1 The Microwave Oven system

In the following, the design of the system is described with a Use Case diagram, a Use Case description, a class diagram and some sequence diagrams.

The code for the classes with their unit tests and integration tests can be found on Brightspace or GitHub as:

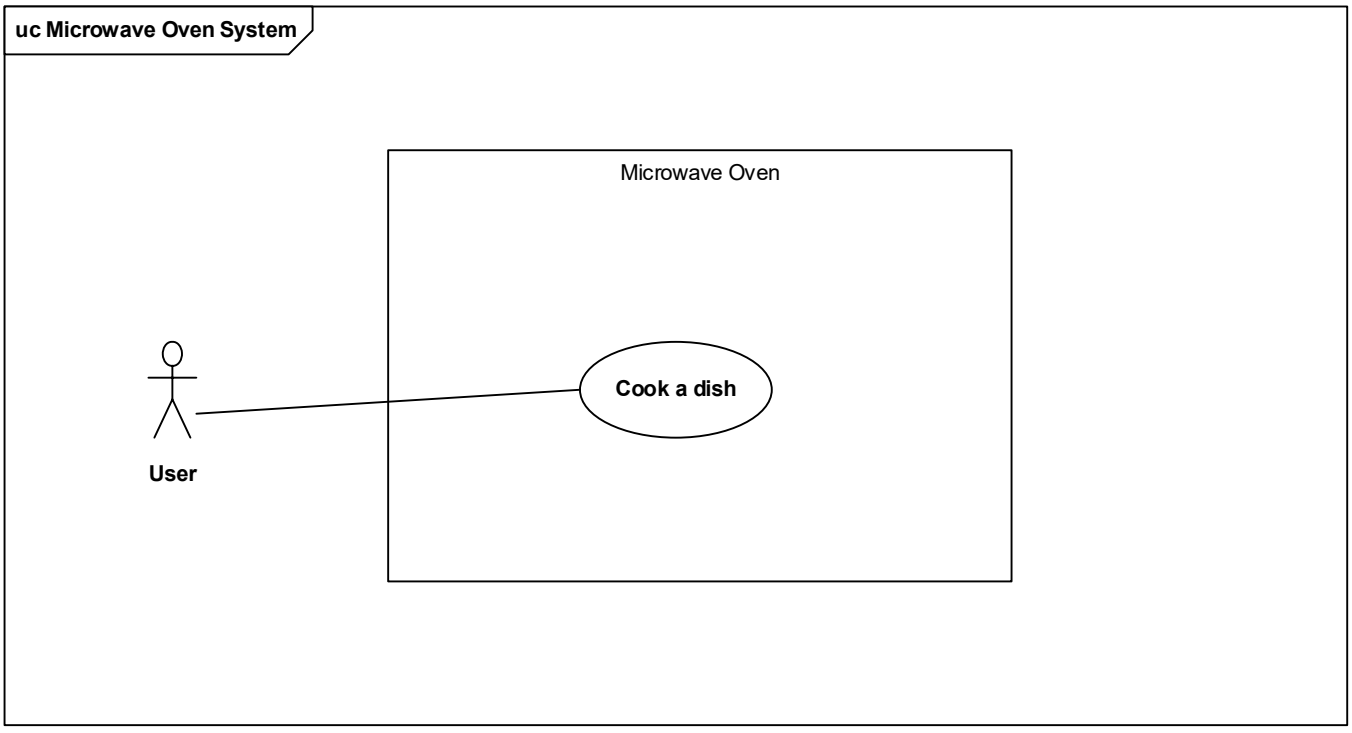
- zipped Visual Studio solution from Brightspace
- copied as template from https://github.com/TeamSWTFrabj/MicrowaveOven to your own GitHub account
- forked from the same place to your own GitHub account
- or downloaded as a zip file from the same place.

When working from the zipped file, you have to add the solution to source control and publish to GitHub.

Be sure to use one of those methods; don't just clone from TeamSWTFrabj, as you cannot use my repository for your further work. The code also contains a demo application.

There is a Unit Test and Coverage job on Jenkins for the code under the name FrabjMicrowaveOven, to demonstrate that I have done unit testing. For integration tests, there is a Unit Test job, it doesn't make sense to calculate coverage for integration tests, it's name is FrabjMicrowaveOvenIntegration.

