

Tobar Segais: User Manual

Stephen Connolly

Tobar Segais: User Manual

by Stephen Connolly

Abstract

User Manual for the Tobair Segais Web application

Table of Contents

- Copyright v
- 1. Introduction 1
- 2. Building 2
- 3. Customization 4
 - Introduction 4
 - Stylesheet customization 4
 - Navbar customization 4
 - Icon customization 5
- Index 6

List of Figures

1.1. Screenshot of the Tobar Segais web application 1

Copyright

Copyright © 2012 Stephen Connolly

‘Oracle’ and ‘Java’ are registered trademarks of Oracle and/or its affiliates.

‘Apache’, ‘Apache Ant’, ‘Apache Lucene’, ‘Apache Maven’, ‘Ant’, ‘Lucene’ and ‘Maven’ are trademarks of The Apache Software Foundation. Used with permission. No endorsement by The Apache Software Foundation is implied by the use of these marks.

‘Eclipse’ is a trademark of the Eclipse Foundation, Inc. No endorsement by the Eclipse Foundation is implied by the use of these marks.

Other names may be trademarks of their respective owners. Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the author(s) were aware of a trademark claim, the designations have been printed in caps or initial caps.

While every precaution has been taken in the preparation of this book, the publisher and authors assume no responsibility for errors or omissions, or for damages resulting from the use of the information contained herein.

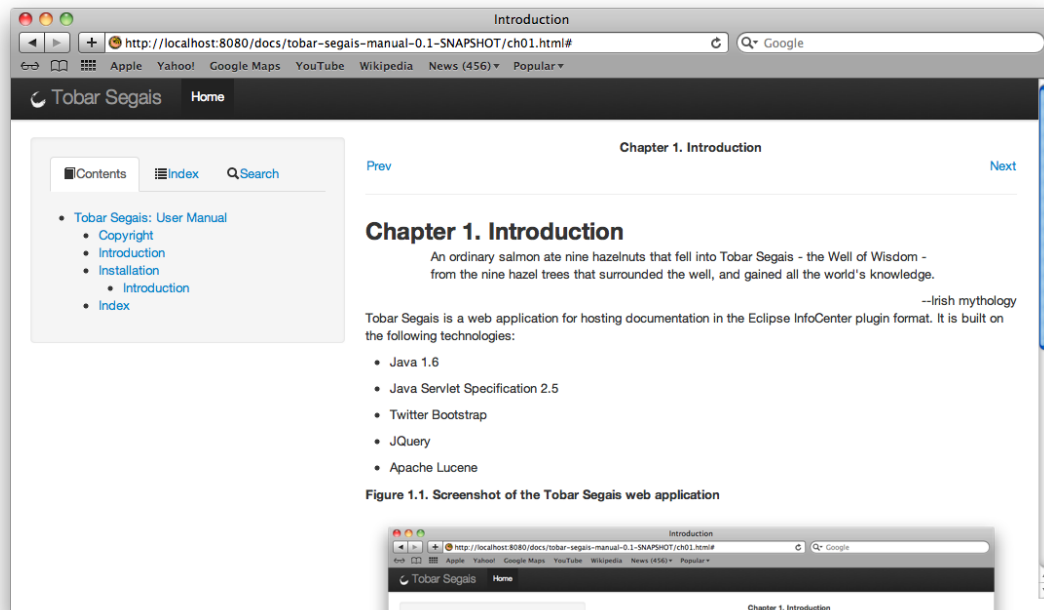
Chapter 1. Introduction

An ordinary salmon ate nine hazelnuts that fell into Tobar Segais - the Well of Wisdom - from the nine hazel trees that surrounded the well, and gained all the world's knowledge.
—Irish mythology

Tobar Segais is a web application for hosting documentation in the Eclipse InfoCenter plugin format. It is built on the following technologies:

- Java 1.6
- Java Servlet Specification 2.5
- Twitter Bootstrap
- JQuery
- Apache Lucene

Figure 1.1. Screenshot of the Tobar Segais web application



Chapter 2. Building

By default the Tobar Segais web application does not bundle any documentation, so in order to have a useful web application you must build a custom version that includes the desired InfoCenter plugin bundles.

