
CS 249 Happy Patients App

CS 249, Fall 2017, Final Project
Presented by -
Rashmeet Kaur Khanuja, Anusha Vijay, Steven Yen

Under guidance of -
Prof. Tanuja Phadke

Outline

- **Happy Patients** is an application that allows users to manage and view patient data.
- Users can add, delete, update, retrieve and **filter** patient information in this application.
- The application uses caching (Memcached) to help improve retrieval speed so that not all requests requires a Cassandra query.
- A **Policy Server** maintains cache policies and users can retrieve and update the cache policies based on the need.
- **Message consumers** listens on a message queue (in ActiveMQ) for messages published by the Happy Patients app and runs appropriate routines when messages are received.

Technologies Used

- Jersey
- Cassandra
- ActiveMQ
- Memcached
- Jetty
- Maven
- Python
- Flask
- Javascript
- Angular

What we already knew?

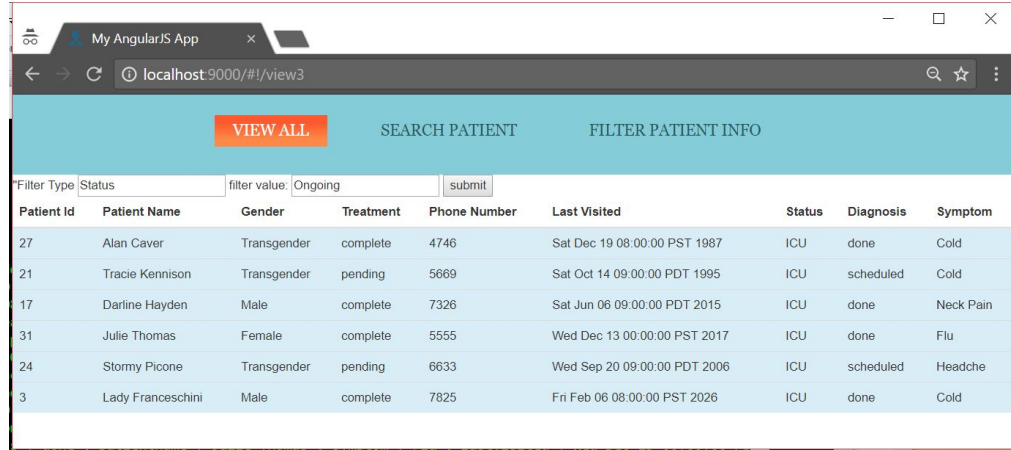
- Python Flask Server
- Jetty Container for Java Projects
- Maven Project and Dependency Management

What we had to learn from scratch?

- Java Jersey Web Services Framework
- Connecting to Cassandra through Java
- Connecting to ActiveMQ through Java, and using JMS API
- Connecting to Memcached

