



Instituto Superior de Engenharia de Lisboa  
Licenciatura em Engenharia Informática e de Computadores

# Proposta de Projeto: i-on Teams

Martim Francisco, n.º 46912, e-mail: a46912@alunos.isel.pt, tel.: 926586586

Afonso Machado, n.º 47217, e-mail: a47217@alunos.isel.pt, tel.: 962181122

Orientador: Pedro Félix, e-mail: pedro.felix@isel.pt

Março de 2022

## 1 Introdução

Este projeto integra-se na iniciativa **i-on** [1] que visa o desenvolvimento de sistemas de apoio a actividades académicas. O **i-on Teams** tem como objectivo a implementação e disponibilização de uma aplicação para a criação e a gestão de repositórios **GitHub** [2] em contexto académico. Entenda-se repositório como um local onde um projeto é armazenado, local ou remotamente, e pode ser acedido e editado por um ou mais colaboradores, com permissões para tal. **GitHub** é uma plataforma de controlo de versões onde ficam armazenados remotamente repositórios.

Este projeto é uma evolução do **i-on CodeGarten** [3], realizado no ano letivo anterior. As principais diferenças entre os dois projetos serão descritas no ponto **2** - Análise - do presente documento.

Algumas unidades curriculares do ISEL usam a plataforma **GitHub** para a partilha de código e a ferramenta **GitHub Classroom** [4] para a supervisão da evolução do trabalho, e o controlo da(s) entrega(s) de cada grupo num dado projeto. Esta abordagem é vantajosa, no sentido em que introduz os alunos a ferramentas de controlo de versões, utilizados a nível empresarial, que promovem a colaboração entre alunos. No entanto, o **GitHub Classroom** apresenta algumas desvantagens, como a falta de automatização para a verificação de entregas, a dificuldade em identificar o aluno através da sua conta **GitHub** e a atualização das cópias locais, existentes nas máquinas dos docentes para acompanhamento e correção dos repositórios dos alunos.

O projeto **i-on Teams** pretende, não só solucionar os problemas anteriormente descritos, como incitar o uso das ferramentas de controlo de versões tanto por alunos como por docentes através de, entre outras vantagens, o aumento da eficiência e produtividade no uso destas ferramentas em contexto académico.

## 2 Análise

O **i-on Teams** será composto por duas aplicações de *frontend*, uma aplicação web e uma aplicação desktop, para alunos e docentes respetivamente. Ambas as aplicações serão suportadas por uma terceira aplicação servidora, denominada **i-on Teams service**, que irá expor uma API.

As aplicações de *frontend* terão funcionalidades diferentes, sendo que, a aplicação desktop será desenvolvida com recurso à tecnologia **Electron** [5] e será responsável por criar e gerir repositórios através da **GitHub** API e da **i-on Teams service** API. Já a aplicação web será usada para a inscrição de alunos em grupos, para a consulta do estado das entregas que o aluno possa ter pendentes nas diferentes turmas e também terá a possibilidade de ver o enunciado dos trabalhos propostos. Esta última comunicará apenas com a **i-on Teams service** API.

A tecnologia **Electron** foi escolhida por ter a versatilidade de produzir uma aplicação multi-plataforma através de tecnologia web, garantindo que conseguimos produzir um cliente que é compatível com o maior número de dispositivos possível.

A **i-on Teams service API** tem como finalidade gerir os dados do sistema, tal como os utilizadores (alunos e docentes), as turmas, os grupos, os projetos, as entregas, entre outros dados, e expô-los através de uma API.

Na figura 1 será apresentada a interação entre os componentes do nosso projeto.

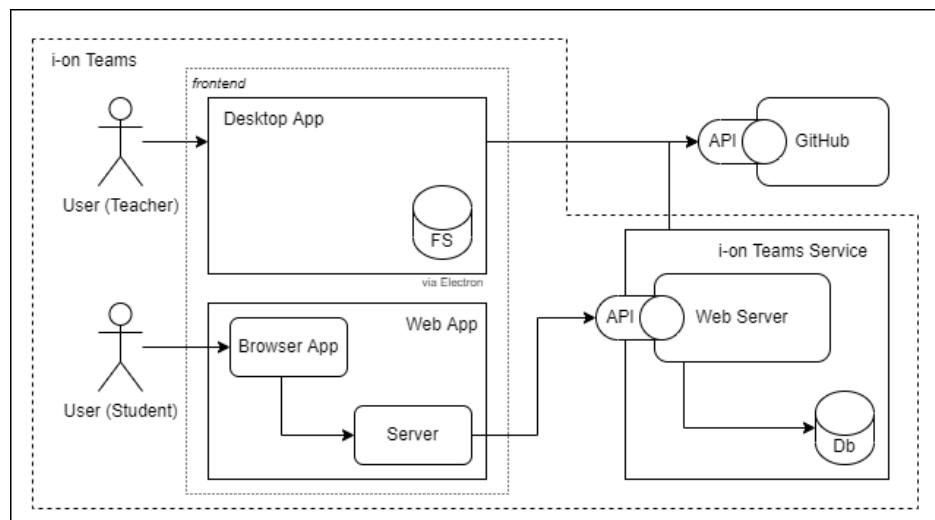


Figura 1: Interação dos componentes i-on Teams

Existem dados armazenados na aplicação servidora que provêm de fontes externas, sendo estes, no contexto do projeto, provenientes do **GitHub**. Para tal, a componente servidora do nosso projeto terá uma ligação com a API do **GitHub** e uma interface que servirá para comunicar com as aplicações clientes, expondo os dados armazenados através de uma API.

