

# Application Manual

## CDC Post-Discharge Stroke Data Reporting App

### Team Health Informaniacs

Fayaz Adam  
Vinh Nguyen  
Anirrudh Saddi  
Gurpreet Singh

Source: <https://github.gatech.edu/gt-hit-fall2016/CDC-Post-Discharge>

## Table of contents

Sr. No.	Title	Page #
1	Project Background	2
2	Architecture Overview	3
3	Design Decisions	5
4	Startup Instructions	6
5	Usages of Stroke Service App	14
6	Source Code	21
7	Acknowledgements	21

### I. Project Background

To get an adequate understanding of which events are more likely to lead to readmission, the Coverdell stroke registry project at CDC is interested in collecting data once the patient transitions from the hospital to their next care setting (e.g. home, inpatient rehabilitation, nursing home). Automated identification of eligible stroke cases, chart abstraction of data elements from EHR and write back of post discharge data into EHR can improve the process by bring consistency, reduce re-entry of data and increase quality of the data reported.

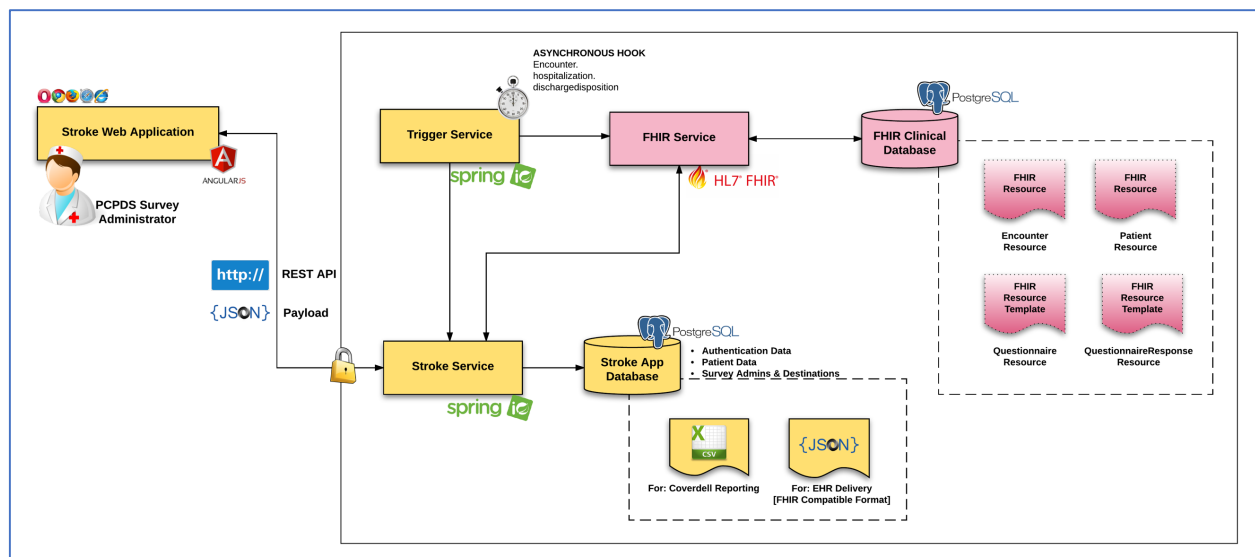
Therefore, the main of objective of this proof of concept is to create a Post-Discharge Stroke web app and supporting services to demonstrate SMART on FHIR can be used in a stroke patient follow-up scenario.

