


	<p>OPAALS PROJECT</p> <p>Contract n° IST-034824</p>
---	--

Workpackage 9
Communication and Dissemination

Deliverable 9.6.1
Cooperative Article Publication Tool (version 1)

	<p>Project funded by the European Community under the "Information Society Technology" Programme</p>
---	--

Contract Number: IST-034824

Project Acronym: OPAALS

Deliverable N°: 9.6

Due date: January 2009

Delivery Date: March 2009

Short Description: This deliverable reports the implementation process of a cooperative article publication system which aims to facilitate users with writing of scientific papers. The idea about how this system should be has changed and now we look for having a system that integrates functionalities for cooperative editing, communicating and direct submitting documents for publishing. An extra time is required to implement and integrate all those functionalities so that we suggested to split the deliverable in two parts. The document here presents what had been done in the first part, showing that a minimum list of requirements are implemented and ready for tests by OPAALS users. The document also present a roadmap for the forthcoming work so that we can deliver the full desired system.

Author: IPTI (Paulo Renato Campos de Siqueira and Saulo Faria Almeida Barretto)

Partners contributed: IPTI

Made available to: OPAALS Consortium and European Commission

Versioning

Version	Date	Name, organization
0.5	15/03/09	IPTI
1.0	20/03/09	IPTI

Quality check

Internal Reviewers: Gerard Briscoe, LSE

Sotiris Moschoyiannis, Surrey

Dependencies:

Achievements*	<p>A first version of a system to allow users to write papers collaboratively was implemented. This version offers a minimum set of functionalities like editing, sharing with other users, visualize the document and comment.</p> <p>Two other tools were also implemented and now are planned to be integrated. A conference system that will users to communicate while they edit a document and a working paper system that will allow users to direct submitt their documents for publishing.</p> <p>A second version of D9.6 is suggested. This version will contains further functionalities for helping users when writing a document (simultaneous editing, control of versions) and also be deliverade as an integrated system (editing, communicate and submitt).</p>
Work Packages	<p>This deliverable will directly benefit the following WPs:</p> <ul style="list-style-type: none"> ●WP 9 (Communication and Dissemination) – the system will improve the communication and collaboration activities. ●WP 10 (Sustainable Community Building) – it will facilitate the process of building a sustainable network community by facilitating the communication and collaboration between the users.
Partners	All partners could benefit reading this deliverable as it is related to a system that will be used by all partners to edit documents in a collaborative way
Domains	All
Targets	Researchers who wants to edit documents (e.g. Deliverables) in a collaborative and interactive way.
Publications*	No work published
PhD Students*	No PhD students involved
Outstanding features*	The integration between an editing tool and a communication (chat and VoIP) tool, as well as the possibility of directly submitting the document, when ready, for a review and publishing process. Another important feature is the simultaneous editing which will allow users to edit at the same time but avoiding to write over the same part of the text.
Disciplinary domains of authors*	<p>Paulo Renato Campos de Siqueira – computer science</p> <p>Saulo Faria Almeida Barretto – civil engineer</p>



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 License. To view a copy of this license, visit: <http://creativecommons.org/licenses/by-nc-sa/3.0/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

1 Introduction

The aim of this deliverable is to report the implementation process of a cooperative article publication system. The idea about what such a system should be has changed in the last months due to two main reasons. First, the fact that the idea of having an online journal has changed into a less ambitious proposal of having a system to support a working paper series, as the challenge for building a scientific online journal is huge and the consortium agreed this plan should be postponed. This new perspective for the publishing system was detailed reported in D9.8. Second, because IPTI has implemented two tools that brings a great potential to enhance the atmosphere of sharing and cooperation. Those tools are the document publishing system (working paper series), based on the functionalities described in D9.8, and an online conference tool (chat and VoIP).

The central point of the cooperative article publication system still remains the same, a system to facilitate users with writing of scientific papers, but now integrated with a communication system and with a publishing system. This will allow users to communicate while they edit their common documents as well as directly publish them in the working paper series tool, by exporting the paper's final version in a PDF file format. This integrated and interactive system will run on top of the DE architecture which includes the peer-to-peer (p2p) layer, considered essential to make the functionality for simultaneous editing and VoIP more efficient.

Taking into account that the deliverables produced by OPAALS require peer reviews, but papers from the outside world might relax a little bit on this restriction, some functionalities that would normally be more suited to an Open Journal concept than to a Working Paper

concept are present, but optional. This way, we can help OPAALS with its deliverable handling, and be useful to other parties as well. The review process, and the change from an Online Journal to a Working Paper has been more deeply investigated in D9.8, like mentioned before.

Due the decision of having a more complex system the implementation process will require more time then previously planned. But it is important to highlight that a version with minimum functionalities for the cooperative article publication system is ready for use and this extra time required will be used to implement some functionalities and integrate this system with the two mentioned above.

This document presents what has been done to achieve the desired system starting by an overview of the DE architecture and the P2P layer. Following this overview we present a detailed description about what was implemented, how it can be used and a roadmap related to the forthcoming work to conclude the system.

2 DE Architecture Overview

A Digital Ecosystem is an environment where services can be made available and consumed in a way determined by the DE itself. In our scenario, this is being defined by OPAALS. Our services should run on top of a fully distributed peer-to-peer network, and have no single points of failure or control. The peer-to-peer network is being implemented based on the P2P network design done by SURREY (WP3).

In summary, this peer-to-peer layer contains two major features: the transaction support (coming in the form of temporary networks (VPTNs) from which Dynamic Virtual Super Peers (DVSPs), emerge (Work Package 3, Deliverable D3.2)), which will allow the tools to exchange messages and work together – executing transactions - in a peer-to-peer fashion, and the distributed storage and database. As mentioned before, the DVSPs are already being implemented, and will be available for usage by the DE elements soon. But the distributed

storage and database is still to be defined, and it is part of the Work Plan for the project's Phase III.

The DE will provide the infrastructure for running the tools we already have in place, which were briefly mentioned in the introduction and will be more detailed later. But which the DE, we will be able to make these tools run according to the OPAALS goals.

3 Workflow for the Article Publication

The process starts with the collaborative document editing tool, which allows users to create documents together, editing their contents simultaneously. This tool will also provide a history of the versions of the documents being created by the community, so that they can see how the document evolves and, if necessary, roll back to older versions.

To assist the users' communications during the creation process, the conference tool will have an important role as well, allowing users to exchange text and voice messages (among other things) in real time, while editing their documents as described above.

At the end of the chain there is the Working Paper Series tool itself, our article publishing system. This is where a document can be sent to be publicly available as a paper, when it is ready. Once sent, the paper will go through a peer review process (described in D9.8), allowing the community itself to handle the evolution of what is available and what isn't, in the environment.

The following sections will describe each one of these tools in more detail, as well as a little bit of the peer-to-peer infrastructure layer.

4 Peer-to-Peer (P2P)

The peer-to-peer layer allows the community to control the future of the system's environment. This is so that no single node is responsible for the infrastructure of the platform or how it runs, but the group of nodes as a whole is.

When this layer gets completely implemented, our scenario will be something like what is described in the next paragraphs, where we basically recap the workflow described previously, but adding a few details regarding the peer-to-peer layer.

First, the editing process will happen directly between the users editing a specific document, without any central server. The nodes will coordinate themselves to make the collaborative edition work, and making sure everyone's contribution is taken into account without losing anything, as well as avoiding people to edit exactly the same sentence or line of the document at the same time as this would result in inconsistencies (conflicting versions of the document).

During the whole editing process, the users will be able to communicate with each other using the conference tool. This tool will be also running on top of the p2p network. This means that everything, from basic text chat communication to proper document editing, will happen between the users without a central server nor dependency on a specific provider for this service; only the structure provided by the p2p network will be used.

When the document gets ready to be published as a paper in the Working Paper Series environment, this will also happen directly in the p2p network environment, and the paper will go through the review process. The editors' and reviewers' nodes will be able to communicate directly at this point, and will ultimately be responsible for the publication process to happen. After being approved, the paper will be made available in the distributed storage, in the p2p network.

At this time, we have most of what is described above implemented, but in a centralized

architecture. The implementation of the p2p layer has started, but it is not ready yet to be used by the tools mentioned above, hence why they are still centralized.

The plan about how the implementation will go forward, and how it will be added to the various tools described here, will be explained in greater detail in the Roadmap section. In what follows we provide a quick overview.

The first thing to note is that the p2p layer being implemented is not something that is coming out of the blue – it is the P2P network design in the core DE architecture. We are in fact implementing the Dynamic Virtual Super Peer (DVSP) model from SURREY (WP 3). The implementation work has been broken in a few logical blocks, and they are being implemented one by one. Parallel to this implementation work, we will start to use this framework being developed in the conference tool, so that we can have proper results while being able to better understand what this infrastructure framework must provide in terms of functionality to the services that will run on top of it.

5 Current Stage of Development

As mentioned in the introduction, two tools have been implemented by IPTI previously, which are the document publishing system and the conference tool. Now we are developing a document editing tool, which allow us to have the workflow defined before.

The document editing tool is the starting point of the workflow, and it is one of the tools which is already ready to use. It only lacks a few features, which are planned and pointed in the roadmap. Right now, we have basic editing capabilities, access control, and the possibility to invite other users to edit the document collaboratively.

Next is the conference tool, which can already be used as well. It lacks some testing and bug fixing only, but all features are already in place, including text messaging, voice calls and a whiteboard.

Finally, there is the Work Paper Series tool. It is also ready for usage, and include features like sending papers to be published, and a implementation of a review process, which is described in D9.8.

The main feature that all of those elements are lacking right now is to have its inner logic working on top of a distributed peer-to-peer network, which is planned to happen soon, as noted in the roadmap section. This will be addressed in more detail later in this document.

6 Tools

As mentioned before the full workflow of the Cooperative Article Publication Tool will be composed by three different tools: Collaborative Document Editing, to allow users to edit documents collectively; Conference, to allow users to participate in a virtual meeting (chat and VoIP) while working on a given document; Working Paper Series, to allow users to publish documents and spread knowledge. These systems descriptions and more detailed instructions are defined in the next topics.

6.1 Document Editing Tool

In the main page, users can see the general information about documents being created (Figure 1). In the upper left side, a counter of all existing documents, and a counter of all documents that the user has permission to collaborate to is displayed. Bellow this section, there is the “Featured Docs” one. There, the system shows the most accessed documents. The single most accessed document can be seen in the middle of the page. On the right side there are lists with the last documents created, the documents that the user contributed to recently and the documents that the friends of the user contributed to. There is also a “Create DOC” button, which the user can press to create a new document.

