Datum	Version	Beschreibung	Autor
3.11.2013	1.0	Erstellen des Use-Cases	Lorenzo Toso Philipp Nitsche Irtaza Syed Sebastian Hüther
08.05.2014	1.1	Hinzufügen des Testscenarios	Lorenzo Toso
12.06.2014	2.0	Update für Endabgabe. Aktualisierung der Screenshots und Texte	Lorenzo Toso

Use-case Specification: Offline-Kalender Aktualisieren

1 Offline-Kalender Aktualisieren

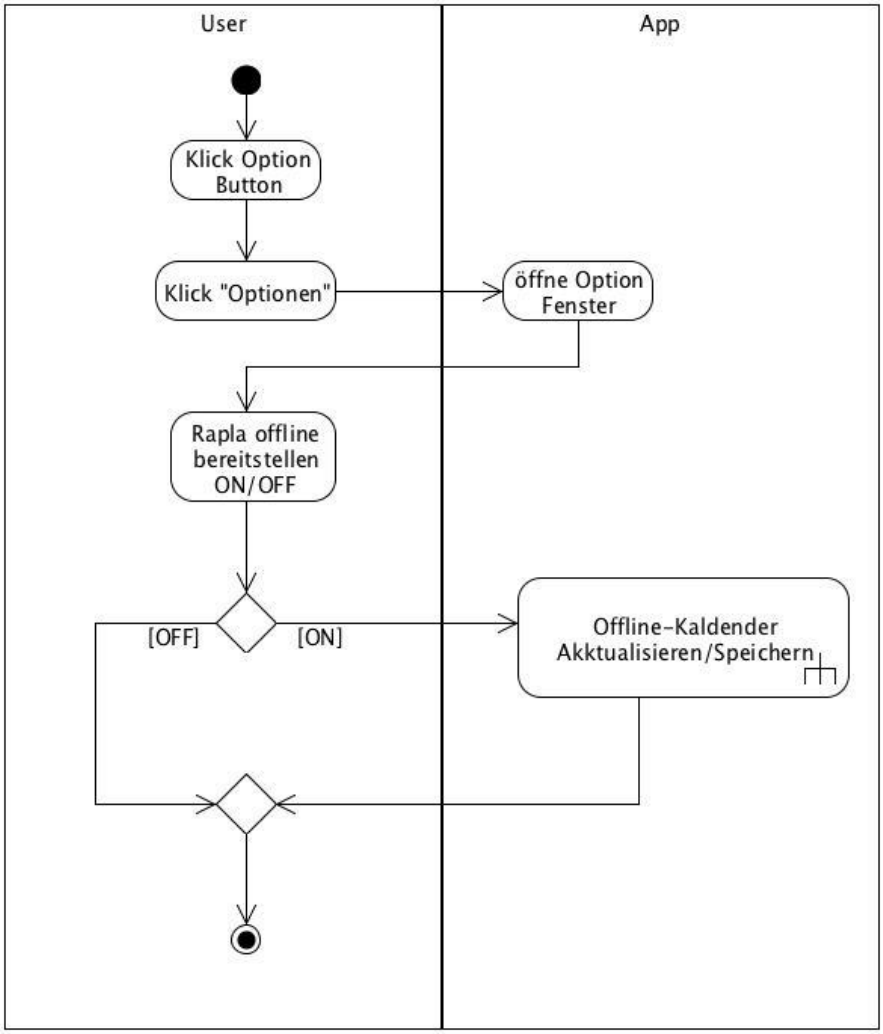
1.1 Brief Description

Dieser Use-Case ist für Personengruppen angedacht, die entweder nicht über eine ständige Verbindung mit dem Internet verfügen oder sich teilweise an Orten befinden, an welchen nur sporadisch oder überhaupt kein Netzzugriff besteht. Dieser Use-Case ermöglicht ihnen, ihre Termine offline verfügbar zu machen, sodass sie den ständigen Zugriff auf ihre Termine behalten. Der Rapla kann so über die App betrachtet werden, anstatt nur einen Webbrowser bereitzustellen.

Events sind ansprechend designt.

