SMART on FHIR

What is SMART on FHIR®?

It is a standardization of the approach using technology and open standards that allows Clinical Systems (eg: EMR) to integrate and run external applications that can interpret, render and visualize in-house data in a secure and replaceable way; this approach supports enabling patients, doctors and healthcare providers to improve clinical care and overall public health by using innovative new solutions integrated in their current application framework. This approach also invites innovation in the health care space as it allows for apps to be substitutable within their capabilities.

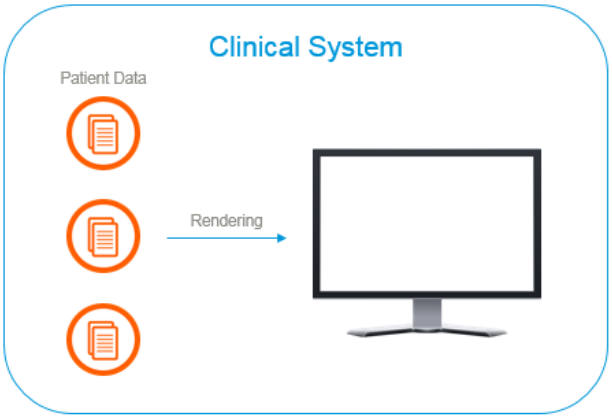
Leveraging SMART, allows Clinical Systems and application vendors to be compatible and re-usable by reducing the cost and complexity of the integration between the app and the Clinical System. Clinical Systems can therefore add new capabilities and functionalities without having to adhere to rigorous change request processes.

Online resources

- SMART on FHIR: a standards-based, interoperable apps platform for electronic health records (White Paper)
- SMART app gallery
- Balloted request SMART launch sequence
- An app platform for health care
- Technical Documentation
- Zulip Stream

Business case

Concept of an EMR



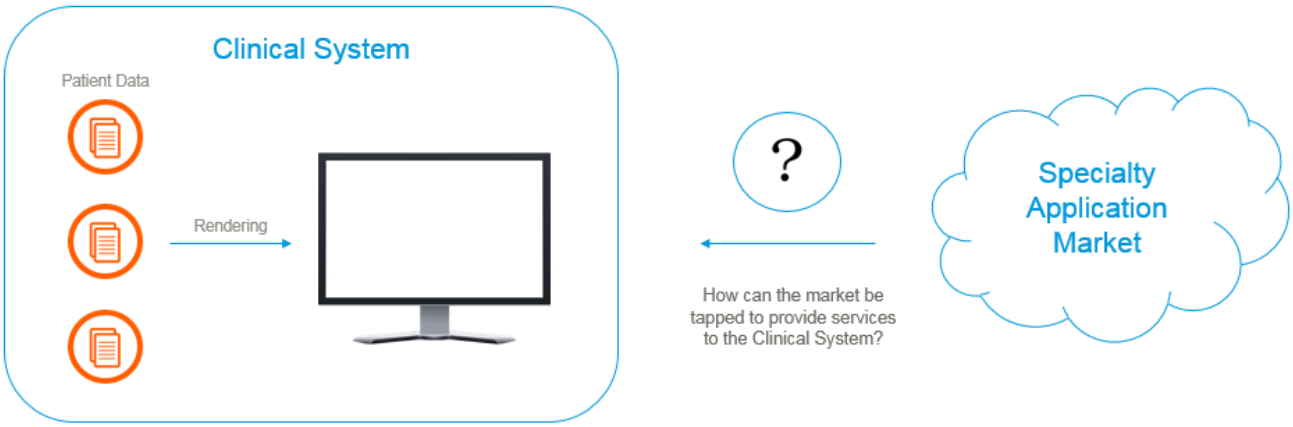
A Clinical System is capable of interpreting and presenting Patient data in a meaningful way, however, it may lack advanced clinical processing abilities or data analysis.

A Clinical System (e.g. EMR, HIS) acts a warehouse for large volumes of health data and is capable of interpreting and presenting this

information. Beyond its primary design however, it may lack advanced information processing abilities or rendering the information in innovative, new ways.