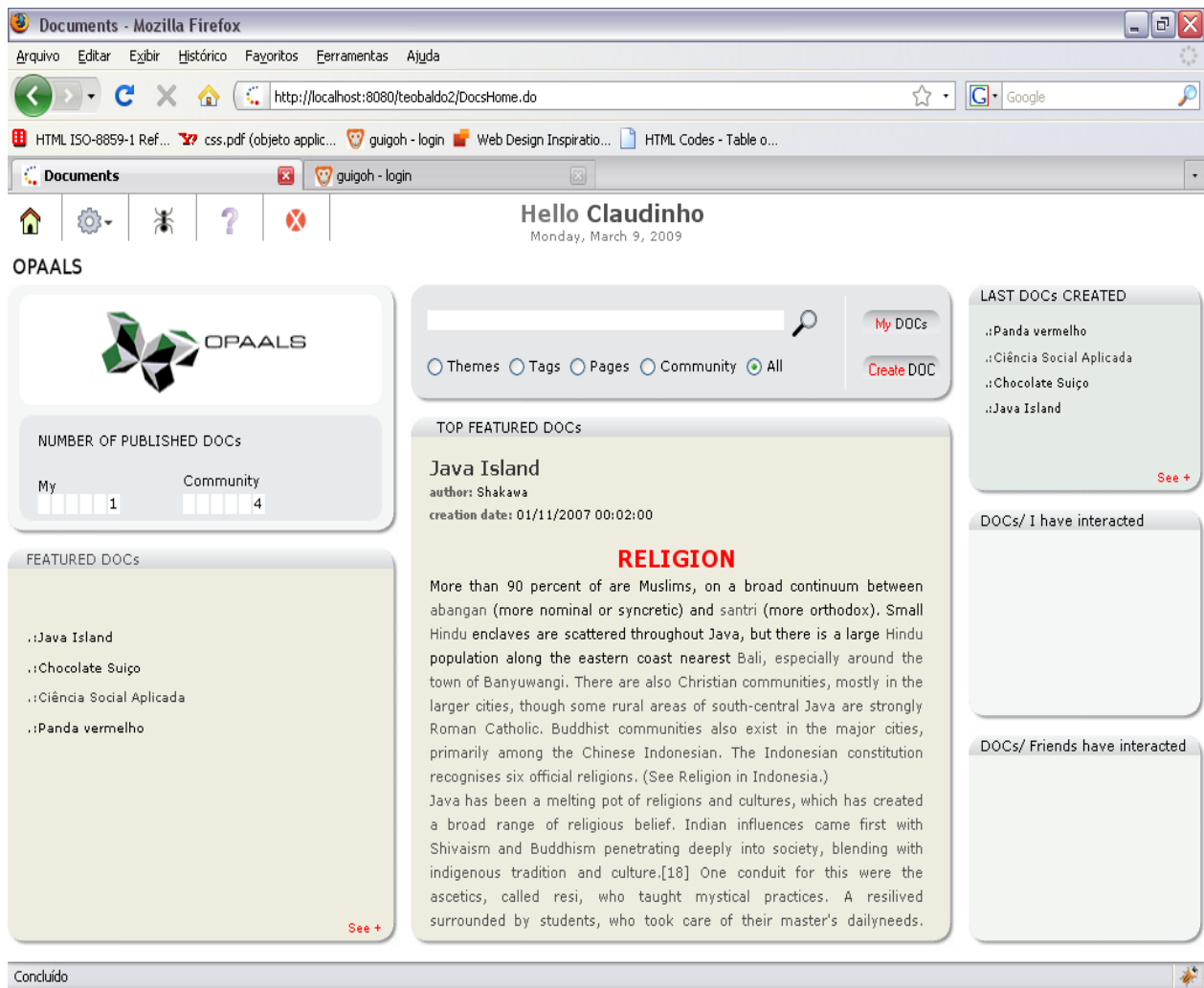


Figure 1. Main page, with lists of documents being created

If the user clicks the button “My Docs”, another page is displayed (Figure 2). In the same fashion, the user can see a counter of the contributions and posts made, all the documents being created, the most accessed documents and finally the comments for the user’s documents.

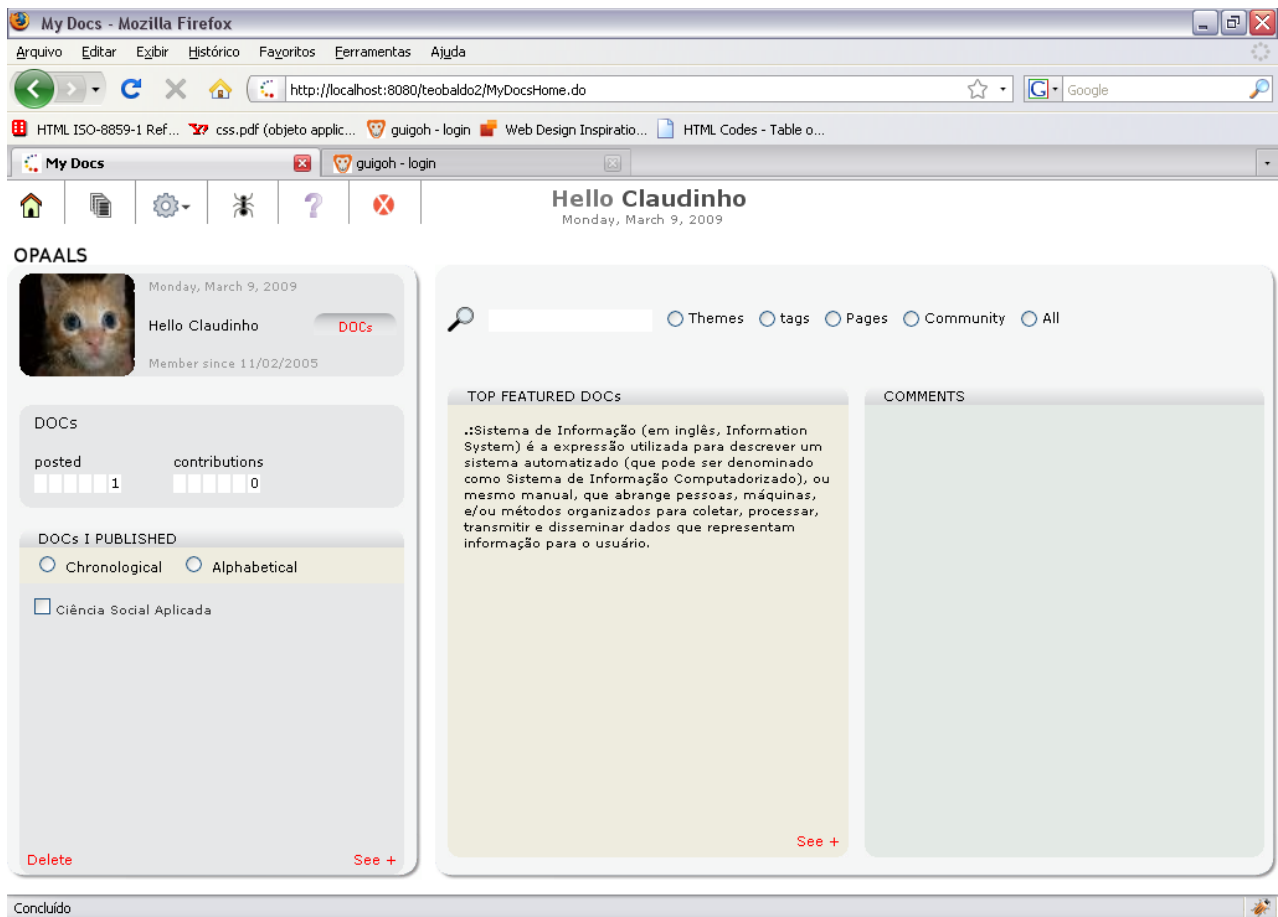


Figure 2. User's documents list page

The document editing page contains a left section with the document information, and a main section, with tabs for editing, viewing, and sharing the document. In the information section, the document title, a file chooser to select an image that illustrates the document, the creation date, the author's name, the abstract of the document, and the tags related to that document can be seen.

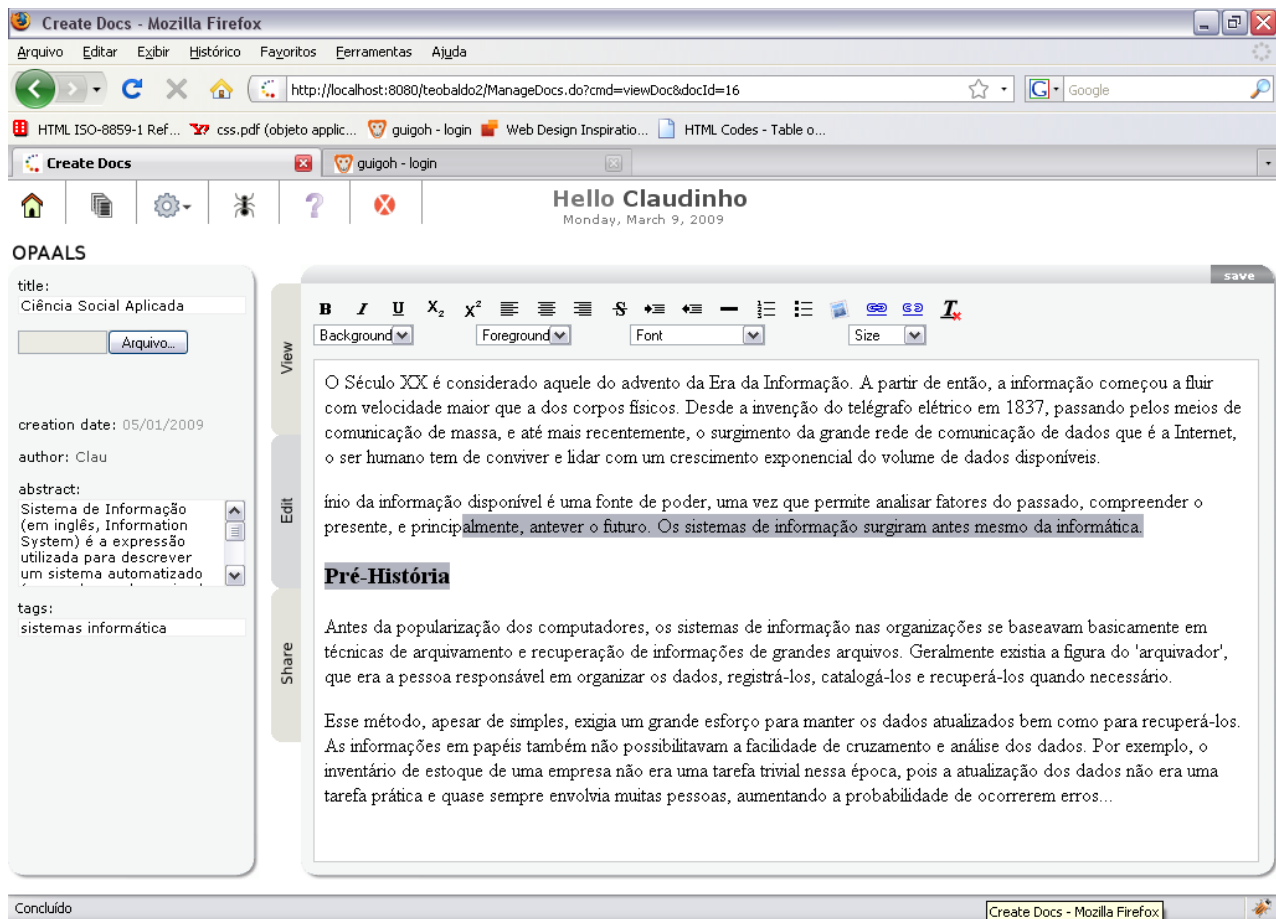


Figure 3. Document editing main page (edit tab)

The edit tab includes the document contents and a toolbar. The author can save the document anytime by clicking the “save” button and also view (Figure 4) how the document will be presented to others by clicking the view tab.

