

945358-BIGPICTURE
Central Repository for Digital Pathology

WP4 - Tools for accessing, annotating and mining digital slides

D4.01 Report on user requirements specification for Cytomine

Lead contributor	15 - ULIEGE
	raphael.maree@uliege.be
Other contributors	1 - RUMC 20 - RO 21 - MUW 23 – HUS 30 - Cytomine 36 - NOVARTIS 37 - JANSSEN And additional contributions from WP2, WP3, and WP4 members

Document History

Version	Date	Description
V0.1	27 Jun 2021	First Draft by Renaud Hoyoux, Ulysse Rubens, Raphaël Marée
V0.2	12 Jul 2021	Second draft based on WP4 and Lygature inputs
V0.3	30 Jul 2021	Final draft based on final review Managing Board

Abstract

This deliverable is the output of the BIGPICTURE Task T4.1 – Develop user requirement specification and aims at establishing precise requirements for end-user tools for accessing, annotating and mining digital slides. We first set up a Cytomine demonstration server (see deliverable D4.02) as a starting point for users to identify missing features. Following a webinar where main existing features were described, we collected more than 100 comments from more than twenty users (incl. clinical pathologists, toxicological pathologists, academic researchers). We then organized one-to-one interviews with members of the consortium to gather additional user requirements and discuss interesting features from other digital pathology tools currently used by EFPIA partners, CRO, and academic partners. This report summarizes all this information and will help drive developments in other WP4 tasks (where prioritization will be decided among partners).

Methods: User requirements specification

To collect user requirements, we first set up a Cytomine demonstration server accessible to all BIGPICTURE partners through online registration (as described in deliverable D4.02). A two-hour webinar was given at the end of March 2021, which was recorded, and made accessible to all BIGPICTURE partners. User feedback was then collected in April and May 2021 through an online, editable, document on BIGPICTURE project shared space. More than twenty BIGPICTURE members wrote about 17 pages of feedback (including screenshots). This document was further analyzed by ULiège, Cytomine, and WP4 members to identify common trends. Interviews (with different members of a given institution/company) were organized in June 2021 where Cytomine developers asked further precision about user feedback to ensure complete understanding of the needs. Discussions about current and future digital pathology practices were also included in these interviews. Overall, 16 distinct individuals (toxicology and clinical pathologists as well as data and computer scientists) were interviewed.

This deliverable summarizes these efforts and organizes the user requirements into main categories of software development that have to be conducted within BIGPICTURE WP4 to fulfil these requirements. Further information about users involved in this process are available in the private report (Deliverable 4.02) to guarantee the protection of personal data.

Results

Using Cytomine current version (v. 3.0.2, <https://github.com/cytomine>) as the basis for user requirements specification, users were asked to describe missing features, issues, and bugs in relation to the future BIGPICTURE platform. While critical feedback was expected, multiple users have anyway positively commented the current version of Cytomine as “intuitive”, “nice tool to use”, “some interface design choices are excellent”, “documentation was fine”, “great for AI developers”, “everything that you would asked for is there”, “great work and many thanks for developing such a great tool”, which suggest that future Cytomine adoption by BIGPICTURE members should be high.

The main (missing) feature requests are listed below and organized in six main sections:

1. Whole slide image (and related data) visualization

Currently, the Cytomine viewer displays a whole-slide image (WSI) after selecting it in the image list of a given project explicitly created by a user. Several BIGPICTURE members asked to improve direct “multiple image viewing”. By selecting an entity that contains multiple images (e.g., a clinical case, or an organ from a given treatment group or a previously registered “image group”), users would like to be able to organize and open all images in one click and display them side by side. An additional feature could allow to link all the displayed images using one click. Currently, the user has to choose the pair of images to link to synchronize their explorations. Users would prefer a more flexible multi-viewer where one could fix one or multiple views (e.g., the H&E slide where magnification and position are fixed) and move in all others (e.g., the IHC slides where either magnification or position could be moved).