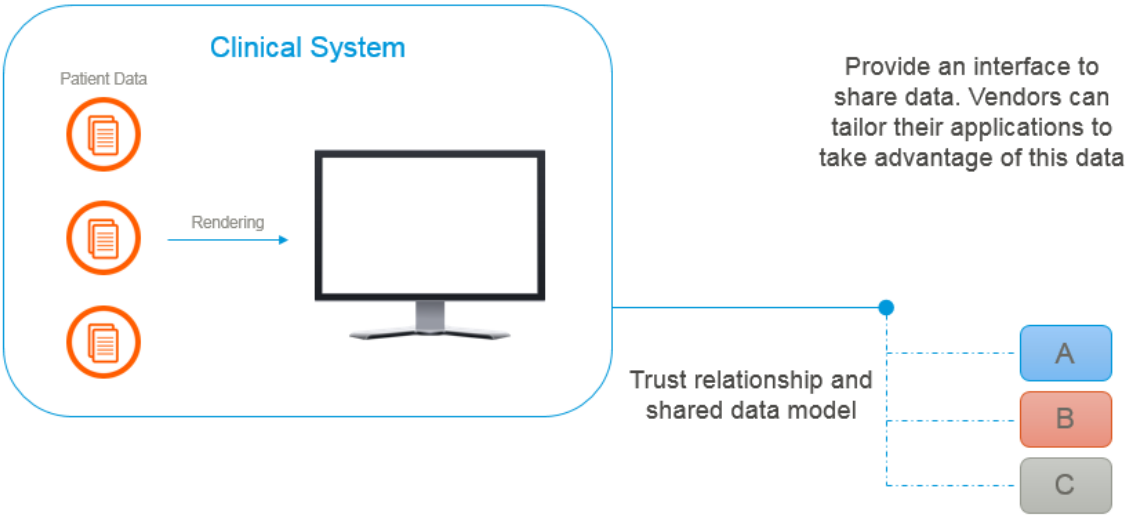
Improvement to these systems would often require significant user-interface and various back-end changes to accommodate any new features.

Adding Specialty Capabilities



Assuming that there was market innovation, what would be the approach to tap into those opportunities by reusing and extending existing systems?

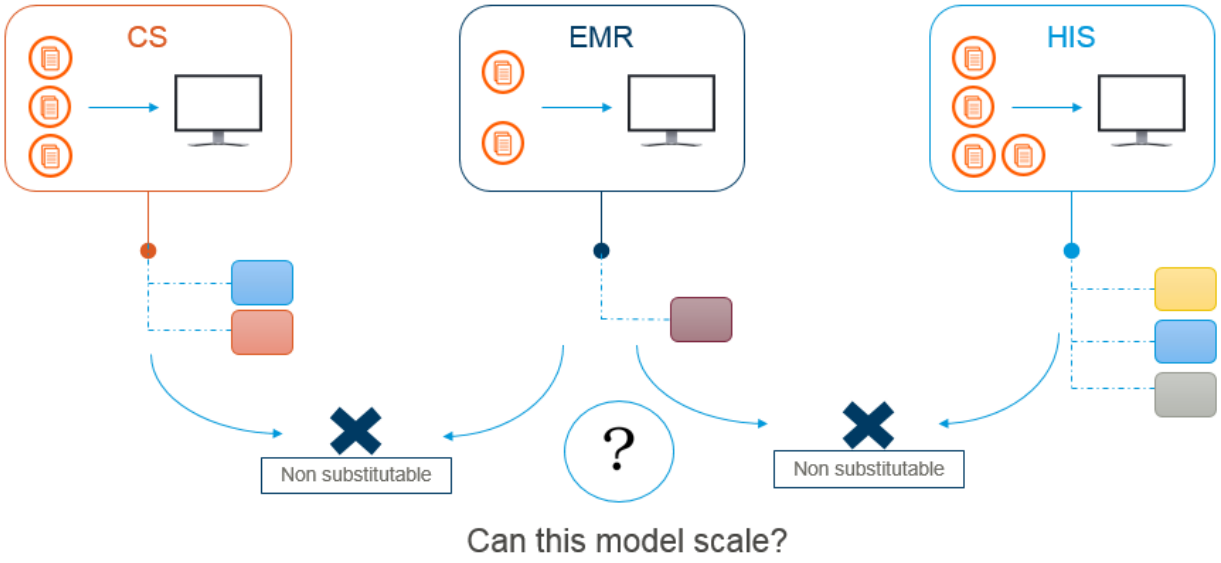
Collaboration Paradigm



One way to extend capabilities is for Clinical System vendors to expose interfaces that app developers could conform to.

