Christopher Markham, Arturo Bramsco, Kaleo C Chase

ATM Project

Table Of Contents

* Problem Statement & System Requirements Page 3
* Functional Requirements Specification Page 4
* System Sequence Diagram Page 5
* Activity Diagram Page 6
* User Interface Specification Page 8
* Project Plan Page 12
* References Page 13

Problem Statement & System Requirements

An ATM system addresses issues with automated machines in many locations. In most locations where ATMs are commonly used, access to banking services can be a problem in suburban and urban areas. In many cases, ATMs only function with buttons, not on the screen, which makes the machine dependent on external connections. Having more sources on the ATM screen can make usage more efficient.

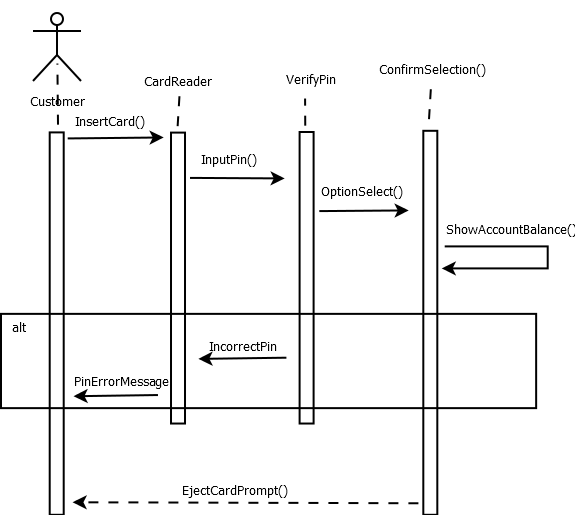
**System Requirements**

* Make users money & data available 24/7. This means the system needs to have nearly 100% uptime.
* Must be an intuitive user experience that is easy to understand for people who are not technically inclined.
* Must be able to connect to the main bank database
* Must be able to count and dispense cash reliably
* Must have 24/7 video surveillance

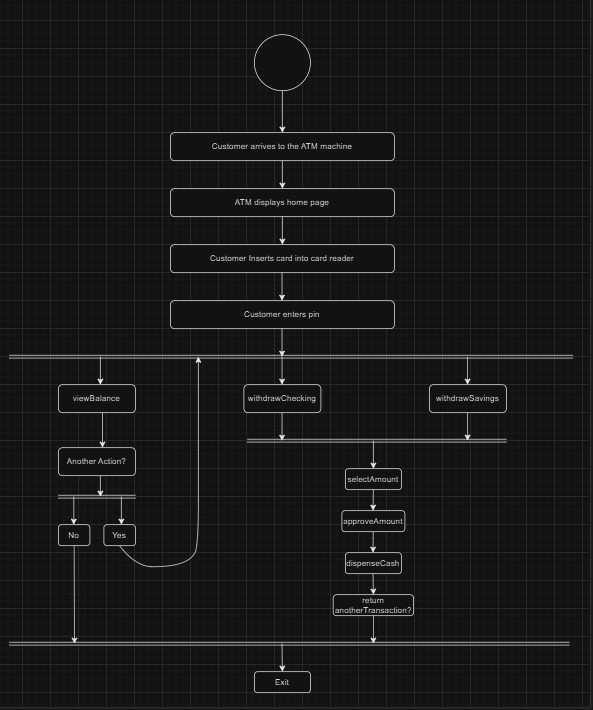
Functional Requirements Specification

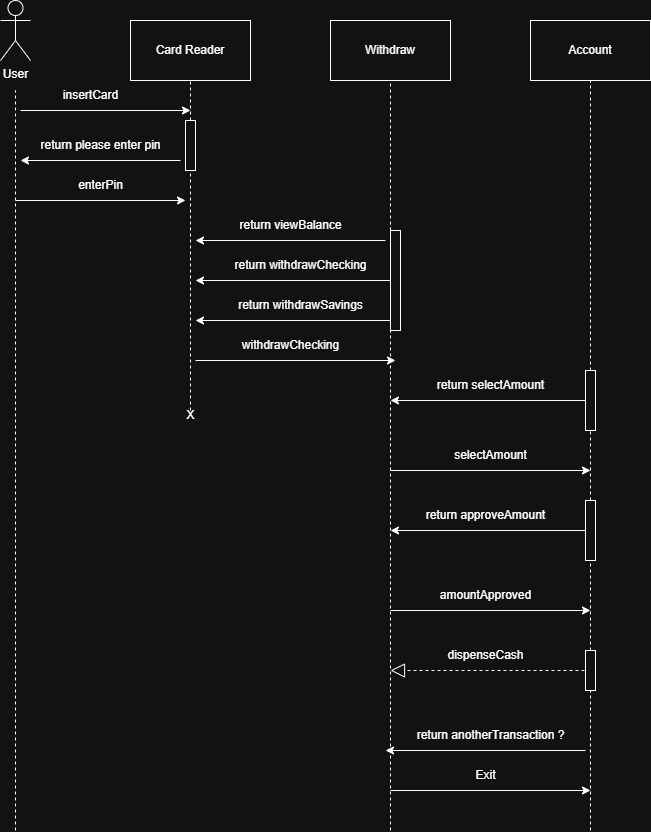
The objective of the ATM system is to allow for various factors to be implemented to become more effective for usage, making the system a lot more beneficial for everyone with minimal problems. For banks or contractors, it would minimize customer service to assist individuals who use ATMs and, in addition, give more options to users to do business at the ATM. This will cut time spent in the bank or other places to deposit cash or check, transfer money to another party, donate, make a payment, check balance, etc. The system in the ATM would be connected to the same server used at your personal bank, which would update any changes made and show up on your personal device. According to the American Bankers Association (ABA), 71% prefer mobile apps, and only 9% prefer in-person. Although you can use these services on your phone, there are still some services you cannot use. The system will be a bank login with your card, which would open up the same details on any ATM using this system, allowing for more and better services at an ATM and effectively cutting time spent on doing services in person.

