Sprint 7

Sprint Planning for Sprint 6 - 04/17/24:

- Interval from 04/17/24 to 05/06/24

Individual Capacities:

- Unit measurement: 1 point to 1 hour

Team member:	Capacity by hours
Julie	50
Jason	40
Kihambo	35
Diego	35
Parth	40
Khuong	40
An	40
Total	280

Work from Backlog:

Work Item	Effort Points (Hours)	Owner
Fixing ui interface	15	Julie
Update backend tests	15	Julie
Connect features that need to be connected	8	Julie
Analysis of DashBoard database and backend	30	Jason
Regression testing	15	Kihambo
Collab Feature	20	Kihambo
Regression testing	25	Diego
Price Range Sorting	10	Parth

Item Creation/Listing/Offering	37	An
Seller Dashboard: Product Editing, Product Deletion, Inventory Stock List, and Financial Progress Report	50	Khuong
End-to-end testing	10	An
Analysis dashboard frontend	30	Parth
End-to-end testing	10	Khuong

Task Breakdown:

Julie's Work Item: (Internal Due Date): 03/25

Fixing UI

1.Fix navbar to only have login and register and recovery (5 hours)

2. Have login route to musicali vs craftverify select website (5 hours)

3. Route all features to new musicali nav bar (5 hours)

Update Backend test

1. Update current existing test (7 hours)
2. Add any missing tests needed (8 hours)

Connect all features to work together

1. Connect collab feature to work with Portfolio (8 hours)
2. Add functionality for user profile correctly to each wesit (7 hours)

Estimate = 45

Kihambo's Work Item Collab Feature: (Internal Due Date): 4/20

1. Redesign the Collab Feature with a HLD and LLD
    a. For higher complexity and better scaling
    b. Make it as meticulous as possible
    c. Make a clear flow so it's easy to understand
2. Modify the existing code in the backend to fit the design
    a. In the services and data access layer
3. Create test cases for Collab Feature for each user story in the BRD
    a. Based on the notes from the last code review, I didn't have any so this time I'm including them
    b. I'll also do this for Scale display
4. Create the frontend button for Collab Feature
5. Debug any errors and warnings

Estimate = 20

Kihambo's Work Item Regression Testing: (Internal Due Date): 5/04

1. Inspect code changes in different features
2. Run tests for each feature
3. If any code changes affect the running tests, I'll fix them
4. Commit and push the fixed changes in the code to Github repository

Estimate = 15

Diego's Work Item: Bingo Board

(Internal Due Date 4/19):

1. Finish Bingo Board for code review
    a. Add page functionality
    b. Fix display issues when switching features
    c. Add interest functionality
2. Regression Testing
    a. Update test cases

Estimation = 35 hrs

Jason's Work Item: (Internal Due Date): 4/13

Analysis Dashboard Database and Backend:

1. Backend for Key Performance Indicators (KPIs) (12 hours total)

    a. The number of success and failed login attempts per month over selected time span (trend chart) (2 hours)
    b. The number of success and failed registrations per month over selected time span (trend chart) (2 hours)
    c. Top 3 longest page visits in seconds (2 hours)
    d. Top 3 most used feature (2 hours)
    e. First application specific feature metric (2 hours)
    f. Second application specific feature metric (2 hours)
2. Controllers for Key Performance Indicators (KPIs) (6 hours total)
    a. The number of success and failed login attempts per month over selected time span (trend chart) (1 hour)
    b. The number of success and failed registrations per month over selected time span (trend chart) (1 hour)
    c. Top 3 longest page visits in seconds (1 hour)
    d. Top 3 most used feature (1 hour)
    e. First application specific feature metric (1 hour)
    f. Second application specific feature metric (1 hour)
3. Unit Testing for Database/Backend for Key Performance Indicators (KPIs) (6 hours total)
    a. The number of success and failed login attempts per month over selected time span (trend chart) (1 hour)
    b. The number of success and failed registrations per month over selected time span (trend chart) (1 hour)
    c. Top 3 longest page visits in seconds (1 hour)
    d. Top 3 most used feature (1 hour)
    e. First application specific feature metric (1 hour)
    f. Second application specific feature metric (1 hour)
4. Bugfixing/manual testing for Database/Backend for Key Performance Indicators (KPIs) based on Unit Testing (12 hours total)
    a. The number of success and failed login attempts per month over selected time span (trend chart) (2 hours)
    b. The number of success and failed registrations per month over selected time span (trend chart) (2 hours)
    c. Top 3 longest page visits in seconds (2 hours)
    d. Top 3 most used feature (2 hours)
    e. First application specific feature metric (2 hours)
    f. Second application specific feature metric (2 hours)

