World Cup Soccer Database - Phase 1

Rebecca Thomas, Dmytry Berkout

April 5, 2016

Contents

1 Purpose **2**
- 1.1 Purpose of the Document . . . 2
- 1.2 Purpose of the Project . . . 2
- 1.3 Purpose of the Phase . . . 2

2 Problems and Solutions **2**

3 Assumptions **3**

4 Environment and Requirements Analysis **3**
- 4.1 Using MondialDB . . . 3
- 4.2 Extract Transform Load Tool . . . 4
- 4.3 Top-Level Information Flow Diagram . . . 4

5 List of Tasks and Task Flow Diagram **6**
- 5.1 Extract, Transform, and Load Task . . . 7
- 5.2 Webpage Server Task . . . 9
- 5.3 Web Query Processor . . . 11
- 5.4 Webpage SQL Query Processor Task . . . 12
- 5.5 Output Results Task . . . 13

6 List of Documents **14**
- 6.1 Webpage . . . 15
- 6.2 Query . . . 15
- 6.3 Database Results . . . 16
- 6.4 Formatted Results . . . 16
- 6.5 Soccer Websites . . . 17
- 6.6 Unformatted Data . . . 18
- 6.7 Formatted Data . . . 18
- 6.8 SQL Script . . . 18

1 Purpose

1.1 Purpose of the Document

This document is an introduction to the MondialDB project. It will provide details on the implementation and purpose of MondialDB. This document will provide a detailed design of MondialDB. It will include an information flow diagram and task forms that go into the inner workings of the project. It will also include the problems we encountered and the solutions we had for them.

1.2 Purpose of the Project

The purpose of the project is to create a database that can access various details about the World Cup. It will contain soccer team names and members and statistics such as player records, team records, penalty information, and the rank obtained. Another major part of this project is creating a tool to extract information about the World Cup from the Internet. It will be able to extract information from various websites and put them into a database readable format. The tool will be able to deal with data from European and American websites and transform it into a standard format. This project will allow for easy and readable queries.

1.3 Purpose of the Phase

In this phase of the project we are to design the database and receive feedback on it. We will attempt to catch any significant errors in design at this stage and prevent a broken or severely flawed implementation from going through. If design errors are caught early it will require considerably less work to fix them.

2 Problems and Solutions

The following list is composed of problems we encountered with the conceptual design of the project.

- Problem: We lack database design experience. Without experience, it is difficult to design a database.

  Solution: Follow project examples and study how databases are designed.

- Problem: We don't understand the World Cup and we don't know where to find data for it.

  Solution: Find out how the World Cup works. Research various World Cup websites. Find several that are usable for our project.

- Problem: We don't know how to extract data from websites.

  Solution: Research data extraction. Make a very basic test script. See what suites are available for use.

- Problem: We don't know how to create websites.

  Solution: Research the creation of websites. Make a very basic website. See what suites are available for use.

- Problem: We don't live close together.

  Solution: Use text messages, phone calls, and email to communicate. Use Google documents to share data.

3 Assumptions

The assumptions we made are the following:

- Our web server will not be overloaded despite not having restrictions on who can use it.
- Our web server software will not fail.
- The database software will be sufficient for the scope of this project.
- The data pulled from websites is accurate.
- We will not need multilingual support. We will support only English.
- We will not need handicap support for our website.
- We will not need a mobile accessible version for our website.
- Our users are average people who do not have a computer science background but are able to comfortably use the internet.

4 Environment and Requirements Analysis

4.1 Using MondialDB

The user will interact with MondialDB through our website. The user will connect to the website using a web browser and a simple webpage will be displayed. The sole purpose of this website is to run specific queries from the web-server through MondialDB and send the results back to the user. The website will only contain a selection of the predefined queries. On selection, a website form will appear which will contain the necessary information to perform the specified query. Both the website and web-server will check the input for validity. Once the query is processed, a table of results will appear underneath the form.

4.2 Extract Transform Load Tool

For this project we will write a python script to pull data from selected websites. The ideal script will be as simple as possible while robust enough to work with a number of different websites. We will input a list of websites and the tool should automatically convert the data into database format and insert it, or provide a script to insert it into MondialDB.

