



Layout de Integração Webservice

Layout de Integração com SIP via Webservices Versão 1.4.03

Sumário

Objetivos	2
Arquitetura.....	2
Versionamento.....	2
Autenticação e Autorização	3
MediaType.....	4
XML.....	4
JSON	4
CSV	4
Ambiente de Testes.....	4
Recursos Disponíveis.....	4



Objetivos

Este layout destina-se a troca de dados entre o SIP e qualquer mecanismo que implemente o protocolo HTTP 1.1 devidamente autenticado, trata-se de uma Web API.

Arquitetura

O SIPWebservices é um componente do SIPWeb e é implementado no estilo arquitetural REST.

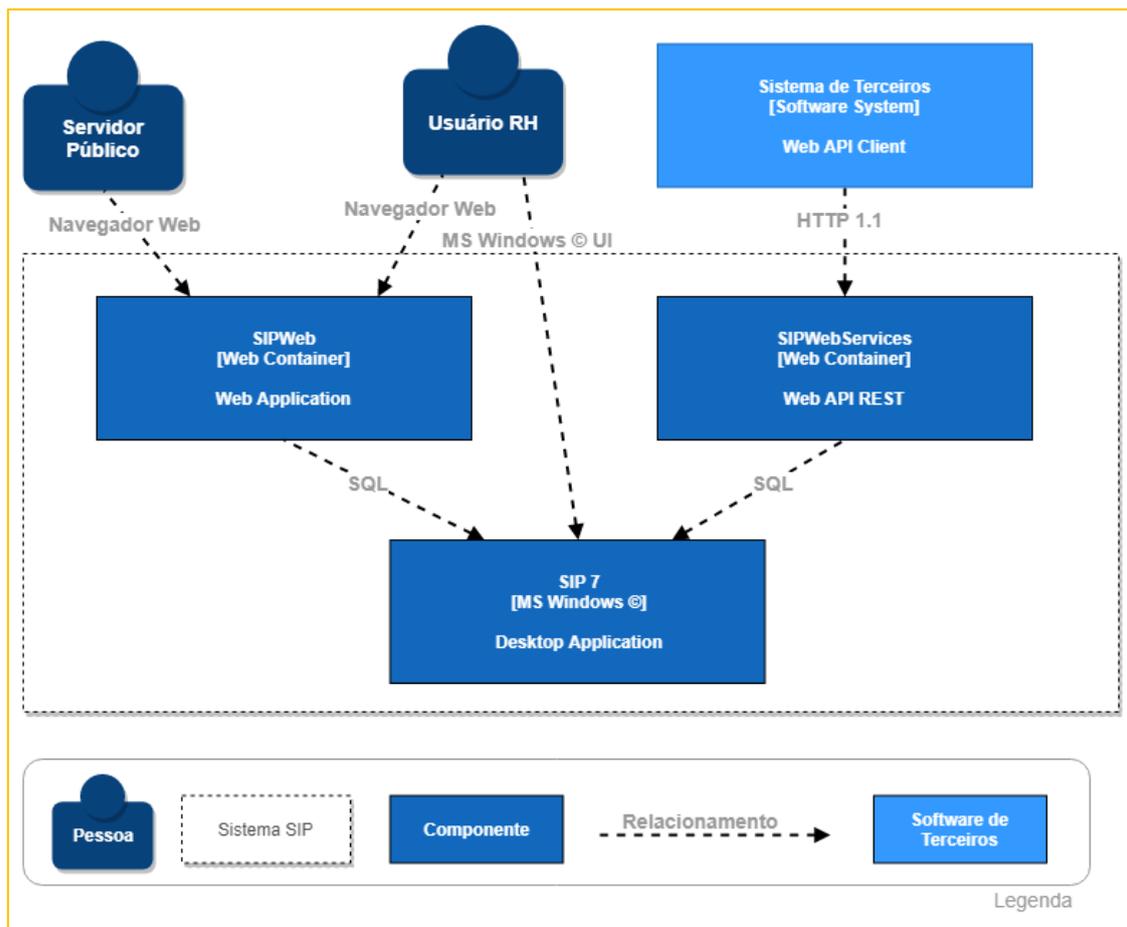


Figura 1 - Arquitetura

Versionamento

Os recursos (URI) e seus comportamentos podem ser modificados ao longo do tempo e para manter a retro compatibilidade a API é versionada. Essa versão caminha numericamente de forma independente da versão do SIPWeb.