Estimation = 36 hours

An's Work Item: (Internal Due Date): 04/15

Item Creation, Listing, and Offering:

1. Item Creation(I.C.):
    a. Revise LLD for I.C. (1 hour)
    b. Create Backend Codes for Item Creation (5 hours)
        i. Data Access Layer Code
        ii. I.C. Service Code
        iii. API endpoints Code
    c. Create Frontend Codes for Item Creation (5 hours)
        i. I.C. HMTL Code
        ii. I.C. JS code
        iii. I.C. CSS Code
2. Item Listing(I.L.):
    a. Revise LLD for I.L. (1 hour)
    b. Create Backend Codes for Item Listing (6 hours)
        i. Data Access Layer Code
        ii. I.L. Service Code
        iii. API endpoints Code
    c. Create Frontend Codes for Item Listing (8 hours)
        i. I.L. HMTL Code
        ii. I.L. JS code
        iii. I.L. CSS Code
3. Offering:
    a. Revise LLD for Offering. (1 hour)
    b. Create Backend Codes for Offering (10 hours)
        i. Data Access Layer Code
        ii. Offering Service Code
        iii. API endpoints Code
    c. Create Frontend Codes for Offering (10 hours)
        i. Offering HMTL Code
        ii. Offering JS code
        iii. Offering CSS Code
4. End-To-End Test:
    a. Perform End-To-End Tests with Khuong (10 hours)

Parth's Work Item: (Internal Due Date): 04/15

Analysis Dashboard Frontend:

1. Planning and Design (6 hrs)

- Research similar dashboards for design inspiration.

- Create wireframes for the layout of the dashboard.
- Decide on a color scheme and fonts.

2. Environment Setup (2 hrs)
    - Set up the development environment (IDE, version control).
    - Configure frontend development tools and libraries (e.g., React.js, Angular).
3. Core Development (16 hrs)
    - Implement the main components of the dashboard:
        - Header/Navbar
        - Data display widgets (charts, graphs, tables)
        - Filters and search functionalities
    - Integrate state management for handling data changes (Redux, Context API).
4. Interactivity and Functionality (4 hrs)
    - Develop interactive elements like buttons, sliders, and dropdown menus.
    - Ensure components are dynamically updated with the backend data.
5. Testing and Debugging (2 hrs)
    - Perform unit and integration tests.
    - Debug and fix any issues.

Estimate = 30 hrs

Price Range Sorting:

1. Research (1 hours)
    - Which features are required to make the price range sorting efficient and ensure the user finds their product faster?
2. Planning and design (1 hour)
    - I want a detailed specification document for the price range sorting feature, including user stories, acceptance criteria, and technical requirements, so the development team has clear guidelines.
3. Front-End Development (3 hours)
    - I need to ensure that the price range sorting interacts seamlessly with the product listing page, updating the displayed products according to the selected price range.
4. Back-End Development (3 hours)
    - I need to implement the logic that filters products within the specified price range, ensuring efficient database queries for a fast response time.
    - I want to create a robust API endpoint that supports price range filtering,

enabling the front end to retrieve filtered products faster.

5. Testing and quality assurance (1 hour)
   - I need to write comprehensive unit tests for the price range sorting feature to ensure it works as expected under various scenarios.
   - I want to test the price range sorting feature in a real-world environment to ensure it meets user needs and is free of bugs.
6. Deployment (1 hour)
   - I need to prepare the deployment environment, ensuring all necessary configurations are in place for the price range sorting feature.

