fEMR Operational Dashboard Technical Specification

By Code Blue
Ryan Tsai, Colin Chun, Leander Balmeo,
Moses Epps, Micah Wibowo

Date Created: November 15, 2022
Last Updated: June 13, 2023

Jira Board

Github Repository

Table of Contents

Introduction

Overview

fEMR is a non-profit organization which deploys instances of electronic health record technology to various organizations. fEMR has operations in several countries to help provide medical care. As the number of clients grows, fEMR has an increasingly difficult time keeping track of all instances deployed. In addition, there is currently no system or interface to manage new instance requests; all requests for new instances are done by email. In order to get information on all fEMR deployments, users must access several different sources of information. The president of fEMR also has to spend precious time helping clients create campaigns.

Due to all the aforementioned reasons, there is a need for a dashboard to manage and see statistics on all active instances. There is also a need for a system to manage customer instance requests.

We propose an operational dashboard that provides data visualizations of active instances, weekly graphs and statistics by campaign, and supports management of instances built with Tableau. This dashboard will be connected to the Chain Gang backend to pull necessary data from the fEMR database.

The primary user groups include medical providers who request and manage fEMR kits, fEMR admins, and ICD-10 coders. This project will enable medical providers to view the activity of deployment instances across their campaigns, billing information, and other relevant statistics to help run and organize campaigns. For the fEMR admin, we work to consolidate data from activity in the fEMR world into a streamlined, accessible dashboard.

Terminology

- **Tableau**: A data visualization software that allows the aggregation of data to be displayed on dashboards
- **fEMR-On-Chain**: fEMR data that is stored on the server to eventually be migrated to an AWS database. Database for fEMR which was the project of a past group, Chain Gang.
- **Legacy**: Database for the fEMR server used in offline deployments
- **Dashboard**: A single area to centralize the data relevant to the users' goals
- **Dashboard Client**: A standard user of the Tableau dashboards. Privileges include requesting an organization, requesting an instance, and viewing aggregated data about the organization.

- **Dashboard Admin**: This user is able to approve organization and instance requests as well as view data of all existing organizations.
- **Instances, Kits**: A single fEMR electronic health record system.
- **Campaigns**: An instance or a collection of instances.
- **Operations**: A collection of campaigns.

Background

An operational dashboard is needed for a few reasons:

- Increases visibility of fEMR activity for admins and clients on their operations.
- Consolidates data into a usable/actionable format from the on-chain database.
- Quickly understand fEMR operational health through statistics and visualizations

The need for an operational dashboard came to fruition because the fEMR team learned that they need an accessible tool for viewing all of the kits launched in the field. Without this dashboard, fEMR admins like Sarah and Andi have to manually approve and view campaign requests, which wastes their time when they can be doing more productive things.

The dashboard and its functionality will allow fEMR admins to better understand the overall operations of fEMR and become more organized. It will reduce manual work by streamlining basic campaign statistics and provide critical data for operations and admins. For example clients can see if certain instances are overflowing with patients and can reorganize resources as necessary.

Goals and Technical Requirements

User Personas

Mr. Imaclient (Customer/Client)
Goals:

- Request a new fEMR Mexico operation for his company
- Request a new campaign in a refugee camp in Matamoros, Mexico
- Pull up graphs and information for own deployments

Frustrations:

- No account system simply or interface to manage requests to the dashboard admin

Jane Doeverything (Dashboard Admin)

Goals:

- Access and see all operations and deployments in the fEMR world
- Pull up graphs and information on various metrics of all deployments
- See leaderboard of all and information on ICD-10 coders

Frustrations:

- Currently no centralized area to view all information in the fEMR world

Ben Coding (ICD-10 Coder)

Goals:

- See progress in each mission trip
- Wants to see how he does compared to other coders

Frustrations:

- Cannot see percentage accurate for each mission trip

User Stories

As an administrator, **I want to** see a map with all deployment locations **so that I** can better manage deployments.

As an administrator, **I want to** click on deployment locations on the map **so I know** all of the information about that specific deployment.

As an administrator, **I want to** see the information on each ICD-10 coder **so I know** the pain points and strengths present in our coders.

As a customer, **I want to** see a map with all my approved deployment locations **so I can** better manage my deployments.

