**Exercise #3 – Redux-Form + Git**

**Goals:**

1. Utilize redux-form library’s wrapper and validation
2. Be able to commit, push codes to a repository
3. Be able to use all previous lectures

**Instructions**

1. Strictly follow naming convention as specified in this document.
2. Ensure all props will be included in propTypes

**registration-panel.jsx**

1. RegistrationPanel will consist of the following components:
   1. Component1(component1.jsx)
   2. Component2(component2.jsx)
2. Registration Panel should be connected to redux

**component1.jsx**

1. Component1 will have the following fields
   1. First Name (string only)
   2. Last Name (string only)
   3. Age (numbers only and not less than 18 not greater than 80)
   4. userId (can be letters and numbers)
2. Button
   1. Calls a function which will call dispatch to create a user with the details entered
   2. Should be disabled if there are errors present in the form
3. Should import registration-validation.jsx
4. Component1 should be wrapped with reduxForm

**component2.jsx**

1. Component2 will display all the users created
2. Component2 should **not** be connected to redux

**registration-validation.jsx**

1. Should perform the following validations:
   1. First Name
      1. Should accept string only
      2. Error message: “First name should only contain alpha characters”
   2. Last Name
      1. Should accept string only
      2. Error message: “Last name should only contain alpha characters”
   3. Age
      1. Should accept numbers 18-80
      2. Error message: “Age should only contain numeric characters”
   4. User ID
      1. Should only accept alphanumeric characters
      2. Error message: “User ID should only contain alphanumeric characters”

**Git Requirements**

1. Project/exercise should be a git repository (there should be a .git folder in your project or you should have initialized your projects via git)
2. Create a branch with the following pattern: <firstname-lastname-exercise3>
   1. Ex: jeffrey-bulanadi-exercise3
3. Repository will be provided on a later date

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**DEADLINE WILL BE ON MARCH 6 (MONDAY) FAILURE TO SUBMIT THIS EXERCISE WILL RESULT IN ESCALATION TO YOUR RESPECTIVE PEMs**

**CASE TO CASE BASIS APPLIES**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***