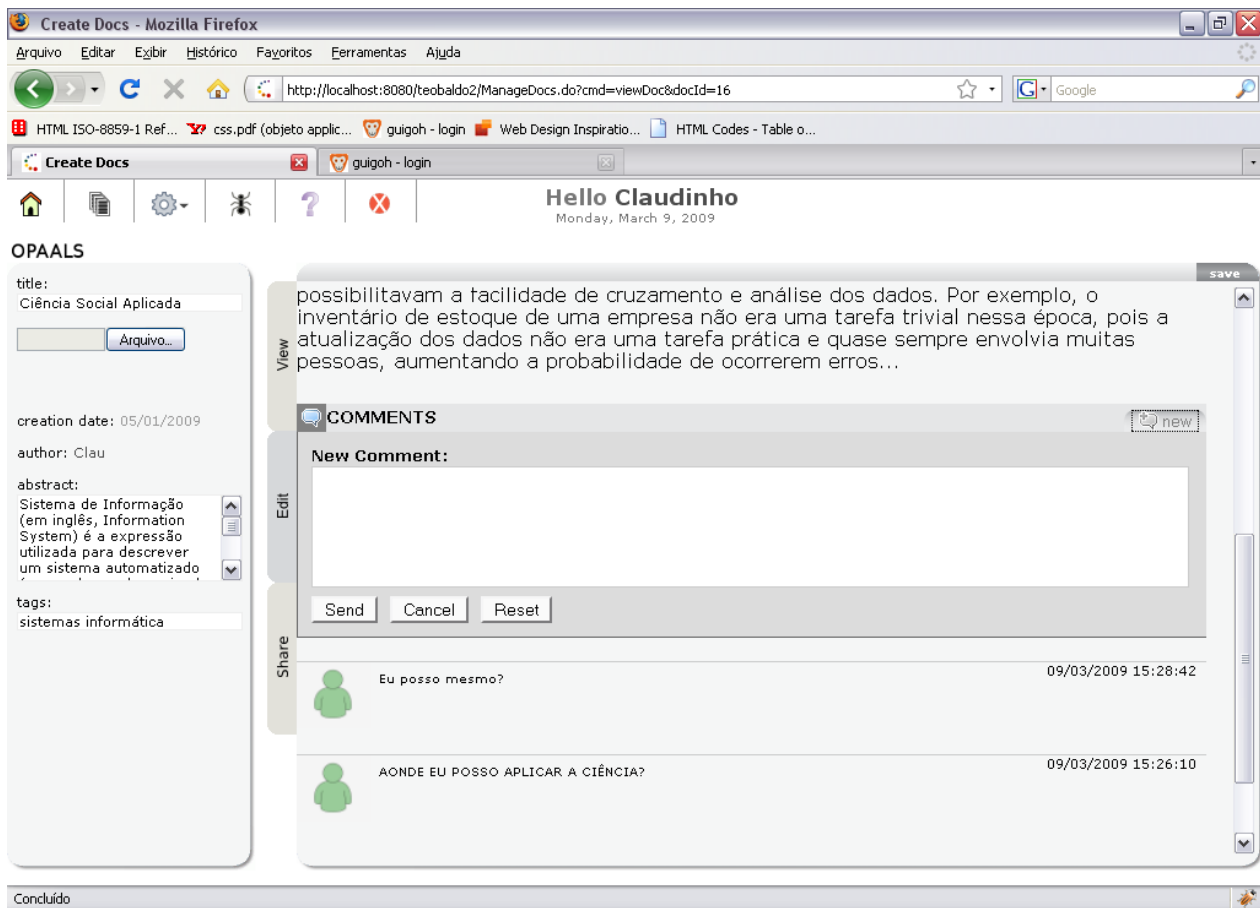


Figure 4. Document editing main page (view tab)

The view tab shows the document as the everyone else will see it, as well as the comments added to that document. The share tab (Figure 5) contains the sharing management function, allowing the user to determine whether the document can be viewed by everyone or not; whether all users can contribute to the document; and a collaborators (contributors) list, to allow the user to choose who will have permission to contribute if the document cannot be edited by everyone.

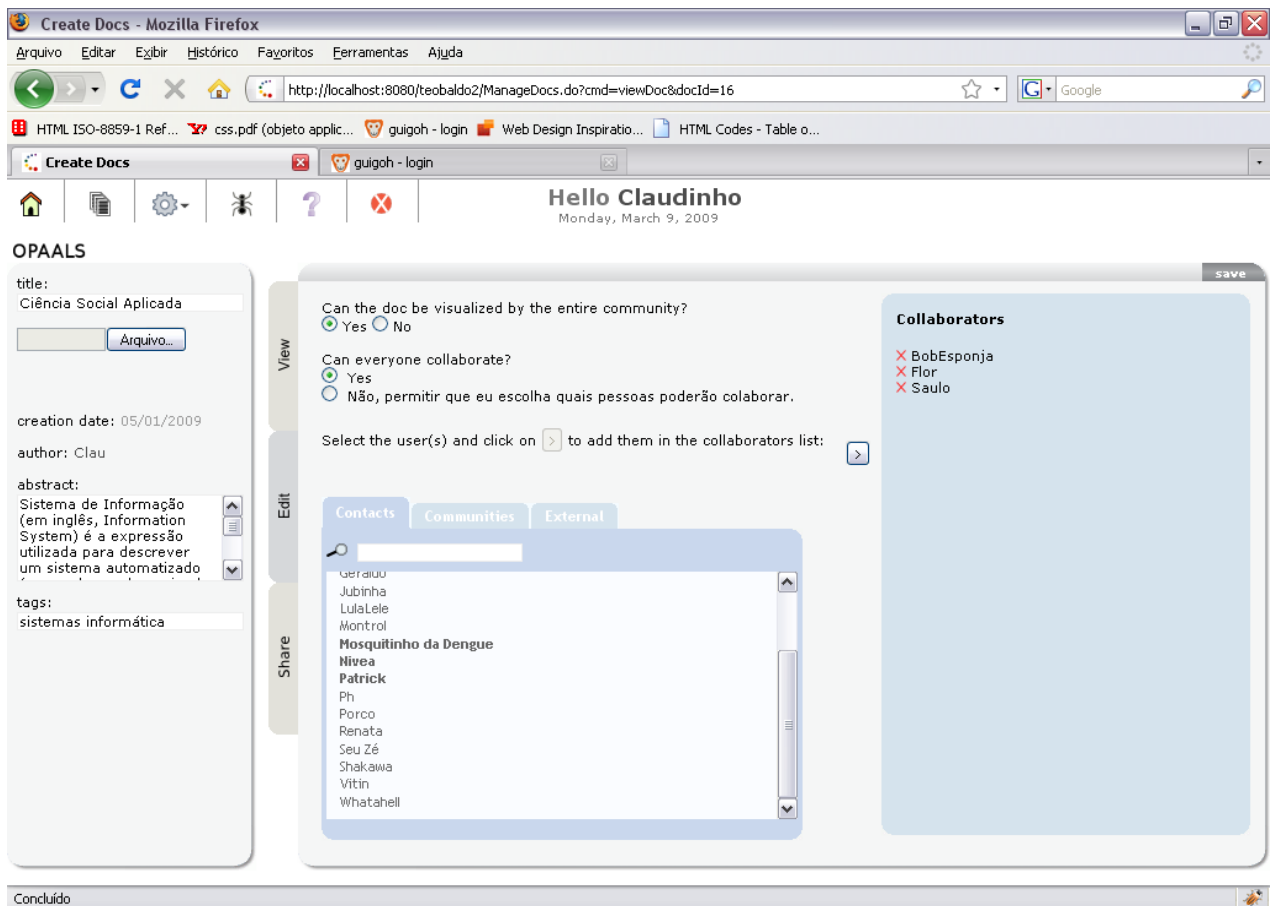


Figure 5. Document editing main page (share tab)

6.2 Conference

The conference tool allows users to have conferences with many features such as participant management, written and voice chat and whiteboard. The system will be launched as an open source project in the near future. When the user enters the conference's main page (Figure 6), the system shows a list of conferences available and a list of conferences he participated before in the bottom.

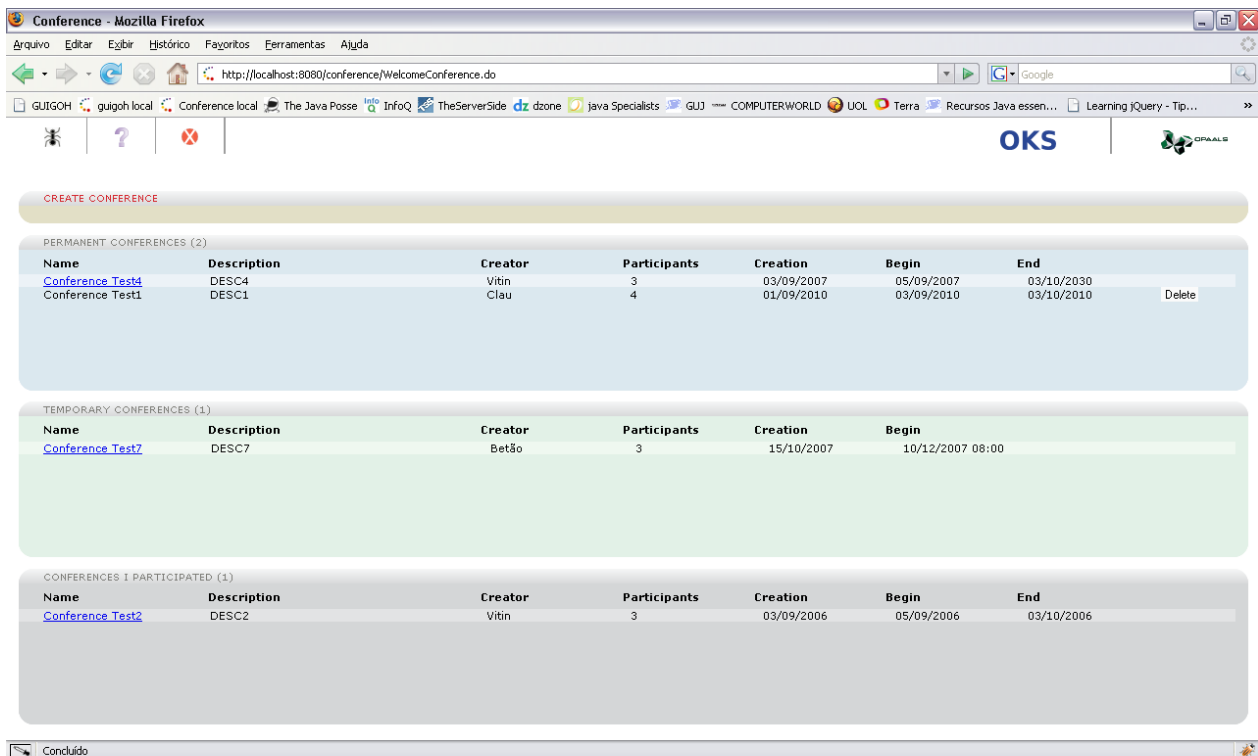


Figure 6. Conference tool main page

In the conference lists there is the conference name, description, owner name, number of participants, creation date, start and end date. If the user is the owner of a conference, the button “Delete” will be visible, allowing him to remove the whole conference. The current running conferences have a link that allows the user to access it and the finished ones have a link allowing access to the chat log of that conference (Figure 7). The first section of the page is used to create a new conference.

