SOCIETIC
SOCiety as Infrastructure for E-Science via technology, innovation and creativity

Deliverable no.	D4.2.
Deliverable name	Cell Images Application Report
Dissemination level	PU
WP no.	4
WP name	Use of the infrastructure. Applications and Creativity
Date	15/05/2013
Date of delivery	31/07/2013
Actual date of delivery	22/07/2013
Status	Final
Author(s)	Eduardo Lostal, Silvia Almonacid
Reviewer (s)	Candida Silva, Francisco Brasileiro, Fermín Serrano

SOCIETIC is supported by the European Commission under Contract Number: RI-312902

Change log

Version	Date	Author/Editor	Reason for change / issue
1	20/5/2013	Eduardo Lostal, Silvia Almonacid	Creation
2	16/07/2013	Candida Silva, Francisco Brasileiro	Update and review
3	22/07/2013	Fermin Serrano	Final

Table of Contents

1. Summary

This deliverable compiles scientific, technological and social aspects regarding the cell images application, covering analysis, deployment, issues appeared during those stages and initial results.

The application is in production, although we are already working in a second version for production. An iterative and incremental development methodology has been followed. That was the methodology chosen given the requirement changes and the interaction with designers in order to get the best of the user interface and usability. Eventually, a production candidate version was deployed in Pybossa production server.

All versions of the application have been developed using HTML5 + JavaScript on the client side, and Python on the server side.

Initially the WP4 plan was designed thinking in having deployed Semantic maps application first (deliverable month 6) and Cells application later (deliverable month 10). We decided to speed up the development and publication of the Cells application taking advantage of the academic year so that secondary schools students from Spain could join the experiments. Initial evaluations, results and feedback have been obtained already. In the 2013-2014 academic year, we plan to launch the experiment at European level. Thus, further information will be included in the deliverable focused on that experiment.

2. Introduction

One of the major lines in cancer research relates to the identification of drugs to fight it. Some of these drugs could function by selective killing of tumor cells, interacting with proteins or metabolic processes (chemical reactions that take place inside cells) that are modified in tumor cells but not in healthy ones.

Therefore, a first step in the search for antitumor drugs could be the identification of chemical compounds that kill tumor cells, selecting in a second step those drugs that do not kill healthy cells. There are two major types of cell death, apoptosis and necrosis. Apoptosis is a type of programmed cell death, physiologically controlled, that eliminates cells in a silent manner. Necrosis, on the other hand, causes an inflammatory response due to cell content release that can damage surrounding tissue. When we want to kill tumor cells inside a tissue we want them to die by apoptosis. A similar successful approach has been used with stem cells, which in some aspects resemble tumor cells.

These two types of cell death can be easily differentiated by microscopy in cultured cells. Advanced optical microscopy techniques allow cells to be cultured directly over the microscope in multiwell plates (cell culture flasks with several individual wells). In this way we can culture tumor cells and treat them with a different chemical (a putative pharmaceutical drug) in each well, and observe which chemicals are more efficient in cell killing, and by which mechanism (apoptosis or necrosis) by observation of changes in different cell structures. These structures can be individually labeled with specific fluorescent dyes and observed using fluorescence.

By photographing cells at different time intervals and assembling consecutive images after the experiment we can obtain videos that show cell movement and changes over time, which is known as time-lapse microscopy. Cells are grown in each well of a 24-well plate, adding a different chemical per well, and photographed every half hour with the appropriate filter for two fluorescence colors and in bright field, obtaining three different images of the same field at each time point, in four different positions per well, in all 24 wells, over a period of 24 hours. This produces 14112 images in a single experiment. The different images obtained in each field at each time point were overlaid (bright field, blue and green fluorescence) to obtain a combined image, and were assembled chronologically to generate videos that show cell movements, cell division and cell death, if they happen, over time.

The resulting sequence of more than the thousand images need to be analysed. This analysis includes classification of the cells according to several parameters like cell and nucleus shape, mitochondria distribution or current cell status. All those huge data sets obtained cannot be analyzed by means of individual researchers. The optimal approach, given the quick advance of the technology, would be to face the analysis making use of computer image analysis techniques in some kind of distributed infrastructure with high performance computational resources. Nevertheless, computers are not as good as human brains in what concerns to pattern recognition. Namely, the classification of the previous parameters, although might be difficult even for humans, becomes harder for computers without an exhaustive training.

An application is developed to provide a framework to the volunteers in order for them to have access to the images and carry out the analysis. Usability has been one of the main concerns. The goal is to get it working for a wide spectrum of people who are not expected to be scientists. Therefore, a key point in the development of the application is that it must be intuitive and user-friendly in order for the volunteer not to get lost with non-relevant features; user must be self-sufficient.

3. Application Specifications

3.1. Description of the application

The application shows one image at a time to be analysed by the citizen scientist. There are nine different parameters that made up the analysis. In order to ease the task to the user, every one of these parameters is analysed and replied in a different step, keeping the image always visible. The end-user has the chance to change the fluorescence channel to properly observe the parameter being studied at a time. Every step has several samples in order to provide the user with a close example of what he or she has to classify. A magnifying glass tool is implemented to allow the citizen scientist to stare at the image in a more accurate manner.

Since the application is expected to be used by any person interested in the science process, a short training must be taken to get a necessary background. That is provided through a tutorial during the first time the application is used. In addition, an endless number of tooltips and step-context specific information are available at any time.

A relevant aspect of this scenario is to get participants involved in the science process. In such a way, a remark section is provided in the last step for citizen scientists to be able to write anything that might have been noticed. The idea is to encourage people to participate in that research process, what in other successful scenarios like the Galaxy Zoo one, has led to significant contributions and discoveries.

Results must be gathered and saved. The same image is sent to a number of users to be analysed. Their replies are matched and through a statistical tool with the whole amount of responses for an image, the final result to be passed to the scientist is obtained. It is the match of the whole replies what sets a more accurate and trustable method. Possible errors are corrected through the collective intelligence of the participants.

3.2. Final requirements

In the following, requirements of the application are listed.

- There will be a tutorial to be displayed on the first time the application is used
- Tutorial will contain context of the project
- Tutorial will contain application usage explanation
- Every task will analyze only one image
- Possibility to magnify or enlarge the image in case image it is too small to look at its details

- Possibility to move the image among the three different channels
- There will be the chance at any moment to access the video the image comes from
- Biotechnology background information to help the analysis must be available at any moment
- There must be present a link to the Global Excursion project, and namely, the cells excursion
- Application will be split into several steps
- It must be clear in which step the user is at every moment
- A task will not be saved unless it had completed the last step
- There will be samples, either images or videos, of every step of the analysis to help the user
- There will be ready a questionnaire to evaluate the application that will be displayed at the end of the first, fifth and tenth task done
- User must be able to provide some comments/remarks at the end of the analysis
- The parameters that must be analized for each cell are:
    - Current status (alive, dead, not sure)
    - Cell shape (elongated, rounded, star-shaped)
    - Cell content release
    - Nucleus shape (rounded, elongated, bean-shaped, condensed/fragmented)
    - Mitochondria shape (worm-shaped, dotted, not sure)
    - Mitochondria distribution (clustered, scattered, not sure)
    - Cell displacement
    - Cell shape change
    - Multinucleated cells
    - Comments, remarks
- Tooltips must be used as another help for the user
- Different sections of the user interface (image, operation area, sample/information area, etc.) must be easily identifiable
- Design must priorize usability and make application as much intuitive as possible

3.3. Privacy concerns

There is no privacy concerns for this application, results can kept open. Results can give a clue about what happened in the cell culture, that is, the image. However, images are not linked to the compound used at that particular culture.

4. Development

Application consists in a web application on HTML 5 plus JavaScript running on top of PyBossa, a Citizen Science Infrastructure. Development has been more complicated than expected given the difficulty to reach a trade-off between usability and application requirements, in addition to a large set of changes in the latter. That is why, development and the versions until releasing the first one on production has been focused on usability and a proper design, intuitive, attractive and motivating for the volunteer.