## II. Architecture Overview

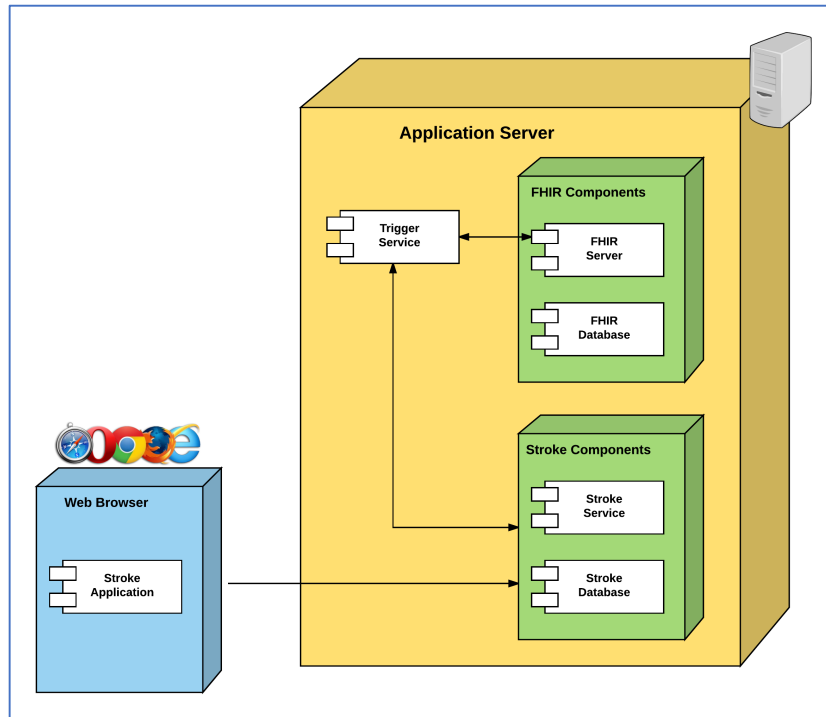
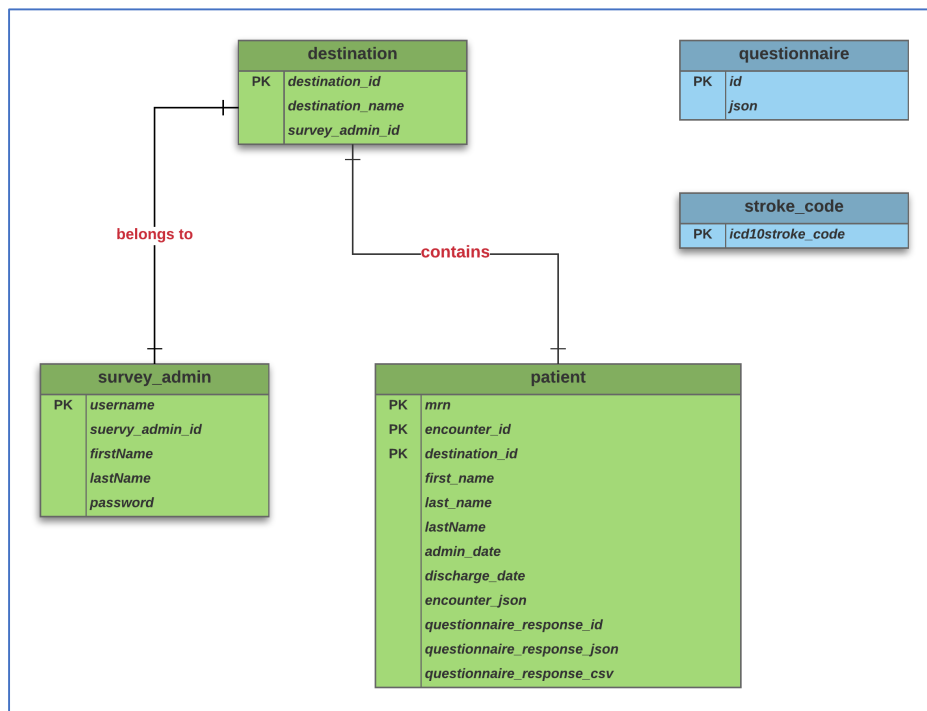
The PDS app will be used by PCPDS Survey Administrators, who collect 30 day post discharge information.

The system consists of the following components:

- FHIR clinical database with FHIR service: Contains all the patient data, and is where the PDSApp will store questionnaires and their responses.
- Trigger Service: Asynchronous trigger which will query the FHIR service to determine if a patient needs to be monitored post discharge for stroke.
- Stroke Web App: The front end of the PDS App which provides an interface for Survey Administrators to view patients that need to be surveyed and conduct surveys.
- Stroke Service: REST interface used by the Trigger Service and Stroke Web App for getting and loading FHIR resources into the FHIR service and getting and loading resources into the PDS App local database.



**Figure:** Architecture Diagram

**Figure:** Component Diagram**Figure:** Logical Data Diagram

### III. Design Decisions

The following decisions and assumptions were made in consultation with our OM (organization mentor).