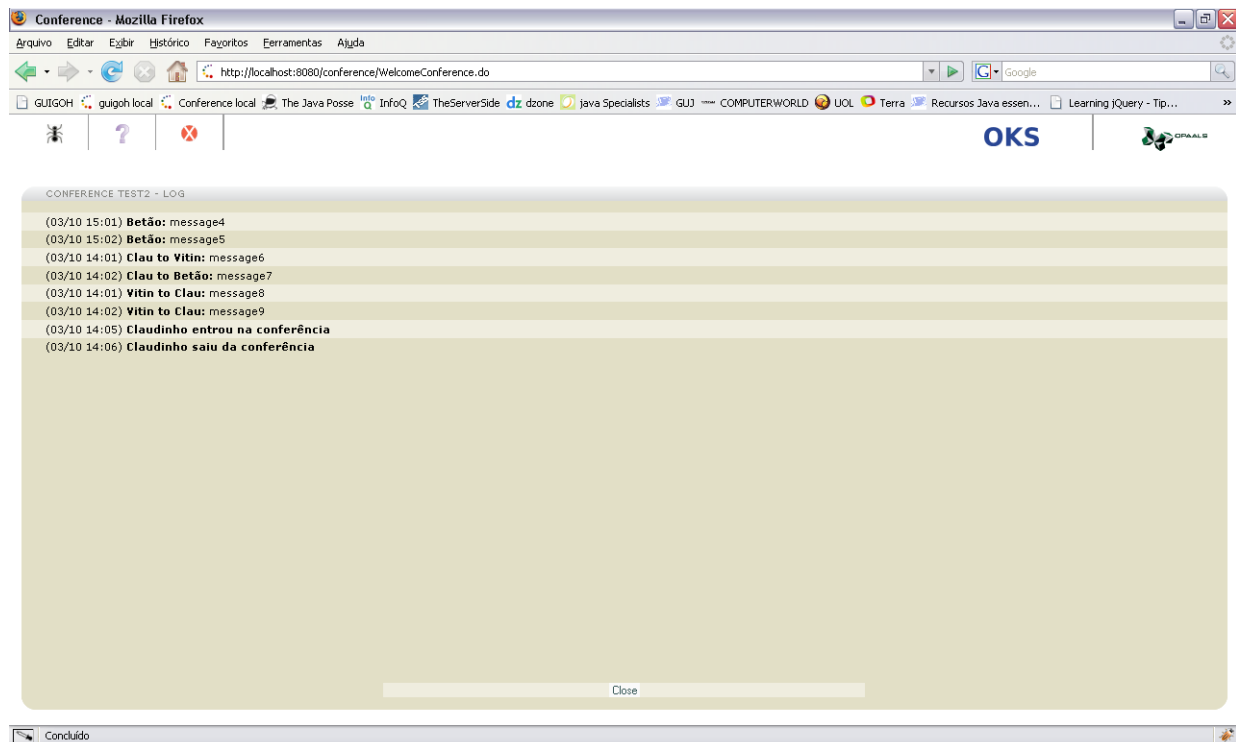


Figure 7. Conference log

If the user wants to create a conference, she can click on the label “Create Conference” and the whole section is maximized without reload time (Figure 8). This section contains the fields name, description and tags pertaining to the conference. Some conference settings can also be configured, such as start and end date (selected by a calendar menu when the user clicks the date box) and the size of the whiteboard of the conference. In the bottom of the page, the conference creator can select the users that will participate in the conference. When a participant is selected, the user’s name goes to the bottom right “Participants” section. After saving the conference, the user returns to the conference main page without any reload time.

Conference - Mozilla Firefox

Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda

http://localhost:8080/conference/WelcomeConference.do

Google

Conference

View Issues - Mantis

OKS **OPAALS**

CREATE CONFERENCE

Name
Opals Features Discussion

Description
Discussion about OKS architecture

Tags
opals oks architecture

Conference type
☒ Permanent
 Start 12/03/2009 dd/mm/yyyy
 Size: Medium

Invite
 Users
 rives
 Patrick
 Ph
 Porco
 Rafael
 Seu Zé
 Shakawa
 Test Man
 Test Woman
 Vitin
 Whatahell

Participants
 Renata
 Saulo
 Zanarias
 Paulinho

Save Cancel

Concluido

Figure 8. Conference creation

To access the conference users must click the conference link in the main page. The conference contains the list of participants, welcome text, mode selection and text chat (Figure 9). In the participants list, the current user is the first on the top. The rest is listed with their status (online – green; offline – gray; away – red). The green telephone button can be used to start a voice conference. The written chat box (yellow box) contains the chat log and the message editing input field, with emoticons.

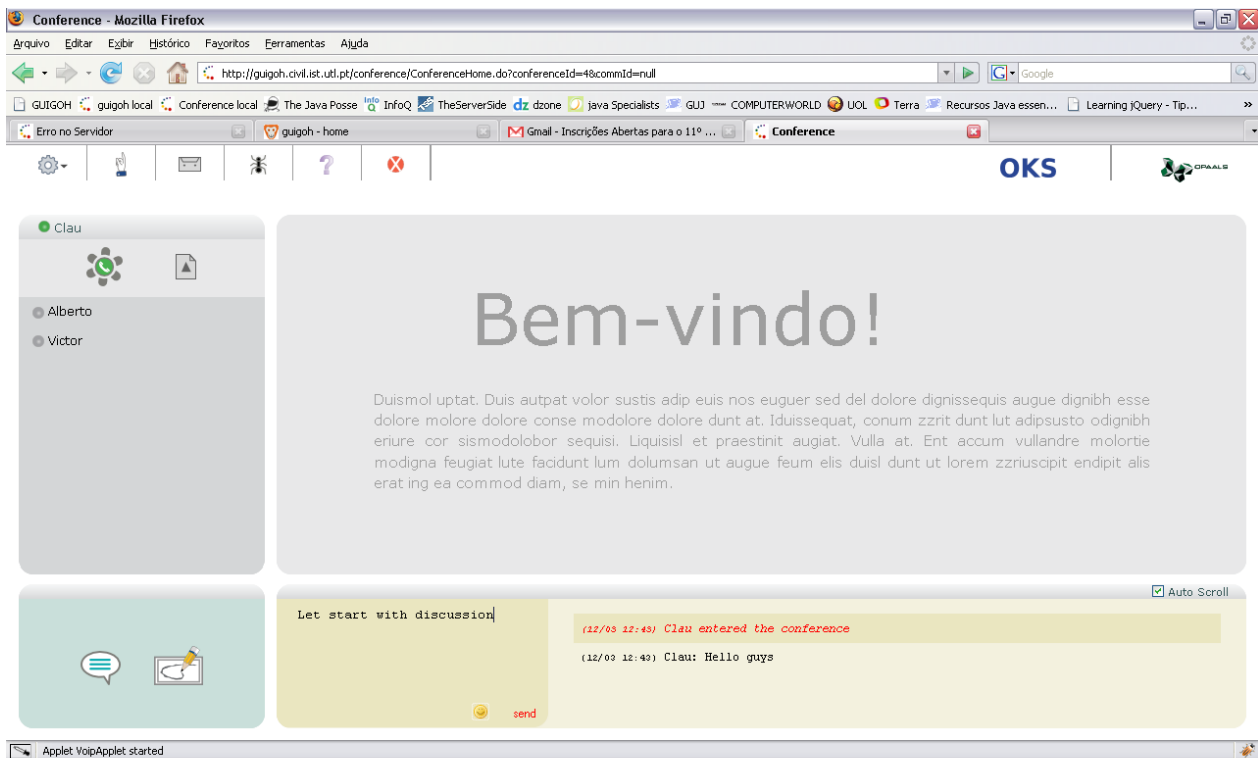


Figure 9. Conference page

At the bottom left is the conference mode selection. A written chat only page can be viewed clicking the first button (chat bubble) (Figure 10).

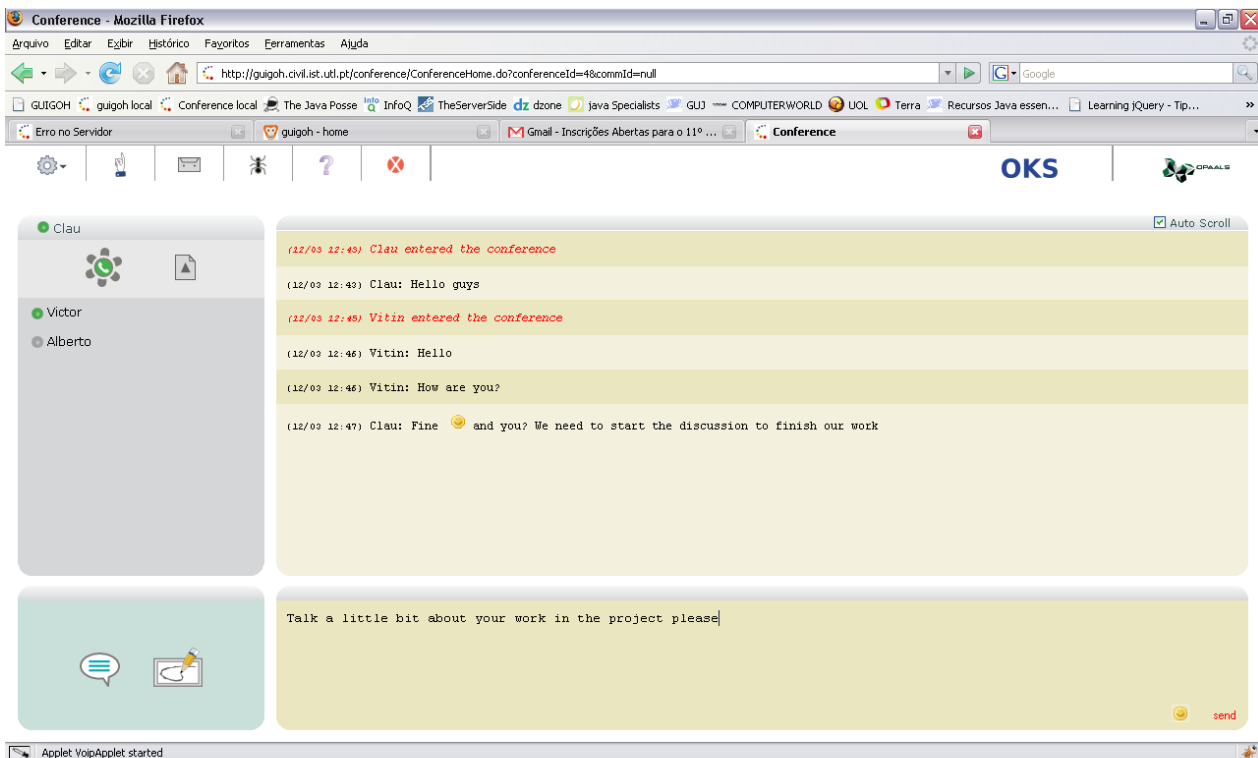


Figure 10. Written chat dedicated mode

When a user is clicked in the list of participants, the user's picture can be visualized and a voice or written private message can be sent (Figure 11). If the user receives a private call, a window will pop to notify it.

