

Pretty print ☐

```
{"openapi":"3.0.1","info":{"title":"Interviews & Group Discussions APIs","description":"Welcome to the API documentation for Mercer | Mettl's Interviews and Group Discussions. These APIs will allow you to seamlessly integrate your platforms and tools with our system, enabling the scheduling, conducting, and managing of interviews and group discussions. With these APIs, you can automate interview processes, fetch candidate feedback & scores, and manage sessions efficiently.","version":"241.1.6"},"servers":[{"url":"api.mettl.pro/interviews"}],"tags":[{"name":"Interview APIs Controller","description":"This controller serves multiple APIs allowing for different operations related to interviews"}, {"name":"PublicReportController","description":"This controller serves the api for various operation related to a public interview report"}, {"name":"Report APIs Controller","description":"This controller serves multiple APIs allowing for different operations related to reports"}, {"name":"FileUploadController","description":"This controller serves the api for all type of file uploads"}, {"name":"JobProfileController","description":"This controller serves the api for various operation related to an job profile"}, {"name":"Job Profile APIs Controller","description":"This controller serves multiple APIs allowing for different operations related to job profiles"}, {"name":"InterviewerController","description":"This controller serves the api for various operation related to interviewer"}, {"name":"ClientReportSettingController","description":"This controller serves the api for various operation related to a client report setting"}, {"name":"ReportController","description":"This controller serves the api for various operation related to an interview report"}, {"name":"InterviewController","description":"This controller serves the api for various operation on an interview"}, {"name":"CandidateController","description":"This controller serves the api for various operation related to candidate"}],"paths":{"/admin/jobProfile":{"put":{"tags":["JobProfileController"],"operationId":"editJobProfile","requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/JobProfileDto"}}},"required":true},"responses":{"500":{"description":"Internal Server Error","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}},"404":{"description":"Not Found","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"403":{"description":"Forbidden","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"401":{"description":"Unauthorized","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"400":{"description":"Bad Request","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"200":{"description":"OK","content":{"application/json":{"schema":{"$ref":"#/components/schemas/JobProfileDto"}}}}}}},"post":{"tags":["JobProfileController"],"operationId":"saveJobProfile","requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/JobProfileDto"}}},"required":true},"responses":{"500":{"description":"Internal Server Error","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"404":{"description":"Not Found","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"403":{"description":"Forbidden","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"401":{"description":"Unauthorized","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"400":{"description":"Bad Request","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"200":{"description":"OK","content":{"application/json":{"schema":{"$ref":"#/components/schemas/JobProfileDto"}}}}}}},"delete":{"tags":["JobProfileController"],"operationId":"deleteJobProfile","parameters":[{"name":"jobProfileUuid","in":"query","required":true,"schema":{"minLength":1,"type":"string"}}],"responses":{"500":{"description":"Internal Server Error","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"404":{"description":"Not Found","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"403":{"description":"Forbidden","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"401":{"description":"Unauthorized","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"400":{"description":"Bad Request","content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"200":{"description":"OK"}}},"/admin/jobProfile/{jobProfileUuid}/display-name":{"put":{"tags":["JobProfileController"],"operationId":"updateDisplayName","parameters":[{"name":"jobProfileUuid","in":"path","required":true,"schema":{"minLength":1,"type":"string"}},
```

## Pretty print

```
{
  "maxLength": 100, "minLength": 0, "type": "string" } } ], "responses": {
    "500": {
      "description": "Internal Server Error", "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      }
    }, "404": {
      "description": "Not Found", "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      }
    }, "403": {
      "description": "Forbidden", "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      }
    }, "401": {
      "description": "Unauthorized", "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      }
    }, "400": {
      "description": "Bad Request", "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      }
    }, "200": {
      "description": "OK"
    }
  }
}, "/admin/interviewer": {
  "put": {
    "tags": [
      "InterviewerController"
    ], "operationId": "updateInterviewer", "requestBody": {
      "content": {
        "application/json": {
          "schema": {
            "$ref": "#/components/schemas/UpdateUserRequestDto"
          }
        }
      }, "required": true
    }, "responses": {
      "500": {
        "description": "Internal Server Error", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "404": {
        "description": "Not Found", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "403": {
        "description": "Forbidden", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "401": {
        "description": "Unauthorized", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "400": {
        "description": "Bad Request", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "200": {
        "description": "OK"
      }
    }
  }, "post": {
    "tags": [
      "InterviewerController"
    ], "operationId": "addInterviewer", "requestBody": {
      "content": {
        "application/json": {
          "schema": {
            "$ref": "#/components/schemas/UserDto"
          }
        }
      }, "required": true
    }, "responses": {
      "500": {
        "description": "Internal Server Error", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "404": {
        "description": "Not Found", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "403": {
        "description": "Forbidden", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "401": {
        "description": "Unauthorized", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "400": {
        "description": "Bad Request", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "200": {
        "description": "OK"
      }
    }
  }
}, "/admin/interview/{interviewUuid}/name": {
  "put": {
    "tags": [
      "InterviewController"
    ], "summary": "This api will update the name of the interview.", "operationId": "updateInterviewName", "parameters": [
      {
        "name": "interviewUuid", "in": "path", "required": true, "schema": {
          "type": "string"
        }
      }, {
        "name": "name", "in": "query", "required": true, "schema": {
          "maxLength": 100, "minLength": 0, "type": "string"
        }
      }
    ], "responses": {
      "500": {
        "description": "Internal Server Error", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "404": {
        "description": "Not Found", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "403": {
        "description": "Forbidden", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "401": {
        "description": "Unauthorized", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "400": {
        "description": "Bad Request", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "200": {
        "description": "OK"
      }
    }
  }
}, "/admin/interview/star/{value}": {
  "put": {
    "tags": [
      "InterviewController"
    ], "summary": "This api will mark the interviews as starred.", "operationId": "updateInterviewStar", "parameters": [
      {
        "name": "value", "in": "path", "required": true, "schema": {
          "type": "boolean"
        }
      }
    ], "requestBody": {
      "content": {
        "application/json": {
          "schema": {
            "type": "array", "items": {
              "type": "string"
            }
          }
        }
      }, "required": true
    }, "responses": {
      "500": {
        "description": "Internal Server Error", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "404": {
        "description": "Not Found", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "403": {
        "description": "Forbidden", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "401": {
        "description": "Unauthorized", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "400": {
        "description": "Bad Request", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "200": {
        "description": "OK"
      }
    }
  }
}, "/admin/interview/reschedule": {
  "put": {
    "tags": [
      "InterviewController"
    ], "summary": "This api will reschedule the interview.", "operationId": "rescheduleInterview", "parameters": [
      {
        "name": "interviewUuid", "in": "path", "required": true, "schema": {
          "type": "string"
        }
      }, {
        "name": "rescheduleTime", "in": "query", "required": true, "schema": {
          "type": "string"
        }
      }
    ], "requestBody": {
      "content": {
        "application/json": {
          "schema": {
            "type": "array", "items": {
              "type": "string"
            }
          }
        }
      }, "required": true
    }, "responses": {
      "500": {
        "description": "Internal Server Error", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "404": {
        "description": "Not Found", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "403": {
        "description": "Forbidden", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "401": {
        "description": "Unauthorized", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "400": {
        "description": "Bad Request", "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }, "200": {
        "description": "OK"
      }
    }
  }
}
}
```

## Pretty print

```

interview": {"operationId": "rescheduleInterview", "requestBody": {"content":
{"application/json": {"schema":
{"$ref": "#/components/schemas/RescheduleInterviewRequest"}}, "required": true, "responses":
{"500": {"description": "Internal Server Error", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "404": {"description": "Not
Found", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}, "403":
{"description": "Forbidden", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "401":
{"description": "Unauthorized", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "400": {"description": "Bad
Request", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "200":
{"description": "OK"}}, "/admin/elastic/users": {"put": {"tags": ["elastic-search-
controller"], "operationId": "loadParticipantsInElastic", "responses": {"500":
{"description": "Internal Server Error", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "404": {"description": "Not
Found", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}, "403":
{"description": "Forbidden", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "401":
{"description": "Unauthorized", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "400": {"description": "Bad
Request", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "200": {"description": "OK"}}, "delete":
{"tags": ["elastic-search-controller"], "operationId": "deleteParticipants", "responses":
{"500": {"description": "Internal Server Error", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "404": {"description": "Not
Found", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}, "403":
{"description": "Forbidden", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "401":
{"description": "Unauthorized", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "400": {"description": "Bad
Request", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "200": {"description": "OK"}}, "/admin/elastic/jobProfiles": {"put": {"tags": ["elastic-search-
controller"], "operationId": "loadJobProfilesInElastic", "responses": {"500":
{"description": "Internal Server Error", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "404": {"description": "Not
Found", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}, "403":
{"description": "Forbidden", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "401":
{"description": "Unauthorized", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "400": {"description": "Bad
Request", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "200": {"description": "OK"}}, "delete":
{"tags": ["elastic-search-controller"], "operationId": "deleteJobProfiles", "responses": {"500":
{"description": "Internal Server Error", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "404": {"description": "Not
Found", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}, "403":
{"description": "Forbidden", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "401":
{"description": "Unauthorized", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "400": {"description": "Bad
Request", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "200": {"description": "OK"}}, "/admin/elastic/interviews": {"put": {"tags": ["elastic-search-
controller"], "operationId": "loadInterviewsInElastic", "responses": {"500":
{"description": "Internal Server Error", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "404": {"description": "Not
Found", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}, "403":
{"description": "Forbidden", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "401":
{"description": "Unauthorized", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "400": {"description": "Bad
Request", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "200": {"description": "OK"}}, "delete":
{"tags": ["elastic-search-controller"], "operationId": "deleteInterviews", "responses": {"500":
{"description": "Internal Server Error", "content": {"*/*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}, "404": {"description": "Not

```

## Pretty print

```
{
  "description": "Forbidden",
  "content": {
    "*/*": {
      "schema": {
        "$ref": "#/components/schemas/InterviewError"
      }
    }
  },
  "401": {
    "description": "Unauthorized",
    "content": {
      "*/*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"
        }
      }
    }
  },
  "400": {
    "description": "Bad Request",
    "content": {
      "*/*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"
        }
      }
    }
  },
  "200": {
    "description": "OK"
  }
},
"/admin/candidate": {
  "put": {
    "tags": [
      "CandidateController"
    ],
    "operationId": "updateCandidate",
    "requestBody": {
      "content": {
        "application/json": {
          "schema": {
            "$ref": "#/components/schemas/UserDto"
          }
        }
      },
      "required": true
    },
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/UserDto"
            }
          }
        }
      }
    },
    "post": {
      "tags": [
        "CandidateController"
      ],
      "operationId": "addCandidate",
      "requestBody": {
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/UserDto"
            }
          }
        },
        "required": true
      },
      "responses": {
        "500": {
          "description": "Internal Server Error",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "404": {
          "description": "Not Found",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "403": {
          "description": "Forbidden",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "401": {
          "description": "Unauthorized",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "400": {
          "description": "Bad Request",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "200": {
          "description": "OK",
          "content": {
            "application/json": {
              "schema": {
                "$ref": "#/components/schemas/UserDto"
              }
            }
          }
        }
      }
    }
  }
},
"/v1/report/action/{interviewUuid}/{candidateUuid}/{candidateStatus}": {
  "post": {
    "tags": [
      "Report APIs Controller"
    ],
    "summary": "This API can be used to change the shortlisting status of a candidate",
    "operationId": "takeAction",
    "parameters": [
      {
        "name": "interviewUuid",
        "in": "path",
        "description": "Unique identifier of the interview",
        "required": true,
        "schema": {
          "minLength": 1,
          "type": "string"
        },
        "example": "7ce6643b"
      },
      {
        "name": "candidateUuid",
        "in": "path",
        "description": "Unique identifier of the candidate",
        "required": true,
        "schema": {
          "minLength": 1,
          "type": "string"
        },
        "example": "e09d5e41"
      },
      {
        "name": "candidateStatus",
        "in": "path",
        "required": true,
        "schema": {
          "type": "string",
          "enum": [
            "SHORTLISTED",
            "REJECTED",
            "ON_HOLD",
            "NOT_EVALUATED"
          ]
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK"
      }
    }
  }
},
"/v1/interview/{interviewUuid}/name": {
  "post": {
    "tags": [
      "Interview APIs Controller"
    ],
    "summary": "This API can be used to update the name of an interview.",
    "operationId": "updateInterviewName_1",
    "parameters": [
      {
        "name": "interviewUuid",
        "in": "path",
        "description": "Unique identifier for the interview",
        "required": true,
        "schema": {
          "type": "string"
        },
        "example": "5080dc7f"
      }
    ],
    "requestBody": {
      "content": {
        "application/json": {
          "schema": {
            "maxLength": 100,
            "minLength": 0,
            "type": "string",
            "description": "Updated name of the interview"
          }
        }
      },
      "required": true
    },
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }
    }
  }
}

```

## Pretty print

```
{
  "/v1/interview/schedule": {
    "post": {
      "tags": ["Interview APIs Controller"],
      "summary": "This API can be used to create and schedule an interview",
      "operationId": "scheduleInterviewV2",
      "requestBody": {
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/ScheduleInterviewApiRequest"
            }
          }
        },
        "required": true,
        "responses": {
          "500": {
            "description": "Internal Server Error",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "404": {
            "description": "Not Found",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "403": {
            "description": "Forbidden",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "401": {
            "description": "Unauthorized",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "400": {
            "description": "Bad Request",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "200": {
            "description": "OK",
            "content": {
              "application/json": {
                "schema": {
                  "$ref": "#/components/schemas/ScheduleInterviewApiResponse"
                }
              }
            }
          }
        }
      }
    }
  },
  "/v1/interview/reschedule": {
    "post": {
      "tags": ["Interview APIs Controller"],
      "summary": "This API can be used to update the scheduled date, time and time zone of an interview",
      "operationId": "rescheduleInterview_1",
      "requestBody": {
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/RescheduleInterviewApiRequest"
            }
          }
        },
        "required": true,
        "responses": {
          "500": {
            "description": "Internal Server Error",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "404": {
            "description": "Not Found",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "403": {
            "description": "Forbidden",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "401": {
            "description": "Unauthorized",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "400": {
            "description": "Bad Request",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "200": {
            "description": "OK"
          }
        }
      }
    }
  },
  "/v1/interview/edit": {
    "post": {
      "tags": ["Interview APIs Controller"],
      "summary": "This API can be used to edit an interview.",
      "operationId": "editInterviewV2",
      "requestBody": {
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/EditInterviewApiRequest"
            }
          }
        },
        "required": true,
        "responses": {
          "500": {
            "description": "Internal Server Error",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "404": {
            "description": "Not Found",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "403": {
            "description": "Forbidden",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "401": {
            "description": "Unauthorized",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "400": {
            "description": "Bad Request",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "200": {
            "description": "OK",
            "content": {
              "application/json": {
                "schema": {
                  "$ref": "#/components/schemas/ScheduleInterviewApiResponse"
                }
              }
            }
          }
        }
      }
    }
  },
  "/report/action/{interviewUuid}/{candidateUuid}": {
    "post": {
      "tags": ["ReportController"],
      "operationId": "takeAction_1",
      "parameters": [
        {
          "name": "interviewUuid",
          "in": "path",
          "required": true,
          "schema": {
            "minLength": 1,
            "type": "string"
          }
        },
        {
          "name": "candidateUuid",
          "in": "path",
          "required": true,
          "schema": {
            "minLength": 1,
            "type": "string"
          }
        }
      ],
      "requestBody": {
        "content": {
          "application/json": {
            "schema": {
              "type": "string",
              "enum": ["SHORTLISTED", "REJECTED", "ON_HOLD", "NOT_EVALUATED"]
            }
          }
        },
        "required": true,
        "responses": {
          "500": {
            "description": "Internal Server Error",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "404": {
            "description": "Not Found",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "403": {
            "description": "Forbidden",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "401": {
            "description": "Unauthorized",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "400": {
            "description": "Bad Request",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "200": {
            "description": "OK"
          }
        }
      }
    }
  },
  "/meeting/recording/completed": {
    "post": {
      "tags": ["meeting-controller"],
      "operationId": "recordingCompleted",
      "parameters": [
        {
          "name": "type",
          "in": "query",
          "required": true,
          "schema": {
            "type": "string"
          }
        }
      ],
      "requestBody": {
        "content": {
          "application/json": {
            "schema": {
              "type": "string"
            }
          }
        },
        "required": true,
        "responses": {
          "500": {
            "description": "Internal Server Error",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          }
        }
      }
    }
  }
}
```

## Pretty print

```

Found": {"content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"type": "string"}}}}}}, "/consent": {"post": {"tags": ["consent-controller"], "operationId": "saveConsentDetails", "requestBody": {"content": {"application/json": {"schema": {"$ref": "#/components/schemas/SaveClientConsentRequest"}}}, "required": true}, "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/ClientConsentDto"}}}}}}}}, "/client/usecase-setting/{category}/{usecase}": {"get": {"tags": ["client-use-case-setting-controller"], "operationId": "getUsecaseSettings", "parameters": [{"name": "usecase", "in": "path", "required": true, "schema": {"type": "string", "enum": ["INTERVIEW", "GROUP_DISCUSSION", "ROLE_PLAY", "ORIENTATION"]}}, {"name": "category", "in": "path", "required": true, "schema": {"type": "string", "enum": ["INTERVIEW", "GROUP_DISCUSSION"]}}], "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ClientUseCaseSettingDto"}}}}}}}}, "post": {"tags": ["client-use-case-setting-controller"], "operationId": "saveUsecaseSettings", "parameters": [{"name": "usecase", "in": "path", "required": true, "schema": {"type": "string", "enum": ["INTERVIEW", "GROUP_DISCUSSION", "ROLE_PLAY", "ORIENTATION"]}}, {"name": "category", "in": "path", "required": true, "schema": {"type": "string", "enum": ["INTERVIEW", "GROUP_DISCUSSION"]}}], "requestBody": {"content": {"application/json": {"schema": {"$ref": "#/components/schemas/ClientUseCaseSettingDto"}}}, "required": true}, "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ClientUseCaseSettingDto"}}}}}}}}, "/client/report-setting": {"get": {"tags": ["ClientReportSettingController"], "summary": "This api will fetch the report setting configured for this client session. If no report setting is configured then default will be returned", "operationId": "getReportSetting", "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}}}}}

```

## Pretty print

```

[{"clientReportSettingController"], "summary": "This api will save or update the report setting this client session.", "operationId": "saveReportSetting", "requestBody": {"content": {"application/json": {"schema": {"$ref": "#/components/schemas/ClientReportSettingDto"}}}, "required": true, "responses": {"500": {"description": "Internal Server Error", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*/*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ClientReportSettingDto"}}}}}}, "/client/emailTemplate": {"get": {"tags": ["client-email-template-controller"], "operationId": "getClientEmailTemplateData", "parameters": [{"name": "sessionCategory", "in": "query", "required": true, "schema": {"type": "string", "enum": ["INTERVIEW", "GROUP_DISCUSSION"]}}, {"name": "key", "in": "query", "required": true, "schema": {"type": "string", "enum": ["ONLINE_INTERVIEW_CANDIDATE_SCHEDULE", "ONLINE_INTERVIEW_CANDIDATE_RESCHEDULE", "ONLINE_INTERVIEW_CANDIDATE_CANCEL", "ONLINE_INTERVIEW_CANDIDATE_UPDATE", "ONLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_REMINDER", "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_INVITE", "ONLINE_INTERVIEW_INTERVIEWER_SCHEDULE", "ONLINE_INTERVIEW_INTERVIEWER_RESCHEDULE", "ONLINE_INTERVIEW_INTERVIEWER_CANCEL", "ONLINE_INTERVIEW_INTERVIEWER_UPDATE", "ONLINE_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_SCHEDULING_REMINDER", "OFFLINE_INTERVIEW_INTERVIEWER_RESCHEDULE", "OFFLINE_INTERVIEW_INTERVIEWER_CANCEL", "OFFLINE_INTERVIEW_INTERVIEWER_UPDATE", "OFFLINE_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_INTERVIEW_INTERVIEWER_SCHEDULING_REMINDER", "OFFLINE_INTERVIEW_CANDIDATE_SCHEDULE", "OFFLINE_INTERVIEW_CANDIDATE_RESCHEDULE", "OFFLINE_INTERVIEW_CANDIDATE_CANCEL", "OFFLINE_INTERVIEW_CANDIDATE_UPDATE", "OFFLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_INTERVIEW_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_INTERVIEW_CLIENT_FINISHED_NOTIFICATION", "ONLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY", "CLIENT_JOB_PROFILE_CANDIDATE_EXCEL_DOWNLOAD", "CLIENT_SESSIONS_USAGE_EXCEL_DOWNLOAD", "ONLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL", "ONLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_REMINDER", "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_INVITE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULING_REMINDER", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULING_REMINDER", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_CLIENT_FINISHED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY", "ONLINE_ROLE_PLAY_CANDIDATE_SCHEDULE", "ONLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE", "ONLINE_ROLE_PLAY_CANDIDATE_CANCEL", "ONLINE_ROLE_PLAY_CANDIDATE_UPDATE", "ONLINE_ROLE_PLAY_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_REMINDER", "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_INVITE", "ONLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "ONLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE", "ONLINE_ROLE_PLAY_INTERVIEWER_CANCEL", "ONLINE_ROLE_PLAY_INTERVIEWER_UPDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_I

```

## Pretty print

```

{
  "OFFLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "OFFLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE", "OFFLINE_ROLE_PLAY_INTERVIEWER_CANCEL", "OFFLINE_ROLE_PLAY_INTERVIEWER_UPDATE", "OFFLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER", "OFFLINE_ROLE_PLAY_CANDIDATE_SCHEDULE", "OFFLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE", "OFFLINE_ROLE_PLAY_CANDIDATE_CANCEL", "OFFLINE_ROLE_PLAY_CANDIDATE_UPDATE", "OFFLINE_ROLE_PLAY_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_FINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_ROLE_PLAY_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY" ] },
  {
    "name": "language",
    "in": "query",
    "required": true,
    "schema": {
      "type": "string",
      "enum": [
        "english", "european_portuguese", "european_spanish", "french", "german", "greek", "italian", "polish", "portuguese_brasil"
      ]
    },
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        },
        "404": {
          "description": "Not Found",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          },
          "403": {
            "description": "Forbidden",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            },
          },
          "401": {
            "description": "Unauthorized",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            },
          },
          "400": {
            "description": "Bad Request",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            },
          },
          "200": {
            "description": "OK",
            "content": {
              "application/json": {
                "schema": {
                  "$ref": "#/components/schemas/GetEmailTemplateDataResponse"
                }
              }
            }
          }
        },
        "post": {
          "tags": [
            "client-email-template-controller"
          ],
          "operationId": "saveEmailTemplate",
          "requestBody": {
            "content": {
              "application/json": {
                "schema": {
                  "$ref": "#/components/schemas/SaveEmailTemplateDataRequest"
                }
              }
            },
            "required": true
          },
          "responses": {
            "500": {
              "description": "Internal Server Error",
              "content": {
                "*/*": {
                  "schema": {
                    "$ref": "#/components/schemas/InterviewError"
                  }
                }
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "200": {
                "description": "OK",
                "content": {
                  "application/json": {
                    "schema": {
                      "$ref": "#/components/schemas/TestMailRequestDto"
                    }
                  }
                }
              }
            },
            "required": true
          },
          "responses": {
            "500": {
              "description": "Internal Server Error",
              "content": {
                "*/*": {
                  "schema": {
                    "$ref": "#/components/schemas/InterviewError"
                  }
                }
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "200": {
                "description": "OK",
                "content": {
                  "application/json": {
                    "schema": {
                      "$ref": "#/components/schemas/TestMailResponseDto"
                    }
                  }
                }
              }
            }
          }
        },
        "/client/emailTemplate/testMail": {
          "post": {
            "tags": [
              "client-email-template-controller"
            ],
            "operationId": "testMail",
            "requestBody": {
              "content": {
                "application/json": {
                  "schema": {
                    "$ref": "#/components/schemas/TestMailRequestDto"
                  }
                }
              },
              "required": true
            },
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "200": {
                "description": "OK",
                "content": {
                  "application/json": {
                    "schema": {
                      "$ref": "#/components/schemas/TestMailResponseDto"
                    }
                  }
                }
              }
            }
          }
        },
        "/client/email-setting": {
          "get": {
            "tags": [
              "client-email-setting-controller"
            ],
            "summary": "This api will fetch the email setting configured for this client session. If no email setting is configured then default will be returned",
            "operationId": "getEmailSetting",
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "200": {
                "description": "OK",
                "content": {
                  "application/json": {
                    "schema": {
                      "$ref": "#/components/schemas/ClientEmailSettingDto"
                    }
                  }
                }
              }
            }
          },
          "post": {
            "tags": [
              "client-email-setting-controller"
            ],
            "summary": "This api will save or update the email setting for this client session.",
            "operationId": "saveEmailSetting",
            "requestBody": {
              "content": {
                "application/json": {
                  "schema": {
                    "$ref": "#/components/schemas/ClientEmailSettingDto"
                  }
                }
              },
              "required": true
            },
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              },
              "200": {
                "description": "OK",
                "content": {
                  "application/json": {
                    "schema": {
                      "$ref": "#/components/schemas/ClientEmailSettingDto"
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}

