Hindawi
Mobile Information Systems
Volume 2018, Article ID 6371793, 16 pages
https://doi.org/10.1155/2018/6371793

Research Article

A Cloud-Based Mobile System to Improve Project Management Skills in Software Engineering Capstone Courses

Andres Neyem, Juan Diaz-Mosquera, and Jose I. Benedetto

Computer Science Department, Pontificia Universidad Católica de Chile, Santiago, Chile

Correspondence should be addressed to Andres Neyem; aneyem@uc.cl

Received 17 August 2017; Revised 3 April 2018; Accepted 10 June 2018; Published 5 July 2018

Academic Editor: María D. Lozano

Capstone project-based courses offer a favorable environment for the development of student skills through an approach incorporating theoretical and practical components. However, it is often difficult to successfully coordinate between students, stakeholders, and the academic team. The absence of suitable tools for addressing this issue, along with time constraints, often prevents students from attaining the expected course outcomes. This raises the question "How can we improve project management skills in computing majors through the use of technology-enhanced learning environments?" This paper presents a Cloud-based mobile system for supporting project management under a framework of best practices in software engineering capstone courses. The Kanban approach was used as a core of the proposed system. Kanban boards are very popular in the software industry today. It has been empirically shown that they provide increased motivation and project activity control due to their inherent simplicity. This helps the students and academic team be aware of the project context as it aids in preventing ambiguities, flaws, or uncertainties in the development of software artifacts.

1. Introduction

Project management is a valuable skill that university students should develop in order to achieve greater success in the industry. To develop this skill, students require enabling environments that establish the conditions necessary to put knowledge into practice. Project-based courses have proven to be the right space for working on this skill; in these courses, students work as a team to face the challenge of developing projects with real-world limitations and needs.

Through project-based learning courses, computer science and software engineering students have the opportunity to exercise the professional skills they will need after obtaining their degree. Projects with real-life constraints based on the needs and specifications of actual clients are particularly relevant as they are the most effective in bridging the gap between industry and academia. Having the abilities to deal with the management of projects is fundamental for success in this context. Through these abilities, it is possible to maintain a greater level of organization of the activities that are being carried out. Beyond that, being able to correctly manage projects allows for the development of better action and contingency plans, since eventual issues that hinder progress may be foreseen in advance.

For universities, teaching project-based courses is a big challenge that involves heavy responsibilities in providing environments where students can develop their professional skills [1]. The IIC2154 course at the Computer Science Department of the Engineering School at Pontificia Universidad Católica de Chile is a capstone course designed around this vision. The intent of the course is to provide a capstone experience that integrates the knowledge gained from all the previous courses in the curriculum by working on a project with realistic challenges. The blended approach between practical and theoretical components in the teaching of software engineering can bring relevant results because it encourages the development of products closer to production quality [2, 3].

In these kinds of courses, as projects move forward, the complexity surrounding them increases substantially. Charts, statistics, metrics, and reports in general become very important to understanding the real state of projects.

Thus, learning tools take on even more importance as they allow academic teams to follow students' learning process. Some investigations are focused on metrics and strategies aligned with the application of project management abilities and software development methodologies such as Scrum, XP, or Kanban [4, 5]. Although these proposals present fresh approaches in the same vein as our work, they lack a unified ecosystem for supporting project management activities. From this perspective, the aid of technological tools that support the activities of students while managing their projects is essential.

There are different ways to provide technological solutions for the implementation of a unified learning environment. Some teams only use shared Excel files for these tasks; however, there are more sophisticated tools available, such as Microsoft Project, that allow for more detailed specification and control of activities. Nevertheless, approaches like those mentioned above lack the flexibility that a Cloud-based mobile system can provide. These systems offer a mobile point of access (mobile devices) to the functions of management and teamwork in projects through a cloud architecture that maintains traditional interfaces such as desktop web browsers.

In this paper, we present a novel approach that tries to fill the gap between software tools and the development of project management skills by providing a technology-enhanced learning environment. This includes tools and analytics for seamless integration with a proven best practice capstone course framework that allows professors and students to derive as much benefit as possible. This approach presents a Cloud-based mobile system that supports the work of students offering tasks, requirements and test cycle management features, tracking, social charts, and other tools for improving project management skills in capstone project teams. Moreover, this platform produces data-rich activity, letting software practitioners and researchers understand how software development teams work using data science for empirical software engineering [6].

The remainder of this article is structured as follows: Section 2 presents the state of the art through a literature review on practices and tools used as support for project management. Section 3 presents the technology-enhanced learning environment for capstone project teams. Section 4 presents an evaluation of the Cloud-based mobile system based on empirical evidence gathered from the capstone course over a year and a half. Finally, Section 5 presents conclusions and future work related to this research.

2. State of the Art

This section presents the results of a literature review on studies in two areas: studies in software engineering education that define the key elements for the development of effective capstone courses and a series of articles that investigate the use of various technologies for supporting project management activities. The results of this literature review provide the foundations for the design and implementation of this proposal.

2.1. Developing Capstone Experiences in Software Engineering. Capstone courses give students the opportunity to demonstrate integrated knowledge and growth in their careers. Real-life projects have become a staple of software engineering capstone courses in several universities due to their pedagogical value in bridging the academic and industry worlds [7]. Moore and Potts wrote one of the first reports of a software engineering capstone course [8]. They propose emulating an industrial organization through the use of a Real World Lab. In this lab, projects are provided by industrial sponsors who act as clients, delivering consulting, reviewing, direction, and resources. The use of Mini-Task, a development process based on the waterfall life cycle, was also suggested to students. However, this cycle is open to changes after the first iteration in order to include methods and techniques they found useful. In the same vein, Sebern [9] also proposes a Real World Lab where students enroll after taking other software engineering courses, therefore ensuring some experience in developing software projects. It relies on the collaboration of real clients who provide students with existing software, aiming for the application of reverse engineering instead of forward engineering, as projects developed from scratch can either be of limited size or have a high risk of not reaching a functional product.

The Software Factory is another capstone course proposal based on the idea of hands-on experience developed across two semesters [10]. In this course, students develop projects either from scratch or undertake maintenance projects. Different roles are assigned to students according to their seniority. The main focus of the course is to meet industry requirements by educating computer science engineers in mastering modern technologies and processes, guaranteeing a solid and persistent knowledge. On the other hand, Robillard et al. [11] propose a capstone course called Studio in software engineering, where all teams have to develop the same project using UPEDU (the Unified Process for Education). To enroll in this course, students have to first take a software engineering course. They reported that most teams' effort was devoted to a few disciplines and that the learning curve of the process only lets development begin in the second half of the course.

Moving onto learning approaches based on agile software practices, Mahnic [12] uses an agile approach in his capstone course. Even though students were enthusiastic about the agile approach, the development would have been more organized and more easily planned if they had not sometimes missed a more detailed up-front design that would have offered them the complete picture. This course encourages the use of agile practices; however, since it always uses real clients, the clients themselves make sure that student teams work toward building the product they want without losing focus. Vanhanen et al. [2] describe another capstone course that uses an ad hoc agile-based process with industrial clients. The interesting feature of this study is the presence of a mentor, who assists and provides orientation during the entire semester. Undergraduate students participate as developers while master's students act as managers. Students considered the course stressful and laborious, but also extremely rewarding.

Weissberger et al. [13] propose a capstone course where students work on software maintenance projects, although in this case, the client is always the university. Before taking this course, students must already be familiar with several software engineering concepts they are expected to take full advantage of (e.g., stand-up meetings, pair programming, and burndown charts). Stettina et al. [14] describe the design, planning, and continuous improvement of another capstone course on software engineering. They conducted a survey to validate their findings. In this survey, they ask students about their satisfaction related to the project, the teamwork and communication within the project, the use of stand-up meetings, and the use of meeting minutes. Finally, Neyem et al. [15] propose a best practices framework for conducting computer science capstone courses, which is the one implemented in our case study. It involves a project-based approach for enhancing the learning experience, as well as an integration of various agile practices and tools. The goal of this framework is to familiarize students with best practices for developing high-quality software on par with commercial standards found in today's top software industries.

2.2. Project Management Tools. Unlike in the industry, project management tools for the development of academic activities are not normally used in educational environments. In some cases, students prefer to take advantage of informal ways of managing and collaborating on projects; an example of this is Google Docs [16, 17]. However, it is known that proper project management tools are essential for the successful planning and managing of projects [18].

When students are working on projects, it is important that all members remain aware of the tasks that others are working on. Kanban tools are an example of tools focused to this end. They allow for the better visualization of tasks through a Kanban Board. These tools have proven to be a suitable option for the coordination of work groups [19]. Generally, these kinds of tools have a focus on general-purpose projects; two examples in this category are Asana and Trello [20]. There are more tools in the market for supporting project management based on the Kanban methodology; the paper in [19] presents an extensive analysis of those. Two of the most known tools are Kanbanery and JIRA [21]. While the first is offered for free to charitable organizations and open source projects, the second requires a hefty subscription fee after 10 users.

Kanban tools are used to manage the flow of materials or information in a process [22]. These tools drive teams to visualize the workflow, limit Work In Progress (WIP) at each workflow stage, measure the cycle time (i.e., average time to complete one task), make process policies explicit, and improve collaboration [23–25]. Kanban tools can be defined as shared workspaces because they support cooperative tasks and provide a virtual space where information can be shared and exchanged [26]. It is very important to recognize the relevance of the virtual space in these tools. The virtual space enables users to clarify team members' awareness of the current production issues and forthcoming tasks [24].

Ahmad et al. [23] present an empirical study that investigates factors that users of Kanban tools consider to be important. It was found that the top two benefits of Kanban tools are the improved visibility of work and improved development flow. These benefits help enhance communication and collaboration within teams and related stakeholders. Under Kanban, a project is broken down into smaller pieces, and team members progressively finish these WIPs one by one, resulting in a constant flow of completed work items to customers. However, it was discovered that teams found it challenging to see the big picture through these smaller pieces. This suggests a need to provide awareness to team members of how each WIP relates to the others and the state of the project as a whole.

