StrategicBanking

Harbir Aujla

aujla@pnw.edu

Isaiah Rosinski

irosinsk@pnw.edu

We will design a bank management application, called StrategicBanking. Our database will hold data related to the operations of a bank, both customer accounts and internal systems. It will be able to determine the bank's overall health, and can adjust the parameters of accounts and loans as needed. We were inspired by the bank management simulator at probanking.com, an academic tool that demonstrates the criterions and constraints required to keep a bank functioning in a healthy, profitable state. This simulator allows the user to bring a bank to profitability by manipulating various decision sets, including quantities of loans issued and deposits accepted, amounts of assets from the bank's portfolio to buy and sell, and interest rates on the various accounts.

Our system will integrate a subset of these typical bank functions. The database will hold customer biographic data, customer deposit and loan information, loans between other banks and the Federal Reserve, bonds held and issued, and current interest rates. The system will be able to manipulate loan or deposit account interest rates, and will update payment requirements for those accounts accordingly. Our application will have two sections, a customer view and an employee view. The customer view will show the balance of a customer's accounts and loans, display the next payment due, and allow the customer to make deposits, withdrawals, and loan payments. The manager view would show data related to the bank's internal operations, such as current reserves and funds on loan from the Federal Reserve, and would permit giving loans and changing the active interest rates for floating-rate accounts. If we have enough time, we may implement the addition of penalties for late payments or overdrafts. We may also implement joint account management, for example for minors or dependents. Our system will not hold investment banking functions, such as the management of stocks or mutual funds, and will not manage customer investments.

We have five examples of queries our system will be able to handle. First: how much reserve does the bank have on hand? This query involves the deposits, customer and bank loans, and bonds held and issued. Second: if interest rates were to change by a given amount, how much equity would be gained or lost? This query would involve the interest rate table, as well as the floating-rate loans and deposit accounts. Third: when a customer leaves the bank, all their corresponding accounts must be removed. This query would involve the customer data table, as well as the various types of accounts. Fourth: How much money is outstanding in customer fees? Fifth: How many payments would a customer need to make in order to pay off a loan?

Our project will not be online; we intend to host the data and the frontend locally, as this makes the application simpler, and allows us to focus more on the database itself. Both of our group members bring valuable experience to the project. Harbir understands the inner workings of the banking system, and will guide the backend structure of the database and surrounding systems to interact according to real-world standards. Isaiah has some experience with application development, so he will set up the application frontend and database interactions. After learning more about database management in class, we will determine who is best able to set up the internal database system. In order to collaborate on the project, we will meet over Zoom weekly or as necessary, and we will share our application code via Github.