ErrorReporting SPI for Jira
Table of Contents

API & Implementation ................................................................. 2
How to configure/use ................................................................. 3
  Classpath ................................................................................. 3
  Configuration Properties .......................................................... 3
  Bootstrapping .......................................................................... 3
Known issues .................................................................................. 4
Dependencies ................................................................................ 5
This module (incodemodule-errorrptjira-impl) provides an implementation of Apache Isis' ErrorReportingService SPI. Whenever an error page is displayed to the user, the implementation will automatically log the details in JIRA and provide the ticket reference to the end-user.
### API & Implementation

This module provides `ErrorReportingServiceForJira` domain service as an implementation of the `ErrorReportingService`:

```java
public interface ErrorReportingService {
    Ticket reportError(final ErrorDetails errorDetails);
}
```

where the returned `Ticket` determines what is shown on the error page for the end-user. In the case of this implementation:

```java
public class Ticket implements Serializable {
    private final String reference;  // ①
    private final String userMessage;  // ②
    private final String details;  // ③
    private final StackTracePolicy stackTracePolicy;  // ④
    private final String kittenUrl;  // ⑤
    ...
}
```

① the JIRA issue number, eg "SUP-1234"

② the main error message to display to the user. This defaults to:

"Our apologies, an error has occurred. Support has been notified."

but can be configured to something else if required (see below).

③ optionally, additional details (shown in a smaller font). This defaults to:

"For more details, see JIRA issue: %s"

(with the place holder holding the URL for the generated JIRA issue), but can be configured to something else if required (see below).

④ whether the end user should be able to view the stack trace. This implementation ALWAYS HIDES the stack trace.

⑤ the URL of an image to display, if any. This implementation never shows any image.
How to configure/use

Classpath

Update your classpath by adding this dependency in your dom project's pom.xml:

```xml
<dependency>
  <groupId>org.incode.module.errorrptjira</groupId>
  <artifactId>incode-module-errorrptjira-impl</artifactId>
</dependency>
```

Check for later releases by searching Maven Central Repo.

Configuration Properties

The following configuration properties are required (either in isis.properties or through the AppManifest); for example:

```properties
isis.service.errorReporting.jira.username=XXXXXX
isis.service.errorReporting.jira.password=XXXXXX
isis.service.errorReporting.jira.base=https://mycompany.atlassian.net/
isis.service.errorReporting.jira.projectKey=PRODSUP
isis.service.errorReporting.jira.issueType="IT Help"
```

Optionally the message text and details can be changed:

```properties
isis.service.errorReporting.jira.userMessage=...
isis.service.errorReporting.jira.detailsFormat=...
```

There currently is no mechanism to localize the message.

Bootstrapping

In the AppManifest, update its getDependencies() method, eg:

```java
@Override
public Set<Module> getDependencies() {
    return Sets.newHashSet(
        ...
        new org.incode.module.errorrptjira.ErrorReportingJiraModule(),
        ...
    );
}
```
Known issues

None known at this time.
Dependencies

Maven can report modules dependencies using:

```
mvn dependency:list -o -pl modules/spi/errorrptjira/impl -D excludeTransitive=true
```

which, excluding Apache Isis itself, returns these compile/runtime dependencies:

```
com.fasterxml.jackson.core:jackson-databind:jar:2.8.0
```