

NOW / NEXT

*M*Pages and *HealthIntent* Open APIs

Patrick Elam

Software Engineer, HealthIntent Open API Team

Scott Worthington

Senior Software Engineer, HealthIntent Care Management Team

About Us

NOW/NEXT

Patrick Elam

Software Engineer,
HealthIntent Open API



I work to enhance the internal and external developer experience with *HealthIntent's* Open API Platform.

Scott Worthington

Senior Software Engineer,
HealthIntent Care Management



I develop solutions using Cerner's Population Health framework, helping patients and providers achieve optimal outcomes.

Goals for Today

Overview of HealthIntent and MPages

Tools for MPages and HealthIntent development

Code examples for a custom component

Live Demo ?

Overview of MPages

MPage Overview

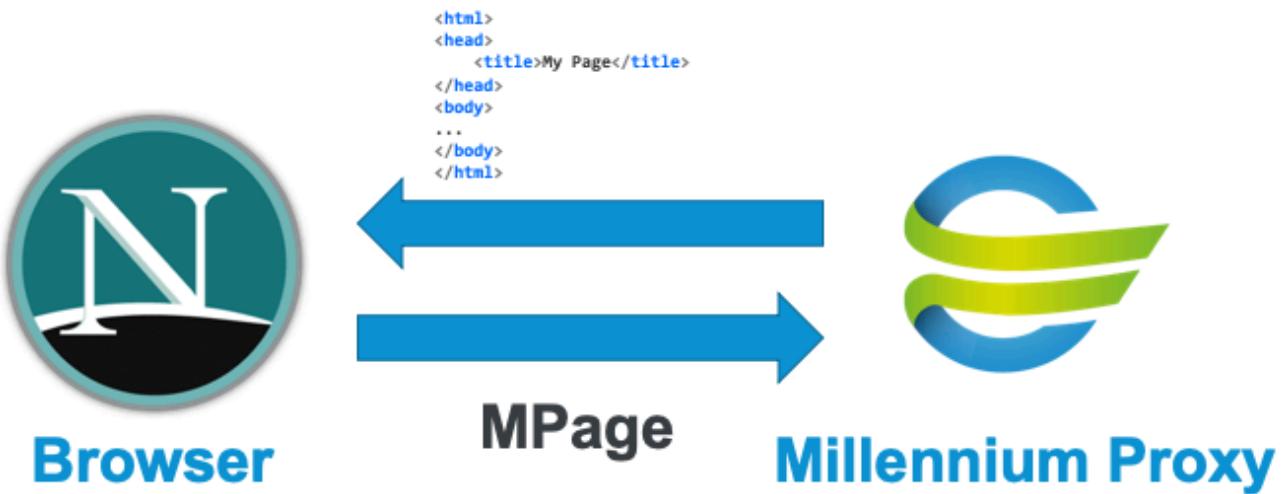
MPages Gaia



MPages Gaia is a **highly extensible** development ecosystem for creating MPages artifacts and **streamlining** the MPages development process.

MPage Overview

MPages Gaia



MPage Overview

MPages Gaia

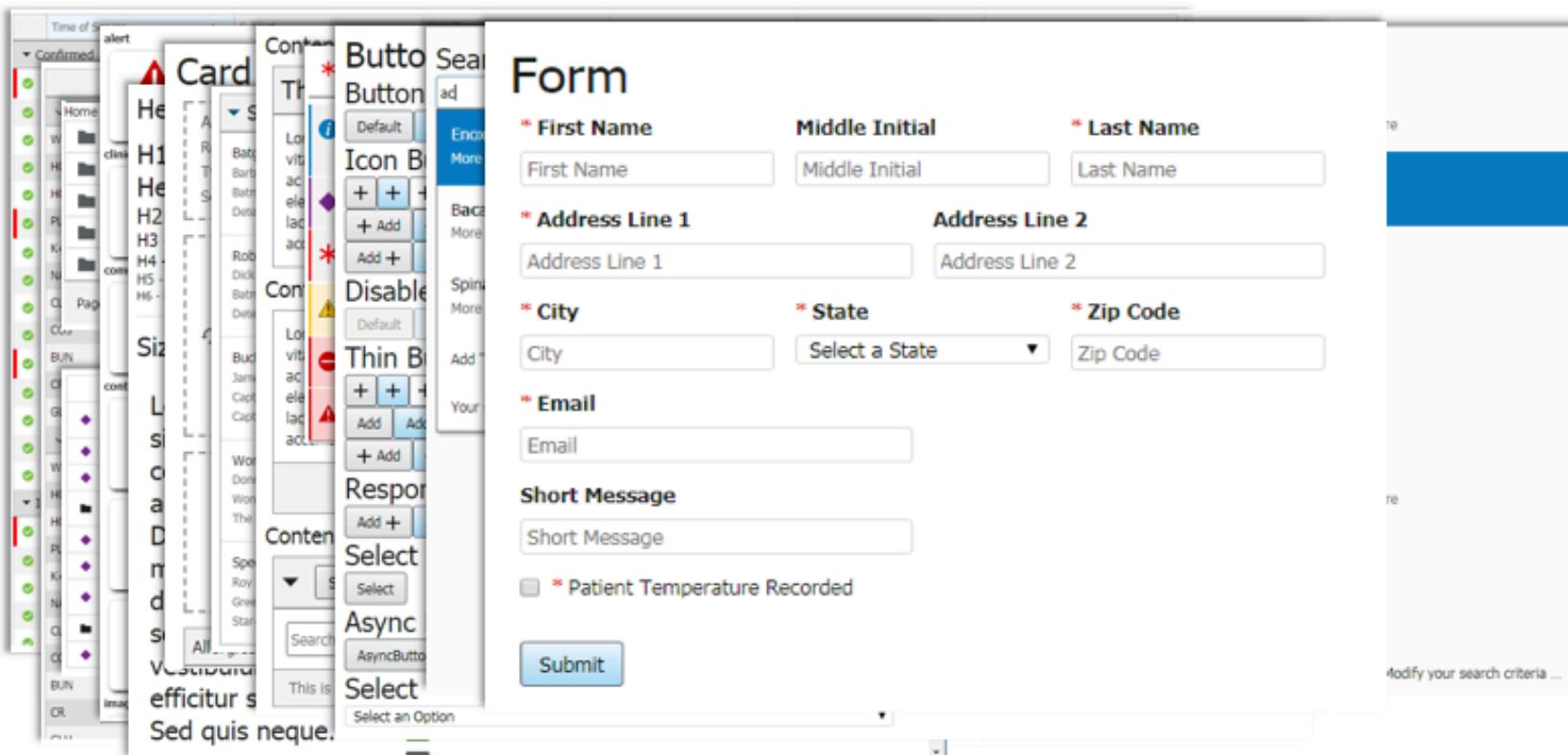


Fusion is a code-companion library of reusable UI elements that are responsive and aligned with Cerner standard style.