```



## Pretty print

```
{
  "500": {
    "description": "Internal Server Error",
    "content": {
      "*/*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"
        }
      }
    },
    "404": {
      "description": "Not Found",
      "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      }
    },
    "403": {
      "description": "Forbidden",
      "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      }
    },
    "401": {
      "description": "Unauthorized",
      "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      }
    },
    "400": {
      "description": "Bad Request",
      "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      }
    },
    "200": {
      "description": "OK",
      "content": {
        "application/json": {
          "schema": {
            "$ref": "#/components/schemas/ClientEmailSettingDto"
          }
        }
      }
    }
  },
  "/admin/welcome": {
    "post": {
      "tags": ["admin-controller"],
      "operationId": "doWelcome",
      "responses": {
        "500": {
          "description": "Internal Server Error",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "404": {
          "description": "Not Found",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "403": {
          "description": "Forbidden",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "401": {
          "description": "Unauthorized",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "400": {
          "description": "Bad Request",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          }
        },
        "200": {
          "description": "OK"
        }
      }
    },
    "/admin/upload/{type}": {
      "post": {
        "tags": ["FileUploadController"],
        "operationId": "upload",
        "parameters": [
          {
            "name": "type",
            "in": "path",
            "required": true,
            "schema": {
              "type": "string"
            }
          }
        ],
        "requestBody": {
          "content": {
            "multipart/form-data": {
              "schema": {
                "$ref": "#/components/schemas/FileUploadRequest"
              }
            }
          },
          "responses": {
            "500": {
              "description": "Internal Server Error",
              "content": {
                "*/*": {
                  "schema": {
                    "$ref": "#/components/schemas/InterviewError"
                  }
                }
              }
            },
            "404": {
              "description": "Not Found",
              "content": {
                "*/*": {
                  "schema": {
                    "$ref": "#/components/schemas/InterviewError"
                  }
                }
              }
            },
            "403": {
              "description": "Forbidden",
              "content": {
                "*/*": {
                  "schema": {
                    "$ref": "#/components/schemas/InterviewError"
                  }
                }
              }
            },
            "401": {
              "description": "Unauthorized",
              "content": {
                "*/*": {
                  "schema": {
                    "$ref": "#/components/schemas/InterviewError"
                  }
                }
              }
            },
            "400": {
              "description": "Bad Request",
              "content": {
                "*/*": {
                  "schema": {
                    "$ref": "#/components/schemas/InterviewError"
                  }
                }
              }
            },
            "200": {
              "description": "OK",
              "content": {
                "*/*": {
                  "schema": {
                    "$ref": "#/components/schemas/FileUploadDetailsDto"
                  }
                }
              }
            }
          }
        },
        "/admin/update-client-details": {
          "post": {
            "tags": ["admin-controller"],
            "operationId": "updateClientInfo",
            "requestBody": {
              "content": {
                "application/json": {
                  "schema": {
                    "type": "object",
                    "additionalProperties": {
                      "type": "object"
                    }
                  },
                  "required": true
                }
              },
              "responses": {
                "500": {
                  "description": "Internal Server Error",
                  "content": {
                    "*/*": {
                      "schema": {
                        "$ref": "#/components/schemas/InterviewError"
                      }
                    }
                  }
                },
                "404": {
                  "description": "Not Found",
                  "content": {
                    "*/*": {
                      "schema": {
                        "$ref": "#/components/schemas/InterviewError"
                      }
                    }
                  }
                },
                "403": {
                  "description": "Forbidden",
                  "content": {
                    "*/*": {
                      "schema": {
                        "$ref": "#/components/schemas/InterviewError"
                      }
                    }
                  }
                },
                "401": {
                  "description": "Unauthorized",
                  "content": {
                    "*/*": {
                      "schema": {
                        "$ref": "#/components/schemas/InterviewError"
                      }
                    }
                  }
                },
                "400": {
                  "description": "Bad Request",
                  "content": {
                    "*/*": {
                      "schema": {
                        "$ref": "#/components/schemas/InterviewError"
                      }
                    }
                  }
                },
                "200": {
                  "description": "OK"
                }
              }
            },
            "/admin/signup": {
              "post": {
                "tags": ["admin-controller"],
                "operationId": "doSignup",
                "requestBody": {
                  "content": {
                    "application/json": {
                      "schema": {
                        "$ref": "#/components/schemas/SignupRequest"
                      }
                    },
                    "required": true
                  }
                },
                "responses": {
                  "500": {
                    "description": "Internal Server Error",
                    "content": {
                      "*/*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      }
                    }
                  },
                  "404": {
                    "description": "Not Found",
                    "content": {
                      "*/*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      }
                    }
                  },
                  "403": {
                    "description": "Forbidden",
                    "content": {
                      "*/*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      }
                    }
                  },
                  "401": {
                    "description": "Unauthorized",
                    "content": {
                      "*/*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      }
                    }
                  },
                  "400": {
                    "description": "Bad Request",
                    "content": {
                      "*/*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      }
                    }
                  },
                  "200": {
                    "description": "OK"
                  }
                }
              },
              "/admin/scheduler/sync/elasticDocuments": {
                "post": {
                  "tags": ["scheduler-controller"],
                  "operationId": "syncElasticDocument",
                  "responses": {
                    "500": {
                      "description": "Internal Server Error",
                      "content": {
                        "*/*": {
                          "schema": {
                            "$ref": "#/components/schemas/InterviewError"
                          }
                        }
                      }
                    },
                    "404": {
                      "description": "Not Found",
                      "content": {
                        "*/*": {
                          "schema": {
                            "$ref": "#/components/schemas/InterviewError"
                          }
                        }
                      }
                    },
                    "403": {
                      "description": "Forbidden",
                      "content": {
                        "*/*": {
                          "schema": {
                            "$ref": "#/components/schemas/InterviewError"
                          }
                        }
                      }
                    },
                    "401": {
                      "description": "Unauthorized",
                      "content": {
                        "*/*": {
                          "schema": {
                            "$ref": "#/components/schemas/InterviewError"
                          }
                        }
                      }
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```

```
{ "$ref": "#/components/schemas/InterviewError" } } } }, "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "200": { "description": "OK" } }, "security": [ { "admin-scheduler-api-key": [] } ] }, "/admin/scheduler/sendReminder/email": { "post": { "tags": ["scheduler-controller"], "operationId": "sendReminderEmail", "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } }, "200": { "description": "OK" } }, "security": [ { "admin-scheduler-api-key": [] } ] }, "/admin/scheduler/cleanup/interview": { "post": { "tags": ["scheduler-controller"], "operationId": "interviewCleanup", "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } }, "200": { "description": "OK" } }, "security": [ { "admin-scheduler-api-key": [] } ] }, "/admin/scheduler/cleanup/file": { "post": { "tags": ["scheduler-controller"], "operationId": "fileCleanup", "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } }, "200": { "description": "OK" } }, "security": [ { "admin-scheduler-api-key": [] } ] }, "/admin/scheduler/autoFinish/interview": { "post": { "tags": ["scheduler-controller"], "operationId": "offlineInterview", "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } }, "200": { "description": "OK" } }, "security": [ { "admin-scheduler-api-key": [] } ] }, "/admin/save-client-details": { "post": { "tags": ["admin-controller"], "operationId": "syncClientInfo", "requestBody": { "content": { "application/json": { "schema": { "type": "object", "additionalProperties": { "type": "object" } } }, "required": true }, "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } }, "200": { "description": "OK" } } }, "/admin/language": { "post": { "tags": ["admin-controller"], "operationId": "updateLanguage", "requestBody": { "content": { "application/json": { "schema": { "type": "object", "additionalProperties": { "type": "object" } } }, "required": true }, "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }, "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } }
```

## Pretty print

```
{
  "description": "Unauthorized",
  "content": {
    "*//*": {
      "schema": {
        "$ref": "#/components/schemas/InterviewError"
      }
    }
  },
  "400": {
    "description": "Bad Request",
    "content": {
      "*//*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"
        }
      }
    }
  },
  "200": {
    "description": "OK"
  },
  "/admin/jobProfile/loadMettlAdminData": {
    "post": {
      "tags": [
        "JobProfileController"
      ],
      "operationId": "loadAdmin",
      "responses": {
        "500": {
          "description": "Internal Server Error",
          "content": {
            "*//*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          },
          "404": {
            "description": "Not Found",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "403": {
            "description": "Forbidden",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "401": {
            "description": "Unauthorized",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "400": {
            "description": "Bad Request",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "200": {
            "description": "OK"
          }
        }
      },
      "/admin/interviewer/status/{status}": {
        "post": {
          "tags": [
            "InterviewerController"
          ],
          "operationId": "updateStatus",
          "parameters": [
            {
              "name": "status",
              "in": "path",
              "required": true,
              "schema": {
                "minLength": 1,
                "type": "string"
              }
            }
          ],
          "requestBody": {
            "content": {
              "application/json": {
                "schema": {
                  "type": "array",
                  "items": {
                    "type": "string"
                  }
                }
              }
            },
            "required": true
          },
          "responses": {
            "500": {
          "description": "Internal Server Error",
          "content": {
            "*//*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          },
          "404": {
            "description": "Not Found",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "403": {
            "description": "Forbidden",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "401": {
            "description": "Unauthorized",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "400": {
            "description": "Bad Request",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            }
          },
          "200": {
            "description": "OK"
          }
        }
      },
      "/admin/interviewer/all/status/{status}": {
        "post": {
          "tags": [
            "InterviewerController"
          ],
          "operationId": "updateStatusForAll",
          "parameters": [
            {
              "name": "status",
              "in": "path",
              "required": true,
              "schema": {
                "type": "string"
              }
            },
            {
              "name": "searchCriteriaJson",
              "in": "query",
              "required": false,
              "schema": {
                "type": "string"
              }
            }
          ],
          "responses": {
            "500": {
              "description": "Internal Server Error",
              "content": {
                "*//*": {
                  "schema": {
                    "$ref": "#/components/schemas/InterviewError"
                  }
                }
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*//*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                }
              },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*//*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                }
              },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*//*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                }
              },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*//*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                }
              },
              "200": {
                "description": "OK"
              }
            }
          },
          "/admin/interview/v2/schedule": {
            "post": {
              "tags": [
                "InterviewController"
              ],
              "summary": "This api will create and schedules an interview. This api can be used if you want to quick create and schedule interview simultaneously.",
              "operationId": "scheduleInterviewV2_1",
              "requestBody": {
                "content": {
                  "application/json": {
                    "schema": {
                      "$ref": "#/components/schemas/ScheduleInterviewRequestV2"
                    }
                  }
                },
                "required": true
              },
              "responses": {
                "500": {
                  "description": "Internal Server Error",
                  "content": {
                    "*//*": {
                      "schema": {
                        "$ref": "#/components/schemas/InterviewError"
                      }
                    }
                  },
                  "404": {
                    "description": "Not Found",
                    "content": {
                      "*//*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      }
                    }
                  },
                  "403": {
                    "description": "Forbidden",
                    "content": {
                      "*//*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      }
                    }
                  },
                  "401": {
                    "description": "Unauthorized",
                    "content": {
                      "*//*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      }
                    }
                  },
                  "400": {
                    "description": "Bad Request",
                    "content": {
                      "*//*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      }
                    }
                  },
                  "200": {
                    "description": "OK",
                    "content": {
                      "application/json": {
                        "schema": {
                          "$ref": "#/components/schemas/CreateAndScheduleInterviewResponse"
                        }
                      }
                    }
                  }
                }
              },
              "/admin/interview/v2/edit": {
                "post": {
                  "tags": [
                    "InterviewController"
                  ],
                  "summary": "This api will edit an interview. This api should be used if the uuid of interviewers and candidates are not available.",
                  "operationId": "editInterviewV2_1",
                  "requestBody": {
                    "content": {
                      "application/json": {
                        "schema": {
                          "$ref": "#/components/schemas/EditInterviewRequestV2"
                        }
                      }
                    },
                    "required": true
                  },
                  "responses": {
                    "500": {
                      "description": "Internal Server Error",
                      "content": {
                        "*//*": {
                          "schema": {
                            "$ref": "#/components/schemas/InterviewError"
                          }
                        }
                      },
                      "404": {
                        "description": "Not Found",
                        "content": {
                          "*//*": {
                            "schema": {
                              "$ref": "#/components/schemas/InterviewError"
                            }
                          }
                        }
                      },
                      "403": {
                        "description": "Forbidden",
                        "content": {
                          "*//*": {
                            "schema": {
                              "$ref": "#/components/schemas/InterviewError"
                            }
                          }
                        }
                      }
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```

```

{"$ref":"#/components/schemas/InterviewError"}}}, "401": {"description":"Unauthorized", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "400":{"description":"Bad Request", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "200":{"description":"OK", "content":{"application/json":{"schema":{"$ref":"#/components/schemas/CreateAndScheduleInterviewResponse"}}}}}}, "/admin/interview/schedule":{"post":{"tags":["InterviewController"], "summary":"Schedule an interview. This api needed interview uuid which can be get from quick create interview response", "operationId":"scheduleInterview", "requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/ScheduleInterviewRequest"}}}, "required":true}, "responses":{"500":{"description":"Internal Server Error", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "404":{"description":"Not Found", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "403":{"description":"Forbidden", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "401":{"description":"Unauthorized", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "400":{"description":"Bad Request", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "200":{"description":"OK"}}}}, "/admin/interview/resubmission/enable":{"post":{"tags":["InterviewController"], "summary":"Enable resubmission for the specified interviewers of the interview", "operationId":"enableResubmission", "requestBody":{"content":{"application/json":{"schema":{"$ref":"#/components/schemas/EnableResubmissionRequest"}}}, "required":true}, "responses":{"500":{"description":"Internal Server Error", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "404":{"description":"Not Found", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "403":{"description":"Forbidden", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "401":{"description":"Unauthorized", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "400":{"description":"Bad Request", "content":{"*//*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}, "200":{"description":"OK"}}}}, "/admin/interview/emailTemplate":{"get":{"tags":["interview-email-template-controller"], "operationId":"getInterviewEmailTemplateDataMap", "parameters":[{"name":"interviewUuid", "in":"query", "required":true, "schema":{"type":"string"}}, {"name":"emailTemplateKeys", "in":"query", "required":true, "schema":{"uniqueItems":true, "type":"array", "items":{"type":"string", "enum":["ONLINE_INTERVIEW_CANDIDATE_SCHEDULE", "ONLINE_INTERVIEW_CANDIDATE_RESCHEDULE", "ONLINE_INTERVIEW_CANDIDATE_CANCEL", "ONLINE_INTERVIEW_CANDIDATE_UPDATE", "ONLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_REMINDER", "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_INVITE", "ONLINE_INTERVIEW_INTERVIEWER_SCHEDULE", "ONLINE_INTERVIEW_INTERVIEWER_RESCHEDULE", "ONLINE_INTERVIEW_INTERVIEWER_CANCEL", "ONLINE_INTERVIEW_INTERVIEWER_UPDATE", "ONLINE_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER", "OFFLINE_INTERVIEW_INTERVIEWER_SCHEDULE", "OFFLINE_INTERVIEW_INTERVIEWER_RESCHEDULE", "OFFLINE_INTERVIEW_INTERVIEWER_CANCEL", "OFFLINE_INTERVIEW_INTERVIEWER_UPDATE", "OFFLINE_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER", "OFFLINE_INTERVIEW_CANDIDATE_SCHEDULE", "OFFLINE_INTERVIEW_CANDIDATE_RESCHEDULE", "OFFLINE_INTERVIEW_CANDIDATE_CANCEL", "OFFLINE_INTERVIEW_CANDIDATE_UPDATE", "OFFLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_INTERVIEW_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_INTERVIEW_CLIENT_FINISHED_NOTIFICATION", "ONLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY", "CLIENT_JOB_PROFILE_CANDIDATE_EXCEL_DOWNLOAD", "CLIENT_SESSIONS_USAGE_EXCEL_DOWNLOAD", "ONLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL", "ONLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_REMINDER", "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_INVITE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE"]}}}], "summary":"Get interview email template data map", "response":{"200":{"description":"OK", "content":{"application/json":{"schema":{"$ref":"#/components/schemas/InterviewEmailTemplateDataMap"}}}}}}}}

```

## Pretty print

```

INTERVIEWER_GRADING_REMINDER", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_GRADING_REMINDER", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_CLIENT_FINISHED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY", "ONLINE_ROLE_PLAY_CANDIDATE_SCHEDULE", "ONLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE", "ONLINE_ROLE_PLAY_CANDIDATE_CANCEL", "ONLINE_ROLE_PLAY_CANDIDATE_UPDATE", "ONLINE_ROLE_PLAY_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_REMINDER", "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_INVITE", "ONLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "ONLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE", "ONLINE_ROLE_PLAY_INTERVIEWER_CANCEL", "ONLINE_ROLE_PLAY_INTERVIEWER_UPDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER", "OFFLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "OFFLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE", "OFFLINE_ROLE_PLAY_INTERVIEWER_CANCEL", "OFFLINE_ROLE_PLAY_INTERVIEWER_UPDATE", "OFFLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER", "OFFLINE_ROLE_PLAY_CANDIDATE_SCHEDULE", "OFFLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE", "OFFLINE_ROLE_PLAY_CANDIDATE_CANCEL", "OFFLINE_ROLE_PLAY_CANDIDATE_UPDATE", "OFFLINE_ROLE_PLAY_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_FINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_ROLE_PLAY_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY"]}}}, "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"type": "object", "additionalProperties": {"$ref": "#/components/schemas/GetEmailTemplateDataResponse"}}}}}, "post": {"tags": ["interview-email-template-controller"], "operationId": "saveEmailTemplate_1", "parameters": [{"name": "interviewUuid", "in": "query", "required": true, "schema": {"type": "string"}}], "requestBody": {"content": {"application/json": {"schema": {"type": "array", "items": {"$ref": "#/components/schemas/SaveEmailTemplateDataRequest"}}}}, "required": true}, "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK"}}}}, "/admin/interview/edit": {"post": {"tags": ["InterviewController"], "summary": "This api will edit an interview. This api should be used if the uuid of interviewers and candidates are available.", "operationId": "editInterview", "requestBody": {"content": {"application/json": {"schema": {"$ref": "#/components/schemas/EditInterviewRequest"}}}}, "required": true}, "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad

```

## Pretty print

```
{
  "$ref": "#/components/schemas/InterviewError"}}, "200": {
  "description": "OK"}}, "/admin/interview/create": {
  "post": {
    "tags": [
      "InterviewController"
    ],
    "summary": "Creates a quick interview without any schedule details",
    "operationId": "createInterview",
    "parameters": [
      {
        "name": "category",
        "in": "query",
        "required": true,
        "schema": {
          "type": "string",
          "enum": [
            "INTERVIEW",
            "GROUP_DISCUSSION"
          ]
        }
      },
      {
        "name": "type",
        "in": "query",
        "required": true,
        "schema": {
          "type": "string",
          "enum": [
            "INTERVIEW",
            "GROUP_DISCUSSION",
            "ROLE_PLAY",
            "ORIENTATION"
          ]
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewBasicDetailsDto"
            }
          }
        }
      }
    }
  }
}, "/admin/interview/clone/{interviewUuid}": {
  "post": {
    "tags": [
      "InterviewController"
    ],
    "summary": "This api will be used to clone an interview. It will generate new interview links and will copy only email templates. The rest of the data will not persist until you schedule new interview with those details",
    "operationId": "cloneInterview",
    "parameters": [
      {
        "name": "interviewUuid",
        "in": "path",
        "required": true,
        "schema": {
          "type": "string"
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewDto"
            }
          }
        }
      }
    }
  }
}, "/admin-panel/resubmission/enable": {
  "post": {
    "tags": [
      "admin-panel-controller"
    ],
    "summary": "Enable resubmission for the specified interviewers of the interview",
    "operationId": "enableResubmission_1",
    "requestBody": {
      "content": {
        "application/json": {
          "schema": {
            "$ref": "#/components/schemas/EnableResubmissionRequest"
          }
        }
      },
      "required": true
    },
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK"}
      }
    }
  }
}, "/admin-panel/interviewer": {
  "post": {
    "tags": [
      "admin-panel-controller"
    ],
    "summary": "Fetch all the interviewer details for the specified interview",
    "operationId": "fetchAllInterviewer",
    "requestBody": {
      "content": {
        "application/json": {
          "schema": {
            "$ref": "#/components/schemas/FetchAllInterviewerRequest"
          }
        }
      },
      "required": true
    },
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/FetchAllInterviewerResponse"
            }
          }
        }
      }
    }
  }
}, "/admin-panel/emailTemplate": {
  "get": {
    "tags": [
      "admin-email-template-controller"
    ],
    "operationId": "getGlobalEmailTemplateData",
    "parameters": [
      {
        "name": "sessionCategory",
        "in": "query",
        "required": true,
        "schema": {
          "type": "string",
          "enum":

```

## Pretty print

```
{ "type": "string", "enum":  
  ["ONLINE_INTERVIEW_CANDIDATE_SCHEDULE", "ONLINE_INTERVIEW_CANDIDATE_RESCHEDULE", "ONLINE_INTE  
VIEW_CANDIDATE_CANCEL", "ONLINE_INTERVIEW_CANDIDATE_UPDATE", "ONLINE_INTERVIEW_CANDIDATE_AUT  
OMATED_REMINDER", "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_REMINDER", "ONLINE_INTERVIEW_CANDIDATE  
_ON_GOING_INVITE", "ONLINE_INTERVIEW_INTERVIEWER_SCHEDULE", "ONLINE_INTERVIEW_INTERVIEWER_RES  
CHEDULE", "ONLINE_INTERVIEW_INTERVIEWER_CANCEL", "ONLINE_INTERVIEW_INTERVIEWER_UPDATE", "ONLIN  
E_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER  
_WITH_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE  
_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOI  
NG_INVITE_WITHOUT_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER", "OFFLINE_INTER  
VIEW_INTERVIEWER_SCHEDULE", "OFFLINE_INTERVIEW_INTERVIEWER_RESCHEDULE", "OFFLINE_INTERVIEW_IN  
TERVIEWER_CANCEL", "OFFLINE_INTERVIEW_INTERVIEWER_UPDATE", "OFFLINE_INTERVIEW_INTERVIEWER_AUT  
OMATED_REMINDER", "OFFLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER", "OFFLINE_INTERVIEW_CANDID  
ATE_SCHEDULE", "OFFLINE_INTERVIEW_CANDIDATE_RESCHEDULE", "OFFLINE_INTERVIEW_CANDIDATE_CANCEL",  
"OFFLINE_INTERVIEW_CANDIDATE_UPDATE", "OFFLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER", "ONL  
INE_INTERVIEW_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICA  
TION", "ONLINE_INTERVIEW_CLIENT_FINISHED_NOTIFICATION", "ONLINE_INTERVIEW_INTERVIEWER_EVALUAT  
ION_SUMMARY", "ONLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_INTERVIEW_CLIENT_EVALUAT  
ED_NOTIFICATION", "OFFLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_INTERVIEW_INTE  
RVIEWER_NOTES_SUMMARY", "CLIENT_JOB_PROFILE_CANDIDATE_EXCEL_DOWNLOAD", "CLIENT_SESSIONS_USAGE  
_EXCEL_DOWNLOAD", "ONLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE", "ONLINE_GROUP_DISCUSSION_CAND  
IDATE_RESCHEDULE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL", "ONLINE_GROUP_DISCUSSION_CANDI  
DATE_UPDATE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSIO  
N_CANDIDATE_ON_GOING_REMINDER", "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_INVITE", "ONLINE_  
GROUP_DISCUSSION_INTERVIEWER_SCHEDULE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE", "ON  
LINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE", "ONL  
INE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_O  
N_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WIT  
HOUT_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLIN  
E_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_GROUP_DISCUSSION  
INTERVIEWER_GRADING_REMINDER", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE", "OFFLINE GROU  
P_DISCUSSION_INTERVIEWER_RESCHEDULE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL", "OFFLINE  
_GROUP_DISCUSSION_INTERVIEWER_UPDATE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMIN  
DER", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_GRADING_REMINDER", "OFFLINE_GROUP_DISCUSSION_CAND  
IDATE_SCHEDULE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE", "OFFLINE_GROUP_DISCUSSION_C  
ANDIDATE_CANCEL", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE", "OFFLINE_GROUP_DISCUSSION_CAND  
IDATE_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLIN  
E_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_CLIENT_FINISHED_  
NOTIFICATION", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_GROUP_DISCUS  
SION_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "O  
FFLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_GROUP_DISCUSSION_INTERVIEW  
ER_NOTES_SUMMARY", "ONLINE_ROLE_PLAY_CANDIDATE_SCHEDULE", "ONLINE_ROLE_PLAY_CANDIDATE_RESCHED  
ULE", "ONLINE_ROLE_PLAY_CANDIDATE_CANCEL", "ONLINE_ROLE_PLAY_CANDIDATE_UPDATE", "ONLINE_ROLE_P  
LAY_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_REMINDER", "ONLINE_RO  
LE_PLAY_CANDIDATE_ON_GOING_INVITE", "ONLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "ONLINE_ROLE_PLA  
Y_INTERVIEWER_RESCHEDULE", "ONLINE_ROLE_PLAY_INTERVIEWER_CANCEL", "ONLINE_ROLE_PLAY_INTERVIEW  
ER_UPDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_INTERVIEWER_  
ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_C  
ANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_I  
NTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDE  
R", "OFFLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "OFFLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE", "OFF  
LINE_ROLE_PLAY_INTERVIEWER_CANCEL", "OFFLINE_ROLE_PLAY_INTERVIEWER_UPDATE", "OFFLINE_ROLE_PLA  
Y_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER", "OFFLINE  
_ROLE_PLAY_CANDIDATE_SCHEDULE", "OFFLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE", "OFFLINE_ROLE_PLAY_  
CANDIDATE_CANCEL", "OFFLINE_ROLE_PLAY_CANDIDATE_UPDATE", "OFFLINE_ROLE_PLAY_CANDIDATE_AUTOMAT  
ED_REMINDER", "ONLINE_ROLE_PLAY_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_E  
VALUATED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_FINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_IN  
TERVIEWER_EVALUATION_SUMMARY", "ONLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_ROLE_PL  
AY_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY", "OFFLI  
NE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY"] },  
{ "name": "language", "in": "query", "required": true, "schema": { "type": "string", "enum":  
  ["english", "european_portuguese", "european_spanish", "french", "german", "greek", "italian", "po  
lish", "portuguese_brasil"] } }, "responses": { "500": { "description": "Internal Server  
Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } }, "404":  
{ "description": "Not Found", "content": { "*/*": { "schema":  
{ "$ref": "#/components/schemas/InterviewError" } } } }, "403":  
{ "description": "Forbidden", "content": { "*/*": { "schema":  
{ "$ref": "#/components/schemas/InterviewError" } } } }, "401":
```

