

# MODEL-DRIVEN DESIGN FOR AN AUTOMATED SYSTEM TO CONNECT PATIENTS AND HOSPITALS BASED ON SERVICE-ORIENTED ARCHITECTURE

Manas Trivedi (181CO231), Omanshu Mahawar (181CO237)  
manasdtrivedi@gmail.com, omanshumahawar1234@gmail.com

National Institute of Technology Karnataka, Surathkal, India

**Abstract.** Model Driven Architecture (MDA) is an approach to design software in the form of UML models. This ensures that the designers of the system are concerned only about meeting the requirements of the system, instead of the platform-specific implementation details. The core idea of MDA is that the code for systems can be generated from models by using model-to-code translators. In the domain of healthcare, software systems must be fast and lightweight, since such systems may be used in time-critical situations. This document focuses on creating a MDA based design for a service-oriented system which connects hospitals and patients, with a focus on users residing in remote areas.

**Keywords:** Model-driven architecture · Healthcare · Design · Architecture

## 1. Introduction

### 1.1 Purpose

This document contains the design and architecture of ‘An Automated System to Connect Patients and Hospitals’ (ASCPH) based on the Model-Driven Architecture (MDA) approach. This document follows the functionalities and requirements identified in the SRS document of the project [2]. The primary audiences of this document are the software developers.

### 1.3 Definitions, acronyms, and abbreviations

**Table 1:** Definitions

Term	Definition
Appointment	Arrangement to meet a doctor working at a hospital at a particular time

Hospital Executive	A person who handles the updation of hospital details and approval and rejection of appointments.
Patient	A person who uses ASCPH to access services such as finding a hospital and booking appointments for the treatment of diseases.
Service	A self-contained unit of software that performs a specific task.

**Table 2.** Acronyms and abbreviations

ASCPH	Automated System to Connect Patients and Hospitals
HEx	Hospital Executive
ID	Identification
MDA	Model-Driven Architecture
CIM	Computation Independent Model
PIM	Platform-Independent Model
PSM	Platform-Specific Model
UML	Unified Modeling Language

## 2 Model-Driven Architecture

The design of systems has evolved from writing high-level documentation and diagrams, to creating complex frameworks. MDA is an approach that addresses this increasing complexity in design. This paradigm shifts the focus from writing code to building models and aims to automate the generation of system implementations for different platforms from the models. The MDA method recommends the following three layers of models:

### 2.1 Computation Independent Model (CIM)

A CIM is a business domain model which presents what the system is intended to do, but does not express the platform-specific details, in order to keep the model independent of the method in which the system will be implemented. CIMs are expressed in natural language. CIMs may be formulated for each view of a system. The CIM aids in bridging

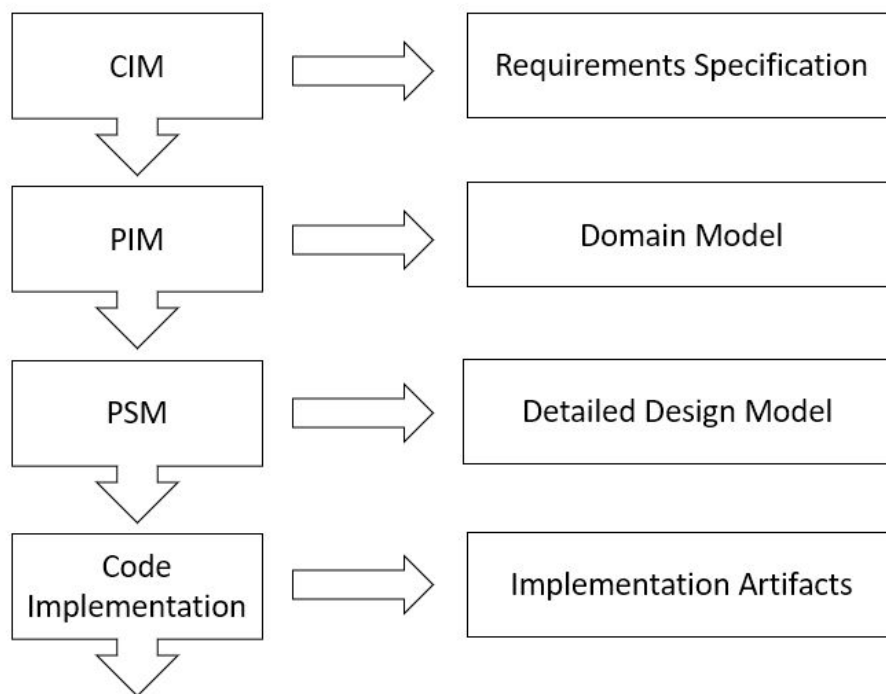
the gap between the persons responsible for the domain model, such as subject matter experts, and those involved in creating the implementation of the system.

## 2.2 Platform Independent Model (PIM)

A PIM represents the working of the system without implementation-specific details. PIMs are expressed in a manner which ensures that they can be mapped to different platforms. They are written in the form of UML diagrams, and show the structure of the system along with the behaviour of the system towards different events.

## 2.3 Platform Specific Model (PSM)

A PSM is created after applying a transformation on the PIM, which adds the details for the implementation of the system on a specific platform.



