

SW Engineering CSC648/848 Summer 2018

# Shallot

Team 02

James Clark

[jclark1@mail.sfsu.edu](mailto:jclark1@mail.sfsu.edu)

Dandan “Jenny” Cai

Michael Lama

Patrick Cocio

Roy Telles

Xisheng “Sam” Zheng

Milestone 4

August 4th, 2018

## **Product Summary:**

Have you or a friend ever needed to find royalty-free, free-use images on the internet, but didn't know where to find one? Have you ever been frustrated with how difficult it is to navigate other stock photo websites to find the content you want? Look no further! Shallot is the next best stock photo website in the market that no one knows about yet! This website will be responsible with providing up to 50 users at a time, all around the world, with royalty-free stock photos for their own personal use, for FREE!

Shallot is a website designed to provide the user with an easy-to-use stock photo browsing and sharing experience. Users will be able to:

- Browse our content by using our powerful search bar
- Register an account with us, which will allow them to:
  - Download the images they like at the click of a button
  - Upload the images they want to share with an easy-to-use interface that allows them to:
    - Add a Title to their image
    - Add a Description to their image

Our users' uploaded content is monitored by an administrator from our team to ensure a pleasant experience for all of our users. We take the steps necessary to prevent content that is inappropriate or not free-to-use from showing up in our site.

Shallot's website can be found here: [shallotco.com](http://shallotco.com)

Note: Our deployment server is also our test server and is not guaranteed to show a stable version of the website until after August 7th, 2018.

## Usability Test Plan:

- **Scope** (What are we testing?): We will be testing the search bar.
- **Purpose** (Why are we testing?): We want to ensure the functionality and usability of the search bar (ease of locating, ease of use, etc.).
- **Sessions** (What is sample size?): A small group (4-5 people).
- **Equipment** (Technology, materials, and resources required): A laptop to visit website will be required.
- **Participants** (Target users, source methods, compensation, role): Participants shall be personas defined in documentation. Preliminary participants shall be CEO and CTO, and close acquaintances. Personal acquaintances shall be compensated appropriately by the team member conducting the test. Materials provided shall be the Likert scale. The role of the participants shall be to provide feedback on the usability of the website's search bar.
- **Scenarios** (Who is the user? Why are they here? What goals do they have?): The user is a student, wanting to locate images of trees to include in a report about nature.

Goal/Output:	Search for tree images using the search bar
Inputs:	Invitee Name
Assumptions:	Laptop already at web site homepage
Steps (assumed):	<ol style="list-style-type: none"><li>1. Navigate to search bar</li><li>2. Enter "tree" in search bar</li><li>3. Hit enter</li><li>4. Achieve results</li></ol>
Success Criteria:	Images of trees successfully found
Notes:	

Questionnaire					
Questions	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
1. The search bar was easy to navigate					
2. The item was easily searchable					
3. I was able to achieve results of item					
Comments:					

### **QA Test Plan:**

1. The objective of this test plan to test the features based on P1 requirements related to the implementation of Shallotco.com
2. Provide detailed activities required to prepare for and conduct the web application test.
3. Define test tools and environment needed to conduct the test.

### **HW Set Up:**

Our application is a web-based application, so we do not have hardware involved.

### **SW Set Up:**

- Get AWS EC2 instance
- Install Apache2
- Install python-flask
- Install Pillow
- Install virtual environment
- Set up Gitup

The following table will illustrate what have been tested and what have not been tested.

Features to be Tested	Features not to be tested
Log In	Can handle 50 upload actions at the same time
Register	User login info shall not be accessed by other users
Upload	Will application save the work after power loss
Download	Can handle 50 users login at the same time
Admin Approval of Images	
Search through Category	
Search one particular image	
Environment test	
Cross-browser test	

Test #	Test Title	Test Description	Test Input	Expected Output	Test Results(chrome) (Pass/Fail)	Test Results(safari) (Pass/Fail)
1	Log in	Check with registered user if it is able to login	Registered user name and user password	It shall redirect from login page to home page with trending images displayed	Pass	Pass
2	Upload Image	Check if users are allowed to upload images	A image file from user with image name, description and category selected	A successful uploading message will be displayed, upload page will still be displayed	Pass	Pass
3	DownLoad Image	Check if users are allowed to download images	Click on download button on individual image	Users shall be able to see the downloaded image in the specified folder	Pass	Pass

## Code Review:

The coding style we chose involves using proper indentation, names that relate well to the content it represents, and comments. As you can see below, this is how this is enforced:

Review Request:

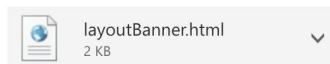
CSC648 Search Bar html



Roy Telles <rtelles64@gmail.com>

Today, 9:35 PM

James Clark ▾



Download Save to OneDrive - San Francisco State University

Hi James,

Please find attached to this email the html file containing the code for the search bar for code review.