4.1. Architecture

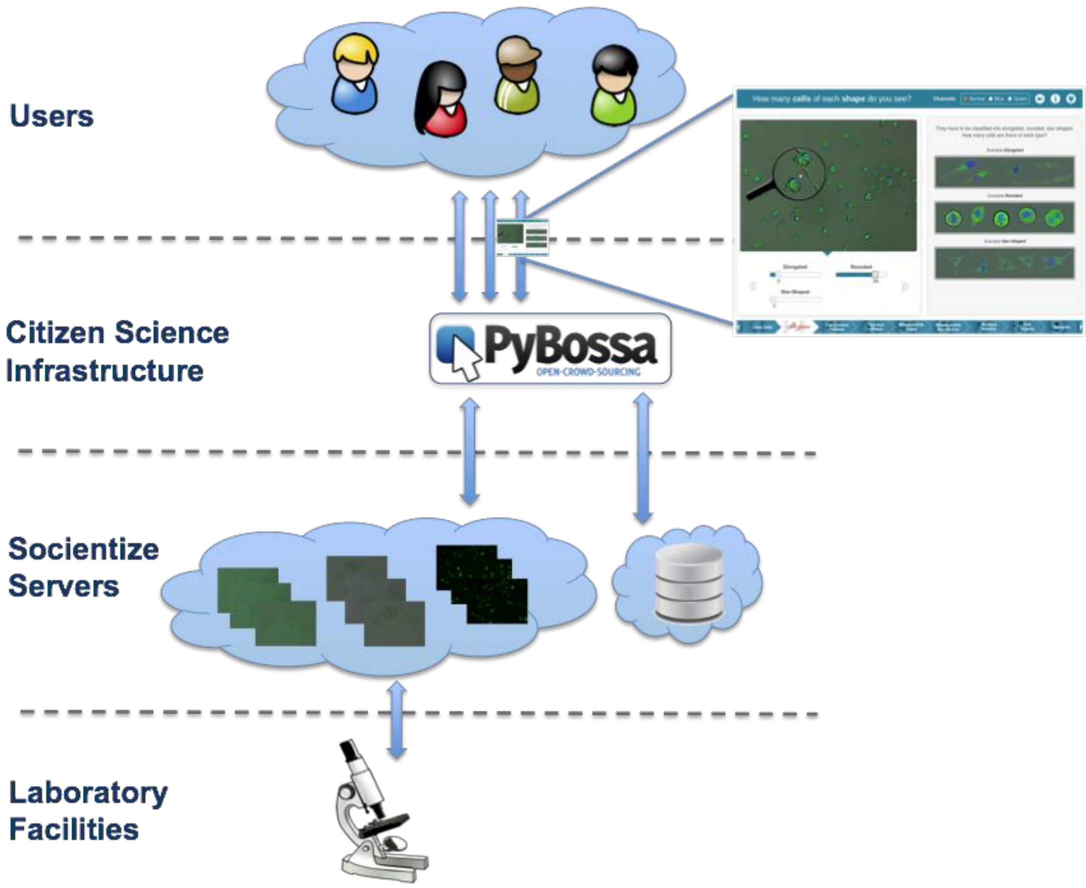
In this subsection the whole architecture will be covered, from data gathering to end-user.

Field experiments are carried out through equipment that comprises a completely automated fluorescence inverted microscope (Leica DMI 6000B), an Optigrid system for semiconfocality and a temperature and CO2 concentration control unit for live cell microscopy. This equipment is placed in a laboratory at BIFI facilities.

Cells are grown in each well of a 24-well plate, adding a different chemical per well, and photographed every half hour with the appropriate filter for two fluorescence colors and in bright field, obtaining three different images of the same field at each time point,
in four different positions per well, in all 24 wells, over a period of 24 hours. This produces 14112 images in a single experiment. The different images obtained in each field at each time point were overlaid (bright field, blue and green fluorescence) to obtain a combined image, and were assembled chronologically to generate videos that show cell movements, cell division and cell death, if they happen, over time. That video obtaining by photographing cells at different time intervals and assembling consecutive images after the experiment is known as time-lapse microscopy.

Images taken from the above mentioned equipment are stored at Ibercivis servers making them accessible. Database and PyBossa are also placed in those servers. PyBossa is the Citizen Science Infrastructure used to support the development. It is a free, open-source crowd-sourcing and micro-tasking platform. A PyBossa server is deployed and kept in touch with the input and the database. Application is developed on top of PyBossa and it is an instance of the application what is created in the PyBossa server. It is that instance, the one that on request accesses either to the data input and the database. PyBossa server, and thus the application, has a public IP through which it can be accessed. Users can access the application through the Internet and make use of it.

The overall architecture is depicted in the picture below. Note that besides PyBossa server, input data (images) and database are all placed within Ibercivis servers, they are split into two different parts in the picture. That is done to clarify the explanation as a separation of concerns, but a technical reader must keep in mind that two inner sections of the picture are indeed one.



Overall Architecture of the Experiment

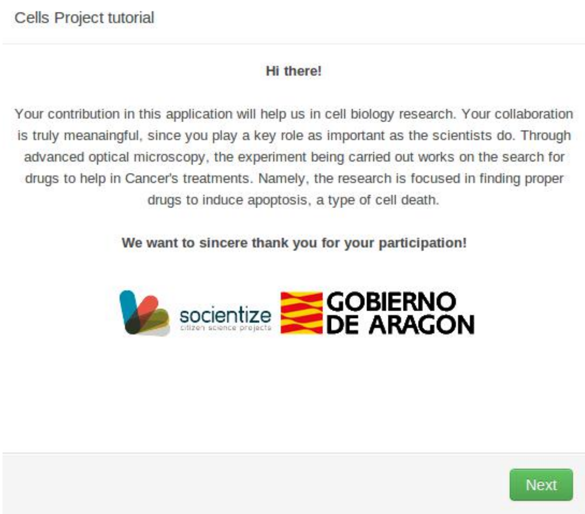
4.2. Application developed

Application result of the development fulfills the whole requirements. In the following, developed application is detailed.

First of all, first time an authenticated user or first time a new IP starts the application a tutorial appears in the form of a pop-up window. If user clicks in any other area of the browser out of the tutorial, the pop-up disappears and it is possible to start contributing pressing the corresponding button.

Tutorial is made up of five steps that introduces the application. The aim is twofold. On one hand, tutorial provides an introduction to the context of the application and biotechnology background. On the other hand, it provides a quick and short tutorial about the application usage. The steps are the following:

1. **Introduction**
   It provides a quick introduction to the project and displays logos of the participant entities.



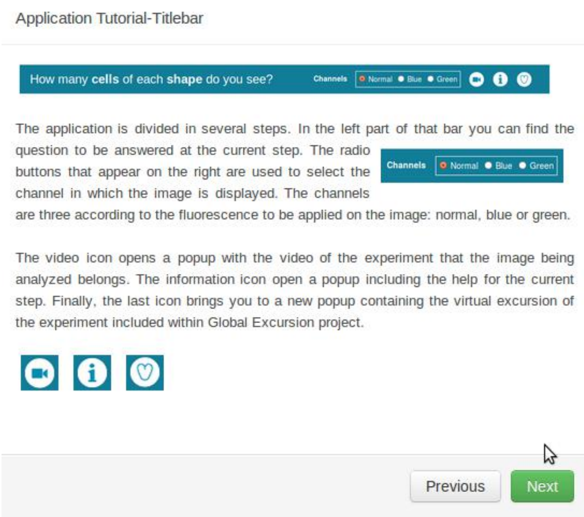
Tutorial first step screenshot

2. **Video Tutorial**
   The second step of the tutorial displays a video of the project. It intends to set the context in which the research is framed, at the same time that provide a starting point for the biotechnology background needed to do the analysis and learn something from it.

   The video is available at:
   - English: http://youtu.be/XXegth8CmM4
   - Spanish: http://youtu.be/HYjnEXXtSe0

   The remaining steps are used to help the user in the usage of the application.

3. **Application Tutorial 1**
The rest of the steps are used to help the user in the usage of the application. Third step explains the contents of the title bar and what every button does.



Tutorial third step screenshot

4. **Application Tutorial 2**
This step is similar to the previous step, but it presents the structure of the central part with an example of a photography to be analyzed, the answers and the samples.

5. **Application Tutorial 3**
The last step is also similar to the previous two. In this case, the bottom bar is explained and the way to move among the steps of the application is described.

Once the tutorial is finished, the application starts. Given PyBossa support, application programming is reduced to work with HTML 5 plus JavaScript. Application makes use of several external JavaScript libraries that are imported. Actually, they are stored in Ibercivis server in order to ensure they are always available when the application is accessed.

Look&feel is always the same for the nine steps that make up the application. However, the answer section and samples change in every step. Design will be discussed later. Thus, in the rest of the section application description, will be provided for its functionality avoiding design regards.

Application contains several buttons with different functionalities that are the same independently of the step on which the user is in. They are the following:

- **Channels**
  User can switch, at any time, the channel of the image to be analyzed between blue, green or bright field fluorescence. It is done through a radio button that only allows to select one a time.

- **Video**
  It opens the video in a pop-up window where the current frame belongs.

- **Information**
  It opens a pop-up window with the information related to the current step in order to explain the user what to do and provide him the necessary knowledge about biotechnology.

- **VisHub**
  It is a button for the Global Excursion project. It opens in a pop-up window with the excursion related to the cells project in the Global Excursion portal.

