

# Saudi America

## Portfolio Project

### Executive Summary

The database design and implementation project has undergone a significant evolution since its inception, driven by iterative feedback and continuous improvement efforts. Initially, the project set out with a comprehensive schema aligned with the database outline and ER diagram, emphasizing consistency in naming conventions, table structures, and data types. However, as the project progressed, peer reviewers provided invaluable insights that guided several key enhancements.

One notable area of focus was the readability and organization of the schema. Peer feedback prompted a re-evaluation of naming conventions and schema layout to ensure clarity and ease of understanding. This led to refinements in the naming of entities and attributes, as well as improvements in the organization of tables within the schema. Additionally, suggestions were incorporated to enhance normalization and address potential non-normalized issues.

Throughout the implementation phase, peer feedback played a pivotal role in evaluating the syntactical correctness of SQL files and ensuring adherence to best practices. Recommendations for optimizing SQL structure, including the incorporation of CASCADE operations to maintain referential integrity, were carefully considered and implemented. These enhancements not only improved the overall integrity of the database but also facilitated smoother data management and manipulation processes.

Feedback from peer reviewers guided the implementation of CRUD functionalities for each table in the schema, ensuring comprehensive data management capabilities. Suggestions for incorporating dynamically populating dropdown lists were integrated to streamline user interactions and improve efficiency. Furthermore, recommendations for enhancing data representation through JOIN queries to display more meaningful information were taken into account. By leveraging JOINS and aliases, the UI was optimized to present relevant data in a user-friendly format, improving usability.

Overall, the project's progression from its initial proposal to its current state reflects a collaborative and iterative approach to database design and implementation. Through continuous feedback and refinement, the project has evolved into a refined and efficient database solution that meets both functional requirements and user expectations. The culmination of these efforts resulted in the successful engineering of a fully functional website equipped with CRUD functionalities and CSS styling, implemented using Express.js and Node.js. These enhancements demonstrate a commitment to delivering a high-quality database solution that is both robust and user-friendly, ultimately contributing to the project's success and effectiveness.

# Project Overview

Saudi America, a burgeoning brokerage startup, is dedicated to bridging the gap between Middle Eastern and American markets, catering to the needs of everyday retail investors. With a substantial clientele of approximately 100,000 users, the platform facilitates an impressive \$5,000,000,000 USD in trades each month. To efficiently manage and streamline their operations, Saudi America requires a robust database-backed system. This system would play a pivotal role in tracking and organizing a multitude of crucial data points, including but not limited to orders, securities, positions, and investor information. The database will enable seamless transaction tracking, portfolio management, and client engagement, providing the scalability needed to accommodate the dynamic nature of a rapidly growing user base and transaction volume.

