AMPATH
Capstone Experience

Team abstract-class
(offline-data-capture service)

Overview

- **Class Objective:** Develop offline capabilities for the AMPATH app.
- **Primary Focus of our Team:** Capturing existing data for offline use.
- **Sprint 1:** Getting Started.
- **Sprint 2:** Deliberating and Conceptualizing.
- **Sprint 3:** Refactoring the Existing Online Tracker.
- **Sprint 4:** Getting Familiarized with PouchDB.
- **Sprint 5:** Improving our Offline Data Capture Implementation.
- **Sprint 6:** Capturing and Displaying Specific Patient Data.
- **Final Thoughts**

Sprint 1

Getting Started

Sprint 1: Topics Covered

- Getting familiar with the overall AMPATH app structure.
- Compilation issues - aka The “ladda” problem.
- CORS: Issues with the Allow-Control-Allow-Origin Google Extension.
- Successfully connecting to the AMPATH server.

Getting Familiar with the AMPATH Application

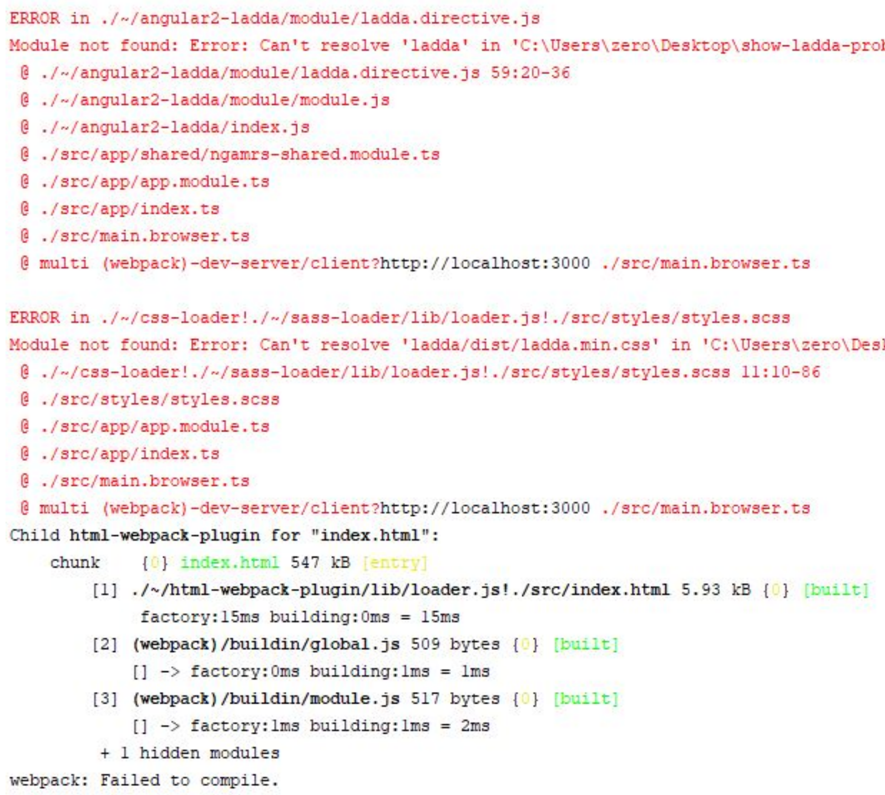
- AMPATH uses many Angular techniques that most of our team members had previously never seen before.

- Therefore, we spent a large portion of this sprint completing tutorials to become better familiarized with such techniques.

- The following tutorials in particular were incredibly helpful for us in better understanding how the AMPATH app works, especially regarding HTTP, REST and Routing:

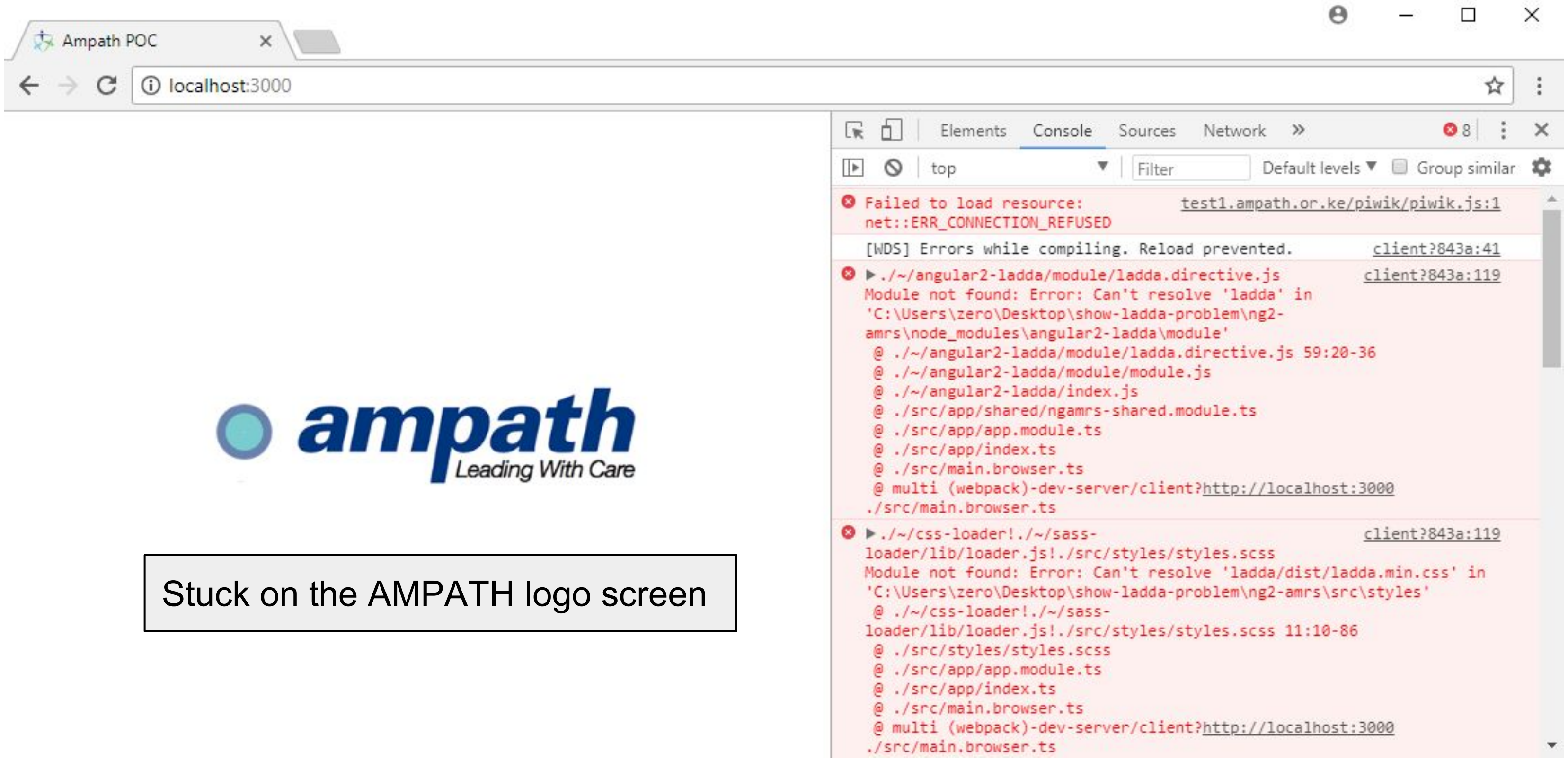
- **Tour of Heroes:** https://angular.io/tutorial
- **Todo-app:** https://www.sitepoint.com/angular-2-tutorial/

Compilation Issues: Remember “ladda” guys?



```
ERROR in ./~/angular2-ladda/module/ladda.directive.js
Module not found: Error: Can't resolve 'ladda' in 'C:\Users\zero\Desktop\show-ladda-problem\ng2-amrs\node_modules\angular2-ladda\module'
 @ ./~/angular2-ladda/module/ladda.directive.js 59:20-36
 @ ./~/angular2-ladda/module/module.js
 @ ./~/angular2-ladda/index.js
 @ ./src/app/shared/ngamrs-shared.module.ts
 @ ./src/app/app.module.ts
 @ ./src/app/index.ts
 @ ./src/main.browser.ts
 @ multi (webpack)-dev-server/client?http://localhost:3000 ./src/main.browser.ts

ERROR in ./~/css-loader!./~/sass-loader/lib/loader.js!./src/styles/styles.scss
Module not found: Error: Can't resolve 'ladda/dist/ladda.min.css' in 'C:\Users\zero\Desktop\show-ladda-problem\ng2-amrs\src\styles'
 @ ./~/css-loader!./~/sass-loader/lib/loader.js!./src/styles/styles.scss 11:10-86
 @ ./src/styles/styles.scss
 @ ./src/app/app.module.ts
 @ ./src/app/index.ts
 @ ./src/main.browser.ts
 @ multi (webpack)-dev-server/client?http://localhost:3000 ./src/main.browser.ts
Child html-webpack-plugin for "index.html":
    chunk    {0} index.html 547 kB [entry]
        [1] ./~/html-webpack-plugin/lib/loader.js!./src/index.html 5.93 kB {0} [built]
             factory:15ms building:0ms = 15ms
        [2] (webpack)/buildin/global.js 509 bytes {0} [built]
            [] -> factory:0ms building:1ms = 1ms
        [3] (webpack)/buildin/module.js 517 bytes {0} [built]
            [] -> factory:1ms building:1ms = 2ms
         + 1 hidden modules
webpack: Failed to compile.
```

Runtime Issues with the “ladda” Problem



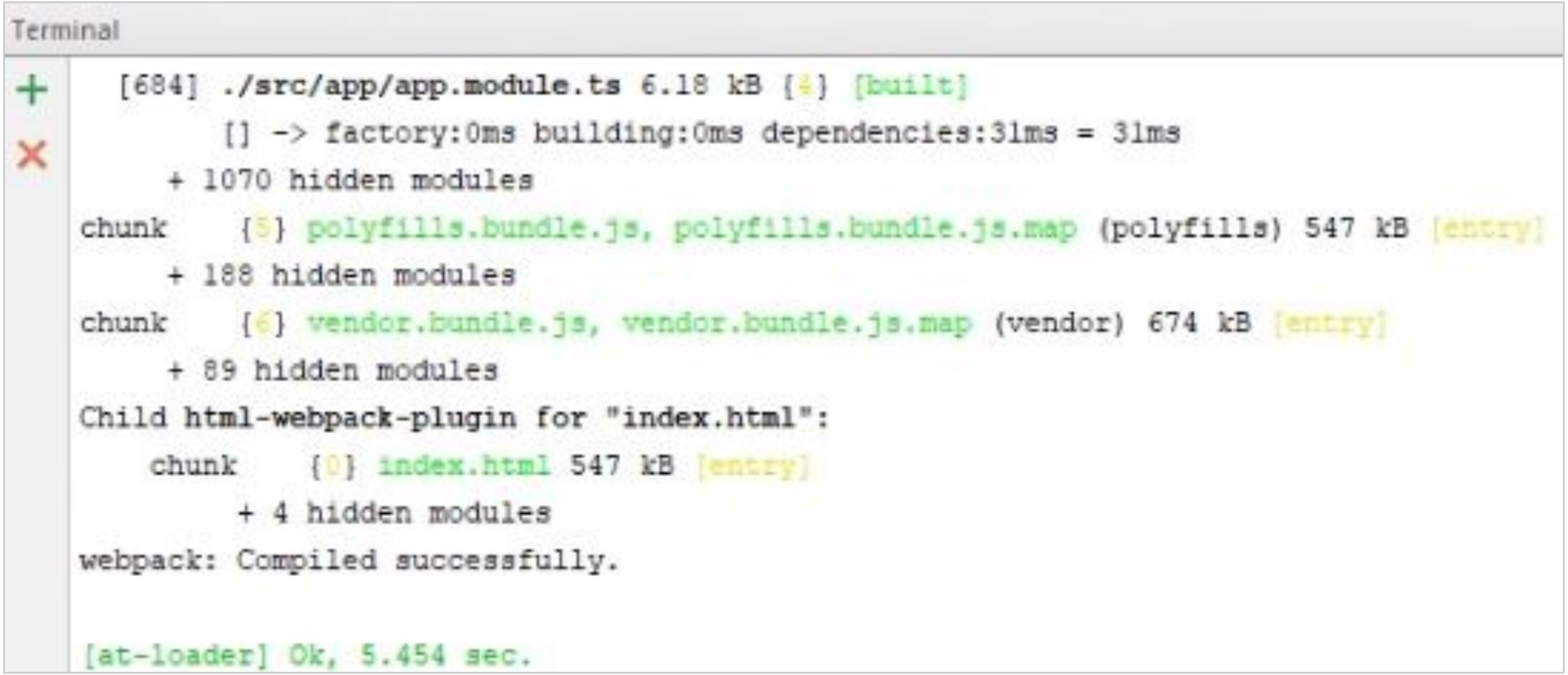
```
Failed to load resource:                test1.ampath.or.ke/piwik/piwik.js:1
net::ERR_CONNECTION_REFUSED
[WDS] Errors while compiling. Reload prevented.              client?843a:41
▶ ./~/angular2-ladda/module/ladda.directive.js             client?843a:119
Module not found: Error: Can't resolve 'ladda' in
'C:\Users\zero\Desktop\show-ladda-problem\ng2-
amrs\node_modules\angular2-ladda\module'
 @ ./~/angular2-ladda/module/ladda.directive.js 59:20-36
 @ ./~/angular2-ladda/module/module.js
 @ ./~/angular2-ladda/index.js
 @ ./src/app/shared/ngamrs-shared.module.ts
 @ ./src/app/app.module.ts
 @ ./src/app/index.ts
 @ ./src/main.browser.ts
 @ multi (webpack)-dev-server/client?http://localhost:3000
./src/main.browser.ts
▶ ./~/css-loader!./~/sass-                                 client?843a:119
loader/lib/loader.js!./src/styles/styles.scss
Module not found: Error: Can't resolve 'ladda/dist/ladda.min.css' in
'C:\Users\zero\Desktop\show-ladda-problem\ng2-amrs\src\styles'
 @ ./~/css-loader!./~/sass-
loader/lib/loader.js!./src/styles/styles.scss 11:10-86
 @ ./src/styles/styles.scss
 @ ./src/app/app.module.ts
 @ ./src/app/index.ts
 @ ./src/main.browser.ts
 @ multi (webpack)-dev-server/client?http://localhost:3000
./src/main.browser.ts
```

Stuck on the AMPATH logo screen

Resolution

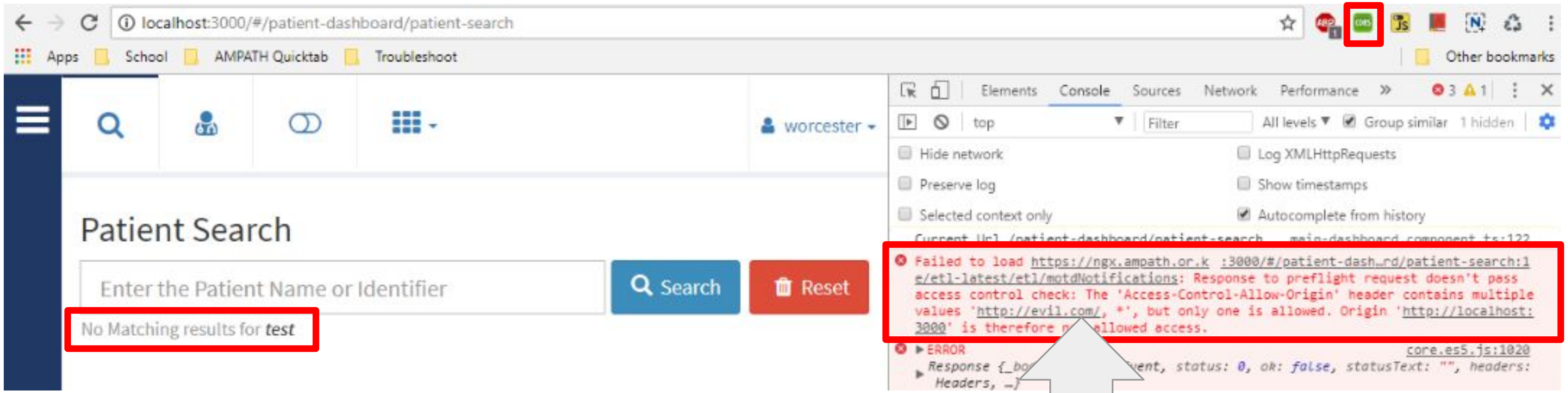
1. Clone a clean copy of ng2-amrs, e.g.: `git clone https://github.com/<your-github-name>/ng2-amrs ng2-amrs-clone`
2. Open ng2-amrs-clone in WebStorm and press ALT-F12 (or Fn-ALT-F12) to start a terminal.
3. Delete package-lock.json from the Project menu in WebStorm (make sure safe delete is checked)
4. In the webstorm terminal, run: `npm install pouchdb @types/pouchdb`
5. Run: `npm install webpack webpack-dev-server karma-cli protractor typescript release-it rimraf -g`
6. Run: `npm install` now that all the required dependencies are installed.
7. Run: `npm start` (This should now be successful without any compilation or runtime errors)

Result: Successful Compilation


Terminal

```
  [684] ./src/app/app.module.ts 6.18 kB {4} [built]
        [] -> factory:0ms building:0ms dependencies:31ms = 31ms
     + 1070 hidden modules
chunk    {5} polyfills.bundle.js, polyfills.bundle.js.map (polyfills) 547 kB [entry]
     + 188 hidden modules
chunk    {6} vendor.bundle.js, vendor.bundle.js.map (vendor) 674 kB [entry]
     + 89 hidden modules
Child html-webpack-plugin for "index.html":
    chunk    {0} index.html 547 kB [entry]
        + 4 hidden modules
webpack: Compiled successfully.

[at-loader] Ok, 5.454 sec.
```

Connecting to the AMPATH Server



Patient Search

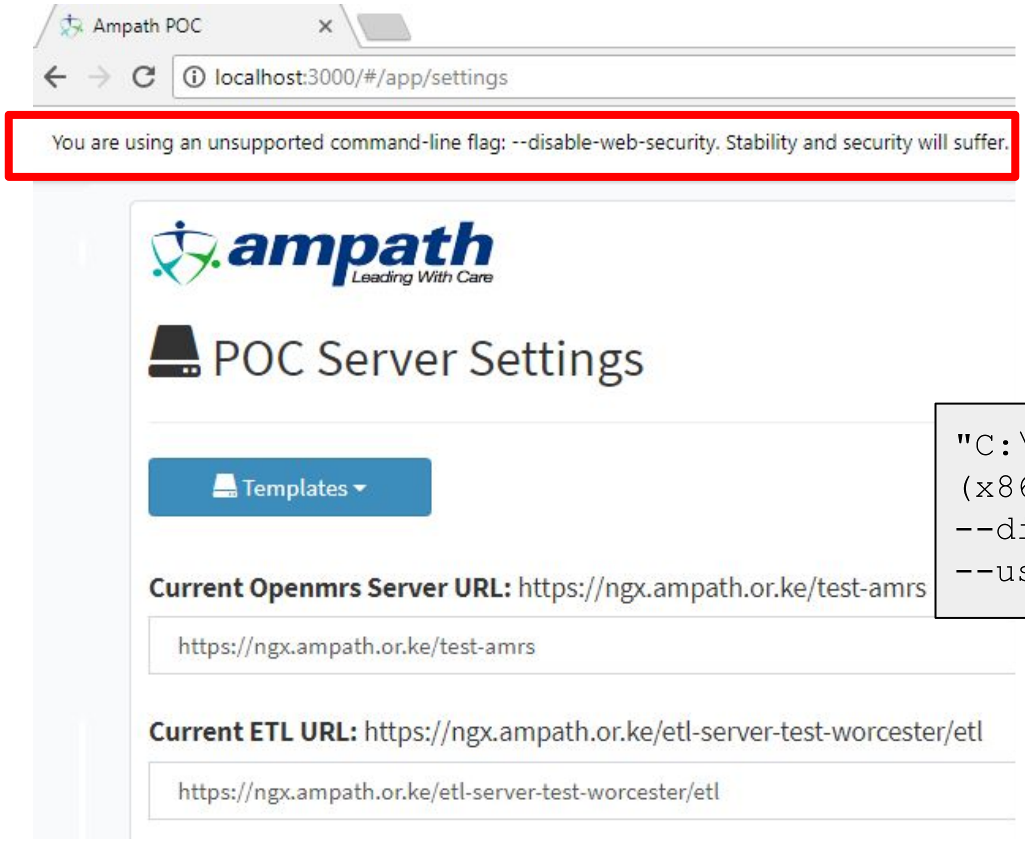
Enter the Patient Name or Identifier

No Matching results for *test*

```
Current Url /patient-dashboard/patient-search    main-dashboard.component.ts:122
Failed to load https://ngx.ampath.or.k :3000/#/patient-dash…rd/patient-search:1
e/etl-latest/etl/motdNotifications: Response to preflight request doesn't pass
access control check: The 'Access-Control-Allow-Origin' header contains multiple
values 'http://evil.com/, *', but only one is allowed. Origin 'http://localhost:
3000' is therefore n  allowed access.
▶ERROR                                                          core.es5.js:1020
▶ Response {_bo            ent, status: 0, ok: false, statusText: "", headers:
   Headers, …}
```

- Problems with the Allow-Control-Allow-Origin Google Extension: **evil.com?**

Resolution: Disable Web Security in Chrome



Ampath POC

localhost:3000/#/app/settings

You are using an unsupported command-line flag: --disable-web-security. Stability and security will suffer.

