

Database Design and Implementation: Project: Part C

Due on November 16, 2018

Professor Zhixiang Chen

David Caballero, Benjamin Garcia, Tomas Torres

Contents

1	Introduction	3
1.1	Purpose	3
1.2	Scope	3
1.3	Definitions, Acronyms, and Abbreviations	3
1.4	References	3
2	System Overview	4
3	Decomposition Description	4
3.1	Data Storage Module	4
3.2	Data Processing Module	5
3.3	User Interface Module	5
4	Dependency Description	6
4.1	Data Storage Dependencies	6
4.2	Data Processing Dependencies	6
4.3	User Interface Dependencies	7
5	Interface Description	7
5.1	Data Storage Module Interface	7
5.2	Data Processing Module Interface	7
5.3	User Interface Module Interface	7
6	Detailed Design	7
6.1	Application Program Design	7
6.2	User Authentication Function Set	7
6.2.1	Authentication	7

1 Introduction

1.1 Purpose

This document we will describe in detail the design of our TourneyDB web-based Data Management system. We will decompose the functionality of this design into interrelated modules that interact with each other through the transferring of data, we will break down these modules to the core tasks we want them to complete and show how they will be implemented as well as how they will interact with each other.

1.2 Scope

This is a preliminary design for our TourneyDB prototype which will likely be modified if implementation presents any issue.

1.3 Definitions, Acronyms, and Abbreviations

TourneyDB: Our web-based software that will aid in the creation, management, and storage of the data relevant to running a tournament.

1.4 References

TourneyDB will consist of three main modules: DataStorage, Data Processing, and UserInterface.

2 System Overview

Our system will provide almost all of the functionality required to handle the data necessary to run a tournament. It will handle user authentication and session management to protect the data created by individuals. It will allow for organizers to create tournaments, and allow guests to view tournaments in their areas, and view local players around them. It will automatically keep track of player performance and generate leaderboards that rank players among each other. It will also provide the functionality to track tournament status in real time.

3 Decomposition Description

Like most web applications our system will depend on 3 core modules, one that stores data, one that processes data, and one that presents that data to the user as well as intakes data from the user. These are represented in Figure 1.

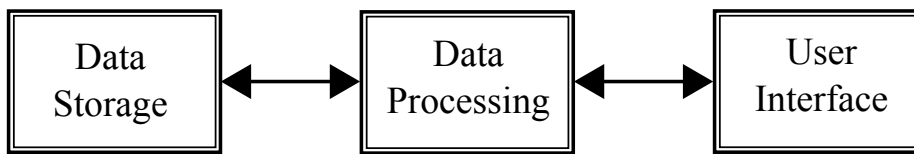


Figure 1: The three core modules of TourneyDB system. Arrows represent the transfer of data.

3.1 Data Storage Module

This module stores all data needed by the TourneyDB system. It interacts directly with the data processing model by providing data needed to create views for the user, authenticate users, etc. It is a relational database consisting of tables that each represent an entity relevant to our system. Our Data Storage Module is represented by a diagram in Figure 2.