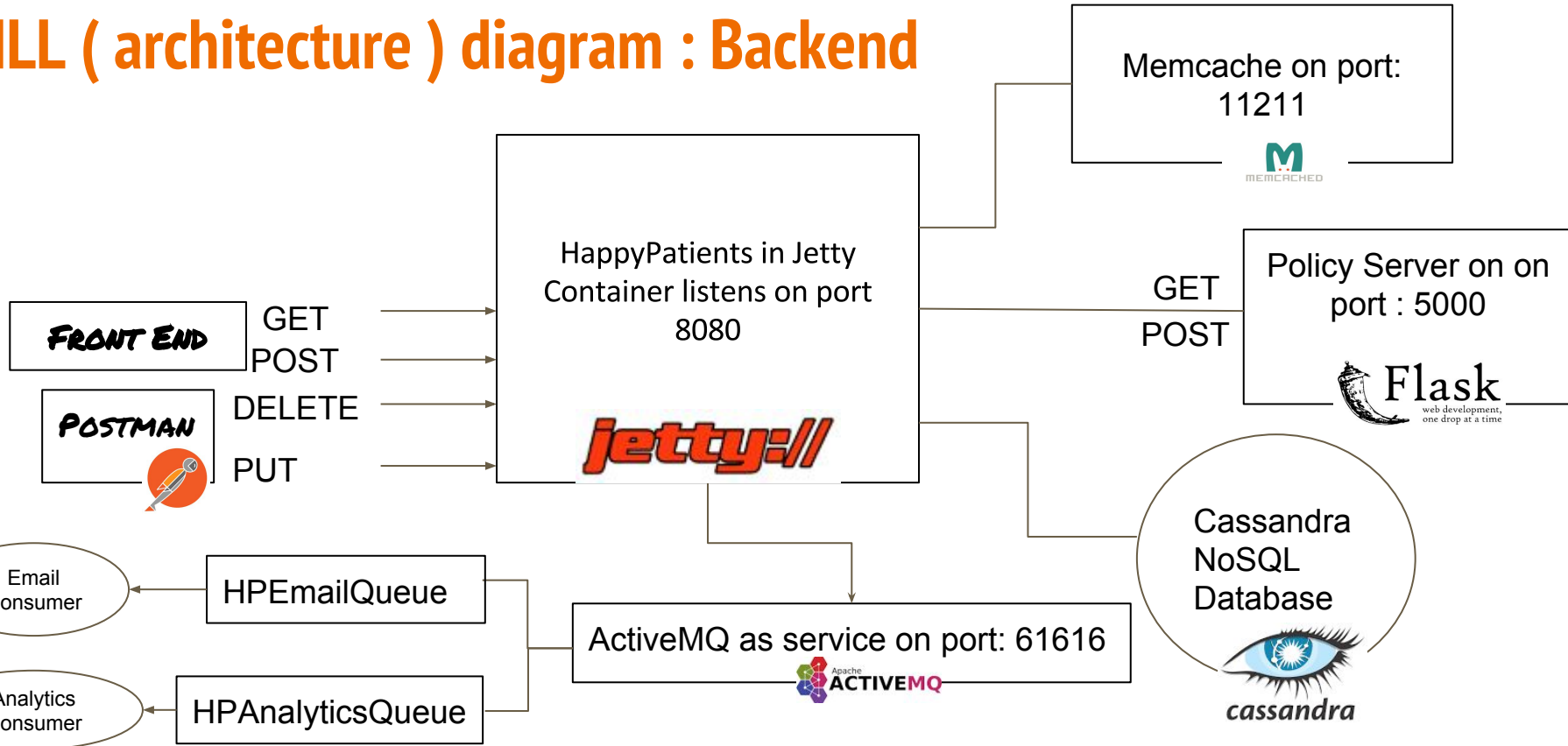
The POC we did

- Generated sample patient data as CSV file and import into Cassandra
- Used POSTMAN to send GET/POST/PUT/DELETE requests to endpoints
- Created mock front-ends that generates HTTP requests and parses the responding JSON object

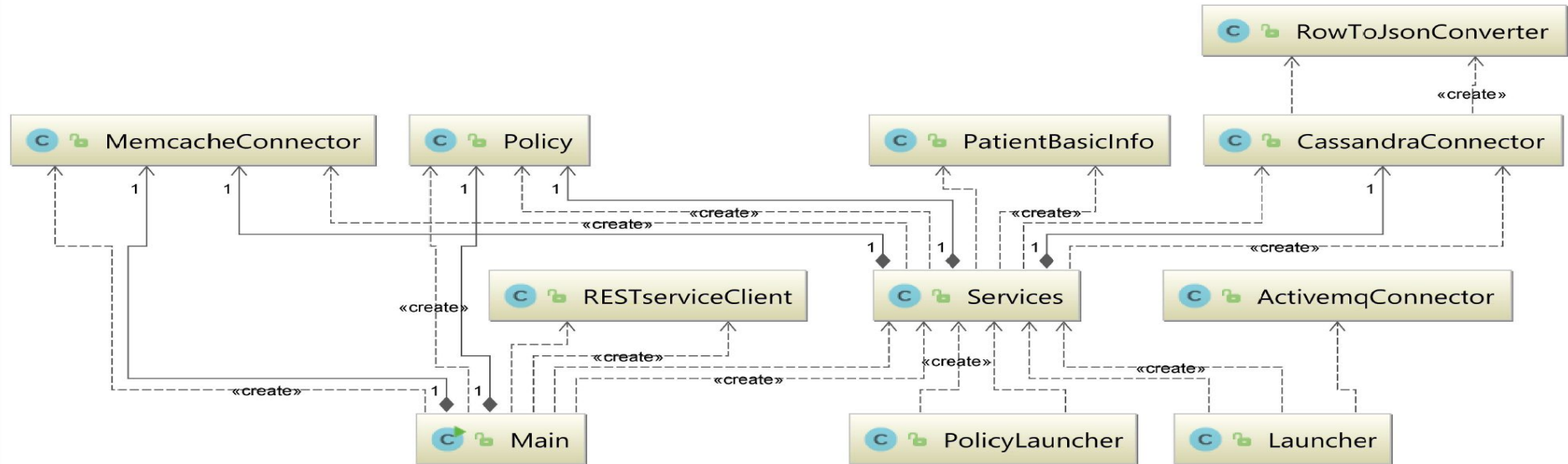


Patient Id	Patient Name	Gender	Treatment	Phone Number	Last Visited	Status	Diagnosis	Symptom
27	Alan Caver	Transgender	complete	4746	Sat Dec 19 08:00:00 PST 1987	ICU	done	Cold
21	Tracie Kennison	Transgender	pending	5669	Sat Oct 14 09:00:00 PDT 1995	ICU	scheduled	Cold
17	Darline Hayden	Male	complete	7326	Sat Jun 06 09:00:00 PDT 2015	ICU	done	Neck Pain
31	Julie Thomas	Female	complete	5555	Wed Dec 13 00:00:00 PST 2017	ICU	done	Flu
24	Stormy Picone	Transgender	pending	6633	Wed Sep 20 09:00:00 PDT 2006	ICU	scheduled	Headche
3	Lady Franceschini	Male	complete	7825	Fri Feb 06 08:00:00 PST 2026	ICU	done	Cold