2 Flow of Events

2.1 Basic Flow



2.2 Alternative Flows

N.a.

3. Postconditions

Der Rapla steht nach Aktualisierung auch offline zur Verfügung. Das beinhaltet, dass im Falle einer nicht vorhandenen Internetverbindung die offline gespeicherte Version des Kalenders aufgerufen und angezeigt wird.

4. Cucumber test scenario

Feature: Offline Synchronization

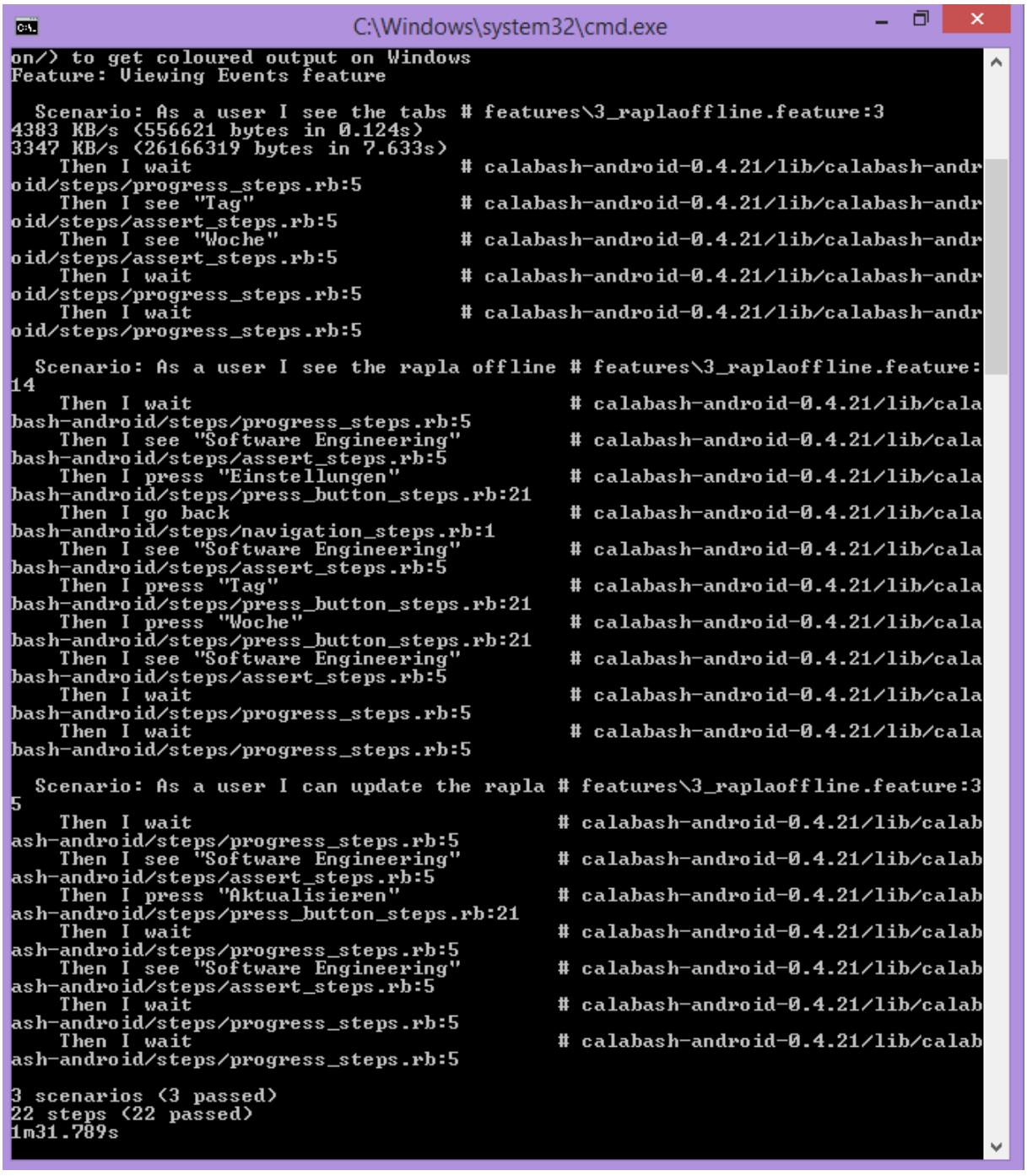
The user can set the synchronization interval