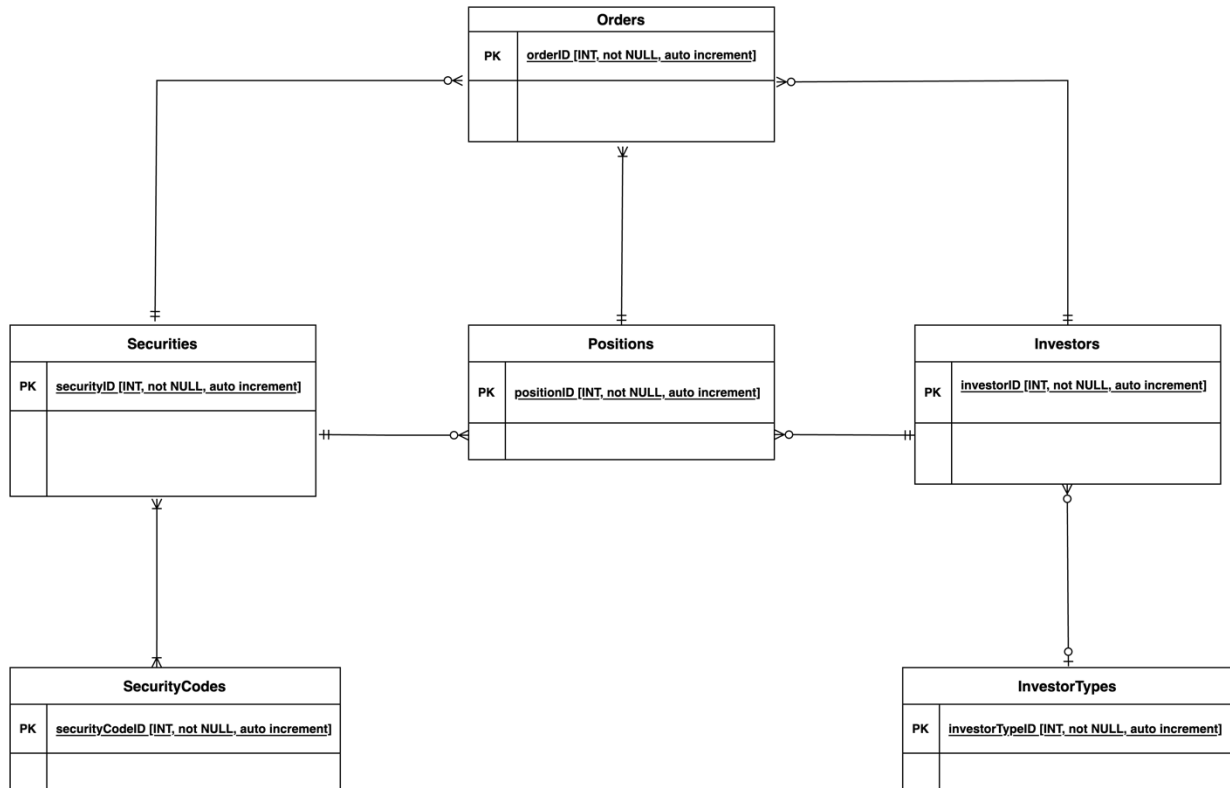
## Database Outline

- **Orders:** Describes an order, the buying or selling, of a particular security.
  - orderID [PK, INT, not NULL, auto increment]
  - investorID [FK, INT, not NULL]
  - securityID [FK, INT]
  - positionID [INT, not Null]
  - orderType [VARCHAR, not NULL]
  - totalUnits [decimal (65, 2)]
  - price [decimal (65, 2), not NULL]
  - tradeTotal [decimal (65, 2), not NULL]
  - date [DATE, not NULL]
  - *Relationships:*
    - **Orders -> Investors = M:1 (Optional Relationship)**
      - Many orders may be associated with a given investor, but only one investor must be associated with any given order.
    - **Orders -> Securities = M:1 (Optional Relationship)**
      - Many orders may be associated with a given stock, but only one stock must be associated with any given order.
    - **Orders -> Positions = M:1 (Optional Relationship)**
      - Many orders may be associated with a given position, but only one position may be associated with any given order.
- **Securities:** Describes a security's name, ticker, current price, and other important info.
  - securityID [PK, INT, not NULL, auto increment]
  - name [VARCHAR, not NULL]
  - ticker [VARCHAR, not NULL]
  - currentPrice [decimal (65, 2), not NULL]
  - *Relationships:*
    - **Securities -> Orders = 1:M (Optional Relationship)**
      - Only one security may be associated with a given order, but many orders may be associated with a given security.

