| Context Project: Multimedia Services Group: 1 | | Reflectio | | | | | | |
|--|------------------------------------|---|-----------------------------------|------------------------------------|---------------|-----|----------------------------|---|
| | | | | | | | | 22-5-2015 |
| | Github/ Waffle.io | | | Estimated Effort | Actual effort | | Done | |
| User Story | Issue # (sorted by priority) | Summary | Task Originally Assigned To | Storypoints (1 sp ~ 2 hours) | Hours | Y/N | Ву | Notes |
| to have a clear | 47-Main | Continue discussion on API contract | Jaap | 5 | 10 | Υ | Jaap | Has been discussed with the team, we have a clear contract now. |
| calls between the | 64-Core | Update room selection API | Gijs | 1 | 4 | Υ | Gijs | Took longer due to some research on filters. |
| User Story As a developer, I want to have a clear interface of the API calls between the frontend and backend. As a user, I want to classify songs based on arousal and valence to improve the system. As a user, when I select a mood, I want to see rooms that are relevant to my current mood. As a user, when a song ends, I want the system to play the next one. As a developer, I want to improve the user interaction on the website. | 14-Frontend | Update room API calls in frontend | Tim | 1 | 2 | Υ | Tim | |
| | 63-Core | Update chat API backend | Gijs | 1 | 2 | Υ | Gijs | Messages are now stored via the room api, as this is a logical part of the room. |
| | 15-Frontend | Update chat API calls frontend | Eva | 2 | 12 | Y | Eva | Took longer due to a lack of angular experience and endpoints that where not properly set. |
| classify songs based | 19-Frontend | Implement frontend function thumbs up and down. | Eva | 3 | 6 | Υ | Eva | |
| | 20-Frontend | Implement frontend function SAMs. | Jan-Willem | 2 | 4 | Υ | Tim | |
| the system ' | 83-Core | Update song vector based on user feedback. | Gijs | 3 | 6 | Υ | Tim | |
| select a mood, I want | 84-Core | Implement nearest neighbours to find songs fitting the current room based on their vector distance. | Tim | 2 | 4 | N | - | Has been pushed to next sprint due to time constraints |
| relevant to my current | 85-Core | Retrieve song data from database. This should also consider the input in the database. | Jaap | 3 | 2 | Υ | Jaap | This issue was already largely covered by the framework |
| song ends, I want the system to play the next one. | | | | | | | | |
| | 86-Core | Create room instances | Jan-Willem | 3 | 6 | Υ | Jan-Willem | |
| to improve the user | 21-Frontend | Style pages and UI elements | Tim | 5 | 10 | Y | Tim, Gijs (pair) | |
| interaction on the | 12-Main | Implement UI states and state transitions | Tim | 2 | 4 | Y | Tim, Jan- Willem (pair) | |
| | 12-Main | Fix duplication bug in state transformation | Tim | 1 | 2 | Υ | Tim | This bug was discovered after implementing the states, causing some delay. The root cause was quickly discovered and a fix had been created |
| | 12-Main | Implement UI state params for rooms, merges | Jan-Willem | 2 | 4 | Υ | Jan-Willem | |

| Context Project: Multimedia Services | | Reflectio | | | | | | |
|--|------------------------------------|---|-----------------------------------|------------------------------------|---------------|-----|---------------------------------------|-------|
| Group: 1 | | | 22-5-2015 | | | | | |
| | Github/ Waffle.io | | | Estimated Effort | Actual effort | | Done | |
| User Story | Issue # (sorted by priority) | Summary | Task Originally Assigned To | Storypoints (1 sp ~ 2 hours) | Hours | Y/N | Ву | Notes |
| As a developer, when I am testing the system I want data to use while manual testing the application | | | | | | | | |
| | 62-Core | Bulk insert song data into database | Jaap | 3 | 6 | Υ | Jaap | |
| As a group member, I want to have a sprint reflection on the last sprint and a sprint | | Write sprint reflection for this sprint | Lead by Gijs | 1 | 2 | Υ | Team effort lead by Gijs | |
| planning for the upcoming sprint | | Write sprint planning for upcoming sprint | Lead by Gijs | 2 | 4 | Y | Team effort lead by Gijs | |
| As a group, I want to schedule SCRUM meetings at the start of the day (9:00 AM) | | We have had daily SCRUM meetings | Lead by Gijs | 3 | 6 | Y | Team effort lead by Gijs | |
| As a group, I want to schedule a meeting with the client at the end of the sprint | | We have had a meeting with our context manager, software engineering teaching assistant and context assistant (our main client) | Lead by Tim | 1 | 3 | Y | Team effort lead by Tim | |
| As a development team, when a pull-request is submitted, we want to review the changes supplied in order to merge the pull-request | | Every pull-request has been 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 | 15 | у | Team effort lead by Jan- Willem | |
| | | | Total: | 54 sp | 114 hours | | | |

| Context Project: Multimedia Services Group: 1 | | 3 | Reflectio | n on Iterati | ion #4 | | | | 22-5-20 | 15 |
|---|------------|------------------------------------|-----------|-----------------------------------|------------------------------------|---------------|-----|------|---------|----|
| | | Github/ Waffle.io | | | Estimated Effort | Actual effort | | Done | | |
| | User Story | lssue # (sorted by priority) | Summary | Task Originally Assigned To | Storypoints (1 sp ~ 2 hours) | Hours | Y/N | Ву | Notes | |

Reflection

Pull requests were open for too long. Because of this the merges we had to perform were way more complicated than they needed to be.

We agreed on everyone finding a point in their day to check if there are PRs open for review. This way we are confident a smoother workflow can be achieved, which should save us a lot of stress.

We had an issue where we had a Pull Reguest that was agreed to be okay. After that, another commit was pushed on the PR that has not been reviewed again. This actually introduced a bug in the master. We agreed that this should not be allowed again.

We broke some API's by changing method signatures that other branches still depended on. This has not been communicated properly, which caused problems while merging. We have to decided to "lock" API's and create new methods instead. If we then decide to remove deprecated code, we should discuss this with the team.

Jan-Willem made less hours on planning, but spent more time reviewing PRs than others.

We need to have a more even distribution, so as mentioned above, we all are going to review PRs more intensive.

The work on the database insertion clashed with other teammembers testing the frontend, which was already linked to the database.

The changes in the entities on the way down were causing problems too. Some field names were changed and this needed to be changed in the frontend, backend and database.

This was eventually fixed when the pull request of the updated entities was merge, the new master was then merged into the bulkrequest PR.

Now the system is consistent and the database schema will suffice for some time.

None of the individual tickets on this sprint (sprint backlog, started, done), have more than 3 story points. This is less than half of the maximum ticketweight of past weeks.

In other words: the granularity has improved.