In the image navigation panel, users would like to be able to adjust the user interface by displaying/hiding existing tools:

- To be able to hide the scale bar
- Display the metadata alongside the image
- Change the current thumbnail/mini map behavior. Switch the “drag the view box” into “click on the new targeted view”. A drag feature could stay only for “pre-exploration” of the image before selecting the area to target.
- Set a “digital zoom” limit in settings to avoid blurry images.
- Add graphic tablet inputs support (as pinch zoom)
- Automatic rotation of the slide label for proper reading.

Additional features were asked related with the image visualization and are planned to be implemented during Task T4.3 – Optimise Cytomine workflows and user experience:

- Support for devices (such as space mouse, programmable keyboards) to reduce movements.
- Further improvement of the multi-image viewing features:
 - Add a button “close all opened images except this one”
 - Be able to select a slide make it “full screen” and then come back easily to the tiled view of all the opened images
 - Allow the users to be able to organize the tiled image in the order they want
 - “Quick & dirty” registration: point a zone in two images and synchronize them through coarse alignment
- To be able to overlay metadata on the top of an image (in addition to annotations) (e.g., heatmaps produced by an algorithm, focus maps of multi-dimensional WSI, custom color lookup tables for labels and continuous values, virtual staining images, any other arbitrary overlays) and to be able to switch easily between these different data sources.
- A “screenshot/snapshot” feature of the current view and for the global view of the image.
- Moving steps with keyboard should be increased (or configurable per user/project) with respect to image size (double space)
- Rotating image should not be limited to encoded degrees but flexible (“freehand”).
- Switch more easily from one image to another (“next image” button is currently hidden in the image “Information” panel) where an image could be another WSI or another version of the current WSI.
- Superimpose square/rectangular/hexagonal grids (where size could be parametrized) to ease several quantification tasks.
- To reduce bias in toxicological pathology studies where multiple organs can be shown on a given WSI, one interesting feature could be to be able to automatically hide some parts of a WSI (e.g., a specific organ) eventually based on AI for organ recognition.

Image format

The main requested feature in terms of image visualization is the support of DICOM WSI format that is currently lacking in Cytomine and that will be implemented by Task T4.5 – Develop a technical implementation of the open digital slide, annotation, and meta-data format. Support for various compression schemes (e.g., JPEG-XL), other WSI formats (e.g. CZI, more flexible TIFF support) and color coding (e.g. 12 bits) have also been requested. In terms of image format, support for different resolution (mpp - micron per pixel) in both dimensions (x/y) has been requested.

Image upload

A seamless workflow has been requested to upload images from a local site (e.g., a clinical laboratory) to the BIGPICTURE archive through the Cytomine tool. This will be investigated in Task 4.3 – Optimise Cytomine workflows and user experience in collaboration with WP2 and WP3. In addition, visualization of uploaded images without the need to explicitly add them into a Cytomine project has been requested. Also, a desirable feature would allow users to link Cytomine to existing images (e.g. on S3) without having to explicitly upload them, therefore also avoiding data duplication.

Additional data visualization

Users would like Cytomine to display additional, related data and not only WSI. It includes the “Slide preview” image (i.e. the whole slide overview taken by the preview camera), the scanned area, focus map, and the “slide label” image (provided these are included in the DICOM file). A more flexible image list display has been requested, e.g., to display/hide some details such as image description, resolution, etc. We will enhance the Cytomine data and image management system to efficiently support this additional information in Task 4.5– Develop a technical implementation of the open digital slide, annotation, and meta-data format.

As a blind mode exists for project contributors in the current Cytomine version, a blind mode for managers has also been requested.