Application is divided in nine steps. It is possible to move among the whole nine steps through the arrows to go to the previous and next steps. Information is kept from step to step, even in the case of going backwards, the information inserted in subsequent steps is kept unless going backwards meant reaching the first step. In the first step, values for all the rest of steps are taken and based on them the whole slidebars are created. Therefore, everytime the user comes back to the first step, it is assumed that changes are done and subsequent step results are erased. The number of cells introduced in this first step is important since the rest of steps check that the total amount of cells selected in the corresponding step match with that number. If they do not match an error message pops up indicating the total expected number of cells. And the possible answer gets a red border to signal the user that something is wrong.

In order to help the user, every element of the application has a tooltip telling the user what it is used for or providing some information about it. The only one that does not have a tooltip is the main picture to be analyzed because on hover the cursor of the mouse changes to be a magnifying glass to help the user to visualize properly the picture.

Application, as mentioned above, is made up of the nine steps, detailed in the following:

1. **Alive Cells**
   The first step must check the amount of alive cells. The whole cells on the picture must be classified as alive, dead or not sure. Since it is not possible to know yet the amount of cells, in this first step, user must insert the amount of cells of each type in the corresponding text box. Since a number is expected, a function has been added to filter any other input from the

text box. Thus, although user types any key, only the numeric keys will be valid and displayed within the text box.

2. **Cell Shape**
   Shape of every cell must be classified among elongated, rounded or star-shaped. As abovementioned (and it is done in the same way in the remainder steps), addition of the number of cells per class must be equal to the total amount of cells inserted in the first step. In order to ease the task to the user, from this step onwards, sliderbars are used to save the result according to each parameter.

3. **Cell Content Release**
   At this step the classification to be done is whether the cells release cell content or not.

4. **Nucleus Shape**
   Shape of the nuclei must be classified in rounded, elongated, bean-shaped or condensed/fragmented. At this step, channel switches automatically to blue fluorescence since nuclei shape should be easier to observe with that channel.

5. **Mitochondria Shape**
   Cell mitochondria shape must be analyzed at this step. Cells must be classified into worm-shaped, dotted or not sure. Similarly to the previous step, channel switches automatically to the green one since mitochondria are dyed with that color.

6. **Mitochondria Distribution**
   This step analyzes mitochondria distribution classifying them into clustered, scattered or not sure. In this step, the channel is also automatically switched to green. Even in the case that in the previous step channel had changed, it will move to green channel.

7. **Nucleus Mobility**
   Mobility of the nuclei is analyzed allowing the classification in mobile and not mobile. Both in the sample and the information a video example appears to show the desired behavior. Since mobility is being analyzed, user must use video button to take a look at the video of the whole experiment where the current picture belongs. Channel switches again to the bright field one.

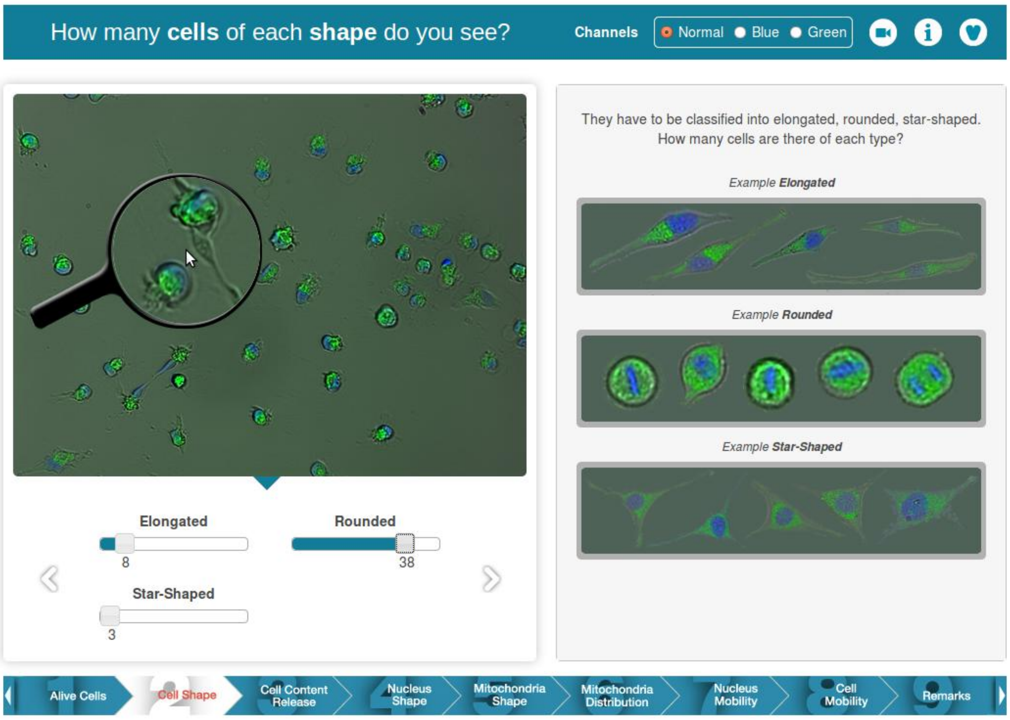
8. **Cell Mobility**
   Changes in the surface of the cell are analyzed at this step setting the classification of the cells in mobile and not mobile. Both in the sample and the information a video example appears to show the desired behavior. Since mobility is being analyzed, user must use video button to take a look at the video of the whole experiment where the current picture belongs.

Channel switches again to the bright field one.

9. **Remarks**
   Last step must count the amount of multinucleated cells. It also provides the user the chance to do any comment or remark about something strange that the picture can contain. Some of these things to write about are proposed in the sample area. They are:

   - Anormally large or small cells
   - Anormally large or small nuclei
   - Cells do not move at all
   - Granularity (little spots) inside the cell
   - Very mobile cells
   - Cells cluster together
   - Cells with aberrant divissions

A screenshot of the application is depicted in the following picture.



Screenshot of the application

Most of the code written is located at the file *cellsPresenter.html*. It contains the HTML and JavaScript. All external JavaScript libraries used are linked within that file and stored in a folder within the server. Namely, these libraries are stored in the path */static/js/cellsJs* within PyBossa folder at Ibercivis servers. In the same folder it is also stored the file *cells.css*, that contains the style sheet. In such a way, the presentation layer is lighter and code is not not that long.

Besides those libraries and the style sheet, there are several files that have to be modified when using PyBossa as the underlying infrastructure. They are the following:

- *app.json*: Defines several parameters of the application like name, short_name (application univoke identifier), thumbnail, description or the question defining the application.
- *CellImagesData.csv*: A CSV file with the initial task to be created.
- *cellsPresenter.html*: It is the HTML file that contains the application itself, including the JavaScript. It can be updated separately, that is, changes that only affect the presenter can be done without affecting the rest of the application and the tasks already created.

- *createTasks.py*: It is the file that runing locally will create or update the application using PyBossa API. Some parameters are set in this file, like the number of replicas, the scheduler used, etc.

- *long_description.html*: An HTML file containing a description of the application that is displayed at the section of the project within PyBossa server.

- *tutorial.html*: An HTML file containing the tutorial to be displayed during the first use of the application.

An additional file must be added for the application to be multilanguage that contains the translations for the languagues supported: *cells-internationalization.js*.

The most important file is cellsPresenter.html, since it contains the application itself. It is described in the following.

At the beginning, necessary css files for JavaScript libraries are linked and so it is done for the css file that provides the customized look&feel to the application. That is stored at *cells.css*. After this, messages pop-ups are detailed and the whole HTML elements of the application. Those elements are organized in nested tables following the order of the elements from top to down and left to right. Thus, first titlebar and its items appear, later the analysis area, that is, picture to be analyzed and response area; samples area, progress bar and finally, the alerts.

After HTML elements, JavaScript libraries are linked and JavaScript code provided. The latter starts with some variable declarations and the function to load the texts according to the language of the browser. Those declarations are for global variables that keep the results of the current replica, the current step, the number of cells inserted by the user or the base name for the files. *clearInputs* function initializes the value of the elements while *changeChannel* changes the channel of the picture being analyzed. *loaduTask* is the function to change from one step to another one. According to the step it shows the proper HTML element for the samples and the response area, set the proper channel by default for the step, updates the title of the title bar, the image of the progress bar and clear the elements of the response area if an error appeared before. *markAsWhite* is the function to carry out that last task, while *markAsRed* shows through a red line that an error in the response area has appeared (usually wrong amount of cells). *resultsGathering* checks on step change whether inserted results are valid. If they are, results are stored on memory until last step is completed. Otherwise, an error is shown. In the case of the first step moving to the second one, the whole slidebars are initialized with the amount of cells inserted by the user in the first step. Later the functions for slider bars creation are placed. Immediately after, a function to escape characters and for binding mouseover event to the magnifying glass functions are added. More JavaScript functions for enabling tooltips and dialog creation are included before the ones for magnifying glass ones. The most significant functions related to PyBossa management of the tasks are added at the end. *changeuTask* manages the changes among steps, while submitTask stores the results in the database and loads the new task as well as preloads the next one (this is a PyBossa feature to speed up the task loading). If questionnaire must appear, it is added in that function. *pybossa.taskLoaded* is the responsible for that task loading. *pybossa.presentTask,* on the other hand, is the responsible for the whole initialization of a new task. *pybossa.setEndpoint* sets the endpoint to the application; without it Pybossa's server cannot serves the tasks properly. *pybossa.run* launches the application.

