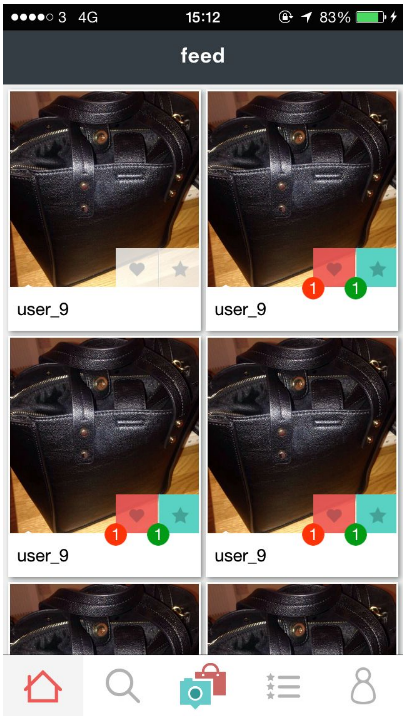
## Task



Build a JavaScript application using a framework of your choosing.
The application should use the Frockadvisor JSON API as its data source.

The application should allow the following:
1- User login with email and password
2- Display user feed. This is a collection of the ‘fotocons’ that are posted by users that you follow. Fotocons are images with captions. You can see how the feed looks in our iOS app (screenshot on the right).

Please take into consideration:
- Appropriate best practice structure/organisation for the application
- Any security considerations
- Basic UI design (it doesn’t have to be pretty, just have some sort of structure so that it is useable)

## API docs

The two API endpoints that are required for this task are:

```
POST /users/login
GET /feed
```

They are documented in the following pages.

The `/users/login` call requires a `Frockadvisor-Api-Key` header.
The `/feed` call requires a `Frockadvisor-Api-Key` header and an `Access-Token` header. The access token is returned from the login call.

**API key:**
`23425j5oi3jtrijgpofhpfokypo5kpokpokokpdofgkdpogkdpfokgpok3po45j30596405849445645644564joidjfogif`

**Valid login credentials:**
email - hello@frockadvisor.com
password - password1234

The API does not return image URLs for the feed – instead it returns unique hash identifiers for each fotocon. The client can then construct the image URL from the hash. The structure of the image URL is as follows:

`http://frockadvisor-api-storage.s3.amazonaws.com/fotocon/[fotocon_hash].jpg`

e.g.
`http://frockadvisor-api-storage.s3.amazonaws.com/fotocon/2f8191e0-79ad-11e5-a3f5-cffc035e1db3.jpg`

It is recommended that you test against the API using a HTTP client like Charles or a similar browser extension (https://chrome.google.com/webstore/detail/advanced-rest-client/hgmloofddffdnphfgcellkdfbfbjeloo?hl=en-US), to verify the API key works and see example responses.

At the end push the code to a Git repository where it can be viewed.

# User Login

```
POST /users/login
```

**Payload**

```
{
  "email":"user@frockadvisor.com",
  "password":"password1234"
}
```

**Response**

Success:

```
# Response returns retailer_id or user_id depending on if the user is an
end-user or retailer
# HTTP 200
{
  "status":"success",
  "data":{
    "token":"oijoicvb-234sdcvdfg-23423kjnkjnkxj-osjnon3on",
    "user_id":123
  }
}
```

Error:

```
# HTTP 500
{
  "status":"error",
  "errors":{
    "message":"User not found",
    "code":"1443102185-3392"
  }
}
```

## User Feed

```
GET /feed
```

**Response**

Success:

```
# HTTP 200
{
  "status":"success",
  "data":[
    {"fotocon_id":152,"caption":"Check out this thing that I
like!","fotocon_hash":"a466fdc0-70d1-11e5-bc81-516edd6ee418","created_at":"20
15-10-12
12:09:01.336319+01","loves_count":2,"wants_count":3,"comments_count":1,"user_
id":462,"username":"user_49","user_hash":"a08b0160-70d1-11e5-a1d1-33bd3cf9308
9","retailer_name":null,"retailer_id":null,"retailer_hash":null,"want":true,"
love":false},
    {"fotocon_id":151,"caption":"Check out this thing that I
like!","fotocon_hash":"a452f400-70d1-11e5-8c56-21414d0f788f","created_at":"20
15-10-12
12:09:01.205205+01","loves_count":0,"wants_count":0,"comments_count":0,"user_
id":461,"username":"user_48","user_hash":"a0813b40-70d1-11e5-b9e9-6d3ba9506db
f","retailer_name":null,"retailer_id":null,"retailer_hash":null,"want":false,
"love":false},

...etc...

  ]
}
```

Error:

```
# HTTP 500
{
  "status":"error",
  "errors":{
    "message":"There was a database error",
    "code":"1443103261-336"
  }
}
```