Performance

Regarding image display, users have noticed lower performance using the remote BIGPICTURE Cytomine demonstration server compared to a regular desktop tool (installed at their own site). While this is expected, other strategies (incl. local caching) might be investigated in Task T4.2 (Extend Cytomine to handle petabyte-sized databases) and Task T4.5 (Develop a technical implementation of the open digital slide, annotation, and meta-data format) in collaboration with WP2, where data security issues might arise. Users noticed:

- First image loading takes 2-3 seconds.
- Image seems to be “jumpy”: irregular loading when browsing from a region to another. User claims the refresh rate is too slow for toxicopathology evaluation (where hundreds of slides might be evaluated per day). Also, zooming creates too much interpolation/scaling artefacts before proper display. Pre-calculation of user displacements has been suggested to speedup display.
- For gamma or color adjustments. The current process is done by fetching new tiles (which are created on the fly) from a backend component which is slower than local computation. An improvement by switching to local processing (assumed by the user browser) e.g. WebGL/shaders has been suggested to speed up the processing. However, backend

processing allows to handle 16bits image (while web browsers only support 8bit images) that might be used for fluorescent imaging.

Other users (incl. data and computer scientists) claimed the refresh rate is sufficient for annotation or data analysis, or they did not notice poor performance or even claimed that Cytomine was faster than other previously tested tools. While some efforts will be dedicated to improve performance (prioritization will be decided within Task T4.2 - Extend Cytomine to handle petabyte-sized databases and Task T4.3 – Optimise Cytomine workflows and user experience), the goal in BIGPICTURE is not to reach performances of a desktop, local, software tool for daily evaluation.

2. Annotations and Metadata (including Ontologies)

Current Cytomine annotation functionalities were appreciated but further improvements or new features have been requested, summarized here.

Complex and cross-image annotations

Users have requested more flexible annotation drawing tools e.g. to enable configurable annotation primitives (e.g. a circle of a given size) and their seamless combination into complex annotations. A brush tool (brush/unbrush with different brush sizes) was also evaluated as highly desirable to speedup manual annotation. In addition, tools to speedup cross-image annotations would be appreciated to annotate more efficiently the same object in multiple images (e.g. a vessel in multiple IHC stains using registration techniques to be developed in Task T4.7 – Develop WSI registration tools). Selection of multiple annotations at once was also requested while Cytomine currently only allows to select one annotation at a time. These additional developments will be performed in Task T4.3 – Optimise Cytomine workflows and user experience.

Annotation selection and navigation

Support for hundreds of thousands/millions of annotations (produced by algorithms) is a request of multiple users (incl. data scientists). It will first require testing the current annotation clustering algorithm and its parameters over different annotation settings (e.g. large numbers of very big annotations) as planned in Task T4.3 – Optimise Cytomine workflows and user experience.

A “Gating” functionality has been requested where users would be able to display/hide annotations based on constraints (e.g., minimum size) and hide/display them accordingly or associate colors (e.g. green if it fulfils constraints, red otherwise) to highlight these objects in WSI.

Content-based image retrieval algorithms will be implemented in Task T4.8 (Develop AI models for content-based image retrieval) to quickly navigate through large sets of annotations.

The possibility to create multiple annotation layers per user is also requested. It is motivated by the need for faster development processes and to ease integration of external annotations without the need to explicitly register each script as a containerized Cytomine apps.

In addition, keyboard shortcuts or new UI buttons have been requested to move from one annotation to another more easily (e.g. with arrow keys between regions of interest).

Semi-automated annotations

Currently, a hybrid workflow is proposed in Cytomine where users first manually annotate regions of interest then train recognition models that are subsequently applied on new images to produce annotations that can be further reviewed by end-users. More interactive and active learning workflows should be implemented to avoid multiple reviewing cycles where launching algorithms should be possible in the image viewing module while it is currently only possible in the “Analysis” Cytomine tab.

The goal of Task T4.4 (Develop AI-assisted WSI annotation tools) is to develop such new interactive annotation tools to speedup annotation processes.

Other requested annotation features include:

- Direct length calculation when drawing an annotation and visualization on the image instead of in the Annotation panel on the side.
- The possibility to create a personal working list which would be a list of annotation task that one user might ask to another user (e.g., a list of WSI where to annotate tumors)
- The possibility to upload annotations directly using XML/JSON files rather than writing code to perform importation.
- Storing of the magnification level when a user creates an annotation (to be later able to assess if the magnification was sufficiently precise to have a reliable annotation for precise measurements or training algorithms)

Ontologies

There is an important need to support efficiently very large ontologies (with hundreds/thousands of terms) with tools facilitating their manipulation such as import/manage/select subsets of existing ontologies for a given project, sharing subsets of ontologies among different projects (without explicit copy but with relative links to the original ontology).