Thank you,

Roy Telles, Jr.

## Code Submitted (that is related to search bar):

```
<!-- SEARCH BAR goes here -->
<div class="container" style="margin-top: 20px;">
  <div class="row">
    <div class="col-xs-8 col-xs-offset-2">
      <div class="input-group">
        <div class="input-group-btn search-panel">
          <form action="/Search" method="post">
            <button type="button" class="btn btn-default dropdown-toggle" data-toggle="dropdown"
              name="category">
              <!-- DROPDOWN menu for categories -->
              <span id="search_concept">All</span> <span class="caret"></span>
            </button>
            <ul class="dropdown-menu" role="menu">
              <li><a href="#All">All</a></li>
              <li><a href="#Clip Art">Clip-art</a></li>
              <li><a href="#Fashion">Fashion</a></li>
              <li><a href="#Food">Food</a></li>
              <li><a href="#Nature">Nature</a></li>
            </ul>
            <!-- END category select -->
          </div>
          <input type="hidden" name="category" value="{{request.form.category}}" id="search_param">
          <input type="text" class="form-control" name="search" maxlength="30" placeholder="Search
            term..." value="{{request.form.search}}">
          <span class="input-group-btn">
            <button class="btn btn-default" type="submit"><span class="glyphicon glyphicon-search">
          </span></button>
          </span>
        </form>
      </div>
    </div>
  </div>
</div>
```

## Checking Functionality:

Nature ▾

Search term...

Q

## Results

- 13 of 13 results for ""



After the code is checked for comments, indentation, functionality, and organization, feedback is given. Since there was nothing more to correct for the search bar code, the following was sent back:



James Clark

Today, 11:55 PM

Roy Telles <rtelles64@gmail.com> ▾

Thank you Roy,

The code looks well-commented, well-indented, and the feature is functional!

Please submit this to our GitHub repo on the "dev" branch.

Thank you!

James Clark



### **Self-check on best practices for security:**

In our database, we are protecting every major asset, including the input of the search bar and the passwords, primarily. However, every input field is protected via limiting the amount of characters allowed in the text box as a first line of defense and as a second line of defense, we are using “%s” validation in our back end Python file. “%s” validation is extremely useful in combating SQL injections because it takes the input which is being placed into an SQL command, and looks at it literally instead of as an extension of the SQL command. This removes the ability to successfully inject an SQL command to drop a table or display all user information as the command would just be seen as input like any other. Also, if an “incorrect” input is inputted, such as “-1”, it’s simply seen as an incorrect input, returns zero results, and causes our most popular images to be displayed. As for passwords, each password gets encrypted via the sha256 algorithm and verified via an algorithm included in the sha256 package. Thus, we never know what passwords our users are using and without cracking sha256, nobody else will be able to either.

### **Self-check on Adherence to original Non-functional specs:**

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0 (some may be provided in the class, some may be chosen by the student team but all tools and servers have to be approved by class CTO). **DONE**
2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of all major browsers: Mozilla, Safari, Chrome. **ON TRACK**
3. Data shall be stored in the team's chosen database technology on the team's deployment server. **DONE**
4. No more than 50 concurrent users shall be accessing the application at any time **DONE**
5. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users. **DONE**
6. The language used shall be English. **DONE**
7. Application shall be very easy to use and intuitive. **ON TRACK**
8. Application shall render well on mobile devices (UI shall be responsive) **ON TRACK**
9. Google analytics shall be added **DONE**
10. No e-mail clients shall be allowed **DONE**
11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated. **DONE**
12. Site security: basic best practices shall be applied (as covered in the class) **DONE**
13. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development **DONE**
14. The website shall prominently display the following exact text on all pages *"SFSU Software Engineering Project CSC 648-848, Summer 2018. For Demonstration Only"* at the top of the WWW page. (Important so as to not confuse this with a real application). **DONE**