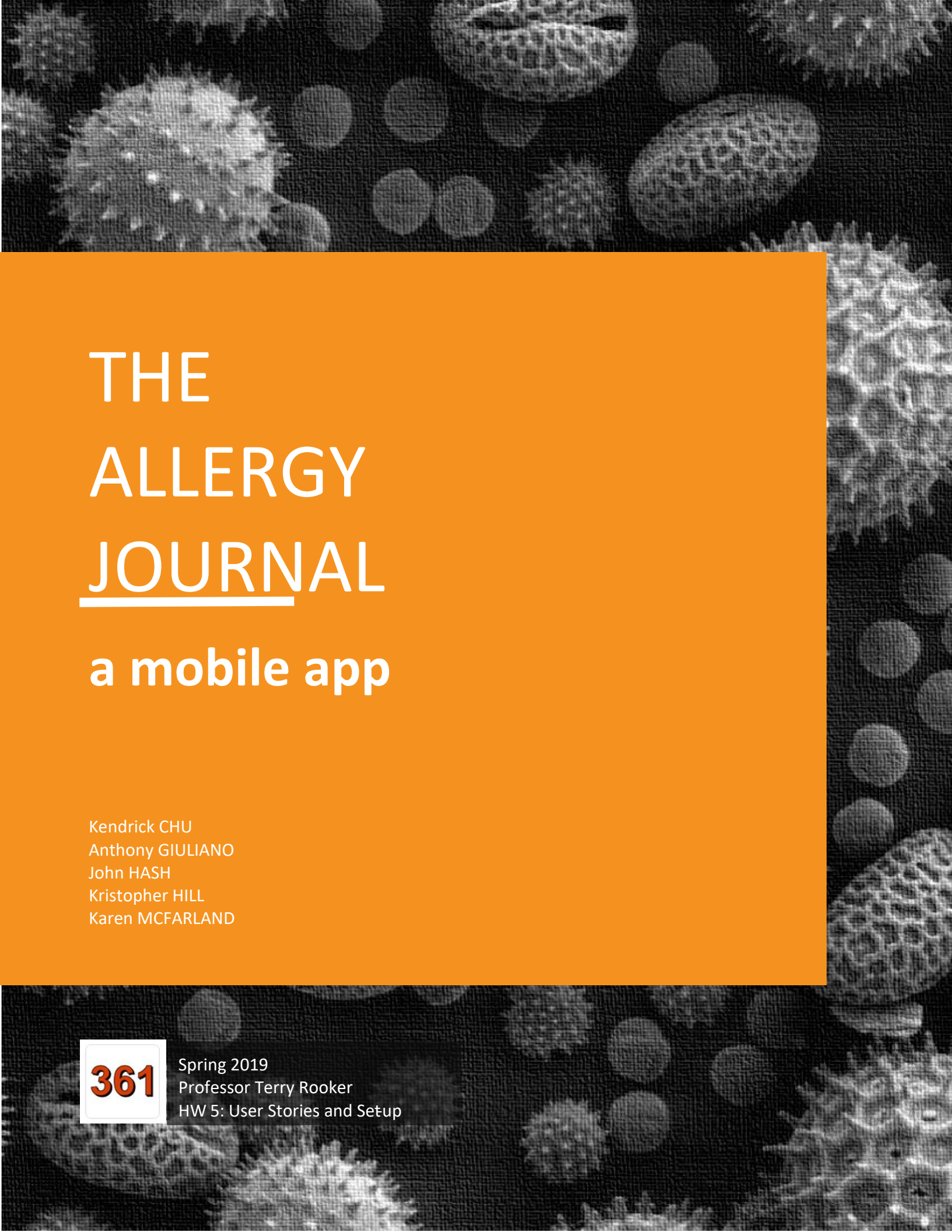


THE ALLERGY JOURNAL

a mobile app

Kendrick CHU
Anthony GIULIANO
John HASH
Kristopher HILL
Karen MCFARLAND

361

Spring 2019
Professor Terry Rooker
HW 5: User Stories and Setup

User Stories: Descriptions, Deadlines and Corresponding Tasks

Sprint 1 (during Week 9):

- 0.1 Environment Setup - Database:
  This task must be finished before tasks 1 – 6 can be finalized.
  [4.5 hrs] {Kris, Karen} Due: Tuesday, May 28, 2019
  - Create and set up a database to store: [2 hrs]
    - product data
    - product use instances
    - reactions instances
    - user info
  - Populate database [2 hrs]
  - Provide credentials to access this database to the team [0.5 hr]

