



UNIVERSIDADE
LUSÓFONA

Plugin Drop Project

Trabalho Final de curso

Relatório Final

Aluno: Bernardo Van Laake Baltazar, 21704025

Orientador: Bruno Cipriano

Coorientador: Pedro Alves

Trabalho Final de Curso | Licenciatura em Engenharia Informática | 30/06/2023

www.ulusofona.pt

Direitos de cópia

Plugin Drop Project, Copyright de Bernardo Van Laake Baltazar, ULHT.

A Escola de Comunicação, Arquitetura, 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 objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Resumo

As ferramentas de avaliação automatizadas melhoram a eficiência e consistência no processo de avaliação das unidades curriculares. O Drop Project é uma aplicação deste conceito, desenvolvido pelos docentes da Universidade Lusófona Pedro Alves e Bruno Cipriano. Esta ferramenta de avaliação automática veio a resolver muitas adversidades do lado da equipa docente, tornando o processo de avaliação mais eficaz e consistente. Mas por outro lado criou adversidades, por parte dos alunos, relativamente à execução de procedimentos necessários para a utilização desta ferramenta, por vezes influenciando o desempenho dos alunos. O projeto ao qual se dará continuidade, será no âmbito de solucionar este problema de maneira que seja uma ferramenta necessária e utilizada no futuro.

Abstract

Automated assessment tools enable the efficiency and consistency in the evaluation process of curricular units. The Drop Project is an application of this concept, developed by Pedro Alves and Bruno Cipriano, professors at the Lusófona University. This automated assessment tool came to solve many adversities of the teaching team, making the assessment process more effective and consistent. But on the other hand, it created other adversities, on the part of the students, regarding the execution of necessary procedures for the use of this tool, sometimes influencing the performance of the students. The project that will be continued will be in the scope of solving this problem so that it becomes a necessary tool that will be used in the future.

Índice

Resumo.....	iii
Abstract	iv
Índice	v
Lista de Figuras	vii
1 Identificação do Problema	1
1.1 Enquadramento.....	1
1.1.1 <i>Automated assesment tool</i>	1
1.1.2 Drop Project	1
1.2 Plugin	2
1.3 Projeto antecessor	2
1.3.1 Plugin 2020.....	3
1.4 Fundamentação.....	4
1.4.1 Novo Drop Project: <i>Application programming interface</i>	4
1.4.2 Token	4
1.4.3 Autenticação com Token.....	4
2 Viabilidade e Pertinência.....	5
3 Benchmarking.....	8
3.1 Raccode	8
3.2 Repl.it	8
3.3 Gradescope	8
3.4 Classroom for GitHub	8
3.5 CodeChef	9
4 Engenharia.....	10
4.1 Levantamento e análise dos Requisitos	10
4.1.1 Requisitos Principais/iniciais	10
4.1.2 Requisitos Pós-testagem	12
4.2 Modelos relevantes.....	14
5 Solução Desenvolvida.....	16
5.1 Introdução.....	16
5.2 Tecnologias e Ferramentas Utilizadas.....	16

5.3	Implementação	17
5.4	Abrangência.....	18
6	Método e Planeamento	19
7	Resultados	20
7.1	Enquadramento.....	20
7.2	Resultados e Sugestões	21
7.2.1	Versões de lançamento:.....	21
7.2.2	Alterações e correções extra	23
7.3	Inquérito Final e Adesão	23
7.3.1	Adesão do plugin.....	27
8	Conclusão e Trabalhos Futuro	30
	Bibliografia.....	31
	Anexos	32
	Glossário	33

Lista de Figuras

Figura 1 - Menu 1º Protótipo	3
Figura 2 - Login 1º Protótipo	3
Figura 3 - Token gerada pelo Drop Project	4
Figura 4 - Modelo conceptual da solução	15
Figura 5 - Diagrama de Sequência de uma submissão do plugin.	15
Figura 6 - Diagrama de Sequência de obtenção de detalhes de uma tarefa.	15
Figura 7 - Timeline do planeamento da avaliação anterior	19
Figura 8 - Planeamento e execução do projeto ao longo do ano	19
Figura 9 - Submissões totais (0=UPLOAD; 2=API)	28
Figura 10 - Submissões projeto prático parte 1 e parte 2 (linhas escuras = UPLOAP; linhas claras = API)	28

1 Identificação do Problema

Esta secção é dedicada à identificação do problema que este trabalho final de curso pretende resolver. Esta divide-se em três subsecções, sendo elas:

- Enquadramento: apresentação da ferramenta de software com a qual a solução irá interagir.
- Plugin: reconhecimento das adversidades que a solução tenta resolver.
- Projeto antecessor: descrição do primeiro protótipo, ao qual se deu continuidade.

