynnsamt och nyttjades under hela projektet. Något som teamet upplevde som problematiskt var hur pass mycket en uppgift bör brytas ned i User Stories, teamet experimenterade således med många små User stories likväl som färre fast större stories. Teamet landade i att lite större User stories som nyttjade den inbyggda funktionen med checklistor var att föredra, detta gjorde stories kunde angripas stegvis utan att förvirra användandet av Trello med allt för många brickor. Teamet kom till denna insikt lite sent i arbetet och hann tyvärr inte nyttja det speciellt länge.



Figur 2 Product Backlog i Trello

Som sin IDE valde teamet **Android Studio**, detta föll sig någorlunda naturligt efter teamet valde att skapa en mobilapplikation. Detta eftersom Android Studio tillhandahöll en markant användarvänlighet och tydlighet som uppskattades enormt då detta var ny mjukvara för teamet. I Android Studio är sedan applikationen uppbyggd av javaklasser, här har teamet dragit nytta av tidigare programmeringskurser och således uppfyllt ett av kursens lärmål.

För att direkt kunna se utfallet av kodningen användes ett plugin till Android Studio, en emulator. I denna emulator kunde sedan applikationen testas för att ge direkt feedback för fortsatt kodning.

För att sedan synka till Git användes **Github Desktop**, en mjukvara som även den är väldigt användarvänlig. Teamet insåg dock sent i projektets framfart att Android Studio har en inbyggd funktion för att synka, hade detta uppmärksammats tidigare hade den nyttjats parallellt med Github Desktop.

Teamet har valt att kommunicera direkt med det Rest API som finns på PORTCDMs server och nyttjar således ingen mellanhand. Denna kommunikation sker med http requests och för att på ett smidigare sätt testa och felsöka dessa förfrågningar har **Restlet Client** nyttjats. Att kommunicera direkt med PORTCDMs REST API visade sig vara smidigare än teamet först förutspått, att lära sig detta knyter även an till ett av kursens lärmål, nämligen att lära sig nya API:er.

Avslutningsvis föll valet naturligt på **Google Drive** för skrivande av rapporten. Möjligheten för flera personer att jobba i dokumentet samtidigt och kommentarsfunktionen gjorde detta till ett bättre alternativ än att jobba i ett dokument som sporadiskt synkas upp till Git. Då teamet använder Google Drive behöver rapporten sedan migreras till Word för att genomföra formatering av dokumentet, trots detta extra moment vägde Google Drive upp nackdelarna som ett synkat dokument på Git medför.

5 RELATING YOUR OWN PROCESS TO LITERATURE AND GUEST LECTURES

Teamets arbete har, i stora drag, liknat den process som Kniberg beskriver i boken *Scrum and XP from the Trenches*. Teamet håller med om det som författaren skriver, att Scrum inte är en metod utan snarare ett ramverk vilket har medfört att teamet tillämpat Scrum på sitt egna vis. Teamet har försökt att ta fasta på de saker som varit viktiga för att framgångsrikt driva projektet framåt, primärt inspirerade av kursens lärare men även av kursens litteratur i allmänhet och Knibergs bok i synnerhet. Speciellt viktig har litteraturen varit i att skapa en mer detaljerad förståelse för vissa koncept. Vilka saker som skulle kunna behandlas under en Sprint Retrospective eller för- respektive nackdelar med parprogrammering är ett par exempel på när litteraturen konsulterats. På det sättet har litteraturen utgjort ett bra komplement till kursens föreläsningar och teamet har genom att kombinera dessa, utvecklat förmågan att tillämpa ett agilt förhållningssätt under projektets genomförandefas.

Teamets arbete knyter även tydligt an till de vägledande principer som Spotify tillämpar, och som Simon redogjorde för, under en gästföreläsning. Att hela tiden utvärdera och förbättra hur vi arbetar och att inte vara rädd att förändra arbetssätten var centralt även i vår process precis som att värdera laget framför jaget. Vidare berörde Simon autonoma teams (squads) vilket även går att likna vårt sätt att arbeta. Våra "squads" var dock inte tvärfunktionella, vilket vid vissa tillfällen genererade problem. Samtidigt var antalet teammedlemmar i minsta laget för att sätta upp flera tvärfunktionella "squads" även om vi nu i efterhand definitivt ser vinsten av ett sådan uppdelning.

Ytterligare en viktig insikt gjordes under gästföreläsningen, nämligen vad som är viktigt när en idé/feature utvecklas. Mer specifikt insåg teamet vikten av att tidigt validera idén gentemot intressenterna och att fortsätta göra så under hela utvecklingsprocessen. Genom att tillämpa Scrum-ramverket blev denna tanke dock lätt att omsätta i praktiken genom möjligheten till veckovis återkommande Sprint Reviews där ny funktionalitet demonstrerades och således även validerades.