MPage Overview

MPages Fusion

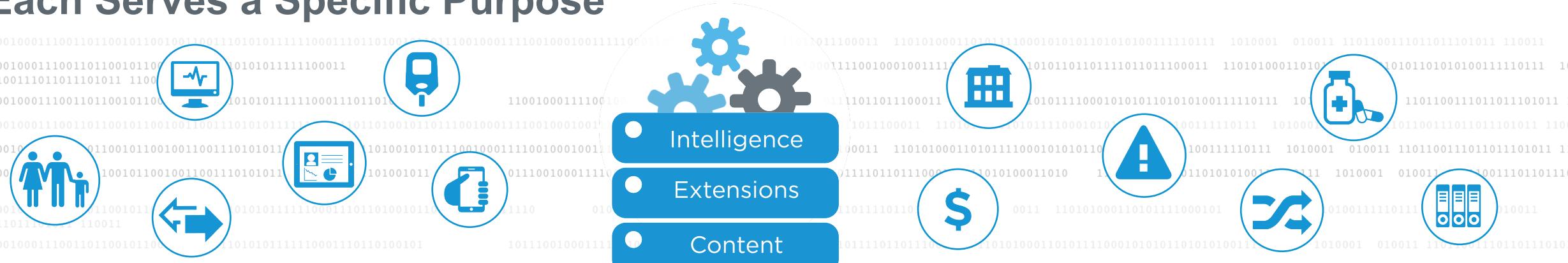
- More than 60 standard controls



Overview of HealthIntent and the Open API Platform

Cerner's Open Platforms

Each Serves a Specific Purpose



CareAware

Device connectivity
and communications



Bidirectional
device data



Capacity
management



Communications
& alerting



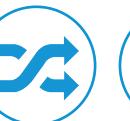
Location/
RTLS



Prescriptions



Patient
record



Clinical
workflows



Revenue
cycle



Research
Longitudinal
Data



Populations
Data



Registries



Cerner Millennium

Electronic health record

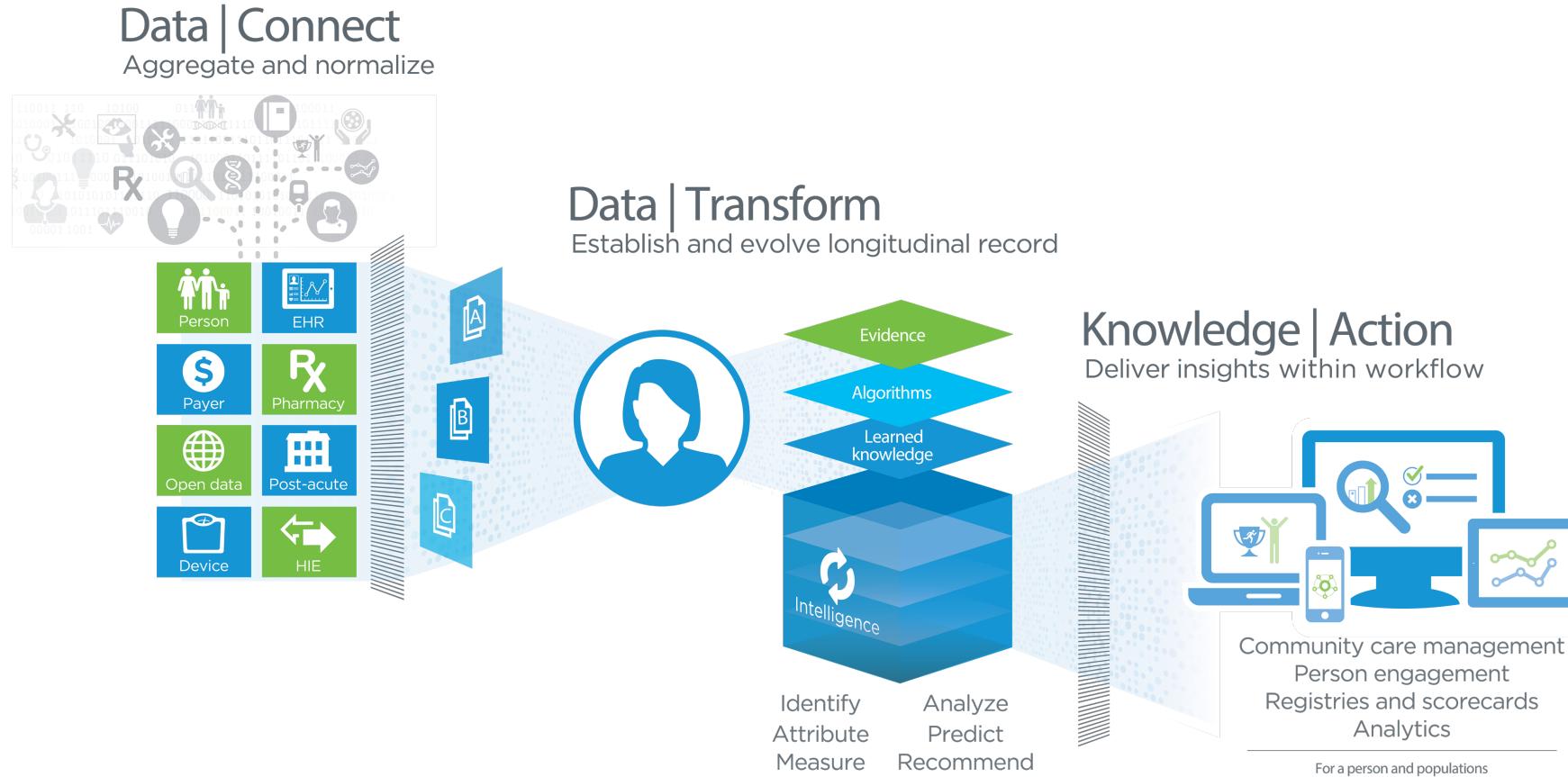


HealthIntent

Population health management

NOW/NEXT

HealthIntent Platform Overview



Traditional Approach: Applications

The screenshot shows the HealtheRegistries application interface. At the top, there is a navigation bar with links for "Scorecards" and "Registries". On the right side of the header, there are icons for "Normal view", "Print", and "0 minutes". Below the header, the user's name "Phyllis Robinson" is displayed along with a search bar.

