



---

# BurstIQ and RemediChain Final Report

Field Session Summer 2021  
Jun 11, 2021

---

## **Client POC:**

BurstIQ: Tyson Henry and Gianna Ricotta  
RemediChain: Philip Baker and Jeff Davis

## **Student Team Blockbusters:**

Katherine Aubert  
Jaidynn Fohr  
Simon Goldberg  
John Konchar  
David Park

---

## INTRODUCTION

After an individual loses their battle with cancer, their families and friends are frequently left with the stress of making arrangements and dealing with loss and grief. Therefore, something as seemingly useless as leftover chemotherapy medications are often discarded, despite the immense value of these medications, sometimes costing between \$10,000 and \$200,000 for the full treatment. It is RemediChain's mission to give a new purpose to these pills, and help treat cancer patients that would otherwise not be able to afford chemotherapy. BurstIQ, a platform for secure and connected data solutions, has decided to partner with RemediChain in order to streamline their donation and recipient process.

The purpose of our project is to facilitate and simplify the logistical problems of connecting chemotherapy medication donors and recipients through our client, RemediChain. By using BurstChain, a private and permissioned blockchain technology from BurstIQ, we envision a web application that will take in donor information and chemotherapy medication, add it to RemediChain's inventory of medications, and display those medications so that candidate recipients can make reservation requests.

## REQUIREMENTS

### Functional Requirements

1. A sign-in page that allows users to access their profile
2. A form to donate medication that creates an asset in inventory on the BurstChain
3. A form to request medication that removes an asset from the inventory on the BurstChain
4. A profile home page that displays a table containing:
  - a. The status of the user's donations
  - b. The total value of donations
  - c. The value of each donation that has been matched

### Non Functional Requirements

1. Pill data traceability
2. Simple and rewarding user experience
3. Allow a user to be both a donor and a recipient of medication
4. Inventory visibility
5. Information protection

---

## SYSTEM ARCHITECTURE

### Website Design

Our website is designed in such a way that it is both aesthetically pleasing and easy to navigate for the end user. It allows for the most efficient and effective way for users to both donate and request various medications. Upon accessing the website which hosts our application, the user must first log in to access the home page. From there, various information about RemediChain is displayed along with the options to either donate or receive a prescription. After choosing either the former or the latter, the user must fill in an HTML form, which links to the back end code and BurstIQ's custom private and decentralized blockchain, BurstChain.

In the case that a user is donating a new prescription, the user fills out the donation form, which directs the backend to create a new asset tied to their own credentials on the BurstChain. The asset is then transferred to the RemediChain inventory wallet with its status marked as "pending". Upon being shipped to RemediChain, their in-house pharmacists inspect the drug to ensure that it is unopened and up to their strict quality standards. If the drug is unopened and passes their quality assurance tests, the drug's status is changed from "Pending" to "Approved", and it will now become visible in the application's inventory.

When a user submits a request to receive an existing prescription from the RemediChain inventory, they must fill out a recipient form and then the back end code stores the information securely on the BurstChain. Once a pharmacist approves the recipient's prescription request, the asset is transferred on the BurstChain from the inventory to the recipient. The prescription is then mailed to the recipient and RemediChain's in-house pharmacists then change the prescription of the asset from "Approved" to "Delivered", such that it is no longer visible in the inventory.

### Data Structures and Blockchain

#### Blockchain Application

One of the primary considerations in this project was data privacy and the ability to track medication ownership over time, so a private and permissioned blockchain implementation is the structure that best fits our client's needs. This blockchain, proprietary to BurstIQ and known as BurstChain, allows separate chains to be created and permits users with public and private key pairs to view specific information, create assets, and transfer assets. In this application, the assets that were created were medications that a user would like to donate. Then, the asset would be available for pharmacists to view and approve for use in RemediChain's inventory, and from there they could be transferred to a recipient user.

## Wallets and Asset Flow

In order to better conceptually track the location of medications in the donation process, we considered the medications as individual assets held by separate user wallets. Medication assets will be transferred between two separate wallet types: user wallets, which can represent either donors or recipients, and a single inventory wallet owned by RemediChain. Once a user fills out the donation form on the website, a medication asset gets created in their wallet. Then, when the form is submitted to RemediChain, the asset is transferred to the inventory wallet, awaiting approval from a pharmacist to exist in the viewable inventory. This asset remains hidden in the inventory wallet and is not visible to users until a pharmacist approves the asset as a valid donation. Once another user submits a prescription request, a pharmacist can assign medication to be matched to that recipient, at which point it's ownership will be transferred into the recipient's wallet. Figure 1 depicts this flow of assets traveling through different wallets from the initial donor to the final recipient.