1. OAUTH2 disabled: To quickly bring up a FHIR sandbox environment and prototype application, the team disabled oauth2 for the FHIR server. There is nothing in the application which would prevent it from supporting OAUTH2.
2. Only ICD-10 codes will be used for identifying stroke patients. The ICD-10 codes are provided by the CDC.
3. Patients who are re-admitted to the hospital for stroke conditions during the 30-day window of the initial discharge will have a unique entry for each encounter.
4. Because of limitations with CDS-Hooks unable to run asynchronously, we decided to use a custom trigger which polls the FHIR service.
5. For an improved user experience, certain questions were grouped together from the given stroke survey.
6. The data contained in the exported CSV will have the group text, linkId, question text and answer text only for the answered questions, since a format was not provided by State Department of Health and CDC.
7. The following resources are saved locally in the PDS App

- a. Questionnaire: The provided survey is created and saved in the PDS App db and pushed to the FHIR service. However, it is only accessed in the app through the local db.
  - b. QuestionnaireResponse: Each patient's response is created and saved in the PDS App db and pushed to the FHIR service. However, it is only accessed in the app through the local db.
  - c. Location: The locations each Survey Administrator is responsible for are saved locally in the PDS App.
8. To test the PDS App synthetic data was generated to simulate stroke patients and loaded into the FHIR service using local tools available in the smart-on-fhir installer.
9. An Admin user for the PDS App was generated to perform administrative duties for the application. For example, uploading the Questionnaire into the application.

#### IV. Startup Instructions

CDC Post-Discharge Stroke Data Reporting App (aka PDS App) is deployed as two Java JARs. They can be deployed on the same server or different servers if there is accessible network path between the two. This app also works in conjunction with FHIR server.

- **triggerservice** - For querying FHIR resources asynchronously for stroke patients
- **strokeservice** - REST interface and web application for hosting survey and saving to FHIR

To deploy the sandbox environment, begin by setting up the FHIR sandbox environment using the instructions located here: <https://github.com/smart-on-fhir/installer>. For the purposes of our GaTech deliverable, a link with a ready VirtualBox appliance will be provided in the shared folder below.

[https://drive.google.com/drive/folders/0Bzoh1sBtLYo\\_TEdLTGJCRHEydUk](https://drive.google.com/drive/folders/0Bzoh1sBtLYo_TEdLTGJCRHEydUk)

Once the FHIR sandbox environment is downloaded, which runs on Ubuntu 14.04, the following steps are required.

### a) Set up environment and synthetic data

#### 1. If JDK version is not 1.8 or later, install JDK1.8

```
sudo add-apt-repository ppa:webupd8team/java -y
sudo apt-get update
sudo apt-get install oracle-java8-installer
```

Set JAVA\_HOME:

```
export JAVA_HOME=/usr/lib/jvm/java-8-oracle
```

#### 2. Setup the backend PostgreSQL database for the application

##### a. Create DB User Id and Password:

```
student@student:~$ sudo -u postgres psql
[sudo] password for student:
psql (9.3.14)

postgres=# ALTER USER postgres PASSWORD 'password';
ALTER ROLE
```

##### b. Create StrokeApp Database:

```
student@student:~$ sudo -u postgres psql
[sudo] password for student:
psql (9.3.14)

postgres=# CREATE DATABASE strokeapp;
```

### 3. Turn off OAuth2 on SMART-on-FHIR and start FHIR service

#### a. Edit main.yml:

```
cd ~/installer/provisioning/roles/common/defaults/
sudo vi main.yml
# Specific settings for the SMART-on-FHIR server authorization
behavior
# Default: fhir_server_use_auth: true. Change true to false to
turn off OAuth
fhir_server_use_auth: false
```

#### b. Save changes in main.yml and restart the FHIR service:

```
sudo ansible-playbook -c local -i 'localhost,' -vvvv \ smart-on-
fhir-servers.yml
```

### 4. Load synthetic data to PDS App. This step uses a custom script for loading sample patients who contain Encounters and/or Conditions which meet the stroke criteria (using given ICD-10 codes). See readme in "generated\_data"

```
cd ~/CDC-Post-Discharge/generated_data
sudo su root ./addStrokePatients.sh
```

### 5. Configuration Parameters for **strokeservice**. Verify these values match database configurations from step 2.

#### **Property file:**



```
~/CDC-Post-Discharge/strokeservice/src/main/resources/application.properties

# =====
# [ Server Configuration Section ]
# =====

server.port = 8888

# =====
# [ Database Configuration Section ]
# =====

spring.jpa.database = POSTGRESQL
spring.jpa.hibernate.ddl-auto = update

spring.datasource.platform = postgres
spring.datasource.url =
jdbc:postgresql://localhost:5432/strokeapp
spring.datasource.username = postgres
spring.datasource.password = password

# =====
# [ Application Configuration Section ]
# =====

admin.password = password
fhirService.baseUrl = http://localhost:9080/
```

