YAML to DOCX

openapi: 3.0.3

# info

title: job-ms

description: This is the YAML for job ms

# contact

email: nikita.rai@tibilsolutions.com

version: 1.0.11

## tags

name: job

description: job-ms APIs

# paths

# /job/job-count

# get

## tags

- job

summary: Get job files

operationId: getJobCount

## parameters

name: org\_id

in: query

required: True

# schema

type: integer

example: 1

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# Job\_count

type: string

example: 5

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/empty\_error

# /job/applicant-count

# get

## tags

- job

summary: get job files

operationId: getApplicantCount

## parameters

name: org\_id

in: query

required: True

# schema

type: integer

example: 1

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# Job\_count

type: string

example: 5

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/empty\_error

# /job/status

# get

## tags

- Applicants

summary: Applicants Status

description: Retrieve the status of job applicants.

operationId: getApplicantsStatus

## parameters

name: context

in: query

# schema

type: string

## enum

- applicant

- job

description: Context parameter to provide filtering for the status request.

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# data

type: array

# items

type: object

# properties

# pid

type: integer

example: 1

# name

type: string

example: Application received

# code

type: string

example: APPLICATION RECEIVED

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/empty\_error

# /job/job

# post

## tags

- job

summary: Create a new job post

# requestBody

required: True

# content

# application/json

# schema

type: object

# properties

# org\_id

type: integer

example: 1

# provider\_id

type: integer

example: 1

# context

type: string

example: onest:work-opportunities

# ojson

type: object

# properties

# id

type: integer

example: 1

# descriptor

type: object

# properties

# name

type: string

example: Blue collar workers female

# long\_desc

type: string

example: Technician with a creative mindset

# media

# items

type: array

# creator

type: object

# properties

# descriptor

type: object

# properties

# name

type: string

example: Tata Electronics

# quantity

type: object

# properties

# available

type: object

# properties

# count

type: integer

example: 150

# location\_ids

# items

type: array

example: 12 ,14

# fulfillment\_ids

# items

type: array

example: 1,2,3

# tags

type: array

# items

## oneOf

type: object

# properties

# display

type: boolean

example: True

# descriptor

type: object

# properties

# code

type: string

example: academic-eligibility

# name

type: string

example: Academic eligibility

# list

type: array

# items

## oneOf

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: course-name

# name

type: string

example: Name of the course

# value

type: string

example: Class-X

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: mandatory-eligibility

# name

type: string

example: Mandatory Eligibility

# value

type: boolean

example: True

# display

type: boolean

example: True

type: object

# properties

# display

type: boolean

example: True

# descriptor

type: object

# properties

# code

type: string

example: job-requirements

# name

type: string

example: Job requirements

# list

type: array

# items

## oneOf

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: req-experience

# name

type: string

example: Required work experience in years

# value

type: string

example: 2.5

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: req-prof-skill

# name

type: string

example: Skills required for the role

# value

type: string

example: android-development

# display

type: boolean

example: True

type: object

# properties

# display

type: boolean

example: True

# descriptor

type: object

# properties

# code

type: string

example: listing-details

# name

type: string

example: Listing details

# list

type: array

# items

## oneOf

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: industry-type

# name

type: string

example: Industry type

# value

type: string

example: IT Services & Consulting

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: department

# name

type: string

example: Department

# value

type: string

example: Engineering - Software & QA

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: employment-type

# name

type: string

example: Employment-type

# value

type: string

example: full-time

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: job-role

# name

type: string

example: Job-role

# value

type: string

example: DevOps Engineer

# display

type: boolean

example: True

type: object

# properties

# display

type: boolean

example: True

# descriptor

type: object

# properties

# code

type: string

example: salary-info

# name

type: string

example: Salary information

# list

type: array

# items

## oneOf

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: gross-min

# name

type: string

example: Minimum gross pay

# value

type: integer

example: 2000000

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: gross-max

# name

type: string

example: Maximum gross pay

# value

type: integer

example: 2500000

# display

type: boolean

example: True

# Applicant\_form\_details

type: object

# properties

# title

type: string

example: Personal Details

# fields

type: array

# items

## oneOf

type: object

# properties

# name

type: string

example: user-full-name

# type

type: string

example: text

# required

type: boolean

example: True

# label

type: string

example: user-full-name

type: object

# properties

# name

type: string

example: user-phone-number

# type

type: string

example: text

# required

type: boolean

example: True

# label

type: string

example: Phone number

# submitButton

type: string

example: Submit

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# message

type: string

example: Data Inserted successfully

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/create\_error

