



UNIVERSIDADE  
LUSÓFONA

# Plugin DropProject V2

## Trabalho Final de curso

Relatório Intercalar 1º Semestre

Nome do Aluno: Diogo Casaca

Nome do Orientador: Bruno Cipriano

Nome do Co-Orientador: Rodrigo Correia

Trabalho Final de Curso | LEI | 22-01-2021

[www.ulusofona.pt](http://www.ulusofona.pt)

## **Direitos de cópia**

*Plugin Drop Project V2, Copyright de Diogo Casaca, ULHT.*

A Escola de Comunicação, Arquitectura, Artes e Tecnologias da Informação (ECATI) e a Universidade Lusófona de Humanidades e Tecnologias (ULHT) têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

---

## Resumo

No âmbito deste trabalho final do curso de LEI, realizei uma abordagem ao trabalho existente (extensão para o IntelliJ). Criei um questionário para os docentes da minha faculdade e fiz um estudo de mercado de aplicações semelhantes, para ajudar a justificar a viabilidade e pertinência do projeto. Usei os resultados do meu estudo e do questionário para alcançar os requisitos finais. Descrevi as funcionalidades existentes da extensão que vamos melhorar e em continuidade propus uma solução para mitigar atuais problemas e acrescentar novas funcionalidades

## **Abstract**

As a matter of this final year project of Computer Engineering undergraduate degree, an approach to the existing project (extension to IntelliJ) was carried out. I created a questionnaire for the teachers of my college, and made a market analysis of similar projects to help justify the feasibility and relevance of my project. I used the results of the study and questionnaire to achieve the final requirements. I described the functionalities that are in the existing plugin that I am going to improve, and proposed a solution to mitigate current problems and add new functionalities.

---

# Índice

Resumo.....	iii
Abstract .....	iv
Índice.....	v
Lista de Figuras .....	vi
Lista de Tabelas .....	vii
1 Identificação do Problema .....	1
2 Viabilidade e Pertinência.....	3
3 Requisitos .....	4
4 Solução Proposta.....	6
5 Benchmarking.....	8
6 Calendário .....	11
Bibliografia .....	12
Anexo 1 – Questionário.....	13
.....	15
.....	16
Glossário.....	17

## **Lista de Figuras**

Figura 1 - Menus e submenus Plugin DropProject

Figura 2 – Login Plugin Drop Project

Figura 3 - Resultado da possível aderência dos docentes ao plugin

Figura 4 - Diagrama da ligação entre o servidor do DP com o InteliJ e seu plugin.

Figura 5 - Interface Raccode dentro do Eclipse

Figura 6 – Exemplo Lista de exercícios submetidos Acode Sistema Web

Figura 7 – Exemplo Relatório de submissão Acode Sistema Web

Figura 8– Exemplo de animação para aprendizagem Java Tool

Figura 9- Resposta à primeira pergunta: “Utilizaria o plugin do DropProject no InteliJ?”

Figura 10- Resposta à segunda pergunta: “Ajudaria no feedback ao aluno a possibilidade de deixar um comentário associado a uma submissão?”

Figura 11- Resposta à terceira pergunta: “Seria prático ter uma opção de fazer download da última submissão do aluno?”

Figura 12 - Resposta à quarta pergunta: “Que opções gostaria de ter para ordenar/filtrar/pesquisar os assignments”

Figura 13 - Resposta à quinta pergunta: “Para Além das opções apresentadas em cima, gostaria de outras opções de ordenação/filtragem/pesquisa? Qual ou quais?”

Figura 14 - Resposta à sexta pergunta: “Que outras funcionalidades gostaria que existissem neste Plugin”

---

# Lista de Tabelas

Tabela 1 - Calendário de tarefas





# 1 Identificação do Problema

## Estado Inicial

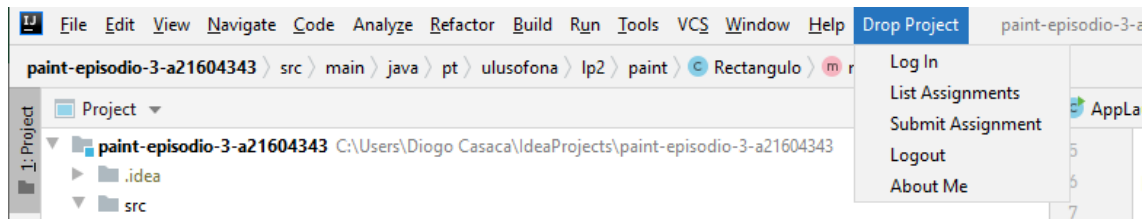
O projeto a ser realizado e descrito vem em contexto de um software já existente, denominado “Drop Project (DP)” [1]. O **DP** foi desenvolvido por docentes da Universidade Lusófona, e tem como objetivo a avaliação automática de programas, com efeito em exercícios de aula, mini-testes, projetos finais e defesas de projetos. A ideia a desenvolver é a continuação do trabalho realizado pelo aluno Yash Jahit, que criou uma extensão (*Plugin*) para o Ambiente de Desenvolvimento Integrado (IDE) pré-selecionado IntelliJ. Este *Plugin* consiste na integração no próprio ambiente de trabalho do *software* em cima referido.