6. Configuration Parameters for **triggerservice**. Verify these values match database configurations from step 2.

**Property file:**

```
~/CDC-Post-Discharge/triggerservice/src/main/resources/application.properties

# =====
# [ Server Configuration Section ]
# =====

server.port = 8887
```

```
# =====  
# [ Stroke Service Configuration Section ]  
# =====  
strokeService.baseUrl = http://localhost:8888/  
  
# =====  
# [ FHIR Service Configuration Section ]  
# =====  
fhirService.baseUrl = http://localhost:9080/  
initialFetchDays = -1
```

## b) Starting Application

The PDS App can be started in two ways, using maven (requires maven to be installed) or with java jars.

To start the application with maven:

### 1. Run **strokeservice**

```
cd ~/CDC-Post-Discharge/strokeservice  
mvn spring-boot:run
```

Optionally, pass the `-Dadmin.password=<password>` property to specify the password for the 'admin' user to log in to the Admin UI. For example:

```
cd ~/CDC-Post-Discharge/strokeservice  
mvn spring-boot:run -Dadmin.password=123456
```

### 2. Run **triggerservice**

*Note: strokeservice is REQUIRED to be running before starting triggerservice*

In a new terminal:

```
mvn spring-boot:run
```

Optionally, pass the `-DinitialFetchDays=<days>` property to specify the number of days back to use when querying patients who have been discharged for stroke conditions.

```
cd ~/CDC-Post-Discharge/triggerservice
mvn spring-boot:run -DinitialFetchDays=90
```

To start the application with the JAVA jars:

### 1. Run **strokeservice**

```
cd ~/CDC-Post-Discharge/strokeservice
java -jar ./target/strokeservice.0.1-SNAPSHOT.jar
```

OR

```
cd ~/CDC-Post-Discharge/strokeservice
java -jar ./target/strokeservice-0.0.1-SNAPSHOT.jar \
-Dadmin.password=123456
```

### 2. Run **triggerService**

Note: strokeservice is REQUIRED to be running before starting triggerservice

In a new terminal:

```
cd ~/CDC-Post-Discharge/triggerservice
java -jar ./target/triggerservice.0.1-SNAPSHOT.jar
```

OR

```
cd ~/CDC-Post-Discharge/triggerservice
java -jar ./target/triggerservice.0.1-SNAPSHOT.jar -
DinitialFetchDays=90
```

To start the application with the JAVA jars:

## 1. Run **strokeservice**

```
java -jar ./target/strokeservice.0.1-SNAPSHOT.jar
```

OR

```
java -jar ./target/strokeservice-0.0.1-SNAPSHOT.jar \  
-Dadmin.password=123456
```

## 2. Run **triggerService**

```
cd ~/CDC-Post-Discharge/triggerService  
java -jar ./target/triggerService.0.1-SNAPSHOT.jar
```

OR

```
cd ~/CDC-Post-Discharge/triggerService  
java -jar ./target/triggerService.0.1-SNAPSHOT.jar -  
DinitialFetchDays=90
```

## c) Verify Applications Running

1. **strokeservice**, which hosts the web application, will listen on port 8888. Verify the application is up and running:  
<http://localhost:8888/>
2. **triggerService**, runs asynchronously in the background and queries the FHIR server nightly at 1 AM.

## d) Running the Application

1. In a web browser (Chrome/Firefox), navigate to URL "http://localhost:8888" where our stroke service resides.
2. Login in as admin using the credentials

**user:** admin

**password:** password (default) or the password entered while initializing the stroke service.

3. If there is a questionnaire present, logout. If not, click on the upload button, navigate to the questionnaire (CDC-Post-Discharge\strokeservice\src\test\resources\data\stroke-questionnaire.json) to be used for collecting 30 days post discharge questionnaire. Logout once the upload is complete.

4. Now, login as a survey administrator using the credentials

**username:** nurse\_jane

**password:** password

You will now see a list of patients assigned to Nurse Jane. The questionnaire column will display a question icon for those patients for whom the questionnaire needs to be completed or the response column displays the response ID of responses completed for the patients.

