TÉCNICO
LISBOA

E-Conf: A Conference Management System

João Pedro Martins Morais Luís

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisors: Prof. Nuno Cavaco Gomes Horta
Prof. Nuno Calado Correia Lourenço

Examination Committee

Chairperson: Prof. José Manuel da Costa Alves Marques
Supervisor: Prof. Nuno Cavaco Gomes Horta
Members of the Committee: Prof. Alberto Manuel Ramos da Cunha

November 2016

Acknowledgments

First of all, I would like to thank my supervisor, Professor Nuno Horta, for all the guidance given during this project. His experience in this field was crucial for the success of this thesis. A special thanks to you.

I would like to thank my parents and my sister, for their patience, support and encouragement throughout my years of study and through the process of researching and developing this project.

I would like to offer a special thanks to Daniela, my girlfriend, for all the support, late nights, and mainly for all the cheering during the course, and specially during this last phase. Your encouragement was essential for me to finish this project.

I would also like to thank Alexandre and João for all the support during the course. Working with you turned what could have been boring, difficult and long hours of work into enjoyable times. We will make that yet to be named game some day!

To all my friends who supported me during this journey, thank you.

Finally, a thanks to God, for all the blessings given to me, and for all the help given through the people mentioned above.

Abstract

Conferences are an essential channel for researchers to exchange information. Nowadays, there are hundreds of thousands of conferences every year, each with its own characteristics. However, organizing one is a difficult and time-consuming process, involving hundreds of people. There are currently many Conference Management Systems to help with this process, some of them analysed in this document, but they can be too complex or lack some needed features.

This work introduces E-Conf, a web based Conference Management System designed and developed to help an organizer planning a conference, automating a great part of the organizing process. The features of E-Conf are divided in four main categories - main app, submissions, reviews and program management - which became the system modules.

E-Conf also includes two algorithms for automatic assignment: AREA, to assign submissions to reviewers, and Chronos, to assign submissions to sessions. When evaluated, both algorithms proof themselves capable, with AREA achieving pretty good results in the used metrics.

Keywords

Conference; Conference management; Paper management; Abstract management.

Resumo

As conferências são um canal essencial para investigadores trocarem conhecimento. Hoje em dia há centenas de milhares de conferências, cada uma com as suas características. Contudo, a organização de uma é um processo difícil e demorado que envolve centenas de pessoas. Existem atualmente muitos Sistemas de Gestão de Conferências para ajudar neste processo, alguns dos quais são analisados neste documento, mas estes podem ser muito completos ou suprimir algumas funcionalidades necessárias.

Este trabalho introduz o E-Conf, um sistema de gestão de conferências baseado na web desenhado e desenvolvido para ajudar os organizadores a planear uma conferência, automatizando grande parte do processo de organização. As funcionalidades do E-Conf estão divididas em quatro categorias principais - aplicação principal, submissões, revisão e gestão de programa - que se tornaram nos módulos do sistema.

O E-Conf também inclui dois algoritmos de atribuição automática: o AREA, para atribuir submissões a revisores, e o Chronos, para atribuir submissões a sessões. Quando avaliados, ambos se mostram capazes, com o AREA a alcançar bons resultados nas métricas utilizadas.

Palavras Chave

Conferência; Gestão de conferências; Gestão de artigos; Gestão de resumos.

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

CMS	Conference Management System
MVC	Model-View-Controller
2FA	Two-factor authentication
UI	User Interface
UX	User Experience
API	Application Programming Interface

1

Introduction

Contents

This chapter provides a background for the work discussed in this document. A brief description of the problem is presented, along with the project goals. Finally, the global structure of this document is described.

1.1 Background

Conferences are an important channel for the exchange of information between researchers. As such, it should not be a surprise that nowadays conferences are numerous and varied - in the calendar year 2012 there were 273700 conventions/conferences/congresses only on the United States [2, table 1]. There are many types of conferences - small and big, themed and general, with many types of events - and they all need to be planned and organized.

Planning is the key to a conference success, so early planning is essential for everything to run smoothly. To begin, the conference team must be formed and the conference parameters agreed - title, themes, topics, committees, dates, venue, sponsors, etc. With this initial planning complete, the first call for papers is issued (usually on the conference website, which is launched around this time), preparing the submission phase. Some organizations opt for a two-phase submission, with a preliminary submission first, to estimate the amount of submissions and the number of reviewers needed.

During the submission phase, researchers submit their work, either the full paper or only the abstract - according to what the organization decided. By this time, submissions can be accepted to all the types of sessions the conference will held (talks, posters, workshops, panels...) or only to some of them, with submissions for the other types being accepted later.

When the submission phase closes, the submitted work is evaluated. For this phase, the organization invites a number of reviewers, according to the number of submissions, so that each submission is reviewed by (usually) at least three reviewers. The submissions are assigned to reviewers, who evaluate them with the criteria the organization decided. Reviewing may be anonymous, hiding the author information for the reviewers.

After the reviewing phase, the conference team makes the final decision on which submissions to accept. The authors of the accepted submission must then submit the full paper, if only the abstracts were submitted, or resubmit the paper making small corrections according to the reviewers' suggestions, along with the presentation materials and the biography info.

Around this time, registration for the conference is open, and may include options for early bird and standard rates, workshops, meals, etc.

Afterwards, it is time to create the conference timetable, creating sessions and assigning events to sessions. The timetable is then published on the conference website. Finally, the conference proceedings are prepared and printed and/or digitally published.

Table 1.1, from Ex Ordo free conference planning eBook [1], illustrates the process, showing the major milestones.

Month	
1	**Start planning**
2	**Planning complete**
3	Launch conference website
	Issue call for papers
4	Invite reviewers
6	Close call for papers
7	Open early bird registration
	Begin peer review
8	Peer review ends
	Accept/reject abstracts
9	Receive copyright, final versions
10	Close early bird registration
	Create conference timetable
11	Print hardcopy book / Publish digitally
12	Close registration
	Conference day
13	Post-conference meeting
	Final report / Handover to next chair

Table 1.1: Conference organization timeline [1, Free conference planning eBook]

1.2 Motivation

This whole process can involve hundreds of people, thousands of e-mails and weeks of an organizer's time. However, most of the tasks can - and, therefore, should - be automated, saving hours of administration time.

As a result, there are currently many CMSs in the market, with varying sets of features, which will be presented in section 2.2. However, it is difficult to find an ideal system among them: some of them are too complex, others lack some important features, and the majority do not offer a pleasant user experience. Additionally, most of them are closed, and so they can not be improved or expanded.

1.3 Objectives

Responding to the difficulty of organizing a conference, the main goal of this project is to design, and then develop, a software that can automate as many of the required tasks as possible, as well as helping organizers with the ones that can not be completely automated. When the system is complete, an organizer should be able to use E-Conf to help the planning and organization of a conference.

To do this, several existing CMSs will be studied and their features analysed, along with other related work. Based on that analysis, the system's set of features will be decided and the architecture of the new system will be designed, trying to answer to the existing problems.

With the system architecture defined, the system will be implemented. Finally, the created system will be tested.

1.4 Document structure

This thesis is organized as follows: Chapter 1 describes the process of organizing a conference, exposing the background needed for the rest of the document. Chapter 2 lists and explains the features present in most CMSs, along with a comparison between them. Further related work is also discussed in this chapter. Chapter 3 describes the requirements for this system, obtained from the analysis in the previous chapter, and the architecture resulting from these requirements. Chapter 4 lists the features implemented on this thesis and details their implementation. The tests of the project and its results are presented on Chapter 5. Chapter 6 contains the conclusions of this thesis, along with further work that can be done on E-Conf.

2

Related work

Contents

For the study required for this work, several commercial and open-source CMSs features were analysed. Unfortunately, for the vast majority of the analysed systems there is not a demo or trial publicly available for the common user to get in touch with the system, so the study had to rely on the feature list available on the systems' websites. Additionally, their architectures are also unavailable, so they can not be analysed.

In general the studied systems have the same overall set of features, helping on most of the tasks described on chapter 1 - set up, submission, reviewing, program management and registration - with the main difference being the level of automation done by the system.

2.1 Features of a CMS

As stated in the introduction, organizing a conference is a process that usually follows a sequence of well-defined and well-delimited steps, a linear workflow described on figure 2.1. As a result, most of the CMSs are designed around this flow and most of their features fit in one of these steps.

This section describes most of the features present in, and expected from, a CMS, organized by steps.

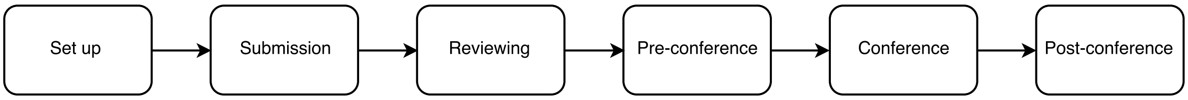


Figure 2.1: Conference flow

2.1.1 Set up

In this first phase, chairs are able to configure the details of their conference, such as names, dates, etc. EasyChair and ConfTool, on their Pro versions, allow complete custom branding, which means the CMS brand is completely hidden from the end user.

EasyChair can import configurations from another conference, allowing a quicker set up to users who organize conferences frequently.

2.1.2 Submission

One of the main functions of a CMS is to automate the articles' submission and processing. As such, a chair must have full control over the submission process.

Some CMSs allow chairs to create custom submission types (for example, for different types of sessions or for prizes).

The submission form is also customizable in some of the systems, allowing for custom fields and/or for custom instructions for the authors. The form is also capable of receiving a file, with some CMSs allowing multiple files or having support for different types of files.

All CMSs studied allow the chair to contact the authors through the system, saving much time that would be used sending hundreds (or thousands) of e-mails.

2.1.3 Revision

Revision is essential to decide which papers are chosen, and therefore another main function of a CMS, as much of its work can be automated.

Firstly, every CMS[1] allows a chair to add reviewers to the program committee, and some of them handle the whole invitation process - sending e-mails and processing their response. In most of the systems, reviewers can pick topics of their interest, to help the matching between submissions and reviewers.

When the committee is complete, reviewers can use the CMS to bid on papers, i.e., to choose which papers they would like to review and/or are more comfortable reviewing. Bidding is available on EasyChair, OpenConf and ConfTool.

The submissions must then be distributed by the reviewers. The majority of the systems allow this assignment to be done manually or automatically. To make this process easier, some of the systems also take into account conflicts of interest, manually marked or automatically detected, as well as other constraints on paper assignment.

EasyChair and EDAS allow reviewers to delegate some of the reviews assigned to them to other users - subreviewers.

The review form is also in the majority of the cases customizable, the form fields or only the evaluation criteria.

Most systems allow for online discussion of the submissions.

After the submissions are reviewed, the reviewers may leave some notes to the author to adjust their paper. Most of the systems allow authors to resubmit their work according to these notes - the author response phase.

Finally, the organizers decide which papers will be accepted to the conference, according to the reviews. Some CMSs allow custom decisions, beyond the more common binary acceptance.

[1] OSEM is the exception to this rule, as it doesn't have any revision functionality.

2.1.4 Pre-conference

The remaining work to be done before the conference can be divided in two main categories: **program management** and **registration**.

2.1.4.A Program management

CMSs can be very helpful when managing the conference program. Most of them allow a chair to create and edit sessions and talks, as well as to assign talks to sessions. A majority of the systems can do this process automatically.

EasyChair and OSEM also handle rooms, allowing to assign rooms (and their capacities) to events.

With the program complete, most systems are also able to publish it.

2.1.4.B Registration

Ex Ordo, EDAS, ConfTool and OSEM also handle registration for a conference, being Ex Ordo the most complete. The majority of them allow chairs to set custom price tiers (such as early bird, 1-day only, etc.), and Ex Ordo also allow extras, such as meals, to be defined. Most of these CMSs handle payments themselves.

Some of them also allow registration to some sessions which require it (e.g., workshops).

2.2 Existing CMSs

There are currently many CMSs. Seven of them were studied for this work: EasyChair, OpenConf, Ex Ordo, EDAS, ConfTool, OSEM and ePapers. To complement the previous section, the systems are presented below, with a full feature comparison available on table 2.1. Some of the systems have different versions (or editions), described below and referenced in the comparative, each providing more features than the previous.

