Matisse Capital Database

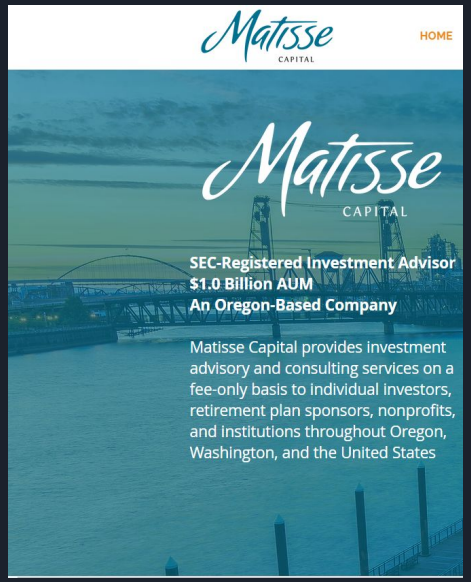
By Daniela Bresler, Peter Vaupel, Joe Torkelson, Riley Barry

Outline

1. Company and Purpose
2. Entity Relationship Diagram
3. Entity Relational Model
4. Front End Application
5. What we learned

Matisse Capital

- A company that provides investment advisory and consulting services for individual investors, retirement plan sponsors, nonprofits, and institution
- Based in Oregon, Washington, and the United States
- **Accounts**
  - Retirement Plan Management
  - Wealth Management
  - Non-profit Management
  - Closed-End Fund Research & Portfolio Management



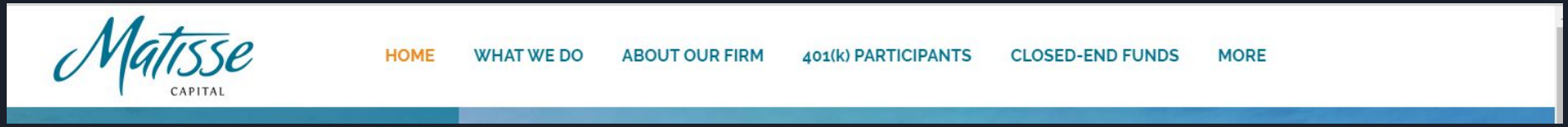
Purpose

Our Purpose:

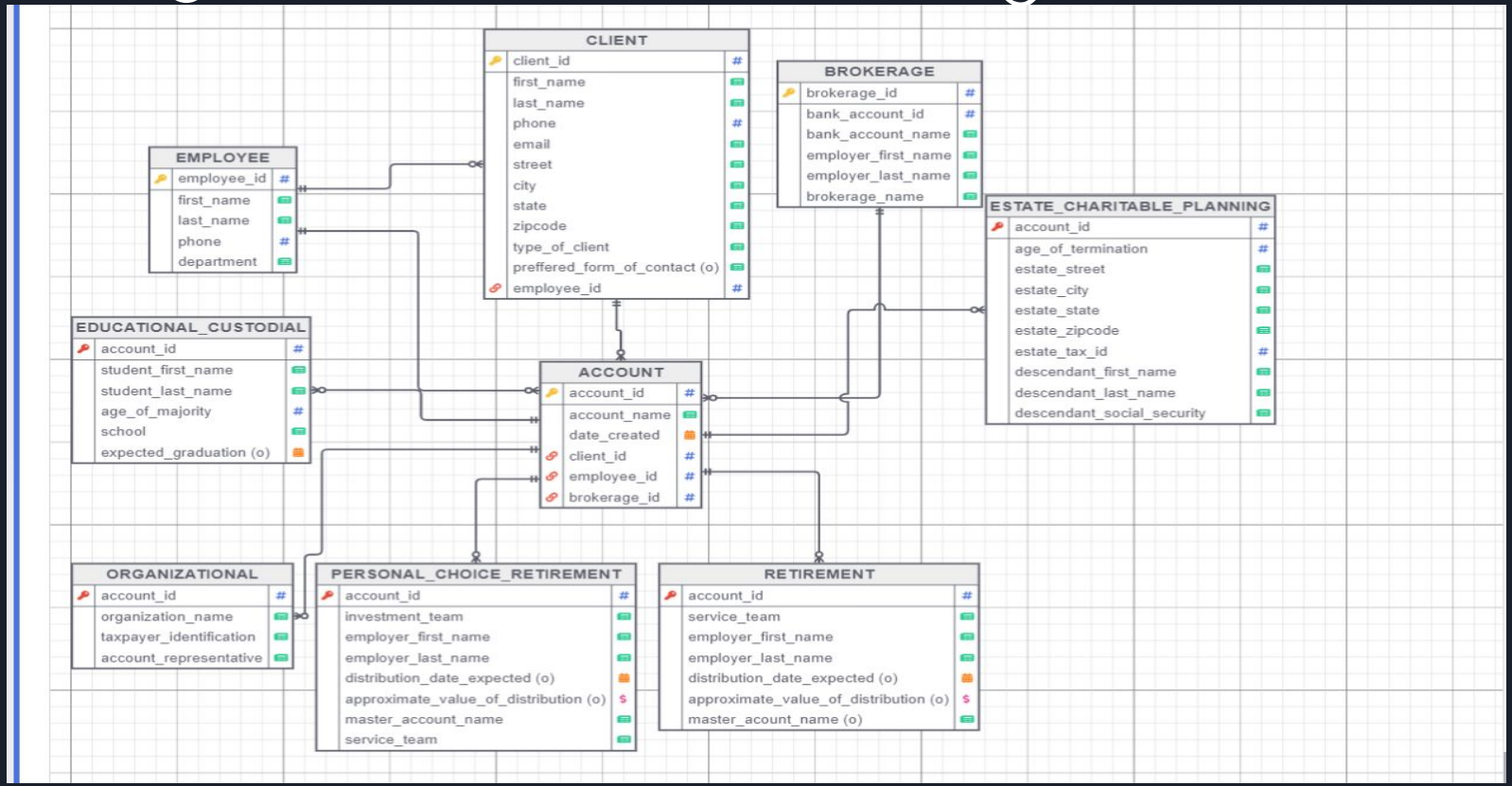
Wanted to create a database that gave clients and employees access their information and help store and organize data more efficiently

