Project Planning

EP = Effort Points = Hours

Developer	Mon	Tue	Wed	Thu	Fri	Total EP Capacity
Aiden	1.5	3	1.5	3	2.5	11.5
Kihambo	2	2	2	3	4	13
Nathan	3	3	3	3	3	15
Diego	2	3	2	4	3	14
Jacob	3	2	3	2	2	12
Kameron	2	2	2	2	2	10 (Vacation HRs)
Team						75.5

Team Capacity: 75.5 x 2 weeks = 151 eps

Team Velocity: Team velocity is a metric used to show the average amount of effort points a team is able to complete during a sprint. This is calculated by taking the average of the amount of effort points completed during each sprint.

- Velocity after Sprint 1: 78.5 EP / 7 days = 11.21 EP per day

Assignments In progress (Based on Priority) (Initial Estimate):

1. BRD (Initial Estimate = 67 EP)
    a. Context (2 Cs * 1.5 EP = 3 EP)
        i. Scope
            1. Security
            2. Valid usernames
        ii. User Types
            1. Preconditions
            2. Success cases
            3. Failure cases
    b. Core Components (4 CCs * 4 EP = 16 EP)

i. Security
    1. Authentication
    2. Authorization
    3. Logout
ii. User Administration
    1. Registration
    2. Account Recovery
    3. Account Deletion
    4. User Management
    5. User Privacy Control
iii. System Observability
    1. Usage Analysis Dashboard
    2. Logging
    3. Archiving(?)
iv. Universal Requirements
    1. Error Handling
    2. UI/UX
    3. Documentation

c. Features (16 features * 3 EP = 48 EP)
    i. Phase 1:
        1. ~~User Management~~
        2. ~~Collaboration Search~~
        3. ~~Ear Training~~
        4. ~~Artist Portfolio~~
        5. ~~Tempo Tool~~
        6. ~~Audio Visualizer~~
        7. ~~Basic Track Editor~~
        8. ~~Artist Profile Calendar~~
        9. ~~Scale Display~~
        10. ~~Bingo Board~~
        11. ~~Gig Review System~~
        12. ~~Collab Feature~~
    ii. Phase 2:
        1. ~~Music-Lesson Hub~~
        2. ~~Advanced Search~~
        3. ~~Tempo Transposition Tool~~
        4. ~~Rhythm Tempo Game~~

2. High-Level Design 6 Eps
    a. Product's overall architecture

        i. Client-side
        ii. Server-side
    b. Defining abstractions and flow of control for cross cutting concerns
        i. Error Handling
        ii. Logging
        iii. Data access
        iv. Security (Authentication & Authorization)

3. Project Plan 4 Eps
    a. Project dates
    b. Project estimates
    c. Comprised of SCRUM Sprints
    d. Milestones for both semesters
    e. Identifying risks & risk mitigations
4. Data Access Library 6 Eps
    a. Low-level design document (feature)
    b. Automated tests
5. Logging Library 6 Eps
    a. Low-level design document (feature)
    b. Automated tests

67 + 48 + 6 + 4 + 6 + 6 = 137 total ep work items

151 - 137 = 14 Eps left over

Task Breakdown Delegation (Individual Estimates):

Devs	**Capacity (Total EP)**	**Tasks (hours allotted)**
Aiden	11.5	● User Administration (5) ● Universal Requirements ● User Management (2.5) ● Ear Training (2.5) ● BRD Finalization

Kihambo	13	• Ear Training (1) • Collaboration Search (2) • Collaboration Request (1) • Universal Requirements • Project Plan
Nathan	15	• Security (3) • Tempo Tool (2.5) • Artist Profile Calendar (2.5) • Universal Requirements • BRD finalization
Jacob	12	• System Observability (4) - Usage dashboard, Logging? • Artist Portfolio (3) • Audio Visualizer (3) • Tempo Transposition (3) • Universal Requirements • Data Access
Diego	14	• Universal reqs. • Bingo board(3) • Gig management(3) • Music lesson hub(1) • Rhythm tempo game(1) • High Level Design
Kameron	10	• Advanced Search (2) • Collab Feature (2) • Scale Display (1) • Basic Track Editor (2) • Collaboration Search (2) • Logging

Analysis:
Our team velocity was pretty high in our previous sprint so we decided to have a high team capacity for this sprint as well. This will allow us to put quality, detailed work into our tasks.

When finished with the first task in an assignment:
Email Vong (CC team) the first finished task in an assignment for feedback. This will allow us to constantly have insight into how our work is progressing. This will also allow for more detailed daily standups and better team coordination since we will have a better grasp of our initial task feedback.

End of Sprint Checklist:

- Commit Code
- Client Demo
- Sprint Retrospective