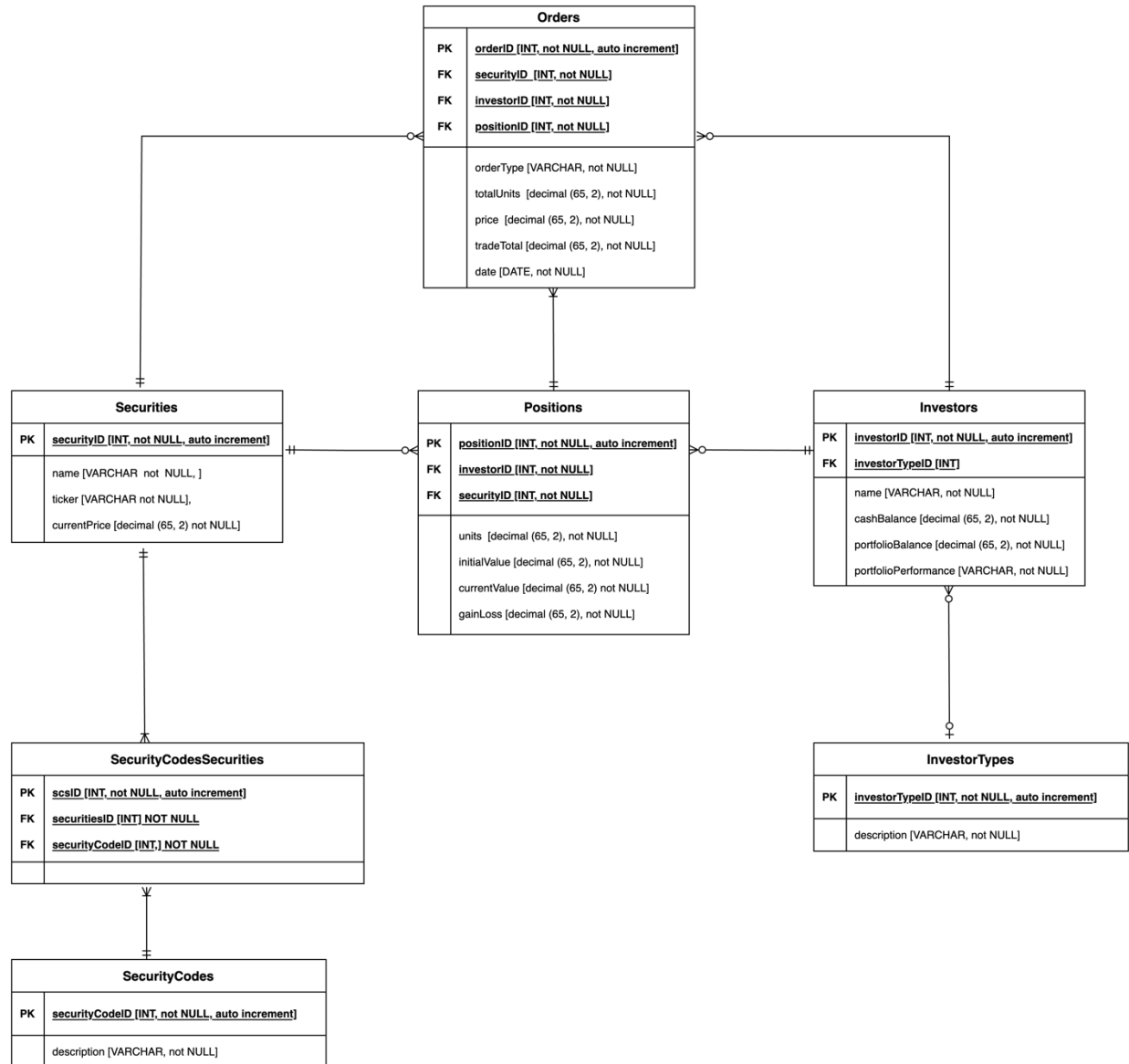
- **Securities -> Positions = 1:M (Mandatory Relationship)**
    - Only one security must be associated with a given position, but many positions may be associated with a given security.
  - **Securities -> SecurityCodes = M:N (Optional Relationship)**
    - A given security may be associated with many security codes, and a given security code may be associated with many securities.
- **SecurityCodes: Category table used to further describe each security, such as type.**
  - securityCodeID [PK, INT, not NULL, auto increment]
  - description [VARCHAR, not NULL]
  - *Relationship*:
    - **SecurityCodes -> Securities = M:N (Optional Relationship)**
      - A given securities code may be associated with many securities and a given security may be associated with many securities codes.
- **SecurityCodesSecurities: Intersection table joining SecurityCodes with Securities.**
  - scsID [PK, INT, not Null, auto increment]
  - securityID [FK, INT]
  - securityCodeID [FK, INT]
  - *Relationships*:
    - **SecurityCodesSecurities -> SecurityCodes: = M:1 (Mandatory Relationship)**
      - Many SecurityCodesSecurities items may be associated with one security code, but only one security code may be associated with a given SecurityCodesSecurities code item.
    - **SecurityCodesSecurities -> Securities = M:1 (Mandatory Relationship)**
      - Many SecurityCodesSecurities items may be associated with one security, but only one security may be associated with a given SecurityCodesSecurities item.
- **Investors: Describes an investor who trades on our platform – name, investor type, various balances, and their portfolio performance.**
  - investorID [PK, INT, not NULL, auto increment]
  - investorType [FK, INT]
  - name [VARCHAR, not NULL]
  - cashBalance [decimal (65, 2), not NULL]
  - portfolioBalance [decimal (65, 2), not NULL]
  - portfolioPerformance [decimal (65, 2), not NULL]
  - *Relationships*:
    - **Investors -> Orders = 1:M (Mandatory Relationship)**
      - Only one investor must be associated with a given order, but many orders may be associated with a given investor.
    - **Investors -> InvestorTypes = M:1 (Optional Relationship)**
      - Many investors may be associated with a given investor type, but only one investor type must be associated with a given investor.