O *Plugin* [2] desenvolvido pelo colega, focou apenas na vertente do aluno que contemplou as seguintes funcionalidades:

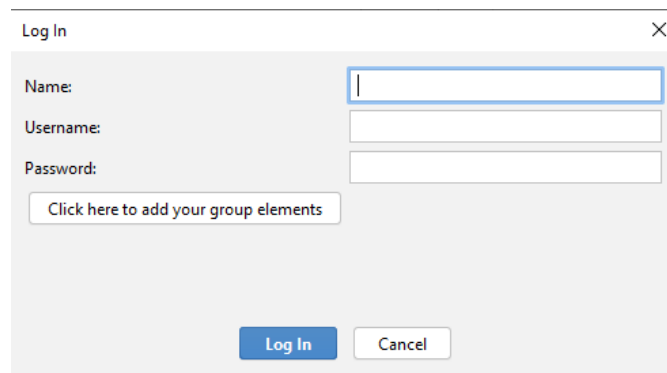
- **Toolbar** : Adicionado ao menu bar do IntelliJ um menu extra denominado “Drop Project”, contendo 5 opções.
- **Login** : Através das credenciais: Nome, número de aluno e *password*. Podendo ainda adicionar o número e nome do respectivo colega de grupo. É criado um ficheiro `authors.txt` contendo as informações do grupo.
- **List Assignments** : Disponibiliza todos os assignments disponíveis. Para cada assignment tem a possibilidade de visualizar as submissões efectuadas, os detalhes e seleccionar o assignment que pretende efectuar os testes.
- **Submit Assignment** : Envia o conteúdo do projeto, incluído o ficheiro `authors.txt` para ser testado pelo DP, atualizando as submissões na lista de assignments.

## Capturas de Ecrã Plugin Drop Project:

Apresentam-se de seguida captura de ecrã do plugin atual do Drop Project. Na Figura 1 apresenta-se a forma como é executado o plugin, podemos ver o menu “Drop Project” adicionado ao IntelliJ e os seus sub-menus listados (*Log in*, *List Assignments*, *Submit Assignment*, *Logout*, *About Me*). Na Figura 2 apresenta-se o ecrã no momento de *login*, mostrando as credenciais necessárias e as opções disponíveis.



**Figura 1 - Menus e submenus Plugin DropProject**



**Figura 2 – Login Plugin Drop Project**

## 2 Viabilidade e Pertinência

Seja a nível escolar ou empresarial, a utilização de ambientes de desenvolvimento de software por parte dos programadores, como os Ambientes de Desenvolvimento Integrado (IDEs), tem vindo a aumentar significativamente.

Estes ambientes permitem o desenvolvimento e a integração de ferramentas (plugins) que, quando bem desenhadas, vêm simplificar as tarefas do utilizador e melhorar a produtividade.

Um plugin é uma extensão, um componente que é possível integrar num IDE, oferecendo-lhe mais funções e recursos.

Um outro ponto é a presença forte da Internet no quotidiano das pessoas, que disponibiliza inúmeras ferramentas capazes de corresponder aos objetivos dos utilizadores. Existem muitos *softwares* para resolução de problemas de programação, mas muito poucos desses softwares são integrados no ambiente de trabalho. Seria útil ao programador fazer a utilização destes serviços num ambiente que lhe é familiar e que utiliza diariamente, ou pelo simples facto de servir como método de aprendizagem, tanto para uma linguagem de programação (ao resolver os problemas) como para o ambiente de trabalho escolhido (através da sua utilização).

Sendo este projeto continuação da extensão já desenvolvida, existindo já um plugin que integra o DP com o IntelliJ contemplando as funcionalidades do aluno bem como de avaliação automática. Deverá ser então viável e possível realizar a vertente do professor e algumas melhorias na vertente do aluno.