4.3. Multilanguage support

With the goal of spread as much as possible the application and encourage its usage, multilanguage support is added. At the time of this document creation, the languages supported are English, Spanish and Portuguese. Translations have been carried out by Portuguese and Spanish partners. PyBossa infrastructure detects browser language and passes it to the application. Thus, language selection is done transparently to the user. According to the language of the browser, the application looks for its translation among the available ones. If it has that language, the proper translation is loaded. Otherwise, default language is loaded (English).

Application code has been adapted to support multilanguage. Every content of the application with some text is tagged with an id. The same for tooltips and images with texts that change. Text is not loaded until language is checked. That is done during page loading, after the whole HTML elements have been created. Consequently, page loading slows down.

Translations are stored in a JavaScript file, *cells-internationalization.js*. For every language, it contains pairs of tag and the proper text for that tag. Thus, the same tag is duplicated for every language translating the text according the language at hand. Consequently, to add an extra language is straighforward: duplicate the tags of a language and translate the texts to the new target language. JavaScript library used to carry out the translation like this is jQuery i18n, a jQuery based Javascript internationalization library.

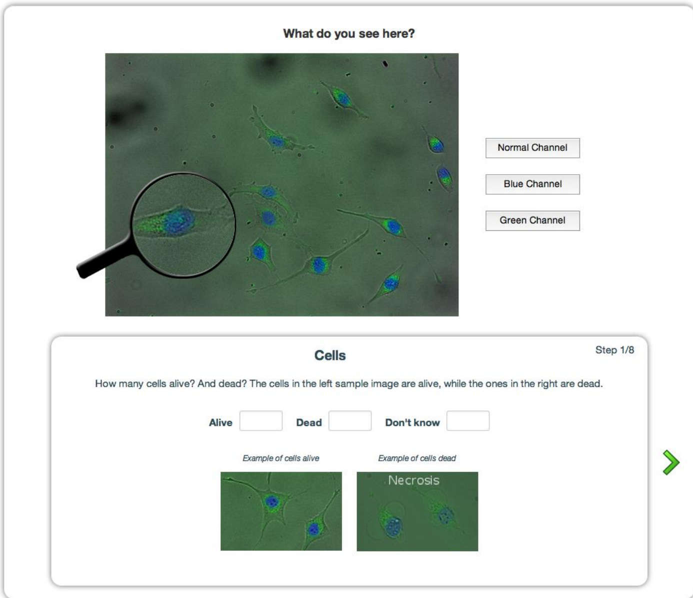
4.4. Development stage

Development has followed an iterative and incremental methodology and, finally, the implementation was performed. Deployment was done in a test server. In such a way, evaluation and test can be done there. Changes in the requirements or issues in the evaluation, repeated the cycle meaning changes in the analysis and the implementation. Main changes were related to design looking for a more usable and easier application. Also, feedback received during the development forced changes in the application.

The first version of the application contained the whole steps mixed in only one. Image appeared on the left, while on the right part thanks to the use of css the nine steps were encapsulated in nine frames. There was not much large information to carry out the analysis or assist the user. At the bottom, there was a picture that could be changed through the arrows at its sides that went through the whole images that composed the sequence that the image belonged to. Below it there was a thumbnail of all these images that on click loaded that thumbnail in the image that could change. The idea was to provide the user the option to check every frame of the whole sequence easily.

Soon that version was discarded because of its lack of usability. Then, application was split into the nine steps that compose it. Reply of every step was kept on the right part. Testing the application, it was found out that the images of the whole sequence confused the volunteers since attention was not focused on the image to be analyzed. Therefore, all those images were removed and replaced by a video.

To ease the task to the user, sample images were added to illustrate what volunteers had to look for. However, volunteers were confused and thought that the analysis was to be over those images. That led to a reorganization of the layout and the use of labels to explicitily mark those images as samples. In the following figure, it is possible to observe a screenshot of the application at that point.



Screenshot of one of the versions of the application

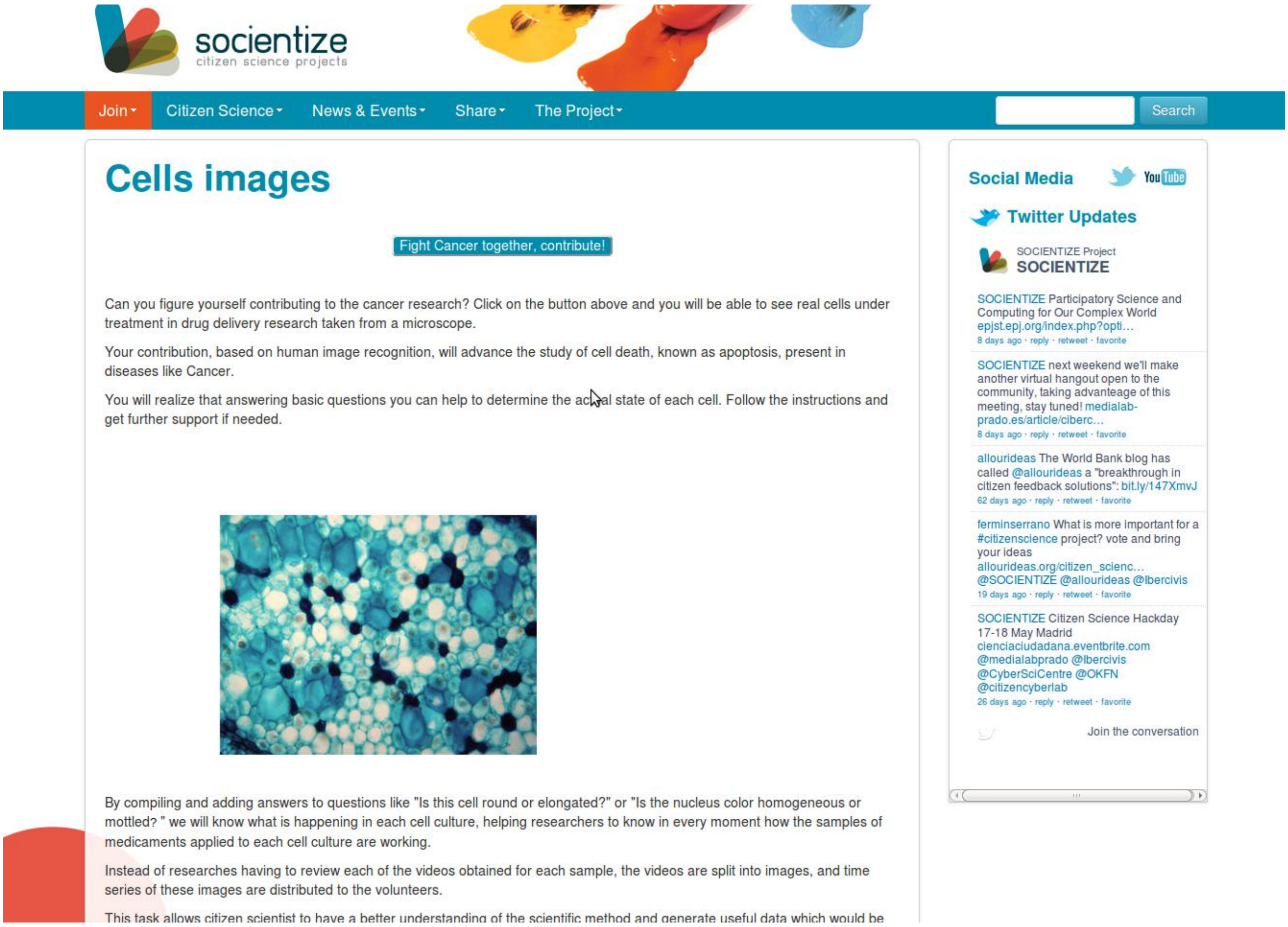
At that point, feedback received from the teachers targeted the work on providing more information to help the volunteers to carry out the analysis. Efforts were focused on improving the tooltips, adding more sample images, information and tutorials.

At some point application functional behavior was working properly, but still design was not attractive for the end user. Therefore, as one of the last steps of the work was focused on the design, layout and look&feel of the application, done with a designer. Final version of the application was the result of those efforts.