## Pretty print

```
{
  "$ref": "#/components/schemas/InterviewError" } } } } } , "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "200": { "description": "OK", "content": { "application/json": { "schema": { "$ref": "#/components/schemas/GetEmailTemplateDataResponse" } } } } } } , "post": { "tags": ["admin-email-template-controller"], "operationId": "saveEmailTemplate_2", "requestBody": { "content": { "application/json": { "schema": { "$ref": "#/components/schemas/SaveEmailTemplateDataRequest" } } } } } , "required": true, "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } } , "200": { "description": "OK" } } } } } , "/admin-panel/emailTemplate/testMail": { "post": { "tags": ["admin-email-template-controller"], "operationId": "testMail_1", "requestBody": { "content": { "application/json": { "schema": { "$ref": "#/components/schemas/TestMailRequestDto" } } } } } , "required": true, "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } } , "200": { "description": "OK", "content": { "application/json": { "schema": { "$ref": "#/components/schemas/TestMailResponseDto" } } } } } } } } } } , "/admin-panel/emailTemplate/copyToGlobal": { "post": { "tags": ["admin-email-template-controller"], "operationId": "copyToGlobalEmailTemplates", "requestBody": { "content": { "application/json": { "schema": { "$ref": "#/components/schemas/CopyEmailTemplateToGlobalRequest" } } } } } , "required": true, "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } } , "200": { "description": "OK" } } } } } , "/admin-panel/emailTemplate/copyToAccount": { "post": { "tags": ["admin-email-template-controller"], "operationId": "copyToAccountEmailTemplates", "requestBody": { "content": { "application/json": { "schema": { "$ref": "#/components/schemas/CopyEmailTemplateToAccountRequest" } } } } } , "required": true, "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } } , "200": { "description": "OK" } } } } } , "/admin-panel/client-account-setting": { "get": { "tags": ["admin-panel-controller"], "summary": "This api will fetch the account setting configured for this client session. If no account setting is configured then default will be returned", "operationId": "getClientAccountSetting", "parameters": [ { "name": "fetchClientAccountSettingRequest", "in": "query", "required": true, "schema": { "$ref": "#/components/schemas/FetchClientAccountSettingRequest" } } ] , "responses": { "500": { "description": "Internal Server Error", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "404": { "description": "Not Found", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "403": { "description": "Forbidden", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "401": { "description": "Unauthorized", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } , "400": { "description": "Bad Request", "content": { "*/*": { "schema": { "$ref": "#/components/schemas/InterviewError" } } } } } } , "200": { "description": "OK" } } } } } } }
```



## Pretty print

```

{"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/ClientAccountSettingDto"}}}}}, {"post": {"tags": ["admin-panel-controller"], "summary": "This api will save or update the account setting for this client session.", "operationId": "saveAccountSetting", "requestBody": {"content": {"application/json": {"schema": {"$ref": "#/components/schemas/SaveClientAccountSettingRequestDto"}}}}, "required": true}, "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK"}}}}, {"get": {"tags": ["Report APIs Controller"], "summary": "This API can be used to fetch the summary of an interview", "operationId": "getSummary", "parameters": [{"name": "interviewUuid", "in": "path", "description": "Unique identifier of the interview", "required": true, "schema": {"type": "string", "example": "7ce6643b"}}, {"responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/InterviewSummaryResponse"}}}}}}}, {"get": {"tags": ["Report APIs Controller"], "summary": "This API can be used to fetch the recordings of an interview", "operationId": "getInterviewRecordings", "parameters": [{"name": "interviewUuid", "in": "path", "description": "Unique identifier of the interview", "required": true, "schema": {"type": "string", "example": "7ce6643b"}}, {"responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"type": "array", "items": {"$ref": "#/components/schemas/MeetingSummaryResponse"}}}}}}}, {"get": {"tags": ["Report APIs Controller"], "summary": "This API can be used to fetch the logs of an interview", "operationId": "getInterviewLogs", "parameters": [{"name": "interviewUuid", "in": "path", "description": "Unique identifier of the interview", "required": true, "schema": {"type": "string", "example": "7ce6643b"}}, {"responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"type": "array", "items": {"$ref": "#/components/schemas/InterviewActivityLogSummaryResponse"}}}}}}}, {"get": {"tags": ["Report APIs Controller"], "summary": "This API can be used to fetch the detailed feedback and scores for an interview", "operationId": "getInterviewFeedback", "parameters":

```

## Pretty print

```

interview": {"required": true, "schema": {"type": "string"}, "example": "/ce6643b-...", "responses": {
  "500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {
    "$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not
Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {
"description": "Forbidden", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "401": {
"description": "Unauthorized", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {
  "application/json": {"schema": {
    "$ref": "#/components/schemas/InterviewFeedback"}}}}}}, "/v1/jobProfile/{uuid}": {"get": {
  "tags": ["Job Profile APIs Controller"], "summary": "This API can be used to fetch all
details of a job profile", "operationId": "getJobProfileDetail", "parameters": {
  [{"name": "uuid", "in": "path", "description": "Unique id of job
profile", "required": true, "schema": {"type": "string"}, "example": "7ce6643b-e287-44c3-93ca-
5d31ce8dc56d"}]}, "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {
  "schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not
Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {
"description": "Forbidden", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "401": {
"description": "Unauthorized", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {
  "application/json": {"schema": {
    "$ref": "#/components/schemas/JobProfileDto"}}}}}}}}, "/v1/jobProfile/all": {"get": {"tags": {
  ["Job Profile APIs Controller"], "summary": "This API can be used to fetch job profiles in
your account in a paginated form with limited data. To fetch complete details of a job
profile please use getJobProfileDetail
method.", "operationId": "getAllJobProfiles", "parameters": {
  [{"name": "pageNumber", "in": "query", "description": "Page Number denotes which next data to be
fetched. For example if page size is 15 and page no 3 the it means 31 to 45 data will be
fetched.", "required": false, "schema": {"type": "string"}, "example": 1},
  {"name": "pageSize", "in": "query", "description": "No of records to be fetched in a
request", "required": false, "schema": {"type": "string"}, "example": 20},
  {"name": "candidateDetails", "in": "query", "description": "Candidate email or candidate name to
search for", "required": false, "schema": {"type": "string"}, "example": "john.doe@example.com"},
  {"name": "jobProfileName", "in": "query", "description": "Job profile name that you want to
search for", "required": false, "schema": {"type": "string"}, "example": "Software
Developer"}]}, "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {
  "schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not
Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {
"description": "Forbidden", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "401": {
"description": "Unauthorized", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {
  "application/json": {"schema": {
    "$ref": "#/components/schemas/JobProfilePage"}}}}}}}}, "/v1/interview": {"get": {"tags": {
  ["Interview APIs Controller"], "summary": "This API can be used to fetch all interviews in
your account in a paginated form", "operationId": "getAllInterviews", "parameters": {
  [{"name": "candidateEmailId", "in": "query", "description": "Candidate email to search
for", "required": false, "schema": {"type": "string"}, "example": "john.doe@example.com"},
  {"name": "pageNumber", "in": "query", "description": "Starts from 0", "required": false, "schema": {
  "type": "integer", "format": "int32"}, "example": 0},
  {"name": "pageSize", "in": "query", "description": "No of records to be fetched in a
request", "required": false, "schema": {"type": "integer", "format": "int32"}, "example": 15},
  {"name": "interviewType", "in": "query", "description": "Type of
interview", "required": false, "schema": {"type": "string", "default": "INTERVIEW", "enum": {
  ["INTERVIEW", "GROUP_DISCUSSION", "ROLE_PLAY", "ORIENTATION"]}, "example": "INTERVIEW"}]}, "respon
ses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not
Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {
"description": "Forbidden", "content": {"*//*": {"schema": {
  "$ref": "#/components/schemas/InterviewError"}}}}, "401": {
"description": "Unauthorized", "content": {"*//*": {"schema":

```

## Pretty print

```

requestContent: {"/v1/interview/timezones": {"get": {"tags": ["Interview APIs Controller"], "summary": "This API can be used to get the list of the supported time zones", "description": "Example - \\"Asia/Kolkata\\": \\"GMT+05:30 Asia/Kolkata\\", \\"America/Toronto\\": \\"GMT-04:00 America/Toronto\\", \\"Europe/Moscow\\": \\"GMT+03:00 Europe/Moscow\\", \\"Pacific/Funafuti\\": \\"GMT+12:00 Pacific/Funafuti\\", \\"UTC\\": \\"GMT+00:00 UTC\\", \\"WET\\": \\"GMT+01:00 WET\\", "operationId": "getAllTimezones", "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"type": "object", "additionalProperties": {"type": "string"}}}}}}}}, "/v1/interview/detail/{interviewUuid}": {"get": {"tags": ["Interview APIs Controller"], "summary": "This API can be used to fetch all details of an interview", "operationId": "getInterviewDetail", "parameters": [{"name": "interviewUuid", "in": "path", "description": "Unique identifier for the interview", "required": true, "schema": {"minLength": 1, "type": "string", "example": "5080dc7f"}}, {"responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/InterviewResponse"}}}}}}}}, "/report/summary/{interviewUuid}": {"get": {"tags": ["ReportController"], "operationId": "getSummary_1", "parameters": [{"name": "interviewUuid", "in": "path", "required": true, "schema": {"type": "string"}}}, {"responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"$ref": "#/components/schemas/InterviewSummaryDto"}}}}}}}}, "/report/recording/{interviewUuid}": {"get": {"tags": ["ReportController"], "operationId": "getInterviewRecordings_1", "parameters": [{"name": "interviewUuid", "in": "path", "required": true, "schema": {"type": "string"}}}, {"responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"type": "array", "items": {"$ref": "#/components/schemas/MeetingSummaryDto"}}}}}}}}, "/report/public/summary/{interviewUuid}": {"get": {"tags": ["PublicReportController"], "operationId": "getSummary_2", "parameters": [{"name": "interviewUuid", "in": "path", "required": true, "schema": {"type": "string"}}}, {"responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"type": "array", "items": {"$ref": "#/components/schemas/MeetingSummaryDto"}}}}}}}}}}

```

## Pretty print

```
{
  "description": "Not Found",
  "content": {
    "*//*": {
      "schema": {
        "$ref": "#/components/schemas/InterviewError"
      }
    },
    "403": {
      "description": "Forbidden",
      "content": {
        "*//*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        },
        "400": {
          "description": "Bad Request",
          "content": {
            "*//*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          },
          "200": {
            "description": "OK",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewSummaryDto"
                }
              }
            }
          }
        }
      }
    },
    "/report/public/recording/{interviewUuid}": {
      "get": {
        "tags": [
          "PublicReportController"
        ],
        "operationId": "getInterviewRecordings_2",
        "parameters": [
          {
            "name": "interviewUuid",
            "in": "path",
            "required": true,
            "schema": {
              "type": "string"
            }
          }
        ],
        "responses": {
          "500": {
            "description": "Internal Server Error",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*//*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  },
                  "403": {
                    "description": "Forbidden",
                    "content": {
                      "*//*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      },
                      "401": {
                        "description": "Unauthorized",
                        "content": {
                          "*//*": {
                            "schema": {
                              "$ref": "#/components/schemas/InterviewError"
                            }
                          },
                          "400": {
                            "description": "Bad Request",
                            "content": {
                              "*//*": {
                                "schema": {
                                  "$ref": "#/components/schemas/InterviewError"
                                }
                              },
                              "200": {
                                "description": "OK",
                                "content": {
                                  "application/json": {
                "schema": {
                  "type": "array",
                  "items": {
                    "$ref": "#/components/schemas/MeetingSummaryDto"
                  }
                }
              }
            }
          }
        }
      }
    },
    "/report/public/logs/{interviewUuid}": {
      "get": {
        "tags": [
          "PublicReportController"
        ],
        "operationId": "getInterviewLogs_1",
        "parameters": [
          {
            "name": "interviewUuid",
            "in": "path",
            "required": true,
            "schema": {
              "type": "string"
            }
          }
        ],
        "responses": {
          "500": {
            "description": "Internal Server Error",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*//*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  },
                  "403": {
                    "description": "Forbidden",
                    "content": {
                      "*//*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      },
                      "401": {
                        "description": "Unauthorized",
                        "content": {
                          "*//*": {
                            "schema": {
                              "$ref": "#/components/schemas/InterviewError"
                            }
                          },
                          "400": {
                            "description": "Bad Request",
                            "content": {
                              "*//*": {
                                "schema": {
                                  "$ref": "#/components/schemas/InterviewError"
                                }
                              },
                              "200": {
                                "description": "OK",
                                "content": {
                                  "*//*": {
                "schema": {
                  "type": "array",
                  "items": {
                    "$ref": "#/components/schemas/InterviewActivityLogSummaryDto"
                  }
                }
              }
            }
          }
        }
      }
    },
    "/report/public/feedbackSummaryForCandidate/{interviewUuid}": {
      "get": {
        "tags": [
          "PublicReportController"
        ],
        "operationId": "getFeedbackSummaryForCandidate",
        "parameters": [
          {
            "name": "interviewUuid",
            "in": "path",
            "required": true,
            "schema": {
              "type": "string"
            }
          },
          {
            "name": "candidateUuid",
            "in": "query",
            "required": true,
            "schema": {
              "type": "string"
            }
          }
        ],
        "responses": {
          "500": {
            "description": "Internal Server Error",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*//*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  },
                  "403": {
                    "description": "Forbidden",
                    "content": {
                      "*//*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      },
                      "401": {
                        "description": "Unauthorized",
                        "content": {
                          "*//*": {
                            "schema": {
                              "$ref": "#/components/schemas/InterviewError"
                            }
                          },
                          "400": {
                            "description": "Bad Request",
                            "content": {
                              "*//*": {
                                "schema": {
                                  "$ref": "#/components/schemas/InterviewError"
                                }
                              },
                              "200": {
                                "description": "OK",
                                "content": {
                                  "*//*": {
                "schema": {
                  "type": "array",
                  "items": {
                    "$ref": "#/components/schemas/FeedbackSummaryDto"
                  }
                }
              }
            }
          }
        }
      }
    },
    "/report/public/feedbackSummary/{interviewUuid}": {
      "get": {
        "tags": [
          "PublicReportController"
        ],
        "operationId": "getFeedbackSummary",
        "parameters": [
          {
            "name": "interviewUuid",
            "in": "path",
            "required": true,
            "schema": {
              "type": "string"
            }
          }
        ],
        "responses": {
          "500": {
            "description": "Internal Server Error",
            "content": {
              "*//*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              },
              "404": {
                "description": "Not Found",
                "content": {
                  "*//*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  },
                  "403": {
                    "description": "Forbidden",
                    "content": {
                      "*//*": {
                        "schema": {
                          "$ref": "#/components/schemas/InterviewError"
                        }
                      },
                      "401": {
                        "description": "Unauthorized",
                        "content": {
                          "*//*": {
                            "schema": {
                              "$ref": "#/components/schemas/InterviewError"
                            }
                          },
                          "400": {
                            "description": "Bad Request",
                            "content": {
                              "*//*": {
                                "schema": {
                                  "$ref": "#/components/schemas/InterviewError"
                                }
                              }
                            }
                          }
                        }
                      }
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```

## Pretty print

```
{"/**":{"schema":{
{"$ref":"#/components/schemas/FeedbackSummaryDto"}}}}},"/report/public/editor/{interviewU
uid":{"get":{"tags":["PublicReportController"],"operationId":"getEditorData","parameters":
[{"name":"interviewUuid","in":"path","required":true,"schema":
{"type":"string"}}],"responses":{"500":{"description":"Internal Server Error","content":
{"**":{"schema":{"$ref":"#/components/schemas/InterviewError"}}},"404":
{"description":"Not Found","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"403":
{"description":"Forbidden","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"401":
{"description":"Unauthorized","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"400":{"description":"Bad
Request","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"200":{"description":"OK","content":
{"**":{"schema":{"type":"array","items":
{"$ref":"#/components/schemas/QuestionDto"}}}}}}},"/report/public/competencySummaryForCand
idate/{interviewUuid":{"get":{"tags":
["PublicReportController"],"operationId":"getCompetencySummaryForCandidate","parameters":
[{"name":"interviewUuid","in":"path","required":true,"schema":{"type":"string"}},
{"name":"candidateUuid","in":"query","required":true,"schema":
{"type":"string"}}],"responses":{"500":{"description":"Internal Server Error","content":
{"**":{"schema":{"$ref":"#/components/schemas/InterviewError"}}},"404":
{"description":"Not Found","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"403":
{"description":"Forbidden","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"401":
{"description":"Unauthorized","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"400":{"description":"Bad
Request","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"200":{"description":"OK","content":
{"**":{"schema":{"type":"array","items":
{"$ref":"#/components/schemas/CandidateCompetencyEvaluationSummaryDto"}}}}}}},"/report/pub
lic/competencySummary/{interviewUuid":{"get":{"tags":
["PublicReportController"],"operationId":"getCompetencySummary","parameters":
[{"name":"interviewUuid","in":"path","required":true,"schema":
{"type":"string"}}],"responses":{"500":{"description":"Internal Server Error","content":
{"**":{"schema":{"$ref":"#/components/schemas/InterviewError"}}},"404":
{"description":"Not Found","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"403":
{"description":"Forbidden","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"401":
{"description":"Unauthorized","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"400":{"description":"Bad
Request","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"200":{"description":"OK","content":
{"**":{"schema":{"type":"array","items":
{"$ref":"#/components/schemas/CandidateCompetencyEvaluationSummaryDto"}}}}}}},"/report/pub
lic/candidateTimeline/{interviewUuid":{"get":{"tags":
["PublicReportController"],"operationId":"getReportTimeline","parameters":
[{"name":"interviewUuid","in":"path","required":true,"schema":{"type":"string"}},
{"name":"candidateUuid","in":"query","required":true,"schema":
{"type":"string"}}],"responses":{"500":{"description":"Internal Server Error","content":
{"**":{"schema":{"$ref":"#/components/schemas/InterviewError"}}},"404":
{"description":"Not Found","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"403":
{"description":"Forbidden","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"401":
{"description":"Unauthorized","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"400":{"description":"Bad
Request","content":{"**":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}},"200":{"description":"OK","content":
{"application/json":{"schema":{"type":"array","items":
{"$ref":"#/components/schemas/CandidateReportDto"}}}}}}},"/report/public/candidateResultSu
mmmary/{interviewUuid":{"get":{"tags":
["PublicReportController"],"operationId":"getCandidateResultSummary","parameters":
[{"name":"interviewUuid","in":"path","required":true,"schema":
{"type":"string"}}],"responses":{"500":{"description":"Internal Server Error","content":
{"**":{"schema":{"$ref":"#/components/schemas/InterviewError"}}},"404":
```

```

{"$ref": "#/components/schemas/InterviewError"}}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":
{"*//*": {"schema": {"type": "array", "items":
{"$ref": "#/components/schemas/CandidateResultSummary"}}}}}}}, "/report/public/candidateFeed
backAndCompetencySummary/{interviewUuid}": {"get": {"tags":
["PublicReportController"], "operationId": "getCandidateFeedbackAndCompetencySummary", "paramete
rs": [{"name": "interviewUuid", "in": "path", "required": true, "schema":
{"type": "string"}]}, "responses": {"500": {"description": "Internal Server Error", "content":
{"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404":
{"description": "Not Found", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":
{"*//*": {"schema": {"type": "array", "items":
{"$ref": "#/components/schemas/CandidateFeedbackAndCompetencyDetail"}}}}}}}, "/report/logs/{
interviewUuid}": {"get": {"tags":
["ReportController"], "operationId": "getInterviewLogs_2", "parameters":
[{"name": "interviewUuid", "in": "path", "required": true, "schema":
{"type": "string"}]}, "responses": {"500": {"description": "Internal Server Error", "content":
{"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404":
{"description": "Not Found", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":
{"application/json": {"schema": {"type": "array", "items":
{"$ref": "#/components/schemas/InterviewActivityLogSummaryDto"}}}}}}}, "/report/feedbackSumm
aryForCandidate/{interviewUuid}": {"get": {"tags":
["ReportController"], "operationId": "getFeedbackSummaryForCandidate_1", "parameters":
[{"name": "interviewUuid", "in": "path", "required": true, "schema": {"type": "string"}},
{"name": "candidateUuid", "in": "query", "required": true, "schema":
{"type": "string"}]}, "responses": {"500": {"description": "Internal Server Error", "content":
{"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404":
{"description": "Not Found", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":
{"application/json": {"schema":
{"$ref": "#/components/schemas/FeedbackSummaryDto"}}}}}}}, "/report/feedbackSummary/{interview
wUuid}": {"get": {"tags":
["ReportController"], "operationId": "getFeedbackSummary_1", "parameters":
[{"name": "interviewUuid", "in": "path", "required": true, "schema":
{"type": "string"}]}, "responses": {"500": {"description": "Internal Server Error", "content":
{"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404":
{"description": "Not Found", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":

```

## Pretty print

```

{
  "$ref": "#/components/schemas/FeedbackSummaryDto",
  "/report/evaluation/reminder/send": {
    "get": {
      "tags": ["ReportController"],
      "operationId": "sendEvaluationReminder",
      "parameters": [
        {
          "name": "remindingToParticipantUuid",
          "in": "query",
          "required": true,
          "schema": {
            "type": "string"
          }
        },
        {
          "name": "interviewUuid",
          "in": "query",
          "required": true,
          "schema": {
            "type": "string"
          }
        }
      ],
      "responses": {
        "500": {
          "description": "Internal Server Error",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"
              }
            }
          },
          "404": {
            "description": "Not Found",
            "content": {
              "*/*": {
                "schema": {
                  "$ref": "#/components/schemas/InterviewError"
                }
              }
            },
            "403": {
              "description": "Forbidden",
              "content": {
                "*/*": {
                  "schema": {
                    "$ref": "#/components/schemas/InterviewError"
                  }
                }
              },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
                "400": {
                  "description": "Bad Request",
                  "content": {
                    "*/*": {
                      "schema": {
                        "$ref": "#/components/schemas/InterviewError"
                      }
                    }
                  },
                  "200": {
                    "description": "OK"
                  }
                }
              }
            }
          }
        },
        "/report/editor/{interviewUuid}": {
          "get": {
            "tags": ["ReportController"],
            "operationId": "getEditorData_1",
            "parameters": [
              {
                "name": "interviewUuid",
                "in": "path",
                "required": true,
                "schema": {
                  "type": "string"
                }
              }
            ],
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "200": {
                "description": "OK",
                "content": {
                  "application/json": {
                    "schema": {
                      "type": "array",
                      "items": {
                        "$ref": "#/components/schemas/QuestionDto"
                      }
                    }
                  }
                }
              }
            }
          }
        },
        "/report/competencySummaryForCandidate/{interviewUuid}": {
          "get": {
            "tags": ["ReportController"],
            "operationId": "getCompetencySummaryForCandidate_1",
            "parameters": [
              {
                "name": "interviewUuid",
                "in": "path",
                "required": true,
                "schema": {
                  "type": "string"
                }
              },
              {
                "name": "candidateUuid",
                "in": "query",
                "required": true,
                "schema": {
                  "type": "string"
                }
              }
            ],
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "200": {
                "description": "OK",
                "content": {
                  "application/json": {
                    "schema": {
                      "type": "array",
                      "items": {
                        "$ref": "#/components/schemas/CandidateCompetencyEvaluationSummaryDto"
                      }
                    }
                  }
                }
              }
            }
          }
        },
        "/report/competencySummary/{interviewUuid}": {
          "get": {
            "tags": ["ReportController"],
            "operationId": "getCompetencySummary_1",
            "parameters": [
              {
                "name": "interviewUuid",
                "in": "path",
                "required": true,
                "schema": {
                  "type": "string"
                }
              }
            ],
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "200": {
                "description": "OK",
                "content": {
                  "application/json": {
                    "schema": {
                      "type": "array",
                      "items": {
                        "$ref": "#/components/schemas/CandidateCompetencyEvaluationSummaryDto"
                      }
                    }
                  }
                }
              }
            }
          }
        },
        "/report/candidateTimeline/{interviewUuid}/{candidateUuid}": {
          "get": {
            "tags": ["ReportController"],
            "operationId": "getReportTimeline_1",
            "parameters": [
              {
                "name": "interviewUuid",
                "in": "path",
                "required": true,
                "schema": {
                  "minLength": 1,
                  "type": "string"
                }
              },
              {
                "name": "candidateUuid",
                "in": "path",
                "required": true,
                "schema": {
                  "minLength": 1,
                  "type": "string"
                }
              }
            ],
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}

```

## Pretty print

