

What will be delivered after this sprint? Final product
 How will we achieve this work?
 CRAFT A SPRINT GOAL - will provide guidance

As a customer, I want to **add an item to my order** so that I can see a summary of my order later at checkout and buy it.

- Design GUI (doesn't need to be final, just functional enough for this use case)
- Implement Utility package
- Unit testing of utility class
- Document ^^^^
- Implement RMI
- Document ^^^^
- Implement Model interface and start implementing some methods on the Model Manager
- Document ^^^^
- Document this sprint (why we started with it, etc)

SPRINT GOAL: By the end of this sprint, we should have a window in which we can add an item to an order, send the order to the server and **see it in the database**.

PB-ID	ID	Task Title	Responsible	Estimate (hours)	Status
1	1	Design GUI	Robert + Kamil	6 (3*2)	Completed
1	2	Implement Utility Package	Daniel + Laura	4 (2*2)	Completed
1	3	Implement Model	Robert	6	Completed
1	4	Implement ViewModel (Includes making the ViewModelFactory + CustomerViewModel)	Robert	6	Completed
1	5	Implement View (Includes making the View Factory)	Kamil	6	Completed
1	6	Update Class Diagram	Robert + Laura	4 (2*2)	Completed
1	7	Document MVVM	Robert + Kamil	8 (4*2)	Completed
1	8	Test MVVM -> unit test for viewmodel	Robert + Kamil	4 (2*2)	Completed
5	9	Add Class about database to Class Diagram	Daniel + Laura	4 (2*2)	Completed
5	10	Implement Database Manager	Daniel + Laura	32 (16*2)	Completed
5	11	Document Database Manager	Daniel + Laura	4 (2*2)	Completed
5	12	Test Database Manager (properly documented -> integration test????)	Daniel + Laura	8 (4*2)	Completed

5	13	Implement RMI	Kamil	2	Completed
5	14	Document RMI	Kamil	2	Completed

DESIGN GUI: Make a sketch

IMPLEMENT UTILITY CLASS: Follow the Class Diagram

IMPLEMENT MODEL: Define the interface, and in the Model Manager only implement the methods relevant to this use case (the other ones are empty without throwing exceptions)

IMPLEMENT VIEWMODEL: Make the ViewFactory and a new ViewModel for this UseCase (could be the CustomerView)

IMPLEMENT VIEW: Use SceneBuilder and make ViewFactory and ViewController for this use case.

UPDATE CLASS DIAGRAM: According to the changes made above

DOCUMENT MVVM:

TEST MVVM: Unit test viewModel, bind to some properties and in the test make the binding and try to set one of the properties and see whether it is doing what is expected. Might not be easy but let's try.

ADD CLASS ABOUT DATABASE TO CLASS DIAGRAM: Figure it out with the youtube video and put it there

IMPLEMENT DATABASE MANAGER: Follow youtube video

DOCUMENT DATABASE MANAGER: Some screenshots & talking about it

TEST DATABASE MANAGER: Trying to add an order and seeing if it gets added to the database.

MAKE TEST CASES FOR THE USE CASE? no. Just integration tests for the trip of the methods.

NOTES FOR THE MEETING:

Show Product Backlog and discuss the use case number zero.

Show First Sprint. **INCEPTION PHASE**

- Analysis is documented :)
- Class diagram is big & nice

- Database is designed and documented
-

Sprint #2