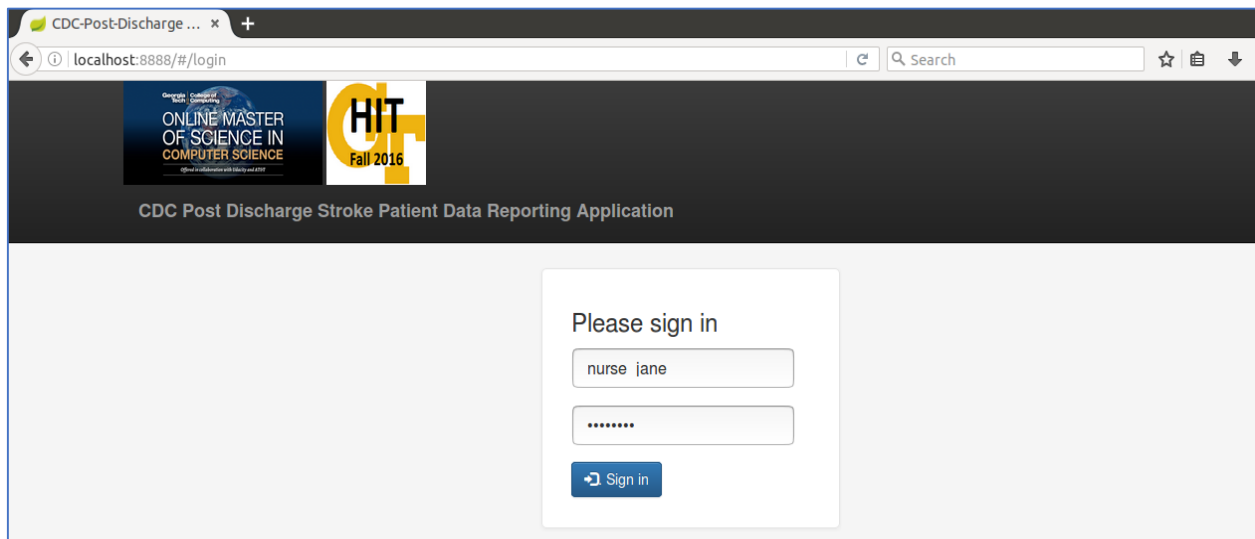
5. You can sort the table on patient first name or days since discharge by clicking on the column header.
6. To complete the questionnaire, click on the question icon next to the patient, it will route you to a questionnaire page. Enter as many applicable fields as you would like and at the top, click the Submit button to submit it, or Cancel button to return home.
7. Once submitted, the app will route you back to home with a response ID next to the patient for whom the questionnaire has just been completed.

- a. To view the JSON response, you can click on the response ID. Here you can download the response either as a .json file or a .csv file by clicking either of the Download button.
- b. You can click on the response ID again to hide the response.