1.1 Enquadramento

Nos cursos associados às ciências da computação, por norma, as suas unidades curriculares envolvem o desenvolvimento de projetos para avaliação, que muitas vezes são programas computacionais. Estes projetos solucionam problemas expostos pelas unidades curriculares, e que para a sua solução, por vezes, necessitam um conjunto grande de instruções. Continuamente, estas instruções terão de ser testadas e avaliadas pela equipa docente da unidade curricular em questão, para que o aluno tenha uma classificação relativa ao projeto desenvolvido. Este processo de classificação, feito pelos docentes, tende-se a ficar muito exaustivo e demorado devido à amplitude dos projetos de cada aluno, o que põe em causa a sua consistência e eficiência.

Para resolver este problema, foram desenvolvidas ferramentas de avaliação automática (*ATT's— Automatic assesment tool*).

1.1.1 Automated assesment tool

Automated assesment tools (AAT's) são softwares com intuito educacional destinados à avaliação automatizada de projetos computacionais, desenvolvidos pelos estudantes. As AAT's começaram a ser desenvolvidas em meados de 1960 [AATs05], sendo a sua utilização mais comum nos cursos de informática [Leht05]. Esta ferramenta possibilita aos estudantes a verificação da satisfação dos requerimentos do trabalho/problema em questão de forma completamente autónoma, ou seja, sem assistência do docente. O Feedback rápido e a identificação de erros, são os principais resultados de um AAT, que consequentemente promove o envolvimento e a motivação, relativa ao curso, do aluno [Leht05]. Através dos algoritmos de avaliação, os professores poupam bastante tempo no seu processo de avaliação. Podendo-se concentrar mais no ensino e no envolvimento do aluno na unidade curricular.

1.1.2 Drop Project

O Drop Project é uma implementação de uma AAT, em que cada aluno submete projetos e recebe feedback da qualidade dos mesmos. Este software está em produção desde 2018, na licenciatura de engenharia informática da universidade Lusófona. O seu desenvolvimento foi motivado por vários fatores, mas principalmente pelo acréscimo da dificuldade e do tempo demorado no processo de avaliação, causado pelo aumento gradual de estudantes inscritos na licenciatura. O Drop Project tornou mais eficaz e consistente o processo de avaliação, viabilizando o aumento da rigorosidade dos projetos das unidades curriculares que utilizam esta ferramenta.

[Drop22] O seu feedback é baseado em diferentes tipos de validação:

- Unit Tests, validam requerimentos funcionais
- Qualidade de código, verifica o grau de conformidade da escrita do código
- Tempo de execução e memória utilizada, relevante para unidades curriculares que avaliem a eficiência.

O Drop Project provém da *framework* Maven, ferramenta de automação de compilação mais popular para linguagens executadas em Java Virtual Machine (JVM) [Apac21]. O software tem vindo a ter melhorias, em diversos aspetos, ao longo dos anos. Isto porque, fundamentalmente, os seus princípios e tecnologias são de integração contínua.

