

# Plugin Development for Confluence

Tiago Brito, Tiago Santos

## Relatório de Actividades

**Resumo**—O projecto consistiu no desenvolvimento de um *plugin* para *Confluence* que mantivesse documentação de um projecto Java actualizado numa página *Confluence*. Todo o projecto foi desenvolvido pelo nosso grupo de duas pessoas, Tiago Santos e Tiago Brito, com o objectivo de melhorar o seu currículo e aprender mais sobre Engenharia de *Software*. O projecto foi desenvolvido na sede da IT4Energy no *Tagus Park* durante os dias úteis desde o início Julho ao início das aulas em Setembro. Para o desenvolvimento tivemos de aprender e seguir as práticas da *startup* IT4Energy em relação ao desenvolvimento Java e reportar o nosso progresso a alunos designados dentro da *startup*. Muitas das funcionalidades tiveram problemas durante o seu desenvolvimento devido à sua documentação problemática e para resolvê-las foi consumido mais tempo do que estava planeado. Apesar de todos os problemas, com os conselhos dos nossos colegas da *startup* mais experientes em Java e a nossa perseverância, o projecto foi um sucesso, providenciando todas as funcionalidades pedidas e manual de utilização no tempo disponível.

**Palavras Chave**—Javadoc, plugin, Confluence, IT4Energy, Tagus Park, Instituto Superior Técnico, desenvolvimento ágil, software, Java, startup, equipa, sprints, marketplace, atlassian, SDK, IDE, Eclipse, Source Tree, Hip Chat, Maven, FTP, SFTP, Git, L<sup>A</sup>T<sub>E</sub>X, relatório, verão, grupo, engenharia.

*EXCESSIVO! Deriam por 5 ou 6!*

## 1 INTRODUÇÃO

*Qual?*

O PROJECTO que desenvolvemos no verão passado consistia no desenvolvimento de um *plugin* para a ferramenta colaborativa de desenvolvimento de *software* *Confluence*, da *Atlassian*. O *Confluence* é uma ferramenta que facilita todos os aspectos do desenvolvimento de *software*, tais como: comunicação entre todos os membros da equipa de desenvolvimento, documentação de projectos, tutoriais e ferramentas organizacionais.

O *plugin* consiste numa fácil criação de páginas de documentação *Javadoc* dentro do *Confluence* que se actualiza automaticamente fazendo uso de um servidor FTP/SFTP.

Este Relatório de Actividade de Grupo de Portefólio Pessoal III foi escrito em L<sup>A</sup>T<sub>E</sub>X usando uma versão modificada da classe IE-EETrans.

*Que relação existe com o assunto?*

- Tiago Brito, nr. 72647,  
E-mail: tiago.de.oliveira.brito@tecnico.ulisboa.pt,
- Tiago Santos, nr. 72960,  
E-mail: tiago.f.dos.santos@tecnico.ulisboa.pt,  
Instituto Superior Técnico, Universidade de Lisboa.

Manuscript received January 3, 2015. *1*  
*PORQUE MOTIVO ESTÁ EM INGLÊS?*

## 2 MOTIVAÇÃO

No fim do ano lectivo 2014/2015 decidimos falar com o nosso professor de Engenharia de *Software*, o professor Paulo Carreira, para podermos participar num projecto que nos permitisse aplicar e aprender melhor metodologias ágeis e boas práticas de desenvolvimento de *software* aprendidas na cadeira. Desta forma podíamos ver como estas técnicas são aplicadas num meio mais formal, diferente do contexto dos projectos das cadeiras.

Foi-nos, então, apresentada a ideia do *Javadoc Plugin*, um *plugin* para o *Confluence* que permitiria uma fácil partilha de documentação de um projecto Java, documentação esta que permaneceria actualizada durante o desenvolvimento do projecto num servidor remoto. Assim, qualquer pessoa *envolvida* no projecto tem uma maneira rápida e fácil de consultar a documentação sem ter de usar uma ferramenta alternativa.

Este *plugin* fazia falta à equipa da IT4Energy porque desta forma toda a documentação dos projectos em que estão envolvidos encontra-se centralizada na ferramenta que usam como intranet.

