Online Bank Management System

A3C Banking

Amber Davis, Andre Harding, Caitlin Benjamin, Aaron Horton

*Computer Science Department, Atlanta Metropolitan State College*

*1630 Metropolitan Parkway SW, Atlanta, GA 30310, United States*

AHORTON@atl.yearup.org

AHarding@atl.yearup.org

cbenjamin@atl.yearup.org

ADavis000@atl.yearup.org

***Abstract*** – **This research paper shows that online banking is the new age banking. Conveniency is what customers looks for. That’s why we developed an online banking system. As you read this paper it will break down some coding and show the usage of HTML, CSS, Java, and the applications that were used to run the codes. So, what you should get out of this paper is how the online banking management system was created.**

Keywords-Java, HTML, Online banking, CSS, Database, Code, GUI, Atom, MySQL

1. Introduction

Are you tired of waiting in long lines to get your own money out of a bank or just to get simple routine banking services? If the answer is yes, then A3C Banking has the perfect solution. My group has developed and online banking system so that customers will be able to access their bank information and handle their money wherever they’re located. Our online banking system will save customers time and gas. Statistic proves that customers favor these online systems “In 2019, there were more than 160 million people using online banking in America, a 20% increase compared to 2014. This includes both online-only banks and branch banks with digital banking channels. (Statista). The first online banking system was introduced in the United States back in 1994. The world’s technology is advancing every day and technology is making services more convenient to customers to do everyday routines. The purpose for this system is to make it easier for customers to handle their money at any time without physically going to a bank. A recent US internet banking survey revealed that only 20% of consumers would rather pay a visit to a physical bank location than do their business via digital channels. (ConsumerAffairs). Although this system will be popular it will not put banks out of business. There will still be customers that will rather get serviced by humans and aren’t as tech savvy. So, the two options should balance to service at A3C banking. The online banking system will provide the same service options that customers would receive at a physical bank such as customers will be able to create an online account, view saving and checking account details, find out the interest rates on a loan, banking statements, transaction history, transfer money etc. In this research paper I will explain how we were able to develop an online banking management system from the front end to the back end.

1. The Start up

Just to give a little background on our bank name “A3C” it comes from our first name initials. The 3 A’s stand for Amber, Andre, Aaron, and the “C” is for Caitlin. We then established roles for each member, now to the good part, how the online banking system was developed. Our color theme is black, white, and green. Green of course represents money and black represents sophistication. Also, we want it to be known that it’s black own so minorities can trust their business with us. We knew that our site needed to provide all the services as if they were in at bank. We were aiming to display balance, statement, saving /checking account etc. We were all already familiar with online banking because we use it. We begin to look at different banking companies such Georgia Own Credit Union, Bank of America, and Chase so we can get an idea. For the program language our group decided to use Java for the back-end coding. We used Java because we’re more familiar with it and java is more flexible / versatile, so you can run it on any machine or platform. And for some front-end work we decided to use HTML and CSS. HTML contribute to the structure on the application. And CSS was used to provide the style.

1. Back-end coding

Now that we figured out the foundation of the application it time to move on to the back-end. Mr. Harding oversaw the back-end coding. He started out seeking examples online for previous online banking projects to help guide him in the right direct. YouTube was the main resource. “Bank Management System | Java Project “YouTube, was the video that walked him through the steps. This video provided two new software applications “NetBeans IDE” and “MySQL”. Mr. Harding used NetBeans to edit the source code, build executables, and debug. We’re used to JGrasp,Vison Studio but they all work pretty much work the same. I will now go over some key details of Mr. Harding’s work. He created multiple classes to represent each page. For example, there’s a class labeled as “sign up” and it will display the sign-up GUI. To a take close look at the code for the login page you’ll see “*JLabel*” that indicates that a label was created for a text field.”*JTexField*” is for the user to enter information in the box. To create buttons for the users to click we used “*JButton*” and you see these example in (e.g Fig. 1) below

Fig.1

Text

Description automatically generated with medium confidence

You will also see “*SetFont*” and “*SetBounds*” alters the color of the button and the border. Action listener is connected to “*JButton*” because when the user clicks the button “*Action Listener”* activates the button to que the next action depending on what button that was clicked, or information was provided by the user. The circle options are created by using “*JRadioButtons*”. And to group the buttons he uses “*ButtonGroup*”. To show dialog box after the user enters information, he used “*JOption Pan*e” as well as “*If-else statement*” which checks if the information was true or false shown in (e.g Fig. 2.)

Fig. 2

Graphical user interface, text, application, email

Description automatically generated

And to create dropdown boxes “JComboBox” is included in the code.