As a customer, **I want to** see the medications currently available or running out **so I can** better assess inventory.

As a coder, **I want to** track my progress and accuracy rate **so that I** become more proficient at assigning correct codes.

As a coder, **I want to** see my ranking compared to other coders **so that I** can track my progress.

When someone clicks on patient history **I want** the system to display entire patient medical history **so that** doctor can correctly address patient history.

As an admin **I want to** be able to track basic statistics of all deployed instances **so that** they can keep track of all deployed instances and see how they are performing.

As an admin **I want to** see pins on a map of all active instances **so that I** can visualize where deployed instances are.

As an admin **I want to** click on the pins **to see more** statistics and data about that specific instance.

Technical requirements

- The system shall be compatible with Chrome and Firefox latest versions.
- The system shall be hosted on the cloud without needing installation on customer locally owned servers.
- The system shall display all instances of kits launched in field for customer and admins
- The system should be able to update once connected to the internet

Out of Scope

- We will not implement Log-In and User Account features as that will be a responsibility for the fEMR central team.
- We will not implement account creation.
- Central database: we will only be pulling data from the central database

Future Goals

- Fall Quarter
    - Setup Github repository
    - Complete HIPAA training
    - Outline project specs, deliverables

    - Mockups for potential dashboards
    - Setup collaborative Tableau environment
- Winter Quarter
    - Design Tableau dashboard layout with user experience in mind
    - Mock data for use in researching Tableau functions
    - Create a fully-functional Tableau dashboard with mock data
- Spring Quarter
    - Design an interactive map with expandable pins
    - Display relevant metrics that differ between user groups (dashboard admin, client, coder admin)
        - Dashboard admins see all metrics
        - Client and coder admins see metrics for relevant deployments
    - Usability Testing on the dashboard
    - Migrate all data visualizations to be compatible with the central database
    - Transition Tableau licensing to the clients

Assumptions

We assume that fEMR will have a free Tableau license for use due to their non-profit status and Tableau can be applied to a web application and continuously read from the central database. We are also assuming that there will be minimal restructuring of the on-chain database and its endpoints to support the operational dashboard.

Solutions

Current Solution

There is no current existing solution. However, the current solution to dealing with new instance requests is to do them manually.

Proposed Solution

We propose to use an external component that will be embedded into the website. That component is Tableau. The Tableau dashboards will interact with fEMR legacy and on-chain databases in a read-only format. Dashboards display information pulled from the database. Currently this proposed solution is dependent on the viability of the On-Chain backend as well as the integration on tableau.

The benefits of using tableau is that it is a clean data visualization platform for building dynamic dashboards with useful analytics. A drawback to our proposed solution is that an internet connection is required to access the dashboard.