POC Server Settings

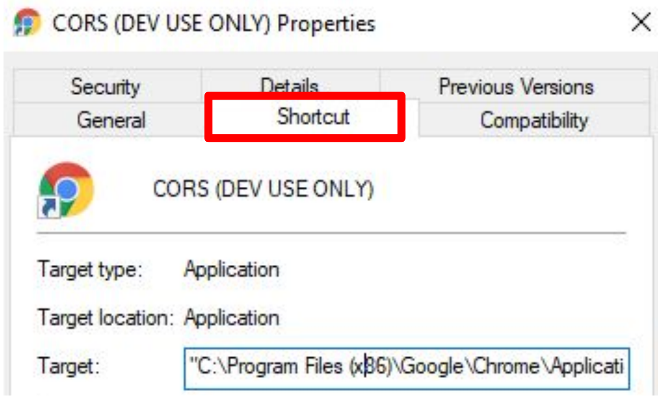
Templates

Current Openmrs Server URL: https://ngx.ampath.or.ke/test-amrs

https://ngx.ampath.or.ke/test-amrs

Current ETL URL: https://ngx.ampath.or.ke/etl-server-test-worcester/etl

https://ngx.ampath.or.ke/etl-server-test-worcester/etl



CORS (DEV USE ONLY) Properties

Security | Details | Previous Versions

General | Shortcut | Compatibility

CORS (DEV USE ONLY)

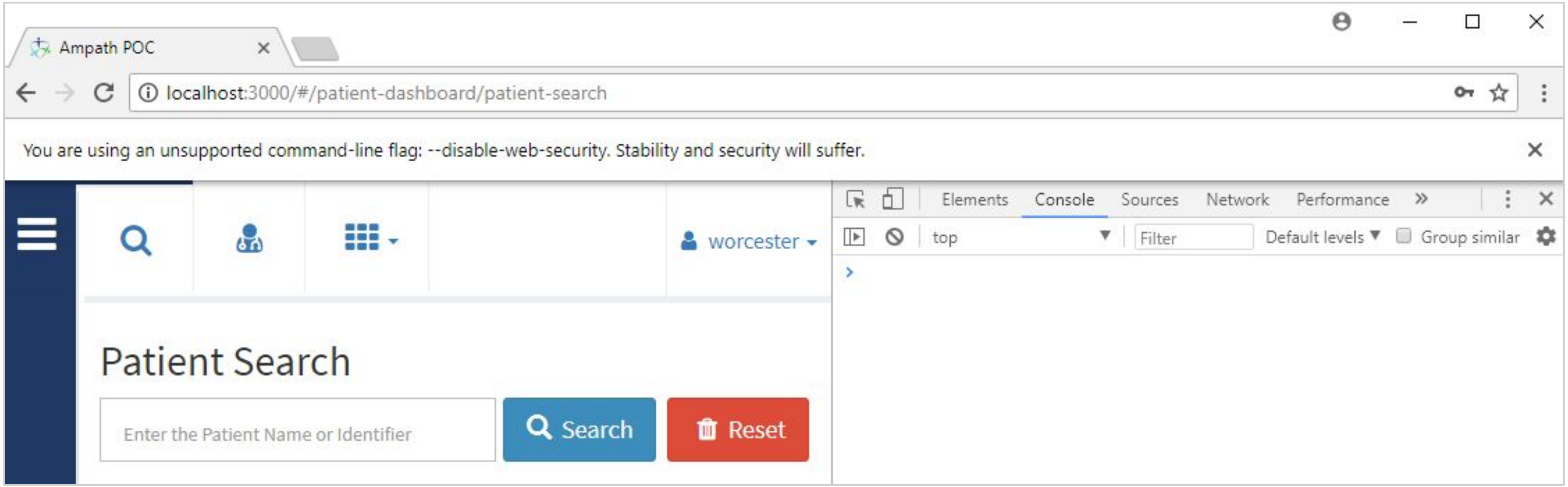
Target type: Application

Target location: Application

Target: "C:\Program Files (x86)\Google\Chrome\Applicati

```
"C:\Program Files
(x86)\Google\Chrome\Application\chrome.exe"
--disable-web-security
--user-data-dir="C:/ChromeDevSession"
```

We’re in without errors!



You are using an unsupported command-line flag: --disable-web-security. Stability and security will suffer.

Patient Search

Sprint 2

Deliberating and Conceptualizing

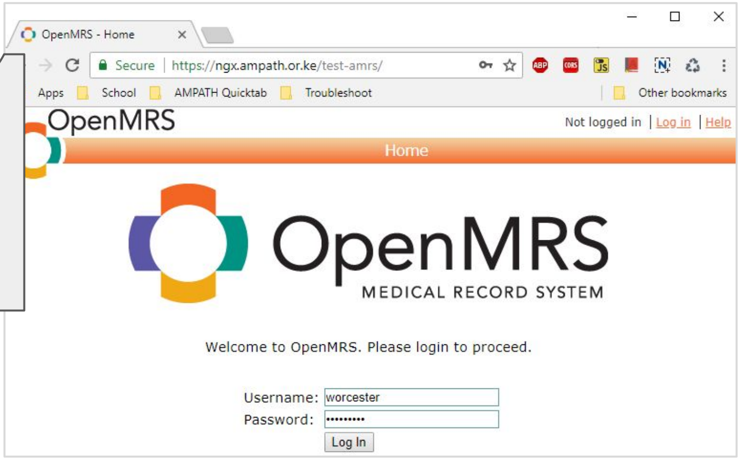
Sprint 2: Topics Covered

- What we learned about AMPATH app so far.
- OpenMRS AMPATH Server.
- ETL AMPATH Server.
- High-level ideas for Offline Implementation.
- High-level ideas for Synchronization.

Some Discoveries During Sprint 2

- The AMPATH app communicates to servers primarily by REST (Representational State Transfer) API.
  - REST is an approach that uses HTTP requests to GET, PUT, POST and DELETE data.
- The app is dependent on not one, but TWO servers, **OpenMRS** and **ETL**.

1



OpenMRS - Home

Secure | https://ngx.ampath.or.ke/test-amrs/

Apps | School | AMPATH Quicktab | Troubleshoot | Other bookmarks

OpenMRS

Not logged in | Log in | Help

Home

OpenMRS MEDICAL RECORD SYSTEM

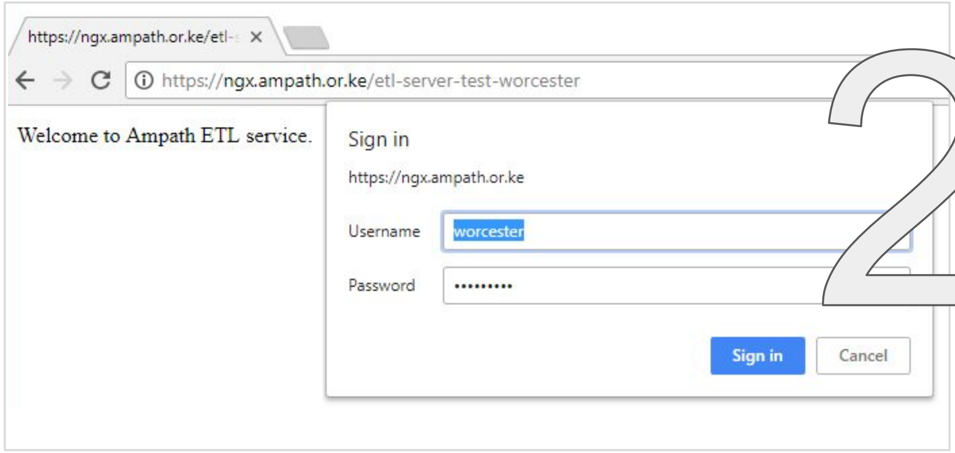
Welcome to OpenMRS. Please login to proceed.

Username: worcester

Password: •••••••••

Log In

2



https://ngx.ampath.or.ke/etl-server-test-worcester

Welcome to Ampath ETL service.

Sign in

https://ngx.ampath.or.ke

Username: worcester

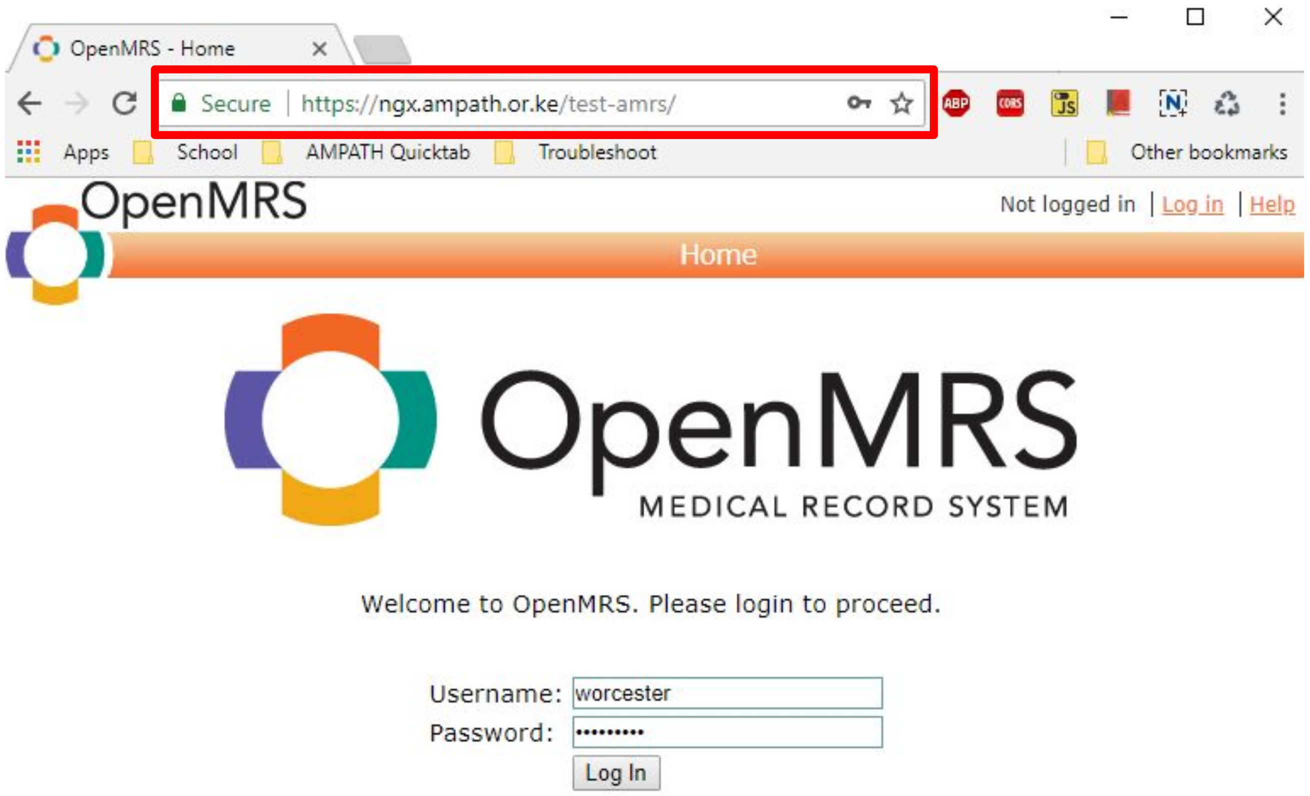
Password: •••••••••

Sign in | Cancel

OpenMRS and ETL

- The majority (if not all) of the patient data seems to exist in OpenMRS.
- The responsibility of the AMPATH ETL server seems to extract the data from the OpenMRS server and flatten certain tables, such as HIV Summary.
- It is worth noting that we did not have access to an AMPATH ETL server until the end of the semester.
- Even after gaining access to ETL, we still couldn’t seem to access certain data (such as HIV summaries and labs) -- will discuss more in depth during Sprint 6 section.

Accessing AMPATH OpenMRS Server Directly

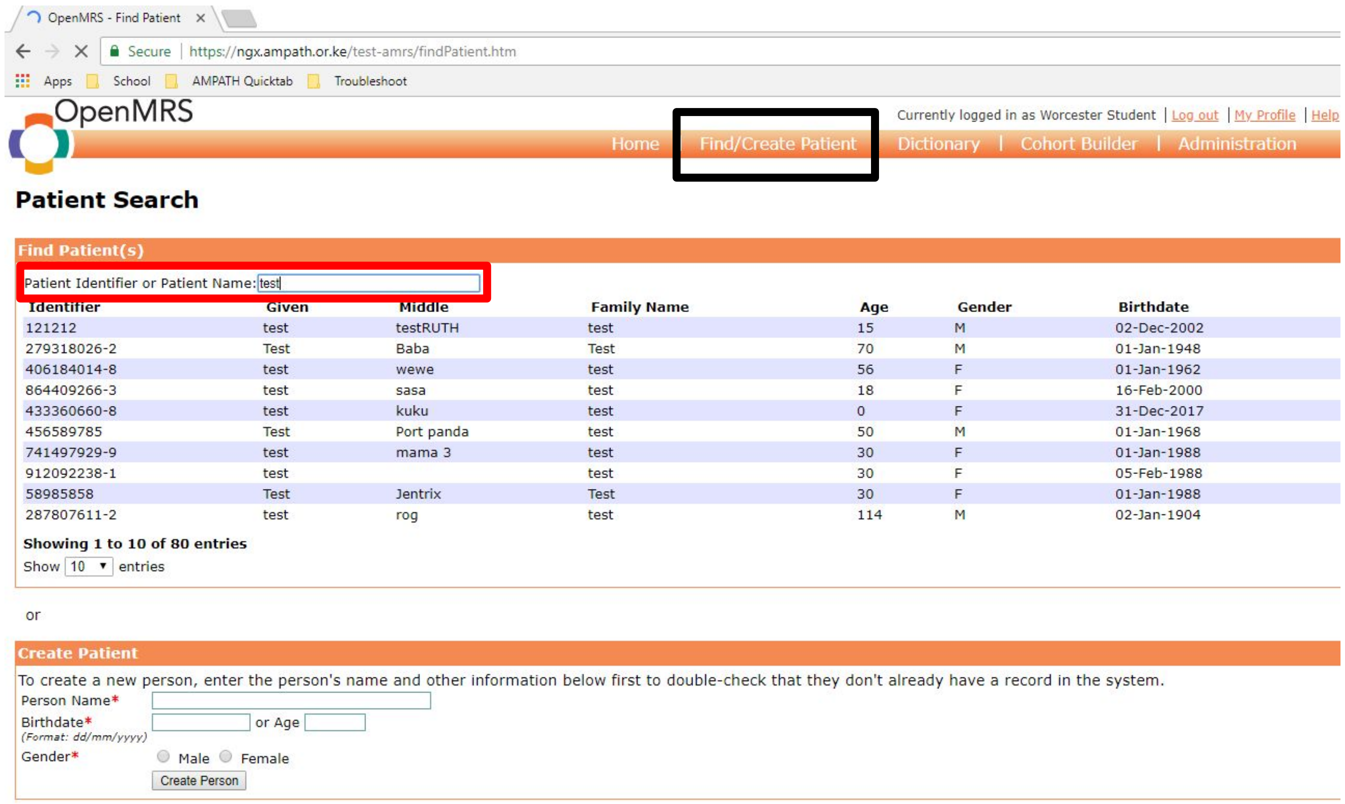


Welcome to OpenMRS. Please login to proceed.

Username:	worcester
Password:	••••••••••

Log In

Accessing AMPATH OpenMRS Server Directly



Currently logged in as Worcester Student | Log out | My Profile | Help

Patient Search

Find Patient(s)

Patient Identifier or Patient Name: test

Identifier	Given	Middle	Family Name	Age	Gender	Birthdate
121212	test	testRUTH	test	15	M	02-Dec-2002
279318026-2	Test	Baba	Test	70	M	01-Jan-1948
406184014-8	test	wewe	test	56	F	01-Jan-1962
864409266-3	test	sasa	test	18	F	16-Feb-2000
433360660-8	test	kuku	test	0	F	31-Dec-2017
456589785	Test	Port panda	test	50	M	01-Jan-1968
741497929-9	test	mama 3	test	30	F	01-Jan-1988
912092238-1	test		test	30	F	05-Feb-1988
58985858	Test	Jentrix	Test	30	F	01-Jan-1988
287807611-2	test	rog	test	114	M	02-Jan-1904

Showing 1 to 10 of 80 entries

Show 10 entries

or

Create Patient

To create a new person, enter the person's name and other information below first to double-check that they don't already have a record in the system.

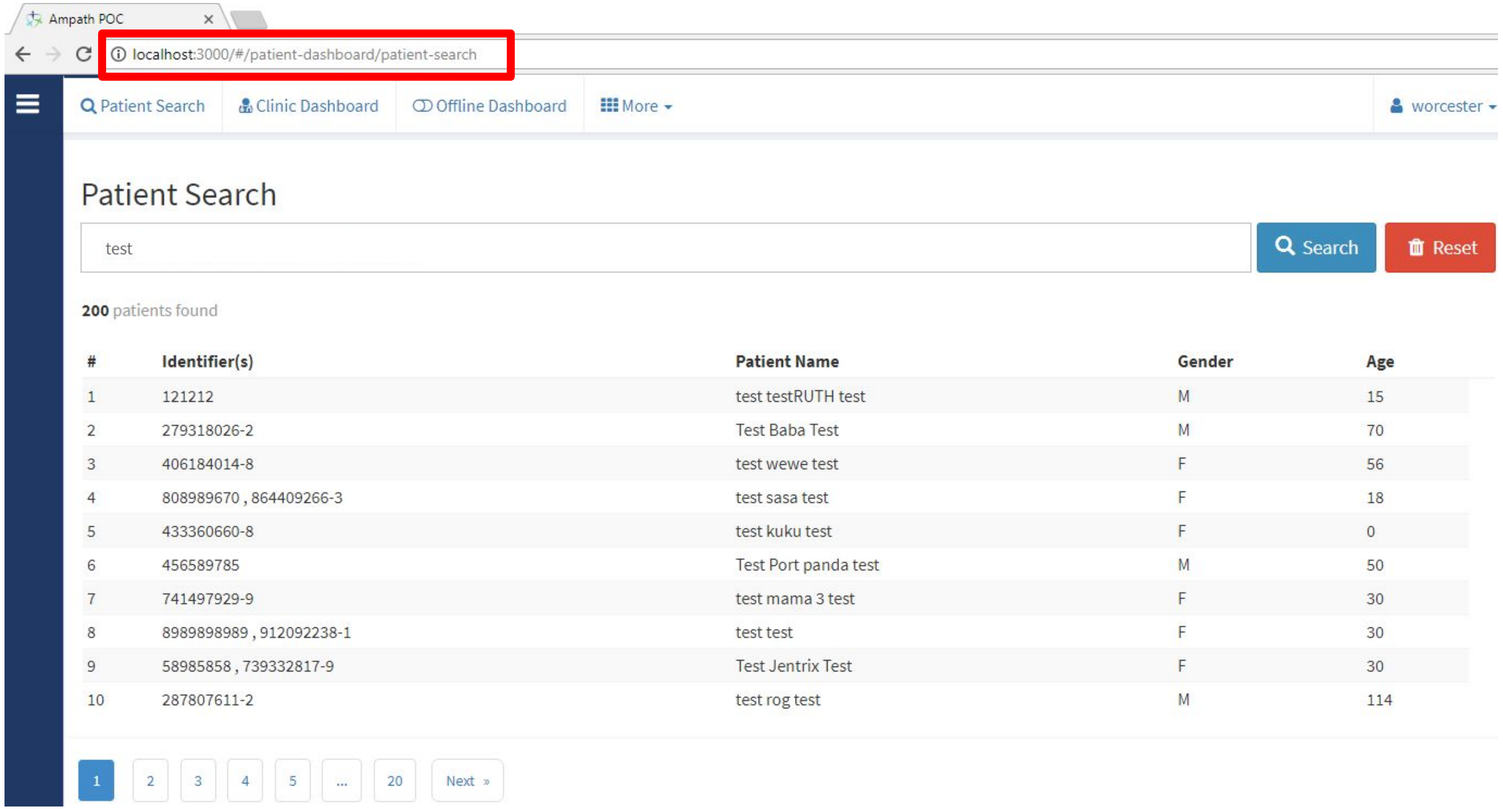
Person Name*

Birthdate* or Age
(Format: dd/mm/yyyy)

