**Database Management System**

Course Project Part II

GuildQuest Project Report

2021 Spring

Rhone Gavois

Ryan Driscoll

**Table of Contents**

1. Project Specifications Page
2. URL of Main Webpage………………………………………………………………. 3
3. Login Credentials……….……………………………………………………………. 3
4. Page Description……..………………………………………………………………. 3
5. Login Page……………………………………………………………………… 3
6. Admin Page…………………………………………………………………….. 3
7. Display Tables Page…………………………………………………………… 4
8. Search Results Page………………………………………………………….. 4
9. Leaderboards Page……………………………………………………………. 4
10. World Home Page……………………………………………………………… 4
11. Player Home Page…………………………………………………………….. 4
12. Quest Page……………………………………………………………………… 5

II. Project Discussion

1. Entity - Relationship Diagram……………………………………………………….. 6
2. Team Members Duties and Responsibilities………………………………………. 7
3. Changes from Part 1…………………………………………………………………. 7

**I. Project Specification**

1. URL of Main Webpage

There are two ways to access the main web page depending on what operating system the user is running. Below are the urls on Linux as well as Windows:

|  |  |
| --- | --- |
| Operating System | URL\* |
| Linux (ubuntu) | 127.0.0.1/GuildQuest/www/index.php |
| Windows | 127.0.0.1/index.php |

\*Depending on Apache configurations the URL may be different

1. Login Credentials

There consist of two main portals in this web application, an administrator portal and a player portal. The login credentials for each view of the web application are listed below.

|  |  |  |
| --- | --- | --- |
|  | Username | Password |
| Administrator Portal | admin | a12345 |
| Player Portal | LimeBat | L12345 |

\*Please make sure to run both populate scripts in the sql directory of the project in order to ensure that all user accounts have been added for logging into the application.

1. Page Descriptions

Note: Scripts are php files that are run in the background to support functionality of the application and used when no updated data needs to be viewed by the user (i.e. banning a user or logging into the application). A script is usually triggered by a button or form submission. The scripts used for each page are listed alongside with the page below.

1. *Login Page (index.php) Scripts (login.php, signup.php)*

This page allows a user of the application to do 1 of 3 things. A user can either login using his/her credentials. If they are not a current user of the application they can then sign up using the sign up form. Or lastly, a curious individual can look at the leaderboards for the game.

1. *Admin Home Page (adminHome.php) Scripts (createWorld.php addQuest.php adminActions.php)*

This page is the main page for an administrator of the web application. The admin can do the following on the page:

1. Display all of the entries in every table located in the DBMS
2. Ban specific accounts from the game
3. Unban specific accounts from the game
4. Remove an account from the DBMS
5. Find a single account
6. Find all banned accounts
7. Find a specific guild
8. Find accounts that have spent over ‘x’ amount of dollars
9. Find a specific world
10. Create a new world that players can then join and play in
11. Create a new quest that players can complete
12. *Display Tables Pages (displayAccounts.php, displayPlayers.php, displayWorlds.php, displayQuests.php, displayGuids.php, displayMissions.php, displayPlots.php).*

The display tables pages all in essence provide the same functionality for the administrator. When the correlating link is clicked in the Admin Home Page, a query will be run selecting all tuples from that selected entity and then the output is displayed on the display page itself.

1. *Search Results Page (search.php)*

This page displays the search results that are obtained from the Admin Home Page in the search form. Depending on what option the administrator selected (i.e. find a single user, find banned users, etc.) the corresponding information is displayed.

1. *Leaderboards Page (leaderboard.php) Scripts (Coins.php Experience.php Defence.php Attack.php Level.php Health.php)*

This page allows a user to view other players statistics within the application. A user can sort the users based on the values of each attribute in the players table. Everytime a header text is clicked, a new query runs on the database selecting all attributes and sorting in ascending order based on the attribute that the user selected.

1. *World Home (worldHome.php) Scripts (addWorld.php, createWorld.php, newWorldLogin.php)*

This page displays all the worlds available for the user. Worlds are divided into two lists, one for registered worlds and another for unregistered worlds. Users may join any registered worlds they have already signed up for. Users may add any unregistered worlds to their worlds list by signing up for them.

1. *Player Home Page (playerHome.php) Scripts (completeQuest.php, addQuest.php, removeQuest.php, joinGuild.php, leaveGuild.php, addPlot.php, removePlot.php)*

This page is the main page for players of the web application. Players can do the following on the page:

1. Display all player stats including:

* General stats
* Resource Inventory
* Active Quests
* Owned Plots
* Guild stats
* Guild members