1.2 Plugin

Apesar do grande impacto que o Drop Project teve nas unidades curriculares que o integraram, ao longo dos anos, a equipa docente apercebeu-se que para a utilização da ferramenta eram necessárias certas ações do utilizador, ou seja, do aluno. Ações essas, que causavam dificuldade e geravam níveis de estresse, principalmente em momentos críticos avaliativos, e são indispensáveis para a submissão do trabalho do aluno. Para as fazer o aluno teria de abdicar do seu tempo e, por vezes, quebrar a sua linha de pensamento. Sendo estas:

- Criação do ficheiro AUTHORS.txt, que identificaria o/os autor/es do trabalho submetido;
- Criação de um ficheiro ZIP, onde se une o trabalho que pretende submeter, com o ficheiro AUTHORS.txt;
- Entre outras ações fundamentais de uma aplicação web, como aceder ao endereço da tarefa atribuída, selecionar e enviar para o website o ficheiro pretendido e reautenticação, no caso de *timeout*

No fundo, são ações relativamente triviais de se fazer, mas quando o aluno está numa situação crítica, como um teste, em que o tempo é limitado, estes procedimentos podem influenciar bastante o desempenho do aluno, e/ou até mesmo prejudicar a nota da avaliação do mesmo.

Com isto, surgiu a ideia, dada pelo professor Pedro Alves, da criação de um plugin que automatizasse estes procedimentos. Fundamentalmente, seria integrar este plugin ao ambiente de desenvolvimento integrado (IDE), onde o aluno escreve o seu código pois desta forma, o fluxo de trabalho do aluno é mais fluido.

Foi, então, escolhido o IDE IntelliJ Idea da empresa JetBrains, dado que é a ferramenta utilizada/recomendada nas unidades curriculares do professor Pedro.

1.3 Projeto antecessor

Em 2020, o ex-aluno Yash Jahit da licenciatura em engenharia informática, criou o primeiro protótipo de plugin no IntelliJ Idea.

Apesar da meta alcançada, este plugin não teve estrutura e condições para entrar em produção. Dado que uma das maiores razões foi o facto de não ter sido testado por alunos, não tendo chegado a uma finalidade concreta relativamente à necessidade deste plugin.

Atualmente, o plugin já não é compatível com a versão mais recente do Drop Project uma vez que, ao longo destes dois anos, o Drop Project foi evoluindo e tornando-se mais robusto em vários aspetos, como por exemplo: autenticação por parte de entidades externas deixou de ser feito através da palavra-chave do DEISI referente ao aluno, mas sim por uma *token* gerada pelo próprio Drop Project, atribuída ao utilizador.

1.3.1 Plugin 2020

Desenhado maioritariamente em Kotlin, utilizando a ferramenta de compilação Gradle conjunta com o IntelliJ Platform SDK, para um ambiente versátil e abrangente como o IntelliJ Idea. Tanto a linguagem quanto o ambiente de desenvolvimento integrado (IDE) são altamente robustos e altamente compatíveis, proporcionando uma ampla gama de possibilidades de implementação, desde a compilação conjunta de arquivos Kotlin e Java até a utilização do Kotlin DSL [Kode19]. Este protótipo dispõe das seguintes funcionalidades:

- Login
- Listar Tarefas
- Submeter Tarefa

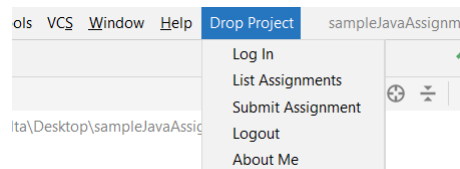


Figura 1 - Menu 1º Protótipo

Login

Responsável pela autenticação do utilizador. É pedido o nome, número de aluno e a respetiva palavra-passe de acesso ao DEISI. Esta autenticação é feita através de um pedido HTTP com autenticação básica, e ao ser bem-sucedida é criado o ficheiro AUTHORS.txt já concebido para o Drop Project

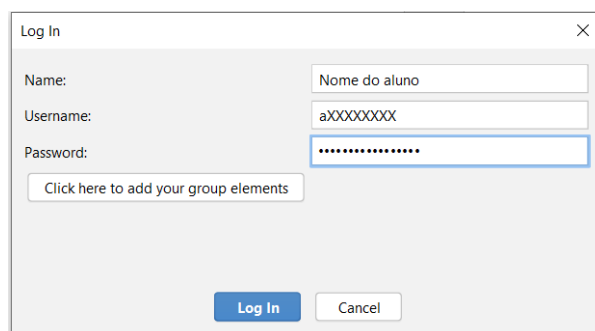


Figura 2 - Login 1º Protótipo

Listar Tarefas

Responsável pela listagem das tarefas atribuídas ao aluno. Nesta página, é também possível:

- Ver a lista de submissões realizadas pelo aluno, referente à tarefa;
- Ver o enunciado da tarefa;
- Selecionar a tarefa para possível eventual submissão.

