**Laboratorio #7**

**“Red Social Dummy”**

**Types:**

Se tendrá un tipo por cada una de las acciones generales que se pueden realizar dentro de la

* POST\_ADDED
* COMMENT\_ADDED
* VOTE\_ADDED

**Actions:**

Se tendrán 3 acciones, una por cada tipo creado.

* addPost: Acción que permita crear un post nuevo con el input que el usuario ingrese.
* addComent: Acción que permita crear un comentario con el input que el usuario ingrese para un post específico.
* addVote: Acción que permita sumarle o restarle a la cantidad de karma (comentarios positivos – comentarios negativos) de un post.

**Reducers:**

Se planea tener 4 reductores divididos en 2 grupos: el de los posts y los de los comments. Estos 4 reductores se unirán para luego formar uno solo.

* postReducers: En este primer grupo, existirán 2 reducers que se encargarán de llevar el control de los posts creados y de ordenarlos de la manera deseada:
  + byId: Reductor que le asignará un valor específico a un post y verificará que contenga todos los aspectos de un post. Se verá algo así:

const byId = (state = {}, action) => {

switch (action.type) {

case types.POST\_ADDED: {

const { id } = action.payload;

const { coment } = state;

const { karma } = state;

}

}

}

* + order: Reductor encargado de ordenar los posts por número de id.
* commentReducers: En este segundo grupo, existirán 2 reducers que se encargarán de llevar el control de los comentarios creados y de ordenarlos de la manera deseada:
  + byId: Reductor que le asignará un valor específico a un comentario y verificará que contenga todos los aspectos del mismo. Se verá algo así:

const byId = (state = {}, action) => {

switch (action.type) {

case types.COMMENT\_ADDED: {

const { id } = action.payload;

const { coment } = state;

}

}

}

* + order: Reductor encargado de ordenar los comentarios por número de id.