The main content area features a large banner for "Phyllis Robinson" with the following information:

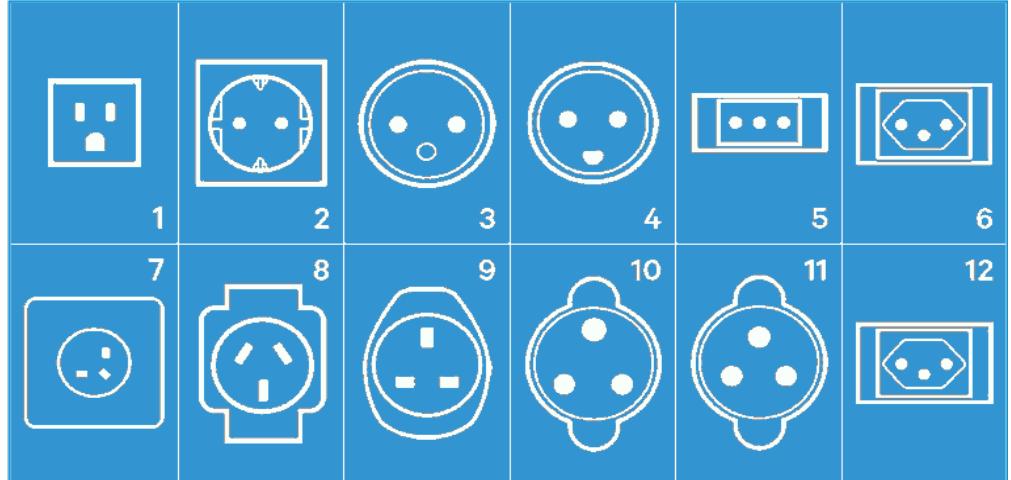
- Risk Score: 0.62
- Organizations: Baseline West Medical Center, Rockcreek Clinic
- A progress bar indicates 36.52% completion.
- Top opportunity to improve score: Blood Pressure < 140/90 mm Hg
- 43% of measure data is completed
- 50 Scorable Persons

Below the banner, there is a section for "Announcements" with one item. There is also a "Help Page Information" box dated Jun 17, 2015. At the bottom, there is a "Registry Population Statistics" section showing data for 73 Persons across three categories: measures achieved, measure data entered, and registries performance.

APIs Have Transformed Other Industries

We believe they can do the same for healthcare!

- Application Programming Interfaces (APIs):
 - Allow technical solutions to talk to one another in the same language
 - Easily expose services to make them available inside and outside an organization
- Cerner's open platforms expose standards that make it possible to connect apps or easily transfer discrete data.



New Approach: API Integrations

The screenshot displays the eClinicalWorks 10e medical software interface. At the top, a header bar shows the URL [10.210.22.45:8094/mobiledoc/jsp/webemr/index.jsp#/>](http://10.210.22.45:8094/mobiledoc/jsp/webemr/index.jsp#/). Below the header is a navigation bar with links for Medical Summary, CDSS, Labs, DI, Procedures, Growth Chart, Imm, T.Inj, Encounters, Patient Docs, Flowsheets, Notes, Progress Note, Scribe, and Orders.

The main patient summary area shows the following details for patient Ibrahim, Aaron:

- Patient: Ibrahim, Aaron
- DOB: 08/31/1983
- Age: 32 Y
- Sex: Male
- Phone: 101-206-4411
- Primary Insurance: Ins: Self Pay
- Address: 5 Heritage St, Westborough, MA 01583
- Account Number: 9294
- Encounter Date: 06/06/2016
- Provider: WILLIS SAM, MD (TEST LICENSE)
- Appointment Facility: Westborough Medical Associates

The interface is divided into sections for Subjective, Objective, Assessment, and Plan. Each section contains dropdown menus for Chief Complaint(s), HPI, Current Medication, Medical History, Allergies/Intolerance, Surgical History, Hospitalization, Family History, Social History, and ROS.

A red box highlights the **UpToDate** module, which is integrated into the software. The UpToDate window displays the following information:

- Quality Score: NA | NA
- PHA Alerts: 0
- Cost Utilization: NA
- Risk Score: 1.4
- Quality Score: 50%

The UpToDate module is organized into sections for EXTERNAL REGISTRIES, HYPERTENSION (5 OUT OF 7 MET), Blood Pressure Management, Blood Pressure Measurement, High Blood Pressure Plan of Care, Serum Creatinine, Tobacco Use Screening and Cessation, SENIOR WELLNESS (6 OUT OF 15 MET), Alcohol Use Screening, Blood Pressure Measurement, and Depression Screening.

New Approach: API Integrations

- A fully integrated panel within AllScripts displays gaps of care for patients via a SmartRegistry API call to SmartData.

The screenshot shows a web-based application window titled "SmartRegistry". At the top, there are fields for "Patient Name: /", "MRN:", "Sex: Female", "DOB:", and "Patient ID:" with a "Close" button. Below this, a message says "Registries information is up-to-date".

The main content area is divided into several sections:

- Adult Diabetes Supporting Facts:**
 - Diabetes mellitus
 - Monitored patient (EMR)
2017-10-13
Mammogram ECRIS Claims (Claim)
 - HEMOGLLOBIN A1c 6.2
 - 2018-04-17
ECWV EMR (EMR)
 - DM II [Diabetes mellitus type II]
 - 2018-04-17
ECWV EMR (EMR)
 - insulin regular, PYXIS
 - 2018-04-12
Monitored patient (EMR)
Encounter with Brown, Jeffrey, Brown, Jeffrey, Wisniski, Sherry D, Brown, Jeffrey and Bryant, Rebecca L
2018-04-11
Monitored patient Seminole CO Hospital, Seminole patient (EMR)
- Adult Wellness** (4 of 9 met)

