

CSE451 Final Project

The final for CSE451 is T 8:00 AM - 10:00 AM - 05/14/19 To 05/14/19 - 006 Benton Hall. We will meet as a class at this time for you to submit and demonstrate your final project.

This project will use:

- Creation of a REST API server
- Use of O-AUTH
- Invoking ToDoist APIs
- Using a new Census API
- Alexa Skills that will use call ToDoist and will also call other RestAPIs
- Storing data in a mysql database
- Use of AJAX calls from a browser based application.

Phase 1 will be for Alexa to get access to the users ToDoist Token. To do this start with the provided Skill and modify it in anyway necessary so that if the ToDoist Token for the current user is not present, the token is obtained.

Phase 2 will be to create a REST API

Phase 3 will be to implement a Web Interface

Phase 4 will be to complete the Alexa Skill from the template code.

You can complete as many phases as you desire. Points for each phase are denoted in the details. You MUST complete phase 1 and can complete the other phases as you desire. They are independent of each other.

Details:

- Phase 1 - MUST BE COMPLETED
 - Title: Alexa OAuth
 - Points: 40
 - The provided code has both the Json for Alexa and the Lambda js code.
 - If the token exists the program will read out the first task. This function exists and does not need to be changed. If the token does not exist, the code calls a function for oauth which you have to implement.
 - Starter code is <https://gitlab.csi.miamioh.edu/campbest/cse451-campbest-node-s19/tree/master/FinalExam>

- Note: Make sure to use the blueprint "trivia" when setting up the lambda code as this will include the needed dynamodb libraries.
- Phase 2
 - Title: Rest API
 - Points: 20
 - Implement a REST API on ceclnx01 that meets the following specifications.
 - Create a mysql database (the create script is in the git project associated with this assignment).
 - Create the following APIs
 - get Comments
 - url: /api/v1/Comments
 - Method: get
 - JSON_IN: N/A
 - JSON_OUT: {"msg":string,"Comments":[{"id":int,"comment":string,"createTime":int (unixtime)},...]}
 - get Last Comment
 - url: /api/v1/Comments/Last
 - method: get
 - JSON_IN: N/A
 - JSON_OUT: {"msg":string,"id":int,"comment":string,"createTime":int}
 - Add Comment to List
 - url: /api/v1/Comment
 - method: put
 - JSON_IN: {"password":"class","comment":string}
 - JSON_OUT: {"msg":string}
 - Delete Comment
 - url: /api/v1/Comment/:id
 - method: put
 - JSON_IN: {"password":"class"}
 - JSON_OUT: {"msg":string}
 - Delete all comments
 - url: /api/v1/Comments
 - method: Delete
 - JSON_IN: {"password":"class"}
 - JSON_OUT: {"msg":string}
 - Store all data inside the mysql database
- Phase 3
 - Title: Web Interface
 - Points: See Below, have to be done in order.
 - Create a web-interface that will call the Above Rest API
 - Create a table that lists the Comments in the collection. 5 points
 - For each Comment, create a link that will delete the comment. 5 Points

- Have a form that will add a comment with good error checking and reporting. (10 points)
 - Must be an AJAX based solution. NO PHP.
- Phase 4
 - Title: Create Alexa Skill
 - Points: 20
 - Create an alexa Skill that will add and read the last comment.

All code used (lambda, javascript, html...) will be submitted to Gitlab and a link shared with the instructor.

Submit to canvas.

- 1) A statement details which phases are working and their status.
- 2) Clickable
- 3) A clickable link to your gitlab. All code including the lambda code should be here.
- 4) A link to a screencast video showing all aspects of your system working. (eg: screencast the alexa app and the webpages.) No more than 2 minutes long.