Architecture development

Architecture Development Part 1: Virtual Machine Configuration/Testing

As a part of this project, I have created Virtual Machines for various hospitals. VM's are best way to test various types of network architectures and developments in safe environment.
Firstly I have selected 5 hospitals Aspirus, Portage health,BCMH,MGH,UPHIE.
For these hospitals I have prepared VM's with my IP address and secured them.

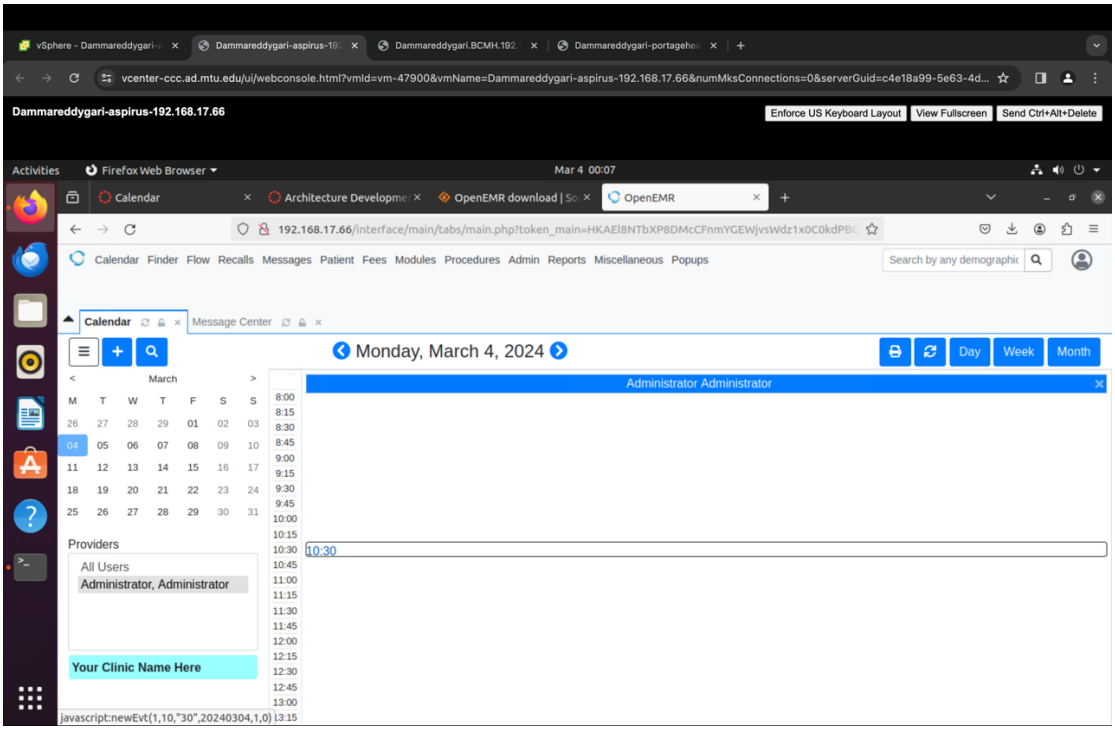
Hospital | HIE	OS Compatible with HAPI-FHIR?	OS Compatible with OpenEHR?	IP Addresss	Successfully Pinged the Other 4 VMs? Yes or No
Asprirus	yes	yes	192.168.17.66	yes
Portage health	yes	yes	192.168.17.67	yes
BCMH	yes	yes	192.168.17.68	yes
MGH	yes	yes	192.168.17.69	yes
UPHIE	yes	yes	192.168.17.70	yes

Architecture Development Part 2: Installation, Configuration, and Security of OpenEMR

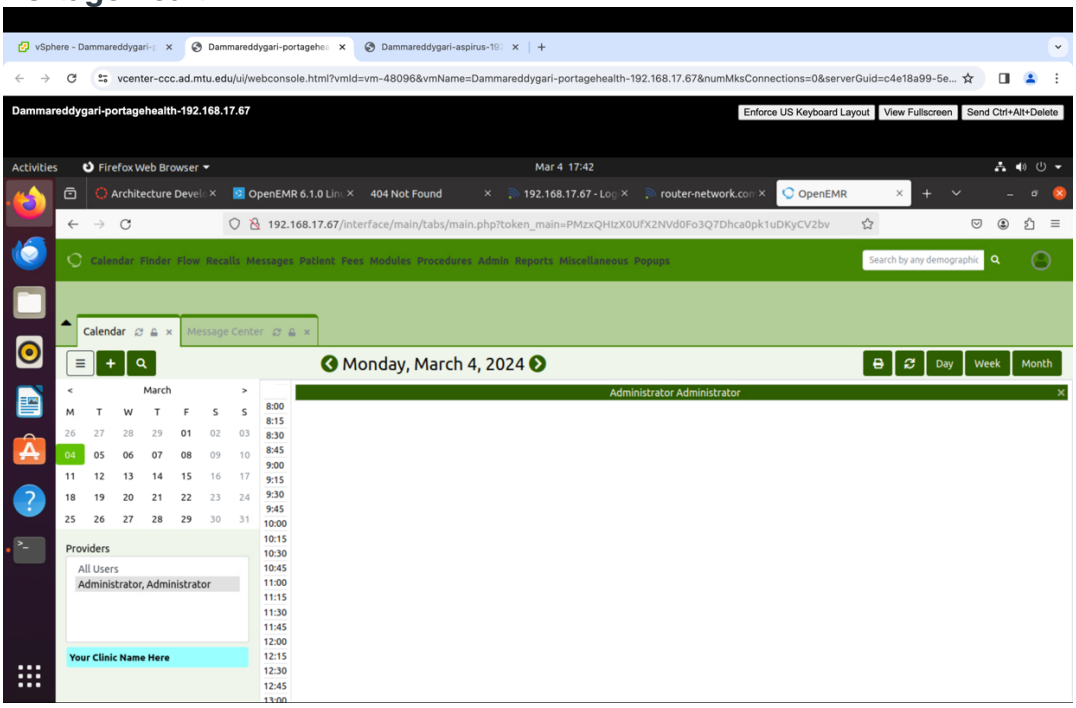
Next I have installed and configured EHR as OpenEMR in each virtual machine
OpenEMR is widely use and open source EHR and medical practice management software system designed to streamline and digitize the workflows of healthcare providers.
OpenEMR offers cost-effective solutions for small to medium practices and low-resource settings due to its open-source licensing. Its customizability is a significant advantage, accommodating various healthcare settings and needs.