*sempre verbais!!*

	ACTIVITY					DOCUMENT						
	Objectives x2	Options x1	Execution x4	S+C x1	SCORE	Structure x0.25	Ortogr. x0.25	Gramm. x0.25	Format x0.25	Title x0.5	Filename x0.5	SCORE
(1.0) Excellent												
(0.8) Very Good												
(0.6) Good												
(0.4) Fair												
(0.2) Weak												
	1.8	0.6	3.6	0.6	6.6	0.2	0.15	0.15	0.25	0.5	0.5	1.75

O *plugin* tinha como objectivo venda, na *Atlassian Marketplace*, e uso por parte da IT4Energy no desenvolvimento dos seus projectos de *software*. Como tal ficámos responsáveis pelo planeamento, gestão, desenho, desenvolvimento e documentação de todo o projecto, o que se demonstrou um enorme desafio técnico, social e organizacional para nós já que nunca tínhamos desenvolvido nada com esta dimensão.

onde?

### 3 AMBIENTE DE TRABALHO

Durante o desenvolvimento deste projecto trabalhamos no Campus do IST situado no *Tagus Park* e incorporados numa *startup* chamada IT4Energy.



Figura 1. Logótipo da IT4Energy

Esta *startup* consiste num grupo de alunos, liderados pelo professor Paulo Carreira, que desenvolvem soluções tecnológicas para aplicações de energia.

Durante o tempo que estivemos na IT4Energy trabalhamos numa pequena sala que acolhia todos os alunos da *startup*. Todos os elementos desta equipa tinham um papel bem definido dentro da organização e por isto foi-nos indicado, logo no primeiro dia, a quem devíamos pedir ajuda caso esta fosse necessária.

Ainda assim, todos se mostraram disponíveis, o que se reflectiu numa valiosa ajuda no projecto a que nos tínhamos comprometido.

Todo o ambiente era descontraído mas sério, todos nós sabíamos que estávamos ali para acabar um projecto, no nosso caso o *Javadoc Plugin for Confluence*. Ainda assim havia espaço e tempo durante as pausas para desanuviar a cabeça dos problemas técnicos que cada um tinha e isto levou a um crescente espírito de

camaradagem entre toda a equipa ao longo do verão, o que tornou toda a experiência gratificante.

Foi fundamental saber qual o lugar de cada um na IT4Energy pois, por estarmos a desenvolver um projecto seguindo as guias e linhas de orientação de *Agile Development* foi necessário seguir regras de trabalho bem definidas. Todos os dias eram realizadas reuniões gerais (*briefings*) para falarmos sobre os progressos nos trabalhos que estávamos a desenvolver, assim como de novos problemas que possam ter surgido.

sempre verbais !!

Para além disto tivemos também que enviar emails diários para o professor Paulo Carreira assim como para outro elemento da IT4Energy (chefe de equipa) de maneira a mantê-los actualizados das dificuldades e actualizações do projecto.

### 4 PREPARAÇÃO E PESQUISA

Como primeiro passo tivémos de saber como se fazia um *plugin* para *Confluence*. Rapidamente descobrimos que a *Atlassian*, a empresa que desenvolveu e vende o *Confluence*, tinha uma SDK disponibilizada para desenvolvimento de *plugins* e que a linguagem usada era Java.

Esta SDK usava *Maven* e o nosso conhecimento de *Maven* era, infelizmente, não existente. Para desenvolvimento de *software* em Java, os nossos colegas da IT4Energy indicaram-nos como proceder em relação ao IDE (*Eclipse*) e ao repositório (*Git*).

Visto que era uma *startup* que seguia o seu conjunto de regras e boas práticas, tivémos que aprender as mesmas e de cumpri-las ao longo de todo o desenvolvimento.

Durante um período de 3 dias estivémos a configurar o *Eclipse* e a aprender as práticas de *GitFlow* usadas na IT4Energy assim como a ganhar prática no uso de ferramentas auxiliares de desenvolvimento usadas com o *Hip Chat* (uma aplicação de *chat* usada entre os membros da equipa), o *Source Tree* (uma ferramenta que facilita o controlo de versões usando o repositório *Git*) e o próprio *Confluence* (usado como intranet dentro da *startup* para partilha de ficheiros, tutoriais e documentação geral dos projectos).

