HRMS

Table of Contents

Chapter 1. Overview

This is an API documentation for HRMS

1.1. Version information

Version : 1.0.0

1.2. URI scheme

Host : localhost:8080
BasePath : /resources
Schemes : HTTP

1.3. Tags

- country : Everything about the country.

- department : It's all about departments.

- designation : Everything about the designation.

- district : Everything about the district.

- employee : Employee related resources.

- jobrole : Everything about the job role.

- meta : Meta/Master Data Access.

- organization : Everything about your organization.

- unit : Everything about the unit.

Chapter 2. Paths

2.1. Returns List of Countries

```
GET /resources/v1/country
```

2.1.1. Description

Returns list of all countries.

2.1.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string

2.1.3. Responses

HTTP Code	Description	Schema
200	The list of countries is retrieved successfully. **Headers** : `Cache-Control` (string) : Caching. **Default** : `"private, no-store, max-age=300"`	CountryModel
403	User does not have permission to execute this command.	No Content
404	No country is found.	ErrorModel
500	Internal Server Error.	ErrorModel

2.1.4. Produces

- `application/xml`
- `application/json`

2.1.5. Tags

- country

2.2. Returns List of Departments in a Given Unit

```
GET /resources/v1/department
```

2.2.1. Description

Returns list of departments within a given unit.

2.2.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Query	**unitId** *required*	The id of the unit whose departments are to be retrieved.	string

2.2.3. Responses

HTTP Code	Description	Schema
200	OK **Headers** : `Cache-Control` (string) : Caching. **Default** : `"private, no-store, max-age=300"`	DepartmentModel
403	User does not have permission to execute this command.	No Content
404	No department is found for the given unit id.	ErrorModel
500	Internal Server Error.	ErrorModel

2.2.4. Produces

- `application/xml`
- `application/json`

2.2.5. Tags

- department

2.3. Returns List of Designations in a Given Job Role.

```
GET /resources/v1/designation
```

2.3.1. Description

Returns list of designations in a given job role.

2.3.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Query	**jobRoleId** *required*	The id of the job role for which the designations are to be returned.	integer

2.3.3. Responses

HTTP Code	Description	Schema
200	The designations are successfully returned. **Headers** : `Cache-Control` (string) : Caching. **Default** : `"private, no-store, max-age=300"`	DesignationModel
403	User does not have permission to execute this command.	No Content
404	Not Found	ErrorModel
500	Internal Server Error.	ErrorModel

2.3.4. Produces

- `application/xml`
- `application/json`

2.3.5. Tags

- designation

2.4. Returns List of Districts

```
GET /resources/v1/district
```

2.4.1. Description

Returns list of districts in a given state.

2.4.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Query	**stateId** *required*	The id of the state whose districts are to be retrieved.	string

2.4.3. Responses

HTTP Code	Description	Schema
200	OK **Headers** : `Cache-Control` (string) : Caching. **Default** : `"private, no-store, max-age=300"`	DistrictModel
403	User does not have permission to execute this command	No Content
404	No district is found for the given state.	ErrorModel
500	Internal Server Error.	ErrorModel

2.4.4. Produces

- `application/xml`
- `application/json`

2.4.5. Tags

- district

2.5. Gets List of Document Types

```
GET /resources/v1/doctype
```

2.5.1. Description

Gets the list of document types. These document types will be mainly used to store various documents

of the employee like passport, birth cirtificate, graduation certificate etc.

2.5.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string

2.5.3. Responses

HTTP Code	Description	Schema
200	The list of document types is successfully retrieved. **Headers** : `Cache-Control` (string) : Caching. **Default** : `"private, no-store, max-age=300"`	DocTypeModel
403	User does not have permission to execute this command.	No Content
404	No document type is found.	ErrorModel
500	Internal Server Error.	ErrorModel

2.5.4. Produces

- `application/xml`
- `application/json`

2.5.5. Tags

- meta

2.6. Gets List of Identity Document Types

```
GET /resources/v1/doctype/identitydoctypes
```

2.6.1. Description

Gets the list of identity document types. Out of all document types, some are that of identity documents like passport for example. Such document type is useful to identify an employee. And the number of such a document will be stored in the employee basic information for quick identification.