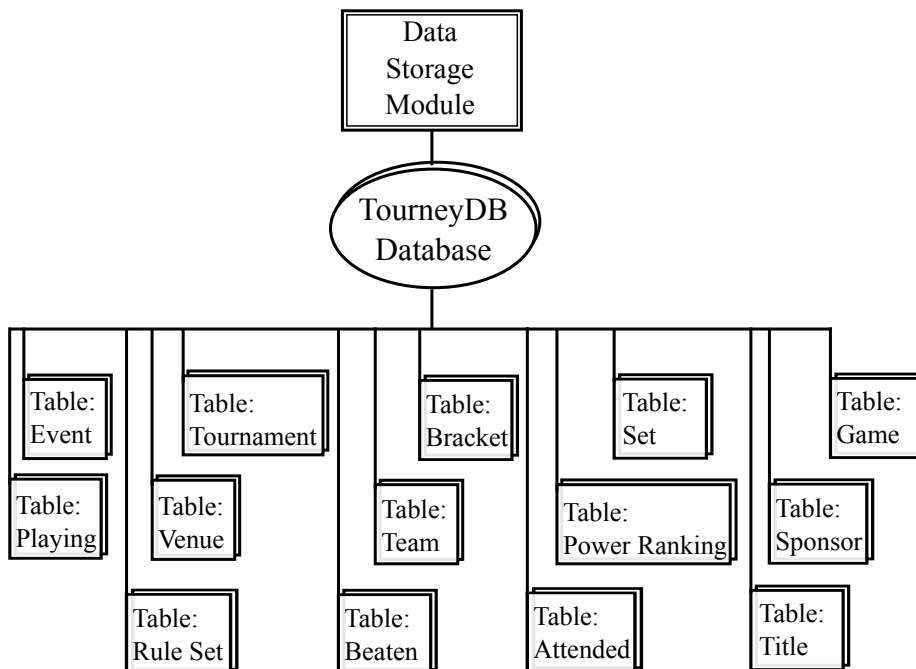


Figure 2: Breakdown of the Data Storage Module

3.2 Data Processing Module

This module will handle all server side processing of data. It will retrieve data from the Data Storage Module, and perform operations on it, such as sorting, formatting, etc. With the data provided from the User Interface it can retrieve specific data relevant to what the user wants. With the data retrieved from the Data Storage Module it can create dynamic web pages. This module will also handle the authentication of users. The Data processing module can further be broken down into specific function sets. These are *User Authentication*, *Database Connection Initiation*, *Data Retrieval*, *Data Update*, *Data Filtering and Computation*, and *Report Generation*.

1. **User Authentication and Authorization:** Allows the logging in of users of this site. Verifies the user has the permissions necessary for a certain action.
2. **Database Connection Initiation:** Handles the initiation of the connection and authentication with the Data Storage Module
3. **Data Retrieval:** Retrieves data from the Storage Module
4. **Data Update:** Creates new data or modifies existing data in the Data Storage Module.
5. **Data Filtering and Computation:** Performs operations on data such as computing certain statistics, sorting, filtering, etc.
6. **Report Generation** Takes a request from the user interface module and generates a respective report.

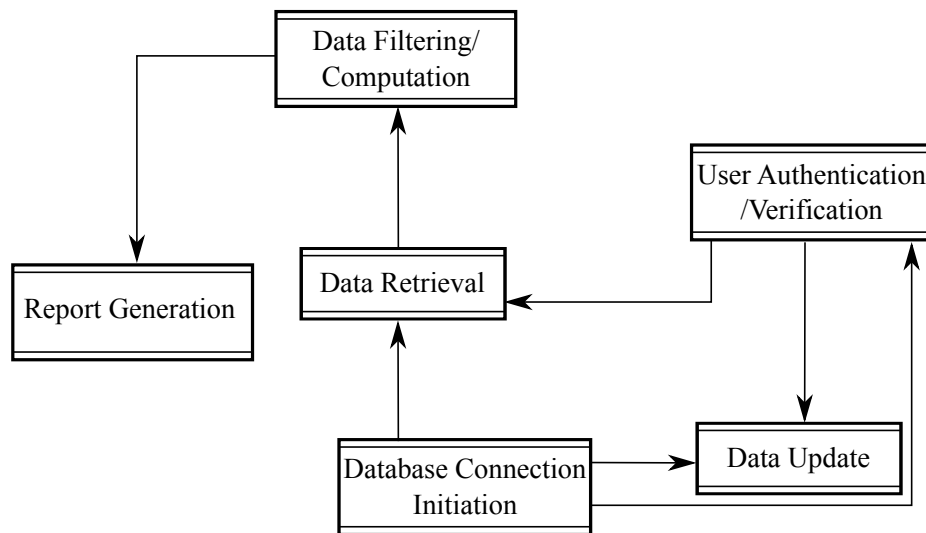


Figure 3: Breakdown of the Data Storage Module

3.3 User Interface Module

The User Interface Module will be where a user directly interactis with our system. It will take data from the user that will be sent to the Data Processing Module, and it will present the data to the user in a viewable manner. There are 9 main pages that the user will interact with. Figure 4 shows how these pages are linked to one another.