EasyChair

EasyChair is currently probably the most commonly used CMS and according to their website, hosted more than 40000 conferences. It is the system with the most complete set of features of the study, although it lacks any registration functionality. EasyChair is provided as a service with three different kinds of licenses: Free, Professional and Executive. [3]

Table 2.1: Feature comparison

	EasyChair	OpenConf	Ex Ordo	EDAS	ConfTool	OSEM	ePapers
Hosting and support							
Hosting on own server		C			V only	X	
Hosting on a high-performance server	F	P	X	X	P		X
Data backup and replication	F						
Encrypted access only	F				P		
Helpdesk and technical support	E	+	X				X
Miscellaneous							
Import configuration from other configuration	P						
Custom branding	P				P		
Authors and submissions							
Anonymous submissions (double-blind review)	F		X	X	V		
Custom submission types	F		X				X
Author notification	F	C	X	X	V	X	X
Instruction for authors	P		X				
Custom fields in submission form	P		X		P		
Multiple file uploads	P	C			P		
Audio, video and ZIP upload	P	P	X		V		
Terms & conditions for authors	E						
Program committee and reviewing							
Event log	F						
Watchlist	F						
Online discussion of submissions	F	+		X	P	X	
Flexible access models for reviewers	F						
Leading reviewer model	F						
Subreviewers	F			X			
Conflict of interest	F	C			P		
Author response phase / Rebuttal	F	+	X		P		
Program committee invitations	F		X	X	P		
Submission topics	F	C	X		V		
Customized decisions (acceptance types)	F	+			V		
Customizable evaluation criteria	F		X	X	P	X	
Statistics	F						
Custom review forms	P		X	X			
Multi-track support	P		X	X	P (Extra)	X	
AAAI model	E						
Paper bidding and review assignment							
Paper bidding	F	+			P		
Automatic submission assignment to reviewers	F	C	X	X	V		
Manual submission assignment to reviewers	F	C	X	X	V		X
Constraints on paper assignment	F						
Data export	P	C		X	P	X	
Email management							
Email to authors and program committee	F	C	X			X	
Email log	F						
Proceedings							
Proceedings creation	F	+	X				
Springer LNCS proceedings	F						
Program and page generation							
Program generation and publishing	P	P	X	X	P		
Page editing	P	P					
Talks and sessions editing	P	P	X			X	X
Assignments of talks to sessions	P	P	X		V	X	X
Scheduling constraints	P						X
Session chair handling	P	P		X			
Room handling	P					X	
Program manager handling	P						
Automatic schedule analysis	P						
Scheduling suggestions	P						
Solutions	P						
List of events	P						
Email to managers, authors and session chairs	P		X	X		X	
Multi-conference program generator	E						
Program page templates	E	P					
Program download	E	P	X				
Registration							
Registration			X	X	V	X	
Set custom prices			X	X	V		
Extras			X				
Payment handling			X	X	V		

EasyChair F - Free, P - Pro, E - Enterprise;
OpenConf C - Community, + - Plus, P - Pro;
ConfTool V - VSIS, P - Pro.

OpenConf

OpenConf biggest lack is on program features, and as EasyChair has no registration features. OpenConf has three editions, Community (free), Plus and Professional, and is provided as a product, i.e., the user downloads and installs it on his server. OpenConf also provides a hosting service, serving the Professional edition. [4]

Ex Ordo

Ex Ordo is provided as a service and has no free version. It has one of the most modern designs, along with OSEM. Differently from the previous systems, Ex Ordo sells a basic (Essential) package, which can be expanded with add-ons. [1]

EDAS

EDAS is also provided as a service and most of the features are included in the basic fee, the notable exception being the setup of the event public page. EDAS can also provide some additional material, like CD-ROMs and USB, for extra fees. [5]

ConfTool

ConfTool has two different versions: VSIS and Pro. VSIS was designed for smaller events, has only the basic functions and is offered for local installation only, free on request. Pro is provided as a service and has substantially more features, being suitable for bigger and more complex conferences. [6]

OSEM

OSEM is an event management system for free software conferences. It is an open-source system and a single installation can manage multiple conferences. Its main lack is the reviewing process, as OSEM does not have any type of reviewer assignment. Along with Ex Ordo, it has one of the most modern designs. [7]

ePapers

ePapers provides two services: submission & review, which handles the whole process, and final collection, which handles only the collection of final manuscripts. According to its website, its main difference is that ePapers is a ”full-service” system instead of a ”self-service” system, meaning that ePapers team prepares the system for the specifics needs of its users. ePapers is the less comprehensive system of the study. [8]

2.3 Other work

There are some articles about CMS solutions that are also important for this study, proposing new systems or new algorithms for some specific tasks.

2.3.1 COMFy

COMFy is a conference management framework, instead of a system, and it is divided in layers. The bottom layer is a relational database. The second layer is composed of three parts: the repositories, which abstract the database to the upper layers; the state machine, which is described as the core of the conference and manages the phase of each submission; and additional modules. The third layer its COMFy, the business logic of the CMS, which handles the requests. This layer is designed as an Model-View-Controller (MVC) pattern and exposes a well defined RESTful API - the fourth layer - which can be used to build applications - the top layer.

To detect conflicts of interest, COMFy uses, along other information, bibliographic data, such as co-authorships. COMFy gets such data from the DBLP API, described on “DBLP XML Requests”, an appendix to [9]. [10]

2.3.2 GRAPE

GRAPE is a review assignment component for CMSs. GRAPE takes advantage from both the papers content (topics) and the reviewers preferences (biddings) and defines two measures to guide the system: reviewer’s gratification and article’s coverage, and tries to maximize both of them for all reviewers and all papers. GRAPE also accounts for conflicts of interest, both explicit by the organization and deducted by itself. [11]

2.4 Conclusion

This chapter presented the related work, mainly through existing CMSs and their features.

Generally speaking, the analysed CMSs share the same set of features. This is not really a surprise, as these features follow very closely the organizing process of a conference. EasyChair is the most comprehensive of the study, automating almost every step in the process.

An analysis of this process and these features provides a foundation for E-Conf requirements, presented in the following chapter.

3

Architecture

Contents

This chapter describes the architecture of the solution. Section 3.1 lists the project requirements, functional and non-functional. The general architecture of E-Conf is presented on section 3.2. The remaining sections describe the architecture of each of the app modules.

3.1 System Requirements

3.1.1 Functional requirements

The functional requirements for E-Conf are based on the feature set offered by existing CMSs, along with experience from conference chairs.

Like section 2.1, the functional requirements will be presented by phase, according to figure 2.1.

3.1.1.A Set up functional requirements

When first preparing E-Conf, an organizer shall be able to set the conference details - name (and short name), brand, dates and topics. An organizer shall also be able to configure the types of events the conference will hold (e.g., papers, workshops, panels...).

3.1.1.B Submission functional requirements

To set up this phase, an organizer shall be able to define the start and end dates of the submission phase. An organizer shall be able to configure the submission form, both by adding text instructions for authors and by adding/removing form fields, including deciding if the full paper should be submitted on this phase or only the abstract. The organizer shall also be able to issue a call for papers from E-Conf.

During the submission phase, authors (i.e., anyone registered on the system) shall be able to submit their work, as many articles as wanted. Authors shall also be able to edit their submissions during this phase, as there can be mistakes when submitting or small changes to the work.

Organizers shall also be able to enable or disable editing outside of this period, if they want. An example use case of this requirement would be to submit the full article after the abstract was approved.

3.1.1.C Revision functional requirements

While preparing this phase, an organizer shall be able to define the start and end dates of the revision phase, as well as of the bidding phase. Organizers shall be able to invite reviewers for the conference through E-Conf, using their emails.

Regarding the review form, organizers shall be able to set the marking scheme for papers, i.e., define the factors and their weights, for E-Conf to be able to calculate a score.

Reviewers shall be able to choose the topics of their interest, to allow for better matching between reviewers and works, as well as be able to bid on submissions, choosing the ones they are willing or more comfortable to review, during the bidding phase.

Organizers shall then be able to assign the submissions to reviewers or order the system to do it, and then be able to edit the assignments. The system shall also detect conflicts of interest, both for manual and automatic assignment.

When the revision period starts, reviewers shall be able to review and mark the submissions assigned to them or delegate them to a subreviewer (who does not need to be invited by an organizer) to do it.

3.1.1.D Pre-conference functional requirements

After the revision phase, an organizer shall be able to accept or refuse submissions. An organizer shall be able to define how many submissions will be accepted (through number of submissions or percentage), and both this value and the submission score shall be shown to organizers as a guide to help accept submissions, although they shall not impose anything.

With the accepted submissions chosen, authors shall be notified (and informed of the reviews) and shall be able to submit the final work and pick which of the authors will be the presenter.

Program management Also when the accepted submissions are chosen, organizers shall be able to prepare the program. For this, organizers shall be able to define the venues and rooms (including their capacity) where the events will take place and create sessions. Organizers shall then be able to assign submissions to sessions or order the system to do it, and then be able to adjust the assignments. The program shall then be made available through the conference page.

3.1.2 Non-functional requirements

Given the functional requirements that were analysed in the previous section, E-Conf has specific requirements that justify the architecture of the solution.

Configurability As stated in the introduction, conferences are numerous and varied. (It is, in fact, the fifth sentence of this document.) As such, E-Conf must be configurable and adapt to the needs of a specific conference and all its phases.

Modularity Section 2.1 show that a conference flow is clearly divided in phases, each with its own set of features. As such, these features should be separated in modules, each module having its own responsibility. This simplifies the system while allowing for easy updating (or possibly reimplementation) of a single module.

Modularity also helps the system configurability, allowing an administrator to install only the modules required for their specific needs.

Extensibility Taking into account the previous two requirements, E-Conf must be also be extensible, i.e., E-Conf should be ready to accept new functionality in the form on new modules. If E-Conf lacks a feature needed for a conference, an administrator shall be able to code it and add it to their E-Conf installation with minor (or, ideally, none) impact to the other functionalities. Additionally, E-Conf shall also be extensible in an administrative way, i.e., new rules or policies may be applied.

Platform compatibility Given the nature of a CMS, E-Conf will be accessed by many and different types of users. As nowadays there are many platforms and screen sizes, a user can access the system from any of them. So, E-Conf shall be easily accessed from any platform.

Reliability Any data loss or down time could be critical for E-Conf. So, the system shall try to avoid any down time and, mainly, try to prevent data from being lost.

Security As different users will be able to access a different set of features, authorization should be made according to the user role. This also implies that the system shall identify the user when accessing E-Conf.

Usability Using the system must not require previous training from the users, not only the committees but specially the researchers, who will use the system only one or two times and should be able to submit their work easily.

3.2 General architecture

This section discusses the design options of E-Conf and overviews the system architecture.

3.2.1 Design options

Some design options made from the above requirements and the analysis are presented below.

Web application The best way to achieve the best platform compatibility is to develop E-Conf as a web app. Most of the devices nowadays, if not all of them, have a web browser, making them able to access the system, while removing the need for the user to install a new software on their device.

Multitenancy Most of the CMSs are provided as a service, and a result a single installation of the software is able to handle multiple conferences, independent from each other. Also, most of the institutions / organizations that host a conference usually host more than one. So, E-Conf must be a multitenant application: each conference (the tenant) has a dedicated share of the instance, including its data, configuration, etc.

However, there may also be cases where this is clearly too much for the user's needs. So E-Conf must also be able to host a single conference, with no further requirements. (This also fills the configurability requirement.)

Multi language Conferences are usually open to authors (and attendants) all around the world, so it would be nice, although not required, to be able to present E-Conf to users in their own language (or in the language they prefer).

Hooks To make E-Conf extensible, a hooks system will be used, like WordPress. This method places actions and hooks through the code so that the system calls package functions at specific times. The use of hooks in WordPress is one of the reasons it is so popular and has so many plug-ins.

3.2.2 System overview

E-Conf comprises four modules, as shown in figure 3.1.

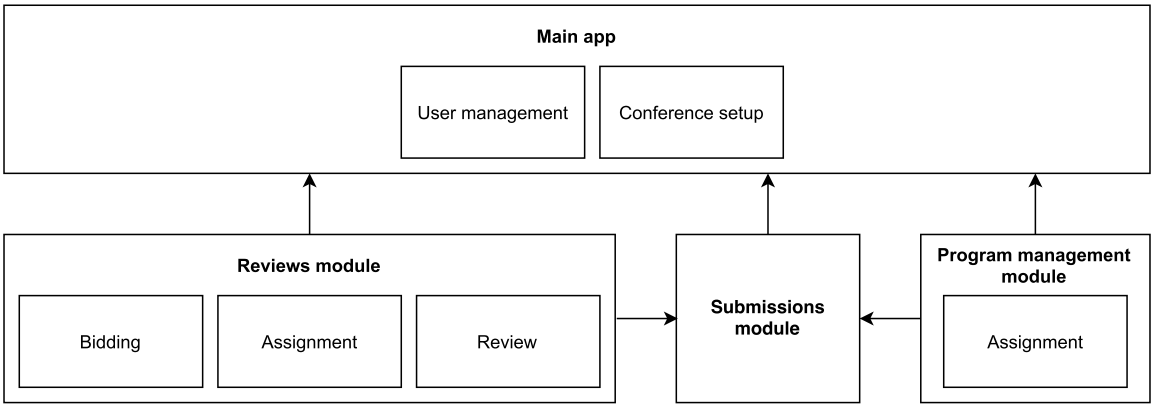


Figure 3.1: System architecture overview

The main app module is the foundation of the system, and as a result all other modules need it to work. The responsibilities and architecture for each module are detailed in the following sections.

3.3 Main app

The main app will handle all the abilities common to all the steps and make them easily usable by all the modules. The main app is responsible for multitenancy, user management and the conference setup.

The data model for this component is presented in figure 3.2 and its parts are described below.