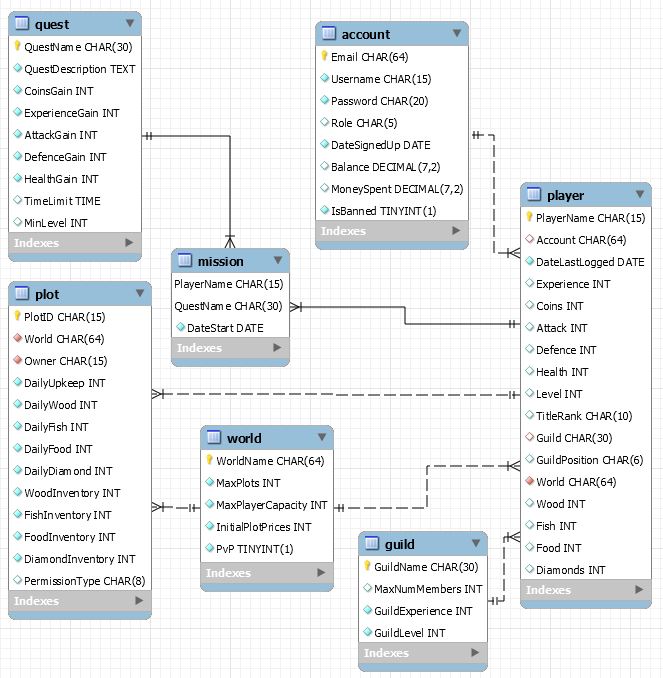
1. Joining a guild
2. Leaving a guild
3. Add plots
4. Remove plots
5. Add quests
6. Removing quests
7. Complete quests to receive player points.
8. *Quest Page (questHome.php) Scripts (createQuest.php, addQuest.php)*

This page lists all available quests for a player to complete depending on their level. Players with higher levels will get access to more quests.

**II. Project Discussion**

1. Entity - Relationship Diagram

The following diagram was created using MySQL.



1. Team Members Duties and Participation

The duties of the project were split evenly among the two group members. The duties each group member held are outlined below. Both participants worked on the database design together in part 1 of the project.

1. *Rhone Gavois*

Rhone was responsible for implementing the player portal of the web application. This included the player homepage and world page. Rhone also worked on the normalization of the database as well as creating the entity-relationship diagram for the application in MySQL..

1. *Ryan Driscoll*

Ryan was responsible for implementing the administrator portal of the web application as well as the welcoming homepage which included the login, sign up, and leaderboard pages of the application.

1. Changes from Part 1

This section will briefly describe all unplanned additions to the web-application along with any attributes removed in the database.

Administrator Use Cases:

* Using the administrator’s portal, administrators are given the ability to to view all records for each table in the database. Additionally, administrators can search for a specific record in the database by using their search feature.
* In addition to admins being able to ban accounts, they can also unban an account or remove an account entirely which will result in permanent removal of account information. Once an account has been removed from the database, all associated players will not be removed from the database allowing player information to be saved.
* Administrators are also given the ability to create worlds using their portal. However, players are not able to create worlds. As a result, the “WorldType” attribute was removed.
* Administrators are also given the ability to to create quests using their portal.

Player Use Cases:

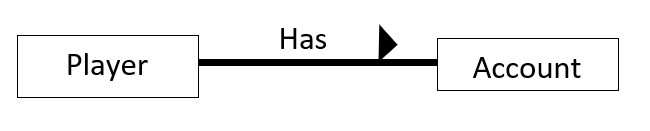
* Originally, after a user has logged off, the account’s new state will be used to update the database once. However due to the possibility of multiple users playing at once, the database would need to be updated after every form is complete.
* Originally we planned for each user to have one player interface. However, we later allowed users to have separate player interfaces for each world they join.
* As new users login, they will first need to register themselves to a world before playing on it.

Entities and Relation Attributes:

* ACCOUNT:
  + Primary key AccountID was replaced with Email for simplicity.
  + LeaderBoardRanking was removed because this attribute could be derived from other tables.
* PLAYER:
  + Primary key PlayerID was replaced with PlayerName for simplicity.
  + All QuertSlots were removed because the many to many relationship between players and quests could be handled by creating a new relation which we called “Missions”.
  + Sword and Armor was removed for simplicity.
* WORLD:
  + Primary key WorldID was replaced with WorldName for simplicity.
  + WorldSize was renamed to MaxPlots which is an integer datatype.
  + CurrentPlayerCount was removed because this attribute could be derived from other tables.
  + GameMode was removed for simplicity.
* GUILD:
  + Primary key GuildID was replaced with GuildName for simplicity.
  + GuildMemberCount was removed and replaced with MaxNumMembers.
  + GuildLeader was removed because it can be handled with in player table as GuildPosition
* PLOT:
  + PvP was removed for simplicity.
  + DailyWood, DailyFish, DailyFood, DailyDiamond, WoodInventory, FishInventory, FoodInventory, and DiamondInventory were added after a discussion about the game mechanics. Resources may only be collected using plots whereas ability stats can only be improved through quests.
* QUEST:
  + Primary key QuestID was replaced with QuestName for simplicity.
  + Other attributes like AttackGain, DefenceGain, and HealthGain were added to stay consistent with the game mechanics.
* MISSION:
  + This relation was added due to the many to many relationships between players and quests. Therefore, on top of PlayerName and QuestName as attributes, DateStart was added as well.
* Other:
  + On top of all changes described above, many default values were added to the attribute definitions to avoid null value occurrences. This allows the frontend to input the least amount of data possible in php insert commands as well as filling tables completely.
  + Not all attributes were accounted for in the development of the web-application. The aspired features which use these attributes were not implemented due to time constraints.

Relationships:

* The only relationship change in our database is the Player and Account relationship. Originally, players had to map to one account. However, we decided to allow each account to have one player interface for each existing world. This means that an account may have multiple player interfaces.



0..1 1..\*

Other:

* Overall, we did not seem to be missing anything important other than the missions relation for the player and quest relationship.
* Overall, the aspiring GuildQuest application resulted in the overwhelming amount of attributes. Not all attributes were fully used to implement all features of the application.
* Due to its nature, Part 1 was general and open for interpretation. Therefore, most of the game mechanics and definitions were discussed during the development of the application.