Vaadin Objects 0.1: Developers' Guide

Vaadin viewer

Version 0.1

Copyright © 2009 Dan Haywood

Permission is granted to make and distribute verbatim copies of this manual provided that the copyright notice and this permission notice are preserved on all copies.

Preface ..... **v**
1. Building from Source **1**
1.1. Prerequisites 1
1.2. Source Code Repository 1
2. Release (Tagging) Process **3**
3. Design **5**
3.1. TODO 5

Preface

Vaadin Objects is a sister project for the Naked Objects framework, allowing JPA-annotated domain objects to be persisted to a relational database (with Hibernate as the underlying JPA implementation).

This developers' guide explains how to build *JPA Objects* from source, allowing you to contribute back and extend the range of capabilities. If you are simply interested in using *JPA Objects* as-is, please consult the user guide.

Vaadin Objects is hosted on SourceForge, and is licensed under Apache Software License v2. *Naked Objects* is also hosted on SourceForge, and is also licensed under Apache Software License v2.

Chapter 1

Building from Source

This chapter describes the (Java) software needed to build the Vaadin Objects project. Follow the steps in this chapter if you want to build the project from source, or want to contribute back to the project.

1.1. Prerequisites

The *Star Objects* developers' guide describes the common set up for all sister projects, including this one. Specifically it describes how to set up your development environment based on whether you are just building from source or whether you are additionally contributing back to the project. It also describes the corporate POM, used as a common parent by all sister projects.

Review the *Star Objects* document and then set up your development environment accordingly (part one of that document). You should end up with Java, Maven, Subversion and Eclipse, all configured appropriately.

1.2. Source Code Repository

The source code directory structure for *Vaadin Objects* is as follows:

```
trunk/
  main/                 # main release
    ...
    documentation/        # this documentation
  support/
  testapp/            # application for testing - not released
tags/
  main                # tags for trunk/main
  ...
```

Chapter 2

Release (Tagging) Process

This chapter describes how to perform a release, prior to deployment. It will only be of interest to contributors with admin privileges to release code.

The release process tags the codebase under `trunk` into `tags`, updating the versions in all POMs in the process removing the "-SNAPSHOT" suffix. After the release is finished, it may then be deployed (see ???). Note that snapshots do not go through this process and can be deployed at any time.

TODO: to complete; adapt from other sister projects when done.

Chapter 3

Design

3.1. TODO

TODO: describe the internal design/architecture here...