## 5 DESENVOLVIMENTO

A maior parte do tempo de desenvolvimento, como em qualquer projecto em que os desenvolvedores não estão familiarizados com tecnologia, foi usado em pesquisa. Pesquisa em relação ao tipo de *plugin* disponíveis [1], como funcionam, que componentes o compõem e como poderíamos usá-lo para cumprir o nosso objectivo. *similarity!*

A maior parte dos problemas que tivemos ao longo do projecto foram devido à documentação da SDK disponibilizada pela *Atlassian* para o desenvolvimento de *plugin's* para as suas ferramentas. A documentação da SDK para o *Confluence* estava desactualizada, contraditória - partes da documentação referiam-se a diferentes versões da SDK - e em alguns casos confusa ou pouco explícita, com exemplos demasiado básicos ou que não se focam nos problemas com que nos deparámos.

Finalmente optámos por um tipo em específico de *plugin* que achámos que iria cumprir os nossos objectivos, o tipo *Macro*. A função de um *plugin* do tipo *macro* é encapsular comportamento a partir de um conjunto de argumentos fornecidos pelo utilizador e reflectir esse comportamento numa página que o utilizador deseje. Comportamento este, que nosso caso seria comunicar com um servidor, fazer *download* do *javadoc* actualizado e representá-lo na página.

Como a SDK era baseada em Java foi fácil aplicar os conhecimentos obtidos em Engenharia de Software e desenvolver uma arquitectura por camadas para o nosso *plugin*. Para simplificar o problema, decidimos usar bibliotecas da *Apache* que permitem fazer comunicação com os servidores desejados e fazer algum tratamento de nomes de ficheiros pois assim não tivemos que implementar tudo no projecto, ao invés confiamos que as bibliotecas da *Apache* são sólidas, funcionais e estão bem construídas.

Uma directiva recorrente em Engenharia Informática é "não inventar a roda" e, sempre que possível, utilizar *software* extensamente testado e de confiança como é o caso das bibliotecas da *Apache*. Ainda assim, apesar desta simplificação não foi sempre trivial o desenvolvimento do *plugin* e, em várias fases, foi necessário o apoio

dos nossos colegas de maneira a ultrapassar os problemas que foram aparecendo.

À excepção das bibliotecas da *Apache* usadas todos os restantes módulos foram desenvolvidos por nós, desde o *back-end* em Java, que liga as funcionalidades das bibliotecas referidas, ao Javascript, HTML e CSS que representam o *front-end* da aplicação.

### 5.1 Múltiplas páginas *Javadoc* numa página *Confluence*

Um dos problemas com que nos deparámos no desenvolvimento do projecto foi a multiplicidade de páginas *Javadoc* associadas a um projecto Java que dependem do número de classes que um projecto tem e, naturalmente, da sua complexidade.

Pensámos inicialmente em criar várias páginas *Confluence*, uma para cada página *Javadoc* existente. Isto teria uma complexidade muito alta, devido à necessidade de sabermos como funciona a hierarquia de páginas no *Confluence* e os módulos programáticos que as gerem.

Felizmente, um colega da *startup*, Pedro Domingues, ajudou-nos a conceber uma solução mais simples para o problema. Ele, como tem muita experiência no desenvolvimento de aplicações *web*, (que é o caso do *Confluence*) ocorreu-lhe que no URL pudesse ser passada a página *Javadoc* desejada como argumento para a página do *Confluence*. Solução esta que foi posta em prática após a descoberta do módulo programático responsável pelo URL.

A solução consistiu na passagem do caminho relativo de uma página, quando se acede a uma hiperligação *Javadoc*, para um argumento no URL através de javascript e uma subsequente leitura do argumento no código Java que permitiu imprimir dinamicamente na mesma página *Confluence* várias páginas *Javadoc* de um dado projecto Java.

### 5.2 Passagem de argumentos

O tipo de *Plugin Macro* permitia que uma *macro* tivesse comportamento dependente dos argumentos que o utilizador lhe fornecia e para

