

MEMORIA TECNICA INTEGRACIÓN DE BITACORA ACCIONES DE USUARIO

Autor:Ing. Juan Carlos García López

19 de mayo de 2020

Contenido

OBJETIVO.....	4
PROPUESTA DE SOLUCIÓN	4
Creación de una cuenta cloud Elastic.....	5
GENERACION SERVICIO PROXY EN EL OSB.....	8
LECCIONES APRENDIDAS.....	9

OBJETIVO

Detallar las estrategias tecnológicas según el marco arquitectónico planteado para la solución de la integración que le dará soporte al control de cambios que son realizados por los usuarios de negocio para la empresa "Automotriz SPS" y de esta forma tener un rastreo de las actividades ejecutadas por los empleados a las reglas de descuento, que se deben aplicar, durante la operación diaria de la empresa.

PROPUESTA DE SOLUCIÓN

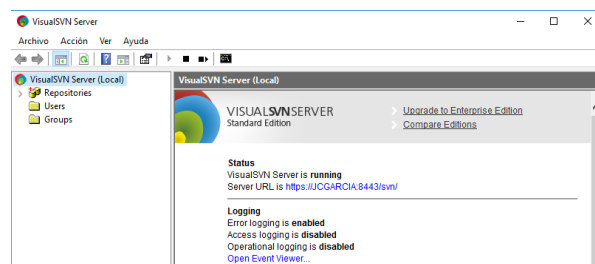
Para la construcción del servicio REST que realizara la inter operatividad con el stack ELK(Elasticsearch, Logstash y Kibana) el cual es un conglomerado de tres productos que ofrece la empresa **elastic**.

Se optó por la construcción del servicio REST "Bitacora" con las siguientes herramientas:

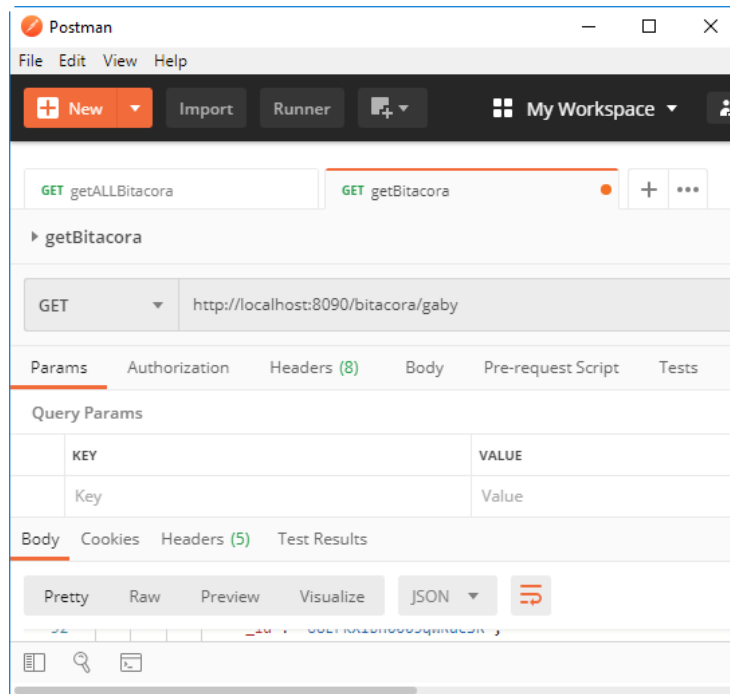
- IDE de desarrollo STS 4(4.6.1.RELEASE)



- Gestión y construcción de la integración: Maven (Apache Maven 3.6.1)
- Se ocupo la versión JDK(1.8.0_211) de java.
- Para la construcción del servicio se usó la biblioteca de spring boot (2.2.2.RELEASE)
- Para la comunicación con Elasticsearch se usó biblioteca elasticsearch(7.6.2) y elasticsearch-rest-client(7.6.2), optando por la comunicación nativa tipo REST.
- Para el control de versiones interno durante el desarrollo se optó por utilizar VisualSVN(2.5.2)



- Como cliente REST se decidió ocupar Postman



- Para generar el contenedor de la integración se ocupó **Docker** con los siguientes para parámetros de configuración en el archivo Dockerfile

```
FROM openjdk:8-jdk-slim
COPY "target/original-bitacora-rest-service-1.0.0.jar" "SPSbitacoraService-1.0.0.jar"
EXPOSE 8090
ENTRYPOINT ["java", "-jar", "SPSbitacoraService-1.0.0.jar"]
```