Web page screenshot of the hospital successful installation of OpenEMR.

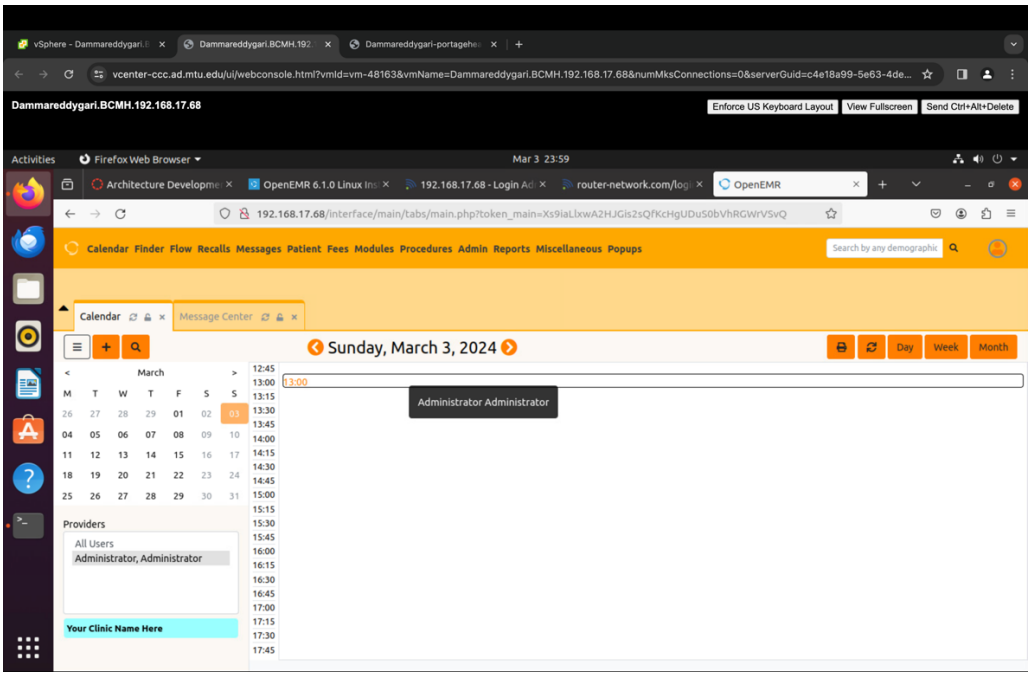
Aspirus



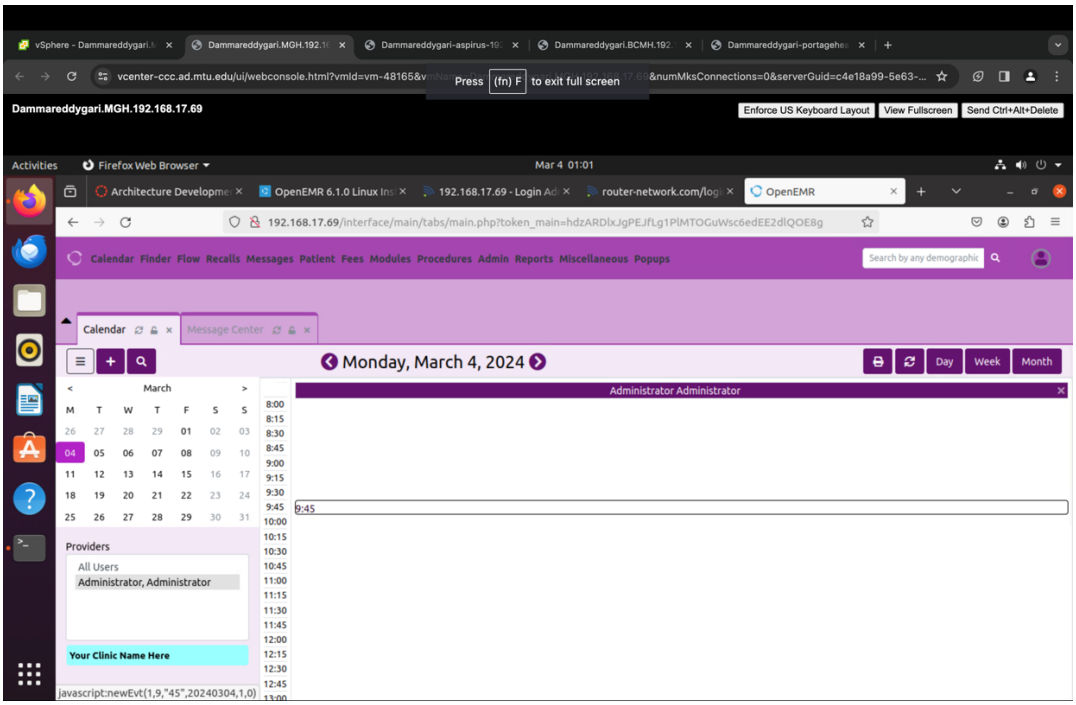
Portage health



BCMH


Dammareddygari.BCMH.192.168.17.68
Enforce US Keyboard Layout
View Fullscreen
Send Ctrl+Alt+Delete
Activities
Firefox Web Browser
Mar 3 23:59
OpenEMR
192.168.17.68/interface/main/tabs/main.php?token_main=Xs9iaLlxwA2HJGis2sQfKcHgUDuS0bVhRGWrVSvQ
Calendar Finder Flow Recalls Messages Patient Fees Modules Procedures Admin Reports Miscellaneous Popups
Calendar
Message Center
Sunday, March 3, 2024
Day
Week
Month
Administrator Administrator
Providers
All Users
Administrator, Administrator
Your Clinic Name Here

MGH

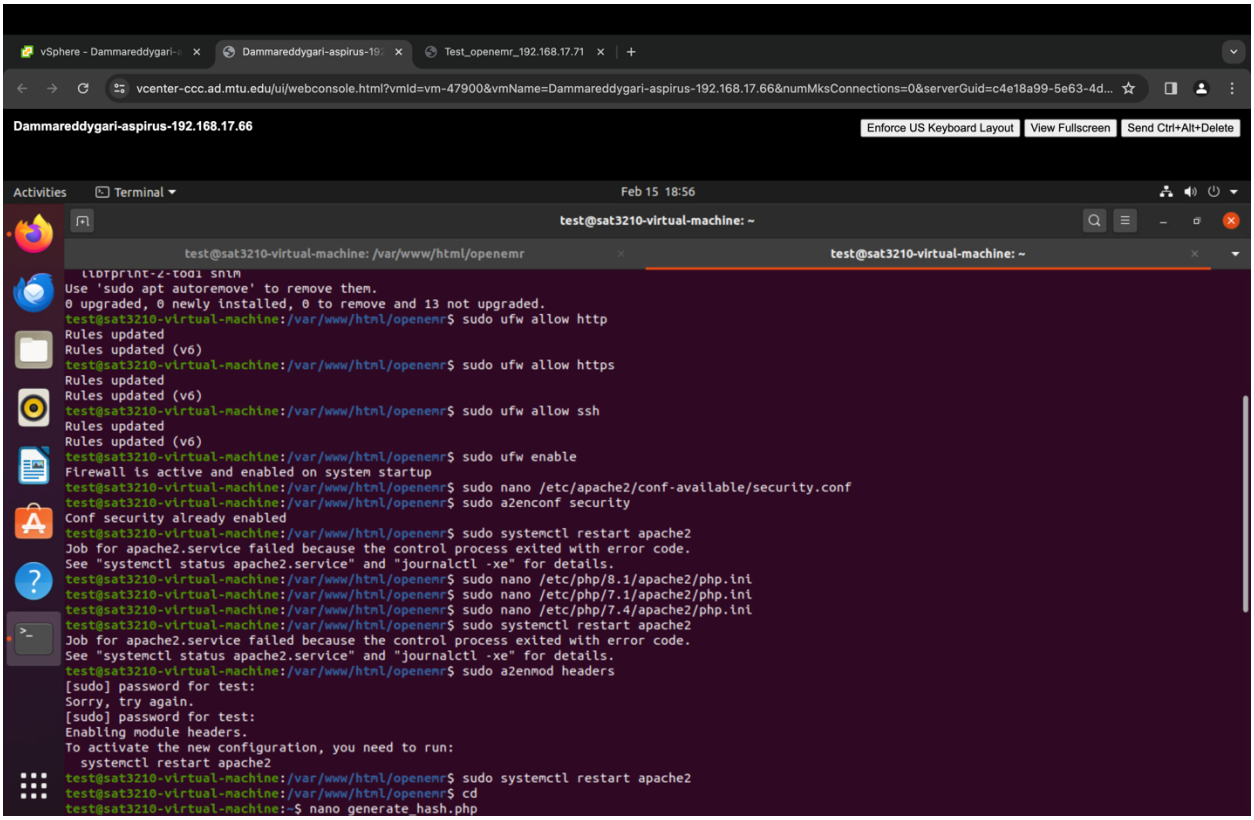

Dammareddygari.MGH.192.168.17.69
Press (fn) F to exit full screen
Enforce US Keyboard Layout
View Fullscreen
Send Ctrl+Alt+Delete
Activities
Firefox Web Browser
Mar 4 01:01
OpenEMR
192.168.17.69/interface/main/tabs/main.php?token_main=hdzARDlxJgPEJfLg1PlMTOGuWsc6edEE2dlQOE8g
Calendar Finder Flow Recalls Messages Patient Fees Modules Procedures Admin Reports Miscellaneous Popups
Calendar
Message Center
Monday, March 4, 2024
Day
Week
Month
Administrator Administrator
Providers
All Users
Administrator, Administrator
Your Clinic Name Here