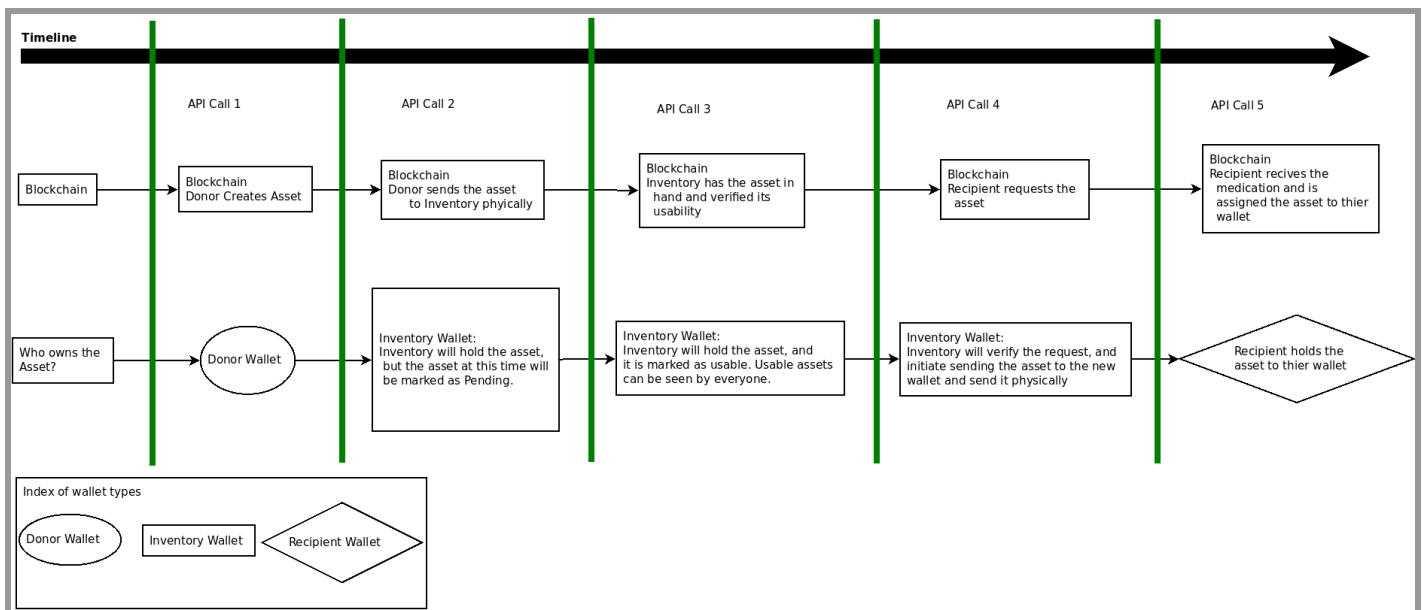


Figure 1: Blockchain flow diagram

---

# TECHNICAL DESIGN

## Website State Diagram

Figure 2 illustrates the flow of a user’s experience through a website state diagram. Once the user navigates to the web application, they are greeted with a home page. Next, they may click the login button on the top menu bar to access a screen which will prompt them for their credentials. When a valid email and password are entered, the application will return them to a second version of the home page that contains the additional “My Account” button on its menu bar. If this is clicked, the user will be redirected to a personalized account page which is set up to display their donation history and tax write-off information as well as allowing them to access links to the donation form page, the inventory page, and the request form page. The user is also able to log out from any point within the application, at which time they will be returned to the original home page.

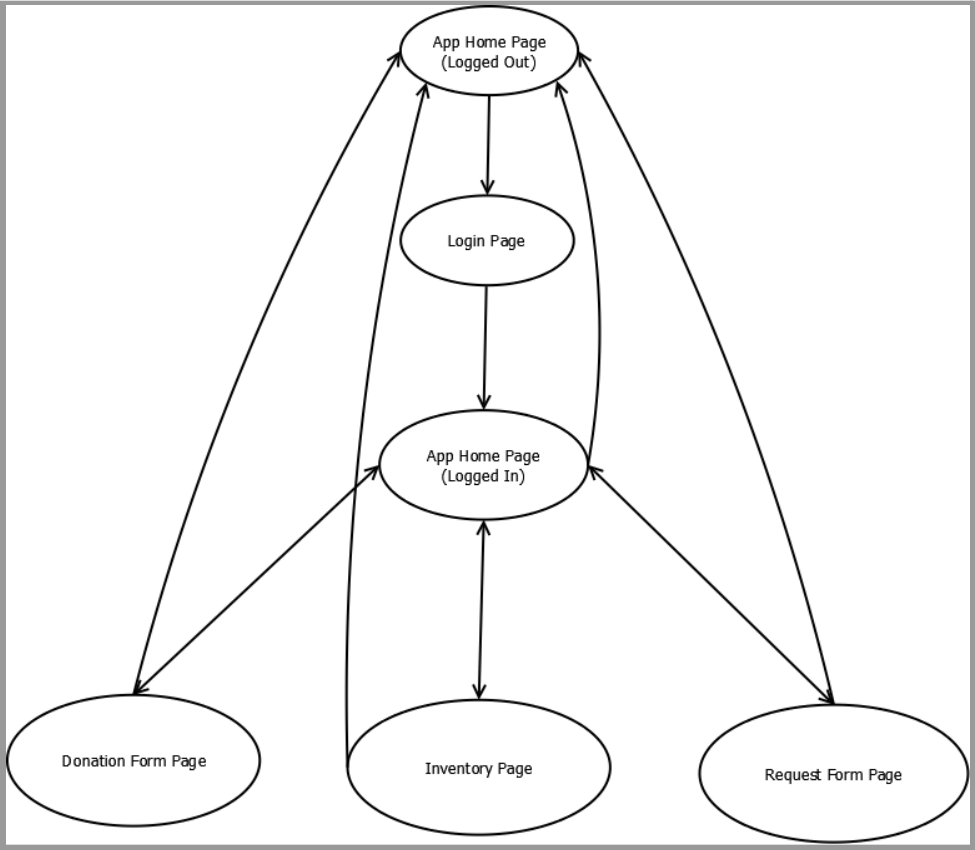


Figure 2: Website State Diagram

## Token Authentication / Login

Our web application includes a login service so that we can ensure that only authorized BurstChain users can access the website's functionality. When a user inputs their username and password, an API GET request is sent through a BurstChain endpoint, returning a JWT user login token that will remain valid for four hours before requiring the user to log in again. All future API calls will utilize this token to authenticate the user's account. The GET call returns an API status, and on a successful '200' return code, the website changes the webpage to show the user's home page. From here, they can access the rest of the website and view their own inventory. Figures 3-4 display the results of this process.