Estimate = 10 hrs

Khuong's Work Item: (Internal Due Date): 04/15

Product Editing, Product Deletion, Inventory Stock List, and Financial Progress Report :

1. Item Modification:
   a. Create LLD for Item Modification (1 hour)
   b. Create Backend Codes for Item Modification (5 hours)
      i. Data Access Layer Code
      ii. Item Modification Service Code
      iii. API endpoints Code
   c. Create Frontend Codes for Item Modification (5 hours)
      i. Item Modification HMTL Code
      ii. Item Modification JS code
      iii. Item Modification CSS Code
2. Item Deletion:
   a. Create LLD for Item Deletion (1 hour)
   b. Create Backend Codes for Item Deletion (4 hours)
      i. Data Access Layer Code
      ii. Item Deletion Service Code
      iii. API endpoints Code
   c. Create Frontend Codes for Item Deletion (4 hours)
      i. Item Deletion HMTL Code
      ii. Item Deletion JS code
      iii. Item Deletion CSS Code
3. Inventory Stock List:
   a. Revise LLD for Inventory Stock List. (3 hour)
   b. Create Backend Codes for Inventory Stock List(5 hours)
      i. Data Access Layer Code
      ii. Inventory Stock List Service Code
      iii. API endpoints Code
   c. Create Frontend Codes for Inventory Stock List(5 hours)

i. Inventory Stock List HMTL Code
ii. Inventory Stock List JS code
iii. Inventory Stock List CSS Code

4. Financial Progress Report:
    a. Revise LLD for Financial Progress Report (3 hours)
    b. Create Backend Codes for Financial Progress Report (7 hours)
        i. Data Access Layer Code
        ii. Financial Progress Report Service Code
        iii. API endpoints Code
    c. Create Frontend Codes for Financial Progress Report (7 hours)
        i. Financial Progress Report HMTL Code
        ii. Financial Progress Report JS code
        iii. Financial Progress Report CSS Code
5. End-To-End Test:
    a. Perform End-To-End Tests with An (10 hours)

Total Estimate = 307 hours from backlog

Individual Breakdown

Julie Reyes

Work Item	Task	Estimation in pts	Team Confirmation
Fixing ui	Fix navbar to only have login and register and recovery	5	AN, DG,KN,PN, JR, JJ, BM, SK
	Have login route to musicali vs	5	AN, DG,KN,PN, JR, JJ, BM, SK

	craftverify select website		
	Route all features to new musicali nav bar	5	AN, DG,KN,PN, JR, JJ, BM, SK
Update Backend test	Update current existing test	4	AN, DG,KN,PN, JR, JJ, BM, SK
	Add any missing tests needed	4	AN, DG,KN,PN, JR, JJ, BM, SK
Connect all features to work together	Connect collab feature to work with Portfolio	6	AN, DG,KN,PN, JR, JJ, BM, SK
	Add functionality for user profile correctly to each website	6	
	Total	35	

1. Fixing ui - 15
2. Backend tests - 8 hours
3. Connecting features - 12

Total New Estimate = 35 hours

50hrs- 35hrs = 15 extra hrs. A lot of hours left over but again as team lead i usually work over and since this is last sprint i want to have abundance of extra time to finish anything that may arise.

Jason Jitsiripol

Work Item	Task	Estimation in pts	Team Confirmation
Analysis Dashboard Database and Backend	1. Backend for Key Performance Indicators for the number of success and failed login	2	AN, DG,KN,PN, JR, JJ, BM, SK

	attempts per month over selected time span (trend chart)		
	2. Backend for Key Performance Indicators for the number of success and failed registrations per month over selected time span (trend chart)	2	AN, DG,KN,PN, JR, JJ, BM, SK
	3. Backend for Key Performance Indicators for top 3 longest page visits in seconds	2	AN, DG,KN,PN, JR, JJ, BM, SK
	4. Backend for Key Performance Indicators for top 3 most used feature	2	AN, DG,KN,PN, JR, JJ, BM, SK
	5. Backend for Key Performance Indicators for First application specific feature metric	2	AN, DG,KN,PN, JR, JJ, BM, SK
	6. Backend for Key Performance Indicators for Second application specific feature metric	2	AN, DG,KN,PN, JR, JJ, BM, SK
	7. Controller for Key Performance Indicators for the number of success	1	AN, DG,KN,PN, JR, JJ, BM, SK

