Deliverable 1

Austin Hood, Joshua Durham, Christina Aragon

Milestone and Deliverable 1:

1. **Describe the software development process model (e.g., agile or waterfall) you will be using on your project. State any variants possible (scrum, etc.) with adequate details**

The software development process we are using on our project is a basic agile methodology. We have set sprints and consistent communication with our primary stakeholder(sponsor) to make sure that they will receive the product that they are asking for. We are also making constant changes to the plans and staying flexible with changing requirements.

2. **Consult with your project mentor (sponsor) and provide the specific deadlines and deliverables that are planned.**

EBO	
January	
Development	Secure messaging utilizing the file conventions of the hypothetical web wallet inbound message
	Secure storage of the data in cloud environment
	Refactoring into a secure message in the file format of the receiving system (Merchant Terminal)

	Secure Messaging in the refactored file format
February	
Development	Refinement of NFC Module Secure Messaging to process in real world environment (Field identification)
	Any augmentation necessary to support barcode systems, UPC, ect.
	System Adminstrator View
March	
Development	ID Verification
	End Point Security
Compliance Implementation	508 Compliance (Applications, JAWS)
	Privacy
April	
Testing	Participant Testing
Final Refinement of Prototype	Any final adjustments necessary
Demonstration of Prototype	April 12th

Agile Methodology

The start and end dates of each sprint are defined in the graph above divided into each month. There is a specific goal for each of our month long sprints, the first being to secure data transfer from a web wallet to a merchant terminal. The second is to secure some real world application of data transfer and be able to communicate with merchant terminals in a consistent way. The third sprint is to add additional security that would be needed for an application that deals with

financial data and potentially sensitive information. The fourth and last sprint's goal is to do any testing and additional adjustments in this month.

Waterfall or its variants:

Specify the start and end dates of each phase, i.e., requirements, design, implementation, testing, deployment, and maintenance.

1. The main tech stack we will be using for our project is a react front-end through the Next.JS framework. This is a full-stack framework so all APIs will be written in java/typescript for continuity of programming language and an easier setup/deployment experience. There will also be a number of APIs and open source libraries that we will be using to achieve the desired functionality. For example, the google wallet dev API for wallet integration on android devices or an open-source web NFC library.
2. The source code repository and version control will be handled by git/github for a streamlined and consistent experience. This is a technology all member of the group are familiar with and we believe is the best option.
3. The method we are using for group task assignment and tracking is a JIRA Kanban board for easy assignment and real time updates. We validate each other's work when a pull request occurs, there must be another member that must verify and validate the other's PR.
4. We are going to manage our project and communicate with each other through weekly meetings with our sponsor as well as additional team meetings as needed throughout each week. Some topics require more meeting with other group members so generally we contact each other through text messaging and if more clarification or a focus group is needed, we meet in person.

Economic Benefit Opportunity: Milestone 2

Sponsor: Thomas Ritter

Team: Joshua Durham, Christina Aragon, Austin Hood

Agile Methodology

I. Requirements

Cloud Environment Requirements

- The cloud environment must securely store the UID number.
- The wallet must pull and decode the UID number from the cloud environment.
- The cloud must pull the UID number back from the NFC device and encode it.
- The NFC device must pull and decode the UID number from the cloud environment
- The cloud environment must securely store the total transaction amount.
- The wallet must pull and decode the total transaction amount from the cloud environment.
- The cloud must pull the total transaction amount back from the NFC device and encode it.
- The NFC device must pull and decode the total transaction amount from the cloud environment
- The cloud environment will store the users private log-in information (Passwords, Username, SSN [possibly])

User/Wallet Requirements

- The user register for app
- The user will sign in for app
- The user will create a username and password
- The user will add new wallet
- The user will fill out any information to obtain a new wallet
- The user will view existing wallet(s)
- Unkown entity will add funds to the user's wallet
- If the user’s EBT application is denied, the system will prompt the user to say that their application has been declined.
- If the user’s EBT application is approved, funds will then be added to the wallet.

