Group Project Timeline

Each Group Project (GP) is to be submitted with an email message, including the URL of your project.

GP 1

Produce a photo upload page. Utilize an OAUTH mechanism to authenticate your interaction with Amazon Web Services. This will be called Page 1, for future reference.

In configuring OAUTH, a facebook developer account should be created, and a URL to your Page 1 in AWS.

Integrate a responsive format to display your results: after uploading the image, the image should appear within the page, inside a styled div, using bootstrap.

After upload, a footer, jumbotron, alert or other tool should be used to communicate with the user, telling them their new image is online, in the cloud.

Resources

1. unit 1 source code, GitHub
2. bootstrap album example, http://getbootstrap.com/docs/4.0/examples/album/
3. youtube videos for unit one, on our playlist, linked on GitHub readme page

GP 2

Utilize AWS CRUD code (create, read, update, delete) to insert a row of data into your nosql database (dynamodb), in the cloud.

Persistent logins should be made for each CRUD operation, going forward (create, update, delete, read).

In your album layout from GP 1, adjust your edit button to link to an additional page (page 2). Instead of Edit, in the original Album, it should read **make**. On page 2, set up several boxes (inputs) to accept several pieces of information from the user, regarding the photo.

Redo the photo upload so that you put the photo upload part on Page 2, also. The name of the photo should be put to the database, in the same put statement.

Page 2 should have a button which allows for the inputs to go to the database, with a put operation. Verify that the data passed to the database afterward.

GP 3

Use your AWS SDK to **get** the row, from your database, which you put in GP 2.

When you load Page 1, get the row you created in GP 2, and load the information on the photo. The photo should appear in the image, with the URL in the database.

The data you get should appear inside the **card** div, within a **col-md-4**.

GP 4

Encode the **Edit** button on Page 1 to direct to an edit page. Utilize URL parameters to take each piece of data in the card div and transport them to the new page, Page 3.

On Page 3, output each URL in a div, so you confirm you can move data between two pages, using URL parameters

GP 5

On Page 3, take the URL parameters from Page 1. Display each parameter in a set of input boxes as you did in GP 2, Page 2. Instead of doing a put, do an edit on the row, using changes to the data.

When Page 1 loads, we should observe the changes in the data you made from Page 3 (edit page).

Be sure to verify that the Edit button within each card will navigate the user to Page 3 (edit page).

GP 6

On a practice page, do a query on your existing photo database, after uploading several photos, and creating several rows in the database.

Form new divs for each row you get back from the database. No bootstrap formatting should be used.

GP 7

Using your query from GP 6, format the output you receive on your practice page so that the divs you produced in GP 6 are now cards, falling within bootstrap container/row format.

Each row should now appear in your album, with interactive buttons. The **make** button in each card should lead to page 2, where new images can be uploaded, with their associated information.

The Edit button within each card should direct to Page 3, the edit page, where all the information in the card should appear, via URL parameters. The information should be editable, as in GP 4.

If you have missed points for things in prior units, they will be accepted for credit for GP 7.

GP 8

Refinements, and completions. Use this week to refine and wrap up any details related to your site. Any credit missed for any item can be made up during GP 8.