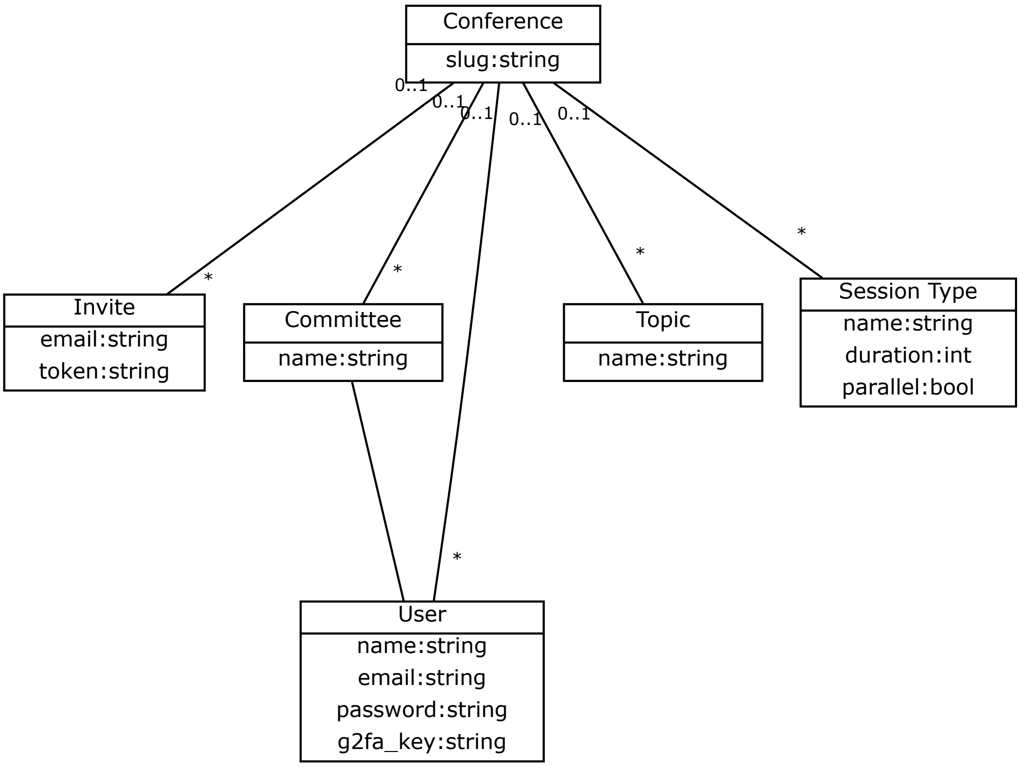


Figure 3.2: Main app data model

3.3.1 Multitenancy

The system administrator must be able to decide if the system will host a single conference or many events.

If they decide for multitenancy, a special type of user must exist - the network manager - that does not belong to any conference and manages the conferences in the system. These users shall be able to access a panel where they can create and remove conferences from the network.

(All the other subsections describe the system from the point of view of a single conference.)

3.3.2 User management

Users shall be able to register through the conference website, and login to access their personal area. Users shall use their email address and a password to login. To improve security, users shall be able enable Two-factor authentication (2FA) for their account.

To ensure proper authorization, i.e., each user can only access what they are supposed to, users shall be able to be assigned to committees, which will then limit access to sections. To add users to committees, an organizer or a committee chair shall be able to send an invite to an email address. When a user receives an invite, they may refuse it or accept it, creating an account if needed. Both the committee and invite systems shall be made available to the other modules.

The main app shall also create the organizing committee.

3.3.3 Conference setup

Conference chairs shall be able to setup the conference details from the main app.

The chair shall be able to define naming and branding details (name, logos, colors) and the conference dates. The chair shall also be able to set up the conference topics and session types.

3.4 Submissions module

The submissions module is responsible for receiving work submissions by authors. It is also through this module that chairs can configure the submission phase.

Acceptance is also handled through this module. Although most of the times acceptance is done considering the review results, that might not happen or the review process may be handled through an external service. As such, acceptance must be handled by this module and not the reviews module.

The data model for this module is presented in figure 3.3.

3.5 Reviews module

The reviews module handles the revision process, from bidding to acceptance. Because this module is so dependent on submissions, it requires the submissions module.

To manage authorization in this module, a program committee, containing all the reviewers, shall be created using the committee system described in 3.3.2.

The revision phase, and thus this module, is divided in three parts: bidding, assignment and revision.

The data model for this component is presented in figure 3.4 and its parts are described below.

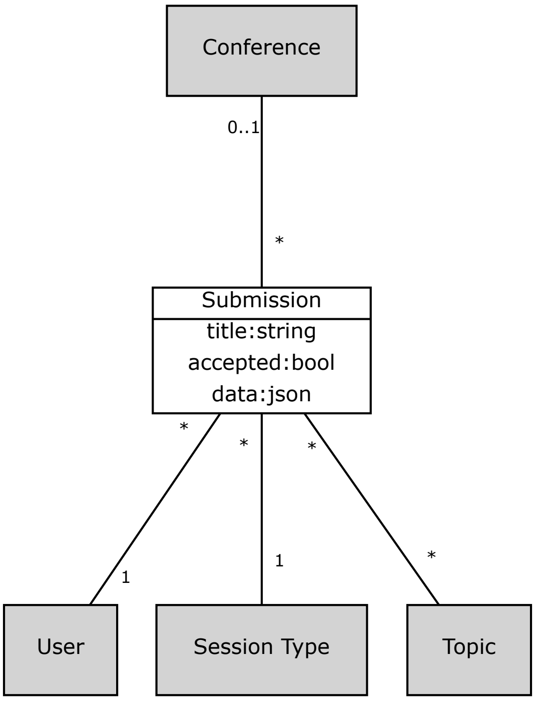


Figure 3.3: Submissions module data model - the entities in grey belong to another module

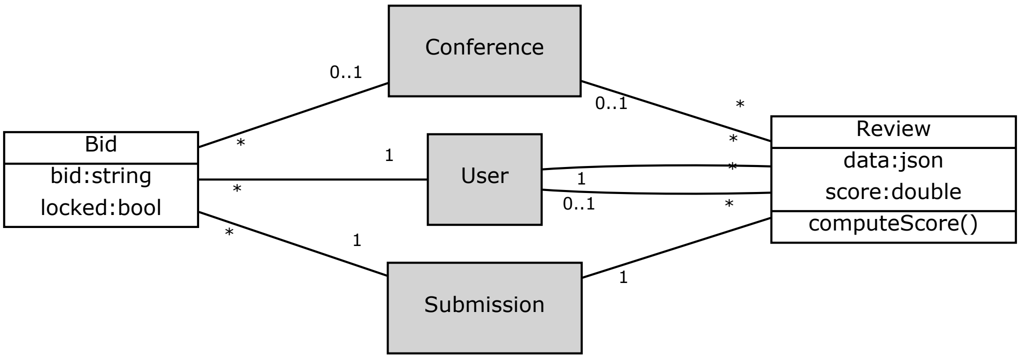


Figure 3.4: Reviews module data model - the entities in grey belong to another module

3.5.1 Bidding

During the bidding phase, reviewers (i.e., users in the program committee) can bid on submissions, choosing one of three values - high, medium or low. Reviewers can also mark if there is a conflict of interest, which are stored as a specific type of bid.

The system shall be able to detect conflicts of interest between submissions and reviewers, and these conflicts need to be marked as an actual conflict by a chair to be applied.

Chairs shall be able to configure the dates of the bidding period and which information will be presented to reviewers during this phase.

3.5.2 Assignment

Assignment occurs between the bidding and revision phases, and is responsible of assigning the submissions to the reviewers.

Chairs shall be able to assign in one of two ways: manually or automatically. Even if assignment is automatic, a chair shall be able to adjust the results.

Assignments will be stored as a review with no score, i.e., a review to be done.

E-Conf shall include an assignment algorithm and be able to receive more easily, through the extension system.

3.5.3 Revision

The revision period is the main part of this module as it is during this phase that reviewers score the submissions, according to the parameters defined by the organization. A chair can define on which parameters submissions will be reviewed, along with their weights.

A reviewer can delegate a review to another user (a subreviewer) if they intend. This invitation will use the invite system described in 3.3.2.

3.5.4 Acceptance

When the reviews module is present, it shall "take over" the acceptance screen, showing the score of each submission and ordering them accordingly. In this case, the system shall highlight the top submissions according to the number defined in the submissions module.

3.6 Program management module

This module is responsible for managing and displaying the conference program.

A chair shall be able to create venues and rooms. A chair shall then be able to create events, and set its details, as well as sessions. Each session has a corresponding event, keeping the timetable logic in one place.

If the submissions module is enabled, and similarly to reviews, a chair shall be able to assign submissions to sessions, either manually or automatically. Like for reviews, E-Conf shall include an algorithm.

The program shall then be shown on the conference public page, with details on the speakers and works.

The data model for this module is presented in figure 3.5.

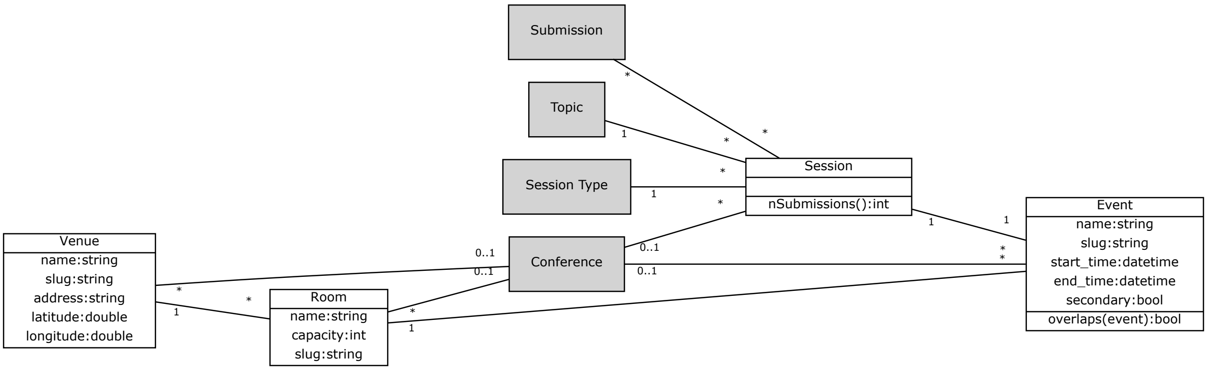


Figure 3.5: Program management module data model - the entities in grey belong to another module

3.7 Conclusion

In this chapter the architecture of the system was defined. As the organizing process is clearly divided in steps, E-Conf will be a modular system, with a module for each step. E-Conf will also be a multitenant system, making it able to handle multiple conferences in a single installation.

In order to be accessible to as many people as possible, E-Conf will be a web-app. Some other design options, such as making the app multi language, were made to improve the User Experience (UX).

4

Implementation

Contents

This chapter details the implementation of this project. First, the general implementation details are described. Then, the implementation of each module is presented.

The project code is available in the GitLab group at `https://gitlab.com/econf`, with the link for the repository of each specific module specified in the respective section. The status of the project at the time this document was written is marked with the tag **1.0.0** on all repositories.

4.1 Implementation options

4.1.1 Technologies

The technologies on which E-Conf is built come from it being a web app (as stated in 3.2.1). To fit that requirement, the app client-side needs to be built using HTML, CSS and JavaScript. There are, however, many different server-side languages to achieve this.

PHP currently powers the vast majority of websites [12], and it is also easily available on any server, including shared hosting options. PHP also has a large community, with many packages available. As such, E-Conf is built on PHP.

As it is common practice in the PHP world, the project dependencies are managed through Composer[1].

To make development easier, and save a few hours and problems, the **Laravel** framework[2] was used. Laravel is currently the most popular PHP framework [13], which results in it having a large community and, as a result, many components already built for the framework. Among the core framework and packages, Laravel offers:

Authentication Authentication is a common requirement on web apps, so Laravel includes a complete authentication (and authorization) system.

Two-factor authentication Security and privacy are very important, so there is a Laravel package implementing 2FA.

Multi language Laravel allows developers to store their strings in language files, making Laravel easily localizable. [3]

Routing Developers can easily maps routes to resources using the framework.

Logging and debugging Through the framework's core and the Debug bar package, Laravel is easy to debug.

[1] `http://getcomposer.org`

[2] `http://laravel.com`

[3] To help with the translation process, E-Conf was translated using Crowdin. The project page is at `https://crowdin.com/project/econf`.

Events The Eventy package allows the use of hooks, stated in 3.2.1 as a design option.

Templates Laravel comes with the Blade template engine, which helps to keep logic and presentation apart.

Form validation To make sure the users' input arrives as expected, forms should be validated before working with the data. Laravel includes all the logic needed for this.

Laravel also suggests (but does not enforce) a MVC pattern, which was used.

Regarding the front-end, LESS[4] was used to pre-process CSS. LESS is one of the most popular CSS pre-processors and extends its syntax to include variables, mixins, nested rule sets, etc. Similarly to the back-end, front-end dependencies are managed through Bower[5].

To store the app data, a database was needed. A relational database was better suited to the presented data models, and so MySQL was used as database engine.

4.1.2 Multitenancy

The multitenancy needed for E-Conf was different from the existing tenancy packages, as in this project it is optional. This feature needed to be abstracted so that it could be ignored when programming the system.

The implementation of this feature was greatly supported by Laravel's features.

First, the system administrator decides if the installation will hold one or multiple conferences in the `config/econf.php` file. This can not be switched through the User Interface (UI) so that it is not switched by accident.

After that, the program needs to know which conference is currently being accessed. This is acheived through a middleware[6]. The created middleware runs before the majority of the routes, when multitenancy is on, and analyses the first URL parameter. If this parameter is the slug of one of the conferences in the system, the system stores the conference details and removes the parameter of the query, so that it can be parsed the same way independently of tenancy.

Knowing which conference is being accessed, the main step is to alter the queries to the database. The `SELECT` queries are altered through a query scope[7] and the `CREATE` queries through a model event[8], both packaged in a PHP trait that is then applied to the models that shall be multitenant.

Finally, some helper functions were created to have similar function to some Laravel helpers but taking into account (possible) multitenancy. As an example, a `m_action` function was created to complement `action`, a function that generates the URL for a specified controller method.

[4] http://lesscss.org

[5] http://bower.io

[6] https://laravel.com/docs/5.2/middleware

[7] https://laravel.com/docs/5.2/eloquent#query-scopes

[8] https://laravel.com/docs/5.2/eloquent#events

4.2 App Module

This module is the core of E-Conf, and as it is the only one able to stand by itself - all other modules are installed on this app, adding them as dependencies through Composer. It is a Laravel app (built from the `laravel/laravel` repository).

The app UI is split in two parts, public page and administration. The UI for both parts can be seen on figure 4.1.

