



**Hunters Inn Pool**  
**v1.0.0-SNAPSHOT**  
*Hunters Inn Pool Application*  
*2015-07-07*



# Table of contents

1. Hunters Inn Pool	Page 1
1.1. Report Overview	
1.2. Issues Analysis	
1.3. Issues Details	

---

# 1. Hunters Inn Pool

This chapter presents an overview of the project measures. This dashboard shows the most important measures related to project quality, and it provides a good starting point for identifying problems in source code.

## 1.1. Report Overview

### Static Analysis

#### Lines of code

**6,065** ↓

N/A packages

77 classes

545 methods

2.7% duplicated lines

#### Comments

**31.2%**

2,755 comment lines

#### Complexity

**2.3**

16.4 /class

1,260 decision points

### Dynamic Analysis

#### Code Coverage

N/A

N/A tests

#### Test Success

N/A

N/A failures

N/A errors

### Coding Rules Issues

#### Technical Debt

**1d** ↑

#### Issues

**30** ↑

## 1.2. Issues Analysis

Most violated rules	
Generic wildcard types should not be used in return parameters	1
"switch" statements should have at least 3 "case" clauses	3
Source files should not have any duplicated blocks	13
Methods should not be too complex	11
Control flow statements "if", "for", "while", "switch" and "try" should not be nested too deeply	2

<b>Most violated files</b>	
DataResource.java	5
FixtureSet.java	3
CompetitionPanel.java	3
Player.java	2
Competition.java	2

<b>Most complex files</b>	
TableUtils.java	87
DataResource.java	76
ReportBuilder.java	67
Fixture.java	64
Competition.java	58

<b>Most duplicated files</b>	
FixtureSetTable.java	43
OutstandingTable.java	43
TableUtils.java	39
OutlookTable.java	32
ParticipantsTable.java	32

### 1.3. Issues Details

Rule	Generic wildcard types should not be used in return parameters	
File	Line	
TableUtils.java	80	

Rule	"switch" statements should have at least 3 "case" clauses	
File	Line	
CompetitionPanel.java	583	
CompetitionTable.java	183	
PlayerTable.java	218	

Rule	Source files should not have any duplicated blocks	
------	--	--

File	Line
ProgramResource.java	N/A
FixtureSetTable.java	N/A
PoolResource.java	N/A
Competition.java	N/A
DataResource.java	N/A
OutlookTable.java	N/A
OutstandingTable.java	N/A
ParticipantsTable.java	N/A
TableUtils.java	N/A
Player.java	N/A
GuiResource.java	N/A
CompetitionPanel.java	N/A
MaintenancePanel.java	N/A

Rule	Methods should not be too complex	
File	Line	
Guilcons.java	155	
CompetitionPanel.java	510	
Player.java	246	
Competition.java	333	
DataResource.java	349	
GuiResource.java	300	
DataResource.java	257, 301	
Fixture.java	336	
FixtureSet.java	286	
DataResource.java	325	

Rule	Control flow statements "if", "for", "while", "switch" and "try" should not be nested too deeply	
File	Line	
FixtureSet.java	365, 351	