What we Did:

Created a database to help clients and employees track and access their information

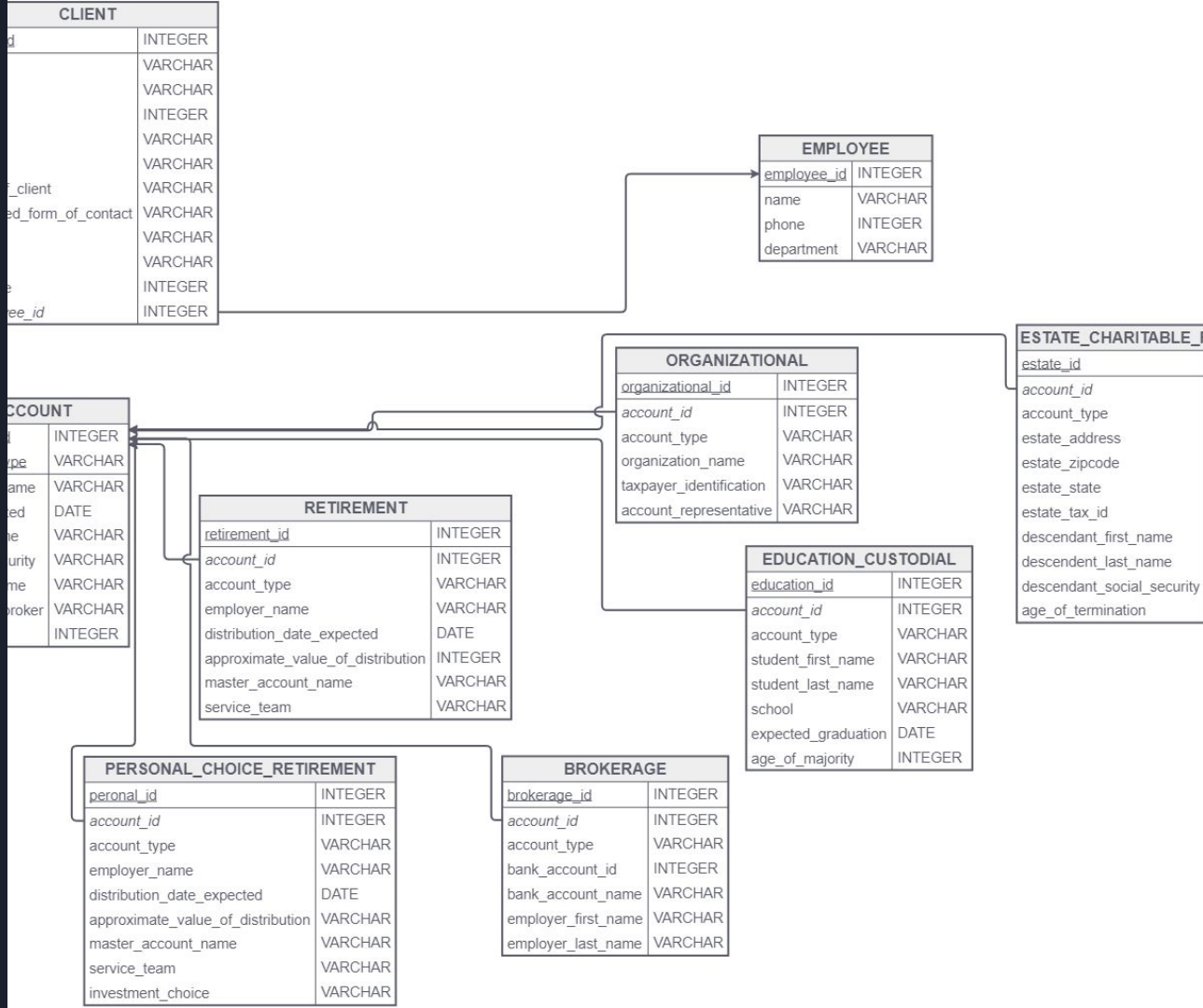


Entity Relationship Diagram



Relational Model:

- Account-types connected to Account entity
- Account entity connected to Client Entity



Front End Application

Employee Accessed Database

Login Page

Any potential user must register before reaching ...

Home Page

- Display Client Information
- Display all Accounts Per Brokerage
- Display Accounts Managed by Each Employee
- Insert new client
- Display A Graph

https://ormond.creighton.edu/courses/354/Groups/Group-A4/structure/matissecapital-login.php

Queries not Incorporated

A possible query we might execute would be to display all of the client names with a retirement account.

```
SELECT first_name, last_name, r.account_id

FROM employee e

JOIN account a ON e.employee_id = a.employee_id

JOIN retirement r ON a.account_id = r.account_id;
```

Display all accounts associated with TD Ameritrade.

```
SELECT brokerage_name, account_id, first_name, last_name
FROM brokerage
