CPSC 304 Project Cover Page

Milestone #: 3
Date: Wed, Nov 2, 2022
Group Number: 11

Name	Student Number	CS Alias (UserID)	Preferred Email Address
Kyle Rich	36316933	h7w2nb	kyledvrich@gmail.com
Eric Liu	95724787	p3v2b	liu.eric103@gmail.com
Anisha Gill	27909993	r4h0d	gill.anisha@outlook.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.

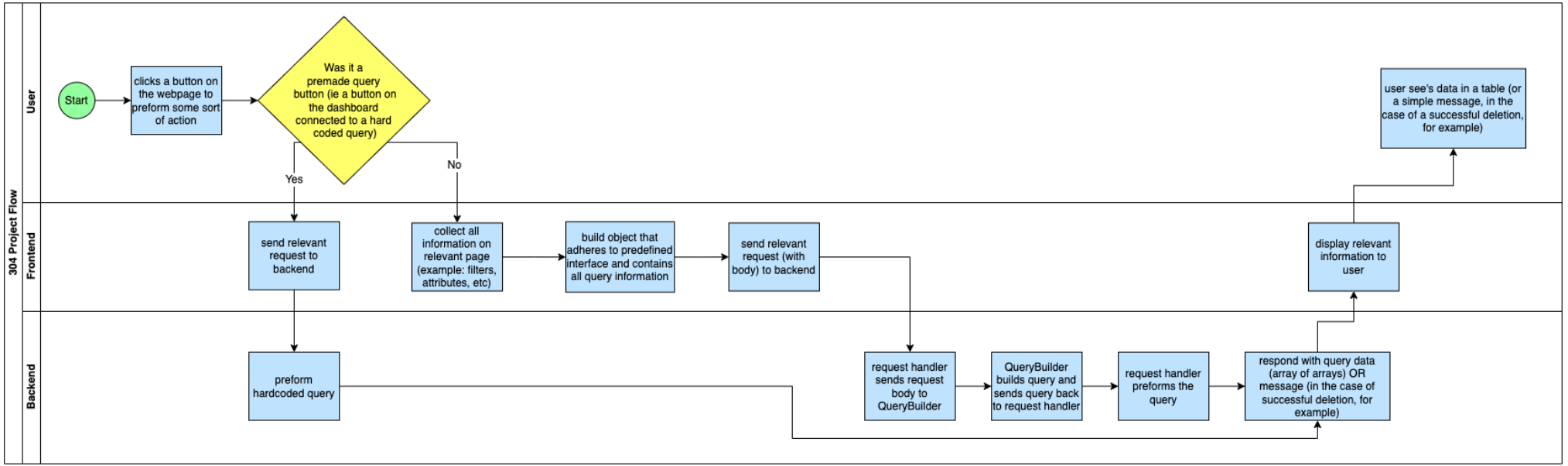
In a general sense, Kyle will implement the overall structure and data flow of the entire project. And Anisha/Eric will do most of the actual implementation, with Kyle supporting where necessary. Overall, we will strive to achieve a 33/33/33% split workload. And will make adjustments as we see fit throughout the project in order to maintain an evenly split workload.

Task	Assigned To	Due Date	Time Needed
Create .sql script to populate the database with dummy data upon running the program	Eric	Nov 7	Medium
Decide on some ideas for the query buttons to fulfill the aggregation etc. requirements (its okay if these are not 100% ready yet)	Anisha	Nov 7	Low
Create backend function stubs for request all handlers (example: handlers for deleting data, inserting data, query buttons, and user defined queries, etc)	Kyle	Nov 7	Low
Create frontend function stubs for sending the requests (these requests will be sent upon a user pressing a button)	Kyle	Nov 7	Low
Create all necessary server routes	Kyle	Nov 7	Low
Create a query builder class in the backend with function stubs	Kyle	Nov 7	Low
Create an interface(s) that can be used inside the request bodys to send necessary information from the frontend to the backend.	Kyle	Nov 7	Low
Implement the query builder (this query builder will take in an interfaced object described above, and output the corresponding SQL statement)	Eric	Nov 14	High
implement the backend request handler stubs	Anisha	Nov 14	High
Write the premade queries (in the dashboard that fulfill the ‘hard coded’ query requirements) and perform those queries in the relevant request handlers	Anisha	Nov 14	Medium

