

Getting Started

Table of contents

1 Build the code.....	2
2 Audience.....	2
3 Scope of this document.....	2
4 Java, JavaScript and PatientOS tools.....	2
5 Threads.....	3
6 Design Patterns.....	3



1. Build the code

The developers guide has chapters on building the source code, configuring and deploying the application. Technical overviews of code generation (coming soon) and how to build new applications is briefly described.

Software Developers Guide

[PDF format](#)

[Word format](#)

2. Audience

This document is primarily written for developers who want to be able to work on any part of PatientOS. You should have a good understanding of Object Oriented Programming, and Java as it is the main programming language used in the project.

However, if you intend to work with the tools and perhaps JavaScript, this document should still be helpful.

3. Scope of this document

This document will try to answer the following questions:

- How is the source code organized?
- Which technologies are used?
- Where do I start?
- How do the components work together?
- How do I contribute code?

4. Java, JavaScript and PatientOS tools

Java is used throughout PatientOS, from the server to the client. The client is Swing though you will find only a moderate amount of Swing specific code in the source as the client is dynamically generated at runtime based upon database settings.

These database settings define the complex hierarchy of forms, menus, toolbars, controls, buttons and everything else that makes up an application. It would very time consuming to enter these settings manually into the database and so PatientOS actually comes with prebuilt tools to create applications, dialogs, forms, tables and much more.

Throughout the UI JavaScript can be used to create dynamic behaviour and workflows while the application is running.

5. Threads

PatientOS is a multithreaded application. You should understand what that means and how both Java and Swing typically handles this complexity. For the most part the framework will insulate you from having to deal with threads but it is helpful to know before working directly on any of the common Swing components

6. Design Patterns

The PatientOS framework and application design is very heavy on the Facade, Mediator, and a few other patterns.