

CDC Population Health Informatics Framework: Population Health Management Data for Public Health Use

Team Purple Dragons

Godswill Oletu (goletu3), Lucas Vollherbst (lvollherbst3), Pingying Zeng (pzeng7)

Deliverable 4: Final Progress Presentation

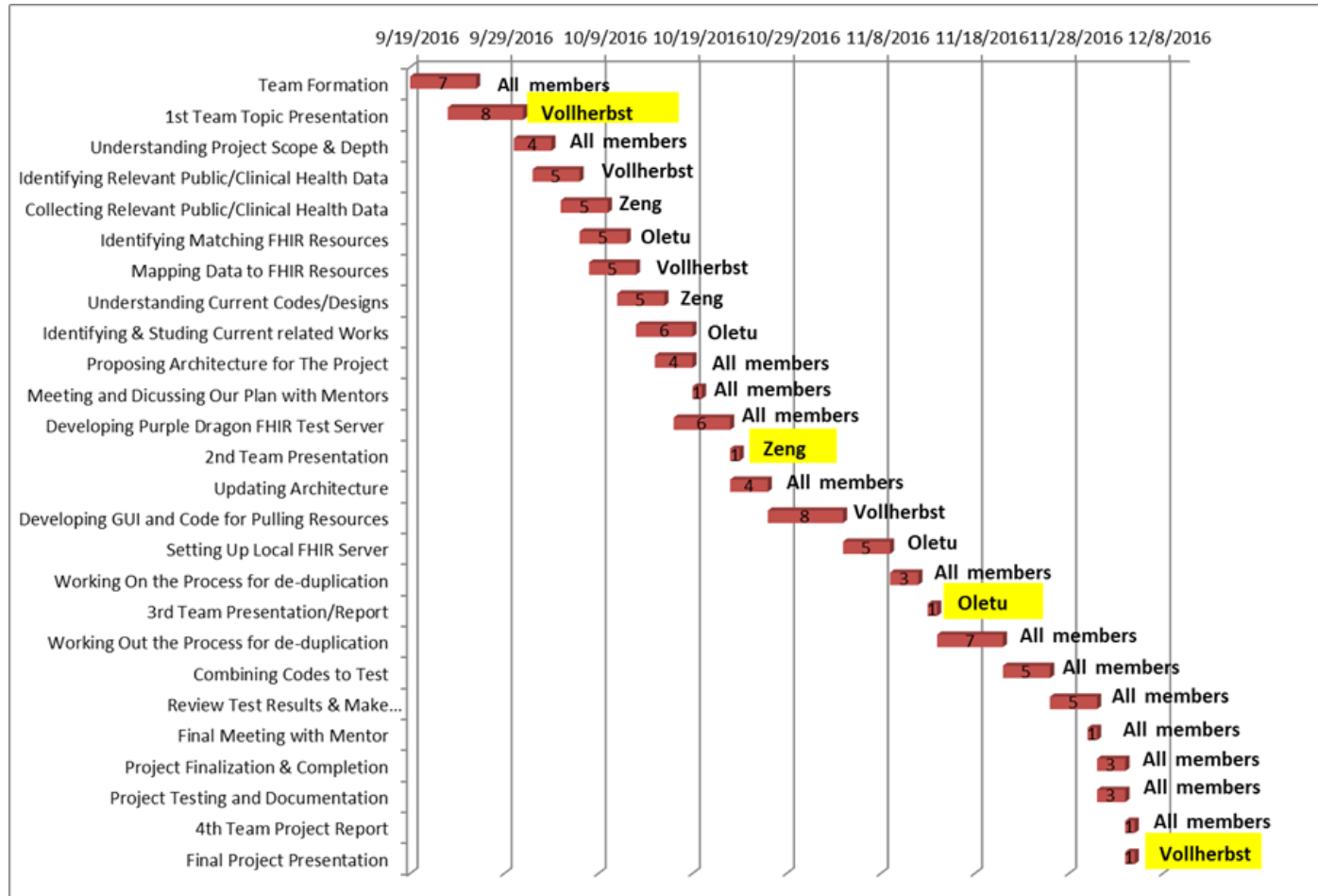
CS 6440: Into to Health Informatics

Fall 2016

Link to presentation video: <https://vimeo.com/194290583>



Complete Update of Gantt Chart



Topic Background

- GOAL: Advance chronic disease management and prevention using EHRs and other health information technology.
- PROBLEM: Applications and services must be used in multiple settings but are limited by the inconsistent data organization, storage formats, and exchange protocols.
- GOAL: Public health agencies would like to use this Population Health Management data to compose a cross-jurisdictional picture of the patient population.
- PROBLEM: Current Population Health Management softwares do not generally collect data by jurisdiction.
- **OBJECTIVE: Build FHIR services that provide data to public health agencies in an identity preserving fashion and present a jurisdictional picture of the patient population.**



