```java
1
2  package acme.features.manager.assignment;
3
4  import java.util.Collection;
5
6  import org.springframework.beans.factory.annotation.Autowired;
7  import org.springframework.stereotype.Service;
8
9  import acme.client.data.models.Dataset;
10 import acme.client.services.AbstractService;
11 import acme.client.views.SelectChoices;
12 import acme.entities.project.Assignment;
13 import acme.entities.project.UserStory;
14 import acme.features.manager.project.ManagerProjectRepository;
15 import acme.roles.Manager;
16
17 @Service
18 public class ManagerAssignmentShowService extends AbstractService<Manager, Assignment> {
19
20     @Autowired
21     private ManagerProjectRepository repository;
22
23
24     @Override
25     public void authorise() {
26         boolean status;
27         int assigmentId = super.getRequest().getData("id", int.class);
28         Manager manager = this.repository.findManagerProjectByAssignmentId(assigmentId);
29         status = manager != null && super.getRequest().getPrincipal().hasRole(Manager.class)
    && manager.getId() == super.getRequest().getPrincipal().getActiveRoleId();
30
31         super.getResponse().setAuthorised(status);
32
33     }
34
35     @Override
36     public void load() {
37
38         Assignment assigment;
39         int id;
40
41         id = super.getRequest().getData("id", int.class);
42         assigment = this.repository.findAssignmentById(id);
43
44         super.getBuffer().addData(assigment);
45
46     }
47
48     @Override
49     public void unbind(final Assignment assigment) {
50         assert assigment != null;
51         int id = super.getRequest().getPrincipal().getActiveRoleId();
52
53         SelectChoices userStoriesChoices;
54         boolean updateable = this.repository.findProjectOfAnAssignmentByAssignmentId
    (assigment.getId()).isDraft();
55
56         Collection<UserStory> userStories = this.repository.findAllUserStoriesOfAManagerById
    (id);
57
58         Dataset dataset;
59
```

```java
60        userStoriesChoices = SelectChoices.from(userStories, "title", assigment.getUserStory
   ());
61
62        dataset = super.unbind(assigment, "project.title", "userStory");
63
64        dataset.put("masterId", assigment.getProject().getId());
65        dataset.put("userStories", userStoriesChoices);
66        dataset.put("updateable", updateable);
67
68        super.getResponse().addData(dataset);
69
70    }
71
72 }
73
```