Develop the frontend html pages and simple JS routes between them. Focus will be on buttons/user input first, to allow for testing, then focus on prettying everything up after.	Kyle (and possibly Eric/Anisha, depending on workload)	Nov 14	High
Create click handler function stubs and describe a clear way to access all of the relevant information once a button is pressed on a certain page (either a filter button, delete button, etc)	Kyle	Nov 14	Low
Implement the frontend button handler and request stubs	Eric + Anisha	Nov 21	High
Bug Fixes / Wrapping up loose ends / improving user experience / assisting where possible	Kyle	Nov 21	Low
Project Submission and any relevant documentation requested by the assignment details	Eric + Anisha	Nov 24	Medium
Any missing implementation that was not finished before Nov 24 (if it wasnt necessary for getting full marks on Milestone 4) Bug fixes / improvements to UI etc.	All	Dec 2	Low

If I missed anything above, it's because it is likely implicitly defined in my head – NOT because we do not have a solid understanding of what is necessary for the project.

We pretty much just need to implement everything in order to enable the entire flow listed below, in addition to making the UI look nice enough.


304 Project Flow
User
Frontend
Backend
Start
clicks a button on the webpage to preform some sort of action
Was it a premade query button (ie a button on the dashboard connected to a hard coded query)
Yes
No
send relevant request to backend
collect all information on relevant page (example: filters, attributes, etc)
build object that adheres to predefined interface and contains all query information
send relevant request (with body) to backend
preform hardcoded query
request handler sends request body to QueryBuilder
QueryBuilder builds query and sends query back to request handler
request handler preforms the query
respond with query data (array of arrays) OR message (in the case of successful deletion, for example)
display relevant information to user
user see's data in a table (or a simple message, in the case of a successful deletion, for example)

