



# **Mode opératoire de mise en exploitation des nouvelles versions de « BiblioMeter »**

Contributeurs :

*Amal Chabli ([amal.chabli@orange.fr](mailto:amal.chabli@orange.fr))*

*Baptiste Refalo, INETUM ([baptiste.REFALO@cea.fr](mailto:baptiste.REFALO@cea.fr))*



Version du **13 décembre 2024**

## Table des matières

I.	Prérequis.....	3
II.	Descriptif du contenu du dépôt.....	4
III.	Procédure de développement.....	5

## I. Prérequis

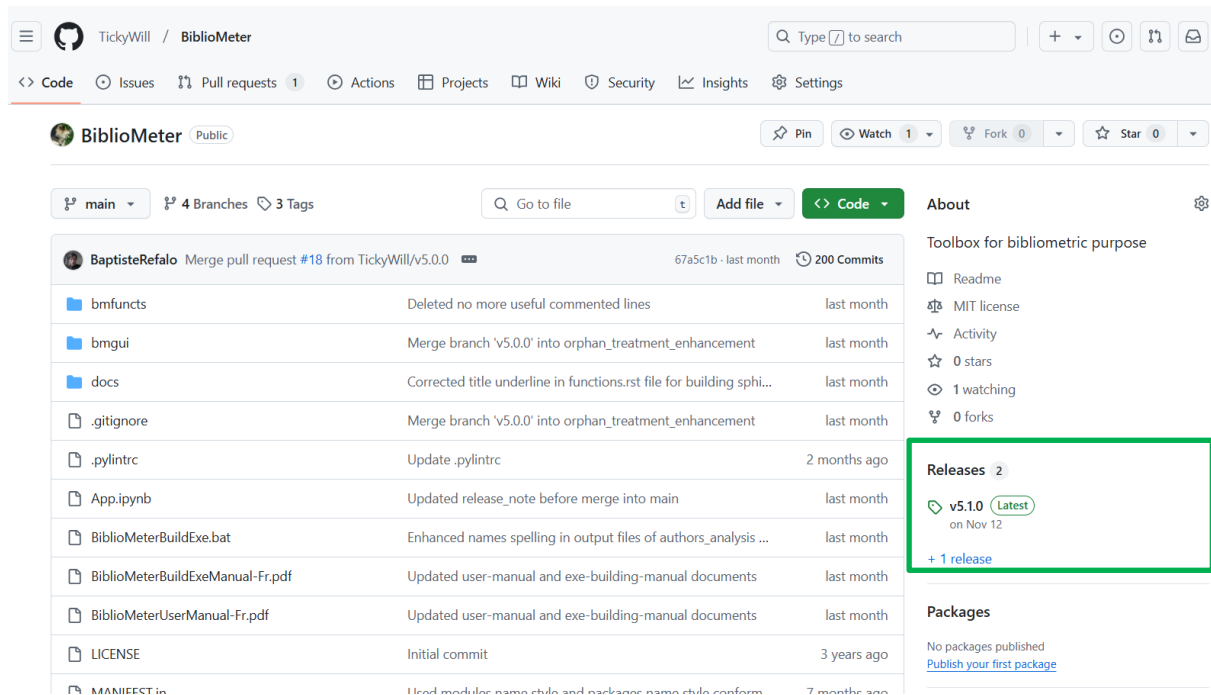
- Avoir un **espace de développement python** (il peut être créé en suivant le mode opératoire écrit par Amal Chabli ([amal.chabli@orange.fr](mailto:amal.chabli@orange.fr)) et François Bertin ([francois.bertin7@wanadoo.fr](mailto:francois.bertin7@wanadoo.fr)) disponible sur demande) ;
- **Être contributeur au dépôt public sur GitHub** du code python de l'application « **BiblioMeter** » : <https://github.com/TickyWill/BiblioMeter> ;
- Disposer des droits **d'accès aux dossiers de l'application « BiblioMeter »** des Instituts sur S (cf : fichier « **BiblioMeterUserManual-Fr.pdf** » disponible sur le dépôt sur GitHub.