Steps Followed in Securing OpenEMR on Virtual Machine using Ubuntu Server

1. **Updated and upgraded Ubuntu Server:**
a.Opened the terminal in my Ubuntu Server virtual machine and Executed the following commands to refresh the package list and upgrade the installed packages:
sudo apt-get update
sudo apt-get upgrade2. **Enabled automatic security updates**:
a. Installed the 'unattended-upgrades' package by executing:
sudo apt-get install unattended-upgrades
b. Enabled automatic updates by executing:
sudo dpkg-reconfigure --priority=low unattended-upgrades
3. **Configured a firewall**:
a.Installed the 'ufw' (Uncomplicated Firewall) package by running
sudo apt-get install ufw
b. Allowed HTTP, HTTPS, and SSH traffic by executing below code
sudo ufw allow http
sudo ufw allow https
sudo ufw allow ssh
c.Enabled the firewall by executing
sudo ufw enable
4. **Secured Apache:**
a. Edited the Apache security configuration file
sudo nano /etc/apache2/conf-available/security.conf
b.Modified the following lines to increase security
ServerTokens Prod
ServerSignature Off
TraceEnable Off
Header set X-Content-Type-Options: "nosniff"
Header set X-Frame-Options: "sameorigin"
Header set X-XSS-Protection: "1; mode=block"
Header set X-Robots-Tag: "none"
Header set X-Download-Options: "noopen"
Header set X-Permitted-Cross-Domain-Policies: "none"
c. Saved and exited the file by pressing Ctrl+X, followed by Y, and then Enter.d.Enabled the new security headers by executing
sudo a2enconf headers
e.Restarted Apache using this command by executing
sudo systemctl restart apache2
5. **Secured PHP:**
a. Edited the PHP configuration file using this command:
sudo nano /etc/php/7.4/apache2/php.ini
b.Modified the following lines to increase the security:

expose_php = Off
display_errors = Off
c.Saved and exited the file by pressing Ctrl+X, followed by Y, and then Enter.
d.Used command
Sudo a2enmod headers
e.Restored Apache by executing
Sudo systemctl restart apache2

Completed the steps to secure OpenEMR.



Architecture Development Part 3: Generation of Synthea Patient and Syndromic Surveillance Data for Hospital EHRs to Simulate Disease Outbreak

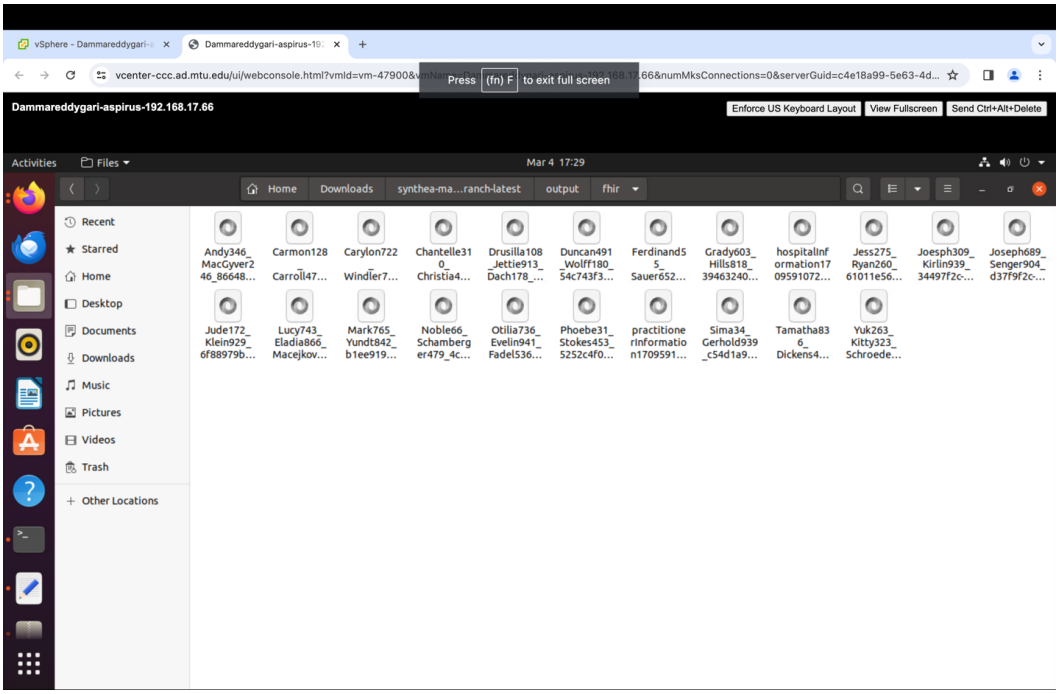
As a Next part in Architecture development I have generated synthetic patient and syndromic surveillance data for hospital EHRs.
Synthea is an open-source software generating realistic patient data for healthcare simulation, research, and testing without compromising patient privacy. It offers diverse datasets for analysis, software development, and training, improving system accuracy. However, it may not capture all real-world complexities and variations, limiting its applicability in some cases. Yet, for disease outbreak surveillance, Synthea's synthetic data is invaluable for simulating and analyzing outbreak scenarios, enhancing public health preparedness and response capabilities.

I have Generated Synthea fornCovid-19 sydromic surveillance disease outbreak HL7 FHIR .json files with the following percentages of each respective hospital population shown below.