Vidare kan teamet även relatera den egna processen till saker som som Christian från 8 Dudes in a Garage belyste under sin gästföreläsning. Framförallt gäller detta Christians 10 egna erfarenheter, och mer specifikt den första:

"Your 'paper idea' is awesome... but you are wrong. Fail fast, adapt and kickass."

Detta var precis vad som hände inledningsvis när teamet utformade en idé för att adressera problemet, enbart utifrån det på förhand givna scenariot. Teamet hade då idén att bygga en webbapplikation eftersom det faktum att kaptenen önskade en mobil lösning med notisfunktion inte gick att utläsa från scenariot. Tacksamt nog failade teamet snabbt, närmare bestämt vid första sprintplaneringen, kom tillbaka upp på banan fort igen och levererade en kickass applikation.

BILAGA 1 KPI:ER

sprint	producerade	godkända	utfall
#1	69	69	1
#2	57	50	0,88
#3	20	20	1
#4	80	80	1
#5	120	95	0,79

TABELL 1 GODKÄNDA INKREMENT

	ideal burndown	verklig burndown
projektstart	300	300
sprint 1	240	231
sprint 2	180	181
sprint 3	120	161
sprint 4	60	81
sprint 5	0	-14

TABELL 2 RELEASE BURNDOWN

sprint	tidigare velocity	godkända	utfall
#1	60	69	115%
#2	69	50	72%
#3	59,5	20	34%
#4	46,33333333	80	173%
#5	54,75	95	174%

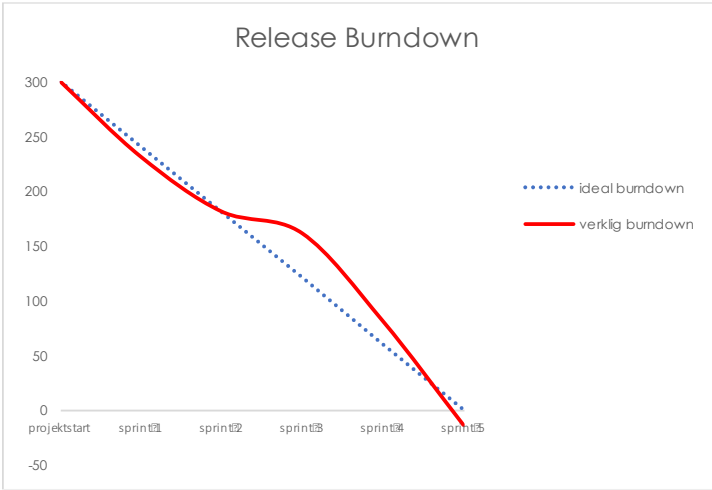
TABELL 3 VELOCITY

	#1	#2	#3	#4	#5
E	20%	50%	20%	70%	60%
G	40%	50%	10%	50%	70%
I	20%	30%	5%	30%	30%
J	50%	40%	40%	50%	70%
L	40%	50%	40%	50%	50%
P	40%	40%	10%	30%	40%
Medelvärde	35%	43%	21%	47%	53%
Maxvärde	50%	50%	40%	70%	70%

