Redmine - Patch #35208

Use `Time.use_zone` instead of `Time.zone=`

2021-05-05 10:04 - Go MAEDA

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:	Go MAEDA		**% Done:**	0%
Category:	Code cleanup/refactoring		**Estimated time:**	0.00 hour
Target version:	5.0.0			

Description

When temporarily overriding a time zone, it is safer to use Time.use_zone than to update Time.zone directly because Time.use_zone will reset the time zone to its original value after the execution of the given block is done.

And RuboCop Rails 2.10.0 will display Rails/TimeZoneAssignment warning when it detects code that directly updates Time.zone.

```
Index: test/unit/mailer_test.rb
===================================================================
--- test/unit/mailer_test.rb    (リビジョン 20961)
+++ test/unit/mailer_test.rb    (作業コピー)
@@ -386,11 +386,10 @@
    issue = Issue.find(3)
    user = User.find(1)
    %w(UTC Paris Tokyo).each do |zone|
-      Time.zone = zone
-      assert_match /^redmine\.issue-3\.20060719190727\.1@example\.net/, Mailer.token_for(issue, user)
+      Time.use_zone(zone) do
+        assert_match /^redmine\.issue-3\.20060719190727\.1@example\.net/, Mailer.token_for(issue, user)
+      end
    end
-  ensure
-    Time.zone = zone_was
  end

  test "#issue_add should notify project members" do
```

Related issues:

Related to Redmine - Patch #35207: Update RuboCop Rails to 2.14	**Closed**

Associated revisions

Revision 20988 - 2021-05-07 06:24 - Go MAEDA

Use `Time.use_zone` instead of `Time.zone=` (#35208).

Patch by Go MAEDA.

History

#1 - 2021-05-05 10:05 - Go MAEDA

- Related to Patch #35207: Update RuboCop Rails to 2.14 added

#2 - 2021-05-07 06:24 - Go MAEDA

- Status changed from New to Closed

- Assignee set to Go MAEDA

- Target version set to 5.0.0

Committed the patch.