	and failed login attempts per month over selected time span (trend chart)		
	8. Controller for Key Performance Indicators for the number of success and failed registrations per month over selected time span (trend chart)	1	AN, DG,KN,PN, JR, JJ, BM, SK
	9. Controller for Key Performance Indicators for top 3 longest page visits in seconds	1	AN, DG,KN,PN, JR, JJ, BM, SK
	10. Controller for Key Performance Indicators for top 3 most used feature	1	AN, DG,KN,PN, JR, JJ, BM, SK
	11. Controller for Key Performance Indicators for First application specific feature metric	1	AN, DG,KN,PN, JR, JJ, BM, SK
	11. Controller for Key Performance Indicators for Second application specific feature metric	1	AN, DG,KN,PN, JR, JJ, BM, SK
	12. Backend unit testing for Key Performance	1	AN, DG,KN,PN, JR, JJ, BM, SK

	Indicators for the number of success and failed login attempts per month over selected time span (trend chart)		
	13. Backend unit testing for Key Performance Indicators for the number of success and failed registrations per month over selected time span (trend chart)	1	AN, DG,KN,PN, JR, JJ, BM, SK
	14. Backend unit testing for Key Performance Indicators for top 3 longest page visits in seconds	1	AN, DG,KN,PN, JR, JJ, BM, SK
	15. Backend unit testing for Key Performance Indicators for top 3 most used feature	1	AN, DG,KN,PN, JR, JJ, BM, SK
	16. Backend unit testing for Key Performance Indicators for First application specific feature metric	1	AN, DG,KN,PN, JR, JJ, BM, SK
	17. Backend unit testing for Key Performance Indicators for Second	1	AN, DG,KN,PN, JR, JJ, BM, SK

	application specific feature metric		
	18. Backend requirement testing/bugfixing for Key Performance Indicators for the number of success and failed login attempts per month over selected time span (trend chart)	2	AN, DG,KN,PN, JR, JJ, BM, SK
	19. Backend requirement testing/bugfixing for Key Performance Indicators for the number of success and failed registrations per month over selected time span (trend chart)	2	AN, DG,KN,PN, JR, JJ, BM, SK
	20. Backend requirement testing/bugfixing for Key Performance Indicators for top 3 longest page visits in seconds	2	AN, DG,KN,PN, JR, JJ, BM, SK
	21. Backend requirement testing/bugfixing for Key Performance Indicators for top 3 most used feature	2	AN, DG,KN,PN, JR, JJ, BM, SK

	22. Backend requirement testing/bugfixing for Key Performance Indicators for First application specific feature metric	2	AN, DG,KN,PN, JR, JJ, BM, SK
	22. Backend requirement testing/bugfixing for Key Performance Indicators for Second application specific feature metric	2	AN, DG,KN,PN, JR, JJ, BM, SK
	Total	36	

1. Usage Analysis Board Frontend and Controllers = 2 + 2 + 2 + 2 + 2 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 = 36

Total New Estimate = 36 hrs

40 hrs - 36 hrs = 4 extra hrs. This is within the individual member's sprint capacity. The extra hours can be used to account for additional testing and bug fixing and/or assistance in establishing the frontend for the Analysis Dashboard

Kihambo

Work Item	Task	Estimation in pts	Team Confirmation
Collab Feature Regression Testing	1. Redesign the Collab Feature with a HLD and LLD	3	AN, DG,KN,PN, JR, JJ, BM, SK

	2. Modify the existing code in the backend to fit the design	8	AN, DG,KN,PN, JR, JJ, BM, SK
	3. Create test cases for Collab Feature for each user story in the BRD	2	AN, DG,KN,PN, JR, JJ, BM, SK
	4. Create the frontend button for Collab Feature	4	AN, DG,KN,PN, JR, JJ, BM, SK
	5. Debug any errors and warnings	8	AN, DG,KN,PN, JR, JJ, BM, SK
	1. Inspect code changes in different features	3	AN, DG,KN,PN, JR, JJ, BM, SK
	2. Run tests for each feature	3	AN, DG,KN,PN, JR, JJ, BM, SK
	2. If any code	8	AN, DG,KN,PN, JR, JJ, BM, SK

	changes affect the running tests, I'll fix them		
	3. Commit and push the fixed changes in the code to Github repository	1	AN, DG,KN,PN, JR, JJ, BM, SK
	Total	35	

1.