JOIN account USING(brokerage_id)
JOIN client USING(client_id)
WHERE brokerage_name = 'TD Ameritrade';
```

Procedures and Views

Create a procedure called add account that adds an additional account to an existing client.

```
DELIMITER $$
CREATE PROCEDURE `add_account`(IN vaccount varchar
BEGIN
 UPDATE `account`
 SET account_name = `vaccount`
 WHERE account_name = `organizational`,
`personal_choice_retirement`,`educational_custodial`,
`estate_charitable_planning`, `retirement`;
 SELECT `first_name`, `last_name`, `account_id`, `account_name`
 FROM account
 WHERE account_name = `vaccount`;
END$$
DELIMITER;
```

A possible view that would be applicable to Matisse's database would be a view that displays all accounts and their affiliated brokerages.

```
CREATE OR REPLACE VIEW
`accounts_and_brokerages` AS
SELECT a.account_id, b.brokerage_name
FROM account a
JOIN brokerage b ON a.brokerage_id =
b.brokerage_id;
```

Value Added Feature

Pie Chart:

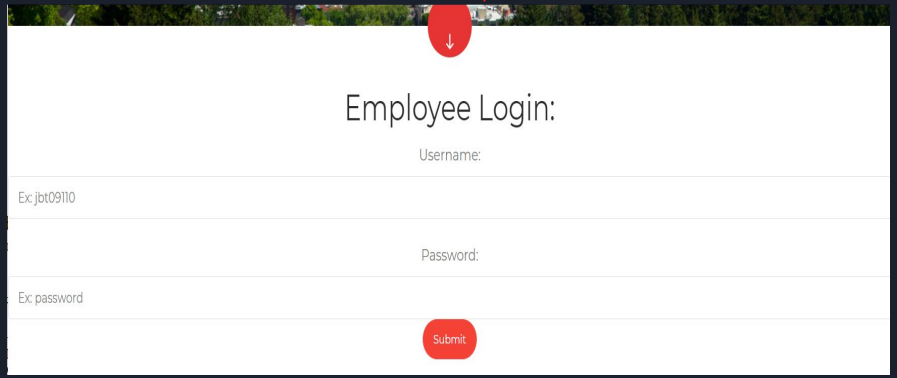
- We added the pie chart as one of the value added features. This helps show accounts clients are using