- 0.2 Environment Setup - Webpages:
  This task must be finished before tasks 1 – 6 can be finalized.
  [4 hrs] {entire team} Due: Wednesday, May 29, 2019
  - High-level design of website. Website will have the following pages: [4 hrs]
    - Splash screen / Options
    - Login (stub)
    - Add a product
    - Document product use
    - Document reaction
    - View recent products used

- 1. User Story: As a person with allergies, I want to be able to add a product by manually entering product information.
  [5.5 hrs][Epic A] {Kris, Anthony} Due: Wednesday, May 29, 2019
  - Create webpage with a form for entering product information [2 hrs]
  - Include button/link to page/pop-up/dynamic form for admin request to add product [1 hr]
  - Create select menu for user to pick product in db from what they entered [1 hr]
  - Create confirmation page/pop-up/button for user to click when they know they have the right product [1 hr]
  - Write SQL query to insert product data into corresponding table(s) [0.5 hr]

- 2. User Story: As a person with allergies, I want to capture information each time I use a product from my personal inventory about which product I used, when I used it and where I used it, so that I can collect data on my product usage.
  [5 hrs][Epic B] {John, Anthony} Due: Wednesday, May 29, 2019
  - Create a panel on user homepage OR unique webpage with a form that includes select menu of all products they already have logged [1 hr]
  - Panel should also include body map or select menu of body parts for user to select from [3 hrs]
  - Write SQL query to insert product usage date into corresponding table(s) [1 hr]

- 3. User Story: As a person with allergies I want to record an allergic reaction: The allergy tracker app users can record an instance of an allergic reaction to a topical agent, like soap, lotion, makeup, etc.
  [5 hrs][Epic D] {John, Kendrick} Due: Wednesday, May 29, 2019
  - Create a webpage with a form for entering allergic reaction information [1 hr]
  - Page should also include body map or select menu of body parts for user to select from [3 hrs]
  - Write SQL query to insert reaction data into corresponding table(s) [1 hr]

- 4. SPK: User Story: As a person with allergies, I want to scan the barcode of a product and add it to my inventory of products being used.
  [10 hrs][Epic A] {John} Due: Wednesday, May 29, 2019
  - Create a webpage that has access to the user's camera [0.5 hr]
  - Connect the web app to a barcode scanning service and product lookup API [9 hrs]
  - Write SQL query to store product data [0.5 hr]

- 5. User Story: As a person with allergies, I want to quickly view a summary of recent products that I've used.
  [5.5 hrs] [Epic E] {Karen, Kendrick} Due: Wednesday, May 29, 2019
  - Create web page to show recent products used [5 hrs]
  - Write SQL query to retrieve needed information [0.5 hrs]

- 6. User Story: As a person with allergies, I want to access a complete list of instances that I've recorded of allergic reactions I've had to a particular topical agent.
  [5.5 hrs][Epic E] {Kris, Kendrick} Due: Wednesday, May 29, 2019
  - Create web page to show all instances of allergic reactions, filtered by topical agent/product [5 hrs]
  - Write SQL query to retrieve allergic reaction data [0.5 hrs]

Backlog:

- User Story: As a person with allergies, I want to be able to fill out a minimal form to request that an administrator add a product to the list of "approved products" when I am unable to find that product in the list of available products.
  [2 hrs][Epic A]
  - Create an observer that directs the user to a page with a request form, if the product is not found
  - Create a form with fields for the user to enter product information

- User Story: As a person with allergies, I want to quickly view past product use and reaction instances in a quick access calendar view.
  [5 hrs][Epic E]
  - Create a webpage connected to main user page structured as a calendar
  - Write SQL query to select data of product use for a user, organized by date
  - Write code for displaying SQL query data in the calendar

- User Story: As a user without internet connection, I want to view a list of products that I should avoid and view recent report summaries.
  [7 hrs][Epic E]
  - Add a function to the page/pop-up where users view products to avoid to allow them to download the list of products
  - Add a function to the page/pop-up where users view report summaries to allow them to download the report summary
    - Should also allow users to download multiple report summaries into one document