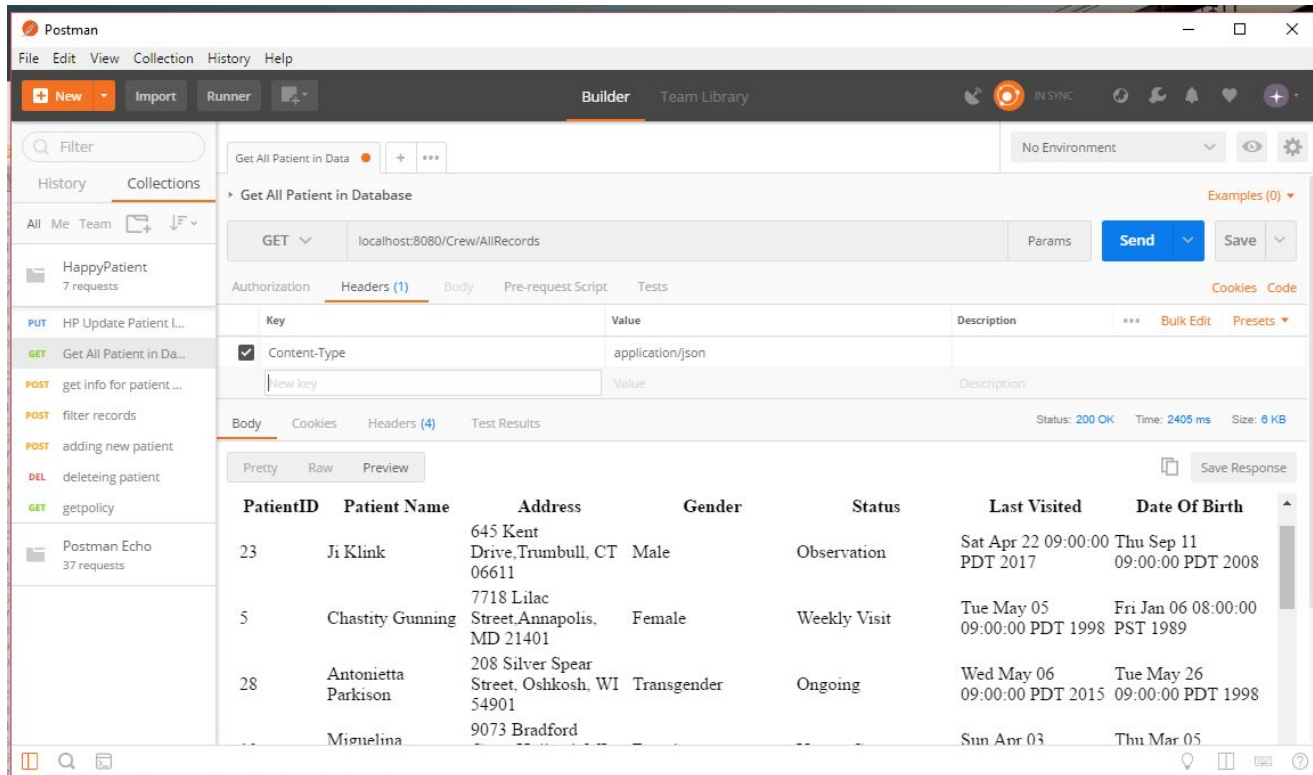
HLL (architecture) diagram : Backend



Class Diagram



GET, Endpoint: /Crew/AllRecords



The screenshot shows the Postman interface with a GET request to `localhost:8080/Crew/AllRecords`. The response is a table of patient records. The table has the following columns: PatientID, Patient Name, Address, Gender, Status, Last Visited, and Date Of Birth. The response status is 200 OK, with a time of 2405 ms and a size of 6 KB.

PatientID	Patient Name	Address	Gender	Status	Last Visited	Date Of Birth
23	Ji Klink	645 Kent Drive, Trumbull, CT 06611	Male	Observation	Sat Apr 22 09:00:00 PDT 2017	Thu Sep 11 09:00:00 PDT 2008
5	Chastity Gunning	7718 Lilac Street, Annapolis, MD 21401	Female	Weekly Visit	Tue May 05 09:00:00 PDT 1998	Fri Jan 06 08:00:00 PST 1989
28	Antonietta Parkison	208 Silver Spear Street, Oshkosh, WI 54901	Transgender	Ongoing	Wed May 06 09:00:00 PDT 2015	Tue May 26 09:00:00 PDT 1998
...	Miguelina	9073 Bradford			Sun Apr 03	Thu Mar 05

Returns a table of information for all patients in the hospital's database

POST, Endpoint: /Crew/PatientRecord

The screenshot shows a REST client interface with the following details:

- Method:** POST
- Endpoint:** localhost:8080/Crew/PatientRecord
- Body Type:** JSON (application/json)
- Request Body:**

```
1 {
2   "Patient Name": "Harry Potter",
3   "Address": "One Washington Square, San Jose, CA 95192",
4   "Gender": "Male",
5   "Created Date": "2017-12-13",
6   "DOB": "1998-05-30",
7   "Phone Number": 5555,
8   "Status": "ICU",
9   "Symptom": "Flu",
10  "Diagnosis": "pending",
11  "Treatment": "pending"
12 }
```
- Status:** 200 OK
- Time:** 2845 ms
- Size:** 105 B
- Response Body:**

```
1 Patient Added! Patient ID is 29
```

Adds a new patient to the hospital's database. And adds the patient's record to cache if it matches the policy.

Male	20	2025-02-03 16:00:00.000000+0000	Dorthea Kintner	7595	Ongoing	Stomach Ache	complete
male	7	2014-07-27 16:00:00.000000+0000	Indira Marcell	5800	Ongoing	Flu	pending
nder	0	2015-04-07 16:00:00.000000+0000	Sheri Rapoza	7343	weekly visit	headche	complete
Male	29	2017-12-13 08:00:00.000000+0000	Harry Potter	5555	ICU	Flu	pending
Male	0	1991-05-25 16:00:00.000000+0000	Lucille Toole	6073	Closed	Cold	in progress
Male	14	2009-12-26 16:00:00.000000+0000	Concetta Tome	7065	Weekly Visit	Stomach Ache	complete
Male	26	2010-03-04 16:00:00.000000+0000	Bong Rosenblatt	4257	Closed	Stomach Ache	in progress
nder	21	1995-10-14 16:00:00.000000+0000	Tracie Kennison	5669	ICU	Cold	pending

PUT, Endpoint: /Crew/PatientRecord




PUT Params

Authorization Headers (1) **Body** Pre-request Script Tests Cookies Code

☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ JSON (application/json)

```
1 {  
2   "Patient ID": 19,  
3   "Gender": "Female",  
4   "Phone Number": 6767,  
5   "Status": "Ongoing"  
6 }
```

Body Cookies Headers (4) Test Results Status: 200 OK Time: 145 ms Size: 875 B

Pretty Raw Preview Text    Save Response

```
1 Updated Info from  
2 [{"treatment": "pending", "address": "662 Sulphur Springs Rd, Fuquay Varina, NC 27526", "gender": "Male", "patientid": 19, "createddate": "Thu Mar 21 08:00:00 PST 1996", "phonenumber": 6049, "diagnosis": "pending", "patientname": "Afton Needham", "symptom": "Flu", "lastvisited": "Sun Jun 07 09:00:00 PDT 2009", "dob": "Sat Sep 17 09:00:00 PDT 1988", "id": 19, "status": "Weekly Visit"}] To  
3 [{"treatment": "pending", "address": "662 Sulphur Springs Rd, Fuquay Varina, NC 27526", "gender": "Female", "patientid": 19, "createddate": "Thu Mar 21 08:00:00 PST 1996", "phonenumber": 6767, "diagnosis": "pending", "patientname": "Afton Needham", "symptom": "Flu", "lastvisited": "Sun Jun 07 09:00:00 PDT 2009", "dob": "Sat Sep 17 09:00:00 PDT 1988", "id": 19, "status": "Ongoing"}]
```

Update a patient's information. This uses the same endpoint as before with a different operation. Adds the patient to Cache if it matches policy after update.