Use Case Diagram



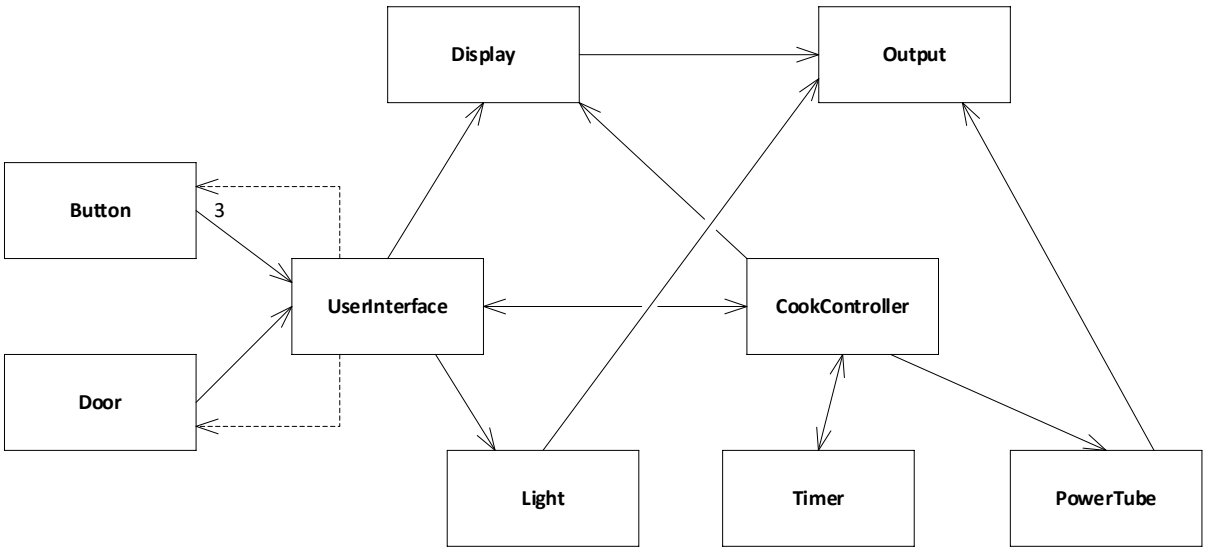
Use Case Description
A beeper has been omitted from this design for the sake of simplicity.

Name	**Cook a dish**
Goal	The user will have a dish of food cooked
Initiation	By the user.
Actors	The User

Precondition	The user has a dish of food on a microwave suitable platter, the MicroWave oven is attached to the mains power supply, no activity is currently taking place in the oven, the display is blank.
Postcondition	The oven is inactive, the food has been heated with the desired microwave power during the desired time. The display is blank.
Main Scenario	1. The user opens the door 2. The light goes on inside the oven 3. The user places the dish in the oven 4. The user closes the door 5. The light goes off inside the oven 6. The user presses the Power button one or more times, to select the desired microwave power. The display shows the currently selected power from 50 to 700 W, starting with 50 W. Each press increases the selected power level with 50 W, until 700, where it will return to 50 W on the next press. [Extension 1: The user presses the Start-Cancel button during power setup] [Extension 2: The user opens the door during setup] 7. The user presses the Time button one or more times to select the desired cooking time. The display shows the currently selected time as minutes:seconds, starting with 01:00. Each press increases the selected time with one minute. [Extension 2: The user opens the door during setup] 8. The user presses the Start-Cancel button. 9. The light goes on inside the oven 10. The powertube starts working at the desired powerlevel 11. The display shows and updates the remaing time every second as minutes:seconds. [Extension 3: The user presses the Start-Cancel button during cooking] [Extension 4: The user opens the Door during cooking] 12. When the time has expired, the power tube is turned off 13. The light inside the oven goes off 14. The display is blanked 15. The user opens the door 16. The light goes on inside the oven 17. The user removes the food 18. The user closes the door 19. The light inside the oven goes off.
Extension	[Extension 1: The user presses the Start-Cancel button during setup] 1. The display is blanked 2. All settings are reset to start values 3. The user can start the Use Case from step 6 or 15, or the Use Case ends. [Extension 2: The user opens the Door during setup] 4. The light goes on inside the oven 5. The display is blanked 6. All settings are reset to start values 7. The user can continue the Use case from step 4 or 17. [Extension 3: The user presses the Start-Cancel button during cooking] 8. The power tube is turned off 9. The display is blanked. 10. The light inside the oven goes off 11. All settings are reset to start values 12. The user can continue the Use Case from step 6 or 15, or the Use Case ends. [Extension 4: The user opens the Door during cooking] 13. The power tube is turned off. 14. The display is blanked. 15. All settings are reset to start values.