Gender* Male Female

Create Person

Compare to AMPATH Patient Search



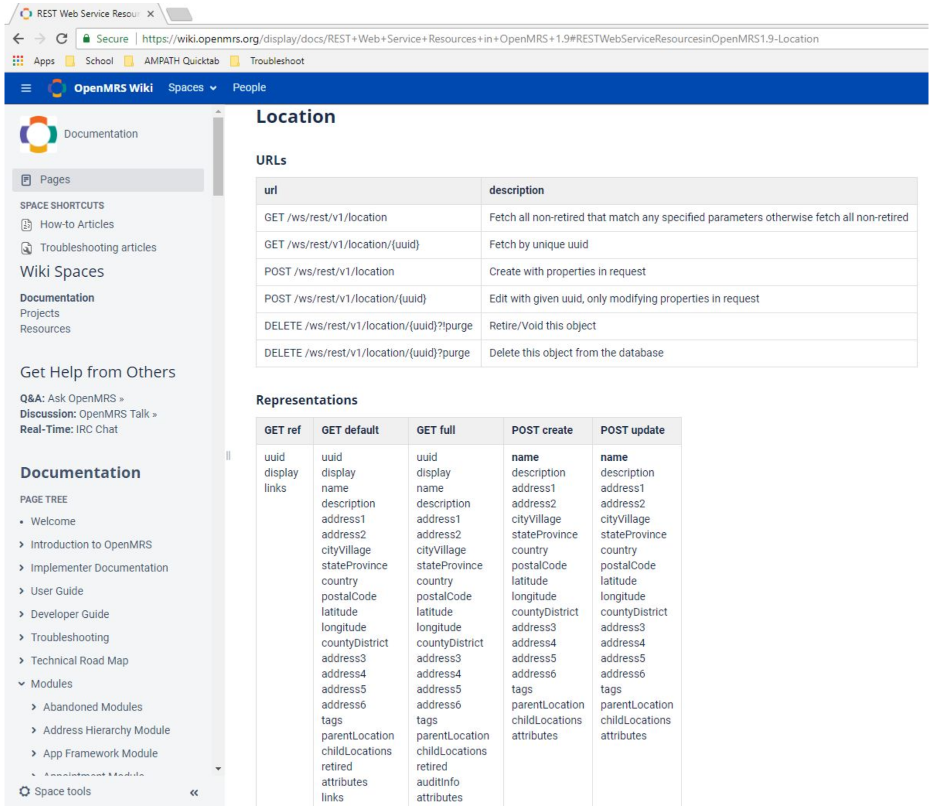
Patient Search

test

200 patients found

#	Identifier(s)	Patient Name	Gender	Age
1	121212	test testRUTH test	M	15
2	279318026-2	Test Baba Test	M	70
3	406184014-8	test wewe test	F	56
4	808989670 , 864409266-3	test sasa test	F	18
5	433360660-8	test kuku test	F	0
6	456589785	Test Port panda test	M	50
7	741497929-9	test mama 3 test	F	30
8	8989898989 , 912092238-1	test test	F	30
9	58985858 , 739332817-9	Test Jentrix Test	F	30
10	287807611-2	test rog test	M	114

OpenMRS Wiki: wiki.openmrs.org



REST Web Service Resou

Secure | https://wiki.openmrs.org/display/docs/REST+Web+Service+Resources+in+OpenMRS+1.9#RESTWebServiceResourcesinOpenMRS1.9-Location

Apps School AMPATH Quicktab Troubleshoot

OpenMRS Wiki Spaces People

Documentation

Pages

SPACE SHORTCUTS

How-to Articles

Troubleshooting articles

Wiki Spaces

Documentation
Projects
Resources

Get Help from Others

Q&A: Ask OpenMRS »
Discussion: OpenMRS Talk »
Real-Time: IRC Chat

Documentation

PAGE TREE

- Welcome
- Introduction to OpenMRS
- Implementer Documentation
- User Guide
- Developer Guide
- Troubleshooting
- Technical Road Map
- Modules
  - Abandoned Modules
  - Address Hierarchy Module
  - App Framework Module

Space tools

Location

URLs

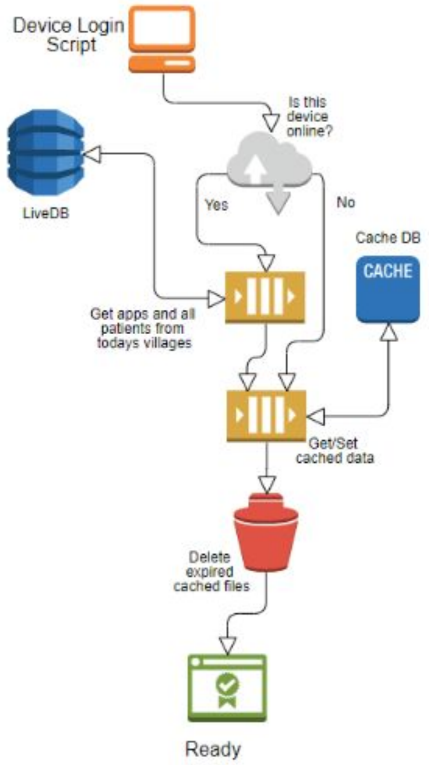
url	description
GET /ws/rest/v1/location	Fetch all non-retired that match any specified parameters otherwise fetch all non-retired
GET /ws/rest/v1/location/{uuid}	Fetch by unique uuid
POST /ws/rest/v1/location	Create with properties in request
POST /ws/rest/v1/location/{uuid}	Edit with given uuid, only modifying properties in request
DELETE /ws/rest/v1/location/{uuid}?!purge	Retire/Void this object
DELETE /ws/rest/v1/location/{uuid}?purge	Delete this object from the database

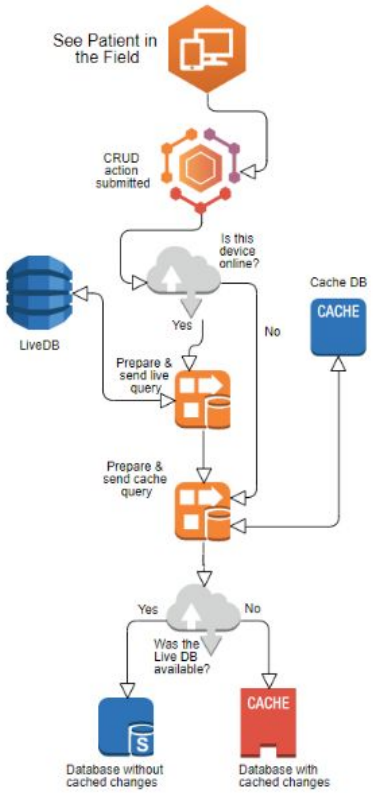
Representations

GET ref	GET default	GET full	POST create	POST update
uuid display links	uuid display name description address1 address2 cityVillage stateProvince country postalCode latitude longitude countyDistrict address3 address4 address5 address6 tags parentLocation childLocations retired attributes links	uuid display name description address1 address2 cityVillage stateProvince country postalCode latitude longitude countyDistrict address3 address4 address5 address6 tags parentLocation childLocations retired auditInfo attributes	**name** description address1 address2 cityVillage stateProvince country postalCode latitude longitude countyDistrict address3 address4 address5 address6 tags parentLocation childLocations attributes	**name** description address1 address2 cityVillage stateProvince country postalCode latitude longitude countyDistrict address3 address4 address5 address6 tags parentLocation childLocations attributes

- Very helpful resource
- Helped us better understand how to make HTTP/REST requests to the AMPATH OpenMRS server.

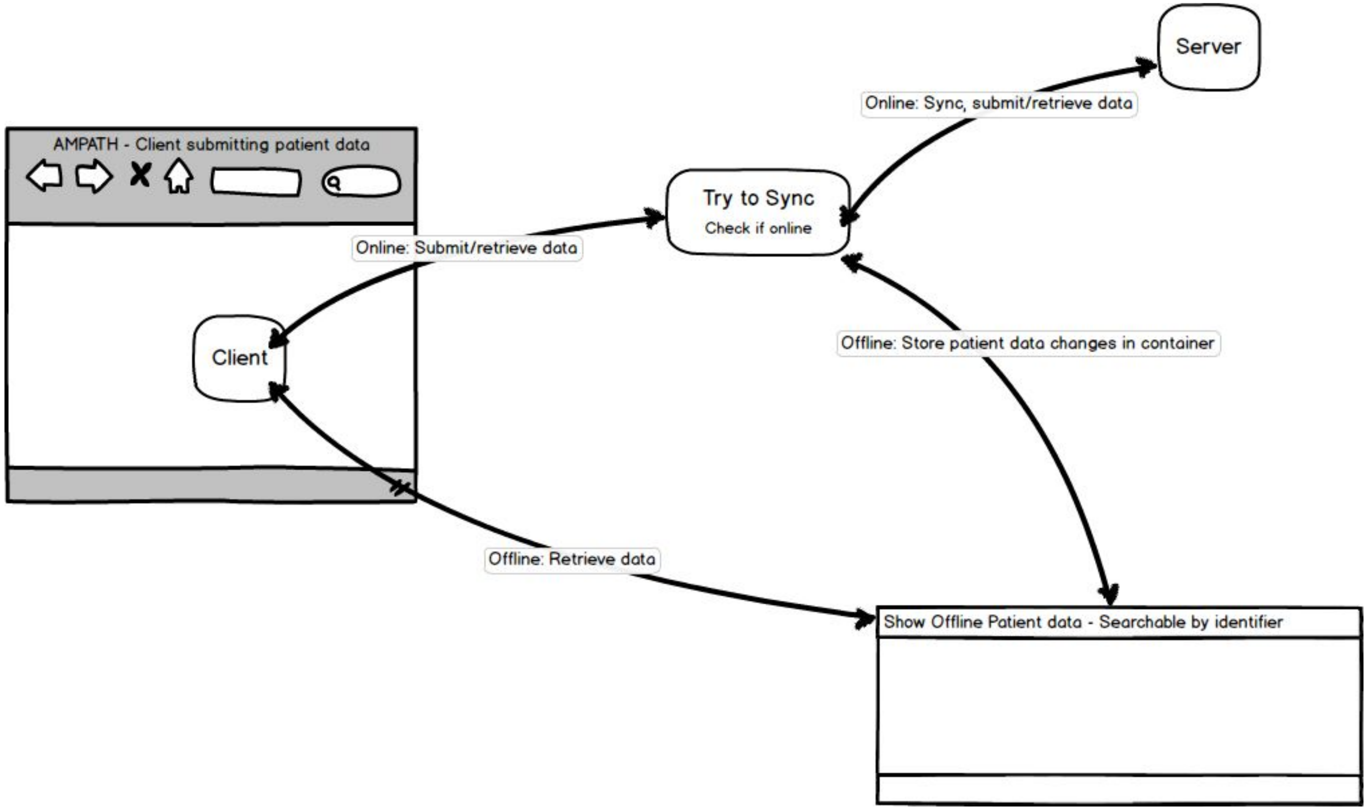
High-Level Ideas for Offline Implementation





See Patient in the Field
CRUD action submitted
Is this device online?
LiveDB
Cache DB
CACHE
Yes
No
Prepare & send live query
Prepare & send cache query
Yes
No
Was the Live DB available?
CACHE
Database without cached changes
Database with cached changes

High-Level idea for Data Synchronization


Server
Online: Sync, submit/retrieve data
AMPATH - Client submitting patient data
Try to Sync
Check if online
Online: Submit/retrieve data
Client
Offline: Store patient data changes in container
Offline: Retrieve data
Show Offline Patient data - Searchable by identifier

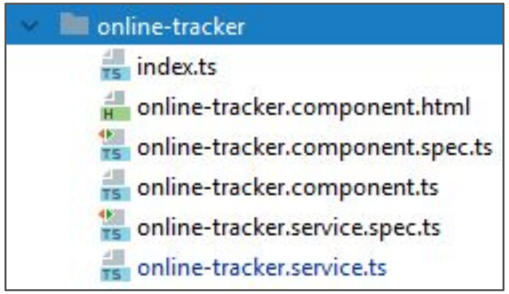
Sprint 3

Refactoring the Existing Online Tracker

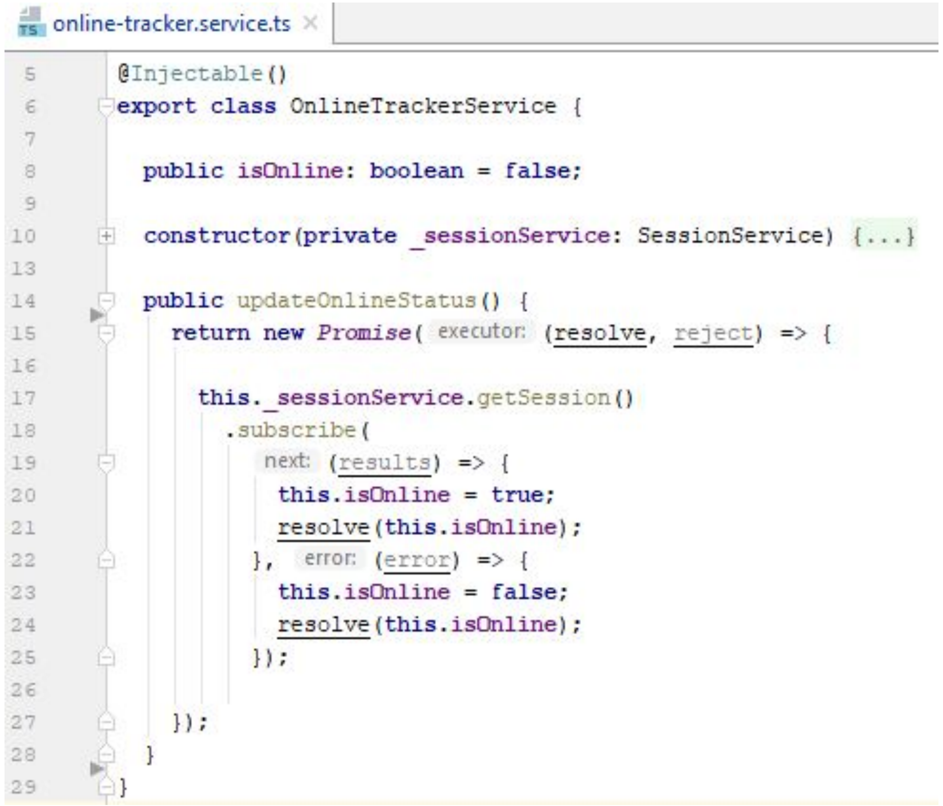
Why we Refactored Online Tracker

- We’re all working on giving the AMPATH app offline capabilities.
- Our class as a whole anticipated we would likely need a way to check for internet connectivity.
- A way of doing so existed, but it was wrapped inside AMPATH’s Online Tracker component.
- We refactored the existing Online Tracker component to include a service that will handle the connectivity checking logic.
- Now we can make our own components call upon the new Online Tracker service to check for connectivity.

Refactoring Online Tracker: Service



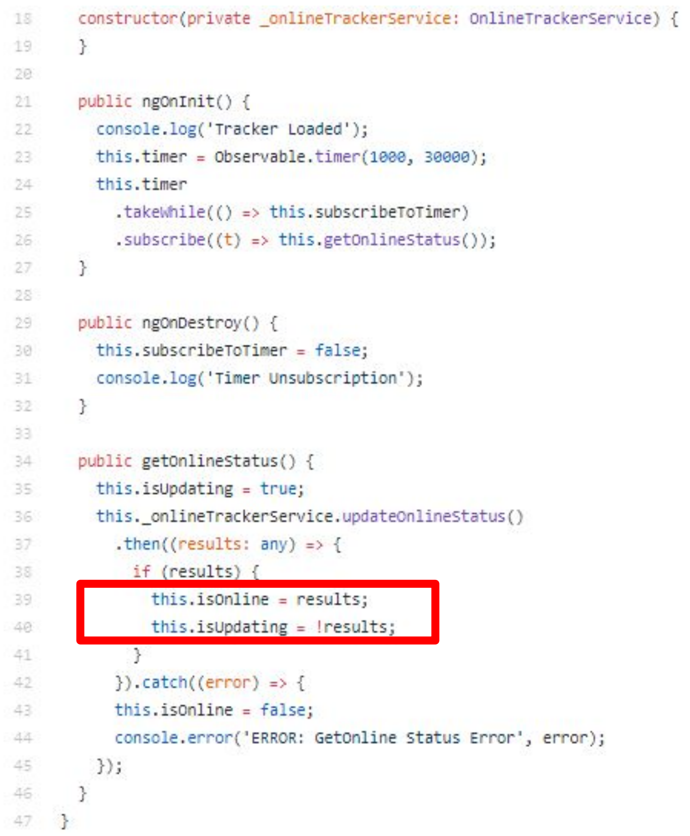
```
online-tracker
    index.ts
    online-tracker.component.html
    online-tracker.component.spec.ts
    online-tracker.component.ts
    online-tracker.service.spec.ts
    online-tracker.service.ts
```



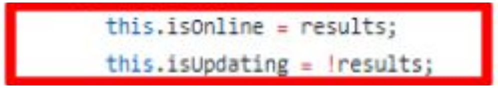
online-tracker.service.ts

```typescript
@Injectable()
export class OnlineTrackerService {

  public isOnline: boolean = false;

  constructor(private _sessionService: SessionService) {...}

  public updateOnlineStatus() {
    return new Promise( executor: (resolve, reject) => {

      this._sessionService.getSession()
        .subscribe(
          next: (results) => {
            this.isOnline = true;
            resolve(this.isOnline);
          }, error: (error) => {
            this.isOnline = false;
            resolve(this.isOnline);
          });

    });
  }
}
```

Refactoring Online Tracker: Component

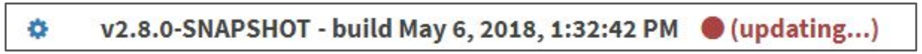


```
  constructor(private _onlineTrackerService: OnlineTrackerService) {
  }

  public ngOnInit() {
    console.log('Tracker Loaded');
    this.timer = Observable.timer(1000, 30000);
    this.timer
      .takeWhile(() => this.subscribeToTimer)
      .subscribe((t) => this.getOnlineStatus());
  }

  public ngOnDestroy() {
    this.subscribeToTimer = false;
    console.log('Timer Unsubscription');
  }

  public getOnlineStatus() {
    this.isUpdating = true;
    this._onlineTrackerService.updateOnlineStatus()
      .then((results: any) => {
        if (results) {
          this.isOnline = results;
          this.isUpdating = !results;
        }
      }).catch((error) => {
      this.isOnline = false;
      console.error('ERROR: GetOnline Status Error', error);
    });
  }
}
```



```
this.isOnline = results;
this.isUpdating = !results;
```

- The above code was causing the following problem:

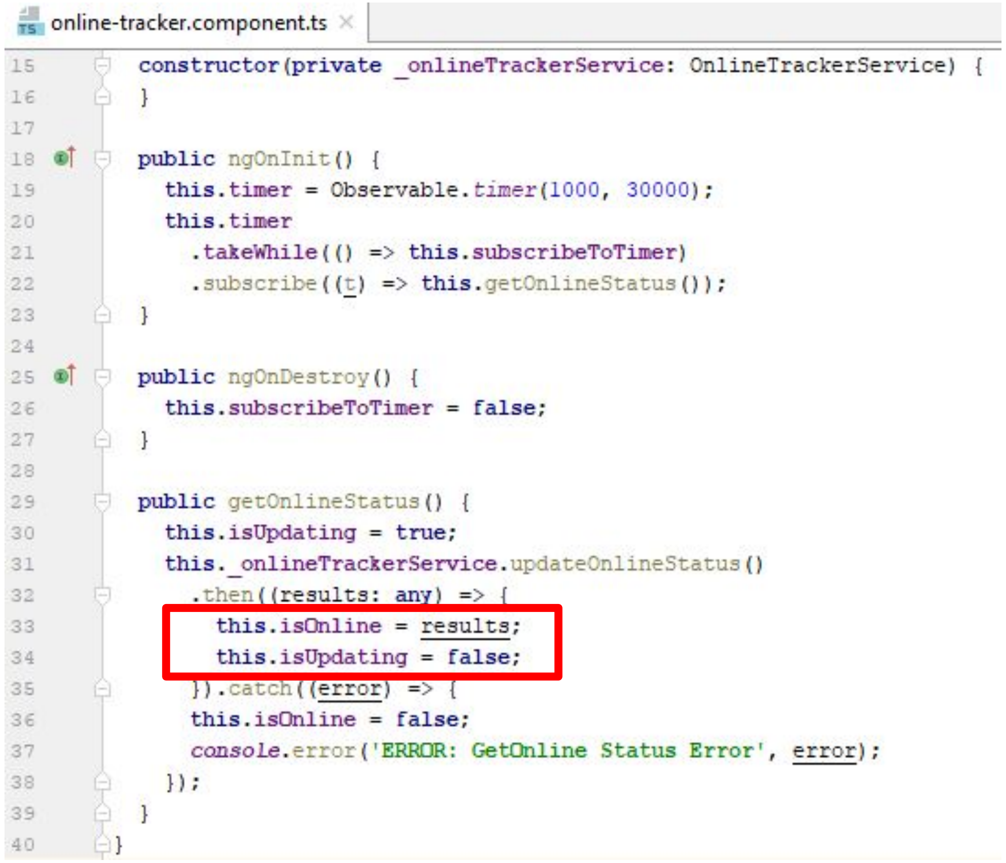


v2.8.0-SNAPSHOT - build May 6, 2018, 1:32:42 PM ● (updating...)

- Online-Tracker light indicator would be stuck in (updating…) status if offline, even after regaining connectivity.

- The **offline-login** team later identified and fixed this bug. Thanks guys!

