title: Debugging a skill weight: 55

Debug your custom skills locally using the skill debugger tool. Use the tool to debug NodeJS skills without having to deploy and register them with Watson Assistant Solutions.

Simulate a converse request from an edge device. See how your skill evaluates the request. View the converse response from the skill.

In the tool, context is saved between calls which allows you to simulate a multi-turn conversation.

Procedure

Complete these steps:

1. Clone the Skill debugger tool.
2. Start your skill. Enter:

```
npm run start
```

3. Start the skill debug tool. Depending on your local platform, start the tool from `../skill_debugger_tool/macOS` or `../skill_debugger_tool/Windows`.
4. If your skill URL is not set to `http://localhost:10011`, go to File > Skill URL to specify the URL.
5. If your skill uses an authentication key, go to File > Skill Key to specify the key.
6. Enter a converse request in the Converse request box. For example:

```
{
    "text": "hello",
    "language": "en-US",
    "userID": "user-001",
    "clientID": "client-001",
    "deviceType": "phone",
    "additionalInformation": {
        "context": {}
    }
}
```

7. (Optional) Provide skill and session context information in the Context box. For example:

```
{
    "skill": {
        "inConversation": false
    },
    "session": {
```

```
            "id": "session-001",
            "new": true,
            "skill": {
            "attributes": {
                "weather-interest": "temperature",
                "inConversation": false
            }
            },
            "attributes": {
            "zone": "city-center"
            },
            "version": "1.0"
        }
    }
```

8. Click **Submit request**.
9. View the evaluation response from the skill. For example:

```
{
    "responseCode": 200,
    "requestResult": "Nlu engine did not return an output",
    "intentities": [
        {
        "intents": [
            {
            "intent": "hello-world",
            "confidence": 1
            }
        ],
        "entities": [],
        "name": "regexp"
        }
    ],
    "handleUtterance": true,
    "context": {
        "application": {
        "id": "app-001",
        "attributes": {}
        },
        "session": {
        "skill": {
            "attributes": {}
        },
        "attributes": {
            "id": "session-001",
            "new": true,
            "skill": {
            "attributes": {
                "weather-interest": "temperature",
                "inConversation": false
```

```
                }
              },
              "attributes": {
              "zone": "city-center"
              },
              "version": "1.0"
            }
          }
        },
        "internal": {}
      }
```

10. View the converse response from the skill.

```
  {
    "reject": false,
    "error": 200,
    "deleteSkillSession": true,
    "captureInput": false,
    "speech": {
      "text": "Hello world"
    },
    "internal": {},
    "skill": {
      "entities": [],
      "intents": [
        {
          "intent": "hello-world",
          "confidence": 1
        }
      ],
      "confidence": 1
    },
    "additionalInformation": {
      "context": {
        "application": {
          "id": "app-001",
          "attributes": {}
        },
        "session": {
          "skill": {
            "attributes": {
              "inConversation": false
            }
          },
          "attributes": {
            "id": "session-001",
            "new": true,
            "skill": {
              "attributes": {
                "weather-interest": "temperature",
```

```
          "inConversation": false
        }
      },
      "attributes": {
        "zone": "city-center"
      },
      "version": "1.0"
    }
  }
 }
}
```

What next?

- For more information about the JSON object that is returned in the evaluation response and the converse response, see Json structure.