

Team Reflection

Team 7

Maj 2020

1 Customer Value and Scope

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

Scopet för vår MVP har hela tiden känts görbart, men vissa features insåg vi tidigt att vi inte skulle hinna med. Trots en något godtycklig velocity har i princip varje sprint haft en rimlig omfattning, och kunnat färdigställas på utsatt dag. Eftersom detta var den sista sprinten och därmed det sista tillfället att skapa värde för uppdragsgivaren var det viktigt att ta reda på vad de ville att fokus skulle ligga, och vad som skulle vara mest värdeskapande på den lilla tid som fanns kvar. Uppdragsgivarna var väldigt nöjda med hur hemsidan såg ut och fungerade på mötet inför veckans sprint, och deras konkreta önskemål var främst att alla sidor på webbplatsen skulle få ett enhetligt utseende.

Under sista sprintplaneringen la vi därför in tasks för att snygga till och få ett enhetligt utseende på hemsidan enligt önskemål, medan de återstående uppgifterna främst handlade om att förfina och förbättra vissa features, samt rensa upp och snygga till koden. Detta för att produkten som helhet skulle få ett större värde för uppdragsgivaren, men det skapade också ett värde för oss som utvecklare att lämna över kod utan massa varningar och som vi tycker ser bra ut. Något som vi dock gärna hade lagt till är en betalningsfunktion inbäddad i bokningssystemet, vilket vi var på väg att implementera i den sista sprinten men prioriterade bort när vi insåg att det skulle ta för stort utrymme från resterande task vi ville slutföra under veckan.

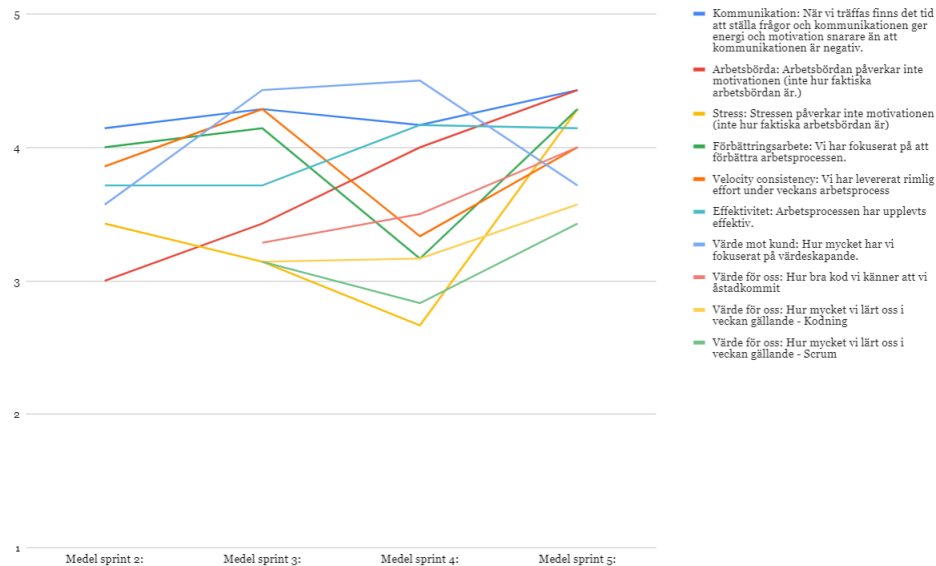
Denna vecka känns det som vi ligger ganska nära målbilden av hur vi vill att det ska fungera, sprinten blev klar i tid och vi hade en rimlig arbetsbörda vilket tyder på att vi lyckats estimerat och prioriterat rätt när vi valde att inte bygga några helt nya features sista veckan. Det hade dock varit önskvärt att få till en betalningsfunktion då detta var något som värderades högt i början av projektet.

Hade projektet fortsatt en vecka till hade nog prioriteringen av betalningsfunktionen varit annorlunda, då vi haft mer tid på oss att hinna med både en betalningsfunktion och de user stories vi prioriterade denna vecka. Vi hade också behövt läsa på om vilken betalningsfunktion som hade varit möjlig att integrera på kort tid, då vi redan insett att Swish inte skulle vara möjligt att bygga in då det behövdes ansökas om samt inte var gratis. En lösning för att inte behöva prioritera bort planerade aspekter i kommande projekt är att tidigare undersöka hur mycket arbete en sådan feature som integrerad betalning faktiskt innebär. Det var inte förrän vid de senaste sprintsen i projektet som vi undersökte huruvida det var genomförbart eller inte, och därför lät vi user storyn ligga kvar i sprint backloggen, vilket kanske gav felaktiga förhoppningar.