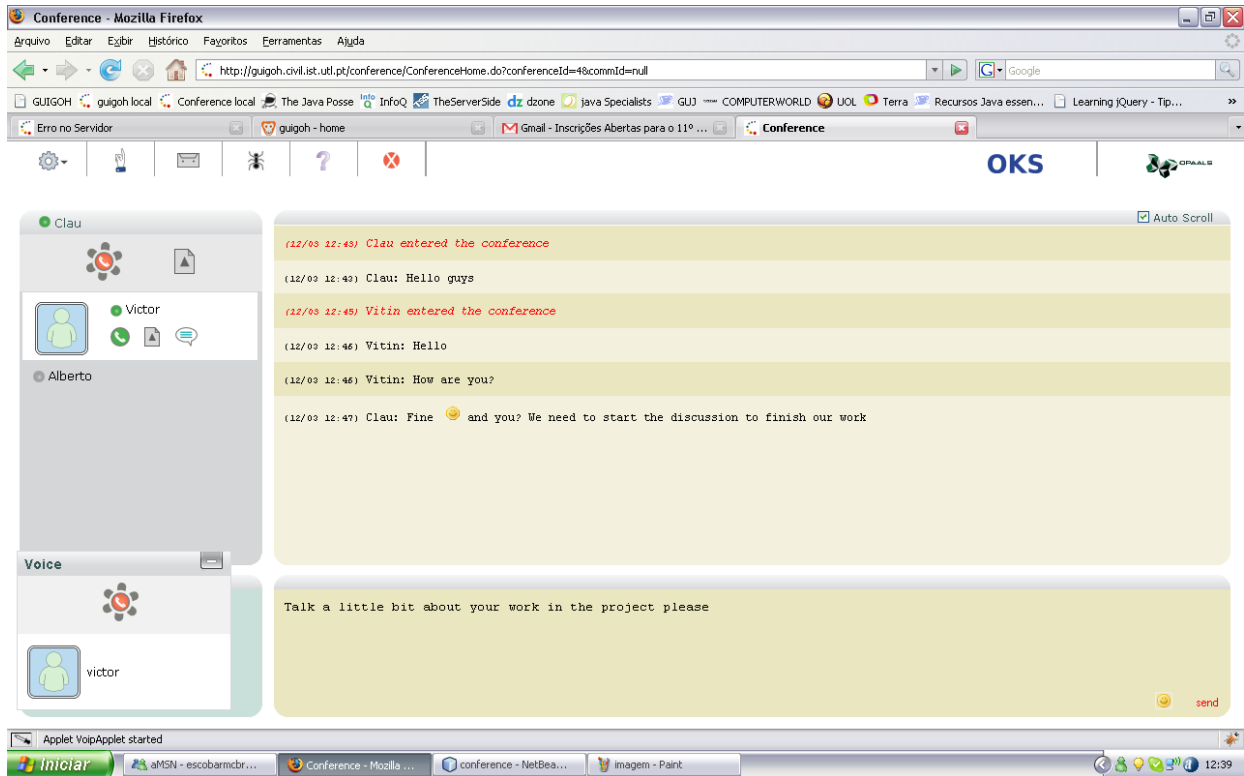


Figure 11. Individual options

The second conference mode is the whiteboard that is selected from the conference mode box. In this mode, users can edit a whiteboard by drawing, writing and including images on a canvas (Figure 12). To edit the canvas, the user must ask for his/her turn by clicking the hand icon in the upper menu. If the hand is marked, it is the user's turn, but if it is not, all the edit buttons are disabled. When a user asks for its turn, the request will go to a queue which order is showed next to the user's name in the participants list.

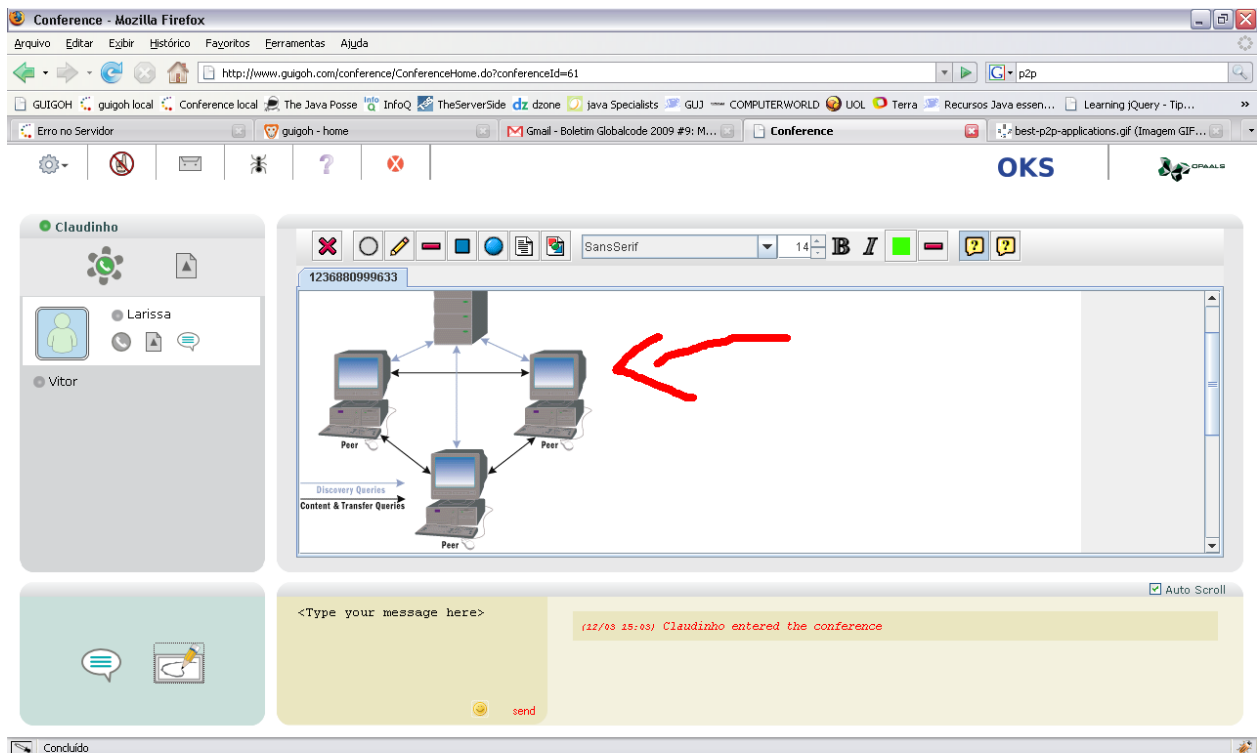


Figure 12. Whiteboard conference mode.

When the upper configuration button is clicked (Figure 13), the user can set the status (online or away) and adjust voice volumes. If the user is the conference owner, he can use the option to “set speaker”, which will hand the turn to any user. That is, it will end the current user’s turn and set a new current editor.

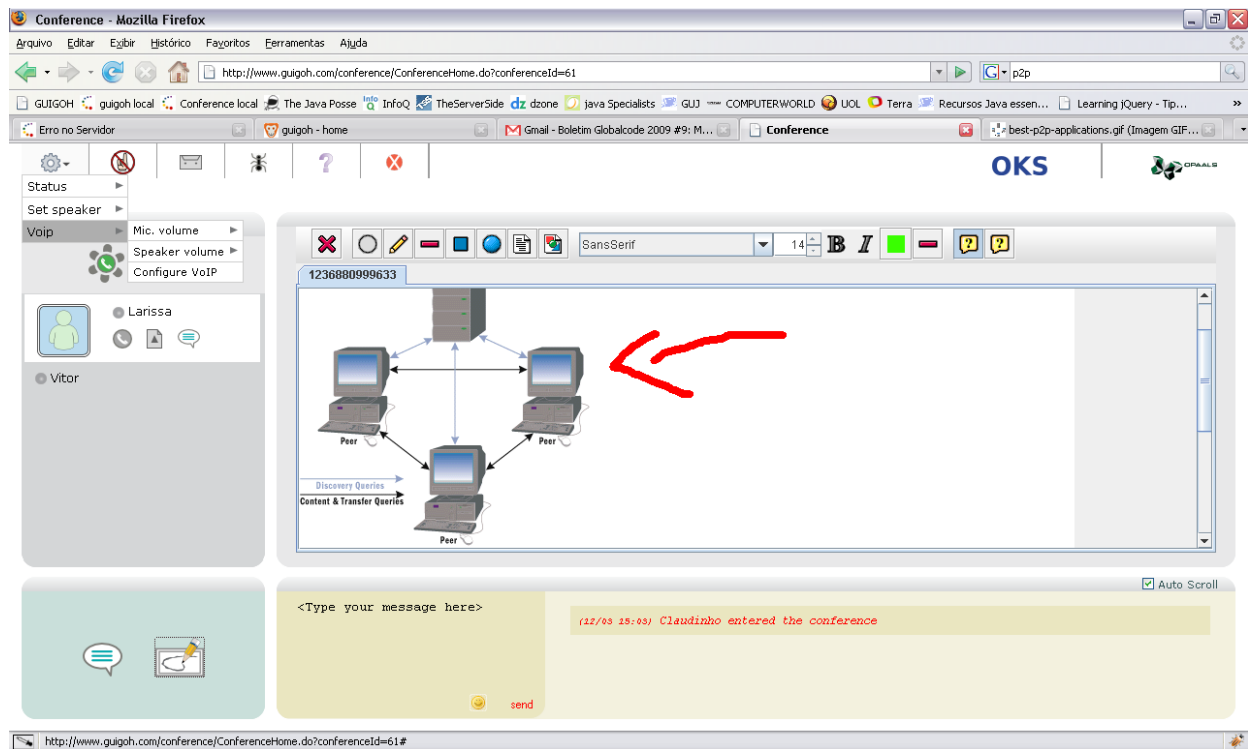


Figure 13. Conference configurations

6.3 Working Paper Series

The Working Paper Series is a system to handle the publishing of papers, and to make those papers go through a formal review process. The Figure 14 shows the tool's login page. As a side note, keep in mind that the title “Digital Ecosystems Open Journal” that appears in the screenshots below are the old name of the “Working Paper Series”. Since the name changed quite recently, the system, and thus the images, couldn't be updated in time for this report.

