University of British Columbia, Vancouver

Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 3

Date: 26/07/2024

Group Number: 38

Name	Student Number	Alias (Userid)	Preferred E-mail Address
Ali Hasan	34975292	ahasan02	ali.hasan9712@gmail.com
Sharjeel Shahid	30717987	sshahi03	sshahi28@uwo.ca
Muhammad Zaid Tahir	26857201	mtahir03	zaidt221325@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Timeline and Task Breakdown

Refer to below task specification to determine which group member is responsible for each task.

Jul 29:

- Create SQL script to generate the database off our logical/conceptual schemas.
  Sharjeel/Muhammad
- Create SQL script to populate the tables, ensuring that data is valid.
  Ali

Jul 31 - The tasks below are split into (1), as in priority 1, and priority 2 (2), tasks. Accordingly, we will prioritize the completion of priority 1 tasks, estimating completion within 3 days of all tasks.

Aug 3 - Focus now on blocking priority 2 tasks, as the baseline functionality from priority 1 tasks (mostly fetch/insert tasks and baseline view mocks) has been completed. Estimated to take 2 days to complete.

Aug 5 - Team meeting and reflection, honing final aspects of the application and ensuring validity of data/scripts/examples. Creation of script and slide deck for demo. Do some manual system stress testing.

Aug 8 - Final rehearsal of slide deck and demo.

Overall Summary:

Our database intends to model a player-centric client for an online, match-based, multiplayer video game. The entirety of player persistent information, centered around individual player profiles and match history, will be modeled in the database. The database will also model a centralized game shop supporting multiple currencies with which a player can transact to populate their inventories. Furthermore, we have now included a front-end functionality that will allow players to dynamically make these queries using intuitive GUI elements. So one can easily perceive and track this information.

Back-End Tasks

Player Profile Operations Muhammad

- Implement route to fetch player profile by ID (1)
- Implement route to insert player information (username, email, password) (1)
- Implement route to retrieve player's overall game statistics (wins, losses, ranking)
- Implement route to update player's ELO rating and total XP (1)

Inventories Muhammad

- Implement route to fetch player's inventory items (2)
- Implement route to consume consumables (2)
- Implement route to add new item to player's inventory (2)
- Implement route to update item quantity in inventory (2)

Match History Ali

- Implement route to populate a new match (randomly generate) (2)
- Implement route to fetch player's match history (2)
- Implement route to retrieve the match statistics for a specific match (2)
- Implement route to retrieve a players profile from a match (2)
- Implement route to retrieve all individual match performances from a match (1)

Shop and Transaction Sharjeel

- Implement route to fetch all items available in some shop (1)
- Implement route to fetch any one players transaction history (2)
- Implement route to process a transaction (purchase item) (2)

Currency Balance Sharjeel

- Create function to fetch player's currency balances (2)
- Implement query to update currency balance after a transaction (2)

Frontend Tasks (Views and Components)

Player Profile Page Muhammad

- Design and implement profile page (1)
- Implement “generate random match” button
- Create form for updating player information (username, email, password) (1)
- Develop component to display overall game statistics
- Implement rank and level visualization

Inventory View Muhammad

- Create basic view for inventory items (2)
- Create page bar for each inventory (2)
- Create filters for different item types (Cosmetic, Consumable) (2)
- Develop quantity indicator for consumable items (2)

Match History Page Ali

- Implement list view for recent matches (2)
- Create basic statistics table (kills, deaths, assists, efficacy, xp gained) within each row (2)
- Develop filters for different game modes (2)

Match Page Ali

- View all players match stats from one match in one page (1)
- Implement linking to other player profiles (1)

Shop Interface Sharjeel

- Implement grid view for shop items (1)
- Implement shop categories and filters (1)
- Create item detail view with purchase option (2)
- Develop shopping cart functionality (2)
- Implement currency balance display (1)

Transaction History View Sharjeel

- Implement list view for recent transactions (1)
- Implement transaction detail popup (1)
- Create date range filter for transactions (1)

Navigation and Layout Ali

- Implement main navigation menu (1)
- Create responsive layout for all pages (1)
- Implement user authentication views (login/logout) (1)