DELETE, Endpoint: /Crew/PatientRecord/{patientID}

The screenshot shows a REST client interface with the following details:

- Request:**
 - Method: DELETE
 - URL: localhost:8080/Crew/PatientRecord/10
 - Params: (empty)
 - Body: (empty, form-data selected)
- Response:**
 - Status: 200 OK
 - Time: 151 ms
 - Size: 157 B
 - Content: Patient Record Deleted!

The interface includes tabs for Authorization, Headers (1), Body, Pre-request Script, and Tests. The Body tab is active, showing a table for key-value pairs. The response is displayed in a text area with a 'Save Response' button.

Removes a patient's entry from the hospital's database (and the cache if it is there).

POST, Endpoint: /Crew/LookupByName

The screenshot displays a REST client interface. The top section shows a POST request to the endpoint `localhost:8080/Crew/LookupByName`. The request body is a JSON object: `{"Patient Name": "Alan Caver"}`. The bottom section shows the response, which is a JSON object with the following fields: `"treatment": "pending", "address": "Nowhere", "gender": "Transgender", "patientid": 27, "createddate": "Mon Jul 12 09:00:00 PDT 1982", "phonenumber": 1234, "diagnosis": "scheduled", "patientname": "Alan Caver", "symptom": "Cold", "lastvisited": "Sat Dec 19 08:00:00 PST 1987", "dob": "Sat Aug 01 09:00:00 PDT 1981", "id": 27, "status": "ICU"`. The status bar indicates a 200 OK response with a time of 2505 ms and a size of 456 B.

POST localhost:8080/Crew/LookupByName

Authorization Headers (1) Body Pre-request Script Tests Cookies Code

form-data x-www-form-urlencoded raw binary JSON (application/json)

```
1 {"Patient Name": "Alan Caver"}
```

Body Cookies Headers (4) Test Results Status: 200 OK Time: 2505 ms Size: 456 B

Pretty Raw Preview JSON

```
1 [
2   {
3     "treatment": "pending",
4     "address": "Nowhere",
5     "gender": "Transgender",
6     "patientid": 27,
7     "createddate": "Mon Jul 12 09:00:00 PDT 1982",
8     "phonenumber": 1234,
9     "diagnosis": "scheduled",
10    "patientname": "Alan Caver",
11    "symptom": "Cold",
12    "lastvisited": "Sat Dec 19 08:00:00 PST 1987",
13    "dob": "Sat Aug 01 09:00:00 PDT 1981",
14    "id": 27,
15    "status": "ICU"
16  }
17 ]
```

Look's up a patient's information given the name. It first checks if the patient's record is in the Cache. Returns from Cache if it is in the Cache. If we get a Cache miss, then the patient's info will be retrieved from Cassandra

POST Endpoint: /Crew/PatientInfoFilter

POST localhost:8080/Crew/PatientInfoFilter Params Send Save Status: 200 OK Time: 184 ms Size: 3.91 KB

Auth Headers (1) Body Pre-req. Tests Cookies Code

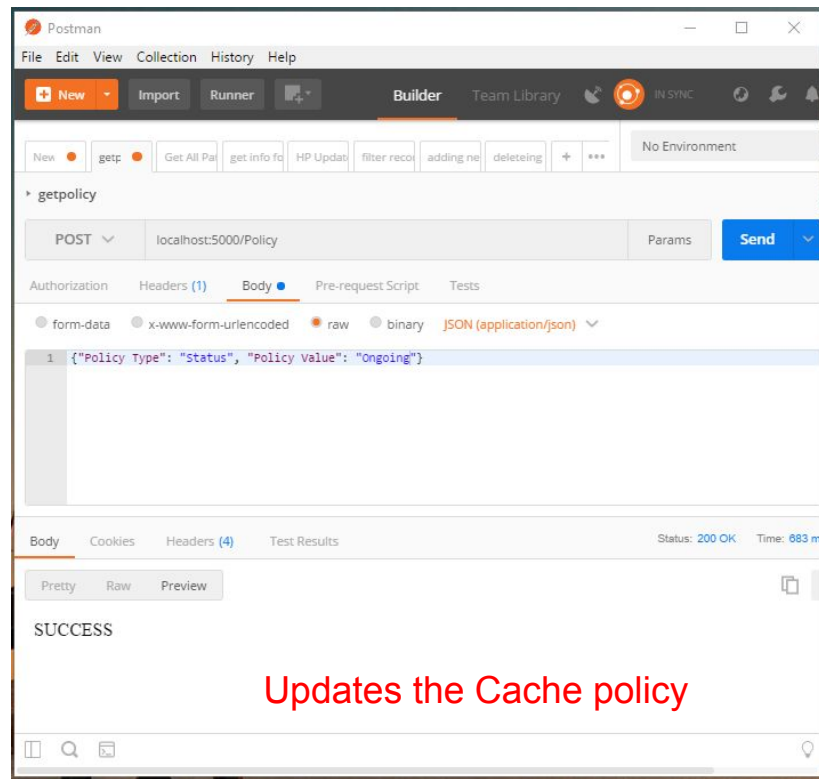
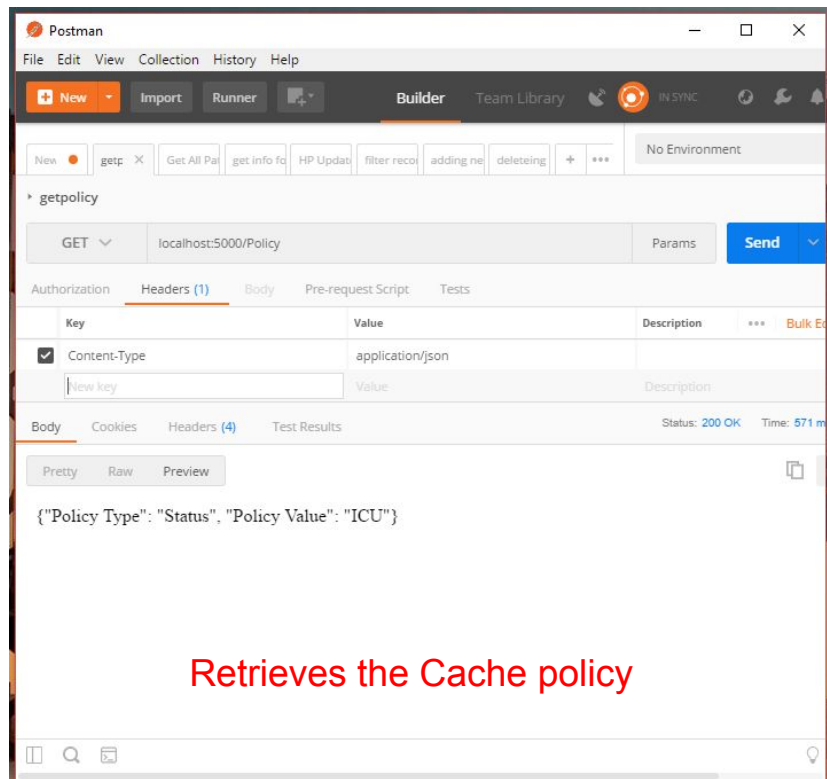
form-data x-www-form-urlencoded raw binary JSON (application/json)

