



`./figures/logo_isel.png`

Corporate Collaboration

David Fidalgo
goncalo9marco@hotmail.com
968 225 962

Pedro Santos
santos.pedroc@gmail.com
962 942 930

Valdemar Antunes
ValdemarPCA@hotmail.com
966 439 259

Orientadores:

Paula Graça, ISEL, paula.graca@isel.pt
Diogo Pacheco, Do iT Lean, diogo.pacheco@doitlean.com

Proposta de projeto realizado no âmbito de Projecto e Seminário,
do curso de licenciatura em Engenharia Informática e de Computadores
Semestre de Verão 2019/2020

14 de Março de 2020

Introduction

Today's market, increasingly technological, demanding and challenging, imposes a work rhythm on companies, that besides managing their core business processes, also have a significant amount of dynamics on internal activities to help them in their growth and competitiveness. In particular, companies in the information and technologies sector keep internal activities such as work fairs, knowledge sharing through informal presentations, software components development, qualification opportunities, playful activities, among others. In order to put these activities in practice, in an efficient way, there must be a coordination of resources that is not easy to obtain, given its allocation to ongoing projects.

However, by developing a collaboration platform we can achieve correct management of the plannings, so that they can be executed in due time, and the whole process can be agile allowing, not only the registration and publication of internal needs, but also the acceptance of applications by the collaborators interested in its development. Therefore, it is intended to develop a web reactive application using the OutSystems platform [1], that functions as a collaboration platform designed for the management of internal corporate necessities in a company, and is accessible in all kinds of devices: browsers, tablets and smartphones.

Requirements

In order to create a fully functional, efficient and trustworthy reactive web application that will work as a collaboration platform for companies to use, there must be a set of requirements that are predefined in due time. The understanding of the OutSystems platform and the exploration of its usability is a key factor that will provide a concise implementation of these requirements. The main requirements are as follows:

1. Registration and authentication in the collaboration platform.
2. Registration of the company's internal necessities, such as the planning of events or job fairs, brown bags to share knowledge between employees, right medium posts, qualification offers, software components development, internal opportunities for career progression and off work activities.
3. Disclosure and scheduling of the company's necessities.
4. Application management and registration for the divulged necessities.
5. Create notifications targeting the author of the necessity when there are new applications and targeting candidates when their application has been accepted or refused, as well as upon the necessity's closure.
6. Possibility of integrating the collaboration platform with Google Maps to mark/indicate the place for events that take place outside the company.

There were defined optional requirements, to be implemented according to how the main requirements are fulfilled. These are described as:

1. Generation of reports that are defined according to how the internal needs of the company/events were fulfilled.
2. Registration of who applies the most to what kinds of events or internal needs, creating a top three or top five representation for each event/internal need.
3. Individual indication of presence or arrival at the place of an event, that can be observed by other participants.

Architecture

The reactive web application will be developed using a low-code platform named OutSystems, that will enable a faster and more efficient development. Its architecture is presented in Figure 1. As shown bellow, the main component of the platform is the Platform Server that allows to generate, optimize, compile and publish the developed applications. This component uses the following services:

1. **Code Generator:** Uses the modeled application in Service Studio and generates the necessary code using standard technologies (such as .NET, SQL Server, HTML, etc.) for the creation of an optimized and secure application.
2. **Deployment Services:** Publishes the code that was previously generated in a server, assuring that the application is installed consistently on each front-end of the infrastructure.
3. **Application Services:** Manages the applications during runtime, through the execution of scheduled batches and asynchronous logging services that allow to store events such as errors, audits and performance metrics.

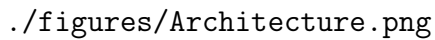
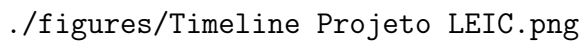
The image is a large rectangular box containing the text `./figures/Architecture.png`. This text likely serves as a placeholder for a diagram or figure that is not visible in the provided image.

Figure 1: OutSystems Architecture

Schedule

The project's management and schedule (Figure 2) will follow an agile [2] designated methodology, where the SCRUM framework [3] will be used since it is the most well-referenced and it enables to increase the productivity and quality of the development. Consequently, the project will be divided into cycles called Sprints, where the development of the project is focused on concluding an incremental solution that results from a previously established objective. Each incremental solution is a fully functional, integrated, tested and documented application in which the final sprint will correspond to the delivery of the final product. In Figure 2 bellow, we can verify the schedule of all the sprints and phases included in the project.



./figures/Timeline Projeto LEIC.png

Figure 2: Project Timeline

The initial phase of the project's development consists of making the OutSystems tutorials followed by a requirements analysis. The sprints that will be made while the development of the project will start with its planning and end with a demonstration and the elaboration of the project's report which will be made progressively throughout the sprints.

References

- [1] <https://www.outsystems.com/>
- [2] <https://www.agilealliance.org/agile101/the-agile-manifesto>
- [3] <https://www.scrum.org/resources/scrum-guide>