# Project 4 – User Management ( FH Inventory Application)

PUT YOUR NAME HERE

|  |
| --- |
| NoName Mchiggens |

PUT GROUP NUMBER HERE

|  |
| --- |
| X99 |

PUT THE NAMES OF YOUR BUDDIES HERE

|  |
| --- |
| Samuel D. Test, Ima Nuther, Buddy Finkerbean |

## Expectations

* Instructor Guidance – **High**
* Independent Effort – **Moderate**
* Originality – **Low**
* Teamwork – **High**

## Introduction

In this project, we are adding a user management feature to an existing solution. By the end of the project, administrators will be able to add a new user, edit a user, and change user passwords. We will also introduce an encrypted password, so that passwords are no longer stored in clear text.

## Scenario

The scenario is basically the same as Project 3. We are still working on an inventory solution for our resource closet charity. You can go back to Project 3 to read about that if you want to!

The idea here is that we have an inventory solution that already exists to manage \*big\* inventory items, and we are adding user management features to it. The \*big\* inventory items are eerily similar to small inventory items. The difference is, of course, is that it is practical to put a QR code sticker on individual big items. There is no need for the Wawa-style “scan book” here.

What that means is that you need different QR codes. The “small” inventory items aren’t in this solution. I have provided you with the new QR codes you need for testing!

## Getting Started

Get started on Project 4 with the following tasks.  Please be advised that \*every\* student will be expected to set up the database and lambda function here… even if it is not the one your project group ends up using.  This is to help ensure that every student can do the setup work!

Use your personal DB credential (found in the MIS Community Gradebook) and login to the database server.

1. Create a “gold” database.
2. Import this file into your gold database: scan4hopedb.zip (unzip it first!)
3. Set up a lambda function using this start file: project4-lambda-start-file.zip
4. Set up your API gateways as in previous projects

Steps 1 through 4 above are covered this video: <https://youtu.be/Ps6rjq3S2Ds>

When you are done, put the endpoint URL into the box below.

***This URL will be different for every student.***

***Box A***

|  |
| --- |
|  |

## Server-side work

Server-side project buddies need to add the user management features to the existing server-side code. You should be able to do that in a single sitting, lasting roughly an hour or two at the most.

* Here’s a video of me doing the work: <https://youtu.be/nOlmBhtzp28>
* Here’s a video of me doing some tests and making fixes: <https://youtu.be/vAdG1WLPb0I>

## Client-side work

Client-side project buddies need to add:

1. An HTML div tag to show a list of users (see: <https://youtu.be/LsEybffZmGY> )
2. An HTML div tag to allow you to create a user (see: <https://youtu.be/fLQ8TxdB6CY> )
3. Some fixes (see: <https://youtu.be/YPViSvUiW9I> )
4. An HTML div tag to allow you to edit a user
(see both: <https://youtu.be/CJcqqJgBt38> and <https://youtu.be/4P1NNQhrLuU> )

Our add and edit functionality could (ideally, theoretically) be accomplished using the same HTML form. But, to make it easier for students to progress through the material, we achieve the desired results with two separate HTML forms… one for “add” and the other for “edit”.

## OK Team, stop and think

Before you rush ahead to the final part of the project, it would be smart to circle up with your project buddies and make sure everything works. It would be smart to make some backups of your work too.

In the next steps you will have to undo / redo some work to fix a security problem.

If you don’t fix the problem, you can still earn 80% of the project grade if everything else is working ok. You should all decide/agree if you want to move forward, or not. Some students are happy with 80%. Some are only happy with 100%. You need to agree among yourselves if what you have is “good enough”.

## Fixing the security issue

In these last two videos I will correct the security bug. Note that once I start fixing that bug, I won’t help you go back to the old/unsecure solution.

* Make some backups and change the database: <https://youtu.be/CLZvg2DA6vY>
* The server-side work: <https://youtu.be/PFqPGXjp3Bc>
* The client-side work and what it should look like when done: <https://youtu.be/QolLZH20S98>

## Turn in your work!

When you are done, upload this document to the corresponding “Project 4” assignment out on canvas.

Be sure to fill out the boxes below.

***Every student must submit a project document.***

Box B: The URL to your solution (should be on S3, should be the same for the whole team).

|  |
| --- |
|  |

Box C: The URL to your web service API endpoint (from Lambda, should be the same for the whole team):

|  |
| --- |
|  |

Box D: The code to your web service (from the index.mjs file on AWS Lambda, should be the same for the whole team)

|  |
| --- |
| <<COPY PASTE YOUR AWS LAMBDA CODE HERE >> |

## How will this project be graded?

|  |  |
| --- | --- |
| Item | Points |
| Your name / Step 1Your group number / Step 2Your group members / Step 3 | 5 |
| BOX A – Functionality (point deductions in 5-point increments)* Login
* Logout
* List users
* Add a User
* Edit a User
 | 25 |
| BOX B – A11y check comes back clean. | 10 |
| BOX B – HTML is free from errors. | 5 |
| BOX B – Client-side code is clean. For example: unused div tags (like div-AAA) were removed from HTML, and related JavaScript code was removed as well. | 5 |
| BOX C – Web service documentation is complete and accurate. Don’t forget to edit “created by” and “last modified by” | 5 |
| BOX C – Web service code is clean. No undocumented features. Code related to missing / unused features removed. | 5 |
| Did you correct the security issue? Yes or no? All or nothing. There should be no more passwords stored in clear text. Passwords are only sent at the time of authentication, and at the time of account creation. | 20 |
| Your contributions to the project ( 0, 5, 10, 15, 20) | 20  |
| **TOTAL** | **100 points** |

***Please see the syllabus for the late policy.***