O DP, atualmente, já tem um forte impacto positivo na aprendizagem e avaliação dos alunos da Universidade Lusófona, sendo usado em quatro disciplinas diferentes (Fundamentos da Programação, Algoritmia e Estruturas de Dados, Linguagem de Programação 2 e Computação Móvel) e abrangendo seis professores, onde facilitaria o seu trabalho, usufruindo diretamente no ambiente de trabalho, de todas as funcionalidades disponibilizadas pelo DP, através da extensão proposta. Até à data o DP já conta com 184 *assignments*, pelos quais participaram 430 alunos realizando um significativo total de 47 000 submissões.

Para maximizar a pertinência e a utilidade, realizou-se um questionário a todos os docentes que interagem com o DP, que tem como objetivo entender as necessidades dos docentes, bem como trabalhar o *workflow* da aplicação, obtendo um ponto de vista de utilizadores ativos no DP, bem como futuros reais utilizadores da extensão em desenvolvimento.

Os resultados do questionário realizado, através do gráfico representado em baixo, demonstra-nos que cem por cento dos docentes inquiridos, apoiam o desenvolvimento deste plugin do DP para o IntelliJ.

Utilizaria o plugin do Drop Project no IntelliJ?

6 respostas

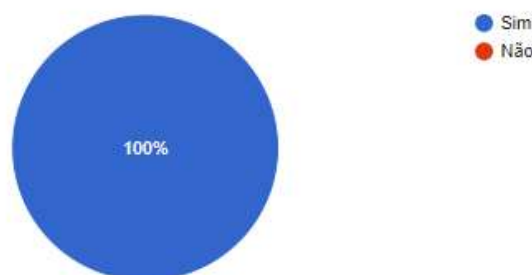


Figura 3 - Resultado da possível aderência dos docentes ao plugin

### 3 Requisitos

Após a análise a aplicações semelhantes no mercado e aos resultados obtidos através do inquérito realizado aos docentes da Universidade Lusófona, foram obtidos os requisitos abaixo referidos.

1 - "Como professor, consigo ver uma lista dos *assignments* aos quais tenho acesso, para poder escolher aquele no qual quero trabalhar."

2 - "Como professor, consigo ver uma lista de todos os alunos que submeteram pelo menos uma vez num *assignment*, para poder saber que alunos estão a trabalhar no mesmo."

3 - "Como professor, consigo ver para um determinado aluno, em quais *assignments* fez pelo menos uma submissão, para avaliar tanto a participação como analisar o seu desempenho."

4 - "Como professor, consigo ordenar/pesquisar *assignments* por: Nome / Data / Ativo ou Arquivado / Última submissão, para facilitar o acesso ao *assignment* pretendido."

5 - "Como professor, consigo ordenar/pesquisar submissões por: Número de aluno/ Nome do aluno / Ordem submissão / Data, para facilitar a identificação das submissões."

6 - "Como professor, consigo criar e/ou alterar um enunciado de um *assignment* no *plugin*, criando um repositório no github com o projeto e criando também um *assignment* no servidor *web* do DP, para facilitar processo todo, evitando a ida ao github e ao próprio servidor *web* do DP."

7 - "Como professor, consigo visualizar o relatório de cada submissão realizada, para rápido conhecimento do estado de progresso do grupo em questão."

8 - "Como professor, consigo fazer download da submissão selecionada ou da última submissão do aluno/grupo, para no segundo caso automaticamente aceder à submissão mais recente do aluno/grupo e para objetivo final, posteriormente aceder ao trabalho do aluno/grupo no seu próprio ambiente de trabalho."

9 - “Como professor, consigo na lista de submissões, selecionar uma ou mais submissões para download abrindo cada uma individualmente em janelas diferenciadas, para evitar a repetição do processo individual de download e facilitando a comparação entre submissões”

10 - “ Como professor, consigo após a análise da submissão do aluno/grupo, deixar um comentário associado à referida submissão analisada, para facilitar, e por conseguinte tornar mais rápido o feedback ao aluno e que este saiba a que submissão corresponde.”

11 - “Como professor, consigo atualizar uma submissão na qual tenha feito o download para a submissão mais recente que o aluno realizou, para agilizar o processo de ir procurar à lista de submissões do aluno num determinado assignment a última submissão realizada.”

12 - “Como professor, consigo personalizar/guardar configurações, para enquadrar no estilo de trabalho de cada docente.”

13 - “Como aluno, consigo pesquisar/ordenar os assignments disponíveis por: Nome / Data / Disciplina, para facilitar ao aluno o acesso ao assignment pretendido.”

14 - “Como aluno, consigo personalizar/guardar configurações, para enquadrar no estilo de trabalho do aluno.”