Creación de una cuenta cloud Elastic

Para este desarrollo se creó una cuenta en el portal Elastic Cloud:

<https://www.elastic.co/es/cloud/>

Siguiendo las especificaciones del documento de requerimiento con los siguientes datos:

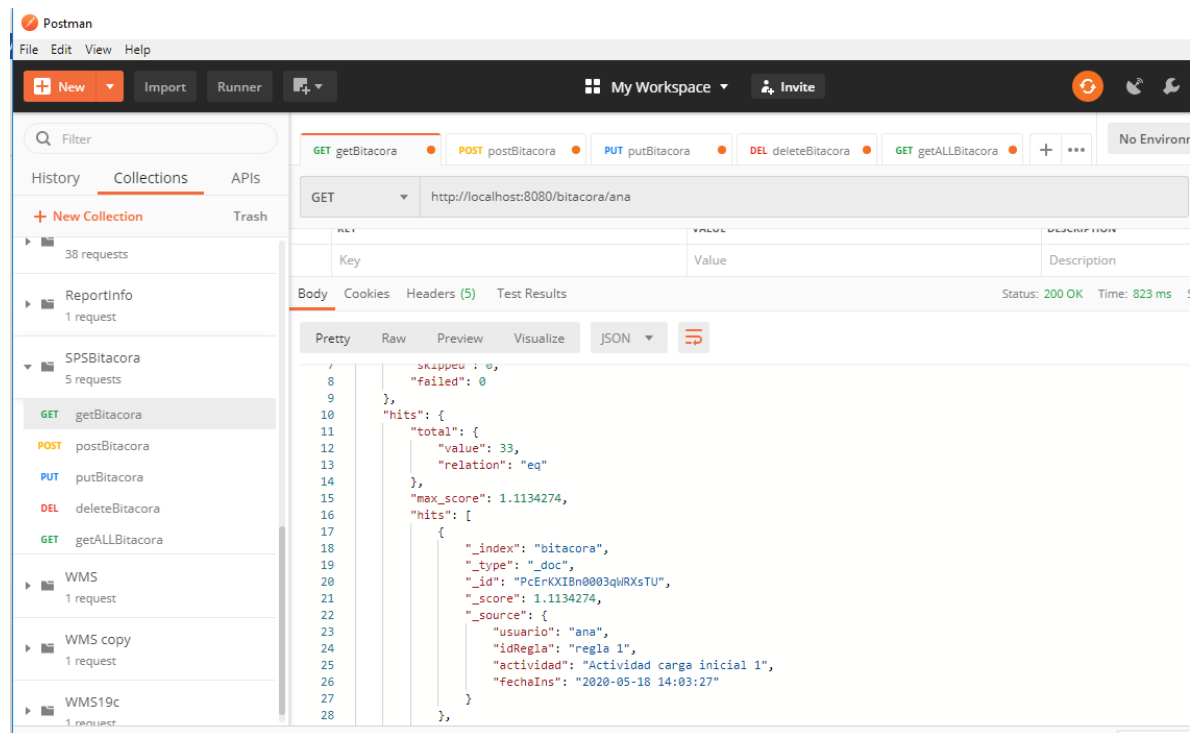
- Nombre del deployment: sps_practica
- Plataforma: Amazon Web Service
- Región: US East (N. Virginia)
- Elastic stack version: más reciente
- Optimize your deployment: I/O Optimized