2.6.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string

2.6.3. Responses

HTTP Code	Description	Schema
200	The identity document types are successfully retrieved. **Headers** : Cache-Control (string) : Caching. **Default** : "private, no-store, max-age=300"	DocTypeModel
403	User does not have permission to execute this command.	No Content
404	No identy document type is found.	ErrorModel
500	Internal Server Error.	ErrorModel

2.6.4. Produces

- application/xml
- application/json

2.6.5. Tags

- meta

2.7. Creates a new employee.

```
POST /resources/v1/employee
```

2.7.1. Description

Creates a new employee.

2.7.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string

2.7.3. Responses

HTTP Code	Description	Schema
201	The employee is sucessfully created. **Headers** : Cache-Control (string) : Caching. **Default** : "private, no-store, max-age=300"	string
403	User does not have permission to execute this command.	No Content
500	Internal Server Error.	ErrorModel

2.7.4. Consumes

- application/xml
- application/json

2.7.5. Produces

- application/xml
- application/json

2.7.6. Tags

- employee

2.8. Returns Autocomplete List for a Given Employee Attribute

```
GET /resources/v1/employee/
```

2.8.1. Description

This API will be mainly used to provide autocomplete experience to users as they type in text fields for searching employees.

2.8.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Query	**attributeName** *required*	Name of the attribute which could be firstName, middleName, lastName or emailAddress.	string
Query	**attributeValuePrefix** *required*	FirstName, middleName, lastName or emailAddress starts with this value.	string
Query	**numberOfItems** *required*	The number of items matching the value to be returned.	integer
Query	**restricted** *required*	The search will be restricted to the hierarchy of the user only if the flag is true. Otherwise, the search will be done across the organizations.	boolean

2.8.3. Responses

HTTP Code	Description	Schema
200	The matching values are retrieved successfully.	< string > array
403	User does not have permission to execute this command.	No Content
500	Internal Server Error	ErrorModel

2.8.4. Produces

- application/xml
- application/json

2.8.5. Tags

- employee

2.9. Gets the Complete Details of an Employee

```
GET /resources/v1/employee/{id}
```

2.9.1. Description

Returns the complete details of a given employee.

2.9.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Path	**id** *required*	The employee id for which the data will be returned.	integer

2.9.3. Responses

HTTP Code	Description	Schema
200	OK **Headers** : `Cache-Control` (string) : Caching. **Default** : `"private, no-store, max-age=100"`	EmployeeModel
403	User does not have permission to execute this command	No Content
404	No employee with the given id is found.	ErrorModel
500	Internal Server Error.	ErrorModel

2.9.4. Produces

- `application/xml`
- `application/json`

2.9.5. Tags

- employee

2.10. Updates the Additional Details of an Employee

```
PUT /resources/v1/employee/{id}/additionaldetails
```

2.10.1. Description

Updates the additional details of a given employee.

2.10.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Path	**id** *required*	The employee id for which the data will be updated.	integer
Body	**reqBody** *optional*	Employee additional data.	EmployeeAddlDetails Model

2.10.3. Responses

HTTP Code	Description	Schema
200	The data is successfully updated.	No Content
403	User does not have permission to execute this command.	No Content
404	No employee is found with the given employee id.	ErrorModel
500	Internal Server Error.	ErrorModel

2.10.4. Consumes

- `application/xml`
- `application/json`

2.10.5. Tags

- employee

2.11. Updates Employee Address Details

```
PUT /resources/v1/employee/{id}/address
```

2.11.1. Description

Updates the employee address details.

2.11.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Path	**id** *required*	Id of the employee whose address will be updated.	string
Body	**reqBody** *optional*	Employee Address parameters in JSON Body	EmployeeAddressModel

2.11.3. Responses

HTTP Code	Description	Schema
200	The address of the given employee is successfully updated.	No Content
403	User does not have permission to execute this command.	No Content
404	The employee with the given id is not found.	ErrorModel
500	Internal Server Error.	ErrorModel

2.11.4. Consumes

- `application/json`
- `application/xml`

2.11.5. Tags

- employee

2.12. Updates the Basic Details of an Employee

```
PUT /resources/v1/employee/{id}/basic
```

2.12.1. Description

Updates the basic details of a given employee.