1- {"Filter Type": "Gender",
2 "Filter Value": "Male"}
3

1- [
2 {
3 "treatment": "in progress",
4 "address": "645 Kent Drive, Trumbull, CT 06611",
5 "gender": "Male",
6 "patientid": 23,
7 "createddate": "Sat Mar 12 08:00:00 PST 2011",
8 "phonenumber": 4787,
9 "diagnosis": "done",
10 "patientname": "Ji Klink",
11 "symptom": "Neck Pain",
12 "lastvisited": "Sat Apr 22 09:00:00 PDT 2017",
13 "dob": "Thu Sep 11 09:00:00 PDT 2008",
14 "id": 23,
15 "status": "Observation"
16 },
17 {
18 "treatment": "pending",
19 "address": "8398 Sunset Dr, Tualatin, OR 97062",
20 "gender": "Male",
21 "patientid": 16,
22 "createddate": "Wed Feb 15 08:00:00 PST 2006",
23 "phonenumber": 6003,
24 "diagnosis": "pending",
25 "patientname": "Delinda Shirk",
26 "symptom": "Fever",
27 "lastvisited": "Tue Apr 20 09:00:00 PDT 2021",
28 "dob": "Tue Dec 26 08:00:00 PST 2006",
29 "id": 16,
30 "status": "Ongoing"
31 }
32]

Filters the records based on specified filter type and value. When applicable, the Cache is checked and the data retrieved from Cache. Otherwise, queries Cassandra.

PolicyServer endpoint: “/Policy” GET and POST



Message Consumers

The screenshot displays a web application interface for updating patient information and an IDE showing the corresponding Java code and message logs.

Web Application Interface:

- Buttons: Get All Patk, adding new, filter record, getpolicy, HP Update, +, ...
- Section: HP Update Patient Info
- Method: PUT, URL: localhost:8080/Crew/PatientRecord
- Headers: Headers (1)
- Body: Body (selected), Pre-request Script, Tests
- Form: form-data, x-www-form-urlencoded, raw, binary, JSON (application/json)
- Body Content:

```
1 {
2   "Patient ID": 23,
3   "Address": "2488 Parca Ave, San Jose, 95988",
4   "Phone number": 8009977755
5 }
6 }
```
- Status: 200 OK
- Test Results: Updated Info from
- Log:

```
1 Updated Info from
2 [{"treatment": "in progress", "address": "645 Kent Drive, Trumbull, CT 06611", "gender": "Male", "patientid": "23", "phonenumber": "8009977755", "diagnosis": "done", "patientname": "Ji Klink", "symptoms": "lastvisited": "Sat Apr 22 09:00:00 PDT 2017", "dob": "Thu Sep 11 09:00:00 PDT 2008", "id": "23", "status": "in progress"}]
3 [{"treatment": "in progress", "address": "2488 Parca Ave, San Jose, 95988", "gender": "Male", "patientid": "23", "phonenumber": "8009977755", "diagnosis": "done", "patientname": "Ji Klink", "symptoms": "lastvisited": "Sat Apr 22 09:00:00 PDT 2017", "dob": "Thu Sep 11 09:00:00 PDT 2008", "id": "23", "status": "in progress"}]
```

IDE (HappyPatientMessageConsumers):

- Project: hpMessageConsumers
- Source: src/main/java
- Files: AnalyticsConsumer.java, EmailConsumer.java, Main.java
- Message Dialogs:
 - New Msg in HPAnalyticsQueue: updated personal information Perform Analytics!
 - New Msg in HPEmailQueue: updated personal information Send Email!
- Run Console:

```
Run Main
"C:\Program Files\Java\jdk1.8.0_101\bin\java" ...
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder"
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder
New Msg in EmailQueue: updated personal information
New Msg in AnalyticsQueue: updated personal information
```

- Email consumer observes ActiveMQ Queue "HPEmailQueue"

- Analytics consumer observes ActiveMQ Queue "HPAnalyticsQueue"

- When info is updated, main HP app sends message to queue triggering action by consumers

How we set up the environment?