Measure Name	Due Date	Outcome	Date
✓ Annual Office Visit	—	Achieved	2017-10-13
● Blood Pressure Measurement	2018-04-10	Not Achieved	2017-04-10
● Body Mass Index	2018-04-10	Not Achieved	2017-04-10
✓ Cervical Cancer Screen (ages 21-64)	2022-10-12	Achieved	2017-10-13
● Depression Screening	2016-07-05	Not Achieved	2015-07-06
✓ HIV Screening	—	Non React	2013-02-06
● Influenza Vaccination	—	Not Achieved	2013-11-04
● Lipid Panel	—	Excluded	2015-07-06
● Tdap Vaccination	—	—	—
✓ Tobacco Use Screening and Cessation	2019-04-10	Performed	2017-04-10
- Supporting Facts:**
 - neversmoker
 - 2017-04-10
ECWV EMR (EMR)
 - Patient screened for tobacco use and received tobacco cessation intervention (counseling, pharmacotherapy, or both), if identified as a tobacco user (PV, CAD)
 - 2018-04-28
Monitored patient ECRIS Claims

1

Industry Standard Technologies

Secure and Easy



Design

- HealthIntent APIs are designed with respect to REST patterns



Invocation

- HTTP is the invocation and communication protocol for all HealthIntent APIs



Messages

- Most HealthIntent APIs use JSON for structuring data



Security

- API security is system-oriented
- Authentication – the OAuth 2.0 standard w/ bearer tokens is used
- Authorization – accounts are granted permissions that are enforced by APIs

HealthIntent APIs

Longitudinal Record

Allergy
Chart Search
Cohort
Condition
Immunization
Medication
Observation
Patient (includes demographics)
Procedure
Risk Assessment

Value-Based and Quality Care

HCC
HealtheRegistries

Care Coordination and Planning

Health Concern
Healthcare Service
Longitudinal Plan
Network
Provider
Referral Request
Specialty
Transition of Care
Risk Assessment
Wellness

Consumer Engagement

Application Experience
Consumer Notifications
Consumer Engagement
HealtheLife Framework SDK

Data Ingestion, Migration, and Core Build

Data Source
Data Syndication
Master Person Management
Ontology
Tenant
Universal Data Ingestion

... and more!