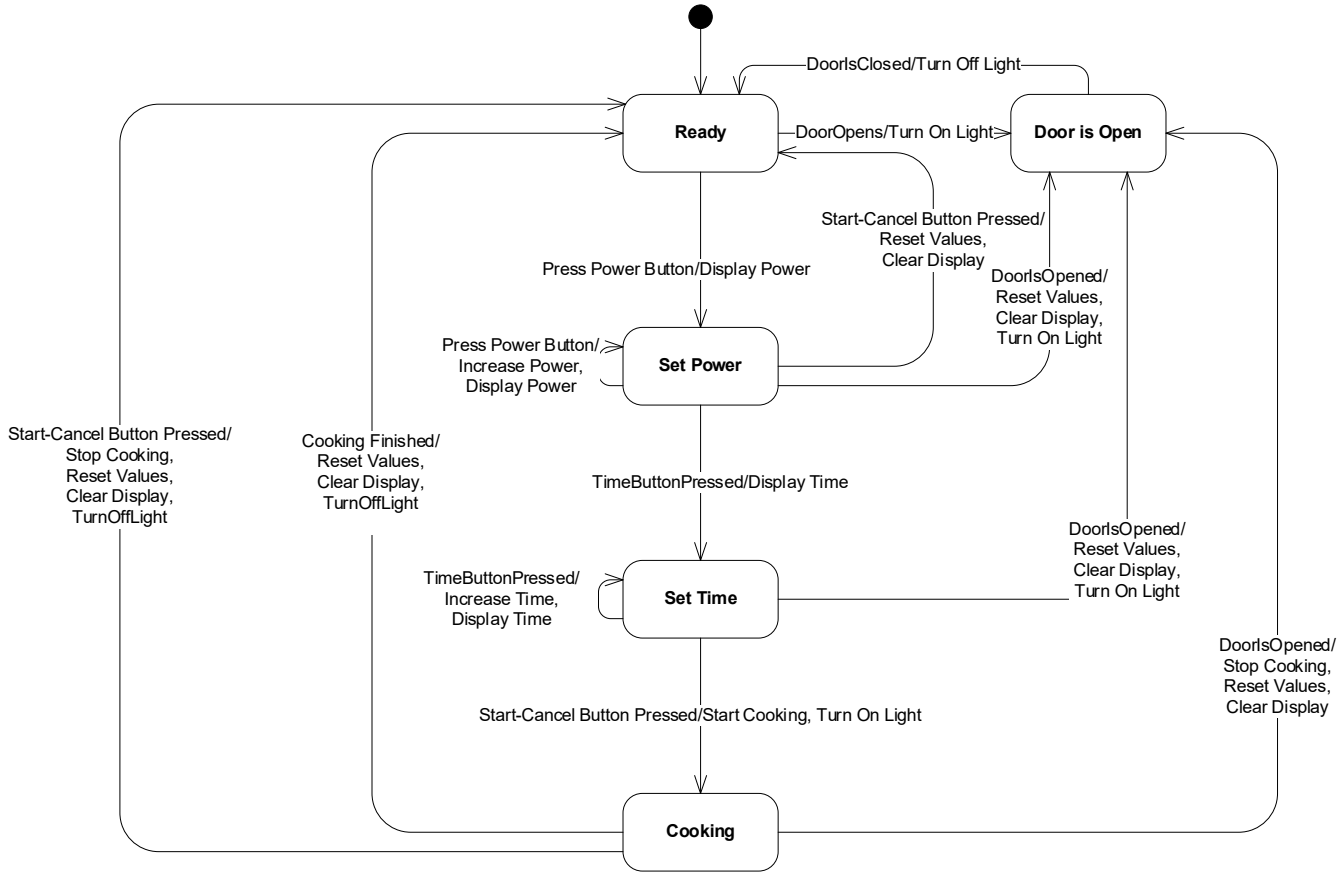
	16. The user can continue the Use Case from step 4 or 17.

