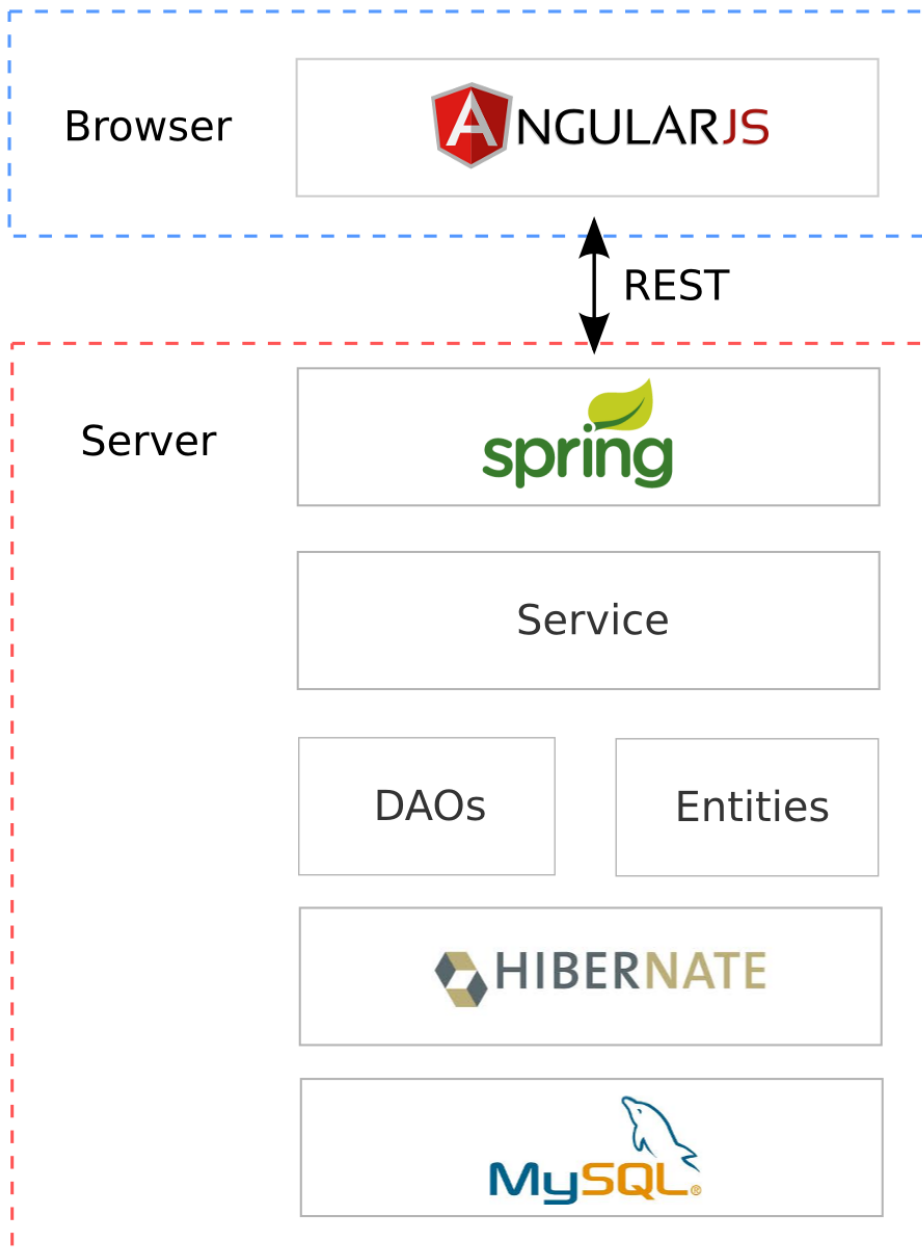


Tech Stack and Technologies Used

Link to <https://github.com/OpenMRS-Australia/openmrs-cpm/wiki/Resources>

The core is built on Java but current development has leveraged REST services to enable client side development in [AngularJS](#) as show in the diagram below



Links to technologies used

- [AngularJS](#) (Training material: <http://docs.angularjs.org/tutorial/>)
- [Spring](#)
- [Hibernate](#)

Build process uses [Gradle](#) (previously was [Maven](#))

BDD via [Cucumber](#)

References

Spring REST

- Spring RestTemplate: <http://www.springframework.net/rest/doc-latest/reference/html/resttemplate.html> / <http://spring.io/blog/2009/03/27/rest-in-spring-3-resttemplate/>
- Consuming a RESTful Web Service with AngularJS: <https://spring.io/guides/gs/consuming-rest-angularjs/>
- Handling exceptions: <http://spring.io/blog/2013/11/01/exception-handling-in-spring-mvc>

Angular

- Services: https://docs.angularjs.org/tutorial/step_11 / [https://docs.angularjs.org/api/ngResource/service/\\$resource](https://docs.angularjs.org/api/ngResource/service/$resource)
- Filters: <https://docs.angularjs.org/guide/filter> / https://docs.angularjs.org/tutorial/step_09
- Directives: <http://www.sitepoint.com/practical-guide-angularjs-directives/> / <https://docs.angularjs.org/guide/directive>
- Routing & HTML partials: https://docs.angularjs.org/tutorial/step_07

RequireJS

Testing

- Cucumber: <http://cukes.info/> / <http://www.ibm.com/developerworks/opensource/library/a-automating-ria/index.html>
- Page Objects: <https://code.google.com/p/selenium/wiki/PageObjects>
- Timing Races: <http://www.bizalgo.com/2012/01/14/timing-races-selenium-2-implicit-waits-explicit-waits/>