

# Marconi Books Architecture Document

---

## Table of Contents

---

- 1.0 Introduction
- 2.0 High Level Hierarchy
  - 2.1 Hierarchy Diagram
  - 2.2 Hierarchy Description
- 3.0 Components Classification
  - 3.1 Presentation Layer
  - 3.2 Controller Layer
  - 3.3 Record Layer
  - 3.4 Data Access Layer
  - 3.5 Database Layer

## 1.0 Introduction

---

The Marconi Books Architecture Document is designed to show and identify the architecture systems used to design and implement the Marconi Books website. The document contains an overall view of the system hierarchy, logical views of the system components, and a process view of the system's communication.

## 2.0 High Level

---

### 2.1 Hierarchy Diagram

---

### 2.2 Hierarchy Description

---

The architecture system for the Marconi Books application is an n-tier architecture. This architecture system is designed to allow for proper information hiding, modular components, and single system dependencies. The abstraction of the presentation layer, and consequently the User Interface (UI). There are multiple layers between the Presentation Layer and the lowest level, due to the significant challenges present in the optimization and control of the Processes design. The Database layer is the lowest level in the hierarchy.

## 3.0 Components Classification

---

### 3.1 Presentation Layer

---

**Purpose:**

To display forms, controls, images and functions to create a comfortable and efficient user experience.

**Specific Nature:**

The presentation layer will be in charge of displaying appropriate images, menus, and functions to the user. This layer provides an interface to enter data into specific forms. When a user clicks a menu on the GUI, the code corresponding to that event will be called.

**Subcomponents:**

Image Viewer, Insert data

- Image Viewer – The Image Viewer subcomponent is used when a user need to view one or more images of the book that he want to buy, and also whn a user want to add photo about books that he want to sell.
- Insert Data – The Insert Data subcomponent is responsible for the action of adding data about books, then this subcomponent push data to lower layer to enable database to store information about the book.

## 3.2 Controller Layer

---

**Purpose:**

Processes and responds to events, typically user actions, and may invoke changes on the model.

**Specific Nature:**

The controller layer in our program will be in charge of collect information received form the lower layer or from the higher layer. It is responsible near the interpretation of the gesture of the user at the presentation level and it is responsible about the data and record that comes from the bottom.

## 3.3 Record Layer

---

**Purpose:**

This layer is in charge of containing the classes that strictly consist of data.

**Specific Nature:**

This layer will be used to store User data, Landmark data, and Error data and Location data. These classes will only contain properties (variables) that describe each data type.

**Associated Constructs:**

User Record, Landmark Record, Error Record

- **\*\*User Record\*\*** User Record will consist of only properties describing a user. This class will be static, meaning there is only one copy of this class in memory.
  - **\*\*User id\*\*** Numeric type descriptor to identify a user record.
  - **User name** Contains the name of the user, used to access the Marconi Books platform. This will be of type string.
- **\*\*Book Record\*\*** – Book Record will consist of only properties describing a Book.
  - **\*\*Book id\*\*** – Will hold the id of the book, to identify it. This property is of type numeric.
  - **\*\*Book name\*\*** – Will hold the name of the book. This property is of type string.
  - **\*\*Book class\*\*** – Will hold the class of the book. This property is of type string.
  - **Book price** – Will hold the price of the book. This property is of type numeric\_.\_

## 3.4 Data Access

---

### Purpose:

This layer is in charge of communicating to the database. This layer should handle all of the database transactions and connectivity.

### Specific Nature:

This layer will be in charge of communication to our database which will in turn lead to populating the record layer.

## 3.5 Database Layer

---

### Purpose:

This layer is in charge of storing data in persistent storage.

### Specific Nature:

This will be our database management system. There will store Books, Errors, and eventually Language Support.