GROUP E

SPRINT RETROSPECTIVE

STORIES

- Total number of stories : 6
- Committed in sprint : 6
- Successfully implemented: 6

- Total number of story points: 11
- Committed in sprint: 11
- Successfully implemented: 11

WHAT DO WE MEAN WITH “IMPLEMENTED”?

- Our definition of “DONE”
  - Code on VCS (GitHub)
  - Code review done
  - Unit testing done (JUnit)
  - End to End tests done

STORY POINTS – #TASKS

- Story 1: 2 - 7
- Story 2: 3 - 7
- Story 3: 2 - 4
- Story 4: 1 - 4
- Story 5: 1 - 2
- Story 6: 2 - 3
- Total #tasks: 27

*Details available in “planning_document.txt” on GitHub

TASKS – ESTIMATION VS REALITY

Story 1:

As a citizen I want to specify a certain type of request so that my request can be fulfilled

Environment settings:	6*1 h*	9 h
Definition and creation of Model	2*1 h*	3 h
Definition and creation of GUI	0.5 h	20 m
Creation of controller class	1 h	0.5 h
Unit tests	1 h	1 h
End to end system testing	0.5 h	0.5 h
Code review	6*0.5 h*	6*10 m
Total:	14h	15h 20m

*6*1h means 6 persons for 1 hour

TASKS – ESTIMATION VS REALITY

Story 2:

As an employee I want to know who the next customer is so that I know what I have to do next.

Modify/Add new features in the model	1.5 h	2.5 h
Definition and update of GUI	1 h	1.5 h
Definition and creation of Controller	0.5 h	1 h
Unit tests	1 h	20 m
Integration test	1 h	0.5 h
End to end system testing	0.5 h	10 m
Code review	6*0.5 h	6*20 m
Total:	8.5h	8h

TASKS – ESTIMATION VS REALITY

Story 3:

As a citizen I want to know which counter is going to serve me so that I know where to go

Update of citizen GUI	0.5 h	1 h
Update of Controller	0.5 h	3 h
Update of Model	/	1.5 h
End to end system testing	0.5 h	1.5 h
Code review	6*0.5 h	6*20 m
Total:	4.5 h	9 h

We underestimated the difficulty of creating a GUI that would be updated whenever the model was modified, without any user stimulus (Observer pattern).

TASKS - ESTIMATION VS REALITY

Story 4:

As a citizen I want to know how many citizens have to be served before me at any moment

Update of citizen GUI	1 h	1 h
Update of Controller	0.5 h	2 h
Update of Model	/	0.5 h
End to end system testing	0.5 h	0.5 h
Code review	6*0.5 h	6*20 m
Total:	5 h	6 h

TASKS – ESTIMATION VS REALITY

Story 5:

As a citizen I want to know an estimation of the waiting time.

Update of citizen GUI	0.5 h	/
Update of model	/	2 h
End to end system testing	0.5 h	0.5 h
Total:	1 h	2.5 h

During the first planning there was no code review, since the GUI code is automatically generated by SceneBuilder. After realising that there was the need to modify the model, we decided to perform the code review for this story together with the last one.

TASKS – ESTIMATION VS REALITY

Story 6:

As an admin I want to know how many clients have been served for each request type and for each counter so that I can compute some statistics

Update of model	3 h	1.5 h
End to end System Testing - final tests	3 h	1 h
Code review - overall examination	6*1 h	6*1 h
Total:	**12h**	**8.5 h**

Main difference here is given to the fact that we decided to use a .txt file instead of a database to store statistics (1.5h in model) and that the final system resulted much more simple to test than expected.

TASKS – SUMMARY

	Estimation	Reality
Story 1:	14 h	15h 20m
Story 2:	8.5 h	8 h
Story 3:	4.5 h	9 h
Story 4:	5 h	6 h
Story 5:	1 h	2.5 h
Story 6:	12 h	8.5 h
Total:	45 h	49h 20 m
Estimation error ratio:	49h 20m / 45 h = 1,096	
Actual hour per task	49h 20m / 27 ≈ 1h 45 minutes	

WHAT ABOUT STORY POINTS?

	Average estimated time	Average elapsed time
1 story point	3 h	4.25 h
2 story points	8.16 h	7.94 h
3 story points	8.5 h	8 h

- Why all this similarity between 2 and 3 story points?

The main problem was that we overestimated the difficulty of implementing the requested algorithm for choosing the next ticket (story 2), while the real problem was managing the interaction between GUI and model (story 3).
However, since we have just one story with 3 points in our sprint, the statistics are not accurate.

WHAT DID WE LEARN?

- We learned how to decompose a problem into stories and then into tasks, while producing a pretty accurate estimation of the relative weight among them.
- We improved our soft skills, interacting with each other while implementing the different tasks.
- We learned how to use the observer pattern
- We improved our skills in MVC programming and GUI designing through Scene Builder

THINGS WE ARE PROUD OF

- We were able to stick to our initial program, delivering all the stories for which we committed for.
- In addition to that, we released a program with all the functionalities requested by the stakeholders, maximising the amount of value delivered in a single sprint.

FOR THE NEXT SPRINT

- Improve the usability of the existing GUIs (bigger labels, fancier ticket with Polito logo…)

- Develop a GUI to read statistics efficiently and without effort

- Introduce the possibility to configure the system (<n> counters and <s> services). The system will read the desired configuration from a "configuration.txt" file.