A forma de indicar qual versão da API deve responder aos *requests* é através do cabeçalho HTTP "versao-sipwebservices", ou seja, para indicar a versão 1.3.10, informe no cabeçalho HTTP "versao-sipwebservices: 1.3.10". Se esse cabeçalho HTTP não for informado, a última versão responderá aos *requests*.

Este manual segue o versionamento da API, portanto existe um documento deste para cada versão disponível. Mesmo que as alterações sejam mínimas, este documento sempre será completo para cada versão.

Autenticação e Autorização

O modelo de autenticação é o *Basic Auth* conforme exemplo na imagem abaixo.

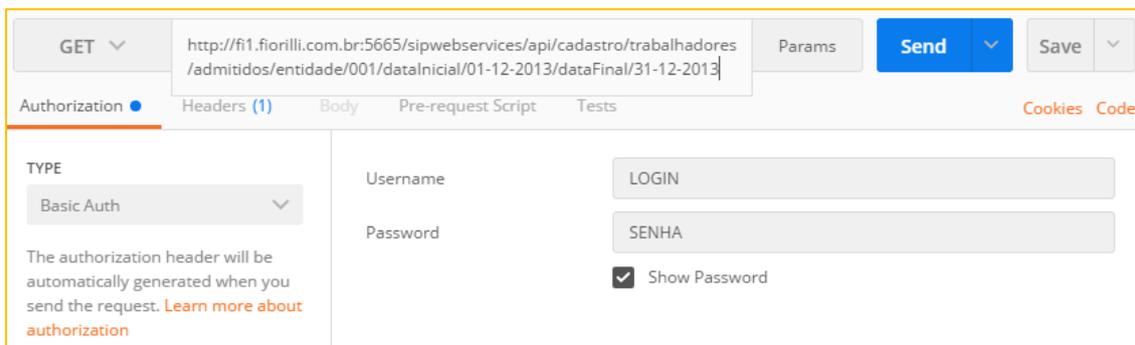


Figura 2 - Reprodução Aplicativo Postman

Para receber a resposta HTTP Status 200 ao acessar a URI <http://localhost:8080/sipwebservices/api/entidade>, por exemplo, é necessário enviar no cabeçalho da requisição HTTP o login e a senha de um usuário que esteja associado a um perfil que tenha a opção “Listar Entidade a partir do SIPWebServices”.

Os recursos (URI) do SIPWebservices são autorizados através do módulo SIP 7, versão desktop. O sistema de autorização do SIP é baseado em perfis, então o usuário utilizado para acessar o SIPWebservices deve estar associado a um perfil com essas autorizações.

Apenas um usuário com perfil de administrador pode criar ou alterar um perfil para acessar o SIPWebservices, portanto não depende da Fiorilli nem do departamento técnico do SIP criar esse usuário e senha. A empresa representante do sistema (*software*) que for acessar o SIPWebservices deve entrar em contato com o órgão público para solicitar a criação desse perfil e desse usuário através do próprio SIP 7.

A imagem abaixo (Figura 3 - Reprodução SIP 7 Desktop) ilustra o módulo SIP 7 exibindo o Cadastro de Perfil de Usuários contendo a opção “Listar Entidade a partir do SIPWebServices” autorizada a ser acessada.