Identity Services

Consumer
Organization
Personnel
Specialty

Building MPages That Call HealthIntent APIs

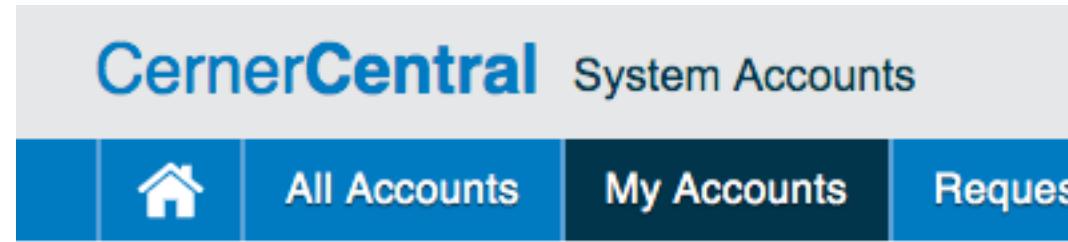
Data Flow

Overview



Authorization Workflow

System Accounts



System Account ID: 1234-5678-91011

\$uPer(/*\$e_creT&ApI^T0Ken

Authorization Workflow

System Account Authorization

HealthIntent Console

Tenants

System Account

ID

Name

Description

OPEN_API_SUPPORT

API Authorization

Longitudinal Plan

Allergy

+ Add

allergy:populations/{population_id}/allergies

Permissions

Read Write

API

Allergy

Application Experience

CareAware Ingestion

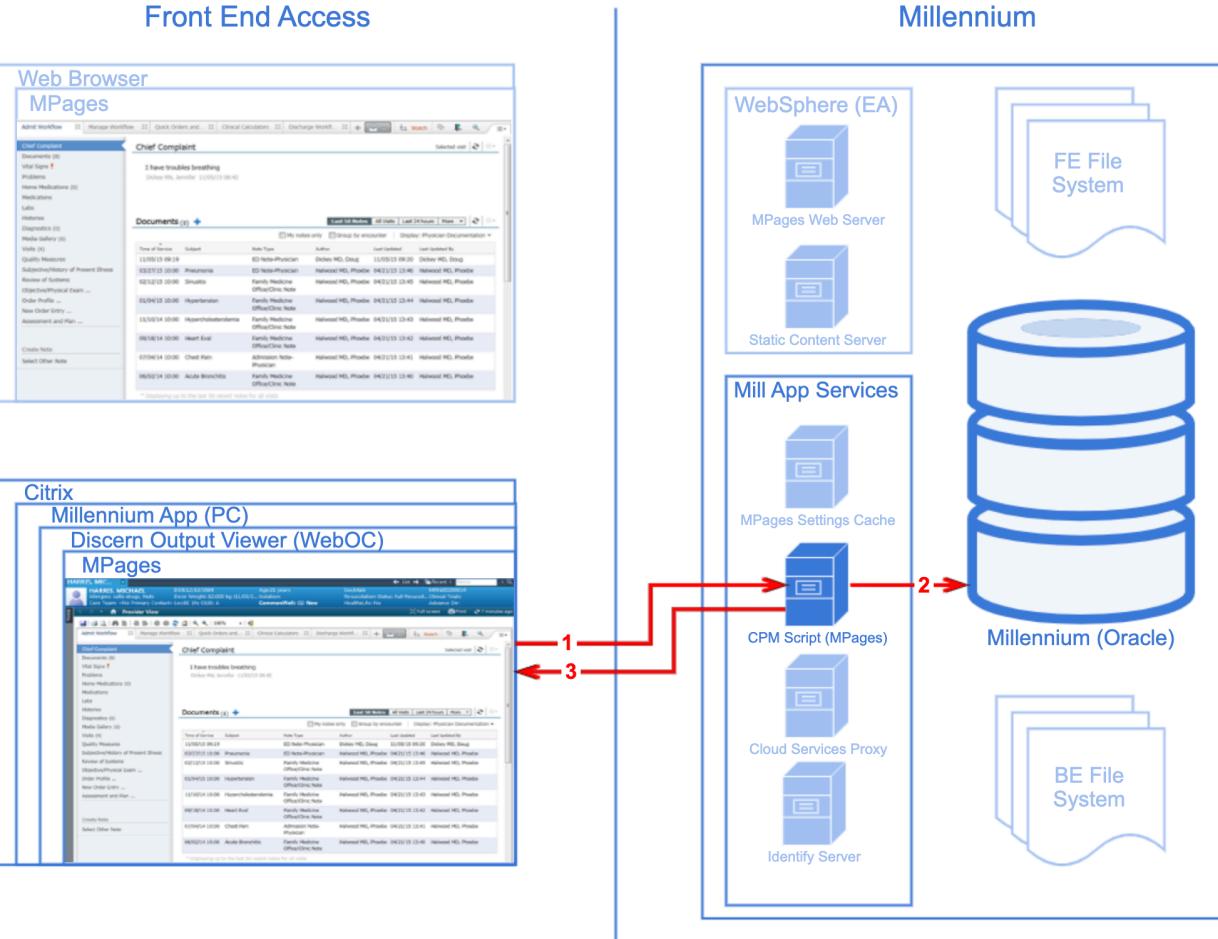
Chart Search

Cohort

The screenshot shows the HealthIntent Console interface for managing system accounts. The left sidebar has tabs for Tenants, Populations, API Authorization (which is selected), and Longitudinal Plan. The main area is titled 'View System Account:' and shows a 'System Account' with ID 'OPEN_API_SUPPORT'. It has fields for Name and Description. Below this, under 'APIs', the 'Allergy' API is selected from a list. To the right, there's a table for managing permissions for the 'Allergy' API, showing a single entry 'allergy:populations/{population_id}/allergies' with 'Permissions' set to 'Read Write'. There are also '+ Add' and '*' buttons.

Data Flow

MPage to CCL



Data Flow

CCL to *HealthIntent* API

Millennium Servers:

- *Millennium* Identity Server (SCP 387)
- HTTP Proxy Server (SCP 543)

HealthIntent CCL Scripts (package 104367 or replacement):

- `hi_http_proxy_post_request`
- `hi_http_proxy_get_request`
- `hi_http_proxy_put_request`
- `hi_http_proxy_delete_request`

Creating a Springboard Example

Springboard Example

Standard *HealthIntent* Call: Patient ID Lookup

GET https://cernerdemo.api.us.healthintent.com/patient/v1/populations/1424e81d-8cea-4d6b-b140-d6630b684a58/patient-id-lookup?dataPartitionId=877307a0-b5f5-4a01-9d4b-9fead6bcf788&sourcePersonId=132676

Send Save

Params ● Authorization Headers (8) Body Pre-request Script Tests Cookies Code Comments (0)

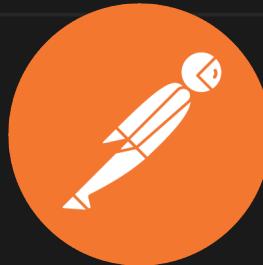
Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> dataPartitionId	877307a0-b5f5-4a01-9d4b-9fead6bcf788			
<input checked="" type="checkbox"/> sourcePersonId	132676			
Key	Value	Description		

Body Cookies Headers (8) Test Results Status: 200 OK Time: 334ms Size: 388 B Save Response

Pretty Raw Preview JSON ⚡

```
1 {
2   "items": [
3     {
4       "sourcePersonId": "132676",
5       "patient": {
6         "id": "001b8e97-c663-477f-aeb6-249997d52bdf"
7       }
8     }
9   ]
10 }
```



A screenshot of a Postman API request interface. The top bar shows a GET request to a Cerner demo API endpoint for a patient ID lookup. The 'Params' tab is selected, showing two query parameters: 'dataPartitionId' with value '877307a0-b5f5-4a01-9d4b-9fead6bcf788' and 'sourcePersonId' with value '132676'. Below the table, the 'Body' tab is selected, showing the JSON response. The response is a single object with a 'items' key containing an array of objects. One object in the array is highlighted with a blue box, showing its 'sourcePersonId' and 'patient' object with an 'id' key. The bottom right corner features a stylized orange circle icon with a white pen nib and a small figure holding it.

Springboard Example

Custom CCL: Patient ID Lookup

```
chc19demo > ccl >  CHC_HI_PATIENT_LOOKUP.prg
 1  drop program chc_hi_patient_lookup go
 2  create program chc_hi_patient_lookup
 3
 4  ; Parameters for the script
 5  ; Output:                      Set output to "MINE"
 6  ; Population ID:              The population ID of the patient.
 7  ; Data Partition ID:          The data partition ID of the patient
 8  ; Millennium Person ID:     The local patient ID in the calling system.
 9  prompt
10  "Output" = "MINE",
11  "Population ID" = "",
12  "Data Partition ID" = "",
13  "Millennium Person ID" = 0.0
14
15  with OUTDEV, populationID, dataPartitionID, millenniumPersonID
16
```

Springboard Example

Custom CCL: Patient ID Lookup

```
25  
26 ; Build URL for patient look-up  
27 set URL = build2("https://", hiTenant  
28 | | | , ".api.us.healtheintent.com/patient/v1/populations/", populationID  
29 | | | , "/patient-id-lookup?"  
30 | | | , "dataPartitionId=", dataPartitionID  
31 | | | , "&sourcePersonId=", CNVTSTRING(millenniumPersonID, 0))  
32  
33 ; Look up HealtheIntent patient ID  
34 execute hi_http_proxy_get_request "MINE", URL  
35
```

Springboard Example

Custom CCL: Patient ID Lookup

```
36 ; Check response code
37 if(proxyReply->httpreply->status = 200)
38
39 ; Declare the record structure to be returned
40 free record OUTREC
41 record OUTREC(
42     1 STATUS_CODE = i4
43     1 HI_PATIENT_ID = vc
44 )
45
```

Springboard Example

Custom CCL: Patient ID Lookup

```
48  
49    ; Pull out HealtheIntent person ID  
50    free record REPLY_REC  
51    set stat = CNVTJSONTOREC(build2(^{"REPLY_REC":^, proxyReply->httpreply->body, "}))  
52    if(validate(proxyReply->httpreply->STATUS))  
53        set OUTREC->STATUS_CODE = proxyReply->httpreply->STATUS  
54    endif  
55    if(validate(REPLY_REC->items[1].patient.id))  
56        set OUTREC->HI_PATIENT_ID = REPLY_REC->items[1].patient.id  
57    endif  
58  
59    ; Return JSON structure containing the Patient ID  
60    call ReturnRecordAsJSON(OUTREC)  
61    call echorecord(OUTREC)  
62
```

