

COURSE PLAN

Course Code: 18UCSC601		Course Name : Object Oriented Modeling & Design	
Course Teacher : Dr. U.P.Kulkarni		Semester Duration: 17 th March to 4 th June 2022	
Semester: 6	Division: A	Credits :4 / Hours: 52	Type : 4-0-0

Coverage: As specified in the PEM based syllabus and Academic calendar published by the Institution.

Unit	Start Date	End Date	No. of Hours
1	17-3-2022	01-4-2022	10
2	05-4-2022	28-4-2022	12
3	29-4-2022	17-5-2022	10
4	18-5-2022	09-6-2022	10
5	10-6-2022	28-6-2022	10
TOTAL			52

Assessment

Tools :

1. Three assessment tool of **written type** examination (IAs), each of 20 marks.
2. Course teacher's assessment (CTA) is for 10 marks based on **implementation based assignments/ Course project** specified by the course teacher/ PEM based syllabus.
3. Preparatory test (**Open Book**) for each IA will be conducted a day before the IA. It is **mandatory** for all students to attend this to become eligible to appear for corresponding IA.

Class Test	Date and Time
1	22-4-2022
2	31-5-2022
3	28-6-2022

4. Solutions for every test will be discussed in the class immediately following the test.
5. CIE out of 50 is calculated based on Sum of the best two IAs plus CTA.
6. Semester End Examination (SEE) is of written type for 100 marks, reduced to 50.
7. Final grading is based on sum of CIE (50) and reduced SEE (50).
8. Minimum marks to be scored in CIE are 20 to appear for SEE.
9. Minimum passing marks is 40 out of 100 in SEE.

Assignments for CTA:

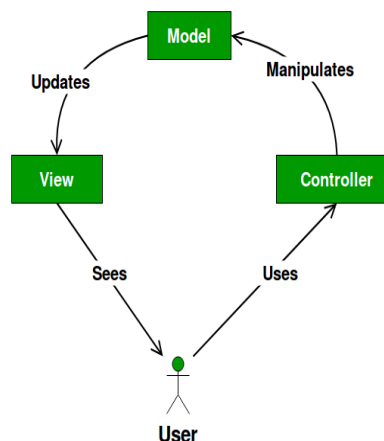
- Following assignments are to be done in a group of two students.
- This group has to collaborate with 4th Semester A division students to mentor their course project.
- Evaluation will be done through seminar and report submitted. Evaluation starts immediately after 2nd IA.

Course work description:

1. Prepare a business scenario and apply **Model View Controller** (MVC) design pattern. Implement the pattern in appropriate language.

The **Model View Controller** (MVC) design pattern specifies that an application consist of a data model, presentation information, and control information. The pattern requires that each of these be separated into different objects.

MVC is more of an architectural pattern, but not for complete application. MVC mostly relates to the UI / interaction layer of an application. You're still going to need business logic layer, maybe some service layer and data access layer.