## 4 Solução Proposta

Em continuidade ao trabalho desenvolvido, o foco principal será em torno da perspectiva do professor (assinalado a verde no diagrama a baixo), onde se pretende disponibilizar ferramentas que facilitem a avaliação e o apoio aos alunos pelos docentes responsáveis bem como melhorar algumas funcionalidades já existentes, anteriormente dedicadas apenas aos alunos.

Diagrama de arquitectura da solução

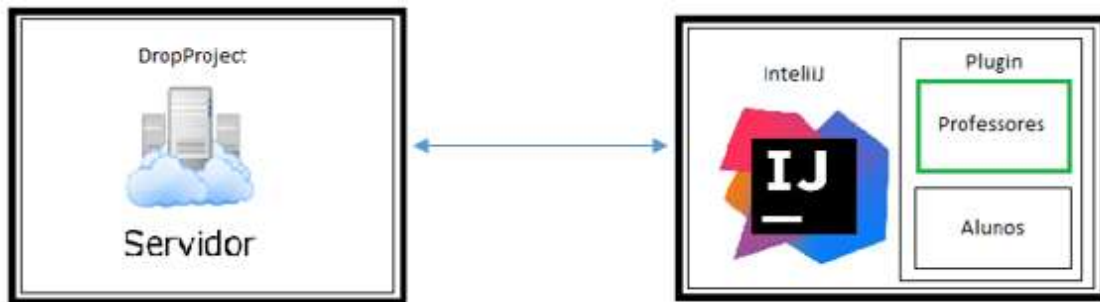


Figura 4 - Diagrama da ligação entre o servidor do DP com o IntelliJ e seu plugin.

### Vertente Docente:

- **Interface** : Criação de uma interface na perspectiva do docente, focando na usabilidade.
- **Login** : Permitir que os professores façam a autenticação com o seu respectivo número, *password* e as disciplinas que leccionam.
- **Listar Assignments do Professor**: Disponibilizar todos os *assignments* em que o respectivo professor tem acesso.
- **Listar alunos que submeteram ao assignment** : Listagem de alunos que já realizaram submissões num determinado *assignment*
- **Listar submissões por aluno por assignment** : Em seguimento do ponto anterior, será disponibilizado ao docente a possibilidade de visualizar todas as submissões de um determinado aluno.
- **Fazer download submissão aluno** : Introduzir a funcionalidade de *download* de uma submissão realizada pelo aluno de um determinado *assignment* para o dispositivo do docente.
- **Preparar download para estar pronto para teste/execução** : Automatizar a abertura e configuração das submissões, permitindo ao docente abrir e testar no seu computador o projeto que o aluno submeteu, agilizando e tornando mais rápido tanto a avaliação como o *feedback* ao aluno.
- **Submissão**: Adaptar a já existente funcionalidade de submissão nos alunos para os docentes terem a possibilidade de testar o projeto pretendido no DP.
- **Filtro/Pesquisa/Ordenação** : Introdução da possibilidade dos professores organizarem/pesquisarem os assignments disponíveis por variáveis como ID, nome, data, se está activo ou arquivado entre outras.
- **Configurações** : Adicionar a possibilidade de personalização de algumas funcionalidades de acordo com os objetivos do professor.

Vertente **Aluno**:

- **Melhorar interface já existente** : A aplicação atual contém uma interface pouco desenvolvida. Será pensada e atualizada para uma vertente mais intuitiva e bem relacionada.
- **Redução de estados para atingir objetivo** : Facilitação da utilização do plugin pelos alunos, focando na usabilidade e simplificação para ambientes de avaliação.
- **Filtro/Pesquisa/Ordenação** : Introdução da possibilidade dos alunos organizarem/pesquisarem os assignments disponíveis por variáveis como ID, nome, data, entre outras.
- **Configurações** : Adicionar a possibilidade de personalização de algumas funcionalidades de acordo com os objetivos do aluno.

Ainda para trabalhar e/ou melhorar algumas funcionalidades, está em análise a possibilidade de se fazer um estudo de usabilidade e/ou um questionário aos alunos utilizadores do DP e do IntelliJ.

## 5 Benchmarking

As áreas de avaliação automática e ambientes de aprendizagem estão em crescimento. Ao analisarmos o mercado em geral, reparamos numa maior incidência em aplicações web e poucas aplicações diretamente integradas no respectivo ambiente de desenvolvimento, realçando a necessidade de aumentar esta vertente do mercado.

Ao focar nessa minoria, encontramos aplicações como:

- Raccode[3]

**Raccode** - Integrou dentro do ambiente de desenvolvimento Eclipse o já existente e utilizado Mooshak[4]. Este *Plugin* inclui uma interface e funcionalidades como visualização do enunciado das questões, submissão de projetos, *feedback* ao aluno, sistema de *rankings* e um ambiente de perguntas e respostas entre alunos.