2.12.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Path	**id** *required*	The employee id for which the data will be updated.	integer
Body	**reqBody** *optional*	Employee basic data.	EmployeeBasicModel

2.12.3. Responses

HTTP Code	Description	Schema
200	OK	No Content
403	User does not have permission to execute this command.	No Content
404	No employee is found with the given employee id.	ErrorModel
500	Internal Server Error	ErrorModel

2.12.4. Consumes

- `application/xml`
- `application/json`

2.12.5. Tags

- employee

2.13. Updates the Hierarchy Status of an Employee

```
PUT /resources/v1/employee/{id}/hierarchystatus
```

2.13.1. Description

Updates the hierarchy status of a given employee.

2.13.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Path	**id** *required*	The employee id for which the data will be updated.	integer
Body	**reqBody** *optional*	New employee hierarchy status data.	EmployeeHierarchy Model

2.13.3. Responses

HTTP Code	Description	Schema
200	The data is successfully updated.	No Content
403	User does not have permission to execute this command	No Content
404	No employee with the given id is found.	ErrorModel
500	Internal Server Error.	ErrorModel

2.13.4. Consumes

- `application/xml`
- `application/json`

2.13.5. Tags

- employee

2.14. Updates the Optional Benefits of an Employee

```
PUT /resources/v1/employee/{id}/optionalbenefits
```

2.14.1. Description

Updates the optional benefits of a given employee.

2.14.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Path	**id** *required*	The employee id for which the data will be updated.	integer
Body	**reqBody** *optional*	Employee optional benefit data.	EmployeeOptionalBenefitModel

2.14.3. Responses

HTTP Code	Description	Schema
200	The data is successfully updated.	No Content
403	User does not have permission to execute this command.	No Content
404	No employee is found with the given employee id.	ErrorModel
500	Internal Server Error.	ErrorModel

2.14.4. Consumes

- application/xml
- application/json

2.14.5. Tags

- employee

2.15. Updates the Profile of an Employee.

```
PUT /resources/v1/employee/{id}/profile
```

2.15.1. Description

Updates the profile of a given employee.

2.15.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Path	**id** *required*	The employee id for which the data will be updated.	integer
Body	**reqBody** *optional*	Employee profile data.	EmployeeProfileModel

2.15.3. Responses

HTTP Code	Description	Schema
200	The data is successfully updated.	No Content
403	User does not have permission to execute this command.	No Content
404	No employee is found with the given employee id.	ErrorModel
500	Internal Server Error.	ErrorModel

2.15.4. Consumes

- `application/xml`
- `application/json`

2.15.5. Tags

- employee

2.16. Updates the Salary of an Employee

```
PUT /resources/v1/employee/{id}/salary
```

2.16.1. Description

Updates the salary of a given employee.

2.16.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Path	**id** *required*	The employee id for which the data will be updated.	integer
Body	**reqBody** *optional*	Employee salary data.	EmployeeSalaryModel

2.16.3. Responses

HTTP Code	Description	Schema
200	OK	No Content
403	User does not have permission to execute this command.	No Content
404	No employee is found with the given employee id.	ErrorModel
500	Internal Server Error.	ErrorModel

2.16.4. Consumes

- `application/xml`
- `application/json`

2.16.5. Tags

- employee

2.17. Returns List of Job Roles in a Given Organization

```
GET /resources/v1/jobrole
```

2.17.1. Description

Returns list of job roles in a given organization.

2.17.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Query	**orgId** *required*	The id of the organization whose job roles are to be returned.	integer

2.17.3. Responses

HTTP Code	Description	Schema
200	OK **Headers** : Cache-Control (string) : Caching. **Default** : "private, no-store, max-age=300"	JobRoleModel
403	User does not have permission to execute this command.	No Content
404	No job role is found for the given organization.	ErrorModel
500	Internal Server Error.	ErrorModel

2.17.4. Produces

- application/xml
- application/json

2.17.5. Tags

- jobrole

2.18. Gets the Optional Benefits of a Given Job Role

```
GET /resources/v1/jobrole/{id}/optbenefit
```

2.18.1. Description

Gets the optional benefits of a given job role.