The code for this module is available at `https://gitlab.com/econf/econf`.

Public page

The public page part includes the login and registration pages, as well as the profile and security options (displayed in figure 4.2). The menu that allows to change E-Conf language is also on the public page.

The included authentication flow (Login form → Validate login → Home) had to be modified to include 2FA for those who opt-in for it (Login form → Validate login → 2FA form → Validate 2FA → Home).

This module also includes the view with the public layout, for other modules to extend to add their own public pages. (Like the submission form described in section 4.3.)

Administration

The administration area is only accessible to users who belong to a committee, and it is locked through a middleware.

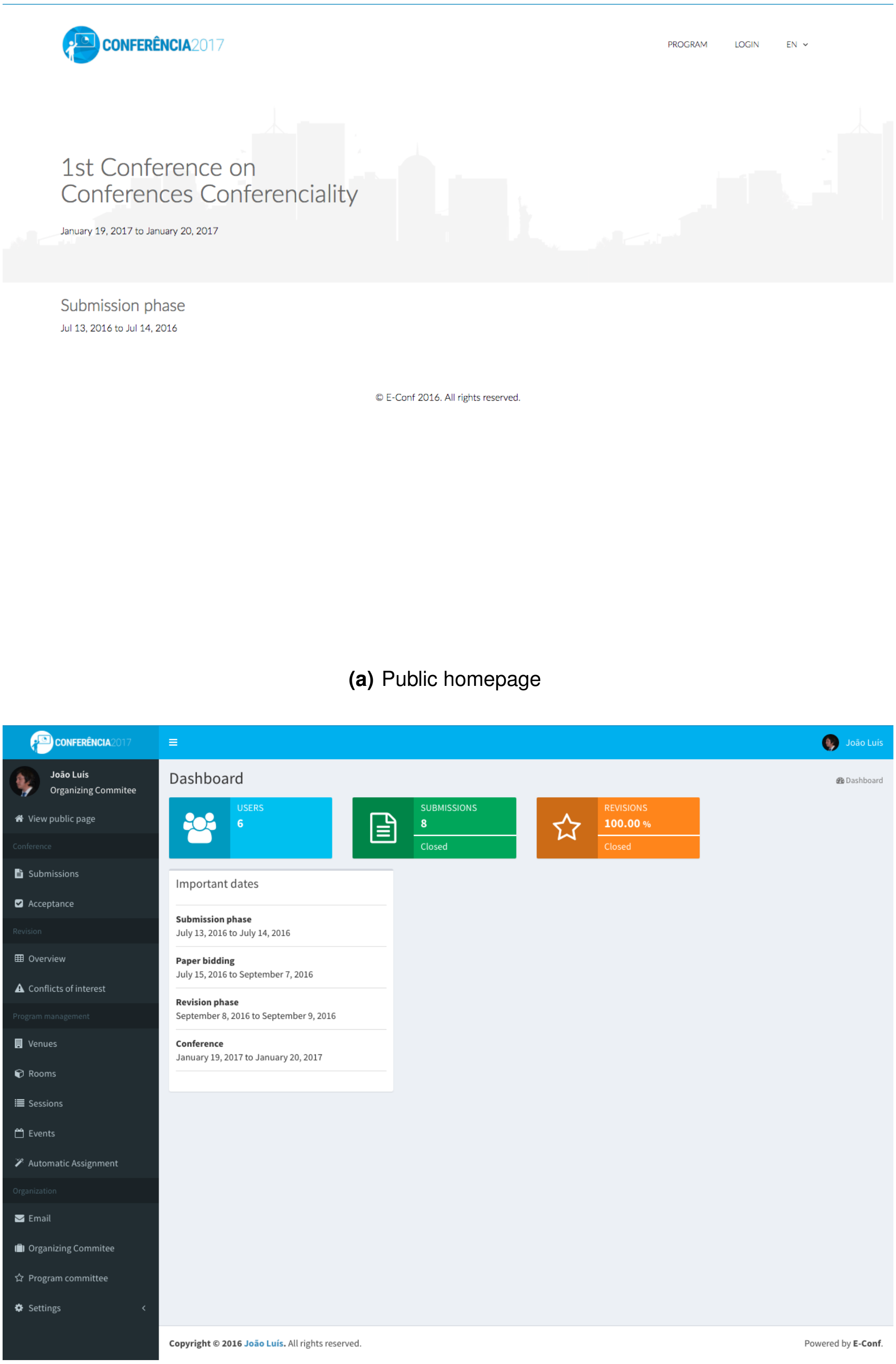
The administration allows members of the organizing committee to configure the conference, including branding. These options are stored using the Laravel Settings package.

Like with the public page, the view with the administration layout is included, for other modules to add their pages.

Network administration

If multitenancy is enabled, the core app will have a third part - the network administration. From this panel, located at `/admin/`, the network administrator can add and remove conferences.

To distinguish network users from conference users, they are stored in the database with the `conference_id` set to `NULL`.



CONFERÊNCIA2017

PROGRAM LOGIN EN

1st Conference on
Conferences Conferenciality

January 19, 2017 to January 20, 2017

Submission phase

Jul 13, 2016 to Jul 14, 2016

© E-Conf 2016. All rights reserved.

(a) Public homepage

CONFERÊNCIA2017 ≡ João Luís

João Luís
Organizing Commitee

- View public page
- Conference
- Submissions
- Acceptance
- Revision
- Overview
- Conflicts of interest
- Program management
- Venues
- Rooms
- Sessions
- Events
- Automatic Assignment
- Organization
- Email
- Organizing Commitee
- Program committee
- Settings

Dashboard

Dashboard

USERS 6

SUBMISSIONS 8 Closed

REVISIONS 100.00 % Closed

Important dates

Submission phase
July 13, 2016 to July 14, 2016

Paper bidding
July 15, 2016 to September 7, 2016

Revision phase
September 8, 2016 to September 9, 2016

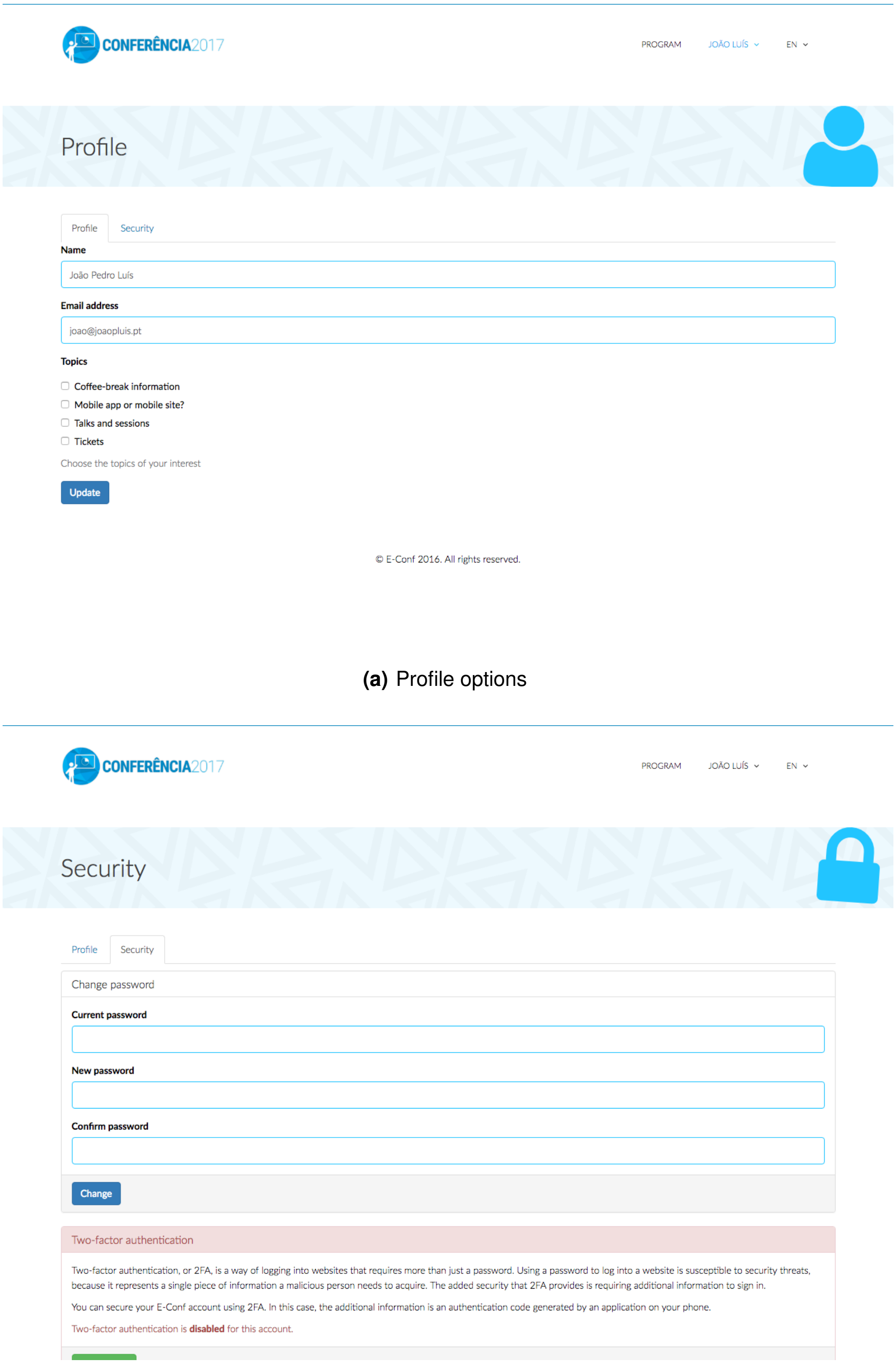
Conference
January 19, 2017 to January 20, 2017

Copyright © 2016 João Luís. All rights reserved.

Powered by **E-Conf**.

(b) Administration homepage

Figure 4.1: App User Interface



Profile

Name

João Pedro Luís

Email address

joao@joaopluis.pt

Topics

□ Coffee-break information
□ Mobile app or mobile site?
□ Talks and sessions
□ Tickets

Choose the topics of your interest

© E-Conf 2016. All rights reserved.

(a) Profile options

Security

Change password

Current password

New password

Confirm password

Two-factor authentication

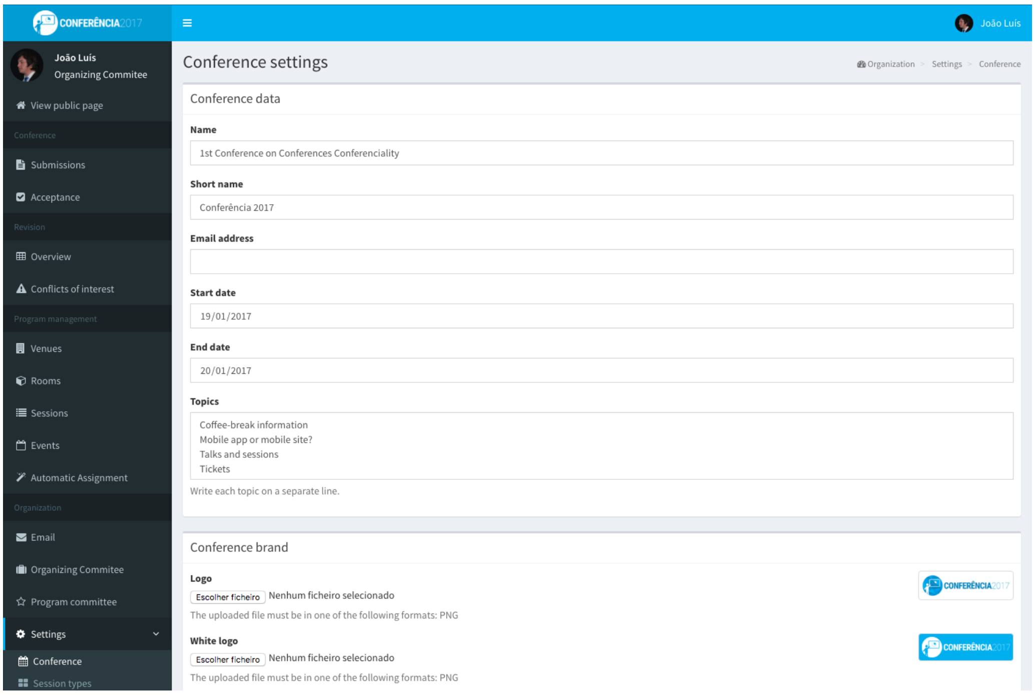
Two-factor authentication, or 2FA, is a way of logging into websites that requires more than just a password. Using a password to log into a website is susceptible to security threats, because it represents a single piece of information a malicious person needs to acquire. The added security that 2FA provides is requiring additional information to sign in.

You can secure your E-Conf account using 2FA. In this case, the additional information is an authentication code generated by an application on your phone.

Two-factor authentication is **disabled** for this account.

(b) Security options

Figure 4.2: User options screens



Conference settings

Conference data

Name

1st Conference on Conferences Conferenciality

Short name

Conferência 2017

Email address

Start date

19/01/2017

End date

20/01/2017

Topics

Coffee-break information
Mobile app or mobile site?
Talks and sessions
Tickets

Write each topic on a separate line.