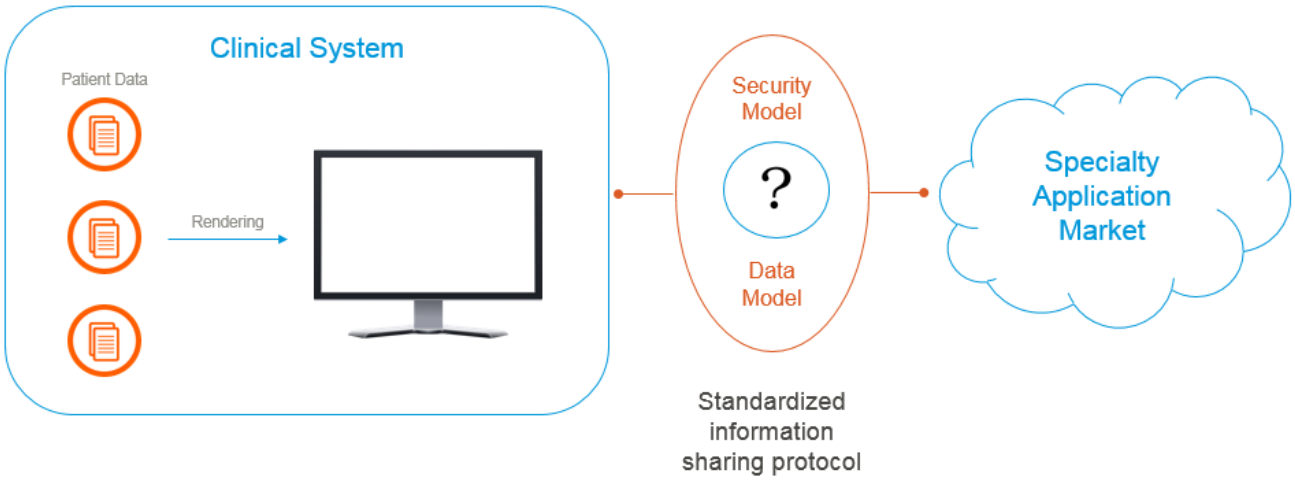
This would have to be realized through a trust relationship and a data model that the app vendor would need to understand.

Scaleability Considerations



How would this particular model scale? Multiple Clinical System would have different interfaces and app makers would have to write a logically similar app multiple times. You will also notice that apps are not easily substituted for one another using this approach.