Figura 5 - Interface Raccode dentro do Eclipse

Focando no mercado mais geral, encontramos várias aplicações como:

- Acode sistema web [5]
- Java Tool [6]
- UVA Tool [7]

Em seguimento apresentam-se informações sobre cada uma das alternativas:

**Acode sistema Web** - Onde se centrou no feedback ao aluno em tempo útil, desenvolvendo um sistema web e um módulo que permite a avaliar o código fonte do aluno consoante o domínio da questão, gerando um relatório com todas as informações positivas e negativas necessárias ao aluno.



**Lista de Exercícios Submetidos**

**Legenda**

● Para que possa ver a descrição mais detalhada, clique no símbolo do projecto respectivo.

**Projectos Avaliados**

# Projecto	# Aluno	Data Entrega	Projecto Entregue	Nota	Ver Relatório
1	●●●●●	15-08-2012 22h05m07s	Download	20 Valores	●●●●●
5	●●●●●	17-08-2012 10h00m43s	Download	20 Valores	●●●●●
8	●●●●●	17-08-2012 18h01m02s	Download	20 Valores	●●●●●
10	●●●●●	17-08-2012 21h34m39s	Download	20 Valores	●●●●●
8	●●●●●	20-08-2012 17h19m07s	Download	17 Valores	●●●●●
6	●●●●●	20-08-2012 17h23m52s	Download	20 Valores	●●●●●
1	●●●●●	20-08-2012 18h18m06s	Download	20 Valores	●●●●●
1	●●●●●	20-08-2012 18h11m41s	Download	19 Valores	●●●●●

Fonte: Marcos Pinto (2012)

Figura 6 – Exemplo Lista de exercícios submetidos Acode Sistema Web

**Relatório Geral**

Nota Final: **20 Valores**

Testes Correctos: **6 de 6** - Nota: **20.00**

Resultado Métricas: **20.00**

**Relatório Completo**

**Relatório do Projecto**

Número de Classes: **4 de 4** - Nota: **20.00**

Número de Métodos: **20 de 20** - Nota: **20.00**

Complexidade Ciclomática: **38 de 38** - Nota: **20.00**

Número de Linhas: **140 de 140** - Nota: **20.00**

Número de Atributos: **16 de 16** - Nota: **20.00**

Nota Final Atribuída: **20.00 Valores**

**Relatório das Classes**

Nome da Classe: Aluno

Complexidade Ciclomática: 8 de 8 - Nota: 20.00

Fonte: Marcos Pinto (2012)

Figura 7 – Exemplo Relatório de submissão Acode Sistema Web

**Java Tool** - é um ambiente de desenvolvimento, desenvolvido em Java, vocacionado para estudantes que frequentam as disciplinas de Introdução à Programação. O principal objectivo desta ferramenta é facilitar o ensino da linguagem de programação Java e deverá ser integrada num ambiente *Web* (i.e. *Moodle*), podendo ser utilizada como ferramenta de trabalho virtual. Esta ferramenta permite ao estudante analisar o algoritmo, seleccionar exercícios propostos pelo docente e produzir uma solução para submeter no sistema. Enquanto ambiente de trabalho integrado, o JavaTool apenas transmite para o estudante mensagens de erro na compilação e execução.