at which the online version of rapla will be copied

and saved locally

Scenario: As a user I see the tabs
Then I wait
Then I see "Tag"
Then I see "Woche"
Then I wait
Scenario: As a user I see the rapla offline
Then I wait
Then I see "Software Engineering"
Then I press "Einstellungen"
Then I go back
Then I see "Software Engineering"
Then I press "Tag"
Then I press "Woche"
Then I see "Software Engineering"
Then I wait
Then I wait
Scenario: As a user I can update the rapla
Then I wait
Then I see "Software Engineering"
Then I press "Aktualisieren"
Then I wait
Then I see "Software Engineering"
Then I wait

Testresult:


C:\Windows\system32\cmd.exe
on/) to get coloured output on Windows
Feature: Viewing Events feature
Scenario: As a user I see the tabs # features\3_raplaoffline.feature:3
4383 KB/s (556621 bytes in 0.124s)
3347 KB/s (26166319 bytes in 7.633s)
Then I wait # calabash-android-0.4.21/lib/calabash-andr
oid/steps/progress_steps.rb:5
Then I see "Tag" # calabash-android-0.4.21/lib/calabash-andr
oid/steps/assert_steps.rb:5
Then I see "Woche" # calabash-android-0.4.21/lib/calabash-andr
oid/steps/assert_steps.rb:5
Then I wait # calabash-android-0.4.21/lib/calabash-andr
oid/steps/progress_steps.rb:5
Then I wait # calabash-android-0.4.21/lib/calabash-andr
oid/steps/progress_steps.rb:5
Scenario: As a user I see the rapla offline # features\3_raplaoffline.feature:
14
Then I wait # calabash-android-0.4.21/lib/cala
bash-android/steps/progress_steps.rb:5
Then I see "Software Engineering" # calabash-android-0.4.21/lib/cala
bash-android/steps/assert_steps.rb:5
Then I press "Einstellungen" # calabash-android-0.4.21/lib/cala
bash-android/steps/press_button_steps.rb:21
Then I go back # calabash-android-0.4.21/lib/cala
bash-android/steps/navigation_steps.rb:1
Then I see "Software Engineering" # calabash-android-0.4.21/lib/cala
bash-android/steps/assert_steps.rb:5
Then I press "Tag" # calabash-android-0.4.21/lib/cala
bash-android/steps/press_button_steps.rb:21
Then I press "Woche" # calabash-android-0.4.21/lib/cala
bash-android/steps/press_button_steps.rb:21
Then I see "Software Engineering" # calabash-android-0.4.21/lib/cala
bash-android/steps/assert_steps.rb:5
Then I wait # calabash-android-0.4.21/lib/cala
bash-android/steps/progress_steps.rb:5
Then I wait # calabash-android-0.4.21/lib/cala
bash-android/steps/progress_steps.rb:5
Scenario: As a user I can update the rapla # features\3_raplaoffline.feature:3
5
Then I wait # calabash-android-0.4.21/lib/calab
ash-android/steps/progress_steps.rb:5
Then I see "Software Engineering" # calabash-android-0.4.21/lib/calab
ash-android/steps/assert_steps.rb:5
Then I press "Aktualisieren" # calabash-android-0.4.21/lib/calab
ash-android/steps/press_button_steps.rb:21
Then I wait # calabash-android-0.4.21/lib/calab
ash-android/steps/progress_steps.rb:5
Then I see "Software Engineering" # calabash-android-0.4.21/lib/calab
ash-android/steps/assert_steps.rb:5
Then I wait # calabash-android-0.4.21/lib/calab
ash-android/steps/progress_steps.rb:5
Then I wait # calabash-android-0.4.21/lib/calab
ash-android/steps/progress_steps.rb:5
3 scenarios (3 passed)
22 steps (22 passed)
1m31.789s

5. Screenshot