- Users can view their information including UID, Transaction payment info, other personal information.
- The user will activate the card payment setting when making a transaction.

NFC Requirements

- The user will tap the reader with a mobile device
- Once the user taps the NFC device, the reader will pull and store the UID in the reader.
- The NFC device will check the user's wallet for available funds.
- The NFC device will complete the transaction if specified funds are available.
- The NFC device will decline and provide an error code.
- The NFC device will remove funds from the wallet once the transaction has been completed
- After the transaction the NFC device will remove the UID from the queue.

II. Software Architecture/Design

(give **consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors. Your report should include how you considered them)**

Summary

The electronic benefit application will consist of **two main components**: **a mobile application for beneficiaries** and **a merchant terminal system**. The mobile application will communicate with the merchant terminal system using NFC technology. The merchant terminal system will be responsible for processing the electronic benefits and authorizing transactions.

The overall architecture will follow a client-server model, with the mobile application serving as the client. The mobile application will be built **Java on Android Studio** and will only support Android devices. The application will use an **API to connect to a Postgre Microsoft Azure Gov Cloud.**

Design Considerations:

Public Health, Safety, and Welfare:

- The application should adhere to industry standards for secure NFC communication to prevent unauthorized access and ensure the privacy of user data.
- The application should be designed with accessibility in mind to accommodate users with disabilities.
- The merchant terminal system should be designed to prevent fraud and abuse by implementing fraud detection algorithms and limiting transaction amounts.
- Global, Cultural, Social, and Economic Factors:
- The application should support multiple languages and currencies to accommodate users from different regions and cultures.
- The user interface should be intuitive and easy to use, regardless of the user's technical expertise or background.
- The application should be designed to minimize data usage and battery consumption to make it accessible to users with limited internet connectivity or older devices.
- The application should be cost-effective for both beneficiaries and merchants and should not place undue financial burden on any party.
- Environmental Factors:
- The application should be designed to minimize its environmental impact by reducing paper waste associated with traditional benefit programs.
- The use of NFC technology will reduce the need for physical cards or vouchers, further reducing waste.
- The application should be designed to minimize energy consumption on mobile devices and merchant terminals to reduce the overall carbon footprint.

In conclusion, the electronic benefit application using NFC technology for merchant terminals should be designed with consideration for various factors, including public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors. Adhering to industry standards, accommodating users from different regions and cultures, minimizing environmental impact, and being cost-effective for all parties involved are key considerations for the successful design and implementation of such an application.

III. Code

Mainframe - https://github.com/jdurham38/HypEBT

User Interface - https://github.com/csaragon1941/UI-HypEBT

IV. Test cases and test reports

Test Cases:

I. NFC Communication Test

Verify that the mobile application can successfully communicate with the merchant terminal system using NFC technology.

Test different scenarios such as moving the mobile device closer and farther away from the terminal and verifying that the transaction is authorized or denied accordingly.

II. User Interface Test

Verify that the user interface is intuitive and easy to use.

Test different scenarios such as selecting different benefits, entering invalid data, and verifying that the appropriate error messages are displayed.

III. Fraud Detection Test

Verify that the merchant terminal system can detect and prevent fraudulent transactions.

Test different scenarios such as entering incorrect benefit amounts, attempting to use expired benefits, and verifying that the transaction is denied.

IV. Accessibility Test

Verify that the application is accessible to users with disabilities.

Test different scenarios such as using a screen reader, navigating using keyboard-only, and verifying that the user interface can be customized for users with color blindness or low vision.

Currently do not have test reports.

V. Backlog

User Story 1: NFC Communication

As a user, I want to be able to use NFC technology to make transactions with the merchant terminal, so that I can use my benefits easily and securely.

Tasks:

Research and select appropriate NFC technology to use

Implement NFC communication in the mobile application

Implement NFC communication in the merchant terminal system

Test NFC communication between the mobile application and the merchant terminal system

Update documentation to include NFC technology and how to use it

User Story 2: User Interface Design

As a user, I want the user interface to be easy to use and intuitive, so that I can use the application without difficulty.