4.3 Top-Level Information Flow Diagram

See Figure 1 on page 5 for the Top-Level Information Flow Diagram. The flow is generally as follows:

1. Data is input into MondialDB from Soccer Websites through Extract-Transform-Load.
2. The user asks for the website and is provided it through the Webpage Server.
3. The Web Query Processor decides what kind of query the user is asking for.
4. The Webpage SQL Query Processor translates the user query into SQL to be executed on MondialDB.
5. The query is executed and database results are returned from MondialDB
6. The results are outputted to the user with the format depending on the kind of query.

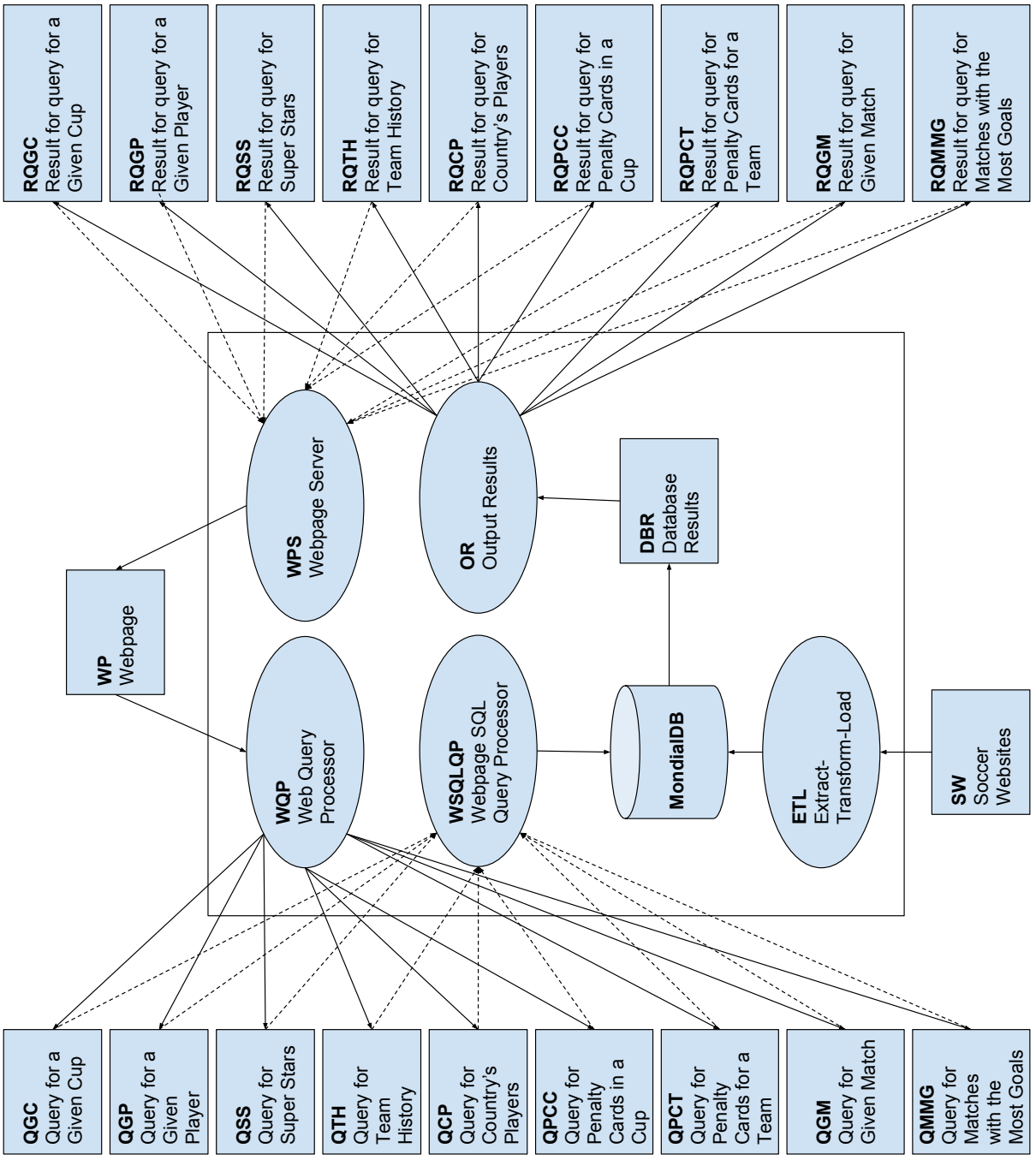

Figure 1: Information Flow Diagram

5 List of Tasks and Task Flow Diagram

We describe the tasks and subtasks necessary to make, populate, and query MondialDB. See Figure 2 on page 6 for the Task Flow Diagram. The tasks are the following:

- Extract, Transform, and Load Task
- Webpage Server Task
- Web Query Processor
- Webpage SQL Query Processor
- Output Results Task

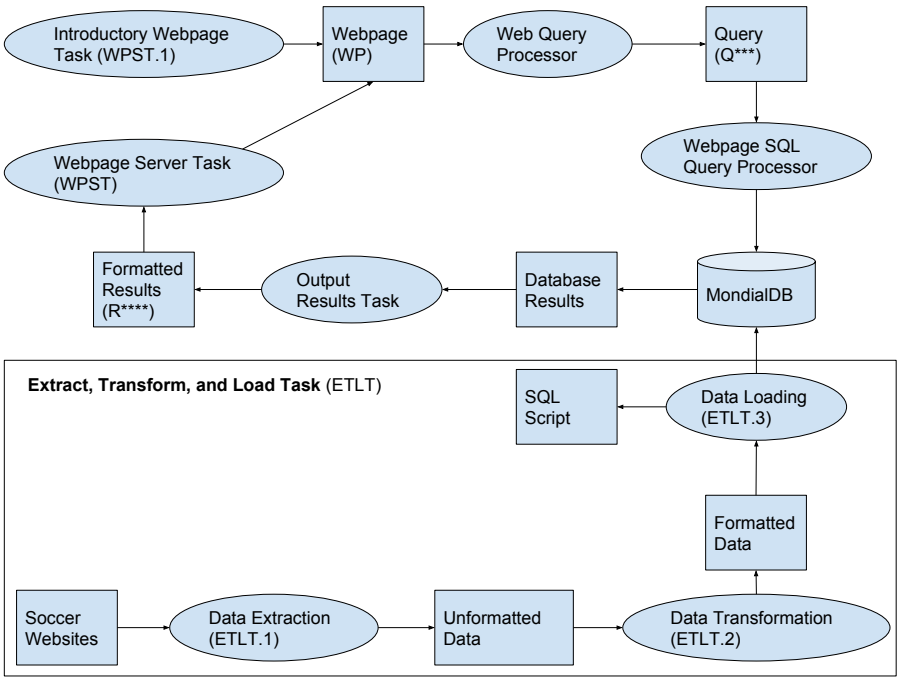

Figure 2: Task Flow Diagram

5.1 Extract, Transform, and Load Task

Task Label ETLT
Task Name Extract, Transform, and Load Task
Performer Python script
Purpose To extract data from Soccer Websites, transform it into a usable format, and send it to MondialDB
Enabling Condition On database creation or database update
Description It takes the information from Soccer Websites and puts the information into MondialDB.
Frequency On database update
Duration It will depend on the extraction, transformation, and load subtasks.
Importance Most important
Maximum Delay It depends on the subtasks.
Input Soccer Websites
Output Copy of the SQL script run by the python script as well as debugging information
Document Use Soccer Websites, Unformatted Data, Formatted Data
Operations Performed Data extraction, data transformation, and data loading
Subtasks Data extraction (ETLT.1), data transformation (ETLT.2), data loading (ETLT.3)
Error Conditions Errors from subtasks

5.1.1 Data Extraction

Task Label ETLT.1
Task Name Data Extraction
Performer Python script
Purpose To extract data from Soccer Websites
Enabling Condition On database data insertion
Description It pulls HTML from the Soccer Websites. It then parses the HTML for data to put into MondialDB.
Frequency On database update
Duration It depends on how quickly websites are scraped and extracted.
Importance Most important
Maximum Delay It depends on how many Soccer Websites are chosen.
Input Soccer Websites
Output Unformatted Data from the Soccer Websites
Document Use Soccer Websites -> Unformatted Data
Operations Performed Data extraction
Subtasks None
Error Conditions Soccer Websites are invalid. The format of the website is invalid or confusing.

5.1.2 Data Transformation

Task Label	ETLT.2
Task Name	Data Transformation
Performer	Python script
Purpose	To transform data from Soccer Websites into a standardized format
Enabling Condition	On database data insertion
Description	It standardizes the data produced by Data Extraction. For example, names that are formatted like "Last Name, First Name" and "First Name Last Name" shall be changed into a standard format.
Frequency	On database update
Duration	It depends on how quickly the data goes from being unformatted to being formatted.
Importance	Important
Maximum Delay	It depends on how badly the original data was formatted.
Input	Unformatted Data from Data Extraction
Output	Formatted Data
Document Use	Unformatted Data -> Formatted Data
Operations Performed	Data transformation
Subtasks	None
Error Conditions	The data are formatted badly.