Research

Background on population health management and public health agencies and their goals:

Help to identify needs and potential uses of a solution

Chronic disease prevention and management: <http://www.ncsl.org/documents/health/chronicdtk13.pdf>

Public health: https://en.wikipedia.org/wiki/Public_health

Current population health management software:

Gain an understanding of the current situation

Identify strengths to play to and shortcomings to improve upon

Cardigm: <https://www.caradigm.com/en-us/>

Wellcentive: <https://www.wellcentive.com/>

Medecision: <https://www.medecision.com/>

FHIR Background

Better understand FHIR, its resources, RESTful services, and interfacing with them.

FHIR Documentation: <https://www.hl7.org/fhir/documentation.html>

FHIR Implementation: <https://www.hl7.org/fhir/implementation.html>

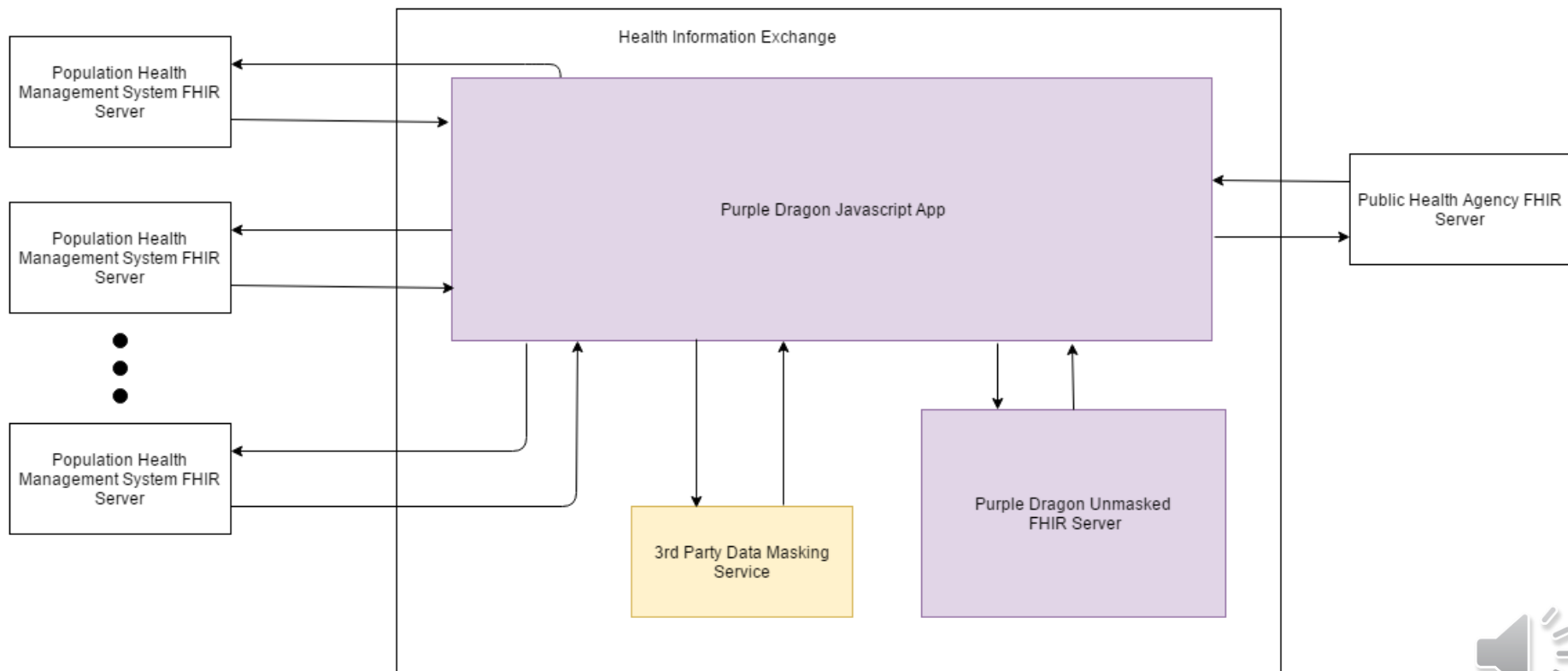


Solution: PurpleDragon app

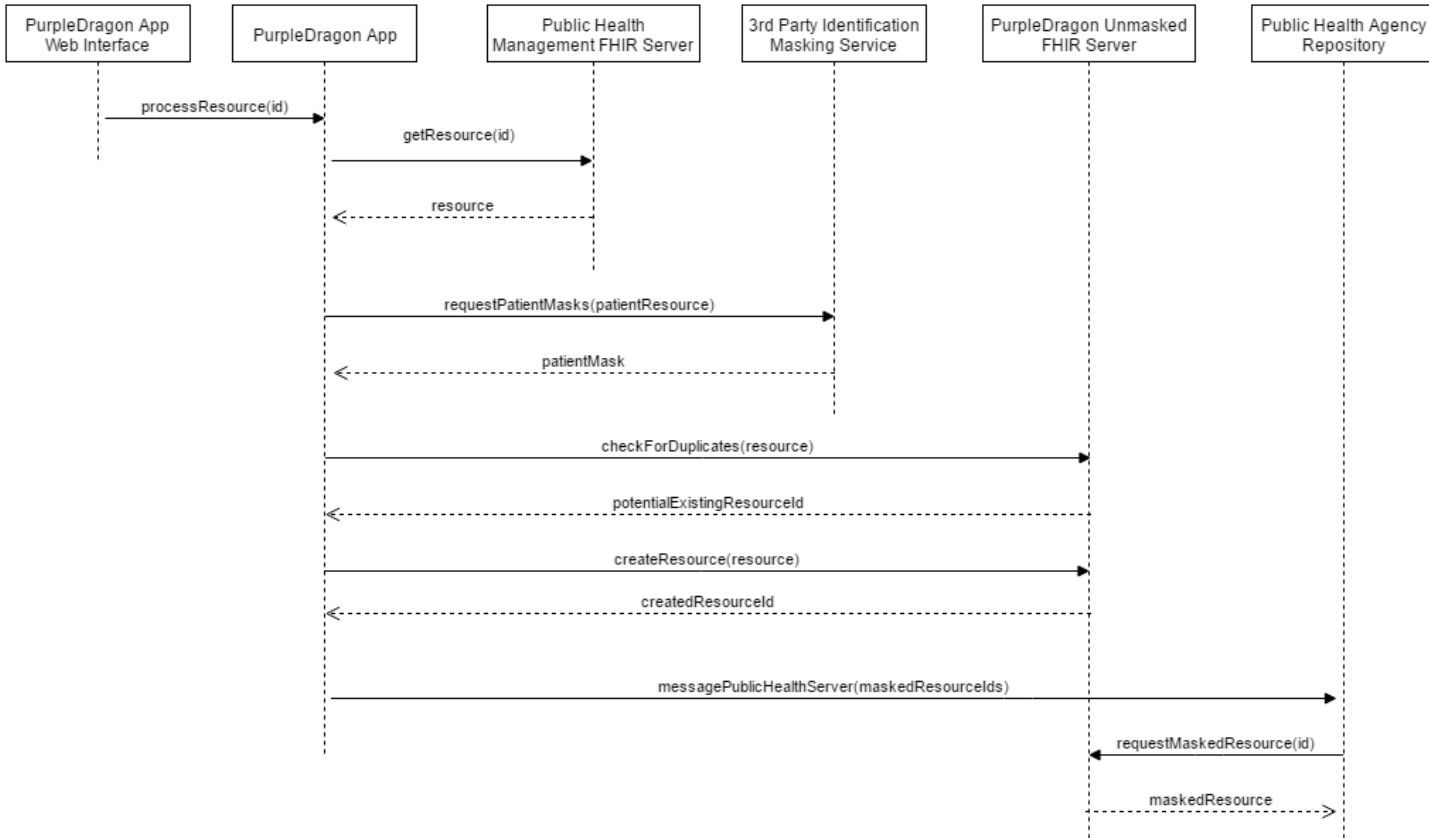
- Javascript web application
- Acts as a centralized hub between many population health management systems and a public health agency
- Intended to be deployed in a centralized location, like an HIE
- Provides services to prevent duplication of data between systems
 - e.g. a Patient that goes to two doctors in the same jurisdiction
- Interfaces with a masking service to properly secure Patient privacy
- Maintains references between Patients and their masked resources to provide potential re-linking