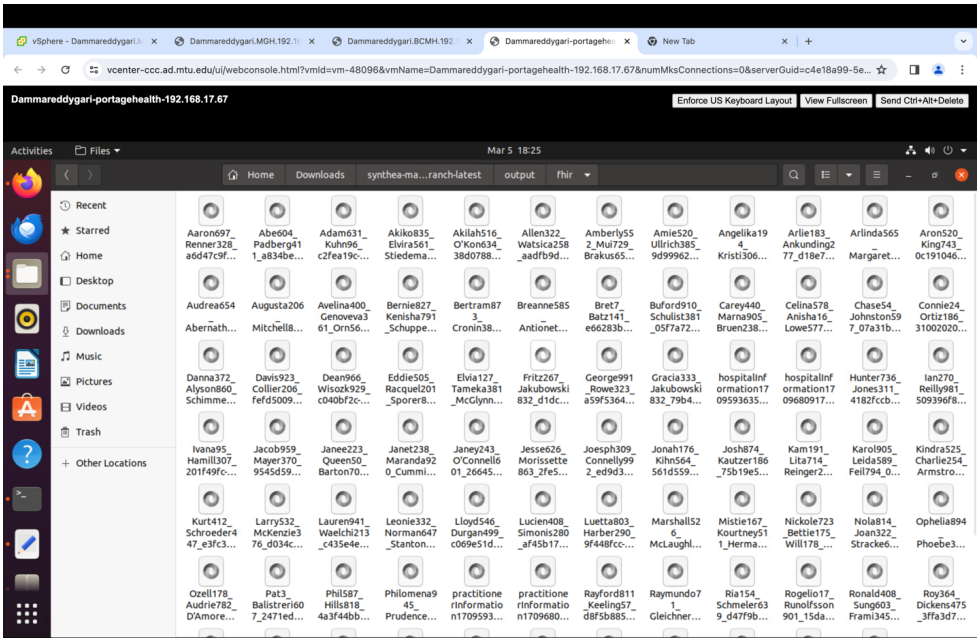
Name	Aspirus Hospital	Portage Health Hospital	BCMH Hospital	MGH Hospital
Generation of Covid-19 HL7 Messages	Low Outbreak (less than 1% of the 5k Population)	Low Outbreak (less than 1% of the 9k Population)	Medium Outbreak (between 1% and 4% of the 7k Population)	High Outbreak (greater than 5% of the 20k Population)