Springboard Example

Custom CCL: Patient ID Lookup

```
61  
62  
63  
64  
65      ; Declare the record structure to be returned  
66      free record OUTREC  
67      record OUTREC(  
68          1 STATUS_CODE = i4  
69          1 ERROR_MESSAGE = vc  
70      )  
71  
72      ; Set return values  
73      if(validate(proxyReply->httpreply->STATUS))  
74          set OUTREC->STATUS_CODE = proxyReply->httpreply->STATUS  
75      endif  
76      if(validate(proxyReply->httpreply->body))  
77          set OUTREC->ERROR_MESSAGE = proxyReply->httpreply->body  
78      endif  
79  
80      ; Return JSON structure containing the error message  
81      call ReturnRecordAsJSON(OUTREC)  
82      call echorecord(OUTREC)  
83  
84      endif
```

Springboard Example

Custom CCL: Patient ID Lookup

```
Command executed:  
1)execute chc_hi_patient_lookup "MINE", "1424e81d-8cea-4d6b-b140-d6630b684a58", "877307a0-b5f5-4a01-9d4b-9fead6bcf788", 132676.0 go  
>>>Begin EchoRecord PREF_RECORD ;HI_PREF_REC  
1 PREFERENCE_VALUES[0,0*]  
>>>End EchoRecord PREF_RECORD Varchar=0, Varlist=0, Fixsize=12, Varsize=0  
test_stat  
URL: https://cernerdemo.api.us.healtheintent.com/patient/v1/populations/1424e81d-8cea-4d6b-b140-d6630b684a58/patient-id-lookup?data  
PartitionId=877307a0-b5f5-4a01-9d4b-9fead6bcf788&sourcePersonId=132676  
>>>Begin EchoRecord OUTREC ;OUTREC  
1 STATUS_CODE= I4 {200}  
1 HI_PATIENT_ID=VC36 {001b8e97-c663-477f-aeb6-249997d52bdf}  
>>>End EchoRecord OUTREC Varchar=1, Varlist=0, Fixsize=10, Varsize=10  
190919:102637 CHC_HI_PATIENT_LOOKUP Cost 0.00 Cpu 0.01 Ela 0.54 Dio 0 00M0R0 P3R0
```

Springboard Example

Custom CCL: Patient Demographics

```
4 ; Parameters for the script
5 ; Output:                      Set output to "MINE"
6 ; Population ID:                The population ID of the patient.
7 ; HealtheIntent Patient ID:    The HealtheIntent patient identifier.
8 prompt
9   "Output" = "MINE",
10  "Population ID" = "",
11  "HealtheIntent Patient ID" = 0.0
12 with OUTDEV, populationID, healtheintentPatientID
13

22 ; Build URL to call the patient demographics list endpoint.
23 ; https://docs.healtheintent.com/api/v1/patient/#retrieve-a-list-of-demographic-records
24 set URL = build2("https://", hiTenant
25   , ".api.us.healtheintent.com/patient/v1/populations/", populationID
26   , "/patients/", hiPatientID
27   , "/demographics")
28
29 ; Make call
30 execute hi_http_proxy_get_request "MINE", URL
31
```

Springboard Example

Custom CCL: Patient Demographics

```
31
32 ; Check response code
33 if(proxyReply->httpreply->status = 200)
34
35 ; Declare the record structure to be returned
36 ; Note that this is a small subset of the information that is provided by
37 ; the API for demonstration purposes.
38 free record OUTREC
39 record OUTREC(
40     1 STATUS_CODE = i4
41     1 FORMATTED_NAME = vc
42     1 DEMOGRAPHIC_ID = vc
43     1 BIRTH_DATE = vc
44     1 BIRTH_SEX = vc
45     1 MARITAL_STATUS = vc
46     1 RACE = vc
47     1 ETHNICITY = vc
48 )
```

Springboard Example

Custom CCL: Patient Demographics

Springboard Example

Custom CCL: Patient Demographics

```
1)execute chc_hi_patient_demographics "MINE", "1424e81d-8cea-4d6b-b140-d6630b684a58", "0004200f-9353-40e8-b953-291263369db1" go
>>>Begin EchoRecord PREF_RECORD ;HI_PREF_REC
1 PREFERENCE_VALUES[0,0*]
>>>End EchoRecord PREF_RECORD Varchar=0, Varlist=0, Fixsize=12, Varsize=0
test_stat
>>>Begin EchoRecord OUTREC ;OUTREC
1 STATUS_CODE= I4 {200}
1 FORMATTED_NAME=VC15 {Jacque Figueiroa}
1 DEMOGRAPHIC_ID=VC64 {63384979344886e5467e9949d669e8b6acc4dfbd230d6a752662b618ba2dcbe}
1 BIRTH_DATE=VC10 {2010-02-13}
1 BIRTH_SEX=VC0 {}
1 MARITAL_STATUS=VC0 {}
1 RACE=VC0 {}
1 ETHNICITY=VC0 {}
>>>End EchoRecord OUTREC Varchar=7, Varlist=0, Fixsize=88, Varsize=149
190919:105747 CHC_HI_PATIENT_DEMOGRAPHICS Cost 0.00 Cpu 0.00 Ela 0.53 Dio 0 00M0R0 P3R0
```

Springboard Example

Custom CCL: Patient Allergies

```
; Declare the record structure to be returned
; Note that this is a small subset of the information that is provided by
; the API for demonstration purposes.
free record OUTREC
record OUTREC(
    1 STATUS_CODE = i4
    1 ALLERGIES[*]
        2 ASSERTED_ON = vc
        2 CATEGORY = vc
        2 CODE = vc
        2 CRITICALITY = vc
        2 ONSET = vc
        2 STATUS = vc
        2 REACTION_TYPE = vc
)
; Convert response to record structure
free record REPLY_REC
set stat = CNVTJSONTOREC(build2(^{"REPLY_REC":^, proxyReply->httpreply->body, "}))  

; Resize allergy list
set numAllergies = size(REPLY_REC->items, 5)
set stat = ALTERLIST(OUTREC->ALLERGIES, numAllergies)
```

Springboard Example

Custom CCL: Patient Allergies

```
57 ; Resize allergy list
58 set numAllergies = size(REPLY_REC->items, 5)
59 set stat = ALTERLIST(OUTREC->ALLERGIES, numAllergies)
60
61 ; Set Return values
62 if(validate(proxyReply->httpreply->STATUS))
63     set OUTREC->STATUS_CODE = proxyReply->httpreply->STATUS
64 endif
65 for(num = 1 to numAllergies)
66     if(validate(REPLY_REC->items[num].assertedon))
67         set OUTREC->ALLERGIES[num].ASSERTED_ON = REPLY_REC->items[num].assertedon
68     endif
69     if(validate(REPLY_REC->items[num].category.text))
70         set OUTREC->ALLERGIES[num].CATEGORY = REPLY_REC->items[num].category.text
71     endif
72     if(validate(REPLY_REC->items[num].code.text))
73         set OUTREC->ALLERGIES[num].CODE = REPLY_REC->items[num].code.text
74     endif
```

