

Team Reflection - Week 7

Group 3

18 maj 2020

1 Kundvärde och omfång

1.1 Success criteria

Slutav användaren fick testa applikationen nu i veckan. Han uppskattade dem nya grafiska och navigationsförändringarna vi hade gjort. Detta ses som väldigt positivt och är något vi aktivt har jobbat mot. Gruppens mål är att göra en applikation som gör det enkelt för spelaren att räkna slag. Gruppen har sedan i början på sprinten att denna veckan skall fokusera på funktionalitet och grafiskt arbete. Detta har lett till att gruppen har utformat sina job stories utefter det här.

1.2 User stories

Gruppen sade att denna sprinten skulle fokusera på att göra applikationen användarvänlig. Detta skulle uppnås genom att göra mycket grafiskt arbete och bugfixar. Så att applikationen blir så enkel att använda som möjligt.

1.3 KPI

Som nämnt i Team reflection för week 4, infördes fem nya KPI:er, i syfte att följa upp hur gruppmedlemmar upplever att samarbetet fungerar och hur de mår.

Då gruppen har sprint review på måndagar, syftade den första utvärderingen av KPI:erna till sprinten vecka 18.

De fem KPI:erna, *kommunikation*, *arbetsbörda*, *stressnivå*, *arbetsprocess* och *tillfört värde* utvärderades utifrån nedanstående skala:

- 1 = Mycket missnöjd
- 2 = Missnöjd
- 3 = OK
- 4 = Nöjd
- 5 = Mycket nöjd

2 Social Contract and Effort

2.1 Social contract

Gruppen har arbetat kollektivt och separat denna veckan. Men samtliga i gruppen har bidragit till respektive del. De flesta i gruppen har varit med på majoriteten av standups. Det har dock varit god kommunikation denna veckan vilket innebär att gruppen har fått gjort alla kritiska moment.

2.2 Time spent

Gruppen har spenderat mycket tid denna veckan på att färdigställa applikationen. Detta har gjort så att gruppen har kunnat färdigställa appen. Detta innebär att gruppen har gjort väldigt mycket mer jobb än det som estimerats i sprinten. Detta eftersom att det är svårt att planera allt fixande och/eller bug fixande i förväg.

3 Designbeslut

3.1 Design decisions support customer value

Vi har valt att lägga banorna på en server. Tidigare var banor hårdkodade i appen vilket innebar att för att lägga till en ny bana var appen tvungen att uppdateras. Genom att ha banorna på en server kan fler banor läggas till utan att appen behöver ändras. Så fort en bana har lagts till på servern finns den i appen. Detta skapar kundvärde då kunden får en smidigare upplevelse. Vill kunden ha en ny bana går det enkelt att lägga till den på servern. Hade nya banor varit tvugna att hårdkodas hade det tvingat kunden till att uppdatera appen vilket inte är ett jobbigt steg för kunden.

3.2 Teknisk dokumentation

Vi använder följande dokumentation som nämnts i tidigare teamreflektioner:

- UML-klassdiagram
- Databas / ER-diagram
- Designbeslut (för kod och arkitektur)
- Flödesdiagram för navigering i appen
- Kodkvalitet & Konventioner
- Testningsprocedur innan merge med master
- Grafisk design och UX
- JavaDocs med beskrivning för hela modellen

3.3 Hur vi använder och uppdaterar dokumentationen

Vi har ett dokument som listar hur vi skall skriva vår kod. Dokumentet ger samtliga i gruppen en slags guideline i hur man vill utforma koden. Detta gör så att all kod skall vara så enhetlig som möjligt. Till exempel så hade det utvecklats ett stycke kod som inte stämde överens med det förbestämda designmönstret Model view Controller (MVC). Då bestämde gruppen att detta skulle skrivas om så att det var utformat som resterande del av applikationen. Ändringen gjorde också så

att den delen av applikation kunde anpassas så att grafiken blev enhetligt i applikationen. Därför är det viktigt att följa den tekniska dokumentation som framtagits. Sedan kommunicerar gruppen med enkla diagram och liknande under standups så att det blir tydligare hur vi skall utforma vissa grafiska delar.

3.4 Code quality and code standards

Under den sista sprinten har en del av arbetet gått åt att se till att den överväldigande majoritet av varningar från IDE:n hanteras. Detta har sett till att vi uppenbara buggar, som exempelvis kan undvikas genom att hantera undantag korrekt, kan undvikas. Utöver detta har stor andel av koden "behandlats med putsmedel", dvs att det kontrollerats utifrån de kodstil standarder som etablerats, försök till att eliminera kodduplicering och bryta ut funktioner för att öka läsbarhet.

4 Application of Scrum

4.1 The roles used within the team

Genom att använda samma roller har vi haft fördelarna och nackdelarna som medföljer statistisk distribution av arbetsområden. Det som kan gjorts annorlunda är att exempelvis ha roterande ansvarsområde, vilket skulle kunna ge en känsla av ägande och förståelse för alla medlemmar gällande just de rollerna. I vårt fall har vi endast haft Scrum Master och dedikerad kundkontakt som statistiska roller utöver den allmänna utvecklarrollen.

Något vi saknat är en sann Product Owner (PO) roll, då vår kundkontakt inte deltagit i projektet i den mån för att uppfylla det som en Product Owner har som roll i agil utveckling. Istället för den dedikerade PO roll [1] har laget istället valt att samlat agera som PO genom att som lag valt att prioritera backlog items utefter slutanvändarens behov och krav. Detta har lett till att vi som lag kanske tagit beslut om hur backlogen prioriteras färgat av som kanske ses som enkelt att implementera. Problemet här kan ju bli att det kanske inte riktigt blir representativt av vad en PO hade gjort, då de inte haft utvecklingssidan i beaktning i samma grad som vi haft.

4.2 Sprint review relationship to scope/value

Vi använder sprint reviewn för att utvärdera vad som har gjorts under den förgående sprinten. Detta gör gruppen tillsammans så att vi kan säkerställa att arbetet ligger i linje med det som gruppen vill få gjort. Utifrån det som slutanvändaren har förmedlat till gruppen. Genom att köra en review kan gruppen avsluta tasks och user stories. Detta för att gruppen alltid ska utveckla en application som hjälper kunden utföra det den vill.

4.3 Agile practices, and their impact

Genom att ha de dagliga standups har vi haft mycket kortare återkopplingstid än ifall man ej etablerat satt regelbundna tider för återkoppling. Detta har den positiva effekten att problem kan snabbt tas om hand på ett sätt som minskar spendering av resurser i tid, genom exempelvis kunskapsdelning.

Sprint retrospective började vi inte applicera föränn en bit in i projektet och kom därför kanske inte lika långt fram i förfiningsprocessen av det. Dock har det varit värdefullt att utvärdera hur man tycker det gått, då det mycket likt daily standup skapar en återkommande återkoppling till vad som kan förbättras i processen. Mycket av värdet kom från den diskussion som hållts efter varje

social KPI mätning som vi utfört. Något som kanske kunnat tilläggas hade varit “Fem varför” för att bättre förstå problematiken som tagits upp vid en retrospektive.

Referenser

- [1] ScrumGuides.org, *Scrum Product Owner*, 2020. Tillgänglig: <https://www.scrumguides.org/scrum-guide.html#team-po> (hämtad 2020-05-24).