As a result I got json files for patient data I have selected for each hospital

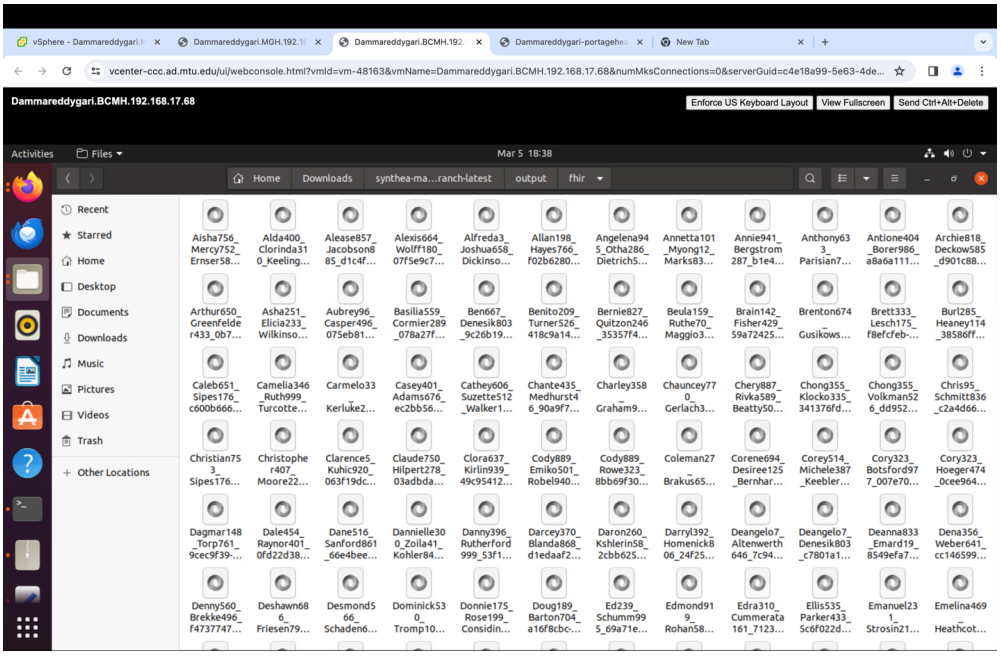
ASPIRUS



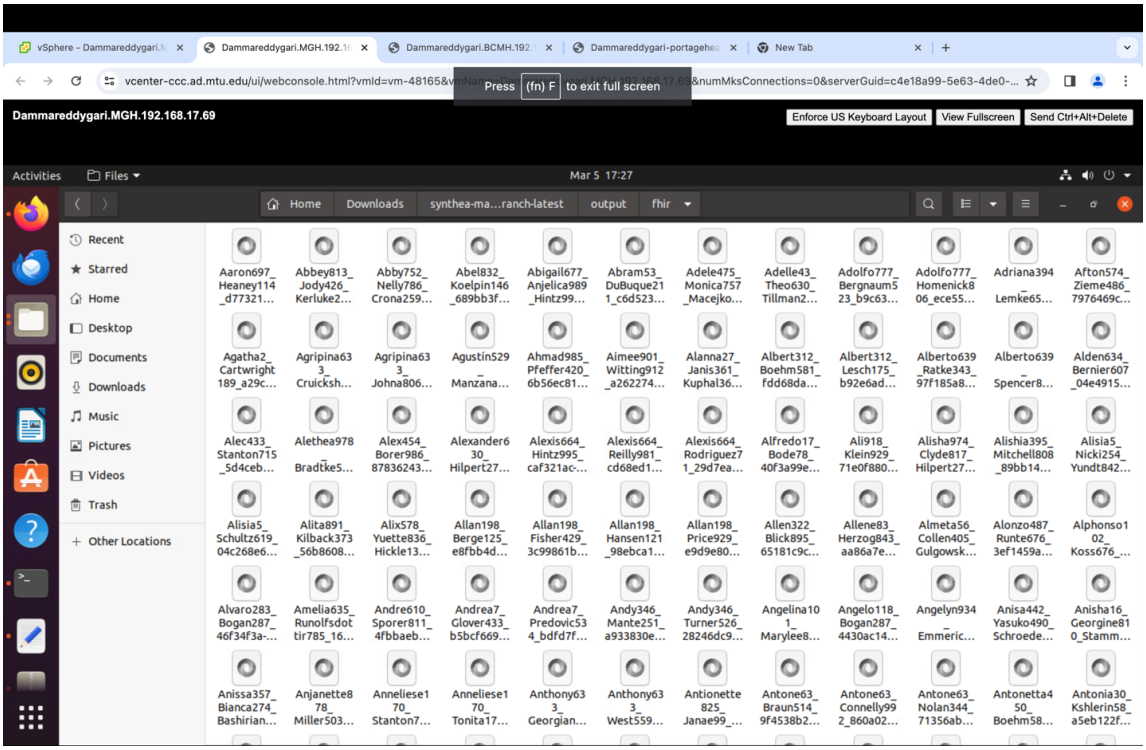
Portage health



BCMH



MGH



HOSPITAL	%USED OF TOTAL SURVED HOSPITAL POPULATION	AMOUNT OF COVID PATIENTS CREATED
ASPIRUS	0.4	20
PORTAGE HEALTH	0.5	45
BCMH	3	210
MGH	6	1200

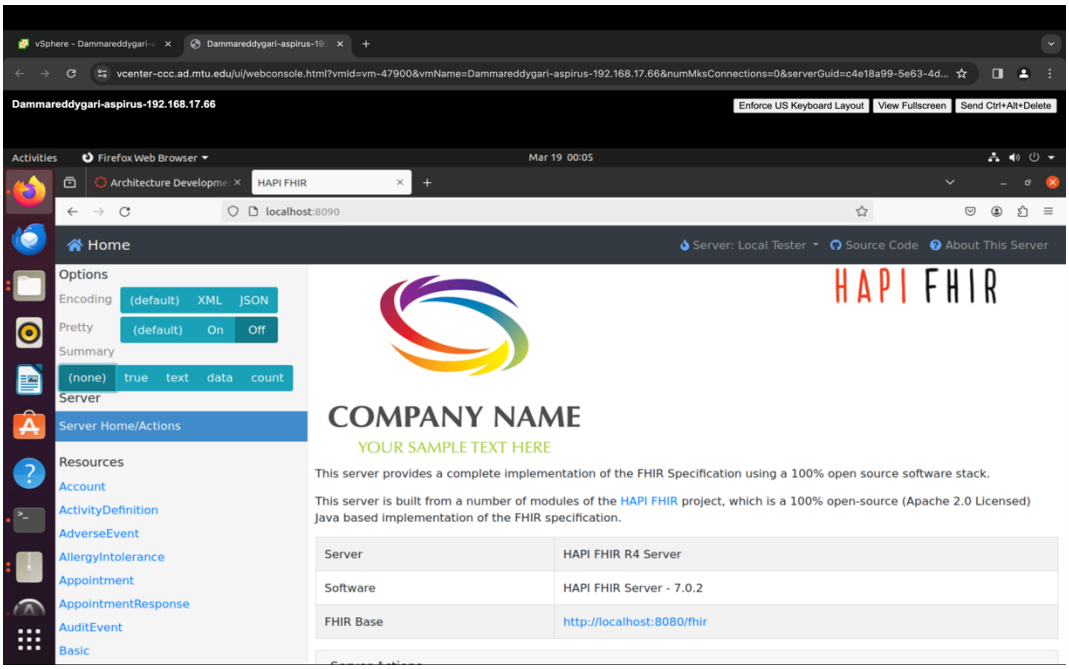
Architecture Development Part 4: Installation and Configuration of Hapi-FHIR Server