Conference brand

Logo

Escolher ficheiro Nenhum ficheiro selecionado

The uploaded file must be in one of the following formats: PNG

White logo

Escolher ficheiro Nenhum ficheiro selecionado

The uploaded file must be in one of the following formats: PNG

Figure 4.3: Conference settings screen

4.2.1 Extension points

The main app module has the majority of the extension points, which will be used by the other modules. These points are listed below:

public.index This action is called when displaying the homepage and allows packages to add content to the conference front page. (Action)

public.menu It is called when creating the public page menu, and it is responsible for adding items to the menu. (Action)

public.assets This filter allows packages to add assets (CSS or JS) to be loaded on a public page. (Filter)

me.profile.show Similarly to public.index, allows modules to add options to the user profile page. (Action)

me.profile.save It is called when saving the user profile, to store the options added in the action above. (Action)

dashboard.data The data for the boxes on the top of the administration dashboard. (Filter)

dashboard.calendar The calendar dates to be displayed on the administration dashboard.

admin.menu Similar to public.menu, but for the administration menu. (Action)

admin.menu.organizing Allows to add items to the ”Organization” section of the administration menu. (Action)

admin.menu.settings Allows to add items to the ”Settings” submenu of the administration menu. (Action)

admin.assets This filter allows packages to add assets (CSS or JS) to be loaded on an administration page. (Filter)

admin.email.groups The list of groups for the email system. (Filter)

admin.email.templates The templates to be used on the email system. (Filter)

admin.email.vars The variables to replace on the email body message. (Filter)

invite.accept.{invite_type} Called when an invite of the type specified is accepted, allowing modules to create their own invites. (Action)

invite.messages.{invite_type} The messages to describe a invite of the specified type. (Filter)

file.can Whether an user can access a file or no. (Filter)

file.transform_name Transforms the requested file name to the stored file name. (Filter)

4.3 Submissions Module

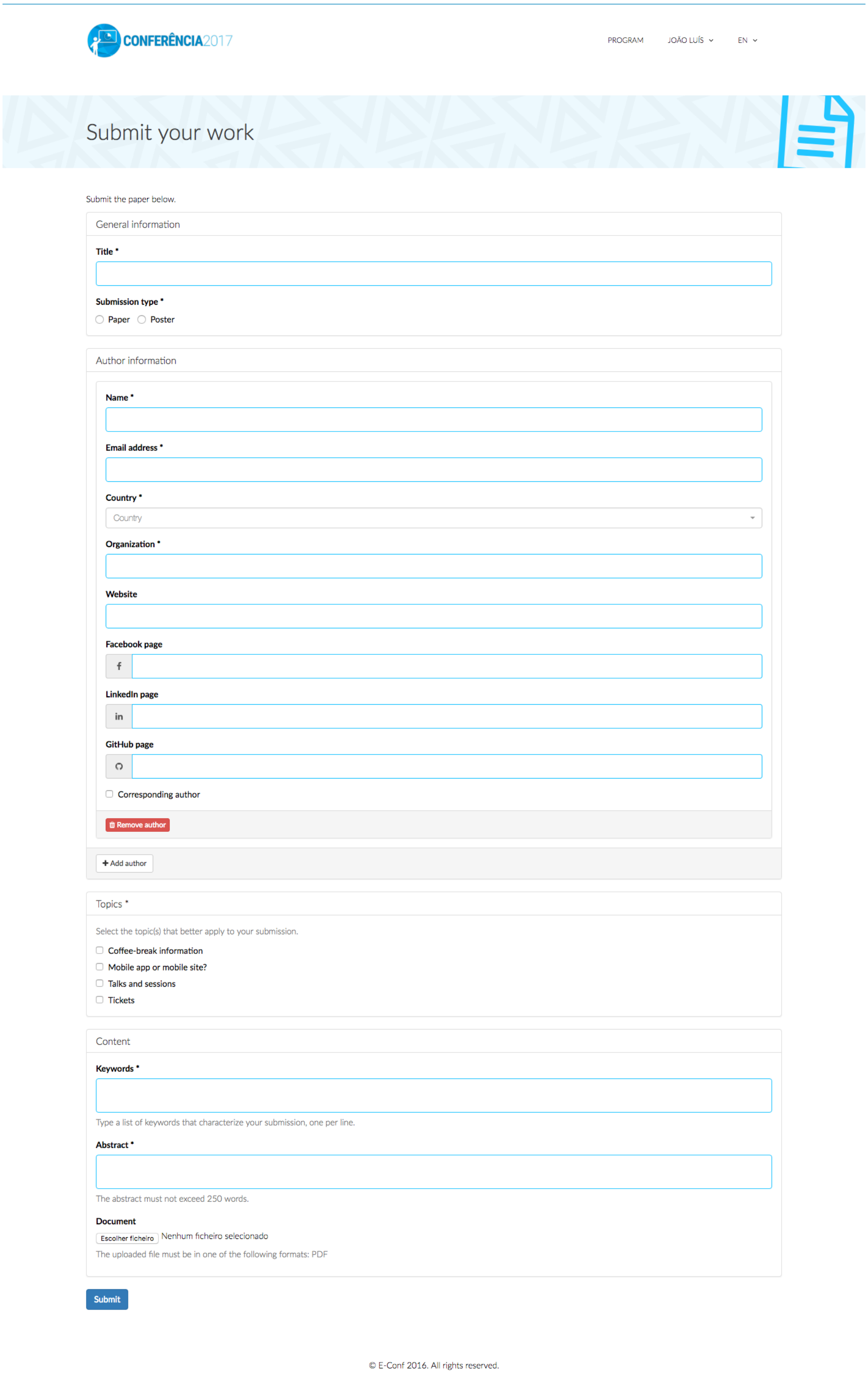
As it was said in section 3.4, this module main responsibility is to receive work submissions from authors.

Submissions can be done by any registered user through the conference public page, during the defined period. The main component of this module is the submission form, which is customizable through the Submission Settings page. The UI for the form is shown on figure 4.4.

A submission can have as many authors as needed, as the forms allows to add and remove authors. The list of fields available for each author is configurable through the settings page and extensible through a filter.

After submitting their work, an user can see their submissions and edit them until the phase is over. The submissions are stored with a sequential ID, however, showing this ID to the users can be a security breach. So, the actual ID is masked as a Hashid[9] and this is the one displayed to end users.

[9] http://hashids.org



Submit your work

Submit the paper below.

General information

Title *

Submission type *

○ Paper ○ Poster

Author information

Name *

Email address *

Country *

Country

Organization *

Website

Facebook page

LinkedIn page

GitHub page

□ Corresponding author

Remove author

+ Add author

Topics *

Select the topic(s) that better apply to your submission.

□ Coffee-break information

□ Mobile app or mobile site?

□ Talks and sessions

□ Tickets

Content

Keywords *

Type a list of keywords that characterize your submission, one per line.

Abstract *

The abstract must not exceed 250 words.

Document

Escolher ficheiro Nenhum ficheiro selecionado

The uploaded file must be in one of the following formats: PDF

Submit

© E-Conf 2016. All rights reserved.

Figure 4.4: Example of a submission form

This module also handles acceptance. When the acceptance period is open - after the submission phase ended - the Acceptance page becomes accessible on the administration panel, allowing an organizer to pick the accepted submissions.

The organizer can also choose to ask for additional information when a submission is accepted, like a final document or the presenter. This data can then be filled through the public page.

To email authors, the module adds some email groups to E-Conf:

- All authors;
- Authors who submitted for a session type (for each type);
- Authors whose submissions were accepted;
- Authors whose submissions were not accepted.

The code for this module is available at https://gitlab.com/econf/submissions.

4.3.1 Extension points

submissions.author-fields Allows to change the list of fields available for each author. (Filter)

submissions.acceptance_open Changes if the acceptance period is open. (Filter)

4.4 Reviews Module

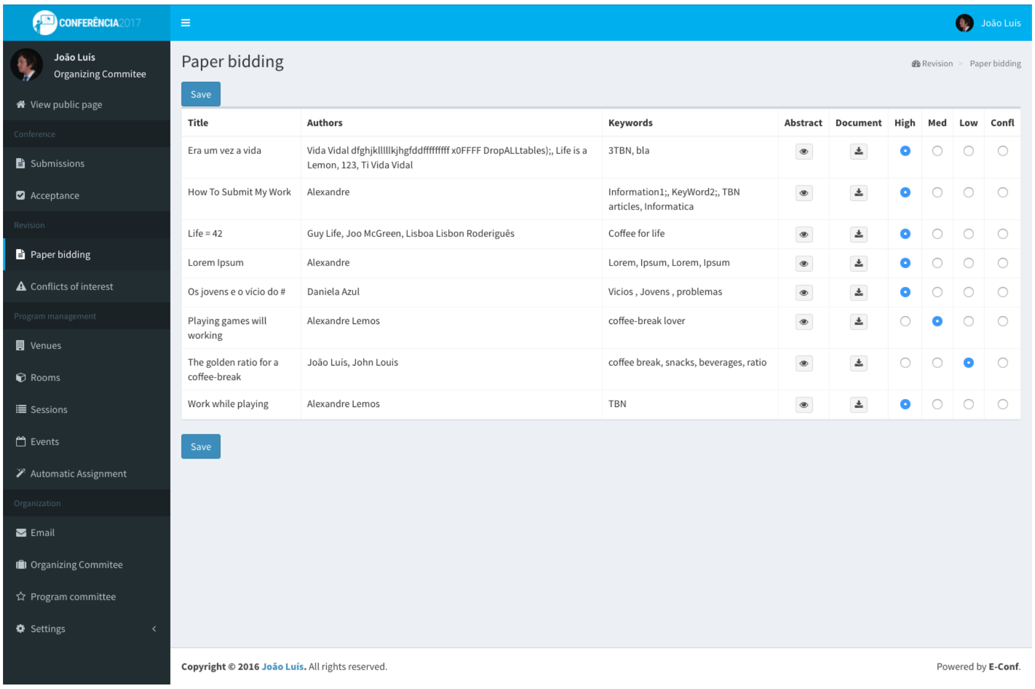
The reviews module is, as stated before, divided in three parts: bidding, assignment and revision. These parts are described below.

The code for this module is available at https://gitlab.com/econf/reviews.

4.4.1 Bidding

Reviewers can bid on submissions through the Bidding page in the administration panel. This page is only accessible by members of the program committee.

In this page, reviewers view a list with all submissions and are able to choose their bid - High, Medium, Low or Conflict. If they want, reviewers can change their bids until the end of the bidding phase. The UI for this page is shown on figure 4.5.



Paper bidding

Title	Authors	Keywords	Abstract	Document	High	Med	Low	Confl
Era um vez a vida	Vida Vidal dfghjklllllkjhgfddffffffffff x0FFFF DropALLtables);, Life is a Lemon, 123, Ti Vida Vidal	3TBN, bla	👁	⬇	●	○	○	○
How To Submit My Work	Alexandre	Information1;, KeyWord2;, TBN articles, Informatica	👁	⬇	●	○	○	○
Life = 42	Guy Life, Joo McGreen, Lisboa Lisbon Roderiguês	Coffee for life	👁	⬇	●	○	○	○
Lorem Ipsum	Alexandre	Lorem, Ipsum, Lorem, Ipsum	👁	⬇	●	○	○	○
Os jovens e o vício do #	Daniela Azul	Vicios , Jovens , problemas	👁	⬇	●	○	○	○
Playing games will working	Alexandre Lemos	coffee-break lover	👁	⬇	○	●	○	○
The golden ratio for a coffee-break	João Luís, John Louis	coffee break, snacks, beverages, ratio	👁	⬇	○	○	●	○
Work while playing	Alexandre Lemos	TBN	👁	⬇	●	○	○	○

Copyright © 2016 João Luís. All rights reserved.

Powered by E-Conf.

Figure 4.5: Bidding page screen

4.4.1.A Conflicts of interest

E-Conf is, by default, able to detect conflicts between a submission authors and a reviewer using two parameters: name and organization.

Name E-Conf compares the author and the review names using PHP built-in `similar_text` function, marking a conflict if this similarity is above 75 %.

Organization Usually organization names can be written in many ways (as an example, Instituto Superior Técnico can also be written as IST or Técnico Lisboa). So, to check organization conflicts, the email suffix is compared, and a conflict is marked if the suffix is the same and it belongs to the Swot list[10], a list of domains that belong to colleges or universities.

A hook placed in the conflict detection code allows packages to add new ways to detect conflicts.

Although these conflicts are detected, a chair of the program committee must approve them for them to be enforced. Chairs can also manually add conflicts.

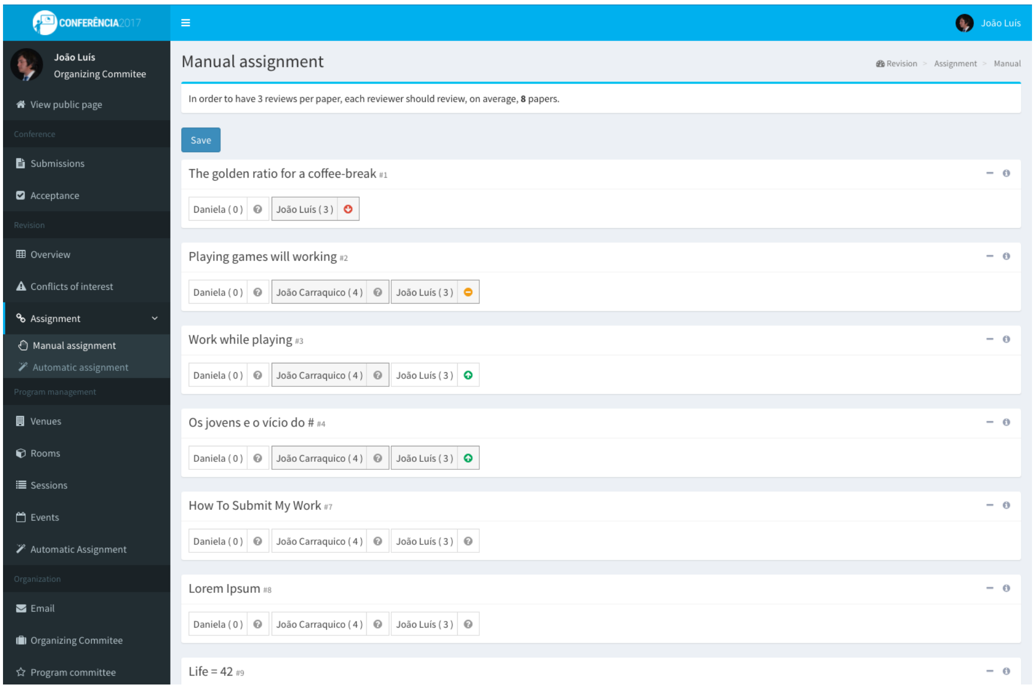
4.4.2 Assignment

The assignment phase occurs automatically between the bidding and review phases. During this phase, the chairs can assign the submissions to reviewers.