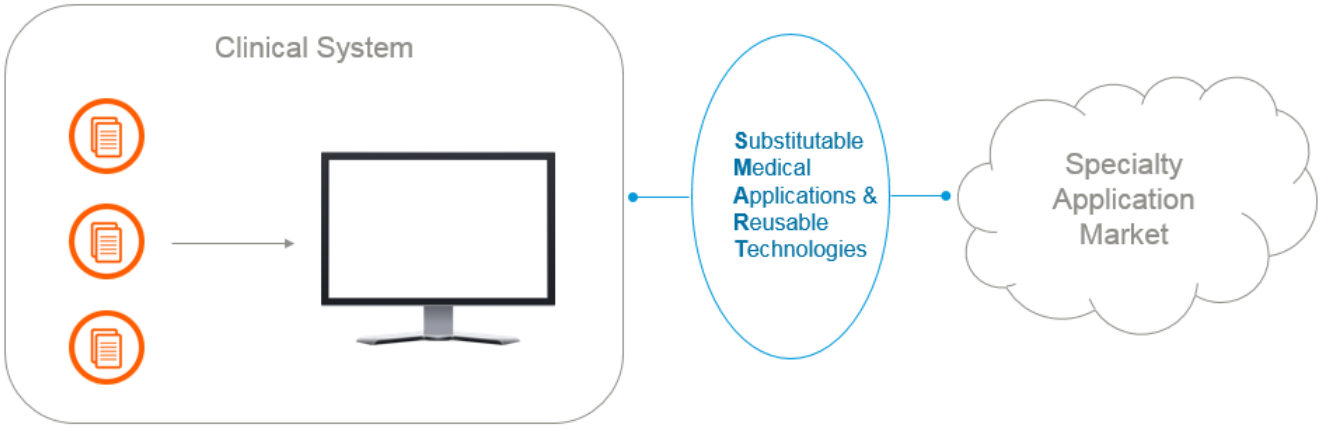
A more scaleable approach



The universal concept of open APIs where a Trust Relationship/Security Model and a common data model that scales very well.

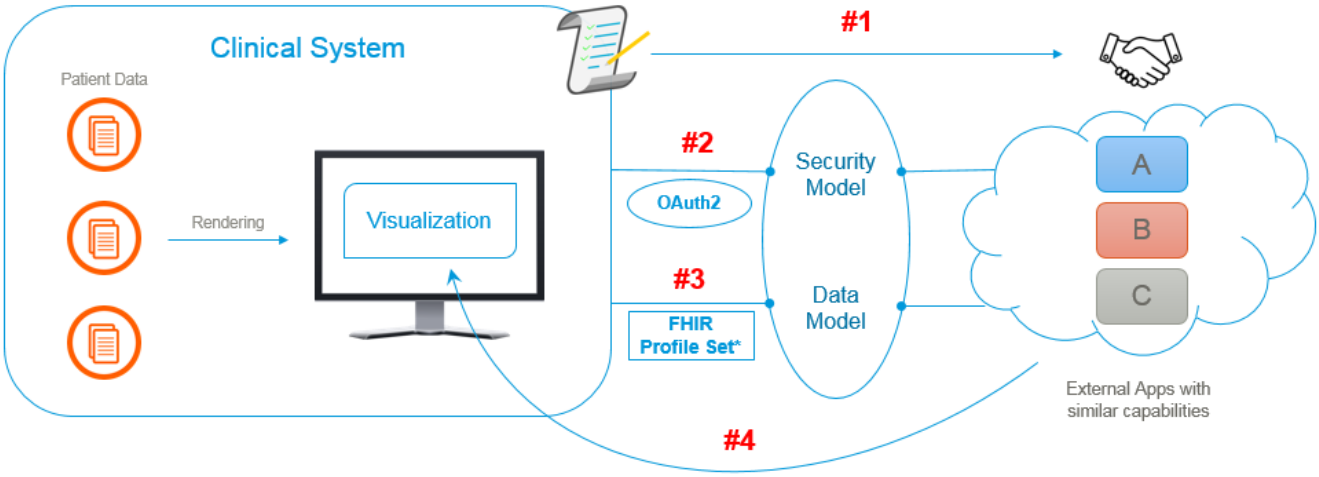
This concept is not new. It has been employed before with RPCs, implementing SOA, etc. so why are we talking about it again? Because there is a game changer rising through the industry - FHIR - it employs an already defined API - REST, the language of the internet and it accommodates security models readily. There still is too much variability. What promises to change that is SMART.

The SMART Approach



This is where the SMART specification comes into play. Substitutable Medical Applications and Reusable Technologies sets its sight on standardizing on how to use trust/security and data (using FHIR) to meaningfully extend data sharing and processing in such a way that web enabled applications can evolve by leaps and bounds using very minimal changes to their architecture.

SMART - Logical Architecture of Legacy Clinical Systems



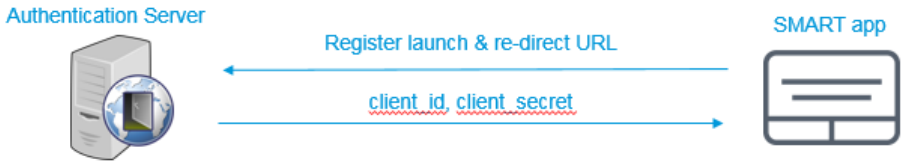
The SMART open standard consists of 3 basic components that enable a Clinical System and SMART Application to interact with one another.