Refactoring Online Tracker: Component

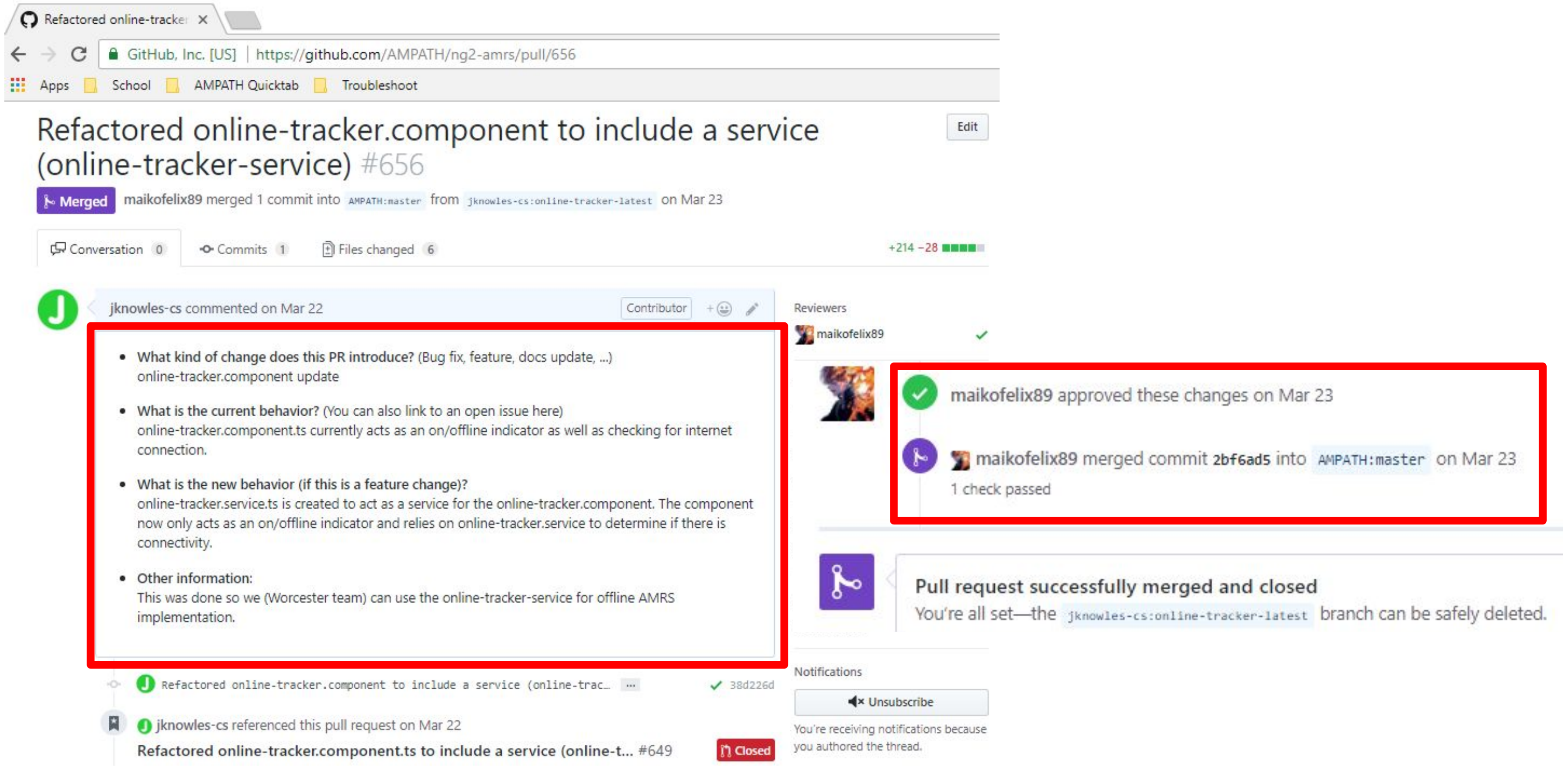


online-tracker.component.ts

```
  constructor(private _onlineTrackerService: OnlineTrackerService) {
  }

  public ngOnInit() {
    this.timer = Observable.timer(1000, 30000);
    this.timer
      .takeWhile(() => this.subscribeToTimer)
      .subscribe((t) => this.getOnlineStatus());
  }

  public ngOnDestroy() {
    this.subscribeToTimer = false;
  }

  public getOnlineStatus() {
    this.isUpdating = true;
    this._onlineTrackerService.updateOnlineStatus()
      .then((results: any) => {
        this.isOnline = results;
        this.isUpdating = false;
      }).catch((error) => {
      this.isOnline = false;
      console.error('ERROR: GetOnline Status Error', error);
    });
  }
}
```

- This is how the Online-Tracker Component looks now, after the bugfix from the **offline-login** team.

Online Tracker: Pull Request Accepted



Refactored online-tracke ✕

GitHub, Inc. [US] | https://github.com/AMPATH/ng2-amrs/pull/656

Apps School AMPATH Quicktab Troubleshoot

Refactored online-tracker.component to include a service (online-tracker-service) #656

Edit

Merged maikofelix89 merged 1 commit into `AMPATH:master` from `jknowles-cs:online-tracker-latest` on Mar 23

Conversation 0 Commits 1 Files changed 6 +214 −28

jknowles-cs commented on Mar 22 Contributor

- **What kind of change does this PR introduce?** (Bug fix, feature, docs update, ...)
  online-tracker.component update
- **What is the current behavior?** (You can also link to an open issue here)
  online-tracker.component.ts currently acts as an on/offline indicator as well as checking for internet connection.
- **What is the new behavior (if this is a feature change)?**
  online-tracker.service.ts is created to act as a service for the online-tracker.component. The component now only acts as an on/offline indicator and relies on online-tracker.service to determine if there is connectivity.
- **Other information:**
  This was done so we (Worcester team) can use the online-tracker-service for offline AMRS implementation.

`Refactored online-tracker.component to include a service (online-trac...` ✓ 38d226d

jknowles-cs referenced this pull request on Mar 22

Refactored online-tracker.component.ts to include a service (online-t... #649 Closed

Reviewers

maikofelix89 ✓

maikofelix89 approved these changes on Mar 23

maikofelix89 merged commit **2bf6ad5** into `AMPATH:master` on Mar 23

1 check passed

Pull request successfully merged and closed

You're all set—the `jknowles-cs:online-tracker-latest` branch can be safely deleted.

Notifications

Unsubscribe

You're receiving notifications because you authored the thread.

Sprint 4

Getting Familiarized with PouchDB

Sprint 4: Topics Covered

- Successfully installing PouchDB within the AMPATH app.

- Got routing to work; created a link so we could access our offline-data-capture component directly, i.e.
  ➔ **`localhost:3000/#/offline-data-capture`**

- Using existing OpenMRS services, such as patient-resource-service to capture some basic patient data.

PouchDB

- ➔ Since our main objective is to capture AMPATH data for offline use, we needed a place to store it, at least temporarily.

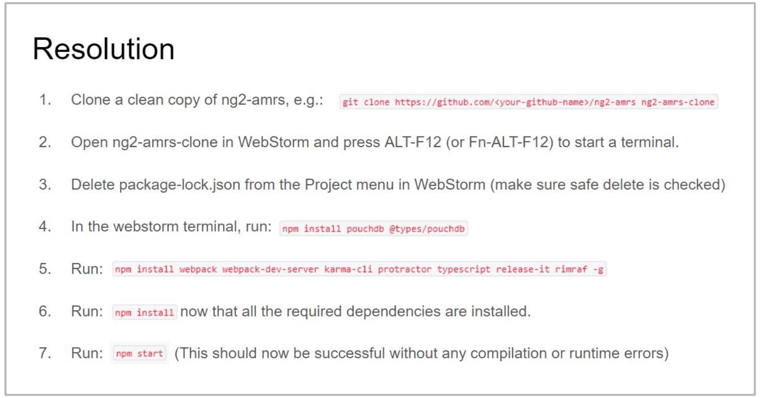
- ➔ We opted to use PouchDB for this purpose.

- **Pros using PouchDB:**
- Other teams were using PouchDB as well.
- A very powerful and straightforward way of storing data once we figured it out.

- **Cons using PouchDB:**
- Very limited documentation for Angular specific use.
- Required a lot of self-teaching and self-discovery, after much trial and error.

Installing PouchDB

- We originally had trouble with this.
- It often seems that when installing a new dependency in the AMPATH app, it is likely to stop working unless starting with a clean copy.
- We had to reclone a clean copy of the ng2-amrs repo and reinstall all the AMPATH dependencies to get PouchDB to work properly.



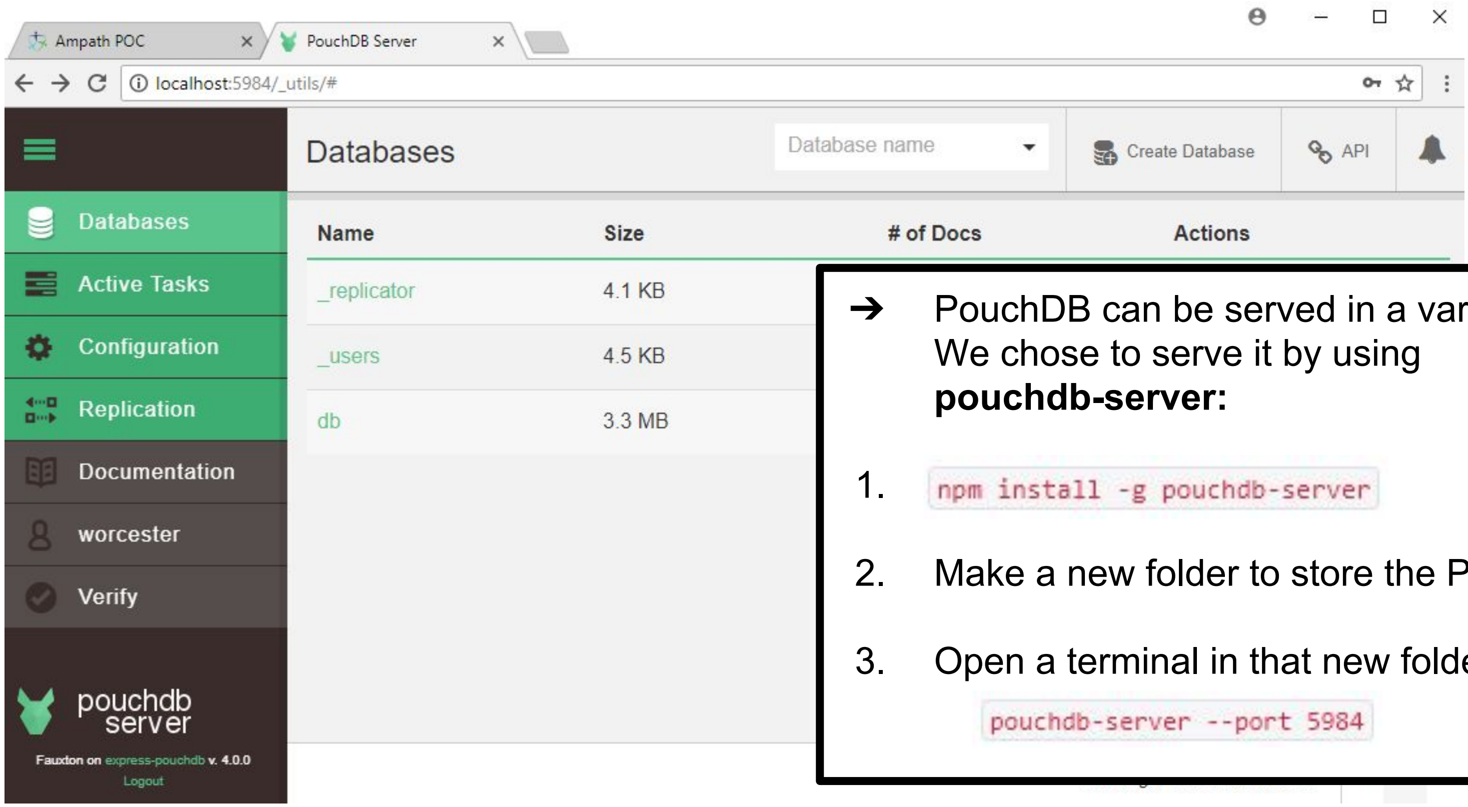
Resolution

1. Clone a clean copy of ng2-amrs, e.g.: `git clone https://github.com/<your-github-name>/ng2-amrs ng2-amrs-clone`
2. Open ng2-amrs-clone in WebStorm and press ALT-F12 (or Fn-ALT-F12) to start a terminal.
3. Delete package-lock.json from the Project menu in WebStorm (make sure safe delete is checked)
4. In the webstorm terminal, run: `npm install pouchdb @types/pouchdb`
5. Run: `npm install webpack webpack-dev-server karma-cli protractor typescript release-it rimraf -g`
6. Run: `npm install` now that all the required dependencies are installed.
7. Run: `npm start` (This should now be successful without any compilation or runtime errors)

Installing PouchDB:
npm install pouchdb @types/pouchdb

Installing PouchDB Server:
npm install -g pouchdb-server

PouchDB Server


Ampath POC
PouchDB Server
localhost:5984/_utils/#

Databases

Name	Size	# of Docs	Actions
_replicator	4.1 KB		
_users	4.5 KB		
db	3.3 MB		

➔ PouchDB can be served in a variety of ways. We chose to serve it by using **pouchdb-server:**

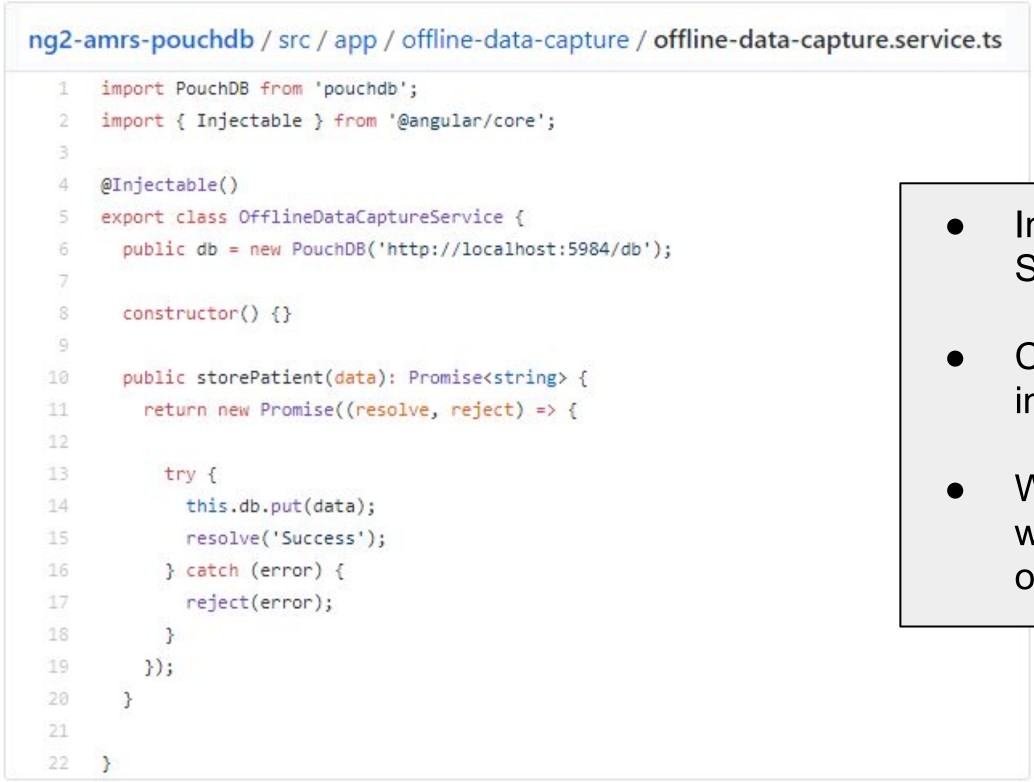
1. `npm install -g pouchdb-server`

2. Make a new folder to store the PouchDB data.

3. Open a terminal in that new folder, then type:

```
pouchdb-server --port 5984
```

Beginning to Implement Offline Data Capture Service



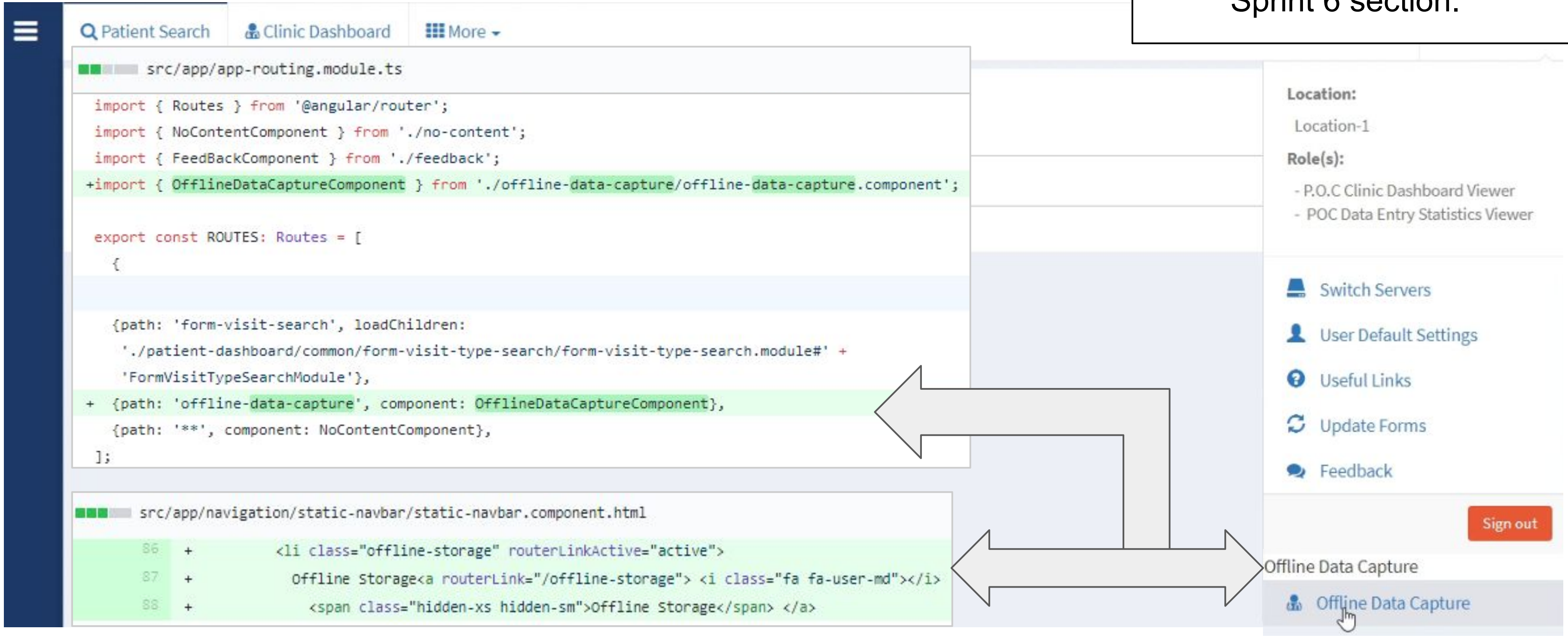
ng2-amrs-pouchdb / src / app / offline-data-capture / offline-data-capture.service.ts

```
import PouchDB from 'pouchdb';
import { Injectable } from '@angular/core';

@Injectable()
export class OfflineDataCaptureService {
  public db = new PouchDB('http://localhost:5984/db');

  constructor() {}

  public storePatient(data): Promise<string> {
    return new Promise((resolve, reject) => {

      try {
        this.db.put(data);
        resolve('Success');
      } catch (error) {
        reject(error);
      }
    });
  }

}
```

- Initial implementation of our service as of Sprint 4.
- Conflicts arise when trying to store a given ID in the PouchDB database more than once.
- We have since made some improvements which we will discuss more in depth during our Sprint 5 section.

Our Original Routing Implementation

- There's a better way to do this.
- We'll explain more in the Sprint 6 section.