Os fatores que diferenciarão este projeto daquele que foi desenvolvido no ano letivo anterior, **i-on CodeGarten**, serão o facto deste ser desenvolvido com recurso a uma aplicação

desktop e uma aplicação web, ao invés de apenas uma aplicação web, a tentativa de otimizar a utilização de armazenamento para viabilizar o seu *deploy* e, por fim, o evitar guardar informação sensível no servidor. Este último ponto traduz-se no facto de o docente ter de aprovar a criação de novos repositórios através de um *token* que apenas este possui localmente.

O **i-on teams** tem como objetivo ser utilizado por todas as instituições de ensino que o desejem. Para que, uma instituição possa utilizar este serviço deverá hospedar uma instância do mesmo. Será também necessário que o docente instale a aplicação desktop na sua máquina por forma a poder gerir as turmas nas quais pretende usufruir do serviço.

### 3 Plano Semanal do Projeto

O desenvolvimento do projecto será planeado às semanas, identificando todas as entregas de Projecto e Seminário. Na figura 2 apresentamos a planificação elaborada.

### Referências

- [1] i-on. <https://github.com/i-on-project>.
- [2] GitHub. <https://github.com/>.
- [3] Diogo Sousa, Tiago David, João Moura. i-on CodeGarten. <https://github.com/i-on-project/codegarten>, 2021.
- [4] GitHub Classroom. <https://classroom.github.com/>.
- [5] ElectronJS. <https://www.electronjs.org/>.

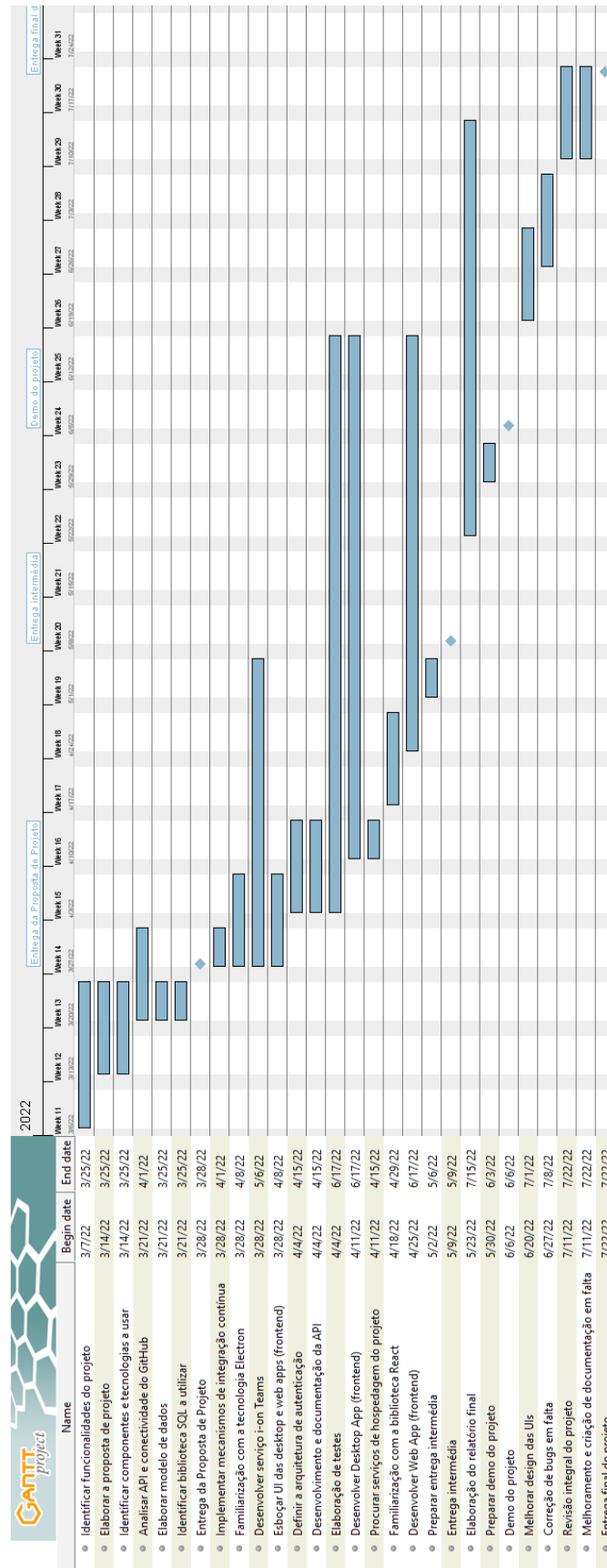


Figura 2: Plano semanal de desenvolvimento