Context Project: Multimedia Services		Reflection on Iteration #7						
Group: 1								**12-6-2015**

User Story	Issue #	Summary	Assignee	Estimated Effort — Storypoints (1 sp ~ 2 hours)	Actual effort — Hours	Done Y/N	Done By	Notes
As a user I want to get rewars for classifying songs.	Core-129	Make points available in the backend.	Gijs	3	4	Y	Gijs	Because of the heavy lifting on the backend and the blocking nature of this ticket for the other tickets in the sprint, this is not as well tested as could be. It will be tested more thoroughly in the next sprint.
	Frontend-45	Make the system use points in the frontend.	Eva	2	8	Y	Eva	This ticket took more time than expected, due to the fact that fetching from the backend costed a lot of time. This was caused by still a lack of angular experience and the persistance loggin of soundcloud, wich changed the user.
	Core-130	Make a leaderboard in the backend	Gijs	1	1	Y	Gijs	
	Frontend-46	Make a leaderboard in the frontend	Eva	2	5	Y	Eva	
	Frontend-47	Use an oath token to keep logged in with soundcloud	Jan-Willem	4	8	Y	Jan-Willem, Gijs	
	Core-114	Refactor the votes of a room	Jan-Willem	2	5	Y	Tim	Was swapped with Frontend-39
	Frontend-48	Make a button for the classify page	Eva	1	2	Y	Eva	
As a user I want to be logged in into the system	Core-113	Create a user log-in structure	Jan-Willem	3	6	Y	Jan-Willem	
	Core-113	Test the user log-in structure in order to verify the behaviour	Tim	-	3	Y	Tim	Personally I had trouble understanding the complex changes, but after I understood the code I was able to test them quite efficiently
As a developer I want to have good datastructure to make the system scalable.	Core-84	Implement the R-tree	Tim Jan-Willem (pair)	5	15	Y	Jan-Willem, Tim	It took a lot of time to get R-Trees to work under both Postgres (production) and H2 (test) using QueryDSL and Hibernate.
As a user I don't want to hear only a few songs.	Core-128	Fix the play history	Jaap	1	8	Y	Jaap, Jan-Willem	RoomInstances had to be completely reworked because threads should not interact Hibernate proxy objects outside transactions.
As a user, I want the system to play the next song after the previous song is finished.	Frontend-39	Synchronize the music with the room.	Tim	3	4	Y	Jan-Willem	Was swapped with Core-114

		Reflection on Iteration #7						
Context Project: Multimedia Services								
Group: 1								**12-6-2015**
				Estimated Effort	**Actual effort**	**Done**		**Notes**
					Hours	**Y/N**	**By**	
User Story	**Issue #**	**Summary**	**Assignee**	**Storypoints (1 sp ~ 2 hours)**				
As a user I want to have more information about a song.	Frontend-49	Show information of the song on the page.	Jaap	4	10	Y	Jaap	Learning Angular went smooth, however it took some time to understand the inner working of all services. I have struggled with getting the current song, sound and room consistent. Within this ticket I thought it would be easy and a nice performance boost to replace the fetching of metadata from SoundCloud with the already retrieved metadata from the backend, but this should have been another ticket/PR.
Poll the chat, so new messages are shown.	Core-131	Make chat polling availible (backend)	Gijs	2	8	Y	Gijs, Jan-Willem, Tim	This ticket became very big and after a walk in the park, we came up with a suitable and easier solution which in the end was more managable
	Frontend-50	Implement chat polling (frontend)	Eva	1		N		Postponed to the next sprint because the changes on the backend took longer than expected.
Emergent Architecture Design Report with the latest changes.	Main-76	Update the Emergent Architecture Design Report with changes.	Jaap	1	2	Y	Jaap	Changed some of the backend text, added SoundCloud integration, updated ER diagrm. Went smooth.
As a group member, I want to have a sprint reflection on the last sprint and a sprint planning for the upcoming sprint.	N/A	Write sprint reflection for this sprint.	Lead by Gijs	1	3	Y		We have had a constructive discussion regarding the past sprint and the summary can be read in the notes
	N/A	Write sprint planning for upcoming sprint.	Lead by Gijs	2	4	Y		
As a group, I want to schedule SCRUM meetings at the start of the day (9:00 AM).	N/A	We will have daily SCRUM meetings.	Lead by Gijs	3	6	Y		
As a group, I want to schedule a meeting with the client at the end of the sprint.	N/A	We will have a meeting with our context manager, software engineering teaching assistant and context assistant (our main client).	Lead by Tim	1	2	Y		
As a development team, when a pull-request is submitted, we want to review the changes supplied in order to merge the pull-request.	N/A	Every pull-request will be reviewed by at least 2 group members. They placed comments for code updates and accepted the pull-request when it is ready to be merged.	Lead by Jan-Willem	8	20	Y		Due to the higher number of pull-requests, the review process took a bit more time than expected
			Total:	**50 sp**	**124**			

Context Project: Multimedia Services		Reflection on Iteration #7						
Group: 1								**12-6-2015**

User Story	Issue #	Summary	Assignee	Estimated Effort Storypoints (1 sp ~ 2 hours)	Actual effort Hours	Done Y/N	By	Notes

Notes

This sprint we were really ambitiuous to implement a lot of new features. However, in a very early stage of the development, we found out that the backend was not working as intended. This is because we had to modifiy the responsibilities for the backend components and because the backend was dependent on objects that it was not allowed to interact with (it interacted with Hibernate entity proxies from a separate thread, outside of transactions- which does not really work). Alltogether, the changes were quite fundamental and we had to restructure quite some things in the backend. These parallel changes in both adding new features, refactoring existing features and rewriting most of the backend caused branches to diverge quite a lot.

This could have been prevented by putting the focus on the refactorings and suspend work on the features. We have however decided to continue work on these features, because we felt quite a press to do so as we are rounding the project. This caused merges to be rather intensive and also introduced some bugs and a not so smooth demo.

For the next sprint we decided that we will reject commits / pull requests that fixes two things (shotgun surgery), because merging gets really complicated when it's not clear why a line has changed and because we have introduced a few bugs by this in the past. Remaining substantional refactorings (we have to move some code around in the frontend) will be realized within a "code lock" (the weekend), so that commits that touches refactored code can be merged with ease.