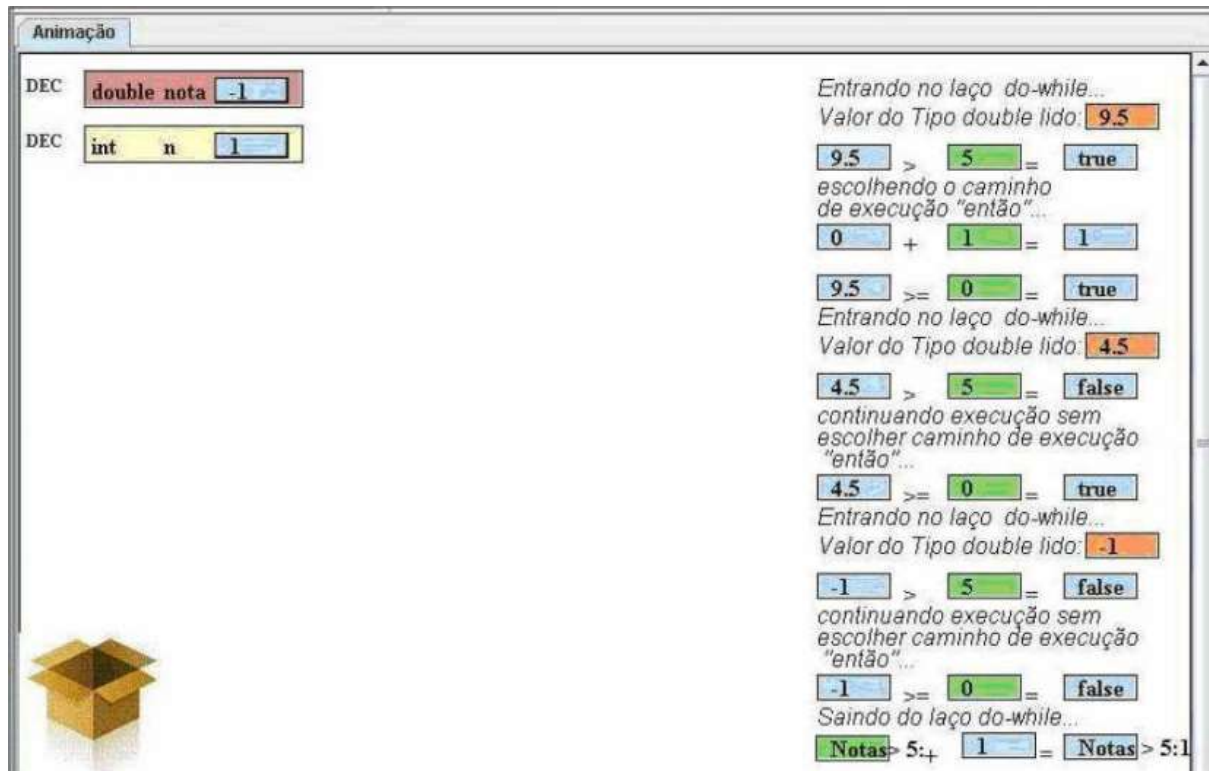


Figura 8— Exemplo de animação para aprendizagem Java Tool

**UVA Tool** - Programa inserido no contexto de programação competitiva que por vezes tem passos demorados e desnecessários, que podem influenciar os resultados e as classificações. Sendo este tipo de competição muitas vezes online e o software de avaliação ser inserido numa página web, Uva Tool procurou então facilitar e automatizar as seguintes tarefas:

- Copiar o código do ambiente de trabalho.
- Esperar que carregue a página da competição para realizar a submissão.
- Colar o código no local pretendido no site da competição
- Seleccionar a linguagem de programação
- Seleccionar o número/id da questão
- Clicar no botão de submissão
- Verificação do veredicto

## 6 Calendário

De forma a gerir a execução deste Trabalho Final de Curso, apresenta-se em seguida o planeamento das várias etapas e suas respectivas datas, necessárias para a conclusão deste projeto. Onde inicialmente irei configurar o *Plugin* existente no meu computador de trabalho, seguido da realização de um questionário e a avaliação dos seus resultados para o levantamento dos requisitos para os docentes. Após a angariação dos requisitos, avançamos para a etapa principal, a implementação das funcionalidades e por último, os testes a realizar à aplicação final. Poderá visualizar a tabela do Calendário em baixo.

**Tabela 1 - Calendário de tarefas**

1	Configuração da aplicação no meu computador de trabalho	24/11/2020	10/12/2020
2	Execução e avaliação do questionário para os docentes	30/11/2020	08/12/2020
3	Levantamento de requisitos com base no questionário	09/12/2020	16/12/2020
4	2º relatório	14/01/2021	22/01/2021
5	Implementação	01/02/2021	15/05/2021
6	3º relatório	15/04/2021	23/04/2021
7	Testes	16/05/2021	07/06/2021
8	Relatório final	08/06/2021	20/06/2021

## Bibliografia

- [1] Drop Project, <https://github.com/yashjahit-21705201/drop-project> , Set. 2020
- [2] Plugin Drop Project, <https://github.com/yashjahit-21705201/Plugin-Drop-Project> , Set. 2020
- [3] Raccode, <https://repositorio-aberto.up.pt/bitstream/10216/118773/2/312198.pdf> , Set. 2020
- [4] Mooshak, <https://mooshak.dcc.fc.up.pt/> , Set. 2020
- [5] Acode, <https://repositorio.iscte-iul.pt/bitstream/10071/8263/1/ACode%20Sistema%20Web%20para%20Avalia%C3%A7%C3%A3o%20Autom%C3%A1tica%20de%20Algoritmos%20Java.pdf> , Set. 2020
- [6] Java Tool, <http://www2.sbc.org.br/csbc2008/pdf/arq0111.pdf> , Set. 2020
- [7] UVA Tool, <https://codeforces.com/blog/entry/56280> , Set. 2020