Springboard Example

Custom CCL: Patient Allergies

```
1)execute chc_hi_patient_allergies "MINE", "1424e81d-8cea-4d6b-b140-d6630b684a58", "002c0901-e16f-464c-bfb0-44f469f5780a" go
>>>Begin EchoRecord PREF_RECORD ;HI_PREF_REC
 1 PREFERENCE_VALUES[0,0*]
>>>End EchoRecord PREF_RECORD Varchar=0, Varlist=0, Fixsize=12, Varsize=0
test_stat
>>>Begin EchoRecord OUTREC ;OUTREC
 1 STATUS_CODE= I4 {200}
 1 ALLERGIES[1,4*]
 2 ASSERTED_ON=VC24 {2015-11-30T08:58:09.000Z}
 2 CATEGORY=VC0 {}
 2 CODE=VC4 {Dust}
 2 CRITICALITY=VC8 {Moderate}
 2 ONSET=VC0 {}
 2 STATUS=VC6 {Active}
 2 REACTION_TYPE=VC7 {Allergy}
 1 ALLERGIES[2,4*]
 2 ASSERTED_ON=VC24 {2015-11-18T12:30:16.000Z}
```

Springboard Example

Custom CCL: Patient Registries

```
25
26 ; Build URL to call the registries summary endpoint.
27 ; https://docs.healtheintent.com/api/alpha/registries/
28 set URL = build2("https://" , hiTenant
29   , ".registries.healtheintent.com/api/populations/" , populationID
30   , "/people/" , hiPatientID
31   , "/registries")
32
33
34 ; Make call
35 execute hi_http_proxy_get_request "MINE" , URL
36
```

Springboard Example

Custom CCL: Patient Registries

```
37 ; Check response code
38 if(proxyReply->httpreply->status = 200)
39
40 ; Declare the record structure to be returned
41 ; Note that this is a small subset of the information that is provided by
42 ; the API for demonstration purposes.
43 free record OUTREC
44 record OUTREC(
45     1 STATUS_CODE = i4
46     1 QUALITY_SCORE = vc
47     1 MARA_RISK_SCORE = f8
48     1 PROGRAMS[*]
49         2 NAME = vc
50         2 MEASURE_COUNT = i4
51         2 MET_COUNT = i4
52         2 MEASURES[*]
53             3 DUE = i4
54             3 NAME = vc
55             3 OUTCOME = vc
56 )
```

Springboard Example

Custom CCL: Patient Registries

```
76  
77    for(i = 1 to numPrograms)  
78        if(validate(REPLY_REC->PROGRAMS[i].name))  
79            set OUTREC->PROGRAMS[i].name = REPLY_REC->PROGRAMS[i].name  
80        endif  
81        if(validate(REPLY_REC->PROGRAMS[i].total_measure_count))  
82            set OUTREC->PROGRAMS[i].measure_count = REPLY_REC->PROGRAMS[i].total_measure_count  
83        endif  
84        if(validate(REPLY_REC->PROGRAMS[i].met_measure_count))  
85            set OUTREC->PROGRAMS[i].met_count = REPLY_REC->PROGRAMS[i].met_measure_count  
86        endif  
87  
88        set numMeasures = size(REPLY_REC->PROGRAMS[i].measures, 5)  
89        set stat = ALTERLIST(OUTREC->PROGRAMS[i].measures, numMeasures)  
90        for(j = 1 to numMeasures)  
91            if(validate(REPLY_REC->PROGRAMS[i].measures[j].measure_due))  
92                set OUTREC->PROGRAMS[i].measures[j].due = REPLY_REC->PROGRAMS[i].measures[j].measure_due  
93            endif  
94            if(validate(REPLY_REC->PROGRAMS[i].measures[j].name))  
95                set OUTREC->PROGRAMS[i].measures[j].name = REPLY_REC->PROGRAMS[i].measures[j].name  
96            endif
```

Springboard Example

Custom CCL: Patient Registries

```
1)execute chc_hi_patient_registries "MINE", "1424e81d-8cea-4d6b-b140-d6630b684a58", "32ec2790-9747-4bbb-9ddc-187c78b71782" go
>>>Begin EchoRecord PREF_RECORD ;HI_PREF_REC
  1 PREFERENCE_VALUES[0,0*]
>>>End EchoRecord PREF_RECORD Varchar=0, Varlist=0, Fixsize=12, Varsize=0
test_stat
>>>Begin EchoRecord OUTREC ;OUTREC
  1 STATUS_CODE= I4 {200}
  1 QUALITY_SCORE=VC3 {42%}
  1 MARA_RISK_SCORE=F8 {0.0400000000}
  1 PROGRAMS[1,3*]
    2 NAME=VC14 {Adult Wellness}
    2 MEASURE_COUNT= I4 {9}
    2 MET_COUNT= I4 {3}
    2 MEASURES[1,9*]
      3 DUE= I4 {1}
      3 NAME=VC21 {Alcohol Use Screening}
      3 OUTCOME=VC8 {ACHIEVED}
    2 MEASURES[2,9*]
      3 DUE= I4 {1}
      3 NAME=VC19 {Annual Office Visit}
      3 OUTCOME=VC12 {MISSING_DATA}
    2 MEASURES[3,9*]
      3 DUE= I4 {1}
      3 NAME=VC26 {Blood Pressure Measurement}
      3 OUTCOME=VC8 {ACHIEVED}
```

Springboard Example

Custom CCL: Returning Output

```
15  with OUTDEV, populationID, dataPartitionID, millenniumPersonID  
16  
17  ; Declare a subroutine used to return the response from the API  
18  declare ReturnRecordAsJSON(record_data = VC(REF)) = null with protect  
19  
20  ; Declare tenant mnemonic  
21  declare hiTenant          = vc with protect, constant("cernerdemo")
```

```
57  endif  
58  
59  ; Return JSON structure containing the Patient ID  
60  call ReturnRecordAsJSON(OUTREC)  
61  call echorecord(OUTREC)  
62  
63  else
```