Support for existing pathology-oriented ontologies is of uttermost importance. It is expected that WP3 (Task T3.03 – Development of ontologies and standardized metadata sets) will decide which ontologies to support among the ICD API (<https://icd.who.int/icdapi>) or SNOMED CT (e.g. <https://github.com/IHTSDO/snowstorm>).

3. Search, filtering and sorting (Meta)Data:

In the current version of Cytomine, every user has access to a Storage, a virtual folder with all whole-slide images he/she has access to. To organize, visualize or annotate these images, they have to be added into Projects, a shared area between some user collaborators.

Feedback and one-to-one interviews showed several pathologists want to be able to visualize and find whole-slide images directly from a bucket of all WSIs they have access to without having to explicitly create a project within Cytomine. From these end-user perspectives, the datasets that the user has been granted ethical/legal access to in the BIGPICTURE archive should be seen as a bucket of slides, and the end-user should be able to search, filter and sort in his bucket (or “smart folder”).

Currently, the Cytomine search bar at top right lets the user search images across projects but is rather simple and not useful for all use-cases. Combining suggestions, it emerges that the archive should be queryable to get whole-slide images matching complex criteria on metadata values. For example:

- Find all livers with fibrosis in it
- Find all livers from that study
- Find all bladder cancers
- Find all groups of specimen related to the right lobe of the liver

An ergonomic and intuitive search form is therefore required (e.g. propose suggestions following user typing) to define complex search rules. The search form can be saved at user-level or shared between multiple users to be further reused. These saved searches (or “filters”) can be used to feed new Cytomine projects such that projects can be seen as smart folders (inspired e.g. by macOS smart folder system). The project (smart folder) manager could enable an option to automatically feed the projects with new images matching the search rules. For power users (e.g. data scientists), support of complex search queries will be implemented through direct API requests. All search queries will be implemented following WP2 data access policies and technologies (including Beacon) during Task T4.2 - Extend Cytomine to handle petabyte-sized databases

Regarding user experience, users welcomed filters in Cytomine (with wildcard as textual search and intuitive sliders for numeric filtering). However, unselecting options in dropdown lists could be more intuitive and some pathologists are more used to filtering at column-level (Excel-like). In such a setting, the results would be displayed in lists with columns and every item result can be expanded to show more details. Metadata shown in columns are however fixed while they should be organisable by the user (choose column order, show/hide some metadata columns, ...). Such a visual configuration should be saveable as a preset at user or project level. Results can then be sorted (at column level) but a single column at a time, while sorting on multiple attributes could be considered. To deal with possibly thousands of slides fulfilling constraints, responsive user interfaces will be designed during Task T4.3 – Optimise Cytomine workflows and user experience.

Users find the Cytomine concept of “Project” very meaningful for annotation as it makes the collaborative annotations of WSIs easy. It is a shared area between users, some of them having rights for project administration. It is worth to note that Cytomine has been designed with annotation tasks in mind. Some pathologists noted that project organization as a shared area seems to be less suitable for usual diagnostic routines (visualization of instantaneous search results would be sufficient).

Inside a project, images are currently organized in a “case-centric” way while some users would like a study-centric organization. Other users have mentioned the need for a hierarchy and grouping by specimen/case/cohorts (e.g., different samples from an organ or a case, different blocks, ...) instead of only one-level “project” hierarchy. It would require slide-to-specimen, specimen-to-case and case-to-cohort relations (following the definition of Entity Relations of the Common Mandatory Metadata Structure in Deliverable D3.02). More generally, the current flat organisation of projects and images within Cytomine should be revamped to support sub-projects and groups of images in a (sub-)project. Grouping images in a project exists as an experimental feature in Cytomine, but like projects, creation is still manual. The system should automate or at least suggest interesting grouping.

