```java
1
2  package acme.features.manager.project;
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.entities.project.Project;
12 import acme.roles.Manager;
13
14 @Service
15 public class ManagerProjectListService extends AbstractService<Manager, Project> {
16
17     // Internal state ---------------------------------------------------
18
19     @Autowired
20     private ManagerProjectRepository repository;
21
22     // AbstractService interface ----------------------------------------
23
24
25     @Override
26     public void authorise() {
27         super.getResponse().setAuthorised(super.getRequest().getPrincipal().hasRole
    (Manager.class));
28     }
29
30     @Override
31     public void load() {
32         int id = super.getRequest().getPrincipal().getActiveRoleId();
33         Collection<Project> objects = this.repository.findProjectsOfAManagerById(id);
34
35         super.getBuffer().addData(objects);
36     }
37
38     @Override
39     public void unbind(final Project object) {
40         assert object != null;
41         boolean userStoriesPublishables;
42         boolean isDraft;
43
44         userStoriesPublishables = this.repository.findAllUserStoriesOfAProjectById
    (object.getId()).stream().allMatch(x -> x.isDraft() == false) &&
    this.repository.findAllUserStoriesOfAProjectById(object.getId()).size() > 0
45             && object.isHasFatalErrors() == false;
46         isDraft = object.isDraft() == true;
47
48         Dataset dataset;
49
50         dataset = super.unbind(object, "title", "code");
51         dataset.put("publishable", userStoriesPublishables);
52         dataset.put("isDraft", isDraft);
53
54         super.getResponse().addData(dataset);
55     }
56
57 }
58
```