CSCE 740 - MEAT Scripting Interface

In the original project description, the customer requested the ability to upload a script of commands to the MEAT system in order to facilitate automated meeting management.

Scripts can be executed through the command line by providing them as an argument to MEAT. For example:
`java - jar MEAT.jar scriptfile.json`

The scripting format shall be a JSON-formatted file, containing one or more **`commands`**. Each command shall have a **`name`** and zero or more **`arguments`**. Each argument has a **`name`** and **`value`**. In the abstract, this is formatted as:

```
[{
    "commands": [
        {
            "name": <command name>,
            "arguments": [
                {
                    "name": <argument name>,
                    "value": <argument value>
                }
            ]
        }
    ]
}]
```

The following commands are supported (these correspond to the major functionality of the system):

<table>
<tr><th>Command Name</th><th>Expected Arguments</th></tr>
<tr><td>add-meeting</td><td>This command is used to add a meeting to the calendar.
<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>start-time</td><td>String, formatted as “HH:MM”. 24-hour clock will be used for scripts.</td></tr>
<tr><td>end-time</td><td>String, formatted as “HH:MM”. 24-hour</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr><td></td><td>clock will be used for scripts.</td></tr>
<tr><td>room-id</td><td>String, should correspond to an entry in the room database.</td></tr>
<tr><td>attendee</td><td>String, argument may be used multiple times. Indicates one attendee each time. Can correspond to either a name (for external attendees) or an employee ID.</td></tr>
<tr><td>description</td><td>String</td></tr>
</table>
</td></tr>
<tr><td>edit-meeting-details</td><td>This command is used to edit all non-attendee details of a meeting. Arguments are used to overwrite existing values. Missing arguments should retain their original values.
<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>meeting-id</td><td>String</td></tr>
<tr><td>date</td><td>Integer, formatted as MMDDYYYY</td></tr>
<tr><td>start-time</td><td>String, formatted as “HH:MM”. 24-hour clock will be used for scripts.</td></tr>
<tr><td>end-time</td><td>String, formatted as “HH:MM”. 24-hour clock will be used for scripts.</td></tr>
<tr><td>room</td><td>String</td></tr>
<tr><td>description</td><td>String</td></tr>
</table>
</td></tr>
<tr><td>edit-meeting-add-attendees</td><td>This command is used to add attendees to an existing meeting.
<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>meeting-id</td><td>String</td></tr>
<tr><td>attendee</td><td>String, argument may be used multiple times. Indicates one attendee each time. Can correspond to either a name (for external attendees) or an employee ID.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td>edit-meeting-remove-attendees</td>
<td>This command is used to remove attendees from an existing meeting.

<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>meeting-id</td><td>String</td></tr>
<tr><td>attendee</td><td>String, argument may be used multiple times. Indicates one attendee each time. Can correspond to either a name (for external attendees) or an employee ID.</td></tr>
</table>
</td>
</tr>
<tr>
<td>delete-meeting</td>
<td>This command is used to delete a meeting, vacation time, or a holiday from the calendar.

<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>meeting-id</td><td>String</td></tr>
</table>
</td>
</tr>
<tr>
<td>add-vacation</td>
<td>This command is used to schedule vacation time.

<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>employee-id</td><td>String, should correspond to an entry in the employee database.</td></tr>
<tr><td>start-date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>end-date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>description</td><td>String</td></tr>
</table>
</td>
</tr>
<tr>
<td>add-holiday</td>
<td>This command is used to schedule a company-wide holiday.

<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>start-date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>end-date</td><td>String, formatted as MMDDYYYY</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr><td>description</td><td>String</td></tr>
</table>
</td></tr>
<tr><td>print-schedule-all</td><td>This command is used to print a company-wide schedule to a log file.

<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>start-date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>end-date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>output-file</td><td>String, name of output file</td></tr>
</table>
</td></tr>
<tr><td>print-schedule-employee</td><td>This command is used to print a employee-specific schedule to a log file.

<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>employee-id</td><td>String, should correspond to an entry in the employee database.</td></tr>
<tr><td>start-date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>end-date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>output-file</td><td>String, name of output file</td></tr>
</table>
</td></tr>
<tr><td>print-schedule-room</td><td>This command is used to print a room-specific schedule to a log file.

<table>
<tr><th>Name</th><th>Value Format</th></tr>
<tr><td>room-id</td><td>String, should correspond to an entry in the room database.</td></tr>
<tr><td>start-date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>end-date</td><td>String, formatted as MMDDYYYY</td></tr>
<tr><td>output-file</td><td>String, name of output file</td></tr>
</table>
</td></tr>
</table>

Names must be formatted as they appear above. Argument names and values must also be formatted as above. Failure to maintain compatibility will result in a loss of points.

The output of any of the schedule commands should also be a JSON-formatted file, in the following abstract format:

```
[{
    "events": [
        {
            "meeting-id": <ID>,
            "date": <Date>,
            "start-time": <Time>,
            "end-time": <Time>,
            "room-id": <Room>,
            "description": <Description>,
            "attendees": [
                {
                    "name": <Name or employee ID>
                }
            ]
        }
    ]
}]
```

If you spot any mistakes or inconsistencies, or have any questions, please let us know.