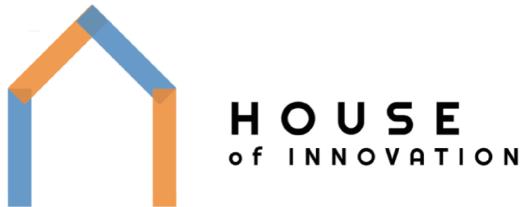


# HIDI Documentation

Innovation Lab - UCI Gap Project



## House of Innovation

Sabrina Lee

Chenyuan Zhang

Abdul Osman

Tianxiong Wu

Christine Trinh

Yihang Yang

March 19, 2021

# Table of Contents

<b>Project Plan</b>	<b>8</b>
Statement of Work	8
1. Scope of Work:	8
2. Location of Work:	8
3. Period of Performance:	8
5. Applicable Standards:	9
6. Acceptance Criteria:	9
7. Special Requirements:	9
Constraints	10
Work Breakdown Structure	11
<b>Business Case</b>	<b>13</b>
1.0 Introduction/ Background	13
2.0 Stakeholders	13
3.0 Competitive Analysis	17
4.0 Current Situation and Problem/ Opportunity statement	19
5.0 Analysis of Options and Recommendation	20
6.0 Preliminary Project Requirements	20
7.0 Budget Estimate and Financial Analysis	21
8.0 Schedule Estimate	21
9.0 Potential Risks	21
<b>Risk Analysis</b>	<b>23</b>
Business Risks	23
Technical Risks	24
Project Management Risks	27
<b>Requirements</b>	<b>30</b>
Functional Requirements	30
External Requirements	32
Software Interfaces	32
Hardware Interfaces	32
Nonfunctional Requirements	32
Use Case Diagram	35
Use Cases/User Stories	36
<b>UI Document</b>	<b>50</b>
Personas	50

Scenarios	53
Comparative Analysis of Interfaces	54
Storyboards	57
User Sign Up and Tutorial Tab Accessing	57
Generalized Transition Network (link)	58
Wireframes	60
Patient Mobile View	60
Patient Desktop View	63
Pharmacist View	67
Usability, Functionality, and Satisfaction Goals	72
Usability Goals	72
Functionality Goals	73
Satisfaction Goals	73
<b>Software Design Document</b>	<b>74</b>
Use-Case Diagram & Summary of Features	74
UML Diagrams	75
App UML Diagram (link)	75
Important Sequence Diagrams Use Cases	76
UML Class Diagrams	82
<b>Component Document</b>	<b>100</b>
Base Components	100
Login and SignUp Buttons	100
Navigation Bar	101
Welcome Summary Message	101
Next Infusion Video Widget	102
Daily Schedule Widget	102
Notification Widget	103
Tutorial Listing	104
Video Player	105
Video Text Section	105
Pharmacist Notes Tab, Step List Tab, and Infusion Notes Tab	107
Daily Infusion Alarms	108
Patient Table	108
Sort and Search on Single Line	109
Frequently Asked Question List	110
Patient Alarm Summary (Pharmacist View)	110
Patient Assigned Tutorials (Pharmacist View)	111

Combined Components	111
Video Tabs	112
Login Combined	113
Sign Up Combined	114
Sign Up Success	115
Pharmacist Settings	117
Pages	118
Welcome Page	118
Error 404 Page	118
Login & SignUp Page	119
Home Page	119
Profile Page	120
Settings Page	121
Tutorial List	122
Single Tutorial Page	123
Schedule Page	124
FAQ Page	125
Contact Information Page	126
<b>Development Environment Setup</b>	<b>127</b>
Credentials	127
Database Setup	128
Client Setup	128
Server Setup	128
Resources	129
<b>REST API Document</b>	<b>129</b>
Patients	130
GET Requests	131
POST Requests	134
Admin	138
GET Requests	138
POST Requests	139
Pharmacists	140
GET Requests	140
POST Requests	143
Events	144
GET Request	144

POST Requests	145
Notifications	146
GET Request	146
POST Requests	147
FAQ	148
GET Requests	148
POST Requests	149
Pharmacy	150
GET Request	150
POST Request	151
Tutorials	152
GET Request	152
POST Requests	153

# Project Plan

## **Statement of Work**

### 1. Scope of Work:

Our team will be building a web-based application to digitize patient care and standardize care instructions in hopes of reducing hospital readmissions. Designing of the application will rely heavily on Figma, an online collaborative user-interface design tool. The product itself will be built using HTML, CSS, JavaScript, ReactJS, AWS DocumentDB, and 3rd Party Security. AWS DocumentDB will be used to host personal user information such as account details and personal medication instructions or reminders. The *3rd party security* is essential for protecting each end-user's privacy.

### 2. Location of Work:

During the quarantine due to the COVID-19 pandemic, all of the work will be performed remotely. We will communicate mostly through Zoom, Discord, and Slack. For technical work, we will be using Google Suite, Figma, and Github for code management.

### 3. Period of Performance:

Starting Date: 10/14/2020

End Date: 3/19/2021

We have 20 weeks to complete the project

We will meet each week with the Innovation Lab stakeholder on Wednesdays at 1:00 pm PST to discuss progress of work as well as general questions about the project.

We also have weekly meetings with Professor Denenberg on Wednesdays at 2:30 pm PST to go over project progression and receive guidance on moving forward.

## 5. Applicable Standards:

Medical content within the web application will be based on reference materials provided by Innovation Lab and will be confirmed by the Innovation Lab subject experts. The web application will include a standard user log-in system.

We will be creating a style/branding guide to keep the User Interface standardized throughout the web application.

## 6. Acceptance Criteria:

### Users

- Digitizing existing documentation of post-care instructional infusion papers
- Notification functionality for patients who wish to be notified about medication times
- Helpful resources to assist patients such as direct connection sources to the pharmacy providing medication
- Post-care patients should be able to easily access their patient-specific instructions through the web platform

### Medical Professionals

- Abilities to assign instructions/notifications to patients.

### Quality & Effectiveness

- The size of the source code should not be mightily large; it is a software characteristic that obviously impacts maintainability.
- The system will be able to run on the most modern web browsers.

## 7. Special Requirements:

- Medical professionals should double check all instructions being added to the web platform to ensure user safety

## Constraints

- ❖ Time
  - We have 20 weeks to complete this project.
- ❖ Lack of Medical Knowledge
  - Our team is largely unfamiliar with the medical field and healthcare industry, such as terminology, processes, and equipment.
  - We are not certified medical professionals and are thus relying on the medical information provided by Innovation Lab to be correct
- ❖ Equipment
  - Video production
    - Insufficient cameras and microphones
    - No access to the medical tools that would need to be filmed
- ❖ Legal Constraints
  - We have to comply with legislation regarding electronic health records and privacy.
  - Examples: Electronic Health Record laws, Health Insurance Portability and Accountability Act (HIPAA)
- ❖ Users
  - Users should have access to an electronic device and have a basic understanding of smart devices.
  - Users need to update their personal information and they can be filched or spoofed.
- ❖ Technical Constraints
  - Innovation Lab requires that all the medical related information/resources (e.g. videos, instructions, etc.) should be prestored in the database. How to efficiently build and fetch information is also a challenge for us.

## **Work Breakdown Structure**

1. Phase 1
  - 1.1. Wireframes
    - 1.1.1. Pharmacist Desktop Wireframes
    - 1.1.2. Patient Desktop Wireframes
    - 1.1.3. Patient Mobile Wireframes
  - 1.2. Mockups
    - 1.2.1. Pharmacist Desktop Wireframes
    - 1.2.2. Patient Desktop Wireframes
    - 1.2.3. Patient Mobile Wireframes
  - 1.3. Database Setup
  - 1.4. Github Repo Setup
  - 1.5. Individual Gantt Chart
  - 1.6. Project Gantt Chart
  - 1.7. Write Project Plan
2. User Testing
  - 2.1. User Testing
    - 2.1.1. User Testing #1
    - 2.1.2. User Testing #2
    - 2.1.3. User Testing #3
  - 2.2. Mockup Revisions
  - 2.3. Frontend Structuring
    - 2.3.1. React Components
  - 2.4. Backend Structuring
    - 2.4.1. Create Models for MongoDB
    - 2.4.2. Create Routes & APIs
  - 2.5. Application Presentation Drafts
    - 2.5.1. Application Presentation Draft 1
    - 2.5.2. Application Presentation Draft 2
3. Phase 3

- 3.1. Software Development
    - 3.1.1. Frontend Development
    - 3.1.2. Backend Development
4. Phase 4
  - 4.1. Communicate with stakeholders
  - 4.2. Communicate with Professor Denenberg
  - 4.3. Communicate with teammates
    - 4.3.1. Distribute work
    - 4.3.2. Make decisions
  - 4.4. Software Testing
  - 4.5. Ensure software implementation follows the high-fidelity mockups
  - 4.6. Software Documentation
    - 4.6.1. Component Documentation
    - 4.6.2. Database Documentation
    - 4.6.3. API & Environment Documentation
5. Phase 5
  - 5.1. Stimulus Package 1
  - 5.2. Draft Stimulus Package 2
  - 5.3. First-Attempt Build on Git
  - 5.4. Stimulus Package 2
  - 5.5. Second-Attempt Build on Git
  - 5.6. Keystroke Analysis
  - 5.7. Deliver final prototype/website to Innovation Lab
  - 5.8. Deliver style/branding guide for future development

# Business Case

**Project Name: Innovation Lab**

**October 16th, 2020**

## **1.0 Introduction/ Background**

The core business goal is to provide an accessible web page for patients to check instructions for at-home medical treatments and consult medical professionals. Our main stakeholders, Innovation Lab, aim to digitize and standardize the hand-out instructions so that patients do not get perplexed with different information from various sources. In addition, patients will get reminder notifications for medical refills, medicine injection times, and re-inspections.

## **2.0 Stakeholders**

1. Innovation Lab
  - a. Priority: 2
    - i. Innovation Lab has the second-highest priority because they commissioned the design of this website and thus determine the requirements and functionality for this project.
    - ii. Only Innovation lab developer administration can modify website contents
  - b. Relation to other stakeholders
    - i. Innovation Lab has the final say on what other stakeholders should be used/ on call for this website
    - ii. Innovation Lab administration license the website to the pharmacies
    - iii. Our team (House of Innovation) reports directly to Innovation Lab on the development process.
  - c. Area of Expertise/ Knowledge

- i. Innovation Lab provides the funding, ideas, and medical-related information to this project.
  - ii. Innovation Lab has set up basic requirements as a guideline for how the web should run and operate.
- d. Primary Concern(s)
- i. Ensuring content of the web page is up to date with proper medical information and instructions
  - ii. Work with pharmacies to add new information if necessary
2. House of Innovation
- a. Priority: 4
- i. The developer team needs to develop the web application.
- b. Relation to other stakeholders
- i. The developer team is building the web application for Innovation Lab.
  - ii. The team reports any progress to Innovation Lab and Professor Denenberg.
- c. Area of Expertise/ Knowledge
- i. Software design
  - ii. Programming capabilities needed for development
  - iii. UI/UX design
- d. Primary concern(s)
- i. Being able to create a website and integrate the necessary requirements to meet patients' needs.
  - ii. The developer team needs to ensure that the UI/UX design is straightforward and simple to use and learn.
  - iii. Privacy laws are being followed to ensure safety and protection of information about their users which include medical professionals and their patients
3. Post-Care Patients (patients)

- a. Priority: 1
  - i. Patients have the highest priority because they are the primary users of the system. It is also essential to ensure that the system is accessible and user-friendly for this group.
- b. Relation to other stakeholders
  - i. Patients will use the system in order to find information and contact other stakeholders (medical professionals) that could aid/ assist their medical needs.
- c. Area of Expertise/ Knowledge
  - i. They are most likely unfamiliar with the treatments they will have to self-administer at home.
  - ii. They may be elderly and not as familiar with using technology.
- d. Primary concern(s)
  - i. Patients' primary concern should be how to find resources they want quickly and accurately.
  - ii. Be able to chat with medical professionals when they need help.
  - iii. Be able to interact and use the system without any obvious lag or bugs.

#### 4. Pharmacist

- a. Priority: 3
  - i. Pharmacists provide professional help by answering questions through FAQ lab/ emergency calls.
- b. Relation to other stakeholders
  - i. Works with Intake Administration
  - ii. Assists Post-Care Patients with medical concerns
- c. Area of Expertise/Knowledge
  - i. Knowledge of medications and correct practices
- d. Primary concern(s)
  - i. Manage multiple patients

- ii. Sign off/double check orders that the Intake Administration makes to Patients' Accounts
  - iii. 24/7 accessibility to Patients
5. Intake Administration
- a. Priority: 3
  - b. Relation to other stakeholders
    - i. Works under the Pharmacists
  - c. Area of Expertise/Knowledge
    - i. Knowledge of medications and correct practices
  - d. Primary concern(s)
    - i. Gathering information for what Patients need
    - ii. Inputting into Patient Accounts what medication, instructions, and alarms they need to follow
6. Professor Denenberg
- a. Priority: 5
    - i. The instructor of the developer team (House of Innovation).
  - b. Relation to other stakeholders
    - i. Consultant to the developer team.
  - c. Area of Expertise/ Knowledge
    - i. Experiences in the IT field
    - ii. Personal experiences of the medical system in the U.S.
  - d. Primary concern(s)
    - i. Weekly meetings
    - ii. Submit deliverables on time

### **3.0 Competitive Analysis**

1. WebMD - <https://www.webmd.com/>
  - a. Description:
    - i. Direct competitor
    - ii. A comprehensive medical information website. Users can find different resources, articles, and help on this website.
  - b. Similarities:
    - i. Provide medical information/instruction
    - ii. Concern about human health
  - c. Differences:
    - i. Medical information/instruction are concluded in text form in articles
    - ii. Users can find professional help such as doctors/ dentist/ symptoms check etc.
    - iii. WebMD provides medical related news/articles
    - iv. WebMD provides only one type user login/register
2. Mayo Clinic - <https://www.mayoclinic.org/>
  - a. Description:
    - i. Direct competitor
    - ii. A professional clinic that users could make appointments and other medical-related supports.
  - b. Similarities:
    - i. Provide medical information/instruction
    - ii. Concern about human health
  - c. Differences:
    - i. Users can find professional help such as doctors/ dentist/ symptoms check etc.
    - ii. Users can request an appointment with a doctor
    - iii. Only patients can register/login

- iv. Medical information/instruction are concluded in text form in articles
  - v. Provides many information related to medical research, education, and job opportunities.
3. Better Me. - <https://betterme.world/>
- a. Description:
    - i. Indirect competitor
    - ii. A fitness app that provides workouts, suggests meals/recipes, and tracks users' daily steps, calories burned, etc.
  - b. Similarities:
    - i. It provides personalized workouts, similar to how the Innovation Lab web application would provide personalized instructions for each patient
    - ii. Has step-by-step instructions for workouts, yoga routines, etc., similar to how the UCI Gap project would provide step-by-step instructions for medical treatments
  - c. Differences:
    - i. Not medical related
    - ii. Has a social media-like portion where users can post and support each other
4. FitBit - <https://www.fitbit.com/global/us/home>
- a. Description:
    - i. Indirect competitor
    - ii. A fitness app that connects to a special wearable device that is able to track heart rate, steps taken, sleep, etc.
  - b. Similarities:
    - i. Provides notifications when fitness milestones are reached, similar to how the UCI Gap project would provide notifications as reminders to take medications
  - c. Differences:
    - i. Not medical related

ii. Has a separate wearable device

Summary:

Compared to its direct competitors, the UCI Gap project has three differences: (1) it mainly focuses on medical instruction rather than news/articles; (2) it provides several types of user accounts such as patients, medical professionals, etc.; and (3) the medical instructions are embedded into media form. All of these features made the project unique from its competitors and more effective for users to gain medical instruction. Although the Gap project doesn't have a function to schedule an appointment with a medical professional, it is expected to have a post-care patient to nurse communication function in the future.

Compared to its indirect competitors, the UCI Gap project will be focused on medical instructions and treatment tracking instead of fitness tracking. However, BetterMe and FitBit both provide personalized dashboards with specific information for each user, and the Gap project will provide this as well. The BetterMe application also has a lot of step-by-step instructions, which will be similar to what the Gap project will include. Overall, the UCI Gap project is fundamentally similar to the indirect competitors but will focus on the healthcare and medical industries.

#### **4.0 Current Situation and Problem/ Opportunity statement**

The proposed Gap project website can be an effective tool for all post-care patients. Nowadays, the current standards are paper instructions sent home for patients to read over. However, not all hand-out instructions are the same; they might vary depending on who is giving them (nurses, doctors, pharmacists, etc.) or the institution. Many post-care patients cannot understand or remember the instructions when they go back home. Some patients incorrectly administer the treatment, which may lead to readmission to the hospital or other serious consequences.

## **5.0 Analysis of Options and Recommendation**

There are three options to address this opportunity:

1. Do nothing.

The current situation of post-care patients is fine. They can keep receiving paper instructions without the Gap project website tool.

2. Design and implement the new Gap project website.

This will standardize instructions and provide more accessible care to patients, overall increasing patient safety.

Based on discussions with stakeholders, we believe that option 2 is the best option.

## **6.0 Preliminary Project Requirements**

The main features of this web application include the following:

1. Integrate medical resources, such as videos and articles that patients can read through and find what they need.
2. Digitized, step-by-step instructions for medical treatments
3. A notification system that reminds patients of major events.
4. Different account types for patients and medical professionals
5. Ability to contact medical professionals 24/7

## **7.0 Budget Estimate and Financial Analysis**

1. Development
  - a. The estimated budget for this project is \$0 for now (10/15/2020) because the developer team is using the free software; however, this number can be

changed in the future as the development process goes. It might require purchasing a URL, database, authentication system, etc.

## 2. Maintenance

- a. After the project is completed, Innovation Lab might need to hire or work with other IT experts to maintain the system. The average cost for a web application is between \$300 to \$2500 per month.  
(<https://www.webfx.com/website-maintenance-pricing.html>)

## 3. Profits

- a. Profits could be possible in the future if other health companies want to purchase the system as a bundle. Ads or other collaborations could also generate revenue.

## **8.0 Schedule Estimate**

Innovation Lab is expecting the project to be completed over the course of 20 weeks starting on October 14, 2020 and ending on March 19, 2021. We should have a proof of concept and the beginning of some prototypes at the end of the first 10 weeks. The remainder of the time will go into fleshing out and refining the product.

## **9.0 Potential Risks**

### 1. Legality

- a. Privacy laws, such as Electronic Health Record laws and the Health Insurance Portability and Accountability Act (HIPAA), make it very difficult to create so many personal accounts with medical information attached to them all

### 2. Security

- a. Personal accounts with medical records have the potential to be hacked into and stolen

### 3. Safety

- a. If instructions uploaded to the web platform are unclear or not validated correctly, it may cause harm to users relying on the correct information

# Risk Analysis

## **Business Risks**

Commercialization

<b>Risk</b>	<b>Description</b>	<b>Solution</b>	<b>Probability</b>	<b>Impact</b>
Website Adoption	Companies and hospitals would not want to adopt this website system.	We can do user research to ensure that potential users will benefit from the website. Innovation Lab can do more marketing to advertise the product.	Low	Medium
Branding	Innovation Lab hopes that many hospitals will use the website, so the branding must be attractive regardless of who is using it.	We can do research on white labeling and create a neutral aesthetic for the website.	Medium	Low

## Budget

Risk	Description	Solution	Probability	Impact
No funding	The budget is currently \$0. However, there may be expenses later in development, such as the purchasing of third-party software or website URL.	We will discuss alternatives or expenses with the Innovation Lab sponsor.	Medium	Medium
No funding for calling API	Innovation Lab sponsor is pushing for a feature where audio and video calls are readily available	Discuss with Innovation Lab that we may need funding to purchase an API from a third party to support the feature or change the scope of the project.	High	High

## Technical Risks

### Hardware

Risk	Description	Solution	Probability	Impact
Incompatibility	If pharmacies have outdated computers that are incompatible with the system built, the system will be rendered unusable	Research what computers pharmacies use and ensure they can use browsers that can support the website.	Low	High
Computer Supply	Pharmacies must have enough computers that can be allocated for use in conjunction with the system to ensure efficiency	Focus the design in ways that would allow one admin to control multiple accounts	Low	Medium

## Software

Risk	Description	Solution	Probability	Impact
Incompatible Browsers	If the browsers in the pharmacies are incompatible with the system built, the system will be rendered unusable	The website can be built using common languages and frameworks to prevent incompatibility	Low	High
Current Medical Records not Transferable	System may need some access to some patient's medical records or information.	If automatic access can not be achieved, may need to allow admins to manually add information	Medium	Medium
Insufficient Software	The tools we planned to use can not accomplish all the tasks we set out to do	We can find software that is commonly used and that has a wide variety of uses.	Low	High
Insufficient Database Size	Database must be able to properly store and retrieve inputted data without memory issues	Implement a database that has lots of storage space	Medium	Medium
Incompatible API	API for video and audio calls should be compatible with the browsers we build the application on	Research compatibility of API before purchasing	Low	High

## Security

Risk	Description	Solution	Probability	Impact
Patient Privacy	System may keep track of informations such as patient's names, phone numbers, etc	Limit the amount of information collected	High	High
Medical	Some medical records may	Limit the amount of	High	High

Records	be stored in the system in order to validate patients	information collected		
---------	---	-----------------------	--	--

## Quality

Risk	Description	Solution	Probability	Impact
Accessibility	Poor accessibility such as poor attention to vision impairments, sight impairments, etc will impact the performance of the system	We will implement accessibility through alternative text, colorblind-friendly color schemes, different size fonts, and more.	High	High
Instruction Verification	The instructions uploaded to the system must be validated and verified to ensure patient safety	We will validate and verify the medical information with the Innovation Lab subject experts.	High	High
Clarity	The system should be designed to make its functionality as clear and easy to understand as possible	Follow design guidelines and conduct user testing	Medium	High
Simplicity	The system should be designed to be as simple and user friendly as possible without compromising functionality	Follow design guidelines and conduct user testing	Medium	High
Pharmacists Training	Pharmacists will need to be trained to use the system properly	We will implement a short tutorial when users first sign up.	High	High
Patient Training	Pharmacists will need to be trained to teach patients how to use the system as well	We will implement a short tutorial when users first sign up. Pharmacists can help guide patients	High	High

		through the tutorial.		
Staff	The system requires a large number of staff to ensure the system works efficiently. For example, there should be a 24 hour live video chat implemented into the system	We will discuss this concern with Innovation Lab.	Medium	High

## Project Management Risks

### Estimation

Risk	Description	Solution	Probability	Impact
Time	We have 20 weeks to finished the whole project, it is possible that we cannot achieve our minimum viable product within the planned time	We should rank the requirements' priority and focus on the most important functions/features first	High	High
Capability	Currently, we probably do not have the complete skills to build/finish the website or certain features in the website.	We should prepare to learn new skills to achieve frontend backend coding ability while we are developing the project.	High	High
Schedule	We may not be able to submit certain deliverables on time due to the wrong estimation of the time it will take.	We need to start early if we are not sure how long the deliverable will take and plan our time precisely.	Low	High

## Scope

Risk	Description	Solution	Probability	Impact
Scope Creep	As the project is developed, the sponsors might require new features/functions, which will increase the complexity of the project.	We should communicate effectively at the beginning and document the requirements so that both sides are clear about what to expect.	High	High

## Communication

Risk	Description	Solution	Probability	Impact
Time	Our meetings with sponsors and the professor are recurring weekly at the same time. Some of our team/sponsor team may have time conflict and cannot attend the meeting.	We will make detailed notes for every meeting we have, so that members who miss a meeting can catch up with our progress.	High	High
Quantity	Meeting once a week may not be enough if we meet other problems or have new questions as we are developing the project	We should communicate with the sponsor team in our slack channel and the message group and make sure we check the message daily.	High	High
Quality	We might meet communication quality problems such as technical	We should ask immediately if we encounter trouble understanding what	High	High

	<p>issues or comprehension, so we cannot hear/understand others' words.</p>	<p>the sponsors want to express and organize our questions/ideas to maximize efficiency.</p>		
Conflict	<p>We all have different ideas when we are designing as a team, disagreement may lead to team conflict</p>	<p>We should listen to each other and respect each other's ideas. We should rationally analyze our ideas and find the best decision for the whole team.</p>	Low	High

# Requirements

## Functional Requirements

### Must have

#	Title	Description
1	(Patient) Access Tutorial Tab	Patients will have the ability to access Tutorials tab
2	(Patient) Access Medication/Infusion Information	Patients will have the ability to view personal medication information
3	(Patient) Select Tutorial Type	Patients will have the ability to select between the three available tutorial types (step-by-step, video, and digital doc)
4	(Patient) Access Profile Page	Patients will have the ability to access their profile page
5	(Patient) Access Dashboard	Patients will have the ability to access their dashboard
6	(Patient) Access FAQ	Patients can access Frequently Asked Questions tab
7	(Patient) View Alarm	Patients can view alarms scheduled for a specific day in a given week
8	(Patient) Signup	Patients will sign up for their account using a given code
9	(All Users) Login	All users will be able to login to their account
10	(Patient) Recover Account	Patients can follow a procedure to recover their account in the case of forgotten login information
11	(Pharmacist / Intake Support) Delete Patient Account	Pharmacist will have the ability to delete patient accounts
12	(Pharmacist / Intake Support) Set Alarms	Pharmacists will have the ability to set patient alarms

13	(Pharmacist / Intake Support) Modify Alarms	Pharmacists will have the ability to modify patient alarms
14	(Pharmacist / Intake Support) Assign Tutorials	Pharmacists will have the ability to assign tutorials to patients based on medical needs
15	(Pharmacist / Intake Support) Edit Tutorials	Pharmacists can edit units required in tutorials
16	(Pharmacist / Intake Support) View Patient Profile	Pharmacists can view specific patient information
17	(Pharmacist / Intake Support) View Patient List	Pharmacists can view patient database specific to their pharmacy
18	(Admin) Create Pharmacy/Pharmacist Account	Admins can create accounts for pharmacists
19	(Admin) View Pharmacy Account List	Admins can view the list of pharmacy accounts

## Nice to have

#	Title	Description
1	(Patient) Use Live Chat	Patients can access live chatting to message or call for assistance from the Live Help bar.
2	(Pharmacist / Intake Support) Update Pharmacy Profile	Pharmacists can update their “about” information for patients to view
3	(Pharmacist / Intake Support) Add Tutorial Notes	Pharmacists will have the ability to add notes to a patient’s tutorial.
4	(Pharmacist) Approve Infusion Changes	Pharmacists will have the ability to approve any changes other intake admins create
5	(Pharmacist / Intake Support ) Recover Account	Pharmacists can follow a procedure to recover their account in the case of forgotten login information

## Great if we can get to it

#	Title	Description
1	(Patient) Use Video Chat	Patients can access video chat to call for assistance from the Live Help bar.

## External Requirements

### Software Interfaces

Software	Description
The Gap Project	The system will work on any smart device with a web browser.
Database	TBD
Javascript, HTML, CSS, web frameworks	Languages such as Javascript, HTML, CSS, and other web frameworks will be used for developing this system.

### Hardware Interfaces

Hardware interfaces include and are limited to computers that the software runs on. It can be a mobile smart phone of any kind, a tablet, or a computer with a web browser such as Chrome, Firefox, Safari, Internet Explorer, Edge, etc.

## Nonfunctional Requirements

- **Compatibility**

PRIORITY: HIGH

- The system must be compatible with any smart device with a web browser.
- The system must be responsive on a range of screen dimensions such as non-desktop environments.

- **Consistency**

PRIORITY: MEDIUM

- The GUI and the layout of the system should be consistent between different screens.

- **Usability**

PRIORITY: HIGH

- All icons and interfaces shall be recognizable by various user groups, regardless of their age, ability, or technology skills.
- The interface must be easy-to-use and intuitive.

- **Security**

PRIORITY: HGIH

- The system shall use a secured database (such as AWS) for authentication and store, transmit sensitive information.
- Patients' accounts will be created by Intake/Pharmacists' accounts, in order to avoid giving access to unauthorized users.
- The information to create Patients' accounts should be minimal, private, and protected.

- **Understandability**

PRIORITY: HIGH

- 95% of the new users can successfully learn to operate the system with proper instruction.
- All main use cases should be accessible on the front page and all main high level goals should be shown on the front page.

- **Maintainability**

PRIORITY: MEDIUM

- Code should be extensible and follow good coding practices.
- The system should be updated regularly to meet more and more users and information storage.
- Written code must be flexible to allow for future changes and improvements.
- Written code should be clean and functional, with appropriate comments.
- The system should use the newest security protocol and the database should regularly clean up useless and erroneous data.

- **Reliability**

PRIORITY: HIGH

- Any incorrect input from a user needs to raise an exception and a prompt until it is resolved.
- All patient data should be presented in real time.
- All infusion related data should be accurately presented to the user.
- All patient data should be stored in a protected database.
- The application should be fully functional and accurate on different browsers.

- **Regulatory**

PRIORITY: HIGH

- The system should be HIPAA compliant, including infrastructure.
- If eCommerce functionality is part of the feature set then PCI DSS3.0 requirements must also be met. Other regulations to consider are GDPR and CCPA data privacy requirements.