Springboard Example

Custom CCL: Returning Output

```
86 subroutine ReturnRecordAsJSON(record_data)
87
88     declare originalMaxStringLength = i4 with noconstant(0)
89     declare newMaxStringLength = i4 with noconstant(0)
90
91     ; Declares a temporary structure to hold the response data as a string
92     record tempJson(
93         1 val = gvc
94     )
95
96     ; Returns the record data as a json string with the record
97     ; name 'record_data' as the top-level key.
98     set tempJson->val = cnvtrectojson(record_data)
99
100    ; Set to the current maximum string length
101    set originalMaxStringLength = curmaxvarlen
102
```

Springboard Example

Custom CCL: Returning Output

```
102  
103 ; Set to the length of the record converted to json, plus a buffer  
104 set newMaxStringLength = textlen(tempJson->val) + 1000  
105  
106 ; If needed, adjust the max string length so that the entire  
107 ; json response can be returned  
108 if (newMaxStringLength > originalMaxStringLength)  
| set modify maxvarlen newMaxStringLength  
109 endif  
110  
111  
112 set _Memory_Reply_String = tempJson->val  
113  
114 ; Reset the max string length  
115 if(newMaxStringLength > originalMaxStringLength)  
| set modify maxvarlen originalMaxStringLength  
116 endif  
117  
118
```

Springboard Example

Custom MPage Component: patientAllergies.js

```
8  const patientAllergies = component => new Promise((resolve, reject) => {
9    const scriptRequest = new FusionComponentScriptRequest();
10   scriptRequest.setName("Patient Allergies Request");
11   scriptRequest.setArtifactInfo({
12     artifactId: "MPagesFusionCustomComponent",
13     functionName: "patientAllergies"
14   });
15
16   scriptRequest.setProgramName("chc_hi_patient_allergies:GROUP1");
17   scriptRequest.setParameterArray([
18     [
19       "^\u00c9MINE^\u00c2",
20       "^\u00c21424e81d-8cea-4d6b-b140-d6630b684a58^\u00c2",
21       "^\u00c2002c0901-e16f-464c-bfb0-44f469f5780a^\u00c2"
22     ]
23   ]);
24   scriptRequest.setResponseHandler((reply) => {
25     if (reply.getStatus() !== "F") {
26       resolve(reply);
27     } else {
28       reject(reply);
29     }
30   });
31   scriptRequest.setComponent(component);
32   scriptRequest.performRequest();
33 });


```

Springboard Example

Custom MPage Component: MPagesFusionCustomComponent.js

```
111     retrieveComponentData() {
112         patientAllergies(this)
113             .then((reply) => {
114                 if (reply.getStatus() === "S") {
115                     this.setStatus(STATUS_TYPES.SUCCESS);
116                     this.renderComponent(reply.getResponse());
117                 } else {
118                     this.setStatus(STATUS_TYPES.NO_DATA);
119                     this.finalizeComponent();
120                 }
121             })
122             .catch((reply) => {
123                 this.setStatus(STATUS_TYPES.ERROR);
124                 this.finalizeComponent();
125             });
126     }
127 }
```

Springboard Example

Custom *MPage* Component: *MPagesFusionCustomComponentBody.js*

```
29  const createResultRows = (results) => {
30    return results.map((result, index) => {
31      return {
32        meta: result,
33        key: index,
34        data: [
35          {
36            display: result.CODE
37          },
38          {
39            display: result.CRITICALITY
40          }
41        ,
42        {
43          display: formatDateDisplay(result ASSERTED_ON)
44        }
45      ];
46    });
47  );
48};
```

Springboard Example

Custom MPage Component: MPagesFusionCustomComponentDetailPanel

```
21 const createActivityDetailsSection = (resultObj) => {
22   return [
23     {
24       id: "activityDetailSection",
25       rows: [
26         [
27           {
28             label: "Allergy Status",
29             text: resultObj.STATUS,
30             span: 6
31           }
32         ]
33       ]
34     }
35   ];
36 };
37
```

Springboard Example

Custom MPage Component

TEST, AVERY
DOB: 03/16/1987 Sex: Male MRN: -- FIN: --

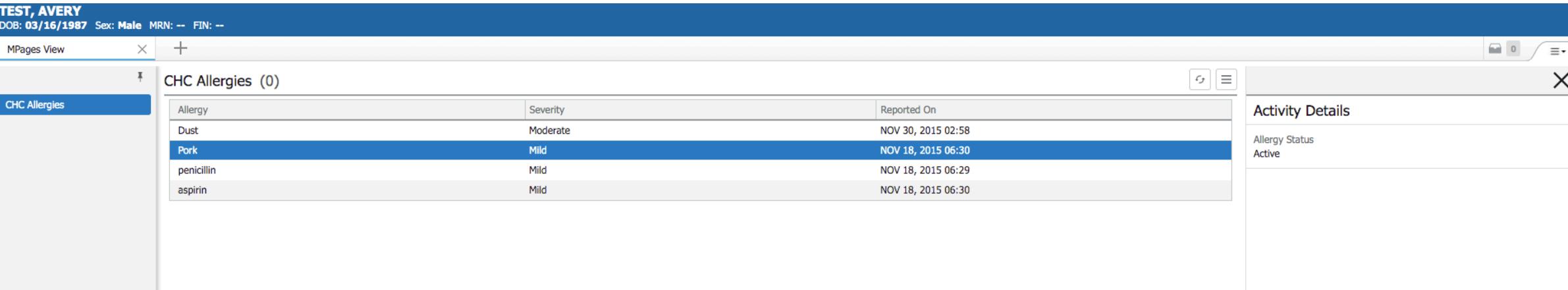
MPages View X +

CHC Allergies (0)

Allergy	Severity	Reported On
Dust	Moderate	NOV 30, 2015 02:58
Pork	Mild	NOV 18, 2015 06:30
penicillin	Mild	NOV 18, 2015 06:29
aspirin	Mild	NOV 18, 2015 06:30

Activity Details

Allergy Status
Active



Conclusion

Additional Resources

MPages

- [MPages End User Community uCern Group](#)
- [MPages Solution Services uCern Group](#)
- [MPages Technical Development uCern Group](#)
- [MPages Fusion Documentation](#)
- [Overview of Mpages Gaia](#)
- [Install Mpages Gaia](#)

Additional Resources

HealthIntent Open APIs

- [HealthIntent Developer Portal](#)
- [Understand HealthIntent API CCL Requests Wiki](#)
- [Lab Content on Github](#)

Reach out to me at patrick.elam@cerner.com

Live Demo?

NOW/NEXT

Thank you!

NOW / NEXT

Template

Subtitle

Text goes here and is Arial Normal 18pt

 Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.