5.1.3 Data Loading

Task Label	ETLT.3
Task Name	Data Loading
Performer	Python script
Purpose	To load formatted data into MondialDB
Enabling Condition	On database data insertion
Description	Makes an SQL script that inserts the standardized data from Data Transformation into MondialDB.
Frequency	On database update
Duration	It depends on how quickly the data is inserted into MondialDB
Importance	Most important
Maximum Delay	It depends on how the slow the connection between the data collector and the database is.
Input	Formatted Data
Output	Log from inserting into MondialDB. Copy of the SQL script run by the python script.
Document Use	Formatted Data -> SQL Script
Operations Performed	Data Loading
Subtasks	None
Error Conditions	There is a faulty connection with the database.

5.2 Webpage Server Task

Task Label	WPST
Task Name	Webpage Server Task
Performer	Web-server
Purpose	Sends webpage to user. Takes requests for data or returns results from a previous query.
Enabling Condition	After database creation
Description	The webpage server will act as the mediator between the MondialDB and the user. It will take user requests and send them to a process that can indirectly interact with MondialDB. It will accept any user connections.
Frequency	Always on
Duration	As long as the database is active
Importance	Most Important
Maximum Delay	Response delay to user requests should be short and less than 10 seconds.
Input	User connection or Formatted Results
Output	Webpage for user to interact with
Document Use	Formatted Results -> Weppage
Operations Performed	Generate and send webpage to user. If new user, then do the operations in WPST.1. Otherwise, serve up the query results in a table.
Subtasks	Introductory Webpage Task (WPST.1)
Error Conditions	The web-server fails or does not respond. Too many users connect at once.

5.2.1 Introductory Webpage Task

Task Label	WPST.1
Task Name	Introductory Webpage Task
Performer	Web-server
Purpose	To give a new user a webpage to interact with. Generate the webpage and interface.
Enabling Condition	On access to MondialDB website
Description	The web-server will generate a website on visit from user. From here the user will be able to do database queries.
Frequency	When a user accesses the website
Duration	Webpage generation should be fast and less than a few seconds. The webpage should remain active as long as the user doesn't disconnect. There might be a possible timeout for inactive users.
Importance	Important
Maximum Delay	In ideal network conditions, it should take less than 5 seconds.
Input	User connection
Output	Introductory Webpage for the user to interact with
Document Use	Webpage
Operations Performed	Generate introductory webpage for user to interact with. Handle valid user requests by passing them on to send data to Web Query Processor. Handle invalid user requests by ignoring them and informing the user why they are invalid.
Subtasks	None
Error Conditions	The web-server fails or does not respond. Too many users connect at once.

5.3 Web Query Processor

Task Label WQPT
Task Name Web Query Processor
Performer Web-server
Purpose Processes queries from the web-server and sends the respective query to the Webpage SQL Query Processor Task.
Enabling Condition After web-server runs
Description It determines the user-specified query from the forms on the Webpage. Once this query is validated, it sends the query on to the Webpage SQL Query Processor.
Frequency Always on
Duration As long as the database is active
Importance Important
Maximum Delay Response delay to user requests should be short and less than 10 seconds.
Input User-inputted form data from the Webpage
Output Sends Query type to Webpage SQL Processor Task. Query types include QGC, QGP, QSS, QTH, QCP, QPCC, QPCT, QGM, and GMMG.
Document Use Webpage -> Query
Operations Performed Validates queries and determines query type.
Subtasks None
Error Conditions The query isn't valid.