## 8. Logout

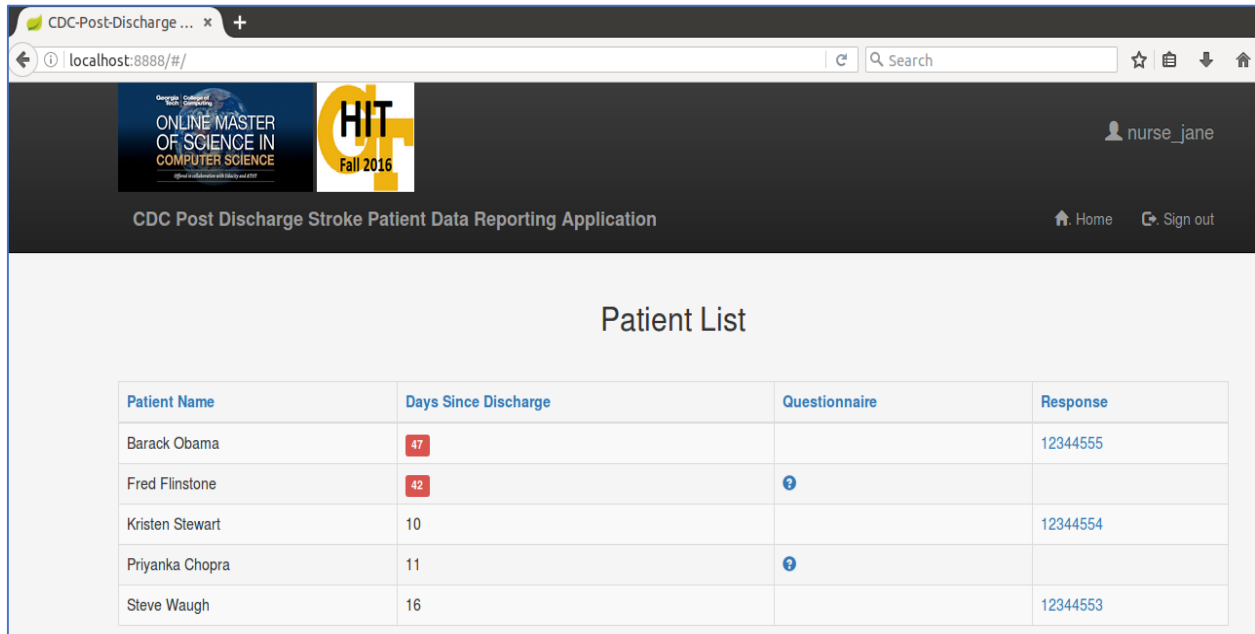
## V. Usages of Stroke Service App

**Note:** Displayed data in all following screenshots are mockup data for demonstration purposes.



**Screenshot:** Survey Administrator Login Page

Both the Survey Administrator and the PDS App Administrators may log in from the initial landing page.



Patient Name	Days Since Discharge	Questionnaire	Response
Barack Obama	47		<a href="#">12344555</a>
Fred Flinstone	42	<a href="#">?</a>	
Kristen Stewart	10		<a href="#">12344554</a>
Priyanka Chopra	11	<a href="#">?</a>	
Steve Waugh	16		<a href="#">12344553</a>

**Screenshot:** Patient List page

After logging in, the Survey Administrator, in this case nurse\_jane (as shown in the top left), will be presented with patients who transitioned post-discharge to the location the Survey Administrator is responsible for (Hospital, Long Term Care, etc). The screen also displays for how many days the patient has been discharged and highlights patients who were discharged more than 30 days ago, indicating they are ready to be surveyed. Additionally, the page provides a link to open the Questionnaire and conduct the survey, and if the survey has been conducted, a link to the response to download as JSON or CSV.

The screenshot shows a web browser window with the address bar displaying 'localhost:8888/#/questionnaire'. The page header includes logos for 'ONLINE MASTER OF SCIENCE IN COMPUTER SCIENCE' and 'HIT Fall 2016', along with the user 'nurse\_jane' and navigation links for 'Home' and 'Sign out'. The main title is 'CDC Post Discharge Stroke Patient Data Reporting Application'. Below this, a patient selection bar shows '[Selected Patient] Fred Flinstone [MRN #] 99900020 [Encounter ID] 1055 [Destination ID] 1' with 'Submit' and 'Cancel' buttons. Below the patient bar, the title '30 day Stroke Followup Questionnaire' is centered. The form contains two input fields: 'Patient's date of hospital discharge' with the value '2016-10-25' and 'Hospital Admission Date (part of in-hospital dataset)' with the value '2016-10-18'.

**Screenshot:** Questionnaire page

When nurse\_jane clicks on (?) to open the Questionnaire for patient Fred Flinstone, nurse\_jane is presented with the 30 day stroke follow up survey. The survey is pre-populated with Patient data when available. In the figure above, the admission and discharge dates are pre-populated.



The screenshot shows a web browser window with the address bar displaying 'localhost:8888/#/questionnaire'. The page header includes a logo for 'ONLINE MASTER OF SCIENCE IN COMPUTER SCIENCE' and 'HIT Fall 2016'. The user is logged in as 'nurse\_jane'. The main title of the application is 'CDC Post Discharge Stroke Patient Data Reporting Application'. The page contains two sections: 'Call placed' and 'Location of patient'. The 'Call placed' section has a question 'Was patient called for follow up post discharge?' with a dropdown menu. The 'Location of patient' section has two questions: 'Where is the patient at the time of follow-up call?' and 'Where is the patient 30 days after discharge?', each with a dropdown menu.