! Continuação

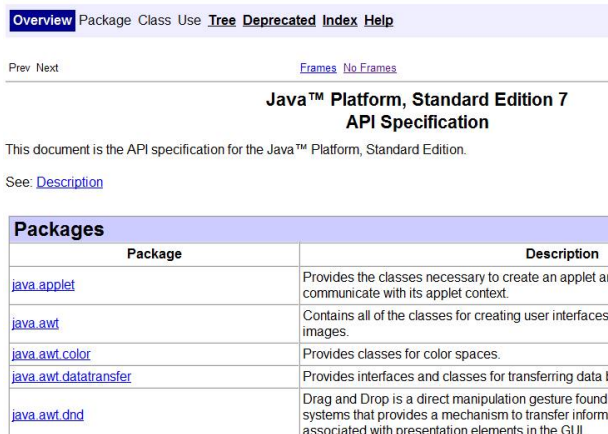


esse efeito disponibilizava uma interface customizável e extensível para o número de argumentos desejado. Devido à natureza do nosso projecto, para aceder a um servidor FTP/SFTP era necessária uma *password* que, em HTML, deveria ser escondida. Infelizmente, apesar de ser possível fazê-lo, a documentação do procedimento estava desactualizada, sendo que foi necessário pesquisar no código fonte disponibilizado pela *Atlassian* e perguntar na plataforma disponibilizada para esse efeito (*Atlassian Answers*) [2] para descobrir o procedimento correcto. Foi necessária uma semana e meia em paralelo com outras tarefas para descobrir a solução a este problema, solução esta partilhada na plataforma [2].

### 5.3 Correção de estilo

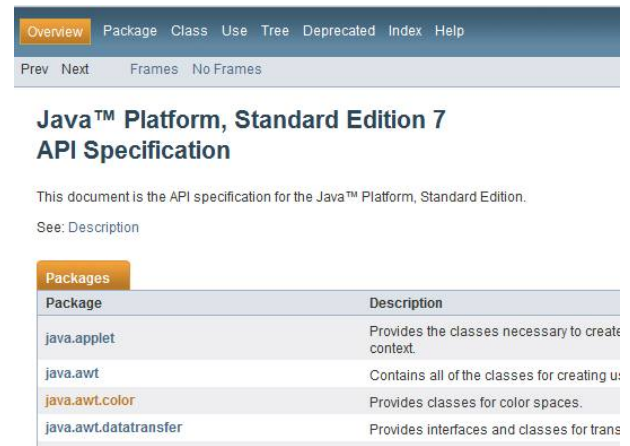
Uma das vantagens das reuniões diárias com o professor Paulo Carreira, que representava, para a ocasião, o nosso cliente, era que se tornava possível entender qual o produto final que havia sido visualizado.

Uma das coisas que entendemos foi que o formato original de um documento *Javadoc* não é esteticamente apelativo quando integrado com o *Confluence*. Como tal, durante uma das reuniões diárias já referidas, propôs-se que fosse desenhado um novo design para uma página *Javadoc* de forma a que esta se integrasse melhor com o *Confluence*. Esta proposta foi aceite pelo professor e como tal começou-se a desenhar um formato de *Javadoc* complementar ao *Confluence*.



Package	Description
<a href="#">java.applet</a>	Provides the classes necessary to create an applet and communicate with its applet context.
<a href="#">java.awt</a>	Contains all of the classes for creating user interface images.
<a href="#">java.awt.color</a>	Provides classes for color spaces.
<a href="#">java.awt.datatransfer</a>	Provides interfaces and classes for transferring data.
<a href="#">java.awt.dnd</a>	Drag and Drop is a direct manipulation gesture found systems that provides a mechanism to transfer information associated with presentation elements in the GUI.

Figura 2. Formato Javadoc original 1



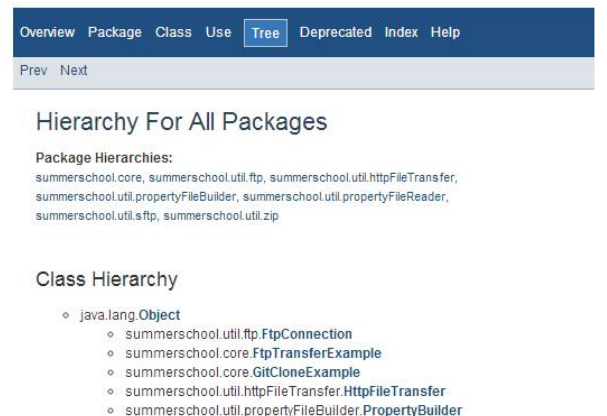
Package	Description
<a href="#">java.applet</a>	Provides the classes necessary to create context.
<a href="#">java.awt</a>	Contains all of the classes for creating user interfaces.
<a href="#">java.awt.color</a>	Provides classes for color spaces.
<a href="#">java.awt.datatransfer</a>	Provides interfaces and classes for transferring data.