4.5. Application entry points

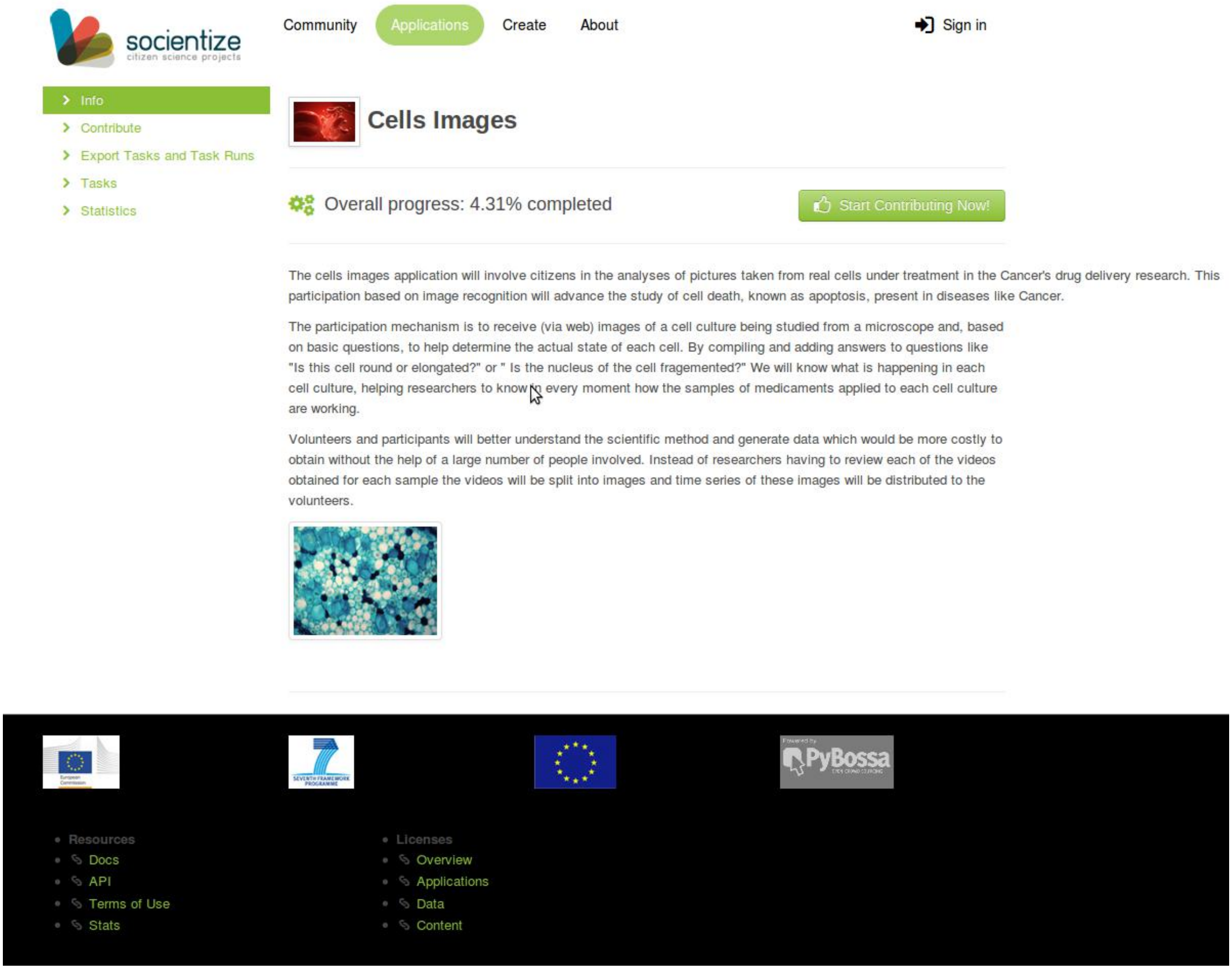
Application is only accessible through PyBossa production server at Ibercivis. It is the only place where application is running and can be accessed. However, given it is a site, it is possible to add the link wherever wished.

In such a way, link is present in Socientize website what means it is accessible from it also. As it is depicted in the figure below, there is a button with the title "*Fight Cancer together, contribute!*" that leads directly to the application in PyBossa.



Cells section in Socientize website

The other entry point is the PyBossa server as abovementioned. There is a section where a small description of the application is provided and a button with the title "*Start Contributing Now*" moves to the application. That is depicted in the picture below this paragraph. From PyBossa server is also possible to entry directly to the application from the main and application pages.



Cell section in PyBossa

Currently, PyBossa does not allow to outsource the application and embed it in other place keeping the authentication. At BIFI, a project to develop that feature is being carried out what will allow in a future to embed the application within Socientize webpage and run from it.

4.6. Questionnaires

In order to gather some information about the application through the end users, questionnaires are used. Taking advantage of the participation of schools in the initial phase, several questionnaires have been prepared and are presented in the following subsections.

4.6.1. Cell images experiment experiences

This is a general survey for every type of user. It asks for an evaluation of the application and its features. The goal is to gather the end user impression about the application. It is prepared and managed by ZSI people. For authenticated users, it appears after the first, fifth and tenth completed task. For an anonymous user, it appears after every completed task.

It is available both in English and Spanish at:
http://survey20.zsi.at/index.php/246162/lang-en

4.6.2. Knowledge

This questionnaire is thought to be filled by the students twice: once before they do nothing about the experiment and once again when they finish working with it. The idea is to measure whether the students actually leearned something from the experiment. In other words, they are used to validate the idea that students may gain knowledge through citizen science approach. Therefore, the questionnaire contains a set of questions about biotechnology related to the field of the experiment being carried out. Questionnaire is only sent to the teachers who distribute it among their students. It is prepared and managed by BIFI people.

It is available only in English at:
https://docs.google.com/a/bifi.es/spreadsheet/viewform?fromEmail=true&formkey=dFZmel90NERSR2FtdHZ2aUJ3ZXpUR3c6MQ

4.6.3. Teacher experience

It is another evaluation questionnaire, but this time oriented to the education field. The idea is that teachers evaluate the application and how it fits with education. It is prepared and managed by ZSI people.

It is available in English and Spanish at:
http://survey20.zsi.at/index.php/225484/lang-es

5. Design and User Experience

In this section design details are provided. It is structured in several subsections according to several design approaches.

5.1. Usability: Functionality and easiness

Application design focuses in the usability, usage easiness and a clear presentation and information strucuture. Following those directives, excessive stetic work is avoided, without leaving aside an attractive look&feel to attract application use from new and usual volunteers. Keeping that in mind graphical resources like setting up a hierarchy for element size and layout, color support, an efficient use of the format, content structure and a legible sans-serif tipography.

5.2. Size and layout

Application is displayed horizontally aligned in the center of the browser. Therefore, it is possible to take advantage of its horizontal margins likewise it happens with the Socientize webpage.

Application sizes are set to be visualized in most of screen resolutions without need for scrolling. Its maximum width is 1024 pixeles and its height displays the whole components without wasting space, keeping a clear and clean structure.

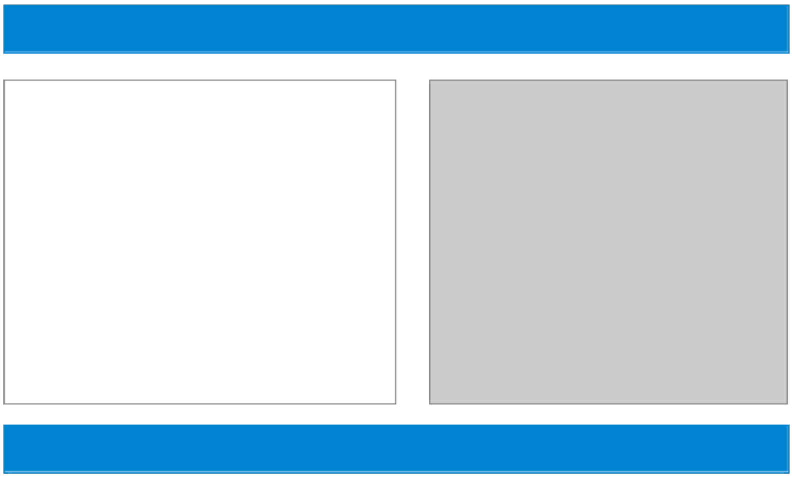
5.3. Color

Design has been developed to be harmonized with Socientize webpage look&feel. Application colors are the ones set in Socientize identity manual. Otherwise, Socientize webpage style is followed. In such a way application look&feel is unified with global Socientize image.

Color is applied to organize information and make application usage more intuitive. Blue color helps to visually delimite the application and address its attention in key points. Among chromatic values from the identity manual, blue color is chosen since that is the one that transmits more comfort to the user and serenity to analize a cellular photography thoroughly. Namely, the blue used is the #127D98.

5.4. Structure

Application information is organized following a logic order. It is divided into four parts: two horizontal bars which vertically delimit that application given their blue color (instead of the rest of elements that have desaturated colors). Between both of them, two modules contain the information. Element placement responds to occidental lecture, placing the most important from left to right and from top to down.