```
{
  "description": "Forbidden",
  "content": {
    "*/*": {
      "schema": {
        "$ref": "#/components/schemas/InterviewError"
      }
    }
  },
  "401": {
    "description": "Unauthorized",
    "content": {
      "*/*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"
        }
      }
    }
  },
  "400": {
    "description": "Bad Request",
    "content": {
      "*/*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"
        }
      }
    }
  },
  "200": {
    "description": "OK",
    "content": {
      "application/json": {
        "schema": {
          "type": "array",
          "items": {
            "$ref": "#/components/schemas/CandidateReportDto"
          }
        }
      }
    }
  }
},
"/report/candidateResultSummary/{interviewUuid}": {
  "get": {
    "tags": [
      "ReportController"
    ],
    "operationId": "getCandidateResultSummary_1",
    "parameters": [
      {
        "name": "interviewUuid",
        "in": "path",
        "required": true,
        "schema": {
          "type": "string"
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "application/json": {
            "schema": {
              "type": "array",
              "items": {
                "$ref": "#/components/schemas/CandidateResultSummary"
              }
            }
          }
        }
      }
    }
  }
},
"/report/candidateFeedbackAndCompetencySummary/{interviewUuid}": {
  "get": {
    "tags": [
      "ReportController"
    ],
    "operationId": "getCandidateFeedbackAndCompetencySummary_1",
    "parameters": [
      {
        "name": "interviewUuid",
        "in": "path",
        "required": true,
        "schema": {
          "type": "string"
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "*/*": {
            "schema": {
              "type": "string"
            }
          }
        }
      }
    }
  }
},
"/i18/public/keys/{interviewUuid}": {
  "get": {
    "tags": [
      "public-i-18n-controller"
    ],
    "operationId": "getInternationalisationKeys2",
    "parameters": [
      {
        "name": "language",
        "in": "query",
        "required": true,
        "schema": {
          "type": "string"
        }
      },
      {
        "name": "interviewUuid",
        "in": "path",
        "required": true,
        "schema": {
          "type": "string"
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "*/*": {
            "schema": {
              "type": "object",
              "additionalProperties": {
                "type": "string"
              }
            }
          }
        }
      }
    }
  }
},
"/i18/keys": {
  "get": {
    "tags": [
      "i-18n-controller"
    ],
    "operationId": "getInternationalisationKeys",
    "parameters": [
      {
        "name": "language",
        "in": "query",
        "required": true,
        "schema": {
          "type": "string"
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      }
    }
  }
}
}
```



## Pretty print

```
{
  "description": "Forbidden",
  "content": {
    "*//*": {
      "schema": {
        "$ref": "#/components/schemas/InterviewError"
      }
    }
  },
  "401": {
    "description": "Unauthorized",
    "content": {
      "*//*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"
        }
      }
    }
  },
  "400": {
    "description": "Bad Request",
    "content": {
      "*//*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"
        }
      }
    }
  },
  "200": {
    "description": "OK",
    "content": {
      "*//*": {
        "schema": {
          "type": "object",
          "additionalProperties": {
            "type": "string"
          }
        }
      }
    }
  }
},
"/codeylsis/metadata": {
  "get": {
    "tags": ["mettl-codeylsis-controller"],
    "operationId": "getAllMetadata",
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "*//*": {
            "schema": {
              "type": "array",
              "items": {
                "$ref": "#/components/schemas/CodeylsisLanguageCompilerVersion"
              }
            }
          }
        }
      }
    }
  }
},
"/codeylsis/codeHistory/{ciid}/uuid/{uuid}/dataSourceType/{dataSourceType}": {
  "get": {
    "tags": ["mettl-codeylsis-controller"],
    "operationId": "getCodeHistoryForCompileUuid",
    "parameters": [
      {
        "name": "uuid",
        "in": "path",
        "required": true,
        "schema": {
          "type": "string"
        }
      },
      {
        "name": "ciid",
        "in": "path",
        "required": true,
        "schema": {
          "type": "integer",
          "format": "int32"
        }
      },
      {
        "name": "dataSourceType",
        "in": "path",
        "required": true,
        "schema": {
          "type": "string"
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "*//*": {
            "schema": {
              "type": "string"
            }
          }
        }
      }
    }
  }
},
"/codeylsis/codeHistory/{ciid}/questionId/{questionId}/metadata": {
  "get": {
    "tags": ["mettl-codeylsis-controller"],
    "operationId": "getCodeHistoryMetadataForCiid",
    "parameters": [
      {
        "name": "ciid",
        "in": "path",
        "required": true,
        "schema": {
          "type": "integer",
          "format": "int32"
        }
      },
      {
        "name": "questionId",
        "in": "path",
        "required": true,
        "schema": {
          "type": "integer",
          "format": "int32"
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "*//*": {
            "schema": {
              "type": "string"
            }
          }
        }
      }
    }
  }
},
"/client/usage-summary/session-count": {
  "get": {
    "tags": ["client-usage-controller"],
    "operationId": "getUsageCounts",
    "parameters": [
      {
        "name": "from",
        "in": "query",
        "required": true,
        "schema": {
          "type": "integer",
          "format": "int64"
        }
      },
      {
        "name": "to",
        "in": "query",
        "required": false,
        "schema": {
          "type": "integer",
          "format": "int64"
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/ClientUsageCountData"
            }
          }
        }
      }
    }
  }
},
"/client/usage-summary/chart-data": {
  "get": {
    "tags": ["client-usage-controller"],
    "operationId": "getUsageChartData",
    "parameters": [
      {
        "name": "from",
        "in": "query",
        "required": true,
        "schema": {
          "type": "integer",
          "format": "int64"
        }
      },
      {
        "name": "to",
        "in": "query",
        "required": false,
        "schema": {
          "type": "integer",
          "format": "int64"
        }
      }
    ],
    "responses": {
      "500": {
        "description": "Internal Server Error",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "404": {
        "description": "Not Found",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*//*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        }
      },
      "200": {
        "description": "OK",
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/ClientUsageChartData"
            }
          }
        }
      }
    }
  }
}
}
```

## Pretty print

```
{
  "name": "to",
  "in": "query",
  "required": false,
  "schema": {
    "type": "integer",
    "format": "int64"
  }
},
{
  "name": "from",
  "in": "query",
  "required": true,
  "schema": {
    "type": "integer",
    "format": "int64"
  }
},
{
  "name": "interval",
  "in": "query",
  "required": true,
  "schema": {
    "type": "string",
    "enum": [
      "DAY",
      "WEEK",
      "MONTH"
    ]
  }
}],
"responses": {
  "500": {
    "description": "Internal Server Error",
    "content": {
      "*/*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"
        }
      }
    },
    "404": {
      "description": "Not Found",
      "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"
          }
        }
      },
      "403": {
        "description": "Forbidden",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        },
      },
      "401": {
        "description": "Unauthorized",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        },
      },
      "400": {
        "description": "Bad Request",
        "content": {
          "*/*": {
            "schema": {
              "$ref": "#/components/schemas/InterviewError"
            }
          }
        },
      },
      "200": {
        "description": "OK",
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/ClientUsageChartData"
            }
          }
        }
      }
    }
  },
  "/client/emailTemplate/default": {
    "get": {
      "tags": [
        "client-email-template-controller"
      ],
      "operationId": "getGlobalEmailTemplateData_1",
      "parameters": [
        {
          "name": "sessionCategory",
          "in": "query",
          "required": true,
          "schema": {
            "type": "string",
            "enum": [
              "INTERVIEW",
              "GROUP_DISCUSSION"
            ]
          }
        },
        {
          "name": "key",
          "in": "query",
          "required": true,
          "schema": {
            "type": "string",
            "enum": [
              "ONLINE_INTERVIEW_CANDIDATE_SCHEDULE",
              "ONLINE_INTERVIEW_CANDIDATE_RESCHEDULE",
              "ONLINE_INTERVIEW_CANDIDATE_CANCEL",
              "ONLINE_INTERVIEW_CANDIDATE_UPDATE",
              "ONLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER",
              "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_REMINDER",
              "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_INVITE",
              "ONLINE_INTERVIEW_INTERVIEWER_SCHEDULE",
              "ONLINE_INTERVIEW_INTERVIEWER_RESCHEDULE",
              "ONLINE_INTERVIEW_INTERVIEWER_CANCEL",
              "ONLINE_INTERVIEW_INTERVIEWER_UPDATE",
              "ONLINE_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER",
              "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE",
              "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE",
              "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE",
              "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE",
              "ONLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER",
              "OFFLINE_INTERVIEW_INTERVIEWER_SCHEDULE",
              "OFFLINE_INTERVIEW_INTERVIEWER_RESCHEDULE",
              "OFFLINE_INTERVIEW_INTERVIEWER_CANCEL",
              "OFFLINE_INTERVIEW_INTERVIEWER_UPDATE",
              "OFFLINE_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER",
              "OFFLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER",
              "OFFLINE_INTERVIEW_CANDIDATE_SCHEDULE",
              "OFFLINE_INTERVIEW_CANDIDATE_RESCHEDULE",
              "OFFLINE_INTERVIEW_CANDIDATE_CANCEL",
              "OFFLINE_INTERVIEW_CANDIDATE_UPDATE",
              "OFFLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER",
              "ONLINE_INTERVIEW_CLIENT_AUTOFINISHED_NOTIFICATION",
              "ONLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION",
              "ONLINE_INTERVIEW_CLIENT_FINISHED_NOTIFICATION",
              "ONLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY",
              "ONLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY",
              "OFFLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION",
              "OFFLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY",
              "OFFLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY",
              "CLIENT_JOB_PROFILE_CANDIDATE_EXCEL_DOWNLOAD",
              "CLIENT_SESSIONS_USAGE_EXCEL_DOWNLOAD",
              "ONLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE",
              "ONLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE",
              "ONLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL",
              "ONLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE",
              "ONLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER",
              "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_REMINDER",
              "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_INVITE",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_GRADING_REMINDER",
              "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE",
              "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE",
              "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL",
              "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE",
              "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER",
              "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_GRADING_REMINDER",
              "OFFLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE",
              "OFFLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE",
              "OFFLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL",
              "OFFLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE",
              "OFFLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER",
              "ONLINE_GROUP_DISCUSSION_CLIENT_AUTOFINISHED_NOTIFICATION",
              "ONLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION",
              "ONLINE_GROUP_DISCUSSION_CLIENT_FINISHED_NOTIFICATION",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY",
              "ONLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY",
              "OFFLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION",
              "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY",
              "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY",
              "ONLINE_ROLE_PLAY_CANDIDATE_SCHEDULE",
              "ONLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE",
              "ONLINE_ROLE_PLAY_CANDIDATE_CANCEL",
              "ONLINE_ROLE_PLAY_CANDIDATE_UPDATE",
              "ONLINE_ROLE_PLAY_CANDIDATE_AUTOMATED_REMINDER",
              "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_REMINDER",
              "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_INVITE",
              "ONLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE",
              "ONLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE",
              "ONLINE_ROLE_PLAY_INTERVIEWER_CANCEL",
              "ONLINE_ROLE_PLAY_INTERVIEWER_UPDATE",
              "ONLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER",
              "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE",
              "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE",
              "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE",
              "ONLINE_ROLE_PLAY_I"
            ]
          }
        }
      ]
    }
  }
}
```

## Pretty print

```

"OFFLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE","OFFLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE","OFF
LINE_ROLE_PLAY_INTERVIEWER_CANCEL","OFFLINE_ROLE_PLAY_INTERVIEWER_UPDATE","OFFLINE_ROLE_PLA
Y_INTERVIEWER_AUTOMATED_REMINDER","OFFLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER","OFFLINE
_ROLE_PLAY_CANDIDATE_SCHEDULE","OFFLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE","OFFLINE_ROLE_PLAY_
CANDIDATE_CANCEL","OFFLINE_ROLE_PLAY_CANDIDATE_UPDATE","OFFLINE_ROLE_PLAY_CANDIDATE_AUTOMAT
ED_REMINDER","ONLINE_ROLE_PLAY_CLIENT_AUTOFINISHED_NOTIFICATION","ONLINE_ROLE_PLAY_CLIENT_E
VALUATED_NOTIFICATION","ONLINE_ROLE_PLAY_CLIENT_FINISHED_NOTIFICATION","ONLINE_ROLE_PLAY_IN
TERVIEWER_EVALUATION_SUMMARY","ONLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY","OFFLINE_ROLE_PL
AY_CLIENT_EVALUATED_NOTIFICATION","OFFLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY","OFFLI
NE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY"]}}},
{"name":"language","in":"query","required":true,"schema":{"type":"string","enum":
["english","european_portuguese","european_spanish","french","german","greek","italian","po
lish","portuguese_brasil"]}}},"responses":{"500":{"description":"Internal Server
Error","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"404":
{"description":"Not Found","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"403":
{"description":"Forbidden","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"401":
{"description":"Unauthorized","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"400":{"description":"Bad
Request","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"200":{"description":"OK","content":
{"application/json":{"schema":
{"$ref":"#/components/schemas/GetEmailTemplateDataResponse"}}}}}}},"/client/email-
setting/emailSenderOption":{"get":{"tags":["client-email-setting-
controller"],"summary":"This api will fetch all the email sender options available for this
client","operationId":"getEmailSenderOptions","responses":{"500":{"description":"Internal
Server Error","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"404":{"description":"Not
Found","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"403":
{"description":"Forbidden","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"401":
{"description":"Unauthorized","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"400":{"description":"Bad
Request","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"200":{"description":"OK","content":
{"application/json":{"schema":
{"$ref":"#/components/schemas/EmailSenderDetailsDto"}}}}}}},"/admin/uisseting":{"get":
{"tags":["admin-controller"],"operationId":"getUiSetting","responses":{"500":
{"description":"Internal Server Error","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"404":{"description":"Not
Found","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"403":
{"description":"Forbidden","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"401":
{"description":"Unauthorized","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"400":{"description":"Bad
Request","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"200":{"description":"OK","content":
{"*/*":{"schema":
{"$ref":"#/components/schemas/AdminUiSetting"}}}}}}},"/admin/jobProfile/{uuid}":{"get":
{"tags":["JobProfileController"],"operationId":"getJobProfileDetail_1","parameters":
[{"name":"uuid","in":"path","required":true,"schema":{"type":"string"}}],"responses":
{"500":{"description":"Internal Server Error","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"404":{"description":"Not
Found","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"403":
{"description":"Forbidden","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"401":
{"description":"Unauthorized","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"400":{"description":"Bad
Request","content":{"*/*":{"schema":
{"$ref":"#/components/schemas/InterviewError"}}}}},"200":{"description":"OK","content":
{"*/*":{"schema":
{"$ref":"#/components/schemas/JobProfileDto"}}}}}}},"/admin/jobProfile/{jobProfileUuid}/lin
ked-mettl-assessments":{"get":{"tags":
["JobProfileController"],"operationId":"getLinkedMettlAssessmentDetail","parameters":
[{"name":"jobProfileUuid","in":"path","required":true,"schema":
{"type":"string"}}],"responses":{"500":{"description":"Internal Server Error","content":
{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}}},"404":

```

```

{"$ref": "#/components/schemas/InterviewError"}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":
{"*//*": {"schema": {"type": "object", "additionalProperties":
{"type": "object"}}}}}}, "/admin/jobProfile/summary": {"get": {"tags":
["JobProfileController"], "operationId": "getJobProfileSummary", "parameters":
[{"name": "jobProfileUuid", "in": "query", "required": true, "schema": {"type": "string"}},
{"name": "category", "in": "query", "required": false, "schema": {"type": "string"}}, "responses":
{"500": {"description": "Internal Server Error", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not
Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":
{"*//*": {"schema":
{"$ref": "#/components/schemas/JobProfileSummary"}}}}}}, "/admin/jobProfile/mettl-
assessments": {"get": {"tags":
["JobProfileController"], "operationId": "getMettlAssessments", "parameters":
[{"name": "searchCriteriaJson", "in": "query", "required": true, "schema":
{"type": "string"}}, "responses": {"500": {"description": "Internal Server Error", "content":
{"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404":
{"description": "Not Found", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":
{"*//*": {"schema":
{"$ref": "#/components/schemas/MettlAssessmentPage"}}}}}}, "/admin/jobProfile/all": {"get":
{"tags": ["JobProfileController"], "operationId": "getJobProfileData", "parameters":
[{"name": "searchCriteriaJson", "in": "query", "required": true, "schema":
{"type": "string"}}, "responses": {"500": {"description": "Internal Server Error", "content":
{"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404":
{"description": "Not Found", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":
{"*//*": {"schema":
{"$ref": "#/components/schemas/JobProfilePage"}}}}}}, "/admin/interviewer/all": {"get":
{"tags": ["InterviewerController"], "operationId": "getAllInterviewers", "parameters":
[{"name": "interviewerCriteriaJson", "in": "query", "required": true, "schema":
{"type": "string"}}, "responses": {"500": {"description": "Internal Server Error", "content":
{"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404":
{"description": "Not Found", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "403":
{"description": "Forbidden", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "401":
{"description": "Unauthorized", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad
Request", "content": {"*//*": {"schema":
{"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content":
{"application/json": {"schema":
{"$ref": "#/components/schemas/InterviewerPage"}}}}}}, "/admin/interview/timezones": {"get":
{"tags": ["InterviewController"], "summary": "This api will return all the supported timezone
in the system", "operationId": "getAllTimezones 1", "responses": {"500":

```

## Pretty print

```
{
  "$ref": "#/components/schemas/InterviewError"}}, "404": {
    "description": "Not Found",
    "content": {
      "*/*": {
        "schema": {
          "$ref": "#/components/schemas/InterviewError"}
        }
      }
    },
    "403": {
      "description": "Forbidden",
      "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"}
          }
        }
      },
    "401": {
      "description": "Unauthorized",
      "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"}
          }
        }
      },
    "400": {
      "description": "Bad Request",
      "content": {
        "*/*": {
          "schema": {
            "$ref": "#/components/schemas/InterviewError"}
          }
        }
      },
    "200": {
      "description": "OK",
      "content": {
        "*/*": {
          "schema": {
            "type": "object",
            "additionalProperties": {
              "type": "string"
            }
          }
        }
      }
    }
  },
  "/admin/interview/reportUrl/{interviewUuid}": {
    "get": {
      "tags": ["InterviewController"],
      "summary": "This api will return report links for an interview. This api can be used to get public and private link of a report.",
      "operationId": "getReportLinks",
      "parameters": [
        {
          "name": "interviewUuid",
          "in": "path",
          "required": true,
          "schema": {
            "type": "string"
          }
        }
      ],
      "responses": {
        "500": {
          "description": "Internal Server Error",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"}
              }
            },
        "404": {
          "description": "Not Found",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"}
              }
            },
        "403": {
          "description": "Forbidden",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"}
              }
            },
        "401": {
          "description": "Unauthorized",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"}
              }
            },
        "400": {
          "description": "Bad Request",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/InterviewError"}
              }
            },
        "200": {
          "description": "OK",
          "content": {
            "*/*": {
              "schema": {
                "$ref": "#/components/schemas/ReportLinkDto"}
              }
            }
          }
        },
        "/admin/interview/new": {
          "get": {
            "tags": ["InterviewController"],
            "summary": "Creates a quick interview without any schedule details",
            "operationId": "getNewInterview",
            "parameters": [
              {
                "name": "category",
                "in": "query",
                "required": true,
                "schema": {
                  "type": "string",
                  "enum": ["INTERVIEW", "GROUP_DISCUSSION"]
                }
              },
              {
                "name": "type",
                "in": "query",
                "required": true,
                "schema": {
                  "type": "string",
                  "enum": ["INTERVIEW", "GROUP_DISCUSSION", "ROLE_PLAY", "ORIENTATION"]
                }
              }
            ],
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "200": {
                "description": "OK",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewBasicDetailsDto"}
                    }
                  }
                }
              },
        "/admin/interview/masterData": {
          "get": {
            "tags": ["master-data-controller"],
            "operationId": "getInterviewMasterData",
            "parameters": [
              {
                "name": "keys",
                "in": "query",
                "required": true,
                "schema": {
                  "uniqueItems": true,
                  "type": "array",
                  "items": {
                    "type": "string",
                    "enum": ["INTERVIEWER", "JOB_PROFILE", "COMPETENCY", "DEFAULT_TIMEZONE", "TIMEZONE", "CANDIDATE"]
                  }
                }
              }
            ],
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "200": {
                "description": "OK",
                "content": {
                  "*/*": {
                    "schema": {
                      "type": "object",
                      "additionalProperties": {
                        "type": "object"
                      }
                    }
                  }
                }
              },
        "/admin/interview/emailTemplate/default": {
          "get": {
            "tags": ["interview-email-template-controller"],
            "operationId": "getDefaultInterviewEmailTemplateData",
            "parameters": [
              {
                "name": "emailTemplateKeys",
                "in": "query",
                "required": true,
                "schema": {
                  "uniqueItems": true,
                  "type": "array",
                  "items": {
                    "type": "string",
                    "enum": ["ONLINE_INTERVIEW_CANDIDATE_SCHEDULE", "ONLINE_INTERVIEW_CANDIDATE_RESCHEDULE", "ONLINE_INTERVIEW_CANDIDATE_CANCEL", "ONLINE_INTERVIEW_CANDIDATE_UPDATE", "ONLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_REMINDER", "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_INVITE", "ONLINE_INTERVIEW_INTERVIEWER_SCHEDULE", "ONLINE_INTERVIEW_INTERVIEWER_RESCHEDULE", "ONLINE_INTERVIEW_INTERVIEWER_CANCEL", "ONLINE_INTERVIEW_INTERVIEWER_UPDATE", "ONLINE"]
                  }
                }
              }
            ],
            "responses": {
              "500": {
                "description": "Internal Server Error",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "404": {
                "description": "Not Found",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "403": {
                "description": "Forbidden",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "401": {
                "description": "Unauthorized",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "400": {
                "description": "Bad Request",
                "content": {
                  "*/*": {
                    "schema": {
                      "$ref": "#/components/schemas/InterviewError"}
                    }
                  },
              "200": {
                "description": "OK",
                "content": {
                  "*/*": {
                    "schema": {
                      "type": "object",
                      "additionalProperties": {
                        "type": "object"
                      }
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```

## Pretty print

```

_WITH_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER", "OFFLINE_INTERVIEW_INTERVIEWER_SCHEDULE", "OFFLINE_INTERVIEW_INTERVIEWER_RESCHEDULE", "OFFLINE_INTERVIEW_INTERVIEWER_CANCEL", "OFFLINE_INTERVIEW_INTERVIEWER_UPDATE", "OFFLINE_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER", "OFFLINE_INTERVIEW_CANDIDATE_SCHEDULE", "OFFLINE_INTERVIEW_CANDIDATE_RESCHEDULE", "OFFLINE_INTERVIEW_CANDIDATE_CANCEL", "OFFLINE_INTERVIEW_CANDIDATE_UPDATE", "OFFLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_INTERVIEW_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_INTERVIEW_CLIENT_FINISHED_NOTIFICATION", "ONLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY", "CLIENT_JOB_PROFILE_CANDIDATE_EXCEL_DOWNLOAD", "CLIENT_SESSIONS_USAGE_EXCEL_DOWNLOAD", "ONLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL", "ONLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE", "ONLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_REMINDER", "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_INVITE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_GRADING_REMINDER", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_GRADING_REMINDER", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE", "OFFLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_GROUP_DISCUSSION_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_CLIENT_FINISHED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY", "ONLINE_ROLE_PLAY_CANDIDATE_SCHEDULE", "ONLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE", "ONLINE_ROLE_PLAY_CANDIDATE_CANCEL", "ONLINE_ROLE_PLAY_CANDIDATE_UPDATE", "ONLINE_ROLE_PLAY_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_REMINDER", "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_INVITE", "ONLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "ONLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE", "ONLINE_ROLE_PLAY_INTERVIEWER_CANCEL", "ONLINE_ROLE_PLAY_INTERVIEWER_UPDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER", "OFFLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "OFFLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE", "OFFLINE_ROLE_PLAY_INTERVIEWER_CANCEL", "OFFLINE_ROLE_PLAY_INTERVIEWER_UPDATE", "OFFLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER", "OFFLINE_ROLE_PLAY_CANDIDATE_SCHEDULE", "OFFLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE", "OFFLINE_ROLE_PLAY_CANDIDATE_CANCEL", "OFFLINE_ROLE_PLAY_CANDIDATE_UPDATE", "OFFLINE_ROLE_PLAY_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_FINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_ROLE_PLAY_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY"]}}},
{"name": "interviewUid", "in": "query", "required": true, "schema": {"type": "string"}}, {"responses": {"500": {"description": "Internal Server Error", "content": {"*/*: {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*/*: {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*/*: {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*/*: {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*/*: {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK", "content": {"application/json": {"schema": {"type": "object", "additionalProperties": {"$ref": "#/components/schemas/GetEmailTemplateDataResponse"}}}}}}, "/admin/interview/detail/{interviewUid}": {"get": {"tags": ["InterviewController"], "summary": "Fetch all details of an interview", "operationId": "getInterviewDetail_1", "parameters":

