

Project Zelula - SCRUM plan #5

Group 5/E:

Felix Akkermans

Niels Doekemeijer

Thomas van Helden

Albert ten Napel

Jan Pieter Waagmeester

June 17, 2012

1 Introduction

In this SCRUM plan we define the tasks we want to complete in this sprint. Every task is assigned to one or two developers and an estimate is provided for the effort required.

For each SCRUM run a milestone is created at GitHub, with issues for the tasks selected. The issues for this milestone can be found on: <https://github.com/FelixAkk/synthbio/issues?milestone=12>.

2 Selection and assignment of tasks

This is the fifth sprint of our project.

Available time

The time available is five mornings in two weeks by five people. That's about twenty hours per person. Including meetings we come to an estimate of 90 hours of actual working.

A list of tasks, assignments and effort estimations is included in a table.

Task	Developer	Effort	
		estimated	actual
#81 Server: Fix simulation results	Albert	8	5
#86 Refactoring circuit servlet	Jieter	3	4
#87 Server: Read new CSV input	Jieter	2	4
#88 Client: Tooltip graph	Niels	2	1
#73 Client: Resize	Felix	6	4
#66 Client: Acceptance testing	Felix + Thomas	8	7
#71 Client: Compound Gates	Niels + Thomas	12	14
#77 Finish Scrum plan 5	Felix	2	2
#52 Final report: Key Problems/Solutions	Thomas + rest	4	2+1
#55 Final report: Reflection Teamwork	Everyone	5 * 1	5 * 1
#89 Final report: Introduction	Jieter	1	0.1
#90 Final report: Description of product	Jieter	4	4
#92 Final report: Design and implementation process	Niels	5	5
#62 Confirmation on close		0	dropped
- Code Review	Everyone	5 * 3	6+2+3+2+1
Total		77	70,1
Optional tasks			
#43 Client: Polish circuit styles	-	-	dropped
#60 Automated QUnit tests from ant	Albert	-	1
#14 Client: Compile .less server side	-	-	dropped
#13 Client: Assign keyboard shortcuts	Felix	-	3
#84 Client: Input and output fields resize and display open connection	-	-	dropped
#82 Client: Migrate stuff to simulation tab	Felix	-	12
#65 Save input/output gate position	Niels	-	2
#43 Client: Deleting of wires and gates using delete	-	-	dropped
#43 Client: Highlighting/selecting gates to move multiple or delete multiple	-	-	dropped
#91 Client: Table with simulation results	-	-	dropped
Total		-	18
Extra tasks			
- Fixing the simulator: writing our own simulator	Jan Pieter	-	20
- Exporting graph: serve our own SVG rasterizer	Jan Pieter	-	4
Total		-	24
Grand Total		77	112,1

3 Reflection on this iteration

Solver issues

During this iteration we discovered a problem with the connection to the solver which was implemented. It showed correct results for the transcription and translation reactions, but degradation did not work. After discussing things with Alexey, Jan Pieter decided to write a solver on its own, which initially took about 12 hours of work. Cleaning up the code and some other adjustments took another 8 hours.

During the development of our own solver, Albert worked together with Alexey to fix the problems in the first approach. The result of this is the availability of two solvers in our GUI.

Graph rasterisation, servlet issues

The graph library used in our GUI provides a way to export the graph to different image formats. This exporting requires a special server which is provided by the author of the library, but we wanted to be independent of that, so we decided to provide the rasterisation service by our own server code.

At first, it seemed very straight-forward, but quite some time was absorbed by the confusion about Tomcat's class-path. It turns out Tomcat only looks in `WEB-INF/lib/*` and not in its subdirectories, resulting in a huge stack of `NoClassDefFoundErrors` to fix.

Overtime

It's obvious that in this iteration we went far over our estimated time. This is due to several factors.

Firstly we found ourselves in an awkward situation when we found the simulator was not working correctly, we set out to fix this, , and because this was an issue that could not be pushed back to another SCRUM iteration, we also set out to write our own solver as an insurance. We managed to fix both the 3rd party solver and complete our own. This was of course a good outcome, but it did take a big bite out of our time expenditure.

Furthermore, this iteration we really started performing the acceptance testing, which brought up some bugs, which required fixing.

Also, because this was quite a long iteration and we really wanted to deliver a great product, we pushed very hard at implementing some should-haves/nice-to-haves.