Application structure from design point of view

5.4.1. Horizontal Top Bar

It is made up of three parts:

1. At the left of the bar a question is placed. It is the most important element within that bar. The question guides the user through the features to be studied in the cells of the photography below. Bold tipography is used to help the user to note the key words of the question, and so the features to put attention in the photography. In addition, to foster user's attention in the question, a serif font is used that contrasts with the sans-serif used in the rest of the application. Question changes at every step of the application and it matchs with the steps displayed at the bottom bar.

2. Next to the question, we place buttons corresponding to each of the color channels among which the user can switch. They are marked within a white line that highlights and delimits them. In order to show channel choice, a radio button has been chosen instead of a drop-down menu since it is more intuitive for the user because the whole alternatives are always visible.

3. At the right part of the bar, optional information is placed for the user. A symbolic representation is chosen for design reasons and because these icons are very representative of the information they contain. These icons are for an information pop-up window about the current step, the video from the current frame is taken and a link to Global Excursion project. These icons have the same style: blue symbol over white circle. In order to do their

usage more user-friendly, the chromatic combination changes on hover showing the biggest color contrast possible according to the chromatic choices from Socientize corporative manual (blue/red-orange), keeping always the maximum legibility.

5.4.2. Horizontal Bottom Bar

It contains a guide for the nine steps that the volunteer must complete. Direction of the steps is reinforced through light arrows that do not bother to the whole. In addition, through highlighting in a chromatic way the current step, changing it from blue to white that step, a help to the user is provided.

Bottom bar

At both sides of the bar an arrow that is used as a button is placed. In order to improve their usability, they change its color on hover from blue to white. They are used to move among the steps.

Usage of shading allows to overlap several information layers in a legible way without adding further color changes. Thus, it is possible to highlight the size of the numbers keeping them behind the text that describes the task of each step. For instance, in the following figure, number five is easily distinguished at the back thanks to its bigger size. Shading avoids the usage of another color to make its shape clear. At the same time, text is also easily readable thanks to the change to white and the overlapping on the number at the back.

Step in bottom bar

5.4.3. Left module

Shading provides to this module a volume that highlights its content and gains significance in contrast to the right module whose information is secundary.

It contains the cellular photography and the answer section. Both the photography and the answer section are related to the picture content. Thus, they are distributed one below the other with the same horizontal centered alignment. An arrow visually leads from the photography to the answer section. Given the relevance of the photography, its position is an exceptional one on the top of the module and it is the element with the biggest size of the whole application. In order to ease the user the visualization of the photography, when the cursor entries in the region of the photography changes to be a magnifying glass that increases the part of the photography where it is placed.

Answer section is made up of three parts: possible answers themselves and two arrows. Arrows are placed at both sides of the possible answers. They are buttons to move to the next step (right arrow) or the previous step (left arrow). These arrows have also a shading similar to the one of the module. On hover the arrow changes its look to simulate a pressed arrow.

5.4.4. Right module

This module is an auxiliary one used to provide samples to the user easing the analysis. It contains either photographies or videos and a brief description about the task to be done. That means, its content changes at every step. Given its auxiliary nature photographies' size is smaller. The module has a light grey color background and a thin border of a medium grey. That light grey for the background makes information of this module to move to the background and loses relevance, visually speaking, in contrast to the left module which is highlighted because of its shading and white background as above mentioned.

As a summary for design section, it is worth mentioning that in order to provide an optimal result, application design must take into account and be aware of technical aspects that programming requires. The aim during the design stage is to offer the information in an easy and comprehensible way.

6. Deployment

Deployment is done in the PyBossa's servers that have been prepared at Ibercivis facilities. Those servers are three: test, alfa and production server. They are detailed in the following:

- Test server
  - Accessible at: test.pybossa.socientize.eu
  - Ip: 155.210.135.168
  - Purpose: It is the server where new features are tested, as its name suggests. Its main usage is for developing new parts of PyBossa.
- Alfa server
  - Accessible at: alfa.pybossa.socientize.eu
  - Ip: 155.210.135.167
  - Purpose: It is a server that has the same configuration that the production one. It is used for testing new applications before moving them to production and new features already tested in the Test server that are almost ready for production.
- Production server
  - Accessible at: pybossa.socientize.eu
  - Ip: 155.210.135.84

- Purpose: It is the production server, the one used by the general public and where applications are ready to use.

At the time of application deployment, only production server was available. Thus, it was deployed directly on it, skipping the deployment on the alfa server. When the other two servers were created, application was also deployed on them also. New versions, updates or modifications are done now on the alfa as a previous step to move them to the production one.

Initial deployment and task creation is done through the following command line:

python createTasks.py -u server -k userApiKey -i inputPath -f taskFile -c

Where:

- *server*: It is the URL where the server is running, for instance the production one: http://pybossa.socientize.eu/pybossa.
- *userApiKey*: API key of the user uploading the application (a valid API key for the server where it is being uploaded must be provided).
- *inputPath*: It is the path within the server where the application will look for the input data, for instance: *static/input/*.
- *taskFile*: It is the file containing the tasks to be created, in this case a CSV file has been used.

Once the application is running, any change in the presentation layer can be updated without modifying the tasks using the -t option:

python createTasks.py -u http://pybossa.socientize.eu/pybossa -k userApiKey -i static/input/ -f CellImagesData.csv -t

Note that parameters have been replaced for valid values (except userApiKey) to illustrate the use of the command line order. In order to be able to run these commands, it is necessary to have installed a version of Python greater or equal to 2.7 and less than 3.0; and pybossa-client (and consequently, pip for installing python packages).

From late March 2013, application is on production. Although it was not until May that it was considered finished with the addition of the final video in the tutorial.

Initial deployment is made up of 4067 different tasks from a unique experiment of the microscopy. In this context, different tasks mean different images. The number of replicas per task is set to 15, to be considered as a proper approach and statistical measure. Therefore, given that task redundancy, the whole amount of tasks (meaning task in that context the different tasks itselves and their replicas) is 61005, that is, 4067*15.

Scheduler chosen for the task delivery to the users is the "*breadth_first*" whose features are detailed in the following:

1. It sends the tasks in the order that were created, first in first out.
2. It ignores the task redundancy value, so it will keep sending tasks even though that value has been achieved.
3. It sends always the task with the least number of task runs in the system.
4. It ignores if a user has participated in the same task, allowing to submit one more answers for the same task by the same user (anonymous or authenticated).
5. A task will never be marked as completed, as the task redundancy is not respected.

From the point of view of the application, the scheduler will be trying to obtain as soon as possible an answer for all the available tasks.

6.1. Initial phase

Besides being in production, which means it can be used by anyone, the initial phase is set out as a proof of concept stage to receive the feedback of the users before its international launch and dissemination.

At this initial phase sixteen schools from the whole region of Aragón, Spain, and at least one from Madrid participate in the experiment. It means between 600 and 800 students working as volunteers as a starting point. This phase is expected to last during the last trimester of the course. In such a way students can carry out their analysis at that period. Therefore, initial phase is expected to finish in late June 2013. It does not mean that the application is removed at the end, but it is the moment to analyze the results and feedbacks in order to improve the application for its international launch.

6.2. Foreseen phases

Next step is its international launch. Since application fits pretty well with education, new version with the improvements from initial phase deployment is scheduled for late summer 2013. In such a way, it will be ready for the beginning of the new course and so, more schools can join the experiment. At the time of the creation of the current document, schools from Portugal and Spain have already shown interest in participating. At the same time new version gets ready, international launch will be done as planned in the dissemination plan.

7. User's Feedback

During development and deployment stages, feedback has been received that made the application evolve and, eventually, lead to a new version.

First significant feedback was received in a meeting hold at BIFI facilities with the teachers participating in the experiment on the 14th January 2013. Current status of the application at that time was presented. Their feedback was very fruitful and encouraged to move development efforts of the development from usability to make easier and more understandable the analysis itself. They liked the application and, besides design could be improved, its look&feel. Usability was good for them. Teachers' main concerns regarded to how difficult was the analysis to be done, images that contained cells where the parameters to analize were not clear for them, lack of more examples, tutorials, etc. As a result of the meeting, the following changes were done:

- Reduce image sizes since Internet connection is not good
- Provide more biotechnology background, improve the tutorial at the beginning
- Provide videos of the experiment
- Add more written information as a complement to the images
- Provide more reference images since analysis is too complex and classify the cells is difficult; more sample images make the task easier

The other relevant fedback came after the deployment of the application from the partners of the project, colleagues and, mainly, end users. That feedback was very positive to find the weak points of the application and design the new version of it. Most of the comments referred to the fact that the images were too complex to analyze since they contain too many cells and the more cells, the more difficult to count them. The other significant critique had to do with the length of the tasks. Tasks are too long, nine steps are too much for the users. In a similar regard, some people complained because they had to leave a task uncompleted because it was too long and could not saved and continued later on. Some people also recommended to make use of the slider bars in every step that it is possible.