[10] https://github.com/leereilly/swot

There are two new pages in the administration panel: manual and automatic assignment.

The manual assignment page lists all the submissions and, for each submission, all the reviewers that can review it, along with the value of the bid they made and the number of submissions assigned to them. This UI, shown in figure 4.6, displays a large amount of information in an easy way to read.



Manual assignment

In order to have 3 reviews per paper, each reviewer should review, on average, **8** papers.

The golden ratio for a coffee-break #1

Daniela (0) | João Luís (3)

Playing games will working #2

Daniela (0) | João Carraquico (4) | João Luís (3)

Work while playing #3

Daniela (0) | João Carraquico (4) | João Luís (3)

Os jovens e o vício do # #4

Daniela (0) | João Carraquico (4) | João Luís (3)

How To Submit My Work #7

Daniela (0) | João Carraquico (4) | João Luís (3)

Lorem Ipsum #8

Daniela (0) | João Carraquico (4) | João Luís (3)

Life = 42 #9

Figure 4.6: Manual assignment screen

The automatic assignment page lists all the available algorithms. After choosing an algorithm, its results will be presented in a page similar to the manual assignment results, without saving them. A chair may then save the results and will be redirected to the manual assignment page, where they can adjust the assignments.

New assignment algorithms can be added by external packages.

4.4.2.A AREA algorithm

The AREA (Automatic REviewer Assignment) algorithm is included with E-Conf. It takes into account the reviewers' bids and the conflicts of interest and tries to maximize the reviewers' satisfaction.

The algorithm is described in algorithm 4.1.

4.4.3 Revision

After the submissions are assigned, reviewers must review their assigned submissions.

Algorithm 4.1: AREA algorithm

```
Function AssignForValue (bidValue)
    $valueReviewers \leftarrow Bids.withValue(bidValue).reviewers.sortBy(numBids)$;
    foreach $valueReviewers$ as $reviewer$ do
        $reviewerBids \leftarrow reviewer.bidsWithValue(bidValue).submissions.sortBy(numBids)$;
        foreach $reviewerBids$ as $submission$ do
            $subs\_revs[submission.id].add(reviewer.id)$;
            $revs\_subs[reviewer.id].add(submission.id)$;
            if $subs\_revs.length \geqslant reviews\_per\_submission$ then
                $submissions.remove(submission)$;
            if $revs\_subs.length \geqslant submissions\_per\_reviewer$ then
                $reviewers.remove(reviewer)$;
                break
Function AssignRemainingSubmissions
    foreach $submissions$ as $submission$ do
        $submReviewers \leftarrow ProgramCommittee.canReview(submission).sortBy(topicMatch)$;
        foreach $submReviewers$ as $reviewer$ do
            $subs\_revs[submission.id].add(reviewer.id)$;
            $revs\_subs[reviewer.id].add(submission.id)$;
            if $subs\_revs.length \geqslant reviews\_per\_submission$ then
                $submissions.remove(submission)$;
            if $revs\_subs.length \geqslant submissions\_per\_reviewer$ then
                $reviewers.remove(reviewer)$;
                break
$submissions \leftarrow Submissions.all()$;
$reviewers \leftarrow ProgramCommittee.members()$;
$bids \leftarrow Bids.all()$;
$subs\_revs \leftarrow []$;
$revs\_subs \leftarrow []$;
$reviews\_per\_submission \leftarrow getFromSettings()$;
$submissions\_per\_reviewer \leftarrow ceil(\frac{submissions.length \times reviews\_per\_submission}{reviewers.length})$;
$AssignForValue(High)$;
$AssignForValue(Medium)$;
$reviewers \leftarrow reviewers.sortBy(numAssignments)$;
$AssignRemainingSubmissions()$;
if not $submissions.empty$ then
    $reviewers \leftarrow ProgramCommittee.members()$;
    $AssignRemainingSubmissions()$;
return $subs\_revs$;
```

A reviewer may review the submissions assigned to them through the administration panel. The review form allows reviewers to score the submission from 1 to 5 on the parameters defined on the Review Settings page and to write text comments to the program committee and the author. A reviewer may change his review any time until the revision phase closes.

Every time a review is saved, its score is computed and stored to the database, so they are not computed each time it is viewed.

The UI for this form is shown on figure 4.7(a).

Chairs are then able to see each individual review. This screen is shown on figure 4.7(b).

Reviewers may also delegate their reviews to other users (subreviewers) through the reviews list. These users do not need to be a part of the program committee.

When a review is delegated, the original reviewer is still stored, and as such they can view the reviews they delegated.

4.4.4 Extension points

conflict.check Allows to add new conflict detection techniques. (Action)

reviews.assignment.algorithms The list of the automatic assignment algorithms. (Filter)

4.5 Program Management Module

The program management module is responsible for managing and showing the conference program.

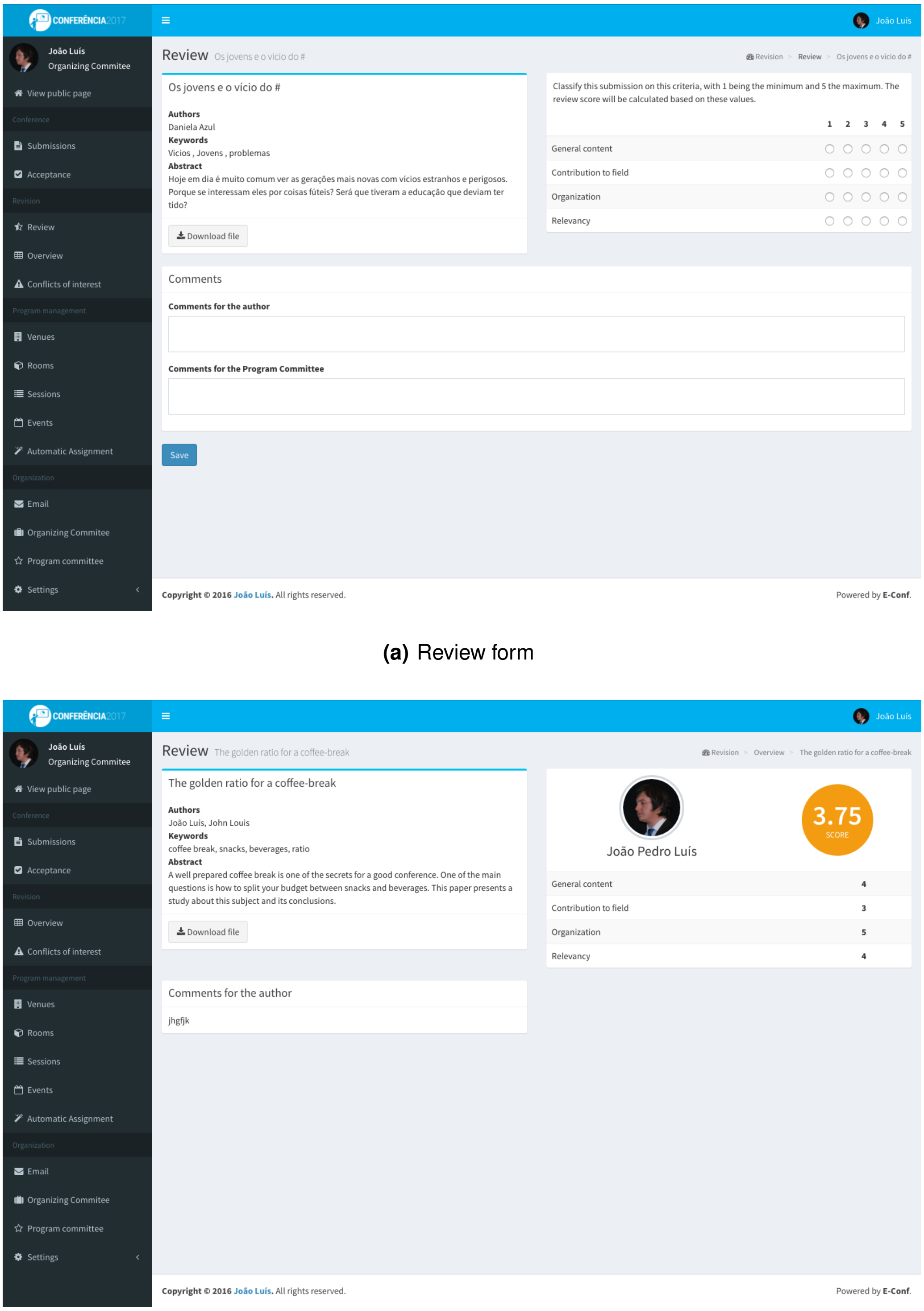
Organizers can manage venues and rooms through the Program section of the administration panel. Organizers can then create events through the panel. Events have the start and end times, the room where they take place and other details.

Organizers can also create sessions. Sessions are special events where submissions will be presented. Each session has a session type and a topic, along with its submissions.

As it was said in 3.6, each session has a corresponding event. This allows queries to the program to be easier and more efficient.

E-Conf also creates a program page on the conference public page. This page generates an overview page, shown in figure 4.8 and a page for each event, shown on figure 4.9.

The code for this module is available at `https://gitlab.com/econf/program-management`.



Review Os jovens e o vício do #

Os jovens e o vício do #

Authors
Daniela Azul
Keywords
Vicios , Jovens , problemas
Abstract
Hoje em dia é muito comum ver as gerações mais novas com vícios estranhos e perigosos. Porque se interessam eles por coisas fúteis? Será que tiveram a educação que deviam ter tido?

Classify this submission on this criteria, with 1 being the minimum and 5 the maximum. The review score will be calculated based on these values.

	1	2	3	4	5
General content	○	○	○	○	○
Contribution to field	○	○	○	○	○
Organization	○	○	○	○	○
Relevancy	○	○	○	○	○

Comments

Comments for the author

Comments for the Program Committee

Copyright © 2016 João Luís. All rights reserved.

Powered by **E-Conf**.

(a) Review form

Review The golden ratio for a coffee-break

The golden ratio for a coffee-break

Authors
João Luís, John Louis
Keywords
coffee break, snacks, beverages, ratio
Abstract
A well prepared coffee break is one of the secrets for a good conference. One of the main questions is how to split your budget between snacks and beverages. This paper presents a study about this subject and its conclusions.

João Pedro Luís

3.75
SCORE

General content	4
Contribution to field	3
Organization	5
Relevancy	4

Comments for the author

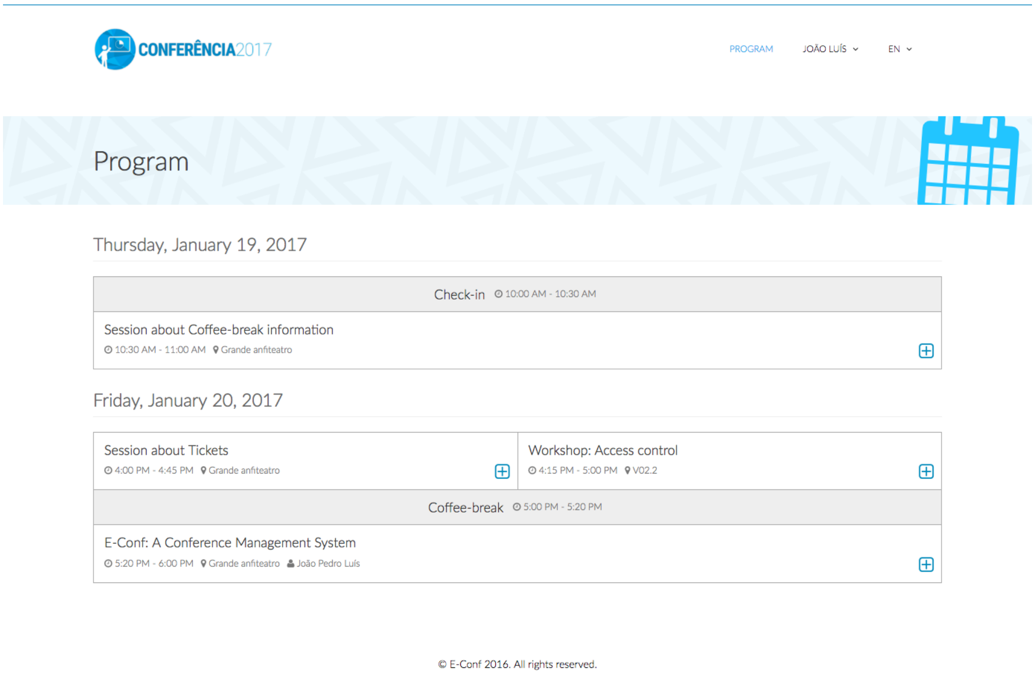
jhgfjk

Copyright © 2016 João Luís. All rights reserved.

Powered by **E-Conf**.