1. The trust relationship - usually set up as part of a Service Level Agreement (SLA) that establishes a relationship between the Clinical System and App Vendor
2. A security model that is realized using the OAuth 2.0 specification
3. A standard data model realized using FHIR and a common profiled baseline

When the concept of SMART was being ruminated by it's innovators, it was originally called SMART classic and it had it's own data model which was described using the Resource Description Framework.

As FHIR was gaining popularity and traction, the innovators decided to switch the RDF for the FHIR data model.

Establishing a Trust Relationship



As part of the OAuth 2.0 specification, a new application must be registered win the service (Ex: Clinical System) that requires it's integration.

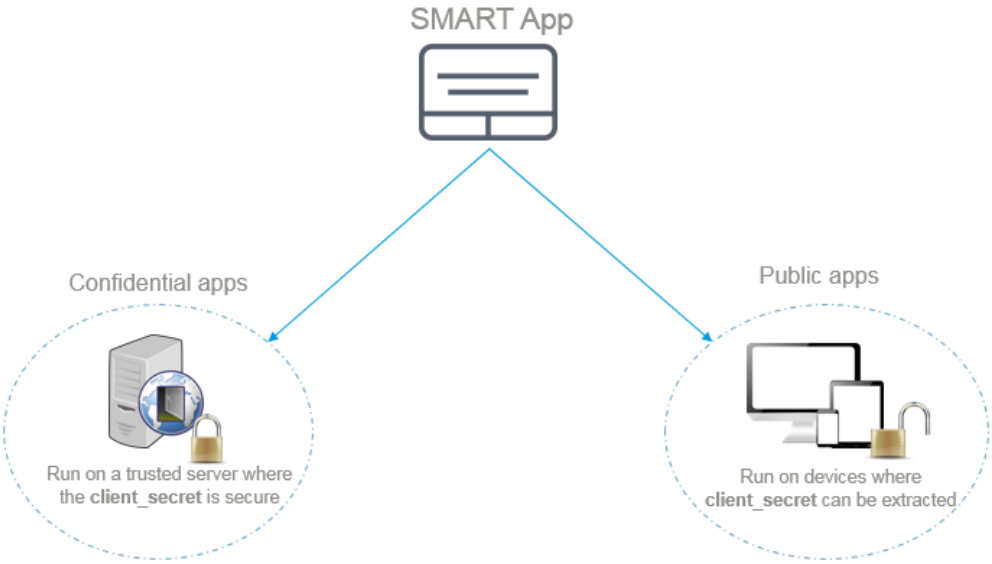
Registration includes basic information such as the application name, website, logo and in addition, a re-direct and launch URL.

In exchange for this, the service will provide the application with a "client_id" and "client_secret" which will help identify the authenticity of the the application during the registration process.

This "client_secret" must be kept confidential. In case the application (such as a single page javascript app, android app etc.) cannot keep this confidential, the "client_secret" must not be used

and a secret must not be issued to the application in the first place. For more information of the OAuth2.0 specification please visit - https://oauth.net/2/