Tasks:

Research and design user interface to meet accessibility guidelines

Develop user interface mockups

Implement user interface design in the mobile application and the merchant terminal system

Test user interface design and make changes as necessary

Update documentation to include user interface design and how to use it

VI. Any major project decisions (e.g., de-scoping and change in plans).

- Bringing in different advisors to aid in the project development.

VII. Each team member's contributions

Josh – Building out read and write capabilities on NFC tags to gain a better understanding of NFC and how they work to integrate with a specified merchant terminal.

Christina – Currently building out the UI and creating a more detailed interface for the app.

Austin – Building out server side/database capabilities.

VIII. Any ethical and professional responsibilities and informed judgments, which considered the impact of engineering solutions in global, economic, environmental, and societal contexts.

- Privacy and Security: The application will contain sensitive information about users' benefits, and it is important to ensure that this information is kept private and secure. This involves implementing appropriate encryption and access controls, as well as adhering to data protection laws and regulations.
- Accessibility: The application should be accessible to all users, including those with disabilities. This involves designing the user interface to be easy to use and implementing accessibility features such as text-to-speech and high-contrast modes.
- Fairness and Non-discrimination: The application should be designed to be fair and non-discriminatory. This involves ensuring that all users have equal access to the benefits, and that the application does not discriminate based on factors such as race, gender, or age.
- Environmental Impact: The development of the application should consider its impact on the environment. This involves using sustainable development practices and minimizing waste and energy consumption.
- Economic Impact: The development of the application should consider its impact on the economy. This involves ensuring that the application is cost-effective and does not place an undue burden on users or merchants.

IX. Project feedback and performance review from your mentors for items 1-7. You must get approval from your project mentor as to what you can document and share in this report.

Approved

Connection through NFC Technology: A Senior Software Project

Joshua Durham, Christina Aragon, Austin Hood

Final Deliverable: Milestone 3

Florida Gulf Coast University

Sprint #1 - March 6 - March 24

Assigned Tasks for Sprint

Joshua	Josh is working on the development of a solution that allows the reading and writing of Mifare NDEF data, the same technology used in credit cards.
Austin	Austin is writing a backend API as well as creating the Postgresql database hosted in the Azure Gov Cloud.
Christina	Christina has taken the lead in designing a new user interface and wireframe that align perfectly with our project's current scope, ensuring a seamless and intuitive user experience.

User Stories & Requirements

As a user, I want to be able to read and write Mifare NDEF data (credit card) with ease, so that I can quickly access my payment information without any hassle.

As a user, I want the user interface to be updated to reflect the current scope, so that I can use the application with a modern, user-friendly design.

As a user, I want to be able to securely authenticate myself and connect to the database, so that I can safely and efficiently store and manage my payment information.

As a user, I want the application to support ISO1443 and ISO8583 standards for reading and writing Mifare NDEF data so that I can use the application with a wide range of payment cards.

As a user, I want the application to securely authenticate me and establish a reliable database connection, so that I can safely store and manage my payment information.

As a user, I want the application to provide me with a seamless and hassle-free authentication experience, so that I can quickly and easily access my payment information.

As a user, I want the application to securely store and encrypt my payment information, so that I can use the application with confidence and peace of mind.

As a user, I want the user interface to be intuitive and easy to navigate, so that I can quickly find the information I need and complete my transactions efficiently.

As a user, I want the application to have a modern and visually appealing design, so that I can use the application with pleasure and pride.

As a user, I want the application to provide clear and concise instructions and feedback, so that I can easily understand and complete my payment transactions.

As a user, I want the application to be accessible and easy to use for people with disabilities so that I can recommend the application to a wider range of users.

As a user, I want the application to be responsive and adaptable to different devices and screen sizes, so that I can use the application on my phone, tablet, or computer with equal ease.

Software Architecture and Design. Ethical Consideration.