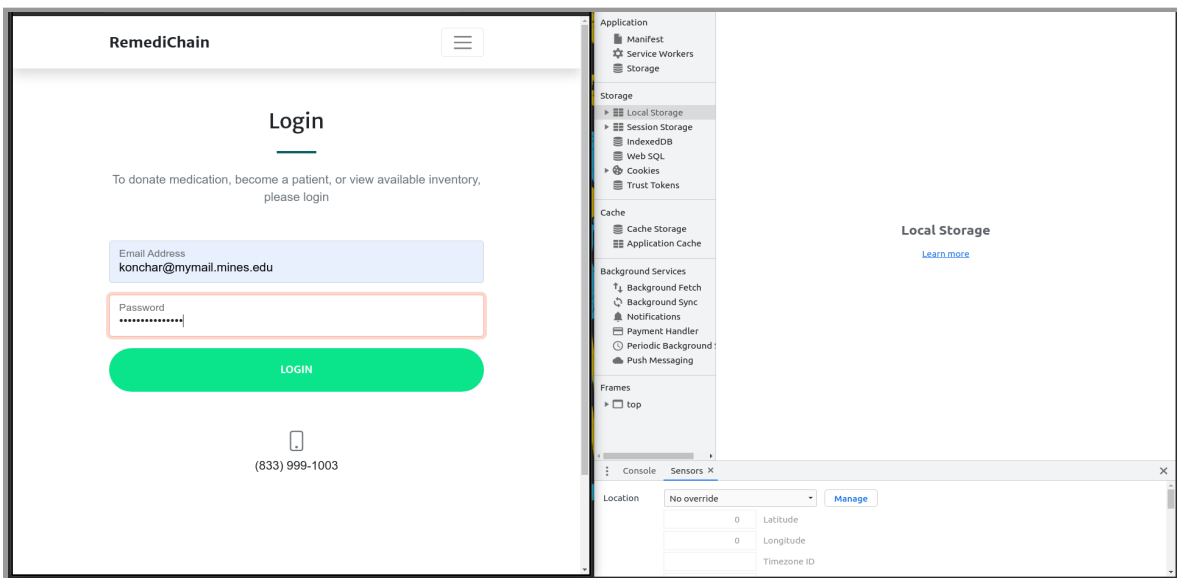


Figure 3: A loginpage with credential fields

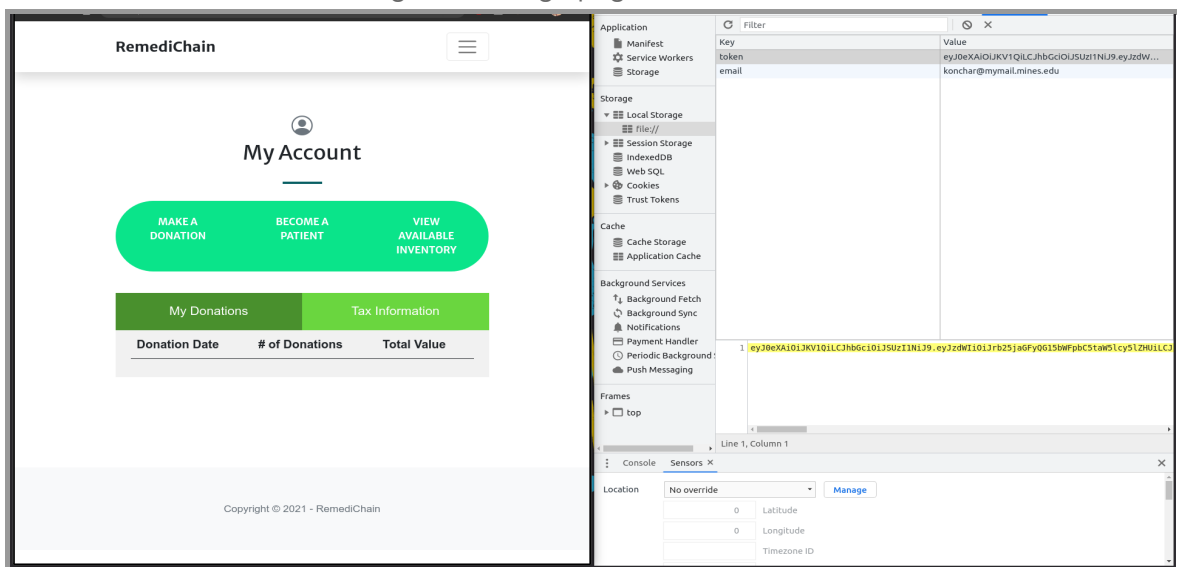


Figure 4: Result after a valid login with the user's JWT token in the computer's local storage

---

## QUALITY ASSURANCE

### User Interface

The web design was created in parallel to the creation of the back end functionality. Because of this, we planned the website around the features and methods being created in the middle and back end. As such, the elected method of ensuring website quality was manual testing due to the predicted changes in the method development. Checklists of manual testing requirements are as follows:

- Page links and buttons require manual checks for functionality.
- Data displayed on pages should be checked against known values to verify accuracy.
- API integration should be evident using the inspect tool on the browser.