SMART Application Profiles

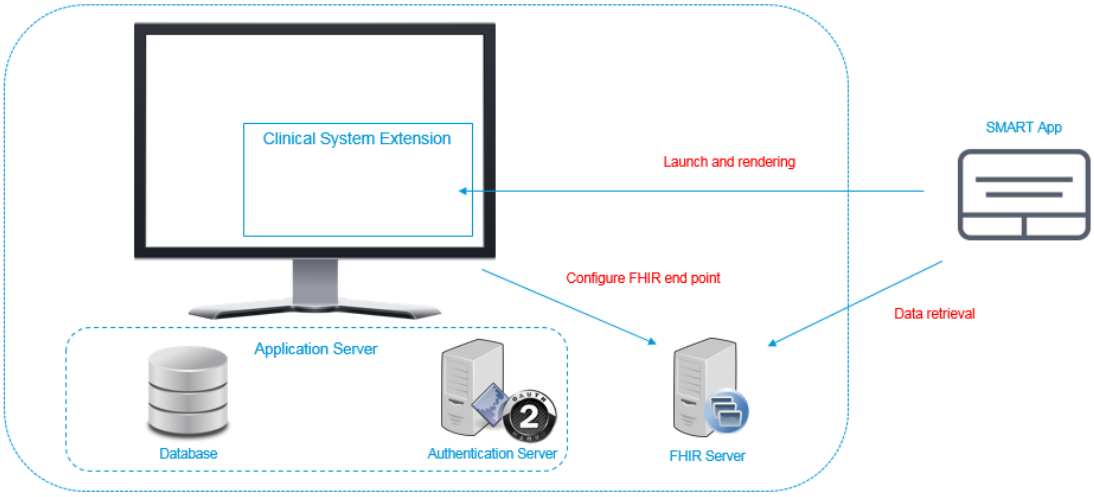


SMART supports the idea of confidential and public app profiles. We have seen that as part of the OAuth 2.0 specification the application gets a "client_id" and "client_secret" after it registers with the Clinical System.

Confidential app Profile	Public app Profile
Usually run on a trusted server that has some business logic such as PHP, .NET, Java etc	Usually run on end-user mobile devices such as Android, iOS, Windows etc
The "client_secret" that they have embedded can be kept secure	The "client_secret" if embedded is prone to being extracted.

For more information on the types of logical architectures supported by SMART, please see - http://argonautwiki.hl7.org/images/4/4c/Argonaut_UseCasesV1.pdf

Logical Architecture - Legacy Clinical Systems



Some common components of existing Clinical Systems:

1. An application server
2. An authorization server
3. A database

These components can be part of a single entity or they can be distributed across the network.

In-order to be SMART compliant, the Clinical System will have to:

1. Have a Trusted Agreement with the application vendor.
2. Serve up FHIR content and this can be achieved by either using a FHIR Server or a implementing a FHIR Facade that is able to serve FHIR content directly from the legacy systems database - note however that the latter requires to act as a compliant FHIR endpoint (i.e. needs to expose and live up to a stated CapabilityStatement).
3. Have an OAuth2 capable authorization server implemented and
4. The Clinical System will need to embed an iframe to render the contents of the SMART Application