- **Accessibility**

PRIORITY: HIGH

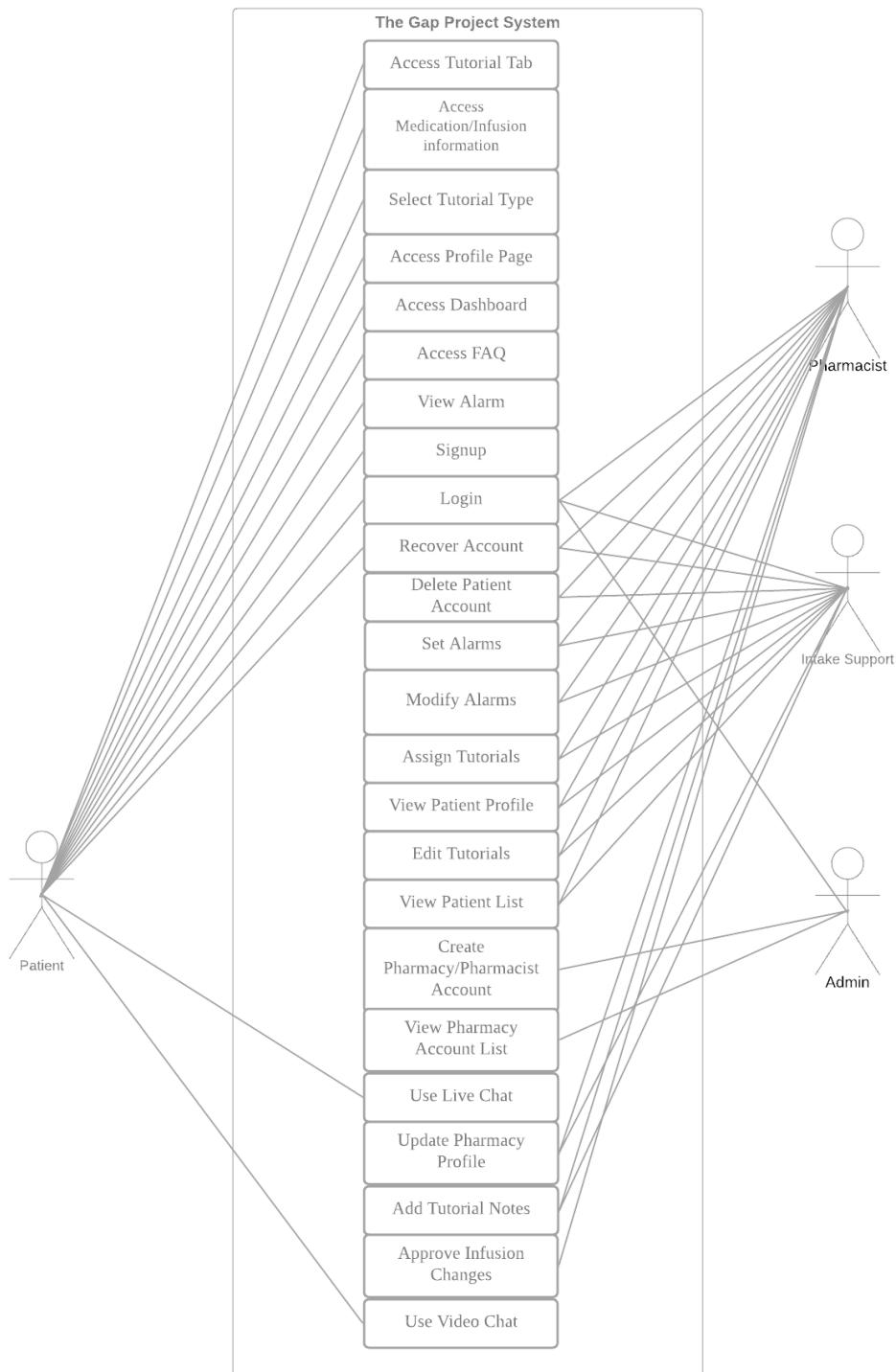
- **Text to Speech:** it will be used to read the text on the screen for the users who potentially cannot read or have vision impairments
- **Increase Contrast:** Increase color contrast between words and background colors.

- **Testability**

PRIORITY: HIGH

- All use cases should be tested.
- 90% requirements should be measurable.
- 100% Must to Have requirements should be met at the end of development.

## Use Case Diagram



## Use Cases/User Stories

Use Case # & Title	01 - Access Tutorial Tab
Priority	Must Have
Description	Patient will have the ability to access Tutorials Tab
Actors	Patient System
Triggers	<ul style="list-style-type: none"> <li>1. Patient clicks Tutorial Tab</li> </ul>
Main Flow	<ul style="list-style-type: none"> <li>1. Patient clicks Tutorial Tab</li> <li>2. New Tutorials page will be shown with three types of tutorials</li> <li>3. Patient clicks on the “how to administer” option within Infusion Info page</li> </ul>
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	02 - Access Medication/ Infusion Information
Priority	Must have
Description	Patient will have the ability to view their personal medication information
Actors	Patient System
Triggers	<ul style="list-style-type: none"> <li>1. Patient clicks on info tab</li> </ul>
Main Flow	<ul style="list-style-type: none"> <li>1. Patient clicks on info tab</li> <li>2. Two options are shown: Infusion information and Medication information</li> <li>3. Selecting Infusion Info <ul style="list-style-type: none"> <li>a. New information page for their specific medication will be shown, as well as a “how to</li> </ul> </li> </ul>

	<p>administer” option</p> <p>4. Selecting Medication Info</p> <ul style="list-style-type: none"> <li>a. New information page for their specific medication will be shown, as well as information on dosage</li> </ul>
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	03 - Select Tutorial Type
Priority	Must have
Description	Patient will have the ability to select between the three available tutorial types (step-by-step, video, and digital doc)
Actors	Patient System
Triggers	<p>1. Patient enters the Tutorial Page</p>
Main Flow	<p>1. Patient can select any of the three tutorial types to progress further</p> <p>2. Step-By-Step</p> <ul style="list-style-type: none"> <li>a. Patient is directed to the step-by-step tutorial that they can follow and click through</li> </ul> <p>3. Video</p> <ul style="list-style-type: none"> <li>a. Patient is directed to the video tutorial that they can play</li> </ul> <p>4. Digital Document</p> <ul style="list-style-type: none"> <li>a. Patient is directed to the instructions tutorial in document form</li> </ul>
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	04 - Access Profile Page
Priority	Must have
Description	Patients will have the ability to access their profile page
Actors	Patient System
Triggers	<ul style="list-style-type: none"> <li>1. Patient clicks the Profile button</li> </ul>
Main Flow	<ul style="list-style-type: none"> <li>1. Patient clicks the Profile picture on the navigation bar</li> <li>2. Patient clicks the Profile button</li> </ul>
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	05 - Access Dashboard
Priority	Must have
Description	Patients will have the ability to access their dashboard
Actors	Patient System
Triggers	<ul style="list-style-type: none"> <li>1. After patient login</li> <li>2. Patient clicks Home button</li> </ul>
Main Flow	<ul style="list-style-type: none"> <li>1. Patient inputs their username and password           <ul style="list-style-type: none"> <li>a. Redirect to the Home page (dashboard)</li> </ul> </li> <li>2. Patient clicks the Home button on the navigation bar</li> </ul> <p>Mobile device:</p> <ul style="list-style-type: none"> <li>1. Patient clicks the Hamburger Menu</li> <li>2. Patient clicks the Home button</li> </ul>
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	06 - Access FAQ
Priority	Must have
Description	Patient can access Frequently Asked Questions tab
Actors	Patient System
Triggers	<ul style="list-style-type: none"> <li>1. Patient clicks on FAQ tab</li> </ul>
Main Flow	<ul style="list-style-type: none"> <li>1. Patient clicks on FAQ tab</li> <li>2. New page is presented with Frequently Asked Questions listed on text format</li> <li>3. Patient can read FAQ</li> </ul>
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	07 - View Alarm
Priority	Must have
Description	Patient can view alarms schedules for a specific day or week
Actors	Patient System
Triggers	<ul style="list-style-type: none"> <li>1. Patient clicks the Alarm Tab.</li> </ul>
Main Flow	<ul style="list-style-type: none"> <li>1. Patient enters Alarm Tab</li> <li>2. Alarm schedule for the week is displayed</li> <li>3. Patient can hover over an alarm to see hover box with information: <ul style="list-style-type: none"> <li>a. Medication + dosage</li> </ul> </li> <li>4. Patient can click “Date” button to change to specific day</li> <li>5. Page will change to show week that the specific day resides in, and will highlight the specified day</li> </ul>
Alternative Flow	NA
Exception Flow	NA

Notes	NA
-------	----

Use Case # & Title	08 - Signup
Priority	Must have
Description	Patients will sign up for their account using a given code
Actors	Patient System
Triggers	<ul style="list-style-type: none"> <li>1. Patient clicks Sign Up button</li> </ul>
Main Flow	<ul style="list-style-type: none"> <li>1. Patient enters information</li> <li>2. Patient clicks Submit button</li> <li>3. Patient is redirected to the login form</li> </ul>
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	09 - Login
Priority	Must have
Description	All users will be able to login to their account
Actors	Patient Pharmacist Intake Support Admin System
Triggers	<ul style="list-style-type: none"> <li>1. Patient/Pharmacist/Admin clicks the login button</li> </ul>
Main Flow	<p>Patient:</p> <ul style="list-style-type: none"> <li>1. Patient sign up (UC#8)</li> <li>2. Patient enters their username and password</li> <li>3. Patient clicks the login button</li> </ul> <p>Pharmacist/Intake:</p>

	<ol style="list-style-type: none"> <li>1. Admin creates pharmacy/pharmacist account (UC#18)</li> <li>2. Pharmacist enters username and password</li> <li>3. Pharmacist clicks the login button</li> </ol> <p>Admin:</p> <ol style="list-style-type: none"> <li>1. Admin enters username and password</li> <li>2. Admin clicks the login button</li> </ol>
Alternative Flow	NA
Exception Flow	<ol style="list-style-type: none"> <li>1. Pharmacist gives the wrong code</li> </ol>
Notes	NA

Use Case # & Title	10 - Recover Account
Priority	Must have
Description	Patients can follow a procedure to recover their account in the case of forgotten login information
Actors	<ol style="list-style-type: none"> <li>1. Patient</li> <li>2. System</li> </ol>
Triggers	<ol style="list-style-type: none"> <li>1. Patient clicks on “Forgot My Login” on the login page</li> </ol>
Main Flow	<ol style="list-style-type: none"> <li>1. Patient clicks on “Forgot My Login” on the login page</li> <li>2. Patient enters in identifying information           <ol style="list-style-type: none"> <li>a. Name</li> <li>b. Date of birth</li> </ol> </li> <li>3. Patient receives an e-mail with their account code</li> <li>4. Patient enters in their code and logs into their account</li> </ol>
Alternative Flow	NA
Exception Flow	NA
Notes	It is assumed that patients will receive the sign-in code provided to them from their initial account creation.

Use Case # & Title	11 - Delete Patient Account
Priority	Must have
Description	<ol style="list-style-type: none"> <li>1. Pharmacist will have the ability to delete inactive patient accounts</li> </ol>
Actors	Pharmacist Intake Support System
Triggers	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake clicks Delete Account button</li> </ol>
Main Flow	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake goes to View Patient List(UC#17) Page</li> <li>2. Pharmacist/Intake selects inactive patient account(s).</li> <li>3. Pharmacist/Intake clicks Delete Account button.</li> </ol>
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	12 - Set Alarms
Priority	Must have
Description	<ol style="list-style-type: none"> <li>1. Pharmacist will have the ability to set patient alarms</li> </ol>
Actors	Pharmacist Intake Support System
Triggers	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake clicks Set Alarm button on patient's page</li> </ol>
Main Flow	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake clicks Set Alarm button on patient's page</li> <li>2. Pharmacist/Intake enters the information about the alarm and the time</li> <li>3. Pharmacist/Intake clicks Save button</li> </ol>
Alternative Flow	NA
Exception Flow	NA

Notes	Changes made by Intake need to be approved by Pharmacists (UC#23).
-------	--

Use Case # & Title	13 - Modify Alarms
Priority	Must have
Description	<ol style="list-style-type: none"> <li>1. Pharmacists will have the ability to modify patient alarms</li> </ol>
Actors	Pharmacist Intake Support System
Triggers	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake selects one specific alarm on patient's page</li> </ol>
Main Flow	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake selects one specific alarm on patient's page</li> <li>2. Pharmacist/Intake modifies the information and/or time</li> <li>3. Pharmacist/Intake clicks Save button</li> </ol>
Alternative Flow	NA
Exception Flow	<ol style="list-style-type: none"> <li>2. Pharmacist/Intake may enter invalid characters which would result in the failure of modifying alarms.</li> </ol>
Notes	Changes made by Intake need to be approved by Pharmacists (UC#23).

Use Case # & Title	14 - Assign Tutorials
Priority	Must have
Description	<ol style="list-style-type: none"> <li>1. Pharmacists will have the ability to assign tutorials to patients based on medical needs</li> </ol>
Actors	Pharmacist Intake Support System
Triggers	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake clicks Tutorial button on patient's</li> </ol>

	page
Main Flow	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake clicks Tutorial button on patient's page</li> <li>2. Pharmacist/Intake assigns Tutorials to the patient</li> <li>3. Pharmacist/Intake clicks the Save button</li> </ol>
Alternative Flow	NA
Exception Flow	NA
Notes	Changes made by Intake need to be approved by Pharmacists (UC#23).

Use Case # & Title	15 - Edit Tutorials
Priority	Must have
Description	Pharmacists can edit units required in tutorials
Actors	Pharmacist Intake Support System
Triggers	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake clicks edit Tutorial button</li> </ol>
Main Flow	<ol style="list-style-type: none"> <li>1. Pharmacist/Intake goes to Tutorials list</li> <li>2. Pharmacist/Intake selects desired Tutorial</li> <li>3. Pharmacist/Intake clicks edit button</li> <li>4. Pharmacist/Intake changes field inputs</li> <li>5. Pharmacist/Intake saves.</li> </ol>
Alternative Flow	NA
Exception Flow	NA
Notes	Changes made by Intake need to be approved by Pharmacists (UC#23).

Use Case # & Title	16 - View Patient Profile
Priority	Must have
Description	Pharmacists can view specific patient information

Actors	Pharmacist Intake Support System
Triggers	1. Pharmacist/Intake clicks a specific patient on Patient page
Main Flow	1. Pharmacist/Intake clicks Patients button 2. Patient list displayed 3. Pharmacist/Intake clicks a specific patient 4. The patient's information displayed
Alternative Flow	3.a. Pharmacist/Intake finds a specific patient through the Search feature.
Exception Flow	1. Pharmacists can view specific patient information
Notes	N/A

Use Case # & Title	17 - View Patient List
Priority	Must have
Description	Pharmacists can view patient database specific to their pharmacy
Actors	Pharmacist System Patient
Triggers	1. Pharmacist clicks Patient List
Main Flow	1. Pharmacist clicks Patient List 2. Pharmacist can view full list of patients
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	18 - Create Pharmacy/Pharmacist Account
Priority	Must have

Description	Admins can create accounts for pharmacists
Actors	Admin System Pharmacist
Triggers	1. Admin clicks Create New button in Pharmacist list
Main Flow	1. Admin clicks Create New button in Pharmacist list 2. Admin enters in Pharmacist information into fields 3. Admin clicks save. 4. New pharmacist account is created
Alternative Flow	NA
Exception Flow	1. Pharmacist may enter invalid characters which would result in the failure of modifying alarms.
Notes	NA

Use Case # & Title	19 - View Pharmacy Account List
Priority	Must have
Description	Admins can view the list of pharmacy accounts
Actors	Admins System
Triggers	1. Admin clicks Pharmacies button
Main Flow	1. Admin clicks Pharmacies button 2. Admin can view full list of pharmacy accounts
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	20 - Use Live Chat
Priority	Nice to have
Description	Patients can access live chatting to message or call for assistance from the Live Help bar.

Actors	Patient Pharmacist Intake Support System
Triggers	1. Patient clicks Live Help button
Main Flow	1. Patient clicks Live Help button 2. Chat box is pulled up 3. Patient can input message into chat box and press send to send message to Pharmacist
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	21 - Update Pharmacy Profile
Priority	Nice to have
Description	Pharmacists can update their “about” information for patients to view
Actors	Pharmacist Intake Support System
Triggers	1. Pharmacist clicks edit on their about information
Main Flow	1. Pharmacist clicks edit on their about information 2. Pharmacist enters in desired changes to information 3. Pharmacist enters save
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	22 - Add Tutorial Notes
Priority	Nice to have
Description	Pharmacists will have the ability to add notes to a patient’s

	tutorial.
Actors	Pharmacist Intake Support Patient System
Triggers	1. Pharmacist clicks Add Notes to tutorial module
Main Flow	1. Pharmacist clicks Add Notes to tutorial module in patient's account 2. Pharmacist enters in desired notes 3. Pharmacist clicks save
Alternative Flow	NA
Exception Flow	NA
Notes	Character limitation applied.

Use Case # & Title	23 - Approve Infusion Changes
Priority	Nice to have
Description	Pharmacists will have the ability to approve any changes other intake admins create
Actors	Pharmacist System Intake Admin
Triggers	1. Pharmacist views patient profile
Main Flow	1. Pharmacist views patient profile to see pending changes 2. Pharmacist either selects Approve or Reject
Alternative Flow	NA
Exception Flow	NA
Notes	NA

Use Case # & Title	24 - Recover Account (Pharmacist/Intake)
Priority	Nice to have

Description	Pharmacists can follow a procedure to recover their account in the case of forgotten login information
Actors	Pharmacist Intake Support System
Triggers	<ol style="list-style-type: none"> <li>1. Pharmacist clicks on “Forgot My Login” on the login page</li> </ol>
Main Flow	<ol style="list-style-type: none"> <li>1. Pharmacist clicks on “Forgot My Login” on the login page</li> <li>2. Pharmacist enters in identifying information             <ol style="list-style-type: none"> <li>a. Pharmacy Address</li> <li>b. Service Subscription Number</li> </ol> </li> <li>3. Pharmacy Account’s primary email receives an email</li> <li>4. Pharmacist clicks the verification button</li> <li>5. Pharmacist creates a new username / password</li> <li>6. Pharmacist saves the username / password</li> </ol>
Alternative Flow	NA
Exception Flow	2b. Pharmacist enters the wrong address or subscription number and is unable to receive an email
Notes	NA

Use Case # & Title	25 - Use Video Chat
Priority	Great if we can get to it
Description	Patients can access video chat to call for assistance from the Live Help bar.
Actors	Patient Pharmacist Intake Support System
Triggers	<ol style="list-style-type: none"> <li>1. Patient clicks Live Help button</li> </ol>
Main Flow	<ol style="list-style-type: none"> <li>1. Patient clicks Live Help button</li> <li>2. Chat box is pulled up</li> <li>3. Patient can click on the video camera icon to initiate</li> </ol>
Alternative Flow	NA

Exception Flow	NA
Notes	NA

## UI Document

### Personas

#### Persona #1: Oliver Thompson (patient)



**PRONOUNS:** he/him  
**AGE:** 83  
**WORK:** writer (retired)  
**LOCATION:** Los Angeles, CA  
**FAMILY:** married, 1 kid  
**TRAITS:** friendly, laidback, easygoing

## OLIVER THOMPSON

---

**BIO**

Oliver is a retired writer who was just released from the hospital. Part of his treatment requires an infusion, but he has trouble remembering each of the steps and was told different details from the nurse and pharmacist. Oliver struggles to use his smartphone and computer. Sometimes he asks his wife to help him with technology and the infusion treatment. In his free time, Oliver likes to read books and the newspaper.

**GOALS**

- properly and safely give himself the infusion treatment
- administer infusions at the correct time

**FRUSTRATIONS**

- the nurse at the hospital and the pharmacist told him different instructions
- he can't remember all the steps for the infusion

## Persona #2: Julianne Chu (pharmacist)



PRONOUNS: she/her

AGE: 37

WORK: pharmacist

LOCATION: Irvine, CA

FAMILY: married, 2 kids

TRAITS: hardworking,  
athletic

# JULIANNE CHU

---

## BIO

Julianne is a pharmacist who works at a busy pharmacy in Irvine, California. On especially busy days, she fears that she will forget to attend to a patient. She is relatively familiar with technology, as she often uses computers during work. Outside of work, she aims to lead an active and healthy lifestyle with her family. Julianne loves going on bike rides with them on the weekends.

## GOALS

- reduce patient confusion and frustration
- improve workflow

## FRUSTRATIONS

- having too many patients to keep track of at work
- patients are being told contradictory information from hospital staff

## Persona #3: Carol Martin (intake admin/pharmacy technician)



**PRONOUNS:** she/her

**AGE:** 43

**WORK:** pharmacy technician

**LOCATION:** Dallas, TX

**FAMILY:** single

**TRAITS:** outgoing, detail-oriented, organized

# CAROL MARTIN

## BIO

Carol is a pharmacy technician in Dallas, Texas. Carol prioritizes organization and efficiency at work so tasks can be done in a timely manner. However, some of her coworkers are not on the same page as each other, which can lead to delays. Carol is familiar with using smartphones and computers, but she's not an expert. When Carol isn't working, she likes to watch romantic comedies and chat with her friends.

## GOALS

- easily onboard patients who require infusion treatments
- manage all her tasks efficiently

## FRUSTRATIONS

- her coworkers are not on the same page all the time

## **Scenarios**

### Scenario 1

Oliver is an 83 year old patient who is married and has gone through infusions before. He is not great with technology and sometimes has to ask his wife for help. No matter how many times he does it however, he can never remember each of the steps. Adding to his frustration, the nurses and pharmacists give him conflicting instructions about the infusion, making the process very confusing. When his pharmacist now tells him that the instructions will be available via a website, he is a little concerned. While creating an account was not difficult, he still asked his wife to watch and help as he navigated through the site. To his surprise, there were not too many buttons on the screen at once. He could see the time of his next infusion at the top and the tutorial button was right underneath. After clicking on the tutorial button, the instructions were separated into a step by step structure with both wording and video formats. Now that he didn't have to remember what different people told him, Oliver felt more assured about what he was doing. He hopes that the pharmacy continues to use the website for his future infusions.

### Scenario 2

Lily is a 19 year old patient who is about to do her first infusion. Her parents were strict on her as a child so she wasn't able to use computers very often and did not receive a smartphone until she turned 18 and was going away to college. Since she isn't as technologically proficient as her friends, she was worried when her pharmacy told her that she would be using the Innovation Lab website to learn how to do the infusion. Upon opening the app, she had to make an account. The steps to make an account were not bad. It was similar to something she's previously done with email accounts. Seeing that the alarm for her infusion was still 2 hours away, Lily chose to watch a movie to help her relax. Unfortunately, the movie she chose was 3 hours long, and Lily missed the alarm for her infusion. Upon realizing her mistake only as the movie ended, Lily began to panic. Opening up the website on her phone, Lily presses on the help button intending to call the pharmacy. To her surprise however, the help button pulled up an FAQ in addition to the "call pharmacy" button. One of the FAQ questions pointed out to her that missing an infusion by an hour is ok and that she can still follow the instructions normally. With a big sigh of relief, Lily goes back to the home page where she navigates to the tutorial section to proceed with the infusion.

### Scenario 3

Carly is a 49 year old intake admin who is single. She cares a lot about her patients, but gets frustrated when she gets calls asking about questions that are frequently asked and have

quick, simple answers. She understands that they are just trying to make sure they aren't making costly mistakes, but sometimes it gets overwhelming when she has to take all the calls in addition to her other duties. Hearing that her pharmacy is going to try a new website from Innovation Lab worries her because she may get a higher influx of calls as patients will be confused about the new technology they have to work with.

#### Scenario 4

Matt is the 28 year old male developer of the Innovation Lab website who is single. He understands the ins and outs of the site after working on it for over a year. The Med Center Pharmacy, one of the pharmacies that has paid for his website, reached out to Matt via the email they found on the pharmacist's version of the website. There is an upcoming approved infusion that they would like to use in the future so they want Matt to add it into the Innovation Lab website's database. Matt consults some of the medical experts on the Innovation Lab team about the new infusion who agree it would be good to add to the website. Matt replies to the Med Center Pharmacy to form future ways of communication. Matt, the medical experts, and the Med Center Pharmacy work together over the next month to research the infusion, write out the proper instructions, film video demonstrations, and upload the information to the website so that it may be assigned properly to patients in the future.

#### Scenario 5

Barney is a 56 year old male pharmacist who is married and has two kids. Barney has an inconsistent work schedule as a pharmacist. There are days where he has to work night shifts and weeks where he has to stay on weekends. Sometimes his wife complains that he always comes home tired and doesn't spend enough time with her or the kids. He became a pharmacist because he wanted to help people. Ever since the Innovation Lab website has been implemented at the pharmacy, he's appreciated that people can go home with a set of instructions that has a video to go along with it. While Barney gets calls occasionally from patients because intake admins are not available, he is glad the website the pharmacy paid for is being used. Moving forward, he is hoping that the Innovation Lab website keeps the patients safer while also making their overall system more efficient and taking some of the workload off of himself.

### **Comparative Analysis of Interfaces**

#### **Context**

Common issues that arise after patients are sent home with infusion instructions include:

- Patients being too tired or sick to interpret written paper instructions

- Patients forgetting to administer their infusions at the necessary times
- Patients having questions or urgent issues to address with little solution for help

As a result of the above scenarios, the dilemma of patients' readmissions into hospitals is unfortunately common. The application being developed on behalf of Innovation Lab aims to ease the process of administering infusions for post-care patients and their caregivers. The application focuses on the following main functionalities:

- Step-by-step video follow-along tutorials for infusion administration
- Alarm system to notify patients of infusion times
- Frequently asked questions section, as well as live video-chat to allow pharmacists to aid patients remotely

There are currently no programs or applications with the exact same conceptual model as the application we are developing, and so comparative analysis will use applications with functionalities similar to the ones listed.

### **Medisafe**

- Purpose: Alert users of medications that they need to take at certain times
- Pros:
  - Easy to understand Dashboard
  - Can choose selected icons for pills for more customization/recognition
- Cons:
  - When not on current week, there is a poor indicator what is the current day; An button with “Today” and an arrow indicating to the left or right is not intuitive and might confuse users
  - Collapsing alarm information has no purpose
  - Alarms are not in order
- Takeaways from this application
  - We will aim to provide a “today” button that will help navigate users back to the current day in Alarms
  - Ensure that alarms are presented in timely order

### **Caring Village**

- Purpose: Healthcare application designed for caregivers to help manage all activities
- Pros:
  - Settings page is scrollable; more spacious and intuitive
  - Very user friendly icons and images used
  - Intuitive design
- Cons:
  - For Calendar system, set events are hard to interpret; starting and ending times are not intuitive to understand and icon on the right is misleading (looks like comments)

- Dates with events are not intuitive (they are colored grey but there is no indication to notify the user that days with events are grey)
- Takeaways from this application
  - Ensure that text descriptions follow icons and other information to avoid ambiguity

### **MyTherapy**

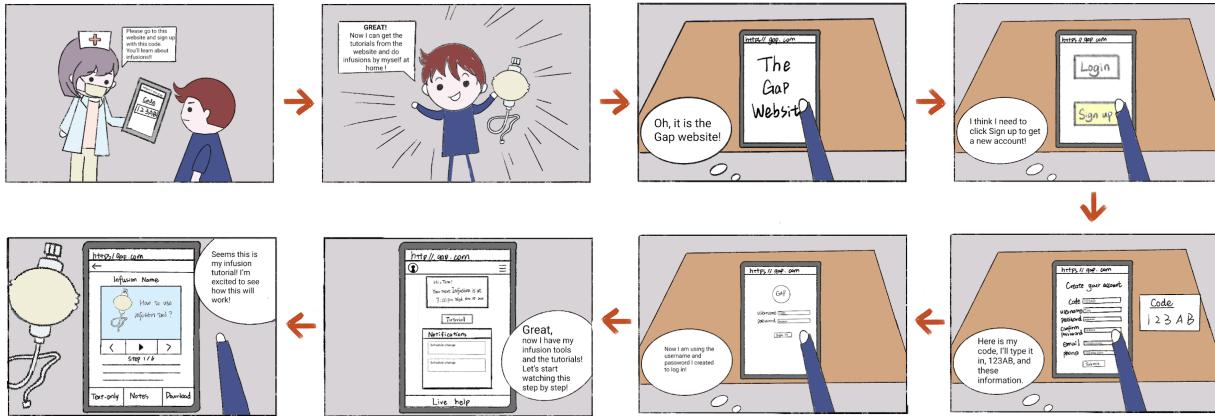
- Purpose - help user create alarms for medications
- Pros:
  - Very unique design for calendar; shows alarms that have been checked off
  - Can choose to view as chart or list
  - Simplified views to help patients
- Cons:
  - Dashboard shows how many “tasks” you’ve completed but does not allow you to expand to see which “tasks”
- Takeaway
  - Will try to keep things simplified for users

### **Overview**

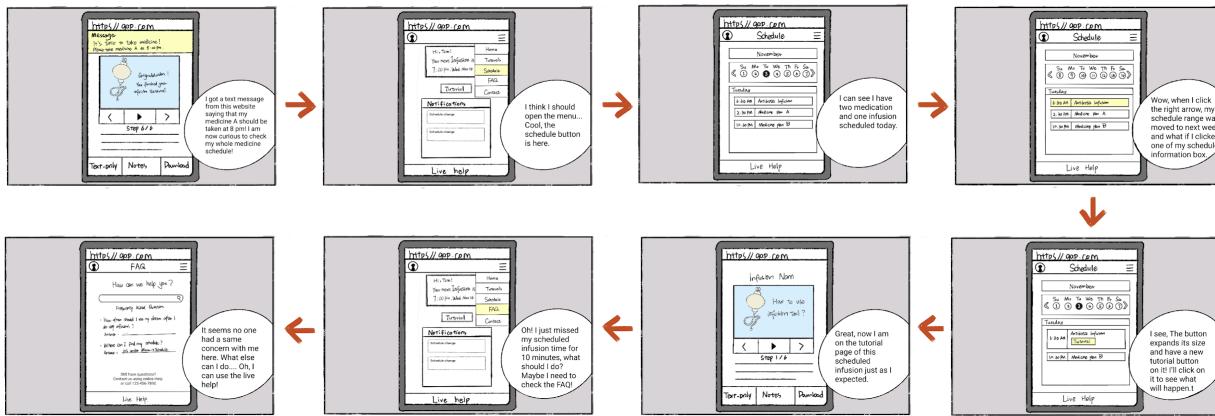
- An important aspect that is seen in all three of these is used-friendly, accessible design. Most likely because the structure of the applications kept in mind the different types of challenges users might encounter.
- Flaws from these designs included ambiguity in icons or colors, which we will try to keep note of when rendering out the final designs for our project.