Class Diagram

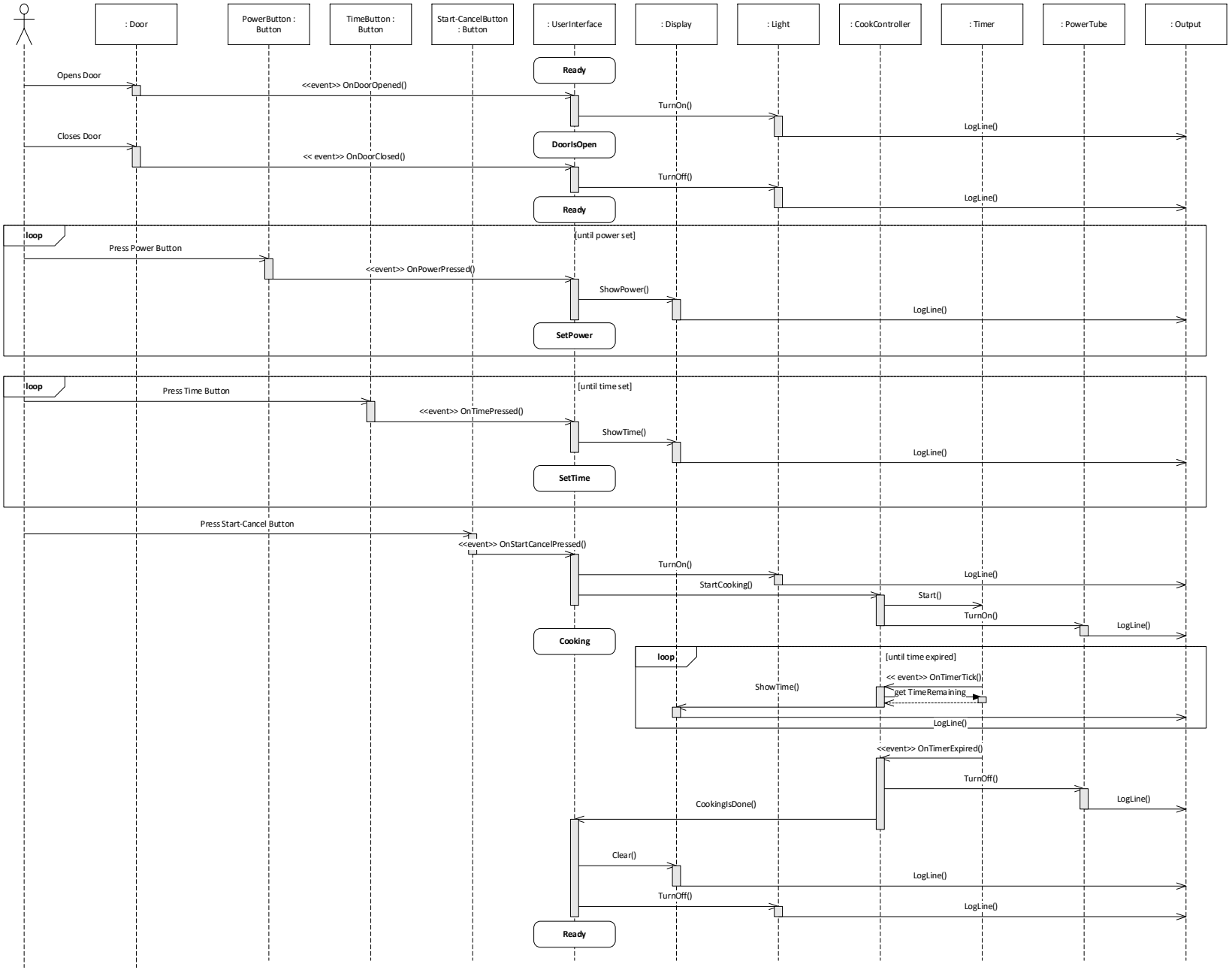


All classes are implementing a relevant interface – not shown on the diagram for simplicity – so dependency isolation is possible for testing. See the supplied unit tests.

