Sprint Planning 11/17 - 12/03 (5 days) (James)

Progress Meeting

      11/30 - 5pm

      12/1 - 7pm

      12/3 - 3pm & 7pm

Capacity:

Member	Available Hours	Assigned Hours
Blake	5	5
Isabel	10	10
Jacob	10	9.5
James	9	7.5
Naeun	8	5
Total	**42**	**37**

Events:

Blake	- Perpetual work throughout the entire sprint
Isabel	- Work Thursday (12/2)
Jacob	- Midterm (11/30)
James	- Midterm (11/30) - Project (12/3)
Naeun	- Term project due (12/2)

Priority:

1. User Management (Assignee: Isabel)
2. Tech Approvals (Assignee: James)
3. Logging (Assignee: Jacob)

Capacity Check: Good to GO!

      Total Story Point Hours = 37 Hours

Epic: Low Level Design					
Story: User Management CRUD (Assignee: Isabel) 25 Hr					
Tasks	Priority	Story Point (Hours)	Internal Due Date	Assignee	Description
CREATE (Assignee: James) 3.5 Hr					
~~CREATE Operation Success Diagram~~	16	1	11/30	James	
CREATE Operation Failure Cases Diagrams	17	2	12/1	James	
CREATE Operation Logging	24	0.5	12/1	James	
UPDATE (Assignee: Isabel) 6 Hr					
UPDATE Orientation Success Diagram	18	2	12/3	Isabel	
UPDATE Operation Success Failure Cases	19	2	12/3	Isabel	
UPDATE Operation Logging	25	2	12/3	Isabel	
DELETE (Assignee: Jacob) 5.5 Hr					
DELETE Operation Success Diagram	20	0.5	12/3	Jacob	
DELETE Operation Failure Diagram	21	5	12/3	Jacob	
DISABLE/ENABLE (Assignee: Naeun) 4 Hr					
DISABLE/ENABLE user account success diagram	22	2	12/3	Naeun	
DISABLE/ENABLE failure diagram	23	2	12/3	Naeun	
Sequence Diagram Fixing (Blake) 3 Hr					

~~Create sequence diagram for password Management~~	~~3~~	~~2~~	~~12/01~~	~~Blake~~	~~Sequence diagram is more specific version of UML~~
~~FIX SPECIFICITY OF UML DIAGRAMS Password Management~~	~~4~~	~~1~~	~~12/02~~	~~Blake~~	~~UML is incorrect refer to office hours video to figure out how to fix~~
Testing (Assignee: Blake)					
Reflect Test Cases from Test Plan in Diagrams					
Associate Test Cases to Each Method Signatures					
Code (Assignee: Blake) 2 hr					
Create Method Signatures for Each & Every Function	26				
Create Pseudo Code for Each Function	27				
Logging	29	1	12/01	Blake	Based on Sequence Diagrams
User Management	28	1	12/01	Blake	Based on CREATE, READ, UPDATE, DELETE diagrams

Epic: Tech Approvals (James) 16 Hr					
Tasks	**Priority**	**Story Point (Hours)**	**Internal Due Date**	**Assignee**	**Description**
Story: Programming Languages (Assignee: James) 4 hr					
Revise Multiplier	1	0.5	12/2	James	
Revise Metric Explanation	2	0.5	12/2	James	
Test Programming Languages for Better Scores/Metrics	3	2	12/3	James	

Final Recommendation Paragraph	4	1	12/3	James	
Story: Network (Assignee: Isabel) 4 hr					
Add quantitative data for EC2	5	1	12/3	Isabel	
Add quantitative data for MariaDB	6	1	12/3	Isabel	
Final Recommendation Paragraphs	8	1	12/3	Isabel	
Add the Tech Approval for Certifications	7	1	12/3	Isabel	
Story: API (Assignee: Jacob) 4 hr					
Conclusion : Tech Approval API NHTSA	10	1	12/3	Jacob	
Revisal of Multiplier/Explanation : Tech Approval API NHTSA	11	1	12/3	Jacob	
Conclusion : Tech Approval API Email Client	9	1	12/3	Jacob	
Revisal of Multiplier/Explanation : Tech Approval API NHTSA	12	1	12/3	Jacob	
Story: Affiliate Programs (Assignee: Naeun) 1 hr					
Update Tech Approval (Affiliate Programs)	13	1	12/3	Naeun	Add 1 more competitor
Story: Testing (Assignee: Blake) 3 hr					
Fix Testing Tech Approval charts so that it is scored how the others are scored	14	1	11/30	Blake	Change ranking criteria
Add Unit Testing Tech Approval	15	2	12/1	Blake	Look at GTest and C# Libraries