Hapi-FHIR is an open-source implementation of the FHIR standard, enabling seamless data exchange in healthcare. It supports interoperability, customization, and collaboration but requires investment in training and infrastructure. Despite challenges, its use in public health surveillance enhances early disease detection and response through real-time data sharing and standardized information exchange, leading to evidence-based policies and improved public health outcomes.

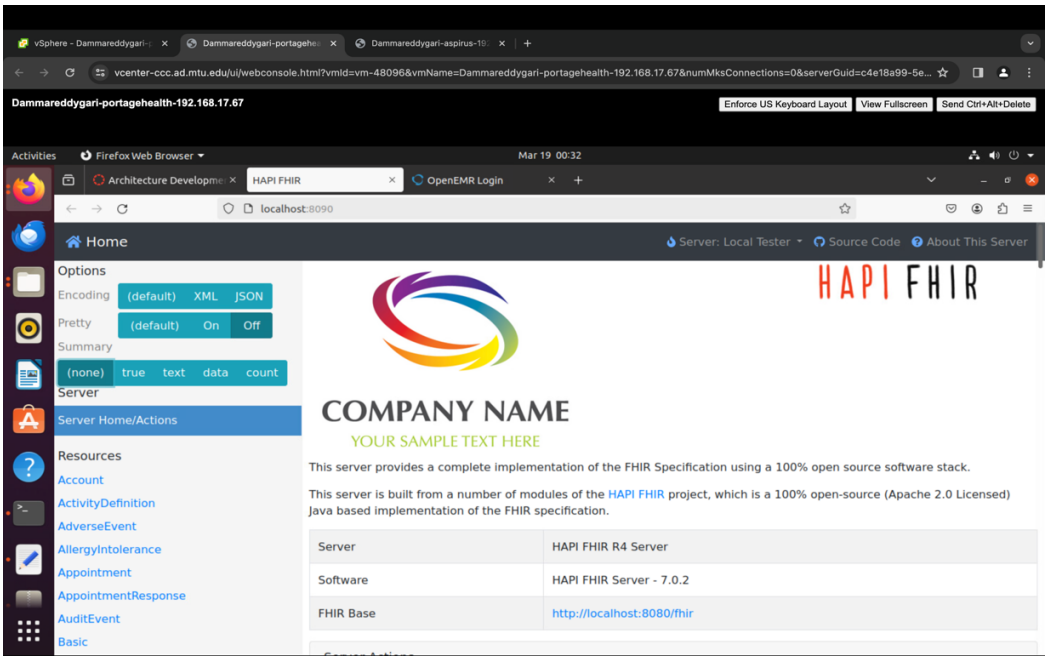
Installed and configured a HAPI FHIR server on virtual machine.