1. Despliega el API REST en tu máquina.



```
{
  "took": 0,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 101,
      "relation": "eq"
    },
    "max_score": 1.0,
    "hits": [
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "ScipKXIBTyXBraH40Qhr",
        "_score": 1.0,
        "_source": {
          "usuario": "Test",
          "idRegla": "regla 1",
          "actividad": "Actividad carga inicial",
          "fechaIns": "2020-05-16 00:00:00"
        }
      },
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "PcErKXIBn0003qWRXsTU",
        "_score": 1.0,
        "_source": {
          "usuario": "ana",
          "idRegla": "regla 1",
          "actividad": "Actividad carga inicial 1",
          "fechaIns": "2020-05-18 14:03:27"
        }
      },
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "PsErKXIBn0003qWRX8RD",
        "_score": 1.0,
        "_source": {
          "usuario": "ana",
          "idRegla": "regla 2",
          "actividad": "Actividad carga inicial 2",
          "fechaIns": "2020-05-18 14:03:28"
        }
      },
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "P8ErKXIBn0003qWRX8SF",
        "_score": 1.0,
        "_source": {
          "usuario": "ana",
          "idRegla": "regla 3",
          "actividad": "Actividad carga inicial 3",
          "fechaIns": "2020-05-18 14:03:28"
        }
      },
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "QMErKXIBn0003qWRX8T",
        "_score": 1.0,
        "_source": {
          "usuario": "ana",
          "idRegla": "regla 4",
          "actividad": "Actividad carga inicial 4",
          "fechaIns": "2020-05-18 14:03:28"
        }
      },
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "aMirKXIBTyXBraH4YAhD",
        "_score": 1.0,
        "_source": {
          "usuario": "ana",
          "idRegla": "regla 5",
          "actividad": "Actividad carga inicial 5",
          "fechaIns": "2020-05-18 14:03:28"
        }
      },
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "QcErKXIBn0003qWRYMSP",
        "_score": 1.0,
        "_source": {
          "usuario": "ana",
          "idRegla": "regla 6",
          "actividad": "Actividad carga inicial 6",
          "fechaIns": "2020-05-18 14:03:28"
        }
      },
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "aclrKXIBTyXBraH4YAjB",
        "_score": 1.0,
        "_source": {
          "usuario": "ana",
          "idRegla": "regla 7",
          "actividad": "Actividad carga inicial 7",
          "fechaIns": "2020-05-18 14:03:28"
        }
      },
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "aslrKXIBTyXBraH4YQgs",
        "_score": 1.0,
        "_source": {
          "usuario": "ana",
          "idRegla": "regla 8",
          "actividad": "Actividad carga inicial 8",
          "fechaIns": "2020-05-18 14:03:28"
        }
      },
      {
        "_index": "bitacora",
        "_type": "_doc",
        "_id": "a8lrKXIBTyXBraH4YQIA",
        "_score": 1.0,
        "_source": {
          "usuario": "ana",
          "idRegla": "regla 9",
          "actividad": "Actividad carga inicial 9",
          "fechaIns": "2020-05-18 14:03:28"
        }
      }
    ]
  }
}
```

2. Realiza pruebas unitarias del API con el cliente REST que más te guste.



3. Conteneriza la aplicación y levanta el contenedor. Expón tu servicio por el puerto 8090.

```

docker-bitacora-rest-service1 latest
Running

Logs
Inspect
Stats

:: Spring Boot :: (v2.2.2.RELEASE)
2020-05-18 23:05:44.380 INFO 1 --- [main] mx.com.sps.RestServiceApplication : Starting RestServiceApplication v1.0.0 on 245d90f8c4id with PID 1 (/usr/bitacora-service-1.0.0.jar started by root in /)
2020-05-18 23:05:44.496 INFO 1 --- [main] mx.com.sps.RestServiceApplication : No active profile set, falling back to default profiles: default
2020-05-18 23:05:44.893 INFO 1 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2020-05-18 23:05:46.914 INFO 1 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2020-05-18 23:05:46.931 INFO 1 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.29]
2020-05-18 23:05:47.014 INFO 1 --- [main] o.s.c.o.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2020-05-18 23:05:47.014 INFO 1 --- [main] o.s.web.context.ContextLoader : Root WebApplicationContext: initialization completed in 2440 ms
2020-05-18 23:05:47.494 INFO 1 --- [main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationTaskExecutor'
2020-05-18 23:05:48.472 INFO 1 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path '/'
2020-05-18 23:05:48.479 INFO 1 --- [main] mx.com.sps.RestServiceApplication : Started RestServiceApplication in 5.189 seconds (JVM running for 6.063)
2020-05-18 23:05:59.524 INFO 1 --- [nio-8080-exec-1] o.s.c.o.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'
2020-05-18 23:05:59.524 INFO 1 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'
2020-05-18 23:05:59.544 INFO 1 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Completed initialization in 20 ms