Oza et al. [27] performed a study that analyzes how the Kanban process aids software teams' communication and collaboration. This study collected data on a software development team that developed a mobile payment software product in six iterations over seven weeks. Data were collected through questionnaires at the end of each iteration. The team was part of the Software Factory, which is an experimental laboratory that provides an environment for research and education in software engineering established by the Department of Computer Science at the University of Helsinki. The team included eight engineers and a coach from the Software Factory. The study found that the use of Kanban had a high impact on both communication and collaboration in the initial iterations.

Ahmad et al. in [28] conduct a systematic review in order to analyze the current trend of Kanban usage in software development and to identify the benefits and challenges of the tool. They report that there is little existing scientific literature that addresses the use of Kanban tools for software development in spite of increasing interest among developers. Similar to the previous study on the implementation of Kanban at a software development team, the systematic review found that team communication and coordination with stakeholders is improved by using Kanban. Furthermore, the visualization of tasks with Kanban enables developers to understand the overall direction of work and helps them manage the workflow. This differs from the results of the empirical study in [27], where some teams found it challenging to see the big picture.

Cicibas et al. in [18] make an interesting comparison between project management tools, including Assembla, BaseCamp, DotProject, GanttProject, Liquid Planner, Artemis View, and Primavera, among others. This work defines a series of criteria for evaluating each tool, including task scheduling, resource management, collaboration, time tracking, project effort estimation, risk assessment, reporting, document management, communication tools, and so on. It was found that Artemis View is the most complete tool according to the criteria defined. Artemis View is an integrated enterprise project and resource management application that runs on Windows and web-based platforms.

Last but not least, according to Schwaber [29], most people responsible for managing projects have been taught a deterministic approach to project management that uses detailed plans, Gantt charts, and work schedules. The agile

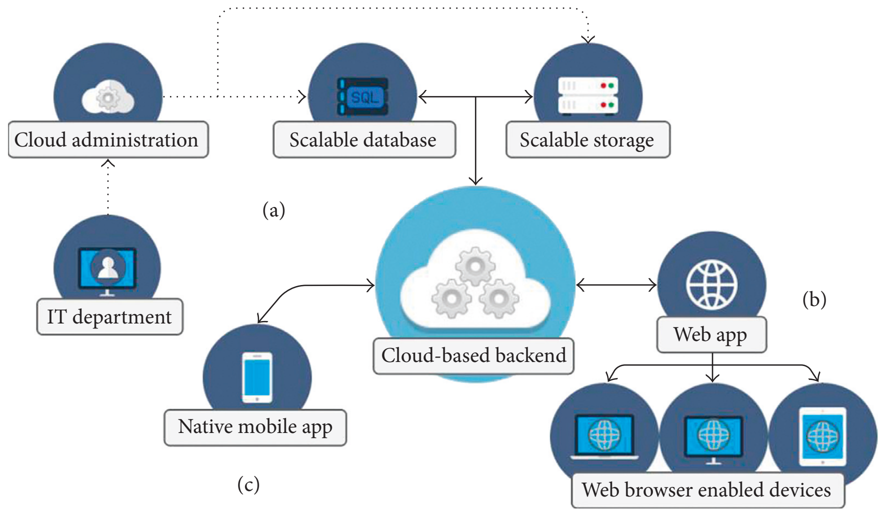


Figure 1: General architecture of the proposed Cloud-based mobile system.

methodology Scrum is the exact opposite. Scrum outlines how to guide a project along its optimal course, which unfolds as the project proceeds. Previously mentioned tools such as JIRA and others such as Pivotal Tracker follow this trend. The platform proposed in this work is based on this approach combined with some of the criteria outlined in [30]; this will be shown in more detail in the next section.

In summary, the review of the selected literature shows that Kanban is a promising tool to adopt for software development and project management. Adoption of Kanban has been shown to improve both communication and collaboration within teams and between related stakeholders. Visualization with Kanban is repeatedly mentioned as an important feature to facilitate project management.

3. The Proposed Cloud-Based Mobile System

3.1. The Architecture of the Cloud-Based Mobile System. The architecture of this system incorporates both a server-hosted component and a mobile client for interacting with the system while in the field. We define such mobile and Cloud Computing (MCC) solutions as Cloud-based mobile systems [31]. We opted for this computational model as mobile devices are becoming a popular way to access educational content, and traditional m-learning has certain limitations [32] that can be overcome with an MCC architecture. Additionally, there is evidence in the literature that these applications succeed in promoting teamwork and collaboration [33] in this context. From here onwards, the term mobile devices will refer to both tablets and smartphones.

Figure 1 shows the general architecture of the proposed system. Its functionalities are separated into three main components: (a) a Cloud-based backend service for computational and data storage purposes, (b) a Web application to enhance collaboration and analysis between students and academic teams, and (c) a mobile application that provides the core Kanban functionalities present in the web application.

The mobile component communicates with the web application via a RESTful API to synchronize data; however, it has its own local database to enable offline operation. Its main purpose is to collect personalized information about a group member's work routine and contributions and to notify them of any relevant status change of any task that was assigned to them.

The proposed platform underwent development for two years by the academic team of the capstone course at the Pontificia Universidad Católica de Chile. It seeks to cover all the basic needs of modern agile software projects and to seamlessly integrate them with an academic tool intended for tracking and evaluating individual progress. We draw attention to the latter, as the primary focus of our solution is its pedagogical value. The platform's features include support for formal methodologies, requirements management, team management, software testing, customer collaboration, and documentation [34]. Moreover, in line with our educational necessities, the platform also supports learning analytics that help to monitor the advance of the projects (Section 3.3) and options to generate standardized documentation for arbitrary software projects and for the course and a public API that allows third parties to query this information for more in-depth analysis (e.g., process modeling, data mining, etc.).

Without the proposed platform, it would be very difficult for the academic team to keep track of the progress made by each individual student in each different project. Furthermore, the data recorded in this platform allow offering more complete feedback to the students about the aspects in which they are succeeding or failing. In contrast, previous editions of the capstone course that lacked support of specialized information technologies made it almost impossible for the academic team to offer personalized guidance, and instead their input was mostly limited to global observations on the status of the projects. Additionally, with no tool to centralize all the information relevant to each project, the practical application of formal agile methodologies was rather limited and project management mostly relied on guesswork. On top

Table 1: Project management criteria: comparison of web and mobile application.

	Mobile	Web
(1) Task scheduling	●	●
(2) Resource management	●	●
(3) Collaboration	●	●
(4) Time tracking	●	●
(5) Estimation	●	●
(6) Risk assessment	◒	◒
(7) Change management	◒	◒
(8) Project analysis/report	○	●
(9) Document management	●	●
(10) Communication tools	●	●
(11) Process development method	●	●
(12) Portfolio management	●	●

● Fully supported; ◒ partially supported; ○ not supported.

of that, the focus on learning analytics of this platform offers a deeper value for both students and teachers, since it enables the platform to be not only a generic project management tool but also a platform enriched with knowledge.

So far, we have placed our focus on the area of software development; however, Conforto et al. show positive experiences in applying agile project management enablers in industries not related to software [35]. With this in consideration, the Cloud-based mobile system is modularly designed to allow for noncomputing industries to disable functionalities irrelevant to their respective fields yet still benefit from the many advantages of agile practices.

In order to determine the project management characteristics of our proposed system, we created a comparative table (Table 1) following the comparison criteria proposed by Ahmad and Laplante [30]. Our main focus is to show what features can be used in the proposed system in the web and mobile application. It is important to highlight that the mobile version does not support reporting, as lack of UI space makes it difficult to display this kind of information in a comprehensive manner. This feature can be better supported in a web browser.

Finally, in both the mobile and the web applications, the core functionalities are centered around "task representations" (from here onwards simply tasks) based on the Kanban board philosophy. It has been empirically shown that they provide increased motivation and project activity control due to their inherent simplicity [24]. While Kanban tasks are no more than granular jobs, our proposed system takes this idea further by allowing participants to enrich them by linking them to other modules in the project. As such, the Kanban board serves as the hub for the entire application. This tool can be found at the following address, under the section "educational tools": http://www.capstonecoursetoolkit.cloud.

3.2. Mobile Application. The mobile application acts as a companion app for the web platform. Its primary purpose is to notify students whenever the status of a task assigned to them changes. It also features the most commonly used functions of the web platform in a user interface optimized for small screens. This allows students to check the global status of the project, modify a task, or share a comment on a relevant matter with the group, even while on the move (e.g., meeting with a client offsite).

In order to reach the widest audience possible, the Android platform was chosen for the development of the mobile component, as it had over 85% of the global mobile OS market share in 2016. Multiplatform and mobile web tools were discarded, as we were interested in tapping into the latest Android-exclusive functionalities.

The mobile application features the core functionalities present in the web application (Figure 2), while the least used functions are relegated to the web platform. In the application, users can manage several organizations and projects (Figure 3(a)). The highlight of the application is the Kanban board (Figure 3(b)). It enables users to see the same global projects overview as in the web counterpart. Here, we include functionalities for creating (Figure 3(c)), editing, and deleting tasks, assigning tasks to different users, submitting task progress reports to keep track of the overall time developers have dedicated to them, file sharing, flag raising, and the ability to share comments between teammates.

All user input is immediately recorded in an offline database, enabling lag-free offline operation. Server synchronization is handled by an Android sync adapter. This allows us to bundle and defer sync operations to a later date. In cases where network connectivity is not available, the OS will trigger the synchronization subroutine only once it returns. It will keep doing so until communication with the server is successful. At the same time, the server will send sync requests whenever there is an update to a relevant task via push notifications. The client can then sync the data back and inform the user on what has changed specifically. Events of interest can be customized to ensure the user is not interrupted by any minor change.

