

Gruppreflektion för vecka 4

Group Joy

1 oktober 2021

1 Social Contract and Effort

1.1 Sociala kontraktet

Your 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)

Vi är nöjda med vårt sociala kontrakt. Det har inte uppkommit något som vi har velat lägga till det. Vi känner att alla våra förväntningar på varandra står i kontraktet. Det vi känner som har varit problemet har varit att folk inte alltid har följt det sociala kontraktet till punkt och pricka. Det har inte varit några större brott mot kontraktet, utan att folk har missat att kommunicera att det har uppkommit förhinder för dem för att delta på möten, eller glömt av möten.

Vi vill att det sociala kontraktet följs av samtliga gruppmedlemmar. När folk får förhinder för att delta på möten bör de kommunicera detta till de andra gruppmedlemmarna. Och de bör definitivt inte glömma av att det är möten.

Vi tror att en stor del av de missade mötena härstammar från att flera gruppmedlemmar har varit ute och rest, samt att det har varit flera större arrangemang som gruppmedlemmar har varit med och arrangerat. Då dessa resor är avslutade och arrangemangen avslutade, tror vi att det kommer bli mycket bättre med närvaron på mötena.

2 Design decisions and product structure

2.1 Hur designval skapar användarvärde

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

Vi valde att visa statistik över hur många personer som har fått sin första Covid-19 vaccinspruta därför att det är dagsaktuellt, för stort sett samtliga invånare i Sverige. Det är fortfarande en pågående pandemi, och den har påverkat människors hälsa i en enorm utsträckning, både direkt och indirekt. Genom att visualisera statistiken genom en heatmap över Sverige gör man det väldigt enkelt för användarna att snabbt få en överblick över dels hur läget ser ut i deras region av landet, men även på eventuella skillnader mellan de olika regionerna. Den informationen kan vara viktig för bland annat ansvarsutkrävande av de ansvariga om det skulle uppstå stora skillnader inom Sverige. Genom utveckla applikationen som en webbapplikation så tillgängliggör vi visualiseringen och statistiken för samtliga användare på alla större, och även mindre, plattformar där man kan köra en modern webbläsare. Detta för att inte riskera att en större grupp användare inte får tillgång till informationen.

Det vi vill göra i framtiden är att tillgängliggöra mer statistik och data på samma sätt som vi har gjort med Covid-19 vaccindatan. Vi vill även göra statistiken mer relevant genom att inkludera befolkningsstatistik. Just nu visar vi endast det absolut antalet personer som har fått sin första spruta av Covid-19 vaccin, vilket inte är relevant utan att veta hur många personer som finns i de olika åldersgrupperna i regionerna.

Vår plan är att få applikationen visa Covid-19 vaccindatan först, för att sedan lägga till befolkningsmängden som en annan datakälla. När vi sedan har detta klart räknar vi med att det kommer vara relativt enkelt att lägga till fler datorkällor i framtiden.

2.2 Vilken dokumentation som vi har använt

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

Vi har använt oss av den officiella dokumentation för bland annat Leaflet, D3, Express.js, Node.js när vi har behövt kolla upp hur man gör saker. Vi har även kollat på tutorials för att lära oss mer. Vi har även kollat upp specifika frågor på Stack Overflow.

Vi planerar att använda liknande dokumentation i framtiden.

2.3 Uppdatering av dokumentation

How you use and update your documentation throughout the sprints

Just nu har vi nästan ingen dokumentation.

Vi vill i framtiden ha dokumentation om hur

- frontenden och backenden kommunicerar med varandra;
- back-end:en hämtar in data; och
- man startar de olika systemen.

Detta gör vi genom att dedicera tid under sprintsen för att skriva dokumentation.

2.4 Kodkvalité och kodstandarder

How you ensure code quality and enforce coding standards

För att uppnå en högre kodkvalité har vi satt upp ett arbetsflöde där gruppmedlemmar

1. skapar en *fork* av huvudrepositoriet (Albins repository);
2. skapar en feature-branch på sin fork;
3. skriver kod;
4. laddar upp koden till sin kopia av repositoriet på Github;
5. skapar en pull request till huvudrepositoriet;
6. scrum master och PO recenserar koden; och slutligen när kodkvalitén är tillräckligt hög
7. mergar in koden.

Just nu har vi ingen satt kodstil, och alla skriver kod på det sättet som vi vill. Vi har inte heller något sätt just nu för att säkerställa att alla följer en kodstil om vi skulle sätta en.

Våran nuvarande process med kodrecensioner tycker vi är bra, och behöver därmed inte ändra den. Det hade varit lite önskvärt att kunna göra det tekniskt omöjligt för folk att merge sin egna kod, men vi har inte hittat någon enkel lösning på det med Github. Vi vill ha en kodstilsstandard i projektet, och även ett automatiserat system som kontrollerar att kod följer standarden.

Standardlösningen för att automatisera kontroll av kodstil är att sätta upp en CI-pipeline som automatiskt testar, kontrollerar, och bygger kod så fort man skapar en pull request till huvudrepositoriet. Vi ska försöka sätta upp detta, men vi har ännu inte gjort någon djupdykning i hur man ska sätta upp det. Om det inte är för svårt så ska vi implementera det. För att säkerställa att detta görs tänker vi dedikera tid under en sprint för att göra detta, med en tillräckligt hög prioritet.