- **Investors -> Positions = 1:M (Mandatory Relationship)**
    - Only one investor must be associated with a given position, but many positions may be associated with one investor.
- **InvestorTypes: Describes the type of investor who trades on our platform – individual, bank, hedge fund, broker, or market maker.**
  - investorTypeID [PK, INT, not NULL, auto increment]
  - description [VARCHAR, not NULL]
  - *Relationship*:
    - **InvestorTypes -> Investors = 1:M (Optional Relationship)**
      - Only one investor type must be associated with a given investor, but many investors may be associated with a given investor type.
- **Positions: Describes an investor's position in a particular security, in terms of the number of units (meaning shares or tokens) owned and the cumulative value of those units.**
  - positionID [PK, INT, not NULL, auto increment]
  - investorID [FK, INT, not NULL]
  - securityID [FK, INT, not NULL]
  - units [decimal (65, 2), not NULL]
  - initialValue [decimal (65, 2), not NULL]
  - currentValue [decimal (65, 2), not NULL]
  - gainLoss [decimal (65, 2), not NULL]
  - *Relationships*:
    - **Positions -> Securities = M:1 (Optional Relationship)**
      - Many positions may be associated with a given security, but only one security must be associated with a given position.
    - **Positions -> Investors = M:1 (Optional Relationship)**
      - Many positions may be associated with a given investor, but only one investor must be associated with a given position.
    - **Positions -> Orders = 1:M (Mandatory Relationship)**
      - Only one position must be associated with a given order, but many orders may be associated with a given position.

# Entity-Relationship Diagram



# Schema



## UI Screenshots

### HOME – NO CRUD OPERATIONS



# SAUDI AMERICA

HOME | ORDERS | SECURITIES | SECURITY CODES | SECURITY CODES SECURITIES | INVESTORS | INVESTOR TYPES | POSITIONS

## Orders

ID	INVESTOR	SECURITY	POSITION	ORDER TYPE	TOTAL UNITS	PRICE	TRADE TOTAL	DATE
1	Keith Gill	GameStop Corp.	1	Buy	200000	1	200000	Tue Apr 20 2021 00:00:00 GMT-0700 (Pacific Daylight Time)
2	Axe Capital	SPDR S&P 500 ETF Trust	2	Buy	500000	360	180000000	Thu Jun 15 2023 00:00:00 GMT-0700 (Pacific Daylight Time)
3	Axe Capital	Tesla Inc.	3	Buy	700000	250	175000000	Thu Jun 15 2023 00:00:00 GMT-0700 (Pacific Daylight Time)
4	J.P. Morgan Chase & Co.	NVIDIA	4	Buy	1000000	400	400000000	Fri Oct 27 2023 00:00:00 GMT-0700 (Pacific Daylight Time)
5	J.P. Morgan Chase & Co.	Bitcoin	5	Buy	100000	25000	2500000000	Fri Oct 27 2023 00:00:00 GMT-0700 (Pacific Daylight Time)
6	Axe Capital	Tesla Inc.	3	Sell	600000	260	156000000	Wed Dec 27 2023 00:00:00 GMT-0800 (Pacific Standard Time)
7	J.P. Morgan Chase & Co.	Bitcoin	5	Sell	50000	40000	2000000000	Wed Jan 24 2024 00:00:00 GMT-0800 (Pacific Standard Time)

### Add Order

Investor

Security

Position (New)

Buy

Total Units

Price

mm/dd/yyyy

Submit



# SAUDI AMERICA

HOME | ORDERS | SECURITIES | SECURITY CODES | SECURITY CODES SECURITIES | INVESTORS | INVESTOR TYPES | POSITIONS

## Securities

ID	NAME	TICKER	CURRENT PRICE
1	Apple Inc.	AAPL	190
2	Bitcoin	BTC	43000
3	Ethereum	ETH	2400
4	GameStop Corp.	GME	14
5	Immutable	IMX	2
6	Invesco QQQ Trust	QQQ	430
7	NVIDIA	NVDA	680
8	SPDR S&P 500 ETF Trust	SPY	500
9	Tesla Inc.	TSLA	185
10	Vanguard Total Stock Market ETF	VTI	250

