ticket.model.js

Share ▾ Open ▾

```
import { action, computed, observable,
import { some } from 'lodash';

import userStore from 'stores/user.sto
import { TABLEAU_STATE as STATE } from

const { AUTHENTICATED, UNAUTHENTICATED

export default class Ticket {
  @observable users = [];

  constructor(config = {}) {
    this.read = config.read;
  }

  fetch() {
    return this.getDataAsync()
      .then((resolved = []) => this.ha
      .catch(error => Promise.reject(e
  }

  async getDataAsync() {
    const response = await this.read()
    return Promise.resolve(response);
  }

  handleChange(resolved = []) {
    if (resolved) this.set(resolved);
    return Promise.resolve();
  }

  @computed get permission() {
    const { users } = toJS(this);
    const { userId: thisUser } = userS
    return (some(users, ['userId', thi
      ? AUTHENTICATED : UNAUTHENTICATE
  }

  @action set(users = []) {
    return this.users.replace(users);
  }
}
```