(b) Chair review view

Figure 4.7: Reviews User Interface



Program

Thursday, January 19, 2017

Check-in 10:00 AM - 10:30 AM

Session about Coffee-break information
10:30 AM - 11:00 AM Grande anfiteatro

Friday, January 20, 2017

Session about Tickets
4:00 PM - 4:45 PM Grande anfiteatro

Workshop: Access control
4:15 PM - 5:00 PM V02.2

Coffee-break 5:00 PM - 5:20 PM

E-Conf: A Conference Management System
5:20 PM - 6:00 PM Grande anfiteatro João Pedro Luís

© E-Conf 2016. All rights reserved.

Figure 4.8: Program overview

4.5.1 Assignment

Organizers can also assign submissions to sessions though this module. This assignment can be done manually or automatically.

The manual assignment can be done from the session page on the administration panel. From this page organizers can add submissions to a session, and set their order.

Automatic assignment works similarly to the reviews automatic assignment, described in 4.4.2. When they access the Automatic assignment page, organizers are presented with a list of algorithms and, after an algorithm is picked, the results of that algorithm are presented. If saved, these results can then be changed on the session page.

New assignment algorithms can be added by external packages.

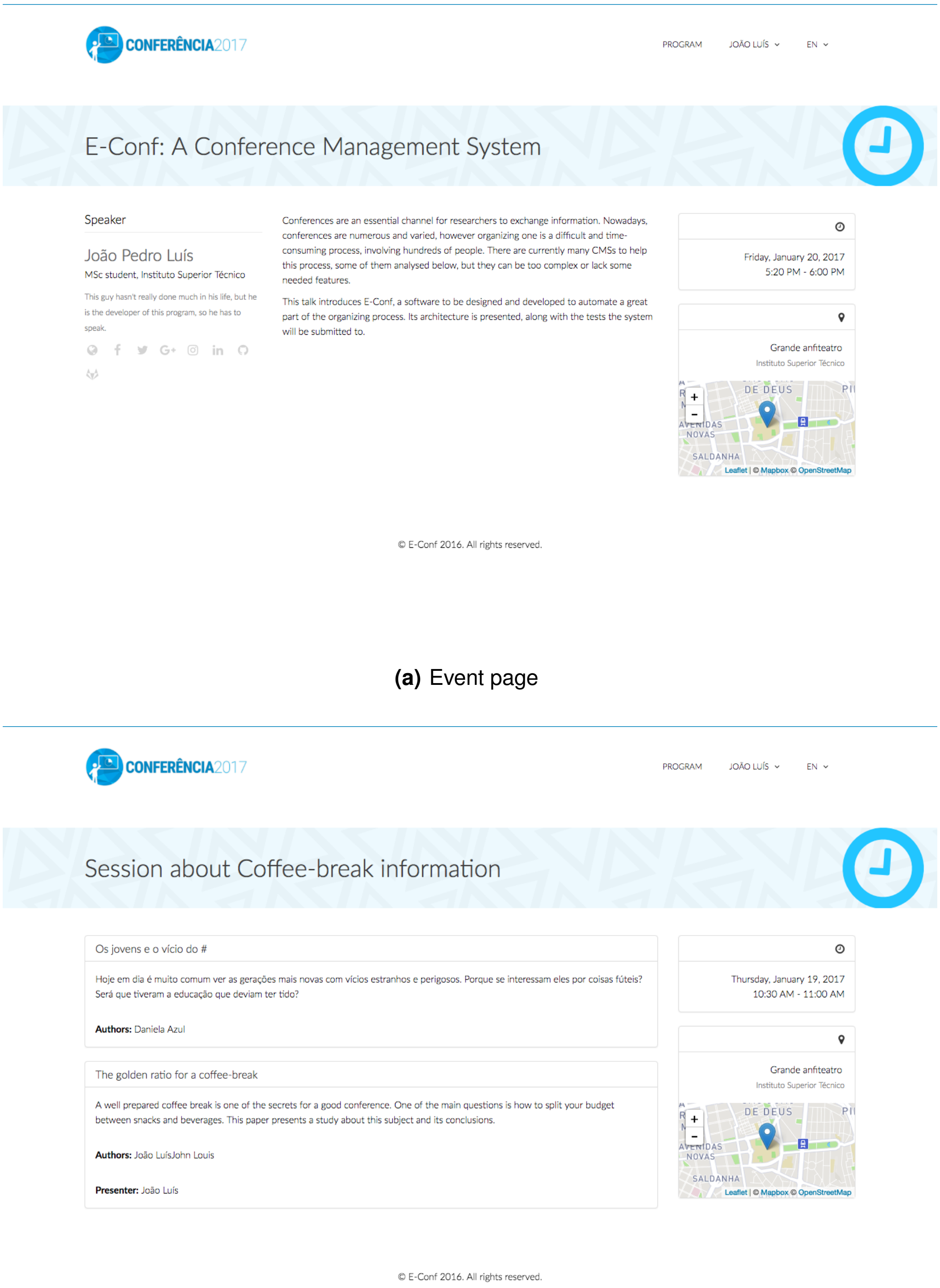
4.5.1.A Chronos algorithm

The Chronos algorithm is included with E-Conf. It takes into account the scores and the topics.

The algorithm is described in algorithm 4.2.

4.5.2 Extension points

program.assignment.algorithms The list of the automatic assignment algorithms. (Filter)



E-Conf: A Conference Management System

Speaker

João Pedro Luís

MSc student, Instituto Superior Técnico

This guy hasn't really done much in his life, but he is the developer of this program, so he has to speak.

Conferences are an essential channel for researchers to exchange information. Nowadays, conferences are numerous and varied, however organizing one is a difficult and time-consuming process, involving hundreds of people. There are currently many CMSs to help this process, some of them analysed below, but they can be too complex or lack some needed features.

This talk introduces E-Conf, a software to be designed and developed to automate a great part of the organizing process. Its architecture is presented, along with the tests the system will be submitted to.

Friday, January 20, 2017
5:20 PM - 6:00 PM

Grande anfiteatro
Instituto Superior Técnico

© E-Conf 2016. All rights reserved.

(a) Event page

Session about Coffee-break information

Os jovens e o vício do #

Hoje em dia é muito comum ver as gerações mais novas com vícios estranhos e perigosos. Porque se interessam eles por coisas fúteis? Será que tiveram a educação que deviam ter tido?

Authors: Daniela Azul

The golden ratio for a coffee-break

A well prepared coffee break is one of the secrets for a good conference. One of the main questions is how to split your budget between snacks and beverages. This paper presents a study about this subject and its conclusions.

Authors: João LuísJohn Louis

Presenter: João Luís

Thursday, January 19, 2017
10:30 AM - 11:00 AM

Grande anfiteatro
Instituto Superior Técnico

© E-Conf 2016. All rights reserved.

(b) Session page

Figure 4.9: Program User Interface

Algorithm 4.2: Chronos algorithm

```
$submissions \leftarrow Submissions.accepted().groupBy(topic)$;
$sessions \leftarrow Sessions.all()$;
$sessions\_submissions \leftarrow []$;
while not $submissions.isEmpty$ or not $sessions.isEmpty$ do
    $submissions.sortByDesc(length)$;
    $sessions.sortByDesc(remainingSubmissions)$;
    $session \leftarrow sessions.shift()$;
    $chosen\_submissions \leftarrow submissions.first.slice(0, session.remainingSubmissions)$;
    $submissions.remove(chosen\_submissions)$;
    if $session.full$ then
        $sessions.remove(session)$;
foreach $sessions\_submissions$ as $session$ do
    $session.shuffle()$;
return $sessions\_submissions$;
```

4.6 Conclusion

This chapter described the implementation of the system. Four different modules were implemented: the main app, submissions, reviews and program management.

E-Conf was implemented using HTML, CSS and JavaScript for the client-side and PHP for the server-side, being the most used language. The Laravel framework was used to avoid solving common problems.

The app module is a Laravel app and is the foundation of the system, with the other modules being added as dependencies using Composer. This module exposes two distinct sections, public page and administration panel, and additionally exposes a network administration panel in multitenant installations. It also exposes almost 20 extension points, to allow other modules to hook into it.

The submissions module handles the submission of work by authors, through a configurable submission form in the public page. It also handles submission acceptance and the submission of additional information for accepted submissions.

The reviews module handles the three parts of the review process: bidding, assignment and revision. One of the strengths of this module is the ability to assign reviewers to submissions automatically, through the algorithm included or external ones. The included algorithm, AREA, was also detailed.

The program management module allows organizers to build the conference program and make it available to everyone through the public page. Similarly to the reviews module, submissions can be assigned to sessions automatically. The Chronos algorithm is also detailed.

5

Evaluation

Contents

Like every other system, E-Conf needs to be tested and evaluated, to check if it is ready to use in a real world scenario.

This chapter describes the tests performed to evaluate E-Conf, along with their results and the respective conclusions.

5.1 Algorithm tests

Both the algorithms created, described in sections 4.4.2.A and 4.5.1.A, were tested and evaluated.

To make the results more accurate, the algorithms were tested with real data, from the 12th Conference on PhD Research in Microelectronics and Electronics (PRIME 2016).

To import the data and compute the values, a fifth module was created (Evaluation). This module imports data from a Microsoft Excel spreadsheet, exported from EasyChair, and imports submissions, topics, program committee members and bids. The other details of this module, specific for each algorithm, are described in the sections below.

5.1.1 Reviewers assignment algorithm (AREA)

Two metrics were used to test this algorithm: reviewer satisfaction and topic matching. These metrics were computed after importing all the data from PRIME and using AREA to assign submissions to reviewers, configured to assign three reviewers per submission.

The results were then compared with the assignments EasyChair made for PRIME.

Reviewers satisfaction

Reviewer satisfaction describes how much a reviewer's assignments match their bids. The method used to compute this value is described on 5.1.

Algorithm 5.1: Reviewer satisfaction computation

```
$total \leftarrow reviewer.assignments.count() \times 2$;
$value \leftarrow 0$;
foreach $reviewer.bids \cap reviewer.assignments$ as $bid$ do
    if $bid.value == high$ then
        $value \leftarrow value + 2$;
    else if $bid.value == medium$ then
        $value \leftarrow value + 1$;
$satisfaction \leftarrow value/total$;
```

The Evaluation module computes this metric for each reviewer and then uses that data to present some statistics.

Taking into account only the reviewers who have done their bidding, the average reviewer satisfaction is 84.69%. While this is a good rate, there is clearly space for improvement.

Topic matching

After assigning submissions to the users who bided, AREA assigns the remaining submissions taking into account the submission topics and the reviewer topics. The method used to compute this metric is described on 5.2.

Algorithm 5.2: Topic matching computation

```
$total \leftarrow reviewer.assignments.count()$;
$value \leftarrow 0$;
foreach $reviewer.assignments$ as $assignment$ do
    $matchingTopics = assignment.submission.topics \cap reviewer.topics$;
    $value \leftarrow matchingTopics.count()/assignment.submission.topics.count()$;
$matching \leftarrow value/total$;
```

Similarly to the previous metric, the Evaluation module computes this metric for each reviewer and uses this data to present some statistics.

Considering all the reviewers who have not done their bidding, the average topic matching is 8.12%. However, if only reviewers who chose more than one topic are considered, the average topic matching raises to 82.56%.

These results show that the algorithm performs better if the reviewers choose more than one topic of interest, because submissions have usually more than one topic.

Comparison with PRIME

Another interesting evaluation to consider is the comparison between the AREA results and the actual assignments from PRIME, both for the metrics above and the assignments done.

Regarding the assignments, 55.86% of the assignments done by AREA were also done for PRIME. This value, by itself, does not say much, although it is still interesting that both methods share more than half of the assignments.

To get the values for reviewers satisfaction and topic matching, the assignments done by AREA were replaced on E-Conf by the ones done for PRIME, and the same computations were done by the module. The reviewers satisfaction for this set of assignments, 81.96%, is pretty close to the value obtained by AREA. However, the topic matching value is greatly inferior: 2.19% for all reviewers with no bids and only 10% for the reviewers with more than one topic.

5.1.2 Session assignment algorithm (Chronos)

The Chronos algorithm is a very simple algorithm that mainly takes into account topics. So, the metric used to evaluate this algorithm was the topic grouping, i.e., whether the submissions in a session shared a topic. The method used to compute this value is described on 5.3.

Algorithm 5.3: Topic grouping computation

```
$session\_topics \leftarrow []$;
foreach $session.submissions$ as $submission$ do
    foreach $submission.topics$ as $topic$ do
        $session\_topics[topic.id] \leftarrow session\_topics[topic.id] + 1$;
$major\_topic \leftarrow session\_topics.sort().last()$;
$grouping \leftarrow major\_topic/session.submissions.count()$;
```

To test the algorithm, the acceptance data from PRIME was imported, a program similar to PRIME's was created on E-Conf and the submissions assigned using Chronos.

The average topic grouping for these assignments were 94.55%. Closer inspection of individual sessions shown that, for the majority of the sessions, the common topic was the same topic. This shows that there is clearly room for improvement, even if taking only in account topics. For instance, topic pairs could be considered, to created sessions with closer submissions.

Other improvements can be done to this algorithm, as stated in 6.1.

5.2 Functionality tests

To test if the system could fulfil its main goal, it was run for the whole organization process, from the set up to the program generation.

To do these tests a mock conference was created - TEST 2016 - and organized using only E-Conf. All the needed data was generated with the help of the Faker package [1] and inserted automatically, but all the remaining interactions, like assigning submissions to reviewers or creating sessions, was done through the E-Conf UI.

First, the system generated 6 topics. Then, it was ordered to create 100 users and, along with those users, created 148 submissions. The program also generated a Program Committee with 78 members.