There are different classes for each GUI. for example, a page for transactions, to sign up, login, and etc... (e.g Fig. 3) will show a GUI for the sign-in page.

Fig. 3

Graphical user interface, application

Description automatically generated

We wanted to have database connect to the program but unfortunately, we couldn’t connect it. Mr. Harding used MySQL to create the database. MySQL is a data base management system. The database would have stored the user’s information such as a password or pin. The database is essential to the project. Data plays a crucial role in any organization. He made different tables for each option for the user to choose from. In e.g (Fig. 4) shows that the varchar was used because Varchar mean variable character. Varchar(20) represent how many characters the user can enter.

Fig. 4

Graphical user interface, text, application

Description automatically generated

1. FRONT-END

We decided to have one person do front-end and one person do the back-end. Ms. Benjamin oversaw the front-end. For the front-end HTML and CSS was used. The group has heard of HTML and CSS, but we didn’t know how to use it. So of course, we had to learn them. W3schools came in handy along with YouTube. W3schools was able to teach Ms. Benjamin how to use HTML and CSS. So, she had to distinguish what they both do. As she learned how to use them w3schools makes it easier for beginner and provides template of layout you can use. The template would basically show you where info on the site such as the header, navigation bar, main content, side content and footer. She was able to get the colors using w3schools.com under CSS. As I mention before the color scheme is green (#3cb371, black (#000000) and white(#ffffff). For colors and design she referred to CSS. And for wording structure she used HTML. Another text editing platform that was used is Atom. Atom is simpler to use for beginners. She used HTML first then she added CSS. Ms. Benjamin also utilized the box model to help organize the layout. Now the details about the code in (e.g Fig 5) it shows <!DOCTYPE html> it means to declare. It informs the web browser about the type of HTML is used. <style> means it’s making it’s own section

Fig. 5

A screenshot of a computer

Description automatically generated with medium confidence

1. Future works

If we were to continue this project, we would do more research or reach out for help to connect the database to the rest of the program. An online banking system needs to have a database. We would add more services for the user to have access to. For example, we didn’t add the loan and credit card options in the application. We would’ve also connected the front-end and the back-end together instead of it being separate. It’s probably better to do the coding together. The back-end code produced a GUI and the front -end looks different. Also, we would continue to program with java but also look to other program languages because other languages might be easier to use. Our group knew the basics about Java, but we didn’t know anything about HTML and CSS. They were easy to catch on. We would continue to use w3schools to learn more. The application would be more decorated, have pictures and provide details about the bank. Also, it would look better and not boring with animations or transitions on the app. We would continue to use ATOM. Maybe if the group had communicated better, we probably would’ve got further in project. One important thing is to add comment in the program so we won’t have to write down notes and forget where it goes and also the group can see the comments. On ATOM they have where you’re able to share your code and whoever you provide the link to can edit the code. That would’ve been helpful. I know we could’ve did more but time wasn’t on our side. NetBeans is pretty easy to use for beginner so I say we would continue to use it .Just to sum it up we would use the same application and software. I don’t believe what we were using was wrong .I believe if we had more time we could’ve had everything running smoothly .Also have the application appealing to the eyes and easy for user to do what they need to do.

1. Conclusion

This project has taught our group a lot. It taught us that communication is key while working in a group. Because if everybody has different roles you have connect with each other so the project can connect. Now in days resources are all on the internet. Just from YouTube and w3school.com we learn a lot. It’s better to do research and do a draft. Technology is always advancing, and companies must stay update to stay relevant and stay in business. And technology is making everything easier for customers. So, who knows what will be developed in the future for banking.

1. References

1. “38+ Insightful Mobile Banking & Online Banking Statistics,” Fortunly, Apr. 30, 2020. https://fortunly.com/statistics/online-mobile-banking-statistics/.

2. 1/4 - Bank Management System | Java Project | Without Drag and Drop | NetBeans & MySQL,” www.youtube.com. https://www.youtube.com/watch?v=MoXEHE21EBQ (accessed Nov. 30, 2021).

3. W3Schools, “W3Schools Online Web Tutorials,” W3schools.com, 2019. https://www.w3schools.com/.

4. “Online Calculators & Tools - RapidTables.com,” www.rapidtables.com. <https://www.rapidtables.com/>.

5. “Introduction to HTML & CSS Course | What Are You Going to Learn in This HTML Course | mmtuts - YouTube,” www.youtube.com. https://www.youtube.com/watch?v=TKYsuU86-DQ&list=PL0eyrZgxdwhwNC5ppZo\_dYGVjerQY3xYU.