SMART compliant CapabilityStatement

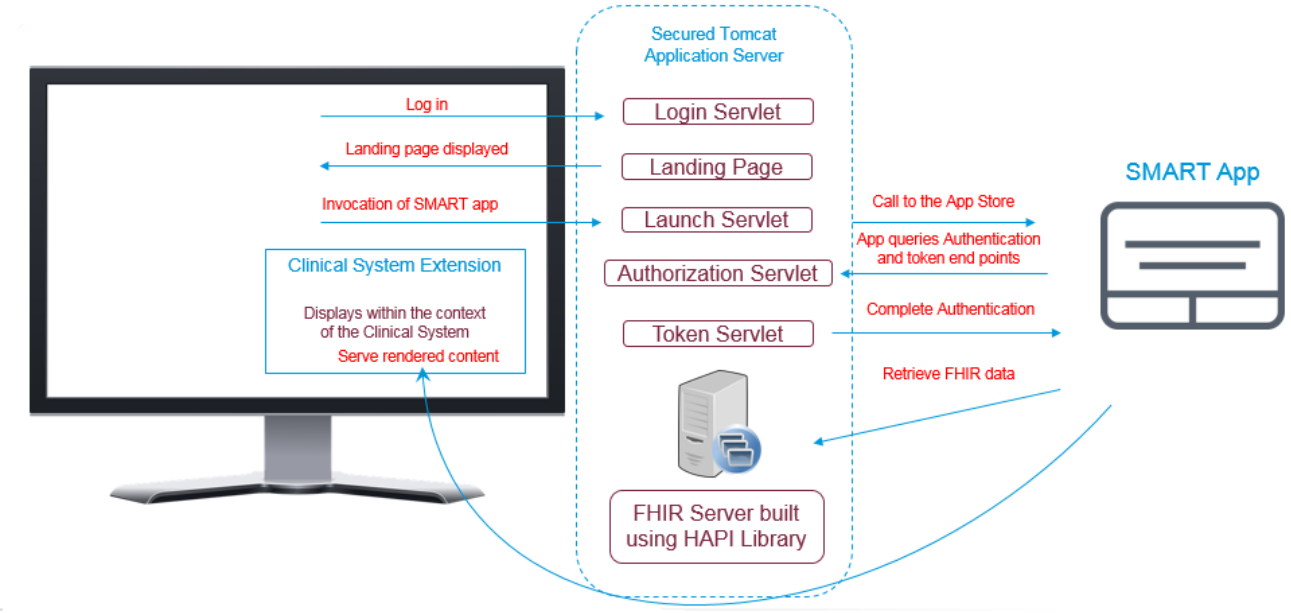
In order for the FHIR endpoint to be SMART compliant, it must have the following extension:

```
{
  "resourceType": "CapabilityStatement",
...
  "rest": [{
   ...
      "security": {
        "service": [
          {
            "coding": [
              {
                "system": "http://hl7.org/fhir/restful-security-service",
                "code": "SMART-on-FHIR"
              }
            ],
            "text": "OAuth2 using SMART-on-FHIR profile (see http://docs.smarthealthit.
          }
        ],
        "extension": [{
          "url": "http://fhir-registry.smarthealthit.org/StructureDefinition/oauth-uris
          "extension": [{
            "url": "token",
            "valueUri": "https://my-server.org/token"
          },{
            "url": "authorize",
            "valueUri": "https://my-server.org/authorize"
          },{
            "url": "manage",
            "valueUri": "https://my-server.org/authorizations/manage"
          }]
```

Extension	Required	Description
authorize	yes	uri indicating the authorization endpoint
token	yes	uri indicating the token endpoint
register	optional	uri indicating the dynamic registration endpoint
manage	optional	uri indicating authorization amangement

For more information about the CapabilityStatement resource, please see - http://www.hl7.org/fhir/smart-app-launch/capability-statement/

