

Especificación de archivos JSON en RES-PwA

RES-PwA: Sistema asistencial para el soporte emocional de personas con Alzheimer utilizando al robot Pepper.

Brayan Ricardo García Morales
María Fernanda Garcés Cala
Juan Sebastián León Suarez
Juan Diego Osorio Hernández

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
BOGOTÁ, D.C.
2020

Contenido

Introducción	5
1. Integración entre el módulo Agent-RESPWA y el módulo Pepper-Robot.	5
2. Integración entre el módulo Pepper Robot y el módulo Agent-RESPWA.	6
3. Configuración del Perfil emocional enviado al Robot.	8

LISTA DE TABLAS

No se encuentran elementos de tabla de ilustraciones.



LISTA DE ILUSTRACIONES

Ilustración 1. JSON del Agente RES-PwA al Robot.....	5
Ilustración 18. JSON del Robot al Agente RES-PwA.....	6

Introducción

En el presente documento se especifican los archivos JSON utilizados en el sistema RES-PwA, con el fin de proveer una guía y un soporte a la implementación de servicios adicionales. A continuación, encontrará la especificación de los JSON de integración entre los módulos Agent-RESPwA y Pepper Robot, en conjunto con la especificación del JSON utilizado para la configuración emocional del robot Pepper. Se anexan además ejemplos de estos archivos utilizados en este trabajo de grado.

Para garantizar un intercambio de información estandarizado, serializable y sencillo, se definieron dos archivos JSON, cada uno correspondiente a un mensaje de solicitud de servicio (Del Agente RES-PwA al robot) y un mensaje de respuesta del servicio (Del robot al agente RESPwA). Cabe mencionar que estos archivos son construidos e interpretados por el facade específico al robot implementado en el módulo Agent-RESPwA, lo que brinda la flexibilidad suficiente para agregar nuevos robots y/o modificar el estándar de comunicación al agregar un nuevo módulo.

En la Ilustración 1. JSON del Agente RES-PwA al Robot, se muestra la estructura del JSON de solicitud y en la Ilustración 2. JSON del Robot al Agente RES-PwA se evidencia la estructura del JSON de respuesta.

1. Integración entre el módulo Agent-RESPwA y el módulo Pepper-Robot.

```
{
  "id": 0,
  "proxyName": "proxyName",
  "methodName": "nombredelmetodo que tiene que llamar",
  "params": {
    "parametro1": "param1",
    "Lista de parametros": [
      "URL",
      "URL2",
      "imagen"
    ]
  }
}
```

Ilustración 1. JSON del Agente RES-PwA al Robot

Como se evidencia en la Ilustración 1. JSON del Agente RES-PwA al Robot, el JSON de integración entre el módulo Pepper Robot y el módulo Agent-ResPwA consta de los siguientes elementos:

- **ID:** El id del mensaje, se utiliza para confirmar la realización del servicio solicitado.
- **ProxyName:** El proxy del cual se desea solicitar el servicio.
- **MethodName:** Según la especificación de la documentación del proxy, como se puede ver en la documentación de Naoqi.
- **Params:** Contiene información necesaria para llamar al método, en forma de parámetros. El contenido de este atributo depende de la especificación de los servicios del sistema.

A continuación, se adjunta el JSON utilizado para llamar el servicio de Tablet, en específico el llamado al método para mostrar un video. Como se puede evidenciar, se envía el parámetro url.

```
{
  "id": 1,
  "proxyName": "TabletServices",
  "methodName": "SHOWVIDEO",
  "params": {"URL": "www.pagina.com/video.mp4"}
}
```

2. Integración entre el módulo Pepper Robot y el módulo Agent-RESPwA.

```
{
  "id": 0, // -1 si es un callback
  "respType": "act",
  "hasEmo": "true",
  "params": {
    "subEstructura": {
      "param1": "valor1"
    },
    "param2": "valor2"
  }
}
```

Ilustración 2. JSON del Robot al Agente RES-PwA

Como se evidencia en la Ilustración 2. JSON del Robot al Agente RES-PwA, el JSON de integración entre el módulo Pepper Robot y el módulo Agent-ResPwA consta de los siguientes elementos:

- **ID:** que corresponde al mensaje de solicitud de retorno. Tiene un valor de -1 en caso de ser un evento de interacción generado independientemente por el robot.
- **RespType:** Tipo de respuesta, que indica el tipo de información contenido por el mensaje. Este puede estar relacionado con la interacción con el PwA (“int”), el estado del hardware del robot y su modelo emocional interno (“rob”), el estado emocional del PwA (“emo”), el estado de la actividad de entretenimiento (“act”) o de la retroalimentación recibida para modificar el perfil del usuario (“act”).
- **Params:** Subestructura de parámetros, que contienen desde objetos hasta listas. Estos parámetros corresponden a información relevante al módulo Agent-RESPwA, de manera que las creencias del agente RobotAgentBDI puedan ser actualizadas y de esta manera tomar decisiones informadas.

- **hasEmo:** Indica que el mensaje contiene información relevante para el EmotionalAnalyzerAgent en caso de que el valor provisto en requestType no sea “emo”. Es de tipo booleano.

A continuación, se adjunta el JSON utilizado retornar la información emocional del PwA. Como se evidencia responde a la petición con id 1, con un tipo de respuesta emocional (“emo”). En params se muestra el contenido de “PersonData”, que corresponde a la estructura provista por el sistema operativo Qi.

```
{ "id": 1, // -1 si es evento
  "respType": "emo"
  "params": { "PersonData": {
    "valence": {value, confidence},
    "attention": {value, confidence},
    "bodyLanguageState": { "ease": {level, confidence} },
    "smile": {value, confidence},
    "expressions": {
      "calm": {value, confidence},
      "anger": {value, confidence},
      "joy": {value, confidence},
      "sorrow": {value, confidence},
      "laughter": {value, confidence},
      "excitement": {value, confidence},
      "surprise": {value, confidence} } } }
}
```

3. Configuración del Perfil emocional enviado al Robot.

```
"emotion":  
  "image": "url",  
  "posture":  
    {  
      "joint_1":  
        {  
          "key": [0.10],  
          "time": [0.76]  
        },  
      "joint_2":  
        {  
          "key": [0.10],  
          "time": [0.76]  
        }  
    }  
}
```

- **Emotion:** hace referencia al nombre del rango emocional que se va a representar. En este caso pueden ser los siguiente: VHAPPY, HAPPY, NORMAL, SAD y VSAD.
- **Image:** imagen que se despliega en la tablet de acuerdo con la emoción.
- **Posture:** incluye la información equivalente a cada una de las articulaciones que realizaran un cambio en su movimiento. Debido a esto, se incluye el nombre de la articulación("joint_1") junto a sus ángulos("key") y tiempos respectivos("time").
- **Key:** lista de ángulos acordes a la animación solicitada
- **Time:** lista de tiempos asociada los ángulos provistos en key, no puede contener valores menores o iguales a 0.

El JSON generado depende de la configuración provista en la base de datos del sistema RES-PwA. Si desea generarlo, ejecute el SMA.