- User Story: As a person with allergies, I want to receive recommendations for alternatives to the products I am using to which I might be allergic, using data pooled from the whole community of users.
  [10 hrs][Epic E]
  - Create a page/pop-up that lists products with same/similar purpose as a product a user uses, but with a list of ingredients that does not include the one or more ingredient(s) that have been identified as a likely allergen for the user
  - Write SQL query to select products that meet the above criteria

- User Story: As a registered user, I can log in and the application will authenticate my credentials.
  [4 hrs][Epic F]
  - Create login page
  - Write SQL query that checks the user login credentials are correct

- User Story: As a forgetful user, I can request a password reset so that I can log in if I forget my password.
  [5 hrs] [Epic F]

- User Story: As a person with allergies, I want to identify information that can be shared about and related to a recorded allergic reaction with a remote physician.
  [5 hrs]
  - Create a page/panel where users can select from list of report summaries to share with a physician

- User Story: As a physician, I want to view reports concerning my patients' allergic reactions.
  [15 hrs]
  - Write code for API that can be used to populate the interface the doctor interacts with relevant data
  - Create a simple, intuitive interface that allows the doctor to quickly see proposed allergens, photos of the reaction submitted by the patient, and history of product use and reactions of their patients

- User Story: As an admin, I want to view new product requests, so that I can verify the accuracy of new, unfound products before they are added to the database.
  [3.5 hrs]
  - Create a page which contains the relevant information of the new product request for admins to view

- User Story: As a registered user, I want to receive a notification email once the app identifies a likely allergen for me, in order to discover what I'm allergic to.
  [50 hrs]
  - Create algorithm to determine a "likely" allergen based on past allergic reactions and product use
  - Run algorithm on given time frame of past product use and allergic reactions from database
  - Write code or use API service to send an email to the user once a "likely" allergen has been determined

- User Story: As a registered user, I want to review my likely allergens, along with my history of each and a confidence measure, in order to understand the calculations behind the app's results.
  [50 hrs]
  - Write an algorithm for determining "likely" allergens with "confidence levels" that pulls data from products user uses and allergic reactions user has recorded and puts it

Epic/Themes:

Epic A: As a person with allergies, I want to add products that I am applying to a personal inventory, so that I can track and monitor products that may be causing allergic reactions.

Epic B: As a person with allergies, I want to document each usage of a product listed in my personal inventory that I am applying, so that I can track and monitor products that may be causing allergic reactions.

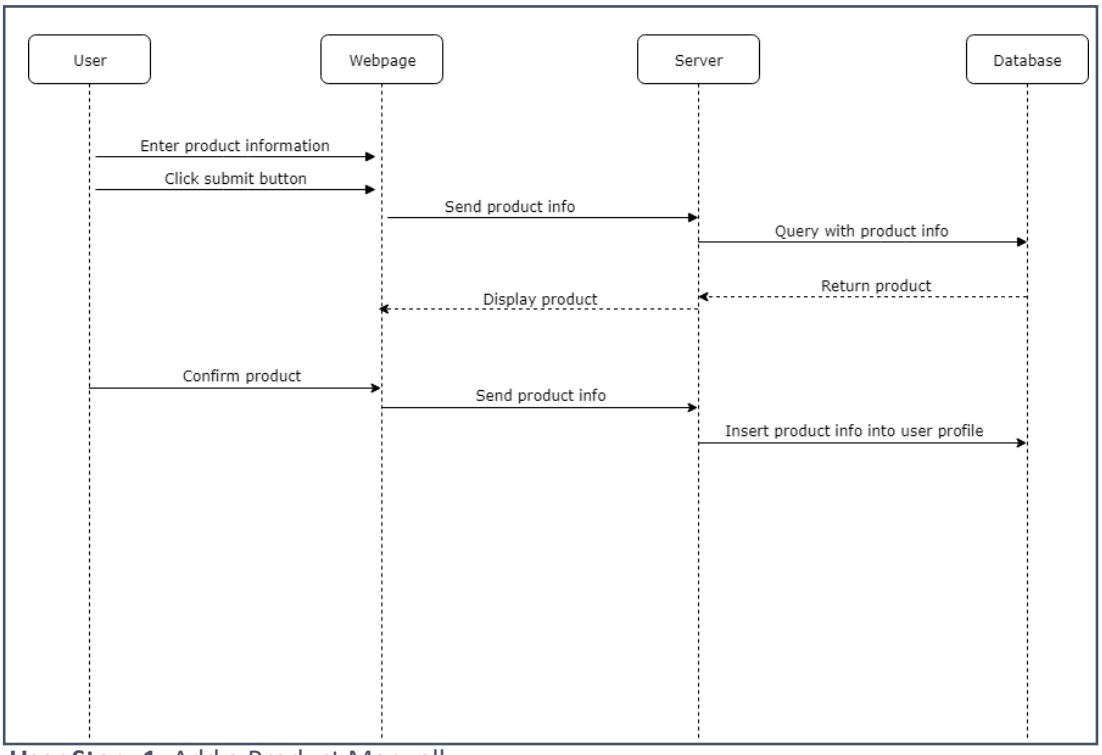
Epic C: As a person with allergies, I want an automated way to document regular product use, so that I can easily enter information about when and where I am using a product.