2.18.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Path	**id** *required*	The job role id for which the data will be retrieved.	integer

2.18.3. Responses

HTTP Code	Description	Schema
200	OK	OptionalSalaryModel
403	User does not have permission to execute this command.	No Content
404	No optional benefit is found with the given job role id.	ErrorModel
500	Internal Server Error	ErrorModel

2.18.4. Produces

- application/xml
- application/json

2.18.5. Tags

- jobrole

2.19. Gets the Salary of a Given Job Role

```
GET /resources/v1/jobrole/{id}/salary
```

2.19.1. Description

Gets the salary of a given job role.

2.19.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string

Type	Name	Description	Schema
Path	**id** *required*	The job role id for which the data will be retrieved.	integer

2.19.3. Responses

HTTP Code	Description	Schema
200	OK	#definitions/Salary Model
403	User does not have permission to execute this command.	No Content
404	No salary component is found with the given job role id.	ErrorModel
500	Internal Server Error.	ErrorModel

2.19.4. Produces

- `application/xml`
- `application/json`

2.19.5. Tags

- jobrole

2.20. Returns List of Organizations

```
GET /resources/v1/organization
```

2.20.1. Description

Returns a complete list of organizations defined in the system.

2.20.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string

2.20.3. Responses

HTTP Code	Description	Schema
200	The list of organizations are successfully retrieved. **Headers** : `Cache-Control` (string) : Caching. **Default** : `"private, no-store, max-age=300"`	OrganizationModel
403	User does not have permission to execute this command.	No Content
404	No organization is found.	ErrorModel
500	Internal Server Error.	ErrorModel

2.20.4. Produces

- `application/xml`
- `application/json`

2.20.5. Tags

- organization

2.21. Gets List of States for a Given Country

```
GET /resources/v1/state
```

2.21.1. Description

Gets the list of states for a given country. This data will be required while saving the address of the employee.

2.21.2. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Query	**countryId** *required*	Country for which the list of states will be retrieved.	string

2.21.3. Responses

HTTP Code	Description	Schema
200	The list of states are successfully retrieved. **Headers** : Cache-Control (string) : Caching. **Default** : "private, no-store, max-age=300"	StateModel
403	User does not have permission to execute this command.	No Content
404	No state found for the given country.	ErrorModel
500	Internal Server Error.	ErrorModel

2.21.4. Produces

- application/xml
- application/json

2.21.5. Tags

- meta

2.22. Returns List of Units in a Given Organization

```
GET /resources/v1/unit
```

2.22.1. Parameters

Type	Name	Description	Schema
Header	**Authorization : Bearer** *required*	Access token to be sent for authentication and authorization.	string
Query	**organizationId** *required*	The id of the organization whose units will be retrieved.	string

2.22.2. Responses

HTTP Code	Description	Schema
200	The units are successfully retrieved. **Headers** : `Cache-Control` (string) : Caching. **Default** : `"private, no-store, max-age=300"`	UnitModel
403	User does not have permission to execute this command.	No Content
404	No unit is found for the given organization.	ErrorModel
500	Internal Server Error.	ErrorModel

2.22.3. Produces

- `application/xml`
- `application/json`

2.22.4. Tags

- unit

Chapter 3. Definitions

3.1. CountryModel

Name	Description	Schema
countryId *required*	Unique identifier of a country.	integer (int(32))
countryName *required*	Name of the country. **Maximal length** : 32	string (varchar)

3.2. DepartmentModel

Name	Description	Schema
address *required*	Address of the department **Maximal length** : 256	string (varchar)
departmentId *required*	To uniquly identify a department. This field will be ignored during department creation.	integer (int(32))
departmentName *required*	Name of the department **Maximal length** : 100	string (varchar)
parentDepartmentId *optional*	To identify the parent of a particular department	integer (int(32))
unitId *required*	To identify the parent of a particular department	integer (int(32))
unitName *required*	Name of the unit that the department belongs to. The field will be ignored during department creation. **Maximal length** : 256	string (varchar)