## Storyboards

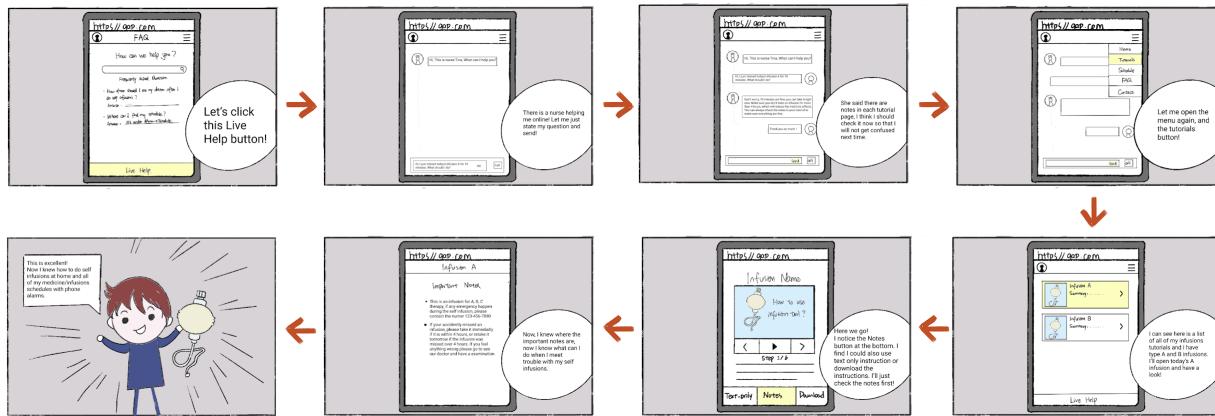
### User Sign Up and Tutorial Tab Accessing



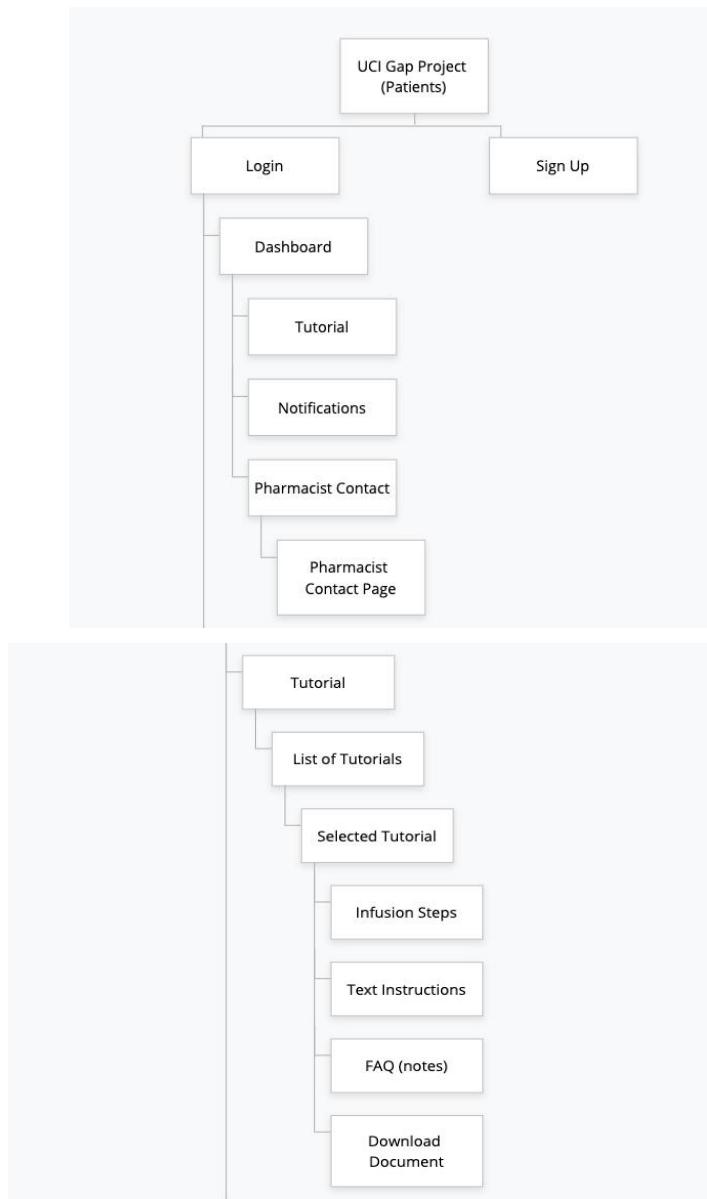
### Schedule Page and FAQ Page Accessing

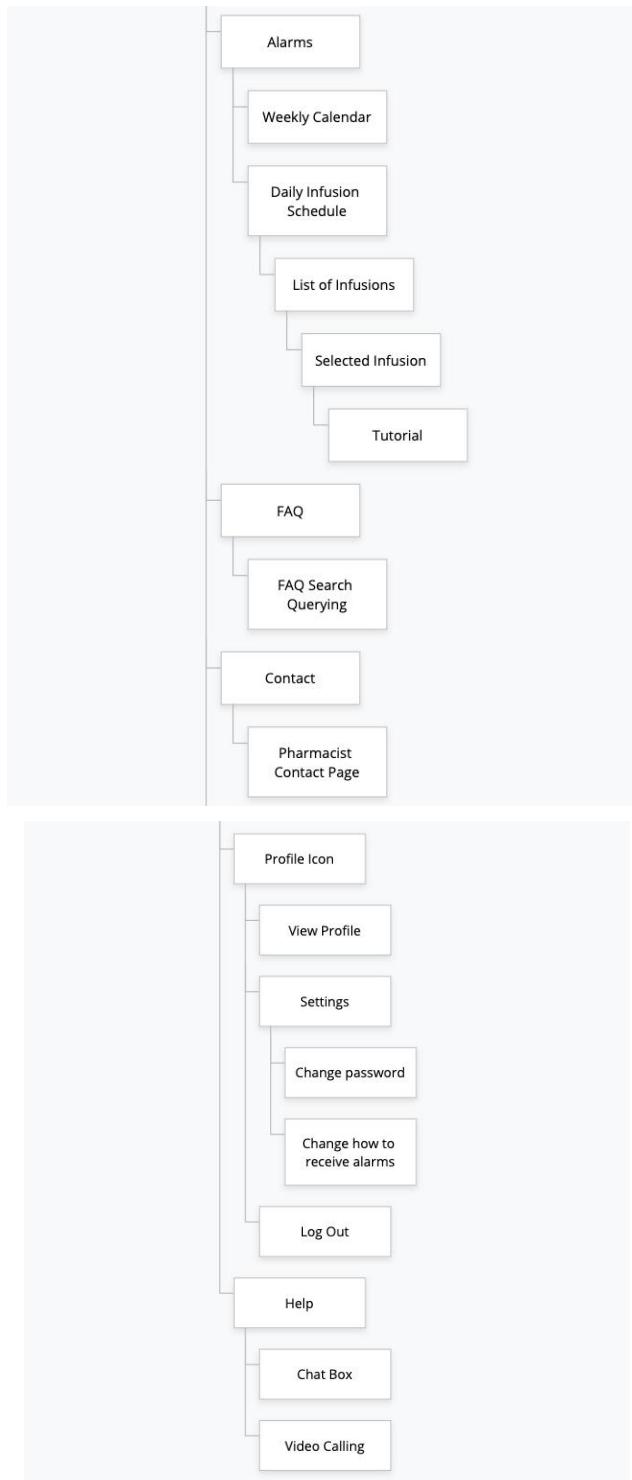


### Live Help Function and Tutorials Page Accessing



## Generalized Transition Network ([link](#))

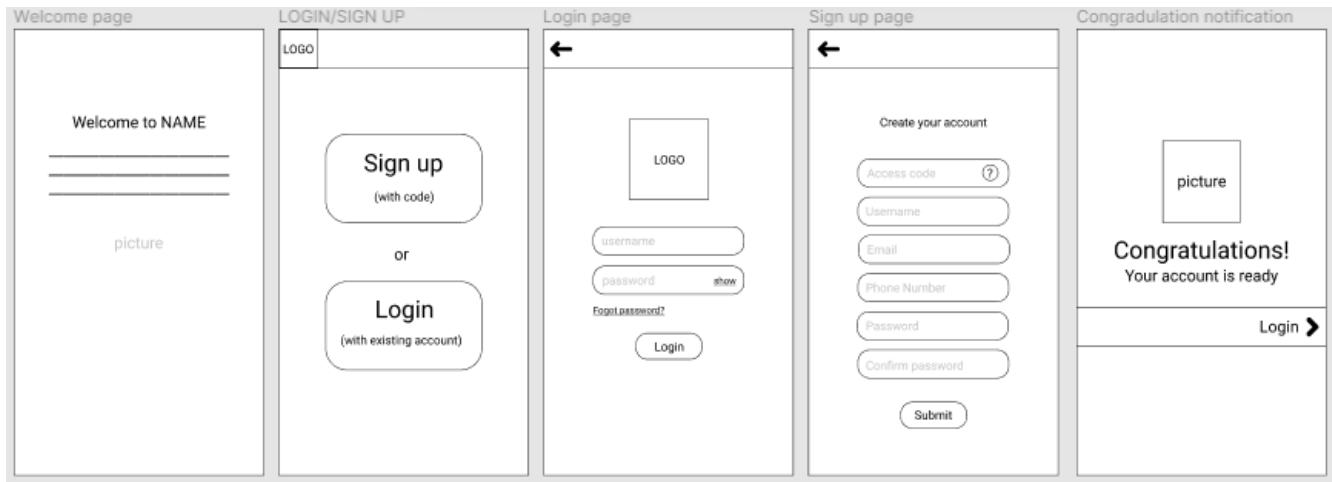




# Wireframes

## Patient Mobile View

### Patient: Login and Onboarding (Mobile)



### Patient: Dashboard, Nav, Chat, Profile, and Settings (Mobile)



## Patient: Tutorials, Text Instructions, Tutorial Notes (Mobile)

**Tutorial List page:**

- Header: Tutorials
- Content: Two infusion tutorial cards (each with a thumbnail, name, summary, and duration).
- Footer: Live Help button.

**Tutorial Page:**

- Header: Infusion NAME
- Content: Step 1 / 6, Description input field, navigation arrows, and three buttons at the bottom: TEXT-ONLY, NOTES, DOWNLOAD.

**Tutorial Text Instruction:**

- Header: Infusion NAME
- Content: Instruction List (5 numbered lines) and "What if I did not \_\_\_\_?" sections.

**Tutorial Notes:**

- Header: Infusion NAME
- Content: Notes section with "What if I did not \_\_\_\_?" sections.
- Footer: More FAQ link.

## Patient: Alarm and Calendar (Mobile)

**Alarms Page (Tuesday):**

- Header: Alarms
- Content: Month view (Su-Mo-Tu-We-Th-Fr-Sa) showing day 3 highlighted. Below is a list of three antibiotic infusions scheduled for Tuesday.
- Footer: Live Help button.

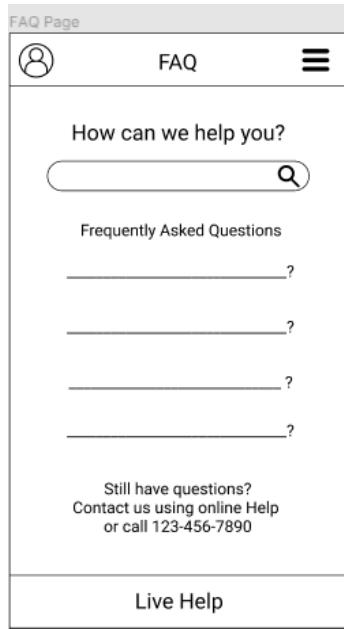
**Alarms Page (Tuesday):**

- Header: Alarms
- Content: Month view (Su-Mo-Tu-We-Th-Fr-Sa) showing day 10 highlighted. Below is a list of three antibiotic infusions scheduled for Tuesday.
- Footer: Live Help button.

**Alarms Page (Wednesday):**

- Header: Alarms
- Content: Month view (Su-Mo-Tu-We-Th-Fr-Sa) showing day 4 highlighted. Below is a list of two antibiotic infusions scheduled for Wednesday, with one entry labeled "Tutorial".
- Footer: Live Help button.

## Patient: Frequently Asked Questions (Mobile)



The wireframe shows a mobile FAQ page titled "FAQ Page". It features a header with a user icon, the word "FAQ", and a menu icon. Below the header is a search bar with a magnifying glass icon. A section titled "Frequently Asked Questions" contains four horizontal lines, each ending in a question mark. At the bottom of the page, there is a message: "Still have questions? Contact us using online Help or call 123-456-7890". A "Live Help" button is located at the very bottom.

## Patient: Pharmacy Contact Page (Mobile)



The wireframe shows a mobile contact information page titled "Contact Information Page". It includes a header with a user icon and a menu icon. The page displays several sections: "About us", "Administration photo", "Name" (with an input field), "Location" (with an input field), and "Contact Information" (which lists "Phone: 123-456-7890" and "Email: 123456@com"). A "Live Help" button is located at the bottom.

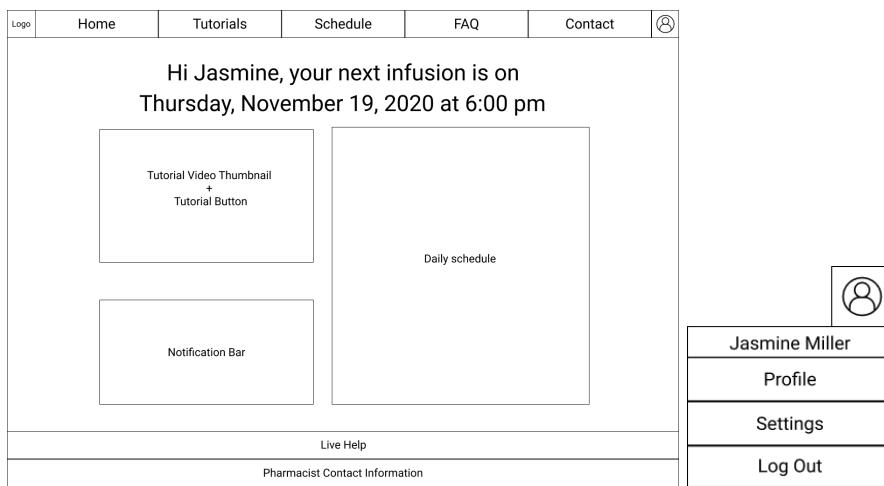
## Patient Desktop View

### Patient: Login Pages

The image displays four wireframe screens for a patient login interface:

- Top Left Screen:** Shows a "Welcome to NAME" message above a horizontal line, followed by a "picture" placeholder.
- Top Right Screen:** Features a "LOGO" placeholder at the top. It contains two rounded rectangular buttons: one labeled "Sign up (with code)" and another labeled "Login (with existing account)". A small "or" text is positioned between them.
- Middle Left Screen:** Displays a "LOGO" placeholder. Below it are two input fields: "username" and "password". A "Forgot password?" link is located next to the password field. A "Login" button is at the bottom.
- Middle Right Screen:** Shows a "Create your account" heading. It includes six input fields: "Access code" (with a help icon), "Username", "Email", "Phone Number", "Password", and "Confirm password". A "Submit" button is at the bottom right.
- Bottom Screen:** Shows a "picture" placeholder. Below it is a "Congratulations! Your account is ready" message. A "Login" button is at the bottom.

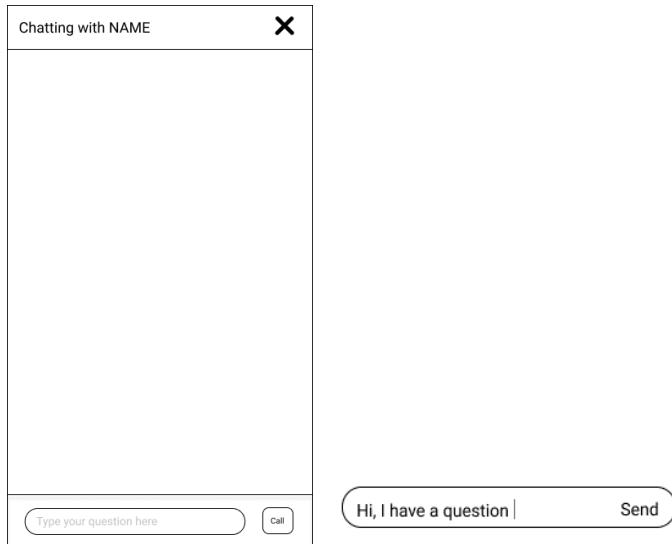
## Patient: Home Page



A wireframe mockup of a patient's home page. At the top, there is a navigation bar with links for Logo, Home, Tutorials, Schedule, FAQ, Contact, and a user icon. Below the navigation bar, a message says "Hi Jasmine, your next infusion is on Thursday, November 19, 2020 at 6:00 pm". To the left, there is a box labeled "Tutorial Video Thumbnail + Tutorial Button". Below that is a "Notification Bar". To the right, there is a "Daily schedule" area. On the far right, there is a vertical sidebar for the user "Jasmine Miller" with options for Profile, Settings, and Log Out.

Logo	Home	Tutorials	Schedule	FAQ	Contact	ouser
Hi Jasmine, your next infusion is on Thursday, November 19, 2020 at 6:00 pm						
Tutorial Video Thumbnail + Tutorial Button		Daily schedule				 Jasmine Miller Profile Settings Log Out
Notification Bar						
Live Help						
Pharmacist Contact Information						

## Patient: Help Chat



A wireframe mockup of a help chat interface. It shows a header "Chatting with NAME" with a close button "X". The main area is a large empty box. At the bottom, there is a text input field with placeholder text "Type your question here", a "Call" button, and a message bubble containing "Hi, I have a question | Send".

Chatting with NAME	X
Type your question here	Call
Hi, I have a question   Send	

## Patient: Profile Page

Logo	Home	Tutorials	Schedule	FAQ	Contact	
<p>Jasmine Miller</p> <p>Pharmacy: (Pharmacy Name) Phone Number: (Phone Number)</p> <p>Summary of Infusion</p>						

## Patient: Settings Page

Logo	Home	Tutorials	Schedule	FAQ	Contact	
<p><b>Settings</b></p> <p>Phone Number: <small>Standard Text Rate Message Apply</small></p> <p><input type="text"/> Phone Number: (123)-456-7891 </p> <p>Email: <input type="text"/> email: email@innovationlab.net </p> <p>Alerts: <small>Select how you would like to be notified</small></p> <p><input type="checkbox"/> Text message: (123)-456-7891</p> <p><input type="checkbox"/> Email message: email@innovationlab.net</p> <p><input type="button" value="Save"/></p> <p>Password Change:</p> <p><input type="text"/> Current Password: *****</p> <p><input type="text"/> New Password: new password</p> <p><input type="text"/> Retype Password: new password</p> <p><input type="button" value="Save"/></p>						

## Patient: Tutorial Pages

Logo	Home	Tutorials	Schedule	FAQ	Contact	
<p>Infusion Name Summary (purpose etc) _____ Duration : 20 min</p> <p>Infusion Name Summary (purpose etc) _____ Duration : 20 min</p>			<p>&lt;          &gt;</p> <p>Step 1 / 6</p> <p>Description: _____</p> <p><input type="button" value="TEXT-TO-SPEECH"/> <input type="button" value="DOCUMENT DOWNLOAD"/></p> <p>Pharmacist Notes   Step List   Infusion Notes</p> <p>What if I did not _____? _____ _____</p>			
<p>Live Help</p>						

## Patient: Schedule Page

Logo Home Tutorials Schedule FAQ Contact 

November						
 Sunday 11/8	Monday 11/9	Tuesday 11/10	Wednesday 11/11	Thursday 11/12	Friday 11/13	
Thursday						
6:30 AM	Antibiotic Infusion					
2:30 PM	Antibiotic Infusion					
10:30 PM	Antibiotic Infusion					

Live Help

## Patient: FAQ Page

Logo Home Tutorials Schedule FAQ Contact 

How can we help you?



Frequently Asked Questions

\_\_\_\_\_?

\_\_\_\_\_?

\_\_\_\_\_?

\_\_\_\_\_?

Still have questions?  
Contact us using online Help  
or call 123-456-7890

Live Help

## Patient: Contact Page

Logo Home Tutorials Schedule FAQ Contact 

About us

Administration photo

Name \_\_\_\_\_

Location \_\_\_\_\_

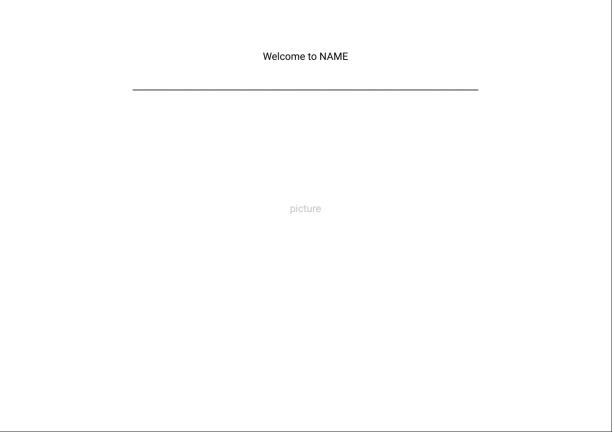
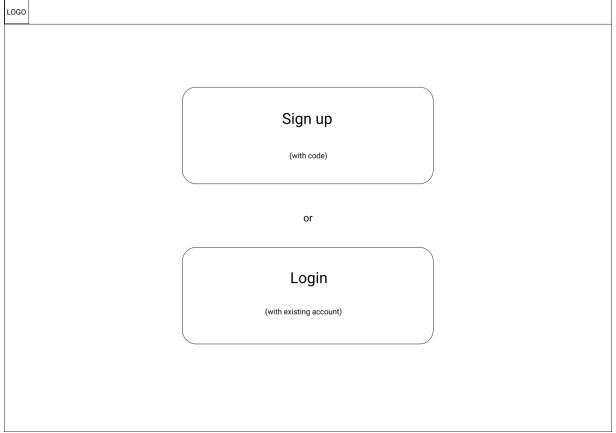
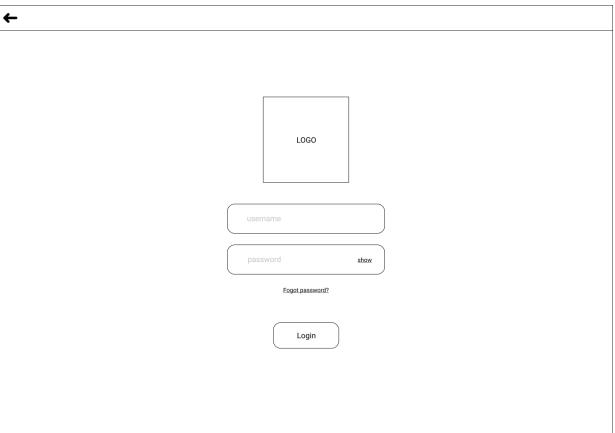
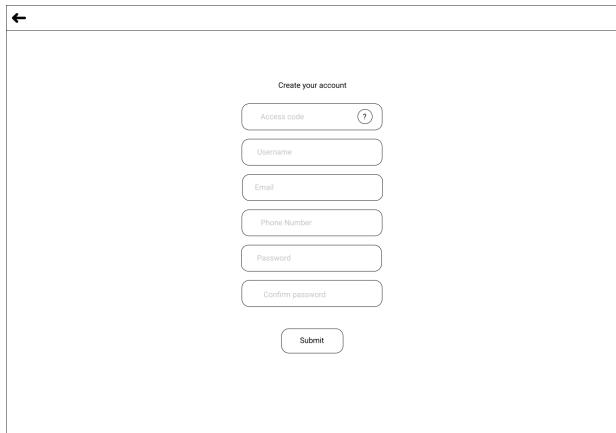
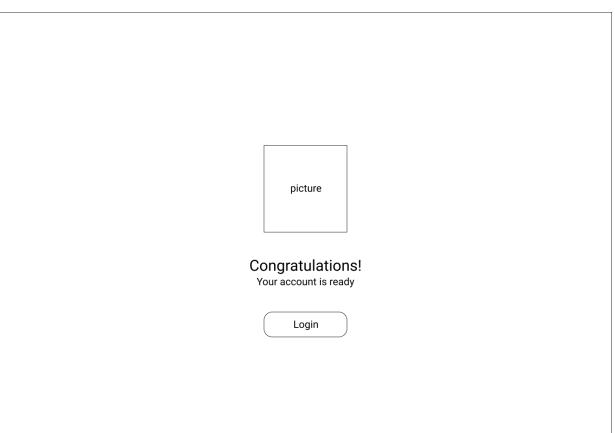
Contact Information

Phone: 123-456-7890  
Email: 123456@com

Live Help

# Pharmacist View (Note: Subject to change as we wait for more information)

## Patient: Login Pages

 <p>Welcome to NAME</p> <p>picture</p> <p>LOGO</p>	 <p>LOGO</p> <p>Sign up (with code)</p> <p>or</p> <p>Login (with existing account)</p>
 <p>←</p> <p>username</p> <p>password <small>Show</small></p> <p><small>Forgot password?</small></p> <p>Login</p> <p>LOGO</p>	 <p>←</p> <p>Create your account</p> <p>Access code <small>(? )</small></p> <p>Username</p> <p>Email</p> <p>Phone Number</p> <p>Password</p> <p>Confirm password</p> <p>Submit</p>
 <p>picture</p> <p>Congratulations! Your account is ready</p> <p>Login</p>	

## Patient: Home Page

Logo    Home    Tutorials    Schedule    FAQ    Contact    

Hi Jasmine,  
you have 52 patients to check up on!

Notification Bar

Patient	ID#	Infusion Types	Last Check Up	Intake Admin

Live Help

Pharmacist Contact Information

 Jasmine Miller  
Profile  
Settings  
Log Out

## Patient: Help Chat

Chatting with NAME 

Type your question here 

Hi, I have a question | 

## Patient: Profile Page

Logo	Home	Tutorials	Schedule	FAQ	Contact	
<p>Jasmine Miller</p> <p>Pharmacy: (Pharmacy Name) Phone Number: (Phone Number)</p> <p>Summary of Infusion</p>						

## Patient: Settings Page

Logo	Home	Tutorials	Schedule	FAQ	Contact	
<p><b>Settings</b></p> <p>Phone Number: <small>Standard Text Rate Message Apply</small></p> <p>Phone Number: (123)-456-7891 </p> <p>Email: <small>email: email@innovationlab.net</small></p> <p>Alerts: <small>Select how you would like to be notified</small></p> <p>Text message: <input type="checkbox"/> (123)-456-7891</p> <p>email message: <input type="checkbox"/> email@innovationlab.net</p> <p></p> <p>Password Change:</p> <p>Current Password: *****</p> <p>New Password: new password</p> <p>Retype Password: new password</p> <p></p>						

## Pharmacist: Tutorial Pages

<p></p> <p><b>Tutorials</b></p> <p>Live Help</p>	<p>Logo</p> <p>Home</p> <p>Tutorials</p> <p>Schedule</p> <p>FAQ</p> <p>Contact</p> <p></p> <p>Pharmacist Notes Step List Infusion Notes</p> <p>Add Note Edit Note Delete Note</p> <p>1. Lorem ipsum dolor sit amet, mazim nonumy eos at, no nemore virtute vis, te noluisse aliquando pro.</p> <p>2. Feugait noluisse abhorreant at sea. Novum impedit salutandi eum cu.</p> <p>3. At phaeidrum indoctum qui, eum ignota invenire ad.</p> <p>&lt;          &gt;</p> <p>Step 1 / 6</p> <p>Description: _____</p> <p>TEXT-TO-SPEECH DOCUMENT DOWNLOAD</p>
--	---

## Pharmacist: Schedule Page

Logo Home Tutorials Schedule FAQ Contact 

November						
 Sunday 11/8	Monday 11/9	Tuesday 11/10	Wednesday 11/11	Thursday 11/12	Friday 11/13	
Thursday						
6:30 AM	Antibiotic Infusion					
2:30 PM	Antibiotic Infusion					
10:30 PM	Antibiotic Infusion					

Live Help

## Pharmacist: FAQ Page

Logo Home Tutorials Schedule FAQ Contact 

How can we help you?



Frequently Asked Questions

\_\_\_\_\_?

\_\_\_\_\_?

\_\_\_\_\_?

\_\_\_\_\_?

Still have questions?  
Contact us using online Help  
or call 123-456-7890

Live Help

## Pharmacist: Contact Page

Logo Home Tutorials Schedule FAQ Contact 

About us

Administration photo

Name \_\_\_\_\_

Location \_\_\_\_\_

Contact Information

Phone: 123-456-7890  
Email: 123456@com

Live Help



## **Usability, Functionality, and Satisfaction Goals**

### Usability Goals

- Effectiveness

Effectiveness means whether users can achieve their goals in a high degree of accuracy. We deem that this should be the primary concern of our UI design; because the majority of our target user group are elders who might not be very familiar with digital devices.

In our UI design, we provide a lot of support texts to users. For example, fixing the phone number and email address fields so that they only accept correct forms of phone number and email address, which could reduce data entry errors and users can reach their goals correctly.

- Ease of use

Ease of use should be the second highest priority of the UI design. We want our users to use this system on a regular basis, hence it is also essential to make the design easy to use and learn.

To accomplish this goal, we use a very common web/phone App layout such as Hamburger Button on the phone App, static top-navigation bar, etc. so that it would match their existing mental models and would not feel overwhelming about design.

- Error Tolerance

We comprehend that sometimes it is difficult to adapt a new system, especially our target user group might not be so familiar with digital devices.

In order to prevent errors, we have decided to use two different URLs (interface) for medical professional users and patient users; so they are unlikely to get perplexed by two login options if they are first time users. Moreover, the static top-navigation bar/Hamburger button always gives users an option to return to the previous page or back to home page.

- Efficiency

Efficiency is about how fast can the user get their goals accomplished.

We carefully examine the number of steps for users to reach specific goals and grouped proximal functions. In addition, we put the core features, such as imminent infusion date, notifications, view tutorials, etc. at the home page with noticeable buttons; so that it would not take a long time for users to obtain the information/resources they need.

### Functionality Goals

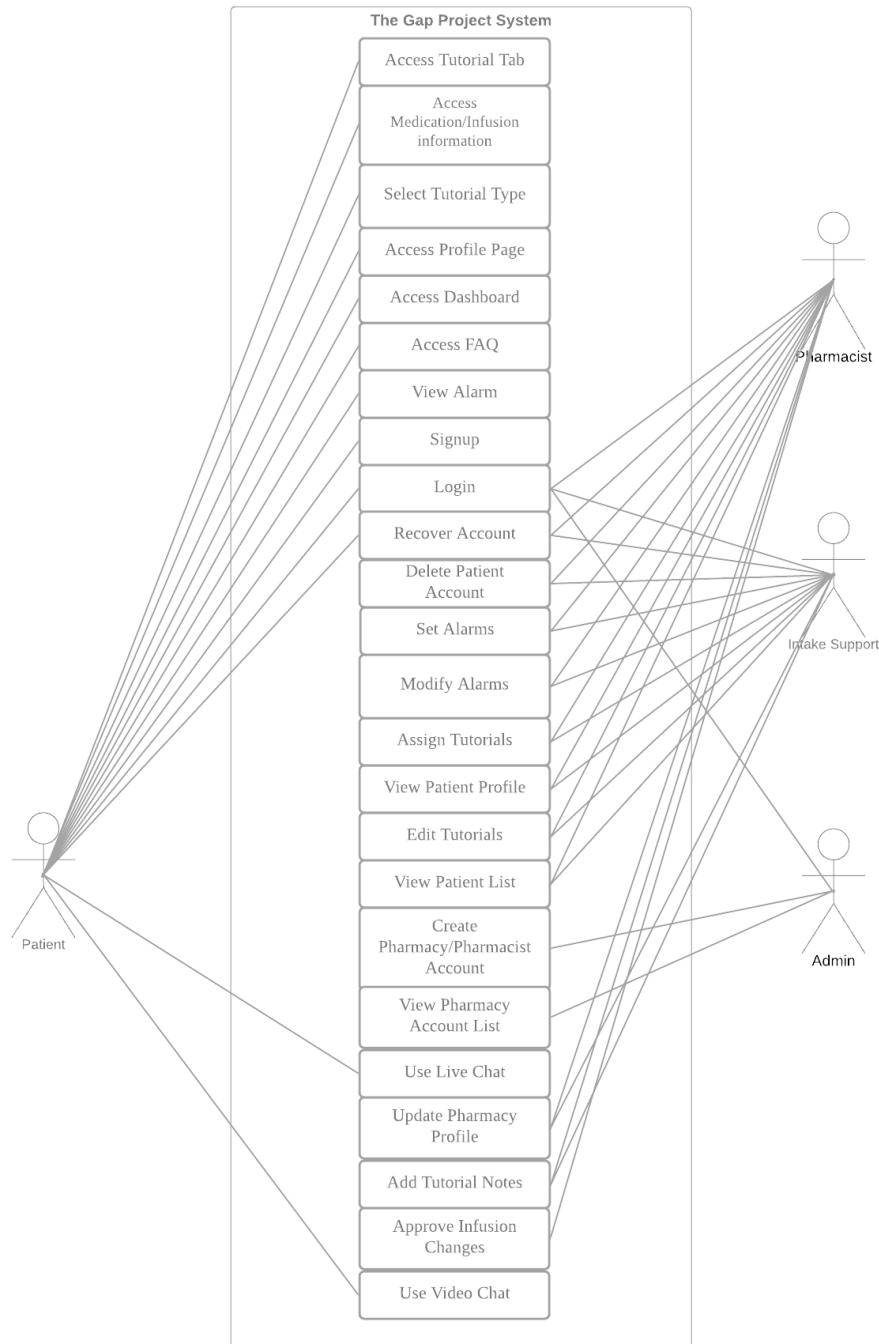
- Integrate medical resources, such as videos and articles that patients can read through and find what they need.
- Digitized, step-by-step, video instructions for medical treatments.
- Customized patient tutorials by medical professionals .
- A notification system that reminds patients of major events.
- Different account types for patients and medical professionals.
- Ability to contact medical professionals 24/7.