Frame 1

Filter | Join | Dashboard | Insert | Update | Delete

Formula 1 Database

Filtering Data

From table

Tables

Filters

Arributes | Operators | Value

\+ -

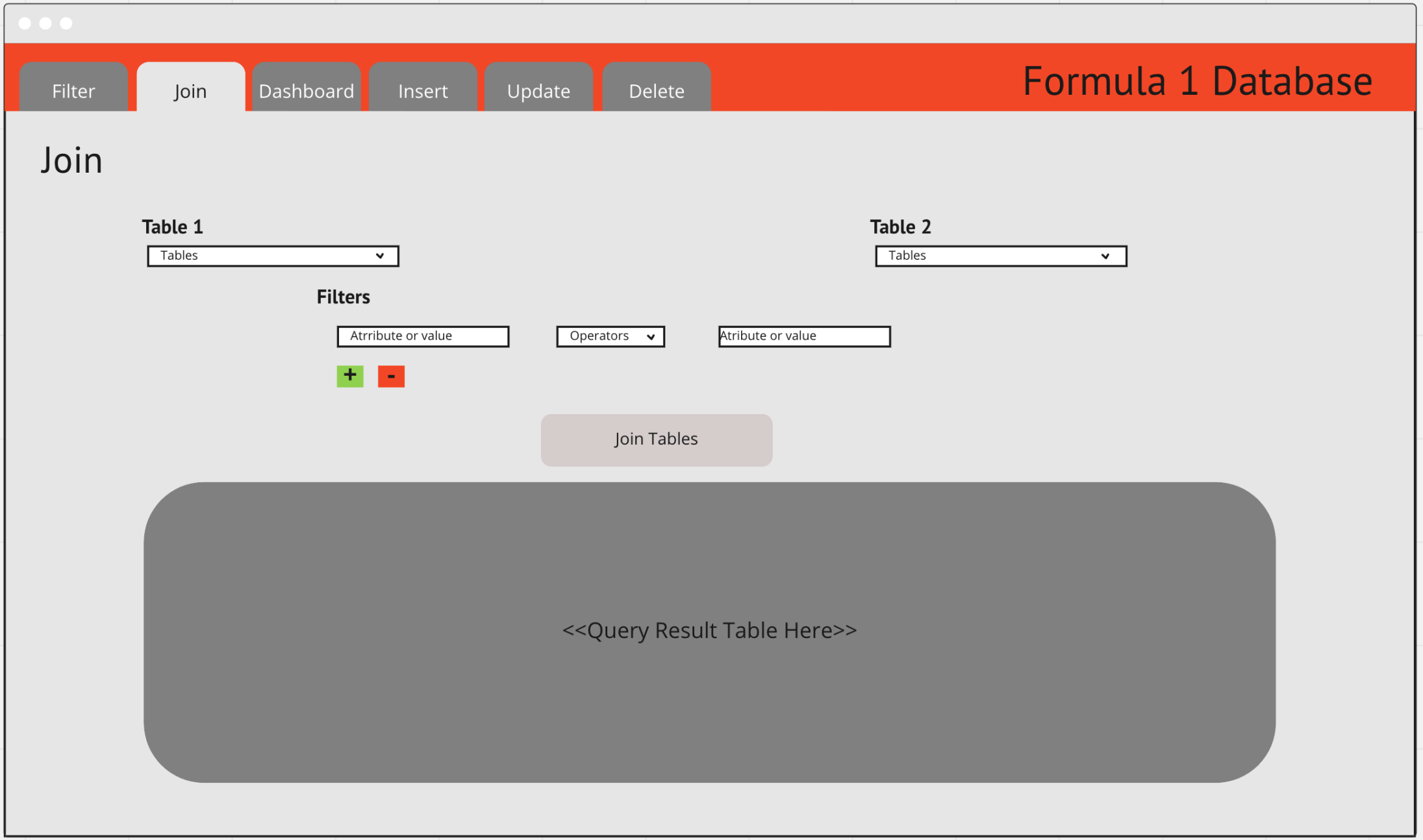
Columns

- Attribute 1
- Attribute 2
- Attribute 3
- Attribute 4
- Attribute 5
- Attribute 6

Get Filtered Results

<<Query Result Table Here>>

Frame 2


Filter
Join
Dashboard
Insert
Update
Delete
Formula 1 Database
Join
Table 1
Tables
Table 2
Tables
Filters
Atrribute or value
Operators
Atribute or value
+
-
Join Tables
<<Query Result Table Here>>