Figure 2 shows an overview of the different layers of the mobile application. The entire application uses state-of-the-art technologies to ensure best practices with regards to code structure, performance, and battery life. The UI was developed using Android's data binding library and architecture components. The data layer was made by leveraging ROOM, Google's latest Android ORM. Lastly, synchronization management is delegated to the operating system through Android's sync adapter pattern to allow for network operation bundling with other applications as well.

There are two synchronization cases that warrant further detail: client-server conflicts and large file handling. Since our platform is decentralized, there exists the possibility for conflicts to arise when different users modify the same task from different clients. In this scenario, a default client-first conflict-solving policy eliminates the need for manual conflict solving. While this policy introduces some border cases in which some data may be lost, in practice, these instances are very rare and therefore deemed an acceptable compromise in favor of usability. On the contrast, our application allows users to share arbitrary documents with one another through a file attachment mechanism. No file limit is imposed, so when it comes to media, for example, there is the possibility of dealing with files in the order of several hundred megabytes. This requires handling with care not only because network interruptions may be frequent, but also because high network traffic through a mobile network

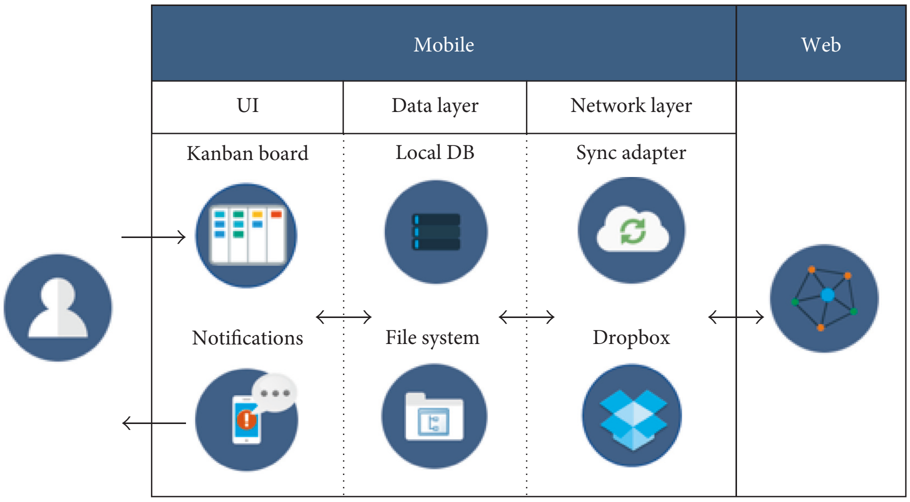
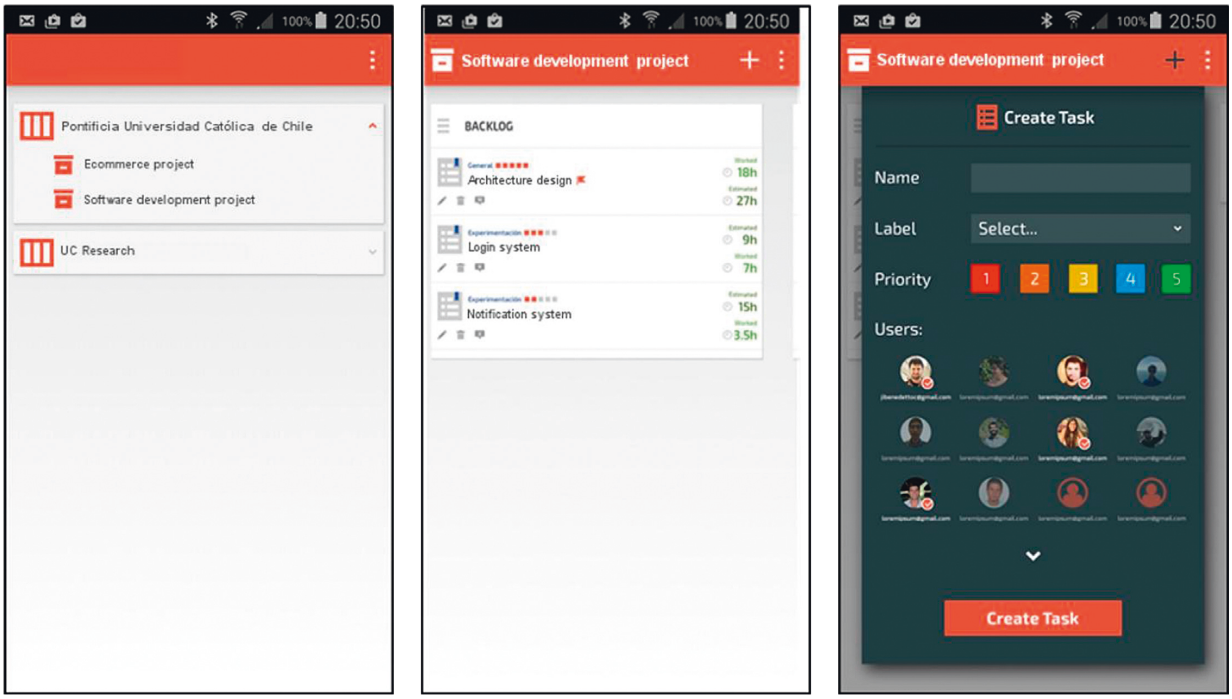


Figure 2: Mobile application overview.



(a)

(b)

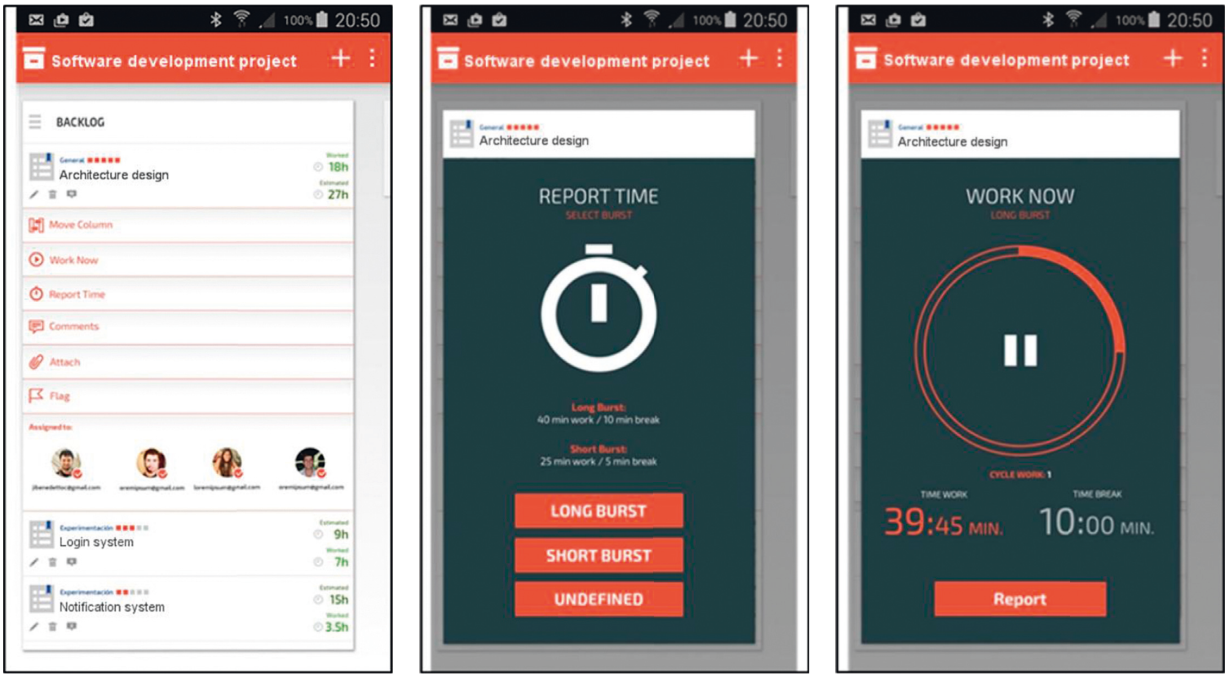
(c)

Figure 3: Mobile client Kanban flow: (a) organizations and projects listing, (b) Kanban board module, and (c) create task screen.

may result in significant financial costs to the end user. For this reason, our platform works in conjunction with the Dropbox file-sharing application, which already implements reasonable file synchronization policies and a resumable upload protocol. Users wishing to share files through our mobile client are therefore first asked to download the Dropbox application to their devices.

The last module of interest in the mobile client is the time reporting component. Users can access this component from the task options menu (Figure 4(a)) and configure their mobile device through our application to keep track of time when they start working (Figures 4(b) and 4(c)). The user can then safely either close the application or turn off the screen. Once he or she has finished working, he or she can return to the application and choose to stop time tracking so that the client can log the session. If the user does not want to time his or her work session, it is also possible to manually log an arbitrary amount of time at any moment. Users are expected to follow proper work ethics to not abuse the system.

Finally, to enhance productivity, alternating work intervals with short period breaks is also suggested. Studies show frequent short breaks that contribute to improving both productivity and wellbeing [36], especially if accompanied by physical activities. Our mobile client can therefore



(a) (b) (c)

Figure 4: Mobile client time reporting flow: (a) task options menu, (b) time reporting module, and (c) burst option selected.