Total New Estimate = 35 hrs

35 hrs - 35 hrs = 0 extra hrs. This is the estimated time for me to complete this work item.

Diego

Work Item	Task	Estimation in pts	Team Confirmation
Bingo Board frontend/backend	1. Add backend logic for applying 2. Add frontend support for applying 3. Implement pagination	10	AN, DG,KN,PN, JR, JJ, BM, SK
Regression Testing	1. Review and revise current test cases to	15	AN, DG,KN,PN, JR, JJ, BM, SK

	work with updated backend		
	Total	35	

1. Bingo board = 10
2. regression tests = 15

Total New Estimate = 25 hrs

35 hrs - 35 hrs =0 hrs. This is the expected work estimation based on the initial work estimate given.

An

Work Item	Task	Estimation in pts	Team Confirmation
Item Creation (I.C.)	1. Revise LLD for I.C.	1	AN, DG,KN,PN, JR, JJ, BM, SK
	2. Create Backend Codes for Item Creation a. Data Access Layer Code b. I.C. Service Code c. API endpoints Code	5	AN, DG,KN,PN, JR, JJ, BM, SK
	3. Create Frontend Codes for Item Creation a. I.C. HMTL Code	5	AN, DG,KN,PN, JR, JJ, BM, SK

	b. I.C. JS code c. I.C. CSS Code		
Item Listing (I.L.)	4. Revise LLD for I.L.	1	AN, DG,KN,PN, JR, JJ, BM, SK
	5. Create Backend Codes for Item Listing a. Data Access Layer Code b. I.L. Service Code c. API endpoints Code	6	AN, DG,KN,PN, JR, JJ, BM, SK
	6. Create Frontend Codes for Item Listing a. I.L. HMTL Code b. I.L. JS code c. I.L. CSS Code	8	AN, DG,KN,PN, JR, JJ, BM, SK
Offering	7. Revise LLD for Offering	1	AN, DG,KN,PN, JR, JJ, BM, SK
	8. Create Backend Codes for Offering	10	AN, DG,KN,PN, JR, JJ, BM, SK

	a. Data Access Layer Code b. Offering Service Code c. API endpoints Code		
	9. Create Frontend Codes for Offering a. Offering HMTL Code b. Offering JS code c. Offering CSS Code	10	AN, DG,KN,PN, JR, JJ, BM, SK
End-to-end Tests	10. Perform End-to-end Tests with Khuong	10	AN, DG,KN,PN, JR, JJ, BM, SK
	Total	60	

1+5+5+1+6+8+1+10+10+10

Total New Estimate = 57 hrs

40 hrs - 57 hrs = -17 hours. This is the expected work estimation based on the initial work estimate given, but it has exceeded my work hours so I will have to delegate some tasks to the next sprint and work extra hours on this sprint.

Parth

Work Item	Task	Estimation in pts	Team Confirmation

Analysis Dashboard Frontend	Planning and Design	6	AN, DG,KN,PN, JR, JJ, BM, SK
	Environment Setup	2	AN, DG,KN,PN, JR, JJ, BM, SK
	Core Development	16	AN, DG,KN,PN, JR, JJ, BM, SK
	Interactivity and Functionality	4	AN, DG,KN,PN, JR, JJ, BM, SK
	Testing and Debugging	2	AN, DG,KN,PN, JR, JJ, BM, SK
Price Range Sorting	Research	1	AN, DG,KN,PN, JR, JJ, BM, SK
	Planning & Design	1	AN, DG,KN,PN, JR, JJ, BM, SK
	Frontend Development	3	AN, DG,KN,PN, JR, JJ, BM, SK
	Backend Development	3	AN, DG,KN,PN, JR, JJ, BM, SK
	Testing and Refinement	1	AN, DG,KN,PN, JR, JJ, BM, SK
	Deployment	1	AN, DG,KN,PN, JR, JJ, BM, SK
	Total	40	

Total New Estimate = 40hrs

40hrs - 40hrs = 0 hrs. This is the expected work estimation based on the initial work estimate given.