### Middleware and API Blockchain Functionality

Utilizing typescript and javascript, our web application interacts with the BurstChain block chain using a series of API calls. API calls were best verified using Postman since authentication testing and scripted testing could be achieved using the application. Examples of the Postman scripts can be seen in the Appendix of this document.

GET, POST, and QUERY interactions were standardized in Postman to verify completeness of functions in the areas of:

- User Authentication
- Asset Creation
- Asset Management
- Accessibility Transfer Protocols

---

## RESULTS

### General Results

Our primary client, BurstIQ, informed us that this was a “Proof of Concept” project and that our separate work on the front-end and middleware code fulfilled all of their specifications, even without connecting the two. They also stated that our user interface was much more sophisticated than they were expecting from developers with our level of experience. Since we have been able to successfully integrate our code to produce a full-stack web application, our client seems highly satisfied with our team results.

### Implemented Functions

- Comprehensive User Interface
- Login Functionality
- Donation and Request Form integrated into the website
- Account Page that leads to the forms and displays donated assets

### Functions Not Implemented

- Pharmacist User Class
- Recipient request and Transfer Function
- Medication Smart Consent Contract
- Tax Information
- User Account Creation

### User Interface Functions Not Implemented

- Account Creation Page
- Forgot Password Page
- Pending Inventory Display Page
- Pending Prescriptions Display Page

### Smart Contract Functions Not Implemented

- Donation Fulfillment Consent Contract
- Tax Information on Donated Assets Form
- Asset Shipping Between Users and Inventory
- Oracle Solution to Confirm RX Prescriptions



---

## **FUTURE WORK**

### **Recipient Request and Transfer**

One set of functions that is necessary for a complete product is recipient request and transfer. For people in need, functionality for recipients to reserve specific medications and functionality for pharmacists to review and send the medications is an important requirement for this web application. Due to time constraints, we were unable to complete this functionality.

### **Pharmacist User Class**

Donated medications need to be physically verified as being recyclable by a qualified pharmacist. As a donor creates a medication asset, it should be assigned a status of pending, so that no recipient users could see or request the medication asset until a pharmacist receives and clears the medication, marking the recyclability of the asset.

We have prepared a 'status' quality on medication assets and created a special user role for pharmacists. We also would have liked to create additional buttons and alter the site appearance for users with pharmacist privileges. Methods with API calls to update assets would need to be created and affixed to buttons to implement this idea.

### **Oracle Solution to Confirm RX Prescriptions**

The vast majority of current medication data is stored off-chain in older databases. It is assumed that these databases are incredibly secure and are following HIPAA compliance. Though this data is currently stored off-chain, it may be implemented into Smart Contracts stored on-chain if an oracle solution were to be utilized. An oracle solution is a technology which connects off-chain data to a blockchain, such that the data acts as the input for one or more predefined Smart Contracts. After much discussion, we have determined that this would be a highly effective solution in the scenario that a user - which will always be a donor in this case - can enter a prescription number (Rx#) to be cross-referenced with the pre-existing off-chain pharmaceutical database.

This comparison between user input and pre-existing data will allow for the full elimination of typos caused by human error related to prescription numbers in the database. The oracle effectively proves the validity of each prescription donated by this cross-referencing process - without the need for any third party to manually verify the prescription in the current database.

Utilizing this solution will save much time and effort, as pharmacists are already extremely busy with their jobs. Additionally, it will ensure that every prescription entering the RemediChain

---

inventory (whether on the blockchain or physically) will in fact be a valid prescription. All RemediChain will then have to do upon receiving the prescription is check that the medication is unopened and can pass all of their quality assurance checks. There is a great abundance of time and effort which would be saved if every single individual medication donated to RemediChain were to be autonomously verified by the process described above.

## Medication Smart Consent Contract

Our product plan also envisioned a donation fulfillment feature where a donor of medications would be notified that the medications they donated were used by a recipient. For example, a donor could log on to their webpage and see all their donated medications on a table, as well as a notification of medication fulfillment for each individual donation. This would serve to give positive feedback to the user for having gone out of their way to donate medications.

The way that we see this being best implemented is through the use of a BurstChain smart contract to grant the donor indefinite permission to view their donated medications for an indefinite time. This contract would only show the original statistics of the donation, as well as a boolean value for whether the asset was owned by the RemediChain inventory wallet or not. This would reveal the status of the medication while maintaining privacy and protecting all personal data concerning the medication's recipient.

## Tax Information

When users donate their unused medication, they may often be eligible for a tax write-off. This tax information will ideally be available for them to reference on their personal account page. At present, we have set a tab on the account page that exists for this purpose, but it is not yet linked to any data or calculations. When BurstIQ and RemediChain are ready to implement that functionality, they will have a place prepared to display this data.

The tax information for medication assets would require code to write a consent contract in order to populate the tab with the proper information. After a donor creates a medication asset and it leaves their wallet to go to the inventory wallet or a recipient wallet, the donor can no longer view the asset. A consent contract allows the donor to be able to see specified attributes of the asset, such as their price and quantity, without compromising the information on the owner of the asset. This would allow a proper tax form to be tabulated, and it would prevent donor users from seeing any extraneous attributes, such as those pertaining to the recipient's personal information.

## User Account Creation

---

Currently, we are using BurstIQ's servers to run our website. When this project started we were added as administrators on their servers. Because of this, we are able to login to the website, but we are not able to create additional accounts. There is no existing page for account creation at present, but it will likely be straightforward for BurstIQ's developers to implement.