4. Algorithms, AI, an API interaction

Integration of algorithms (machine learning, computer vision, artificial intelligence) into Cytomine front-end is already possible using containerization technologies and some programming conventions (such as specifying algorithm parameter types in a descriptor file) that enable execution of algorithms through the web user interface (batch execution mode) or remote command lines, and visualize annotation results on top of WSI. Additional mechanisms and features have been requested:

- Ease of use: Simplification of the UI for algorithm execution within the image viewer (instead of from the “Analysis” tab would be a must (“one-click” button to run a workflow with templates with pre-saved parameter values), see also Section on semi-automated annotations.
- Cross-project execution: Currently an algorithm can be associated to a Cytomine project and run within the context of this project. Users would like to be able to leverage images and annotations from multiple projects to train recognition models
- Complex outputs: Currently integrated algorithms output results such as annotations (geometries and ontology term). While this allows to cover multiple image analysis tasks (tissue segmentation and delineation, cell counting, object recognition,...), Cytomine should be extended to support complex outputs that are useful for other tasks, incl. new whole-slide (or sub-) images (such as generated by pre-processing, virtual staining, or a transformed image following registration), heatmaps (such as probability maps generated by segmentation models), deformation settings and complex links to original WSI (such as those generated by image registration algorithms to be developed in Task T4.7 – Develop WSI registration tools), additional data (such as a grade in gleason grading). The API and clients should then allow injecting and manipulating these data efficiently, e.g., by thresholding heatmap probabilities in the web viewer to visualize regions of interest according to algorithm predictions.
- Workflows (both in terms of UI and data models) for image registration are lacking, as well as basic scripts to manipulate annotations between slides (e.g., transfer of annotations between sections using estimation of transformations). These will be designed during Task T4.7 – Develop WSI registration tools.
- Flexible support of heterogeneous GPU devices (see deliverable D4.02) will be investigated during Task T4.6 – Implement a container-based infrastructure to integrate, connect and run external AI-algorithms.
- Content-based image retrieval algorithms for searching annotations will be implemented in Task 4.8 – Develop AI models for content-based image retrieval.

Presentation (listing) of algorithms should also be improved to allow end-users to more easily identify the purpose of a given algorithm. Structuration of algorithms into types of image analysis tasks and precise description might be useful to provide a more comprehensive list. An algorithm management plan should also be defined.

Additional notes regarding reproducibility and the private execution model are given in the private report on *Cytomine demonstration server and source code (D4.02)*.

5. Security

Feedback related to this section were mainly given by computer scientists and fall into two categories: the security of the source code and the security of the user authentication. In Task T4.2 (Extend Cytomine to handle petabyte-sized databases), Cytomine will be extended to integrate with the WP2 infrastructure and connect to ELIXIR Authentication and Authorisation infrastructure (AAI) for user credential management and secure dataset access.

Improvement of user key management

The main concern is related to the **keys** used by Cytomine to authenticate users and scripts through the REST API (e.g., in headless mode such as AI tools dealing with Cytomine data).

Currently, every user owns a set of keys. These keys are stored in the Cytomine database and shown to its owner into its *Account* page. This set of keys is presented as public and private key pairs, suggesting an asymmetric-key scheme while it is more a shared secret key scheme.

It is asked to improve security relative to user keys. At first, the private key should be hidden in the *Account* page, and the user password has to be entered to see it.

A better practice is to use a true asymmetric-key cryptographic scheme by discarding the private key after generation and only showing it once to the user. The server will then **not** store the private key of the users. If a user lost its keys, it could always regenerate a complete set of keys.

It is recommended to track and show the usage of these keys for detecting malicious use.

These improvements will increase the security related to authentication and thus the data access and the detection of any suspicious behavior allowing administrators to quickly take appropriate actions and countermeasures.

We also had two requests for extending the connection keys feature:

1. Provide multiple key-pairs which can be used for different projects. With this feature, a connection to Cytomine established through a set of keys will not always have the same permissions as a connection through the Web UI and could have more limitations.
2. Provide multiple key-pairs which have different levels of access (read-only, read/write). This request, related to the access level and not the considered objects, is similar to the previous point and offers the same security advantages and drawbacks.