System Sequence Diagram



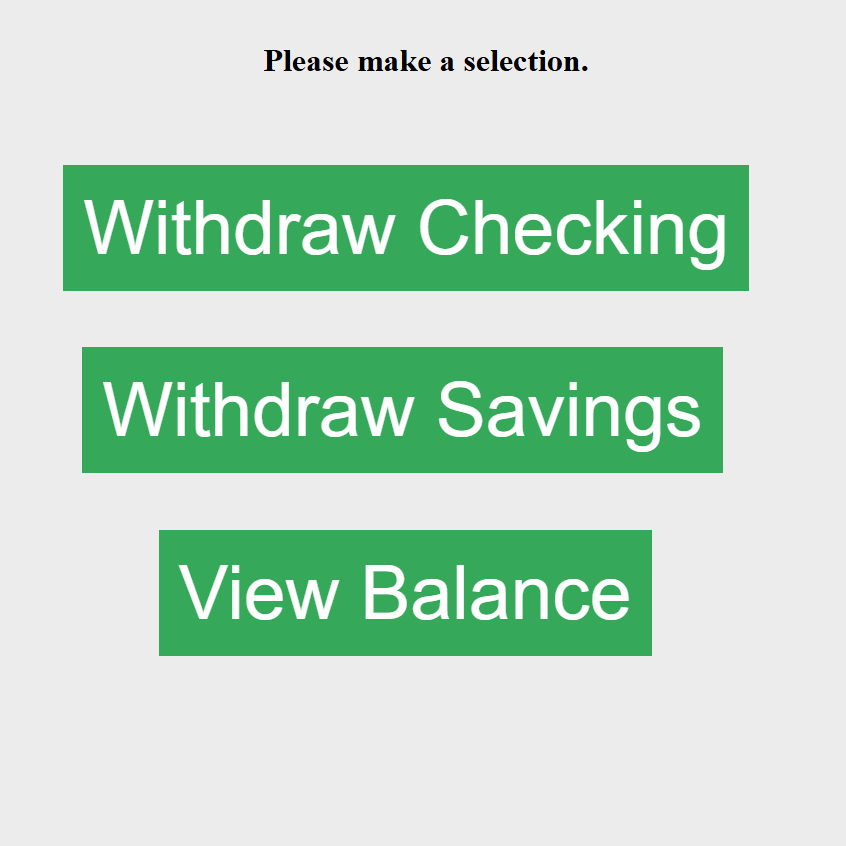
Activity Diagram



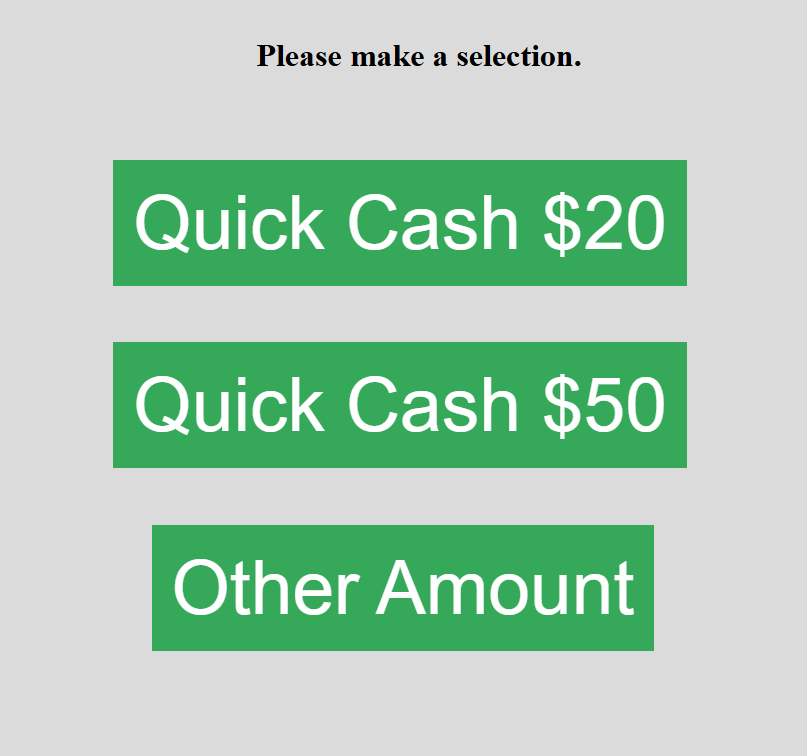


User Interface Specification

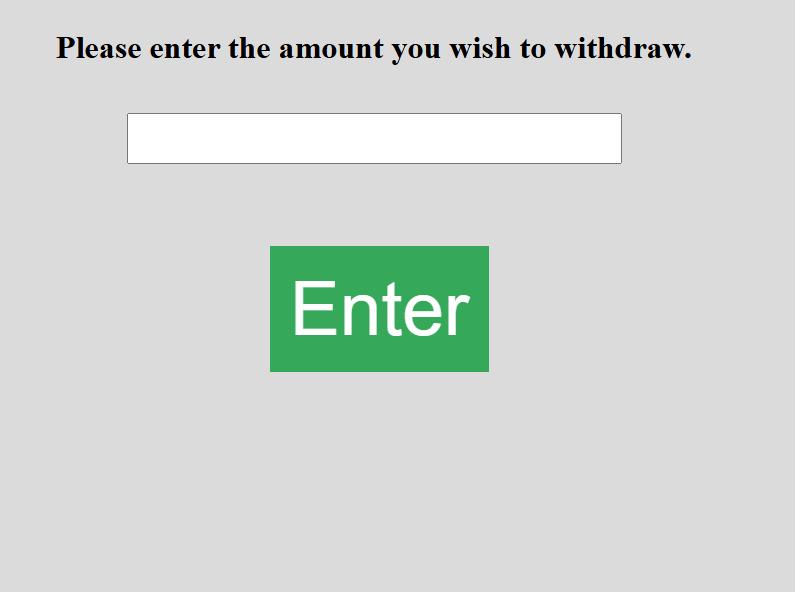
Our ATM application's icons will be big and bold, as it is a touch-screen application. Our prerelease software will be used with a mouse and keyboard, but the final product will be touch-screen capable. The screen will look similar to the one below.



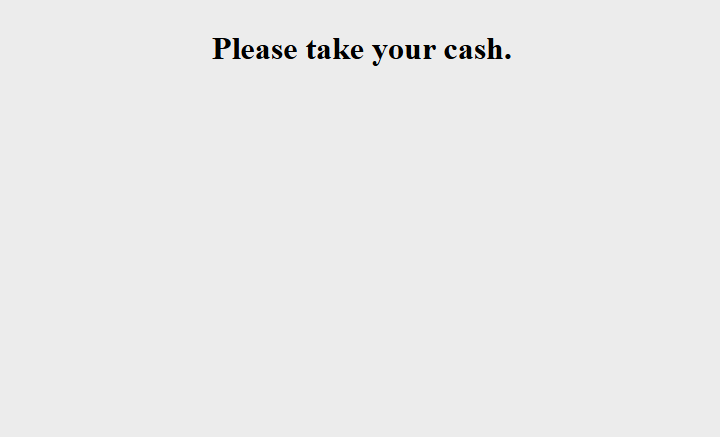
The screenshot on the previous page shows the main screen of our ATM system. After entering the system, all navigation starts here. With this prerelease demo, only the Withdraw Checking is currently functional. Each of the green rectangles is a button. This system is intended for users of all age groups; thus, it is straightforward with very few distractions. Any user-interactable item will be bold and green.



Quick cash is as simple as clicking the button. The specified amount is deducted from checking and presented to the user. The Other Amount is for any other amount, not 50 or 20. You can still deduct 50 or 20 using others. Those are options as they are some of the most popular amounts drawn.



If the user clicks another amount, they are presented with the above screen. Here, they can enter the amount they want, and it will deduct that amount from the checking account.



The very last screen is a simple “Please take your cash.” In a future release, there will be a button to return to the home page to start another transaction, etc.

Project Plan

**W1 – 2:** Determine and build the framework to establish the system's structure. Frameworks will give an idea of how the system will look when it's all finished. Connect the front end to the back end to start the development process. Build the layout of the pages (does not include buttons, pictures, or anything extra).

**W3 – 5:** Build the login system and set it up for banks, managers, or contractors. Also, implement basic page features, such as the introductory screen, transaction options, balance information, amount selection, instructions, confirmation, error messages, security alerts, etc.

**W6:** Test all the added features and record a demo for the mid-term. This will allow us to find bugs or areas that can affect the system's future usage.

**W7 – 9:** Improve current features based on customer feedback, if any, and continue to work on perfecting the features on the system. The system will show the user's activity for verification and any verification needed to protect the information or identity of the user. Focus more on the security aspect of the system, as it is essential to protect sensitive information. Improve the banking aspect to ensure smooth usage of the system. Write test cases for the implemented features or continue building features to develop the system further. Record a demo for the final presentation to give insights into the system.

References

* Christopher Markham
* Arturo Bramsco
* Kaleo C Chase