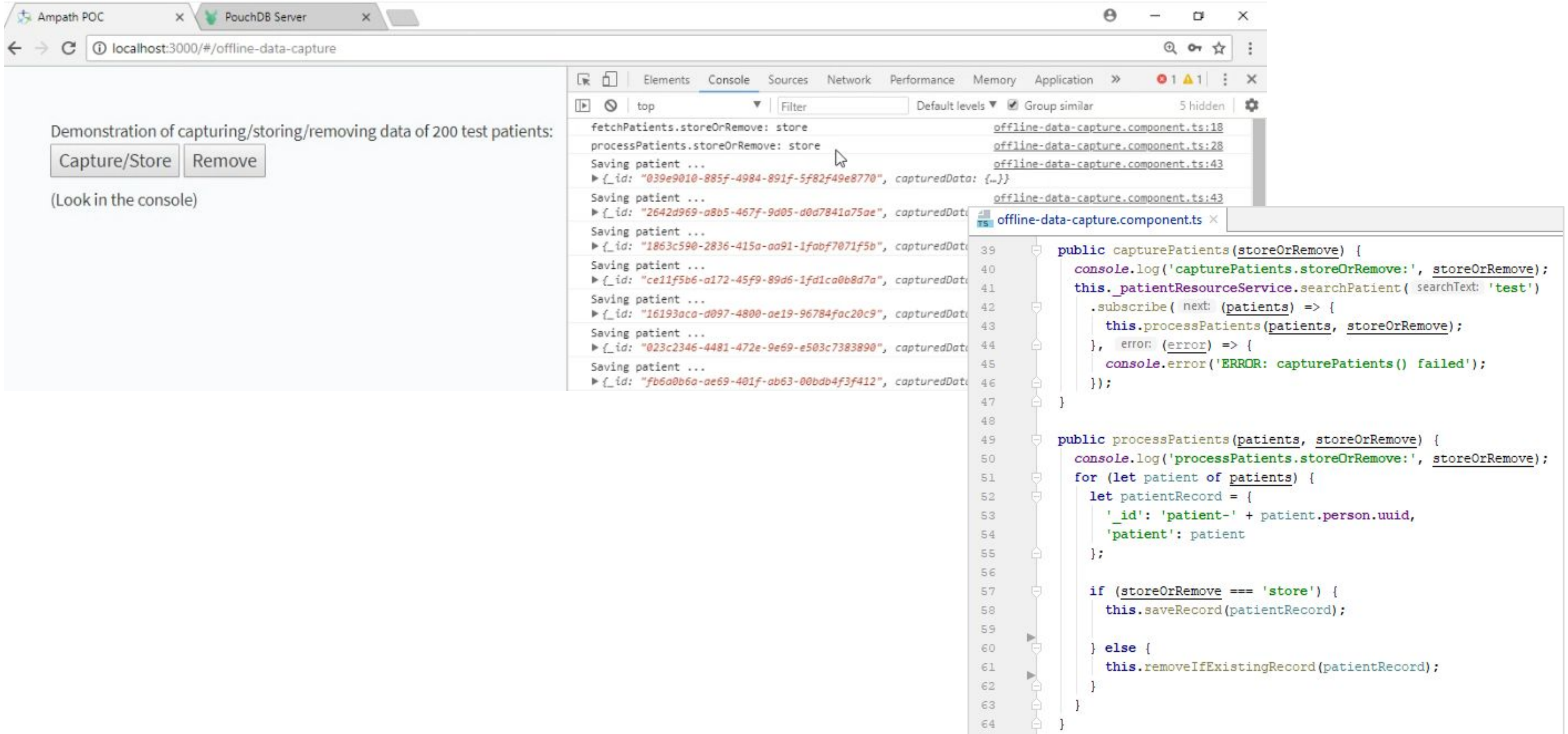
src/app/app-routing.module.ts

```
 import { Routes } from '@angular/router';
 import { NoContentComponent } from './no-content';
 import { FeedBackComponent } from './feedback';
+import { OfflineDataCaptureComponent } from './offline-data-capture/offline-data-capture.component';

 export const ROUTES: Routes = [
   {


   {path: 'form-visit-search', loadChildren:
    './patient-dashboard/common/form-visit-type-search/form-visit-type-search.module#' +
    'FormVisitTypeSearchModule'},
+  {path: 'offline-data-capture', component: OfflineDataCaptureComponent},
   {path: '**', component: NoContentComponent},
 ];
```

src/app/navigation/static-navbar/static-navbar.component.html

```
+         <li class="offline-storage" routerLinkActive="active">
+           Offline Storage<a routerLink="/offline-storage"> <i class="fa fa-user-md"></i>
+             <span class="hidden-xs hidden-sm">Offline Storage</span> </a>
```

Introducing our Offline Data Capture Component



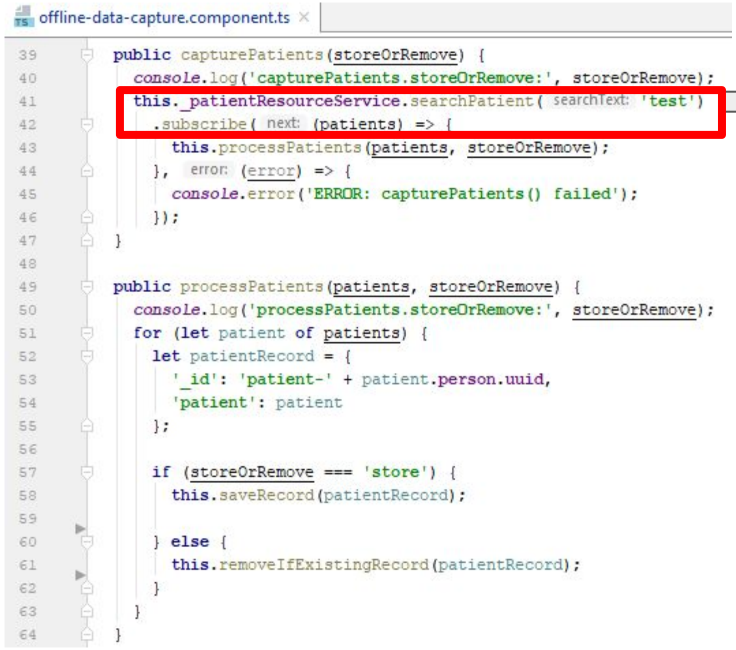
Demonstration of capturing/storing/removing data of 200 test patients:

(Look in the console)

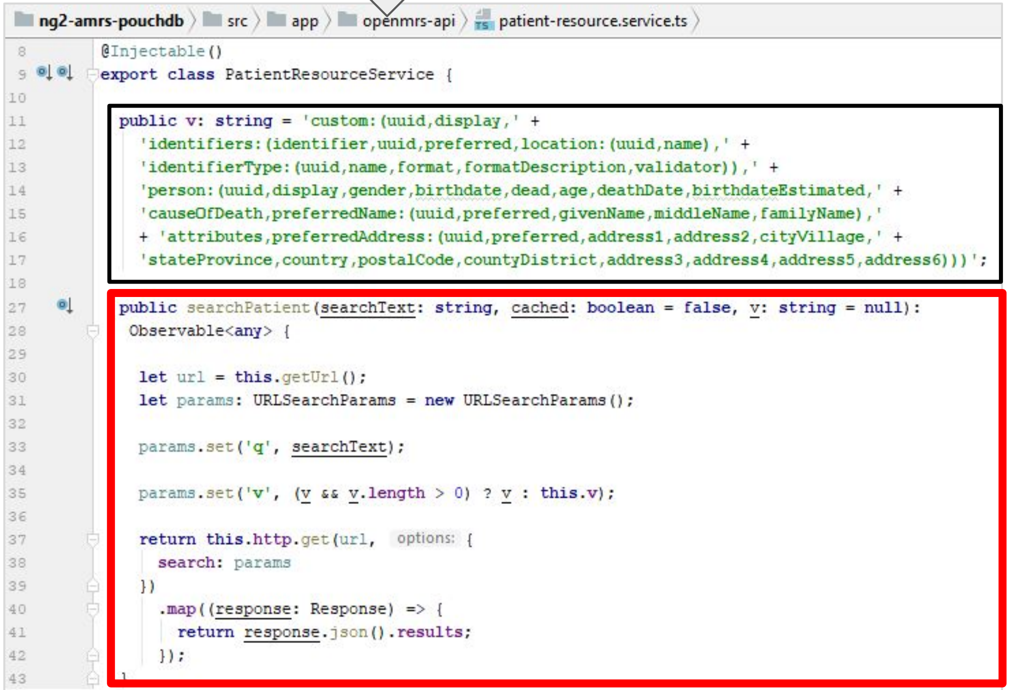
```
fetchPatients.storeOrRemove: store                                  offline-data-capture.component.ts:18
processPatients.storeOrRemove: store                                offline-data-capture.component.ts:28
Saving patient ...                                                  offline-data-capture.component.ts:43
▶ {_id: "039e9010-885f-4984-891f-5f82f49e8770", capturedData: {…}}
Saving patient ...                                                  offline-data-capture.component.ts:43
▶ {_id: "2642d969-a8b5-467f-9d05-d0d7841a75ae", capturedDat
Saving patient ...
▶ {_id: "1863c590-2836-415a-aa91-1fabf7071f5b", capturedDat
Saving patient ...
▶ {_id: "ce11f5b6-a172-45f9-89d6-1fd1ca0b8d7a", capturedDat
Saving patient ...
▶ {_id: "16193aca-d097-4800-ae19-96784fac20c9", capturedDat
Saving patient ...
▶ {_id: "023c2346-4481-472e-9e69-e503c7383890", capturedDat
Saving patient ...
▶ {_id: "fb6a0b6a-ae69-401f-ab63-00bdb4f3f412", capturedDat
```

```
public capturePatients(storeOrRemove) {
  console.log('capturePatients.storeOrRemove:', storeOrRemove);
  this._patientResourceService.searchPatient( searchText: 'test')
    .subscribe( next: (patients) => {
      this.processPatients(patients, storeOrRemove);
    },  error: (error) => {
      console.error('ERROR: capturePatients() failed');
    });
}

public processPatients(patients, storeOrRemove) {
  console.log('processPatients.storeOrRemove:', storeOrRemove);
  for (let patient of patients) {
    let patientRecord = {
      '_id': 'patient-' + patient.person.uuid,
      'patient': patient
    };

    if (storeOrRemove === 'store') {
      this.saveRecord(patientRecord);

    } else {
      this.removeIfExistingRecord(patientRecord);
    }
  }
}
```

Our capturePatients() uses an OpenMRS Service


offline-data-capture.component.ts

```
public capturePatients(storeOrRemove) {
  console.log('capturePatients.storeOrRemove:', storeOrRemove);
  this._patientResourceService.searchPatient( searchText: 'test')
    .subscribe( next: (patients) => {
      this.processPatients(patients, storeOrRemove);
    }, error: (error) => {
      console.error('ERROR: capturePatients() failed');
    });
}

public processPatients(patients, storeOrRemove) {
  console.log('processPatients.storeOrRemove:', storeOrRemove);
  for (let patient of patients) {
    let patientRecord = {
      '_id': 'patient-' + patient.person.uuid,
      'patient': patient
    };

    if (storeOrRemove === 'store') {
      this.saveRecord(patientRecord);

    } else {
      this.removeIfExistingRecord(patientRecord);
    }
  }
}
```


ng2-amrs-pouchdb 〉 src 〉 app 〉 openmrs-api 〉 patient-resource.service.ts

```
@Injectable()
export class PatientResourceService {

  public v: string = 'custom:(uuid,display,' +
    'identifiers:(identifier,uuid,preferred,location:(uuid,name),' +
    'identifierType:(uuid,name,format,formatDescription,validator)),' +
    'person:(uuid,display,gender,birthdate,dead,age,deathDate,birthdateEstimated,' +
    'causeOfDeath,preferredName:(uuid,preferred,givenName,middleName,familyName),'
    + 'attributes,preferredAddress:(uuid,preferred,address1,address2,cityVillage,' +
    'stateProvince,country,postalCode,countyDistrict,address3,address4,address5,address6)))';

  public searchPatient(searchText: string, cached: boolean = false, v: string = null):
   Observable<any> {

    let url = this.getUrl();
    let params: URLSearchParams = new URLSearchParams();

    params.set('q', searchText);

    params.set('v', (v && v.length > 0) ? v : this.v);

    return this.http.get(url, options: {
      search: params
    })
      .map((response: Response) => {
        return response.json().results;
      });
```

Sprint 5

Improving our Offline Data Capture Implementation

Sprint 5: Topics Covered

- Discussing the Conflict issues arising from our Offline Data Capture Service implemented in Sprint 4.
- Fixing our Offline Data Capture Service.
- Deciding what data is most useful to capture.
- Manipulating existing components to capture extensive data when a given patient is searched for.

Our Offline Data Capture Service had Problems...

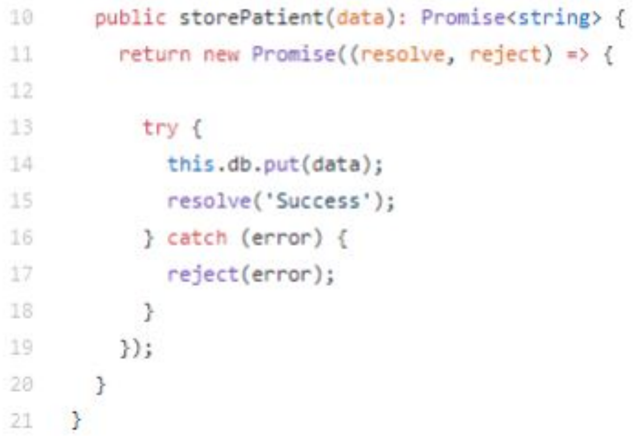
- Attempting to store captured data by a given ID more than once would cause a conflict (HTTP Status Code 409).

- During Sprint 5, we were able to identify the problem and update our Offline Data Capture Service accordingly.

Updating our Offline Data Capture Service

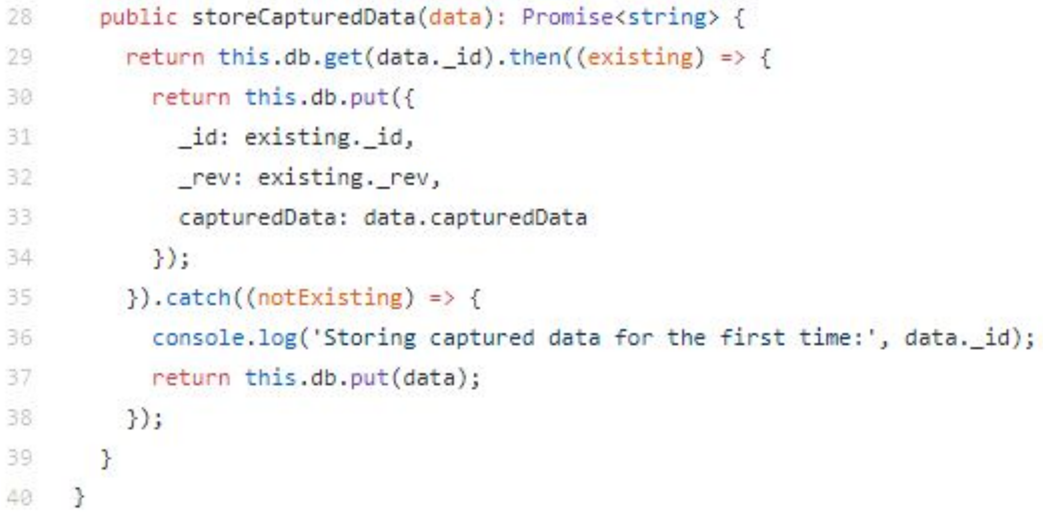
ng2-amrs-pouchdb / src / app / offline-data-capture / offline-data-capture.service.ts

Before...



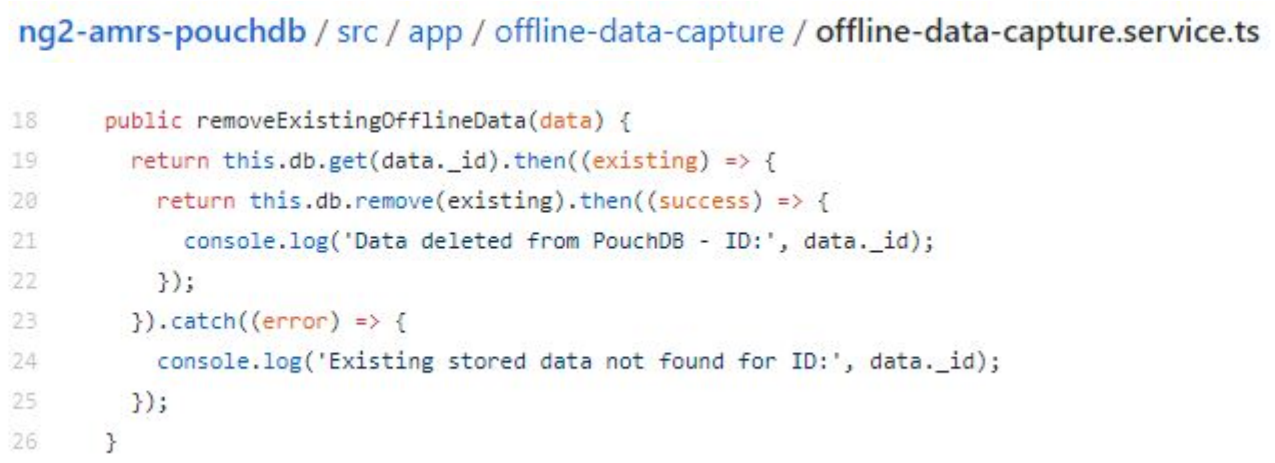
```
  public storePatient(data): Promise<string> {
    return new Promise((resolve, reject) => {

      try {
        this.db.put(data);
        resolve('Success');
      } catch (error) {
        reject(error);
      }
    });
  }
}
```

After:



```
  public storeCapturedData(data): Promise<string> {
    return this.db.get(data._id).then((existing) => {
      return this.db.put({
        _id: existing._id,
        _rev: existing._rev,
        capturedData: data.capturedData
      });
    }).catch((notExisting) => {
      console.log('Storing captured data for the first time:', data._id);
      return this.db.put(data);
    });
  }
}
```

We also added a PouchDB remove() function



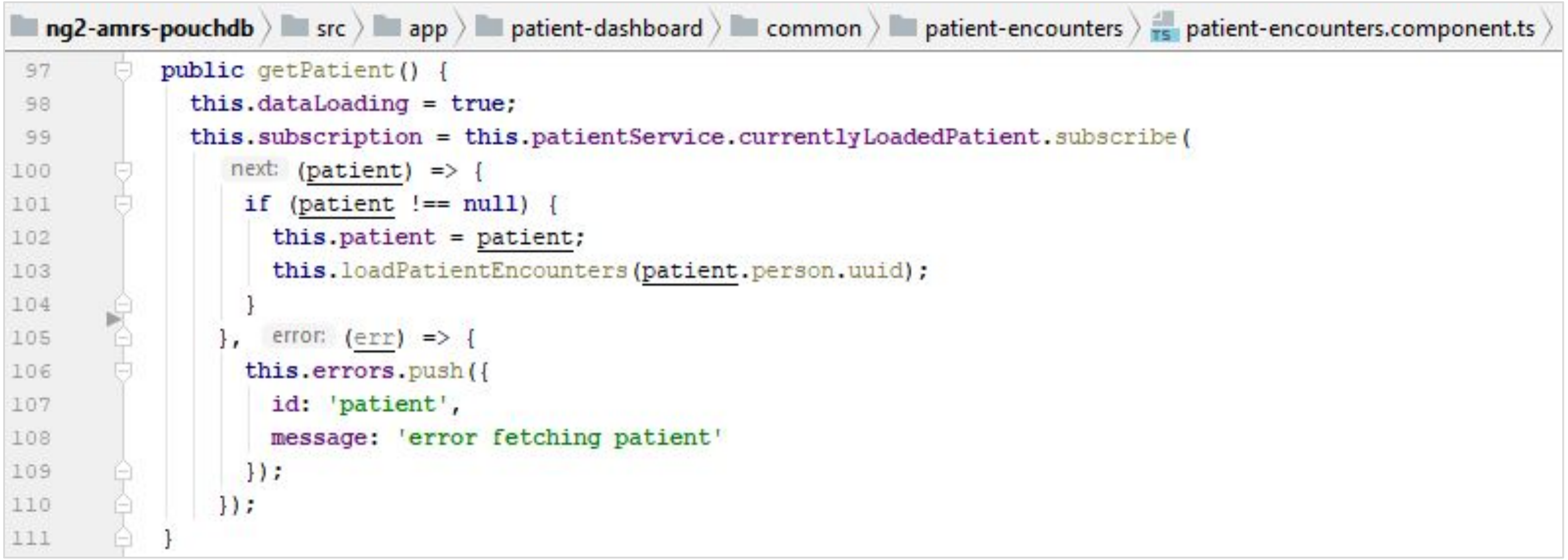
ng2-amrs-pouchdb / src / app / offline-data-capture / offline-data-capture.service.ts

```
public removeExistingOfflineData(data) {
  return this.db.get(data._id).then((existing) => {
    return this.db.remove(existing).then((success) => {
      console.log('Data deleted from PouchDB - ID:', data._id);
    });
  }).catch((error) => {
    console.log('Existing stored data not found for ID:', data._id);
  });
}
```

Deciding what to Capture: Exploring our Options

- Now that we’ve fixed our service, we wanted to capture more data.
- It’s worth noting at this point (midway through Sprint 5) that we have only been capturing basic patient information. (name, address, etc)
- We wanted to be able to capture more detailed patient information, such as vitals, patient visits, lab results, etc.
- We noticed the existing **Patient Encounters Component** was fetching a substantial amount of information when a given patient was searched for.