Continuous security audit of the source code

In addition to a global security audit of the web assets (with source code scanning tools like netsparker.com), some consortium members have asked for a process for Cytomine update auditing to avoid the introduction of potential security breaches.

Besides Cytomine source code, a continuous code dependency analysis process should be considered and will be implemented in Task T4.13 – Tests, documentation, and validation.

A further detailed analysis and additional comments regarding the source code and security are included in the private report on *Cytomine demonstration server and source code (D4.02)* at the *Security evaluation* section.

6. Documentation, installation, help, and community management

The first points are related to the first interaction of a person with the Cytomine ecosystem: when installing, browsing, or reading documentation about Cytomine. The last point is more specific and is related to the contribution process with the Open Source Cytomine platform.

Documentation

A common feedback of users was about the scarcity of the documentation with outdated sections, and a hard-to-understand version definition (between Cytomine official version and Cytomine ULiège R&D version).

A “developer guide” helping the developers to use the Cytomine clients and API was missing. More examples or documentation for the Python client were explicitly asked for their global or specific (how

to query the image backend directly) uses.

Since then, a whole new documentation server has been set up. This new documentation contains additional sections dedicated to data scientists with a focus on the Cytomine API and the related external clients.

Task T4.13 – Tests, documentation, and validation will be based on its new documentation engine allowing versioning and collaborative contributions.

Installation

The installation scripts (Cytomine-bootstrap) and procedure (on the documentation) are currently dedicated to installation on a unique server through bash scripts. Deployment on multiple servers is possible and documented but it requires substantial manual configuration.

More details have been requested on how to deploy Cytomine in more complicated settings or with some container orchestrators (e.g., Kubernetes).

Help when using Cytomine

In addition to the positive comment related to the intuitive user interface, we received the following improvement propositions that should help a user during its browsing experience:

- Set pop-up tags (or “floating help”) for each tool or function displayed with explanations. They already exist for some of the annotation tools and should be extended to all tools. A user will have first information on a displayed tool without the need to search detailed explanation into an external documentation website.
- Allow further configurable simplification of the UI (improvements to the Configuration tab of a project)
- Display more descriptive error messages into the frontend. Some error messages are too generic and do not inform the end user about the experienced problem.

Community management

As Cytomine is an Open-Source project with different versions and “flavors”, users have requested to clarify how to coordinate development, bug reporting, contribution, and support of Cytomine. For example, on which flavors and platform report the issues.

The current way is through the GitHub repositories <https://github.com/cytomine/> but more detailed information is needed and will help for the adoption of Cytomine by BIGPICTURE stakeholders.

Discussion

No deviations from DoA and contingency plans were needed.

Conclusion

We have been able to establish a first set of precise requirements for BIGPICTURE platform by: 1) providing access to BIGPICTURE members to a server with current Cytomine version 2) by organizing a webinar (recorded) to describe existing features 3) by organizing interviews with end-users to better characterize missing features.

This list will help drive developments in other WP4 tasks. Developments will be prioritized within each WP4 task based on the objectives of the BIGPICTURE project to develop a web platform to remotely search, annotate, and analyze millions of slides, and to facilitate collaboration between pathologists and data scientists.

Based on these results WP4 efforts will be prioritized to achieve the following:

1. DICOM WSI support (Task T4.5 – Develop a technical implementation of the open digital slide, annotation, and meta-data format);
2. security and interoperability with WP2 infrastructure (Task – T4.2 Extend Cytomine to handle petabyte-sized databases);
3. Flexible and efficient mechanisms to search and filter slides, metadata, and annotations (Task T4.2 – Extend Cytomine to handle petabyte-sized databases);
4. Smart mechanisms to perform semi-automated annotations based on AI (Task T4.4 – Develop AI-assisted WSI annotation tools),
5. implementation of support for complex algorithm outputs (Task T4.6 – Implement a container-based infrastructure to integrate, connect and run external AI-algorithms),
6. Cytomine Web UI ergonomic improvements (Task T4.3 – Optimise Cytomine workflows and user experience).