```

```

{"minLength":1,"type":"string"},"responses":{"500":{"description":"Internal Server Error","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"404":{"description":"Not Found","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"403":{"description":"Forbidden","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"401":{"description":"Unauthorized","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"400":{"description":"Bad Request","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"200":{"description":"OK","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewDto"}}}}}}},"/admin/interview/all_new":{"get":{"tags":["InterviewController"],"summary":"This is paginated api used to fetch all interview of a client","operationId":"getAllInterviews_new","parameters":[{"name":"interviewCriteriaJson","in":"query","required":true,"schema":{"type":"string"}}],"responses":{"500":{"description":"Internal Server Error","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"404":{"description":"Not Found","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"403":{"description":"Forbidden","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"401":{"description":"Unauthorized","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"400":{"description":"Bad Request","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"200":{"description":"OK","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewPage"}}}}}}}}},"/admin/info":{"get":{"tags":["admin-controller"],"operationId":"getAdminInfo","responses":{"500":{"description":"Internal Server Error","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"404":{"description":"Not Found","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"403":{"description":"Forbidden","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"401":{"description":"Unauthorized","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"400":{"description":"Bad Request","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"200":{"description":"OK","content":{"*/*":{"schema":{"$ref":"#/components/schemas/AdminDto"}}}}}}}}},"/admin/clientsInterviewSetting":{"get":{"tags":["admin-controller"],"operationId":"getClientInfo","parameters":[{"name":"mettlClientId","in":"query","required":true,"schema":{"type":"integer","format":"int64"}}],"responses":{"500":{"description":"Internal Server Error","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"404":{"description":"Not Found","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"403":{"description":"Forbidden","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"401":{"description":"Unauthorized","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"400":{"description":"Bad Request","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"200":{"description":"OK","content":{"*/*":{"schema":{"$ref":"#/components/schemas/ClientAccountSettingDto"}}}}}}}}},"/admin/candidate/all":{"get":{"tags":["CandidateController"],"operationId":"getAllCandidates","parameters":[{"name":"searchCriteriaJson","in":"query","required":true,"schema":{"type":"string"}}],"responses":{"500":{"description":"Internal Server Error","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"404":{"description":"Not Found","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"403":{"description":"Forbidden","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"401":{"description":"Unauthorized","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"400":{"description":"Bad Request","content":{"*/*":{"schema":{"$ref":"#/components/schemas/InterviewError"}}}},"200":{"description":"OK","content":{"*/*":{"schema":{"type":"array","items":{"$ref":"#/components/schemas/UserDto"}}}}}}}}},"/v1/interview/cancel/{interviewUuid}":{"delete":{"tags":["Interview APIs Controller"],"summary":"This API can be used to cancel an interview","operationId":"cancelInterview","parameters":

```

## Pretty print

```

interview": {"required": true, "schema": {"type": "string"}, "example": "5080dc71"},
{"name": "isSendEmail", "in": "query", "description": "Should send email", "required": true, "schema": {"type": "boolean"}, "example": true}, "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK"}}}, "/admin/interview/cancel/{interviewUuid}": {"delete": {"tags": ["InterviewController"], "summary": "This api will cancel an interview", "operationId": "cancelInterview_1", "parameters": [{"name": "interviewUuid", "in": "path", "required": true, "schema": {"type": "string"}}, {"name": "isSendEmail", "in": "query", "required": true, "schema": {"type": "boolean"}}], "responses": {"500": {"description": "Internal Server Error", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "404": {"description": "Not Found", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "403": {"description": "Forbidden", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "401": {"description": "Unauthorized", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "400": {"description": "Bad Request", "content": {"*//*": {"schema": {"$ref": "#/components/schemas/InterviewError"}}}}, "200": {"description": "OK"}}}}, "components": {"schemas": {"AdminDto": {"type": "object", "properties": {"email": {"type": "string"}, "timeZone": {"type": "string"}, "client": {"$ref": "#/components/schemas/ClientDto"}, "uiSetting": {"$ref": "#/components/schemas/AdminUiSetting"}}, "AdminUiSetting": {"type": "object", "properties": {"showSignupPage": {"type": "boolean"}, "showWelcomePage": {"type": "boolean"}, "language": {"type": "string"}}, "CandidateBasicDetails": {"type": "object", "properties": {"firstName": {"type": "string", "description": "First name of candidate", "example": "John"}, "lastName": {"type": "string", "description": "Last name of candidate", "example": "Doe"}, "email": {"type": "string", "description": "Email of candidate", "example": "john.doe@example.com"}, "participantUuid": {"type": "string", "description": "Unique id of candidate", "example": "7ce6643b-e287-44c3-93ca-5d31ce8dc56d"}, "participantStatus": {"type": "string", "description": "Current status of candidate", "example": "INVITED", "enum": ["INVITED", "WAITING", "JOINED_INTERVIEW", "LEFT_INTERVIEW", "FEEDBACK_SUBMITTED", "AWAITING_FEE", "NOT_ACTIVE", "DENIED", "ALLOWED", "DISCONNECTED", "LEFT_LOBBY"]}, "resume": {"$ref": "#/components/schemas/FileDetailsResponse"}, "description": "List of candidates in the interview"}, "CandidateCompetencyEvaluationSummary": {"type": "object", "properties": {"competencyName": {"type": "string", "description": "Name of the competency", "example": "Problem Solving"}, "averageScore": {"type": "number", "description": "Average score obtained by candidate", "format": "float", "example": 12.3}, "maxScore": {"type": "number", "description": "Max score of the competency", "format": "float", "example": 20}, "scoringType": {"type": "string", "description": "Scoring type of competency", "example": "STAR", "enum": ["STAR", "SCORE"]}, "interviewerCompetencyFeedbackDtoList": {"type": "array", "items": {"$ref": "#/components/schemas/InterviewerCompetencyFeedbackResponse"}}, "notAssessed": {"type": "boolean", "description": "If the competency is not assessed by all interviewers", "example": false}, "candidateSubCompetencyEvaluationSummaryDtoList": {"type": "array", "description": "Evaluation summary for sub competencies if there are any", "items": {"$ref": "#/components/schemas/CandidateCompetencyEvaluationSummary"}}}}, "CandidateCompetencyEvaluationSummaryDto": {"type": "object", "properties": {"competencyName": {"type": "string"}, "averageScore": {"type": "number", "format": "float"}, "maxScore": {"type": "number", "format": "float"}, "scoringType": {"type": "string", "enum": ["STAR", "SCORE"]}, "interviewerCompetencyFeedbackDtoList": {"type": "array", "items": {"$ref": "#/components/schemas/InterviewerCompetencyFeedbackDto"}}, "notAssessed": {"type": "boolean"}, "candidateSubCompetencyEvaluationSummaryDtoList": {"type": "array", "items": {"$ref": "#/components/schemas/CandidateCompetencyEvaluationSummaryDto"}}}}, "CandidateDto": {"type": "object", "properties": {"uuid": {"type": "string"}, "status": {"type": "string", "enum": ["INVITED", "WAITING", "JOINED_INTERVIEW", "LEFT_INTERVIEW", "FEEDBACK_SUBMITTED", "AWAITING_FEE", "NOT_ACTIVE", "DENIED", "ALLOWED", "DISCONNECTED", "LEFT_LOBBY"]}, "user":

```



```

{"type": "boolean"}, "candidate": {"type": "boolean"}, "platformFeedbackSubmitted": {"type": "boolean"}, "lastActiveTime": {"type": "integer", "format": "int64"}, "networkInfo": {"type": "integer", "format": "int64"}, "joinedOnce": {"type": "boolean"}, "joinedDuration": {"type": "integer", "format": "int64"}, "joinedLobby": {"type": "boolean"}, "inviteReminderCount": {"type": "integer", "format": "int32"}, "evaluationReminderCount": {"type": "integer", "format": "int32"}, "userStatusAtInterviewEnd": {"type": "string", "enum": ["ENABLED", "DISABLED"]}, "resumeUrl": {"type": "string"}, "resumeFileName": {"type": "string"}, "resumeFileUuid": {"type": "string"}, "candidateStatus": {"type": "string", "enum": ["SHORTLISTED", "REJECTED", "ON_HOLD", "NOT_EVALUATED"]}, "averageMarksObtained": {"type": "number", "format": "float"}, "maxMarks": {"type": "number", "format": "float"}, "notAssessed": {"type": "boolean"}, "averageMarksObtainedPercentage": {"type": "number", "format": "float"}, "fullName": {"type": "string"}, "invitedOrJoined": {"type": "boolean"}}, "CandidateEvaluationSummaryDto": {"type": "object", "properties": {"interviewerUuid": {"type": "string"}, "interviewerName": {"type": "string"}, "interviewerStatus": {"type": "string", "enum": ["INVITED", "WAITING", "JOINED_INTERVIEW", "LEFT_INTERVIEW", "FEEDBACK_SUBMITTED", "AWAITING_FEE", "DBACK", "NOT_ACTIVE", "DENIED", "ALLOWED", "DISCONNECTED", "LEFT_LOBBY"]}, "interviewerJoinedDuration": {"type": "integer", "format": "int64"}, "evaluationResult": {"type": "string", "enum": ["RECOMMENDED", "NOT_SURE", "NOT_RECOMMENDED", "NOT_EVALUATED"]}, "feedback": {"type": "string"}, "updatedOn": {"type": "integer", "format": "int64"}, "evaluationReminderCount": {"type": "integer", "format": "int32"}, "userStatus": {"type": "string", "enum": ["ENABLED", "DISABLED"]}, "userStatusAtInterviewEnd": {"type": "string", "enum": ["ENABLED", "DISABLED"]}, "skipEvaluation": {"type": "boolean"}, "skipEvaluationReason": {"type": "string"}, "invited": {"type": "boolean"}, "joinedOnce": {"type": "boolean"}}, "CandidateEvaluationSummaryResponse": {"type": "object", "properties": {"interviewerUuid": {"type": "string"}, "description": "Unique id of interview", "example": "7e30d29a"}, "interviewerName": {"type": "string"}, "description": "Name of interviewer", "example": "Interviewer 1"}, "interviewerStatus": {"type": "string"}, "description": "Interviewer status in interview", "example": "FEEDBACK_SUBMITTED", "enum": ["INVITED", "WAITING", "JOINED_INTERVIEW", "LEFT_INTERVIEW", "FEEDBACK_SUBMITTED", "AWAITING_FEE", "DBACK", "NOT_ACTIVE", "DENIED", "ALLOWED", "DISCONNECTED", "LEFT_LOBBY"]}, "interviewerJoinedDuration": {"type": "integer", "description": "Joined duration interviewer in minutes", "format": "int64", "example": 30}, "evaluationResult": {"type": "string", "description": "Evaluation result by interviewer", "example": "RECOMMENDED", "enum": ["RECOMMENDED", "NOT_SURE", "NOT_RECOMMENDED", "NOT_EVALUATED"]}, "feedback": {"type": "string", "description": "Feedback by interviewer", "example": "Feedback by interviewer"}, "skipEvaluation": {"type": "boolean", "description": "If the evaluation is skipped by interviewer", "example": false}, "skipEvaluationReason": {"type": "string", "description": "Reason for skipping evaluation", "example": "Reason for skipping evaluation"}}, "CandidateFeedbackAndCompetencyDetail": {"type": "object", "properties": {"candidate": {"$ref": "#/components/schemas/CandidateBasicDetails"}, "feedback": {"$ref": "#/components/schemas/FeedbackSummaryDto"}, "competenciesEvaluation": {"type": "array", "items": {"$ref": "#/components/schemas/CandidateCompetencyEvaluationSummaryDto"}}}}, "CandidateFeedbackAndResultSummary": {"type": "object", "properties": {"candidate": {"$ref": "#/components/schemas/CandidateBasicDetails"}, "feedback": {"$ref": "#/components/schemas/FeedbackSummaryResponse"}, "candidateResultSummaryResponse": {"$ref": "#/components/schemas/CandidateResultSummaryResponse"}}, "CandidateInfo": {"required": ["email", "firstName"], "type": "object", "properties": {"email": {"type": "string", "description": "Email of the user.\nMax length: 200 characters", "example": "John.doe@example.com"}, "firstName": {"type": "string", "description": "First name of the user.\nMax length: 100 characters", "example": "John"}, "lastName": {"type": "string", "description": "Last name of the user.\nMax length: 100 characters", "example": "Doe"}}, "description": "List of candidates for the interview"}, "CandidateInfoDto": {"required": ["userId"], "type": "object", "properties": {"userId": {"minLength": 1, "type": "string"}, "resumeFileUuid": {"type": "string"}}, "CandidateReportDto": {"type": "object", "properties": {"reportType": {"type": "string", "enum": ["METTL_ASSESSMENT", "INTERVIEW"]}, "reportName": {"type": "string"}, "submitTime": {"type": "integer", "format": "int64"}, "shareUrl": {"type": "string"}, "downloadUrl": {"type": "string"}, "interviewUuid": {"type": "string"}, "sessionCategory": {"type": "string", "enum": ["INTERVIEW", "GROUP_DISCUSSION"]}, "interviewType": {"type": "string", "enum":

```

## Pretty print

```
{
  "type": "object",
  "additionalProperties": {
    "type": "string"
  },
  "interviewerList": {
    "type": "array",
    "items": {
      "type": "string"
    }
  },
  "CandidateResponse": {
    "type": "object",
    "properties": {
      "uuid": {
        "type": "string",
        "description": "Unique id of the participant",
        "example": "d8ccc0a"
      },
      "status": {
        "type": "string",
        "description": "Current status of the participant",
        "example": "INVITED",
        "enum": [
          "INVITED",
          "WAITING",
          "JOINED_INTERVIEW",
          "LEFT_INTERVIEW",
          "FEEDBACK_SUBMITTED",
          "AWAITING_FEE",
          "DBACK",
          "NOT_ACTIVE",
          "DENIED",
          "ALLOWED",
          "DISCONNECTED",
          "LEFT_LOBBY"
        ]
      },
      "user": {
        "$ref": "#/components/schemas/UserResponse",
        "invited": {
          "type": "boolean",
          "description": "If the participant is invited for the interview",
          "example": true
        },
        "joinedOnce": {
          "type": "boolean",
          "description": "If the participant has joined once in the interview",
          "example": true
        },
        "joinedDuration": {
          "type": "integer",
          "description": "Duration of participant in the interview",
          "format": "int64",
          "example": 17465881
        },
        "resumeUrl": {
          "type": "string",
          "description": "Url of the uploaded candidate resume",
          "example": "https://resume-url.com",
          "resumeFileName": {
            "type": "string",
            "description": "Name of the uploaded candidate resume",
            "example": "john_resume.pdf"
          },
          "candidateStatus": {
            "type": "string",
            "description": "Current status of the candidate",
            "example": "SHORTLISTED",
            "enum": [
              "SHORTLISTED",
              "REJECTED",
              "ON_HOLD",
              "NOT_EVALUATED"
            ]
          },
          "averageMarksObtained": {
            "type": "number",
            "description": "Average marks obtained by candidate",
            "format": "float",
            "example": 78.3
          },
          "maxMarks": {
            "type": "number",
            "description": "Max marks",
            "format": "float",
            "example": 100
          },
          "notAssessed": {
            "type": "boolean",
            "description": "Flag telling if the candidate is not assessed",
            "example": true
          }
        },
        "CandidateResultSummary": {
          "type": "object",
          "properties": {
            "candidate": {
              "$ref": "#/components/schemas/CandidateBasicDetails",
              "status": {
                "type": "string",
                "enum": [
                  "SHORTLISTED",
                  "REJECTED",
                  "ON_HOLD",
                  "NOT_EVALUATED"
                ]
              },
              "recommendationCounts": {
                "type": "object",
                "additionalProperties": {
                  "type": "integer",
                  "format": "int64"
                },
                "totalAverageScore": {
                  "type": "number",
                  "format": "float"
                },
                "totalMaxScore": {
                  "type": "number",
                  "format": "float"
                },
                "notAssessed": {
                  "type": "boolean"
                },
                "competencySummaryList": {
                  "type": "array",
                  "items": {
                    "$ref": "#/components/schemas/CandidateCompetencyEvaluationSummaryDto"
                  }
                },
                "joinedOnce": {
                  "type": "boolean"
                },
                "joinedLobby": {
                  "type": "boolean"
                },
                "participantStatus": {
                  "type": "string",
                  "enum": [
                    "INVITED",
                    "WAITING",
                    "JOINED_INTERVIEW",
                    "LEFT_INTERVIEW",
                    "FEEDBACK_SUBMITTED",
                    "AWAITING_FEE",
                    "DBACK",
                    "NOT_ACTIVE",
                    "DENIED",
                    "ALLOWED",
                    "DISCONNECTED",
                    "LEFT_LOBBY"
                  ]
                }
              },
              "CandidateResultSummaryResponse": {
                "type": "object",
                "properties": {
                  "totalAverageScore": {
                    "type": "number",
                    "description": "Total average score obtained by Candidate",
                    "format": "float",
                    "example": 76.3
                  },
                  "totalMaxScore": {
                    "type": "number",
                    "description": "Total max score",
                    "format": "float",
                    "example": 100
                  },
                  "notAssessed": {
                    "type": "boolean",
                    "description": "Whether evaluations are not assessed by interviewers",
                    "example": false
                  },
                  "competencySummaryList": {
                    "type": "array",
                    "items": {
                      "$ref": "#/components/schemas/CandidateCompetencyEvaluationSummary"
                    }
                  },
                  "description": "If there is any job profile in interview, then this will provide overall score and summary of evaluations provided by all interviewers for competencies",
                  "ChartData": {
                    "type": "object",
                    "properties": {
                      "title": {
                        "type": "string"
                      },
                      "data": {
                        "type": "object",
                        "additionalProperties": {
                          "type": "integer",
                          "format": "int64"
                        }
                      }
                    }
                  },
                  "ClientAccountSettingDto": {
                    "required": [
                      "clientEmail"
                    ],
                    "type": "object",
                    "properties": {
                      "clientEmail": {
                        "minLength": 1,
                        "type": "string"
                      },
                      "enableAdminPanel": {
                        "type": "boolean"
                      },
                      "enableResubmission": {
                        "type": "boolean"
                      },
                      "meetingType": {
                        "type": "string",
                        "enum": [
                          "ZOOM",
                          "TOKBOX"
                        ]
                      },
                      "enableMeetingRecording": {
                        "type": "boolean"
                      }
                    }
                  },
                  "ClientConsentDto": {
                    "type": "object",
                    "properties": {
                      "category": {
                        "type": "string",
                        "enum": [
                          "ONLINE_INTERVIEW_CANDIDATE_NON_RECORDED",
                          "ONLINE_INTERVIEW_CANDIDATE_RECORDED",
                          "ONLINE_IN",
                          "TERVIEW_INTERVIEWER_NON_RECORDED",
                          "ONLINE_INTERVIEW_INTERVIEWER_RECORDED",
                          "ONLINE_GROUP_DIS",
                          "CUSSION_CANDIDATE_NON_RECORDED",
                          "ONLINE_GROUP_DISCUSSION_CANDIDATE_RECORDED",
                          "ONLINE_GROUP",
                          "DISCUSSION_INTERVIEWER_NON_RECORDED",
                          "ONLINE_GROUP_DISCUSSION_INTERVIEWER_RECORDED",
                          "OFFLIN",
                          "E_INTERVIEW_INTERVIEWER_NON_RECORDED",
                          "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_NON_RECORDED"
                        ]
                      },
                      "language": {
                        "type": "string"
                      },
                      "consentDetailList": {
                        "type": "array",
                        "items": {
                          "$ref": "#/components/schemas/ConsentDetailDto"
                        }
                      },
                      "createdOn": {
                        "type": "integer",
                        "format": "int64"
                      },
                      "updatedOn": {
                        "type": "integer",
                        "format": "int64"
                      }
                    }
                  },
                  "ClientDto": {
                    "type": "object",
                    "properties": {
                      "email": {
                        "type": "string"
                      },
                      "mettlClientId": {
                        "type": "integer",
                        "format": "int64"
                      },
                      "logoUrl": {
                        "type": "string"
                      },
                      "name": {
                        "type": "string"
                      },
                      "organisationName": {
                        "type": "string"
                      },
                      "favIconUrl": {
                        "type": "string"
                      },
                      "primarySupportNumber": {
                        "type": "string"
                      },
                      "secondarySupportNumber": {
                        "type": "string"
                      },
                      "allowInterviewSimulatorApp": {
                        "type": "boolean"
                      },
                      "accountSetting": {

```

```
{
  "required": [
    "emailSenderType", "type", "object", "properties": {
      "sendParticipantReminder": {
        "type": "boolean", "participantReminderIntervalInMin": {
          "type": "integer", "format": "int32", "sendAdminNotification": {
            "type": "boolean", "adminNotificationOnSessionEvaluated": {
              "type": "boolean", "adminNotificationOnSessionAutofinished": {
                "type": "boolean", "adminNotificationOnSessionFinished": {
                  "type": "boolean", "emailSenderType": {
                    "type": "string", "enum": ["GENERIC", "CLIENT", "ADMIN", "CUSTOM"]}, "customSenderName": {
                      "type": "string", "customSenderEmail": {
                        "type": "string"}, "ClientReportSettingDto": {
                          "type": "object", "properties": {
                            "showSessionRecording": {
                              "type": "boolean", "showSessionLogs": {
                                "type": "boolean", "showCandidateTimeline": {
                                  "type": "boolean", "allowDownloadingPublicReports": {
                                    "type": "boolean", "allowDownloadingSessionRecording": {
                                      "type": "boolean", "ClientUsageChartData": {
                                        "type": "object", "properties": {
                                          "dailyChartDataList": {
                                            "type": "array", "items": {
                                              "$ref": "#/components/schemas/SessionDataPoint"}}, "ClientUsageCountData": {
                                                "type": "object", "properties": {
                                                  "totalSessionScheduled": {
                                                    "type": "integer", "format": "int64", "totalSessionConducted": {
                                                      "type": "integer", "format": "int64", "totalParticipants": {
                                                        "type": "integer", "format": "int64"}}, "ClientUseCaseSettingDto": {
                                                          "type": "object", "properties": {
                                                            "allowedQuestionTypes": {
                                                              "uniqueItems": true, "type": "array", "items": {
                                                                "type": "string", "enum": ["CODING", "WHITEBOARD", "DIAGRAM", "TEXT_EDITOR", "METTL_QUESTION"]}}, "enableCandidateProfile": {
                                                                  "type": "boolean", "enablePreviousSessionReports": {
                                                                    "type": "boolean", "showOverallRecommendation": {
                                                                      "type": "boolean", "enableCopyPaste": {
                                                                        "type": "boolean", "openInVideoMode": {
                                                                          "type": "boolean", "enableInvite": {
                                                                            "type": "boolean", "enableEvaluationPanel": {
                                                                              "type": "boolean", "enableRecording": {
                                                                                "type": "boolean", "language": {
                                                                                  "type": "string"}, "CodelysisLanguageCompilerVersion": {
                                                                                    "type": "object", "properties": {
                                                                                      "name": {
                                                                                        "type": "string", "compilerVersion": {
                                                                                          "type": "string"}, "CompetencyDto": {
                                                                                            "required": ["name"], "type": "object", "properties": {
                                                                                              "uuid": {
                                                                                                "type": "string", "name": {
                                                                                                  "maxLength": 100, "minLength": 0, "type": "string", "description": {
                                                                                                    "maxLength": 1000, "minLength": 0, "type": "string", "createdOn": {
                                                                                                      "type": "integer", "format": "int64", "scoringType": {
                                                                                                        "type": "string", "enum": ["STAR", "SCORE"]}, "marks": {
                                                                                                          "type": "number", "format": "float", "subCompetencies": {
                                                                                                            "type": "array", "items": {
                                                                                                              "$ref": "#/components/schemas/CompetencyDto"}}, "calculationMethod": {
                                                                                                                "type": "string", "enum": ["SUM", "AVERAGE"]}, "type": {
                                                                                                                  "type": "string", "enum": ["ADMIN", "SELF"]}}}, "ConsentDetailDto": {
                                                                                                                    "type": "object", "properties": {
                                                                                                                      "uuid": {
                                                                                                                        "type": "string", "consentText": {
                                                                                                                          "type": "string", "inputRequired": {
                                                                                                                            "type": "boolean", "optional": {
                                                                                                                              "type": "boolean", "createdOn": {
                                                                                                                                "type": "integer", "format": "int64"}}, "ConsentDetailRequest": {
                                                                                                                                              "required": ["consentText", "inputRequired"], "type": "object", "properties": {
                                                                                                                                                            "consentText": {
                                                                                                                                                              "maxLength": 15000, "minLength": 1, "type": "string", "inputRequired": {
                                                                                                                                                                "type": "boolean", "optional": {
                                                                                                                                                                  "type": "boolean"}}}, "CopyEmailTemplateToAccountRequest": {
                                                                                                                                                                  "required": ["destinationAccountEmail", "language", "mode", "sessionCategoryAndUseCaseAndEmailCategoryDataList", "sourceAccountEmail"], "type": "object", "properties": {
                                                                                                                                                                      "sourceAccountEmail": {
                                                                                                                                                                        "minLength": 1, "type": "string", "destinationAccountEmail": {
                                                                                                                                                                          "minLength": 1, "type": "string", "language": {
                                                                                                                                                                            "type": "string", "enum": ["english", "european_portuguese", "european_spanish", "french", "german", "greek", "italian", "polish", "portuguese_brasil"]}, "mode": {
                                                                                                                                                                              "type": "string", "enum": ["ONLINE", "OFFLINE"]}, "sessionCategoryAndUseCaseAndEmailCategoryDataList": {
                                                                                                                                                                                "minItems": 1, "type": "array", "items": {
                                                                                                                                                                                  "$ref": "#/components/schemas/SessionCategoryAndUseCaseAndEmailCategoryData"}}, "CopyEmailTemplateToGlobalRequest": {
                                                                                                                                                                                  "required": ["language", "mode", "sessionCategoryAndUseCaseAndEmailCategoryDataList", "sourceAccountEmail"], "type": "object", "properties": {
                                                                                                                                                                                      "sourceAccountEmail": {
                                                                                                                                                                                        "minLength": 1, "type": "string", "language": {
                                                                                                                                                                                          "type": "string", "enum": ["english", "european_portuguese", "european_spanish", "french", "german", "greek", "italian", "polish", "portuguese_brasil"]}, "mode": {
                                                                                                                                                                                            "type": "string", "enum": ["ONLINE", "OFFLINE"]}, "sessionCategoryAndUseCaseAndEmailCategoryDataList": {
                                                                                                                                                                                              "minItems": 1, "type": "array", "items": {
                                                                                                                                                                                                "$ref": "#/components/schemas/SessionCategoryAndUseCaseAndEmailCategoryData"}}, "CreateAndScheduleInterviewResponse": {
                                                                                                                                                                                                "type": "object", "properties": {
                                                                                                                                                                                                  "uuid": {
                                                                                                                                                                                                    "type": "string", "pin": {
                                                                                                                                                                                                      "type": "string", "participants": {
                                                                                                                                                                                                        "type": "array", "items": {
                                                                                                                                                                                                          "$ref": "#/components/schemas/UserWithLinkDto"}}, "DynamicKeyData": {
                                                                                                                                                                                                          "type": "object", "properties": {
                                                                                                                                                                                                            "key": {
                                                                                                                                                                                                              "type": "string", "description": {
                                                                                                                                                                                                                "type": "string", "descriptionI18nKey": {
                                                                                                                                                                                                                  "type": "string"}}, "EditInterviewApiRequest": {
```