Business Logic

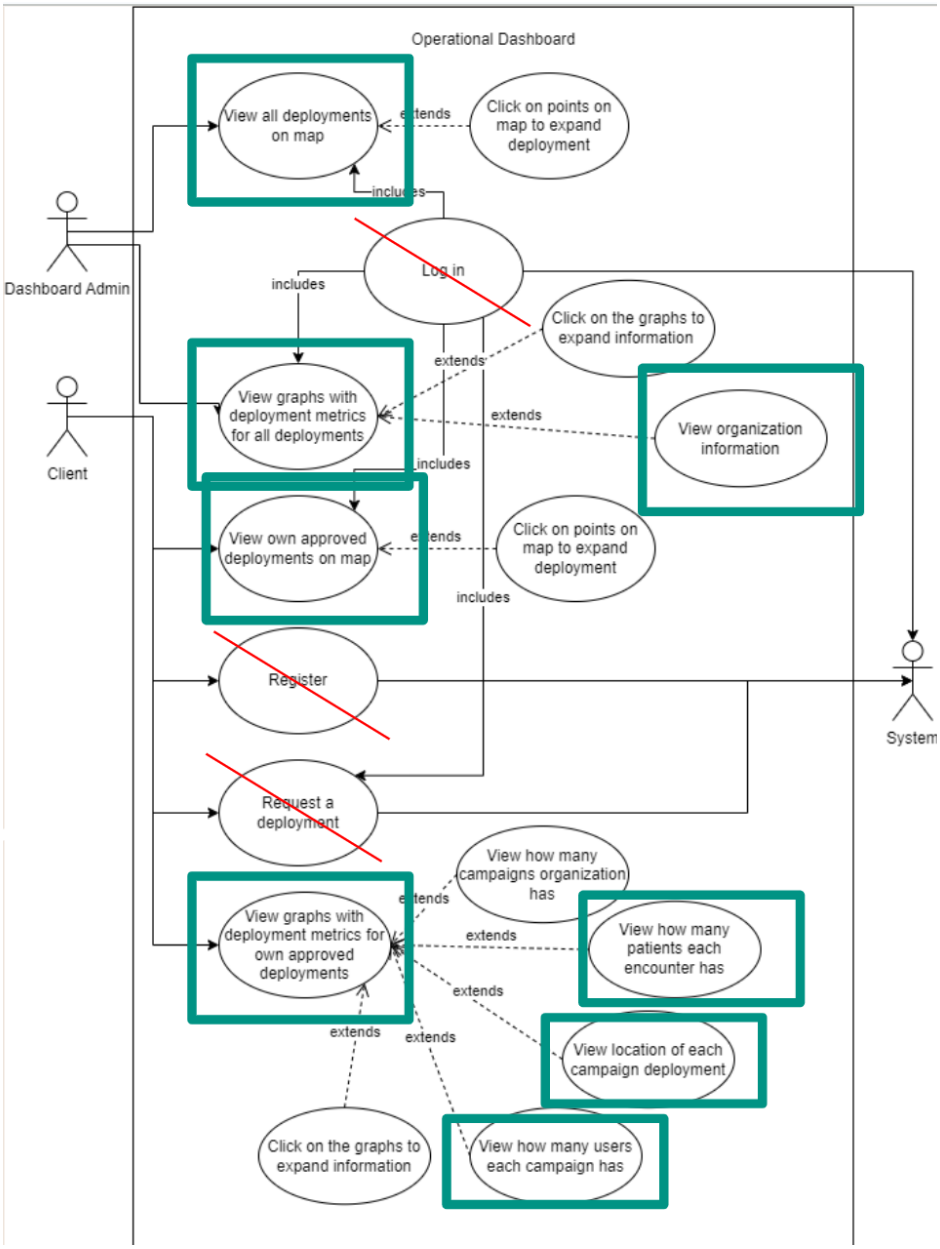
Pseudocode

Download mock data and connect to tableau. Use the workspace area to use map, area charts, text tables, bar charts, and building a dashboard.

Environments that hurry developers/ Conditions that lead to errors and failures.

If the developers are put under stressful environments then decisions will not be thoroughly adjusted to provide excellent code production. Unclear breakdown of direction from clients or restricted assets to generate the wanted product.

Modified Use Case Diagram



Legend:

Green - Implemented/In Progress

Crossed Out (Red) - No longer implementing

No mark - Future work

Backend Schema

Link: **syncroDB.pdf**

UI Wireframe

Figma

https://www.figma.com/file/CNwNuxHIGZaB3LSepZuiNj/Dashboard-Mock-up?node-id=0%3A1&t=0RwhJ8IVvLUJy28p-1

Other Considerations

When it comes to scalability, this isn't seen as a huge issue since the website won't be handling millions of requests at a time. One consideration is about performance of the website, i.e. load times for web pages if we have a lot of packages that need to be loaded like material ui. Limitations really are dependent on data availability from the on-chain backend and potentially data visualizations, if tableau integration does not work with react.

Test Plan

Continuous Integration

Our pipeline will automatically test the build of any new code we implement. In the future we plan on adding continuous deployment as well to further streamline the CI/CD process. We will use GitHub Actions for continuous integration.

Unit Tests

We'll focus on testing for correct endpoints in Tableau and checking that data is appropriate to be used by Tableau.

User Testing

We'll regularly collaborate with our target fEMR users to make sure that the dashboard experience and usage is satisfactory.

We conducted user testing through our peers in the class and had people from each team use our product while noting any defects or issues they came across.

Automated Testing

We tried looking for a suitable automated testing solution for our embedded Tableau dashboard. We ended up using Selenium WebDriver, which works perfectly, but can be a little difficult

to set up due to the embedded Tableau visualization using a shadow DOM. Furthermore, Tableau visualizations hold output values in a container which makes them inaccessible, making Selenium not the best choice for testing database output values. We only used Selenium for testing dashboard functionality like clicking on buttons and seeing certain tables disappear as a result. Selenium testing base code is found in our project repository.