## II. Descriptif du contenu du dépôt

Les versions en exploitation et en développement de l'application sont définies par la séquence de nombres « #.#.# ».

La version en exploitation est celle de la branche « **main** » sur GitHub et correspond au numéro de version de la dernière « **Release** » indiqué par le cadre vert sur la page du dépôt sur GitHub ci-dessous.

Dans cet exemple, la version en exploitation est la **5.1.0**.



The screenshot displays the GitHub repository page for 'BiblioMeter' by 'TickyWill'. The repository is public and has 4 branches and 3 tags. The main branch is selected. The file list on the left includes 'bmfuncts', 'bmgui', 'docs', '.gitignore', '.pylintrc', 'App.ipynb', 'BiblioMeterBuildExe.bat', 'BiblioMeterBuildExeManual-Fr.pdf', 'BiblioMeterUserManual-Fr.pdf', 'LICENSE', and 'MANIFEST.in'. The pull request section shows a merge pull request #18 from TickyWill/v5.0.0. The right sidebar contains the 'About' section with links to Readme, MIT license, Activity, 0 stars, 1 watching, and 0 forks. The 'Releases' section is highlighted with a green box, showing 'v5.1.0 (Latest)' as the current release, with a link to '+ 1 release'. The 'Packages' section shows 'No packages published' and a link to 'Publish your first package'.

### III. Procédure de développement

Les actions de la séquence générale décrite dans le tableau ci-dessous donnent la procédure générale à suivre.

Action
1. Créer une branche secondaire à partir de la branche « <b>main</b> » nommée « <b>v#.#.#</b> » avec <#.#.#> <b>le nouveau numéro de version</b> et mettre à jour ce numéro de version dans les différents fichiers qui l'utilisent (« setup.py », « init.py » des « packages », « bmfuncts.gui_globals.py »...).
2. Créer, <b>sur S</b> , une copie du dossier de travail de chaque Institut en la nommant « <b>BiblioMeter_Files-#.#.#</b> » avec <#.#.#> <b>le nouveau numéro de version</b> .
3. Créer une branche secondaire à partir de la branche « <b>v#.#.#</b> » <b>nommée en librement en fonction de l'évolution envisagée</b> dans le code de l'application.
4. Effectuer <b>les modifications et leur test</b> en travaillant au niveau de la branche créée à l'étape <b>3</b> et en utilisant comme dossier de travail le dossier créé à l'étape <b>2</b> . Rem : pour préserver les développements, pousser régulièrement les changements sur GitHub depuis la branche de travail.
5. Faire une relecture de code à l'aide de « <b>pylint</b> ».
6. Mettre à jour la note descriptive des modifications ou « <b>Release note</b> » avec les modifications apportées.
7. Faire une demande de revue par les collaborateurs en émettant un « <b>Pull request</b> » et prendre en compte les retours en reprenant à l'étape <b>4</b> jusqu'à un accord sur les modifications.
8. Effectuer l'intégration des modifications dans la branche « <b>v#.#.#</b> » créée à l'étape 1 par un « <b>merge</b> ».
9. Répéter depuis l'étape <b>3</b> autant que nécessaire.
10. Mettre à jour, si besoin, les documents descriptifs de l'application sur GitHub (modes opératoires, « Readme », documentation Sphinx...).
11. Convenir de la date de mise en exploitation de la version « <b>#.#.#</b> » de l'application avec INETUM et effectuer l'intégration des modifications dans la branche « <b>main</b> » par un « <b>merge</b> ».
12. Informer les utilisateurs de chaque Institut de la mise en exploitation de la nouvelle version avec la mise à jour des dossiers de travail de chaque Institut en tenant compte des éventuels traitements effectués par les utilisateurs depuis l'étape <b>2</b> .