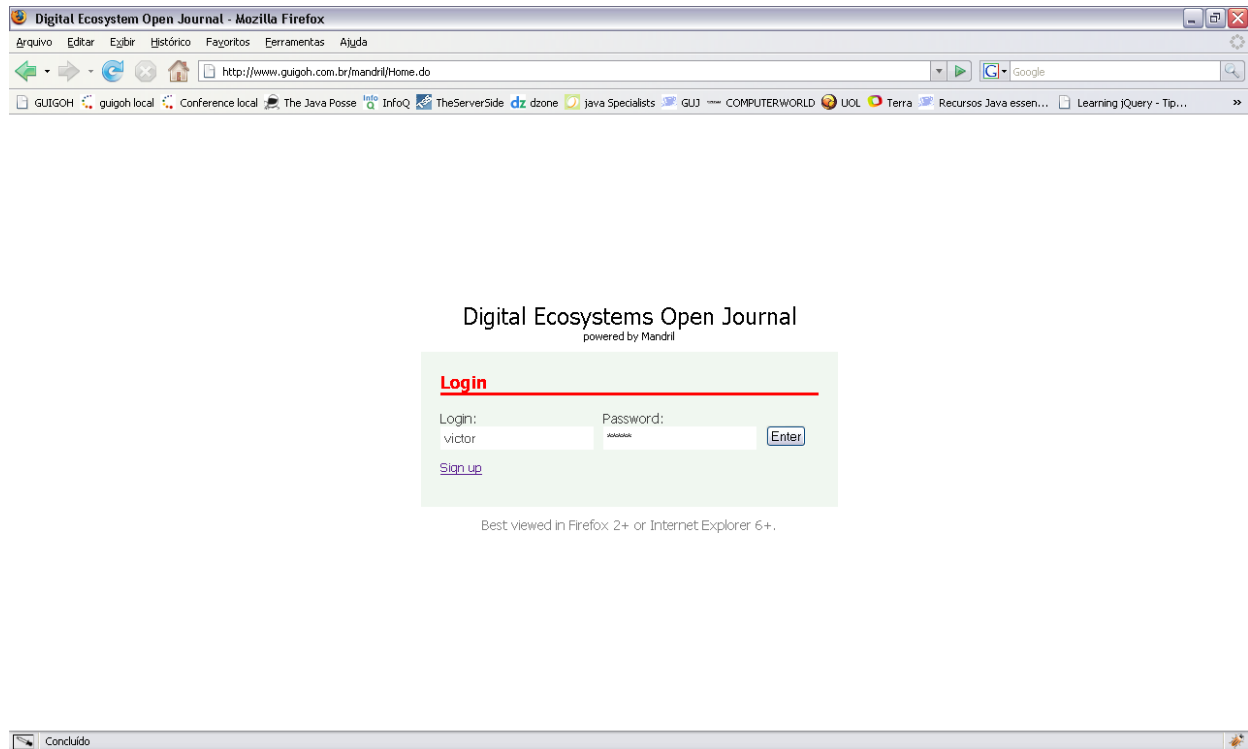


Figure 14. Login page.

In that step, the user sign in using the login name and password. If the author does not have a login yet, she can sign up by clicking the link “Sign up”. The information needed to register is: name, last name, login name, e-mail, password and country (Figure 15). The register fields are being evaluated and might be extended and/or changed.

The screenshot shows a Mozilla Firefox browser window with the title "Digital Ecosystem Open Journal - Mozilla Firefox". The address bar displays the URL "http://www.guigoh.com.br/mandril/RegisterUserHome.do". The page content is a registration form titled "Register user" in red. The form has two columns of input fields. The first column contains fields for "Name:" (filled with "Victor"), "Login" (filled with "victor"), and "Password:" (with a small "adicionar" link below it). The second column contains fields for "Last name:" (filled with "Rocha"), "e-mail" (filled with "victor@ipti.org.br"), and "Confirm password:" (with a small "adicionar" link below it). At the bottom of the form are two buttons: "Back" and "Register". The browser's status bar at the bottom shows a green progress bar and the word "Concluido".

Register user	
Name: Victor	Last name: Rocha
Login victor	e-mail victor@ipti.org.br
Password: <small>adicionar</small>	Confirm password: <small>adicionar</small>
<input type="button" value="Back"/> <input type="button" value="Register"/>	

Figure 15. Registration page.

After logging in the user can access all the published papers in the system (Figure 16). Published papers can be found through the Filter tool, located in the upper side of the page, where any character typed is searched for among the published papers' Titles or Tags. The Featured section (on the right side) shows the most downloaded papers.

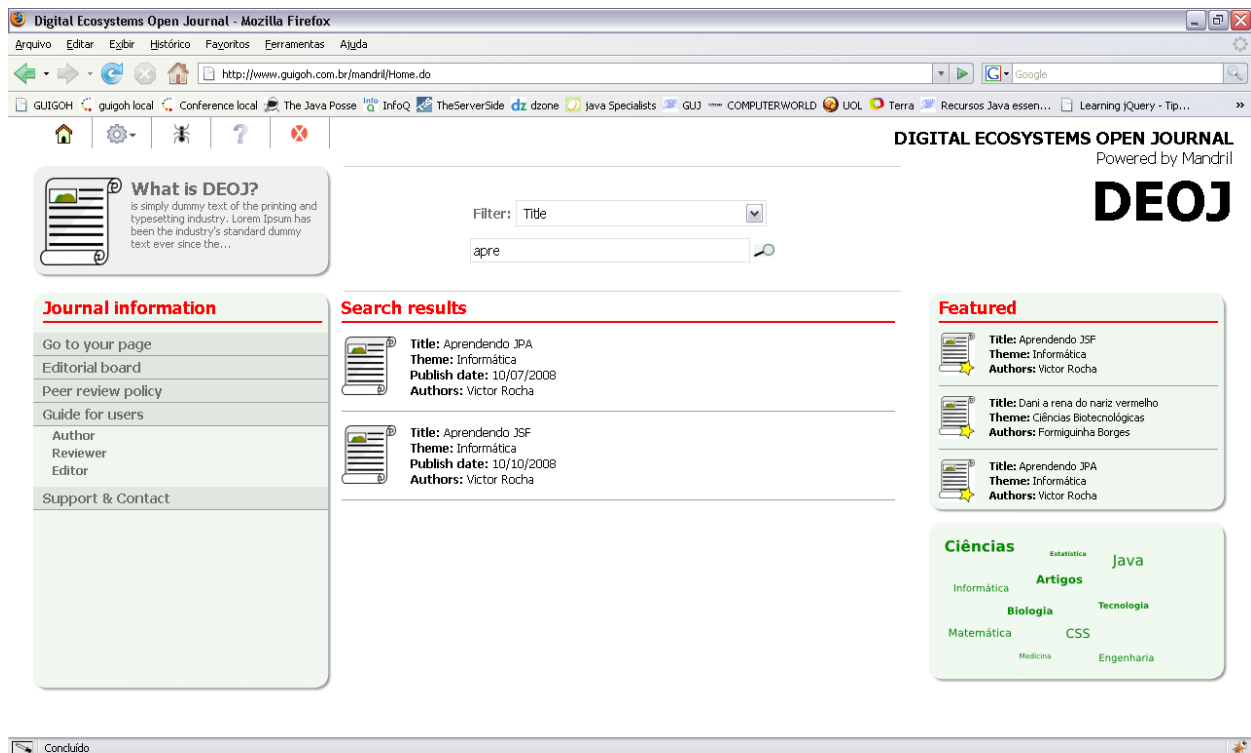


Figure 16. Paper search page.

If a paper is selected, the system presents the title and the publish date, the theme, the abstract and the tags, responsible authors, earlier reviews and a download link to the paper (Figure 17).

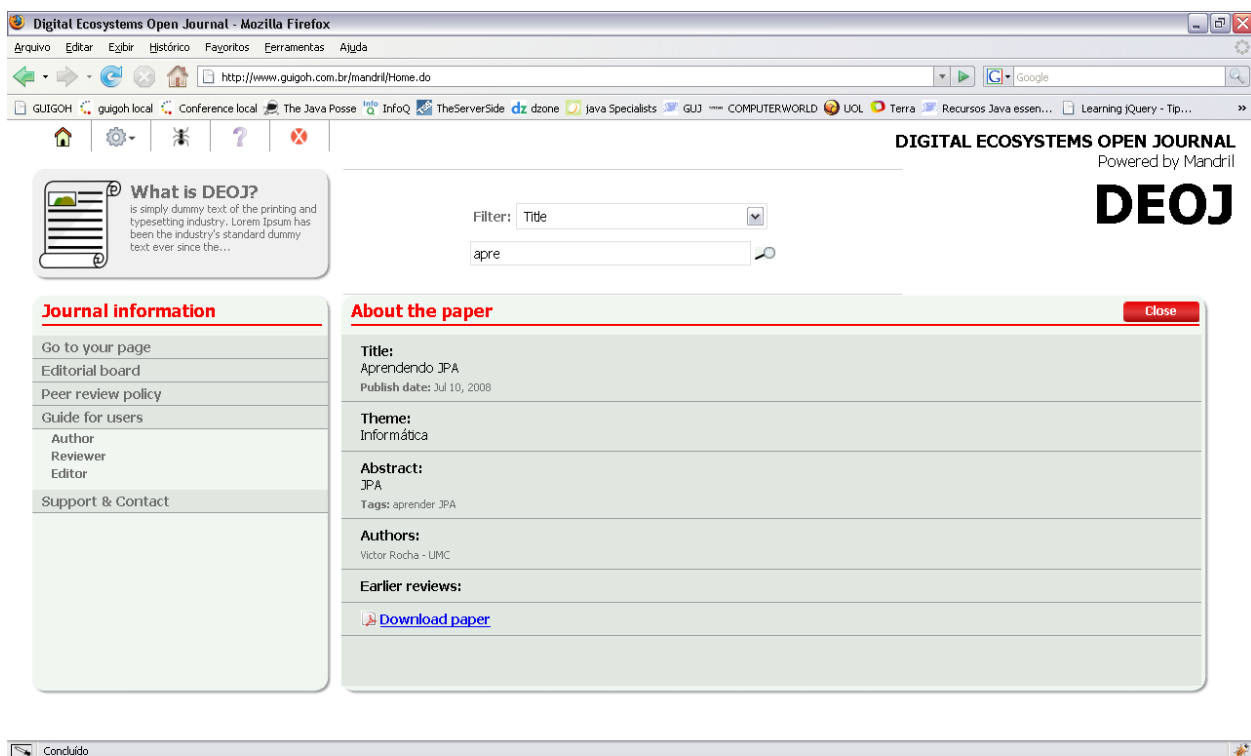


Figure 17. About the paper page.

On the left menu, there is a “Go to your page” link. This link takes the user to the personal paper management step (Figure 18). There, the user can find his papers that are on any review stage. If the person is not a reviewer and/or editor, only the “My” menu is displayed. But if the person is a reviewer and/or editor, the system also presents all the pending and finished papers to be reviewed and/or edited.