Introduction: **consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors. Your report should include how you considered them.**

When creating the software architecture and design, we took into account many ethical factors. For **Public Health, Safety, and Welfare**, we ensured that the software could run on various operating systems, allowing anyone with an NFC device to access it. We also followed industry best practices and standards through research, training, and guidance. We reviewed

security protocols, privacy policies, and data protection measures to ensure the safety of our customers' card data, complying with legal and regulatory requirements. To accomplish this, we implemented rigorous testing and verification procedures to detect and prevent potential vulnerabilities.

We were also very mindful of **global, cultural, social, environmental, and economic** factors. Taking into account the diverse needs and preferences of users, we designed the app to ensure accessibility for people with different abilities and cultural backgrounds. We also considered the environmental impact of our app and found that it has a positive effect. By digitizing credit cards, we can reduce plastic waste, making it more sustainable and profitable. We believe that by prioritizing public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors, we can build a software system that is not only functional and efficient but also ethical and sustainable.

Code

GitHub Links

Josh (NFC Capabilities): https://github.com/jdurham38/EBS

Austin (Azure Database): Private repository

Christina (UI/UX Desgin): https://github.com/csaragon1941/UI-HypEBT

Backlog

Develop NFC reader functionality for Android platforms

Design and implement user interface for the payment app

Integrate payment gateway with the app for seamless transactions

Implement encryption and security protocols to protect user data

Add support for multiple payment methods (credit/debit cards, mobile wallets, etc.)

Develop a database to store payment information securely

Test and optimize the app for performance and user experience

Create a user guide and help documentation for the app

Implement internationalization features for the app to support different languages and currencies

Develop a loyalty program for frequent users of the app

Major Project Decisions

There were no major project decisions during the duration of the project.

Each Team Member's Contribution

Joshua	Worked on payment and transaction information as well as NFC writing capabilities.
Christina	Worked on project documentation, as well as creating a UI/UX design for the product, including an in-depth wireframe.
Austin	Worked on the structure of the backend and authentication integration within the Azure

Gov Cloud

Any ethical and professional responsibilities and informed judgments, which considered the impact of engineering solutions in global, economic, environmental, and societal contexts.

The team has prioritized public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors in the design and implementation of the software. This includes ensuring accessibility for users with diverse needs and preferences, as well as considering the environmental impact of the product by reducing plastic waste through digitization. The team also implemented rigorous testing and verification procedures to detect and prevent potential vulnerabilities, thereby protecting users' privacy and adhering to legal and regulatory requirements.

Project feedback and performance review from your mentors for items 1-7. You must get approval from your project mentor as to what you can document and share in this report.

Sprint #2 - March 27 - April 14

Assigned Tasks for Sprint

Joshua	The read and write functionality of Mifare NDEF data has been completed. Additionally, the main application that will read and write Mifare data to the terminal is almost finished.
Austin	Hosting of the API in the Azure gov cloud and integration of API to client using HTTP requests.
Christina	Designing the animated version of the new user interface and wireframe.

User Stories & Requirements

Same as before, just making more progress.

Test Cases & Test Reports

Test Case: Verify Credit Card Data Reading

Test Objective: To verify if the app can read credit card data from a MIFARE compatible card.

Preconditions:

The app is installed on a device with an NFC sensor.

The credit card being tested is a MIFARE compatible card.

The credit card is within close proximity to the device's NFC sensor.

Test Steps:

Launch the app.

Hold the credit card near the device's NFC sensor.

Wait for the app to read the credit card data.

Expected Result:

The app accurately reads the credit card data.

The credit card data is displayed on the screen.

The credit card data displayed on the screen matches the information on the physical credit card.

Test Report:

Test Case Name: Verify Credit Card Data Reading

Test Case ID: NFC-001

Tester: Josh Durham

Date: April 13,2023

Test Results: Succeeded

Comments: The app accurately read the credit card data and displayed it on the screen without any issues. The information displayed on the screen matched the information on the physical credit card. The test was conducted on a Samsung Galaxy s20 with app version 1.0. Overall, the test was successful and the app passed the test case.