## Anexo 1 – Questionário

Perguntas realizadas:

- Utilizaria o plugin do Drop Project no IntelliJ?
- Ajudaria no feedback ao aluno a possibilidade de deixar um comentário associado a uma submissão?
- Ao ter acesso aos alunos que já fizeram uma submissão, seria prático ter uma opção de fazer download da última submissão desse aluno, mesmo podendo fazer download dessa submissão quando abrir todas as submissões desse aluno e selecionar a última para download?
- Que opções gostaria de ter para ordenar/filtrar/pesquisar os assignments além destas (ID, Nome/Tag, Data (de \*\*/\*\*/\*\* a \*\*/\*\*/\*\*), ativo ou arquivado) ?
- Que funcionalidades gostaria que existissem neste Plugin?

### Resultados Questionário

Após a realização do questionário a um total de seis docentes, obteve-se os seguintes resultados:

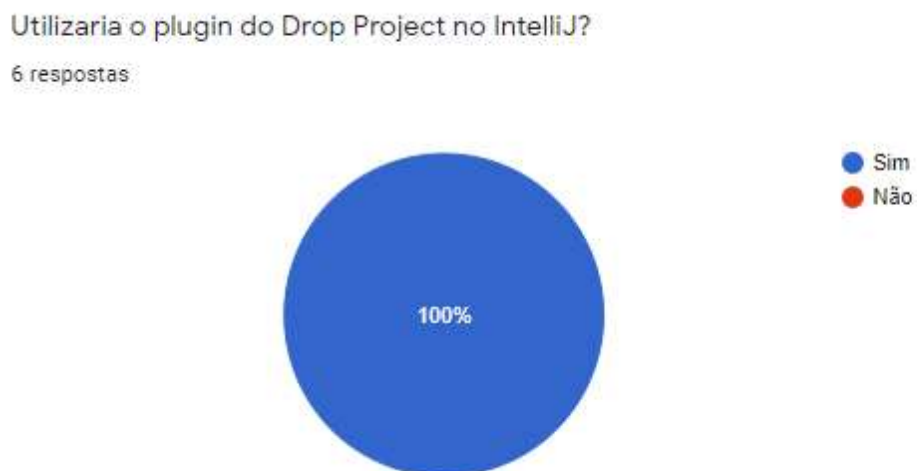
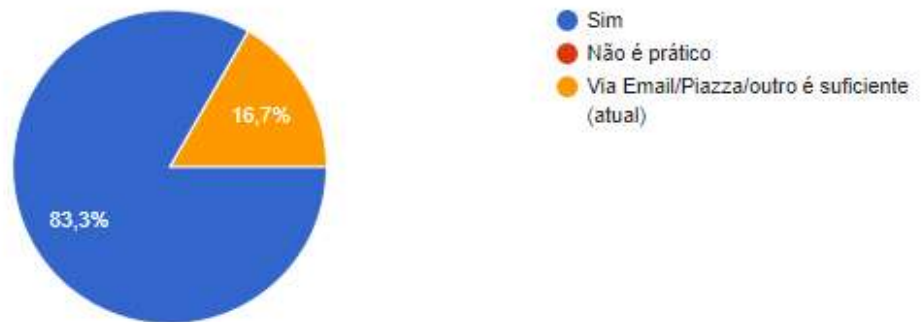


Figura 9- Resposta à primeira pergunta: “Utilizaria o plugin do DropProject no IntelliJ?”

Ajudaria no feedback ao aluno a possibilidade de deixar um comentário associado a uma submissão?

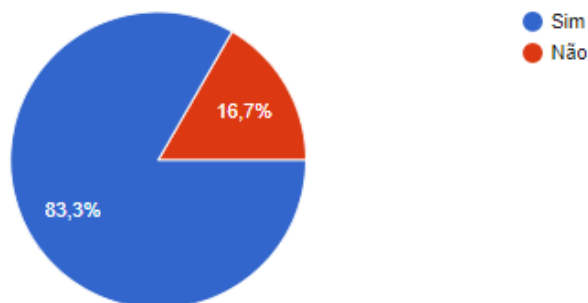
6 respostas



**Figura 10- Resposta à segunda pergunta: “Ajudaria no feedback ao aluno a possibilidade de deixar um comentário associado a uma submissão?”**

Ao ter acesso aos alunos que já fizeram uma submissão, seria prático ter uma opção de fazer download da ultima submissão desse aluno, mesmo podendo fazer download dessa submissão quando abrir todas as submissões desse aluno e selecionar a última para download?

