Peter M Piper

An interactive and demonstrative program that implements a Hash Table and explores different hashing functions and collision resolution.

Data Structures 2

Hash

**Reason for Test Case \_\_\_\_\_\_\_\_ Input Values\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_Expected Output**

Create initial admin user address: Peter@gmail.com,

password: Japan1 Admin account created

Logging in – user does not exist address: foo@bar.com

password: baz Incorrect username and password

Logging in – admin user –valid password address: Peter@gmail.com,

password: japan1 Welcome back admin!

Logging in – admin user –invalid password address: Peter@gmail.com

password: baz Incorrect username and password

Add new user – non existing

Add new user – existing

Delete existing user – non existing

Delete existing user - existing

Display all existing users in a readable format

\*Change user role – non existing

\*Change user role - existing

Logging in – non-admin user

Change password – old password matches old password: Japan1

new password: Japan2

confirmation: Japan2 Password has been changed

Change password – old password does not match old password: Japan3

new password: Japan2

confirmation: Japan2 Old password incorrect. Password not changed.

Change password – confirmation matches old password: Japan1

new password: Japan2

confirmation: Japan2 Password has been changed

Change password – confirmation does not match old password: Japan1

new password: Japan2

confirmation: Japan2 "New password and confirmation do not match. Password not changed.

\*Forget password – user does not exist

\*Forget password – user exists