### Satisfaction Goals

- Easily accessible for patients and medical professionals users.
- All error and error actions can be modified by returning.
- Feedback on success and failure after operation(s) does not disturb the user
- All types of users can use the system easily with some proper instructions.
- Careful UI/UX design that prevents problems from occurring in the first place.
- Users want to use this system on a regular basis.

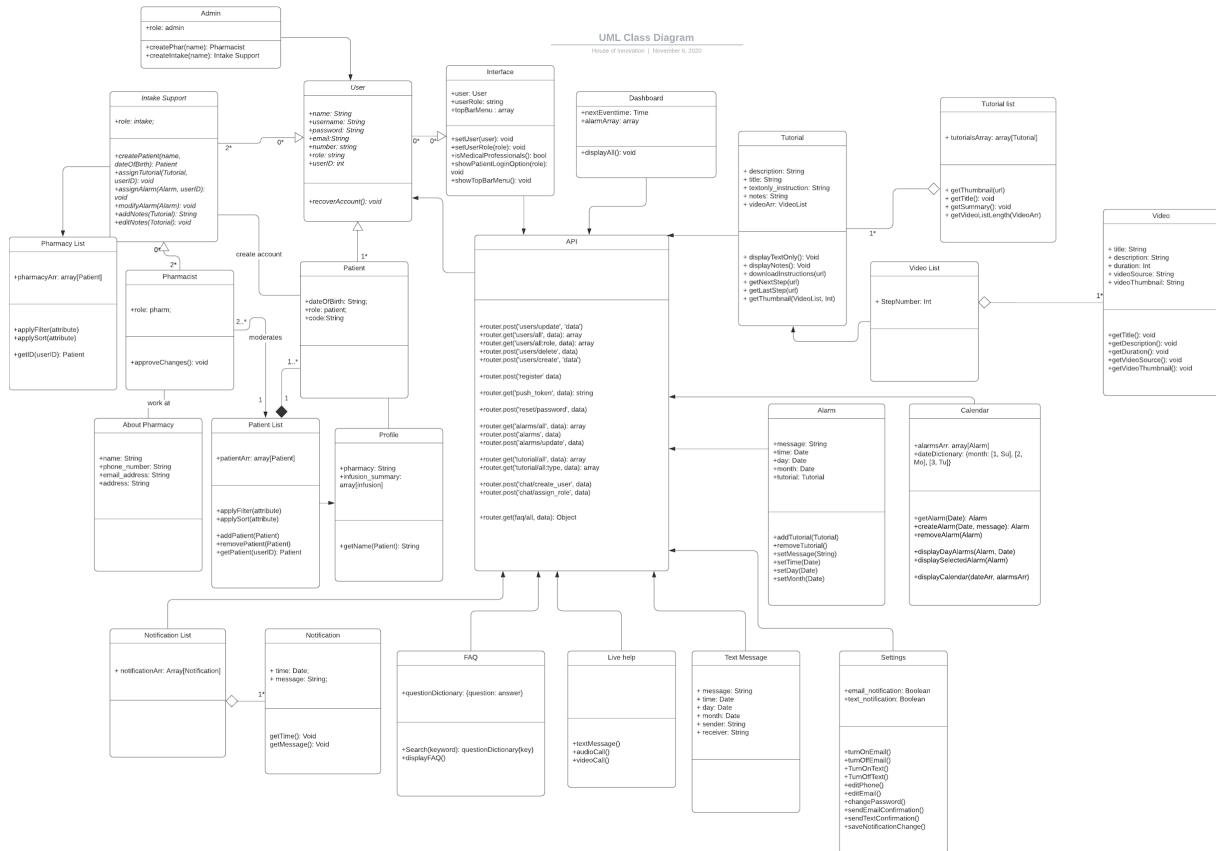
# Software Design Document

## Use-Case Diagram & Summary of Features



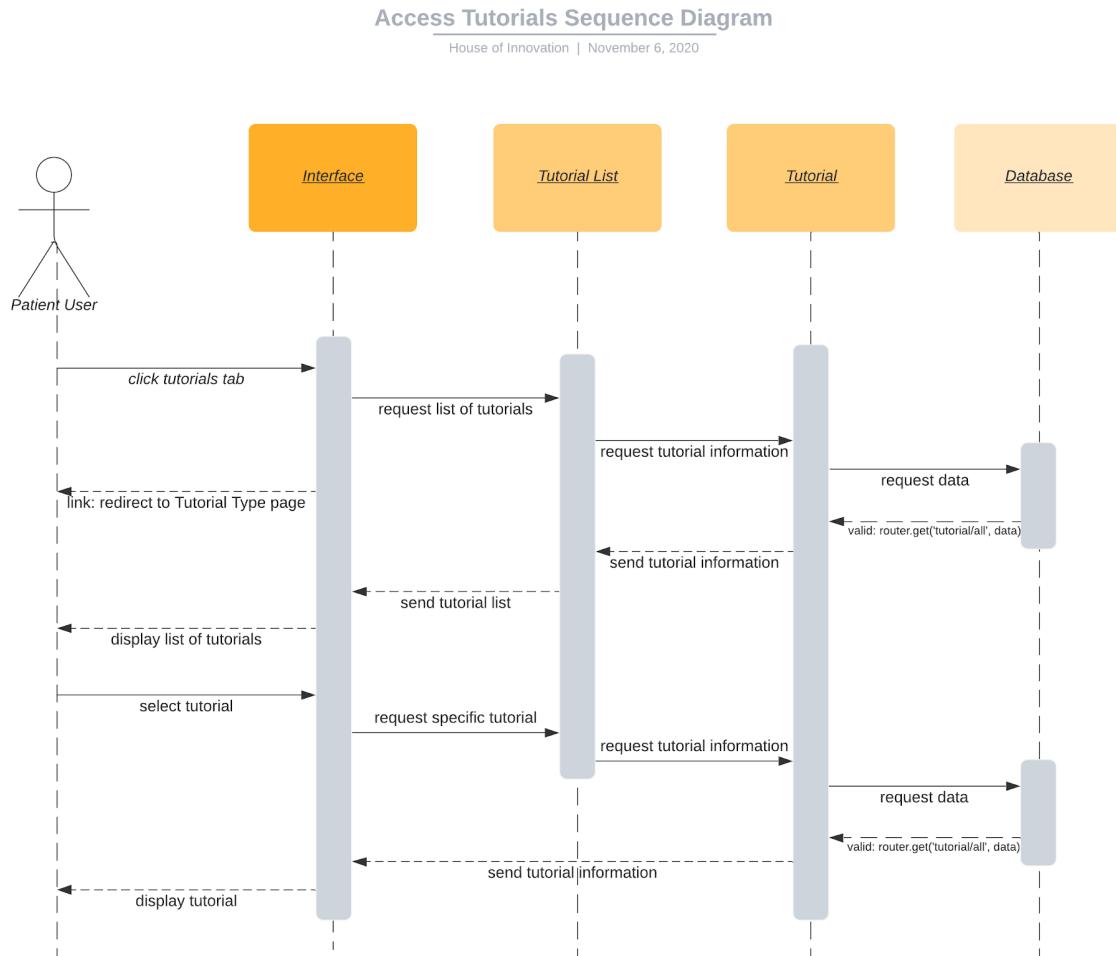
# UML Diagrams

## App UML Diagram ([link](#))



# Important Sequence Diagrams Use Cases

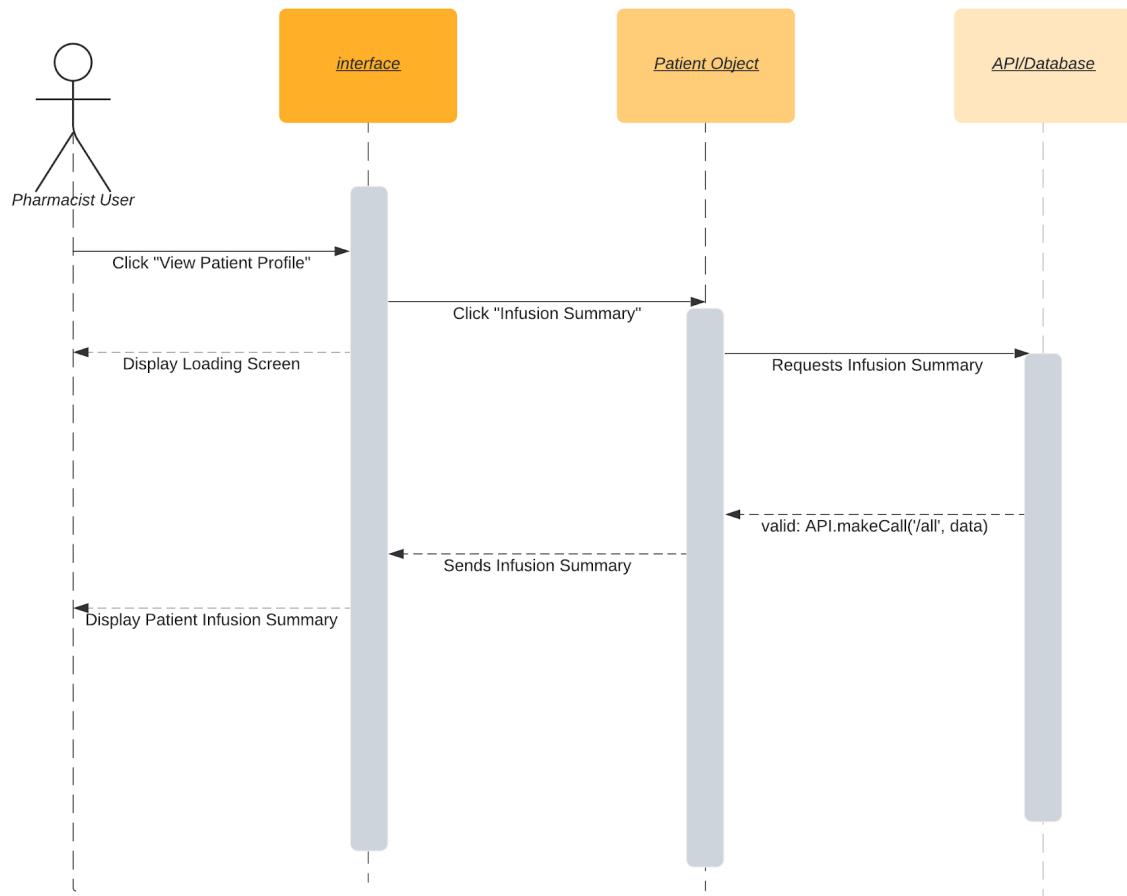
## 1. Access Tutorial



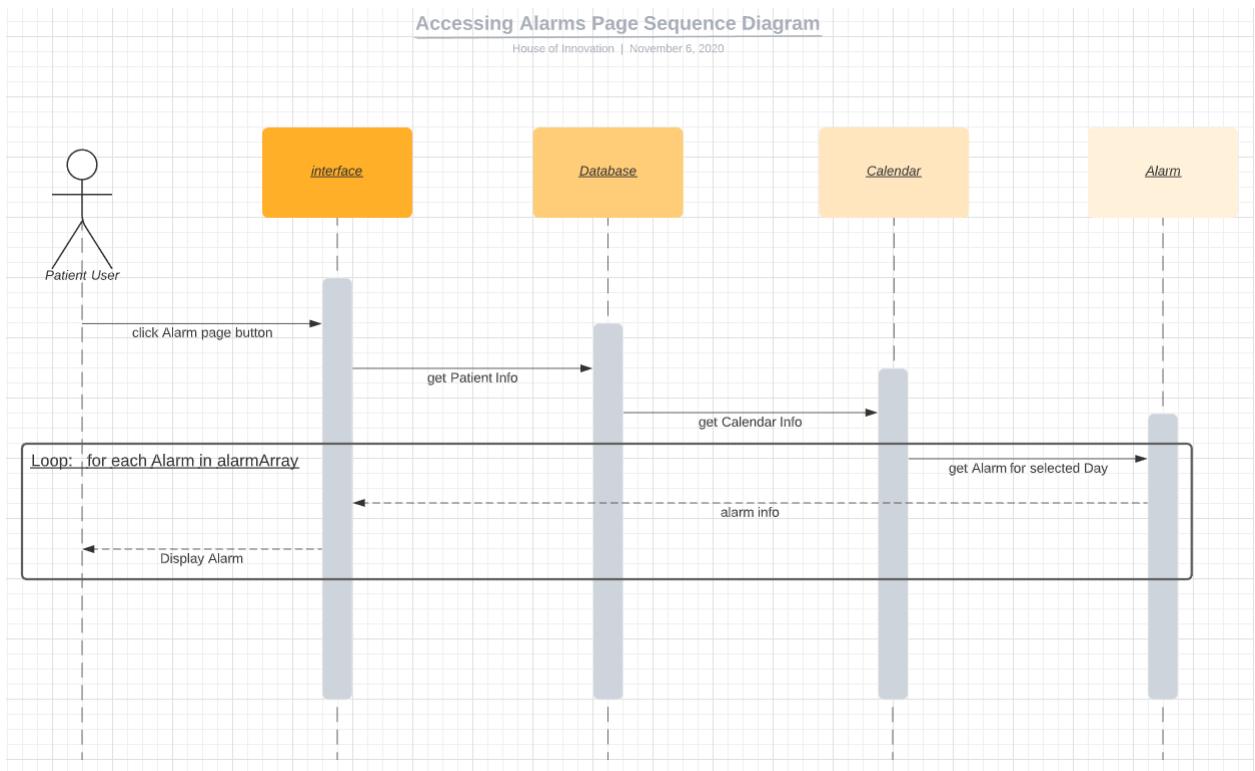
## 2. Access Medication/Infusion Information

**Access Infusion Information Sequence Diagram**

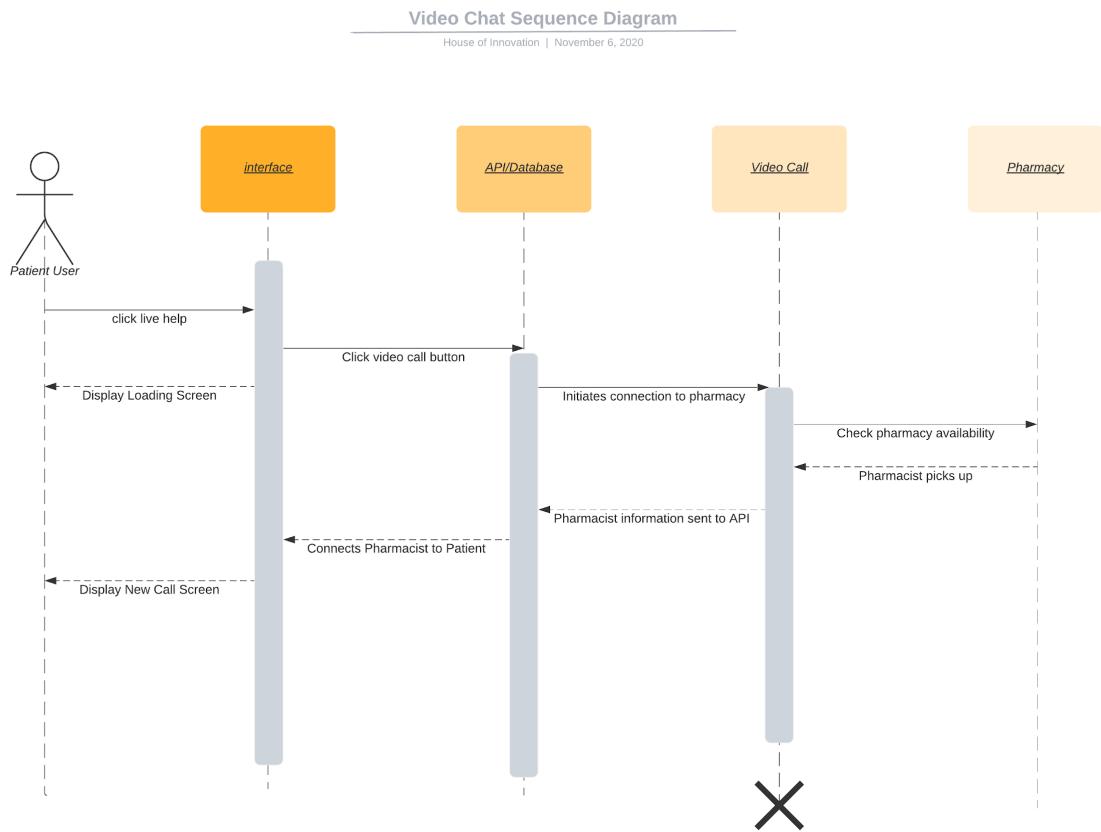
House of Innovation | November 6, 2020



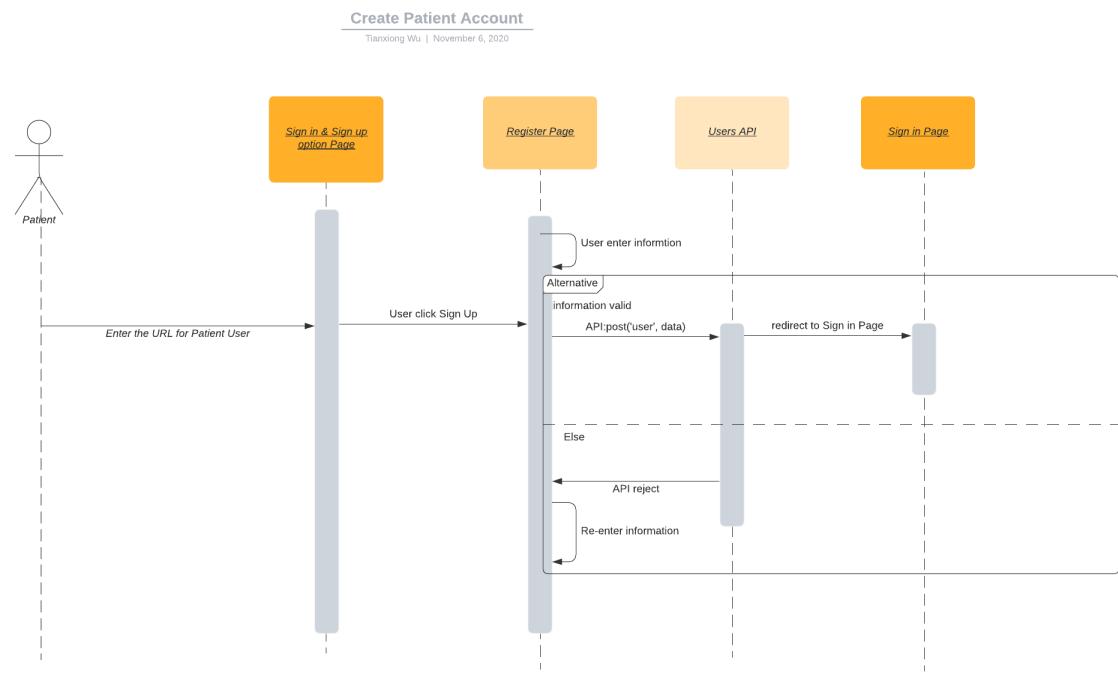
### 3. View and Receive Alarm



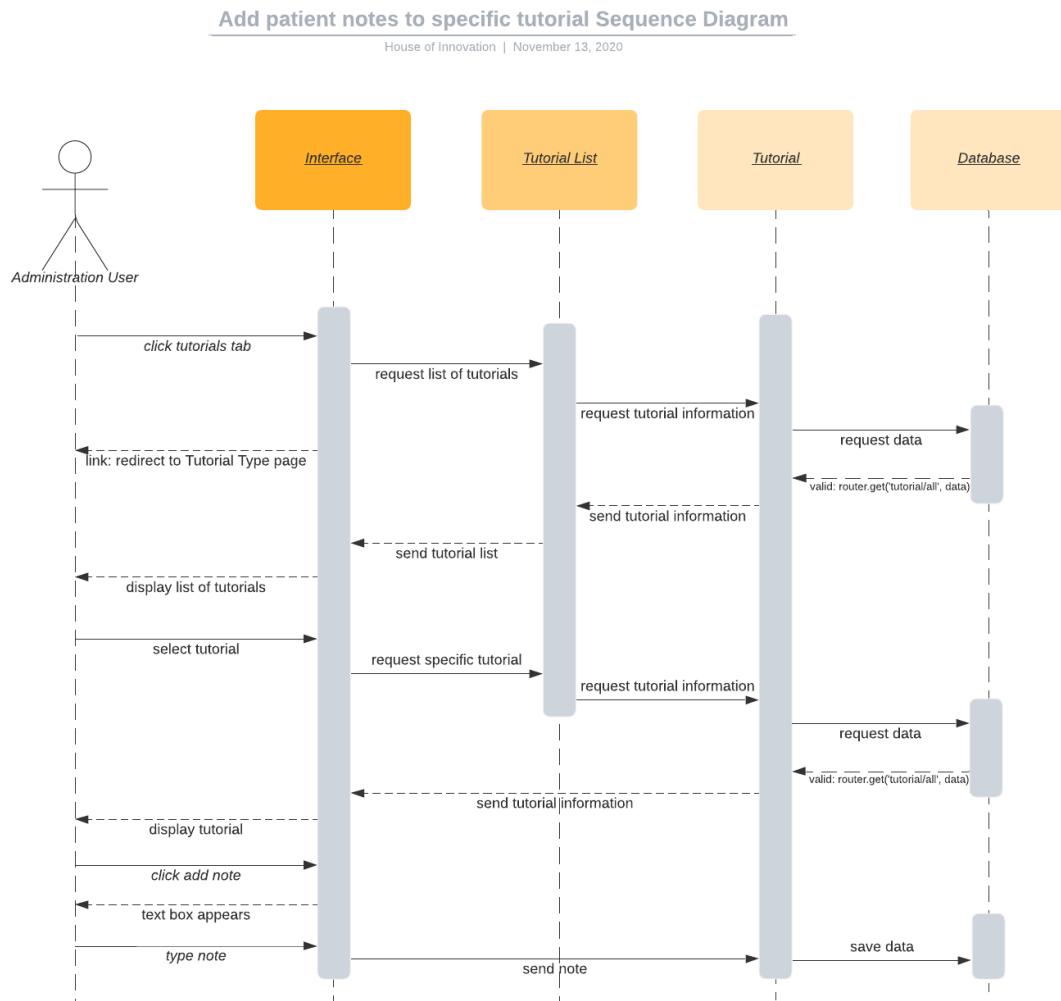
## 4. Video Chat



## 5. Create Patient Account

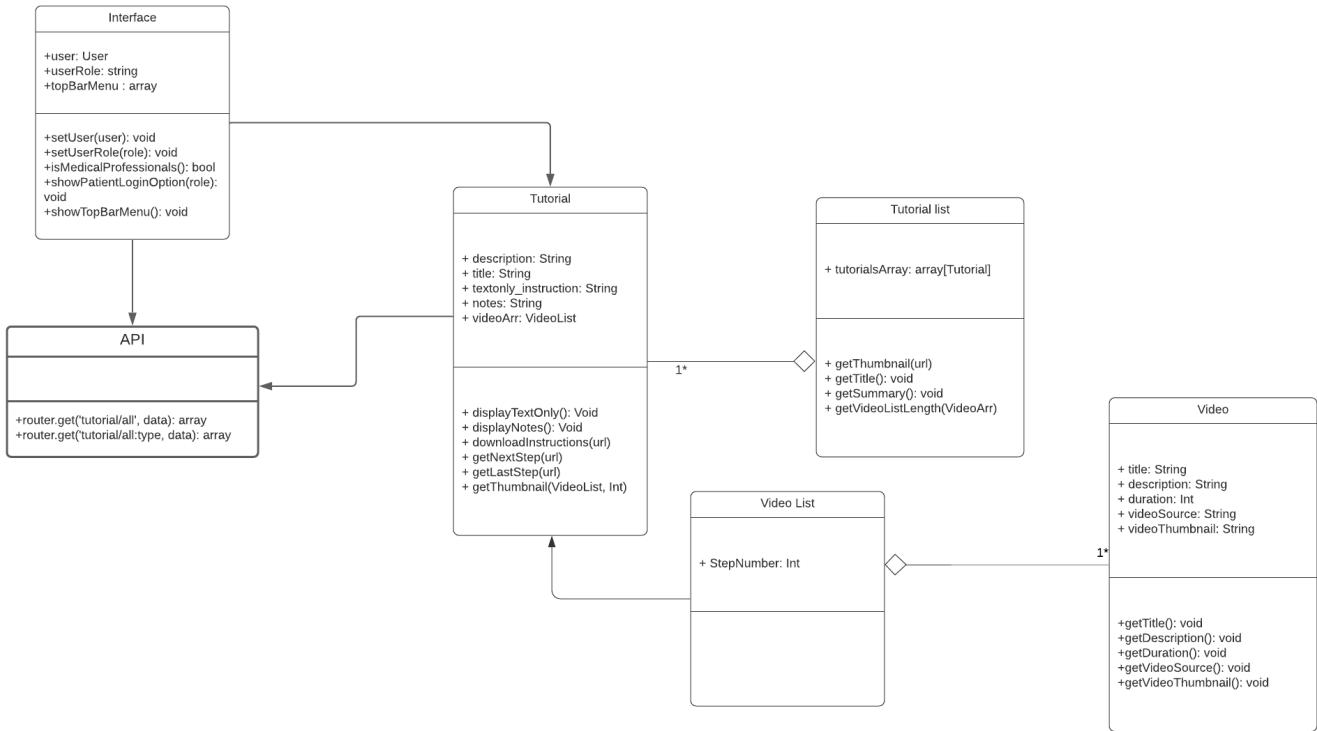


## 6. Add Patient Notes to Specific Tutorial



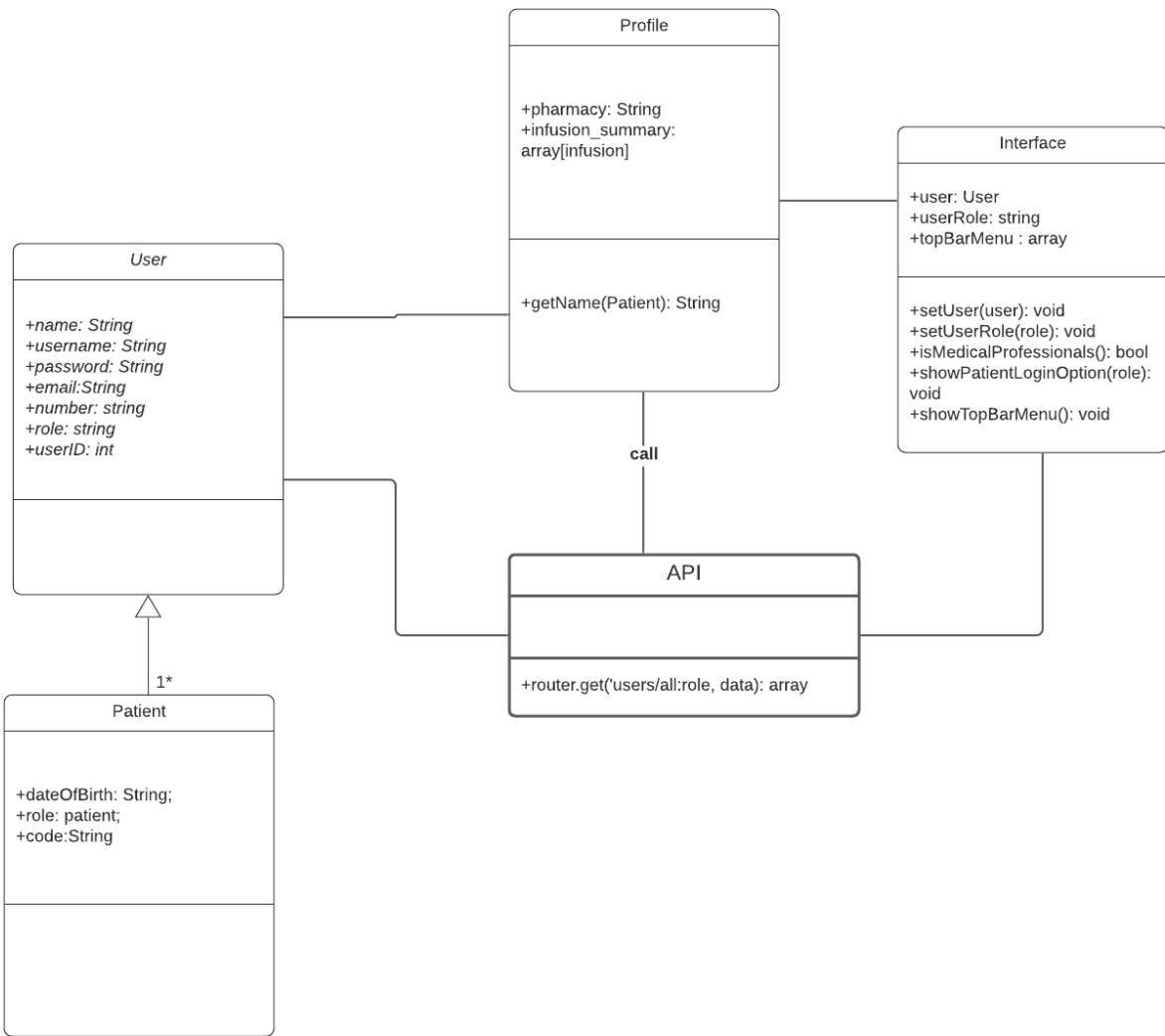
## UML Class Diagrams

UC #1 Access Tutorial Tab, UC #2 Access Infusion Information, UC #3 Select Tutorial Type ([link](#))



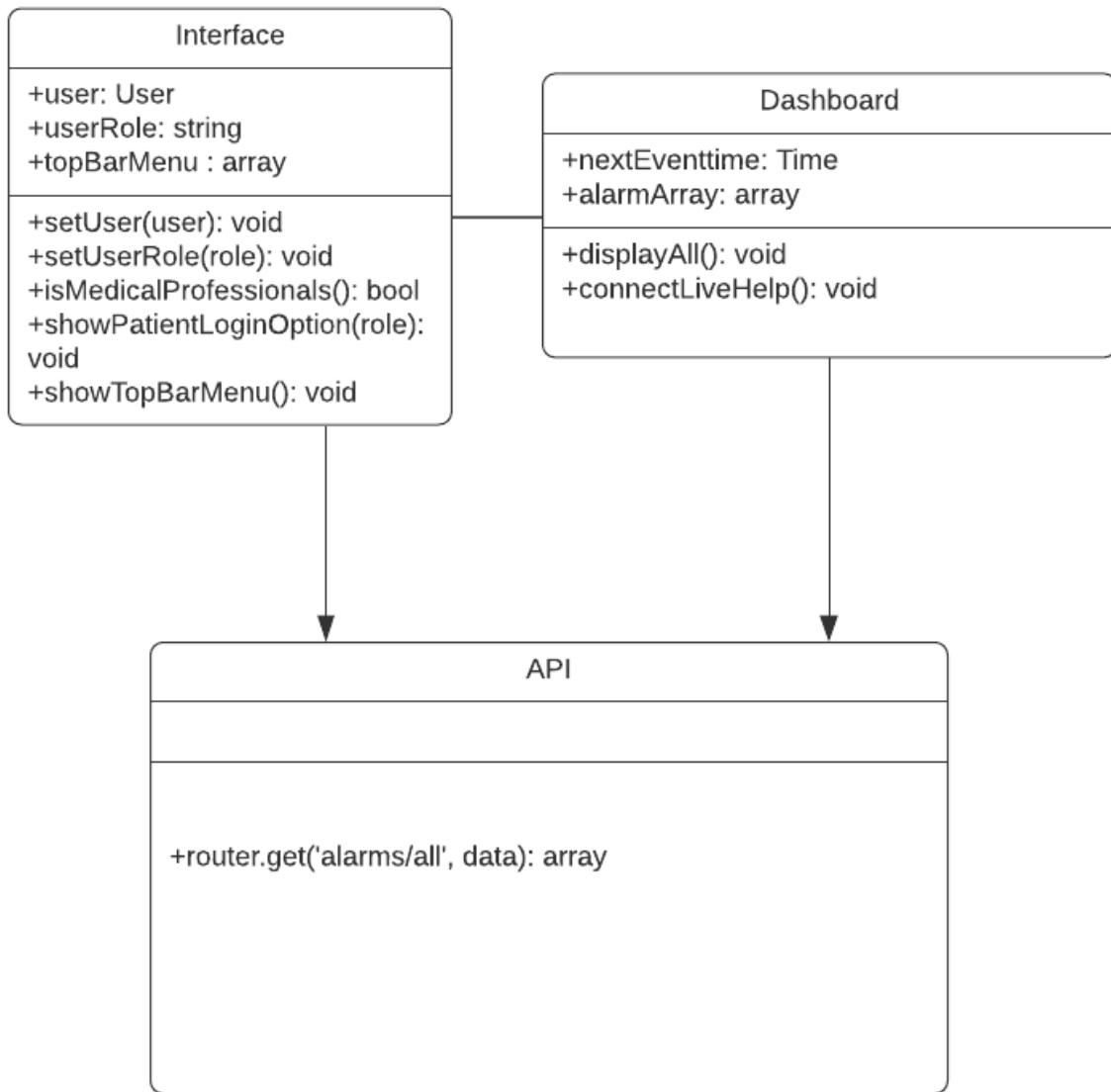
- **API**: Through the API call, the web App would fetch all the Tutorials and Infusion information and update the Patient's infusion table accordingly.
- **Tutorial**: Tutorial object includes: tutorial description, title, notes, videos, and text only display.
- **Video**: Video object that stores information about tutorial videos.
- **Interface**: Display different interface and menu options based on the user's type.

