

APIs

Overview

There are two APIs that we must provide currently. One to get employees and one to generate the paychecks for accounting. These provide ways for other silos to gather information that we store.

Endpoints

Method	URL	Description
GET	/api/employee	Gather employee details with filters. All other silos will likely use this endpoint. Results are returned in a JSON object with pagination.
POST	/api/salary	Accounting will use this endpoint as a trigger. This will calculate each employees paycheck (including base pay, commission pay, and bonus) and return a JSON consisting of the pay start date, pay end date, and the lump sum that needs to be deducted from accounts. This will also store a paycheck history object in HR's database.

Details

GET /api/employee

Returns a list of employees and their information based on filters provided in the url params

Request Object:

Name	Type	Required	Description	Example
------	------	----------	-------------	---------

id	integer	no	ID of the employee.	21
department	string	no	Enables filtering of employees based on department	"Sales"
position	string	no	Filtering employees based on the position	"Sales Representative"
region	string	no	Filtering employees based on region they belong too.	"Southwest"
pageNum	integer	yes	The page number of the information the user wants to return based on limit	2
limit	integer	yes	The maximum number of records returned per page	26

Example Requests

URL: /api/employee?id=3

Name	Type	Description	Example
id	integer	ID of the employee.	3
firstName	string	First name of the employee	"Zach"
lastName	string	Last name of the employee	"D"
dateOfBirth	string	DateTime string of the date of birth of the employee	"1996-11-05T00:00:00"
regionName	string	The name of the region the employee is working	"Rochester"
roleName	string	The role of the employee	"Admin"
departmentName	string	The department that the employee is working in	"Accounting"
positionName	string	The position title that the employee is working	"Board Member"

/api/employee?pageNum=1&limit=4

Name	Type	Description	Example
------	------	-------------	---------

data	array	Contains an array of objects. Each object pertains to one employee (as described above)	Example Below
Page	object	Contains parameters pertaining to the url for the next pagination page.	Example below

Example Response Object

```
{
  "data": [
    {
      "id": 1,
      "firstName": "Zachary",
      "lastName": "DiPasquale",
      "dateOfBirth": "1996-11-05T00:00:00",
      "regionName": "Rochester",
      "roleName": "Admin",
      "departmentName": "HR",
      "positionName": "CEO"
    },
    {
      "id": 3,
      "firstName": "Zach",
      "lastName": "D",
      "dateOfBirth": "1996-11-05T00:00:00",
      "regionName": "Rochester",
      "roleName": "Admin",
      "departmentName": "Accounting",
      "positionName": "Board Member"
    }
  ],
  "page": {
    "nextPage": "?pageNum=2&limit=4"
  }
}
```

POST /api/salary

Calculates and stores all paychecks for all employees within a given range. Returns the total amount expended by the paychecks and the dates the paychecks were calculated for.

Request Object

Name	Type	Description	Example
startDate	string	Indicates the start for the range of dates paychecks should be calculated for.	"2008-05-01 12:00:00Z"

endDate	string	Indicates the end for the range of dates paychecks should be calculated for. This should be after the start date.	"2008-05-02 12:00:00Z"
quarterlyBonus	string	Signifies a boolean indicating whether the quarterly bonus should be applied to a paycheck.	"true"
monthlyBonus	string	Signifies a boolean indicating whether the monthly bonus should be applied to a paycheck.	"false"

Return Object

Name	Type	Description	Example
startDate	string	Indicates the start for the range of dates paychecks have been calculated for.	"2008-05-01 12:00:00Z"
endDate	string	Indicates the end for the range of dates paychecks have been calculated for.	"2008-05-02 12:00:00Z"
lumpSum	string	Signified the total amount of money that has been expended by the payout of all paychecks for all employees within the range.	"3400002.24"

APIs to be used

Name	Silo	Description
Sales Revenue Information	Sales	Within salary calculation, there is a need to know individual sales, company sales, and regional sales in order to properly calculate bonus and commission for paycheck. As such, Sales has agreed that there will be an endpoint available to gather this information.
Customer Support Employee Reviews	Customer Support	When an employee goes to their profile page, they are allowed to see various reviews that people have left for them such as managers and customers. As such, Customer support has opened an endpoint that allows HR to GET customer support reviews for a specific employee.

APIs

Method	URL	Description
GET	/auth/	This feature is slated for R2. This API endpoint will take in username and a password of an employee. Verify the employee information is correct and send back a json web token (jwt).

A JWT is returned when authenticated with our system. The JWT returned in Base64 encoded. At this time it is not encrypted and can easy be spoofed. We are planning on discussing encryption in future team coordination meetings. As defined by the JWT standar there are three parts to a JWT: Header, Payload and Secret; Each separated with a '.' once encoded. The JWT is signed to verify that the token has not changed along the way. An example JWT is displayed below:

Header:

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

Payload:

```
{
  "result": {
    "username": "test2",
    "password": "testing123",
    "id": 1
  },
  "iat": 1522635247
}
```

```
HMACSHA256(
  base64UrlEncode(Header) + "." +
  base64UrlEncode(Payload),
  `secretkey`
)
```

POST /auth/login

Logs in a user by authenticating against the database.

Request Object

Name	Type	Description	Example
username	string	The username of the account	"test2"
password	string	Password of the account. This is not to be hashed.	"testing123"

Return Object

Name	Type	Description	Example
status	boolean	Indicates whether the request was successful at authenticating or not.	true
token	string	JSON web token that is to be used while interacting with the api gateway endpoints.	See Below
		"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyZXN1bHQiOnsidXNlcm5hbWUiOiJ0ZXN0MilsInBhc3N3b3JkljoidGVzdGluZzEyMyIsInR5cGUiOiJqdXN0b21lciJ9LCJpYXQiOiJlMjMzUyNDd9.xod4sN7kAC7Fj_uHCAekMJ-ORGuwnQtYxcGeZdaPoml"	See above for decoded JWT

POST /auth/create

Create an account for a general user with a "username" and "password"

Request Object

Name	Type	Description	Example
username	string	The username of the account	"test2"
password	string	Password of the account. This is not to be hashed.	"testing123"
type	string	Type of account	"employee" or

			"customer"
--	--	--	------------

Return Object

Name	Type	Description	Example
status	boolean	Indicates whether the request was successful at authenticating or not.	true
token	string	JSON web token that is to be used while interacting with the api gateway endpoints.	See Below
		"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyZXN1bHQiOnsidXNlcm5hbWUiOiJ0ZXN0MilsInBhc3N3b3JkljoidGVzdGluZzEyMyIsInR5cGUiOiJqdXN0b21lciJ9LCJpYXQiOiJlMjI2MzUyNDd9.xod4sN7kAC7Fj_uHCAekMJ-ORGuwnQtYxcGeZdaPoml"	See above for decoded JWT
message	String	Indicates what type of errors or success the request has encountered	"Account Created"