5.4 Webpage SQL Query Processor Task

Task Label WSQLQPT
Task Name Webpage SQL Query Processor Task
Performer Webpage server
Purpose Sends the query from the user-specified query task to MondialDB
Enabling Condition User submits valid query
Description The WSQLQPT is the final step before the query reaches MondialDB. It will send a perform a raw SQL query in MondialDB
Frequency Triggers on every received valid user query
Duration The SQL shouldn't take longer than 15 seconds to run.
Importance Important
Maximum Delay The SQL shouldn't take at most longer than 45 seconds to run.
Input User-specified query from the following list:
QGC (Query for a given cup)
QGP (Query for a given player)
QSS(Query for Super Stars)
QTH(Query for Team History)
QCP(Query for Country's Players)
QPCC(Query for Penalty Cards in a Cup)
QPCT (Query for Penalty Cards given to a Team)
QGM (Query for a Given Match)
QMMG (Query for Matches with the Most Goals)
Output Sends SQL query to MondialDB
Document Use Query
Operations Performed If the query is a valid query from the aforementioned list, then create the SQL command. If it is not a valid query, send an error message back.
Subtasks None
Error Conditions The user-specified query is not valid.

5.5 Output Results Task

Task Label OR
Task Name Output Results Task
Performer Webpage Server
Purpose Sends the query results to the user
Enabling Condition MondialDB receives query and sends output to Output Results Task
Description The WSQLQPT is the final step before the query reaches MondialDB. It will send a raw SQL query to MondialDB.
Frequency Triggers on every received request
Duration In ideal network conditions, it should be short and less than 10 seconds.
Importance Most Important
Maximum Delay Response delay to user requests should be short and less than 10 seconds.
Input For a user-specified query, Database Results
Output The Formatted Result from the following results:
RQGC (Result for Query for a Given Cup)
RQGP (Result for Query for a Given Player)
RQSS (Result for Query for Super Stars)
RQTH (Result for query for Team History)
RQCP (Result for query for Country's Players)
RQPCC (Result for Query for Penalty Cards in a Cup)
RQPCT (Result for query for Penalty Cards for a team)
RQGM (Result for Query for Given Match)
RQMMG (Result for query for Matches with the Most Goals)
Document Use Database Results -> Formatted Results
Operations Performed Sends query results to Webpage Server Task
Subtasks None
Error Conditions MondialDB gave back error conditions instead of valid results. Connection to user is disrupted. Web server becomes overloaded.

6 List of Documents

See page 14 for repeated Task Flow Diagram. The documents are the following:

- Webpage
- Query
- Database Results
- Formatted Results
- Soccer Websites
- Unformatted Data
- Formatted Data
- SQL Script

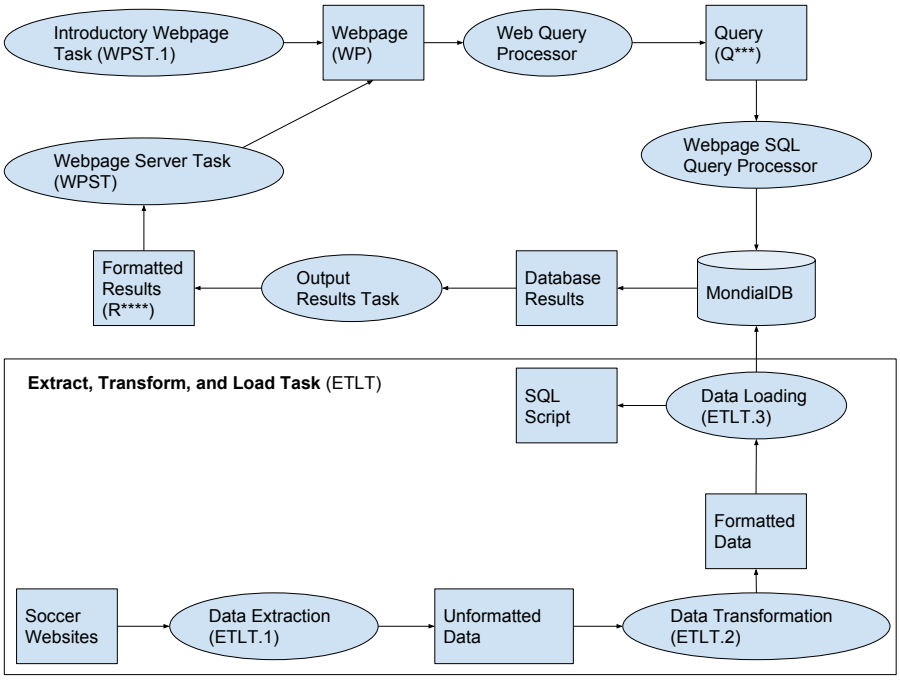

Figure 3: Task Flow Diagram

6.1 Webpage

The website will display either an introductory webpage or a webpage that displays results from a previous query.

