

Computer Science Programming Project Documentation 16/17

By Will Morris

Analysis

- Describe the problem**
 - A bank has a base of customers** all of which have single or multiple, savings or general accounts that they use to store and manage their money. The data for all of these accounts and customers are currently stored in paper form, this is done via a paper form for each customer which contains their personal details such as name, age, email, DOB etc. These forms are stored in individual "folders" and are identified by their fore/surname which are stored in cabinets in a massive file room in alphabetical order. For example a cabinet would represent surnames with initials A-C.

If a customer wants to open an account with the bank a form must be filled out stating the details of it, such as: the customers full name so that the new account is linked to an individual; the type of account(savings, investment ISA etc); date that it was opened etc. This is then stored ideally with the customers personal form so that it can always be identified. The customers personal form will also have a list or field for accounts so that it can be identified that he/she has x number of accounts and they are listed by the name that the customer gave that account when they opened it for example "Retirement Fund".

An accounts balance is stored on a server database which can only be accessed by the bank and various credit card machines where a user might use a card to by something in which case the card uses the name and account number to access the banks database and request funds. At this point a transaction is logged if the transaction is successful. However the customer can only review their transaction at the end of the month when a paper statement is sent out for each account that the customer possesses showing the transactions for that month. The customer can also go into the nearest branch in person to request the same paper copy to be printed off to show an up-to date balance and transaction log.
 - Problems with this system**
 - Firstly, the system for sending out a paper statement every month means that a lot of paper and ink is used in printing which for a bank with thousands of customers can amount to a large sum of money that could be negated by being replaced with an online system. This gets far worse when customers start to have multiple accounts.
 - All records of accounts and customers are stored in paper format which poses multiple problems. Say an account document is removed from the customers file to be changed, when they go an put the document back they have the account number and customers name to go by however there is no unique customer identification field which can be put on the account form to say that the account belongs to this person and this person only. Instead the employee has to index the account by surname only and then if there are multiple of that surname maybe they check each of those customer folders to see which customer holds an account with that number. This can all go wrong if the employee is feeling lazy or for some reason doesn't realize that there are multiple customers in that surname and they just put it in the first one they see. All of this leads to misplaced and eventually lost information if the issues aren't resolved quickly.
 - Related to this is the problem that if a person tries to find a customers file there is a chance that they could pick up the wrong one. This is because each customer does not have any single unique identifier field that only one person can hold. It is unlikely that the wrong record would be chosen if multiple fields are used to search and verify a person(eg: name, date of birth) however it is still possible to have multiple of the same record with all the same data that are physically different customers.
 - Data redundancy is also possible if a customer leaves the bank for whatever reason and then re-joins and the bank does not ensure that the old one is removed or the problem stated above occurs and the wrong file is removed. Both ways can end up with repetition of data which can cause issues when keeping both files up to date.
 - The bank only holds one copy of each document and has no accessible backup meaning that if there is a fire or natural disaster and all of their records are destroyed customers can lose track of where their money is/should be held and the bank could go out of business very quickly.
 - Customers do not have quick access to their transaction history or current balance having to go into branch to request this information which can be a problem for people with a very tight schedule who just want to be able to access their banking information quickly through their smart phone or laptop that they may have on them at all time. This means customers can also go overdrawn unintentionally which can give them issues with credit score if a customer does not know he or she is overdrawn because they did not have easy access to their banking information.
 - Justify why it can be solved by a computer** The banks current system is ideal to be modeled by a computer system for a number of reasons.
 - Firstly, half of the banks system is already stored on a database. This covers just the transactions which are linked to the various accounts by having an account number as a foreign key to the paper account form which the bank stores. However this link between the physical account database and the computerized transaction database does nothing. For example the transaction database does not have the ability to check the account "form" to see how much over draft that type of account is allowed meaning there is no way to safeguard a customers overdraft if they are unaware of their current balance in that account. A computer system can be used in the form of a database for the same reason that the transaction database can.
 - Identify stakeholders**
 - The customers of the bank** will be the main stakeholders of this project as they are the individuals who will gain most from the production of an online system to replace the paper one. the customers needs from this program is that they must be able to access it anywhere with an internet connection so that they can view their details at a time flexible to them. Also it must replace the monthly paper statements if the customer so chooses if they feel that it is no longer needed due to this new online system. This will allow customers to manage their accounts more in real time rather than having to wait for the next opportunity to go out of their way to visit a branch making for a much more convenient and efficient banking experience. It will also provide a way of making transactions between different accounts or even to other customers of the bank which will be a very quick process again due to the lack of having to go into the bank to withdraw cash or write a check and send it to the target person.
 - Employees of the banking company** will now have a much easier way to access and edit customer records with a new database that connects with the tables of the transaction database so that they both work together and are linked in such a way to avoid data redundancy and maintain data integrity. This will make the job of the bankers more efficient because it will provide a level of human error protection so that data is always kept up-to-date in all places and that no data can be repeated. The database will provide a way of uniquely identifying each customer through a primary key or a secondary key which ideally will be more human readable so that a customer can be identified uniquely by their email for example. Their email will also be tied to this database so the bank can send emails about urgent issues with their account or the ability to send out messages about new account types that they are offering. The online database will also have the ability to be easily copied and backed up off site which increases the security and safety of the information that they are holding which in turn makes their bank much more attractive to new customers as they know their money and personal data is safe.
 - Research and justification of approach**
 - While I am in the design and development of my banking application I will be taking inspiration from different examples of already developed/deployed banking systems and observing the features that stand out most to me. Namely I will be looking for the features that cater to the user through asking questions like:
 - Does it look easy to use?
 - Are the navigational tools well designed and accessible?
 - Is there too much information on screen? Too many options to navigate?
 - Do different buttons have tool tips to display information about what the button does?
 - Is the account information displayed in a manner in which it is easy to identify separate accounts/ transactions?
 - Is it responsive? Does it respond to changes in screen size?(this is not measureable here)
1. **Example 1** - This application is a tablet android based application however its design is in a style which I would like to implement in my final application. Some of its key features will be listed.
- Example 1**
- Clear navigation bar on the right with headers to group linked button actions together. In this example "Start a transaction" all the actions for setting up payments are grouped.
 - There is a search box button so that the transactions can be queried to find specific records. I would like to implement this with different criteria so that the user can search only for records in a certain price bracket for example.
 - Details that do not need to be displayed are hidden behind drop down menus("More Details") so that they can be accessed when they are needed. This makes for a very abstracted view and only shows the most important data to start with so the user can just glance at the display and already get a good picture of how their account is looking currently.
 - Text sizing highlights the account balance and the smaller text highlights the transactions. This gives a very natural feel to the information being displayed as the main balance is made up of the smaller transactions and this is represented well by the text sizing.
 - The listed section of accounts on the side shows all accounts that the customer currently holds. This is a feature which I will likely adopt so the user can access their different account seamlessly. In this example it allows the user to quickly switch between accounts through a scroll menu at the side.
 - Flat design. No ugly shadows or beveled buttons. This makes the application more user friendly and not overwhelming to look at.
1. **Example 2** – Login Screen Example. I like this login screen example because of it simplistic look and feel. It is very abstracted through how it has one simple logo and just the links you need to either access your account or in my example create an account. It doesn't show any irrelevant information and the focus is solely on the login of the application.
- Example 2**
1. **Example 3** - This example shows a basic `ch` program however it displays many of the basic functions that I want in my program to have. It uses a box to highlight related options/ information which in this example surrounds the transaction functions. It gives you a number of options which can be operated because the user has already selected their account shown by the account number and each of these buttons use that to manipulate and access various information about that account which will change a database in the background.
2. It also has a logout button which I would also like to add to allow users to exit and save changes made to their accounts before closing the application when they are finished. However this example is a little basic and the colors used are not inviting which makes the program ugly and old fashioned looking.
- Example 3**
1. **Example 4** - Another example of a tablet application which I believe is a good fit for the design of my banking application due to its attractive look and simple display of information and options.
- Example 4**
- Clear navigation bar with headers to group linked button actions together. In this example "Start a transaction" all the actions for setting up payments with other people are grouped.
 - There is a search box so that the transactions in this account can be queried to find specific records. I would like to implement this with different criteria so that the user can search for only records in a certain price bracket for example.
 - Details that do not need to be displayed are hidden behind drop down menus so that they can be accessed when they are needed. This makes for a very abstracted view and only shows the most important data to start with so the user can just glance at the display and already get a good picture of how their account is looking currently.
 - Text sizing highlights the account balance and the smaller text highlights the transactions. This gives a very natural feel to the information being displayed as the main balance is made up of the smaller transactions and this is represented well by the text sizing.
 - This allows the user to quickly switch between accounts though a scroll menu at the side.
- Flat design. No ugly shadows or beveled buttons.
- Explain the features of the solution**