Figura 3. Formato Javadoc original 2

Optou-se por manter a estrutura original do documento já que o problema da falta de complementaridade entre os formatos se devia maioritariamente às cores usadas no *Javadoc* original assim ao footer original que para o caso se tornava desnecessário.

Assim procedeu-se à alteração das cores do formato original assim como à remoção do footer e à remoção de hiperligações e outros detalhes desnecessários para a documentação Java.

De seguida podemos ver a versão *Javadoc* que foi desenhada especificamente para a aplicação.



Package Hierarchies:
summerschool.core, summerschool.util.ftp, summerschool.util.httpFileTransfer, summerschool.util.propertyFileBuilder, summerschool.util.propertyFileReader, summerschool.util.sftp, summerschool.util.zip

Class Hierarchy
java.lang.Object
summerschool.util.ftp.FtpConnection
summerschool.core.FtpTransferExample
summerschool.core.GitCloneExample
summerschool.util.httpFileTransfer.HttpFileTransfer
summerschool.util.propertyFileBuilder.PropertyBuilder

Figura 4. Formato Javadoc re-desenhado.

### 5.4 Atalhos

Sempre que um utilizador quer criar uma página *Confluence* com documentação Java tem que inserir manualmente a *Macro* (*plugin*) na página.

*figura deve estar referida no texto!*

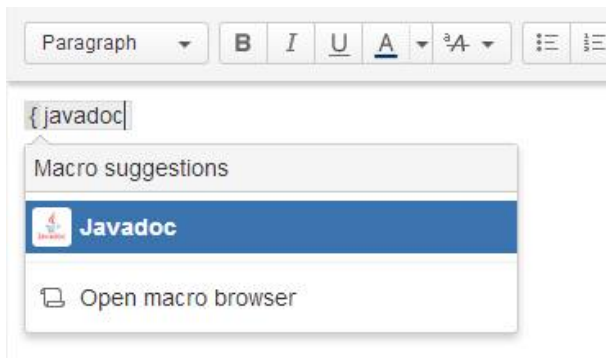


Figura 5. Inserção manual da *Macro* numa página *Confluence*.

Tratando-se este projecto de uma aplicação desenvolvida para o uso regular deve ser simples, acessível e fácil de aprender e como tal procurámos desenvolver, dentro do *Confluence*, opções que permitissem executar a nossa aplicação com o número mínimo de passos possíveis para o utilizador.

Para possibilitar isto procurámos vários mecanismos que permitem inserir uma *Macro* numa página do *Confluence*. Foi então que descobrimos como colocar em destaque o nosso *plugin* na opção de inserção de *Macro's* assim como criar uma página *Confluence* que contém já a nossa *Macro*, a chamada *Blueprint*.

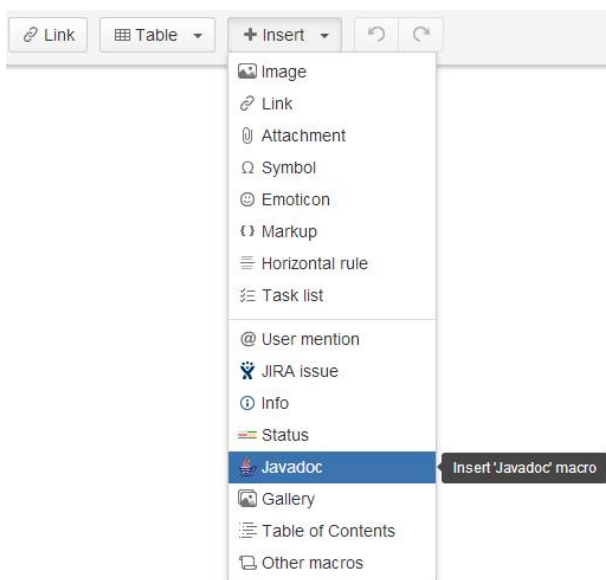


Figura 6. Menu de inserção de *Macro's* *Confluence*.

A *Blueprint* permite que um utilizador consiga criar uma página *Confluence* para *Javadoc*

sem ter que, manualmente, inserir a *Macro* numa página em branco, o que dá a toda a aplicação um novo nível de usabilidade.

