# How to release RTFPPL version

Feature list

TBD

Create release candidate

Create release\_candidate\_xxx branch from develop or hotfix\_xxx branch from Master

Set rc-version number

in GitHub\Gift\Libraries\FPPositioningLib\Src\version.h

#define FPPE\_VERSION\_MAJOR 1

#define FPPE\_VERSION\_MINOR 12

#define FPPE\_VERSION\_BUILD 0

#define FPPE\_VERSION\_RELEASE\_ID FPPE\_VERSION\_RELEASE\_ID\_RC

Release testing

Windows release KPI tests –

* use template: rc-kpi-x.x.x.bat from

\\cayyc-proj01\compute02\FPL\_DATA\test\_results\RTFPPL\test\_scripts\

* in template: set correct names and values for script settings (see comments in script for details)
* place bat-file in folder [\\cayyc-proj01\compute02\dchurikov\Polygon\auto-run-scripts\](file:///\\cayyc-proj01\compute02\dchurikov\Polygon\auto-run-scripts\)
* Compare results with previous version. The results have to be bit-exact or can differ for sensible values

Linux KPI tests

* use template: rc-kpi-x.x.x.sh in

\\cayyc-proj01\compute02\FPL\_DATA\test\_results\RTFPPL\test\_scripts\

* in template: set correct names and values for script settings (see comments in script for details)
* run script on cayyc-lc003 server
* Compare results with previous version. The results are supposed to be bit-exact or can differ for sensible values.

Console reprocessing

* use template: rc\_rep\_console-x.x.x.bat in

\\cayyc-proj01\compute02\FPL\_DATA\test\_results\RTFPPL\test\_scripts\

* in template: set correct names and values for script settings (see comments in script for details)
* In folder specified in console\_collection\_path create sub-folders with RTFPPL console and dll
  + 1.15.4 – for regular tests
  + 1.15.4-vdr – for VDR tests with vdr enabled in console (uncomment FPc.setPlatformType(pftp\_FORKLIFT); in main.cpp)
  + 1.15.4-b10 – for special proximity tests in ICA venue (uncomment FPc.setBLEBias(10, 0); in main.cpp)
  + **1.15.4\_baro\_off for special** no baro tests in ICA venue (uncomment FPc.setUseBarometer(false); // in main.cpp)
* place bat-file in folder [\\cayyc-proj01\compute02\dchurikov\Polygon\auto-run-scripts\](file:///\\cayyc-proj01\compute02\dchurikov\Polygon\auto-run-scripts\)
* asses test result by comparison with previous version for bit-exactness or !!!visually!!! if any difference in position output found.

Save test results in file ReleaseAssessmentHistory.xlsx in [\\cayyc-proj01\compute02\FPL\_DATA\test\_results\RTFPPL\](file:///\\cayyc-proj01\compute02\FPL_DATA\test_results\RTFPPL\)

If all tests have been passed go-to next step else fix bugs and repeat the all tests.

RelNotes

Add new feature description and bugfixes in ..Gift\Libraries\FPPositioningLib\README.md

Set release-version number

#define FPPE\_VERSION\_MAJOR 1

#define FPPE\_VERSION\_MINOR 12

#define FPPE\_VERSION\_BUILD 0

#define FPPE\_VERSION\_RELEASE\_ID FPPE\_VERSION\_RELEASE\_ID\_RELEASE

Release merge

* Checkout MASTER
* Merge release candidate into MASTER
* Push MASTER
* Checkout DEVELOP
* Merge MASTER in to DEVELOP

Set develop-version number in DEVELOP

#define FPPE\_VERSION\_MAJOR 1

#define FPPE\_VERSION\_MINOR 12

#define FPPE\_VERSION\_BUILD 1

#define FPPE\_VERSION\_RELEASE\_ID FPPE\_VERSION\_RELEASE\_ALFA

Push DEVELOP

Coursa-Venue server build

Open these Jenkins URLs

Android: RTFPPL : [CoursaVenue-Android-Bridgestone-Release](http://ussjc-zeus01:8080/job/Coursa%20Venue/job/RTFPPL/job/CoursaVenue-Android-Bridgestone-Release/)

iOS: [CoursaVenue-iOS-Bridgestone-Release](http://ussjc-zeus01:8080/job/Coursa%20Venue/job/RTFPPL/job/CoursaVenue-iOS-Bridgestone-Release/)

and check that projects have been built successfully.

Coursa-Venue smoke testing

1. Android test
   1. Install Coursa Venue for Android. Run the application and select ICA venue. Push “start” button and collect data for ~1-minute test rout.
   2. RTFPPL version check – Download Coursa Venue dataset from previous test and check for FPL version. See line with “FPL:x.x.x.x” in xxx\_ipldbg.txt. It must be the number of released version.
   3. BD initialization check. Check correctness of initialization of fingerprints and databases in pf\_log. All fingerprints and databased must be checked: MFP, WFP, BFP, BLP and Map Matching.
2. iOS test
3. Install Coursa Venue for iOS. Run the application and select ICA venue. Push “start” button and collect data for ~1-minute test rout.
4. RTFPPL version check – Download Coursa Venue dataset from previous test and check for FPL version. See line with “FPL:x.x.x.x” in xxx\_ipldbg.txt. It must be the number of released version.
5. BD initialization check. Check correctness of initialization of fingerprints and databases in pf\_log.

All fingerprints and databased must be checked: MFP, WFP, BFP, BLP and Map Matching.

Release mail – if necessarily

Extended testing - if necessarily

Multiseed x12 win release KPI tests