

EMR Plugins

Introduction

In this assignment we are going to create a new EMR plugin. When the plugin is created, then we are going to define rules in it to add some restrictions to scheduling an appointment.

Assignment

Creating an EMR plugin

1. Navigate to Tables > More > EMR plugins in Raintree.
2. Press 'A' to add a new plugin to the plugins List.
3. The code of the created plugin should be your first name + last name. For example if name is Mait Sarv, then the code should be MAITSARV.
4. Plugin folder should be the same as plugin code.
5. Set plugin description to "Custom rules".
6. Save the plugin.

Creating rules

1. Open the plugin that was created.
2. Click on "Rules Setup" button at the bottom left corner of the window.
3. In the rules list press 'A' to add a new rule.
4. Set rule category to 'Scheduler'.
5. Set rule Action to 'BeforeSave'.
6. Set script type to 'RSL'.
7. The File name field insert 'schedulerRule.frm'.
8. Make sure that the Func name is 'Scheduler_BeforeSaveRule()'.
9. Set rule priority to '1'.
10. Click on 'Open script in RIDE' button.
11. Now you can save the file in RIDE if you like to. If the function is empty, then RIDE will prompt to delete empty functions. Make sure that you do not delete to function.
12. Save the rule in Raintree.

Adding logic inside the rule

Now what we would like to do is to prevent users from scheduling appointments more than 7 days into past and more than 6 months into future. When user tries to schedule outside of the date range that is not allowed, then there should be a warning message shown to user. Users that belong to _DB user group should also see the warning, but they should see it in a prompt

that would ask them if they want to continue. If user in _DB group selects that they want to continue scheduling, then it should allow to schedule the appointment. Please implement this logic in the rule that was created in previous step.

Some notes:

- Variable *apdate* holds the appointment date.
- Agent function *getUserGroupsForUser* will return the users groups that the user belongs to.
- Variable *today* holds the value of today's date.
- Agent functions *changedate* and *changemonth* can be used for date manipulations.
- Command *forcemode 0* will stop the scheduling of the appointment.
- Please use functions in *rslib:message.lib* to show the user messages.

Please test your solution after you have completed the rule.

- It should not be possible to add new appointments outside the defined date range.
- It should not be possible to reschedule existing appointments outside the defined date range.

Scheduler tools

Test if it possible to add new appointments outside the defined date range with Scheduler tools. Try adding new appointments with 'Block Book Appointments' tool and try rescheduling with 'Cancel/Reschedule Future Appointments' tool.

Bonus assignment will be to apply the same rules to the Scheduler tools without changing any of the standard scripts. This means that handlers should not be changed all the changes should be made in events file.

Submission

Please check in all the files that you created or changed during this assignment.

Additionally create a report of the assignment and answer following questions.

1. Why should the scheduler rule that was created in this assignment be 'BeforeSave' and not 'Add' rule.
2. Why do you think the priority of the rule should be set to '1'? Feel free to test scenarios where the priority is not '1' and report the results. Make sure that you restore the priority value after testing.
3. What benefits or disadvantages do you see in using the standard library to show the messages instead of just using the specific built in commands/functions?

Send the report to <email>