Epic D: As a person with allergies, I want to record allergic reactions that I experience, so that I can track and monitor my allergic reactions and discover any that are caused by my product usage.

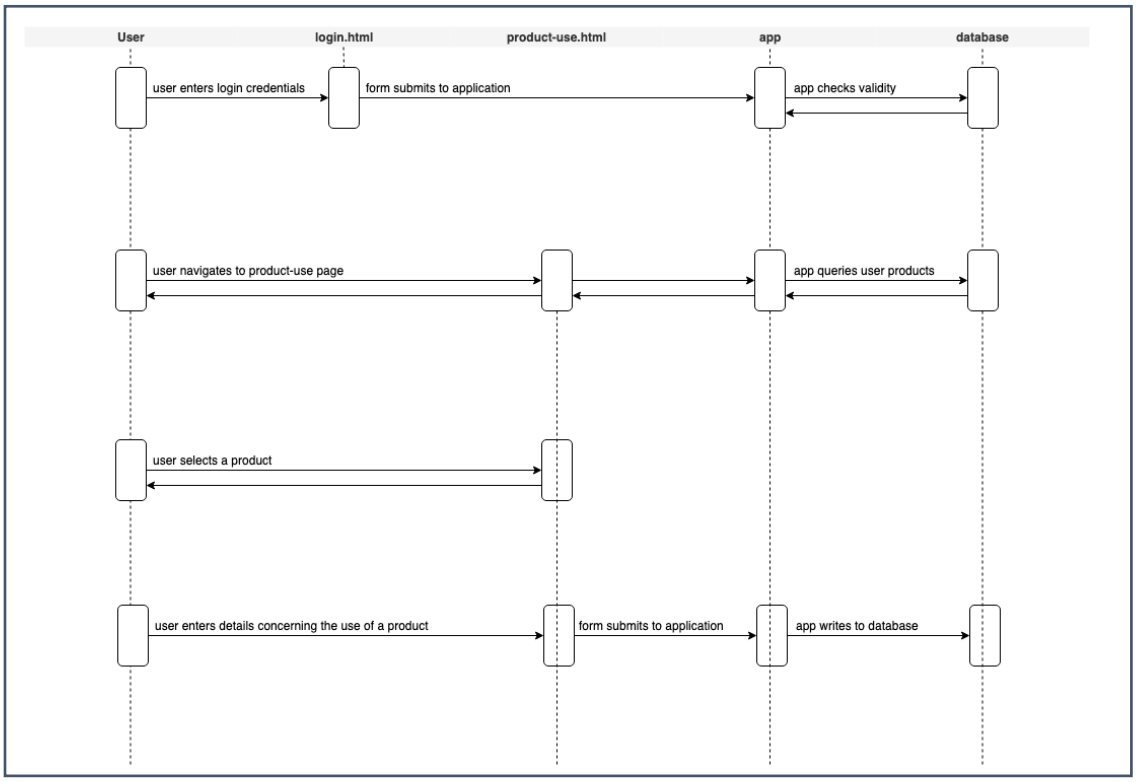
Epic E: As a person with allergies, I want to be able to review the information that I have entered, so that I can discover patterns and determine if there are products I should not be using.

Epic F: As a registered user, I want to log in, so that I can use the application.

UML sequence diagram and Spike Descriptions



User Story 1: Add a Product Manually



User Story 2: Add Instance of Product Use

3. User Story: As **a person with allergies** I want to **record an allergic reaction**: The allergy tracker app users can record an instance of an allergic reaction to a topical agent, like soap, lotion, makeup, etc.

