RapidRecall System | Cottage Health

Cindy Zhao, Emily Zheng, Jennifer Lopez, John Lins, Leo Zhu

Introduction

Project Overview

The goal of this project is to build an agentic medical product recall system that integrates with Cottage Health's enterprise resource planning (ERP) system. This system automates the process of receiving, classifying, and handling recall information, ensuring timely identification, evaluation, and action on critical recall notices.

Problem Statement

Unsafe or ineffective medical devices present significant risks to patients. In 2024, more than 1,000 medical device recalls were issued in the U.S. Each recall generates multiple notification variations, flooding healthcare systems with mail and creating 'alarm fatigue.'

Today, there is no effective standardization of manufacturer recall notices. It is up to the recipient to read and decipher whether any given notice is relevant and urgent at their facility. At Cottage Health, recall notices may arrive via certified mail, standard mail, private delivery carriers, emails, phone calls, and they are sent to different recipients, sometimes based on the vendor's individual contact at the facility. Once a recall notice is received by Cottage's Supply Chain team, staff checks electronic inventory records using the manufacturer name, code, lot number and date of purchase, to determine whether any of the recalled item is within the system.

Standardizing and improving the recall notification process would involve regulatory action from the U.S. Food and Drug Administration. This could take years. In the meantime, preventable patient harm could occur. To better protect patients, our team aims to design a recall management system that rapidly and consistently responds to recall information within Cottage Health.

Existing Solutions

ECRI is the market leader in the recall notification services field. However, its current recall management system, Recall Management, is only compatible with Workday, making it difficult to integrate with healthcare systems nationwide.

Objective

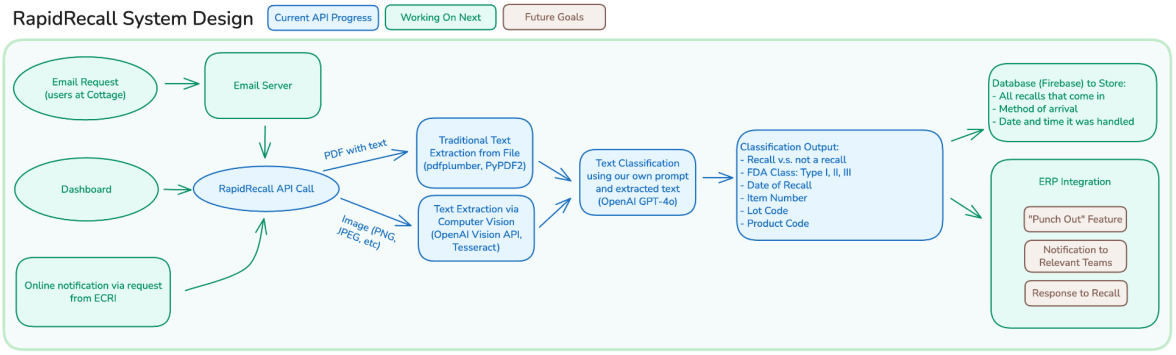
RapidRecall aims to build an automated healthcare product recall management system that reduces manual work while ensuring accuracy (will correctly classify), reliability (will alert), and speed (saves time) and integrates with existing healthcare systems.

Scope

- Extracts text and other info pdf and images
- Classifies recall info into FDA priority class
- Alerts relevant teams that use the recalled product to stop using the product
- Generate response to recall notices
- Maintain a database of recall notice and statuses (pending, alert complete, response given)
- Accessible by healthcare org users and our team

System Overview

The website takes as input a file which can either be a PDF or an image, the backend then uses AI to convert to text, process and classify the recall, send an email to the appropriate person, and then store the recall in the database for later reference. Alternatively, the user may interface with the system using email.



System Features

- Manage recalls
    - Handle each recall that comes in
    - Update response information if the notice is responded to manually
    - Identify recall priority
    - Log information about when a recall came in and how it was addressed (liability issues)

External Interface Requirements

- User Interfaces:
    - Intuitive dashboard for website users
    - Search interface for database
    - Admin users can edit historic recalls
- Software Interfaces:
    - Connects to Firebase database

    - Connects to email server

Requirements

Functional Requirements

- *FR1:* User Login
    - The system will authenticate Cottage Health users upon entering system's dashboard
- *FR2:* Upload Recall via Email and Dashboard
    - Users will be able to email recall PDFs to system's email server and/or upload recalls via dashboard
- *FR3:* Recall Alerts
    - The system will classify all recalls and send out alert notifications based on urgency
- *FR4:* Recall Documentation
    - The system will log all recalls that come in and when they are responded to
- *FR5*: Connection to Cottage Health Database
    - The system will query Cottage Health's database for product storeroom location