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

Vi har funnit det väldigt intressant att följa våra KPIer varje vecka och tycker att det varit värdeskapande för oss. Under varje sprint review har vi diskuterat resultatet och vad det kan bero på att det skiljer sig från föregående vecka. De veckor vi haft en kommande sprint har vi även stolpat upp mål för vad vi bör göra för att ändra ex. en nedåtgående trend.

Vad vi kunde se i grafen var att effektiviteten upplevts minska något, kanske eftersom vissa tog på sig tasks som de var osäkra på för att utmana sig själv. Detta kan leda till minskad effektivitet då det kan vara timmar man bara sitter och bankar huvudet i väggen. Det som förlorades i effektivitet kompenseras dock för med nya lärdomar. Bristen på motivation till följd av stressen har minskat drastiskt jämfört med föregående vecka. Troligtvis beroende på att halva teamet hade en stressig vecka med kandidatarbetet förra veckan. Potentiellt upplevdes därmed denna veckan som en lång utandning. Förbättringsarbete ökade jämfört med föregående sprint. Möjligen eftersom att vi la tid på att formulera acceptance criterias för alla user stories denna vecka.



Hade projektet fortlöpt hade det varit fortsatt intressant att följa upp med denna veckovisa enkät för att se fortsatt utveckling av projektet. Vi tycker det är oerhört intressant att titta tillbaka på den, försöka förstå potentiella dippar, och av det lära oss vad vi kunde ha gjort annorlunda i arbetet för att undvika dem. I framtida projekt känns det som ett bra verktyg att tillämpa ett liknande sätt att mäta arbetet på. Vi tror det kan vara än viktigare desto större team man är också eftersom det potentiellt är svårare att kommunicera med alla om gruppen är större.

Något som är lite synd är att vi inte fick koll på vad KPIer innebar förrän tills sprint 2, vilket innebär att vi förlorade första veckans reflektion kring dessa områden. Hade varit intressant att se värden ändra från start och det är något som vi tar med oss i framtida projekt, att vi nu vet vad det handlar om och hur man kan upprätta, samt följa upp sina KPIer på ett snabbt och enkelt sätt.

2 Social Contract and Effort

2.1 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)

Denna veckan känns det som om vi spenderat aningen mer tid på projektet än innan. Detta kan ha att göra med att det var sista veckan och vi ville komma fram med ett resultat vi kunde vara stolta över. Det var därmed en hel del småfix som behövde göras och saker som dykte upp under sprinten. Dessutom fick en del tid läggas på merge-konflikter och dylikt som uppkom eftersom det var många som behövde jobba i flertalet filer på samma gång. Det som dessutom tog extra tid denna vecka var att vi utöver veckans vanliga uppgifter skulle förbereda oss för presentationen, vilket kan ta tid om man vill åstadkomma något intressant och kvalitativt.

I framtida projekt kan det vara bra att strukturera vissa tasks på ett tydligare och bättre sätt så att det blir mindre småfix som behöver göras såhär i slutet. Exempelvis att vi under sista veckan kom på att alla olika pages bör ligga i en container så att de får ett enhetligt utseende gällande marginaler etc. Hade detta gjorts i början hade detta designproblem aldrig uppkommit och den tiden hade inte behövts läggas denna vecka i samma utsträckning.

För att lyckas få en bättre grundstruktur tänker vi oss att man ska vara lugnare i projektets början och verkligen låta detta ta tid. Att man inte ska skynda sig igång och vara för ivrig. Dock är det väsentligt inom agil programmering att komma med ett första utkast fort och utveckla det eftersom. Vad vi menar är snarare att man borde ha definierat globala containrar och kort exempelvis i ett tidigt skede. Det hade även gjort det enklare att ändra saker dynamiskt eftersom vi fick input av uppdragsgivare.