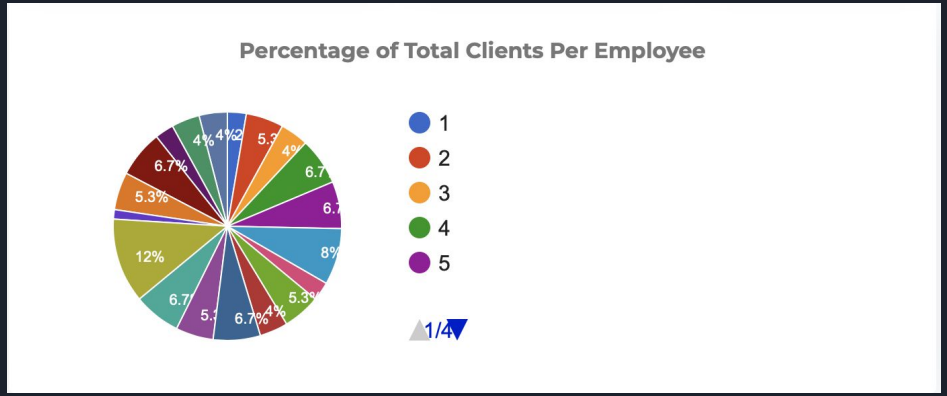
Login:

- The login feature helps all clients and employees enter the database

Storing Files

- This feature can help clients electronically store paper forms and contracts





Reflection

What we learned:

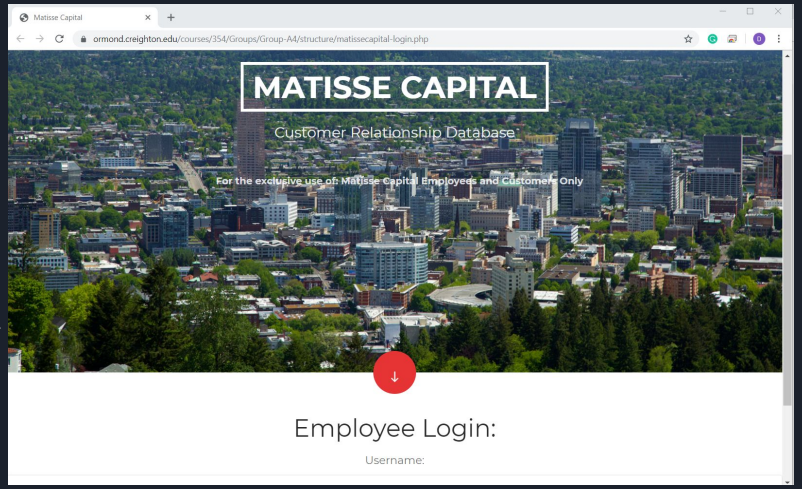
- We grew a lot in our knowledge about working with a team and building our skills of sql and php

What we talked about:

- Help make it easier for clients to get information

What we would have done differently:

- Plan out the pieces of the code and website before creating them




```
#A description of any (several more than three) queries you would need to perform against your database with a brief explanation of why this query is important. After identifying all queries, include the SQL for at least three of these queries. Two of these queries need to demonstrate the joining of two or more tables.
   #Display all account ids with basic client information (client_id, account_id, brokerage_id). This is important when initially speaking with a client to get a basic sense of their standing with the company.
SELECT client_id, first_name, last_name, account_id, brokerage_id
FROM client
RIGHT OUTER JOIN account USING(client_id)
RIGHT OUTER JOIN brokerage USING(brokerage_id);
   #Display all accounts associated with TD Ameritrade.  If TD Ameritrade comes to us with questions regarding their associated accounts with Matisse Capital, this querey can show quick results with helpful information regarding their accounts and relationship.
SELECT brokerage_name, account_id, first_name, last_name
FROM brokerage
JOIN account USING(brokerage_id)
JOIN client USING(client_id)
WHERE brokerage_name = 'TD Ameritrade';
    #Display all accounts linked to employees including client information. This is an important query when trying to find the employee associated with a client account in case a client needs their representive's information.
SELECT e.employee_id, e.first_name AS e_first_name, e.last_name AS e_last_name, c.first_name AS c_first_name, c.last_name AS c_last_name, account_id, account_name
FROM employee e
LEFT OUTER JOIN client c USING(employee_id)
JOIN account USING(client_id);
```

Questions