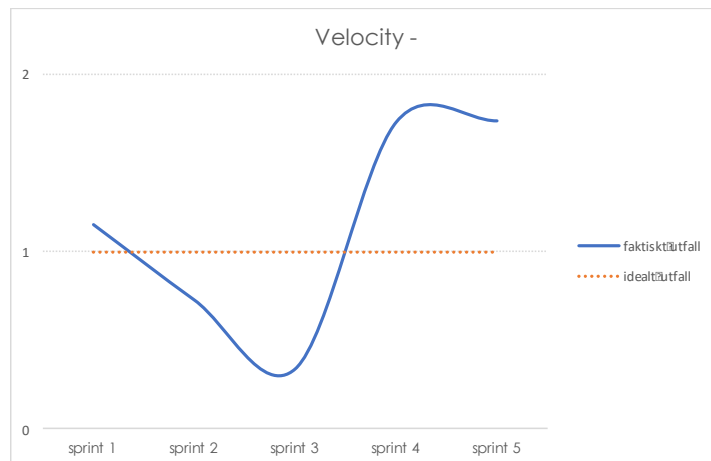
TABELL 4 UPPLEVD ARBETSBELASTNING

	Sprint 1						Sprint 2						Sprint 3					
	E	G	I	J	L	P	E	G	I	J	L	P	E	G	I	J	L	P
Hur glad är du över att vara en del av teamet?	100%	100%	100%	80%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Hur nöjd är du med dina arbetsuppgifter?	100%	70%	70%	80%	50%	50%	100%	80%	70%	90%	80%	50%	100%	100%	60%	100%	80%	50%
Hur värdefull känner du dig för teamet?	100%	40%	40%	60%	50%	50%	100%	50%	50%	40%	60%	50%	90%	10%	10%	70%	60%	50%
Hur stolt känner du dig för teamet?	100%	90%	100%	80%	100%	100%	100%	90%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Hur stolt känner du dig för det teamet levererar?	100%	100%	100%	90%	100%	100%	100%	100%	100%	100%	100%	90%	60%	50%	50%	70%	100%	60%
Medelvärde:	83%						87%						79%					
Minvärde:	40%						40%						10%					
	Sprint 4						Sprint 5											
	E	G	I	J	L	P	E	G	I	J	L	P						
Hur glad är du över att vara en del av teamet?	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%						
Hur nöjd är du med dina arbetsuppgifter?	100%	90%	90%	100%	100%	100%	100%	100%	100%	100%	100%	100%						
Hur värdefull känner du dig för teamet?	100%	75%	50%	80%	80%	60%	100%	65%	40%	50%	50%	50%						
Hur stolt känner du dig för teamet?	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%						
Hur stolt känner du dig för det teamet levererar?	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%						
Medelvärde:	94%						92%											
Minvärde:	50%						40%											

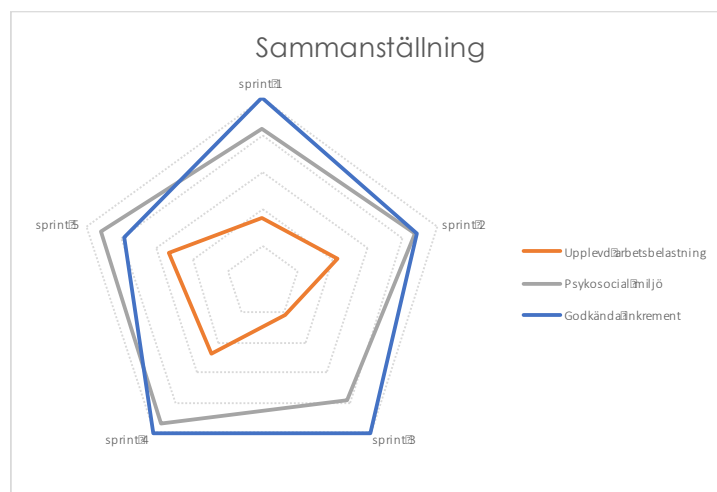
TABELL 5 PSYKOSOCIAL MILJÖ



GRAF 1 RELEASE BURNDOWN



GRAF 2 VELOCITY: RESULTATET AV GODKÄNDA INKREMENT DIVIDERAT MED VECKANS VELOCITY.



GRAF 3 SAMMANSTÄLLNING

BILAGA 2 SPRINT RETROSPECTIVE DOCUMENTATION

Sprint	Vad gick bra? (Keep doing)	Vad gick mindre bra? (Stop doing)	Förslag till förbättring (Try doing)
#1	Genom arbetsfördelningen kunde överlappande arbete undvikas	För övergripande målbild	Tydligare nedbrytning/user stories
	Tillämpning av Daily Scrum föll ut väl	Bristande synkning mellan programmeringsgrupper	Få till ett fysiskt måte under sprinten för koordinering
	Tydlig arbetsinriktning/målbild	Uppföljning / uppdaterat vad som är gjort och vad som skall göras	Föra tydligare loggbok samt utse Trello-ansvarig
	Agilt förhållningsätt under första sprintplaneringen	Svårigheter med att nå backend	Konsultera andra grupper vilka löst problemet
	Bra "progress"		
#2	Gjorda teknikval känns bra och ändamålsenliga	User Stories är belastning snarare än hjälpmedel	Kika på INVEST Criteria-modellen
	Bra insikter nåtts om backenden	Snöade in på "grooming-detaljer"	Konsultera övriga teammedlemmar som ej är insatta i detaljer för att undvika tunnelseende
	Bra samverkan med övriga grupper	Ostrukturerad Scrum board	Utse Trello-ansvarig
#3	Teamet lyckades leverera värde trots lite avsatt tid under sprinten	Tog lång tid att få svar från utvecklare på Slack	Tagga berörd/berörda utvecklare vid kommunikation via Slack
	Lyckades fånga upp och korrigera det som inte godkändes av PO vid sprint review #2	Sprint backloggen var inte tillräckligt genomarbetad	Lägga mer tid vid sprintplaneringen bl.a. för att säkerställa en ändamålsenlig sprint backlog
	Teamet lyckades fokusera på rätt saker	Delvis överlappande arbete	Tydligare arbetsuppdelning vid sprintplaneringen
#4	Samtliga programmeringsgrupper sitter tillsammans	Svårt att prioritera uppgifter	Att vid nästa sprintplanering enas med produktägaren om en tydlig prioriteringslista
	Sprintbackloggen fyllde funktionen som ett hjälpmedel väl	Frustration när appen "havererade"	Programmeringsgrupper informerar övriga redan innan vissa kritiska åtgärder vidtas
	Tvärfunktionellt arbete		