Revision history

|  |  |  |
| --- | --- | --- |
| Revision # | Date of Revision | Changes Made |
| 1.0 | NA | First Release |
|  |  |  |

Reference documents

|  |  |  |
| --- | --- | --- |
| S.No | Document Title | Date |
| 1 |  | 05 Jun 2017 |

Glossary

| **Terms** | **Description** |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Table of contents

[1 Predix 6](#_Toc489465976)

[2 Back End 6](#_Toc489465977)

[3 Front End 6](#_Toc489465978)

[4 Deployment 6](#_Toc489465979)

[5 Dev Environment Setup 6](#_Toc489465980)

[6 Project management 6](#_Toc489465981)

THIS PAGE IS INTENTIONALLY LEFT BLANK

# Predix

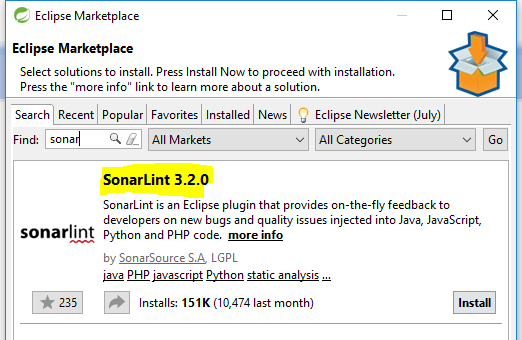
TBD

# Back End

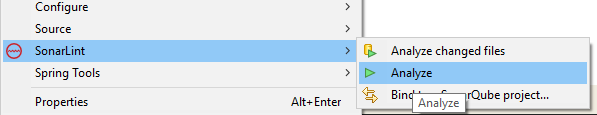
## Learning

1. Plugged in SonarLint for code quality check in STS

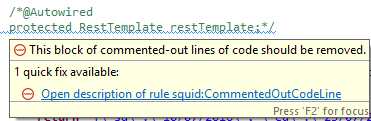
* Go to Eclipse Marketplace, search for sonar and install latest SonarLint.



* Once installation completes, restart Eclipse/STS and you will be able to see SonarLint on right-click of the project.



* After completely analyzed your project by SonarLint, you will be able to see Sonar defects and fix them while in development phase. Ref. below snapshot of issue captured.



## Issue Resolution

1. Identifying the HTTP port number which should connect Consumer and Provider services through Register Service.

**Resolution:**

* No need to explicitly declare any port number in Consumer service. Although port number is needed in Provider and Register services.
* Define default constructor for all the Application Beans in Consumer microservices. No need to define default constructor for application beans in Provider services.
* Declare @LoadBalanced annotation for restTemplate() method in Consumer service bootstrap class.
* Declare dependence for Ribbon in Consumer’s pom.xml

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-ribbon</artifactId>

</dependency>

# Front End

## Learning

1. Before closing the browser we have to trigger some event to warn user about the unsaved Changes(if any in application) , same thing we need it for invalidating the session also by calling the REST API.
2. We have faced issue in implementing and importing the Highcharts library with  our application, which we have resolved it by importing it with alternate way.
3. Angular – 2 command to make component

* ng g component nameComponent --module=app.module.ts(specific to a module)

1. Lifecycle hook of Angular2
   1. constructor
   2. ngOnChanges − When the value of a data bound property changes, then this method is called.
   3. ngOnInit − This is called whenever the initialization of the directive/component after Angular first displays the data-bound properties happens.
   4. ngDoCheck − This is for the detection and to act on changes that Angular can't or won't detect on its own.
   5. ngAfterContentInit − This is called in response after Angular projects external content into the component's view.
   6. ngAfterContentChecked − This is called in response after Angular checks the content projected into the component.
   7. ngAfterViewInit − This is called in response after Angular initializes the component's views and child views.
   8. ngAfterViewChecked − This is called in response after Angular checks the component's views and child views.
   9. ngOnDestroy − This is the cleanup phase just before Angular destroys the directive/component.

## Issue Resolution

1. We have faced issue while fetching the latitude and longitude from table , we have written the code of map in ngOninit(one of lifecycle hooks which comes into the control while loading of component) so at the time of page rendering map get loaded  and we unable to bind the data in the map which already got loaded

**Resolution:**

* Soltn=> Now we are using the different component for map which is now a child component(map-child component) and by the help of ngOnchanges(it triggered on changes only) we will be able to identify the changes happens in child component

# Deployment

## Learning

## Issue Resolution

# Dev Environment Setup

## Learning

## Issue Resolution

# Project management

## Learning

## Issue Resolution