



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

Proposta de Projeto - CodeGarten

Projeto e Seminário

Orientador: Pedro Félix (pedrofelix@cc.isel.ipl.pt)

Diogo Sousa (46008)
Tiago David (46078)
João Moura (46107)

Abril de 2021

Introdução

Atualmente, a forma mais utilizada pelos docentes para a gestão de código desenvolvido pelos seus alunos é a utilização da plataforma *GitHub Classroom*. Porém, esta apresenta várias inconveniências e funcionalidades em falta que dificultam o trabalho dos docentes, como por exemplo:

- a falta de automatização da verificação de entregas dos trabalhos;
- dificuldade em relacionar utilizadores GitHub com os alunos;
- dificuldade em criar/atualizar cópias locais dos repositórios dos alunos.

Com o objetivo de facilitar esta gestão, assim como resolver os problemas existentes no *GitHub Classroom*, foi planeado o desenvolvimento de uma aplicação denominada CodeGarten, que será uma aplicação *web* suportada por uma API REST.

Análise

Como foi referido na introdução, o CodeGarten será uma aplicação *web* suportada por uma API REST. Esta decisão foi feita para possibilitar a criação de aplicações diferentes com o mesmo propósito. Por exemplo, poderão existir aplicações *Android* e *Web* do CodeGarten, e ambas têm as mesmas funcionalidades, comunicando com a mesma API.

A API será desenvolvida utilizando a tecnologia *Spring* na linguagem de programação *Kotlin*. Sendo que a API irá necessitar de armazenar informação, será utilizada um sistema de gestão de base de dados PostgreSQL. No caso da aplicação *web*, esta será desenvolvida em *Node.js* utilizando a biblioteca *React*. Visto que o conhecimento desta biblioteca ainda não se encontra devidamente aprofundado, a aplicação será primeiramente desenvolvida utilizando *Bootstrap*, visto que este foi utilizado anteriormente noutras unidades curriculares. Devido a ser necessário realizar ações no GitHub para gerir os repositórios, é também necessário o uso de uma *GitHub App*. Esta irá agir em nome do CodeGarten na criação e gestão de repositórios.

O CodeGarten terá de ser instalado individualmente em cada instituição de ensino. Isto significa que existirá uma instância da aplicação por cada instituição que pretender utilizar a mesma. Visto que a aplicação depende da existência de uma *GitHub App* para efetuar ações no *GitHub*, cada instância da aplicação irá também necessitar de uma *GitHub App* dedicada a si mesma.

O objetivo principal é obter uma aplicação semelhante ao *GitHub Classroom*, onde seja possível gerir repositórios (relacionados a trabalhos) de alunos, com fácil verificação de entregas e ligação entre utilizador GitHub e aluno da instituição por parte do docente.

Plano Semanal do Projeto

Semana 0 (08 março – 14 março)	Analisar a API do <i>GitHub</i> ¹
Semana 1 (15 março – 21 março)	Analisar o <i>GitHub Classroom</i> ² Esboçar a interação/representação da interface do utilizador Escrever e entregar o formulário do projeto Identificar e documentar requisitos funcionais
Semana 2 (22 março – 28 março)	Investigar a API fluente do <i>JDBI</i> ³ Definir o modelo de dados
Semana 3 (05 abril – 11 abril)	Definir a arquitetura de autenticação e <i>tokens</i> de acesso Escrever e entregar a proposta do projeto
Semana 4 (12 abril – 18 abril)	Atualizar o modelo de dados para suportar a arquitetura de autenticação Finalizar a documentação inicial do projeto Iniciar o desenvolvimento da API (utilizando <i>Spring</i> ⁴) Investigar e configurar Integração Contínua
Semana 5 (19 abril – 25 abril)	Prosseguir com o desenvolvimento da API Iniciar o desenvolvimento de testes Iniciar a documentação da API
Semana 6 (26 abril – 02 maio)	Iniciar o desenvolvimento de um protótipo da aplicação <i>web</i> Investigar possibilidades de hospedagem para a aplicação
Semana 7 (03 maio – 09 maio)	Efetuar o <i>deploy</i> da versão alfa da aplicação
Semana 8 (10 maio – 16 maio)	Iniciar a aprendizagem da biblioteca <i>React</i> ^{5*} Iniciar a migração da aplicação <i>web</i> para <i>React</i> *
Semana 9 (17 maio – 23 maio)	Finalizar a migração da aplicação <i>web</i> para <i>React</i> * Preparar a apresentação do progresso atual do projeto
Semana 10 (24 maio – 30 maio)	Adicionar funcionalidades em falta à aplicação <i>web</i> Apresentar o progresso atual do projeto
Semana 11 (31 maio – 06 junho)	Iniciar a escrita do relatório do projeto
Semana 12 (07 junho – 13 junho)	Melhorar a interface gráfica da aplicação <i>web</i>
Semana 13 (14 junho – 20 junho)	Realizar a demonstração da versão inicial do relatório
Semana 14 (21 junho – 27 junho)	Adicionar documentação relativa a correr uma instância da aplicação
Semana 15 (28 junho – 04 julho)	Finalizar trabalhos pendentes
Semana 16 (05 julho – 13 julho)	Finalizar o relatório do projeto Entregar a versão final do projeto

Texto a negrito define uma milestone

* Esta decisão está sujeita a alterações. A aplicação final poderá utilizar as mesmas tecnologias do protótipo

¹ <https://docs.github.com/en/rest>

² <https://classroom.github.com/>

³ <https://jdbi.org/>

⁴ <https://spring.io/>

⁵ <https://reactjs.org/>