1. **Home Page:** First page a user sees when entering our site.
2. **Login/Register:** Page where the player can enter their information to be authenticated or create a new account.
3. **User Profile:** Page where the user can vee and update their own information.
4. **My Tournaments:** Accessible through *User Profile*. Shows the tournament created by a user.
5. **Leaderboard:** Lets the user see the ranking of players that use the site.

6. **Player View** Shows the information of another user.
7. **Create Tournament**: The page for the creation of a tournament.

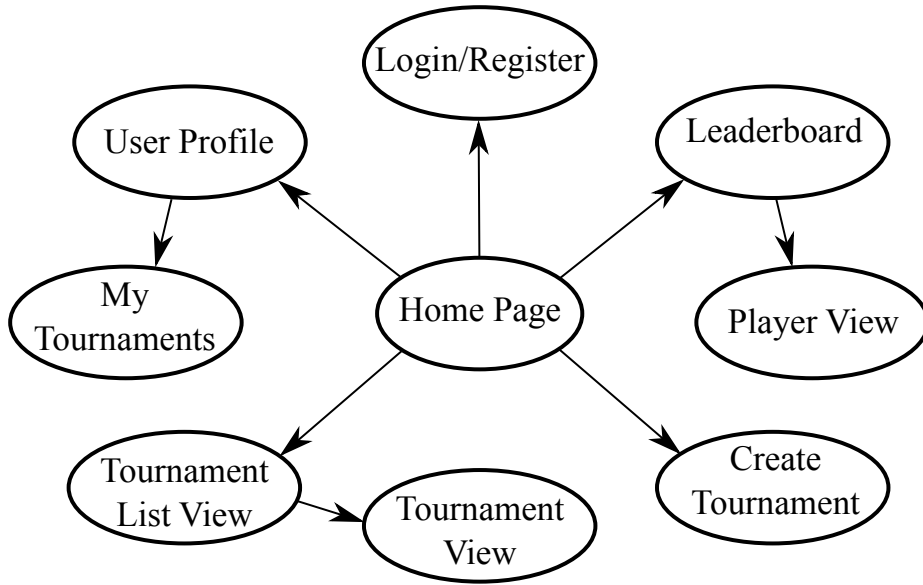


Figure 4: Flowchart of the User Interface

4 Dependency Description

This section will describe both the hardware and the software that our TourneyDB system depends on.

4.1 Data Storage Dependencies

The Data Processing Module is a MySQL server hosted by AWS on a *t2.micro* Ubuntu server instance. It is named tourneydb and has one authenticated user, authentication is necessary to start a connection with our database.

Hardware	Software
t2.micro instance 1 vCPU .5 GiB Memory 20 GB Storage	Ubuntu Server 18.04 MySQL 5.6.40

4.2 Data Processing Dependencies

Our data processing module will be hosted on the same Ubuntu server. Our site will be hosted on a PHP 5.4.0 server running on port 80. All data will be processed by PHP functions.

Hardware	Software
t2.micro instance 1 vCPU .5 GiB Memory 20 GB Storage	Ubuntu Server 18.04 PHP Server 5.4.0

4.3 User Interface Dependencies

The User Interface Module is located on the same Ubuntu server, it will generate HTML documents dynamically using PHP and send those to clients requesting it. The structure and design will really on Bootstrap templates and CSS stylesheets.

5 Interface Description

5.1 Data Storage Module Interface

The MySQLi PHP library will be used to connect and transfer data to the Data Storage Module. On loading our site, a connection will be created with our MySQL server through MySQLi, requiring authorized user credentials to be provided. We will execute queries and updates using SQL queries dynamically generated using PHP.

5.2 Data Processing Module Interface

Our PHP server will be accessible through port 80 of our Ubuntu server. It will be handed data through HTTP requests, and from there take the appropriate action such as loading of a new page, filtering of data, updating the Data Storage Module, *etc.*

5.3 User Interface Module Interface

Our User Interface will be accessible through any desktop browser. It will use HTML forms to take in data from the user and send HTTP requests to send these to our Data Processing Module.

6 Detailed Design

6.1 Application Program Design

The root folder of our application will be in our TourneyDB directory. This folder will contain the PHP files that the server will run. Any resources needed such as images or videos will be in TourneyDB/resources.

6.2 User Authentication Function Set

6.2.1 Authentication

Authenticates a user with a username and a password