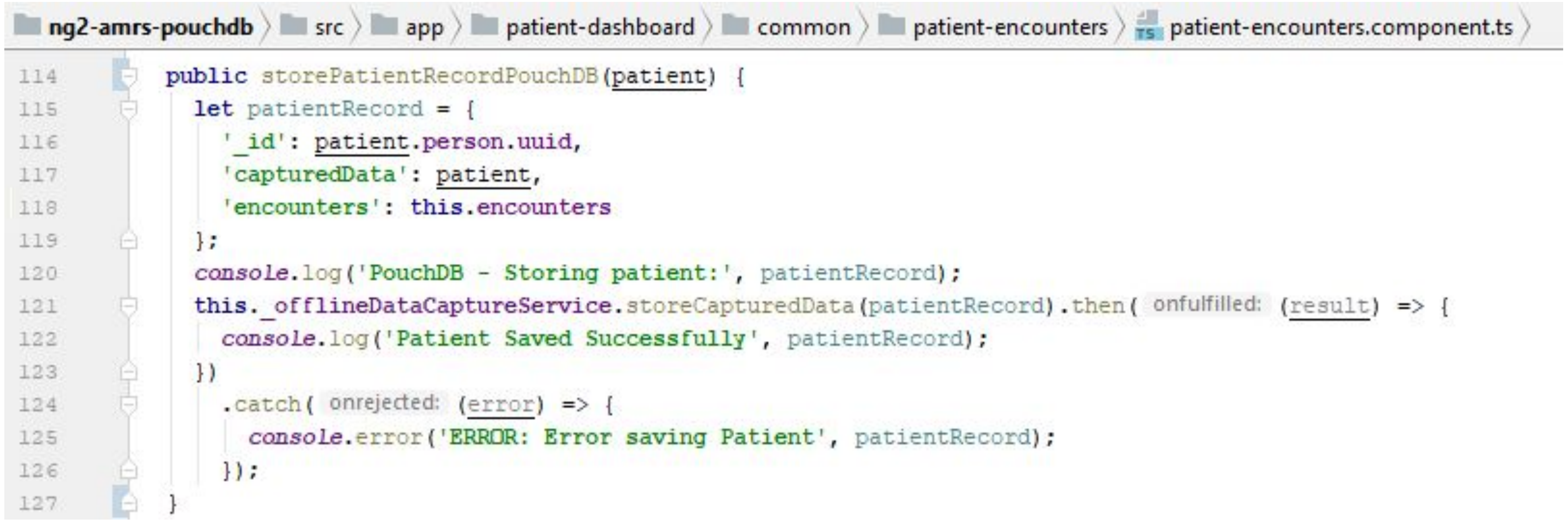
Existing AMPATH Patient Encounters Component



ng2-amrs-pouchdb 〉 src 〉 app 〉 patient-dashboard 〉 common 〉 patient-encounters 〉 patient-encounters.component.ts

```
public getPatient() {
  this.dataLoading = true;
  this.subscription = this.patientService.currentlyLoadedPatient.subscribe(
    next: (patient) => {
      if (patient !== null) {
        this.patient = patient;
        this.loadPatientEncounters(patient.person.uuid);
      }
    }, error: (err) => {
      this.errors.push({
        id: 'patient',
        message: 'error fetching patient'
      });
    });
}
```

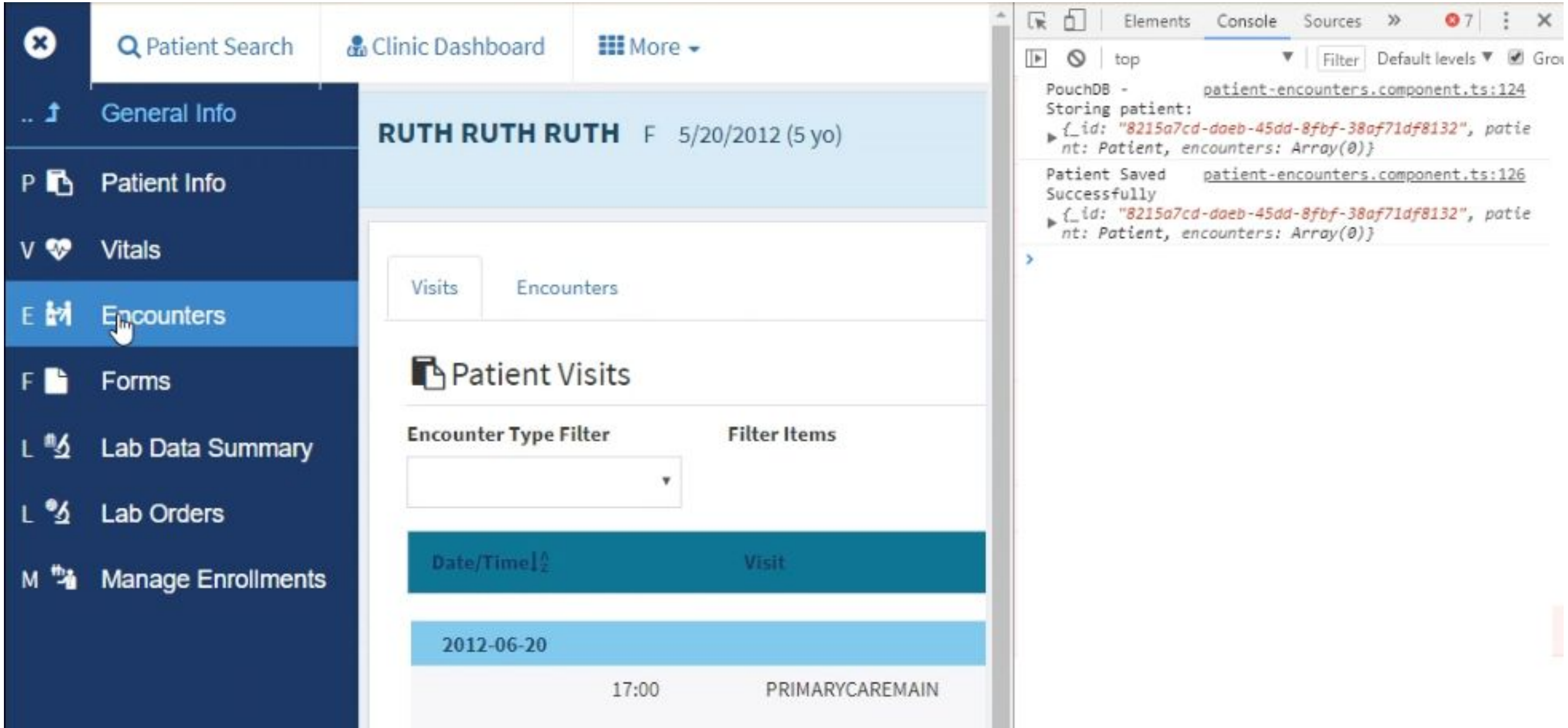
We added a function in it to call our service...


ng2-amrs-pouchdb 〉 src 〉 app 〉 patient-dashboard 〉 common 〉 patient-encounters 〉 patient-encounters.component.ts 〉

```
public storePatientRecordPouchDB(patient) {
  let patientRecord = {
    '_id': patient.person.uuid,
    'capturedData': patient,
    'encounters': this.encounters
  };
  console.log('PouchDB - Storing patient:', patientRecord);
  this._offlineDataCaptureService.storeCapturedData(patientRecord).then( onfulfilled: (result) => {
    console.log('Patient Saved Successfully', patientRecord);
  })
    .catch( onrejected: (error) => {
      console.error('ERROR: Error saving Patient', patientRecord);
    });
}
```

Then we called `this.storePatientRecordPouchDB(patient)` within the existing getPatient() function.

Demonstrating storing data offline when searched for...



RUTH RUTH RUTH F 5/20/2012 (5 yo)

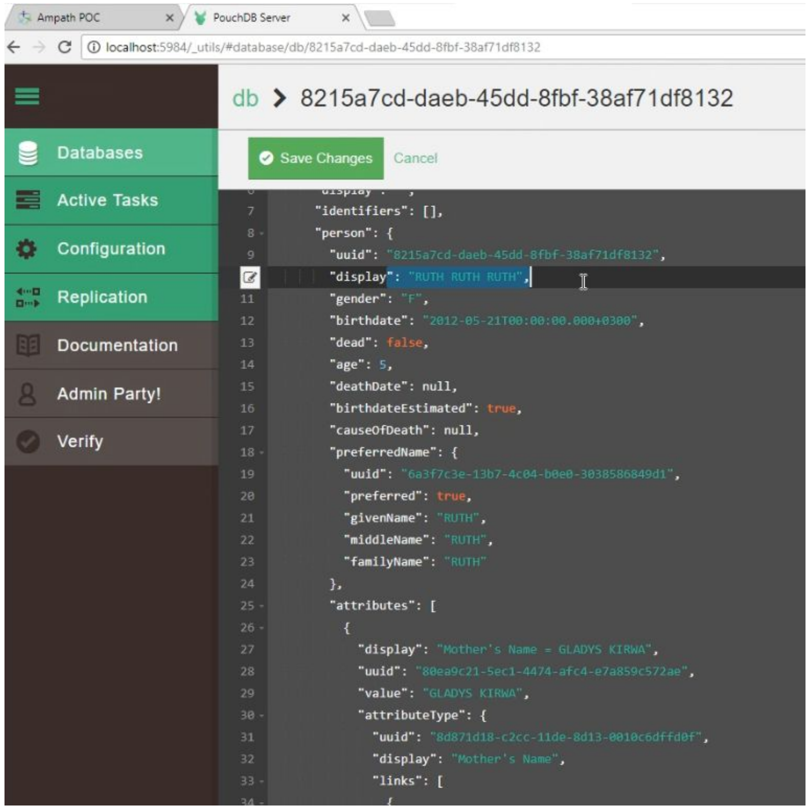
Patient Visits

Encounter Type Filter Filter Items

Date/Time	Visit
2012-06-20	
17:00	PRIMARYCAREMAIN

```
PouchDB -                patient-encounters.component.ts:124
Storing patient:
▶ {_id: "8215a7cd-daeb-45dd-8fbf-38af71df8132", patie
  nt: Patient, encounters: Array(0)}
Patient Saved            patient-encounters.component.ts:126
Successfully
▶ {_id: "8215a7cd-daeb-45dd-8fbf-38af71df8132", patie
  nt: Patient, encounters: Array(0)}
>
```

...and the extensive Patient Data is Stored in PouchDB!



```
    "identifiers": [],
    "person": {
      "uuid": "8215a7cd-daeb-45dd-8fbf-38af71df8132",
      "display": "RUTH RUTH RUTH",
      "gender": "F",
      "birthdate": "2012-05-21T00:00:00.000+0300",
      "dead": false,
      "age": 5,
      "deathDate": null,
      "birthdateEstimated": true,
      "causeOfDeath": null,
      "preferredName": {
        "uuid": "6a3f7c3e-13b7-4c04-b0e0-3038586849d1",
        "preferred": true,
        "givenName": "RUTH",
        "middleName": "RUTH",
        "familyName": "RUTH"
      },
      "attributes": [
        {
          "display": "Mother's Name = GLADYS KIRWA",
          "uuid": "80ea9c21-5ec1-4474-afc4-e7a859c572ae",
          "value": "GLADYS KIRWA",
          "attributeType": {
            "uuid": "8d871d18-c2cc-11de-8d13-0010c6dffd0f",
            "display": "Mother's Name",
            "links": [
```

```
Approximate Lines: 2500+
File Size: ~300kb per patient
```

- We're probably capturing too much data here.
- We have to keep in mind the customer.
- (i.e. Limited internet connectivity and limited storage space)
- We began addressing these issues in Sprint 6.

Sprint 6

Capturing and Displaying Specific Categories

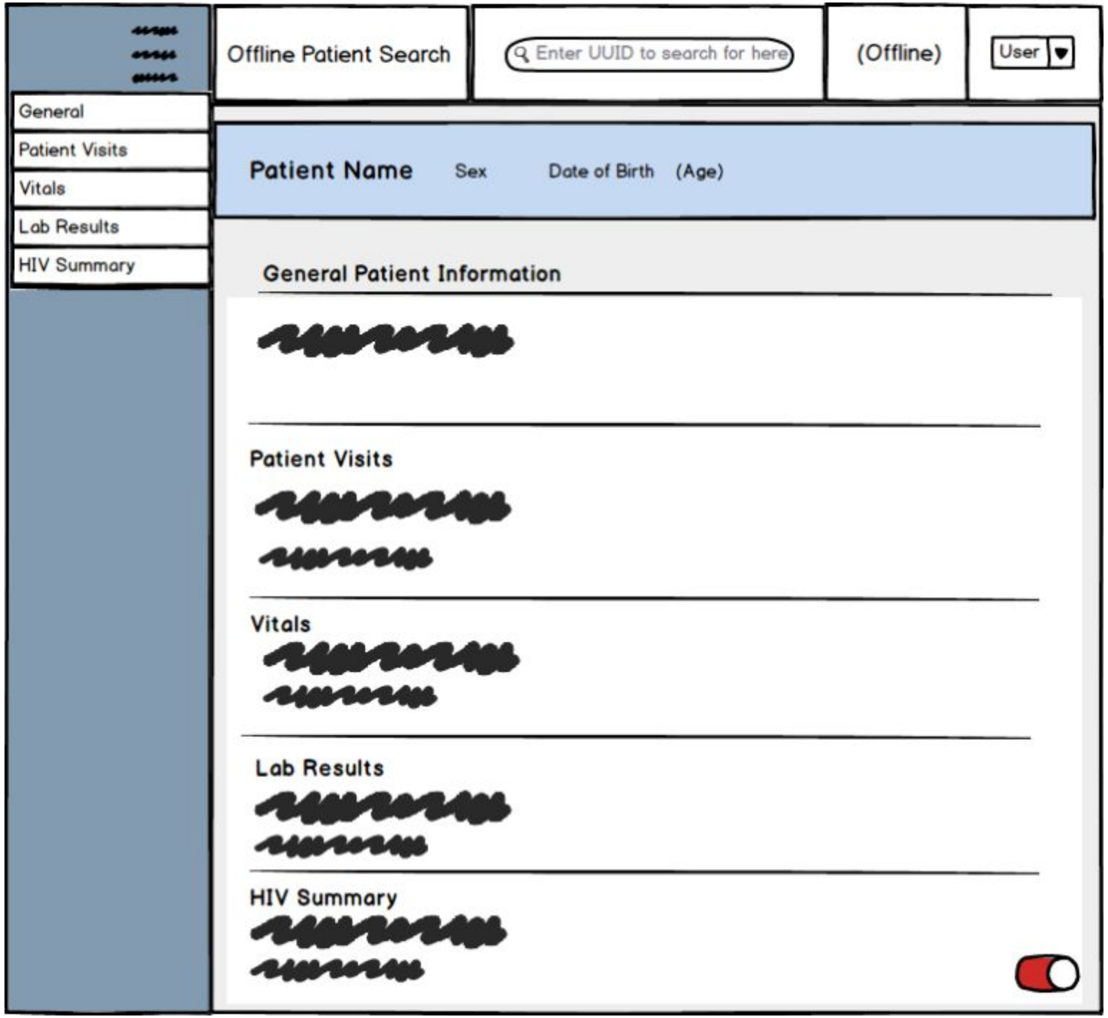
Sprint 6: Topics Covered

- Reconsidering the extensive amount of data currently being captured.
- Creating a mock diagram demonstrating GUI implementation ideas.
- Developing an offline GUI to display specific offline data.
- Capturing the specific data requested by the AMPATH team.
- Storing this specific data into the PouchDB database.
- Retrieving data from PouchDB when user is offline to display in our GUI.

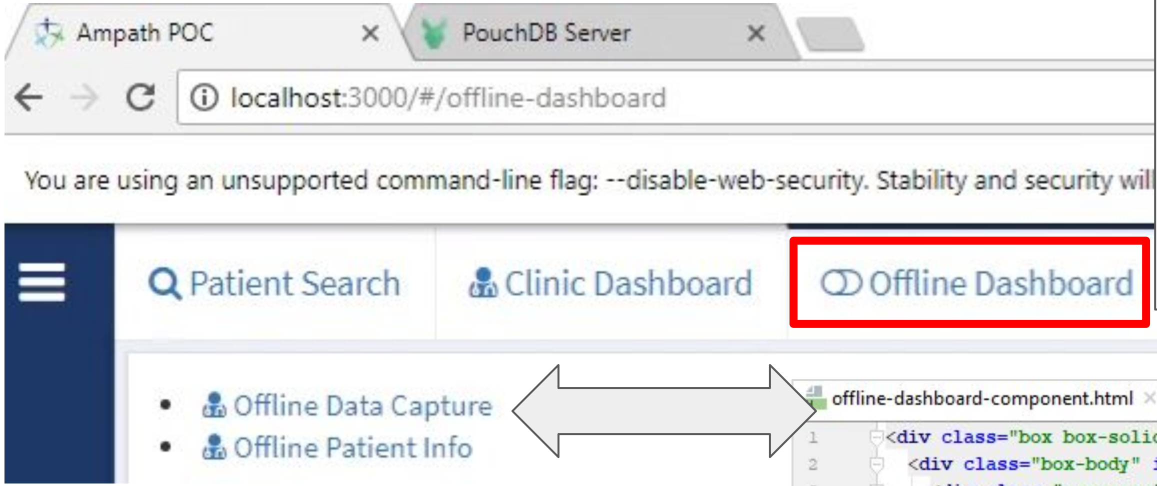
Mock Diagram for Offline GUI

- Based on the mockup by the **offline-storage team** with small changes.

- The categories listed here are based on what we and the AMPATH team decided to be most useful to capture for offline use.

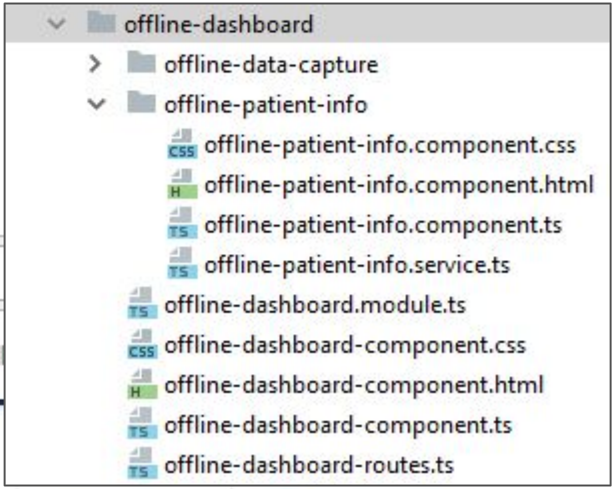


Offline GUI: Implementation

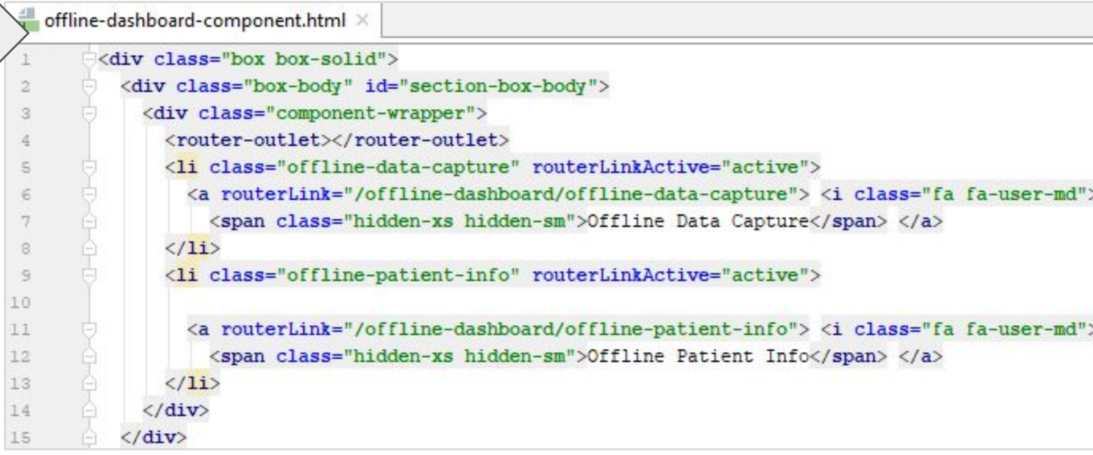


You are using an unsupported command-line flag: --disable-web-security. Stability and security wil

- Offline Data Capture
- Offline Patient Info



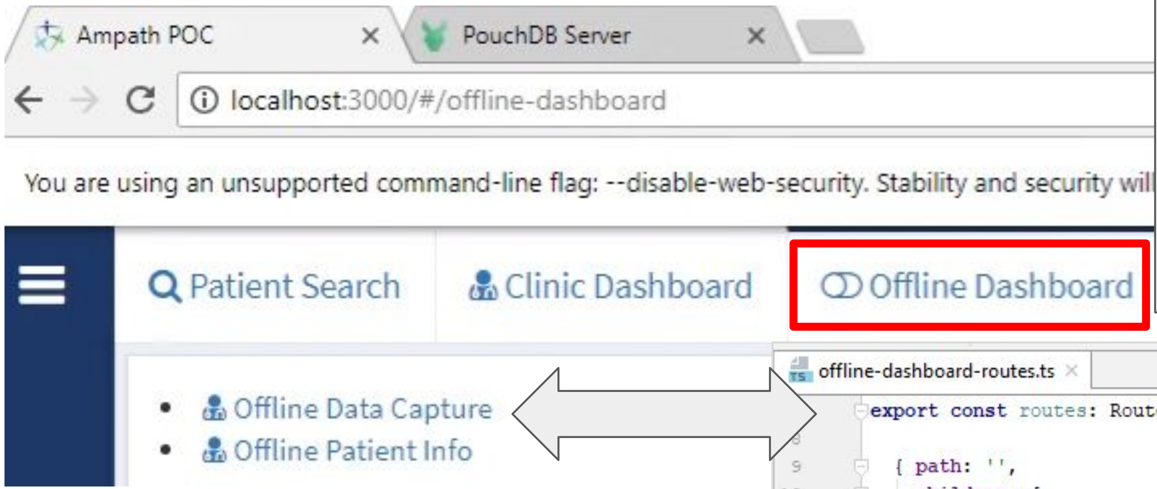
```
offline-dashboard
    offline-data-capture
    offline-patient-info
        offline-patient-info.component.css
        offline-patient-info.component.html
        offline-patient-info.component.ts
        offline-patient-info.service.ts
    offline-dashboard.module.ts
    offline-dashboard-component.css
    offline-dashboard-component.html
    offline-dashboard-component.ts
    offline-dashboard-routes.ts
```



offline-dashboard-component.html

```
<div class="box box-solid">
  <div class="box-body" id="section-box-body">
    <div class="component-wrapper">
      <router-outlet></router-outlet>
      <li class="offline-data-capture" routerLinkActive="active">
        <a routerLink="/offline-dashboard/offline-data-capture"> <i class="fa fa-user-md"
          <span class="hidden-xs hidden-sm">Offline Data Capture</span> </a>
      </li>
      <li class="offline-patient-info" routerLinkActive="active">

        <a routerLink="/offline-dashboard/offline-patient-info"> <i class="fa fa-user-md"
          <span class="hidden-xs hidden-sm">Offline Patient Info</span> </a>
      </li>
    </div>
  </div>
```