7.1. New version

Taking into account the feedback about the complexity of the application, the new version aims to simplify the task to the user without forgetting the search for valid scientific results. It reduces the number of steps of the application. Instead of having nine steps per task, it will have between four and five steps removing the less interesting steps but keeping the most relevant for the research. The idea is that less steps mean less time to perform a task what may attract more volunteers to use the application. Otherwise, too long tasks make the application boring and may take volunteer to loose interest. The other significant change is that no text fields or slider bars are used. Instead of loading the user with the task of counting either by typing the amount of cells or by moving the slider bar, user only has to click on the cell. In such a way, user is released from the tedious task of counting a large amount of cells. User must select the option of the current step to mark and going through the image clicking on the cells that meet that feature. For instance, in the first step, the one where user must classify cells in alive or dead, there will be three buttons: alive, dead or not sure. Clicking the button for living cells, user can go through the image clicking on the cells that, he thinks, are alive that keep marked as so.

These are the main changes included in the new version of the application. The latter one will be also very useful for the comparison of the results got from the citizen science analysis and the automatic computer vision software to be developed.

We are also working in the naming for this application, in order to improve marketing and dissemination. Candidates are "Cell Spotting", "Cell Discovery", "Fight For Cell" between others.

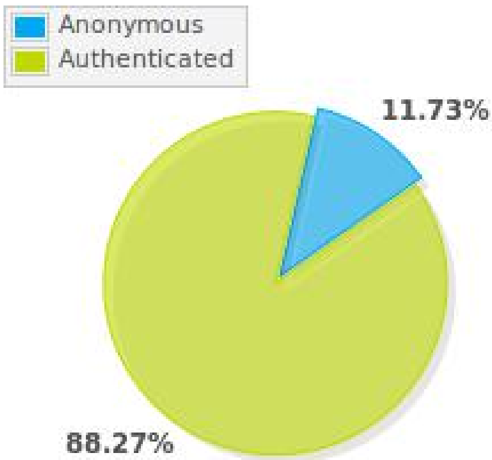
8. Results

As mentioned above, results are open and can be downloaded by anyone at any time. They are available at: http://pybossa.socientize.eu/pybossa/app/cellsimages/export. They can be exported either in CSV or JSON format. One example of the JSON format is shown in the following with the whole fields for the analyzed parameters.

[{"info": {"mitocohondria_worm": 27, "remarks": "", "mitocohondria_distribution_notsure": 0, "movility_nucleus_null": 30, "cellular_shape_star": 7, "nucleus_elongated": 8, "mitocohondria_shape_notsure": 0, "cells_alive": 33, "cellular_shape_elongated": 23, "nucleus_bean": 2, "release_no": 34, "cells_dont_know": 0, "mitocohondria_scattered": 33, "mitocohondria_rounded": 7, "multinuclei": 0, "nucleus_fragmented": 0, "movility_nucleus_mobile": 4, "mitocohondria_grouped": 1, "movility_surface_mobile": 4, "release_yes": 0, "task": "http://societic.ibercivis.es/pybossa/static/input/feb2013-10-19", "movility_surface_null": 30, "cells_dead": 1, "cellular_shape_rounded": 4, "nucleus_rounded": 24}, "user_id": 337, "task_id": 9264, "created": "2013-04-18T12:04:07.321949", "finish_time": "2013-04-18T12:04:07.321979", "calibration": null, "app_id": 421, "user_ip": null, "timeout": null, "id": 3153},

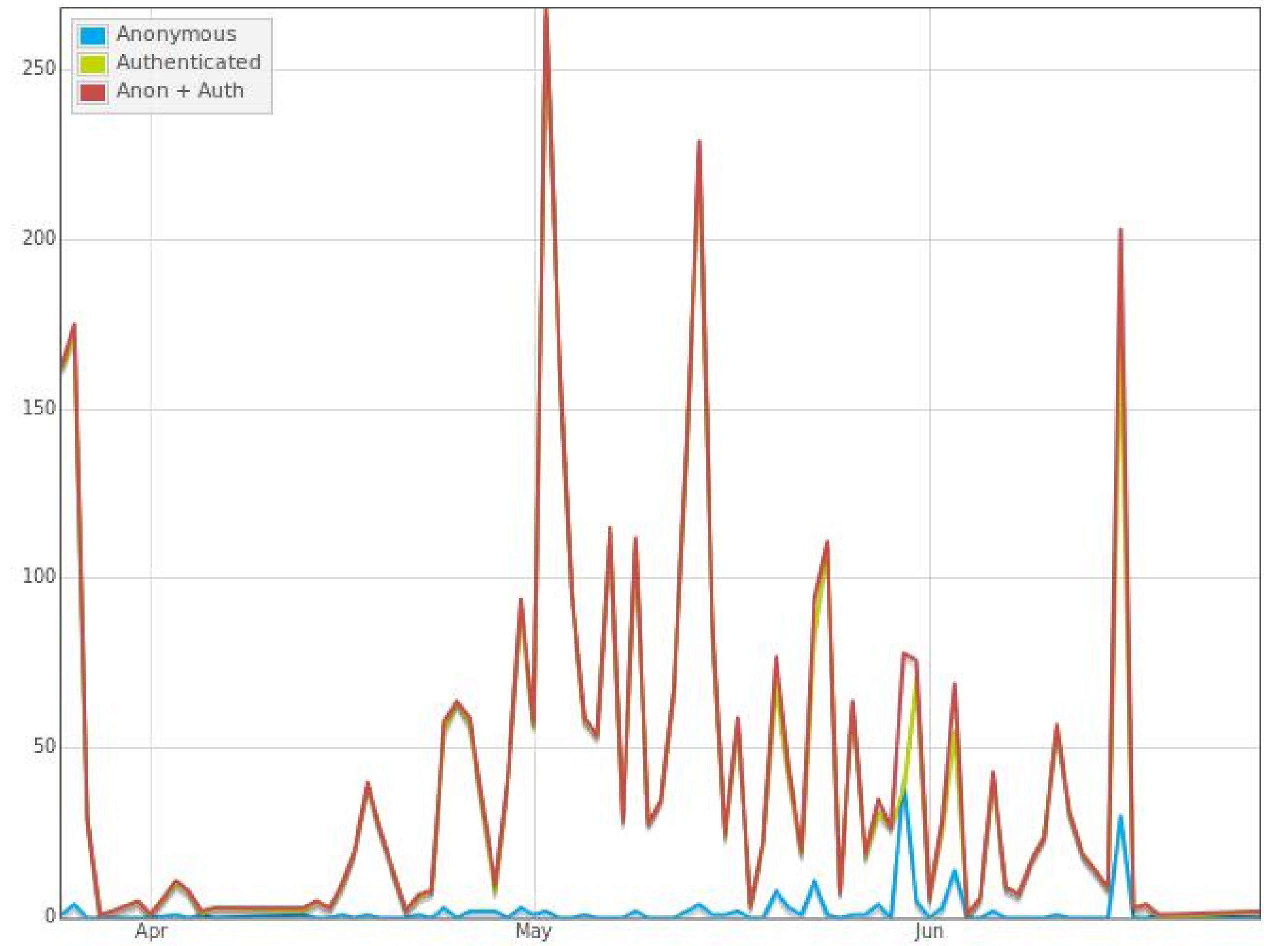
{"info": {"mitocohondria_worm": 30, "remarks": "", "mitocohondria_distribution_notsure": 0, "movility_nucleus_null": 2, "cellular_shape_star": 17, "nucleus_elongated": 19, "mitocohondria_shape_notsure": 0, "cells_alive": 35, "cellular_shape_elongated": 18, "nucleus_bean": 1, "release_no": 19, "cells_dont_know": 0, "mitocohondria_scattered": 6, "mitocohondria_rounded": 6, "multinuclei": 3, "nucleus_fragmented": 2, "movility_nucleus_mobile": 34, "mitocohondria_grouped": 30, "movility_surface_mobile": 34, "release_yes": 17, "task": "http://societic.ibercivis.es/pybossa/static/input/feb2013-10-19", "movility_surface_null": 2, "cells_dead": 1, "cellular_shape_rounded": 1, "nucleus_rounded": 14}, "user_id": 328, "task_id": 9264, "created": "2013-04-18T12:09:56.454279", "finish_time": "2013-04-18T12:09:56.454310", "calibration": null, "app_id": 421, "user_ip": null, "timeout": null, "id": 3157}, {"info": {"mitocohondria_worm": 17, "remarks": "", "mitocohondria_distribution_notsure": 4, "movility_nucleus_null": 32, "cellular_shape_star": 3, "nucleus_elongated": 11, "mitocohondria_shape_notsure": 9, "cells_alive": 29, "cellular_shape_elongated": 29, "nucleus_bean": 7, "release_no": 11, "cells_dont_know": 1, "mitocohondria_scattered": 11, "mitocohondria_rounded": 11, "multinuclei": 4, "nucleus_fragmented": 3, "movility_nucleus_mobile": 5, "mitocohondria_grouped": 22, "movility_surface_mobile": 5, "release_yes": 26, "task": "http://societic.ibercivis.es/pybossa/static/input/feb2013-10-19", "movility_surface_null": 32, "cells_dead": 7, "cellular_shape_rounded": 5, "nucleus_rounded": 16}, "user_id": 336, "task_id": 9264, "created": "2013-04-18T11:55:23.663656", "finish_time": "2013-04-18T11:55:23.663684", "calibration": null, "app_id": 421, "user_ip": null, "timeout": null, "id": 3150}]

At the time of this deliverable creation, the percentage of tasks done is 6.06% what means around 3697 tasks completed (0.0606*4067*15). According to PyBossa statistics, most of the users were authenticated before using the application as it is depicted in the graph below. Only 11% of the users were anonymous versus the almost 89% of the authenticated users. That unexpected behaviour seems to be due to the schools participation during the initial phase since students had to signed up. And also, the absence of the international and official launch keeps application not so widespread.