- We set up the environment on both MAC and Windows
 - Install DataStax Community wrapper for Cassandra
 - Install Apache ActiveMQ as a service running in background
 - Download and run Memcache.exe
 - Do pip install for Flask for Python and its dependencies
 - JSON
 - Requests

Environment: Cassandra

Cassandra CQL Shell

20	7256 North Courtland Drive,Middletown, CT 0645	2015-05-10 16:00:00.000000+0000	2008-11-15 16:00:00.000000+0000	Male	20	2025-02-03 16:00:00.000000+0000	Dorthea Kintner	7595	Ongoing
7	11 Leeton Ridge St,Llnt, MI 48504	2010-03-09 16:00:00.000000+0000	2003-07-20 16:00:00.000000+0000	Female	7	2014-07-27 16:00:00.000000+0000	Indira Marcell	5800	Ongoing
6	56 Glen Eagles Rd,Desoto, TX 75115	1994-01-14 16:00:00.000000+0000	1988-07-24 16:00:00.000000+0000	Transgender	6	2013-04-07 16:00:00.000000+0000	Sheri Rapoza	7343	Weekly Visit
29	82 East 10th Street,Loxahatchee, FL 33470	1987-01-04 16:00:00.000000+0000	1986-02-24 16:00:00.000000+0000	Transgender	29	1999-12-27 16:00:00.000000+0000	null	7043	Weekly Visit
9	84 Fordham St,Lithonia, GA 30038	1988-10-11 16:00:00.000000+0000	1988-08-20 16:00:00.000000+0000	Male	9	1991-05-25 16:00:00.000000+0000	Lucille Toole	6073	Closed
14	79 Miles Ave,Dearbourn Heights, MI 48127	2005-06-08 16:00:00.000000+0000	2005-03-10 16:00:00.000000+0000	Male	14	2009-12-26 16:00:00.000000+0000	Concetta Tome	7065	Weekly Visit
26	350 NE. Gonzales St,Encino, CA 91316	1998-04-26 16:00:00.000000+0000	1998-11-20 16:00:00.000000+0000	Male	26	2010-03-04 16:00:00.000000+0000	Bong Rosenblatt	4257	Closed
21	8963 Peachtree Lane,Little Rock, AR 72209	1988-10-07 16:00:00.000000+0000	1984-01-01 16:00:00.000000+0000	Transgender	21	1995-10-14 16:00:00.000000+0000	Tracie Kennison	5669	ICU
17	9487 Blackburn Ave,Glendale Heights, IL 60139	2003-03-01 16:00:00.000000+0000	2003-11-09 16:00:00.000000+0000	Male	17	2015-06-06 16:00:00.000000+0000	Darline Hayden	7326	ICU
24	89 Branch Rd,Champlin, MN 55316	1986-09-03 16:00:00.000000+0000	1986-03-26 16:00:00.000000+0000	Transgender	24	2006-09-20 16:00:00.000000+0000	Stony Pitcone	6633	ICU
25	53 Spruce St,Wilmington, MA 01887	1994-12-29 16:00:00.000000+0000	1991-10-08 16:00:00.000000+0000	Female	25	2002-12-11 16:00:00.000000+0000	Grazyna Parkson	6817	Observation
12	878 Bank St,Elgin, IL 60120	1997-01-02 16:00:00.000000+0000	1993-01-23 16:00:00.000000+0000	Transgender	12	2007-03-17 16:00:00.000000+0000	Hipolito Inouye	7087	Urgent Care
3	815 San Pablo Street,Danbury, CT 06810	2010-07-16 16:00:00.000000+0000	2006-07-18 16:00:00.000000+0000	Male	3	2026-02-06 16:00:00.000000+0000	Lady Franceschini	7825	ICU

(30 rows)

cqlsh:test001> select * from BasicInfo;