Solution Architecture



Solution Sequence Diagram



Solution Environment: Purple Dragon App

Purple Dragons APP:

- Publicly hosted on Godaddy Linux server with default drivers, libraries, etc.
- Registered Domain Name: www.purpledragonapp.com
- The core app, purpledragon.js
 - Provides a Javascript object to be used to instantiate a connection with each population health management system.
 - Handles processing of resources
 - Dependent on jQuery - www.jquery.com
- exampleinterface.js and index.html
 - Create the web interface specifically intended for use with the modeled system
 - Makes use of Twitter Bootstrap styling - <http://getbootstrap.com/>



Demonstration Environment Setup

All Five Hapi FHIR Servers:

- Publicly hosted with Amazon EC2 cloud services.
- Ubuntu 16.04 (64 bits) micro cloud server.
- Java 1.8, Maven 3.3.9, Snapshot 2.0 (Derby installed by Maven).
- Customized Hapi FHIR Server installation with hibernated Derby database.
- Model the following servers:
 - 3 population health management FHIR servers
 - 1 PurpleDragon FHIR server used to store unmasked data for de-duplication and relinking
 - 1 public health agency FHIR server



Purple Dragon

Purple Dragon is a Javascript app built on using the FHIR standards framework. The app provides three main features:

- **Resource aggregation:** The app aims to provide easy aggregation of FHIR resources for a number of clients in a geographical region and provide these resources to a centralized public health agency for analysis.
- **De-duplication:** Purple Dragon aims to prevent duplication of patient and other resources that may exist in multiple client databases within the region to provide an accurate representation of the region to the public health agency.
- **Anonymization:** The app also interfaces with patient masking services to provide anonymization of patient resources to protect patient confidentiality when data is passed to organizations outside of the patients' care facilities.

Instructions: First, choose the client you wish to retrieve resources from. Then for each available resource (Observation, Condition, Medication, and MedicationOrder), enter a comma separated list of resource IDs to send to the public health agency (e.g. '1,2,3'). Click 'Process resources' to initiate the process.

Choose a client to work with:

client1

Observation resources to pull:

Condition resources to pull:

Medication resources to pull:

MedicationOrder resources to pull:

Process Resources

Activity:

*** READY ***



Deployment - Purple Dragons' FHIR APP

- Include “purpledragon.js” script in a web environment
- Gives access to the PurpleDragon object
- Contains needed methods for processing available resources
- Create an instance of the object with the following parameters:
 - baseUrl - base url of the FHIR server that will be used to store unmasked aggregate data, used to perform de-duplication and relinking
 - clientUrl - base url of the population health management FHIR server that is currently accessing the interface
 - agencyUrl - base url of the public health agency FHIR server that will store the masked FHIR resources that have been processed through the system
- Then call the processResource(id) methods using the created PurpleDragon instance.
- Example to process Observation/4035:

```
var pd = new PurpleDragon(baseUrl, clientUrl, agencyUrl);  
pd.processObservation("4035");
```



Deployment - Purple Dragons' FHIR Servers

- Install Ubuntu 16.04 (64 bit) with at least 80G Storage & 2G Memory.
- Clone git Repository.
 - Git clone <https://github.gatech.edu/gt-hit-fall2016/Population-Health-Management-Data>
- Install at least java 1.8.
 - `sudo apt-get install openjdk-8*`
- Install Maven 3.3.9.
 - `sudo apt-get install maven`
- Change Directory to `/hapi-fhir-jpaserver-example/`.
 - `cd ~/fhir-agency-server/hapi-fhir-jpaserver-example/`
- Compile the FHIR server binaries.
 - `mvn install`
- Launch FHIR Server
 - `Screen`
 - `mvn jetty:run`
- Repeat the steps above to install the other Management/MASKING FHIR servers.



Outstanding elements

- More resources
 - Currently limited to:
 - Observation
 - Condition
 - Medication
 - MedicationOrder
 - Use current resource processes to help guide further additions
- Authentication/login
 - Helps to ensure clients can only offer their own resources
- Patient masking service integration



Thank you for watching.