## UC#4 Access Profile Page ([link](#))



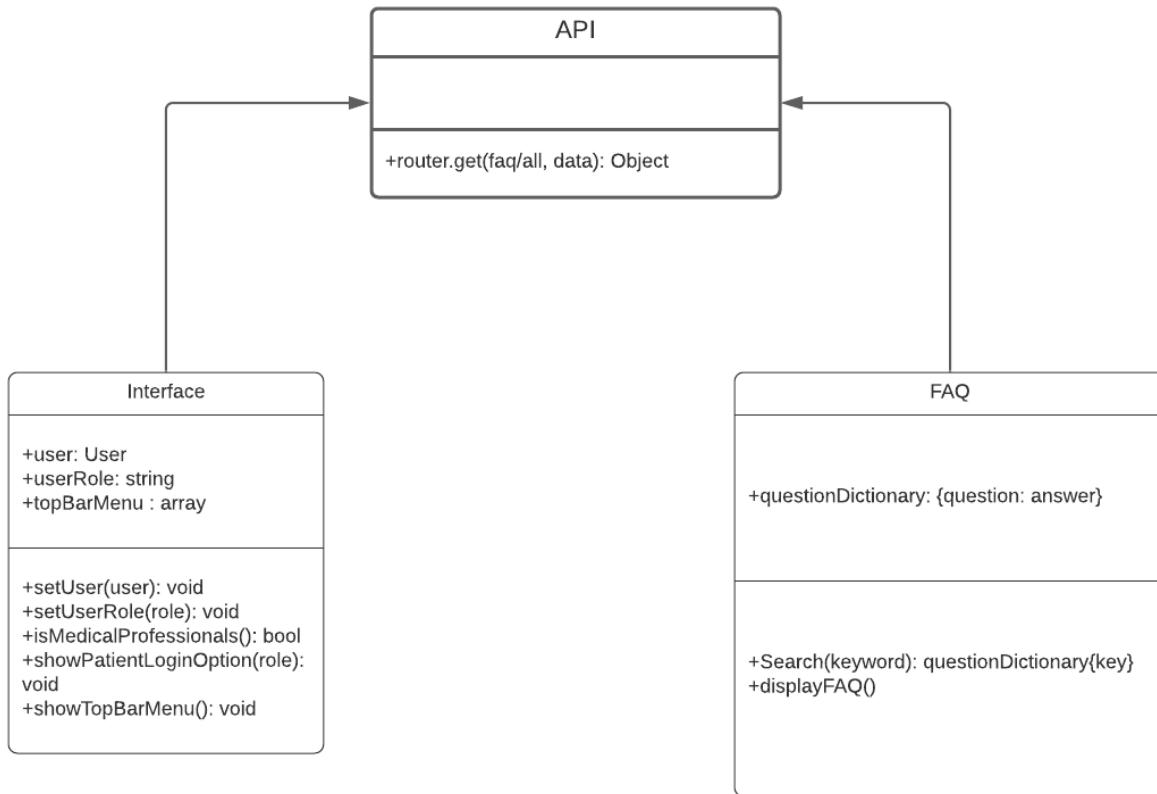
- **API:** Through the API call, the web App would fetch the user information and update the Profile page.
- **User:** User object that stores basic information that shared by every user
  - **Patient:** An inheritance object that has all attributes from the User object. The unique attributes include date of birth, and code.
- **Profile:** Profile object fetches information through API and displays them to users.
- **Interface:** Display different interface and menu options based on the user's type.

UC#5 Access Dashboard (Patients) ([link](#))



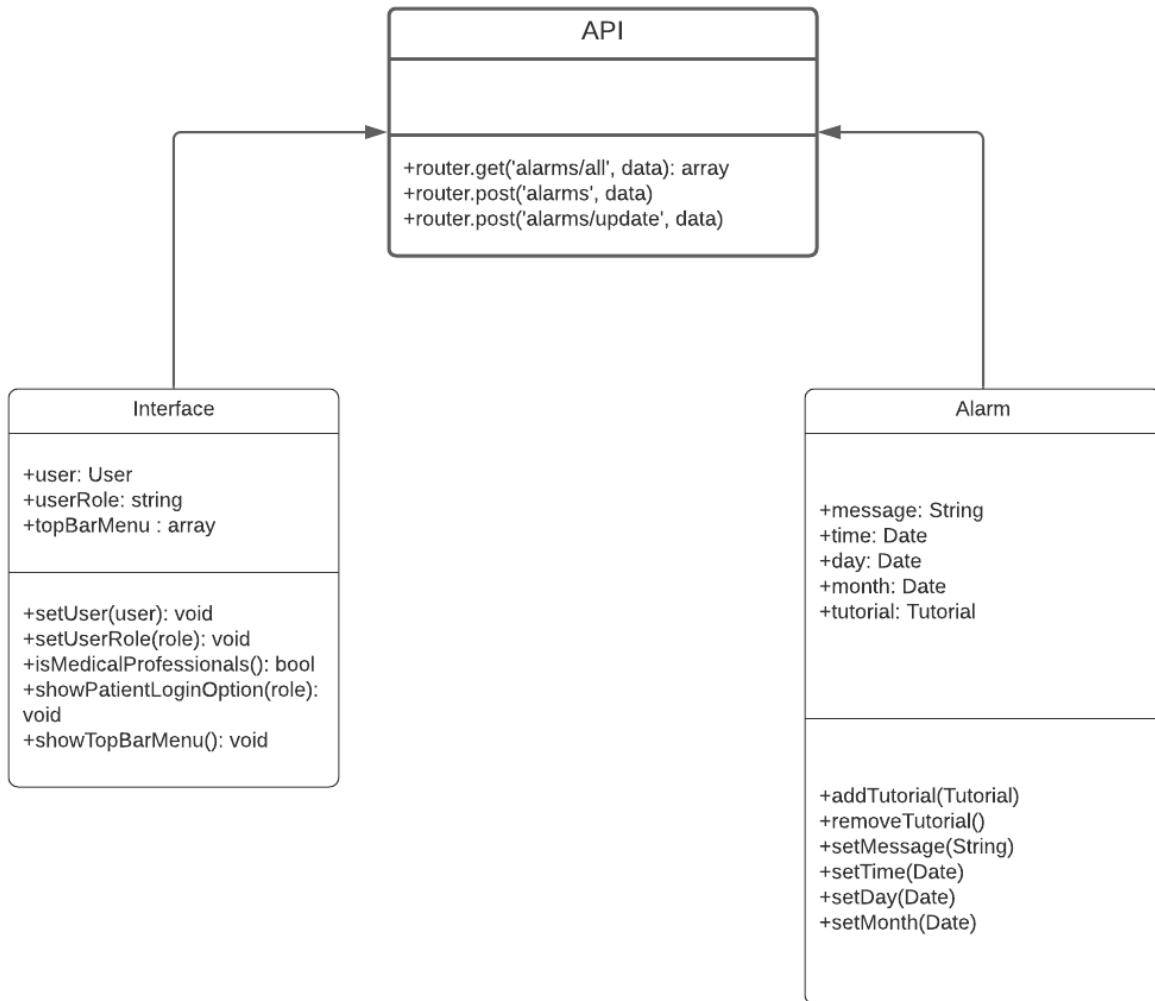
- **API**: Through the API call, the web App would fetch the necessary information for displaying on the home page (dashboard).
- **Dashboard**: Displays user's imminent events and other information such as pharmacy contact etc.
- **Interface**: Display different interface and menu options based on the user's type.

## UC #6 Access FAQ ([link](#))



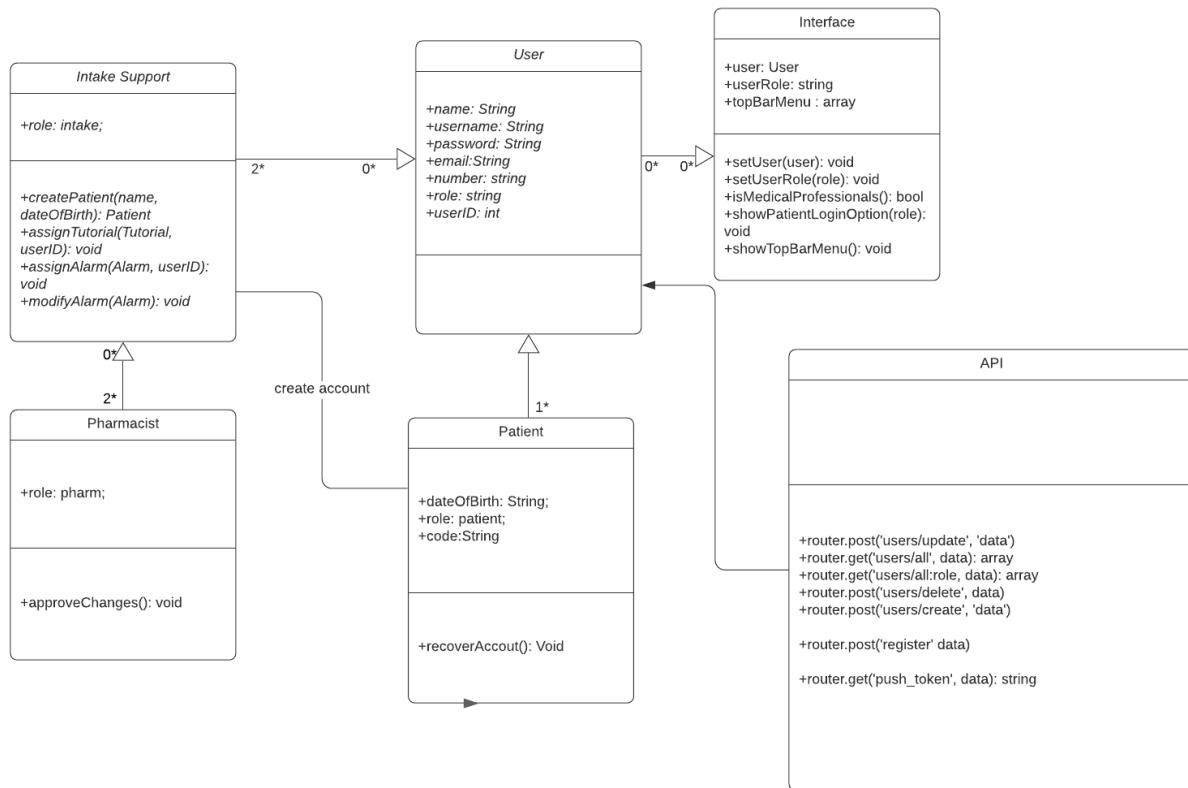
- **API**: Through the API call, the web App would fetch the prestored FAQ for displaying on the FAQ page.
- **FAQ**: Displays FAQ and provides a search function.
- **Interface**: Display different interface and menu options based on the user's type.

## UC #7 View Alarm ([link](#))



- **API:** Through the API call, the web App would fetch the alarms and update them to calendar.
- **Alarm:** Alarm object includes message, time, and associated tutorial(s).
- **Interface:** Display different interface and menu options based on the user's type.

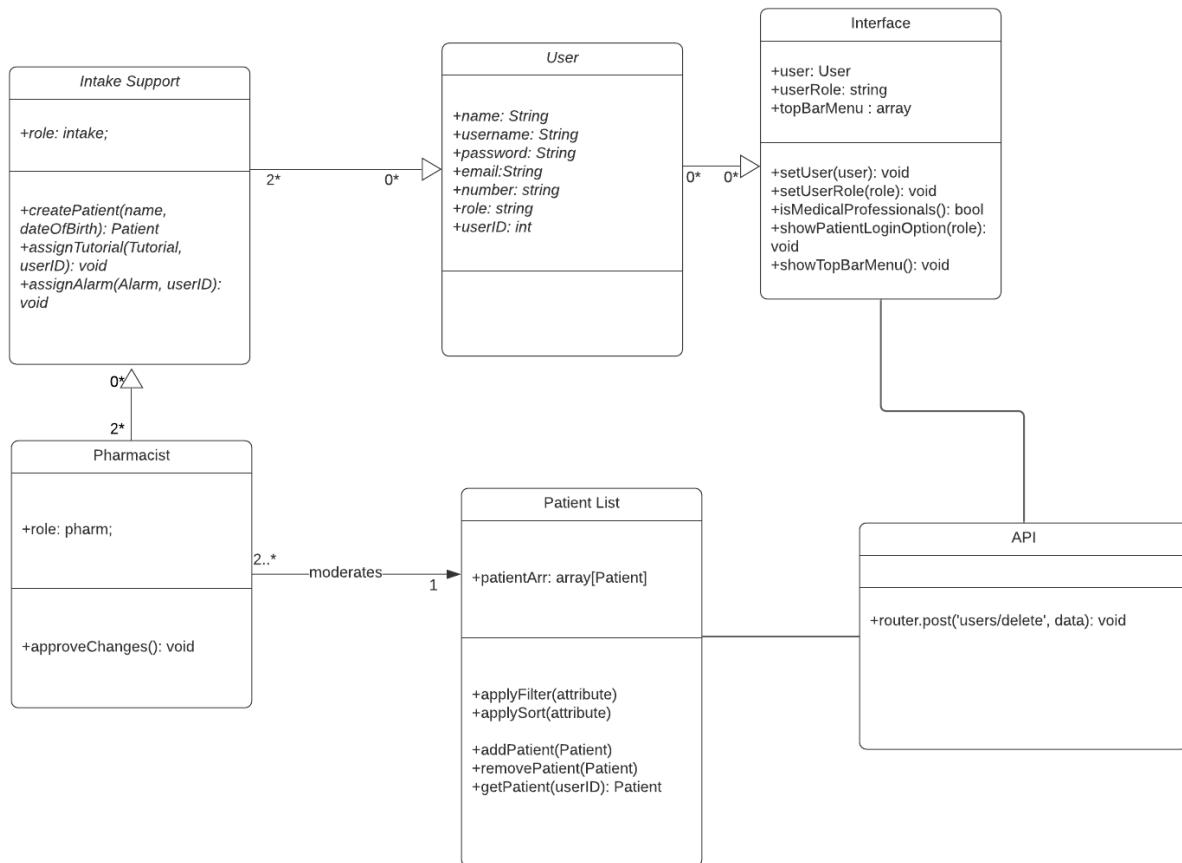
## UC #8 Signup, UC #9 Login, UC #10 Recover Account (Patients) ([link](#))



### - API

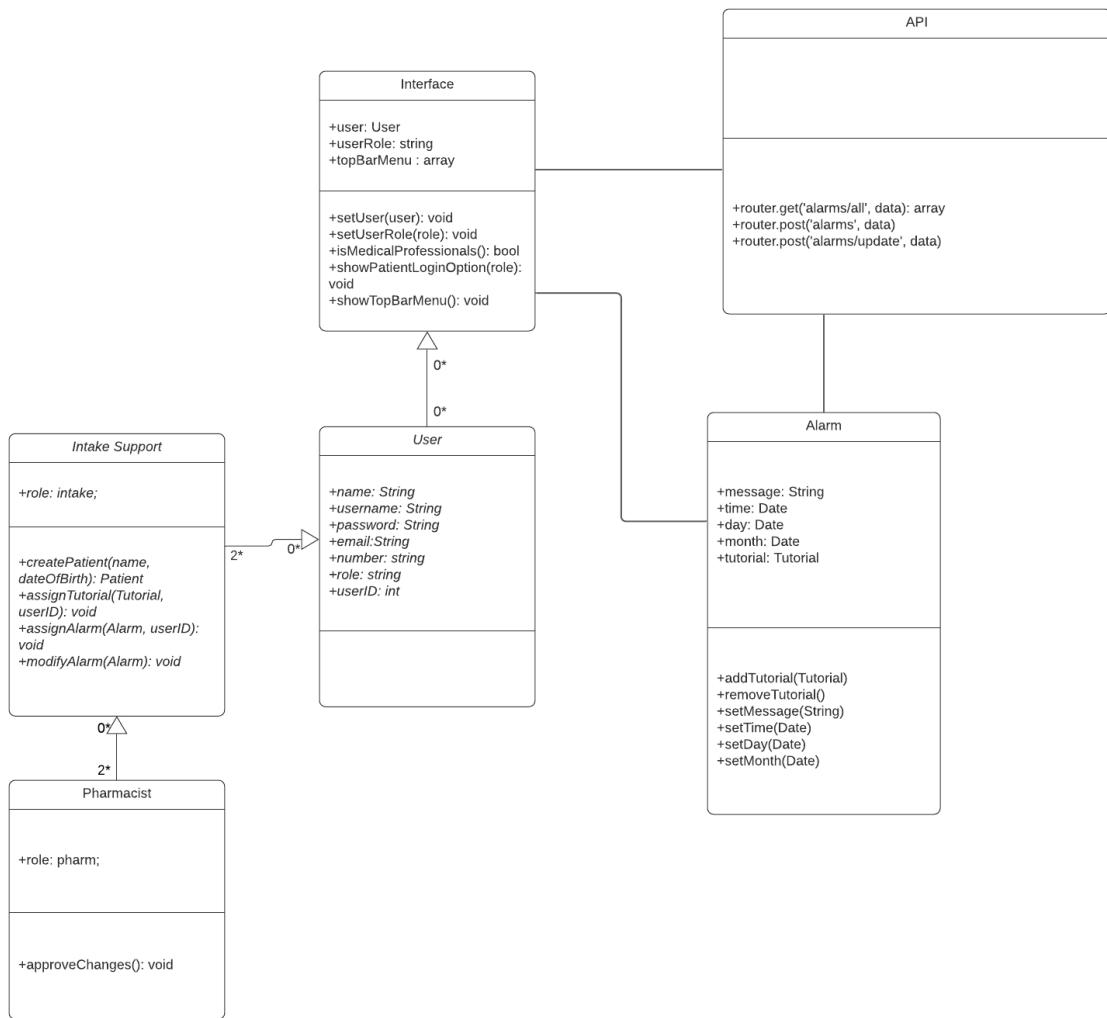
- Sign up: through API call, it will create a Patient user and store it in the database.
- Login: through the API call to verify a user
- Recover account: fetch a specific user and call router('reset', data) to recover account(s) through API call.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
    - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
  - **Patient**: An inheritance object that has all attributes from the User object. The unique attributes include date of birth, and code.
- **Interface**: Display different interface and menu options based on the user's type.

## UC#11 Delete Patient Account ([link](#))



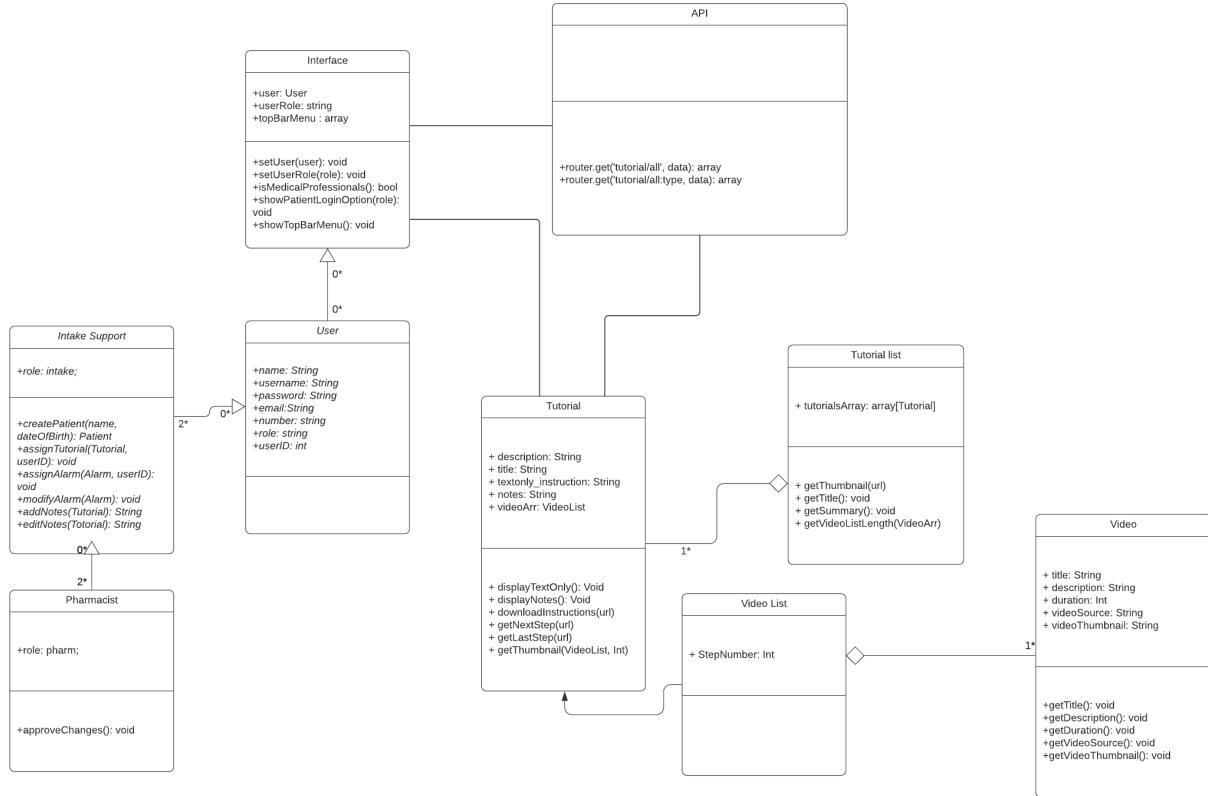
- **API**: Through an API call, the patient account will be deleted from the database.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
  - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
- **Interface**: Display different interface and menu options based on the user's type.
- **Patient List**: Array of Patient objects, can be filtered and sorted based on criterion

## UC #12 Set Alarms, 13 Modify Alarms ([link](#))



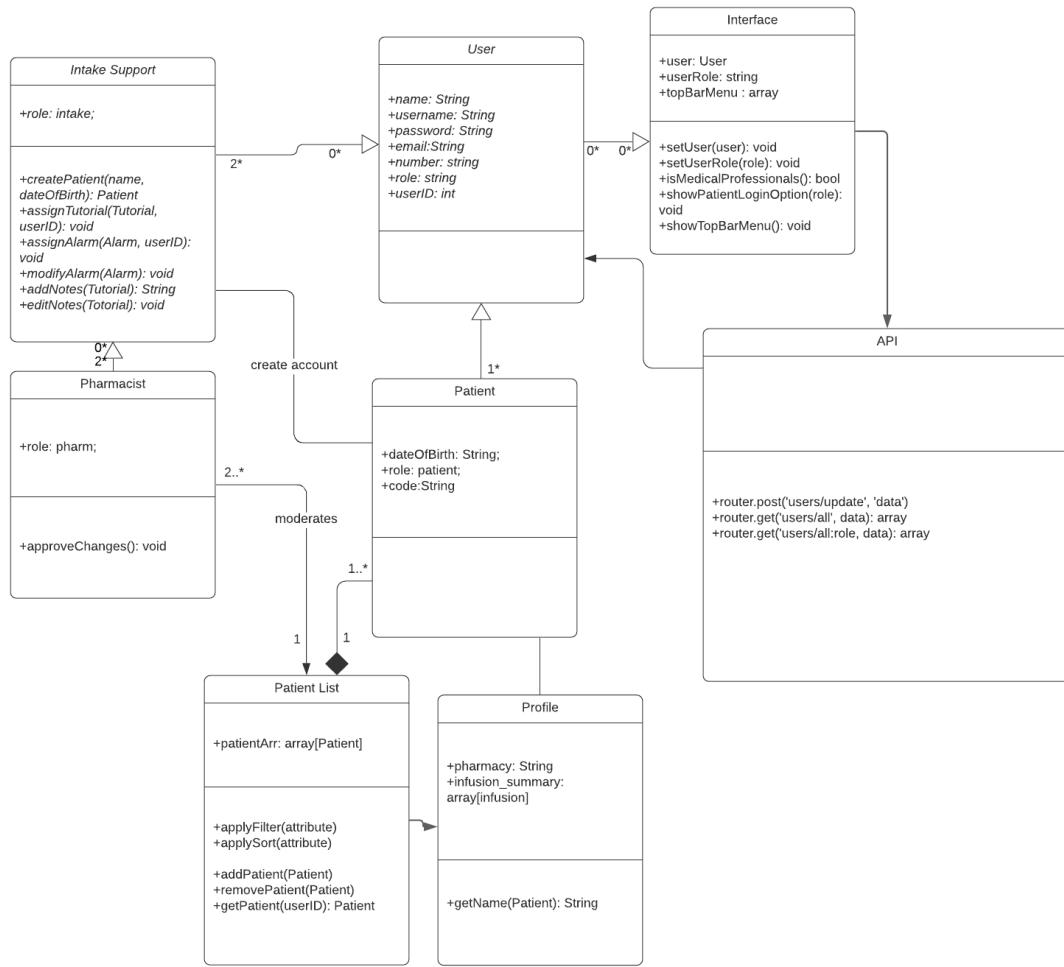
- **API**: The API will receive alarm data and save it in the database.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
  - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
- **Interface**: Display different interface and menu options based on the user's type.
- **Alarm**: Alarm object includes message, time, and associated tutorial(s).

## UC #14 - Assign Tutorials, UC #15 - Edit Tutorials, UC #22 - Add Tutorial Notes ([link](#))



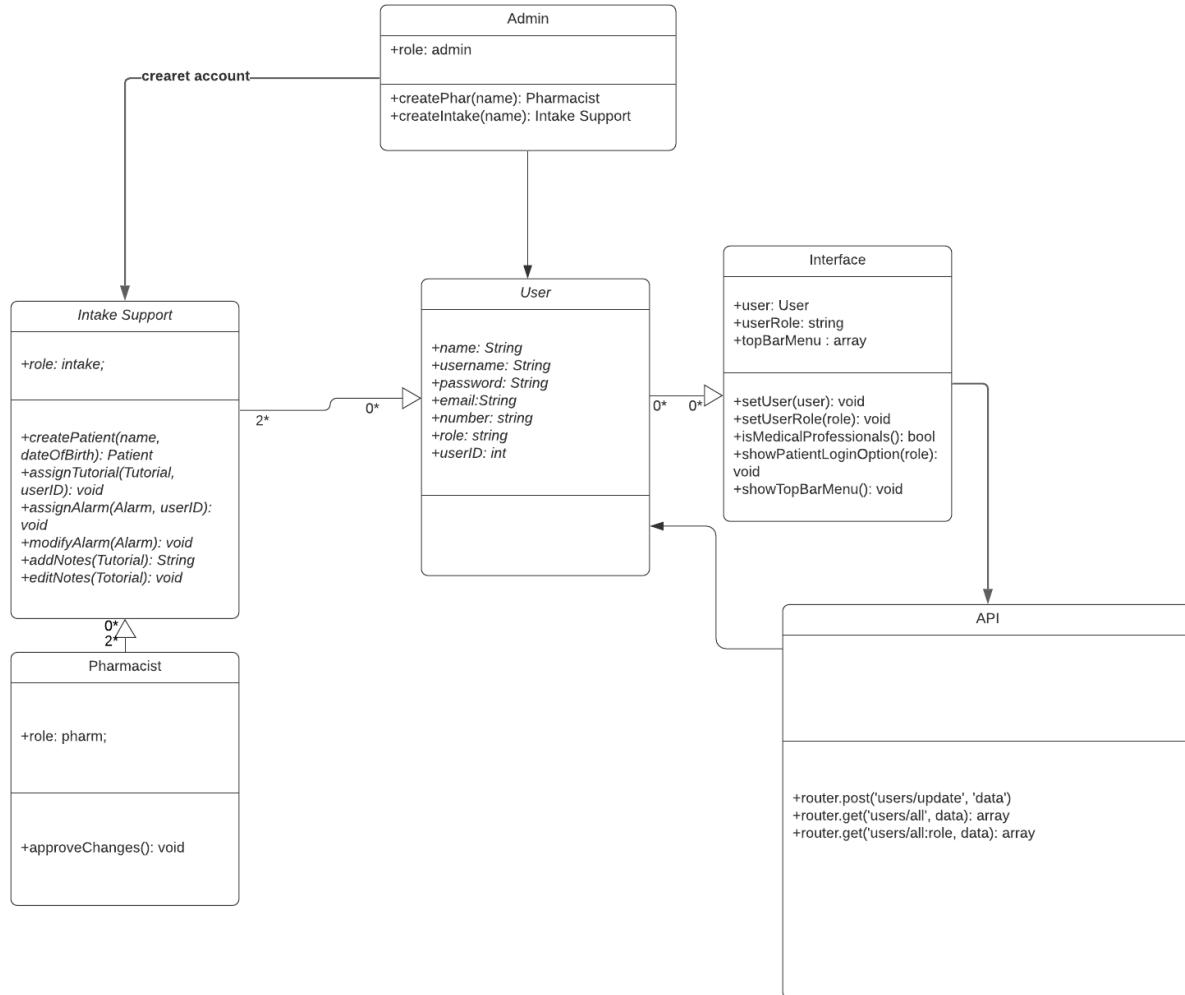
- **API**: Through an API call, pharmacists can retrieve certain video resources for a tutorial stored in the database.
- Through an API call, pharmacists can add notes for a patient's tutorial and save it in the database.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
  - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
- **Interface**: Display different interface and menu options based on the user's type.
- **Tutorial List**: Array of tutorial objects
  - **Tutorial**: Tutorial object includes title, description, text instructions, Video objects, etc
- **Video List**: List of video objects
  - **Video**: Video objects include title, description, thumbnail, duration, source, etc.

