

Sprints

Velocity: 10 (2 per person/week)

Task values: 1, 2, 3, 4, 5, ?

Sprint 1

Discussion about how to handle task breakdown. Decided to go with more general tasks where the person assigned to it would get to break it down into smaller pieces themselves. No formal user stories.

General application structure, i.e. clearly definable parts of the application we want:

- Travel
- Pay
- Account
- History
- Settings

Division of **Travel** into tasks (cost in parentheses):

- REST - class (2)
 - ElectriCity + Västtrafik
- Front-end: Choose source-screen (4)
 - GPS-check for stop
 - Close by stops
 - Choose line based on stop
- Front-end: Choose destination-screen (2)
 - Get stops from Västtrafik
- Front-end: Travel info-screen (2)
 - Identify bus by wifi or with gps
 - Show when the bus will arrive
 - Show destination
 - Show arrival time
 - Toggle Stop
 - Toggle Alarm
 - (Show delays)

Definition of done:

Properly tested, code reviewed by non-implementer before merge into master.

Sprint 2

Underestimated workload of API. Decision to scrap definition of done. Perform informal testing (no unit tests), code doesn't have to be formally reviewed before merge into master. Improve later if issues arrive. Wrote user stories (though never used throughout the project).

User stories:

As a user I want multiple languages

As a user I want to be able to change settings

As a user I want the app to press the stop button automatically

As a user I want the app to notify me when the vehicle approaches my final destination

Finished from previous sprint:

Front-end: Choose source-screen

Identify bus by wifi

Now we properly define tasks, including priority and workload, in spreadsheet. Assign tasks through Trello.

Tasks:

Front-end: Travel info-screen	(2)
Alarm-functionality	(2)
Front-end: Settings	(2)
Refactor from Activities to Fragment	(1)
Hamburger menu	(2)
Continuation of API	(?)

Sprint 3

We decided if you are done with a task, you can continue with some other task. Bugfixing and fixing other small things doesn't need to have a formal task. Just branch, work, and merge with master.

Finished from previous sprints:

Front-end: Travel info-fragment

Västtrafik API

Hamburger menu

Settings

Tasks:

Send data between fragments (1)

Back-end: Destination-fragment (2)

Back-end: Departure-fragment (2)

Back-end: Info-fragment (2)

Front-end: Payment (2)

Sprint 4

Finished from previous sprints:

Front-end: Payment

Back-end: Destination-fragment

Back-end: Travel-fragment

Back-end: Info-fragment

All screens in the app are now mostly functional. Time to put them all together properly.

Tasks:

More robust data passing between fragments (2)

Bugfixes and small things (?)

Sprint 5

More bugfixing. Write documentation. Sprints mostly abandoned. Handle what's left of development by communicating and Trello.

State of spreadsheet (task names do not match this document):

Task	Priority	Description	Cost	Done
Change travelactivity to fragment	1	Have the same functionality and look but as a fragment instead	1	DONE
Hamburger menu	2	"Länka" till fragment	1	DONE
Västtrafik API	3	REST class communicating with Västtrafik	3	DONE
Create destination fragment	4	Create a fragment containing	2	DONE
Reseinfo-fragment	5	Make front-end for the info fragment	2	DONE
Link fragment data	6	Make the fragments communicate the travel data in between	1	DONE
Departure fragment gets data from VT-api	7	Make the departure fragment use real data from VT	2	DONE
Larmfunktionalitet	8	Add functionality for notification when almost at destination	2	DONE
Destination fragment gets data from VT-api	9	Add some good backend to destination fragment so we actually see the real stops	1	DONE
Create settings fragment	10	Choose language, turn larm on/off, ???	2	DONE
Reseinfo-fragment hämtar data från VT-api & EC-api mha wifi-finder	11	Add some good backend to travel fragment so we actually see the real stops	4	DONE
Betalningsfragment	12	Add front-end for payment	2	DONE
App-ikon	13	Gör en app-ikon	1	DONE
Lägg till färdiga fragment i hamburgermenyn	14		1	
Developer documentation	99	Se exempel på github	5	
Användarmanual	100	Skriv användarmanual	1	
			30	