```

```

{
  "took":0,"timed_out":false,"shards":{"total":1,"successful":1,"skipped":0,"failed":0},"hits":{"total":{"value":104,"relation":"eq"},"max_score":1.0,"hits":
  [
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "ScIpKXI8TyXBrAh40Qhr",
      "_score": 1.0,
      "_source": {
        "usuario": "Test",
        "idRegla": "regla ",
        "actividad": "Actividad carga inicial",
        "fechaIns": "2020-05-16 00:00:00"
      }
    },
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "PcErKXI8n0003qWRXsTU",
      "_score": 1.0,
      "_source": {
        "usuario": "ana",
        "idRegla": "regla 1",
        "actividad": "Actividad carga inicial 1",
        "fechaIns": "2020-05-18 14:03:27"
      }
    },
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "PsErKXI8n0003qWRX8RD",
      "_score": 1.0,
      "_source": {
        "usuario": "ana",
        "idRegla": "regla 2",
        "actividad": "Actividad carga inicial 2",
        "fechaIns": "2020-05-18 14:03:28"
      }
    },
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "P8ErKXI8n0003qWRX8Sf",
      "_score": 1.0,
      "_source": {
        "usuario": "ana",
        "idRegla": "regla 3",
        "actividad": "Actividad carga inicial 3",
        "fechaIns": "2020-05-18 14:03:28"
      }
    },
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "QcErKXI8n0003qWRX8T",
      "_score": 1.0,
      "_source": {
        "usuario": "ana",
        "idRegla": "regla 4",
        "actividad": "Actividad carga inicial 4",
        "fechaIns": "2020-05-18 14:03:28"
      }
    },
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "aIrKXI8TyXBrAh4YAH0",
      "_score": 1.0,
      "_source": {
        "usuario": "ana",
        "idRegla": "regla 5",
        "actividad": "Actividad carga inicial 5",
        "fechaIns": "2020-05-18 14:03:28"
      }
    },
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "QcErKXI8n0003qWRX8SP",
      "_score": 1.0,
      "_source": {
        "usuario": "ana",
        "idRegla": "regla 6",
        "actividad": "Actividad carga inicial 6",
        "fechaIns": "2020-05-18 14:03:28"
      }
    },
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "aCIrKXI8TyXBrAh4YAjB",
      "_score": 1.0,
      "_source": {
        "usuario": "ana",
        "idRegla": "regla 7",
        "actividad": "Actividad carga inicial 7",
        "fechaIns": "2020-05-18 14:03:28"
      }
    },
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "aSiRkXI8TyXBrAh4YQgs",
      "_score": 1.0,
      "_source": {
        "usuario": "ana",
        "idRegla": "regla 8",
        "actividad": "Actividad carga inicial 8",
        "fechaIns": "2020-05-18 14:03:28"
      }
    },
    {
      "_index": "bitacora",
      "_type": "_doc",
      "_id": "a8IrKXI8TyXBrAh4VQIA",
      "_score": 1.0,
      "_source": {
        "usuario": "ana",
        "idRegla": "regla 9",
        "actividad": "Actividad carga inicial 9",
        "fechaIns": "2020-05-18 14:03:28"
      }
    }
  ]
}

```

4. Realiza pruebas unitarias del API, ahora corriendo en el contenedor.

```

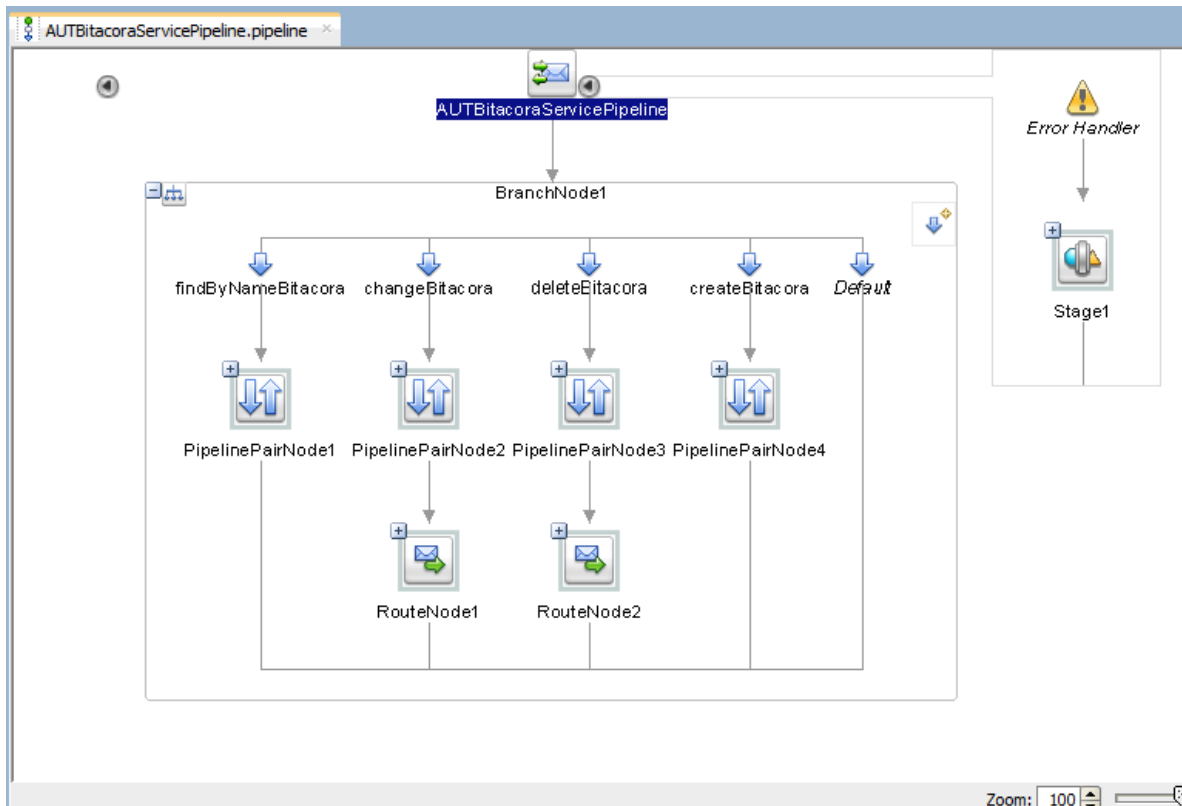
GET getALLBitacora
GET http://localhost:8090/bitacora/
Send

Pretty Raw Preview Visualize JSON
14
15
16     "max_score": 1.0,
17     "hits": [
18       {
19         "_index": "bitacora",
20         "_type": "_doc",
21         "_id": "ScIpKXI8TyXBrAh40Qhr",
22         "_score": 1.0,
23         "_source": {
24           "usuario": "Test",
25           "idRegla": "regla ",
26           "actividad": "Actividad carga inicial",
27           "fechaIns": "2020-05-16 00:00:00"
28         }
29       },
30       {
31         "_index": "bitacora",
32         "_type": "_doc",
33         "_id": "PcErKXI8n0003qWRXsTU",
34         "_score": 1.0,
35         "_source": {
36           "usuario": "ana",
37           "idRegla": "regla 1",
38           "actividad": "Actividad carga inicial 1",
39           "fechaIns": "2020-05-18 14:03:27"
40         }
41       }
42     ]
43   }
44 }