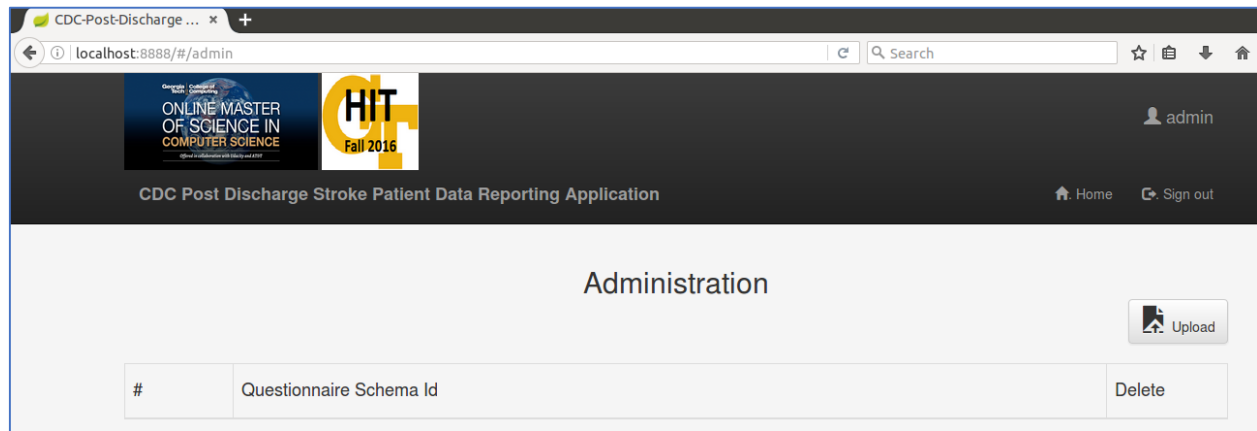
**Screenshot:** Questionnaire page (cont..)

nurse\_jane then conducts the survey by going through each of the questions and collect answers. When all the required questions are answered completely, nurse\_jane can either submit them by clicking on Submit button to have data persisted into Stroke the application database and FHIR resource or click on Cancel button to cancel the transaction.

Patient Name	Days Since Discharge	Questionnaire	Response
Barack Obama	48		12344555
Fred Flinstone	43		<a href="#">5844fa4e5d8f37ba6aff0c6</a>
<pre>Object   resourceType: "QuestionnaireResponse"   questionnaire: Object   subject: Object   group: Object</pre>			
<a href="#">CSV</a> <a href="#">JSON</a>			
Steve Waugh	17		12344553
Priyanka Chopra	12		
Kristen Stewart	11		12344554

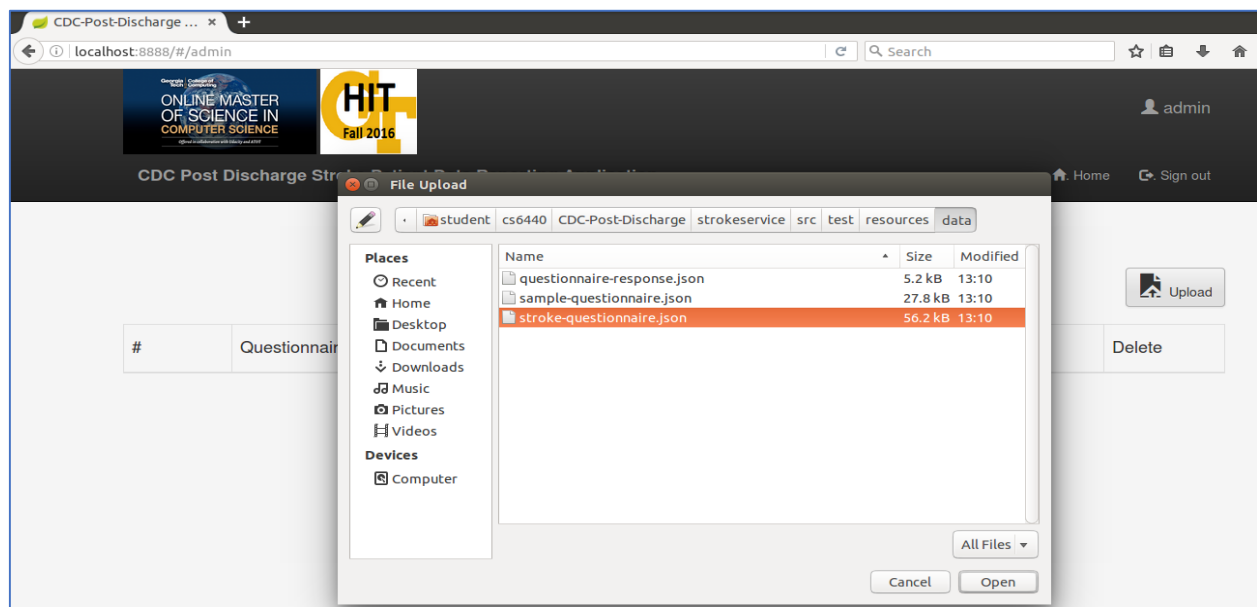
### Screenshot: Patient List page - Completed surveys

