Sprint retrospective

Office Queue Management

SE2 - Team 8

Barco Luca

Chen Jinzhuo

Farhadov Elchin

Finocchiaro Loredana

Laudani Daniele

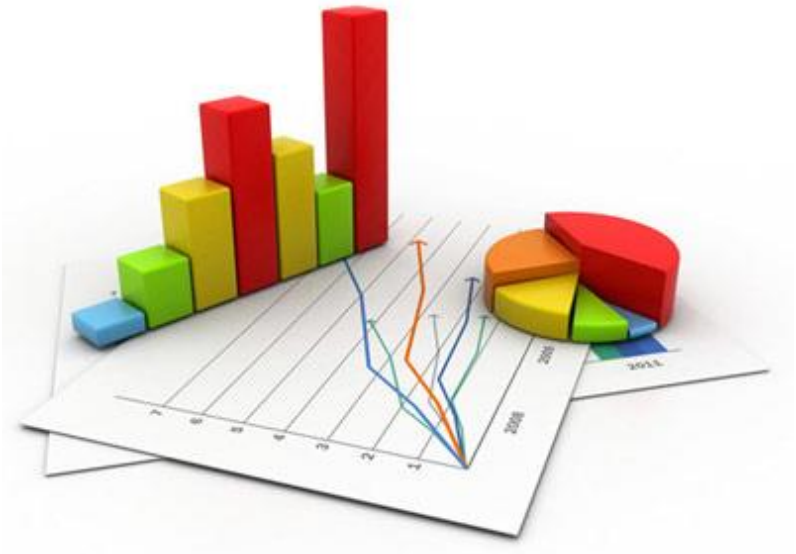
Nicolò Antonino

Implementation choices

- Web application + client-server model

- **Coding**
- Client: React.js
- Server: Node.js
- DBMS: Sqlite

- **Testing**
- Jest + Enzyme



Process

Macro and detailed statistics

Stories (included in the Sprint)

- OQ-1 AS AN officer I WANT TO press a button SO THAT I can call and serve the next user
- OQ-2 AS A customer I WANT TO know the average waiting time needed to process my request
- OQ-3 AS A customer I WANT TO know the ticket number that is being served in a certain moment
- OQ-4 AS A customer I WANT TO know the ticket number assigned by the counter
- OQ-5 AS A customer I WANT TO see on the main display board the queue's length
- OQ-9 AS A customer I WANT TO pick a ticket for a specific service type SO THAT I can be inserted in the queue

Stories (not included)

- OQ-6 AS A manager I WANT TO configure service time for each service type
- OQ-7 AS A manager I WANT TO configure the request types that could be served for each counter SO THAT counters could be organized properly
- OQ-8 AS A manager I WANT TO list the history of requests for each service type SO THAT I can compute statistics per day/week/month

Macro statistics

- Number of stories committed vs done: 6/6
- Total points committed vs done: 17/17

- Team of 6 members → Planned hours: 6 * 8 ph = 48 ph
- Number of hours planned vs spent (as a team): 48/50

- For detailed timesheets, we used the Clockwork Free application.

Detailed statistics

Story ID	Number of tasks	Points	Total hours estimated	Total hours spent
OQ-1	3	5	9 ph	13.5 ph
OQ-2	1	3	1 ph	1.5 ph
OQ-3	1	2	1.5 ph	3 ph
OQ-4	1	2	0.5 ph	0.25 ph
OQ-5	2	2	0.5 ph	0.5 ph
OQ-9	3	3	11 ph	10.5 ph

- Hours per task: 2.65 (avg), 4.27 (σ)
- Total task estimation error ratio (hours estimated/tot): 23.5/29.25 = 0.8

(Missing hours regard more general tasks, which were not considered as subtasks of stories)

Statistics about horizontal tasks

Such as «Toolkit setup», «Discuss about architecture to be choosen for the system», «Implement communication with the db», «Study how to perform testing», …

Number of tasks	Total hours estimated	Total hours spent
9	16	20.75 ph

- Hours per task: 2.3 (avg), 3.68 (σ)
- Total task estimation error ratio (hours estimated/tot): 16/20.75 = 0.77

Quality

Testing and coding review

Unit Testing

- Total hours estimated: 7 ph
- Total hours spent: 9 ph
- N° of automated unit tests: 11

Integration testing

- Total hours estimated: 3 ph
- Total hours spent: 5 ph
- N° of automated unit tests: 3

Coverage

CLIENT

All files

14.01% Statements `29/207` **14.13%** Branches `13/92` **17.28%** Functions `14/81` **14.72%** Lines `29/197`

Press *n* or *j* to go to the next uncovered block, *b*, *p* or *k* for the previous block.

File	Statements		Branches		Functions		Lines	
src	3.88%	4/103	0%	0/54	3.33%	1/30	3.96%	4/101
src/Components	22.03%	13/59	44.44%	8/18	22.58%	7/31	25%	13/52
src/Components/Officer	26.67%	12/45	25%	5/20	30%	6/20	27.27%	12/44

SERVER

All files

84.21% Statements `96/114` **50%** Branches `19/38` **100%** Functions `17/17` **84.21%** Lines `96/114`

Press *n* or *j* to go to the next uncovered block, *b*, *p* or *k* for the previous block.

File	Statements		Branches		Functions		Lines	
countersDao.js	88.24%	30/34	50%	4/8	100%	6/6	88.24%	30/34
db.js	83.33%	5/6	50%	2/4	100%	0/0	83.33%	5/6
serviceTypesDao.js	88.24%	15/17	50%	2/4	100%	4/4	88.24%	15/17
ticketsDao.js	80.7%	46/57	50%	11/22	100%	7/7	80.7%	46/57

Code review

- Total hours estimated: ?
- Total hours spent: ?

Assessment

How did it go?

Did you complete all the planned stories? If no, why?

- Yes, we were able to complete all the planned stories.
- Although, we made an error: we considered as «Done» stories correctly implemented, but not yet tested.
- Testing was considered among more general tasks, that were not completed entirely.

What caused your errors in estimation (if any)?

- We spent, more or less, 6 ph more with respect to the estimation. Furthermore, we noticed that hours spent for each story are often not so coherent with the points assigned to that story.
- The reason for underestimating is maybe due to not so great experience in managing the choosen implementation technology and in aligning own work with those of other members, that caused a bit of a waste of time.
- We can conclude we surely need to learn still a lot about estimation.

What lessons did you learn (both positive and negative) in this Sprint?

- ✓ Coordination between team members was quite good, but it's always possible to do better for the future.
- ✓ We reached our objective about completing all the committed stories.
- ✓ We gained many positive feedbacks from the stakeholders.

- × We had many issues concerning testing, because of very little experience about that.
- × We should improve our capability of estimating.

Propose 1 or 2 improvement goals for the next sprint and specify how to achieve them

(technical tasks, team coordination, etc.)

- Each of us should learn how to do testing in the choosen framework (since this is new for all of us) and test all the functions as soon as these are completed (only at that point the task can be considered as «Done» officially).
  This can be reached by studying documentations online individually and then we can meet and discuss together, trying to find a common line.
- Aligning skills as much as possible could be useful for learning more, better and reciprocally, considering that some of us don’t have so many skills about React.js.
  This can be easily obtained by permitting team members to assist other team members while they’re working.

One thing you are proud of as a team

- We are 6 different people, with different timetables/interests/skills (and time zones, too), but this first Sprint demonstrated that working together with «love and agreement» is possible, thanks to the fact that all of us are willing to listening, dialogue and discuss peacefully.
- We surely think there are many perspectives of improveness as a team and that there are all the conditions to do a good job together.