Bank Project CS 611 Final Project

George Padavick, Shilpen Patel, Justin DiEmannuele, Matt Gilgo

Requirements Specification

- Generic Requirements:
 - There will be two types of users interfacing with the bank: a Customer and Manager that will each have secure login credentials.
 - Users should be able to easily navigate the functionalities of the Bank through an easy to use Graphical User Interface (GUI)
- Customer Requirements:
 - Must be able to create/delete checking, savings, trading, and loan accounts
 - Must be able to manipulate each type of account
- Manager Requirements:
 - o Must be able see who the customers are and see what transactions the customers have made
 - Must be able to maintain and manipulate the stock market
- Account Requirements:
 - There should be four types of accounts: Savings, Checking, Loans, and Brokerage accounts
- Stock Market Requirements:
 - The Bank Manager would like the bank to offer brokerage services to customers who have accounts that satisfy certain metrics.
 - Customers with brokerage accounts should be able to trade stocks in the stock market and see realized and unrealized gains
- Persistence Requirement
 - o if bank service goes down, the user information/money should not go away

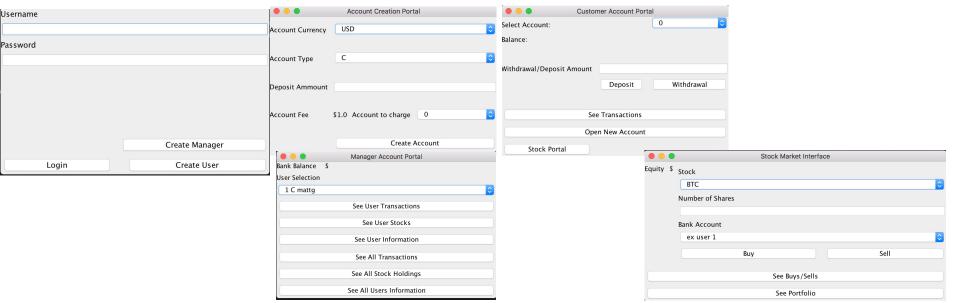
Database Design

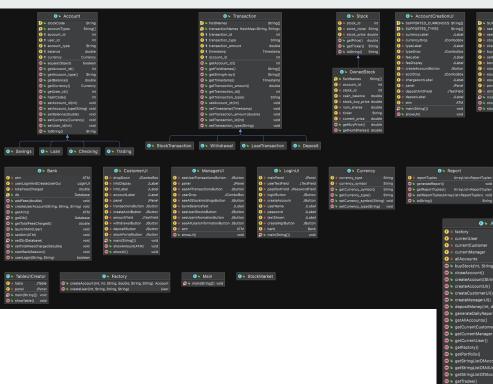
- Five tables developed to retain information needed for application
 - Users all users including Customers and Managers
 - o Accounts stores all accounts include savings, checking, stocks, and loan accounts
 - Transactions log of all transactions in the database including withdrawals, deposits, stock trades
 - Stocks master list of stocks, allows manager to update stock prices
 - o Stocks owned manages stock trading capability by tracking the current shares for each user

users		accounts		transactions		stocks		stocks_owned	
PK	user_id	РК	account id	PK	transaction id	PK	stock id	РК	stock instance owned id
	user_type		user_id		transaction_type		stock_ticker		account_id
	username		account_type		transaction_amount		stock_price		stock_id
	password		balance		timestamp				cash_balance
			currency_name		account_id				stock_buy_price
			currency_symbol						num_shares

GUI Design

- Preliminary GUI layouts were generated using powerpoint to map out what the user should be able to see and how the user experience will flow
- Finalized GUI's are shown:





Object Oriented Design



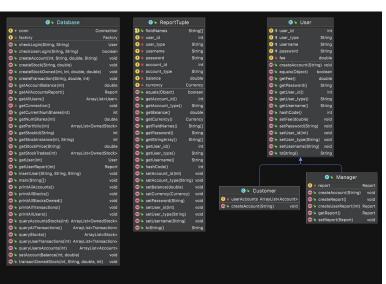
10 a SUPPORTED_STOCKS String[]

6 buyButton

6 a equityLabe

⊕ ≥ stocksString()

(f) a panel



Demo