Screenshot of the HAPI-FHIR server's default web UI

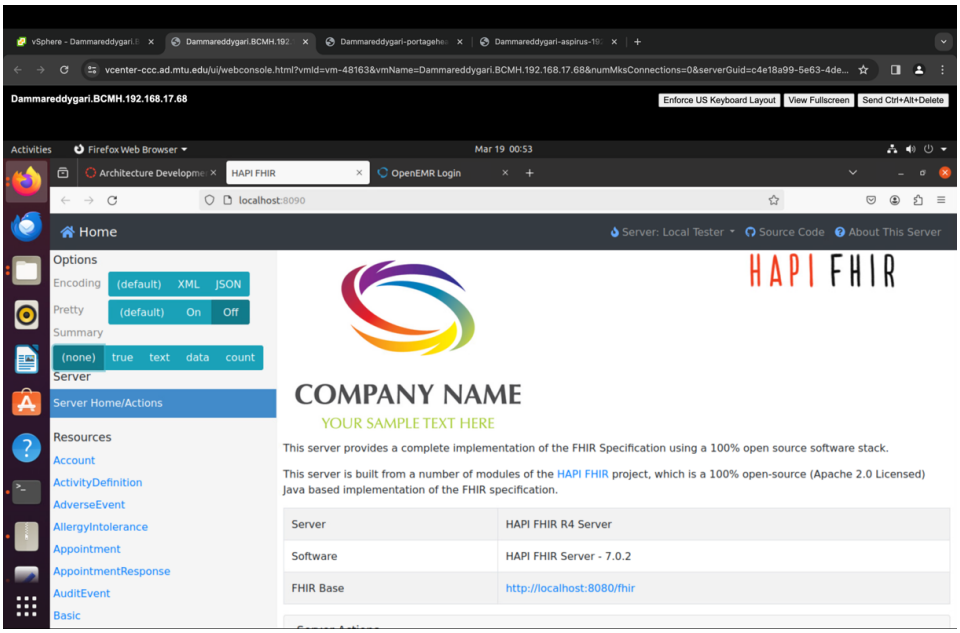
ASPIRUS



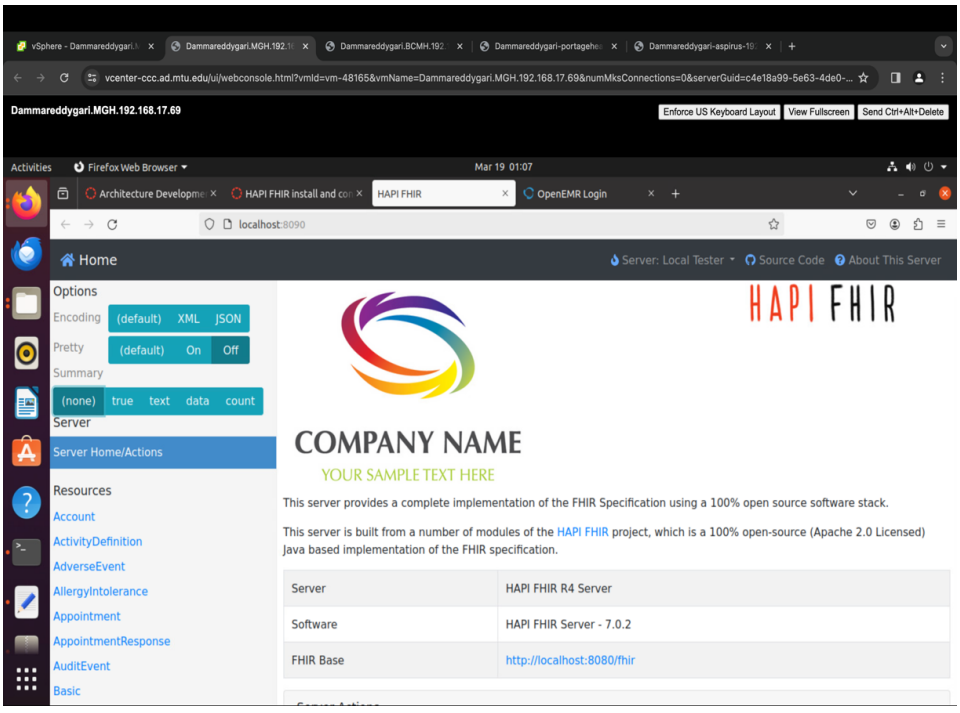
PORTAGE HEALTH



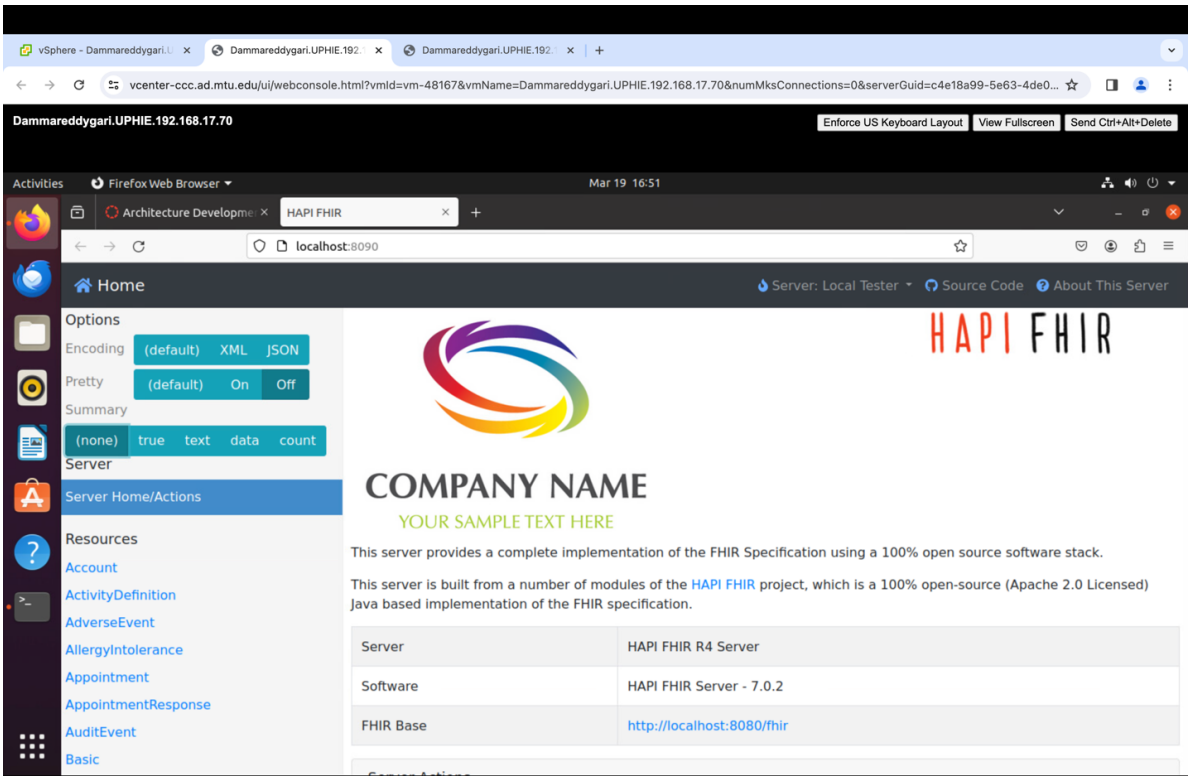
BCMH


Dammareddygari.BCMH.192.168.17.68
Enforce US Keyboard Layout
View Fullscreen
Send Ctrl+Alt+Delete
Activities
Firefox Web Browser
Mar 19 00:53
HAPI FHIR
OpenEMR Login
localhost:8090
Home
Server: Local Tester
Source Code
About This Server
Options
Encoding
(default) XML JSON
Pretty
(default) On Off
Summary
(none) true text data count
Server
Server Home/Actions
Resources
Account
ActivityDefinition
AdverseEvent
AllergyIntolerance
Appointment
AppointmentResponse
AuditEvent
Basic
HAPI FHIR
COMPANY NAME
YOUR SAMPLE TEXT HERE
This server provides a complete implementation of the FHIR Specification using a 100% open source software stack.
This server is built from a number of modules of the HAPI FHIR project, which is a 100% open-source (Apache 2.0 Licensed) Java based implementation of the FHIR specification.
Server
HAPI FHIR R4 Server
Software
HAPI FHIR Server - 7.0.2
FHIR Base
http://localhost:8080/fhir

MGH


Dammareddygari.MGH.192.168.17.69
Enforce US Keyboard Layout
View Fullscreen
Send Ctrl+Alt+Delete
Activities
Firefox Web Browser
Mar 19 01:07