3 Design decisions and product structure

3.1 How you use and update your documentation throughout the sprints

Vi har dokumenterat det mesta vi gjort under projektet, till exempel finns protokoll för alla möten i helgrupp, möten med uppdragsgivaren och sprint reviews i separata dokument. Dessa är strukturerade i olika mappar i den gemensamma google drive som hela gruppen har tillgång till. Därigenom har det varit möjligt att hela tiden ha koll på vilka beslut som tagits och vad som diskuterats vid vilken tidpunkt. Mötesprotokollen från de möten vi haft med uppdragsgivaren har varit väldigt användbara. Vissa dokument, såsom sammanställningen av våra KPI:er och den funktionella checklistan som används för testning av koden har istället uppdaterats kontinuerligt. Sammanställningen av KPI:erna har används vid varje sprint review för att kunna diskuteras i gruppen. Checklistan har uppdaterats så fort en feature färdigställts och det finns en pull request på github. Att hålla detta dokument uppdaterat är således hela gruppens ansvar. Den digitala scrumboarden uppdateras också gemensamt i helgrupp under möten men det finns även ett ansvar att flytta färdigställda uppgifter från exempelvis "in progress" till "done" när en uppgift är klar och uppfyller den "definition of done" som finns uppsatt.

För att säkerställa att checklistan direkt blev uppdaterad vid färdigställandet av en feature skrevs denna uppgift in i definition of done. Trots det har det frångåtts i några fall, vilket kan bero på glömska eller viljan att bli klar med sin uppgift. Om projektet skulle fortgå en vecka till hade vi kanske behövt påminna varandra om vikten av att följa det arbetssätt vi satt upp, då vi redan gjort en åtgärd för att säkerställa detta.

Vi har ständigt strävat efter ett strukturerat arbetssätt, vilket visar sig bland annat med våra dagordningar. Detta är något som vi kommer att ta med oss till framtida projekt, vikten av struktur för att ett helt team ska kunna hålla ihop och vara med på samma spår. I kommande projekt kommer vi att ta med oss kunskapen vi fått gällande arbetsprocessen men absolut även försöka utveckla den. Bara för att detta sätt att dokumentera arbetet fungerat väl för det här projektet betyder inte det att det kommer att fungera i en annan kontext, exempelvis under ett längre/större projekt med fler involverade medlemmar.

En sista key-takeaway värd att nämna är att vi fortsättningsvis kommer att värdera användandet av en scrumboard. Eftersom att alla ansvarade för att flytta sina tasks kunde man som medlem hela tiden vara medveten om vad som hände i andra delar av projektet och om någon task legat i "in progress" under en lång tid var det enkelt att fråga den tillsatta personen om den var i behov av hjälp.

3.2 How you ensure code quality and enforce coding standards.

Den kodstandard vi från början kom överens om var att vi skulle ha själv-dokumenterande kod, att den inte skulle innehålla onödiga kommentarer eller stora mängder utkommenterad kod, samt vara formaterad. Utöver detta har vi även kontinuerligt genom projektet använt oss av pull-requests för att upprätthålla detta.

Förra reflektionen talade vi om att prova byta vilka som granskar varandras kod samt eventuellt lägga till ett krav på minst två personers godkännande för att få mergea sin branch med mastern. Detta prioriterades bort då majoriteten av sakerna som gjorts denna vecka har varit mindre än tidigare, vilket innebär att det har funnits fler pull requests än vanligt. Om vi dessutom bytt partners samt ställt ett högre krav på antal godkännande hade detta tagit för mycket tid. I ett framtida projekt hade det varit intressant att ställa högre krav på antalet personer som reviewar kod för att kunna garantera att kodstandarder upprätthålls. För att göra detta tror vi dock att det hade krävts att mer tid lades åt sidan för detta. Något som i sådana fall skulle behövas ta hänsyn till redan tidigt i projektet och som räknades in i varje sprint.

Den kodstandard vi kom överens om hade under processen kunnat förtydligas tidigare. Det kom upp frågor gällande mängd kommentarer, vilka komponenter som skulle vara funktions - eller klassbaserade och liknande under de sista två veckorna. Gällande kommentarer togs beslutet att vi inför sista veckan skulle försöka kommentera den kod vi skrev. Detta upplevde många som väldigt svårt då mycket av den koden som skrevs sista veckan var små saker i olika delar av koden, vilket upplevdes som svårt att kommentera. Samt som att det skulle vara en stor mängd arbete om vi skulle gå igenom gammal kod och kommentera den på ett relevant sätt. Något som i slutfasen av projektet inte skulle skapa särskilt mycket värde för någon, varken kund eller oss själva som utvecklare.