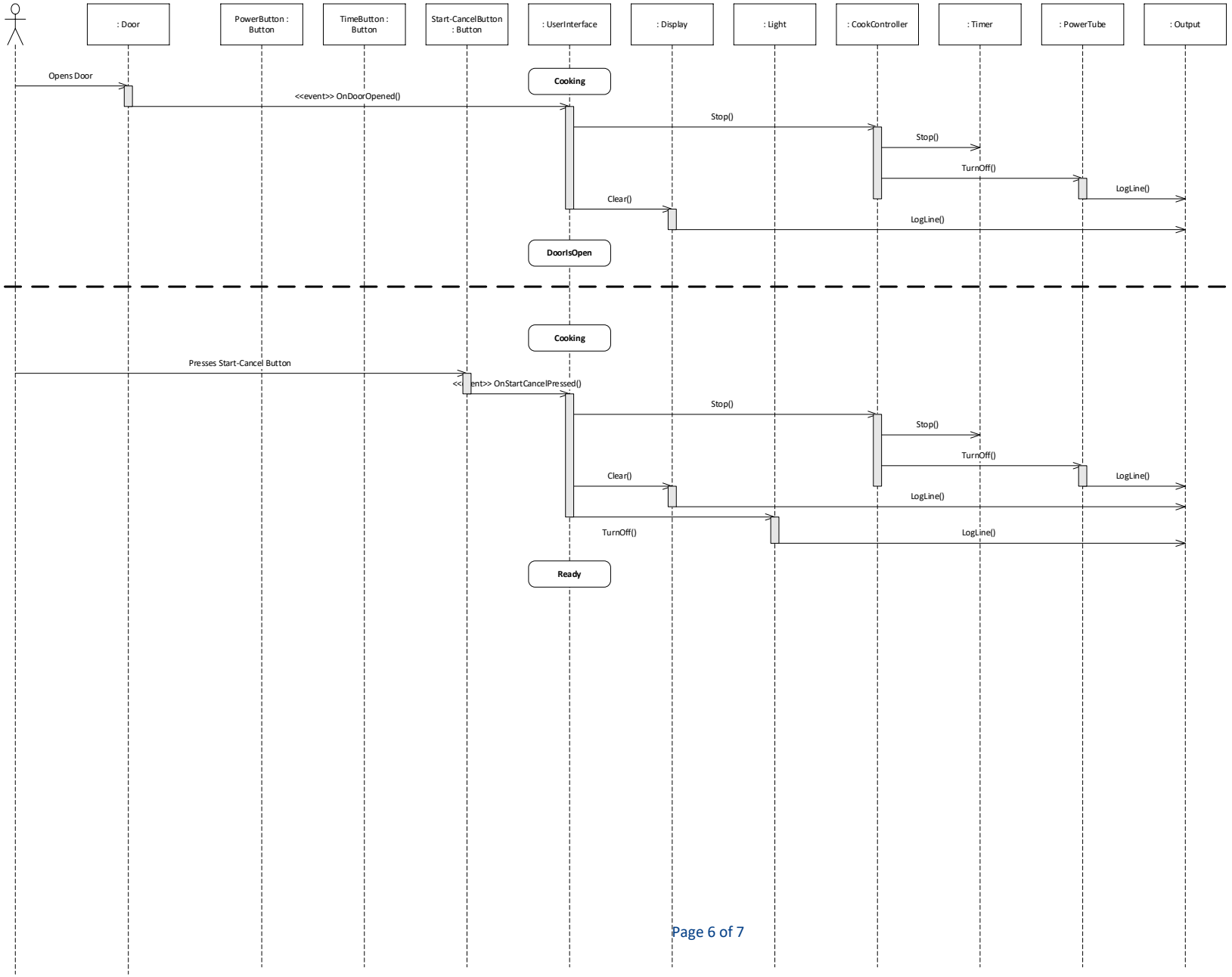
State Machine diagram for UserInterface



On the next page: Sequence diagram for the main scenario


: Door
PowerButton : Button
TimeButton : Button
Start-CancelButton : Button
: UserInterface
: Display
: Light
: CookController
: Timer
: PowerTube
: Output
Ready
Opens Door
<<event>> OnDoorOpened()
TurnOn()
LogLine()
Closes Door
DoorIsOpen
<< event>> OnDoorClosed()
TurnOff()
LogLine()
Ready
loop
[until power set]
Press Power Button
<<event>> OnPowerPressed()
ShowPower()
LogLine()
SetPower
loop
[until time set]
Press Time Button
<<event>> OnTimePressed()
ShowTime()
LogLine()
SetTime
Press Start-Cancel Button
<<event>> OnStartCancelPressed()
TurnOn()
LogLine()
StartCooking()
Start()
TurnOn()
LogLine()
Cooking
loop
[until time expired]
<< event>> OnTimerTick()
ShowTime()
get TimeRemaining
LogLine()
<<event>> OnTimerExpired()
TurnOff()
LogLine()
CookingIsDone()
Clear()
LogLine()
TurnOff()
LogLine()
Ready

And these are are sequence diagrams for the some of the Extensions.



This page is intentionally left blank.