The easiest way to build the Tobar Segais web application is using Apache Maven's WAR Overlays feature. Start by creating a basic `pom.xml` that depends on the Tobar Segais Web application:

```
<?xml version="1.0" encoding="utf-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.mydomain.reversed</groupId>
  <artifactId>docs-webapp</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>war</packaging>

  <dependencies>
    <dependency>
      <groupId>org.tobarsegais</groupId>
      <artifactId>tobar-segais-webapp</artifactId>
      <version>1.5.1</version>
      <type>war</type>
    </dependency>
    <dependency>
      <groupId>com.mydomain.reversed</groupId>
      <artifactId>docs-infocenter-plugin</artifactId>
      <version>1.0-SNAPSHOT</version>
      <scope>provided</scope>
    </dependency>
  </dependencies>

  <build>
    <plugins>
      <plugin>
        <artifactId>maven-dependency-plugin</artifactId>
        <executions>
          <execution>
            <phase>generate-resources</phase>
            <goals>
              <goal>copy-dependencies</goal>
            </goals>
            <configuration>
              <outputDirectory>
                ${project.build.directory}/${project.build.finalName}/WEB-INF/bundles
              </outputDirectory>
              <includeScope>provided</includeScope>
            </configuration>
          </execution>
        </executions>
      </plugin>
    </plugins>
  </build>
</project>
```

```
        </executions>
    </plugin>
</plugins>
</build>

</project>
```

To build the web application, just invoke Apache Maven with a phase of “package” or later, e.g.

mvn package

The generated `.war` file can then be deployed in any Servlet container that supports the Servlet 2.5 specification.

Chapter 3. Customization

Introduction

Not everyone will be happy with everything about how the Tobar Segais web application looks. However, you will be happy to know that the Tobar Segais web application supports a number of customization points to enable you to get it looking just the way you want.

Stylesheet customization

The core look of Tobar Segais is based on Twitter Bootstrap. Any custom full build of Bootstrap 2.0.x should work as a drop in replacement that offers the customized look and feel.

Assuming you are building your Tobar Segais web application using the technique described in Chapter 2 all you need do is create the file `src/main/webapp/css/bootstrap.min.css` containing the minified version of your custom build of Twitter Bootstrap.

If you have customized versions of the Glyph Icons, these should be put at the paths `src/main/webapp/img/glyphicons-halflings.png` and `src/main/webapp/img/glyphicons-halflings-white.png`

If you just want to apply some simple tweaks to the CSS after Twitter Bootstrap is loaded all you need to do is create the file `src/main/webapp/css/custom.css` this file is included after all the other CSS files have been included.

Navbar customization

The Navbar at the top of the page can be easily customized by providing a replacement JSP Fragment.

Assuming you are building your Tobar Segais web application using the technique described in Chapter 2 all you need do is create the file `src/main/webapp/WEB-INF/custom-navbar.jspf` containing the custom navbar. For example, the default Navbar has the following `custom-navbar.jspf`:

```
<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<div class="navbar navbar-fixed-top no-print">
  <div class="navbar-inner">
    <div class="container-fluid">
      <a class="btn btn-navbar" data-toggle="collapse" data-target=".nav-collapse">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </a>
      <a class="brand" href="http://www.tobarsegais.org/">
        
        Tobar Segais
      </a>
      <div class="nav-collapse">
        <ul class="nav">
          <li class="active"><a href="{pageContext.request.contextPath}/docs/">Home</a></li>
        </ul>
      </div>
    </div>
  </div>
</div>
```

```
</div>  
</div>  
</div>
```

Icon customization

There are a number of icons which can be replaced quite simply.

The following instructions assume you are building your Tobar Segais web application using the technique described in Chapter 2.

- *Header logo* - `src/main/webapp/img/header-logo.png`
- *Favicon* - `src/main/webapp/img/favicon.ico`
- *Apple Touch Icons* - `src/main/webapp/img/apple-touch-icon.png`, `src/main/webapp/img/apple-touch-icon-72x72.png`, `src/main/webapp/img/apple-touch-icon-114x114.png` and `src/main/webapp/img/apple-touch-icon-144x144.png`,

Index

S

Screenshot, 1