---

## LESSONS LEARNED

### Team Coding Experience

Sun Tzu once said, “If you know the enemy and know yourself, you need not fear the result of a hundred battles.” When starting a new project using new languages, a new technology, and a new team, it is very important to spend time early on in the project understanding the capabilities of the team. Planning how to best use the talents of the team, and also planning on measures to improve on the team's deficiencies, is vital to project success.

For example, none of the team had experience using HTML/CSS and Javascript, but the project required these languages. While the team was able to study the languages in order to complete the requirements of the project, the team's various programming skill levels limited overall production. One proposed solution was to spend a day or two studying the languages and blockchain together, ensuring a common understanding of the project's challenges. Without having done this, certain people in the team were too specialized in their fields, and collaboration became difficult.

### Client / Team Expectations

Our project brought the unique challenge of working for a partnership between two companies, and this required our team to meet with both and collect information on what they each desired for the end result of our project. When we initially met with our primary client, BurstIQ, they were not initially aware of what exactly was within the scope of our project requirements, and it was not until a second meeting with representatives from both BurstIQ and RemediChain that we were able to begin to grasp what we would need to accomplish. We were also presented with a much larger amount of desired user stories than we were able to tackle in the short time frame designated for us. Consequently, we had to learn to rely on our primary client's expectations without being sidetracked with additional ideas that would not be feasible to implement.

We also had to manage expectations within our own team. Since we were constantly discussing how best to structure our code and better conceptualize what we needed to accomplish, there were a large number of ideas flying around. At times, we had to work together to ensure that we were all on the same page regarding the priorities for this project. It is unfortunately very easy to become distracted by conversations or functions that are not necessarily furthering the goals of the desired product, and there was a major point when we set aside time to reset and return to more realistic expectations.

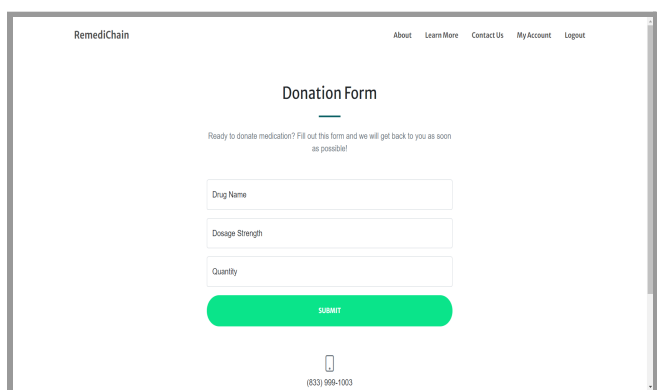
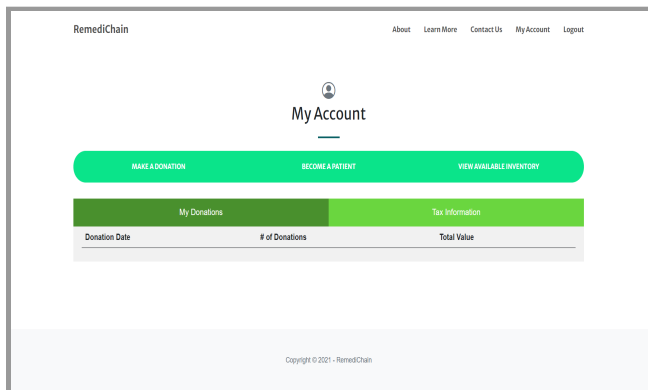
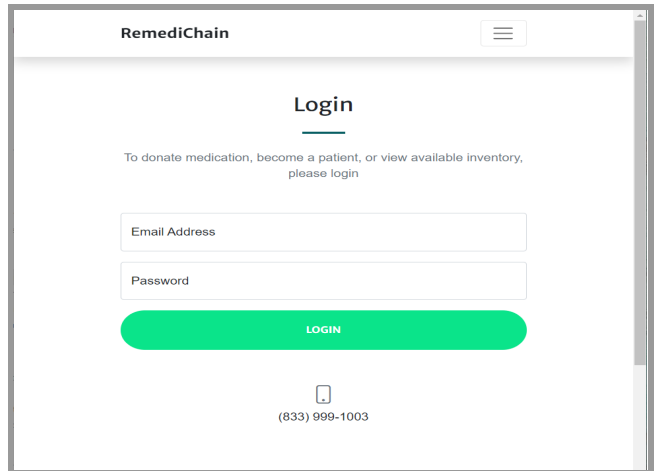
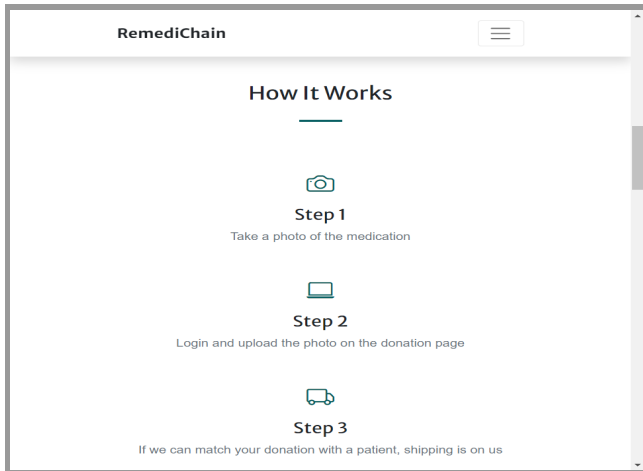
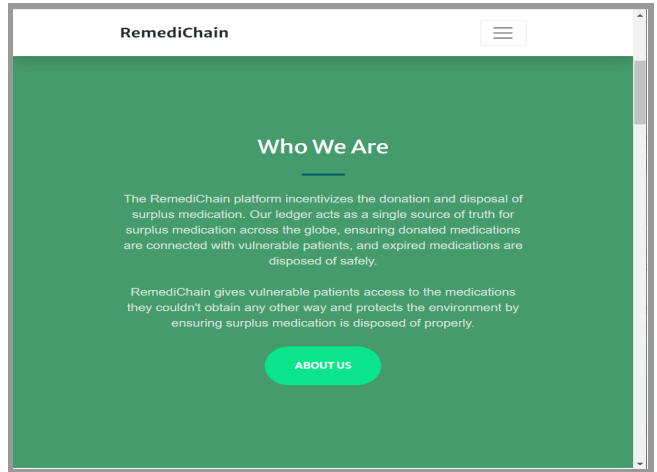
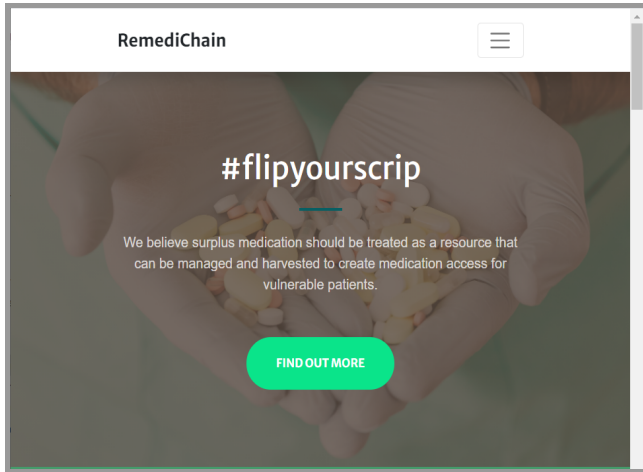
---