Submeter tarefas

Responsável pela submissão da tarefa previamente selecionada pelo utilizador. É nesta funcionalidade que o ficheiro ZIP é criado, com o ficheiro do projeto e o ficheiro AUTHORS nele contidos. De seguida é enviado para o Drop Project, através de um pedido HTTP POST.

1.4 Fundamentação

Dado que este primeiro protótipo já não funciona com a versão atual do Drop Project, é necessário modificá-lo e adaptá-lo. Modificações e melhorias essas fundamentaram este trabalho.

1.4.1 Novo Drop Project: *Application programming interface*

No presente, o Drop Project apresenta uma API (application programming interface) redesenhada, com alterações notáveis: principalmente nos endereços de pedidos ao serviço (*endpoints*), nas entidades e estruturas de resposta aos pedidos, e na autenticação, sendo esta feita por tokens.

1.4.2 Token

Com a existência da token garante-se outro nível de segurança, pois, desta forma, o plugin não armazena no disco a palavra-chave do aluno que garante o acesso a todos os serviços da Universidade Lusófona.

A token é gerada pelo Drop Project e tem uma validade que após a sua expiração, o acesso ao Drop Project, através da mesma, não é possível.

Personal access tokens

Tokens you have generated that can be used to access the Drop Project API

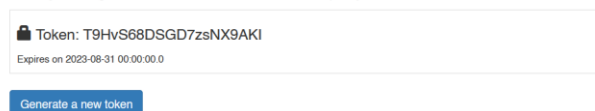


Figura 3 - Token gerada pelo Drop Project

1.4.3 Autenticação com Token

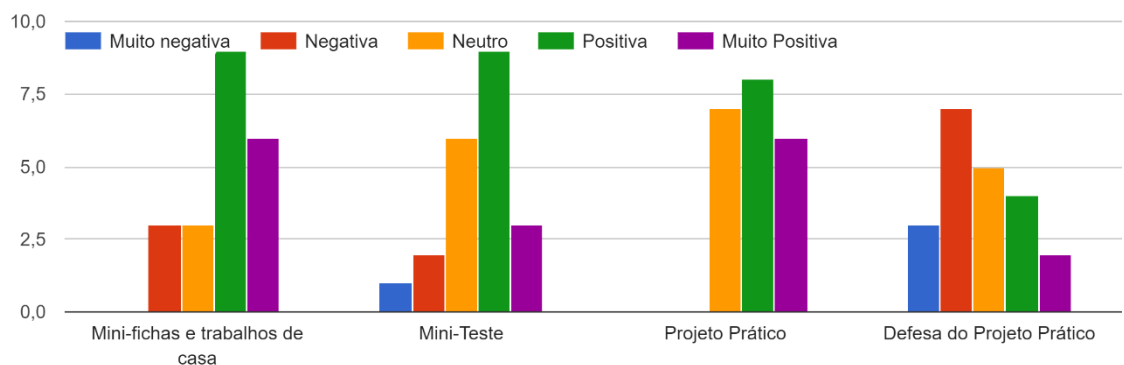
Esta autenticação é de acesso básico, e é requerida em todos os pedidos à API. E pretende-se tornar o plugin compatível com este sistema.

