**A picture containing text, clipart

Description automatically generated**

**Formative Assignment \_\_**

|  |  |
| --- | --- |
| Module Name | Web Development Using Platforms |
| Course Name | Bachelor’s Degree in Software Engineering |
| Assignment Title | Develop an Application |

|  |  |
| --- | --- |
| Learner Name | Francis Roel L. Abarca |

|  |
| --- |
| Learner declaration |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student signature: Date: 3/31/23 |

**Trainer--- University training, SSG training,**

**Content development**

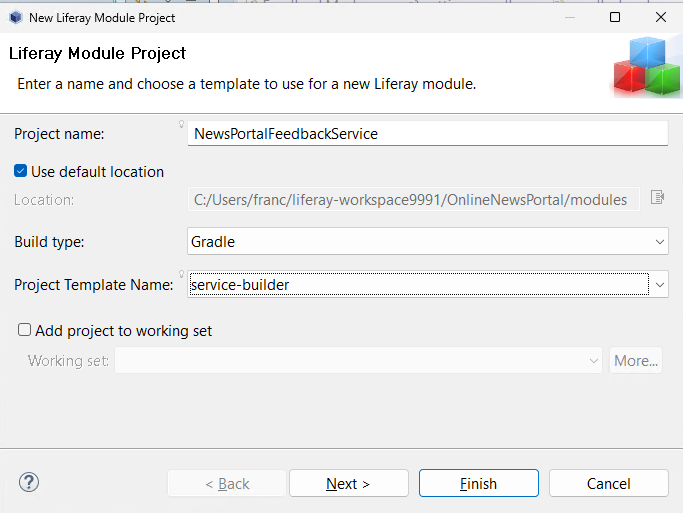
**Scope of this assignment**

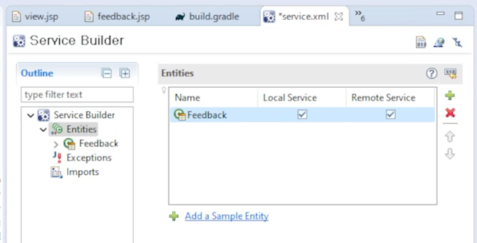
1. To implement a database-driven application, right-click on previous Feedback MVC project -> create a new Liferay service builder with package path "com.liferay.docs.feedback". Use the namespace as FB.
2. Create an entity as a local service in service.xml with entity name Feedback.
3. Define the columns as the following. feedbackID, feedbackDate, feedbackMessage, feedbackSubject, companyID, groupID, userID.
4. In service.xml, define the order of feedback entity by ascending feedbackId.
5. In service.xml, define Service Entity Finder Methods for GroupId, CompanyId and feedbackText cloumns.
6. Build your service. If build success, check the service Builder generates classes and interfaces belonging to the persistence layer, service layer, and model layer.
7. Expose the Finder in the Service Layer \*LocalServiceImpl. Implement a service layer for Feedback entity and create a method "addFeeback" that adds Feedbacks to the database.
8. Update controller (your portlet class) action method to handle the Feedback form.
9. Update view.jsp page to display all the feedback.
10. Build your service application, deploy your portlet, and try adding some Feedback for your portal.

**Task Wise Solutions**

**To implement a database-driven application, right-click on previous Feedback MVC project -> create a new Liferay service builder with package path "com.liferay.docs.feedback". Use the namespace as FB.**

**Graphical user interface, text, application, email

Description automatically generated**

**Create an Entity as a local service in Service.xml with the entity name Feedback.** ****

Graphical user interface, text, application, email

Description automatically generated

**After that, define the colums as the following:**

**Graphical user interface

Description automatically generated**

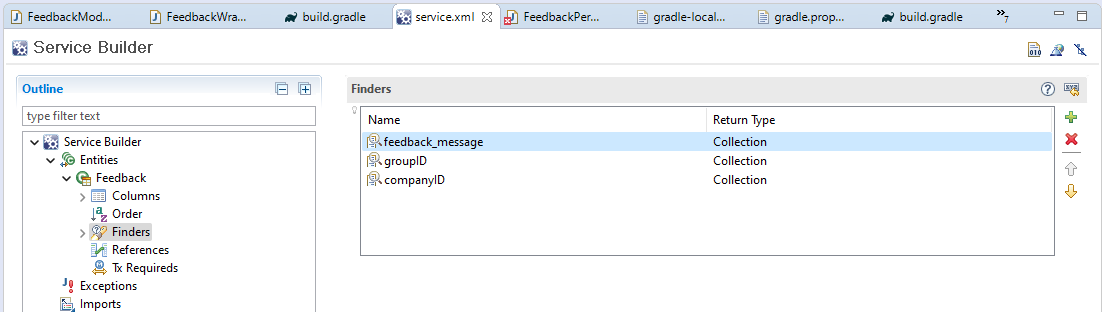
**Define the order of feedbackId by ascending.**

**Graphical user interface, text

Description automatically generated  
Graphical user interface, text, application, email

