

CS487 OAD Acceptance Test

1. Try to go to /administrator.html. You should be redirected to a login page.
2. Attempt to login with an invalid username and password. You should be rejected.
3. Attempt to login with username:john password:password. You should be redirected to /administrator.html
4. Fill out the fields containing advertiser making sure to use a unique email. The advertiser should appear under "Manage Listings" > "Create Listing" > "Advertiser Name" if successfully created (after the page is refreshed) and should return with a successful response in the console.
5. Fill out the fields containing advertiser repeating the same email but a different business name. The advertiser should not be created.
6. In the "Manage Listings" tab, create a Category with subcategories. Multiple subcategories should be added to a list when the "Add Subcategory" button is clicked. The Category and its subcategories should appear under the "Create Listing" tab after the page is refreshed.
7. Fill out the "Create Listing" form making sure each field is selected. Select 'Normal' as the type. Clicking Submit once the form is selected should return a successful response. NOTE: THE LISTING IMAGE URL MUST BE HTTPS AND MUST BE A VALID IMAGE.
8. Repeat #7 with different information selecting 'Homepage Featured' as the featured type.
9. Repeat #7 with different information selecting 'Category Featured' as the featured type.
10. Go to /index.html. The homepage featured listing should be seen at the top of the page and the category created should be below it.
11. Click on the + button. The category featured listing should be displayed with the subcategory created below it.
12. Click the + button on the subcategory. The listing created should be displayed below.
13. Every + button on every listing should expand the listing to display the description, a MAP button and a WEBSITE button.
14. The category featured listing should be displayed to the left of the listings with a header defining what category it belongs to (desktop view only).

API Definitions

Title: Display Advertisers

URL: /admin/advertiser

Method: GET

Success Response: 200

Error Response: 401 UNAUTHORIZED

Title: Create Advertiser

URL: /admin/advertiser

Method: PUT

Data Params:

```
{  
  "name" : [string],  
  "phone" : [string],  
  "website" : [string],  
  "email" : [string]  
}
```

Success Response: 200

Error Response: 401 UNAUTHORIZED

Title: Display Categories

URL: /admin/category

Method: GET

Success Response: 200

Error Response: 401 UNAUTHORIZED

Title: Create Category

URL: /admin/category

Method: PUT

Data Params:

```
{  
  "name" : [string],  
  "subcategories" : [array{string}]  
}
```

Success Response: 200

Error Response: 401 UNAUTHORIZED

Title: Create Listing

URL: /admin/listing

Method: PUT

Data Params:

```
{  
  "name" : [string],  
  "advertiser" : [string],  
  "image" : [string],  
  "address" : [string],  
  "city" : [string],  
  "neighborhood" : [string],  
  "phone" : [alphanumeric],  
  "description" : [string],  
  "website" : [string],  
  "startDate" : [alphanumeric],  
  "endDate" : [alphanumeric],  
  "featureType" : [string],  
  "category" : [string],  
  "subcategories" : [string]  
}
```

Success Response: 200

Error Response: 401 UNAUTHORIZED

Title: Display All Listings

URL: /listing

Method: GET

Success Response: 200

Title: Find Listings

URL: /listing/search?query=[string]&city=[string]&neighborhood=[string]

Method: GET

Success Response: 200

Title: Display Locations

URL: /location

Method: GET

Success Response: 200

Issues

Issues that need IMMEDIATE attention

- Admin Login functionality needs better implementation. The login information is currently stored as plaintext with no way to change user information or create new users.
- Create Listing image field needs type checking and exception handling to prevent non-https sites from being passed. Non-https images passed to the back end will BREAK the GET listings request.

Issues that should be addressed before deployment

- Front end Find functionality needs to be fixed before “Find Listing” use case works.
- Report use cases have not been implemented.
- Start and End Date on listings is just stored; listings will still display even if date is outside of start-end range.

Issues that are not essentially, but should eventually be addressed.

- Addresses when creating listings should be broken up into multiple lines if too long in order to maintain appropriate listing display.
- REST request returns should be REST standard compliant. The requests currently return plaintext such as “Advertiser Created” instead of the appropriate REST response.
- URLs should be changed to be REST standard compliant.
- More extensive exception handling can be done on both the front end and the back end.
- Front End should be reworked to be more Object-Oriented and less rigid.