The system then generated the biddings. Only a random percentage of the reviewers would bid, each bid with the same probability of being high, medium, low or a conflict. 52 of the 78 program committee members made their bids.

With these bids done, the submissions were assigned to reviewers using AREA. Although E-Conf did the job (and with a reviewer satisfaction of 100%), it took a few minutes to complete it. After the

[1] https://github.com/fzaninotto/Faker

assignments were done, the reviews were generated, and the top 36 submissions were accepted. (To create 6 1-hour sessions).

The required sessions were created manually and then the submissions were assigned using Chronos.

Besides the struggle with time when storing the assignments, E-Conf was able to handle the whole process with no problems, and requiring minimal input from the organizer.

5.3 Stress tests

When this work was first planned, the scale of conferences E-Conf would take and the finer details were yet unknown, as the analysis was not done and the architecture not defined. By that time, stress tests were thought needed.

However, the development of the work showed that they were not needed, because the scenarios they would test - e.g., many submissions at the same time - do not happen when actually organizing a conference. As a result, stress tests were not done.

5.4 Conclusion

The evaluation performed both tested E-Conf with real and simulated scenarios, for different purposes.

The AREA algorithm had very good results, achieving an average reviewer satisfaction of approximately 85% for PRIME data. The Chronos algorithm, on the other hand, had clearly room for improvement, although it was functional and makes an acceptable assignment.

When used from the beginning to the end of the organizing process, this time using simulated, random data, E-Conf shows no problem handling its tasks.

6

Conclusion

Contents

Conferences are an essential channel for the exchange of information, and as result there are hundreds of thousands of conferences every year. Organizing a conference is a long and hard process, involving hundreds of people. Most of the tasks are repetitive and, therefore, ideal to delegate to a computer. There are already several systems to solve this problem, Conference Management Systems, although they have some restrictions. The work shown in this document proposes E-Conf, a new CMS.

To find which features were required, other CMSs were analysed. Although there are some small differences among the set of features offered, they share the same overall set of features, which can be grouped in five main categories - submission, reviewing, pre-conference, registration and public page. The set of features on E-Conf, based on this analysis, covers three of this five areas.

E-Conf was implemented as a web-app, using the most used web technologies today, making E-Conf accessible to anyone with a web browser. The clear separation of the features in categories was used to group the functionality in modules, one for the main app and one for each category.

The evaluation performed showed that E-Conf can fulfil its purpose. The AREA algorithm evaluation also showed positive results (similar or better than PRIME's) for the used metrics. The Chronos algorithm, however, despite the good results in the metrics used, clearly needs improvement - or to be replaced.

6.1 Future Work

Two of the five main feature categories, registration and public page, are not covered by E-Conf. The addition of new modules focusing on these categories would make E-Conf a CMS able to cover the whole process, with no external tools required.

The assignment algorithms are another area that could be enhanced in the future. E-Conf includes two basic algorithms, but a thorough study on these aspects could produce much better algorithms, specially for the sessions assignment. This algorithm could take into account schedule conflicts and restrictions, room capacity and submission scores, topic pairs, etc.

Regarding UI and customization, E-Conf could benefit from a theming engine, allowing chairs to easily change their public page appearance. This is already possible if a module overrides most of the templates, but this process could be made easier.

Another feature that would open many doors for E-Conf is the creation of an Application Programming Interface (API). This could allow the integration of E-Conf in other systems and many other creative uses, e.g. a complete new interface for the system.

Bibliography

[1] "Ex ordo — award winning abstract management software," http://exordo.com/, (Visited on 14/Dec/2015).

[2] L. PricewaterhouseCoopers, "The economic significance of meetings to the us economy," 2014.

[3] "Easychair," http://www.easychair.org/, (Visited on 14/Dec/2015).

[4] "Openconf peer-review, conference and abstract management software system," http://www.openconf.com/, (Visited on 14/Dec/2015).

[5] "Edas: Editor's assistant," https://edas.info/doc/, (Visited on 14/Dec/2015).

[6] "Conftool: Conference and event management software," http://www.conftool.net/, (Visited on 14/Dec/2015).

[7] "Open source event manager," http://osem.io/, (Visited on 14/Dec/2015).

[8] "Welcome to epapers!" http://www.epapers.org/main/, (Visited on 14/Dec/2015).

[9] M. Ley, "Dblp: Some lessons learned," *Proc. VLDB Endow.*, vol. 2, no. 2, pp. 1493–1500, Aug. 2009. [Online]. Available: http://dx.doi.org/10.14778/1687553.1687577

[10] C. Caldera, R. Berndt, and D. W. Fellner, "Comfy –a conference management framework," *Inf. Serv. Use*, vol. 33, no. 2, pp. 119–128, Apr. 2013. [Online]. Available: http://dl.acm.org/citation.cfm?id=2596858.2596867

[11] N. D. Mauro, T. M. A. Basile, and S. Ferilli, "Grape: An expert review assignment component for scientific conference management systems." in *IEA/AIE*, ser. Lecture Notes in Computer Science, M. Ali and F. Esposito, Eds., vol. 3533. Springer, 2005, pp. 789–798. [Online]. Available: http://dblp.uni-trier.de/db/conf/ieaaie/ieaaie2005.html#MauroBF05

[12] "Usage of server-side programming languages for websites," https://w3techs.com/technologies/overview/programming_language/all, (Visited on 05/Oct/2016).

[13] B. Skvorc, “The best php framework for 2015: Sitepoint survey results,” http://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-results/, Mar. 2015, (Visited on 02/Jan/2016).

A

Program generated by Chronos

Below is presented the program generated by Chronos during the evaluation. For each session, it is presented the common topic and the submissions and their topics.

Session 1 - Analog/Digital/Mixed/RF IC Design

Submission 8 Analog/Digital/Mixed/RF IC Design

Submission 3 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, RF, Microwave and mm-wave Circuits, VLSI and SoC Applications

Submission 7 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing

Submission 6 Analog/Digital/Mixed/RF IC Design, Biomedical Applications

Submission 2 Analog/Digital/Mixed/RF IC Design

Session 2 - Analog/Digital/Mixed/RF IC Design

Submission 10 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, RF, Microwave and mm-wave Circuits, VLSI and SoC Applications

Submission 12 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, Sensors/Systems and MEMS

Submission 9 Analog/Digital/Mixed/RF IC Design

Submission 16 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 14 Analog/Digital/Mixed/RF IC Design

Session 3 - Analog/Digital/Mixed/RF IC Design

Submission 66 Analog/Digital/Mixed/RF IC Design, Sensors/Systems and MEMS

Submission 69 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing

Submission 73 Analog/Digital/Mixed/RF IC Design

Submission 75 Analog/Digital/Mixed/RF IC Design, Sensors/Systems and MEMS

Submission 68 Analog/Digital/Mixed/RF IC Design

Session 4 - RF, Microwave and mm-wave Circuits

Submission 101 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 111 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 100 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits, VLSI and SoC Applications

Submission 114 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 94 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Session 5 - Emerging Technologies and Applications

Submission 18 Emerging Technologies and Applications

Submission 11 Analog/Digital Signal Processing, RF, Microwave and mm-wave Circuits, Technical Trends & Challenges, Emerging Technologies and Applications, Sensors/Systems and MEMS

Submission 21 VLSI and SoC Applications, Semiconductors, Emerging Technologies and Applications

Submission 42 Semiconductors, Technical Trends & Challenges, Emerging Technologies and Applications

Submission 19 Emerging Technologies and Applications

Session 6 - Sensors/Systems and MEMS

Submission 13 Biomedical Applications, Sensors/Systems and MEMS

Submission 23 Sensors/Systems and MEMS

Submission 86 Sensors/Systems and MEMS

Submission 70 Emerging Technologies and Applications, Sensors/Systems and MEMS

Submission 55 Semiconductors, Sensors/Systems and MEMS

Session 7 - Analog/Digital/Mixed/RF IC Design

Submission 125 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 122 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits, Sensors/Systems and MEMS, Flexible Electronics

Submission 116 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing

Submission 123 Analog/Digital/Mixed/RF IC Design, VLSI and SoC Applications, Sensors/Systems and MEMS

Submission 117 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, RF, Microwave and mm-wave Circuits

Session 8 - Analog/Digital Signal Processing

Submission 15 Analog/Digital Signal Processing

Submission 88 Analog/Digital Signal Processing, VLSI and SoC Applications, Emerging Technologies and Applications

Submission 85 Analog/Digital Signal Processing

Submission 92 Analog/Digital Signal Processing

Submission 65 Analog/Digital Signal Processing, Biomedical Applications

Session 9 - Emerging Technologies and Applications

Submission 133 Technical Trends & Challenges, Integrated Power ICs

Submission 109 RF, Microwave and mm-wave Circuits, Emerging Technologies and Applications

Submission 72 Integrated Power ICs

Submission 110 Emerging Technologies and Applications

Session 10 - Analog/Digital/Mixed/RF IC Design

Submission 27 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, Semiconductors, Integrated Power ICs

Submission 30 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, VLSI and SoC Applications

Submission 31 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 24 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, Semiconductors, Emerging Technologies and Applications

Submission 22 Analog/Digital/Mixed/RF IC Design, VLSI and SoC Applications, Semiconductors, Emerging Technologies and Applications

Session 11 - Analog/Digital/Mixed/RF IC Design

Submission 64 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, Technical Trends & Challenges, Emerging Technologies and Applications

Submission 44 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits, Semiconductors

Submission 51 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 58 Analog/Digital/Mixed/RF IC Design, Emerging Technologies and Applications

Submission 50 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing

Session 12 - Sensors/Systems and MEMS

Submission 95 Sensors/Systems and MEMS

Submission 104 Analog/Digital Signal Processing, Sensors/Systems and MEMS

Submission 99 Sensors/Systems and MEMS

Submission 121 Emerging Technologies and Applications, Sensors/Systems and MEMS

Submission 106 Emerging Technologies and Applications, Sensors/Systems and MEMS, Energy Harvesting

Session 13 - VLSI and SoC Applications

Submission 61 VLSI and SoC Applications

Submission 96 VLSI and SoC Applications, Integrated Power ICs

Submission 93 VLSI and SoC Applications, Emerging Technologies and Applications, Integrated Power ICs, Energy Harvesting

Submission 57 VLSI and SoC Applications, Technical Trends & Challenges

Submission 105 VLSI and SoC Applications

Session 14 - Flexible Electronics

Submission 150 Analog/Digital/Mixed/RF IC Design, Semiconductors

Submission 38 Flexible Electronics

Submission 20 Flexible Electronics

Submission 149 Analog/Digital/Mixed/RF IC Design

Session 15 - Analog/Digital Signal Processing

Submission 107 Analog/Digital Signal Processing

Submission 102 Analog/Digital Signal Processing, Integrated Power ICs

Submission 131 Analog/Digital Signal Processing, Sensors/Systems and MEMS

Submission 108 Analog/Digital Signal Processing, Biomedical Applications

Submission 129 Analog/Digital Signal Processing, VLSI and SoC Applications, Sensors/Systems and MEMS

Session 16 - Semiconductors

Submission 76 Semiconductors, Technical Trends & Challenges

Submission 124 Semiconductors

Submission 134 Semiconductors, Sensors/Systems and MEMS

Submission 103 Semiconductors

Submission 118 Semiconductors

Session 17 - Analog/Digital/Mixed/RF IC Design

Submission 143 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits, VLSI and SoC Applications

Submission 147 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 141 Analog/Digital/Mixed/RF IC Design, VLSI and SoC Applications

Submission 137 Analog/Digital/Mixed/RF IC Design

Submission 140 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, Semiconductors

Session 18 - Biomedical Applications

Submission 144 Biomedical Applications

Submission 139 Biomedical Applications

Submission 138 Biomedical Applications, Emerging Technologies and Applications, Sensors/Systems and MEMS, Flexible Electronics

Submission 113 Energy Harvesting

Submission 142 Biomedical Applications

Session 19 - Emerging Technologies and Applications

Submission 63 RF, Microwave and mm-wave Circuits, Semiconductors, Emerging Technologies and Applications

Submission 71 Semiconductors, Emerging Technologies and Applications

Submission 81 Emerging Technologies and Applications

Submission 62 Technical Trends & Challenges, Biomedical Applications, Emerging Technologies and Applications

Submission 47 Semiconductors, Emerging Technologies and Applications

Session 20 - Analog/Digital/Mixed/RF IC Design

Submission 37 Analog/Digital/Mixed/RF IC Design

Submission 43 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing

Submission 34 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 36 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing

Submission 40 Analog/Digital/Mixed/RF IC Design, Energy Harvesting

Session 21 - Analog/Digital/Mixed/RF IC Design

Submission 126 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 132 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, RF, Microwave and mm-wave Circuits, Biomedical Applications, Sensors/Systems and MEMS

Submission 128 Analog/Digital/Mixed/RF IC Design

Submission 130 Analog/Digital/Mixed/RF IC Design, Technical Trends & Challenges

Submission 127 Analog/Digital/Mixed/RF IC Design, Biomedical Applications

Session 22 - Analog/Digital/Mixed/RF IC Design

Submission 82 Analog/Digital/Mixed/RF IC Design, VLSI and SoC Applications, Emerging Technologies and Applications

Submission 84 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits

Submission 91 Analog/Digital/Mixed/RF IC Design, Analog/Digital Signal Processing, Integrated Power ICs

Submission 83 Analog/Digital/Mixed/RF IC Design, VLSI and SoC Applications, Emerging Technologies and Applications

Submission 90 Analog/Digital/Mixed/RF IC Design, RF, Microwave and mm-wave Circuits