3.3. DesignationModel

Name	Description	Schema
designationId *required*	To uniquely identify the designation. This field will be ignored durin the designation creation.	integer (int(32))
designationName *required*	Name of the designation **Maximal length** : 32	string (varchar)

3.4. DistrictModel

Name	Description	Schema
districtId *required*	Unique identifier of a district.	integer (int(32))
districtName *required*	Name of the district. **Maximal length** : 32	string (varchar)
stateId *required*	Unique identifier of a state.	integer (int(32))
stateName *required*	Name of the state. **Maximal length** : 32	string (varchar)

3.5. DocTypeModel

Name	Description	Schema
docTypeId *required*	Unique identifier of a document type viz. passport, birth certificate etc. This field will be ignored during doc type creation.	integer (int(32))
docTypeName *required*	The name of the document type. **Maximal length** : 32	string (varchar)
identityDoc *required*	If the document if a identity document, the value will be true, otherwise false.	boolean

3.6. EmployeeAddlDetailsModel

Name	Description	Schema
dependentNo *optional*	The number of dependents of the employee.	integer (int(32))
emergencyContactName *optional*	The name emergency contact. **Maximal length** : 30	string (varchar)
emergencyContactNo *optional*	The contact number of the emergency contact. **Maximal length** : 15	string (varchar)
medicalReportComment *optional*	The overall medical report comment. **Maximal length** : 32	string (varchar)

Name	Description	Schema
nomineeName 1 *optional*	The name of the first nominee. **Maximal length** : 120	string (varchar)
nomineeName 2 *optional*	The name of the second nominee. **Maximal length** : 120	string (varchar)
nomineeName 3 *optional*	The name of the third nominee. **Maximal length** : 120	string (varchar)
nomineeShare 1 *optional*	The share of the first nominee.	integer (int(32))
nomineeShare 2 *optional*	The share of the second nominee.	integer (int(32))
nomineeShare 3 *optional*	The share of the third nominee.	integer (int(32))
preMedicalCheckUpDate *optional*	The contact number of the emergency contact.	string (date-time)
siblingNo *optional*	The number of siblings of the employee.	integer (int(32))

3.7. EmployeeAddressModel

Name	Description	Schema
area *optional*	The name of the area. **Maximal length** : 32	string (varchar)
countryId *required*	The unique identifier of the country.	integer (int(32))
countryName *required*	The name of the country. This field will be ignored during employee creation. **Maximal length** : 32	string (varchar)
description *optional*	The description or landmark of the address. **Maximal length** : 100	string (varchar)
districtId *required*	The unique identifier of the district.	integer (int(32))

Name	Description	Schema
districtName *required*	The name of the district. This field will be ignored during employee creation. **Maximal length** : 32	string (varchar)
houseNo *required*	The number of the house. **Maximal length** : 32	string (varchar)
pinno *required*	The zip code of the location. **Maximal length** : 15	string (varchar)
region *optional*	The name of the region. **Maximal length** : 32	string (varchar)
stateId *required*	The unique identifier of the state.	integer (int(32))
stateName *required*	The name of the state. This field will be ignored during employee creation. **Maximal length** : 32	string (varchar)
streetName *required*	The name of the street. **Maximal length** : 32	string (varchar)

3.8. EmployeeBasicModel

Name	Description	Schema
contactNo *required*	The contact number of the employee. **Maximal length** : 15	string (varchar)
department *required*	The department the employee belongs to.	DepartmentModel
dob *optional*	Date of birth of the employee.	string (date-time)
doj *required*	date of joining of the employee.	string (date-time)
emailId *optional*	Email address of the employee. **Maximal length** : 30	string (varchar)
empFirstName *required*	First name of the employee. **Maximal length** : 60	string (varchar)
empId *required*	Unique identifier of the employee. This field will be ignored during employee creation.	integer (int(32))
empLastName *optional*	Last name of the employee. **Maximal length** : 60	string (varchar)