## UC #16 - View Patient Profile, UC#17 - View Patient List ([link](#))



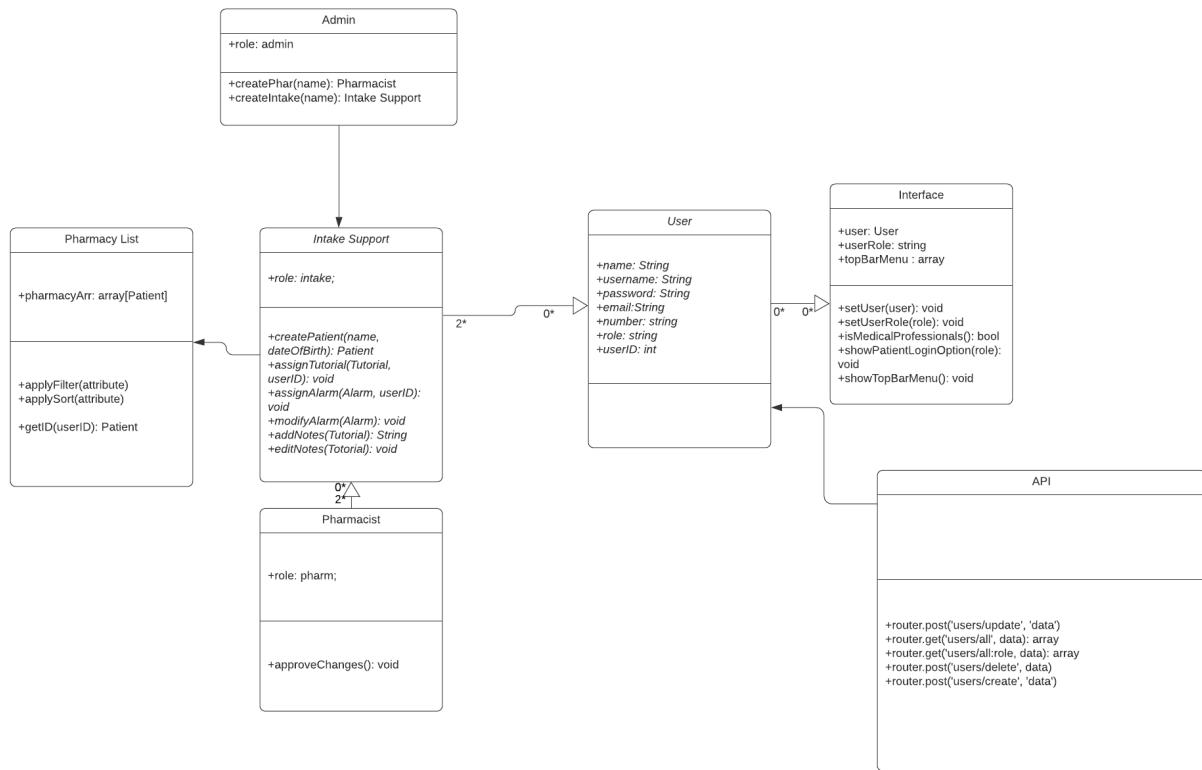
- **API**: Viewing a patient's profile will retrieve their information from the database through an API call and display it to the screen. Pharmacists can view multiple patient information through an API call that populates an array.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
    - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
  - **Patient List**: Array of Patient objects, can be filtered and sorted based on criterion
    - **Patient**: An inheritance object that has all attributes from the User object. The unique attributes include date of birth, and code.
- **Interface**: Display different interface and menu options based on the user's type.
- **Profile**: Profile object fetches information through API and displays them to users.

## UC #18 - Create Pharmacy/Pharmacist Account ([link](#))



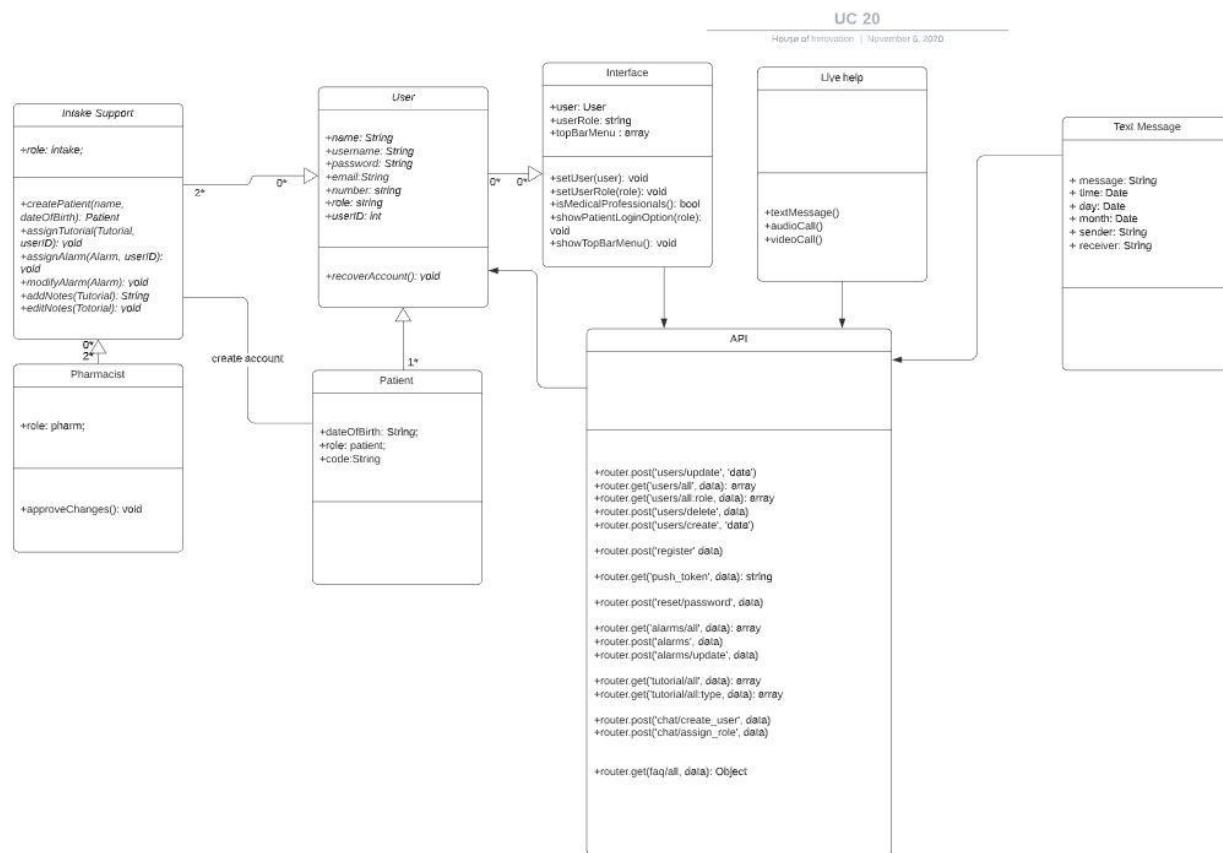
- **API**: Through an API call, admins will create pharmacist accounts and store them in the database.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
  - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
- **Administration**: Administration object that can create pharmacist and intake support accounts.
- **Interface**: Display different interface and menu options based on the user's type.

## UC #19 - View Pharmacy Account List ([link](#))



- **API:** Through an API call, admins can view all pharmacist accounts stored in the database.
- **User:** User object that stores basic information that shared by every user
  - **Intake support:** Intake support object that can create a patient account.
  - **Pharmacist:** Pharmacist object that shares all attributes and functions in Intake object.
    - **Pharmacy List:** An array object that contains all of the pharmacist account objects.
- **Interface:** Display different interface and menu options based on the user's type.

## UC #20 - Use Live Chat ([link](#))

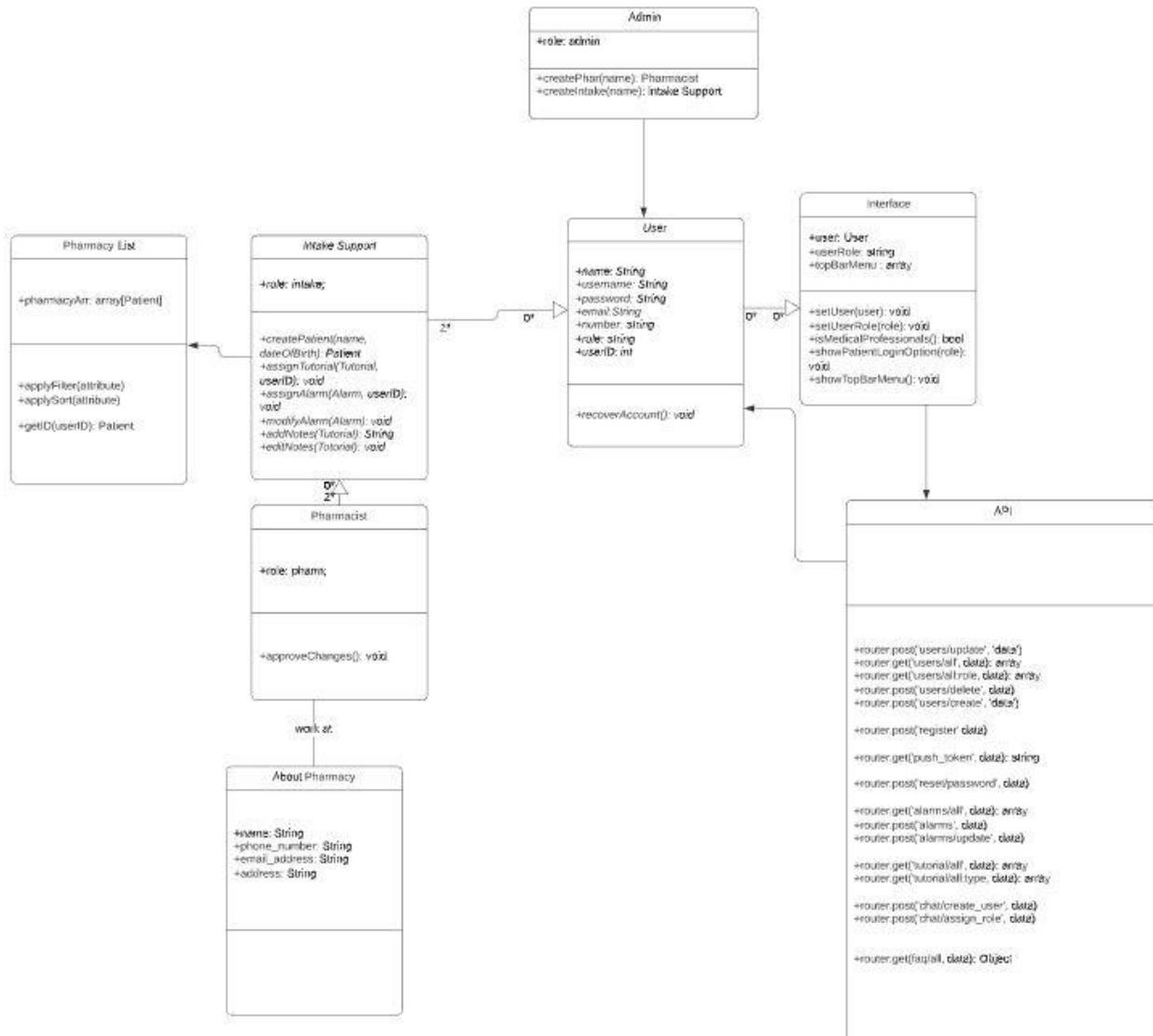


- **API**: Patient initiates a connection which searches for available pharmacists through an API call. If successful, a live chat begins.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
  - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
- **Patient**: An inheritance object that has all attributes from the User object. The unique attributes include date of birth, and code.
- **Interface**: Display different interface and menu options based on the user's type.
- **Live Help**: Live Help object is an interface feature that allows patients to message, call, and video chat.
- **Text Message**: Text Message object is a message sent to a patient's phone number

## UC #21 - Update Pharmacy Profile ([link](#))

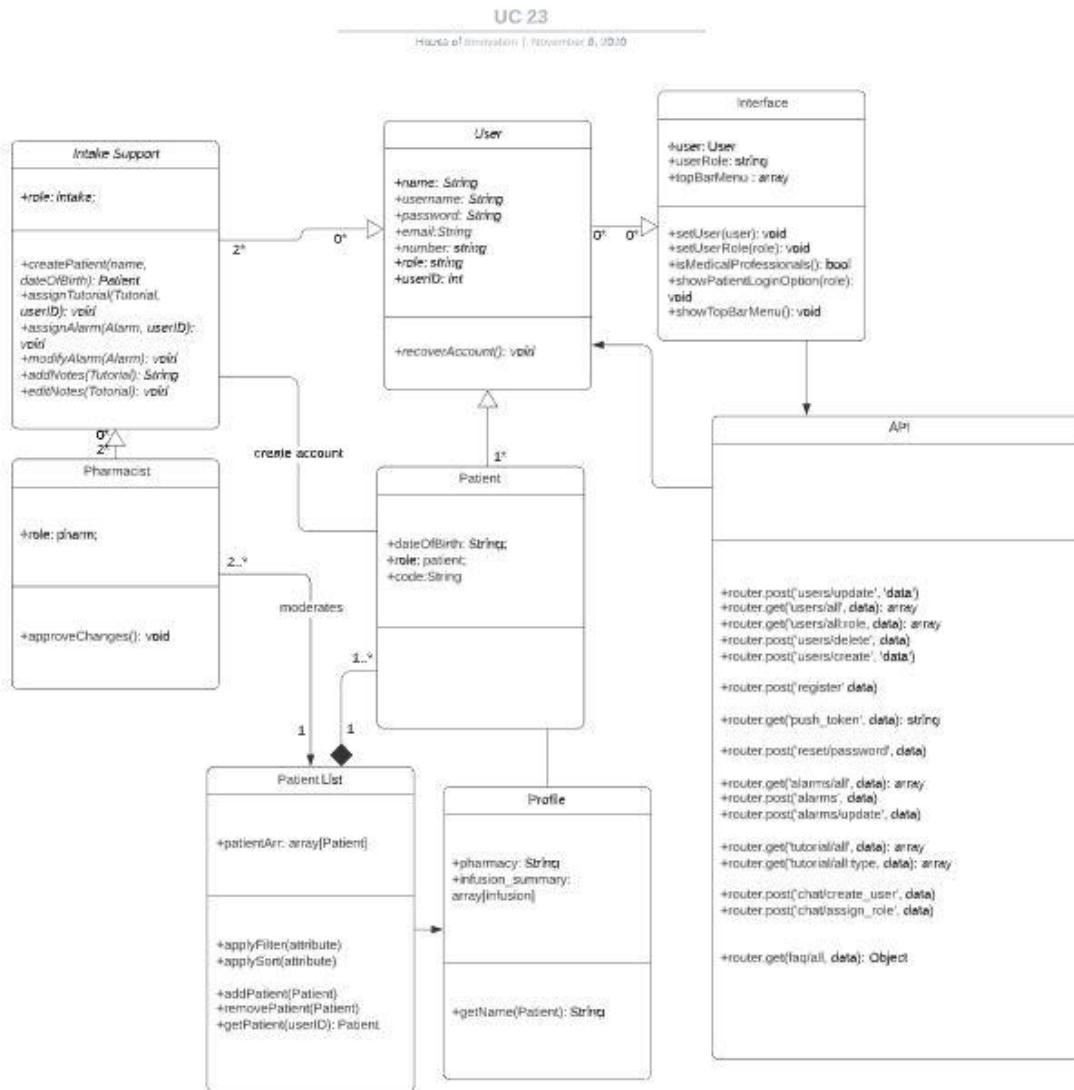
UC 21

House of Innovation | November 6, 2020



- **API:** Pharmacists can update their profile and save it to the database. Patients will be able to retrieve that data from the database when they view the contact page of their interface.
- **User:** User object that stores basic information that shared by every user
  - **Intake support:** Intake support object that can create a patient account.
    - **Pharmacy list:** Pharmacy list object contains information for each pharmacy. Updating pharmacy information updates its listing in the pharmacy list.
    - **Pharmacist:** Pharmacist object that shares all attributes and functions in Intake object.
  - **Administration:** Administration object that can create pharmacist and intake support accounts.
- **Interface:** Display different interface and menu options based on the user's type.
- **About Pharmacy:** About Pharmacy object includes the name, address, etc. of a patient's pharmacy

## UC #23 - Approve Infusion Changes ([link](#))

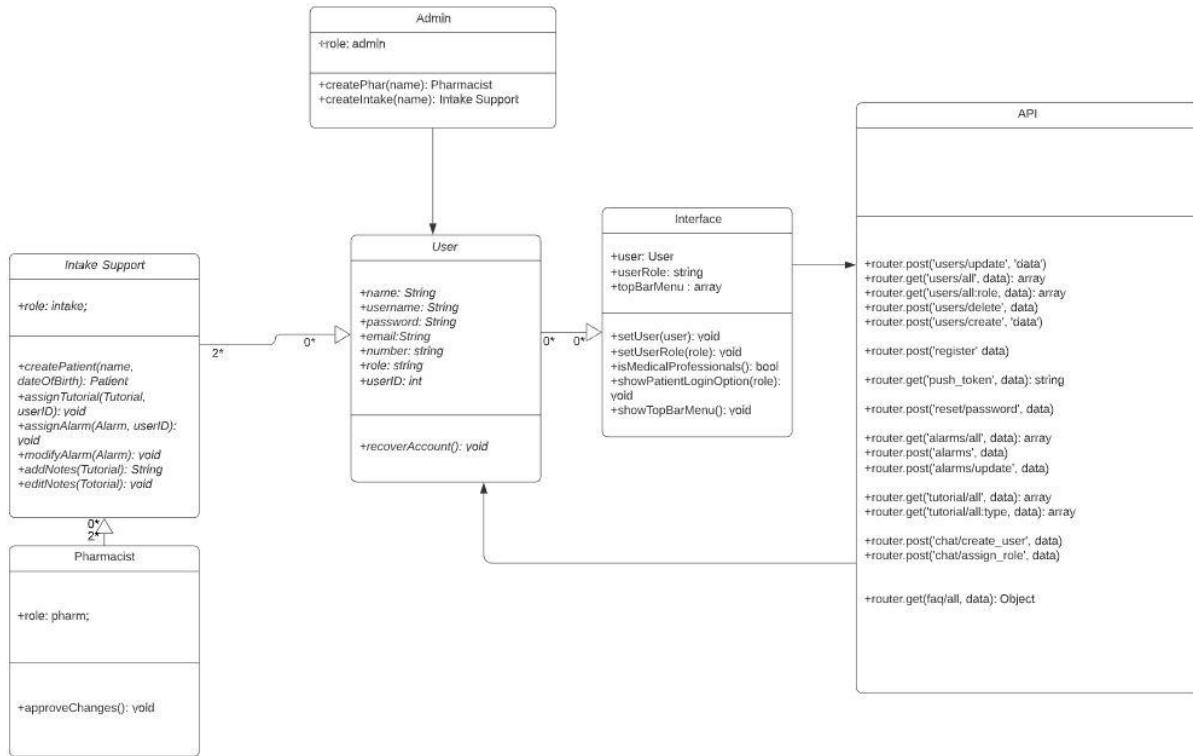


- **API**: Pharmacist users can receive “change updates” from an Intake Support through an API call. When changes are approved, the changes are saved to the database.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
  - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
- **Patient List**: Array of Patient objects, can be filtered and sorted based on criterion
  - **Patient**: An inheritance object that has all attributes from the User object. The unique attributes include date of birth, and code.
- **Interface**: Display different interface and menu options based on the user's type.
- **Profile**: Profile object fetches information through API and displays them to users.

## UC #24 - Recover Account (Pharmacist/Intake) ([link](#))

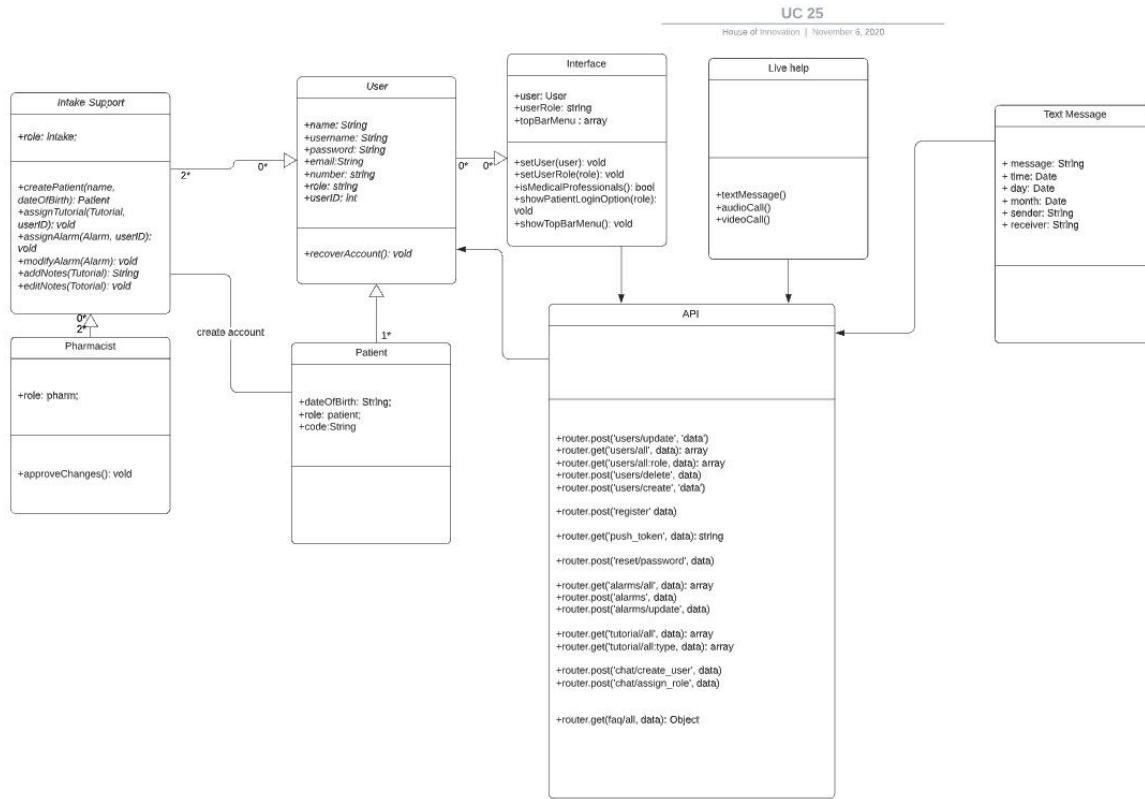
UC 24

House of Innovation | November 6, 2020



- **API**: Fetch a specific user and call router('reset', data) to recover account(s) through API call.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
  - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
- **Interface**: Display different interface and menu options based on the user's type.

## UC #25 - Use Video Chat ([link](#))

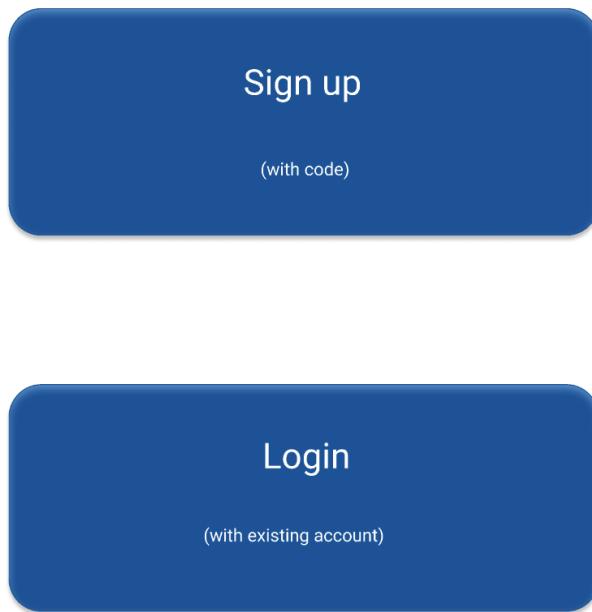


- **API**: Patient initiates a connection which searches for available pharmacists through an API call. If successful, a live chat begins. Users can then request to connect through a video chatting API.
- **User**: User object that stores basic information that shared by every user
  - **Intake support**: Intake support object that can create a patient account.
  - **Pharmacist**: Pharmacist object that shares all attributes and functions in Intake object.
  - **Patient**: An inheritance object that has all attributes from the User object. The unique attributes include date of birth, and code.
- **Interface**: Display different interface and menu options based on the user's type.
- **Live Help**: Live Help object is an interface feature that allows patients to message, call, and video chat.
- **Text Message**: Text Message object is a message sent between a pharmacist and patient

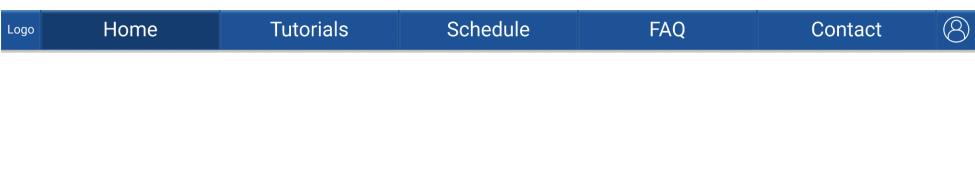
# Component Document

## Base Components

Login and SignUp Buttons

Image	
Description	<p><a href="https://material-ui.com/components/buttons/">https://material-ui.com/components/buttons/</a></p> <p>The button API can be used here. Customize event handler and style ourselves</p>
Functionality	Used for patient sign up or login

## Navigation Bar

Image	
Description	<p><a href="https://material-ui.com/components/app-bar/">https://material-ui.com/components/app-bar/</a></p> <p>Might want to consider viewing: "App Bar with Menu"</p> <ul style="list-style-type: none"><li>- All the components you'll need for the nav are in this section under source code</li></ul>
Functionality	Used for site navigation

## Welcome Summary Message

Image	Hi Jasmine, your next infusion is on Thursday, November 19, 2020 at 6:00 pm
Description	<a href="https://material-ui.com/components/typography/">https://material-ui.com/components/typography/</a> Standalone typography component
Functionality	Used for landing page  Use props to pass in the user's name and next infusion info from the backend

## Next Infusion Video Widget

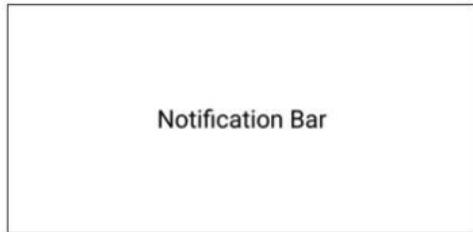
Image	 <p><b>Tutorial Video Thumbnail</b> + <b>Tutorial Button</b></p>
Description	A widget on the home page for quick access to the next infusion's tutorial  <a href="https://material-ui.com/components/cards/">https://material-ui.com/components/cards/</a>  Img tag or card component with href to the correct page?
Functionality	Pull from the patient's infusion summary and relevant assigned tutorial  Clicking on it takes them to the proper tutorial

## Daily Schedule Widget

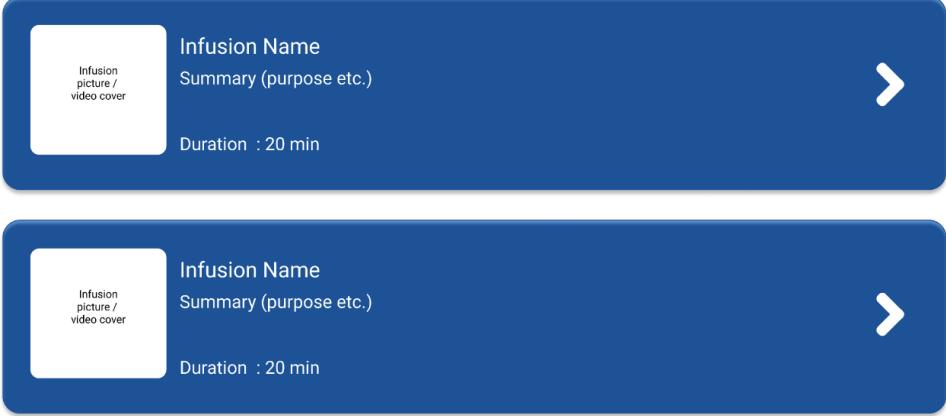
Image	 <p>Daily schedule</p>	
-------	--	--

Description	A widget on the home page that summarizes a patient's infusion schedule for the day. May be combined with "Daily Infusion Alarms Component" below
Functionality	Displays the daily infusion schedule