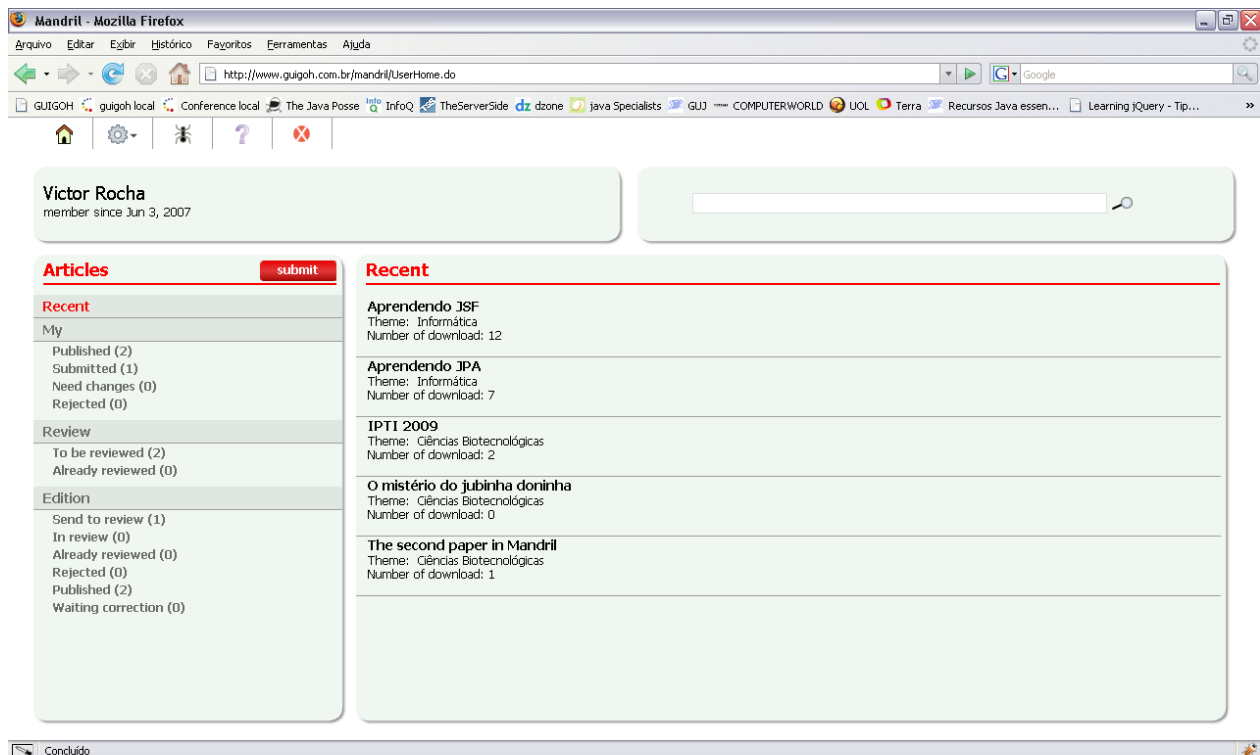


Figure 18. Personal paper page.

To submit a paper, the user just needs to click the “submit” button and fill the fields: file path, title, tags, theme, abstract, name of the authors and each affiliation (Figure 19). After that, the Submitted counter on the menu is incremented.

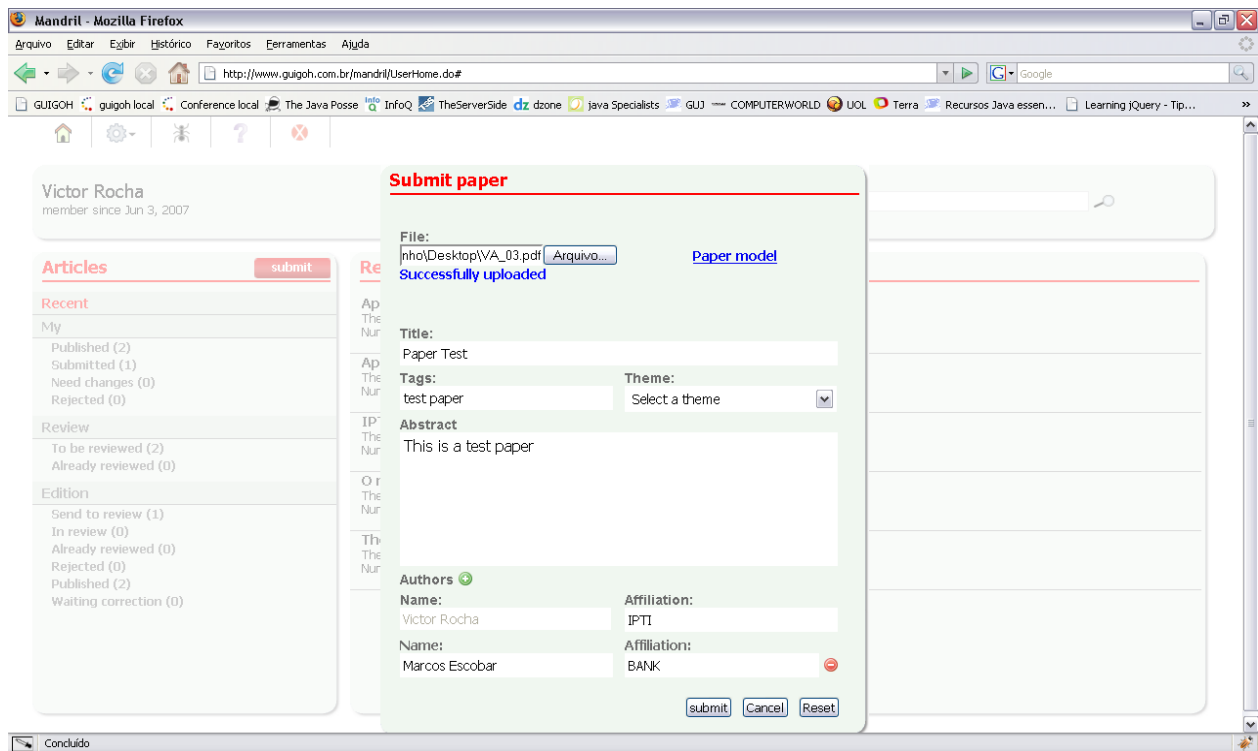


Figure 19. Paper submit page.

Then, the editor can access the paper and send it to review. When the editor sends a paper to review, a reviewer must be selected (Figure 20). The editor can also reject the paper before reviewing. For each reviewer, the system shows a counter with the number of pendent papers that reviewer has. The list of reviewers appears sorted by the less occupied on top.

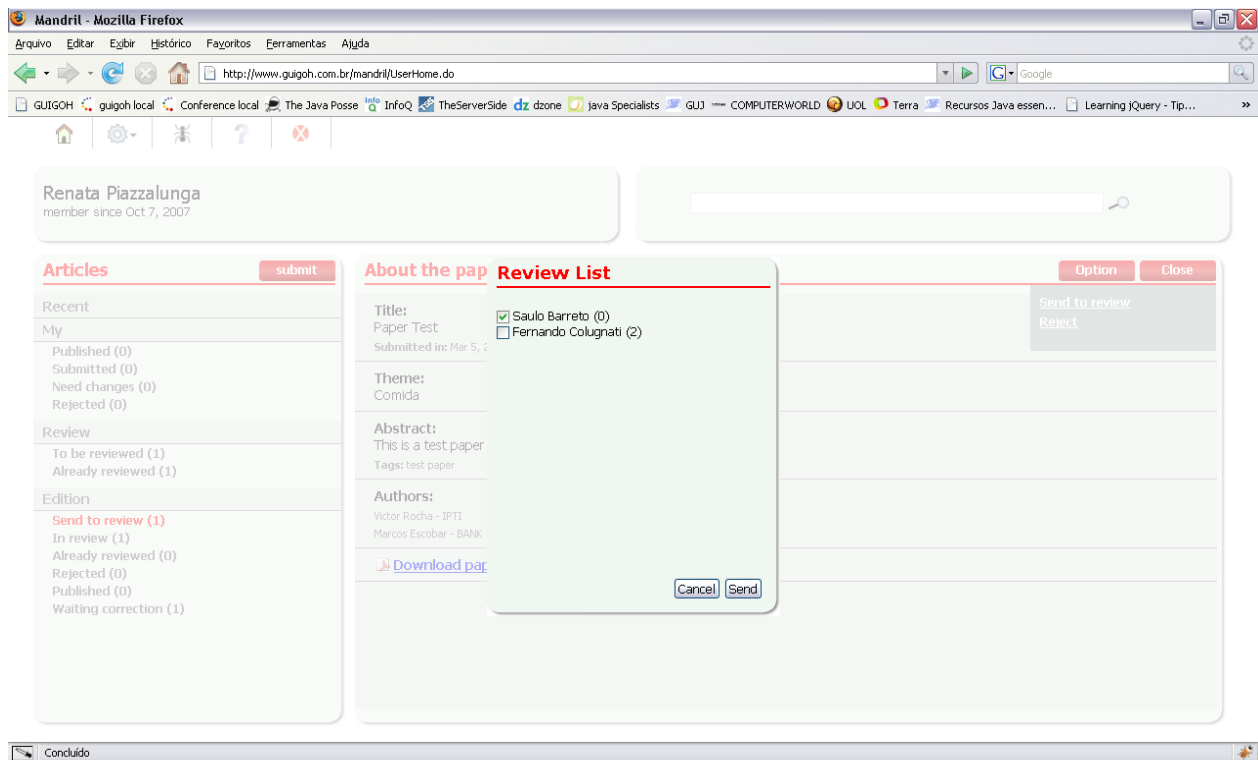


Figure 20. Send a paper to a reviewer.

After that, the reviewer can see the papers to be reviewed by clicking the correspondent link. Selecting a pendent paper show an “About the paper” page and the reviewer can download the PDF file to read it. The next step is to send the review to the author (Figure 21). The fields of the Review Form are: “The paper content is relevant to the theme?”, “The content is grammatically correct?”, “The content is understandable and objective?”, “Are there references?” and the open comments to the editor and to the author. Finally the reviewer can choose if the paper is accepted, rejected or accepted with revision.

The screenshot shows a web browser window titled "Mandril - Mozilla Firefox". The address bar displays "http://www.guigoh.com.br/mandril/UserHome.do". The page content is organized into three columns:

- Left Column:**
 - User profile for "Saulo Barreto" (member since Sep 10, 2007).
 - Articles** section with a "submit" button and a list of article statuses: Recent, My, Published (0), Submitted (0), Need changes (0), Rejected (0), Review, To be reviewed (1), Already reviewed (1), Edition, Send to review (1), In review (1), Already reviewed (0), Rejected (0), Published (0), and Waiting correction (1).
- Central Column:**
 - About ti** section with fields for Title ("Paper Tes"), Theme ("Comida"), Abstract ("This is a te"), and Tags ("test p").
 - Review Form** section with questions in Portuguese:
 - "O conteúdo do paper condiz com o tema?" (yes)
 - "O conteúdo está correto gramaticamente?" (yes)
 - "O conteúdo está claro e objetivo?" (absolutely)
 - "As citações bibliográficas foram apresentadas?" (Good authors)
 - Comments** section with "To editor:" (Good comments) and "To author:" (Passed).
 - Decision** section with radio buttons for "Accept" (selected), "Reject", and "Accept with revision".
- Right Column:**
 - A search bar.
 - A "Review" button and a "Close" button.
 - A list of reviews, each with a "Review" button and a "Close" button.

Figure 21. Review Form page.

The reviewer can access the reviewing and paper information later. The editor can check the recent modified papers and view the review and comments. The editor can also moderate the comment of the reviewer, asking more details of the revision or any other information (Figure 22).