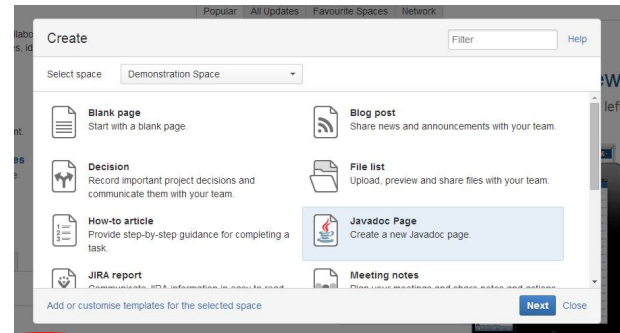


Figura 7. Menu de inserção de *Macro's* *Confluence*.

## 6 MANUAL DE UTILIZADOR

Como o objectivo do *plugin* era a venda do mesmo, também produzimos o manual de utilizador. O manual foi produzido sem grandes complicações e foi sendo melhorado ao longo do processo com *input* tanto dos nossos colegas, pertencentes à *startup*, como do professor.

Link para o download do Guia do Utilizador

## 7 CONCLUSÃO

Todo o trabalho foi uma experiência fantástica em termos de aprendizagem. Aprendemos como funciona uma *startup* de perto, visto que estivemos na sala onde esta funcionava.

Aprendemos a lidar com cada problema e falar com o nosso cliente (neste caso o Professor Paulo Carreira) em relação às diversas soluções possíveis assim como sobre sugestões em relação à aparência do projecto. Isto mostrou-nos quão difícil pode ser fazer uma "pequena" alteração para cumprir as expectativas do cliente.

Finalmente, fazer um programa de início ao fim, desenvolvendo todos os passos iniciais, intermédios e finais deu-nos uma perspectiva de como trabalhar num projecto de grandes dimensões e com um propósito que não a avaliação académica.

Infelizmente não conseguimos colocar ainda o *plugin* à venda, mas esperamos que seja possível fazê-lo num futuro próximo.

*Figura deve estar referida no texto!*

## AGRADECIMENTOS

Gostaríamos de agradecer ao professor Paulo Carreira por nos providenciar preciosa oportunidade. Ao Gonçalo Almeida por ter toda a paciência do mundo para esclarecer as nossas dúvidas. Ao João Pinho por nos ter guiado quando mais estávamos perdidos no projecto. E ao Filipe Bento, por ter sido um companheiro em tecnologias alienígenas.

## REFERÊNCIAS

- [1] (2014) "confluence plugin module types". [Online]. Available: <https://developer.atlassian.com/display/CONFDEV/Confluence+Plugin+Module+Types>
- [2] (2014) "atlassian answers". [Online]. Available: <https://answers.atlassian.com/>

**Tiago Brito** Sempre tentei aprender novas coisas e ser um membro activo na comunidade.

Sempre estive envolvido em desportos como as artes marciais (Judo, Karate e Aikido), nas quais cheguei a níveis de reconhecimento e posições de pódio em algumas competições nacionais, aprendi a velejar num barco da classe Optimist e sempre fui um grande fã de carros, nomeadamente de karts e competições de trial todo-o-terreno.

Em 2009 fui a Marrocos para trabalhar como voluntário numa expedição todo-o-terreno cujo objectivo era viajar até aldeias remotas, cujas condições de acesso são débeis, e doar roupas, água, comida e outros recursos.

Sempre fui um interessado em construir e imaginar coisas novas, interessantes e práticas e foi isso que me levou a concluir a Licenciatura em Engenharia Informática e de Computadores pelo Instituto Superior Técnico assim como a continuar a minha aventura académica pelo mestrado nesta mesma instituição e, quem sabe, continuar no futuro sempre a aprender novas coisas.

**Tiago Santos** O meu nome é Tiago Santos, e acabei no ano lectivo passado a minha Licenciatura em Engenharia Informática e de Computadores no Instituto Superior Técnico. Estou agora no primeiro ano do Mestrado na mesma área majorando em Engenharia de Software e minorando em Sistemas Inteligentes. Sempre gostei de Engenharia de Software e gostaria de ser arquitecto de Software no futuro.

**APÊNDICE****COMPROVATIVOS DA ACTIVIDADE**

Serão entregues nos dias posteriores à entrega deste relatório.