```

GENERACION SERVICIO PROXY EN EL OSB

Se generó un proxy en el OSB que consumiera las operaciones CRUD expuestas por el servicio REST de BITACORA expuesto como micro servicio



Y finalmente se hizo una prueba de la operación Get para obtener la lista de registros del índice “Bitacora” creado en Elasticsearch.

localhost:7101/lwpcfconsole/testdialog.portal?_nfpb=true&_windowLabel=ServiceTestDialogPortlet&ServiceTestDialogPortlet_actionOverride=%2Ftest%2FSe...

Prueba de Pipeline - AUTBitacoraServicePipeline Ayuda

Atrás Cerrar

Documento de Petición

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header>
    <v1:header xmlns:v1="http://spsolutions.com.mx/core/commonTypes/v1/">
      <v1:transactionId>testFind1</v1:transactionId>
    </v1:header>
  </soap:Header>
  <soap:Body>
    <v1:findBitacoraRequest xmlns:v1="http://xmlns.spsolutions.com.mx/aut/BitacoraServices/v1/">
      <v1:usuario>Test</v1:usuario>
    </v1:findBitacoraRequest>
  </soap:Body>
</soap:Envelope>
```

Documento de Respuesta

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header>
    <v1:header xmlns:v1="http://spsolutions.com.mx/core/commonTypes/v1/">
      <v1:transactionId>testFind1</v1:transactionId>
    </v1:header>
  </soap:Header>
  <soap:Body>
    <tns:findBitacoraResponse xmlns:nxsd="http://xmlns.oracle.com/pcbpel/nxsd" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
      xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:inp2="http://spsolutions.com.mx/core/commonTypes/v1/"
      xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://xmlns.spsolutions.com.mx/aut/BitacoraServices/v1/">
      <tns:listBitacora>
        <tns:usuario>Test</tns:usuario>
        <tns:idRegla>regla</tns:idRegla>
        <tns:actividad>Actividad carga inicial</tns:actividad>
        <tns:fechaIns>2020-05-16 00:00:00</tns:fechaIns>
      </tns:listBitacora>
    </tns:findBitacoraResponse>
  </soap:Body>
</soap:Envelope>
```

LECCIONES APRENDIDAS

El presente desarrollo me ayuda a familiarizarme más con el concepto cloud al implementar una interfaz de comunicación con los servicios que provee Elasticsearch, además de comprender mejor el concepto de realizar micro servicios, al exponer mi apiRest en un contenedor Docker.