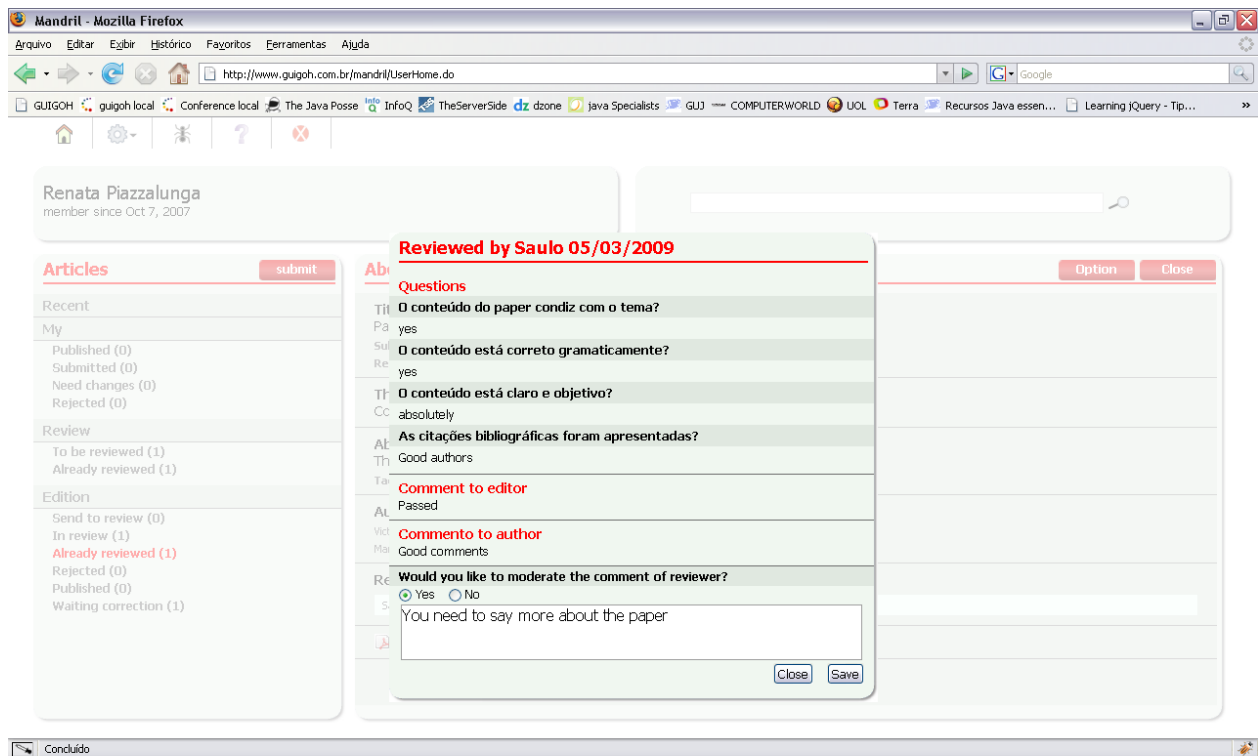


Figure 22. Editor revision viewer page.

After checking the revision, the editor can reject the paper, publish or request corrections. If the editor choose to publish the paper, a comment can be made for that paper (Figure 23). But if the editor requests corrections, the author should submit the same paper with the adjustments

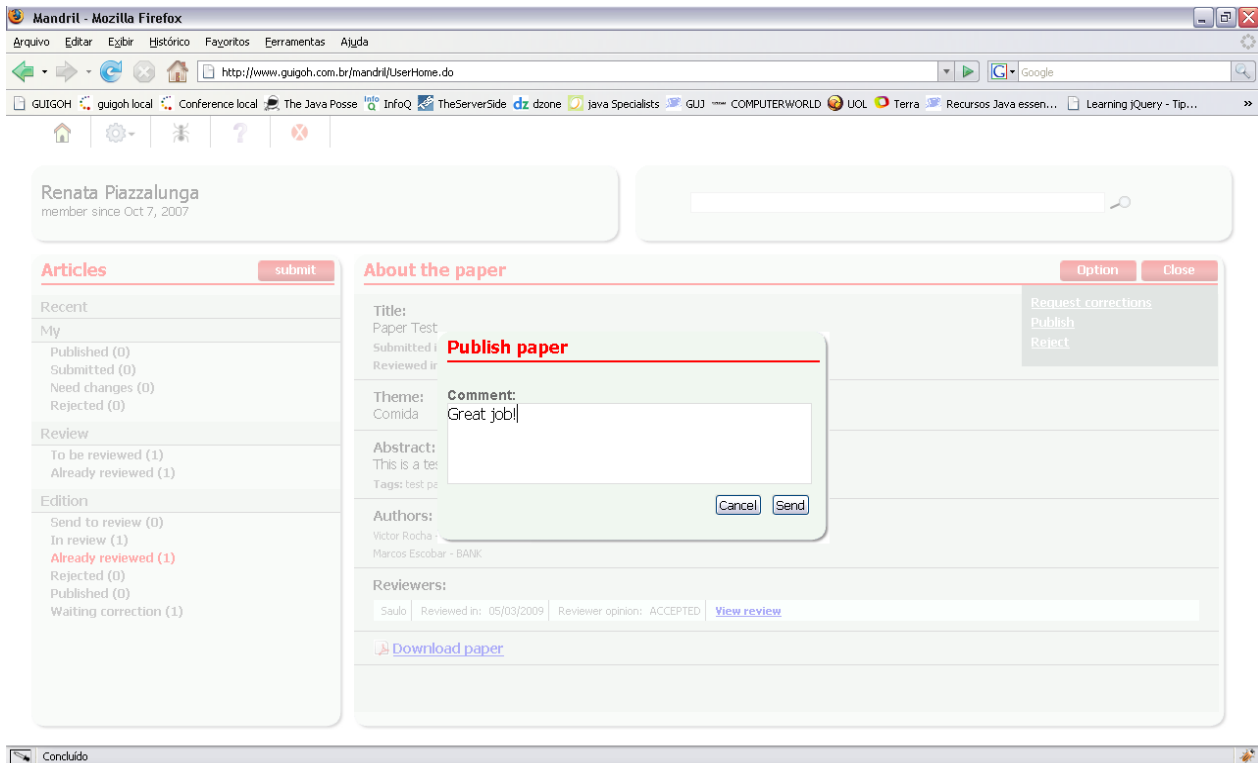


Figure 23. Editor comment page.

The editor and the author will then see the published paper in the “published”, “rejected” or “waiting for correction” menu item.

6.4 Common Architecture for the Tools

The common features and technical specification of the tools are described next. It explains the technologies used in the project, such as programming languages, servers, databases and integrated projects.

The Working Paper Series is already available as an open source project called *Mandril*. *Mandril* is divided in two main projects: the *Mandril-core* and the *Mandril-opaals*. *Mandril-core* is the core of the paper review logic structure. This project contains the basic features for document reviewing. The *Mandril-opaals* project is a specialized implementation of the *Mandril-core*. The goals of the project are the submitting and reviewing system for papers in the PDF format. *Mandril-opaals* also has a specific interface

developed considering the project objective. The Mandril project is simple to deploy and its properties are configured easily with a properties file. All the tools described above use the same architecture that will be explained next.

To organize the user base model, a project called *Primata* was created, and is used by all the tools. This project provides a social networking base system that contains the database model and the logic structure to handle users, communities and ties between the users. Also, all tools are exposing its services through a REST (Representational State Transfer) interface, and using REST to call the necessary services from other the tools.

The REST architecture was preferred for its simplicity. This is because the architecture uses the universal URL syntax and it does not depend on the programming language or operational system chosen. The response for a request is delivered using JSON syntax. The JSON response is text based and lighter than a XML response. It represents data structures and arrays with a human-readable format.

To implement each of the three tools system, the Java language version 1.6 was used. Java is a programming language that is multi-platform and it supports a variety of tools and libraries that were also used in the project. The application server selected was JBoss (v. 4.x), because it is one of the most popular application servers nowadays and it is maintained by an active open source community. JBoss also contain a REST library called RestEasy, which facilitates the building of REST based applications. As a side note, the usage of the JBoss application server is likely to be dropped when we move to the peer-to-peer infrastructure. For the database, PostgreSQL was chosen because it is one of the most advanced open source database management system. To handle the data from the database, the Hibernate Framework was employed. This framework manages data mapping tables into objects. It also generates SQL queries and assists the development avoiding human errors on SQL querying and data manipulation.

The user interfaces were developed with *Java Server Pages* (JSP) and *Asynchronous Javascript And XML* (AJAX) technologies. JSP is native Java technology for web systems. AJAX makes the interface friendlier, avoids reloading and handles requests, among other

features. All projects mentioned have a test suite developed using JUnit and JMock (in some cases). Another testing tool used was Selenium, which targets the tests on the user interface. The development uses Mercurial tool for source code version control, which maintains a distributed repository. Liquibase, a tool that helps organizing database versions through xml files was used as well

7. Roadmap for Future Work

Here we present a roadmap for the implementation of the functionalities mentioned throughout this document that are not implemented yet.

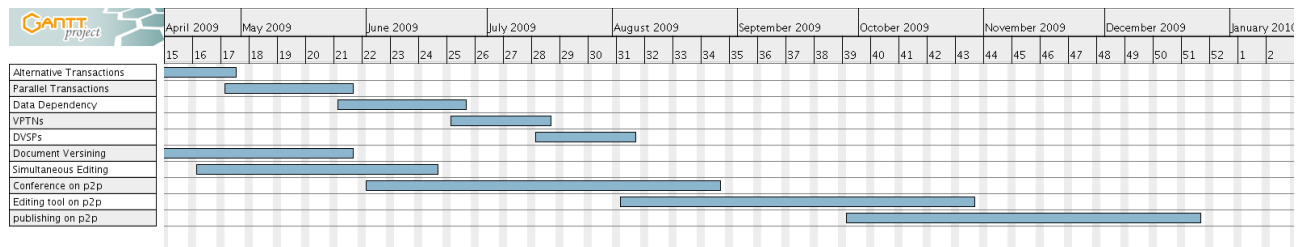
In terms of the peer-to-peer structure being implemented, we only have the functionality for sequential transactions done. In other words, in the current prototype implementation only actions happening serially, one after the other, are allowed. The next steps are, in this order: alternative transactions, parallel transactions, data dependency among transactions, VPTNs and DVSPs. Each of those should take from a couple of weeks to a month.

Also, there are the new functionalities to be added to the document editing tool: the handling of different document versions and the simultaneous edition of documents from multiple participants. Each of those should take from one to two months, and the later depends on some output from the transaction model, and in particular, the semantics and the locking scheme described in Deliverable D3.2 to address concurrency issues. So this extension will require some support from University of Surrey, as will the whole peer-to-peer layer being implemented.

Now, to conclude the planned work, the elements of the OKS used to provide the Working Paper Series a nice environment will have to be switched to work on top of the peer-to-peer layer. This will happen to the conference tool and its sub-elements, including text and Voice chat and the whiteboard, the document editing tool, allowing collaboration between partners without having to rely on centralized servers, and finally the Working Paper Series itself,

allowing the whole community to have access to all the published documents, again without having to rely on centralized servers.

The following diagram illustrates all of this information:



8. Conclusions

The system we envisage for supporting cooperative article writing has now become a more interactive and complete tool, integrating functionalities for editing, communicating and direct submitting for publishing in PDF file format. This change in relation to the original plan means a delay in terms of implementing the full system but we were able to implement a minimum list of requirements which allow OPAALS members to use and test the system, including both coupled one (conference and publication).

The integrated system will run over the DE architecture which means using the p2p network to support the work, mainly for the simultaneous editing and the communication through voice. A unique user interface design will also be part of this second version of the system and it will be conceived through a process of sharing and agreement between the consortium in order to achieve a most usable platform as possible.

In order to be able to conclude the implementation work the task related to this deliverable was split in two parts, being this one here reported related to the first part. A D9.6.2 is suggested based on a roadmap of implementation and should be delivered by M39.