

Sprint Retrospective, Iteration 4						
Context Project: Programming Life						
Group: PL-2 (Pantzerfaust)						
User story	Task	Assignee	Estimated effort (hours)	Actual effort (hours)	Done (Yes/No/Partly)	Notes
As a user, I must be able to see the whole graph in the upper-most view.	Show the amount of difference between bubbles in graphs	Cas	8.0	8.0	Partly	Still needs a lot of attention
	Create 'random' bubbles to be able to show top level overview	Cas	10	10	Partly	Works for straight sequences
	Modify the data containers of the current graph comparison algorithm, so further calculates can be done on the graph	Faris	2.0	1.5	Yes	
As a user, I want to load my data in an instantaneous manner	Design persistent database structure to store the 'default' representation of the graph	Justin & Wouter	8.0	Justin - 1.5 Wouter - 1.0	Partly	Got code that uses a (mysql) database as backend. See problem 2, the whole data-server part is delayed / scrapped for now. Also, the user story is "done" (for 328 graph), as loading the data is fast enough.
	Setup Streaming Framework to support streamed parsing	Justin	4.0	0.0	No	See above
	Design API and queries to provide to the GUI	Wouter	8.0	0.0	No	Problem 1 and 2 combined to this not happening. Furthermore, this task is completely cancelled
	Profiling and Performance Improvement	Justin & Wouter	12.0	Justin 1.0	No	See problem 1&2. Got some tooling to run to profile our application but after "problem" 2, finding bottlenecks is not a big priority anymore.
As a user, I must be able to zoom in into the bubbles	Finish zooming algorithm for bubbles	Casper	8.0	8.0	No	Still some issues with the zooming
	Refactor code relate to filtering/zooming bubbles	Casper	8.0	8.0	No	Due to the fact that there were still some issues with the zooming, the refactoring hasn't been done either yet
As a user, I want to be able to see the phylogenetic tree next to the graph so I know which genomes I am comparing.	Create a better transition fom the tree to the graph. The tree should still be visible when viewing the graph, so the user can see what is being compared in the graph.	Faris	4.0	5.0	Yes	
As a user, I want to be able to easily distinguish edges containing many genomes from edges containing few genomes	Make the thickness of the edges containing many genome paths thicker than the edges containing few genome paths.	Faris	1.5	2.0	Yes	

As a user ,I want to be able to distinguish nodes containing many bases (ATGCs) from nodes containing few bases	Make the nodes containing many bases bigger than the nodes which contain few bases.	Faris	1.5	1.0	Yes	
As a user, I want to be able to compare any pair (or set) of random genomes which I can select in the graph	Make it possible to select an arbitrary set of nodes to compare	Faris	3.0	0.0	No	This will be moved to the next sprint. It will be implemented more generically (so it can be used for multiple kinds of functionality). See next sprint plan: Dragging nodes from the tree to the graph. Instead of this I have improved the performance of the parser and algorithms (this is actually more important, as our application was pretty slow before. The 328 genome set can be loaded in a couple of seconds now!). We only noticed that this was an issue after we started working on some of the other tasks of this sprint for which the dataset of 328 genomes was preferred, because it gives a better image of the usability of the software than the set of 10 does.
As a user, I want to be able to see the evolutionary distance between different children in the phylogenetic tree	Use the edge length of the phylogenetic tree in the user interface.	Faris	4.0	3.0	Yes	
As a developer, I want to get an accurate analysis of my code	Exclude 3rd party code from analysis tools	Wouter	1.0	1.0	Partly	Mostly spent time on fixing other issues in the build jobs.
As a developer, I want to work with maintainable code	Refactor SplitGraphs and test it	Wouter	7.0	7.0	Yes	Done in response to the shift in priorities
Main Problems Encountered		Adjustments for the next Sprint				
		1. Make sure that all group members split high priority tasks evenly, so that in case of sickness, not everything is stalled.				
Problem 1	Sickness during Sprint					
Description:	Wouter fell sick.					
Reaction:	His low priority tasks were postponed, high priority tasks were reallocated.					
		Motivate any adjustments that will be made for the next Sprint.				
Problem 2	At the start of the sprint, priorities shifted heavily.	1. This is mostly a preventive measure, because it did not really happen this sprint. However, we have seen the effect of losing man power and do not want this to affect us more than necessary.				

Description:	Because of a heavy refactoring done by Faris, the application ran very smooth even without data preprocessing. This made putting huge effort in data parsing less desirable, contrary to the sprint plan.					
Reaction:	we changed the planning rather than stubbornly following it to address the current needs of the client and the development process.					