HAPI FHIR install and con
HAPI FHIR
OpenEMR Login
localhost:8090
Home
Server: Local Tester
Source Code
About This Server
Options
Encoding
(default) XML JSON
Pretty
(default) On Off
Summary
(none) true text data count
Server
Server Home/Actions
Resources
Account
ActivityDefinition
AdverseEvent
AllergyIntolerance
Appointment
AppointmentResponse
AuditEvent
Basic
HAPI FHIR
COMPANY NAME
YOUR SAMPLE TEXT HERE
This server provides a complete implementation of the FHIR Specification using a 100% open source software stack.
This server is built from a number of modules of the HAPI FHIR project, which is a 100% open-source (Apache 2.0 Licensed) Java based implementation of the FHIR specification.
Server
HAPI FHIR R4 Server
Software
HAPI FHIR Server - 7.0.2
FHIR Base
http://localhost:8080/fhir

UPHIE



Architecture Development Part 5: Interoperability- FHIR Data Exchange with HAPI-FHIR

Hapi-FHIR's API enables developers to extract specific information from these messages, aiding in disease outbreak analysis. Real-time data collection and analysis empower health authorities to swiftly identify trends, monitor outbreaks, and implement evidence-based interventions, ultimately improving population health outcomes.

\

Successfully explored HAPI FHIR API CLIENT.