To develop and publish BroadAI Agents

BroadAI Agentic Framework

Editor

```
...
    // - agent
    agent:{
        // name your agent to the likes of a persona, e.g. StockBroker,
        Analyst, Linguist, etc.
        "agent-name": "--??--",
        // describe what kind of overall capabilities this agent will
        provide. Better description will help BroadAI MAS utilize it at the
        right place.
        "capability": "--??--",
         // list of all the skills this agent possesses.
        "skills": [
          {
            // name the skill like a function, e.g. translateText,
    composeEmail, etc.
            "skill-name": "skillName",
            // describe what the skill will help achieve. Better
    description will help BroadAI MAS utilize it at the right place.
            "objective": "--??--",
            // MANDATORY: Must match the skill-name
            "action": skillName,
            // parameters the action needs to execute
            "parameters": {
              // MANDATORY: Parameter name used in the function definition
    above
              "param": "--data-type-- | --description and possibly an
    example--"
              // RECOMMENDATION: 'param' value must describe the data-type
    expected followed by brief description and possibly an example. Better
    description and examples will help BroadAI MAS determine appropriate
    parameter values so that the action can execute completely.
            }
          },
          /* add additional skills using the same structure above */
        ]
    }
...
```

Define the `action`, which is essentially a NodeJS `function`

6

Editor

S

p

D

A

N

r

u

n

P

m

r

e

W

S

,

r

e

C

t

C

1

t

r

m

t

1

1

A

P

C

,

t

S

B

,

n

t

p

C

e

n

,

r

W

t

t

n

W

n

L

Export as **`npm package`** and:

- use it as a **private agent**, or
- send it to BroadAI to publish as a **public agent** so that developers can plug-and-play it into their applications

7

```
const skillName = (param) => {

  // RECOMMENDATION: returns Promise
  return new Promise((resolve, reject) => {

    // ... You may use Agentic accelerator methods
    // ... > to call external API resource, or
    // ... > to call LLM using BroadAI's configuration
    // ... depending on your use case.
    // ... Alternatively, you are free to
    // ... write the entire logic on your own.


    resolve();
  });
};
```

BroadAI Agentic Framework

To develop and publish BroadAI Agents

Editor

```
...
    // - agent
    agent:{
        // name your agent to the likes of a persona, e.g. StockBroker,
        Analyst, Linguist, etc.
        "agent-name": "--??--",
        // describe what kind of overall capabilities this agent will
        provide. Better description will help BroadAI MAS utilize it at the
        right place.
        "capability": "--??--",
         // list of all the skills this agent possesses.
        "skills": [
          {
            // name the skill like a function, e.g. translateText,
    composeEmail, etc.
            "skill-name": "skillName",
            // describe what the skill will help achieve. Better
    description will help BroadAI MAS utilize it at the right place.
            "objective": "--??--",
            // MANDATORY: Must match the skill-name
            "action": skillName,
            // parameters the action needs to execute
            "parameters": {
              // MANDATORY: Parameter name used in the function definition
    above
              "param": "--data-type-- | --description and possibly an
    example--"
              // RECOMMENDATION: 'param' value must describe the data-type
    expected followed by brief description and possibly an example. Better
    description and examples will help BroadAI MAS determine appropriate
    parameter values so that the action can execute completely.
            }
          },
          /* add additional skills using the same structure above */
        ]
    }
...
```

6 Define the `action`, which is essentially a NodeJS `function`

Editor

```
const skillName = (param) => {

  // RECOMMENDATION: returns Promise
  return new Promise((resolve, reject) => {

    // ... You may use Agentic accelerator methods
    // ... > to call external API resource, or
    // ... > to call LLM using BroadAI's configuration
    // ... depending on your use case.
    // ... Alternatively, you are free to
    // ... write the entire logic on your own.

   resolve();
  });
};
```

7 Export as `npm package` and:

- use it as a **private agent**, or
- send it to BroadAI to publish as a **public agent** so that developers can plug-and-play it into their applications