





**FTMSManager** – We are using the singleton design pattern. The FTMSManager has a static instance that manages all the items it points to.

**FTMSApplication** – This class simply serves as the entry point to the application. It calls upon the PersistenceFTMS subsystem to load persistence data, and starts the UI.

**FTMSPage** – This class is the desktop view. It calls the Java Swing & AWT APIs in order to display appropriate content retrieved via the controllers. It also fires events which call controller methods.

**OrderController** – This is a controller class containing all the methods necessary for handling an order. Currently it is called statically from events fired by the View.

**ScheduleMaker** – This is a controller class containing all methods for interacting with staff schedules. It contains all the necessary functions and utilities for schedule creation and display. Called from the View.