Offline GUI: Implementation



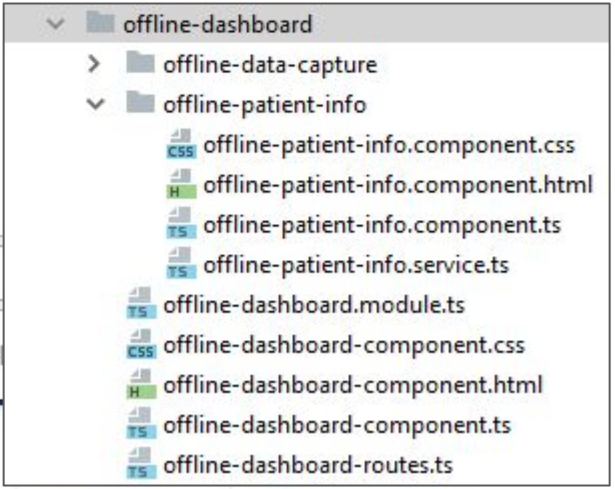
Ampath POC

PouchDB Server

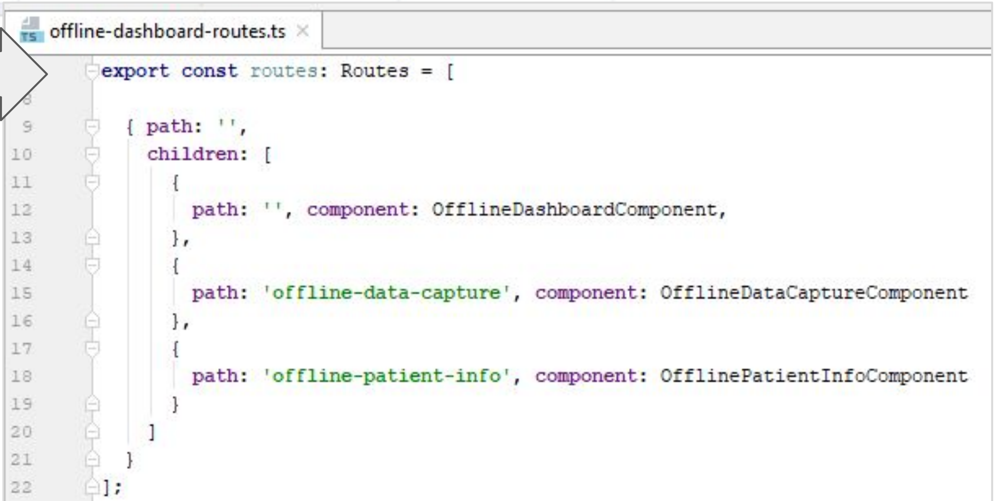
localhost:3000/#/offline-dashboard

You are using an unsupported command-line flag: --disable-web-security. Stability and security will

- Offline Data Capture
- Offline Patient Info



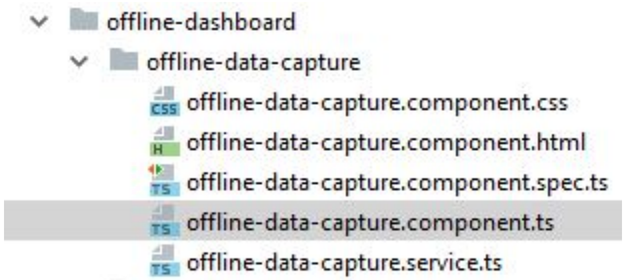
```
offline-dashboard
    offline-data-capture
    offline-patient-info
        offline-patient-info.component.css
        offline-patient-info.component.html
        offline-patient-info.component.ts
        offline-patient-info.service.ts
    offline-dashboard.module.ts
    offline-dashboard-component.css
    offline-dashboard-component.html
    offline-dashboard-component.ts
    offline-dashboard-routes.ts
```



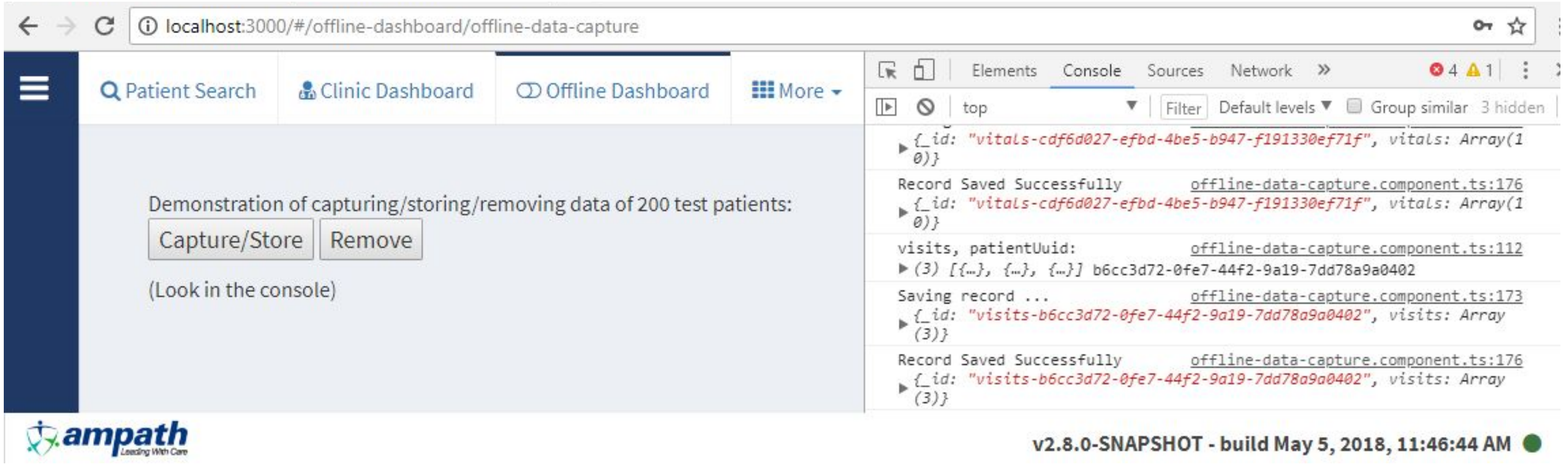
offline-dashboard-routes.ts

```
export const routes: Routes = [

  { path: '',
    children: [
      {
        path: '', component: OfflineDashboardComponent,
      },
      {
        path: 'offline-data-capture', component: OfflineDataCaptureComponent
      },
      {
        path: 'offline-patient-info', component: OfflinePatientInfoComponent
      }
    ]
  }
];
```

Updated: Capturing Specific Data when Online



```
offline-dashboard
  offline-data-capture
    offline-data-capture.component.css
    offline-data-capture.component.html
    offline-data-capture.component.spec.ts
    offline-data-capture.component.ts
    offline-data-capture.service.ts
```



Demonstration of capturing/storing/removing data of 200 test patients:

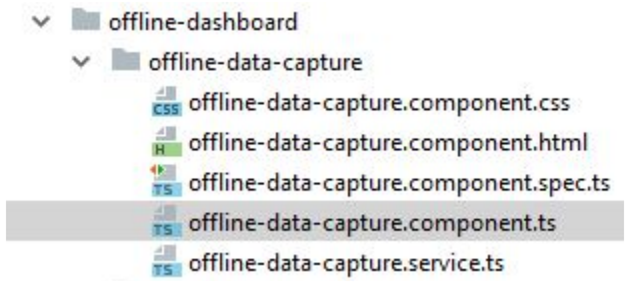
Capture/Store Remove

(Look in the console)

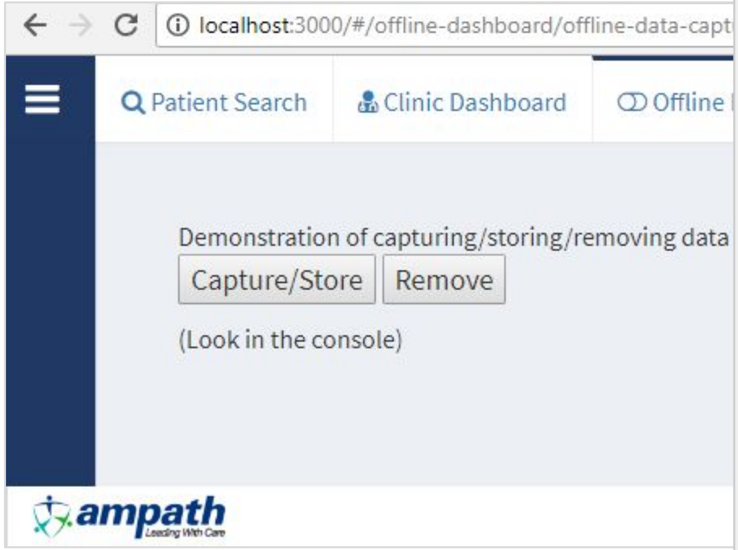
```
▶ {_id: "vitals-cdf6d027-efbd-4be5-b947-f191330ef71f", vitals: Array(1
  0)}
Record Saved Successfully          offline-data-capture.component.ts:176
▶ {_id: "vitals-cdf6d027-efbd-4be5-b947-f191330ef71f", vitals: Array(1
  0)}
visits, patientUuid:               offline-data-capture.component.ts:112
▶ (3) [{…}, {…}, {…}] b6cc3d72-0fe7-44f2-9a19-7dd78a9a0402
Saving record ...                  offline-data-capture.component.ts:173
▶ {_id: "visits-b6cc3d72-0fe7-44f2-9a19-7dd78a9a0402", visits: Array
  (3)}
Record Saved Successfully          offline-data-capture.component.ts:176
▶ {_id: "visits-b6cc3d72-0fe7-44f2-9a19-7dd78a9a0402", visits: Array
  (3)}
```

v2.8.0-SNAPSHOT - build May 5, 2018, 11:46:44 AM

Updated: Capturing Specific Data when Online

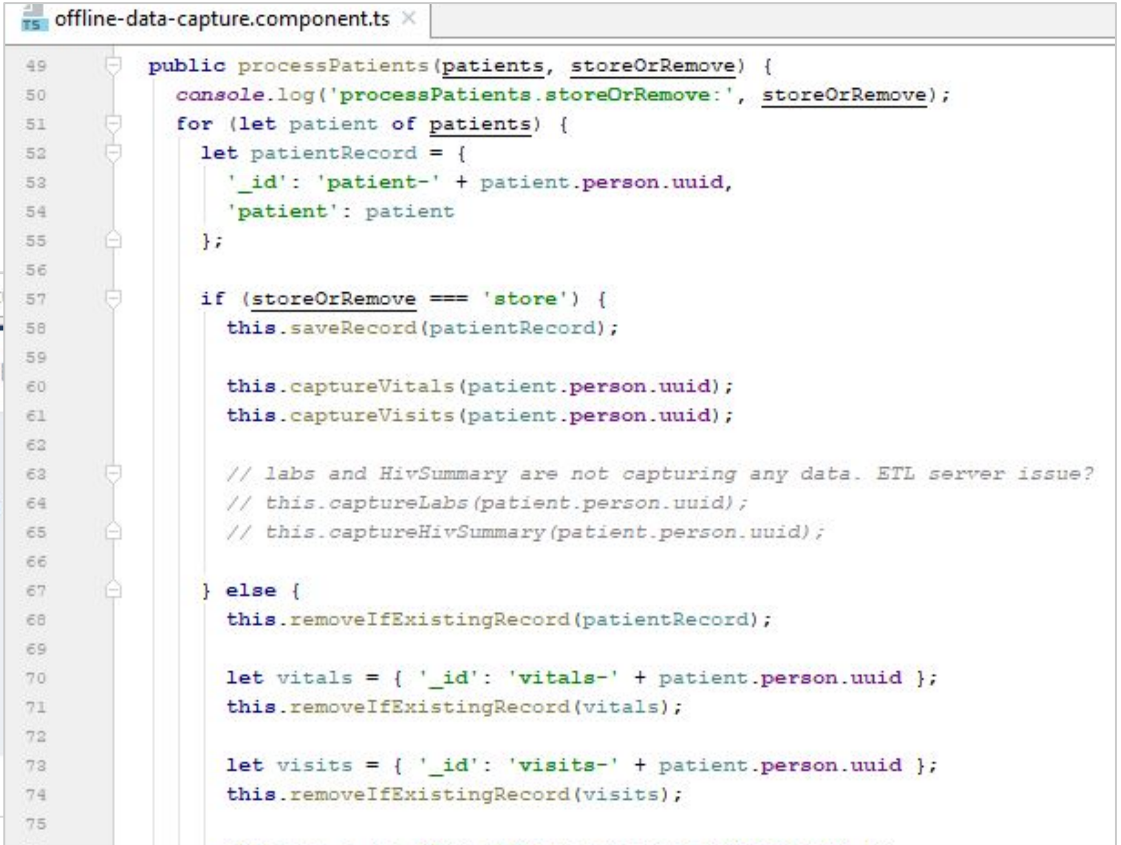


```
offline-dashboard
  offline-data-capture
    offline-data-capture.component.css
    offline-data-capture.component.html
    offline-data-capture.component.spec.ts
    offline-data-capture.component.ts
    offline-data-capture.service.ts
```



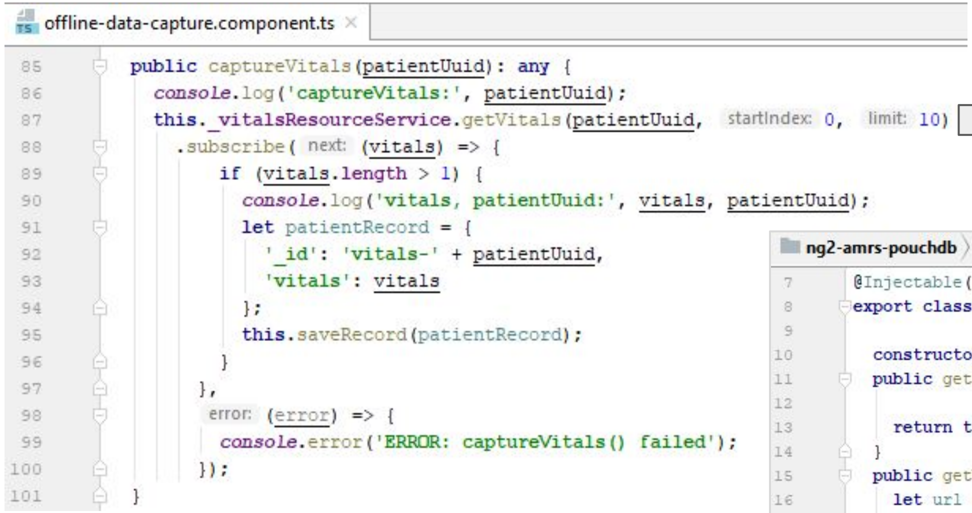
Demonstration of capturing/storing/removing data

(Look in the console)



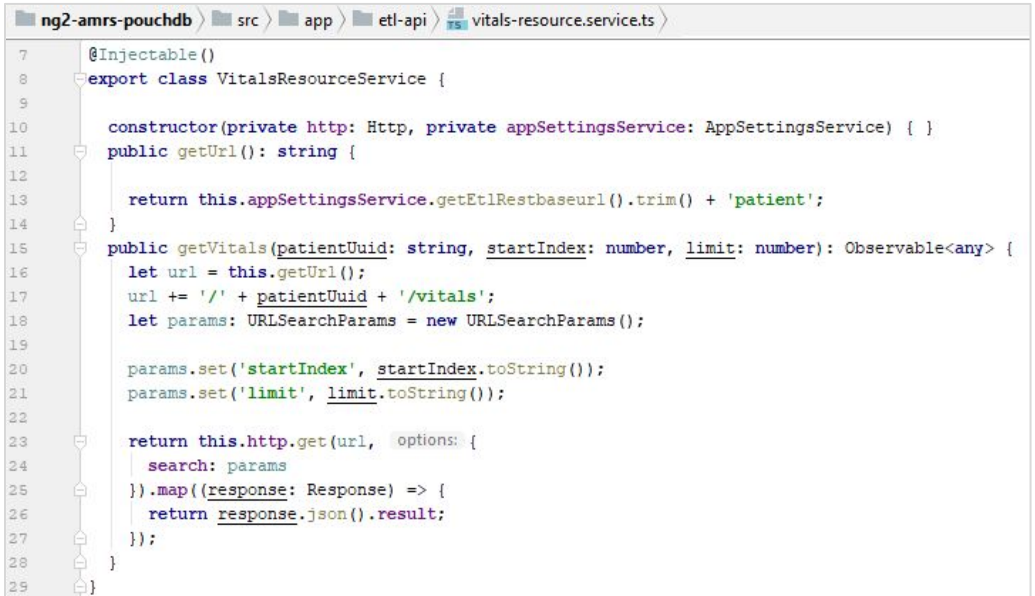
```
public processPatients(patients, storeOrRemove) {
  console.log('processPatients.storeOrRemove:', storeOrRemove);
  for (let patient of patients) {
    let patientRecord = {
      '_id': 'patient-' + patient.person.uuid,
      'patient': patient
    };

    if (storeOrRemove === 'store') {
      this.saveRecord(patientRecord);

      this.captureVitals(patient.person.uuid);
      this.captureVisits(patient.person.uuid);

      // labs and HivSummary are not capturing any data. ETL server issue?
      // this.captureLabs(patient.person.uuid);
      // this.captureHivSummary(patient.person.uuid);

    } else {
      this.removeIfExistingRecord(patientRecord);

      let vitals = { '_id': 'vitals-' + patient.person.uuid };
      this.removeIfExistingRecord(vitals);

      let visits = { '_id': 'visits-' + patient.person.uuid };
      this.removeIfExistingRecord(visits);

