

Sprint Retrospective 4

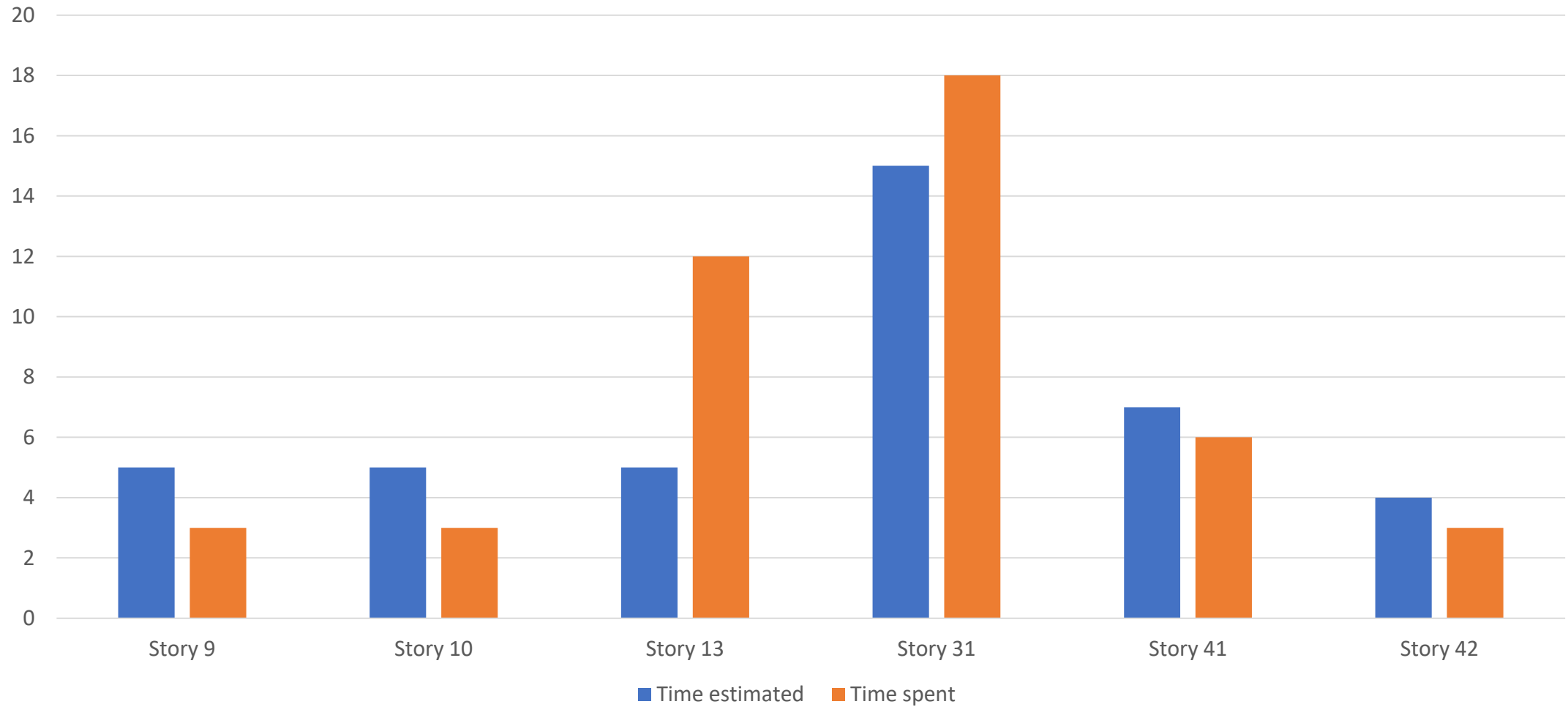
GROUP K

Process Measures

Macro Statistics

- In this sprint 4, we decide to attend 9 stories and we also decide to make major improvements of code.
- Total points of all stories are 233 (with all the stories), and we committed 54 of them and delivered 24.
- The number of points of delivered stories is low because we take one of the hardest stories for us and we don't make it works

Statistics of the sprint



Detailed statistics

Id Story	Number of Tasks	Points	Total Hour Estimation	Total Hour Spent
9	4	3	5	3
10	3	3	5	3
13	3	3	5	12
31	3	8	15	18
41	3	5	7	6
42	2	2	4	3

Statics of the time repartition

- Estimating time was 45 hours for stories and also 5 hours to improve the sonarqube and 10 hours of improvements of code.
- We have 3 tasks per stories approximately.
- The rate (time spend/time expected): $60/60=1$

Quality Measures

Unit Testing

- Total Hours estimated : 2 hours
- Total Hours spent : 3 hours
- Number of automated unit test : 0, we don't use automated unit test
- Coverage : not working

Differents Test

	E2E	Code Review	Technical Debt Management
Total Hours Estimated	4	5	6
Total Hours Spent	4	5	6

Overall Technical Debt at the demo

- Problem with SonarQube, when we update our code and we do analysis, all the errors don't change. So we can't evaluate the different value.

Assesssment

What caused your errors in estimation

- For the stories, the causes of errors are django problem with pip and problem with some features of the logiciel.
- It was difficult to understand how to create constraint on Django especially on the client side, because all the tutorials we find, use the server side to create constraint.
- We have problem with pip because some libraries aren't updated

What lessons did you learn?

- We learn that using a library is complicated and that we have to be smart to find a solution that has no background.
- We also learn how to use Sonarqube and what kind of mistake we can avoid in order to reduce the technical debt

Which improvement goal set in the previous retrospective were you able to achieve

- Find a solution to a difficult stories and a comprise between the different team member
- Reduce the technical debt and improve the code

Which ones you were not able to achieve?
Why?

- Find a solution to optimize the model.
- Make all the stories planned because of their difficulty

One thing you are proud of

- Solving most of our initial problems like communication between team members