Alternative Solutions

As one of the most popular and efficient open-source libraries for frontend and UI development, React would be very helpful in creating the operational dashboards fEMR is asking for. React allows for the easy build of a frontend web page of the team's design.

Pros:

- A very common library and development tool so it would be easy to onboard new developers
- Easily link to backend and databases
- Some packages available for data analytics and visualizations

Cons:

- Less suited to data visualization and graphs which is the main focus of the dashboards, unlike Tableau

If we did not use Tableau, Excel could be implemented but the process is simpler in Tableau.

There are other applications available but the cost could be exponentially more. An example is using Qlik which is a tool that facilitates the analysis of daily user data. It has the ability to write data back to a database from within the application which allows greater flexibility. It has an in-memory data model that enables fast data discovery. Qlik excels in data visualization, scalability, platform experience, access control & security, API integration and developer tools.

Pros:

- Easy to start
- Easy to use
- Fast and agile

Cons:

- Experts are needed to implement and use it right away
- Less benefits without trained users

- Expensive to scale to hundred - thousands users

Further Considerations

Security Considerations

Security Threats

When it comes to potential security threats there are a few we need to consider:

- Non-dashboard admins viewing metrics not intended for them
- Unauthorized users accessing fEMR data to target marginalized groups
- Most user based security concerns are out of scope for this project as will be handled by another fEMR team
- Adherence to HIPAA standards regarding security issues

How will they be mitigated?

- Separate user groups for dashboard admins and clients
- Minimizing amount of information shown to avoid giving away specific locations
- We utilized database views in sql as a way to abstract out data that need to remain protected in accordance to HIPAA regulations

How will the solution affect the security of other components, services, and systems?

- Solutions will affect accessibility and usage of other fEMR portal functionalities implemented later on

Privacy Considerations

- Data to be used from fEMR-On-Chain and Kits are patient data and thus need to be treated within HIPAA compliance.
- All group members will complete HIPAA training
- Data will be anonymized and queries will only return metadata, not any Protected Health Information (PHI)

Accessibility considerations

Users require a stable internet connection to access the operational dashboard. Users will also need to have visual ability to assess what is on the dashboard. We will conduct usability testing to evaluate the system's accessibility for target users.

Operational Considerations

- Data is stored on AWS. AWS has preventive features for making sure your data is saved if harddrive fails.
- We would restart the server and clean data that needs to be cleaned.
- To keep costs down we will use AWS hosting tools like amplify to keep costs lower. As well as if possible use AWS lambda since you have 1 million free api calls a month with them

Risks

- Tableau Online dependency
    - Reliant on Tableau Online for collaborative work
    - Assumes that tableau workbooks and other hooks are transferable to fEMR's non-profit account on deployment
- Our current solution is always subject to change due to the nature of software development projects

Are there risks that once taken can't be walked back?

- Potential issues transferring to fEMR's non-profit account

What is the cost-benefit analysis of taking these risks?

- Time to implement the dashboard is definitely worth it because the fEMR CEO will ultimately have more time doing important company work.
- Tableau flexibility allows for quick turnaround time on deliverables relative to other frameworks
- Save medical professionals' field operating time by knowing complete patient history

Deliberations

Discussion

Initially, we were debating on using text message or Slack as the primary form of group communication. We settled for text messaging because of accessibility and familiarity. We also had a hard time deciding on the method of communication with the client. The client ultimately preferred writing all questions on a shared Google Doc and answering the questions later. Both of these decisions might change in the future due to new team members and client needs.

We still need to agree on API endpoints and metrics to add to the operational dashboard. We also need to decide with Skeletubbies what metrics are more relevant to coders and how to best display the data.

Open Questions

- What metrics are most relevant to you?
- What metrics do you want most seen?
- Are there any non-English speaking fEMR users?
- Are there any fEMR users with disabilities we need to provide accommodations for?

References and Acknowledgements

https://www.tableau.com/learn/training/20223

Special thanks to Prof. Klingenberg, Sarah, and Andi for their continuous support and contribution to the design of this system!