Data Fetching and State Management Ali

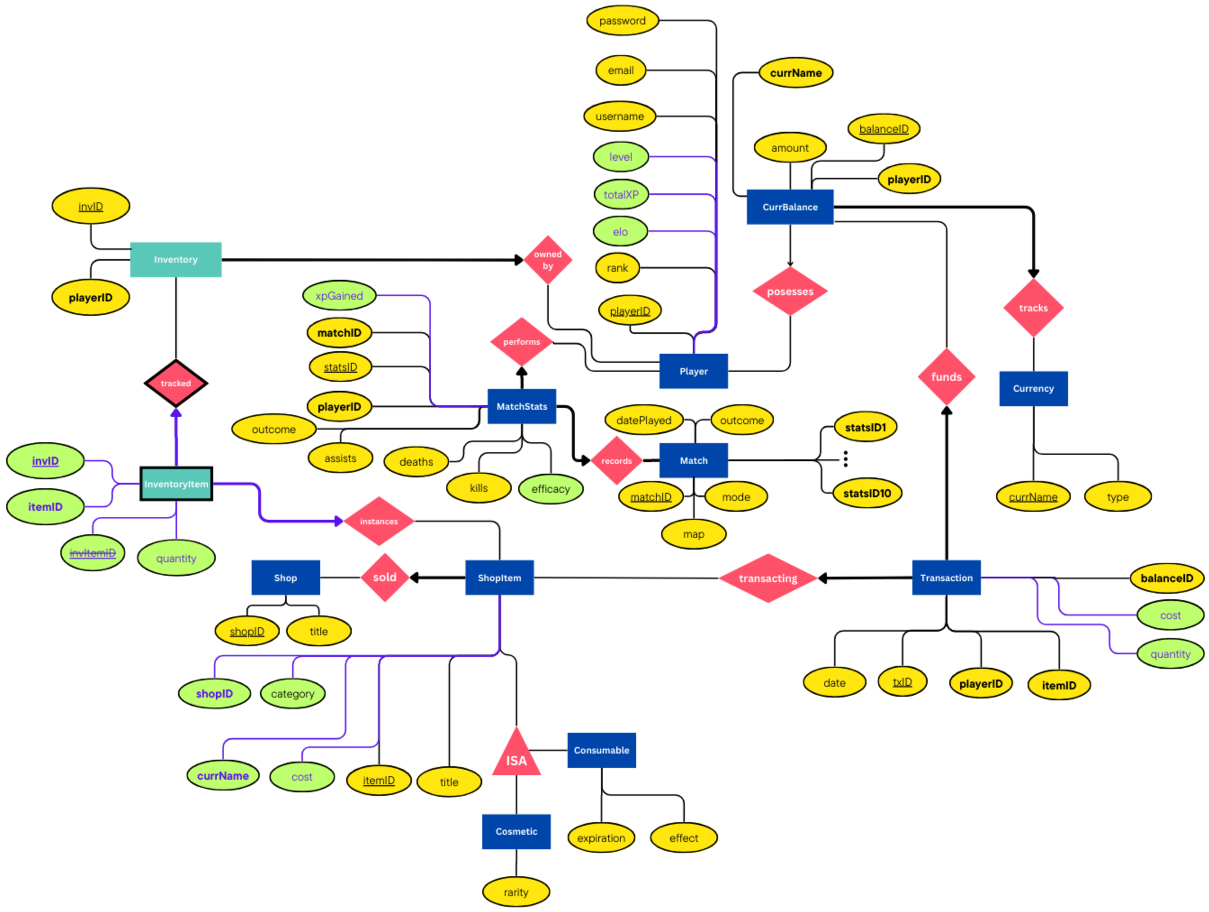
- Implement API service for backend communication (2)
- Create loading and error states. (2)

Styling and UI/UX Ali/Sharjeel/Muhammad

- Maintain a consistent theme (1)
- Create animations for transitions and loading states (1)
- Create custom icons for item categories.

University of British Columbia, Vancouver
Department of Computer Science

UPDATED ER DIAGRAM:


password
email
username
level
totalXP
elo
rank
playerID
currName
amount
balanceID
playerID
CurrBalance
invID
Inventory
playerID
owned by
possesses
tracks
xpGained
matchID
statsID
playerID
performs
Player
funds
Currency
tracked
MatchStats
outcome
assists
deaths
kills
efficacy
datePlayed
outcome
records
Match
matchID
mode
map
statsID1
statsID10
currName
type
invID
itemID
InventoryItem
~~invItemID~~
quantity
instances
Shop
sold
ShopItem
transacting
Transaction
balanceID
cost
quantity
shopID
title
shopID
category
date
txID
playerID
itemID
ISA
Consumable
currName
cost
itemID
title
Cosmetic
expiration
effect
rarity