I kommande projekt hade det varit bra att ha en tydligare definierad kodstandard så att det från första början är tydligt hur olika saker hanteras på vad gruppen anser är 'bästa' sätt. Med mer kommentarer kan det förenkla för alla medverkande att vara införstådda i vad koden innebär och gör oavsett tidigare samarbete och kodkompetens. För att göra detta hade det varit bra att i början kunna spara exempel på situationer som representerar olika varianter kodstandard som gruppen sedan kan ta ställning till varefter en stil väljs ut. Därefter skulle en kunna gå bak och titta på konkreta exempel om en vore osäker i någon situation.

4 Application of Scrum

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

När vi började planera denna sprint ville vi prova att definiera en checklista för acceptance criterias på varje user story. Detta var något som vi gjorde under tidigare sprints, men som under projektets gång föll mellan stolarna när vi troligen tänkte att det var för tidskrävande och att alla ändå var införstådda i vad storyn innebar. Anledningen till att vi valde att prova detta igen var för att det uppdagades under förra sprinten att det ibland var svårt att veta om ens task var klar eller inte. Vi upptäckte att detta var ett värdefullt tillvägagångssätt och att det hjälpte oss att snabbare kunna definiera vilka tasks som behövdes under respektive user story, samt att det som utvecklare var enklare att veta när man var klar. Alltså uppfyllde detta våra förväntningar.

Att definiera acceptance criterias, och låta det momentet ta den tid det behövde, är något som vi kommer att ta med oss i framtiden. Det har gett tydligare specifikation av vad som faktiskt efterfrågas av varje utvecklare i sprinten, vilket är en positiv förbättring. För att säkerställa att denna lärdom tas med i framtida utvecklingsprojekt, kommer vi vara noga med att lyfta de fördelar som tillkommer med acceptance criterias ifall det kommer diskussion för att checklistorna skall genomföras. På så sätt ökar sannolikheten att acceptance criterias verkligen implementeras.

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

Kniberg (2015) redogör för vikten av scrum backloggen och hur viktiga de user stories som finns i backloggen är för scrum-metodiken. Han lyfter magkänslan som riktlinje vid val av aktuella user stories, till skillnad från det tillvägagångssätt för val av user stories vi främst använt under projektet. Vi valde de user stories som vi trott skulle ge uppdragsgivarna mest värde för de timmar vi lagt ned på att koda, vilket till stor del har fungerat bra. Vår magkänsla har säkert varit med i bakgrunden men mest fokus har varit på uppdragsgivarna. En sak som vi hade kunnat göra bättre utefter Knibergs (2015) rekommendationer är att se till att varje sprint har ett specifikt mål. Under projektets gång har vi inte riktigt haft den övergripande planen, dels på grund av otydligt specificerade ansvarsområden mellan studenterna, men också för att vi helt enkelt missat den aspekten. Inför kommande projekt ser vi ett stort värde i att specificera ett mål för varje sprint och kommer därför försöka implementera aspekten vid varje sprint planning eftersom det först är då lagmedlemmarna med säkerhet vet vad som skall göras under veckan.

I föreläsning 4 den 1 april diskuterades främst fenomenet "slicing the cake".

Det har mestadels gått bra, eftersom vi försökt använda INVEST-ramverket vid utstakandet av samtliga user stories. Vi försökte se till att våra User stories var vertikala tårtbitar, vilket har gjort det möjligt för oss att arbeta på flera user stories under samma sprint utan större merge-konflikter på github när utvecklad kod skulle samköras. Även för de tasks som definierats vid varje sprintplanering har vi haft ambitionen att följa SMART-kriterierna och göra dessa oberoende av varandra, men där har det varit svårare. Det har lett till att det stundtals inte riktigt gått att börja med en task förrän andra tasks färdigställts. Problemet uppstod främst under projektets sista sprint, eftersom fokus då var på att snygga till och fixa de sista funktionerna, vilket redogörs för mer under 2.2. I ett optimalt fall hade vi lyckats bibehålla vertikala user stories och oberoende tasks även in i slutet, kanske genom att fixa till vissa av de tasks vi lämnade till slutet i tidigare sprints. Exempelvis hade vi kunnat säkerställa att den gemensamma stylingen med css färdigställdes och faktiskt användes redan i någon av de tidigaste sprintsen, så utvecklarna som skapat nya sidor och funktioner direkt vetat hur det skulle se ut. Det är något vi kommer ta med oss till nästkommande utvecklingsprojekt och försökt implementera tidigare för att slippa det dubbelarbete som vi behövde göra i detta projekt.