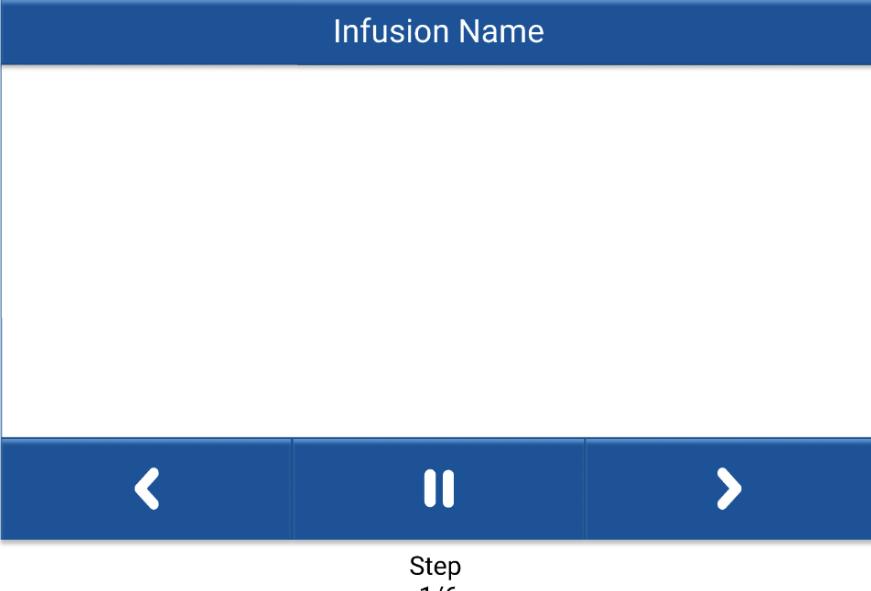
### Notification Widget

Image	
Description	Components TBD based on final design.
Functionality	Displays any changes that a pharmacist made to their account as well as infusion updates

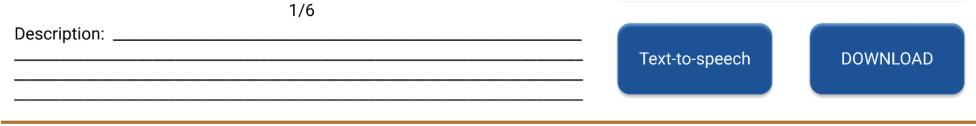
## Tutorial Listing

Image	
Description	<p><a href="https://material-ui.com/components/cards/">https://material-ui.com/components/cards/</a></p> <p>Use Grid and Typography as well for formatting.</p> <p>Use Button and <a href="https://material-ui.com/components/material-icons/">https://material-ui.com/components/material-icons/</a> for the arrow buttons</p>
Functionality	Used for listing tutorials in the tutorial listing page. A thumbnail of the video should appear on the left. The videos' summaries are next and a button that takes a user to the individual tutorial page is on the right. MaterialUI's ChevronRight icon matches the one from this image

## Video Player

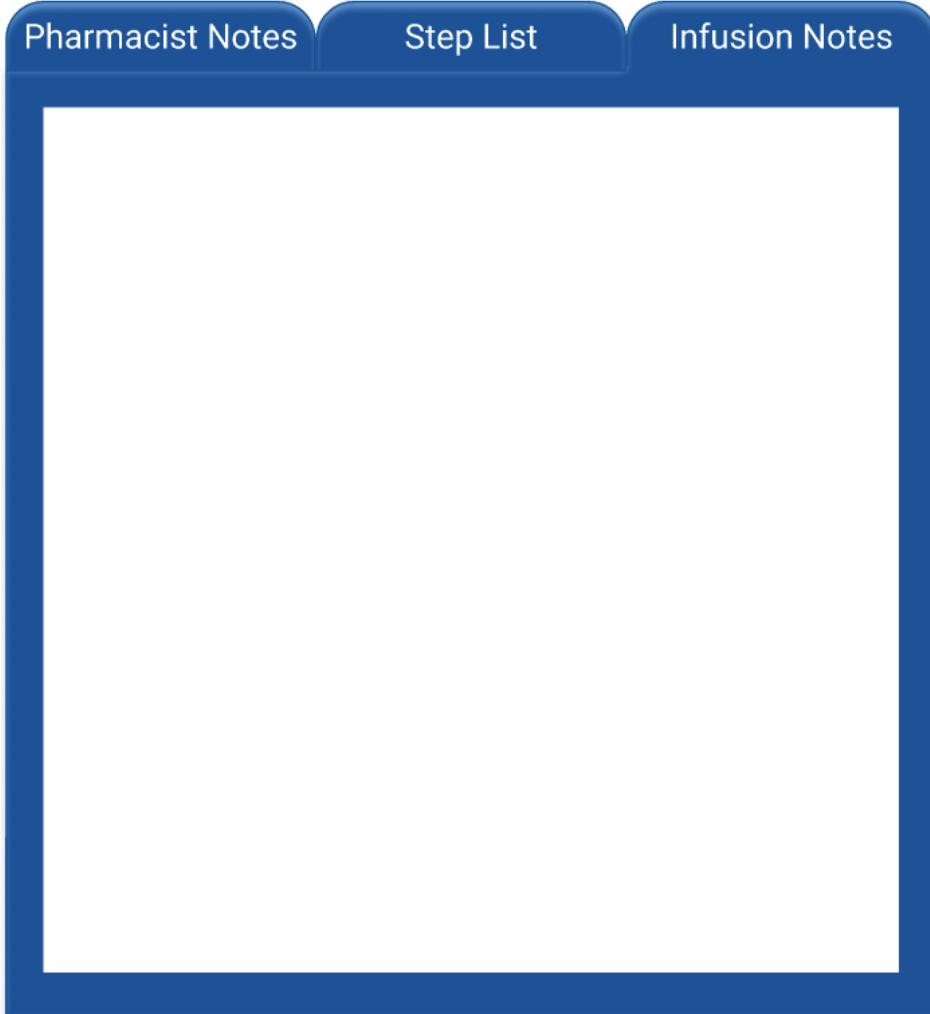
Image	
Description	<p><a href="https://material-ui.com/components/paper/">https://material-ui.com/components/paper/</a></p> <p>For the bottom row, maybe a sticky bottom nav or a Grid api with 3 buttons inside?</p>
Functionality	<p>Used for single tutorial page</p> <p>Plays a single video in an array of videos. Options to pause, prev Video, next Video</p>

## Video Text Section

Image	
-------	--

Description	Button API for TTS and Document Download Grid for formatting Typography for Descriptions and Counter
Functionality	Used for single tutorial page Renders description for the video Clicking on TTS allows a user to listen to an audio transcription Clicking on Document Download allows a user to download any supplied pdfs for the given video

## Pharmacist Notes Tab, Step List Tab, and Infusion Notes Tab

Image	
Description	Just use Typography for each component and set up a rough structure to be filled in with props

Functionality	Used to combine with tabs component Allows for supplementary information for each tutorial
---------------	---

## Daily Infusion Alarms

Image	<p>The image shows a blue card component with a title "Thursday" at the top. Below the title are three horizontal rows, each consisting of a time slot on the left and a corresponding infusion type on the right. The times are 6:30 AM, 2:30 PM, and 10:30 PM, and the infusion type is "Antibiotic Infusion" for all three entries.</p>
Description	<p>Typography and props for the given day  <a href="https://material-ui.com/components/cards/">https://material-ui.com/components/cards/</a></p> <ul style="list-style-type: none"> <li>- Props.time for the left</li> <li>- Props.description for the right</li> </ul>
Functionality	<p>Used for the Schedule page Displays infusion schedule for the day</p>

## Patient Table

Image	<table border="1"> <thead> <tr> <th>Patient Name</th><th>ID #</th><th>Date of Birth</th><th>Most Recent Check-In</th><th>Next Check-In</th><th>Infusion Type</th><th>Notification Information</th></tr> </thead> <tbody> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> </tbody> </table>	Patient Name	ID #	Date of Birth	Most Recent Check-In	Next Check-In	Infusion Type	Notification Information																																																								
Patient Name	ID #	Date of Birth	Most Recent Check-In	Next Check-In	Infusion Type	Notification Information																																																										
Description	<p><a href="https://material-ui.com/components/tables/">https://material-ui.com/components/tables/</a></p> <ul style="list-style-type: none"> <li>- Use conditional rendering to generate the fields and change the color if even/odd # in the array</li> </ul>																																																															
Functionality	<p>Used for the Pharmacist home page</p> <p>Displays patient array information for the day</p>																																																															

## Sort and Search on Single Line

Image	
Description	<p><a href="https://www.npmjs.com/package/material-ui-search-bar">https://www.npmjs.com/package/material-ui-search-bar</a></p> <p><a href="https://material-ui.com/components/menus/">https://material-ui.com/components/menus/</a></p> <p>Can be put on the same line if you use grid api</p>
Functionality	<p>Used for the Pharmacist home page</p> <p>Displays patient array information for the day</p>

## Frequently Asked Question List

Image	
Description	<p><a href="https://material-ui.com/components/accordion/">https://material-ui.com/components/accordion/</a></p> <p>Provide an expansive question bar.</p>
Functionality	<p>Used for the Patient FAQ page</p> <p>Used for question and answer listing.</p>

## Patient Alarm Summary (Pharmacist View)

Image	
Description	<p><a href="https://material-ui.com/api/paper/">https://material-ui.com/api/paper/</a></p> <p>Maybe just need to build the custom paper part with typography title... create a component for an individual alarm entry</p>

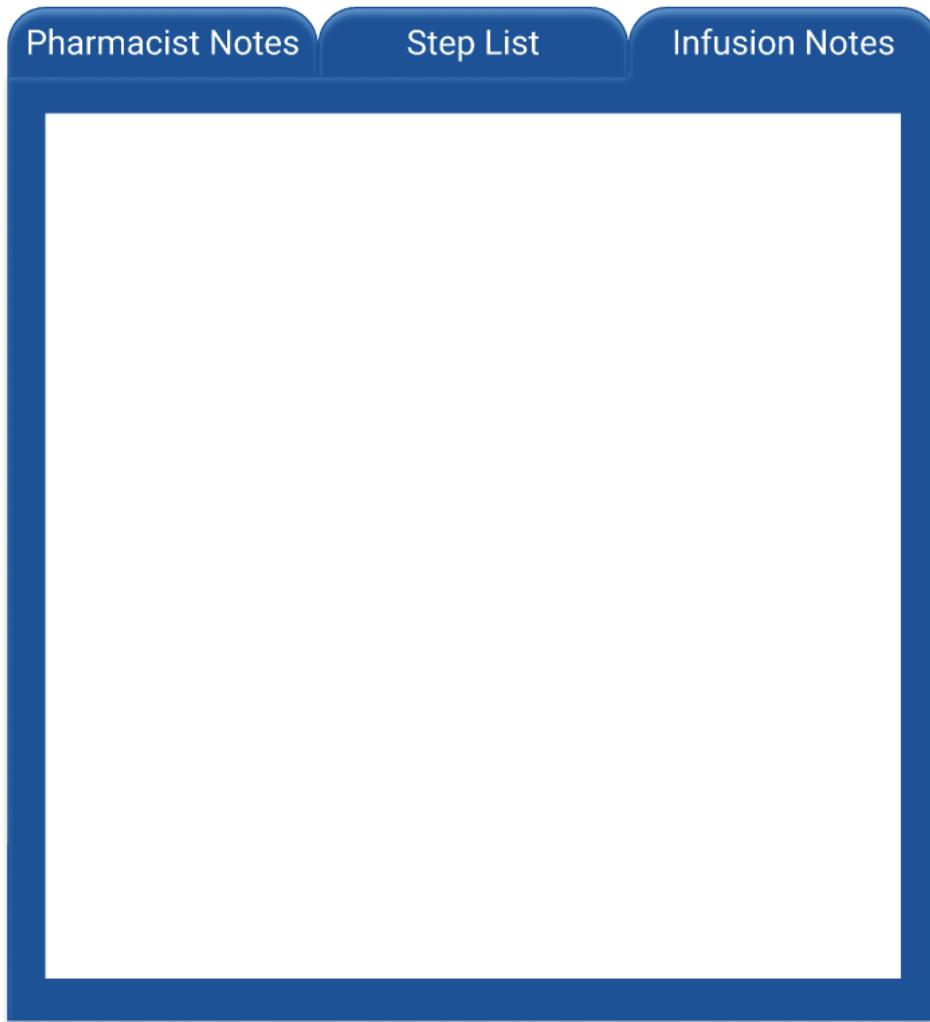
Functionality	Used for the Pharmacist's view of a patient's profile Displays patient weekly alarms
---------------	---

### Patient Assigned Tutorials (Pharmacist View)

Image	
Description	<p><a href="https://material-ui.com/api/paper/">https://material-ui.com/api/paper/</a></p> <p>Add tutorial component as entries</p>
Functionality	Used for the Pharmacist's view of a patient's profile Displays patient assigned tutorials

## Combined Components

Video Tabs

Image	
Description	<p><a href="https://material-ui.com/components/tabs/">https://material-ui.com/components/tabs/</a></p> <p>Combined with Components for Pharmacist Notes, Step List, and Infusion Notes</p>

Functionality	Used for single tutorial page Allows for supplementary information for each tutorial
---------------	---

## Login Combined

Image	<p>The wireframe shows a layout for a combined login and logo page. On the left, there is a vertical column labeled "Image". To its right, at the top center, is a large rectangular box labeled "LOGO". Below it, there is a horizontal input field with rounded corners containing the placeholder "username". Further down, another horizontal input field with rounded corners contains the placeholder "password" and a "show" link to its right. Below these fields is a blue button with the word "Login" in white. At the bottom center of the page is a blue link labeled "Forgot password?".</p>
-------	--

Description	Img tag Two input fields WITH validation Two buttons <ul style="list-style-type: none"><li>- Forgot button with no outline</li><li>- Login button with outline</li></ul>
Functionality	Allows a patient to login

### Sign Up Combined

Image	<p style="text-align: center;"><b>Create your account</b></p> <div style="text-align: center;"> <input style="border: 1px solid #ccc; border-radius: 10px; padding: 5px; width: 200px; margin-bottom: 10px;" type="text" value="Access code"/> <span style="border: 1px solid #ccc; border-radius: 50%; padding: 2px 5px; font-size: small;">?</span> </div> <div style="text-align: center;"> <input style="border: 1px solid #ccc; border-radius: 10px; padding: 5px; width: 200px; margin-bottom: 10px;" type="text" value="Username"/> </div> <div style="text-align: center;"> <input style="border: 1px solid #ccc; border-radius: 10px; padding: 5px; width: 200px; margin-bottom: 10px;" type="text" value="Email"/> </div> <div style="text-align: center;"> <input style="border: 1px solid #ccc; border-radius: 10px; padding: 5px; width: 200px; margin-bottom: 10px;" type="text" value="Phone Number"/> </div> <div style="text-align: center;"> <input style="border: 1px solid #ccc; border-radius: 10px; padding: 5px; width: 200px; margin-bottom: 10px;" type="text" value="Password"/> </div> <div style="text-align: center;"> <input style="border: 1px solid #ccc; border-radius: 10px; padding: 5px; width: 200px; margin-bottom: 10px;" type="text" value="Confirm password"/> </div> <div style="text-align: center;"> <input style="background-color: #005a99; color: white; border: 1px solid #005a99; border-radius: 10px; padding: 10px; width: 150px;" type="button" value="Submit"/> </div>
-------	---

Description	<p><a href="https://material-ui.com/api/form-control/">https://material-ui.com/api/form-control/</a></p> <ul style="list-style-type: none"><li>- A parent-container for forms in Material UI</li><li>- Ensure each one has validation as needed. Stack Overflow has lots of regex you can adapt from</li></ul> <p>Button API</p>
Functionality	Allows a patient to sign up provided that all details are correct, pass validation, and that the username is unique to the database

## Sign Up Success

Image

picture

**Congratulations!**  
Your account is ready

Login

Description

Img tag and typography. Button redirects to login

Functionality	Signals to a user that their sign up was successful and that they are now ready to log in to the system.
---------------	--

## Pharmacist Settings

Image	
Description	<p><a href="https://material-ui.com/components/tabs/">https://material-ui.com/components/tabs/</a>  <a href="https://material-ui.com/components/text-fields/">https://material-ui.com/components/text-fields/</a></p> <p>Can use Paper or card components for the About section..</p>

Functionality	Signals to a user that their sign up was successful and that they are now ready to log in to the system.
---------------	--

## Pages

### Welcome Page

Image	 <p>The image placeholder shows a wireframe representation of a web page. At the top, a header bar is labeled "Welcome page". Below the header, the main content area contains the text "Welcome to NAME" centered above a horizontal line. In the bottom right corner of the content area, the word "picture" is written next to a small square placeholder.</p>
Description	A simple landing page for patients
Functionality	Greeting page that leads to login or sign up

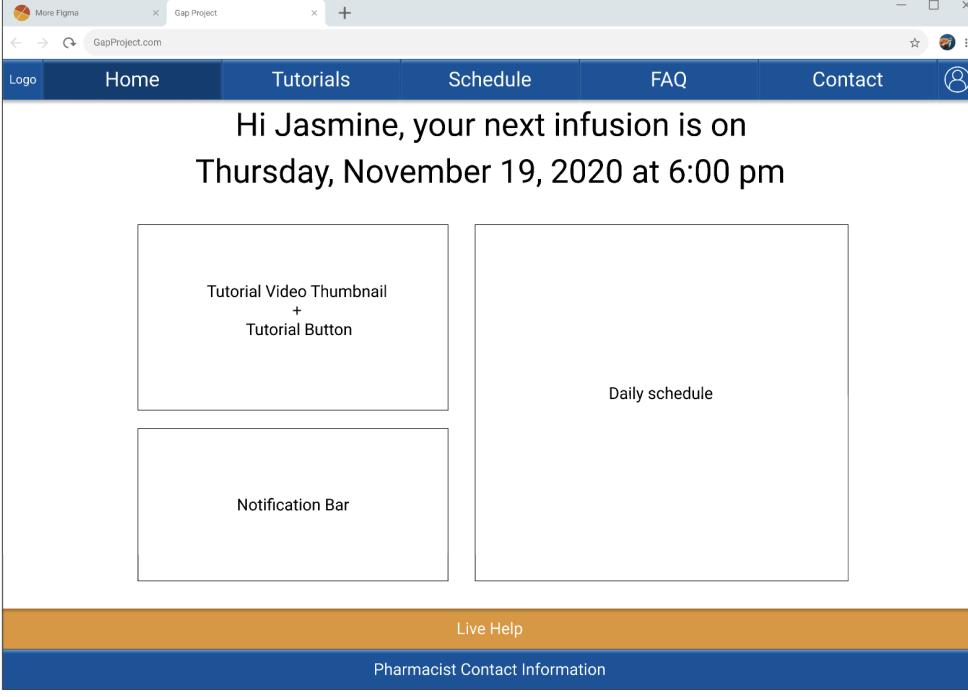
## Error 404 Page

Image	[ PROVIDE IMAGE HERE ]
Description	A simple error 404 page if the app does not connect to the server / endpoints  Will probably need to use Grid for spacing and reformatting and img tag for the provided vectors
Functionality	Signals to a user that there was an error in trying to connect to the database. Provide an explanation for a positive user experience

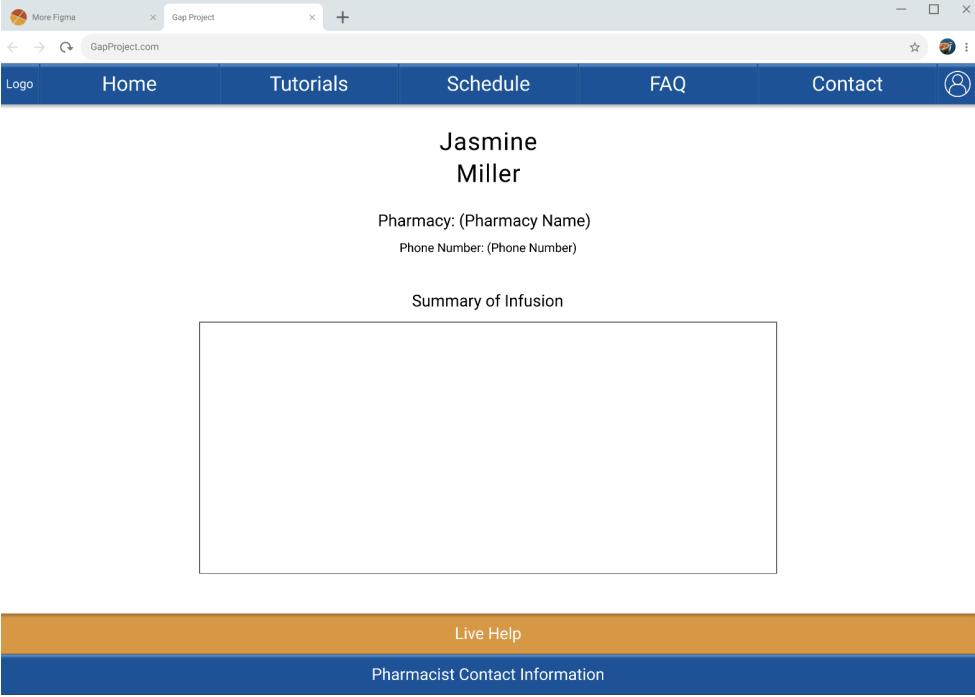
## Login & SignUp Page

Flow	Welcome → Login Combined OR Sign Up Combined → Form → Success
Description	All of the involved components will be placed into this login and sign up component but will be conditionally rendered. Use event handlers to toggle between the components and save the state (user entered information) all in a single file
Functionality	Signals to a user that their sign up was successful and that they are now ready to log in to the system.

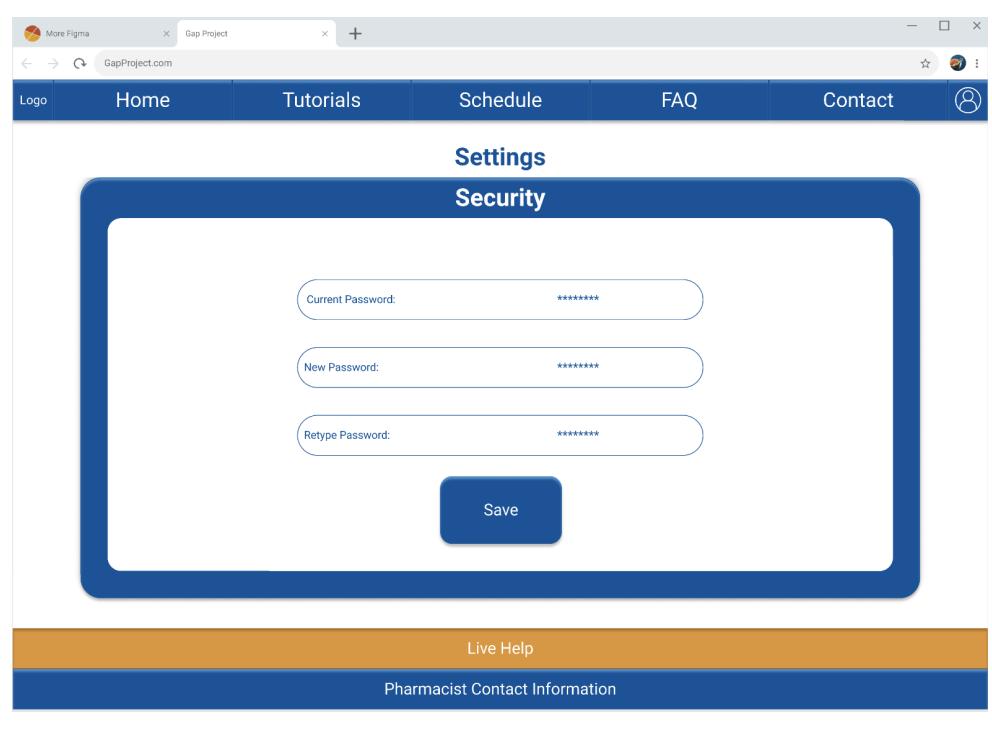
## Home Page

Image	 A Figma wireframe of a web browser window titled "Gap Project". The address bar shows "GapProject.com". The navigation bar includes "Logo", "Home", "Tutorials", "Schedule", "FAQ", "Contact", and a user icon. The main content area displays a welcome message: "Hi Jasmine, your next infusion is on Thursday, November 19, 2020 at 6:00 pm". Below this are two large rectangular boxes: one labeled "Tutorial Video Thumbnail + Tutorial Button" and another labeled "Notification Bar". To the right of the "Notification Bar" box is a smaller box labeled "Daily schedule". At the bottom of the page are two horizontal bars: an orange "Live Help" bar and a dark blue "Pharmacist Contact Information" bar.
Description	<p>Components</p> <ul style="list-style-type: none"><li>- Welcome Summary</li><li>- Daily Schedule Widget</li><li>- Notification Widget</li><li>- Next Infusion Tutorial Widget</li><li>- Live Help</li><li>- Pharmacist Contact</li></ul>
Functionality	An at-a-glance view relating to the patient's infusion

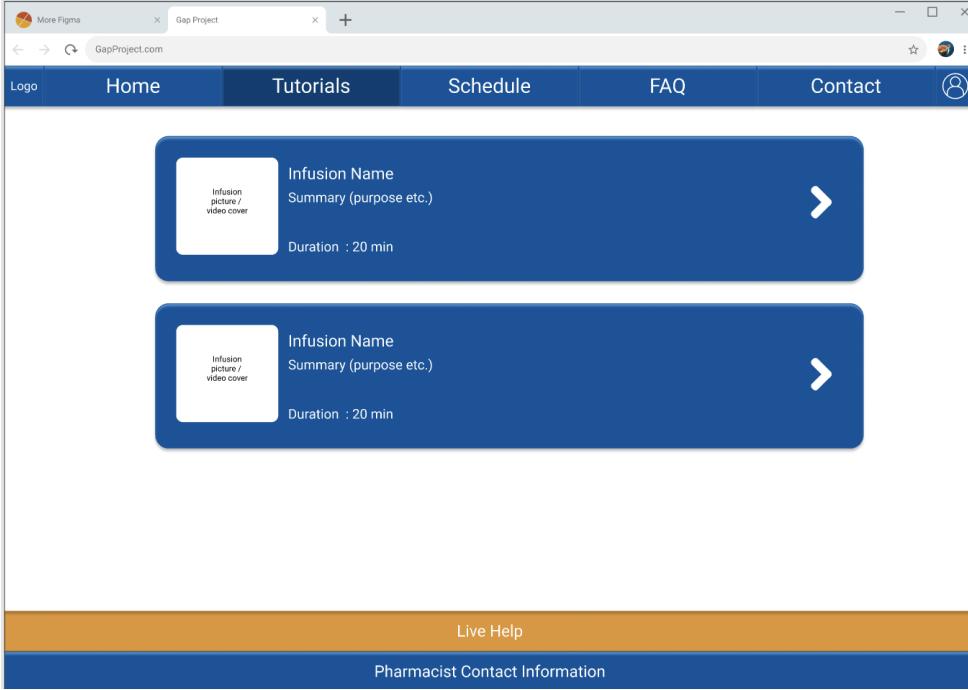
## Profile Page

Image	
Description	<p>Components</p> <ul style="list-style-type: none"><li>- Typography for Name &amp; Pharmacy Info</li><li>- Summary of Infusion (created here, not a base component)<ul style="list-style-type: none"><li>- MUI components TBD based on design</li></ul></li></ul>
Functionality	Displays patient overview including name, pharmacy, and infusion summary

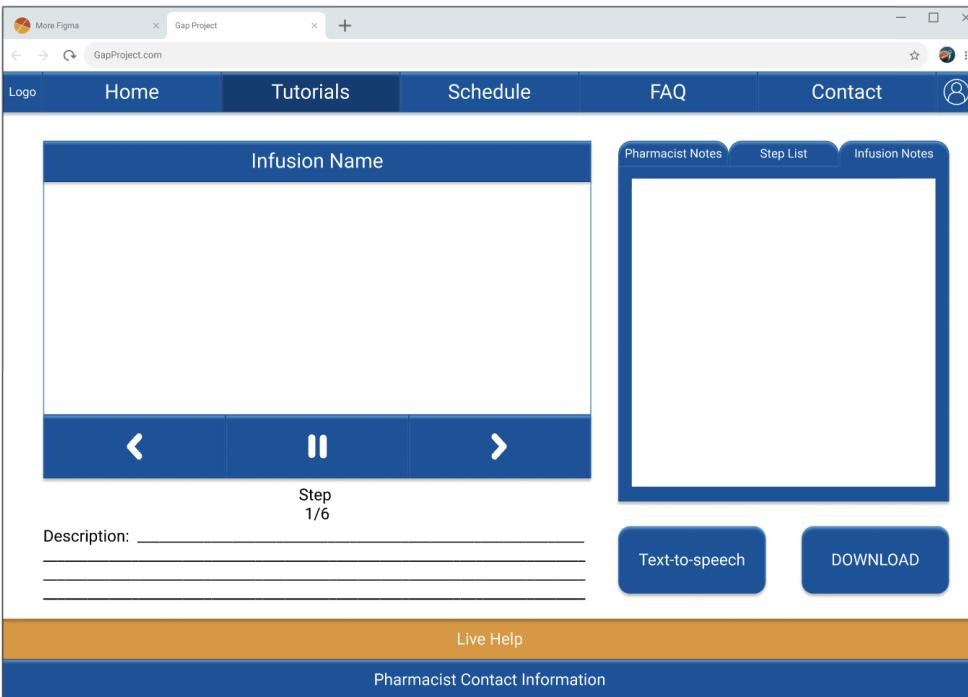
## Settings Page

Image	 A screenshot of a web browser window titled "Gap Project". The navigation bar includes "Logo", "Home", "Tutorials", "Schedule", "FAQ", "Contact", and a user icon. A blue header bar contains the text "Settings" and "Security". Below this is a form with three input fields: "Current Password" (placeholder: "*****"), "New Password" (placeholder: "*****"), and "Retype Password" (placeholder: "*****"). A blue "Save" button is centered below the inputs. At the bottom of the page are two horizontal bars: a yellow one labeled "Live Help" and a dark blue one labeled "Pharmacist Contact Information".
Description	Components: <ul style="list-style-type: none"><li>- Paper</li><li>- Typography</li><li>- Form Group</li><li>- Checkbox</li><li>- Button</li></ul>
Functionality	Loads saved information initially.  Clicking on edit icons allow a patient to update contact information, alert preferences, and their account password.

## Tutorial List

Image	
Description	<p>List of infusion cards assigned to a patient</p> <p>Components:</p> <ul style="list-style-type: none"><li>- Tutorial Listing</li></ul>
Functionality	Displays all assigned tutorials to a patient. Clicking on the arrow takes the patient to the respective tutorial.