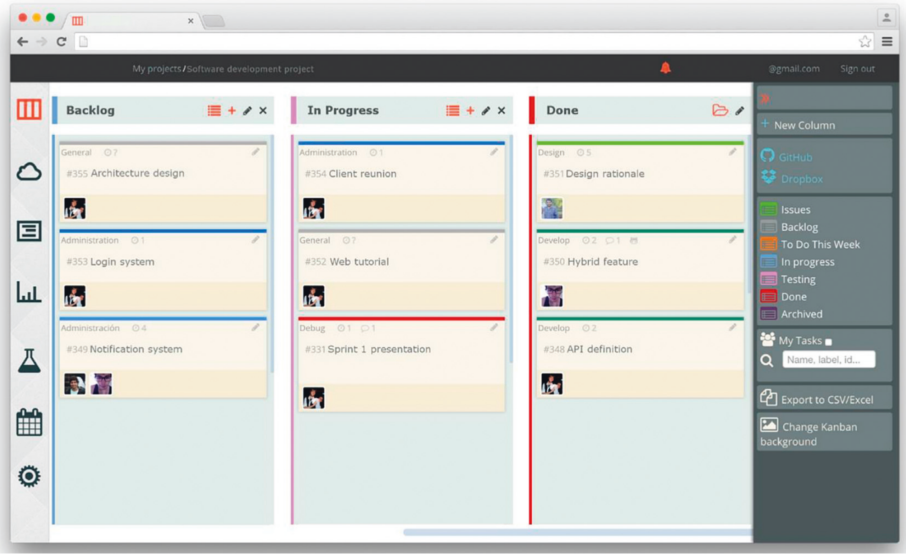


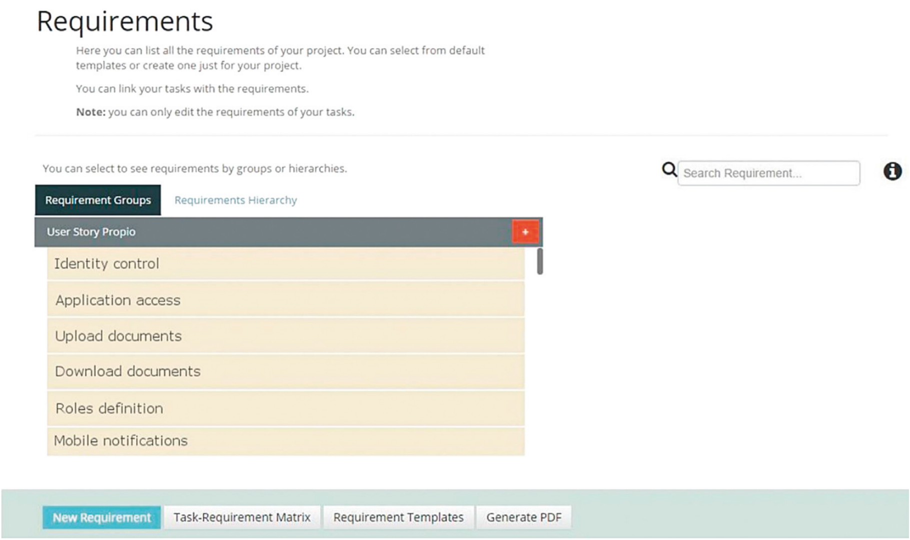
Figure 5: Kanban board in the web application.

be configured to notify the user at regular intervals when to take a short break and when to resume working.

3.3. Web Application. The web application is built using HTML5, CSS, and JavaScript, with Ruby on Rails as a server-side framework. The deployment process is standardized by Capistrano, which, among other benefits, enables fast replication of the application on different servers.

Figure 5 shows the Kanban board: a virtual shared space that offers the entire team a global view of the project, along with all the tasks currently being worked on. Team members can assign tasks to one or more teammates, update their state, share their input via comments, and categorize them through labels. As previously stated, the Kanban board is one of the most effective ways to visualize the workflow and enable student teams to take control over what is to be done and what is currently being worked on.

In the web application, students have the ability to keep track of a project's requirements. These can be associated with tasks registered in the Kanban board. User stories or use cases can be created to reflect the requirements desired by



Requirements

Here you can list all the requirements of your project. You can select from default templates or create one just for your project.

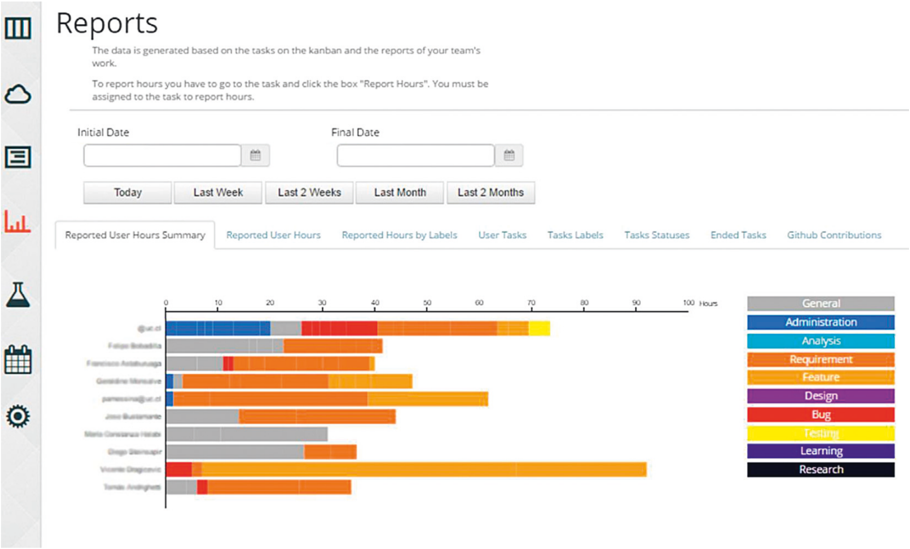
You can link your tasks with the requirements.

Note: you can only edit the requirements of your tasks.

You can select to see requirements by groups or hierarchies.

Identity control
Application access
Upload documents
Download documents
Roles definition
Mobile notifications

Figure 6: Requirements module.



Reports

The data is generated based on the tasks on the kanban and the reports of your team's work.

To report hours you have to go to the task and click the box "Report Hours". You must be assigned to the task to report hours.

Figure 7: Summary of reported user hours.

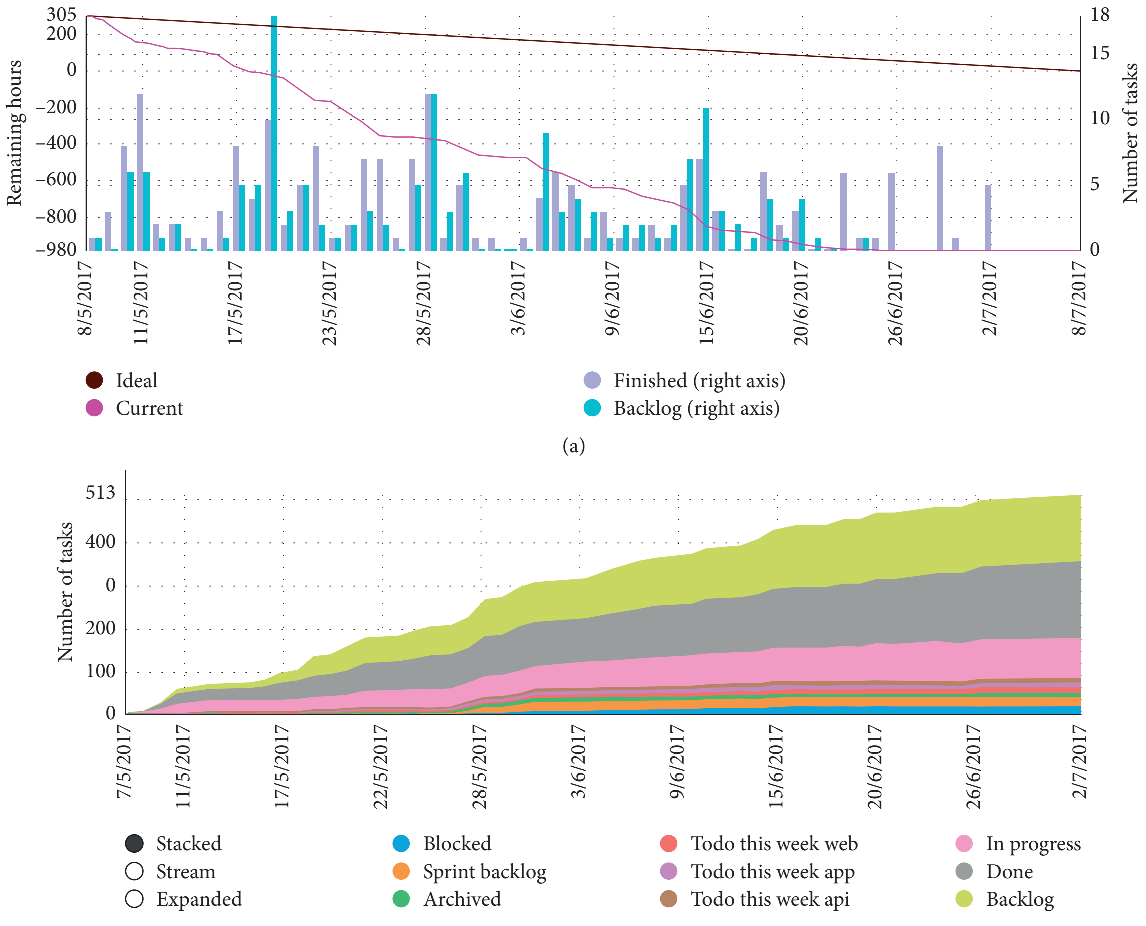
stakeholders (Figure 6); additionally, tasks can be linked with test cycles with the purpose of ensuring the high quality of the software artifacts delivered.

Figure 7 shows the teamwork-reporting module of the web application. Managers and members of the academic team can use this tool to visualize the contributions of each team member through various charts. Users can see their reported hours by categories such as feature development, testing, analysis, and so on. Moreover, the web application enables visualization of the tasks cataloged as finished and reported hours through labels, among other reports. These options allow users to apply several filters over the information. If users need more detail, they can click on the bars to display related information.