## APPENDICES

### Appendix A: User Instructions

To start our application, open the GitHub repository and click on the “home.html” file within the BurstIQ-RemediChain-Web-Application folder. This will open up the homepage where users can learn more about what RemediChain is doing. From there they can click on the login button at the top right corner of the page. This will open a login page where users can sign in to the application using their BurstIQ login credentials. Upon signing in, the user will be directed to the home page. From here, the user can use the top menu bar to access their profile page - which contains links to pages for the donation form, medication request form, and the available inventory - or they can access any of those three pages directly as well.

## Appendix B: Website Photos





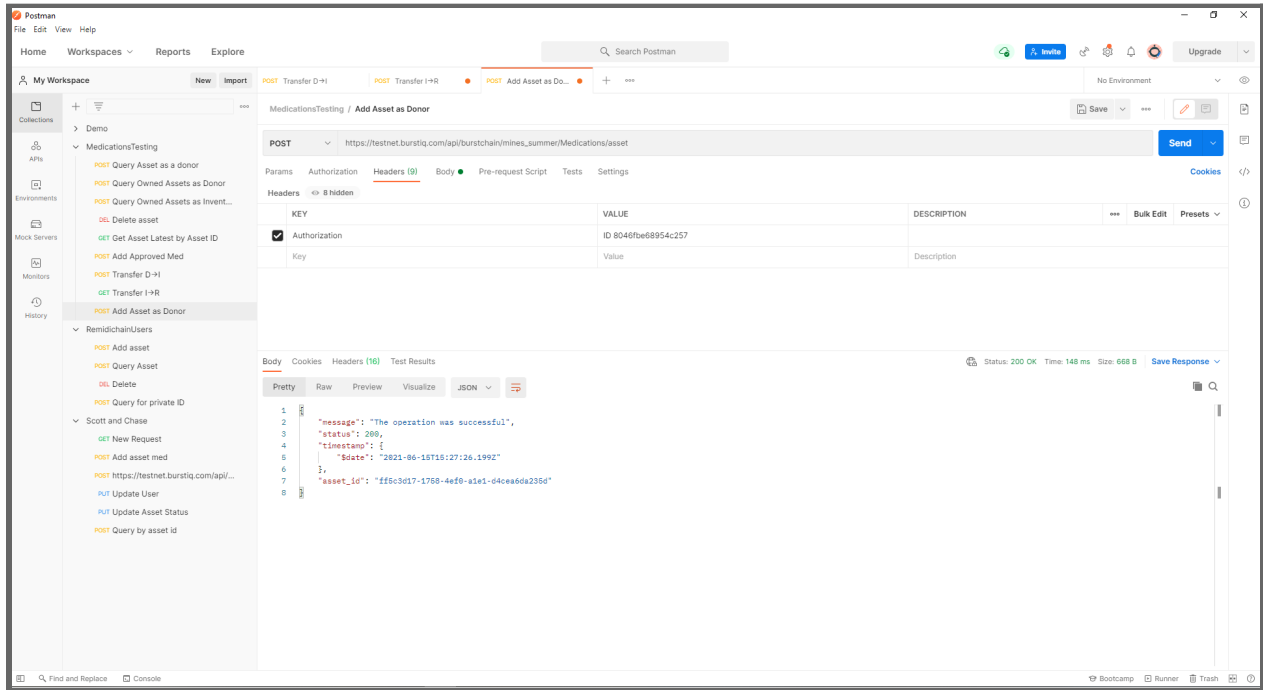


Figure 6: Create an Asset

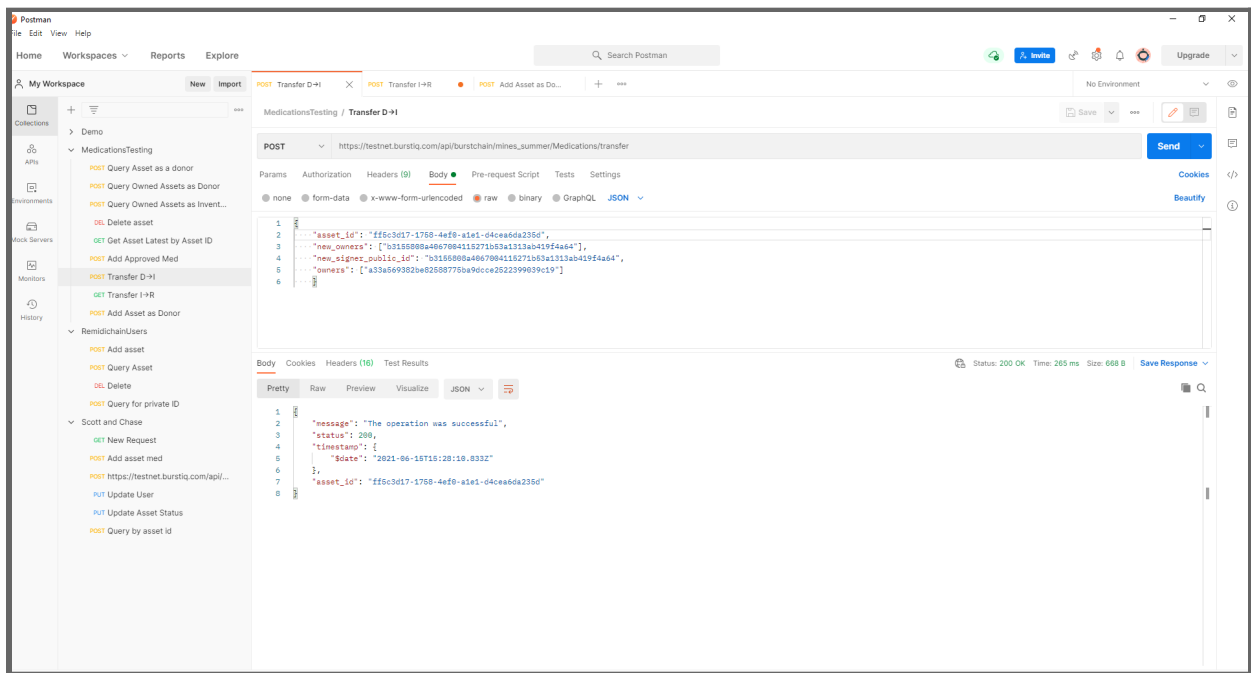


Figure 7: Transfer an Asset from Donor to Inventory



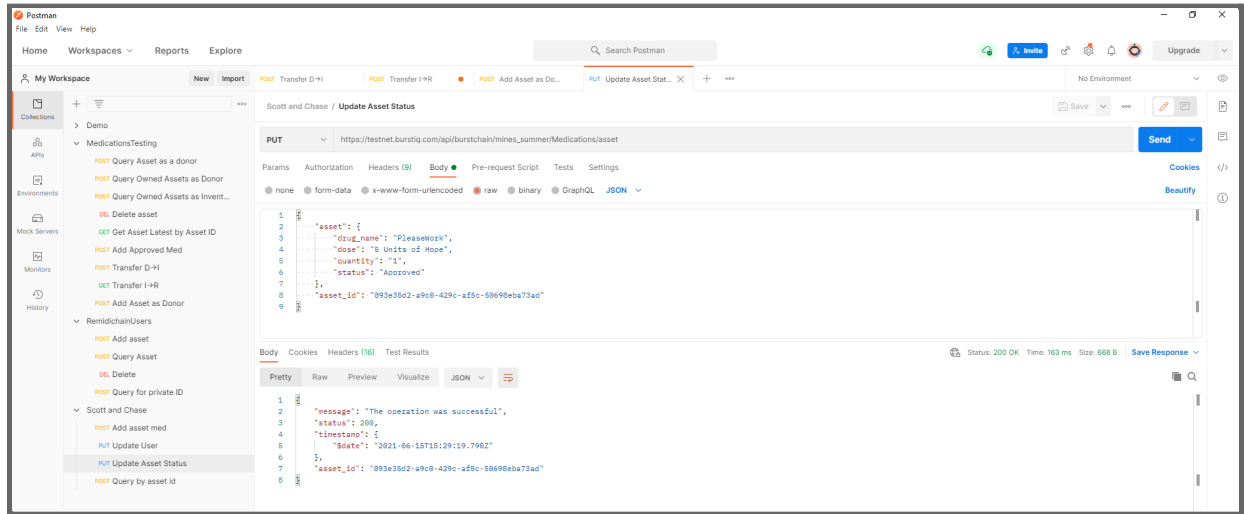


Figure 8: Update the Status Attribute of an Asset from 'Pending' to 'Approved'