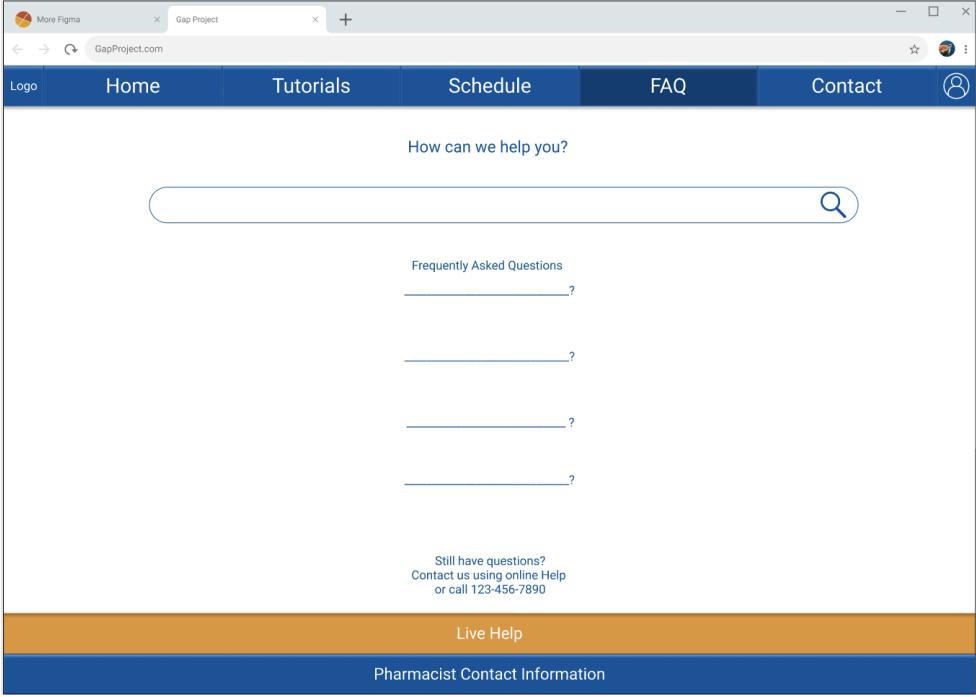
## Single Tutorial Page

Image	
Description	<p>View of a single tutorial page</p> <p>Components:</p> <ul style="list-style-type: none"><li>- Video Player</li><li>- Video Text Section</li><li>- Video Tabs<ul style="list-style-type: none"><li>- 3 tabs in one parent tab component</li></ul></li></ul>
Functionality	User Interface for controlling the tutorial process. Offers video, text, and supplementary info provided by the pharmacist.

## Schedule Page

Image	<p>The screenshot shows a web browser window with a dark blue header bar containing a logo, 'Home', 'Tutorials', 'Schedule' (which is highlighted in light blue), 'FAQ', 'Contact', and a user icon. Below the header is a navigation bar for the month of November, with arrows for previous and next months, and days 15 through 21 labeled. A large, rounded rectangular box highlights the 'Thursday' section. Inside this box, three time slots are listed: '6:30 AM Antibiotic Infusion', '2:30 PM Antibiotic Infusion', and '10:30 PM Antibiotic Infusion'. At the bottom of the page is a yellow 'Live Help' bar and a dark blue footer bar with the text 'Pharmacist Contact Information'.</p>
Description	<p><a href="https://material-ui.com/components/tabs/">https://material-ui.com/components/tabs/</a> (see scrollable tab)          Typography for the month with props          Daily Infusion Summary/Alarms Component</p> <p><a href="https://github.com/jquense/react-big-calendar">https://github.com/jquense/react-big-calendar</a>          Calendar component  <b>Examples:</b>  <a href="https://codesandbox.io/examples/package/react-big-calendar">https://codesandbox.io/examples/package/react-big-calendar</a></p>
Functionality	<p>Display schedule for a selected day in a set week          Schedule lists infusion time and medicine thawing time</p>

## FAQ Page

Image	
Description	<p>Input component. Example seen in <a href="https://material-ui.com/components/app-bar/">https://material-ui.com/components/app-bar/</a> (section “App Bar with Search Field”)</p> <p>FAQ section filled with props into the FAQ component if questions are changed dynamically</p>
Functionality	<p>Search page for patients to look up questions. Results will appear below.</p> <p>Frequently asked questions will appear below results? Or perhaps to the side?</p>

## Contact Information Page

Image	A wireframe of a 'Contact Us' page from a website called 'GapProject.com'. The page features a header with a logo, Home, Tutorials, Schedule, FAQ, and Contact links. Below the header is a section titled 'Contact Us' containing a placeholder for an 'Administration photo'. Underneath this is a 'Pharmacy' section and a 'Contact Information' section, both enclosed in blue-bordered boxes. The 'Pharmacy' section contains fields for 'Name: Pharmacy' and 'Location: 2345 N. RandStreet, Tustin Ave, CA 92618'. The 'Contact Information' section contains fields for 'Phone Number: 123-456-7890' and 'Email: 123456@.com'. At the bottom of the page are two buttons: a yellow 'Live Help' button and a blue 'Pharmacist Contact Information' button.
Description	Pharmacy contact and information page Components: <ul style="list-style-type: none"><li>- Img</li><li>- typography</li></ul>
Functionality	Displays a patient's pharmacy information

## Development Environment Setup

**Written by:** House of Innovation

**Edited by:** House of Innovation

**Last Revision:** Mar 02, 2021

## Credentials

**Ask your project manager at Innovation Lab for credentials.**

## Mongo DB

**Sign in with Google:**

Email: [innovationlab233@gmail.com](mailto:innovationlab233@gmail.com)

Password: lab1234!

**Database User (for connection)**

username: innovationlab

Password: 1234

Before following this guide, please ensure that you have a Github account, React, and Node.js, and npm package manager installed on your computer.

Steps to install node:

1. Go to <https://nodejs.org/en/>
2. install recommended version.
3. To confirm installation open terminal (command prompt for Windows)
4. Type command: node --version
5. To confirm npm installation: npm --version

## Database Setup

In order to connect to the database, you will need to login credentials (above) for MongoDB. After logging in, you will be greeted with the homepage of the Cluster. You can connect the database to MongoDB Compass and the project (already did) through the **Connect** button; and check your saved data through the **Collections** button. To add yourself to the database connections, click on **Network Access -> Add IP Address**. This will add your ip as a connection to the database, allowing you to access it through running the server locally.

## Client Setup

After cloning and pulling the master branch of the remote [repository](#), run the command **npm install** to download all the necessary packages. To start the client page, run the command **npm start**. The react application should set up the development server and open the website at localhost:3000.

# Server Setup

After cloning and pulling the master branch of the remote [repository](#), run the command **npm install** to download all the necessary packages. To start the server, run the command **nodemon server**. You should see **Server is running on port: 5000** and **MongoDB database connection established successfully** in your console.

You can test your API via **Insomnia**.

## Resources

### React JS

1. [Material-UI](#)
2. [Material-UI Icons](#)

### MongoDB

1. [MongoDB Crash Course](#)
2. [Install MongoDB Community Edition on macOS – MongoDB Manual](#)
3. [Collection Methods – MongoDB Manual](#)

### Test your API

1. [Insomnia | The API Design Platform and REST Client](#)

### Others

1. [Passport.js](#)
2. [bcrypt - npm](#)
3. [axios/axios: Promise based HTTP client for the browser and node.js](#)

# REST API Document

This API document is the backend development by the House of Innovation Team for INF4MATX 191. This API documentation further explains the routing GET and POST calls. All responses currently sent by the api follow raw JSON formatting for GET requests, HTTP status code 404 and 400 for invalid responses, 500 for any other server errors, and HTTP status 200 for successful POST requests made. (Chrome extension: **JSON Formatter** is recommended)

To access the current production API, the base URL for API requests is located at:

ⓘ localhost:5000



for example, in order to access all users in database:

---

← → ⌂ ⓘ localhost:5000/patients/all

---

For the development environment, the following information will be displayed in terminal during run-time:

- Server is running on port: 5000
- MongoDB database connection established successfully

# Patients

## GET Requests

Type	Route	Parameters	Description
GET	patients/all	None	return all patients in database

Code	Response Format [JSON]	Description
200	<pre>{   "role": "patient",   "notification": [],   "notificationType": "both",   "_id": "602abele792bfc56f79940a6",   "firstName": "tianxiong",   "lastName": "wu",   "phoneNumber": "6268332091",   "password": "\$2as10\$4105zpX80ogM/gol9K05.0JCz3meJWdy2bfmbcezCKQkgC0cPBQq",   "email": "tianxiongwu@hotmail.com",   "birthday": "2021-02-06T00:00:00.000Z",   "gender": "male",   "infusionArray": [],   "events": [],   "createdAt": "2021-02-15T18:31:58.574Z",   "updatedAt": "2021-02-15T18:31:58.574Z",   "__v": 0 },</pre>	
500		Catch all server side error responses.

Type	Route	Parameters	Description
GET	patients/:id	{_id: mongoose.Schema.Types.ObjectId}	Get a patient via its id

Code	Response Format [JSON]	Description

200	<pre>{   "role": "patient",   "notification": [],   "notificationType": "both",   "_id": "602abefc792bcfc56f79940a6",   "firstName": "tianxiong",   "lastName": "wu",   "phoneNumber": "6268332091",   "password": "\$2as10\$4105zPx80ogM/qo19K05.0JCz3meJWdy2bPmbcezCK0kgC0cPBQg",   "email": "tianxiongwu@hotmail.com",   "birthday": "2021-02-06T00:00:00.000Z",   "gender": "male",   "infusionArray": [],   "events": [],   "createdAt": "2021-02-15T18:31:58.574Z",   "updatedAt": "2021-02-15T18:31:58.574Z",   "__v": 0 },</pre>	
404	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\\" at path \"_id\" for model \"User\""	Invalid ID
500		Catch all server side error responses.

Type	Route	Parameters	Description
GET	patients/infusion/:id	{_id: mongoose.Schema.Types.ObjectId}	get a patient's infusion list via ID

Code	Response Format [JSON]	Description
200	<pre>[   ▶ { ... }, // 7 items   ▶ { ... }, // 7 items   ▶ { ... } // 7 items ]</pre>	return user's infusion list
404	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\\" at path \"_id\" for model \"User\""	Invalid ID
500		Catch all server side error responses.

Type	Route	Parameters	Description
GET	patients/notif ication/:id	{_id: mongoose.Schema.Types.ObjectId}	get a user's notification list via ID

Code	Response Format [JSON]	Description
200	<pre> 1 ▶ [ 2   "A", 3   "b" 4 ] </pre>	return user's notification list
404	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\" at path \"_id\" for model \"User\""	Invalid ID
500		Catch all server side error responses.

// NB: APIs below needs more testing

Type	Route	Parameters	Description
GET	patients/logout		logout

Code	Response Format [JSON]	Description
200		
400		
500		

//END

# POST Requests

Type	Route	Parameters	Description
POST	patients/register	{firstName: String, lastName: String, password: String, phoneNumber: String, email: String, birthday: Date, gender: enum [male, female, ohters], role: String -> default: patient}	patient register

Code	Response Format [JSON]	Description
200	<pre>▼{data: {…}, status: 200, statusText: "OK", headers: {…}, config: {…}, …}   ►config: {url: "http://localhost:5000/patients/register", method: "post", data: …   ►data: {role: "patient", notification: Array(0), notificationType: "both", _id: …   ►headers: {content-length: "436", content-type: "application/json; charset=utf-8"}   ►request: XMLHttpRequest {readyState: 4, timeout: 0, withCredentials: false, upl…   ►status: 200   ►statusText: "OK"   ►__proto__: Object</pre>	success
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	patients/login	{email: String, password: String}	patient login

Code	Response Format [JSON]	Description
200	<pre>▼{data: "OK", status: 200, statusText: "OK", headers: {…}, config: {…}, …}   ►config: {url: "http://localhost:5000/patients/login", method: "post", data: "…   ►data: "OK"   ►headers: {content-length: "2", content-type: "text/plain; charset=utf-8"}   ►request: XMLHttpRequest {readyState: 4, timeout: 0, withCredentials: false, upl…   ►status: 200   ►statusText: "OK"   ►__proto__: Object</pre>	success
404	<pre>● ▶ POST http://localhost:5000/patients/login 404 (Not Found)           inspector.js:44 Error: Request failed with status code 404                                loginForm.js:48   at createError (createError.js:16)   at settle (settle.js:17)   at XMLHttpRequest.handleLoad (xhr.js:62)</pre>	information failed to authenticate
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	patients/updateInfo/:id	<code>{_id: mongoose.Schema.Types.ObjectId, phoneNumber: String, email: String, birthday: Date}</code>	update user personal info

Code	Response Format [JSON]	Description
200	<pre>{   "checkinList": [],   "infusionType": [],   "notification": [],   "_id": "5ff6af2dc9a93b891866357a",   "firstName": "world",   "lastName": "hello",   "phoneNumber": "33123",   "email": "123456789@gmail.com",   "role": "intake",   "createdAt": "2021-01-07T06:53:06.003Z",   "updatedAt": "2021-01-08T07:12:15.476Z",   "__v": 0 }</pre>	return updated user JSON data
400	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\" at path \"_id\" for model \"User\""	Invalid id
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	patients/updateInfusion/:id	<code>{_id: mongoose.Schema.Types.ObjectId, infusionArray: Array}</code>	update user infusion info

Code	Response Format [JSON]	Description

200	<pre>{   "checkinList": [],   "infusionType": [],   "notification": [],   "_id": "5ff6af2c9a93b891866357a",   "firstName": "world",   "lastName": "hello",   "phoneNumber": "33123",   "email": "123456789@gmail.com",   "role": "intake",   "createdAt": "2021-01-07T06:53:06.003Z",   "updatedAt": "2021-01-08T07:12:15.476Z",   "__v": 0 }</pre>	return updated user JSON data
400	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\" at path \"_id\" for model \"User\""	Invalid id
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	patients/updateEvents/:id	{_id: mongoose.Schema.Types.ObjectId, events: Array}	update user event

Code	Response Format [JSON]	Description
200	<pre>{   "checkinList": [],   "infusionType": [],   "notification": [],   "_id": "5ff6af2c9a93b891866357a",   "firstName": "world",   "lastName": "hello",   "phoneNumber": "33123",   "email": "123456789@gmail.com",   "role": "intake",   "createdAt": "2021-01-07T06:53:06.003Z",   "updatedAt": "2021-01-08T07:12:15.476Z",   "__v": 0 }</pre>	return updated user JSON data
400	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\" at path \"_id\" for model \"User\""	Invalid id
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	patients/updateNotification/:id	{_id: mongoose.Schema.Types.ObjectId, notification: Array{}}	update user notification

Code	Response Format [JSON]	Description
200	{ "checkinList": [], "infusionType": [], "notification": [], "_id": "5ff6af2c9a93b891866357a", "firstName": "world", "lastName": "hello", "phoneNumber": "33123", "email": "123456789@gmail.com", "role": "intake", "createdAt": "2021-01-07T06:53:06.003Z", "updatedAt": "2021-01-08T07:12:15.476Z", "__v": 0 }	return updated user JSON data
400	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\" at path \"_id\" for model \"User\""	Invalid id
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	patients/updateCheckin/:id	{_id: mongoose.Schema.Types.ObjectId, nextCheckIn: Date{}}	update user next checkin time

Code	Response Format [JSON]	Description

200	<pre>{   "checkinList": [],   "infusionType": [],   "notification": [],   "_id": "5ff6af2c9a93b891866357a",   "firstName": "world",   "lastName": "hello",   "phoneNumber": "33123",   "email": "123456789@gmail.com",   "role": "intake",   "createdAt": "2021-01-07T06:53:06.003Z",   "updatedAt": "2021-01-08T07:12:15.476Z",   "__v": 0 }</pre>	return updated user JSON data
400	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\" at path \"_id\" for model \"User\""	Invalid id
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	patients/updateRecentCheckin/:id	{ _id: mongoose.Schema.Types.ObjectId, recentCheckIn: Date}	update user recent checkin time

Code	Response Format [JSON]	Description
200	<pre>{   "checkinList": [],   "infusionType": [],   "notification": [],   "_id": "5ff6af2c9a93b891866357a",   "firstName": "world",   "lastName": "hello",   "phoneNumber": "33123",   "email": "123456789@gmail.com",   "role": "intake",   "createdAt": "2021-01-07T06:53:06.003Z",   "updatedAt": "2021-01-08T07:12:15.476Z",   "__v": 0 }</pre>	return updated user JSON data
400	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\" at path \"_id\" for model \"User\""	Invalid id
500		Catch all server side error responses.

# Admin

## GET Requests

// NB: APIs below needs more testing

Type	Route	Parameters	Description
GET	patients/logout		logout

Code	Response Format [JSON]	Description
200		
400		
500		

// END

## POST Requests

Type	Route	Parameters	Description
POST	admin/register	{firstName: String, lastName: String, password: String, role: String -> default: admin}	admin register

Code	Response Format [JSON]	Description
200		success

500		Catch all server side error responses.
-----	--	--

Type	Route	Parameters	Description
POST	admin/login	{email: String, password: String}	admin login

Code	Response Format [JSON]	Description
200		success
404		information failed to authenticate
500		Catch all server side error responses.

## Pharmacists

### GET Requests

Type	Route	Parameters	Description
GET	pharmacists/ all	None	return all pharmacists in database

Code	Response Format [JSON]	Description
------	------------------------	-------------

200	<pre>  "v": 1,     "v": {       "assignedPatient": [],       "role": "pharmacist",       "_id": "603aac61062b6e0ff72464eb",       "firstName": "adf",       "lastName": "asd",       "password": "\$2as10\$Fx6ubojk23mC35sCQ7ARRuYeaSqmz90.nOZsnw3246GJF5ut3lI8q",       "phoneNumber": "6268332091",       "email": "363221244@gg.com",       "birthday": "2021-02-04T00:00:00.000Z",       "createdAt": "2021-02-27T20:32:33.559Z",       "updatedAt": "2021-02-27T20:32:33.559Z",       "__v": 0     }   }</pre>	
500		Catch all server side error responses.

Type	Route	Parameters	Description
GET	pharmacists/:id	{_id: mongoose.Schema.Types.ObjectId}	Get a pharmacist via its id

Code	Response Format [JSON]	Description
200	<pre>  "v": 1,     "v": {       "assignedPatient": [],       "role": "pharmacist",       "_id": "603aac61062b6e0ff72464eb",       "firstName": "adf",       "lastName": "asd",       "password": "\$2as10\$Fx6ubojk23mC35sCQ7ARRuYeaSqmz90.nOZsnw3246GJF5ut3lI8q",       "phoneNumber": "6268332091",       "email": "363221244@gg.com",       "birthday": "2021-02-04T00:00:00.000Z",       "createdAt": "2021-02-27T20:32:33.559Z",       "updatedAt": "2021-02-27T20:32:33.559Z",       "__v": 0     }   }</pre>	
404	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\" at path \"_id\" for model \"User\""	Invalid ID
500		Catch all server side error responses.

Type	Route	Parameters	Description
GET	pharmacists/:id	{_id: mongoose.Schema.Types.ObjectId}	Get a pharmacist via its id

Code	Response Format [JSON]	Description
200	<pre>{   "assignedPatient": [],   "role": "pharmacist",   "_id": "603aac61062b6e0ff72464eb",   "firstName": "adif",   "lastName": "asd",   "password": "\$2a\$10\$Fx6u8ojk23mC35eCQ7ARRuYEaSqmz9O.n0Zenw3246GJF5ut3lI8q",   "phoneNumber": "6268332091",   "email": "363221244@qq.com",   "birthday": "2021-02-04T00:00:00.000Z",   "createdAt": "2021-02-27T20:32:33.559Z",   "updatedAt": "2021-02-27T20:32:33.559Z",   "__v": 0 }</pre>	
404	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\\" at path \"_id\" for model \"User\""	Invalid ID
500		Catch all server side error responses.

Type	Route	Parameters	Description
GET	pharmacists/assignedPatient/:id	{_id: mongoose.Schema.Types.ObjectId}	Get assigned patient array via pharmacist' id

Code	Response Format [JSON]	Description
200		
404	"Error: CastError: Cast to ObjectId failed for value \"<SOME ID>\\" at path \"_id\" for model \"User\""	Invalid ID
500		Catch all server side error responses.

// NB: APIs below needs more testing

Type	Route	Parameters	Description
GET	pharmacists/logout		logout

Code	Response Format [JSON]	Description
200		
400		
500		

//END

## POST Requests

Type	Route	Parameters	Description
POST	pharmacists/register	{firstName: String, lastName: String, password: String, phoneNumber: String, email: String, birthday: Date, gender: enum [male, female, others], role: String -> default: pharmacist}	pharmacist register

Code	Response Format [JSON]	Description
200	<pre>signUpForm.js:115 &lt; (data: {}, status: 200, statusText: "OK", headers: {}, config: {}, &lt;) &gt; config: {url: "http://localhost:5000/pharmacists/register", method: "post", data: {"firstName": "John", "lastName": "Doe", "password": "123456", "phoneNumber": "123-456-7890", "email": "john.doe@example.com", "birthday": "1990-01-01", "gender": "male", "role": "pharmacist"}, headers: {content-length: "368", content-type: "application/json; charset=utf-8"}}, &lt; &gt; XMLHttpRequest (readystate: 4, timeout: 0, withCredentials: false, upload: XMLHttpRequestUpload, status: 200, statusText: "OK", &lt;proto&gt;: Object)</pre>	SUCCESS
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	pharmacists/login	{email: String, password: String}	pharmacist login

Code	Response Format [JSON]	Description
200	<pre> {   assignedPatient: Array(0), role: "pharmacist", _id: "603aac61062b6e0ff72464eb", firstName: "asd",   lastName: "asd", __proto__: Object } + assignedPatient: [] + birthdate: "1990-04-08T00:00:00.000Z" + createdAt: "2021-02-27T20:32:33.559Z" + email: "363221244@qq.com" + firstName: "asd" + lastName: "asd" + password: "\$2a\$10\$fx6u8o)k2mC35sC07ARRuYe5qnz90.n0Zsnw3Z46GJF5ut3l1Bq" + phoneNumber: "626833232091" + role: "pharmacist" + updatedAt: "2021-02-27T20:32:33.559Z" + __v: 0 + __id: "603aac61062b6e0ff72464eb" + __proto__: Object </pre>	SUCCESS
404		information failed to authenticate
500		Catch all server side error responses.

## Events

### GET Request

Type	Route	Parameters	Description
GET	events/all	None	fetch all event objects

Code	Response Format [JSON]	Description

200	<pre>[{"description": "this is a test event", "_id": "603d45b42474b1395cb13dac", "title": "test event", "start": "2020-01-01T08:00:00.000Z", "end": "2020-12-20T08:00:00.000Z", "notifyAt": "2020-12-19T08:00:00.000Z", "__v": 0}, {"description": "this is a test event 2", "_id": "603d45e92474b1395cb13dad", "title": "test event 2", "start": "2021-01-01T08:00:00.000Z", "end": "2021-02-20T08:00:00.000Z", "notifyAt": "2021-01-19T08:00:00.000Z", "__v": 0}],</pre>	success
500		Catch all server side error responses.

## POST Requests

Type	Route	Parameters	Description
POST	events/add	{title: String, start: Date, end, Date, notifyAt: Date, description: String}	add an event

Code	Response Format [JSON]	Description
------	------------------------	-------------

200	<pre>{   "description": "this is a test event 2",   "_id": "603d4a2d2474b1395cb13db2",   "title": "test event 2",   "start": "2021-01-01T08:00:00.000Z",   "end": "2021-02-20T08:00:00.000Z",   "notifyAt": "2021-01-19T08:00:00.000Z",   "__v": 0 }</pre>	
400	<pre>"Error: ValidationError: notifyAt: Cast to date failed for value \"NaN\" at path \"notifyAt\""</pre>	type errors
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	events/delete/:id	{_id: mongoose.Schema.Types.ObjectId}	delete an event

Code	Response Format [JSON]	Description
200	<pre>"deleted"</pre>	
400	<pre>"Error: CastError: Cast to ObjectId failed for value &lt;ID&gt; at path \"_id\" for model \"events\""</pre>	invalid id
500		Catch all server side error responses.

## Notifications

### GET Request

Type	Route	Parameters	Description
GET	notifications/all	None	fetch all notification objects

Code	Response Format [JSON]	Description
200	<pre>[{"description": "this is a test notification",   "_id": "603d49202474b1395cb13db0",   "title": "test notification 1",   "notifyAt": "2021-01-19T19:00:00.000Z",   "__v": 0},  {"description": "this is a test notification 2",   "_id": "603d49662474b1395cb13db1",   "title": "test notification 2",   "notifyAt": "2021-10-26T22:00:00.000Z",   "__v": 0}]</pre>	
500		Catch all server side error responses.

## POST Requests

Type	Route	Parameters	Description
POST	notifications/add	{title: String, notifyAt: Date, description: String}	add a notification

Code	Response Format [JSON]	Description
200	<pre>{"description": "this is a test notification 2",   "_id": "603d49662474b1395cb13db1",   "title": "test notification 2",   "notifyAt": "2021-10-26T22:00:00.000Z",   "__v": 0}</pre>	
400	<pre>Error: ValidationError: notifyAt: Cast to date failed for value \"NaN\" at path \"notifyAt\""</pre>	type errors

500		Catch all server side error responses.
-----	--	--

Type	Route	Parameters	Description
POST	notifications/d elete/:id	{_id: mongoose.Schema.Types.ObjectId}	delete an event

Code	Response Format [JSON]	Description
200	"deleted"	
400	"Error: CastError: Cast to ObjectId failed for value <ID> at path \"_id\" for model \"events\""	invalid id
500		Catch all server side error responses.

## FAQ

### GET Requests

Type	Route	Parameters	Description
GET	faqs/all	None	fetch all FAQ objects

Code	Response Format [JSON]	Description
------	------------------------	-------------

200	<pre> 1 [ 2   { 3     "_id": "600f8eaa3349d7aa4file46", 4     "question": "Why are infusions performed at home?", 5     "answer": "The technological advances that enabled safe and effective administration of infusion therapies in the home, the desire of patients to resume normal lifestyles and work activities while recovering from illness, and the cost-effectiveness of home care are important.", 6     "__v": 0 7   }, 8   { 9     "_id": "600f8eaa3349d7aa4file46", 10    "question": "How often does a nurse visit in the home and what will they do to help?", 11    "answer": "Infusion nurses will visit in the beginning of your course of treatment and at intervals during your course of treatment. They typically provide an evaluation and assessment, education and training for the patient or caregiver, and home assessment for aseptic administration. The nurse will observe and/or your caregiver's ability to administer your medication so you can give your own IV doses at the prescribed times.", 12    "__v": 0 13  }, 14  { 15    "_id": "600f8eaa3349d7aa4file47", 16    "question": "What do I do if I can't attach the pump to my IV/PICC line?", 17    "answer": "If you are unable to access your IV line, contact the Infusion Pharmacy for direction. It is important to not miss a dose of your medication. The pharmacy will be able to connect you with a pharmacist or nurse to assist you in problem solving so you can administer your medication.", 18    "__v": 0 19  }, 20  { 21    "_id": "600f8eaa3349d7aa4file470", 22    "question": "Does it matter if my medication is cold when I administer it?", 23    "answer": "The medication should not be cold. You will be asked to remove a dose from the refrigerator at least 30 minutes before it's time to administer to bring the medication to room temperature." 24    "__v": 0 25  } ] </pre>	
500		Catch all server side error responses.

## POST Requests

Type	Route	Parameters	Description
POST	faqs/add	{question: String, answer: String}	add a faq

Code	Response Format [JSON]	Description
200	<pre> 1 { 2   "_id": "603f17792474b1395cb13db3", 3   "question": "this is a test faq", 4   "answer": " this is a test faq", 5   "__v": 0 6 } </pre>	
400		type errors
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	faqs/delete/:id	{_id: mongoose.Schema.Types.ObjectId}	delete a faq

Code	Response Format [JSON]	Description
200	{"deleted": true}	
400	"Error: CastError: Cast to ObjectId failed for value <ID> at path \"_id\" for model \"faqs\""	invalid id
500		Catch all server side error responses.

## Pharmacy

### GET Request

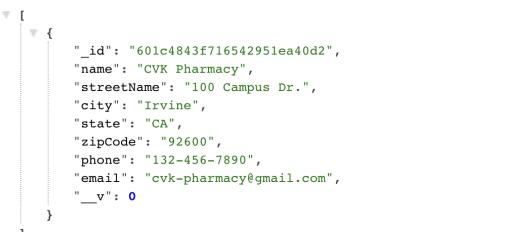
Type	Route	Parameters	Description
GET	pharmacy/all	None	fetch all Pharmacy objects

Code	Response Format [JSON]	Description
200	[{"_id": "601c4843f716542951ea40d2", "name": "CVK Pharmacy", "streetName": "100 Campus Dr.", "city": "Irvine", "state": "CA", "zipCode": "92600", "phone": "132-456-7890", "email": "cvk-pharmacy@gmail.com", "__v": 0}]	
500		Catch all server side error

		responses.
--	--	------------

## POST Request

Type	Route	Parameters	Description
POST	pharmacy/add	{name: String, streetName: String, city: String, state: String, zipCode: String, phone: String, email: String}	add a pharmacy

Code	Response Format [JSON]	Description
200	 A JSON object representing a single pharmacy record. It includes fields: _id, name, streetName, city, state, zipCode, phone, email, and __v. The __v field is highlighted in blue. <pre>{   "_id": "601c4843f716542951ea40d2",   "name": "CVK Pharmacy",   "streetName": "100 Campus Dr.",   "city": "Irvine",   "state": "CA",   "zipCode": "92600",   "phone": "132-456-7890",   "email": "cvk-pharmacy@gmail.com",   "__v": 0 }</pre>	
400		type errors
500		Catch all server side error responses.

Type	Route	Parameters	Description
POST	pharmacy/upda teInfo/:id	{_id: mongoose.Schema.Types.ObjectId}	update a pharmacy

Code	Response Format [JSON]	Description
------	------------------------	-------------

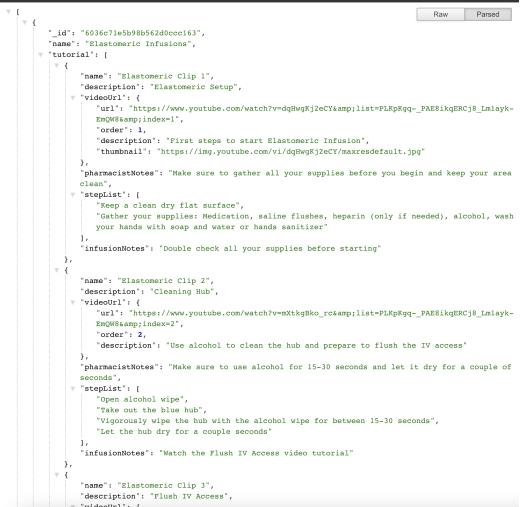
200	<pre>   [     {       "_id": "601c4843f716542951ea40d2",       "name": "CVK Pharmacy",       "streetName": "100 Campus Dr.",       "city": "Irvine",       "state": "CA",       "zipCode": "92600",       "phone": "132-456-7890",       "email": "cvk-pharmacy@gmail.com",       "__v": 0     }   ] </pre>	return updated pharmacy
400	"Error: CastError: Cast to ObjectId failed for value <ID> at path \"_id\" for model \"faqs\""	invalid id
500		Catch all server side error responses.

## Tutorials

### GET Request

Type	Route	Parameters	Description
GET	tutorials/all	None	fetch all Pharmacy objects

Code	Response Format [JSON]	Description
------	------------------------	-------------

200		
500		Catch all server side error responses.

## POST Requests

Type	Route	Parameters	Description
POST	tutorials/add	{name: String, description: String, duration: String, tutorials: tutorialSchema*}	add a tutorial object  *See tutorial.model.js for details

Code	Response Format [JSON]	Description
200		return added tutorial object
400		type errors
500		Catch all server side error responses.