Figure 8(a) shows a breakdown of a teammate's contributions. This is a burndown chart that displays the current rate of work done against an ideal rate that is calculated according to the daily rate of team productivity defined by users. Additionally, this report allows users to check the number of tasks in the backlog or tasks with a "finished" status that they have reported each day. Figure 8(b) is a well-known chart in the world of Scrum, called a cumulative chart. As the name implies, this chart shows the sum of work accumulated along different stages of a project in progress. This chart allows for visualizing the workload according to different categories, letting users decide where the project needs the most attention.

There is also a feature where students can visualize the main topics in a project in a treemap (Figure 9). It uses text mining techniques to select the most relevant words and shows these according to the level of relevance (tiles of different sizes). This feature can help students and the academic team to determine where they should pay more attention and apply better project management strategies.

In software project management, providing traceability between a task and its associated code becomes a key factor



(a)

(b)

Figure 8: Agile reporting: (a) breakdown of a teammate's contributions and (b) cumulative chart of a teammate's contributions.

for maintaining wider tracking of project activities. For this reason, the web application in this work is integrated with the code repository platform GitHub. Through this special feature, users can easily revise code changes, in the same way as on GitHub (Figure 10).

The last feature of interest is the Process Mining- (PM-) based Student Teamwork Assessment. It implements PM techniques [37] to provide descriptive process models (Figure 11(a)) using the information from the proposed system. It helps to understand the workflow inside each project team. The line thickness denotes the level of interaction between activities, and the color of each box represents the frequency of each activity (darker colors mean higher frequency). This view allows the students and the academic team to identify situations out of the ordinary in order to take proper action and also to identify strong points in management behavior.

Likewise, PM allows identifying interactions between team members, as the graph in Figure 11(b) represents. The nodes in this graph show a flow between an administrator figure and members of their team; in other words, it corroborates that the administrator works as the central axis of the project.

3.4. Cloud Service as a Backend. There are multiple ways to deploy backend services in cloud computing architectures.

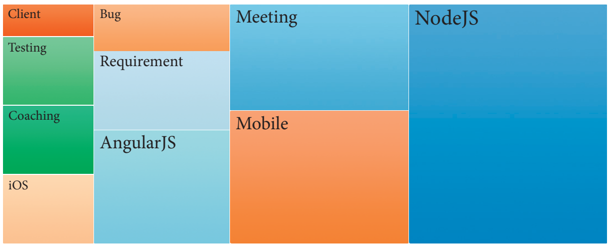
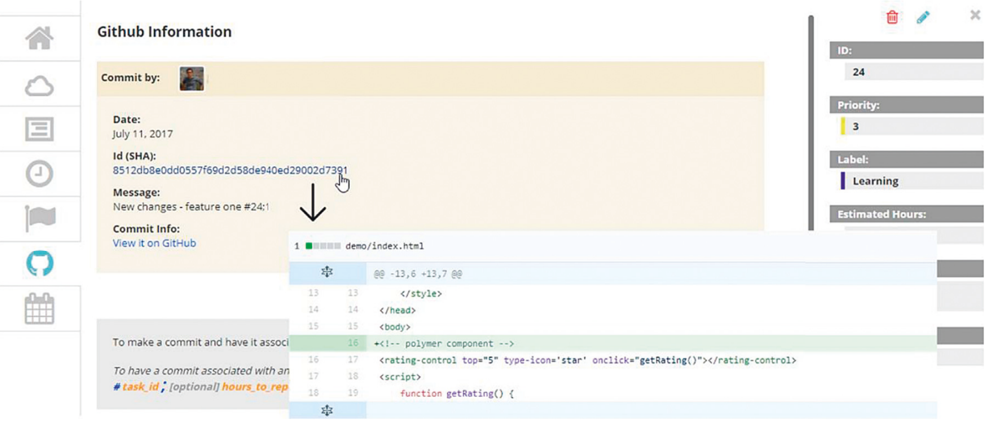


Figure 9: Treemap with the main topics in a project.

One is to implement the whole backend logic using the software distribution model IaaS (Infrastructure as a Service). In this scenario, users have total control of their applications, data, runtime environments, and operating systems. In IaaS, users do not need to spend time on the complexity of building and maintaining the physical infrastructure typically associated with developing and launching an app.

Backend as a Service (BaaS) is another option, consisting of a cloud computing approach for providing services that should be able to be easily accessed. It aims for an integrated way of using a shared pool of computing resources or services that can be rapidly distributed with minimal management effort. It reduces the internal complexity that



Github Information

Commit by:

Date:
July 11, 2017

Id (SHA):
8512db8e0dd0557f69d2d58de940ed29002d7391

Message:
New changes - feature one #24:1

Commit Info:
View it on GitHub

To make a commit and have it associ

To have a commit associated with an
# task_id ; [optional] hours_to_rep

ID:
24

Priority:
3

Label:
Learning

Estimated Hours:

```
1 demo/index.html
@@ -13,6 +13,7 @@
      </style>
   </head>
   <body>
+<!-- polymer component -->
   <rating-control top="5" type-icon='star' onclick="getRating()"></rating-control>
   <script>
      function getRating() {
```

Figure 10: GitHub integration.

developers of these services have to face, offering common preset features such as user management, push notifications, storage, social networking integrations, triggers, and so forth. In essence, the advantages of using this approach for mobile and web apps are as follows:

(i) Availability on demand: the services can be used and accessed at any time.

(ii) Agile and fast provisioning: it is not necessary to set up complex environments; furthermore, scalability for apps is easily achievable.

(iii) Multiple access support: the services can be accessed through the Internet from different kinds of devices.

(iv) Security: offers centralized solutions for authentication and authorization.

(v) Better processing capability: complex tasks can be processed in the backend, reducing computational time compared to devices with limited resources.

The backend for our proposed Cloud-based mobile system is built with a two-fold approach. It uses Firebase (as an instance of BaaS) and the Ruby on Rails (RoR) framework (as an instance of IaaS). With Firebase, we share and distribute data to several users and several devices and platforms. To achieve this, we adopted the Firebase Cloud Messaging technology, which was used to send nonsensitive information from the cloud service to mobile devices for providing push notifications.

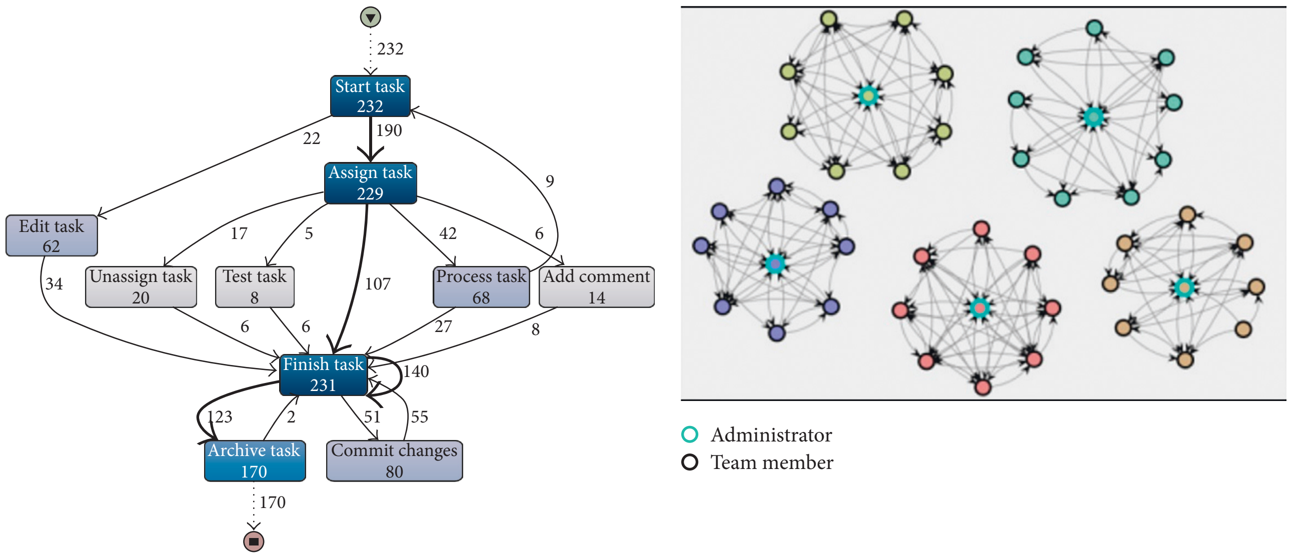
The IaaS component of our system contains a RESTful API deployed on Amazon EC2 that provides all the services for project management activities. Only authenticated users are allowed to interact with our web services. For storage purposes, our cloud system uses high availability Amazon services such as RDS (Relational Database Service) and S3 (Simple Storage Service). In this way, our whole ecosystem relies on Amazon technologies, which allows for easy horizontal and vertical scalability.

4. Evaluation

The following section serves to demonstrate the positive results of using our Cloud-based mobile system to support the development of project management skills. The evaluation was conducted by carrying out 15 projects over 3 semesters (2016-1, 2016-2, and 2017-1) at Pontificia Universidad Católica de Chile (PUC). Our main focus is to study the participants' perceptions of the project management skills that they acquired using the Cloud-based mobile system proposed in this work. Additionally, we study the relevance of the tool in accomplishing this objective. To complement this study, we also include an analysis of the clients' opinions on the work carried out by our students and the final deliverables they received.