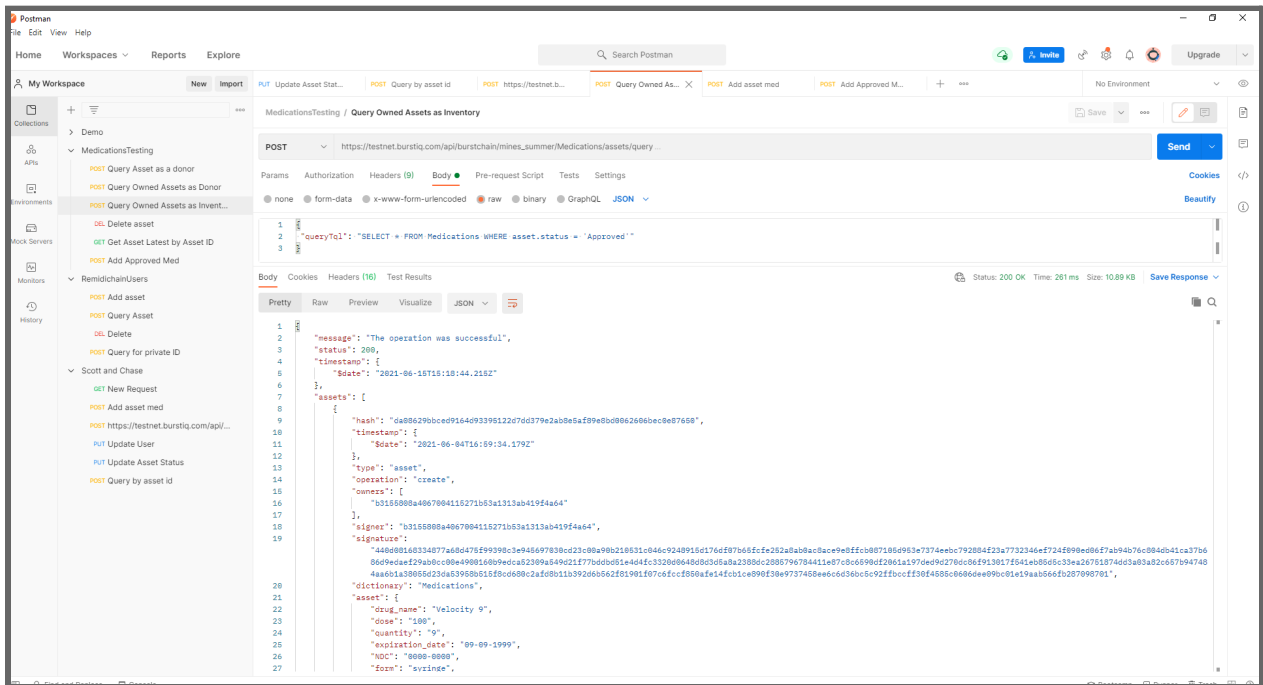


Figure 9: Query the Inventory to View all Approved Medication Assets

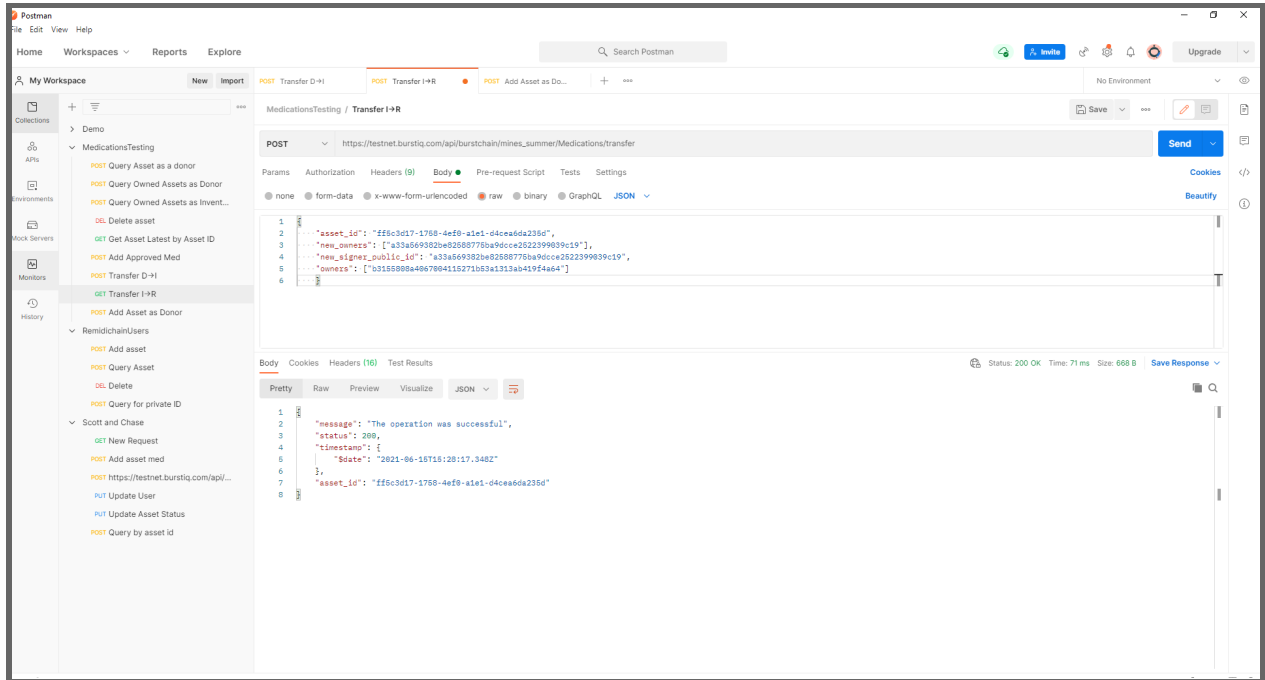


Figure 10: Transfer an Asset from Inventory to Recipient