patientid	address	createddate	dob	gender	id	lastvisited	patientname	phonenumber	status
23	645 Kent Drive,Trumbull, CT 06611	2011-03-12 16:00:00.000000+0000	2008-09-11 16:00:00.000000+0000	testnum	23	2017-04-22 16:00:00.000000+0000	Ji Klink	4787	Observation
5	7718 Liliac Street,Annapolis, MD 21401	1992-08-02 16:00:00.000000+0000	1989-01-06 16:00:00.000000+0000	Female	5	1998-05-05 16:00:00.000000+0000	Chastity Gunning	6426	Weekly Visit
8	208 Silver Spear Street, Oaksh, MI 54901	1999-04-04 16:00:00.000000+0000	1999-04-04 16:00:00.000000+0000						
10	9073 Bradford Court,Holland, MI 49423	1999-06-14 16:00:00.000000+0000	1998-03-05 16:00:00.000000+0000						
16	8398 Sunset Dr,Tualatin, OR 97062	2006-02-15 16:00:00.000000+0000	2006-12-26 16:00:00.000000+0000						
13	8388 Sheffield St,Bolot, WI 53511	2009-04-15 16:00:00.000000+0000	2007-02-02 16:00:00.000000+0000						
11	6 Tallwater Street,West Chicago, IL 60185	2011-05-28 16:00:00.000000+0000	2005-09-06 16:00:00.000000+0000						
1	8654 Edgefield St,Brick, NJ 08723	1996-04-21 16:00:00.000000+0000	1990-04-25 16:00:00.000000+0000						
19	662 Sulphur Springs Rd,Fuquay Varina, NC 27526	1996-03-21 16:00:00.000000+0000	1988-09-17 16:00:00.000000+0000						
8	423 Beacon Ave,Lorton, VA 22079	1991-09-25 16:00:00.000000+0000	1989-04-15 16:00:00.000000+0000						
0	9482 Newport St,Whitestone, NY 11357	2002-05-20 16:00:00.000000+0000	1997-08-22 16:00:00.000000+0000						
2	8704 Hillside St,Freehold, NJ 0772	1996-09-20 16:00:00.000000+0000	1996-02-15 16:00:00.000000+0000						
4	7969 Hanover Court,Maspeth, NY 11378	1997-06-16 16:00:00.000000+0000	1996-05-15 16:00:00.000000+0000						
18	9389 Bayport Street,Eastpointe, MI 48021	2002-03-14 16:00:00.000000+0000	2001-09-07 16:00:00.000000+0000						
15	535 Del Monte St,Farmington, MI 48331	1980-01-16 16:00:00.000000+0000	1980-05-17 16:00:00.000000+0000						
22	49 S Glenholme St,Christiansburg, VA 24073	1992-03-28 16:00:00.000000+0000	1992-09-24 16:00:00.000000+0000						
27	8758 Goldfield St,Eastpointe, MI 48021	1982-07-12 16:00:00.000000+0000	1981-08-01 16:00:00.000000+0000						
20	7256 North Courtland Drive,Middletown, CT 0645	2015-05-10 16:00:00.000000+0000	2008-11-15 16:00:00.000000+0000						
7	11 Leeton Ridge St,Llnt, MI 48504	2010-03-09 16:00:00.000000+0000	2003-07-20 16:00:00.000000+0000						
6	56 Glen Eagles Rd,Desoto, TX 75115	1994-01-14 16:00:00.000000+0000	1988-07-24 16:00:00.000000+0000						
29	82 East 10th Street,Loxahatchee, FL 33470	1987-01-04 16:00:00.000000+0000	1986-02-24 16:00:00.000000+0000						
9	84 Fordham St,Lithonia, GA 30038	1988-10-11 16:00:00.000000+0000	1988-08-20 16:00:00.000000+0000						
14	79 Miles Ave,Dearbourn Heights, MI 48127	2005-06-08 16:00:00.000000+0000	2005-03-10 16:00:00.000000+0000						
26	350 NE. Gonzales St,Encino, CA 91316	1998-04-26 16:00:00.000000+0000	1998-11-20 16:00:00.000000+0000						
21	8963 Peachtree Lane,Little Rock, AR 72209	1988-10-07 16:00:00.000000+0000	1984-01-01 16:00:00.000000+0000						
17	9487 Blackburn Ave,Glendale Heights, IL 60139	2003-03-01 16:00:00.000000+0000	2003-11-09 16:00:00.000000+0000						
24	89 Branch Rd,Champlin, MN 55316	1986-09-03 16:00:00.000000+0000	1986-03-26 16:00:00.000000+0000						
25	53 Spruce St,Wilmington, MA 01887	1994-12-29 16:00:00.000000+0000	1991-10-08 16:00:00.000000+0000						
12	878 Bank St,Elgin, IL 60120	1997-01-02 16:00:00.000000+0000	1993-01-23 16:00:00.000000+0000						
3	815 San Pablo Street,Danbury, CT 06810	2010-07-16 16:00:00.000000+0000	2006-07-18 16:00:00.000000+0000						

(30 rows)

Task Manager

File Options View



Processes Performance App history Startup Users Details Services

Name	PID	Description	Status	Group
cplspcon		Intel(R) Content Protection HD...	Stopped	
CryptSvc	3932	Cryptographic Services	Running	NetworkService
CxAudMsg	4004	Conexant Audio Message Ser...	Running	
CxUtilSvc	3908	CxUtilSvc	Running	
DataStax DDC Server	13044	DataStax DDC Server 3.9.0	Running	
dbupdate		Dropbox Update Service (dbu...	Stopped	
dbupdateam		Dropbox Update Service (dbu...	Stopped	
DbxSvc	3952	DbxSvc	Running	
DcomLaunch	560	DCOM Server Process Launcher	Running	DcomLaunch

Fewer details | Open Services

Environment: ActiveMQ

← → ↻ ⓘ localhost:8161/admin/queues.jsp;jsessionid=1kbmh6tte2p7k1kjku9pw3tbm 🔍 ☆ 🛡



Home | Queues | Topics | Subscribers | Connections | Network | Scheduled | Send Support

Queue Name Create Queue Name Filter Filter

Queues:

Name ↑	Number Of Pending Messages	Number Of Consumers	Mess
EmailQueue	0	0	0
getAllIQ_a	0	0	0
getAllI1_a	0	0	0
getAllQ_20171208_1921	1	0	0
getAllQ_a	0	0	0
getAllQ_asdf	7	0	0
getAllQ_blah	3	0	0
getAllQ_ttt20171208_1959	1	0	0

Task Manager

File Options View

Processes Performance App history Startup Users Details Services

Name	PID	Description	Status	Group
ActiveMQ	3896	ActiveMQ	Running	
AdobeARMservice	3916	Adobe Acrobat Update Service	Running	
AESMSERVICE	9864	Intel® SGX AESM	Running	
AJRouter		AllJoyn Router Service	Stopped	LocalServiceNet...
ALG		Application Layer Gateway Ser...	Stopped	
AppIDSvc		Application Identity	Stopped	LocalServiceNet...
Appinfo	6360	Application Information	Running	netshvc
AppReadiness		App Readiness	Stopped	AppReadiness
AppXSvc		AppX Deployment Service (A...	Stopped	wsappx

