Click

ScheduleServiceDev1

CallClickService.cs

RequestAppointmentBookingController.cs

<< Cancel Task>>

SDT APP

ScheduleServiceDev1.OptionalParameters() { CallerIdentity = "MUST" };

ScheduleServiceDev1.Task();

public ActionResult CancelTask(string CancelReason)

public ActionResult CancelTask(string CancelReason)

ScheduleServiceDev1.GEHCCancellationReasonReference objCancellation = new NewSDTApplication.ScheduleServiceDev1.GEHCCancellationReasonReference() { Name = CancelReason };

ScheduleServiceDev1.GEHCRejectionReasonReference objRejectionReason = new NewSDTApplication.ScheduleServiceDev1.GEHCRejectionReasonReference() { Name = "Other" };

ScheduleServiceDev1.GEHCIncompleteReasonReference objIncompleteReason = new NewSDTApplication.ScheduleServiceDev1.GEHCIncompleteReasonReference() { Name = CancelReason };

ScheduleService.ProcessTaskEx(processTaskExRequest.OptionalParameters, ref processTaskExRequest.Task, ref processTaskExRequest.Assignment, processTaskExRequest.ReturnAssignment,

processTaskExRequest.RelatedTasks, processTaskExRequest.SchedulingWorkflow, processTaskExRequest.LogicDomain,

processTaskExRequest.SchedulingHorizon, processTaskExRequest.TaskRequestedProperties, processTaskExRequest.AssignmentRequestedProperties,

processTaskExRequest.ReturnSchedulingError, out processTaskExResponse.LinkedTasks, out processTaskExResponse.SchedulingError);

var cancelStatus = ScheduleService.ProcessTaskEx(processTaskExRequest.OptionalParameters, ref processTaskExRequest.Task, ref processTaskExRequest.Assignment, processTaskExRequest.ReturnAssignment,

processTaskExRequest.RelatedTasks, processTaskExRequest.SchedulingWorkflow, processTaskExRequest.LogicDomain,

processTaskExRequest.SchedulingHorizon, processTaskExRequest.TaskRequestedProperties, processTaskExRequest.AssignmentRequestedProperties,

processTaskExRequest.ReturnSchedulingError, out processTaskExResponse.LinkedTasks, out processTaskExResponse.SchedulingError);

processTaskExRequest.Task.Status = new ScheduleServiceDev1.TaskStatusReference() { Name = taskStatus };

return Json(new {message = cancelStatus});

return bool value;

ScheduleService.ProcessTaskEx(processTaskExRequest.OptionalParameters, ref processTaskExRequest.Task, ref processTaskExRequest.Assignment, processTaskExRequest.ReturnAssignment,

processTaskExRequest.RelatedTasks, processTaskExRequest.SchedulingWorkflow, processTaskExRequest.LogicDomain,

processTaskExRequest.SchedulingHorizon, processTaskExRequest.TaskRequestedProperties, processTaskExRequest.AssignmentRequestedProperties,

processTaskExRequest.ReturnSchedulingError, out processTaskExResponse.LinkedTasks, out processTaskExResponse.SchedulingError);

ScheduleService.ProcessTaskEx(processTaskExRequest.OptionalParameters, ref processTaskExRequest.Task, ref processTaskExRequest.Assignment, processTaskExRequest.ReturnAssignment,

processTaskExRequest.RelatedTasks, processTaskExRequest.SchedulingWorkflow, processTaskExRequest.LogicDomain,

processTaskExRequest.SchedulingHorizon, processTaskExRequest.TaskRequestedProperties, processTaskExRequest.AssignmentRequestedProperties,

processTaskExRequest.ReturnSchedulingError, out processTaskExResponse.LinkedTasks, out processTaskExResponse.SchedulingError);

Click