Description automatically generated**

**Inside service.xml, define the Finder methods for feedback\_message, groupID and companyID.**

****

**Graphical user interface, text

Description automatically generated**

**Build your service then check the service builder and generate classes belonging to the persistence layer, service layer and model layer.**

**Graphical user interface, text, application, Word

Description automatically generated**

**Text

Description automatically generated with medium confidence** **Graphical user interface, text

Description automatically generated with medium confidence**

Graphical user interface, text, application, email

Description automatically generated

**Go to FeedbackPersistenceImpl.java inside the Service layer then add the following method for it to be connected to the database.**

**Text

Description automatically generated with medium confidence**

**Then add these lines of code inside:  
findBygroupID**

@Override

**public** List<Feedback> findBygroupID(**long** groupID) {

**return** findBygroupID(

groupID, QueryUtil.***ALL\_POS***, QueryUtil.***ALL\_POS***, **null**);

}

@Override

**public** List<Feedback> findBygroupID(**long** groupID, **int** start, **int** end) {

**return** findBygroupID(groupID, start, end, **null**);

}

@Override

**public** List<Feedback> findBygroupID(

**long** groupID, **int** start, **int** end,

OrderByComparator<Feedback> orderByComparator) {

**return** findBygroupID(groupID, start, end, orderByComparator, **true**);

}

**findBycompanyID**

@Override

**public** List<Feedback> findBycompanyID(**long** companyID) {

**return** findBycompanyID(

companyID, QueryUtil.***ALL\_POS***, QueryUtil.***ALL\_POS***, **null**);

}

@Override

**public** List<Feedback> findBycompanyID(**long** companyID, **int** start, **int** end) {

**return** findBycompanyID(companyID, start, end, **null**);

}

@Override

**public** List<Feedback> findBycompanyID(

**long** companyID, **int** start, **int** end,

OrderByComparator<Feedback> orderByComparator) {

**return** findBycompanyID(companyID, start, end, orderByComparator, **true**);

**findByfeedback\_message**

@Override

**public** List<Feedback> findByfeedback\_message(String feedback\_message) {

**return** findByfeedback\_message(

feedback\_message, QueryUtil.***ALL\_POS***, QueryUtil.***ALL\_POS***, **null**);

}

@Override

**public** List<Feedback> findByfeedback\_message(

String feedback\_message, **int** start, **int** end) {

**return** findByfeedback\_message(feedback\_message, start, end, **null**);

}

@Override

**public** List<Feedback> findByfeedback\_message(

String feedback\_message, **int** start, **int** end,

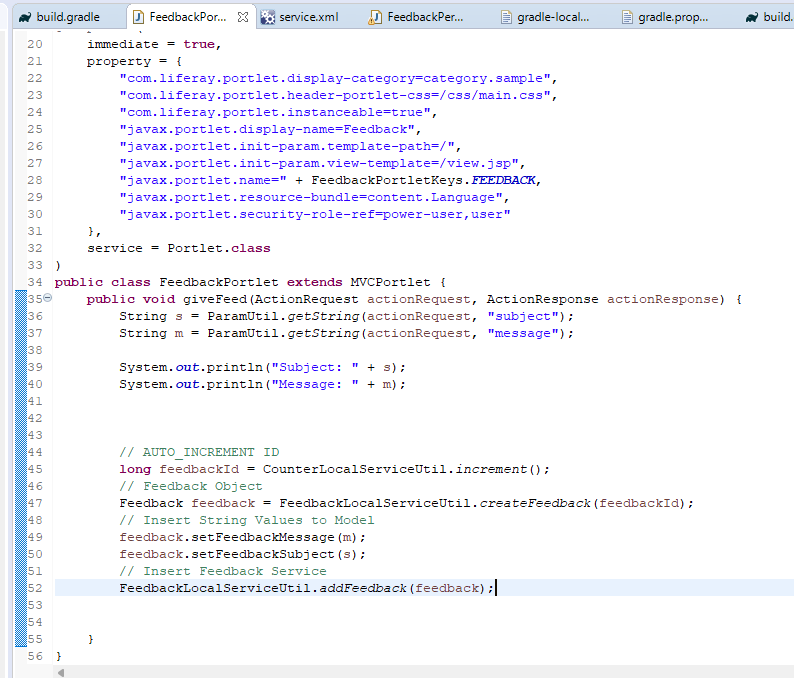
OrderByComparator<Feedback> orderByComparator) {

**return** findByfeedback\_message(

feedback\_message, start, end, orderByComparator, **true**);

}

**Update your controller (Portlet class)’s action method to handle the Feedback form.**

****

**Update view.jsp page to display all the feedback.**

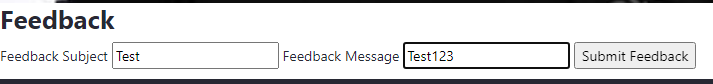
**Graphical user interface, text, application

Description automatically generated**

**Build the service application, deploy the portlet and test the feedback page on your portal.**

**Graphical user interface, website

Description automatically generated**



**A picture containing shape

Description automatically generated**