⬅ Fewer details | 🔗 Open Services

Browse Active Consumers

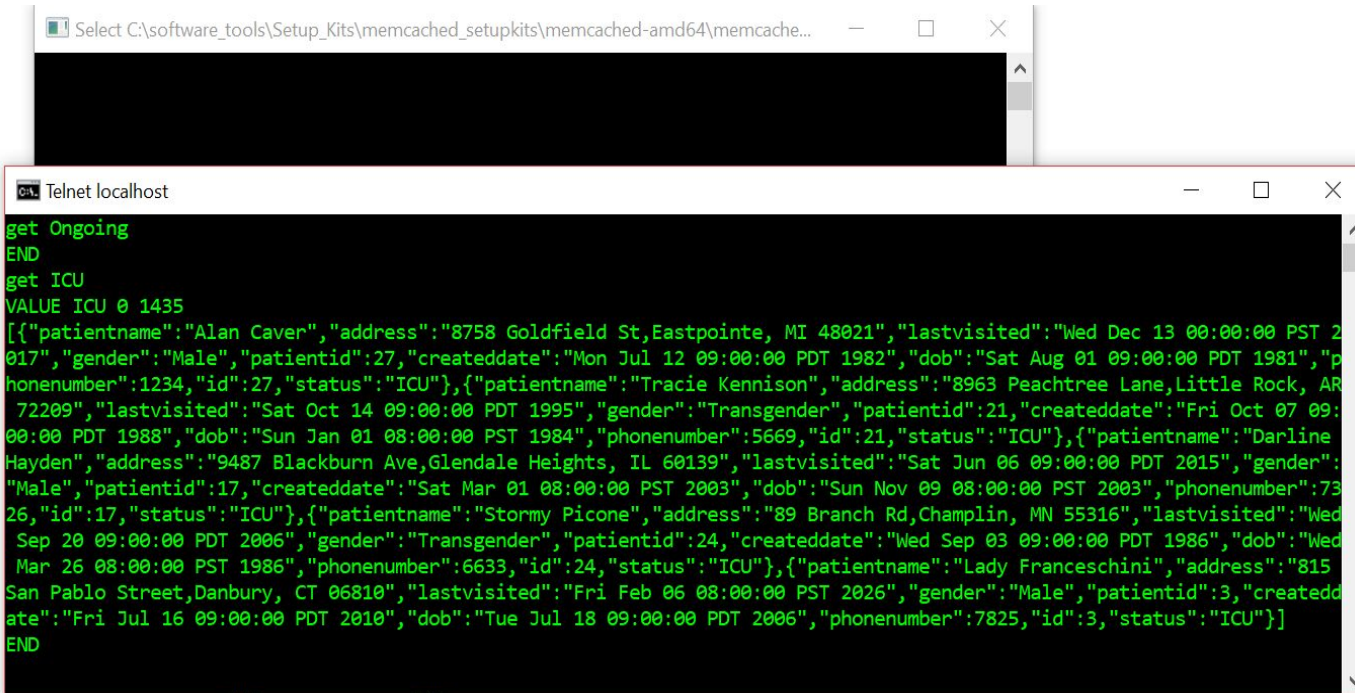
Active Producers

Send To Purge Delete

Browse Active Consumers

Environment: Memcache

Memcache telnet console: When the policy is set to ICU



```
Select C:\software_tools\Setup_Kits\memcached_setupkits\memcached-amd64\memcache...  
C:\ Telnet localhost  
get Ongoing  
END  
get ICU  
VALUE ICU 0 1435  
[{"patientname":"Alan Caver","address":"8758 Goldfield St,Eastpointe, MI 48021","lastvisited":"Wed Dec 13 00:00:00 PST 2017","gender":"Male","patientid":27,"createddate":"Mon Jul 12 09:00:00 PDT 1982","dob":"Sat Aug 01 09:00:00 PDT 1981","phonenummer":1234,"id":27,"status":"ICU"}, {"patientname":"Tracie Kennison","address":"8963 Peachtree Lane, Little Rock, AR 72209","lastvisited":"Sat Oct 14 09:00:00 PDT 1995","gender":"Transgender","patientid":21,"createddate":"Fri Oct 07 09:00:00 PDT 1988","dob":"Sun Jan 01 08:00:00 PST 1984","phonenummer":5669,"id":21,"status":"ICU"}, {"patientname":"Darline Hayden","address":"9487 Blackburn Ave, Glendale Heights, IL 60139","lastvisited":"Sat Jun 06 09:00:00 PDT 2015","gender":"Male","patientid":17,"createddate":"Sat Mar 01 08:00:00 PST 2003","dob":"Sun Nov 09 08:00:00 PST 2003","phonenummer":7326,"id":17,"status":"ICU"}, {"patientname":"Stormy Picone","address":"89 Branch Rd, Champlin, MN 55316","lastvisited":"Wed Sep 20 09:00:00 PDT 2006","gender":"Transgender","patientid":24,"createddate":"Wed Sep 03 09:00:00 PDT 1986","dob":"Wed Mar 26 08:00:00 PST 1986","phonenummer":6633,"id":24,"status":"ICU"}, {"patientname":"Lady Franceschini","address":"815 San Pablo Street, Danbury, CT 06810","lastvisited":"Fri Feb 06 08:00:00 PST 2026","gender":"Male","patientid":3,"createddate":"Fri Jul 16 09:00:00 PDT 2010","dob":"Tue Jul 18 09:00:00 PDT 2006","phonenummer":7825,"id":3,"status":"ICU"}]  
END
```

What we referred to learn concepts

- Jersey example:
 - <https://www.mkyong.com/webservices/jax-rs/jersey-hello-world-example/>
- Java Cassandra interaction:
 - <http://www.baeldung.com/cassandra-with-java>
- Java ActiveMQ Interaction:
 - <http://activemq.apache.org/hello-world.html>
 - <https://www.youtube.com/watch?v=yQcC6Z0PPes&t=879s>
 - <https://www.youtube.com/watch?v=3T-IDT-vALE>
- Memcache Interaction:
 - <https://www.tutorialspoint.com/memcached>

Code walkthrough

- 3 Main Projects:
 - “HappyPatients” main App
 - “CS249PolicyServer” cache policy server
 - “hpMessageConsumers” email and analytics consumers
- Sample Data Generated through Python

Thank You