### Add Security

Name

Ticker

Current Price

Submit

# SAUDI AMERICA

HOME | ORDERS | SECURITIES | SECURITY CODES | SECURITY CODES SECURITIES | INVESTORS | INVESTOR TYPES | POSITIONS

## Security Codes

ID	DESCRIPTION
1	Cryptocurrency
2	Stock
3	Exchange Traded Fund
4	Popular/trending securities
5	Stock will report quarterly earnings this week
6	Ethereum-based cryptocurrency

### Add Security Code

Enter Description

Submit

# SAUDI AMERICA

HOME | ORDERS | SECURITIES | SECURITY CODES | SECURITY CODES SECURITIES | INVESTORS | INVESTOR TYPES | POSITIONS

## Security Codes Securities

Note: This is an intersection table. For a complete list of securities and security codes, please refer to the 'Securities' and 'Security Codes' pages.

ID	SECURITY	SECURITY CODE		
1	Apple Inc.	Cryptocurrency	Delete	Update
2	Bitcoin	Stock	Delete	Update
3	Bitcoin	Stock	Delete	Update
4	Ethereum	Exchange Traded Fund	Delete	Update
5	GameStop Corp.	Popular/trending securities	Delete	Update
6	Immutable	Stock will report quarterly earnings this week	Delete	Update
7	Immutable	Stock will report quarterly earnings this week	Delete	Update
8	Invesco QQQ Trust	Ethereum-based cryptocurrency	Delete	Update

### Add SCS

Security

Security Code


Submit

# SAUDI AMERICA

[HOME](#) | [ORDERS](#) | [SECURITIES](#) | [SECURITY CODES](#) | [SECURITY CODES SECURITIES](#) | [INVESTORS](#) | [INVESTOR TYPES](#) | [POSITIONS](#)

## Update SCS

Security:

Select Security 

Security Code:

Select Security Code 

Update

# SAUDI AMERICA

HOME | ORDERS | SECURITIES | SECURITY CODES | SECURITY CODES SECURITIES | INVESTORS | INVESTOR TYPES | POSITIONS

## Investors

ID	INVESTOR TYPE	NAME	CASH BALANCE	PORTFOLIO BALANCE	PORTFOLIO PERFORMANCE
1	Bank	J.P. Morgan Chase & Co.	2000000000	8950000000	4450000000
2	Hedge Fund	Axe Capital	1560000000	2685000000	695000000
3	Individual	Keith Gill	0	2800000	2600000
12		Ting Ting	0	0	0
14		Superman	0	0	0
15		Two Sigma	0	0	0

## Add Investor

Investor Type

▼

Name

Cash Balance

Portfolio Balance

Portfolio Performance

Submit

# SAUDI AMERICA

HOME | ORDERS | SECURITIES | SECURITY CODES | SECURITY CODES SECURITIES | INVESTORS | INVESTOR TYPES | POSITIONS

## Investor Types

ID	DESCRIPTION
1	Bank
2	Hedge Fund
3	Individual
4	Broker
5	Market Maker
6	

### Add Investor Type

Enter Description

Submit

# SAUDI AMERICA

## Positions

ID	INVESTOR	SECURITY	UNITS	INITIAL VALUE	CURRENT VALUE	GAIN / LOSS	
1	Keith Gill	GameStop Corp.	200000	200000	2800000	2600000	<button>Update</button>
2	Axe Capital	SPDR S&P 500 ETF Trust	500000	180000000	250000000	70000000	<button>Update</button>
3	Axe Capital	Tesla Inc.	100000	175000000	18500000	-500000	<button>Update</button>
4	J.P. Morgan Chase & Co.	NVIDIA	1000000	400000000	680000000	280000000	<button>Update</button>
5	J.P. Morgan Chase & Co.	Bitcoin	50000	250000000	215000000	165000000	<button>Update</button>

## Add Position

Investor

Security

Units

Initial Value

Current Value

Gain/Loss

Submit

# SAUDI AMERICA

HOME | ORDERS | SECURITIES | SECURITY CODES | SECURITY CODES SECURITIES | INVESTORS | INVESTOR TYPES | POSITIONS

## Update Position

Position ID, Investor Name, Security:

1 - Keith Gill - GameS

▼

Units:

Units

Initial Value:

Initial Value

Current Value:

Current Value

Gain/Loss:

Gain/Loss

Update