Authentication Statistics

In the following graph it is possible to observe the distribution of the answers over the time. Distribution shape makes sense keeping in mind that most of the users of the initial phase were students. Therefore, main peaks (amount of tasks done) are between May and June, becoming almost null the amount of tasks done since the end of June (end of the school period).



Answer distribution over time

In what concerns to the questionnaires, it is considered that there are not enough samples to draw conclusions.

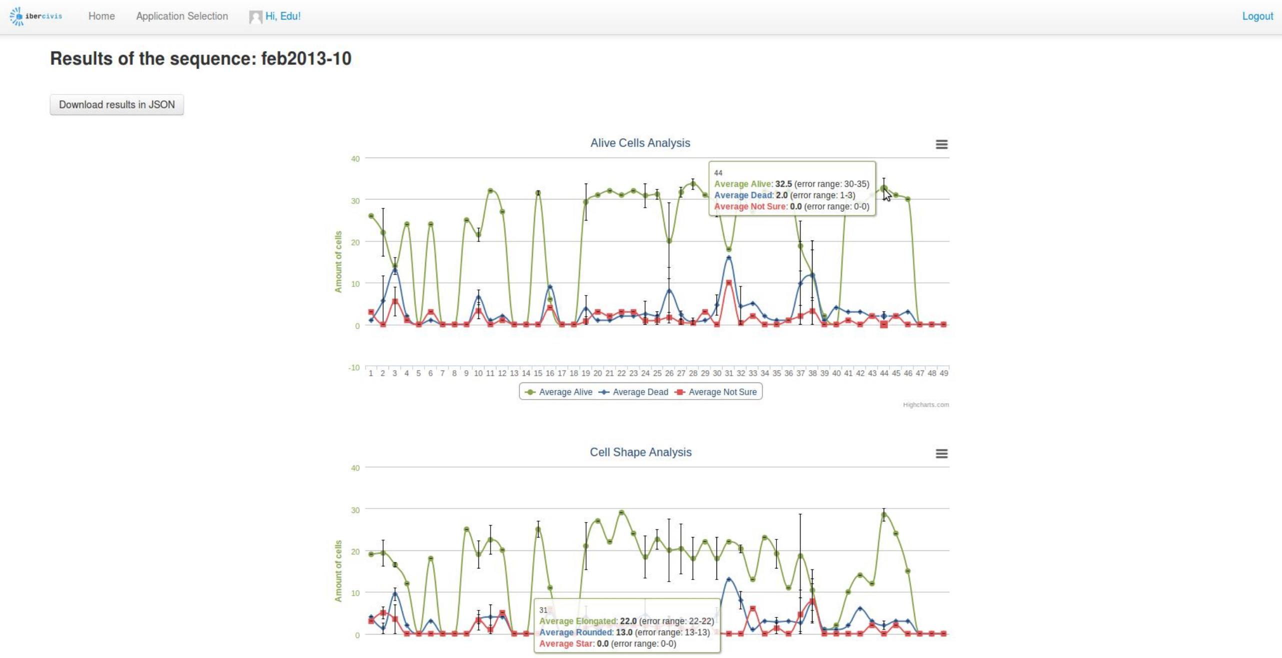
8.1. Result's presentation

Aiming to ease results' analysis to the researcher a site has been developed. Results can be download in the form above displayed. However, those formats (JSON and CSV) might not be comfortable to work with for the researcher who has to invest time to process the results. Therefore, since result data sets generated as a consequence of the use of the application are not easy to handle, a tool was considered necessary.

A generic site has been built to manage the results of several Citizen Science applications by the researchers. A functional version has been developed, but it is not public yet, since design and look&feel aspects have still been left aside. At the moment, authentication is done through openID what means that researcher must have some account linked to that mechanism like for instance a Gmail, Google account. Researchers are granted only with access to the applications they manage. In addition, the idea behind this site is to make public-accessible results for those researchers who are willing to do so, but keep them private to those who do not. Currently, all results are private, but in the foreseen work this will change. PyBossa instead, does not currently allow to make results private.

Site has been developed combining Flask (a micro-framework for Python) and bootstrap (a powerful front-end framework for faster and easier web development) working on a local virtual machine managed by Vagrant (an open-source software for creating and configuring virtual development environments).

In what concerns to the cells project, results visualization and election have been customized. Since every sequence is divided in 49 frames that become the tasks which are replicated 15 times each. Thus, with the goal to summarize, ease and speed up result visualization, tasks are grouped by sequences so in task selection only sequences are displayed. Once one of them is selected, results for that sequence are shown in the screen. Four of the parameters have been selected to appear in the results. These are: cells alive/dead, cells shape, cell content release and mitochondria distribution. Results for these parameters are depicted in graphs one per parameter. In the x axis the time line is presented divided in 49 slots, while the y axis describes the amount of cells. Average of results for every option of the parameter are displayed as the lines in the graphs. For each value in the time line, the standard deviation is represented as a black vertical line. On hover over the points along the line, statistics for the option of the parameter at the corresponding slot are shown. A screen shot for illustrating this paragraph is provided at the following figure.



Results Site Graphs

Besides displaying these graphs, the researcher can still download the raw results data set in JSON format either for the sequence at hand in the graph page or for the whole tasks in the task selection page. At the bottom, below the four graphs, remarks and comments done by the users for the current sequence are also displayed as depicted in the next figure.

Results site users' remarks

9. Publications

As a result of the work done, two publications were submitted and accepted in an international conference and an international workshop.

- *Cell Images Analysis as a Case of Citizen Science for Advanced Education: Laboratory and School, Back and Forth.*
  E. Lostal Lanza, F. Serrano Sanz, J. A. Carrodeguas Villar, P. Martínez Alonso, F. Sanz García, C. Val Gascón
  In Proceedings of the 7th International Technology, Education and Development Conference (INTED 2013), Valencia, Spain, 2013, IATED

- *A case of Citizen Science for Cell Biology Images Analysis.*
  Eduardo Lostal Lanza, Fermín Serrano Sanz, Carlos Val Gascón, Francisco Sanz García, Patricia Martínez Alonso, J. A. Carrodeguas Villar
  Accepted for VII e-Science Workshop of the Brazilian Computing Society, Maceio, Brazil, 2013

The work was also presented at the VI National Conference BIFI 2013 by Eduardo Lostal under the title “Citizen Science for the analysis of microscopy images in cell models of disease”.

10. Future Work

In the following, future work lines at the time of document creation are presented.

- Given the feedback received, work in a new version. That version must be simpler with less steps and quicker to analyze. Provide a way to do the analysis graphically is one of the prioritary improvements to be done.
- Create a result repository in order to ease the query of the data result to the researcher.
- Improve the way in that tasks are created and input is provided.
- Keep working on the software tool for automatic analysis. On parallel, a software tool is being developed for an automatic analysis of those photographies. Through the datasets provided by citizen science, that is through the application developed, a machine learning software can train the software to get a proper analysis of the pictures taken by the microscopy.
- Improve results site, more features, look&feel, etc.

11. Conclusion

Application was developed with the whole functionalities and requirements of the researcher and is in production since late March 2013. It is accessible from either PyBossa production server at Ibercivis facilities or Socientize webpage:

- http://pybossa.socientize.eu/pybossa/app/cellsimages/newtask
- http://socientize.eu/?q=eu/node/382

Application is fully operative and working properly with a very attractive design. However, received feedback indicates that the application is too complex for the general public, what must be ammended in the next version.

No further dissemination or publicity has been done waiting for the international launch. Anyhow, during the first three months of the project, volunteers have completed almost three thousand tasks. That is a satisfactory result for the initial phase. In addition, the work has been accepted for publication in two conferences and presented in another one.