## Pretty print

```
{
  "interviewUuid": {
    "type": "string",
    "description": "Unique identifier for the interview you want to edit.",
    "example": "4de94450",
    "name": {
      "type": "string",
      "description": "Name of the interview. Only applicable when your interview type is 'GROUP_DISCUSSION'",
      "example": "Interview with john",
      "interviewers": {
        "type": "array",
        "description": "List of interviewers for the interview",
        "items": {
          "$ref": "#/components/schemas/InterviewerInfo"
        },
        "candidates": {
          "type": "array",
          "description": "List of candidates for the interview",
          "items": {
            "$ref": "#/components/schemas/CandidateInfo"
          },
          "jobProfileUuid": {
            "type": "string",
            "description": "Unique id of job profile",
            "example": "7ce6643b-e287-44c3-93ca-5d31ce8dc56d",
            "settings": {
              "$ref": "#/components/schemas/InterviewSettingRequest"
            }
          },
          "EditInterviewRequest": {
            "required": ["candidates", "interviewUuid", "interviewerUids"],
            "type": "object",
            "properties": {
              "interviewUuid": {
                "minLength": 1,
                "type": "string",
                "name": {
                  "type": "string",
                  "interviewerUids": {
                    "maxItems": 5,
                    "minItems": 1,
                    "type": "array",
                    "items": {
                      "type": "string"
                    },
                    "candidates": {
                      "type": "array",
                      "items": {
                        "type": "string"
                      },
                      "jobProfileUuid": {
                        "type": "string",
                        "settings": {
                          "$ref": "#/components/schemas/InterviewSettingsDto"
                        }
                      },
                      "EditInterviewRequestV2": {
                        "required": ["candidates", "interviewUuid", "interviewers"],
                        "type": "object",
                        "properties": {
                          "candidates": {
                            "type": "array",
                            "items": {
                              "$ref": "#/components/schemas/UserDto"
                            },
                            "interviewUuid": {
                              "minLength": 1,
                              "type": "string",
                              "interviewers": {
                                "maxItems": 5,
                                "minItems": 1,
                                "type": "array",
                                "items": {
                                  "$ref": "#/components/schemas/UserDto"
                                },
                                "jobProfileUuid": {
                                  "type": "string",
                                  "name": {
                                    "type": "string",
                                    "settings": {
                                      "$ref": "#/components/schemas/InterviewSettingsDto"
                                    }
                                  },
                                  "EmailSenderDetailsDto": {
                                    "type": "object",
                                    "properties": {
                                      "emailSenderOptions": {
                                        "type": "array",
                                        "items": {
                                          "$ref": "#/components/schemas/EmailSenderOptionDto"
                                        },
                                          "EmailSenderOptionDto": {
                                            "type": "object",
                                            "properties": {
                                              "type": {
                                                "type": "string",
                                                "enum": ["GENERIC", "CLIENT", "ADMIN", "CUSTOM"],
                                                "senderName": {
                                                  "type": "string",
                                                  "senderEmail": {
                                                    "type": "string"
                                                  },
                                                  "EmailTemplateData": {
                                                    "type": "object",
                                                    "properties": {
                                                      "subject": {
                                                        "type": "string",
                                                        "body": {
                                                          "type": "string",
                                                          "calendarInviteTitle": {
                                                            "type": "string",
                                                            "key": {
                                                              "type": "string",
                                                              "enum": [
                                                                "ONLINE_INTERVIEW_CANDIDATE_SCHEDULE",
                                                                "ONLINE_INTERVIEW_CANDIDATE_RESCHEDULE",
                                                                "ONLINE_INTERVIEW_CANDIDATE_CANCEL",
                                                                "ONLINE_INTERVIEW_CANDIDATE_UPDATE",
                                                                "ONLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER",
                                                                "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_REMINDER",
                                                                "ONLINE_INTERVIEW_CANDIDATE_ON_GOING_INVITE",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_SCHEDULE",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_RESCHEDULE",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_CANCEL",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_UPDATE",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER",
                                                                "OFFLINE_INTERVIEW_INTERVIEWER_SCHEDULE",
                                                                "OFFLINE_INTERVIEW_INTERVIEWER_RESCHEDULE",
                                                                "OFFLINE_INTERVIEW_INTERVIEWER_CANCEL",
                                                                "OFFLINE_INTERVIEW_INTERVIEWER_UPDATE",
                                                                "OFFLINE_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER",
                                                                "OFFLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER",
                                                                "OFFLINE_INTERVIEW_CANDIDATE_SCHEDULE",
                                                                "OFFLINE_INTERVIEW_CANDIDATE_RESCHEDULE",
                                                                "OFFLINE_INTERVIEW_CANDIDATE_CANCEL",
                                                                "OFFLINE_INTERVIEW_CANDIDATE_UPDATE",
                                                                "OFFLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER",
                                                                "ONLINE_INTERVIEW_CLIENT_AUTOFINISHED_NOTIFICATION",
                                                                "ONLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION",
                                                                "ONLINE_INTERVIEW_CLIENT_FINISHED_NOTIFICATION",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY",
                                                                "ONLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY",
                                                                "OFFLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICATION",
                                                                "OFFLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY",
                                                                "OFFLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY",
                                                                "CLIENT_JOB_PROFILE_CANDIDATE_EXCEL_DOWNLOAD",
                                                                "CLIENT_SESSIONS_USAGE_EXCEL_DOWNLOAD",
                                                                "ONLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE",
                                                                "ONLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE",
                                                                "ONLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL",
                                                                "ONLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE",
                                                                "ONLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER",
                                                                "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_REMINDER",
                                                                "ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_INVITE",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE",
                                                                "ONLINE_GROUP_DISCUSSION_INTERVIEWER_GRADING_REMINDER",
                                                                "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE",
                                                                "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE",
                                                                "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL",
                                                                "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE",
                                                                "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER",
                                                                "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_GRADING_REMINDER",
                                                                "OFFLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE",
                                                                "OFFLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE",
                                                                "OFFLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL",
                                                                "OFFLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE",
                                                                "OFFLINE_GROUP_DISCUSSION_CAND"

```

```

"GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_CLIENT_FINISHED_NOTIFICATION", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_NOTES_SUMMARY", "ONLINE_ROLE_PLAY_CANDIDATE_SCHEDULE", "ONLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE", "ONLINE_ROLE_PLAY_CANDIDATE_CANCEL", "ONLINE_ROLE_PLAY_CANDIDATE_UPDATE", "ONLINE_ROLE_PLAY_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_REMINDER", "ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_INVITE", "ONLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "ONLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE", "ONLINE_ROLE_PLAY_INTERVIEWER_CANCEL", "ONLINE_ROLE_PLAY_INTERVIEWER_UPDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE", "ONLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER", "OFFLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE", "OFFLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE", "OFFLINE_ROLE_PLAY_INTERVIEWER_CANCEL", "OFFLINE_ROLE_PLAY_INTERVIEWER_UPDATE", "OFFLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER", "OFFLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER", "OFFLINE_ROLE_PLAY_CANDIDATE_SCHEDULE", "OFFLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE", "OFFLINE_ROLE_PLAY_CANDIDATE_CANCEL", "OFFLINE_ROLE_PLAY_CANDIDATE_UPDATE", "OFFLINE_ROLE_PLAY_CANDIDATE_AUTOMATED_REMINDER", "ONLINE_ROLE_PLAY_CLIENT_AUTOFINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_EVALUATED_NOTIFICATION", "ONLINE_ROLE_PLAY_CLIENT_FINISHED_NOTIFICATION", "ONLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY", "ONLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY", "OFFLINE_ROLE_PLAY_CLIENT_EVALUATED_NOTIFICATION", "OFFLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY", "OFFLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY"]}, "mode": {"type": "string", "enum": ["ONLINE", "OFFLINE"]}, "type": {"type": "string", "enum": ["INTERVIEW", "GROUP_DISCUSSION", "ROLE_PLAY", "ORIENTATION"]}, "language": {"type": "string", "enum": ["english", "european_portuguese", "european_spanish", "french", "german", "greek", "italian", "polish", "portuguese_brasil"]}, "userType": {"type": "string", "enum": ["INTERVIEWER", "CANDIDATE", "CLIENT"]}, "category": {"type": "string", "enum": ["SCHEDULE", "RESCHEDULE", "CANCEL", "UPDATE", "ON_GOING_REMINDER", "ON_GOING_REMINDER_WITH_CANDIDATE", "ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ON_GOING_INVITE", "ON_GOING_INVITE_WITH_CANDIDATE", "ON_GOING_INVITE_WITHOUT_CANDIDATE", "AUTOMATED_REMINDER", "GRADING_REMINDER", "AUTOFINISHED_NOTIFICATION", "EVALUATED_NOTIFICATION", "FINISHED_NOTIFICATION", "EVALUATION_SUMMARY", "NOTES_SUMMARY", "JOB_PROFILE_CANDIDATE_EXCEL_DOWNLOAD", "SESSIONS_USAGE_EXCEL_DOWNLOAD"]}}, "EnableResubmissionRequest": {"required": ["interviewUuid", "interviewers"], "type": "object", "properties": {"interviewUuid": {"minLength": 1, "type": "string"}, "interviewers": {"maxItems": 5, "minItems": 1, "type": "array", "items": {"type": "string"}}, "requesterEmail": {"type": "string"}, "requesterName": {"type": "string"}}, "FeedbackSummaryDto": {"type": "object", "properties": {"candidateStatus": {"type": "string", "enum": ["SHORTLISTED", "REJECTED", "ON_HOLD", "NOT_EVALUATED"]}, "candidateEvaluationSummaryDtoList": {"type": "array", "items": {"$ref": "#/components/schemas/CandidateEvaluationSummaryDto"}}, "FeedbackSummaryResponse": {"type": "object", "properties": {"candidateStatus": {"type": "string", "description": "Status of candidate", "example": "SHORTLISTED", "enum": ["SHORTLISTED", "REJECTED", "ON_HOLD", "NOT_EVALUATED", "SHORTLISTED", "REJECTED", "ON_HOLD", "NOT_EVALUATED"]}, "evaluationResultVsCount": {"type": "object", "additionalProperties": {"type": "integer", "description": "Recommendation by interviewer counts", "format": "int64"}, "description": "Recommendation by interviewer counts", "example": "{\n  \"RECOMMENDED\": 1,\n  \"NOT_RECOMMENDED\": 0,\n  \"NOT_SURE\": 1,\n  \"NOT_EVALUATED\": 1,\n}"}}, "candidateEvaluationSummaryDtoList": {"type": "array", "items": {"$ref": "#/components/schemas/CandidateEvaluationSummaryResponse"}}, "description": "This will provide summary of overall evaluation provided by all interviewers"}, "FetchAllInterviewerRequest": {"required": ["clientEmail", "interviewUuid"], "type": "object", "properties": {"clientEmail": {"minLength": 1, "type": "string"}, "interviewUuid": {"minLength": 1, "type": "string"}}, "FetchAllInterviewerResponse": {"type": "object", "properties": {"interviewUuid": {"type": "string"}, "interviewerList": {"type": "array", "items": {"$ref": "#/components/schemas/InterviewParticipantDto"}}, "FetchClientAccountSettingRequest": {"required": ["clientEmail"], "type": "object", "properties": {"clientEmail": {"minLength": 1, "type": "string"}}, "FileDetailsDto": {"type": "object", "properties": {"uuid": {"type": "string"}, "name": {"type": "string"}, "url": {"type": "string"}}, "FileDetailsResponse": {"type": "object", "properties": {"uuid": {"type": "string", "description": "Unique id of file", "example": "11239494-b5f0-440c-ab97-d4748ae7fd9b"}, "name": {"type": "string", "description": "Name of the file", "example": "candidate-resume.pdf"}, "url": {"type": "string", "description": "Url of the file", "example": "https://file-

```

## Pretty print

```
{
  "type": "string",
  "uploadKey": {
    "type": "string",
    "filename": {
      "type": "string",
      "proxyUrl": {
        "type": "string",
        "fileSizeInB": {
          "type": "integer",
          "format": "int64"
        }
      },
      "FileUploadRequest": {
        "required": [
          "file"
        ],
        "type": "object",
        "properties": {
          "file": {
            "type": "string",
            "format": "binary"
          }
        }
      },
      "GetEmailTemplateDataResponse": {
        "type": "object",
        "properties": {
          "fromEmailAddress": {
            "type": "string"
          },
          "emailTemplateData": {
            "$ref": "#/components/schemas/EmailTemplateData"
          },
          "dynamicKeyDataList": {
            "type": "array",
            "items": {
              "$ref": "#/components/schemas/DynamicKeyData"
            }
          }
        }
      },
      "InterviewActivityLogSummaryDto": {
        "type": "object",
        "properties": {
          "createdOn": {
            "type": "integer",
            "format": "int64"
          },
          "eventType": {
            "type": "string",
            "enum": [
              "PARTICIPANT_STATUS_EVENT",
              "QUESTION_STATUS_EVENT",
              "INTERVIEW_LOCK_EVENT",
              "INTERVIEW_STATUS_EVENT",
              "PARTICIPANT_DEVIATION_EVENT",
              "PARTICIPANT_REMINDER_EVENT",
              "QUESTION_LOCK_EVENT",
              "QUESTION_TITLE_EVENT",
              "QUESTION_DESCRIPTION_EVENT",
              "INTERVIEWSETTINGS_EVENT"
            ]
          },
          "currentActivityType": {
            "type": "string",
            "enum": [
              "PARTICIPANT_INVITED",
              "PARTICIPANT_WAITING",
              "PARTICIPANT_ALLOWED",
              "PARTICIPANT_DENIED",
              "PARTICIPANT_JOINED",
              "PARTICIPANT_LEFT",
              "PARTICIPANT_DISCONNECTED",
              "PARTICIPANT_LEFT_LOBBY",
              "PARTICIPANT_AWAITING_FEEDBACK",
              "PARTICIPANT_FEEDBACK_SUBMITTED",
              "PARTICIPANT_RECEIVED_FEEDBACK",
              "PARTICIPANT_REMINDER_SENT",
              "QUESTION_ADDED",
              "QUESTION_DELETED",
              "QUESTION_TEXT_UPDATED",
              "QUESTION_LOCKED",
              "QUESTION_UNLOCKED",
              "QUESTION_RENAMED",
              "INTERVIEW_LOCKED",
              "INTERVIEW_UNLOCKED",
              "INTERVIEW_UNSCHEDULED",
              "INTERVIEW_SCHEDULED",
              "INTERVIEW_RESCHEDULED",
              "INTERVIEW_IN_PROGRESS",
              "INTERVIEW_AWAITING_FEEDBACK",
              "INTERVIEW_EVALUATED",
              "INTERVIEW_DEACTIVATED",
              "INTERVIEW_INTERRUPTED",
              "INTERVIEW_CANCELLED",
              "BROWSER_TAB_SWITCH_BLUR",
              "BROWSER_TAB_SWITCH_FOCUS",
              "INTERVIEW_SETTING_COPY_PASTE_TOGGLE"
            ]
          },
          "firstParticipantName": {
            "type": "string"
          },
          "secondParticipantName": {
            "type": "string"
          },
          "duration": {
            "type": "integer",
            "format": "int64"
          },
          "newQuestionLabel": {
            "type": "string"
          },
          "oldQuestionLabel": {
            "type": "string"
          },
          "questionType": {
            "type": "string",
            "enum": [
              "CODING",
              "WHITEBOARD",
              "DIAGRAM",
              "TEXT_EDITOR",
              "METTL_QUESTION"
            ]
          },
          "settingChangedValue": {
            "type": "string"
          },
          "eventData": {
            "type": "object",
            "additionalProperties": {
              "type": "object"
            }
          },
          "InterviewActivityLogSummaryResponse": {
            "type": "object",
            "properties": {
              "createdOn": {
                "type": "integer",
                "description": "Time when this log was created",
                "format": "int64",
                "example": 12312378909
              },
              "eventType": {
                "type": "string",
                "description": "Event type of log",
                "example": "INTERVIEW_STATUS_EVENT",
                "enum": [
                  "PARTICIPANT_STATUS_EVENT",
                  "QUESTION_STATUS_EVENT",
                  "INTERVIEW_LOCK_EVENT",
                  "INTERVIEW_STATUS_EVENT",
                  "PARTICIPANT_DEVIATION_EVENT",
                  "PARTICIPANT_REMINDER_EVENT",
                  "QUESTION_LOCK_EVENT",
                  "QUESTION_TITLE_EVENT",
                  "QUESTION_DESCRIPTION_EVENT",
                  "INTERVIEWSETTINGS_EVENT"
                ]
              },
              "currentActivityType": {
                "type": "string",
                "description": "Activity performed for the event type",
                "example": "INTERVIEW_STATUS_EVENT",
                "enum": [
                  "PARTICIPANT_INVITED",
                  "PARTICIPANT_WAITING",
                  "PARTICIPANT_ALLOWED",
                  "PARTICIPANT_DENIED",
                  "PARTICIPANT_JOINED",
                  "PARTICIPANT_LEFT",
                  "PARTICIPANT_DISCONNECTED",
                  "PARTICIPANT_LEFT_LOBBY",
                  "PARTICIPANT_AWAITING_FEEDBACK",
                  "PARTICIPANT_FEEDBACK_SUBMITTED",
                  "PARTICIPANT_RECEIVED_FEEDBACK",
                  "PARTICIPANT_REMINDER_SENT",
                  "QUESTION_ADDED",
                  "QUESTION_DELETED",
                  "QUESTION_TEXT_UPDATED",
                  "QUESTION_LOCKED",
                  "QUESTION_UNLOCKED",
                  "QUESTION_RENAMED",
                  "INTERVIEW_LOCKED",
                  "INTERVIEW_UNLOCKED",
                  "INTERVIEW_UNSCHEDULED",
                  "INTERVIEW_SCHEDULED",
                  "INTERVIEW_RESCHEDULED",
                  "INTERVIEW_IN_PROGRESS",
                  "INTERVIEW_AWAITING_FEEDBACK",
                  "INTERVIEW_EVALUATED",
                  "INTERVIEW_DEACTIVATED",
                  "INTERVIEW_INTERRUPTED",
                  "INTERVIEW_CANCELLED",
                  "BROWSER_TAB_SWITCH_BLUR",
                  "BROWSER_TAB_SWITCH_FOCUS",
                  "INTERVIEW_SETTING_COPY_PASTE_TOGGLE"
                ]
              },
              "eventData": {
                "type": "object",
                "additionalProperties": true,
                "description": "EventData will provide details about the events of occurred during this activity log."
              }
            }
          },
          "InterviewBasicDetailsDto": {
            "type": "object",
            "properties": {
              "uuid": {
                "type": "string"
              },
              "candidateLink": {
                "type": "string"
              },
              "interviewerLink": {
                "type": "string"
              },
              "pin": {
                "type": "string"
              },
              "createdOn": {
                "type": "integer",
                "format": "int64"
              },
              "linkExpiryDate": {
                "type": "integer",
                "format": "int64"
              },
              "settings": {
                "$ref": "#/components/schemas/InterviewSettingsDto",
                "name": {
                  "type": "string"
                },
                "mode": {
                  "type": "string",
                  "enum": [
                    "ONLINE",
                    "OFFLINE"
                  ]
                },
                "type": {
                  "type": "string",
                  "enum": [
                    "INTERVIEW",
                    "GROUP_DISCUSSION",
                    "ROLE_PLAY",
                    "ORIENTATION"
                  ]
                },
                "category": {
                  "type": "string",
                  "enum": [
                    "INTERVIEW",
                    "GROUP_DISCUSSION"
                  ]
                }
              },
              "InterviewDto": {
                "type": "object",
                "properties": {
                  "uuid": {
                    "type": "string"
                  },
                  "candidateLink": {
                    "type": "string"
                  },
                  "interviewerLink": {
                    "type": "string"
                  },
                  "pin": {
                    "type": "string"
                  },
                  "createdOn": {
                    "type": "integer",
                    "format": "int64"
                  },
                  "linkExpiryDate": {
                    "type": "integer",
                    "format": "int64"
                  },
                  "settings": {
                    "$ref": "#/components/schemas/InterviewSettingsDto",
                    "name": {
                      "type": "string"
                    },
                    "mode": {
                      "type": "string",
                      "enum": [
                        "ONLINE",
                        "OFFLINE"
                      ]
                    },
                    "type": {
                      "type": "string",
                      "enum": [
                        "INTERVIEW",
                        "GROUP_DISCUSSION",
                        "ROLE_PLAY",
                        "ORIENTATION"
                      ]
                    },
                    "category": {
                      "type": "string",
                      "enum": [
                        "INTERVIEW",
                        "GROUP_DISCUSSION"
                      ]
                    },
                    "startDate": {
                      "type": "integer",
                      "format": "int64"
                    },
                    "duration": {
                      "type": "integer",
                      "format": "int64"
                    },
                    "timezone": {
                      "type": "string"
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```

## Pretty print