Name	Description	Schema
empMiddleName *optional*	Middle name of the employee. **Maximal length** : 30	string (varchar)
empType *required*	Permanent or Contract employee. **Maximal length** : 15	string (varchar)
entryBy *optional*	The username that entered the record. **Maximal length** : 32	string (varchar)
entryDate *optional*	Entry date and time.	string (date-time)
fatherName *optional*	Father's name of the employee. **Maximal length** : 120	string (varchar)
hrFlag *optional*	Indicates if the employee is a HR.	boolean
identityDocType *required*	The identity document type.	DocTypeModel
identityNumber *required*	The identification document number of the employee. **Maximal length** : 32	string (varchar)
maritalStatus *required*	Marital status of the employee. **Maximal length** : 10	string (varchar)
nationality *required*	Nationality of the employee. **Maximal length** : 32	string (varchar)
organization *required*	The organization the employee belongs to.	OrganizationModel
sex *required*	Sex of the employee. **Maximal length** : 10	string (varchar)
supervisorFlag *optional*	Indicates if the employee is a supervisor.	boolean
title *required*	Title of the employee. **Maximal length** : 5	string (varchar)
unit *required*	The unit the employee belongs to.	UnitModel

3.9. EmployeeHierarchyModel

Name	Description	Schema
cl *required*	The casual leaves of the employee.	integer (int(32))
hrId *required*	The employee id of the HR. **Maximal length** : 20	string (varchar)
maternityLeave *required*	The maternity leaves of the employee.	integer (int(32))
noticePeriodEndDate *optional*	The notice period end date.	string (date-time)
paternityLeave *required*	The paternity leaves of the employee.	integer (int(32))
pl *required*	The paid leaves of the employee.	integer (int(32))
probationPeriodEndDate *required*	The probation period end date.	string (date-time)
specialLeave *required*	The special leaves of the employee.	integer (int(32))
status *required*	The status of the employee. **Maximal length** : 4	string (varchar)
supervisorId *required*	The employee id of the supervisor. **Maximal length** : 20	string (varchar)

3.10. EmployeeModel

Name	Description	Schema
employeeAddlDetails *optional*	The employee additional details.	EmployeeAddlDetailsModel
employeeAddress *optional*	The employee address.	EmployeeAddressModel

Name	Description	Schema
employeeBasicDetails *optional*	The employee basic model.	EmployeeBasicModel
employeeHierarchy *optional*	The employee hierarchy.	EmployeeHierarchyModel
employeeOptionalBenefit *optional*	The employee optional benefit.	EmployeeOptionalBenefitModel
employeeProfile *optional*	The employee profile.	EmployeeProfileModel
employeeSalary *optional*	The employee salary.	EmployeeSalaryModel

3.11. EmployeeOptionalBenefitModel

Name	Description	Schema
benefitValue *required*	The benefit value for the employee.	number (double)
frequency *optional*	Frequency of the benefit in months.	integer (int(32))
iterations *optional*	Number of times the benefit will be given to the employee. Value -1 means infinite, Value 0 indicates the current benefit has been completed and will no longer be given to the employee. After every due date when the payroll runs, the iterations will be reduced by one.	integer (int(32))
nextDueDate *optional*	The next due date of this benefit for the employee.	string (date-time)
optSalaryComponent *required*	Optional salary component.	SalaryOptComponentModel
startDate *required*	The start date of this benefit for the employee.	string (date-time)
stopDate *optional*	The stop date of this benefit for the employee.	string (date-time)

3.12. EmployeeProfileModel

Name	Description	Schema
comments *optional*	The comments. **Maximal length** : 52	string (varchar)
description *optional*	The description of the employee. **Maximal length** : 256	string (varchar)
qualification *optional*	The highest qualification of the employee. **Maximal length** : 32	string (varchar)

3.13. EmployeeSalaryModel

Name	Description	Schema
salaryComponent *required*	Salary Component.	#definitions/SalaryComponentModel
salaryValue *required*	The salary value of the employee for the given salary component.	number (double)

3.14. ErrorModel

Name	Description	Schema
errorMessage *required*	Detailed Error Message.	string

3.15. GradeModel

Name	Description	Schema
gradeId *required*	To uniquly identify a grade. This field will be ignored during grade creation.	integer (int(32))
gradeName *required*	Name of the grade **Maximal length** : 32	string (varchar)

