Split It: Acceptance Testing

Latest revision: 2014-05-07

Product backlog

I want to be able to...	Importance (1-5)	Story points (1-10)
add contacts from my phone as contacts in the application	4	2
split a debt between several contacts	4	3
create an "event" with an associated debt	3	6
add contacts to an event	3	5
in an event, let the added contacts suggest what they should pay	3	6
add personal debts in events (if someone should pay more or less from a receipt)	3	
have a direct link to Swish	1	10
delete a debt	4	2
accept/reject a debt that someone associates to you	3	9
"scratch" a debt when it is "small enough" (which the user defines)	2	3
change a debt	4	5
add a picture to a debt (of for instance a receipt)	2	8
add a picture to an event	2	8
connect to Facebook, Google or other social medias	1	10

Sprint #1 (Deadline 26-04-14) FINISHED

I want to be able to...	Importance (1-5)	Story points (1-10)
add contacts to my contact list	5	2
add a debt to a contact	5	3
accumulate debts with a contact	5	4
view what my total owings are	4	3
view what others total debts to me are	4	3
get a status of my total owings/debts (total owings from others - total debts I have)	4	3

What to do
Learn the basics of how to use the Android development platform, APIs and how to work to work with GitHub. Learn how to connect an Android phone to the application. Create basic functionalities of the app. These functionalities consists of user stories regarding creating contacts and debts. Another task for this time period is to get used to how to work with scrum.

How to do it
Continuous contact with the teams supervisor. Use the internet to find relevant guides and tutorials. Use lecture slides to find out how to work with software engineering. Develop the application according to scrum – daily scrum, sprint review, sprint retrospective and sprint planning.

Sprint #2 (Deadline 14-05-14)

I want to be able to...	Importance (1-5)	Story points (1-10)
split a debt between several contacts	4	3
create an "event" with an associated debt	3	6
add contacts to an event	3	5
delete a debt	4	2
change a debt	4	5

What to do

Continue working on the application. Create the event functionality, including the possibility to split a debt. Be able to change and delete debts.

How to do it

Continuous contact with the teams supervisor. Use the internet to find relevant guides and tutorials. Use lecture slides to find out how to work with software engineering. Develop the application according to scrum – daily scrum, sprint review, sprint retrospective and sprint planning.

Sprint #3 (Deadline 21-05-14)

I want to be able to...	Importance (1-5)	Story points (1-10)
add contacts from my phone as contacts in the application	3	2
add personal debts in events (if someone should pay more or less from a receipt)	3	
add a picture to a debt (of for instance a receipt)	2	8
add a picture to an event	2	8

What to do

Continue working on the application. Connect the application with the phone book. Improve the event functionality. Add various pictures.

How to do it

Continuous contact with the teams supervisor. Use the internet to find relevant guides and tutorials. Use lecture slides to find out how to work with software engineering. Develop the application according to scrum – daily scrum, sprint review, sprint

retrospective and sprint planning.

Sprint #4 (Deadline 28-05-14)

I want to be able to...	Importance (1-5)	Story points (1-10)
in an event, let the added contacts suggest what they should pay	3	6
have a direct link to Swish	1	10
accept/reject a debt that someone associates to you	3	9
"scratch" a debt when it is "small enough" (which the user defines)	2	3
connect to Facebook, Google or other social medias	1	10

What to do

Continue working on the application. Work with the connection between users on different phones. Check out future possibilites to connect with other applications, such as Swish and Facebook.

How to do it

Continuous contact with the teams supervisor. Use the internet to find relevant guides and tutorials. Use lecture slides to find out how to work with software engineering. Develop the application according to scrum – daily scrum, sprint review, sprint retrospective and sprint planning.