The proposed tool is used throughout the course, but it especially helps students to show their progress in weekly meetings with the academic team. This is a process that is fundamentally based on the information recorded by the students on the platform. The data contained within include the record of hours used in the development of the activities and the registration of requirements and test cycles for the developed software artifacts. All of the above may be further complemented by attaching arbitrary documents that are shared through the application. With this information, the academic team can offer better support to the students (e.g., early detection of risks and establishment of contingency plans). It is important to highlight that our intention is to show the platform as a whole; therefore, we evaluate the experience of the students in regard to the complete approach offered in the course and not only in regard to the web and mobile tool used in an isolated context. In this section, we describe the background, the participants, and the instruments used. Then, we discuss the findings that emerged from the quantitative and qualitative analyzes from the students. Finally, we show the opinions of real clients regarding aspects such as product quality, communication, and their perceptions on the project management abilities developed by students.

4.1. Experiments: Context and Participants. The evaluation of the platform was carried out with a total of 146 students with different levels of expertise in the area of software development. These students belong to the capstone course in computing science major at the Computer Science Department. In this evaluation, we incorporated the opinions of 15 clients on the work done by the students. During the



(a)

(b)

Figure 11: Process mining approach: (a) PM model from capstone course project and (b) identifying team interactions using PM.

course, students follow a general outline of the best practices framework [15], which consists of five phases: previous work, introduction and three separate software development sprints. The previous work phase should focus on project selection; the introduction phase should emphasize teaching course methodology and applied software engineering principles; and finally, the software development sprints delve into the actual practical development (requirements, design, coding, testing, and deployment).

Over the course of three semesters, the capstone course was able to encompass a large group of projects. These include startups, established businesses, foundations, and university projects. Among those that stand out the most, we have one project consisting of a technological platform to carry out online evaluations and interact with a multimedia encyclopedia (images, videos, and 3D) on anatomy; another one consisting of a system for generating conference-focused websites that allow users to create and manage the flow of events of this type in an intuitive way; and a third one consisting of the development of embedded systems to be used as beacons for a more user-participatory marketing and advertising strategy.

The aforementioned projects, along with 12 others, involve the development of web, mobile, and embedded systems. These projects have an average of 10 members (an administrator included) who share different tasks throughout the duration of the course. These activities demand a high level of management and teamwork as they are frequently interconnected. Moreover, these activities represent different challenges that students have to face. Among the list of challenges are the integration of external systems, the development of APIs, the construction of e-commerce systems, the implementation of payment systems, and the development of notification modules, among other activities.

4.2. Instruments and Data Collection. A survey was conducted at the end of each semester in order to collect quantitative and qualitative data and analyze them according to the focus of this work. This research used a mixed-method approach with a concurrent design, meaning quantitative and qualitative questions were asked in parallel or in the same survey [38]. In this survey, we included three open-ended questions intended to assess the development of project management skills amongst the students. Additionally, clients were asked one more questions intended to evaluate the overall experience of the course. We sent this survey to 146 students (52 in 2016-1, 49 in 2016-2, and 45 in 2017-1) and 15 clients during the three semesters of the capstone course. The level of participation varied in each semester.

The data gathered from the open-ended questions allow us to perform a qualitative analysis with the participants' opinions. Details on the contents of the survey are available in Table 2. Question number 3 evaluates the achievement level that participants perceive for that item and comprises four possible scores. The evaluation scale considers 1 as "unsatisfactory," 2 as "developing," 3 as "satisfactory," and 4 as "exemplary."

4.3. Results. According to the participants' opinions, in overall, the mixed approach between the proposed system and the methodology implemented in the capstone course gave positive results. The qualitative data were analyzed using the grounded theory method in empirical software engineering [39]. The following is a specific analysis for each research-related question asked in the survey.

4.3.1. Question 1: Approach Evaluation. By analyzing the first open-ended question, we discovered that the proposed system played an important role in carrying out the work of task planning. Some participants stated that task planning was carried out weekly and that the proposed system allowed

Table 2: Questions in the survey.

Question	Category	Participant
(1) What do you think was the approach that allowed you to exercise the development of the ability “organizational skill” during the course?	Open-ended	Students
(2) What do you think about the proposed tool as a means of support for the development of your project?	Open-ended	Students
(3) At what level were you able to develop project management skills (such as planning, scheduling, tracking, and teamwork) present in a software project?	Multiple choice	Students
(4) What are your opinions about the work done by students in the capstone course (strategies implemented, tools used, students’ performance, delivered product, etc.?	Open-ended	Clients

a clear division of tasks through the Kanban approach (Quotes 1 and 2).

Quote 1: *Using a system like the one we used in the course has been fundamental for better task planning.*

Quote 2: *The system’s Kanban approach allows a clear division of activities.*

Being able to define tasks and prioritize them and establish their time limits has allowed for a better management of the functionalities that the students have to deliver (Quotes 3 and 4). In addition, the implementation of user stories promoted by the system has also allowed for better organization (Quote 5).

Quote 3: *The number of tasks that are programmed in the system for each week has allowed us to exercise our organizational skills, since we need to prioritize the development of each one.*

Quote 4: *The use of the system and its integrated Google Calendar feature has allowed us to organize the deadlines of the tasks that must be delivered.*

Quote 5: *Using user stories and being able to define several requirements per week has helped us a lot.*

4.3.2. Question 2: Tool Evaluation. We addressed this question from three perspectives: the utility, the user experience, and the learnability of the tool. The students’ responses and their respective analysis are approached in that order.

From a utility perspective, the responses led to the discovery that regardless of the required time for registering information in the proposed system, students considered it useful as it enabled more control over their projects (Quotes 6, 7, and 8).

Quote 6: *I am aware that the use of tools such as the Kanban board proposed in this course made our project more organized.*

Quote 7: *The proposed system mainly offers order. Another aspect is that the group members can visualize the activities that everyone is performing. Really, if we used it with even more dedication we could obtain greater benefits.*

Quote 8: *With the tasks organized by phases in the tool we have a better perspective of what we have to do.*

Some students highlight the fact that they can easily navigate through the different functionalities of the tool (Quote 9). They also highlight the characteristics related to social awareness such as roles, activities, privileges, and group history that can be easily identified and used in the platform (Quotes 10 and 11).

Quote 9: *The navigation in the tool is simple; having organizations and projects in a hierarchical way makes the management of these very practical. On the other hand, the central functionality, “the Kanban board,” is very flexible and allows adapting the projects to the needs of each work team.*

Quote 10: *To know what my teammates are doing, I would read the titles in the tasks on the Kanban board. If I wanted more details, I would open the tasks and look at the description.*

Quote 11: *In my case, being the team leader, I have to constantly check what my teammates are doing. I do that in different ways, for example by chat, face-to-face meetings, shared documents or with the activities registered on the Kanban board.*

In the context of tasks, it is important to know the answers to the following questions: who, what, where, when, and how. Some features of the proposed system allowed students to answer these questions (Quotes 12 and 13).

Quote 12: *The notifications at the start of the system let us see who created or modified the tasks.*

Quote 13: *The visualization of requirements and test cycles is useful to better understand what is going on in projects.*

From a user experience perspective, some students see the Kanban board feature as a way of incentivizing their own work. Watching the work of the other students can generate positive motivation (Quote 14). Moreover, the design of the platform makes them feel comfortable due to the distribution of the content and the style used for the mobile and web application (Quotes 15 and 16).

Quote 14: *I think that the Kanban (board) has a large benefit. On the one hand, I can see how much work the rest of the project members are doing. This can incentivize you to work more.*

Quote 15: *The tool handles a good balance both in the combination of colors and the distribution of its characteristics. It’s nice to work on the tool; it even allows some degree of interesting customization.*

Quote 16: *In my opinion, the style of the web and mobile tools is quite modern and intuitive. The tool really allows you to have a real immersion in a project with agile methodologies.*

Some participants highlighted more aspects that they found attractive when using the proposed system (Quotes 17 and 18). Some even recognized that the system could be helpful in a business environment (Quote 19). This is an important finding, because it lets us understand the impact that the system might have on the working experience of the participants.

Quote 17: *The system helps you set short goals and visualize them in the Kanban board.*

Quote 18: *With the system, I can see how well my teammates are doing. Besides watching what they do, I can help them or ask for help if they are doing something related to what I have to do.*

Quote 19: *Actually, I would use the proposed tool for project management in my future working life.*

Regarding the learnability of the tool, it was noted that the first contact of the students with the tool was not difficult. The ease of access, the proper location of the menus, and design similar to tools widely known in the market let the students quickly learn how to use the tool (Quotes 20 and 21).

Quote 20: *The first time I tried the tool it reminded me of Trello. For me it was very easy to understand how to use the tool.*

Quote 21: *I have already had the opportunity to use tools of this style; I can say that the learning curve for this tool is not too steep.*

4.3.3. Question 3: Performance Level. As we previously mentioned, this question used a four-point scale. *Unsatisfactory* means that the participant felt he or she was not able to demonstrate understanding of the knowledge and skills referred to in this question. *Developing* indicates that the participant felt able to demonstrate understanding of the knowledge and skills referred to in this question, however, only in the case of environments similar to the one seen in the course. *Satisfactory* refers to the ability that the participant has to describe, apply, and solve problems using the knowledge and the skills referred to by this question; moreover, the participant can face and overcome situations in settings different from that of the course. Finally, *Exemplary* means that the participant felt able to show mastery of the knowledge and skills involved in the question, being able to synthesize, organize, plan, manage, evaluate, and/or teach others, in diverse situations covered by the question.

According to the results, the participants considered that they had gained the skills required to manage projects properly. All responses indicated that participants had acquired a satisfactory or exemplary level (Figure 12) of understanding. It is interesting to note that in all cases, the approach used in the capstone course influenced the development of project management skills, and we are certain that the Cloud-based mobile system was a relevant support in that regard.

4.3.4. Question 4: Client Opinions

(1) Product Quality. Throughout the capstone course, different types of projects are carried out, which represent a great technical and coordination challenge both for the students and the academic teams. Despite the complexity of the projects, the students have been able to present products with similar quality to that of professional software development companies.