- Title
- Description
- Link-to-Usage-Documentation
- Query-Request-Form
- Result-Table

6.2 Query

The types of Queries:

- QGC (Query for a given cup)
- QGP (Query for a given player)
- QSS (Query for Super Stars)
- QTH (Query for Team History)
- QCP (Query for Country's Players)
- QPCC (Query for Penalty Cards in a Cup)
- QPCT (Query for Penalty Cards given to a Team)
- QGM (Query for a Given Match)
- QMMG (Query for Matches with the Most Goals)

Each query might not require all parts of the following general document form.

- Query-Type (See above)
- Cup-Year (All, 1930, 1934, 1938, ...)
- Team-Name (All, Brazil, Germany, Italy, ...)
- Player-Name (All, Carbajal, Lahm, Klose, ...)
- Want-Match-Info (True, False)
- Match-Query-Info
  - Winning-Team (Any, Brazil, Germany, Italy, ...)
  - Losing-Team (Any, Brazil, ...)
  - Get-Match-With-Most-Goals-For-Winning-Team (True, False)
- Want-Penalty-Cards (True, False)

6.3 Database Results

The Database Results are the exact results from MondialDB. The Result-Data will be in the format that MondialDB returns.

- Query-Type (See section 6.2 on page 15 for the list of queries)
- Result-Type (See section 6.4 on page 16 for the list of queries)
- Result-Data

6.4 Formatted Results

The types and document forms for Formatted Results:

- RQGC (Result for Query for a Given Cup)
  - Cup-Year
  - Number-Goals-Per-Team-List
  - Team-Name-List
  - Team-Position-List
- RQGP (Result for Query for a Given Player)
  - Player-Name
  - Cup-Name-List
  - Number-Goals-Per-Cup-List
  - Number-Penalties-Per-Cup-List
  - Team-Name-List
- RQSS (Result for Query for Super Stars)
  - Player-Name
  - Cup-Name-List
  - Number-Goals-Per-Cup-List
  - Number-Penalties-Per-Cup-List
  - Team-Name-List
- RQTH (Result for query for Team History)
  - Team-Name
  - Cup-Name-List
  - Number-Goals-Per-Cup-List

- RQCP (Result for query for Country's Players)
  - Team-Name
  - Player-List
  - Number-Goals-Per-Player-List
  - Number-Penalties-Per-Player-List
- RQPCC (Result for Query for Penalty Cards in a Cup)
  - Cup-Year
  - Number-Penalties-Per-Team-List
  - Team-Name-List
- RQPCT (Result for query for Penalty Cards for a team)
  - Team-Name
  - Cup-Name-List
  - Number-Goals-Per-Cup-List
  - Number-Penalties-Per-Cup-List
- RQGM (Result for Query for Given Match)
  - Winning-Team-Name
  - Losing-Team-Name
  - Cup-Name
  - Winning-Team-Goals
  - Winning-Team-Penalties
  - Losing-Team-Goals
  - Losing-Team-Penalties
- RQMMG (Result for query for Matches with the Most Goals)
  - Winning-Team-Name
  - Losing-Team-Name
  - Cup-Name
  - Winning-Team-Goals
  - Losing-Team-Goals

6.5 Soccer Websites

The Soccer Websites will be scraped for World Cup data to insert into MondialDB.

- SW-ID
- URL
- Publisher

6.6 Unformatted Data

The Unformatted Data is extracted data from the Soccer Websites. The HTML-Selector is the exact path of where to find the data on the relevant webpage. The access date is included because websites change over time.

- SW-ID
- Access-Date
- HTML-Selector
- Data-Value

6.7 Formatted Data

The Formatted Data is a transformation of the Unformatted Data into something that MondialDB will accept.

- SW-ID
- Access-Date
- Number-Goals
- Number-Penalties
- Cup-Year (All, 1930, 1934, 1938, ...)
- Team-Name (All, Brazil, Germany, Italy, ...)
- Player-Name (All, Carbajal, Lahm, Klose, ...)
- Is-Match (True, False)
- Match-Data
  - Winning-Team (Brazil, Germany, Italy, ...)
  - Losing-Team (Brazil, ...)
  - Winning-Team-Goals
  - Losing-Team-Goals

6.8 SQL Script

The SQL Script is the result from the Data Loading Task (ETLT.3). It will be run on MondialDB by the python script handling ETLT.3.

- File-Name
- Date-Executed
- File-Contents