MDA enables the designers of the system to construct the same without concerns for platform-specific implementation, at a high level of abstraction. This enables the rapid creation of system models and reduces the chances of errors which may arise in a platform-specific implementation.

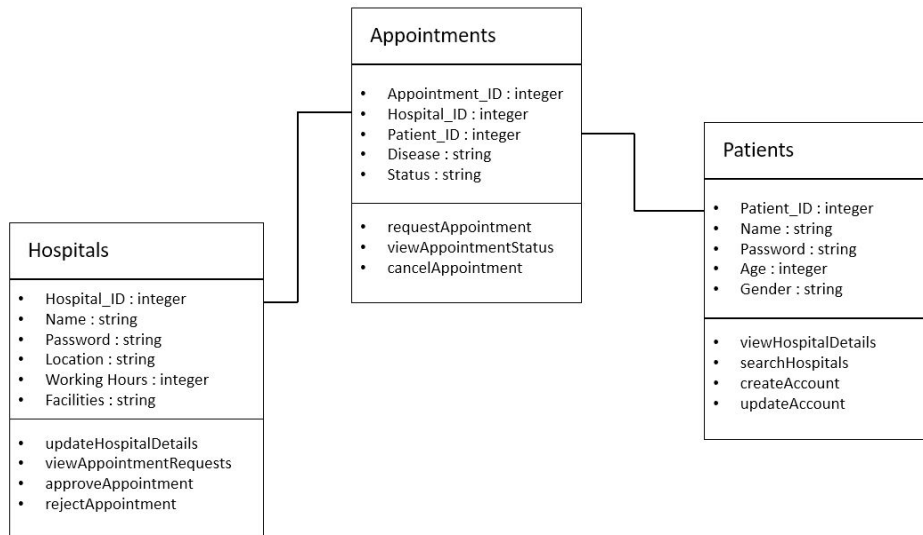
These system models need to be transformed into actual implementation through a series of transformations. Each transformation applied to a model adds more details. These transformations are carried out with powerful tools, which result in system implementations for the same model on different platforms [1].

According to the MDA approach, the transformations between models must happen in a completely automated manner. This would mean that executable code would be generated automatically from a PSM for a given application. But practically, for nontrivial systems, the transformations from PSMs to executable code cannot occur without human intervention. On the other hand, PIM to PSM transformation is relatively simple, and tools exist for the same, which translate PIMs to common platforms. Hence, a single PIM is required for a system to be run on multiple platforms, and a translator is required for each platform for which the PIM to PSM conversion is required.

### 3 Platform Independent Model

The PIM for ASCPH is formulated by modeling the business domain objects using UML. The relationships between these objects are represented in UML class diagrams. This ensures that the requirements of ASCPH are expressed as independent models without any concerns of platform-specific technicalities. The PIM will be transformed into a Java PSM, which would be performed using a model-to-model translator.

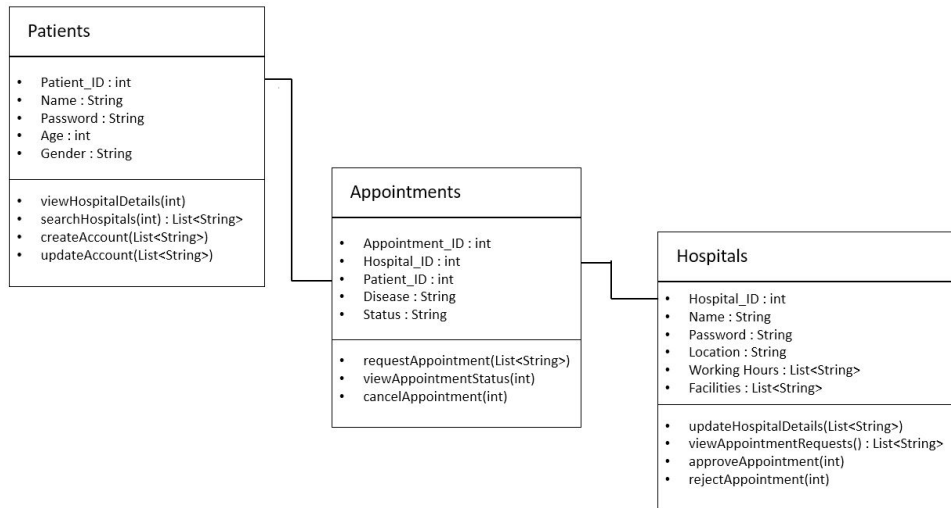
A component of the PIM, a UML Class Diagram, is shown in figure 2.



**Figure 2.** PIM for ASCPH

## 4 Platform Specific Model

The PSM for ASCPH, generated from the PIM, is created for the Java platform. The PSM generated contained issues which were resolved by making changes to the PSM manually. The Java PSM is shown in figure 3.



**Figure 3.** Java PSM for ASCPH

## Conclusion

An MDA-based design for ASCPH has been described in this document. The PIM for the application consists of the entities along with their attributes and associated methods. The PSM for the application is derived from the PIM, and contains the variables and functions for the ASCPH entities, with Java as the platform.

## References

1. Sommerville, I.: Software Engineering, 10th ed., pp. 490-519. Pearson India (2017)
2. Trivedi, M., Mahawar, O.: Software Requirements Specification for An Automated System to Connect Patients and Hospitals Based on Service-Oriented Architecture (2020)