TASK MANAGEMENT SYSTEM DOCUMENTATION

Demo Users,

user1=>[victor@gmail.com](mailto:victor@gmail.com), password=>victor123, role=>normal user

user2=>[brenda@gmail.com](mailto:brenda@gmail.com), password=>password, role=>nomal user

user3=>[alice@gmail.com](mailto:alice@gmail.com), password=>alice123, role=> department head

## Task Management App ##

This web application allows supervisors and their juniors to easily collaborate in managing their day to day tasks

### How to setup on server ###

- clone the repository

- run \*composer install\*

### How to setup for local development ###

- clone the repository

- run \*composer install\*

- Install NodeJS

- Install Gulp globally by running - \*npm install -g gulp\*

- run \*gulp && gulp watch\* - this will create a local server on port localhost:3000. Ensure Laravel development server is running so as to enable API calls to the server

### Seeding database ###

- Run \*php artisan migrate --seed\* to migrate the database tables and seed some information

### Demo ###

Seeding the database with the above command will create the following users and respective roles

1. Alice Kiprop - alice@gmail.com - role "department head"

1. Brenda Wambui - brenda@gmail.com - role "normal"

1. Victor Macharia - victor@gmail.com - role "normal"

### API Spec ###

The entire back end is completely API Based. You will see the API Specification file in this folder, called ApiSpec.json

TASK MANAGEMENT SYSTEM API SPECIFICATION

Login

POST /api/users/login

{

"user": {

"email":

"password":

}

}

Create Tasks

POST /api/tasks

Constraints: Authenticated users

This api will also subscribe the given users to a task

{

"tasks": {

"id":

"user\_id":

"department\_id":

"category\_id":,

assignee\_id:,

"name": "title of task"

"summary": "summary of task"

"access\_level": "public"

"prority": "Normal"

"due\_date": "12/8/2017"

"progress\_status": "0",

"notify": {

department\_subscribers: [],

user\_subscribers: []

}

}

}

Post Progress message to a Task

POST /api/task/{task}/progress

Constraints: Authenticated and Authorized[Policy]

Authorized means you are either the owner(reporter) of the task, or task is assigned to you

{

"progress": {

"user\_id"

"task\_id"

"overall\_status"

"message

"progress\_status"

}

}

Subscribe to a task

POST /api/task/{task}/subscriptions

Constraints: Authenticated

{

"subscribe": true

}

Unsubscribe from a task

DELETE /api/task{task}/subscriptions

Constraints: Auth

{

"subscribe": false

}