Non-Functional Requirements

- *Performance:* Recall PDFs/images should be uploaded/processed within a reasonable amount of time
- *Scale:*
    - The system should support multiple website and/or email users using the platform at the same
    - The system should integrate with other ERP platforms at other healthcare organizations
- *Reliability:* Regardless of using the email or website to perform the upload, adding recall information to our database should not interfere with existing data
- *Security:*
    - Only admin users can access the system data
    - Validate that email senders belong to Cottage Health before processing the recall notification
- *Availability:* There should 99% uptime for hosted version of our backend
- *User-Friendly:* clean UI following accessibility best practices

Milestones

1. Explore existing product recall system software (e.g., ECRI) and pinpoint its practices and limitations

2. Create a product/system design and specification document that has been iterated with mentor feedback
3. Craft an initial MVP for the auto-recall system
4. Develop end-to-end prototype of authentication, dashboard, process endpoint, database endpoint
5. Evaluate the performance (accuracy and latency) of our application
6. Test the application in Cottage Health in internal pilot

User Stories

As a Cottage Health admin user, I can

- Review and correct recall classifications, so that I can ensure that recall notifications are sent out with with the appropriate level of urgency
    - Main Flow:
        - Admin logs into the platform and accesses the list of recalls
        - Admin selects a recall classification to review
        - Admin evaluates the classification based on provided criteria
        - Should be needed, admin updates the classification and submits the update
        - The system logs the change and triggers the appropriate recall notification process
    - Postconditions: Recall is updated in systems and the change is logged; Notification process reflects change

As a Cottage health email user, I can

- Receive a confirmation email after I have submitted a recall notice to RapidRecall, so that I know when my submission was successfully received and is being processed.
    - Main Flow:
        - User scans or receives recall
        - User sends recall to our email
        - Recall is processed on the backend
        - Confirmation email is sent back to the user, they may override if they choose
    - Alternative Flows: Having to manually upload through the application
    - Postconditions: Recall is stored on the backend and email is sent from our servers on behalf of the user

As a Cottage Health website user, I can

- Drag and drop scanned recall notifications onto the RapidRecall website, so that recall notifications can be processed automatically and quickly with minimal effort from me.
    - Main Flow:
        - User logs in to dashboard

        - User clicks “upload document” and submits a PDF/image
        - Recall is processed on the backend
        - Confirmation and results are presented to the user
        - Recall status is updated in our database and shown on the website
    - Alternative Flows: If we fail to extract information from the PDF/image, we alert the user with “Bad image” or “Try again.”
    - Postconditions: Recall notification is fully processed and stored; notifications and alerts are sent.
- Search through a database of past documents I have uploaded and view the information and recall status for each one.
    - Main Flow:
        - User logs in to dashboard and navigates to the Documents database
        - The Documents page displays a database of recall notices that the user has historically uploaded
        - User clicks on the search bar and type in keywords to search for a specific recall
        - Matching recalls are shown to the user
        - User can click on each recall status and see the saved information
    - Alternative Flows: User can scroll through the database
    - Postconditions: All recall notifications are fully processed and stored. The database contains the structured data and zipped PDFs of each recall.

As a Cottage Health employee, I can

- Receive a detailed notification of a recalled item that is relevant to my team, so that I can follow the instructions to safely recall the item.
    - Main Flow:
        - User receives email for recent recalls directed towards that department
    - Alternative Flows: A Cottage Health employee can log onto the dashboard as a website user to view a recall relevant to their team
    - Postconditions: Recall notification is sent to the proper team and should appear in the alerts section based on urgency

Appendix

Platforms/Technologies

- Project Management:
    - Git/Github (version control)
    - Github projects (sprint planning)
    - Discord (communication)
- Backend:

    - Flask (Python web server)
    - Process API: Tesseract OCR (image processing), Gemini 2.5 Pro (text classification)
    - Database: Firebase Firestore
- Frontend: React, Vite, Tailwind CSS
- Development Tools:
    - Authentication: Google OAuth
    - CI/CD: Jenkins

Glossary:

- Email User: Unregistered user uploading PDF through email server
- App User: Registered user allowed to access recall database
- Management Team: Team that we can alert.

References:

- IEEE Std. 830-1998, Software Requirements Specification format