2 Viabilidade e Pertinência

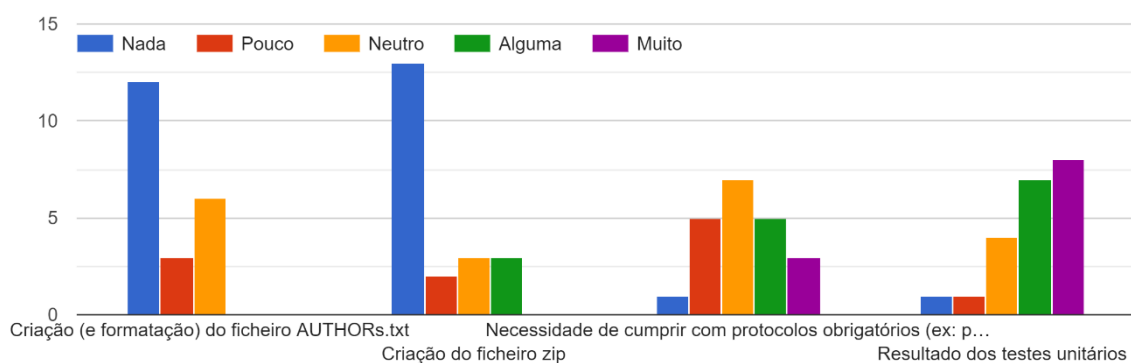
O Drop Project está disponível para os alunos desde 2018, como dito anteriormente (secção 1.1.2). E desde então, tem sido a ferramenta principal no processo de avaliação das unidades curriculares de fundamentos de programação, algoritmia e estrutura de dados e linguagens de programação.

Esta secção tem como objetivo demonstrar a viabilidade e pertinência do projeto, fundamentando-se num inquérito realizado numa fase inicial do projeto. Os resultados deste inquérito revelaram uma tendência positiva em relação à pertinência do plugin, ao abordar os problemas que o projeto tenta resolver.

Como classificaria a sua experiência do Drop Project nos seguintes casos:

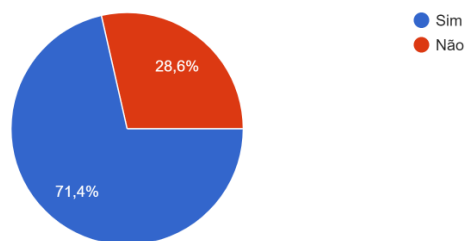


Considere o uso do Drop Project para avaliações presenciais (mini-testes e defesas de projecto). Nesse âmbito, considera que o proces...ansiedade e/ou nervosismo, nos seguintes casos ?



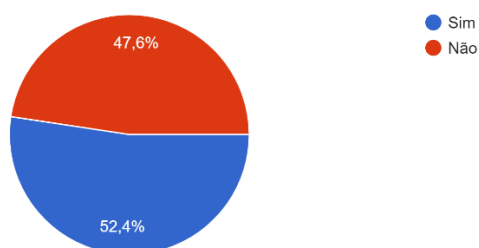
Diria que o seu desempenho durante momentos avaliativos presenciais (ex: mini-teste) é influenciado pelo processo de submissão do Drop Project?

21 respostas

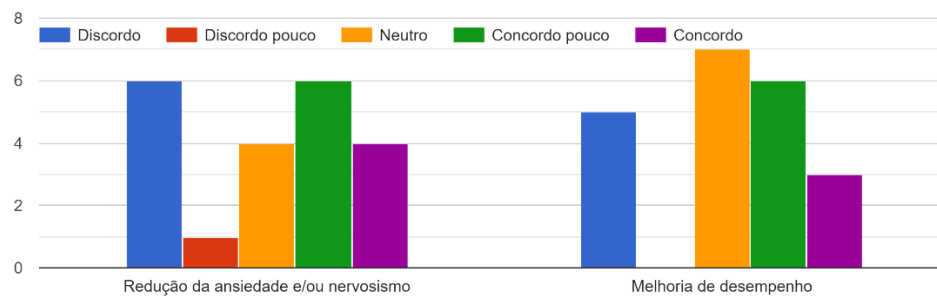


Diria que o seu desempenho durante momentos avaliativos não presenciais (ex: projeto prático final) é influenciado pelo processo de submissão do Drop Project