Frame 3

Filter | Join | Dashboard | Insert | Update | Delete

Formula 1 Database

Useful Everyday Query Buttons

<<Aggregation with group>>

<<Name 1>>

<<Nested Aggregation with Group>>

<<Name 3>>

Surprise Query

<<Aggregation with Having>>

<<Name 2>>

<<Division>>

<<Name 4>>

Surprise Query

<<Query Result Table Here>>

Frame 4

Filter | Join | Dashboard | Insert | Update | Delete

Formula 1 Database

Add Data

Table

Tables

Values

Attributes	Value
Attributes	Value
Attributes	Value

Insert Row

<<Entry Result Here>>

Frame 5

Filter | Join | Dashboard | Insert | Update | Delete

Formula 1 Database

Update Data

Table

Tables

Filters

Arributes | Operators | Value

\+ -

New Values

Attributes | Value

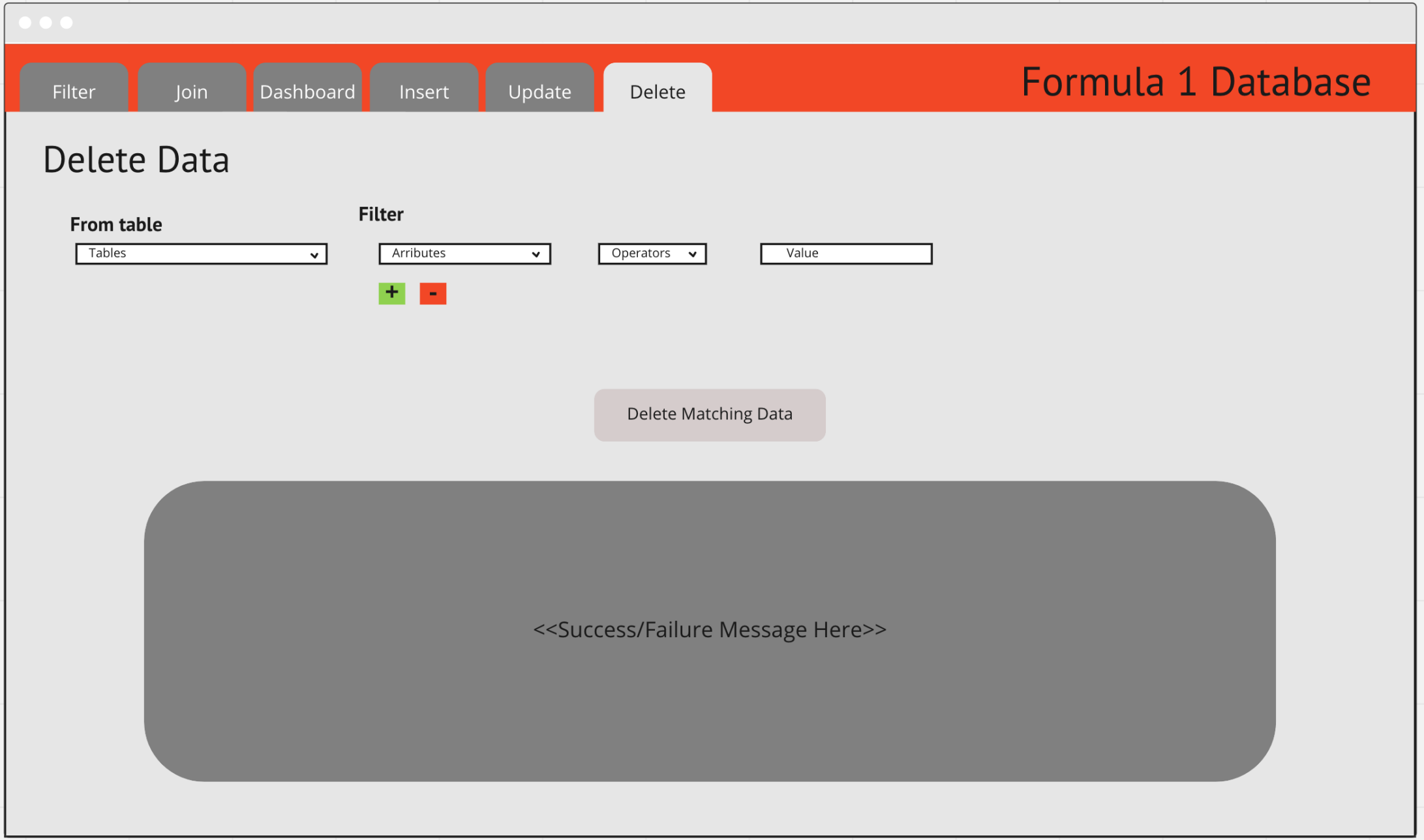
Attributes | Value

Attributes | Value

Update Data

<<Updated Result Here>>

Frame 7


Filter
Join
Dashboard
Insert
Update
Delete
Formula 1 Database
Delete Data
From table
Tables
Filter
Arributes
Operators
Value
+
-
Delete Matching Data
<<Success/Failure Message Here>>