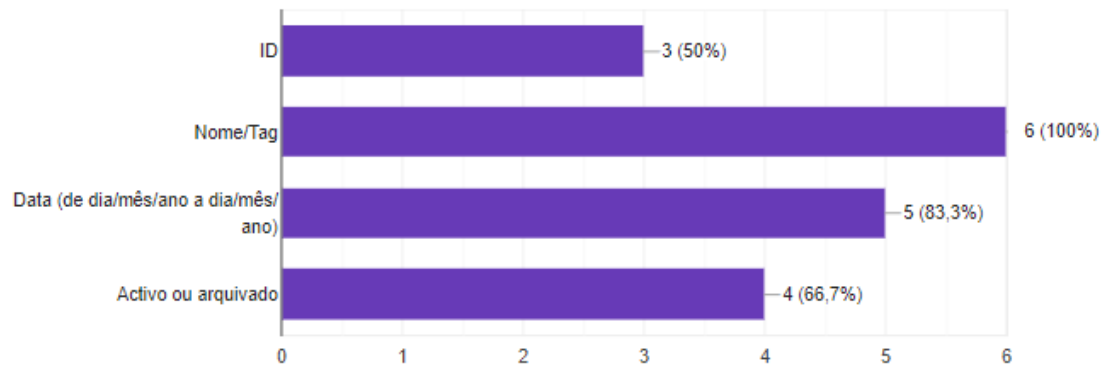
6 respostas



**Figura 11- Resposta à terceira pergunta: “Seria prático ter uma opção de fazer download da última submissão do aluno?”**

Que opções gostaria de ter para ordenar/filtrar/pesquisar os assignments.

6 respostas



**Figura 12 - Resposta à quarta pergunta: “Que opções gostaria de ter para ordenar/filtrar/pesquisar os assignments”**

Para além das opções apresentadas em cima, gostaria de ter outras opções de ordenação/filtragem/pesquisa? Qual ou quais?

4 respostas

parecem-me suficientes as opções

Pesquisar Assignments por nome de aluno que submeteu

- \* Gostaria que houvesse uma ordenação por omissão, se calhar por número de aluno
- \* Ou talvez que o sistema guardasse as minhas preferências, ou a última "coisa" que fiz
- \* Ordenar por grupo(?)

Ordenação dos assignments por data da última submissão (do mais recente para o mais antigo)

**Figura 13 - Resposta à quinta pergunta: “Para Além das opções apresentadas em cima, gostaria de outras opções de ordenação/filtragem/pesquisa? Qual ou quais?”**

## Que outras funcionalidades gostaria que existissem neste Plugin?

\* Ver o último relatório para cada Submissão (sem ter que correr o maven/java/projecto/o\_que\_for) \* Talvez escolher as colunas que queria ver nas várias listagens (e que o IntelliJ guardasse as minhas preferências) \* Poder exportar para excel (com a mesma capacidade anterior - escolher o que exportar) \* Na exportação para excel, ter links directos para a submissão a que corresponde e ao código git (se for esse o caso) \* Ter funcionalidade para fazer download de todas as submissões marcadas como finais \* Ter funcionalidade para poder ver código das submissões (por exemplo, serem "módulos"/"projectos" no IntelliJ - devia ser possível navegar no código, correr, etc) \* Possivelmente podia querer, como professor, actualizar o código do Assignment sem ser preciso ir à página web do DP (possivelmente no futuro, vou querer todas as funcionalidades que há na vertente web do DP)

A mais importante de todas é conseguir "abrir" a submissão de um aluno com um click (equivalente a fazer download, descomprimir e importar como maven). "Nice to have" seria a possibilidade de refrescar uma submissão prévia de um aluno para a última versão (equivalente ao git pull). O meu objectivo principal é conseguir fazer debug das submissões dos alunos o IntelliJ. Também gostava de conseguir criar um assignment a partir de um projecto do IntelliJ que: (1) criasse um repo no github e colocasse lá o projeto e (2) criasse o assignment no DP automaticamente associado ao repo.

Saber, para um determinado aluno, em que assignments fez pelo menos uma submissão. Poder escolher 2 ou 3 submissões (do mesmo assignment) e abrir as mesmas no IntelliJ ao mesmo tempo (2 ou 3 janelas novas).

parecem-me suficientes

**Figura 14 - Resposta à sexta pergunta: "Que outras funcionalidades gostaria que existissem neste Plugin"**



## **Glossário**

LEI	Licenciatura em Engenharia Informática
DP	Drop Project
IDE	Integrated Development Environment
TFC	Trabalho Final de Curso