```

Our captureVitals() uses ETL VitalsResourceService



offline-data-capture.component.ts

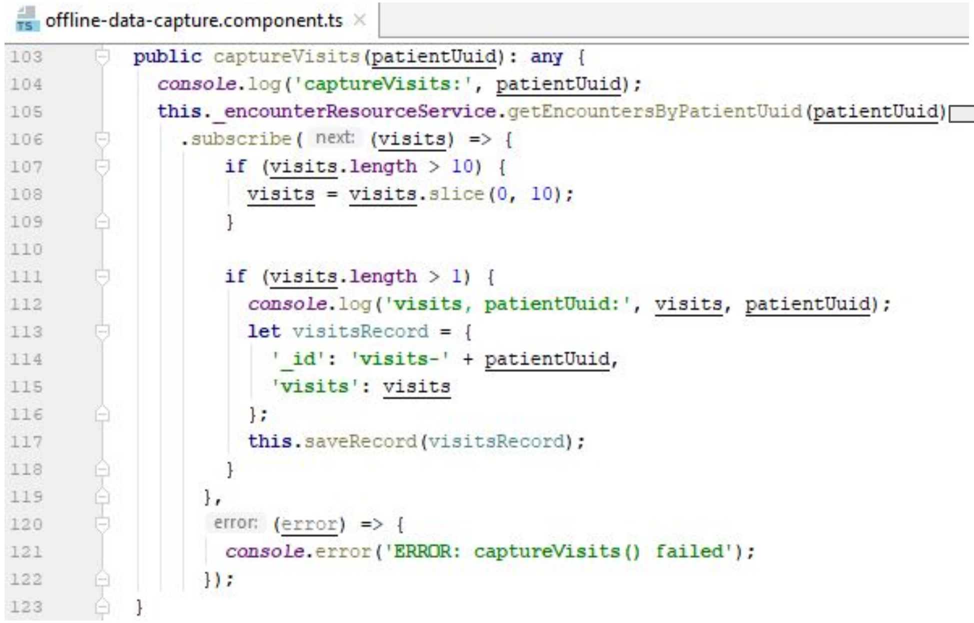
```
public captureVitals(patientUuid): any {
  console.log('captureVitals:', patientUuid);
  this._vitalsResourceService.getVitals(patientUuid,  startIndex: 0,  limit: 10)
    .subscribe( next: (vitals) => {
        if (vitals.length > 1) {
          console.log('vitals, patientUuid:', vitals, patientUuid);
          let patientRecord = {
            '_id': 'vitals-' + patientUuid,
            'vitals': vitals
          };
          this.saveRecord(patientRecord);
        }
      },
       error: (error) => {
        console.error('ERROR: captureVitals() failed');
      });
}
```



ng2-amrs-pouchdb 〉 src 〉 app 〉 etl-api 〉 vitals-resource.service.ts 〉

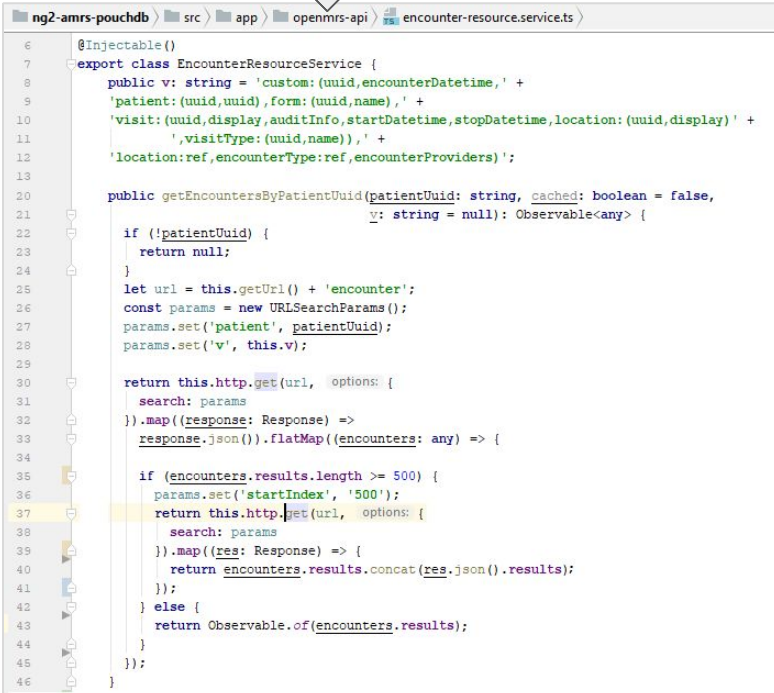
```
@Injectable()
export class VitalsResourceService {

  constructor(private http: Http, private appSettingsService: AppSettingsService) { }
  public getUrl(): string {

    return this.appSettingsService.getEtlRestbaseurl().trim() + 'patient';
  }
  public getVitals(patientUuid: string, startIndex: number, limit: number): Observable<any> {
    let url = this.getUrl();
    url += '/' + patientUuid + '/vitals';
    let params: URLSearchParams = new URLSearchParams();

    params.set('startIndex', startIndex.toString());
    params.set('limit', limit.toString());

    return this.http.get(url,  options: {
      search: params
    }).map((response: Response) => {
      return response.json().result;
    });
  }
}
```

Our captureVisits() uses an OpenMRS Service



offline-data-capture.component.ts

```
public captureVisits(patientUuid): any {
  console.log('captureVisits:', patientUuid);
  this._encounterResourceService.getEncountersByPatientUuid(patientUuid)
    .subscribe( next: (visits) => {
        if (visits.length > 10) {
          visits = visits.slice(0, 10);
        }

        if (visits.length > 1) {
          console.log('visits, patientUuid:', visits, patientUuid);
          let visitsRecord = {
            '_id': 'visits-' + patientUuid,
            'visits': visits
          };
          this.saveRecord(visitsRecord);
        }
      },
      error: (error) => {
        console.error('ERROR: captureVisits() failed');
      });
}
```



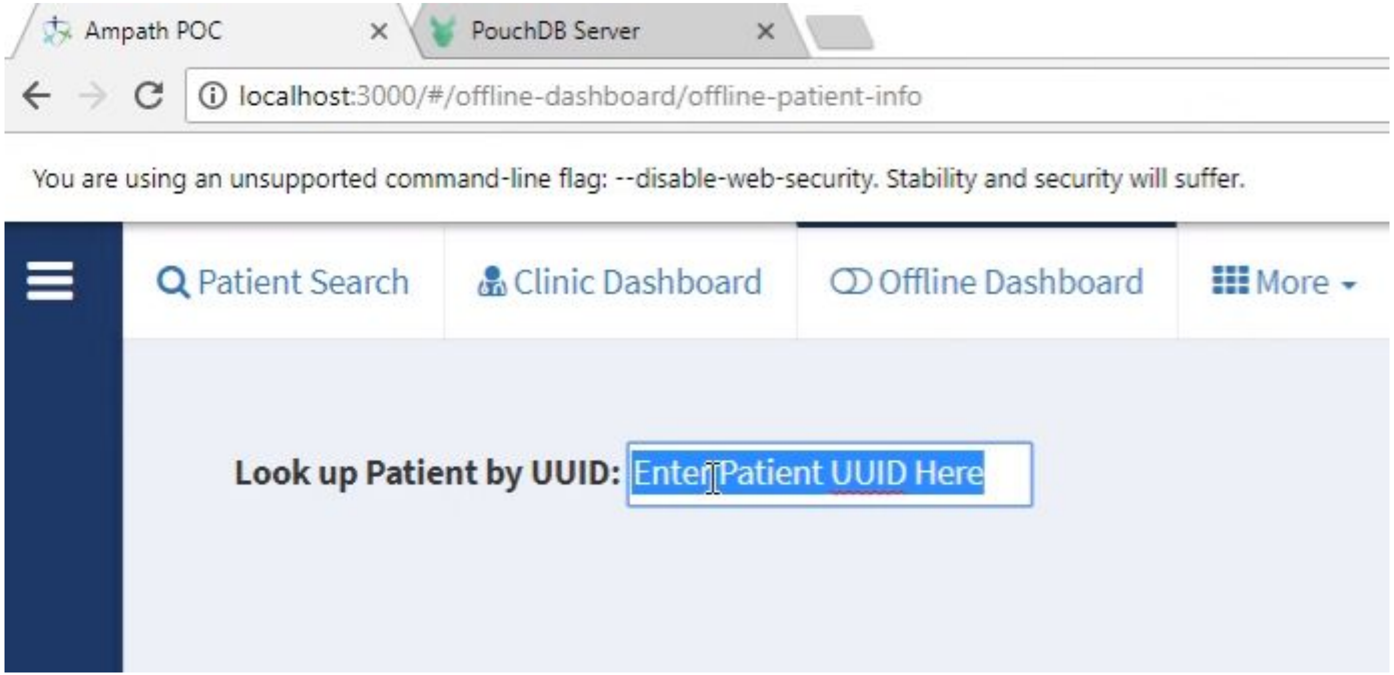
ng2-amrs-pouchdb 〉 src 〉 app 〉 openmrs-api 〉 encounter-resource.service.ts 〉

```
@Injectable()
export class EncounterResourceService {
    public v: string = 'custom:(uuid,encounterDatetime,' +
    'patient:(uuid,uuid),form:(uuid,name),' +
    'visit:(uuid,display,auditInfo,startDatetime,stopDatetime,location:(uuid,display)' +
            ',visitType:(uuid,name)),' +
    'location:ref,encounterType:ref,encounterProviders)';

    public getEncountersByPatientUuid(patientUuid: string, cached: boolean = false,
                                      v: string = null): Observable<any> {
      if (!patientUuid) {
        return null;
      }
      let url = this.getUrl() + 'encounter';
      const params = new URLSearchParams();
      params.set('patient', patientUuid);
      params.set('v', this.v);

      return this.http.get(url,  options: {
        search: params
      }).map((response: Response) =>
        response.json()).flatMap((encounters: any) => {

        if (encounters.results.length >= 500) {
          params.set('startIndex', '500');
          return this.http.get(url,  options: {
            search: params
          }).map((res: Response) => {
            return encounters.results.concat(res.json().results);
          });
        } else {
          return Observable.of(encounters.results);
        }
      });
    }
```

Labs and HIV Summary

- Note: We implemented functions to capture Labs/HIV Summary data as well.
- Unfortunately, we were unable to verify if these functions worked properly.
- It appears this may be due to either one or both of the following reasons:
    1. We could not properly access the ETL server for this particular data, or
    2. There was no data available for Labs/HIV Summary.
       (Nothing was captured when we tried to run the functions)

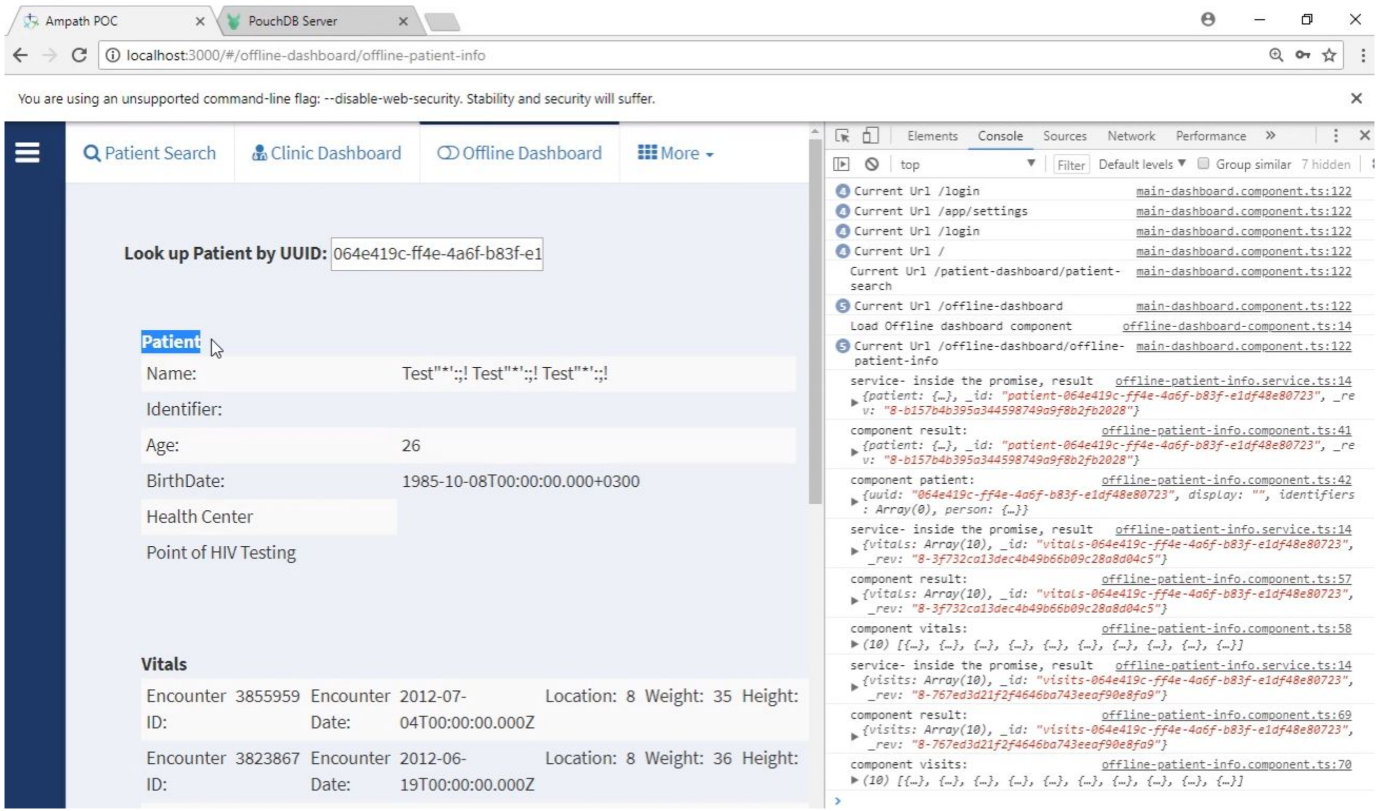
Displaying PouchDB data in Offline GUI



You are using an unsupported command-line flag: --disable-web-security. Stability and security will suffer.

Look up Patient by UUID: Enter Patient UUID Here

Displaying PouchDB data in Offline GUI



You are using an unsupported command-line flag: --disable-web-security. Stability and security will suffer.

Look up Patient by UUID: 064e419c-ff4e-4a6f-b83f-e1

Patient

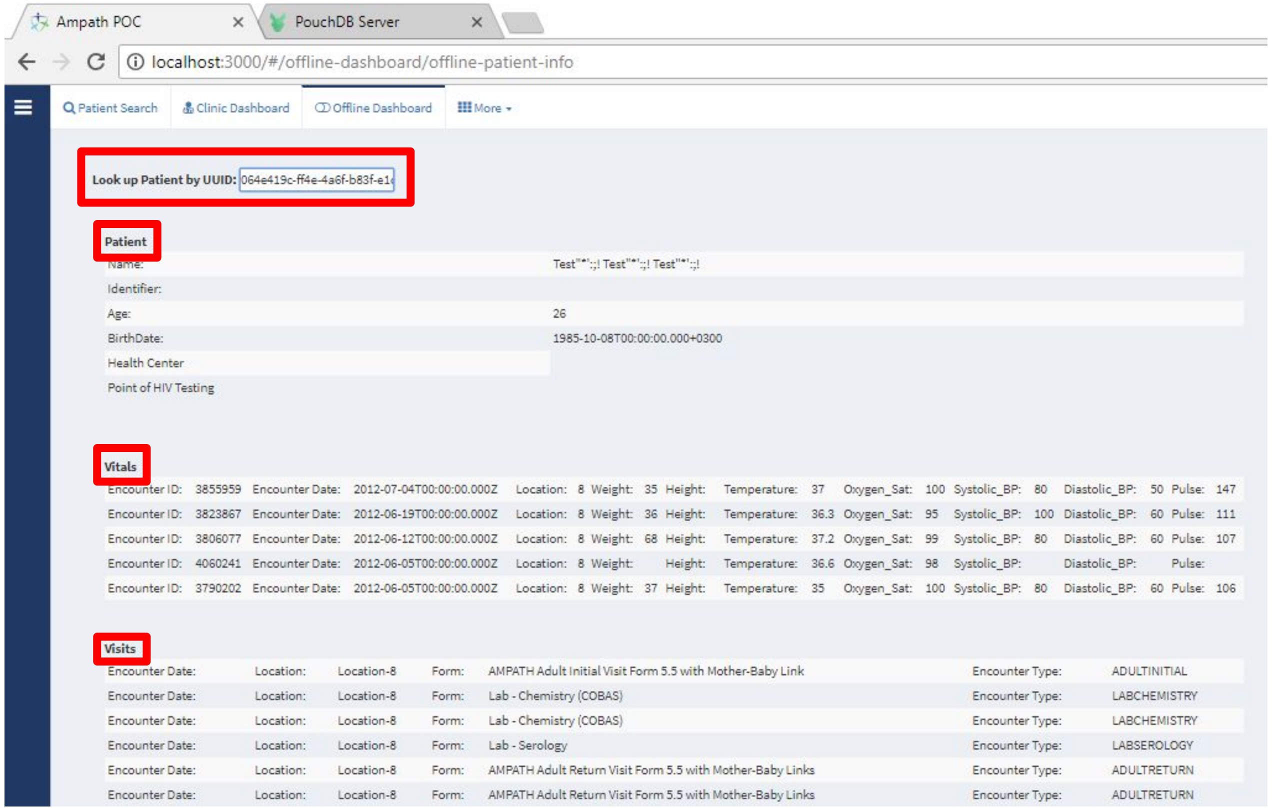
Name:	Test"*':;! Test"*':;! Test"*':;!
Identifier:	
Age:	26
BirthDate:	1985-10-08T00:00:00.000+0300
Health Center	
Point of HIV Testing	

Vitals

Encounter ID:	3855959	Encounter Date:	2012-07-04T00:00:00.000Z	Location:	8	Weight:	35	Height:
Encounter ID:	3823867	Encounter Date:	2012-06-19T00:00:00.000Z	Location:	8	Weight:	36	Height:

```
4 Current Url /login                                  main-dashboard.component.ts:122
4 Current Url /app/settings                           main-dashboard.component.ts:122
4 Current Url /login                                  main-dashboard.component.ts:122
4 Current Url /                                       main-dashboard.component.ts:122
  Current Url /patient-dashboard/patient-             main-dashboard.component.ts:122
  search
5 Current Url /offline-dashboard                      main-dashboard.component.ts:122
  Load Offline dashboard component                 offline-dashboard-component.ts:14
5 Current Url /offline-dashboard/offline-             main-dashboard.component.ts:122
  patient-info
  service- inside the promise, result          offline-patient-info.service.ts:14
  ▶ {patient: {…}, _id: "patient-064e419c-ff4e-4a6f-b83f-e1df48e80723", _re
    v: "8-b157b4b395a344598749a9f8b2fb2028"}
  component result:                          offline-patient-info.component.ts:41
  ▶ {patient: {…}, _id: "patient-064e419c-ff4e-4a6f-b83f-e1df48e80723", _re
    v: "8-b157b4b395a344598749a9f8b2fb2028"}
  component patient:                         offline-patient-info.component.ts:42
  ▶ {uuid: "064e419c-ff4e-4a6f-b83f-e1df48e80723", display: "", identifiers
    : Array(0), person: {…}}
  service- inside the promise, result          offline-patient-info.service.ts:14
  ▶ {vitals: Array(10), _id: "vitals-064e419c-ff4e-4a6f-b83f-e1df48e80723",
     _rev: "8-3f732ca13dec4b49b66b09c28a8d04c5"}
  component result:                          offline-patient-info.component.ts:57
  ▶ {vitals: Array(10), _id: "vitals-064e419c-ff4e-4a6f-b83f-e1df48e80723",
     _rev: "8-3f732ca13dec4b49b66b09c28a8d04c5"}
  component vitals:                          offline-patient-info.component.ts:58
  ▶ (10) [{…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}]
  service- inside the promise, result          offline-patient-info.service.ts:14
  ▶ {visits: Array(10), _id: "visits-064e419c-ff4e-4a6f-b83f-e1df48e80723",
     _rev: "8-767ed3d21f2f4646ba743eeaf90e8fa9"}
  component result:                          offline-patient-info.component.ts:69
  ▶ {visits: Array(10), _id: "visits-064e419c-ff4e-4a6f-b83f-e1df48e80723",
     _rev: "8-767ed3d21f2f4646ba743eeaf90e8fa9"}
  component visits:                          offline-patient-info.component.ts:70
  ▶ (10) [{…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}]
>
```

Displaying PouchDB data in Offline GUI



Look up Patient by UUID: 064e419c-ff4e-4a6f-b83f-e1

Patient

Name:	Test"*':;! Test"*':;! Test"*':;!
Identifier:	
Age:	26
BirthDate:	1985-10-08T00:00:00.000+0300
Health Center	
Point of HIV Testing	

Vitals

Encounter ID:	3855959	Encounter Date:	2012-07-04T00:00:00.000Z	Location:	8	Weight:	35	Height:		Temperature:	37	Oxygen_Sat:	100	Systolic_BP:	80	Diastolic_BP:	50	Pulse:	147
Encounter ID:	3823867	Encounter Date:	2012-06-19T00:00:00.000Z	Location:	8	Weight:	36	Height:		Temperature:	36.3	Oxygen_Sat:	95	Systolic_BP:	100	Diastolic_BP:	60	Pulse:	111
Encounter ID:	3806077	Encounter Date:	2012-06-12T00:00:00.000Z	Location:	8	Weight:	68	Height:		Temperature:	37.2	Oxygen_Sat:	99	Systolic_BP:	80	Diastolic_BP:	60	Pulse:	107
Encounter ID:	4060241	Encounter Date:	2012-06-05T00:00:00.000Z	Location:	8	Weight:		Height:		Temperature:	36.6	Oxygen_Sat:	98	Systolic_BP:		Diastolic_BP:		Pulse:	
Encounter ID:	3790202	Encounter Date:	2012-06-05T00:00:00.000Z	Location:	8	Weight:	37	Height:		Temperature:	35	Oxygen_Sat:	100	Systolic_BP:	80	Diastolic_BP:	60	Pulse:	106

Visits

Encounter Date:	Location:	Location-8	Form:	AMPATH Adult Initial Visit Form 5.5 with Mother-Baby Link	Encounter Type:	ADULTINITIAL
Encounter Date:	Location:	Location-8	Form:	Lab - Chemistry (COBAS)	Encounter Type:	LABCHEMISTRY
Encounter Date:	Location:	Location-8	Form:	Lab - Chemistry (COBAS)	Encounter Type:	LABCHEMISTRY
Encounter Date:	Location:	Location-8	Form:	Lab - Serology	Encounter Type:	LABSEROLOGY
Encounter Date:	Location:	Location-8	Form:	AMPATH Adult Return Visit Form 5.5 with Mother-Baby Links	Encounter Type:	ADULTRETURN
Encounter Date:	Location:	Location-8	Form:	AMPATH Adult Return Visit Form 5.5 with Mother-Baby Links	Encounter Type:	ADULTRETURN

Overall Project

Final Thoughts

Overall Project: Final Thoughts

- Learned a lot about Angular & AMPATH app.
- Working in teams was a great way to learn.
- Overall project felt like a real world experience.
- Online Tracker refactoring seems like a success.
- Offline Data Capture implementation needs work.

Overall Project: Final Thoughts

We wish we were able to complete the following, but ran out of time.

- Having tabs for the offline GUI similar to our mockup.
- Complete implementation of our offline GUI (editing, submission, etc)
- Synchronization between offline and online data.
- Complete testing of our implementations.
- Incorporate all teams' implementations as a whole.

We hope our research will help future AMPATH developers complete such tasks!