Result: Feasible and will be implemented

A subtask for this user story involves creating an interactive image of a human body for users to tap to indicate where a product was applied or where a reaction occurred. Upon taping the primary region, a pre-populated list of subregions appears beside the body image for more detailed documentation.

During our spike, the main problem we wanted to explore was how to create an interactive image of the body. Manipulating the DOM to create a table of sub regions after taping on the body was not need in the spike because we already knew how to implement that part. We settled on using a simple image map, where the x,y coordinates of polygon vertices were mapped for each main body region.

User Story 3: Add Instance of Reaction

4. SPK: User Story: As a **person with allergies**, I want to **scan the barcode of a product and add it to my inventory of products being used**.

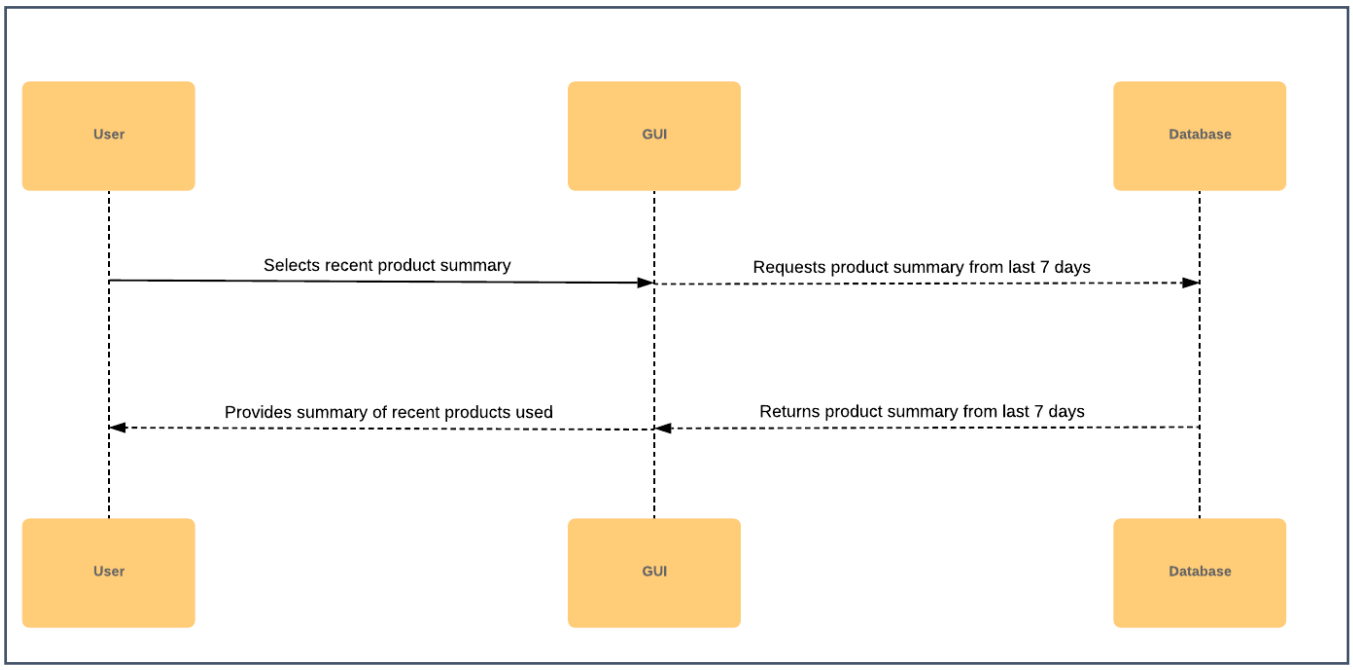
Result: Unfeasible and will not be implemented due to lack of SSL certification

The ability to scan a barcode, retrieve product information, and store it in the user's inventory is an important feature of our application. Access to the user's built-in camera is necessary for being able to read barcodes and perform other functions such as taking a picture of a reaction to send the doctor. Without camera access, the user will be able to still enter products manually.