Once the questionnaire is completed, the response ID for the QuestionnaireResponse will be displayed as a clickable link under the "Response" column. When nurse\_jane clicks on the response ID, nurse\_jane can view details of this patient's responses and download a JSON or CSV file which captures the survey responses.



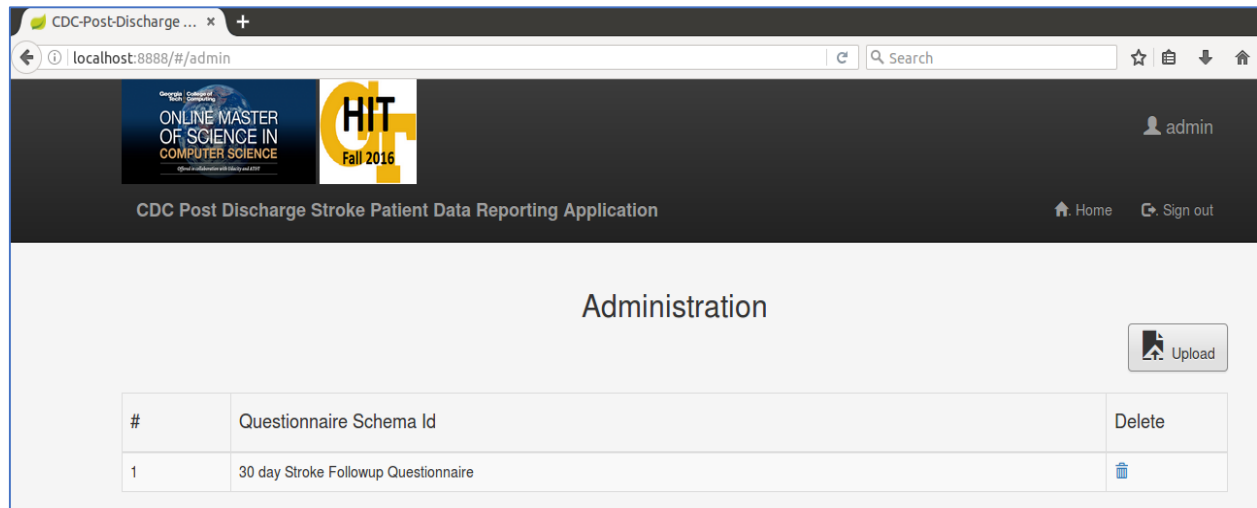
**Screenshot:** Load new questionnaire page

PDS Stroke app has feature to allow an Admin to upload the survey Questionnaire into the application.



**Screenshot:** Load new questionnaire page (cont...)

The PDS App Admin can click on an Upload button to locate a JSON file which stores the survey as a FHIR Questionnaire resource and upload it. Once uploaded, form would be persisted in Stroke App database.



**Screenshot:** Load new questionnaire page (cont...)

Following screenshot shows how survey admin can remove questionnaire form. To do so, the PDS App Admin can click on Trash bin icon whose form is wished to be removed. Once removed, this form would be purged from Stroke app database.

Only user who has admin privilege can delete or upload questionnaires for the PDS App database. If a survey needs to be modified, the PDS App Admin can delete the existing questionnaire form then upload an updated form.

## **VI. Source Code**

Source code can be downloaded from:

<https://github.gatech.edu/gt-hit-fall2016/CDC-Post-Discharge>

Please contact GA Tech Team Health Informaniacs if you cannot access to this link or acquire further information or collaboration.

## **VII. Acknowledgements**

We wish to thank the Ga Tech. Professor, Dr. Mark Braunstein, and TAs, Tia Pope Bolinger, Kelli McLure, and others and Dr. Arunkumar Srinivasan and CDC team for giving us opportunity and guidance to work on this project. Also, thank you all team members of Ga Tech. Team Health Informaniacs who had spent great amount of time and effort to complete this project in short time.