Migration plan :

After the admin user logs in or registers , upon authentication , the cookie is set and the session id is set .if the admin is logging in user name is checked by using the session cookie, then a check if performed to confirm if the user is the admin , it it is then the session token is set in the datastore and the cookie is set for the admin to display his contents from the datastore ,the index page is loaded and the old data is loaded from the datastore by a query , the data is called by a get function and the its is rendered on the index .html page for the user .

Scenarios:

- A user desires to change their username
  After login , users can alter username by a special button , this will have implications that need updates in several places on the server and datastore. The session token for the user needs to be updated and the new username must be set for the session token . The datastore entity username will be updated . Care must be taken that the session id is updated synchronously with the datastore updates.

- A user desires to change their password:
  Redirect users to a password reset page. Ask for the old password if it is not already verified .Get a new password , recheck password by asking user to enter password again, hash the password and store the password for that user entity by querying the data store by the username , get the entity , update the password by hashing the current password and update the datastore .Allow user to login with new password and create a new session token .

- A user desires to delete their account and all associated data
  In this case the session tokens associated with his username will be destroyed , the datastore entry for the user , with his credential data removed. All user old data and new data must be discarded from the datastore after the user presses a button on the HTML page .Possibly add a javascript alert to recheck the user's

decision. Session tokens for that user should be destroyed .Query the datastore for the entity of that user , delete the entity from the datastore by it's id .

- A user loses their password
  Provide a button for forgot password which redirects to a page that asks for a security question . Answers to the security question are encrypted and stored in the datastore when the user registers .After the user enters the answer , the answer can be checked against the stored encrypted anwer , after verification it allows the user to reset password . update the password for the user's entity.

- A user has their password stolen and used by someone else
  In these scenarios the first step is to stop the beach and identify the reason behind the breach . methods for eradication of the attack vary depending on the type of attack itself; it can be done by reformatting the affected assets and restoring them, or blacklisting an IP address from where the attack originated. Check for illegal access with other users. One way to check this by looking at sessions' , if there are signs that the user is unaware of or the logs .