Khuong

Work Item	Task	Estimation in pts	Team Confirmation
Item Modification	1. Create LLD for Item Modification	1	AN, DG,KN,PN, JR, JJ, BM, SK
	2. Create Backend Codes for Item Modification a. Data Access Layer Code b. Item Modification Service Code c. API endpoints Code	5	AN, DG,KN,PN, JR, JJ, BM, SK
	3. Create Frontend Codes for Item Modification a. Item Modification HMTL Code b. Item Modification JS code c. Item Modificati	5	AN, DG,KN,PN, JR, JJ, BM, SK

	on CSS Code		
Item Deletion	4. Create LLD for Item Deletion	1	AN, DG,KN,PN, JR, JJ, BM, SK
	5. Create Backend Codes for Item Deletion a. Data Access Layer Code b. Item Deletion Service Code c. API endpoints Code	4	AN, DG,KN,PN, JR, JJ, BM, SK
	6. Create Frontend Codes for Item Deletion a. Item Deletion HMTL Code b. Item Deletion JS code c. Item Modification CSS Code	4	AN, DG,KN,PN, JR, JJ, BM, SK

Inventory Stock List	7. Revise LLD for Inventory Stock List	3	AN, DG,KN,PN, JR, JJ, BM, SK
	8. Create Backend Codes for Inventory Stock List a. Data Access Layer Code b. Inventory Stock List Service Code c. API endpoints Code	5	AN, DG,KN,PN, JR, JJ, BM, SK
	9. Create Frontend Codes for Inventory Stock List a. Inventory Stock List HMTL Code b. Inventory Stock List JS code c. Inventory Stock List CSS Code	5	AN, DG,KN,PN, JR, JJ, BM, SK

Financial Progress Report	11. Revise LLD for Financial Progress Report	3	AN, DG,KN,PN, JR, JJ, BM, SK
	12. Create Backend Codes for Financial Progress Report a. Data Access Layer Code b. Financial Progress Report Service Code c. API endpoints Code	7	AN, DG,KN,PN, JR, JJ, BM, SK
	13. Create Frontend Codes for Financial Progress Report a. Financial Progress Report HMTL Code b. Financial Progress Report JS code	7	AN, DG,KN,PN, JR, JJ, BM, SK

	c. Financial Progress Report CSS Code		
End-to-end Tests	14. Perform End-to-end Tests with Khuong	10	AN, DG,KN,PN, JR, JJ, BM, SK
	Total	60	

1+5+5+1+4+4+3+5+5+3+7+7+10 = 60

Total New Estimate = 60 hrs

40 hrs - 60 hrs =0. This is the expected work estimation based on the initial work estimate given, but it has exceeded my work hours so I will have to delegate some tasks to the next sprint and work extra hours on this sprint.

Team individual estimate total = 301

Final Analysis:

Do we accept the **Core Tasks**?:

Julie	Yes
Jason	Yes
Shane	Yes
Kihambo	Yes
Diego	Yes
Parth	Yes
Khuong	Yes
An	Yes

8Yes/0 No: We will accept the current effort estimation.

Do we accept **Individual Features**?:

Julie	Yes
Jason	Yes
Shane	Yes
Kihambo	Yes
Diego	Yes
Parth	Yes
Khuong	Yes
An	Yes

8 Yes/0 No: We will accept the current effort estimation.

Is Team Phoenix within our sprint capacity?:

	Original Estimate	New Team Estimate
Total	307	301

Team capacity - estimated hours:

280 - 301= -21 hours

We are assigning 21 more hours over our estimated capacity but some members have fallen a little behind so the hours ended up compiling to their assigned tasks. Even though it is over 15% we will have to work a little more to hopefully finish strong this semester.