Demo Implementation

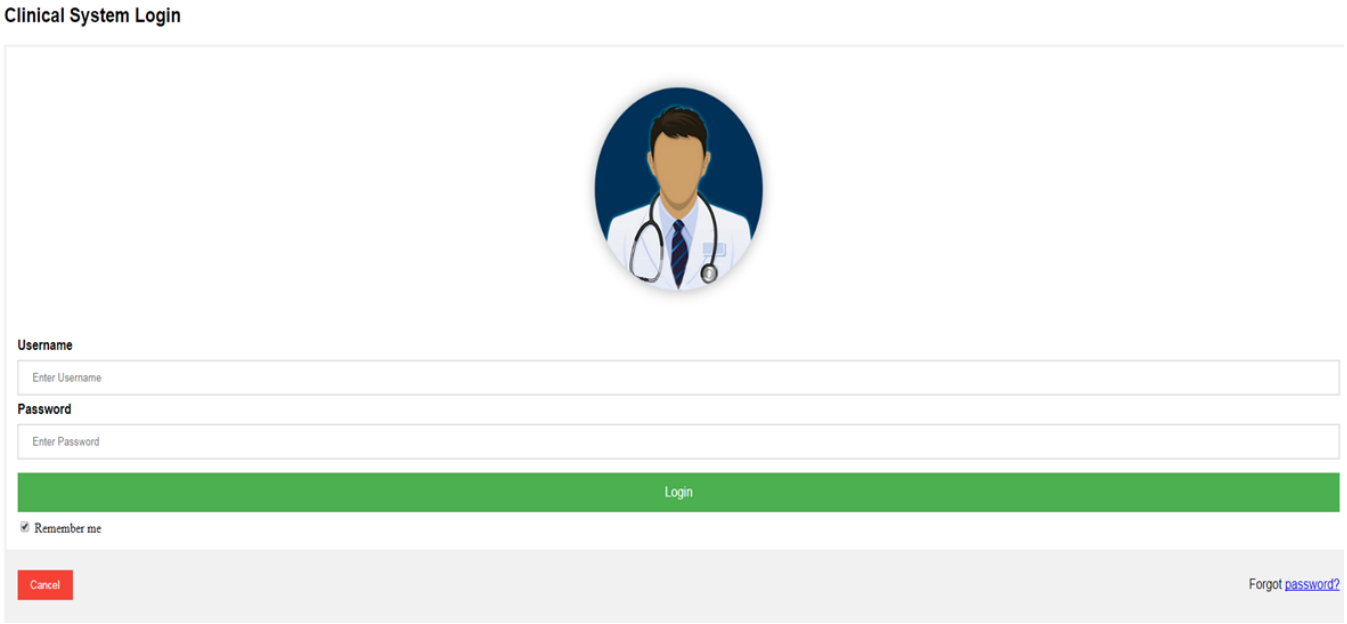


For the demonstration, we will simulate an existing Clinical System which is implemented through the use of a web application hosted on a secured Tomcat server. A typical workflow would entail a medical practitioner such as a doctor logging into the system which is handled by the

login servlet. The practitioner lands on to a page displaying the relevant patient context and this handled by the launch servlet. At this point the practitioner can launch the SMART app hosted in the SMART app gallery (https://apps.smarthealthit.org/) and this is handled by the launch servlet.

The SMART app will then query the FHIR endpoint (built using the HAPI library) to gain access to the health data. Once the data is processed by the app, the rendered result is shown in the embedded iframe of the clinical system.

User logins:



User is presented with Patient Context:

Patient Profile - John Smith

Name (Full name)	John Smith
Date Of Birth	1990-12-08
Gender	Male Female Other
Marital Status:	Married Unmarried
Permanent Address	CA 94117 1524 Haight St
Primary Occupation	Musician
Phone number	773-888-1212 Secondary Phone number
Email Address	heyjoe@aol.net

User wishes to view growth chart:

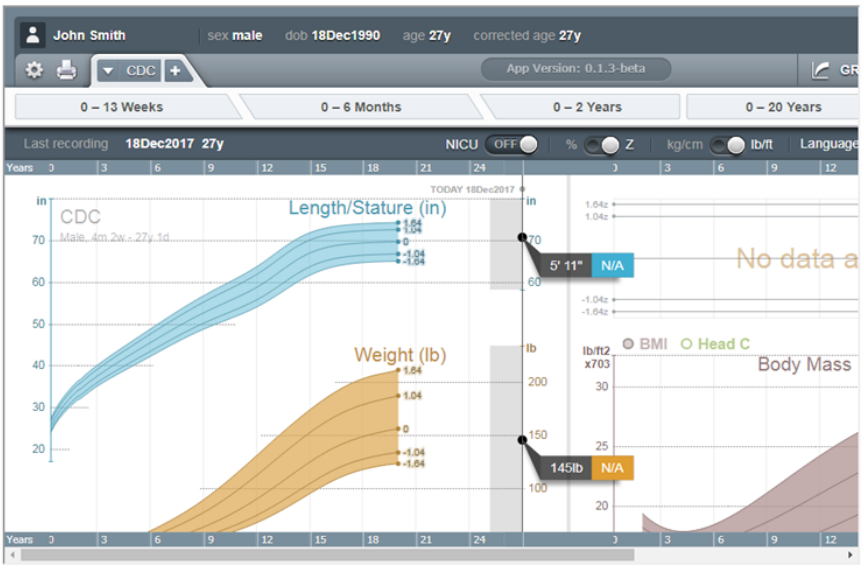
Height & Weight - John Smith

Age	Weight (kg)	Height (cm)
11	36.2874	145.32
15	38.13	147.372
18	42.56	155.372
20	50.12	160.78
22	60.35	165.75
25	61.34	178.12
27	65.78	179.56

View Growth Chart

SMART app is launched and rendered in iframe within the current context:

Height & Weight - John Smith



View Table