# /job/job/{pid}

# put

## tags

- job

summary: Update the Job

description: updateJob

operationId: updateJob

## parameters

name: pid

in: path

required: True

# schema

type: integer

example: 1

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# message

type: string

example: job updated successfully

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/empty\_error

# /job/jobs

# get

## tags

- job

summary: Get job files

operationId: getJobByFilter

## parameters

name: org\_id

in: query

required: True

# schema

type: integer

example: 1

name: context

in: query

required: True

# schema

type: string

## enum

- onest:work-opportunities

- onest:financial-support

name: job\_id

in: query

required: True

# schema

type: integer

example: 1

name: jobTitle

in: query

# schema

type: string

example: OLA

name: industry

in: query

# schema

type: string

example: software

name: status

in: query

# schema

type: string

example: active

name: fromDate

in: query

# schema

type: string

example: yyyy-mm-dd

name: toDate

in: query

# schema

type: string

example: yyyy-mm-dd

name: experience

in: query

# schema

type: integer

example: 5

name: location

in: query

# schema

type: integer

example: pid

name: jobProvider

in: query

# schema

type: string

example: ABC Academy

name: Skills

in: query

# schema

type: string

example: Ms-Excel

name: searchTerm

in: query

# schema

type: string

## enum

- designation

- job\_description

- company\_name

- Location

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# job\_id

type: integer

example: 1

# totalnumberofapplicants

type: integer

example: 45

# actuallyFilledVacancies

type: integer

example: 25

# job\_provider

type: integer

example: 1

# postedBy

type: string

example: Priya T

# location

type: array

# items

type: object

# properties

# id

type: integer

example: 1

# name

type: string

example: Delhi

# created\_at

type: string

example: 2024-02-05 14:27:56.687000+00:00

# status

type: string

example: active

# ojson

type: object

# properties

# id

type: integer

example: 1

# descriptor

type: object

# properties

# name

type: string

example: Blue collar workers female

# long\_desc

type: string

example: Technician with a creative mindset

# media

# items

type: array

# creator

type: object

# properties

# descriptor

type: object

# properties

# name

type: string

example: Tata Electronics

# quantity

type: object

# properties

# available

type: object

# properties

# count

type: integer

example: 150

# location\_ids

# items

type: array

example: 12 ,14

# fulfillment\_ids

# items

type: array

example: 1,2,3

# tags

type: array

# items

## oneOf

type: object

# properties

# display

type: boolean

example: True

# descriptor

type: object

# properties

# code

type: string

example: academic-eligibility

# name

type: string

example: Academic eligibility

# list

type: array

# items

## oneOf

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: course-name

# name

type: string

example: Name of the course

# value

type: string

example: Class-X

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: mandatory-eligibility

# name

type: string

example: Mandatory Eligibility

# value

type: boolean

example: True

# display

type: boolean

example: True

type: object

# properties

# display

type: boolean

example: True

# descriptor

type: object

# properties

# code

type: string

example: job-requirements

# name

type: string

example: Job requirements

# list

type: array

# items

## oneOf

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: req-experience

# name

type: string

example: Required work experience in years

# value

type: string

example: 2.5

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: req-prof-skill

# name

type: string

example: Skills required for the role

# value

type: string

example: android-development

# display

type: boolean

example: True

type: object

# properties

# display

type: boolean

example: True

# descriptor

type: object

# properties

# code

type: string

example: listing-details

# name

type: string

example: Listing details

# list

type: array

# items

## oneOf

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: industry-type

# name

type: string

example: Industry type

# value

type: string

example: IT Services & Consulting

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: department

# name

type: string

example: Department

# value

type: string

example: Engineering - Software & QA

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: employment-type

# name

type: string

example: Employment-type

# value

type: string

example: full-time

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: job-role

# name

type: string

example: Job-role

# value

type: string

example: DevOps Engineer

# display

type: boolean

example: True

type: object

# properties

# display

type: boolean

example: True

# descriptor

type: object

# properties

# code

type: string

example: salary-info

# name

type: string

example: Salary information

# list

type: array

# items

## oneOf

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: gross-min

# name

type: string

example: Minimum gross pay

# value

type: integer

example: 2000000

# display

type: boolean

example: True

type: object

# properties

# descriptor

type: object

# properties

# code

type: string

example: gross-max

# name

type: string

example: Maximum gross pay

# value

type: integer

example: 2500000

# display

type: boolean

example: True

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/empty\_error

# /job/applicant/

# get

## tags

- Applicants

summary: get applicant filter

operationId: getApplicantsList

## parameters

name: job\_id

in: query

required: True

# schema

type: integer

name: pid

in: query

# schema

type: integer

example: 1

required: True

name: status

in: query

# schema

type: integer

example: 4

name: fromDate

in: query

# schema

type: string

example: From (yyyy-mm-dd)

name: toDate

in: query

# schema

type: string

example: To (yyyy-mm-dd)