- i. The program must be able to allow users to create an online account and connect to their respective accounts using the account number. They will create an account by entering the same data as is on their paper form in the bank and then choose a password. In future they will use this password along with their email to login with. Once they have created an account their currently active accounts will show up and their data will automatically be linked to their customer database.
- ii. Once a user has logged in the allow users to view a list of the accounts they have linked them. They should be able to select one of these accounts and view: balance, a transaction list which should include an option to refine the transactions to just in or outgoing transactions. They should also have multiple search options which allows a customer to search the transactions by date, ascending/descending alphabetical order, by transaction size and others as I think of them.
- iii. There should also be the option to select "make a transaction" which should prompt you to enter the account number you would like to send funds to, a drop down box to enter the account you would like to pull from and an amount box to specify the quantity you would like to transfer. This should talk to the transaction database to create a new record of this transaction and then edit the account database by way of a foreign key to link and edit the balance of the accounts.

- **Justify hardware and software requirements**
 - The target audience for this program is mainly the customer therefore the customer has to be able to run this with their

Design

Test strategy(last subtitle for design):

Unit testing is testing each program as it is written. The dev tests the program with data as the program is being made. Get screen shots of code as you are making it. Show testing as the program is being added to. In design you must create a test plan for each program or procedure.

Test No.	Test Data	Reason for test	Expected Result	Actual Result
1	Surnames = ("Graham", "Lee", "Appleyard", "Brigg")	To test Bubble sort	Surnames printed in correct order	-

White box testing - differs to unit testing by testing a whole program or module of code and putting test data into the program that tests all possible pathways through the program. Must be planned in design section and carried out in the development.

Post development testing:

Improvements:

- Could make more secure by using a temporary key so people can't just claim your account using a stolen account number
- Could have security questions
- Sql injection