

Uses swagger to document Grails Controllers

Swaggydoc Plugin - Reference Documentation

Authors: Rahul Somasunderam

Version: 0.20

Table of Contents

- 1 Introduction
- 2 Customization

1 Introduction

Swaggydoc makes it easy to document rails controllers using swagger.

This guide documents configuration defaults and describes how to configure the Swaggydoc plugin for Grails applications.

Release History

- 2015-05-01 - 0.20
 - Added support for items in Operation
 - Docs on api url and customization
- 2015-05-01 - 0.19
 - Readded support for rails 2.3
- 2015-05-01 - 0.18
 - Fixed Infinite loops in self referencing models
- 2015-04-24 - 0.17
 - Support for multiple values
 - No responseModel when responseModel is 'void'. This improves codegen.
 - integer params are now properly documented.
 - Codegen now works for test data
- 2015-04-23 - 0.16
 - Added support for overriding produces and consumes per method
 - Added support for documenting command objects
 - Added support for enums as dataTypes in ApiOperation
 - Added support for responseContainers
 - Added support for response classes for ApiResponse
- 2015-04-20 - 0.15
 - Fully documented supported annotations
 - Support for extraParams
 - Support for enumerated operation parameters and min/max constraints.
- 2015-04-18 - 0.14

- Overridable path params
- Removed blank methods
- Most primitives and wrappers are now accounted for
- Added support for enums
- Significantly enhanced type system
- 2015-04-15 - 0.13
 - Removed boring methods
 - Lots of static compilation
 - Added toString to parameter. Also removed ref to 'demo'
 - Log and proceed when type information is not available.
 - Upgraded to 2.4.5. Reloading fixed.
 - added a method to enrich parameter information with constraints information (felipe)
 - Assert Grails Domain definitions (marcos-carceles)
 - removed unnecessary configuration files and updated .gitignore (felipe)
- 2015-04-04 - 0.12
 - Support for multiple http methods on a method
 - Newified internal classes
 - More static types
- 2015-03-13 - 0.11
 - Declared "marshallers" dependency as test-only, so it doesn't propagate (David Pisoni)
- 2015-02-26 - 0.10

- Improved documentation of hierarchical path parameters (David Pisoni)
- Added some sense to the response content-type filter (David Pisoni)
- Fixed minor bugs (David Pisoni)
- Added optional support for marshallers plugin for emitting domain-based models (David Pisoni)
- Models based on Domains now use persistentProperties to determine fields (David Pisoni)
- Fixed broken string in responseMessages (David Pisoni)
- Cleaned up default response messages (David Pisoni)
- "produces" content types are now dynamically determined by Controller.responseTypes (David Pisoni)
- Fixed bug where single-resource url mappings were getting "id" parameter (David Pisoni)
- Fixed support for controller/action UrlMappings (David Pisoni)
- Added consistent support for hierarchical resource path parameters when annotations are used (David Pisoni)
- Consolidated workaround to rails 2.4.4 rendering bug into Controller (David Pisoni)
- Use UrlMappings to dynamically document actions (first pass) (David Pisoni)
- Refactored service-y stuff into a service (David Pisoni)
- 2015-02-12 - 0.9
 - Redirect for ease of development
 - Added support for Non domain classes
- 2015-02-04 - 0.8
 - Workaround terrible bug in rails 2.4.4 that fails to render lists to JSON (David Pisoni)
- 2015-01-07 - 0.7
 - Updated swagger-ui
 - Fix for 'Code breaks when the domain class has explicit type defined for hasMany attribute'
- 2014-12-25 - 0.6
 - Grails Dependency reduced to 2.2
 - Now using Travis's docker infrastructure
- 2014-10-14 - 0.5
 - Added support for required fields based on nullable
 - Travis can now automatically publish to rails plugin portal
- 2014-09-24 - 0.4
 - Added support for domain relationships
 - Enabling travis

- 2014-09-22 - 0.3
 - No dupes for parametrized methods. ImplicitParams are optional.
- 2014-09-22 - 0.2
 - No domain? no problem. Also q is no longer forced
- 2014-09-22 - 0.1
 - Added documentation in README
 - 1.7 is clearly defined as jdk
 - Added shorthand annotations for standard methods
 - Added support for summarized annotations - List and Show
 - Better Markup
 - Working API Renderer
 - Implemented documentation

2 Customization

By default, the api page is rendered at the uri `/api`. If you prefer changing that to something else, please keep in mind that the api page is rendered by a controller. You can do something like this:

```
class UrlMappings {
  static mappings = {
    "/*controller/*action?/*id?" {
      constraints {
        // apply constraints here
      }
    }

    "/myapi" (controller: "api")
    "/" (view: "/index")
    "500" (view: '/error')
  }
}
```

This now moves the api page to the uri `/myapi`.