

Sprint 24

Team Capacity Start: 92hrs

- Joseph: 16hrs
- Joshua: 8hrs
- Rhoy: 16hrs
- Frank: 20hrs
- Ghabe: 16hrs
- David: 16hrs

Work items

Work item	Hours	Name	Goal Date
Reputation Implementation pt 3	16	Rhoy	04/23/23
Services Implementation	20	Frank	04/23/23
Picture Upload Implementation	16	David	04/23/23
Events Implementation	16	Joseph	04/23/23
Refactor Back end	8	Ghabe	04/23/23
Ui Core Implementation	8	Ghabe	04/23/23
Account Deletion	8	Josh	04/23/23

Total: 92

FileUpload Implementation Task Breakdown - David

Task	Hours	Notes
Implementation Manager	3	
Implementation Services	2	
Implementation SqlDAO	2	
Implementation Entry Point	2	

Add file buttons to map frontend	3	
Add end to end test	4	

Total: 16

Reputation Implementation Task Breakdown - Rhoy

Task	Hours	Notes
Implementation "View Reports Functionality"	2	In order for this functionality to be effective I'm trying to figure out how to partition the data from the database to display it back to the view so that the server won't slow down when users request so see reports
Implementation Controller	1	Since I started implementing the controller yesterday I don't think it should take very long but I still don't know how to do any configuration files or dependency injection. Basically any data that the later layers needs to know in order to function properly.
Implementation Testing in backend	2	I used the ReputationUnitTest.cs to ensure that the functionalities I worked on were working in the most simplest conditions. But I have to refactor it so that it tests the things listed on the BRD such as finish executing before reaching a maximum duration.
Implementation Front-end	6	By far the most challenging thing I'll need to figure out before the end of this week. Was not able to start on it last sprint because I was getting most of the backend done. But hopefully by referencing everything that's already been done I'll be able to get the most simplest views to work with what I've done in the backend.
Implementation end-to-end testing	2	Since I'm also not entirely sure what an e2e test is supposed to do I think I'll be stuck on it for a bit but I'm making sure not to waste any time on things I don't know and instead work on the things I do know.
Refactoring and Reviewing	3	Only if I'm able to complete all the above task items in timely manner will I try and review everything to see if I'm missing anything related to the coding standards that the professor expects us to follow but as far as I'm aware I'm already following it to the best of my ability.

Total: 16

User Services Implementation Task Breakdown - Frank

Task	Hours	Notes
Write out tests	2	Writing out each of the test cases may take shorter than the 2 hours that I think it will take but this is for all tests unit and integration.
Setup controller	2	I am not exactly well versed in what needs to be done in order to get the controller working properly so I will be taking queues from the controllers that are already implemented.
Database Implementation	2	I have an idea of how the database should be setup but I think that I may need some time to make sure that each of the data tables make sense.
Data Access Implementation	3	This will likely be one of the longer implementations to be completed it holds most of the connection to the database so all the data needs to be transferred here.
Manager Implementation	3	I think this will probably where I spend most of my time, solving any errors or issues will probably lead me back to here.
Service Implementation	2	The service layer implementation does not seem to hard as the value this layer has is mostly to check if the data is valid when returning from the database.
Finish Front-end Implementation	2	I have most of the frontend html completed it just needs to be hooked up with the js so that it can connect to the controller. I am also missing the admin view.
Implementation end-to-end testing using cypress	2	I honestly have no idea how cypress works so it will take some time and learning from what my team members have done to get this completed.
Bug fixing deployment	2	I don't know the exact number on this but this is something that needs to be accounted into somewhere.

Total: 20

Conclusion/summary	Name
I honestly think that this is the idealized vision of completing this in the shortest	Frank

amount of work hours. I think total hours is somewhere around 30-32 hours like was originally in the project plan. I am going to try and get this done because I have to.

Refactor back end Task Breakdown - Ghabe

Task	Hours	Notes
Account Controller Refactor	2	
Analysis Controller Refactor	2	
Recovery Controller Refactor	2	
Make sure tests still work	2	

Total: 8

Ui Core Implementation Task Breakdown - Ghabe

Task	Hours	Notes
Implementation usermanagement.js	2	
Add integrate usermanagement view	2	
Implement display current user	2	
create e2e test	2	

Total: 8

Events Implementation Task Breakdown - Joseph

Task	Hours	Notes
Services Layer	2	
Manager Layer	2	
Unit Test	2	
Integration test	2	
Ui	2	Familiar with html and js structure, I can probably make it in 1 hr
Controller	2	Have an idea how it will work, but might cost more hrs

Service.js	3	although Im familiar with JS, it may require more work with handling and presenting data in the front end.
end to end test	2	not familiar with technology so it may require even more hours to accomplish it. Hoping that 1hr is enough to get the basic down.

Total: 18

Conclusion/summary	Name
Since my capacity is at 16 hours I will increase it to 18hrs.	Joseph

Account Deletion Task Breakdown - Josh

Task	Hours	Notes
Account Deletion Controller Refactor	2	Need to adjust the controller to meet coding standards
Adjust Views	2	Views are mostly completed. Need restricted access
Testing	2	Code needs to be retested and evaluated as working code

Total: 4

Alerts Implementation Task Breakdown - Josh

Task	Hours	Notes
Fix controller	1	Coding standards are not met.
Make zip codes work	1	Process has been annoying but I believe that it shouldn't be too long
Testing	2	Need time to test feature

Total: 4

Sprint Conclusion

Original items:

Team Capacity: 92hrs

- Joseph: 16hrs
- Joshua: 8hrs
- Rhoy: 16hrs
- Frank: 20hrs
- Ghabe: 16hrs
- David: 16hrs

Work Items

- File Upload Implementation | David | 16
- Reputation Implementation part 3 | Rhoy | 16
- User Services Implementation | Frank | 20
- Refactor back-end | Ghabe | 8
- Core Components Ui | Ghabe | 8
- Events Implementation | Joseph | 16
- Account Deletion | Josh | 8

Total | 92

After Task Breakdown:

Team Capacity: 94hrs

- Joseph: 18hrs
- Joshua: 8hrs
- Rhoy: 16hrs
- Frank: 20hrs

- Ghab: 16hrs
- David: 16hrs

Work Items

- File Upload Implementation | David | 16
- Reputation Implementation part 3 | Rhoy | 16
- User Services Implementation | Frank | 20
- Refactor back-end | Ghab | 8
- Core Components Ui | Ghab | 8
- Events Implementation | Joseph | 18
- Account Deletion | Josh | 4
- Alerts Implementation | Josh | 4

Total | 94

In conclusion our team capacity is equal to our own capacity. Since we are unable to help each other, we are all required to beef up on our hours in order to be in time of code review for next week. Since 5/6 of us are working on our own app specific features , that leaves Ghab in anchoring the team by picking up core components that are not fully finished.