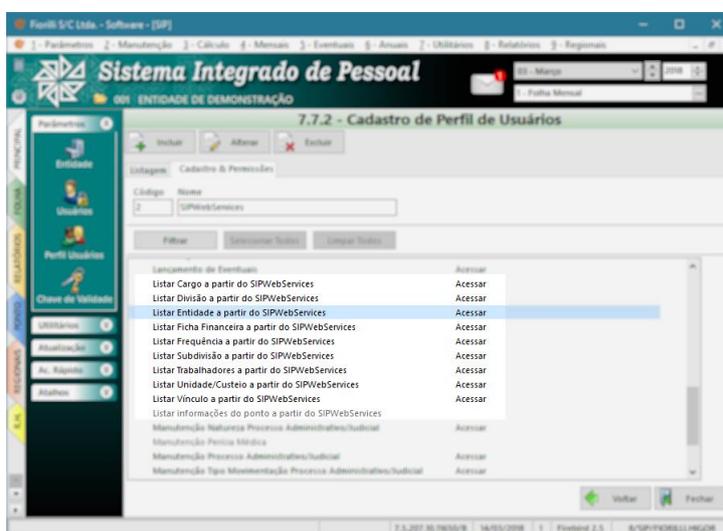


Figura 3 - Reprodução SIP 7 Desktop



MediaType

Os MediaTypes disponíveis são:

Nome	Formato do Retorno
text/csv	CSV
text/plain	CSV
application/json	JSON
application/xml	XML

XML

O tipo application/xml pode retornar uma lista da tag desejada ou apenas uma tag com seus atributos, por exemplo, ao solicitar o recurso trabalhadores alterados no período, o XML de retorno será sempre uma listagem, mas ao solicitar o recurso trabalhador por matrícula, o XML de retorno será sempre uma tag com os atributos daquele trabalhador.

JSON

O tipo application/json pode retornar um objeto array ou apenas um objeto simples com seus atributos, por exemplo, ao solicitar o recurso trabalhadores alterados no período, o JSON de retorno será sempre um objeto que contém um array, mas ao solicitar o recurso trabalhador por matrícula, o JSON de retorno será sempre um objeto simples com os atributos daquele trabalhador.

CSV

Os tipos text/plain ou text/csv retornam um arquivo no formato CSV no qual a primeira linha é sempre o nome das colunas da tabela. Este arquivo é separado por vírgula e possui aspas duplas. Nem todos os recursos desta API retornam CSV, pois existem aqueles que não são listagens, então apenas XML ou JSON estão disponíveis e a aplicação cliente receberá o erro HTTP Status 500: *No match for accept header*.

Ambiente de Testes

Para realizar testes disponibilizamos a porta 5665 no endereço fi1.fiorilli.com.br, online de segunda-feira 07:00 a sexta-feira 19:00. Alertamos que ele pode sofrer indisponibilidades sem prévio aviso. A credencial para testes é o login **testews**, senha **123456789**.

Recursos Disponíveis

Todas as URIs disponíveis devem ser dinamicamente exploradas para se identificar o conteúdo das respostas HTTP. Uma vez que o formato é especificado no cabeçalho HTTP, resta apenas saber qual o conteúdo das respostas de cada URI. Essa flexibilidade e dinamicidade é característico de Web APIs REST, diferenciando-a do SOAP, que é um protocolo estaticamente tipado.

Esta API foi desenhada para desencorajar a verbosidade limitando os recursos disponíveis expondo o máximo de URI que retornam apenas listagens para evitar sistemas de sincronização por chave primária. Apenas alguns endereços retornam um único objeto para atender casos específicos.

Para ter acesso a listagem dos recursos disponíveis acesse o endereço `/sipwebservices/doc`, neste documento HTML estão as URIs e seus parâmetros, sejam eles *path* ou *query*.



Além da exploração dinâmica da API é possível usar um gerador de códigos para sua plataforma se ela for suportada pela comunidade. Para isso baixe a ferramenta swagger-codegen-cli-X.Y.Z.jar disponível no endereço <https://swagger.io/swagger-codegen/>. Em seguida coloque numa pasta vazia e digite o comando abaixo:

```
java -jar swagger-codegen-cli-2.2.1.jar generate -i  
http://servidor:porta/sipwebservices/api/swagger.json -l python
```

Este comando gera o cliente de acesso na plataforma Python.