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# data

type: array

# items

type: object

# properties

# pid

type: integer

example: 1

# name

type: string

example: Rahul Singh

# email

type: string

example: rahulsingh@gmail.com

# phone

type: integer

example: 7896342341

# Work Experience

type: integer

example: 2

# resume

type: string

example: pdf

# created\_at

type: string

example: 2024-02-05 14:27:56.687000+00:00

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/empty\_error

# put

## tags

- Status

summary: updateApplicantDetail

description: updateApplicantDetail

operationId: updateApplicantDetail

## parameters

name: pid

in: query

required: True

# schema

type: integer

example: 1

name: status\_pid

in: query

required: True

# schema

type: integer

example: 2

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# message

type: string

example: status updated successfully

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/create\_error

# /job/locations

# get

## tags

- Locations

summary: get Location List

operationId: getLocationDetail

## parameters

name: searchTerm

in: query

# schema

type: string

name: offset

in: query

# schema

type: integer

example: 1

name: limit

in: query

# schema

type: integer

example: 10

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# data

type: array

# items

type: object

# properties

# id

type: integer

example: 1

# name

type: string

example: Delhi

# state\_pid

type: integer

example: 6

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/empty\_error

# /job/{org\_id}/provider

# post

## tags

- Provider

summary: Create a new provider

## parameters

name: org\_id

in: path

required: True

# schema

type: integer

# requestBody

# content

# application/json

# schema

type: object

# properties

# provider\_name

type: string

example: InfoTech

# provider\_website

type: string

example: Infotech@gmail.com

# address

type: string

example: Thane, Mumbai India

# provider\_logo

type: string

example: base64 URL

# context

type: string

example: Finance Support

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# message

type: string

example: Data Inserted successfully

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/create\_error

# /job/{org\_id}/provider/{pid}

# put

## tags

- Provider

summary: Update Providers List

operationId: Provider

## parameters

name: org\_id

in: path

required: True

# schema

type: integer

name: pid

in: path

required: True

# schema

type: integer

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# message

type: string

example: Data Updated Successfully

# 400

description: No Records Found

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 1000

# message

type: string

example: Failed to Updated Data

# /job/{org\_id}/providers/

# get

## tags

- Provider

summary: get Providers List

operationId: getProvidersList

## parameters

name: org\_id

in: path

required: True

# schema

type: integer

name: pid

in: query

required: True

# schema

type: integer

name: offset

in: query

# schema

type: integer

example: 1

name: limit

in: query

# schema

type: integer

example: 10

name: context

in: query

required: True

# schema

type: string

## enum

- onest:work-opportunities

- onest:financial-support

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# data

type: array

# items

type: object

# properties

# provider\_pid

type: integer

example: 1

# media

type: string

example: base64 URL

# provider\_name

type: string

example: InfoTech

# applicationrecevied

type: integer

example: 1

# activejobs

type: integer

example: 2

# totalCount

type: integer

example: 10

# totalPages

type: integer

example: 100

# 400

description: No Records Found

# content

# application/json

# schema

$ref: #/components/schemas/generic\_error

# /job/transactionCount

# get

## tags

- Scholarship

summary: get transactionCount

operationId: getTransactionCount

# responses

# 200

description: Successful operation

# content

# application/json

# schema

type: object

# properties

# code

type: integer

example: 2000

# transactionCount

type: string

example: 120

# data

type: array

# items

type: object

# properties

# name

type: string

example: Sarthak

# job\_id

type: integer

example: 1

# state\_name

type: string

example: Application received

# 400

description: Invalid Input

# content

# application/json

# schema

$ref: #/components/schemas/empty\_error

# components

# schemas

# drive

type: object

# properties

# code

type: integer

example: 2000

# nextPageToken

type: string

example: 15263!@#$%^&JM<

# totalCount

type: integer

example: 1

# files

type: array

# items

type: object

# properties

# mimeType

type: string

example: application/pdf

# parents

type: array

# items

type: string

example: 1YgcgFin7rvksL7fElQgLmw9hkR3GAQOK

# id

type: string

example: 1YgcgFin7rvksLasd7fElQgLmw9hkR3GAQOK

# name

type: string

example: test.pdf

# createdTime

type: string

example: 2023-08-08T01:24:58.301Z

# modifiedTime

type: string

example: 2023-08-08T01:24:58.301Z

# thumbnailLink

type: string

example: aeq67we1n2RFTY78-6789

# generic\_error

type: object

# properties

# code

type: integer

example: 1000

# message

type: string

example: Invalid Input

# empty\_error

type: object

# properties

# code

type: integer

example: 1000

# message

type: string

example: No record found

# create\_error

type: object

# properties

# code

type: integer

example: 1000

# message

type: string

example: Failed to Insert the Data

# securitySchemes

# bearerAuth

type: http

scheme: bearer

bearerFormat: JWT