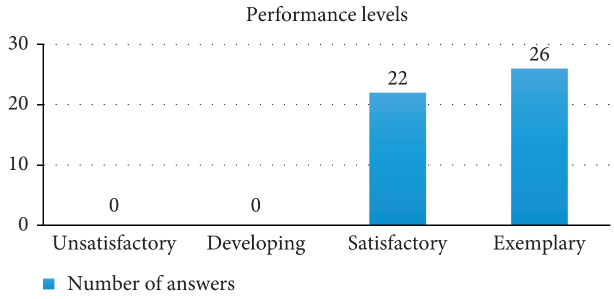


Figure 12: Results of question 3.

Generally, clients declare that the projects developed by the students meet the objectives expressed throughout the course. In accordance with the requirements, in the large majority of cases, 100% of the requested functionalities are achieved (Quotes 22 and 23). The approach that allows obtaining these results is the clear and timely administration of all the activities that take place in the projects through the management tool that this proposal presents.

Quote 22: *I was surprised with the product delivered by the students. I can assure that they fully complied with everything we had talked about from the beginning.*

Quote 23: *The platform that the students made demonstrates how professional they can be. I have in my hands a product that is ready to be put into production.*

Some clients stated that they received more than they expected. When they came to the capstone course, they had a basic idea of what they are looking for; however, at the end of the course, they were left with the satisfaction of having obtained a product that could put into practice what they had imagined (Quotes 24 and 25). The combination of a framework that coordinates the clients, students, and academic team well and a management tool that allows for a detailed follow-up of the tasks, their managers, and the state of these, constitutes for us the success of our approach.

Quote 24: *I am not an expert in technology, which is why my ideas could be a bit difficult to understand; however, the capstone students managed to understand my requirements and make software with more than I expected.*

Quote 25: *I can say that the students in charge of my project knew how to make and guide me towards a product that exceeded my expectations.*

(2) Communication with Clients. Communication with clients was achieved in different ways, through formal meetings and in-progress presentations, while periodically using the course's project management tool. Clients had the ability to access the Kanban board where all the pending, in-process or completed activities within the project, were recorded (Quotes 26 and 27).

Quote 26: *When I wanted to know what the students were doing, it was enough for me to go to the Kanban board and see the activities registered by them.*

Quote 27: *Through the tool I found an easy way to leave comments about the functionalities that were being developed for my project, in this way I clarified the doubts of the students in a timely manner.*

(3) Perceived Project Management. Clients expressed their opinion on how they viewed the management of the projects carried out by the students. During the three project development cycles, they could see how the activities aimed at delivering functional software artifacts were being managed. Having the requirements, the test cycles and the activities clearly divided by phases, categories, and priorities was a clear indicator of good management (Quotes 28 and 29).

Quote 28: *The way in which I perceived that students were having good project management was noticing that they had clearly defined roles and tasks for each member, and they were quite supported by the platform that manages the course.*

Quote 29: *I thought it was excellent that there was a clearly defined timeline where it was established what was going to be done and under what priority in each cycle of the project.*

In each of the meetings with the clients, the students relied on indicators generated by the tool, which reported the hours of work employed in the management and development of the project. The clients declared that it was easy to see the level of commitment of the students when visualizing the charts that the tool offers (Quotes 30 and 31).

Quote 30: *The students were very transparent in showing and explaining the time spent in the development of the functionalities proposed by us by using the tool.*

Quote 31: *With the indicators shown by the students we could clearly notice the time and commitment with which our project was developed. I consider in general terms that the students knew how to work together and manage the project very well.*

5. Discussion and Conclusions

This paper presents a novel Cloud-based mobile system for assisting students in a capstone course in software engineering. The combination of the tool and a framework of best practices in the capstone course generated positive results for the development of different skills in students. This work places special emphasis on skills relevant to project management, which we consider to be one of the most important aspects in the correct execution and control of tasks in work groups.

The proposed tool involves a series of features that revolve around a Kanban approach, which offers better visualization and monitoring of the activities defined in a project during its different stages. The MCC component of this system is fundamental in more broadly supporting the development of project management skills and therefore teamwork. In this context, the mobile part of the tool is a subset of features that complements the entire proposed system.

The analytics or visualizations presented in this work are an important source of information that allows both the students and the academic team to identify different aspects of work and performance within the projects. It is important to highlight that this work is largely based on the use of learning analytics, which offer a way to use the academic information offered by the capstone course to understand and subsequently optimize learning and the environments where it takes place. Our proposed system is a suitable environment for converting information into knowledge using different techniques in the area of Data Science for Empirical Software Engineering; specifically, we present in this work, an approach using process mining and text mining techniques.

The evaluation instrument used in this study allowed us to recognize that the proposed system had an impact on the students' work experience. However, we must recognize that this would not have been possible without the framework established in the course that motivated a greater organization of the activities they carried out. This framework encourages students to rely on the information registered in the proposed system to be able to better control and track the projects, especially in weekly meetings, where progress in these projects is shared with the academic team.

The investigation bridges two separate worlds: academia and industry. With the proposed platform, greater possibilities are offered to students to recognize and live a software development experience that is more similar to the real world. Web and mobile platforms are a means to facilitate the process of organizing student tasks and carrying out better course coordination by teachers. At the end of each semester, surveys are conducted that allow us to obtain more evidence about the application of our approach. In the future, we hope to integrate additional features with the proposed system, among which we have considered the development of a recommendation system that will offer students assistance in the activities or tasks they want to accomplish. This system will consist of the consolidation of knowledge or experiences of other people who performed similar activities in the past. It will use this information to offer suggestions based on how the affected people were able to overcome such problems before.

Another approach we seek to pursue in the future is machine learning. With information captured from our system and consolidated in a training database, we want to discover special factors for objectively determining, assessing, and predicting student teamwork outcomes by applying machine learning methods to the training database [40]. Moreover, we want to promote the participation of students with the proposed system through gamification strategies that generate better-defined spaces of collaboration.

In summary, the evidence obtained from research data leads us to conclude that systems of this type in educational environments can generate a significant influence on the future work experience of students. This is derived from the analysis of the opinions obtained from the survey. The work conducted in this study presents a technology-enhanced learning environment where a framework of best practices in capstone courses converges with MCC components for the support of student and academic team activities. In essence, this approach constitutes a proposal to improve learning and teaching of critical skills for the proper professional development of students.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors would like to acknowledge the Computer Science Department, the Education Engineering Unit of the Engineering School, and the VRA FondeDOC for the continued support of this course. Finally, the authors would like to thank all the students from the IIC2154 capstone course who were involved in this educational research project.

References

[1] A. T. Chamillard and K. A. Braun, "The software engineering capstone: structure and tradeoffs," *ACM SIGCSE Bulletin*, vol. 34, no. 1, pp. 227–231, 2002.

[2] J. Vanhanen, T. O. Lehtinen, and C. Lassenius, "Teaching real-world software engineering through a capstone project course with industrial customers," in *Proceedings of the First International Workshop on Software Engineering Education Based on Real-World Experiences*, pp. 29–32, IEEE Press, Zurich, Switzerland, June 2012.

[3] M. Vasilevskaya, D. Broman, and K. Sandahl, "Assessing large-project courses: model, activities, and lessons learned," *ACM Transactions on Computing Education*, vol. 15, no. 4, pp. 1–30, 2015.

[4] L. Alperowitz, D. Dzvonyar, and B. Bruegge, "Metrics in Agile project courses," in *Proceedings of the 38th International Conference on Software Engineering Companion*, pp. 323–326, ACM, Austin, TX, USA, May 2016.

[5] S. Mohan, S. Chenoweth, and S. Bohner, "Towards a better capstone experience," in *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, pp. 111–116, ACM, Raleigh, NC, USA, February 2012.

[6] T. Menzies, L. Williams, and T. Zimmermann, *Perspectives on Data Science for Software Engineering*, Morgan Kaufmann, Burlington, MA, USA, 2016.

[7] A. Rusu, A. Rusu, R. Docimo, C. Santiago, and M. Paglione, "Academia-academia-industry collaborations on software engineering projects using local-remote teams," *ACM SIGCSE Bulletin*, vol. 41, no. 1, pp. 301–305, 2009.

[8] M. Moore and C. Potts, "Learning by doing: goals and experiences of two software engineering project courses," in *Proceedings of the 7th SEI CSEE Conference on Software Engineering Education*, pp. 151–164, San Antonio, TX, USA, January 1994.

[9] M. J. Sebern, "The software development laboratory: Incorporating industrial practice in an academic environment," in *Proceedings of the 15th Conference on Software Engineering Education and Training (CSEE&T 2002)*, pp. 118–127, IEEE, Covington, KY, USA, 2002.

[10] J. D. Tvedt, R. Tesoriero, and K. A. Gary, "The software factory: combining undergraduate computer science and software engineering education," in *Proceedings of the 23rd International Conference on Software Engineering (ICSE 2001)*, pp. 633–642, IEEE, Toronto, ON, Canada, May 2001.

[11] P. N. Robillard, P. Kruchten, and P. d'Astous, "Software engineering using the Upedu," Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2002.

[12] V. Mahnic, "A capstone course on agile software development using Scrum," *IEEE Transactions on Education*, vol. 55, no. 1, pp. 99–106, 2012.

[13] I. Weissberger, A. Qureshi, A. Chowhan, E. Collins, and D. Gallimore, "Incorporating software maintenance in a senior capstone project," *International Journal of Cyber Society and Education*, vol. 8, no. 1, pp. 31–38, 2015.