```
[{"type": "UNSCHEDULED", "SCHEDULED", "IN_PROGRESS", "AWAITING_FEEDBACK", "EVALUATED", "DEACTIVATED", "INTERRUPTED", "CANCELLED"}], "updatedOn": {"type": "integer", "format": "int64"}, "InProgressDuration": {"type": "integer", "format": "int64"}, "webhookUrl": {"type": "string"}, "linkExpiryDurationInHr": {"type": "integer", "format": "int32"}, "location": {"type": "string"}, "locked": {"type": "boolean"}, "interviewers": {"type": "array", "items": {"$ref": "#/components/schemas/InterviewParticipantDto"}}, "candidates": {"type": "array", "items": {"$ref": "#/components/schemas/CandidateDto"}}, "starred": {"type": "boolean"}}, "InterviewerCompetencyFeedbackDto": {"type": "object", "properties": {"interviewerName": {"type": "string"}, "interviewerStatus": {"type": "string", "enum": ["INVITED", "WAITING", "JOINED_INTERVIEW", "LEFT_INTERVIEW", "FEEDBACK_SUBMITTED", "AWAITING_FEEDBACK", "NOT_ACTIVE", "DENIED", "ALLOWED", "DISCONNECTED", "LEFT_LOBBY"]}, "score": {"type": "number", "format": "float"}, "comment": {"type": "string"}, "updatedOn": {"type": "integer", "format": "int64"}, "userStatus": {"type": "string", "enum": ["ENABLED", "DISABLED"]}, "userStatusAtInterviewEnd": {"type": "string", "enum": ["ENABLED", "DISABLED"]}, "skippedEvaluation": {"type": "boolean"}, "skippedEvaluationReason": {"type": "string"}, "notAssessed": {"type": "boolean"}, "invited": {"type": "boolean"}, "joinedOnce": {"type": "boolean"}}, "InterviewerCompetencyFeedbackResponse": {"type": "object", "properties": {"interviewerName": {"type": "string", "description": "Interviewer name", "example": "Interviewer 1"}, "interviewerStatus": {"type": "string", "description": "Current interview status of the interviewer", "example": "FEEDBACK_SUBMITTED", "enum": ["INVITED", "WAITING", "JOINED_INTERVIEW", "LEFT_INTERVIEW", "FEEDBACK_SUBMITTED", "AWAITING_FEEDBACK", "NOT_ACTIVE", "DENIED", "ALLOWED", "DISCONNECTED", "LEFT_LOBBY"]}, "score": {"type": "number", "description": "Score provided by interviewer for competency", "format": "float", "example": 14.4}, "comment": {"type": "string", "description": "Comment provided by interviewer for competency", "example": "A comment for candidate for competency"}, "skippedEvaluation": {"type": "boolean", "description": "If the evaluation is skipped by interviewer", "example": false}, "skippedEvaluationReason": {"type": "string", "description": "Reason for skipping evaluation", "example": "Reason for skipping evaluation"}, "notAssessed": {"type": "boolean", "description": "If the interviewer not assessed the competency", "example": false}}}, "InterviewerData": {"type": "object", "properties": {"uuid": {"type": "string"}, "firstName": {"type": "string"}, "lastName": {"type": "string"}, "email": {"type": "string"}, "interviewCount": {"type": "integer", "format": "int64"}, "typeVsCountMap": {"type": "object", "additionalProperties": {"type": "integer", "format": "int64"}}, "status": {"type": "string", "enum": ["ENABLED", "DISABLED"]}, "createdOn": {"type": "integer", "format": "int64"}, "updatedOn": {"type": "integer", "format": "int64"}}, "InterviewerInfo": {"required": ["email", "firstName"], "type": "object", "properties": {"firstName": {"type": "string", "description": "First name of the user.\nMax length: 100 characters", "example": "John"}, "lastName": {"type": "string", "description": "Last name of the user.\nMax length: 100 characters", "example": "Doe"}, "email": {"type": "string", "description": "Email of the user.\nMax length: 200 characters", "example": "John.doe@example.com"}}, "description": "List of interviewers for the interview"}, "InterviewerPage": {"type": "object", "properties": {"number": {"minimum": 0, "type": "integer", "format": "int32"}, "size": {"maximum": 100, "minimum": 1, "type": "integer", "format": "int32"}, "hasMorePage": {"type": "boolean"}, "totalPages": {"type": "integer", "format": "int32"}, "totalRecords": {"type": "integer", "format": "int64"}, "interviewers": {"type": "array", "items": {"$ref": "#/components/schemas/InterviewerData"}}}, "InterviewError": {"type": "object", "properties": {"status": {"type": "integer", "format": "int32"}, "message": {"type": "string"}, "i18nKey": {"type": "string"}}, "InterviewFeedback": {"type": "object", "properties": {"interviewSummary": {"$ref": "#/components/schemas/InterviewSummary"}, "candidateFeedbackAndCompetencyDetailResponseList": {"type": "array", "items": {"$ref": "#/components/schemas/CandidateFeedbackAndResultSummary"}}}, "InterviewPage": {"type": "object", "properties": {"number": {"type": "integer", "description": "Data of this page number will be provided in case of pagination.", "format": "int32", "example": 1}, "size": {"type": "integer", "description": "Number of records available in request", "format": "int32", "example": 20}, "hasMorePage": {"type": "boolean", "description": "If there is more data available in next pages", "example": true, "enum": [true, false]}, "totalPages": {"type": "integer", "description": "Number of pages that can be fetched for complete data", "format": "int32", "example": 50}, "totalRecords": {"type": "integer", "description": "Number of total records available that can be fetched in paginated way", "format": "int64", "example": 300}, "interviewList": {"type": "array", "description": "List of interviews", "items":
```

## Pretty print

```
{
  "type": "object",
  "properties": {
    "uuid": {
      "type": "string"
    },
    "status": {
      "type": "string",
      "enum": [
        "INVITED",
        "WAITING",
        "JOINED_INTERVIEW",
        "LEFT_INTERVIEW",
        "FEEDBACK_SUBMITTED",
        "AWAITING_FEEBACK",
        "NOT_ACTIVE",
        "DENIED",
        "ALLOWED",
        "DISCONNECTED",
        "LEFT_LOBBY"
      ]
    },
    "user": {
      "$ref": "#/components/schemas/UserDto",
      "participantType": {
        "type": "string",
        "enum": [
          "INTERVIEWER",
          "CANDIDATE"
        ]
      },
      "invited": {
        "type": "boolean"
      },
      "platformFeedbackSubmitted": {
        "type": "boolean"
      },
      "lastActiveTime": {
        "type": "integer",
        "format": "int64"
      },
      "networkInfo": {
        "type": "integer",
        "format": "int64"
      },
      "joinedOnce": {
        "type": "boolean"
      },
      "joinedDuration": {
        "type": "integer",
        "format": "int64"
      },
      "joinedLobby": {
        "type": "boolean"
      },
      "inviteReminderCount": {
        "type": "integer",
        "format": "int32"
      },
      "evaluationReminderCount": {
        "type": "integer",
        "format": "int32"
      },
      "userStatusAtInterviewEnd": {
        "type": "string",
        "enum": [
          "ENABLED",
          "DISABLED"
        ]
      },
      "fullName": {
        "type": "string"
      },
      "invitedOrJoined": {
        "type": "boolean"
      }
    },
    "InterviewParticipantResponse": {
      "type": "object",
      "properties": {
        "uuid": {
          "type": "string",
          "description": "Unique id of the participant",
          "example": "d8ccc0a"
        },
        "status": {
          "type": "string",
          "description": "Current status of the participant",
          "example": "INVITED",
          "enum": [
            "INVITED",
            "WAITING",
            "JOINED_INTERVIEW",
            "LEFT_INTERVIEW",
            "FEEDBACK_SUBMITTED",
            "AWAITING_FEEBACK",
            "NOT_ACTIVE",
            "DENIED",
            "ALLOWED",
            "DISCONNECTED",
            "LEFT_LOBBY"
          ]
        },
        "user": {
          "$ref": "#/components/schemas/UserResponse",
          "invited": {
            "type": "boolean",
            "description": "If the participant is invited for the interview",
            "example": true
          },
          "joinedOnce": {
            "type": "boolean",
            "description": "If the participant has joined once in the interview",
            "example": true
          },
          "joinedDuration": {
            "type": "integer",
            "description": "Duration of participant in the interview",
            "format": "int64",
            "example": 17465881
          }
        },
        "InterviewResponse": {
          "required": [
            "duration",
            "timezone"
          ],
          "type": "object",
          "properties": {
            "uuid": {
              "type": "string",
              "description": "A unique identifier of the interview",
              "example": "4de94450"
            },
            "candidateLink": {
              "type": "string",
              "description": "Interview link for the candidate.",
              "example": "https://interviews.mettl.com/candidate/d8ccc0a"
            },
            "interviewerLink": {
              "type": "string",
              "description": "Interview link for the interviewer.",
              "example": "https://interviews.mettl.com/interviewer/d8ccc0a"
            },
            "pin": {
              "type": "string",
              "description": "PIN to access interview for interviewers",
              "example": "HTG7SF"
            },
            "createdOn": {
              "type": "integer",
              "description": "Timestamp when interview was created",
              "format": "int64",
              "example": 17465881
            },
            "linkExpiryDate": {
              "type": "integer",
              "description": "Timestamp when the interview link will expire",
              "format": "int64",
              "example": 17465881
            },
            "settings": {
              "$ref": "#/components/schemas/InterviewSettingsResponse",
              "name": {
                "type": "string",
                "description": "Name of the interview.",
                "example": "Interview with John"
              },
              "mode": {
                "type": "string",
                "description": "Mode of interview",
                "example": "ONLINE",
                "enum": [
                  "ONLINE",
                  "OFFLINE"
                ]
              },
              "type": {
                "type": "string",
                "description": "Type of interview",
                "example": "INTERVIEW",
                "enum": [
                  "INTERVIEW",
                  "GROUP_DISCUSSION",
                  "ROLE_PLAY"
                ]
              },
              "startDate": {
                "type": "integer",
                "description": "Start date(epoch time) of interview",
                "format": "int64",
                "example": 1726403385000
              },
              "duration": {
                "type": "integer",
                "description": "Duration for interview in minutes.",
                "format": "int64",
                "example": 120
              },
              "timezone": {
                "type": "string",
                "description": "Timezone for interview.",
                "example": "Asia/Kolkata"
              },
              "interviewStatus": {
                "type": "string",
                "description": "Current status of the interview",
                "example": "SCHEDULED",
                "enum": [
                  "UNSCHEDULED",
                  "SCHEDULED",
                  "IN_PROGRESS",
                  "AWAITING_FEEDBACK",
                  "EVALUATED",
                  "DEACTIVATED",
                  "INTERRUPTED",
                  "CANCELLED"
                ]
              },
              "webhookUrl": {
                "type": "string",
                "description": "A webhook url where interview events are published",
                "example": "https://any-web-hook-url.com"
              },
              "linkExpiryDurationInHr": {
                "type": "integer",
                "description": "Duration for which interview link will be active",
                "format": "int32",
                "example": 12
              },
              "location": {
                "type": "string",
                "description": "Location of interview. Only required when mode is 'OFFLINE'",
                "example": "221B Baker Street"
              },
              "jobProfileUuid": {
                "type": "string",
                "description": "Unique id of job profile",
                "example": "7ce6643b-e287-44c3-93ca-5d31ce8dc56d"
              },
              "interviewers": {
                "type": "array",
                "items": {
                  "$ref": "#/components/schemas/InterviewParticipantResponse"
                }
              },
              "candidates": {
                "type": "array",
                "items": {
                  "$ref": "#/components/schemas/CandidateResponse"
                }
              },
              "description": "List of interviews"
            },
            "InterviewSettingRequest": {
              "type": "object",
              "properties": {
                "allowedQuestionTypes": {
                  "uniqueItems": true,
                  "type": "array",
                  "description": "Allowed question types for interview",
                  "example": "CODING, WHITEBOARD, DIAGRAM, METTL_QUESTION",
                  "items": {
                    "type": "string",
                    "description": "Allowed question types for interview",
                    "example": "CODING, WHITEBOARD, DIAGRAM, METTL_QUESTION",
                    "enum": [
                      "CODING",
                      "WHITEBOARD",
                      "DIAGRAM",
                      "METTL_QUESTION"
                    ]
                  }
                },
                "enableCopyPaste": {
                  "type": "boolean",
                  "description": "Enable copy/paste option during the"
                }
              }
            }
          }
        }
      }
    }
  }
}
```



## Pretty print

```
{
  "type": "boolean",
  "description": "Enable/disable recording of the interview",
  "example": true,
  "openInVideoMode": {
    "type": "boolean",
    "description": "Start the interview in video mode",
    "example": true,
    "showOverallRecommendation": {
      "type": "boolean",
      "description": "Show the overall recommendations",
      "example": true,
      "sendEmails": {
        "type": "boolean",
        "description": "Send emails of interview activity",
        "example": true,
        "enableInvite": {
          "type": "boolean",
          "description": "Enable inviting other participants in a interview",
          "example": true,
          "language": {
            "type": "string",
            "description": "Language of the interview",
            "example": "English",
            "enum": ["English", "German", "Italian", "French", "European Portuguese", "European Spanish", "Polish", "Greek"]
          },
          "enablePreviousSessionReports": {
            "type": "boolean",
            "description": "Show all previous session reports in a session",
            "example": true,
            "enableEvaluationPanel": {
              "type": "boolean",
              "description": "Show evaluation panel in the session",
              "example": true,
              "enableCandidateProfile": {
                "type": "boolean",
                "description": "Show candidate profile in the session",
                "example": true,
                "description": "Interview settings you want to enable for this interview",
                "InterviewSettingsDto": {
                  "type": "object",
                  "properties": {
                    "allowedQuestionTypes": {
                      "uniqueItems": true,
                      "type": "array",
                      "items": {
                        "type": "string",
                        "enum": ["CODING", "WHITEBOARD", "DIAGRAM", "TEXT_EDITOR", "METTL_QUESTION"]
                      }
                    },
                    "keywordMappings": {
                      "type": "object",
                      "additionalProperties": {
                        "type": "string"
                      }
                    },
                    "enableCopyPaste": {
                      "type": "boolean"
                    },
                    "enableRecording": {
                      "type": "boolean"
                    },
                    "openInVideoMode": {
                      "type": "boolean"
                    },
                    "showOverallRecommendation": {
                      "type": "boolean"
                    },
                    "recordingDisabledFromClient": {
                      "type": "boolean"
                    },
                    "enableCopyPasteSettingModified": {
                      "type": "boolean"
                    },
                    "sendEmails": {
                      "type": "boolean"
                    },
                    "enableCandidateProfile": {
                      "type": "boolean"
                    },
                    "enableInvite": {
                      "type": "boolean"
                    },
                    "enableReports": {
                      "type": "boolean"
                    },
                    "maxInterviewerCount": {
                      "type": "integer",
                      "format": "int32"
                    },
                    "maxCandidateCount": {
                      "type": "integer",
                      "format": "int32"
                    },
                    "language": {
                      "type": "string"
                    },
                    "enablePreviousSessionReports": {
                      "type": "boolean"
                    },
                    "enableEvaluationPanel": {
                      "type": "boolean"
                    },
                    "maxInterviewersAllowedWaitingCount": {
                      "type": "integer",
                      "format": "int32"
                    },
                    "maxCandidateAllowedWaitingCount": {
                      "type": "integer",
                      "format": "int32"
                    }
                  }
                },
                "InterviewSettingsResponse": {
                  "type": "object",
                  "properties": {
                    "allowedQuestionTypes": {
                      "uniqueItems": true,
                      "type": "array",
                      "description": "Allowed question types for interview",
                      "example": ["CODING", "WHITEBOARD", "DIAGRAM", "METTL_QUESTION"],
                      "items": {
                        "type": "string",
                        "description": "Allowed question types for interview",
                        "example": ["CODING", "WHITEBOARD", "DIAGRAM", "METTL_QUESTION"],
                        "enum": ["CODING", "WHITEBOARD", "DIAGRAM", "METTL_QUESTION"]
                      }
                    },
                    "enableCopyPaste": {
                      "type": "boolean",
                      "description": "Enable copy/paste option during the interview",
                      "example": true,
                      "enableRecording": {
                        "type": "boolean",
                        "description": "Enable/disable recording of the interview",
                        "example": true,
                        "openInVideoMode": {
                          "type": "boolean",
                          "description": "Start the interview in video mode",
                          "example": true,
                          "showOverallRecommendation": {
                            "type": "boolean",
                            "description": "Show the overall recommendations",
                            "example": true,
                            "recordingDisabledFromClient": {
                              "type": "boolean",
                              "description": "This flag shows if recording is disabled from client level",
                              "example": true,
                              "enableCopyPasteSettingModified": {
                                "type": "boolean",
                                "description": "This flag shows if the copy paste setting is modified at least once in the current interview",
                                "example": true,
                                "sendEmails": {
                                  "type": "boolean",
                                  "description": "Send emails of interview activity",
                                  "example": true,
                                  "enableInvite": {
                                    "type": "boolean",
                                    "description": "Enable candidate profile in the interview",
                                    "example": true,
                                    "language": {
                                      "type": "string",
                                      "description": "Language of the interview",
                                      "example": "English",
                                      "enum": ["English", "German", "Italian", "French", "European Portuguese", "European Spanish", "Polish", "Greek"]
                                    },
                                    "description": "Details of applicable interview settings."
                                  },
                                  "InterviewSummary": {
                                    "type": "object",
                                    "properties": {
                                      "duration": {
                                        "type": "integer",
                                        "description": "Duration of interview in millis",
                                        "format": "int64",
                                        "example": 868396
                                      },
                                      "startTime": {
                                        "type": "integer",
                                        "description": "Start time of interview in millis",
                                        "format": "int64",
                                        "example": 1715158648725
                                      },
                                      "endTime": {
                                        "type": "integer",
                                        "description": "End time of interview in millis",
                                        "format": "int64",
                                        "example": 1715158656027
                                      }
                                    },
                                    "description": "This field will provide interview summary",
                                    "InterviewSummaryDto": {
                                      "type": "object",
                                      "properties": {
                                        "duration": {
                                          "type": "integer",
                                          "format": "int64"
                                        },
                                        "startTime": {
                                          "type": "integer",
                                          "format": "int64"
                                        },
                                        "endTime": {
                                          "type": "integer",
                                          "format": "int64"
                                        },
                                        "competencyAvailable": {
                                          "type": "integer",
                                          "format": "int64"
                                        },
                                        "meetingRecordingAvailable": {
                                          "type": "boolean"
                                        }
                                      }
                                    },
                                    "interviewSettings": {

```

```
{
  "type": "string", "enum": ["ONLINE", "OFFLINE"]}, "candidates": {"type": "array", "items": {
    "$ref": "#/components/schemas/CandidateBasicDetails"}}, "jobProfile": {
    $ref": "#/components/schemas/JobProfileBasicDetails"}, "location": {
      "type": "string"}, "clientReportSetting": {
        $ref": "#/components/schemas/ClientReportSettingDto"}, "category": {"type": "string", "enum": [
          "INTERVIEW", "GROUP_DISCUSSION"]} }, "InterviewSummaryResponse": {
      "type": "object", "properties": {
        "duration": {"type": "integer", "description": "Duration of interview in millis", "format": "int64", "example": 868396}, "startTime": {
          "type": "integer", "description": "Start time of interview in millis", "format": "int64", "example": 1715158648725}, "endTime": {
            "type": "integer", "description": "End time of interview in millis", "format": "int64", "example": 1715158656027}, "interviewSettings": {
              $ref": "#/components/schemas/InterviewSettingsResponse"}, "name": {
                "type": "string", "description": "Name of interview", "example": "Interview with john"}, "type": {
                  "type": "string", "description": "Type of interview", "example": "INTERVIEW", "enum": [
                    "INTERVIEW", "GROUP_DISCUSSION", "ROLE_PLAY"]}, "mode": {"type": "string", "description": "Mode of interview", "example": "ONLINE", "enum": ["ONLINE", "OFFLINE"]}, "candidates": {
                    "type": "array", "description": "List of candidates in the interview", "items": {
                      $ref": "#/components/schemas/CandidateBasicDetails"}}, "jobProfile": {
                      $ref": "#/components/schemas/JobProfileBasicDetailsResponse"}, "location": {
                        "type": "string", "description": "Location of interview. Only required when mode is 'OFFLINE'", "example": "221B Baker Street"}, "meetingRecordingAvailable": {
                          "type": "boolean", "description": "Is meeting recording available", "example": true, "enum": [true, false]}}, "JobProfileBasicDetails": {"type": "object", "properties": {
                            "name": {"type": "string"}, "jobDescription": {
                              $ref": "#/components/schemas/FileDetailsDto"}, "displayName": {
                                "type": "string"}}}, "JobProfileBasicDetailsResponse": {"type": "object", "properties": {
                                  "name": {"type": "string", "description": "Name of job profile", "example": "Java Developer"}, "jobDescription": {
                                    $ref": "#/components/schemas/FileDetailsResponse"}}, "description": "Details of job profile associated with this interview"}, "JobProfileData": {"type": "object", "properties": {
                                      "name": {"type": "string", "description": "Name of job profile", "example": "Software Test Engineer"}, "uuid": {"type": "string", "description": "Unique id of a job profile", "example": "7ce6643b-e287-44c3-93ca-5d31ce8dc56d"}, "type": {
                                        "type": "string", "description": "Type of job profile", "example": "ADMIN", "enum": [
                                          "ADMIN", "SELF", "ADMIN", "SELF"]}, "createdOn": {"type": "integer", "description": "Timestamp (in millis) when this job profile was created", "format": "int64", "example": 12312378909}, "modifiedOn": {
                                          "type": "integer", "description": "Timestamp (in millis) when this job profile was last modified", "format": "int64", "example": 12312378909}, "isEditable": {
                                            "type": "boolean", "description": "Is this job profile editable", "example": true, "enum": [true, false]}, "isFeedbackSubmittedForCandidate": {"type": "boolean", "description": "Is any feedback is submitted for the candidates which are present in the session linked to this job profile", "example": true, "enum": [true, false]}, "typeVsCountMap": {
                                              "type": "object", "additionalProperties": {
                                                "type": "integer", "format": "int64"}}}, "description": "List of job profiles"}, "JobProfileDto": {"required": ["name"], "type": "object", "properties": {
                                                  "uuid": {"type": "string", "name": {"maxLength": 100, "minLength": 0, "type": "string"}, "displayName": {
                                                    "maxLength": 100, "minLength": 0, "type": "string"}, "descriptionFileUuid": {
                                                      "type": "string", "description": "descriptionFileName": {
                                                        "type": "string", "description": "descriptionFileUrl": {"type": "string"}, "format": "int64"}, "modifiedOn": {
                                                          "type": "integer", "format": "int64"}, "competencies": {"type": "array", "items": {
                                                            $ref": "#/components/schemas/CompetencyDto"}}, "type": {"type": "string", "enum": [
                                                              "ADMIN", "SELF"]}, "mettlAssessments": {"pattern": "\\{\\s*(\\(\\d+\\))\\s*:\\s*(\\[<|? \\\"\\.+,\\\\\\\\\\/]*\\\\\\\"\\s*(\\(\\d+\\))\\s*:\\s*(\\[<|? \\\"\\.+,\\\\\\\\\\/]*\\\\\\\"\\s*)*\\\\\"", "type": "string", "maxMarks": {
                                                                "type": "number", "format": "float", "guide": {
                                                                  "maxLength": 150000, "minLength": 0, "type": "string"}, "allowNotAssessed": {
                                                                    "type": "boolean"}}}}, "JobProfilePage": {"type": "object", "properties": {
                                                                      "number": {"type": "integer", "description": "Data of this page number will be provided in case of pagination.", "format": "int32", "example": 1}, "size": {"type": "integer", "description": "Number of records available in request", "format": "int32", "example": 20}, "hasMorePage": {
                                                                        "type": "boolean", "description": "If there is more data available in next pages", "example": true, "enum": [true, false]}, "totalPages": {
                                                                          "type": "integer", "description": "Number of pages that can be fetched for complete data", "format": "int32", "example": 50}, "totalRecords": {"type": "integer", "description": "Number of total records available that can be fetched in paginated
```

## Pretty print

```
{
  "type": "array",
  "description": "List of job profiles",
  "items": {
    "$ref": "#/components/schemas/JobProfileData"
  }
}, {
  "type": "object",
  "properties": {
    "name": {
      "type": "string"
    },
    "competencyCount": {
      "type": "integer",
      "format": "int32"
    },
    "chartDataList": {
      "type": "array",
      "items": {
        "$ref": "#/components/schemas/ChartData"
      }
    }
  }
}, {
  "type": "object",
  "properties": {
    "playUrl": {
      "type": "string"
    },
    "downloadUrl": {
      "type": "string"
    },
    "startTime": {
      "type": "string"
    },
    "endTime": {
      "type": "string"
    },
    "size": {
      "type": "integer",
      "format": "int64"
    },
    "status": {
      "type": "string",
      "enum": ["STARTED", "STOPPED", "AVAILABLE"]
    }
  }
}, {
  "type": "object",
  "properties": {
    "playUrl": {
      "type": "string",
      "description": "Play url of the recorded meeting",
      "example": "https://video_play_url.com"
    },
    "downloadUrl": {
      "type": "string",
      "description": "Download url of the recorded meeting",
      "example": "https://video_download_url.com"
    },
    "startTime": {
      "type": "string",
      "description": "Start time of the recorded meeting",
      "example": "1715158648725"
    },
    "endTime": {
      "type": "string",
      "description": "End time of the recorded meeting",
      "example": "1715158648725"
    },
    "size": {
      "type": "integer",
      "description": "Size of the recorded meeting",
      "format": "int64",
      "example": 1715158648725
    },
    "status": {
      "type": "string",
      "description": "status of the recorded meeting",
      "example": "AVAILABLE",
      "enum": ["STARTED", "STOPPED", "AVAILABLE"]
    }
  }
}, {
  "type": "object",
  "properties": {
    "assessmentId": {
      "type": "integer",
      "format": "int64"
    },
    "name": {
      "type": "string"
    },
    "assessmentType": {
      "type": "string"
    },
    "createdAt": {
      "type": "integer",
      "format": "int64"
    }
  }
}, {
  "type": "object",
  "properties": {
    "number": {
      "minimum": 0,
      "type": "integer",
      "format": "int32"
    },
    "size": {
      "maximum": 100,
      "minimum": 1,
      "type": "integer",
      "format": "int32"
    },
    "hasMorePage": {
      "type": "boolean"
    },
    "totalPages": {
      "type": "integer",
      "format": "int32"
    },
    "totalRecords": {
      "type": "integer",
      "format": "int64"
    },
    "mettlAssessmentList": {
      "type": "array",
      "items": {
        "$ref": "#/components/schemas/MettlAssessment"
      }
    }
  }
}, {
  "type": "object",
  "properties": {
    "uuid": {
      "type": "string"
    },
    "questionLabel": {
      "type": "string"
    },
    "questionType": {
      "type": "string",
      "enum": ["CODING", "WHITEBOARD", "DIAGRAM", "TEXT_EDITOR", "METTL_QUESTION"]
    },
    "questionText": {
      "type": "string"
    },
    "questionData": {
      "type": "string"
    },
    "questionCreatorUuid": {
      "type": "string"
    },
    "questionStatus": {
      "type": "string",
      "enum": ["LOCKED", "UNLOCKED", "DELETED"]
    }
  }
}, {
  "type": "object",
  "properties": {
    "publicUrl": {
      "type": "string"
    },
    "privateUrl": {
      "type": "string"
    }
  }
}, {
  "required": ["duration", "interviewUuid", "sendEmail", "startDate", "timezone"],
  "type": "object",
  "properties": {
    "interviewUuid": {
      "type": "string",
      "description": "Unique identifier for the interview",
      "example": "7ce6643b-e287-44c3-93ca-5d31ce8dc56d"
    },
    "duration": {
      "type": "integer",
      "description": "Duration for interview in minutes. Max duration: 180",
      "format": "int64",
      "example": 45
    },
    "startDate": {
      "type": "integer",
      "description": "Start date in epoch time",
      "format": "int64",
      "example": 1726403385000
    },
    "timezone": {
      "type": "string",
      "description": "Valid timezone for interview. Use /api/interview/timezones (getAllTimezones) to get a list of valid timezones",
      "example": "Asia/Kolkata"
    },
    "sendEmail": {
      "type": "boolean",
      "description": "Will email be send once rescheduled successfully?",
      "example": true
    },
    "location": {
      "type": "string",
      "description": "Location of interview. Only required when mode is 'OFFLINE'",
      "example": "221B Baker Street"
    }
  }
}, {
  "required": ["duration", "interviewUuid", "startDate", "timezone"],
  "type": "object",
  "properties": {
    "interviewUuid": {
      "minLength": 1,
      "type": "string"
    },
    "duration": {
      "maximum": 180,
      "minimum": 1,
      "type": "integer",
      "format": "int64"
    },
    "startDate": {
      "minimum": 1,
      "type": "integer",
      "format": "int64"
    },
    "timezone": {
      "minLength": 1,
      "type": "string"
    },
    "sendEmail": {
      "type": "boolean"
    },
    "location": {
      "type": "string"
    }
  }
}, {
  "required": ["clientEmail"],
  "type": "object",
  "properties": {
    "clientEmail": {
      "minLength": 1,
      "type": "string"
    },
    "enableAdminPanel": {
      "type": "boolean"
    },
    "enableResubmission": {
      "type": "boolean"
    }
  }
}, {
  "required": ["category", "language"],
  "type": "object",
  "properties": {
    "category": {
      "type": "string",
      "enum": ["ONLINE_INTERVIEW_CANDIDATE_NON_RECORDED", "ONLINE_INTERVIEW_CANDIDATE_RECORDED", "ONLINE_INTEVIEWER_NON_RECORDED", "ONLINE_INTERVIEWER_RECORDED", "ONLINE_GROUP_DISCUSSION_CANDIDATE_NON_RECORDED", "ONLINE_GROUP_DISCUSSION_CANDIDATE_RECORDED", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_NON_RECORDED", "ONLINE_GROUP_DISCUSSION_INTERVIEWER_RECORDED", "OFFLINE_INTERVIEW_INTERVIEWER_NON_RECORDED", "OFFLINE_GROUP_DISCUSSION_INTERVIEWER_NON_RECORDED"]
    },
    "language": {
      "type": "string",
      "enum": ["english", "european_portuguese", "european_spanish", "french", "german", "greek", "italian", "polish", "portuguese_brasil"]
    }
  }
}, {
  "type": "array",
  "items": {
    "$ref": "#/components/schemas/ConsentDetailRequest"
  }
}, {
  "type": "object",
  "properties": {}
}
```

## Pretty print