3.16. JobRoleModel

Name	Description	Schema
NoticePeriod *required*	Notice period if the employee returns after the probation period.	integer (int(32))
grade *required*	Name of the grade model	#definitions/GradeModel
jobRoleId *required*	To uniquly identify the job role within the organization. This field will be ignored durin the job role creation.	integer (int(32))
organization *required*	Name of the organization model	#definitions/OrganizationModel
probationNoticePeriod *required*	Notice period if the employee resigns during the probation period.	integer (int(32))

3.17. OptionalSalaryModel

Name	Description	Schema
maxAllowedLimit *required*	The maximum allowable limit.	number (double)
optSalaryComponent *required*	The Salary component.	#definitions/SalaryOptComponentModel
salaryValue *required*	The default value of the salary Component for the paricular job role id.	number (double)

3.18. OrganizationModel

Name	Description	Schema
address *required*	Address of the organization **Maximal length** : 256	string (varchar)
orgId *required*	unique identifier of the organization. This field will be ignored during organization creation.	integer (int(32))
orgName *required*	Name of the organization. **Maximal length** : 100	string (varchar)
orgType *required*	The organization type.	OrganizationTypeModel

3.19. OrganizationTypeModel

Name	Description	Schema
description *optional*	The description of the organization type. **Maximal length** : 100	string (varchar)
orgTypeId *required*	unique identifier of the organization type. This field will be ignored during organization type creation.	integer (int(32))
orgTypeName *required*	The name of the organization type. **Maximal length** : 100	string (varchar)

3.20. SalaryComponentModel

Name	Description	Schema
compId *required*	Unique identifier of the salary component. This field will be ignored during salary component creation.	integer (int(32))
compName *required*	Name of the salary component. **Maximal length** : 32	string (varchar)
description *optional*	Description of the salary component. **Maximal length** : 32	string (varchar)
salComponent *optional*	Salary componenet. **Maximal length** : 32	string (varchar)

3.21. SalaryModel

Name	Description	Schema
maxAllowedLimit *required*	The maximum allowable limit.	number (double)
salaryComponent *required*	The Salary component.	SalaryComponentModel
salaryValue *required*	The default value of the salary Component for the paricular job role id.	number (double)

3.22. SalaryOptComponentModel

Name	Description	Schema
creditDebitFlag *required*	This flag indicates whether the amount will be credited © to the employee's account or debited (D) from the employee's account. **Maximal length** : 1	string (varchar)
description *optional*	Description of the salary component. **Maximal length** : 32	string (varchar)
empOptOutFlag *required*	This flag indicates whether the employee can opt out from the benefit (Y) or no (N). **Maximal length** : 1	string (varchar)
endDate *required*	The end date of this benefit in company policy.	string (date-time)
frequency *required*	Frequency of the component in months.	integer (int(32))
optCompId *required*	Unique identifier of the optional salary component. This field will be ignored during optional salary component creation.	integer (int(32))
optCompName *required*	Name of the salary component. **Maximal length** : 32	string (varchar)
salOptComponent *optional*	Salary componenet. **Maximal length** : 32	string (varchar)

3.23. StateModel

Name	Description	Schema
countryId *required*	Unique identifier of a state.	integer (int(32))
countryName *required*	Name of the country. **Maximal length** : 32	string (varchar)
stateId *required*	Unique identifier of a state.	integer (int(32))
stateName *required*	Name of the state. **Maximal length** : 32	string (varchar)

3.24. UnitModel

Name	Description	Schema
address *required*	Address of the unit. **Maximal length** : 256	string (varchar)
empIdPrefix *required*	Prefix of employee id. **Maximal length** : 10	string (varchar)
empIdSeqName *required*	**Maximal length** : 20	string (varchar)
orgId *required*	unique identifier of the organization.	integer (int(32))
orgName *required*	Name of the organization. This field will be ignored during unit creation. **Maximal length** : 100	string (varchar)
unitId *required*	To uniquly identify an unit.	integer (int(32))
unitName *required*	Name of the unit. **Maximal length** : 100	string (varchar)