[14] C. J. Stettina, Z. Zhou, T. Bäck, and B. Katzy, "Academic education of software engineering practices: towards planning and improving capstone courses based upon intensive coaching and team routines," in *Proceedings of the 2013 IEEE 26th Conference on Software Engineering Education and Training (CSEE&T)*, pp. 169–178, IEEE, San Francisco, CA, USA, May 2013.

[15] A. Neyem, J. I. Benedetto, and A. F. Chacon, "Improving software engineering education through an empirical approach: lessons learned from capstone teaching experiences," in *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, pp. 391–396, ACM, Atlanta, GA, USA, March 2014.

[16] X. Tan and Y. Kim, "Cloud computing for education: a case of using Google Docs in MBA group projects," in *Proceedings of the 2011 International Conference on Business Computing and Global Informatization (BCGIN 2011)*, pp. 641–644, IEEE, Shanghai, China, July 2011.

[17] S. Dekeyser and R. Watson, "Extending Google Docs to collaborate on research papers," Technical Report, University of Southern Queensland, Toowoomba, QLD, Australia, 2006.

[18] H. Cicibas, O. Unal, and K. A. Demir, "A comparison of project management software tools (PMST)," in *Proceedings of the Software Engineering Research and Practice*, pp. 560–565, Las Vegas, NV, USA, July 2010.

[19] E. Corona and F. E. Pani, "A review of lean-Kanban approaches in the software development," *WSEAS Transactions on Information Science and Applications*, vol. 10, no. 1, pp. 1–13, 2013.

[20] H. A. Johnson, "Trello," *Journal of the Medical Library Association*, vol. 105, no. 2, p. 209, 2017.

[21] P. Li, *Jira 7 Essentials*, Packt Publishing Ltd., Birmingham, UK, 2016.

[22] P. Klipp, *Getting Started with Kanban*, Kanbanery, Kraków, Poland, 2011.

[23] M. O. Ahmad, J. Markkula, and M. Oivo, "Insights into the perceived benefits of Kanban in software companies: practitioners' views," in *Proceedings of the International Conference on Agile Software Development*, pp. 156–168, Springer International Publishing, Edinburgh, UK, May 2016.

[24] M. Ikonen, E. Pirinen, F. Fagerholm, P. Kettunen, and P. Abrahamsson, "On the impact of Kanban on software project work: an empirical case study investigation," in *Proceedings of the 2011 16th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS 2011)*, pp. 305–314, IEEE, Las Vegas, NV, USA, April 2011.

[25] E. M. Schön, D. Winter, J. Uhlenbrok, M. J. Escalona, and J. Thomaschewski, "Enterprise experience into the integration of human-centered design and Kanban," in *Proceedings of the ICSOFT-EA*, pp. 133–140, Lisbon, Portugal, July 2016.

[26] A. Neyem, S. Ochoa, and J. Pino, "Designing mobile shared workspaces for loosely coupled workgroups," in *Proceedings of the International Conference on Groupware: Design, Implementation, and Use (CRIWG 2007)*, pp. 173–190, Bariloche, Argentina, September 2007.

[27] N. Oza, F. Fagerholm, and J. Munch, "How does Kanban impact communication and collaboration in software

engineering teams?," in *Proceedings of the 2013 6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE 2013)*, pp. 125–128, IEEE, San Francisco, CA, USA, May 2013.

[28] M. O. Ahmad, J. Markkula, and M. Oivo, "Kanban in software development: a systematic literature review," in *Proceedings of the 2013 39th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA 2013)*, pp. 9–16, IEEE, Santander, Spain, September 2013.

[29] K. Schwaber, *Agile Project Management with Scrum*, Microsoft Press, Redmond, WA, USA, 2004.

[30] N. Ahmad and P. A. Laplante, "Software project management tools: making a practical decision using AHP," in *Proceedings of the 30th Annual IEEE/NASA Software Engineering Workshop (SEW'06)*, pp. 76–84, IEEE, Washington, DC, USA, April 2006.

[31] D. Carver, W. K. Chan, C. K. Chang, and H. Yang, "Software engineering and applications for cloud-based mobile systems," *IEEE Transactions on Services Computing*, vol. 9, no. 5, pp. 742–744, 2016.

[32] A. M. Rosado da Cruz, *Modern Software Engineering Methodologies for Mobile and Cloud Environments*, IGI Global, Hershey, PA, USA, 2016.

[33] S. Abolfazli, Z. Sanaei, M. H. Sanaei, M. Shojafar, and A. Gani, "Mobile cloud computing," in *Encyclopedia of Cloud Computing*, S. Murugesan and I. Bojanova, Eds., John Wiley & Sons, Ltd., Chichester, UK, 2016.

[34] N. Uikey and U. Suman, "An empirical study to design an effective agile project management framework," in *Proceedings of the CUBE International Information Technology Conference and Exhibition*, pp. 385–390, ACM, Pune, India, September 2012.

[35] E. C. Conforto, F. Salum, D. C. Amaral, S. L. da Silva, and L. F. M. de Almeida, "Can agile project management be adopted by industries other than software development?," *Project Management Journal*, vol. 45, no. 3, pp. 21–34, 2014.

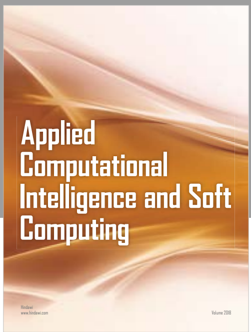
[36] R. A. Henning, P. Jacques, G. V. Kissel, A. B. Sullivan, and S. M. Alteras-Webb, "Frequent short rest breaks from computer work: effects on productivity and well-being at two field sites," *Ergonomics*, vol. 40, no. 1, pp. 78–91, 1997.

[37] W. M. Van der Aalst, *Process Mining: Data Science in Action*, Springer, Berlin, Germany, 2016.

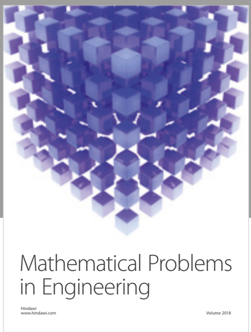
[38] J. W. Creswell and V. L. P. Clark, *Designing and Conducting Mixed Methods Research*, Sage Publications, Thousand Oaks, CA, USA, 2007.

[39] S. Adolph, W. Hall, and P. Kruchten, "Using grounded theory to study the experience of software development," *Empirical Software Engineering*, vol. 16, no. 4, pp. 487–513, 2011.

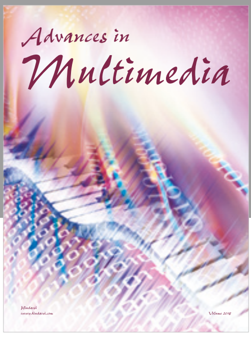
[40] D. Petkovic, "Using learning analytics to assess capstone project teams," *Computer*, vol. 49, no. 1, pp. 80–83, 2016.

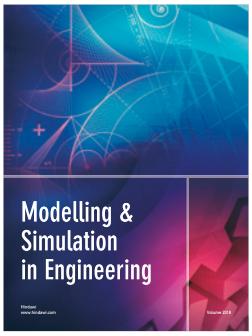

Applied Computational Intelligence and Soft Computing

The Scientific World Journal


Mathematical Problems in Engineering

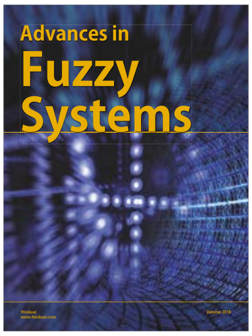
Journal of Engineering

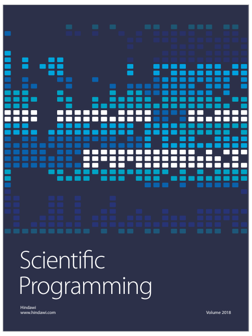

Advances in Multimedia

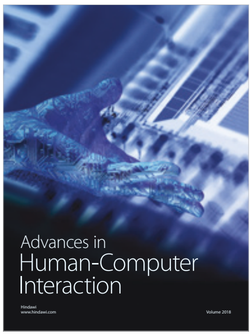

Modelling & Simulation in Engineering

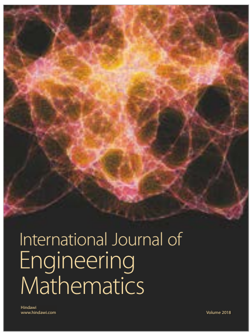
Advances in Artificial Intelligence

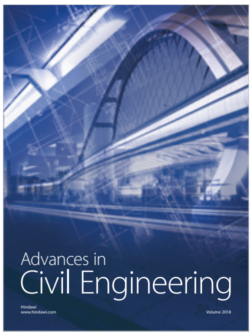
International Journal of Reconfigurable Computing

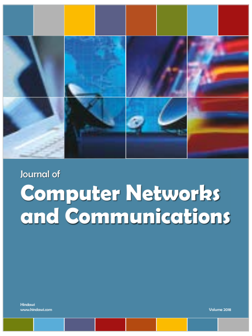

Advances in Fuzzy Systems


Scientific Programming


Advances in Human-Computer Interaction

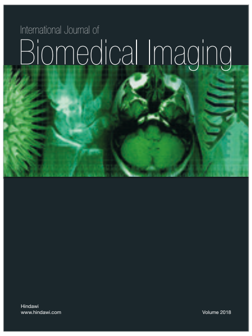

International Journal of Engineering Mathematics


Advances in Civil Engineering

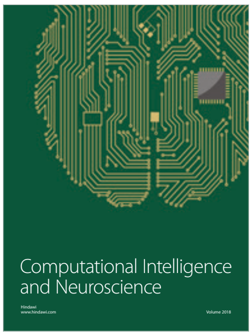

Journal of Computer Networks and Communications

Journal of Robotics

International Journal of Computer Games Technology


International Journal of Biomedical Imaging

Journal of Electrical and Computer Engineering


Computational Intelligence and Neuroscience