```

{"minLength":1,"type":"string"},"body":
{"maxLength":150000,"minLength":0,"type":"string"},"calendarInviteTitle":
{"type":"string"},"key":{"type":"string","enum":
["ONLINE_INTERVIEW_CANDIDATE_SCHEDULE","ONLINE_INTERVIEW_CANDIDATE_RESCHEDULE","ONLINE_INTE
RVIEW_CANDIDATE_CANCEL","ONLINE_INTERVIEW_CANDIDATE_UPDATE","ONLINE_INTERVIEW_CANDIDATE_AUT
OMATED_REMINDER","ONLINE_INTERVIEW_CANDIDATE_ON_GOING_REMINDER","ONLINE_INTERVIEW_CANDIDATE
_ON_GOING_INVITE","ONLINE_INTERVIEW_INTERVIEWER_SCHEDULE","ONLINE_INTERVIEW_INTERVIEWER_RES
CHEDULE","ONLINE_INTERVIEW_INTERVIEWER_CANCEL","ONLINE_INTERVIEW_INTERVIEWER_UPDATE","ONLIN
E_INTERVIEW_INTERVIEWER_AUTOMATED_REMINDER","ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER
_WITH_CANDIDATE","ONLINE_INTERVIEW_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_CANDIDATE","ONLINE
_INTERVIEW_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE","ONLINE_INTERVIEW_INTERVIEWER_ON_GOI
NG_INVITE_WITHOUT_CANDIDATE","ONLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER","OFFLINE_INTER
VIEW_INTERVIEWER_SCHEDULE","OFFLINE_INTERVIEW_INTERVIEWER_RESCHEDULE","OFFLINE_INTERVIEW_IN
TERVIEWER_CANCEL","OFFLINE_INTERVIEW_INTERVIEWER_UPDATE","OFFLINE_INTERVIEW_INTERVIEWER_AUT
OMATED_REMINDER","OFFLINE_INTERVIEW_INTERVIEWER_GRADING_REMINDER","OFFLINE_INTERVIEW_CANDI
DATE_SCHEDULE","OFFLINE_INTERVIEW_CANDIDATE_RESCHEDULE","OFFLINE_INTERVIEW_CANDIDATE_CANCEL"
,"OFFLINE_INTERVIEW_CANDIDATE_UPDATE","OFFLINE_INTERVIEW_CANDIDATE_AUTOMATED_REMINDER","ONL
INE_INTERVIEW_CLIENT_AUTOFINISHED_NOTIFICATION","ONLINE_INTERVIEW_CLIENT_EVALUATED_NOTIFICA
TION","ONLINE_INTERVIEW_CLIENT_FINISHED_NOTIFICATION","ONLINE_INTERVIEW_INTERVIEWER_EVALUAT
ION_SUMMARY","ONLINE_INTERVIEW_INTERVIEWER_NOTES_SUMMARY","OFFLINE_INTERVIEW_CLIENT_EVALUAT
ED_NOTIFICATION","OFFLINE_INTERVIEW_INTERVIEWER_EVALUATION_SUMMARY","OFFLINE_INTERVIEW_INTE
RVIEWER_NOTES_SUMMARY","CLIENT_JOB_PROFILE_CANDIDATE_EXCEL_DOWNLOAD","CLIENT_SESSIONS_USAGE
_EXCEL_DOWNLOAD","ONLINE_GROUP_DISCUSSION_CANDIDATE_SCHEDULE","ONLINE_GROUP_DISCUSSION_CAND
IDATE_RESCHEDULE","ONLINE_GROUP_DISCUSSION_CANDIDATE_CANCEL","ONLINE_GROUP_DISCUSSION_CANDI
DATE_UPDATE","ONLINE_GROUP_DISCUSSION_CANDIDATE_AUTOMATED_REMINDER","ONLINE_GROUP_DISCUSSIO
N_CANDIDATE_ON_GOING_REMINDER","ONLINE_GROUP_DISCUSSION_CANDIDATE_ON_GOING_INVITE","ONLINE_
GROUP_DISCUSSION_INTERVIEWER_SCHEDULE","ONLINE_GROUP_DISCUSSION_INTERVIEWER_RESCHEDULE","ON
LINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL","ONLINE_GROUP_DISCUSSION_INTERVIEWER_UPDATE","ONL
INE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMINDER","ONLINE_GROUP_DISCUSSION_INTERVIEWER_O
N_GOING_REMINDER_WITH_CANDIDATE","ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_REMINDER_WIT
HOUT_CANDIDATE","ONLINE_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE","ONLIN
E_GROUP_DISCUSSION_INTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE","ONLINE_GROUP_DISCUSSION_
INTERVIEWER_GRADING_REMINDER","OFFLINE_GROUP_DISCUSSION_INTERVIEWER_SCHEDULE","OFFLINE_GROU
P_DISCUSSION_INTERVIEWER_RESCHEDULE","OFFLINE_GROUP_DISCUSSION_INTERVIEWER_CANCEL","OFFLINE
_GROUP_DISCUSSION_INTERVIEWER_UPDATE","OFFLINE_GROUP_DISCUSSION_INTERVIEWER_AUTOMATED_REMIN
DER","OFFLINE_GROUP_DISCUSSION_INTERVIEWER_GRADING_REMINDER","OFFLINE_GROUP_DISCUSSION_CAND
IDATE_SCHEDULE","OFFLINE_GROUP_DISCUSSION_CANDIDATE_RESCHEDULE","OFFLINE_GROUP_DISCUSSION_C
ANDIDATE_CANCEL","OFFLINE_GROUP_DISCUSSION_CANDIDATE_UPDATE","OFFLINE_GROUP_DISCUSSION_CAND
IDATE_AUTOMATED_REMINDER","ONLINE_GROUP_DISCUSSION_CLIENT_AUTOFINISHED_NOTIFICATION","ONLIN
E_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION","ONLINE_GROUP_DISCUSSION_CLIENT_FINISHED_
NOTIFICATION","ONLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY","ONLINE_GROUP_DISCUS
SION_INTERVIEWER_NOTES_SUMMARY","OFFLINE_GROUP_DISCUSSION_CLIENT_EVALUATED_NOTIFICATION","O
FFLINE_GROUP_DISCUSSION_INTERVIEWER_EVALUATION_SUMMARY","OFFLINE_GROUP_DISCUSSION_INTERVIEW
ER_NOTES_SUMMARY","ONLINE_ROLE_PLAY_CANDIDATE_SCHEDULE","ONLINE_ROLE_PLAY_CANDIDATE_RESCHED
ULE","ONLINE_ROLE_PLAY_CANDIDATE_CANCEL","ONLINE_ROLE_PLAY_CANDIDATE_UPDATE","ONLINE_ROLE_P
LAY_CANDIDATE_AUTOMATED_REMINDER","ONLINE_ROLE_PLAY_CANDIDATE_ON_GOING_REMINDER","ONLINE_RO
LE_PLAY_CANDIDATE_ON_GOING_INVITE","ONLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE","ONLINE_ROLE_PLA
Y_INTERVIEWER_RESCHEDULE","ONLINE_ROLE_PLAY_INTERVIEWER_CANCEL","ONLINE_ROLE_PLAY_INTERVIEW
ER_UPDATE","ONLINE_ROLE_PLAY_INTERVIEWER_AUTOMATED_REMINDER","ONLINE_ROLE_PLAY_INTERVIEWER_
ON_GOING_REMINDER_WITH_CANDIDATE","ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_REMINDER_WITHOUT_C
ANDIDATE","ONLINE_ROLE_PLAY_INTERVIEWER_ON_GOING_INVITE_WITH_CANDIDATE","ONLINE_ROLE_PLAY_I
NTERVIEWER_ON_GOING_INVITE_WITHOUT_CANDIDATE","ONLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDE
R","OFFLINE_ROLE_PLAY_INTERVIEWER_SCHEDULE","OFFLINE_ROLE_PLAY_INTERVIEWER_RESCHEDULE","OFF
LINE_ROLE_PLAY_INTERVIEWER_CANCEL","OFFLINE_ROLE_PLAY_INTERVIEWER_UPDATE","OFFLINE_ROLE_PLA
Y_INTERVIEWER_AUTOMATED_REMINDER","OFFLINE_ROLE_PLAY_INTERVIEWER_GRADING_REMINDER","OFFLINE
_ROLE_PLAY_CANDIDATE_SCHEDULE","OFFLINE_ROLE_PLAY_CANDIDATE_RESCHEDULE","OFFLINE_ROLE_PLAY_
CANDIDATE_CANCEL","OFFLINE_ROLE_PLAY_CANDIDATE_UPDATE","OFFLINE_ROLE_PLAY_CANDIDATE_AUTOMAT
ED_REMINDER","ONLINE_ROLE_PLAY_CLIENT_AUTOFINISHED_NOTIFICATION","ONLINE_ROLE_PLAY_CLIENT_E
VALUATED_NOTIFICATION","ONLINE_ROLE_PLAY_CLIENT_FINISHED_NOTIFICATION","ONLINE_ROLE_PLAY_IN
TERVIEWER_EVALUATION_SUMMARY","ONLINE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY","OFFLINE_ROLE_PL
AY_CLIENT_EVALUATED_NOTIFICATION","OFFLINE_ROLE_PLAY_INTERVIEWER_EVALUATION_SUMMARY","OFFLI
NE_ROLE_PLAY_INTERVIEWER_NOTES_SUMMARY"]},"language":{"type":"string","enum":
["english","european_portuguese","european_spanish","french","german","greek","italian","po
lish","portuguese_brasil"]},"sessionCategory":{"type":"string","enum":
["INTERVIEW","GROUP_DISCUSSION"]}}},"ScheduleInterviewApiRequest":{"required":
["candidates","duration","interviewers","startDate","timezone"],"type":"object","properties
":{"interviewers":{"type":"array","description":"List of interviewers for the
interview"},"items":{"$ref":"#/components/schemas/InterviewerInfo"},"candidates":

```

## Pretty print

```
{
  "$ref": "#/components/schemas/CandidateInfo",
  "startDate": {
    "type": "integer",
    "description": "Start date in epoch time",
    "format": "int64",
    "example": 1726403385000,
    "timezone": {
      "type": "string",
      "description": "Valid timezone for interview. Use /api/interview/timezones (getAllTimezones) to get a list of valid timezones",
      "example": "Asia/Kolkata",
      "duration": {
        "type": "integer",
        "description": "Duration for interview in minutes. Max duration: 180",
        "format": "int64",
        "example": 120,
        "jobProfileUuid": {
          "type": "string",
          "description": "Unique id of the job profile. Use /api/jobProfile/all to get a list of job profiles for client",
          "example": "7ce6643b-e287-44c3-93ca-5d31ce8dc56d",
          "webhookUrl": {
            "type": "string",
            "description": "A webhook url where interview events are published",
            "example": "https://any-web-hook-url.com",
            "settings": {
              "$ref": "#/components/schemas/InterviewSettingRequest",
              "name": {
                "type": "string",
                "description": "Name of the interview. Only applicable when your interview type is 'GROUP_DISCUSSION'",
                "example": "Interview with john",
                "type": {
                  "type": "string",
                  "description": "Type of interview",
                  "example": "INTERVIEW",
                  "enum": ["INTERVIEW", "GROUP_DISCUSSION", "ROLE_PLAY"]},
                "mode": {
                  "type": "string",
                  "description": "Mode of interview",
                  "example": "ONLINE",
                  "enum": ["OFFLINE", "ONLINE"]},
                "location": {
                  "type": "string",
                  "description": "Location of interview. Only required when mode is 'OFFLINE'",
                  "example": "221B Baker Street"},
                "description": "This model is used to schedule interview."},
                "ScheduleInterviewApiResponse": {
                  "type": "object",
                  "properties": {
                    "uuid": {
                      "type": "string",
                      "description": "A unique identifier of the newly created interview",
                      "example": "4de94450"},
                    "pin": {
                      "type": "string",
                      "description": "PIN to access interview for interviewers",
                      "example": "HTG7SF"},
                    "participants": {
                      "type": "array",
                      "items": {
                        "$ref": "#/components/schemas/UserWithLinkResponse"
                      }
                    }
                  }
                },
                "ScheduleInterviewRequest": {
                  "required": ["candidates", "duration", "interviewUuid", "interviewerUuids", "startDate", "timezone"],
                  "type": "object",
                  "properties": {
                    "interviewerUuids": {
                      "maxItems": 5,
                      "minItems": 1,
                      "type": "array",
                      "items": {
                        "type": "string"
                      }
                    },
                    "candidates": {
                      "type": "array",
                      "items": {
                        "$ref": "#/components/schemas/CandidateInfoDto"
                      }
                    },
                    "startDate": {
                      "minimum": 1,
                      "type": "integer",
                      "format": "int64",
                      "name": {
                        "type": "string"
                      },
                      "interviewUuid": {
                        "minLength": 1,
                        "type": "string",
                        "timezone": {
                          "type": "string"
                        },
                        "duration": {
                          "maximum": 180,
                          "minimum": 1,
                          "type": "integer",
                          "format": "int64"
                        },
                        "sendEmail": {
                          "type": "boolean",
                          "deprecated": true
                        },
                        "jobProfileUuid": {
                          "type": "string"
                        },
                        "settings": {
                          "$ref": "#/components/schemas/InterviewSettingsDto"
                        },
                        "webhookUrl": {
                          "type": "string"
                        },
                        "linkExpiry": {
                          "minimum": 1,
                          "type": "integer",
                          "format": "int32"
                        },
                        "assessmentId": {
                          "minimum": 1,
                          "type": "integer",
                          "format": "int64"
                        },
                        "mode": {
                          "type": "string",
                          "enum": ["ONLINE", "OFFLINE"]
                        },
                        "location": {
                          "type": "string"
                        }
                      }
                    },
                    "ScheduleInterviewRequestV2": {
                      "required": ["candidates", "duration", "interviewers", "startDate", "timezone"],
                      "type": "object",
                      "properties": {
                        "candidates": {
                          "type": "array",
                          "items": {
                            "$ref": "#/components/schemas/UserDto"
                          }
                        },
                        "category": {
                          "type": "string",
                          "enum": ["INTERVIEW", "GROUP_DISCUSSION"]
                        },
                        "duration": {
                          "maximum": 180,
                          "minimum": 1,
                          "type": "integer",
                          "format": "int64"
                        },
                        "interviewers": {
                          "maxItems": 5,
                          "minItems": 1,
                          "type": "array",
                          "items": {
                            "$ref": "#/components/schemas/UserDto"
                          }
                        },
                        "jobProfileUuid": {
                          "type": "string"
                        },
                        "linkExpiry": {
                          "minimum": 1,
                          "type": "integer",
                          "format": "int32"
                        },
                        "location": {
                          "type": "string"
                        },
                        "mode": {
                          "type": "string",
                          "enum": ["ONLINE", "OFFLINE"]
                        },
                        "name": {
                          "type": "string"
                        },
                        "settings": {
                          "$ref": "#/components/schemas/InterviewSettingsDto"
                        },
                        "startDate": {
                          "minimum": 1,
                          "type": "integer",
                          "format": "int64"
                        },
                        "timezone": {
                          "type": "string"
                        },
                        "type": {
                          "type": "string",
                          "enum": ["INTERVIEW", "GROUP_DISCUSSION", "ROLE_PLAY", "ORIENTATION"]
                        },
                        "webhookUrl": {
                          "type": "string"
                        }
                      }
                    },
                    "SessionCategoryAndUseCaseAndEmailCategoryData": {
                      "required": ["emailCategories", "sessionCategory", "useCase"],
                      "type": "object",
                      "properties": {
                        "sessionCategory": {
                          "type": "string",
                          "enum": ["INTERVIEW", "GROUP_DISCUSSION"]
                        },
                        "useCase": {
                          "type": "string",
                          "enum": ["INTERVIEW", "GROUP_DISCUSSION", "ROLE_PLAY", "ORIENTATION"]
                        },
                        "emailCategories": {
                          "minItems": 1,
                          "uniqueItems": true,
                          "type": "array",
                          "items": {
                            "type": "string",
                            "enum": ["SCHEDULE", "RESCHEDULE", "CANCEL", "UPDATE", "ON_GOING_REMINDER", "ON_GOING_REMINDER_WITH_CANDIDATE", "ON_GOING_REMINDER_WITHOUT_CANDIDATE", "ON_GOING_INVITE", "ON_GOING_INVITE_WITH_CANDIDATE", "ON_GOING_INVITE_WITHOUT_CANDIDATE", "AUTOMATED_REMINDER", "GRADING_REMINDER", "AUTOFINISHED_NOTIFICATION", "EVALUATED_NOTIFICATION", "FINISHED_NOTIFICATION", "EVALUATION_SUMMARY", "NOTES_SUMMARY", "JOB_PROFILE_CANDIDATE_EXCEL_DOWNLOAD", "SESSIONS_USAGE_EXCEL_DOWNLOAD"]
                          }
                        }
                      }
                    },
                    "SessionDataPoint": {
                      "type": "object",
                      "properties": {
                        "timePeriod": {
                          "type": "string"
                        },
                        "scheduledSessionCount": {
                          "type": "integer",
                          "format": "int64"
                        },
                        "conductedSessionCount": {
                          "type": "integer",
                          "format": "int64"
                        }
                      }
                    },
                    "SignupRequest": {
                      "required": ["email", "name", "phone", "usecase"],
                      "type": "object",
                      "properties": {
                        "email": {

```

## Pretty print

```
{
  "type": "integer", "format": "int64", "organization": {
    "type": "string", "jobrole": {
      "type": "string", "usecase": {
        "minLength": 1, "type": "string"
      }
    }, "TestMailRequestDto": {
      "required": ["body", "subject"], "type": "object", "properties": {
        "subject": {
          "minLength": 1, "type": "string", "body": {
            "maxLength": 150000, "minLength": 0, "type": "string", "testEmailAddress": {
              "type": "string", "previewOnly": {
                "type": "boolean"
              }
            }, "TestMailResponseDto": {
              "type": "object", "properties": {
                "subject": {
                  "type": "string", "body": {
                    "type": "string", "fromEmailAddress": {
                      "type": "string"
                    }
                  }, "UpdateUserRequestDto": {
                    "required": ["firstName", "uuid"], "type": "object", "properties": {
                      "uuid": {
                        "minLength": 1, "type": "string", "firstName": {
                          "maxLength": 100, "minLength": 1, "type": "string", "lastName": {
                            "maxLength": 100, "minLength": 0, "type": "string"
                          }
                        }, "UserDto": {
                          "required": ["email", "firstName"], "type": "object", "properties": {
                            "firstName": {
                              "maxLength": 100, "minLength": 1, "type": "string", "lastName": {
                                "maxLength": 100, "minLength": 0, "type": "string", "type": {
                                  "type": "string", "enum": ["INTERVIEWER", "CANDIDATE"]
                                }, "email": {
                                  "maxLength": 200, "minLength": 1, "type": "string", "createdOn": {
                                    "type": "integer", "format": "int64", "uuid": {
                                      "type": "string", "status": {
                                        "type": "string", "enum": ["ENABLED", "DISABLED"]
                                      }, "updatedOn": {
                                        "type": "integer", "format": "int64", "fullName": {
                                          "type": "string"
                                        }
                                      }, "UserResponse": {
                                        "type": "object", "properties": {
                                          "uuid": {
                                            "type": "string", "description": "A unique identifier for the newly created user. This uuid must be used to perform an operation for a user.", "example": "cc338777-9e3f-4c98-b8cc-ef8f87de1b0d", "firstName": {
                                              "type": "string", "description": "First name of the user", "example": "John", "lastName": {
                                                "type": "string", "description": "Last name of the user", "example": "Doe", "status": {
                                                  "type": "string", "description": "Status of the user", "example": "ENABLED", "enum": ["ENABLED", "DISABLED"]
                                                }, "email": {
                                                  "type": "string", "description": "Email of the user", "example": "john.doe@example.com", "type": {
                                                    "type": "string", "description": "Type of user. CANDIDATE - If this object is a Candidate. INTERVIEWER - If this object is an Interviewer", "example": "CANDIDATE", "enum": ["CANDIDATE", "INTERVIEWER"]
                                                  }, "description": "Details of the user", "UserWithLinkDto": {
                                                    "required": ["email", "firstName"], "type": "object", "properties": {
                                                      "firstName": {
                                                        "maxLength": 100, "minLength": 1, "type": "string", "lastName": {
                                                          "maxLength": 100, "minLength": 0, "type": "string", "type": {
                                                            "type": "string", "enum": ["INTERVIEWER", "CANDIDATE"]
                                                          }, "email": {
                                                            "maxLength": 200, "minLength": 1, "type": "string", "createdOn": {
                                                              "type": "integer", "format": "int64", "uuid": {
                                                                "type": "string", "status": {
                                                                  "type": "string", "enum": ["ENABLED", "DISABLED"]
                                                                }, "updatedOn": {
                                                                  "type": "integer", "format": "int64", "link": {
                                                                    "type": "string", "fullName": {
                                                                      "type": "string"
                                                                    }, "UserWithLinkResponse": {
                                                                      "type": "object", "properties": {
                                                                        "uuid": {
                                                                          "type": "string", "description": "A unique identifier for the newly created user. This uuid must be used to perform an operation for a user.", "example": "cc338777-9e3f-4c98-b8cc-ef8f87de1b0d", "firstName": {
                                                                            "type": "string", "description": "First name of the user", "example": "John", "lastName": {
                                                                              "type": "string", "description": "Last name of the user", "example": "Doe", "status": {
                                                                                "type": "string", "description": "Status of the user", "example": "ENABLED", "enum": ["ENABLED", "DISABLED"]
                                                                              }, "email": {
                                                                                "type": "string", "description": "Email of the user", "example": "john.doe@example.com", "type": {
                                                                                  "type": "string", "description": "Type of user. CANDIDATE - If this object is a Candidate. INTERVIEWER - If this object is an Interviewer", "example": "CANDIDATE", "enum": ["CANDIDATE", "INTERVIEWER"]
                                                                                }, "link": {
                                                                                  "type": "string", "description": "Link for interview.", "example": "https://interviews.mettl.com/candidate/d8ccc0a"
                                                                                }
                                                                              }
                                                                            }
                                                                          }
                                                                        }
                                                                      }
                                                                    }
                                                                  }
                                                                }
                                                              }
                                                            }
                                                          }
                                                        }
                                                    }
                                                  }
                                                }
                                              }
                                            }
                                          }
                                        }
                                      }
                                    }
                                  }
                                }
                              }
                            }
                          }
                        }
                      }
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```