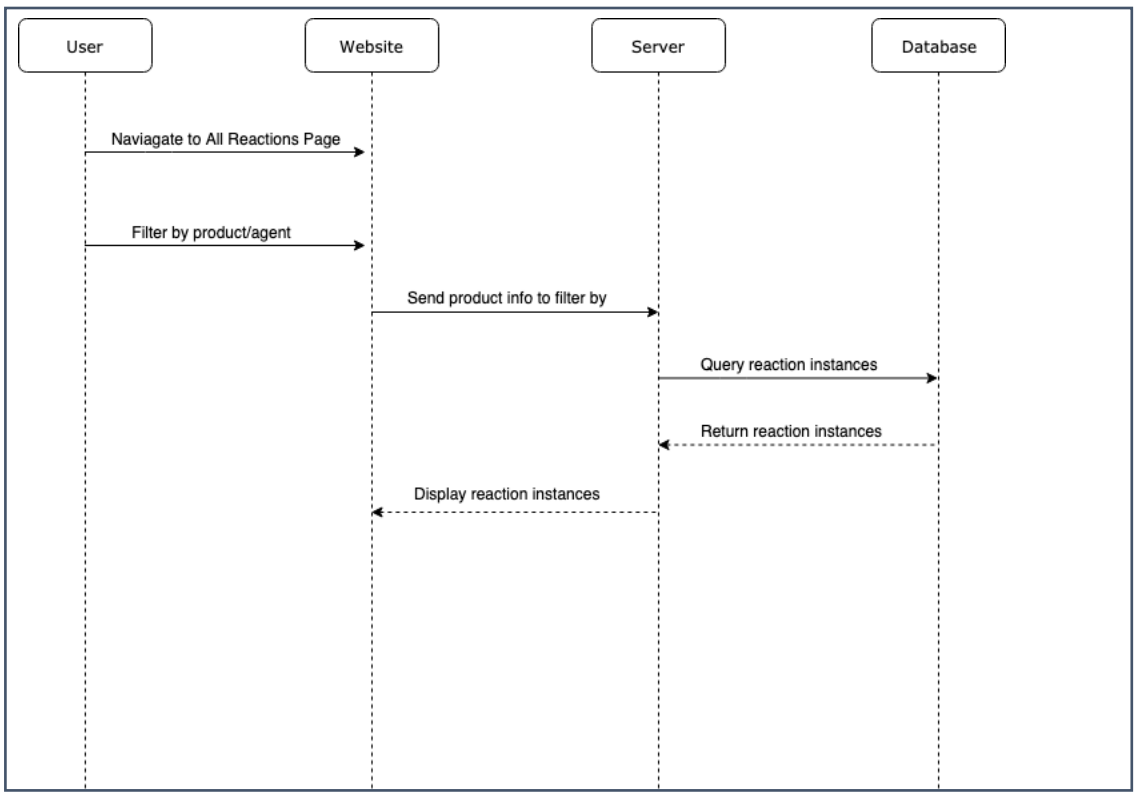
New features for HTML5 includes a video tag that allows video from the camera or another source in the browser without plug-ins. Furthermore, the JavaScript API, getUserMedia, created by WebRTC and supported by all major browsers, lets developers grab images from the live camera feed. We also tested a demo of the very powerful SCANDIT SDK for detecting and reading barcodes, were able to determine that it would be appropriate for our application.

We tested the camera and image grab locally and were successful. However, when we attempted to deploy the spike code on the OSU server, it did not work. We learned that camera access is only available on secure connections and decided to abort further implementation of the SCANDIT SDK.

User Story 4: Add a Product Using a Barcode



User Story 5: Provide List of Reactions



User Story 6: Provide List of Product Used

Implementation Plan

- Environment Setup - Database: Kris and Karen will deliver Tuesday, May 28, 2019.
- Environment Setup - Webpages: the entire is working on this. These will be delivered as part of individual tasks
- User Story 1: Kris and Anthony will deliver Wednesday, May 29, 2019.
- User Story 2: John and Anthony will deliver Wednesday, May 29, 2019.
- User Story 3: John and Kendrick will deliver Wednesday, May 29, 2019.
- User Story 5: Karen and Kendrick will deliver Wednesday, May 29, 2019.
- User Story 6: Kris and Kendrick will deliver Wednesday, May 29, 2019.

Customer Availability

Our customer was willing and able to meet with us on Sunday, May 19, 2019. He also continues to be very generous with the team by answering questions in slack!

Contributions of team members

	Kendrick CHU	Anthony GIULIANO	John HASH	Kristopher HILL	Karen MCFARLAND
Describe user stories			X	X	X
Attribute user stories (due date, tasks, dependencies, story points)			X	X	X
UML sequence diagram/spike description	X	X	X	X	X
Implementation plan			X		X
Stitch together document					X
Review final draft		X	X		X
Submit document					X