Team reflection week 7

Ellen Widerstrand, Henrik Lagergren, Isabelle Ermeryd Tankred, Louise Larsson, Nils-Martin Robeling, Oscar Orava Kilberg, Sara Hillström

16th of May 2019

1 Customer value and scope

This sprint, as well as during the last sprint, we have narrowed down our user stories and our scope to create the main customer value for both parts of the platform; to get an easy shopping experience for the customers and to easily upload goods and see how many goods have been sold and after that update the inventory for the producers.

As we talked about in our last two team reflections, we have a greater focus on learning how to estimate effort and velocity better. We all feel that during this week we have gained a greater understanding of how to estimate the effort needed for different user stories and break down their different tasks. This is probably partly because we have narrowed down our user stories which has made our working progress more standardized, partly because we are getting more used to coding in React and using the tools needed, which makes estimating effort easier. We also feel that our velocity as a team has increased, probably due to the same reasons.

As the weeks go by we have gotten to know each other and each others way to work, which has made the sprints more successful and efficient. We have learned to implement SCRUM better during this sprint, which has been a success criteria of ours. This is probably due to us getting more used to working agile and with SCRUM-methods. This sprint we have really seen how we create value with our web-based platform, which has been an important success criteria for us as a team. Our main goal for this sprint was to get a value creating flow for both the customer and the producer. When it's time for our presentation we want to be able to show how a customer easily can log in to the page, search for a farmer, choose the products of interest and finally go through with a purchase. We also want to show how the producer can easily update his inventory and see

a change in it when a customer has made a purchase. This goal was set after interviewing potential clients on both side of the platform.

As in previous sprints the acceptance tests has been approved by at least two peers and the product owner. This sprint it has been easier to check acceptance criteria since we have more clearly defined and narrowed down our user stories. This, combined with the fact that the site as a whole now provides value (since the user stories during this sprint have consisted of connecting different functions) has made it easier as a PO to actually see what value is provided.

During this sprint we have continued to use our three different KPIs. Here is the result for the 'Team member satisfaction' of the week:

(Higher is better) Last weeks results:

Workload: 6.9 out of 10
Contribution: 7.6 out of 10
Support: 8.6 out of 10

This weeks results:

Workload: 8.7 out of 10
Contribution: 7.4 out of 10
Support: 8.6 out of 10

The rating of contribution and support have barely changed since last week. We believe that the constant and high average rating of these are due to the fact that our team work is working well. The workload rating on the other hand has increased, which we believe is due to that we have a greater understanding of how to work in React and how to define the different user stories.

Seen to the defect KPI we had 1 defect last week which was resolved this week. This is great statistics, but we have found 1 new defect during this week that we haven't solved yet. Overall we are satisfied with this result since the new defect is rather recent and we haven't had the time to solve it yet.

Our estimated velocity for this week was 31 (5 of these consist in creating sprint retrospective and team reflection). Our actual velocity turned out to be the same, though we made some cuts to requirements of the log in user story which means we undervalued the effort required. So next week we will try to make sure that we don't underestimate the amount of work required, especially when it comes to functionality which we are not familiar with implementing.

2 Social contract and effort

Our social contract has worked good for us this week as well. We are all committed to following the contract and we believe that it gives us a good structure. During this project we have learned how to work together and we respect each others' work. Last week we mentioned that we want to get better at commenting our code, since it eases the workload when it's time to combine and connect the different parts of the project. For us to get better at following through with commenting code, we added this to our definition of done in the social contract this week.

This week we spent about 15 hours on average per person working on the project, this includes both general meetings and individual work. It is fewer hours than last week, but some of us struggled a lot with issues last week, whilst things have run more smoothly this week. Since we are getting closer to the end of the project, a lot of the user stories left covers smaller changes that connect different parts and they are often dependent on each other, which makes it more difficult to start with a new user story. If someone is working on one part and is not yet done with it, it's not possible to start working on a new user story. Even though less hours has been spent on the project this week we believe we're doing all right and that we're on phase with the project. We delivered more than what we had planned this week.

3 Design decisions and project structure

Our plan was to use Google's API for auto-completion when searching for producers near a certain address. Unfortunately, it turned out that Google charges a fee for the API, so we decided to skip that design feature and instead use a mock-up with hard coded values. We have also implemented a feature where the user can choose to give their current location to the application, which makes it easy for them to see nearby farmers.

To save time and create value for the user we have implemented auto-fill in most of the text fields in the checkout. We have also provided a simple validation in the checkout to secure that the user have written correct information.

We have had great use of our Trello scrum board throughout our project. One positive effect from getting better at defining user stories has been that the use of the board became more standardized. When the stories became more specific the pairs could pick one without discussing interaction with their stories with the entire team.

Last week we took it upon ourselves to increase the amount of comments we

add to components/files to improve the ability for someone from a different team to understand and work with/on that piece of code. With GitInspector we saw that on average 7% of our lines of code are comments, a positive result! However, we should look to balance comments per file reasonably. One way of making sure we don't forget to comment our code is by adding it to our definition of done and social contract, which we have done.

To ensure code quality we require peer review from at least two team members as well as approval from PO before the user story is done. Our experience is that the code quality ameliorated as the teams understanding and knowledge in React increased.

4 Application of scrum

The roles used in the team haven't changed this sprint. One possible addition is however a Scrum-board manager. Having someone who keeps track of changes made could be helpful as every now and then when a sprint is finished, a lot of pieces and requirements have changes or have moved. A quick team recap of any changes to the Scrumboard could make it easier to know how things are going and would force us to think more about exactly how a story stands development wise.

Our recent sprint retrospective mentioned effort estimation as something that could be improved with further experience of our working environment. Estimating correctly helps us alter the sprint scope to deliver value more regularly. As our KPI on velocity shows, we have probably become better at this.

Over time our ability to use Git has become better. It was hard to make sure all required packages and files were present on our local devices. By having meetings rather regularly, this has proven itself easier than expected to handle since we help each other use the sometimes confusing git commands and remind everyone of what new packages have been implemented.