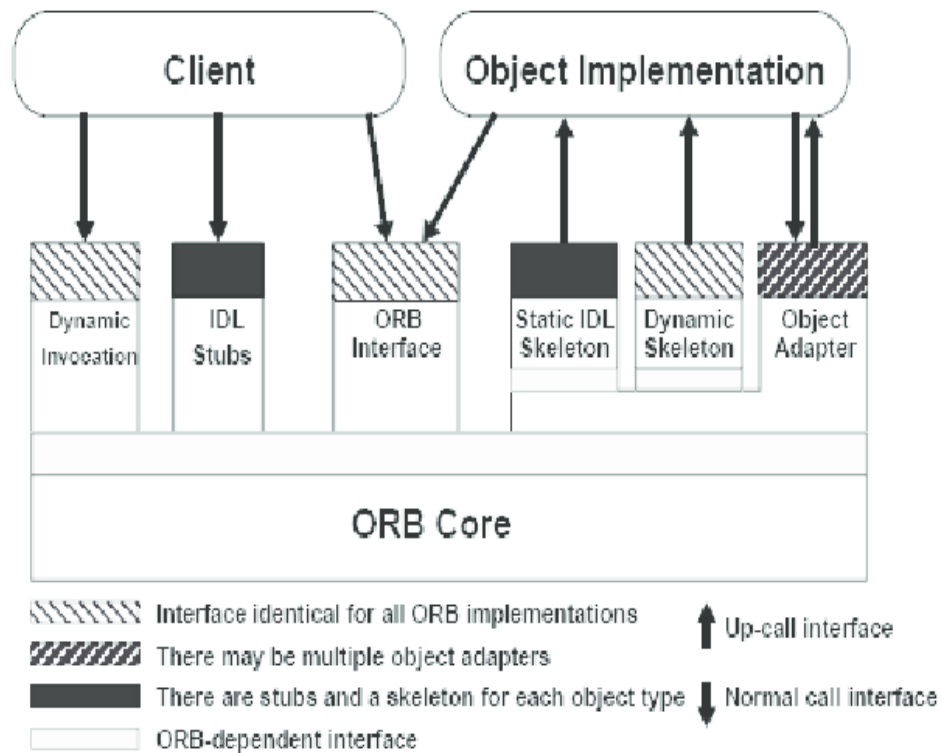
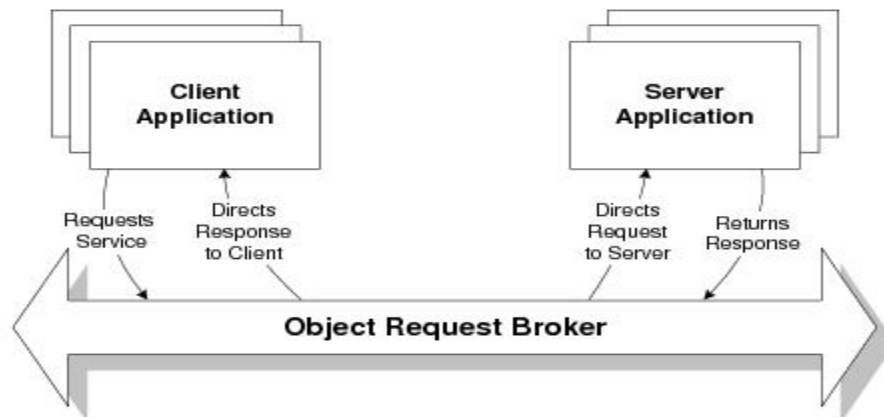
Design components

- The **Model** contains only the pure application data; it contains no logic describing how to present the data to a user.
- The **View** presents the model's data to the user. The view knows how to access the model's data, but it does not know what this data means or what the user can do to manipulate it.
- The **Controller** exists between the view and the model. It listens to events triggered by the view (or another external source) and executes the appropriate reaction to these events. In most cases, the reaction is to call a

method on the model. Since the view and the model are connected through a notification mechanism, the result of this action is then automatically reflected in the view.

2. Prepare the business scenario and write a program to show the **CORBA**. The **Common Object Request Broker Architecture** (CORBA) is a standard defined by the Object Management Group (OMG) that enables software components written in multiple computer languages and running on multiple computers to work together.

Note: As a preparatory work, you are expected to implement **Java RMI**.



3. Presentation of **Reusable objects and writing test scripts** work completed in Software Engineering course of 5th semester.

4. **Mentoring** the course project of Object Oriented Programming of 4th semester A division.

Reference Books:

- 1) Michael Blaha, James Rumbaugh, "Object-Oriented Modeling and Design with UML", 2/E, Pearson Education, 2007.
- 2) Frank Buschmann, RegineMeunier, Hans Rohnert, Peter Sommerlad, Michael Stal, "Pattern-Oriented Software Architecture", A System of Patterns Volume 1, John Wiley and Sons, 2006.
- 3) Len Bass, Paul Clements, Rick Kazman, "Software Architecture in Practice", 2/E, Pearson Education, 2003.
- 4) Grady Boochetai, "Object-Oriented Analysis and Design with Applications", 3/E, Pearson Education, 2007.
- 5) Ali Bahrani, "Object oriented systems development", McGrawHill, 1999.
- 6) Mary Shaw and David Garlan, "Software Architecture Perspectives on an Emerging Discipline", Prentice-Hall of India, 2007