21 respostas

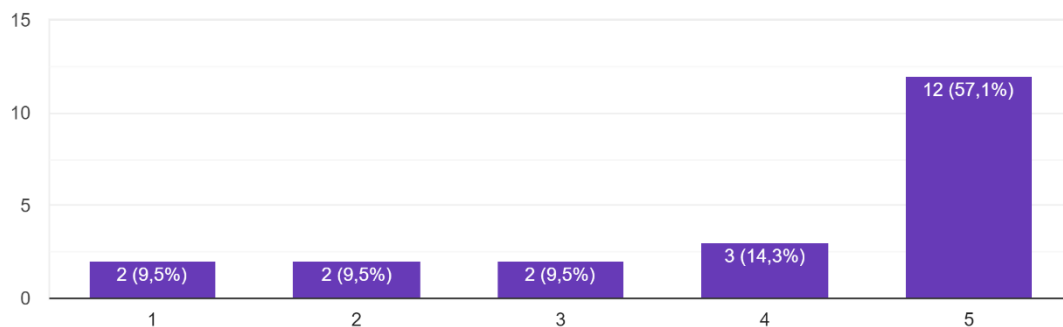


Concorda que se o processo de submissão de projetos, isto é, criação do AUTHORS.txt e criação do ficheiro zip (contendo a pasta /src e o AUTHOR...tomático, contribuiria para a os seguintes casos?



Se existisse uma forma de fazer submissões para o Drop Project diretamente a partir do IntelliJ, o quão provável seria que passasse a usar essa forma?

21 respostas



Embora os resultados obtidos no inquérito tenham apresentado diversidade, é possível identificar uma tendência geral de que a utilização de uma ferramenta integrada no ambiente de desenvolvimento (IDE) é algo vantajoso, desde que seja bem aproveitada. Isso deve-se ao facto desses alunos estarem acostumados com a metodologia "clássica" do Drop Project, e, além disso, não haviam experimentado nenhuma outra ferramenta semelhante anteriormente.

É importante destacar que o grupo de alunos que respondeu ao inquérito já tinha uma certa familiaridade com a metodologia tradicional e, portanto, pode ter mostrado alguma resistência inicial à mudança. No entanto, a tendência geral é de que a utilização de uma ferramenta integrada no IDE fosse vantajosa, pois permite aos alunos aproveitar as funcionalidades do plugin para melhorar a eficiência e a qualidade do seu trabalho.

Além disso, ao longo do ano letivo, alunos utilizaram ativamente esta solução, o que resultou numa maior familiaridade com a mesma. Essa evolução foi evidenciada pelos resultados obtidos num inquérito final realizado. Os detalhes e análises desse inquérito podem ser encontrados na secção de Resultados. [7].

3 Benchmarking

Embora as AAT's estejam há bastante tempo no mercado, e existirem bastantes ferramentas com este intuito, não existem muitas implementações expansíveis para um ambiente de desenvolvimento integrado. No entanto, as existentes são muito úteis e requisitadas pelas universidades.

3.1 Raccode

Integração de uma AAT, Mooshak, no ambiente de desenvolvimento Eclipse. Este plugin, desenvolvido por André Silva, José Paulo Leal e José Carlos Paiva [AnJJ12], tem como funcionalidades:

- Visualização de enunciados;
- Submissão de trabalhos;
- Levantamento de feedback ao aluno;
- Disponibilização de um sistema de ranks
- Área social onde é possível expor perguntas e repostas

O plugin Raccode envolve este mercado, que é considerado grande devido à pouca oferta contendo, comparativamente, funcionalidades similares, abordando o problema exposto neste trabalho.

3.2 Repl.it

O Repl.it é uma plataforma de desenvolvimento em nuvem que fornece um IDE para uma variedade de linguagens de programação. A plataforma também tem uma vertente de tarefas, que permite que os professores criem tarefas e classifiquem automaticamente as submissões dos alunos. A plataforma tem um plugin que se integra com IDEs populares e permite que os utilizadores enviem código para uma certa tarefa, criada pelo professor, para avaliação. Para além disso, a plataforma inclui uma vertente de colaboração, que habilita os utilizadores trabalharem em conjunto num mesmo projeto (programação a pares)

3.3 Gradescope

O Gradescope é uma plataforma web desenhada pra simplificar o processo de avaliação para professores. Permite aos professores criar e avaliar tarefas, especialmente as que requerem muita escrita. Para além de trazer as funcionalidades essenciais de uma ferramenta de avaliação automatizada, a plataforma contém uma componente de deteção de plágio, evitando assim cópias. Esta plataforma também fornece um plug-in para vários IDEs populares, incluindo Visual Studio Code, IntelliJ IDEA e Eclipse. Este plugin permite aos alunos a submissão de projetos diretamente do IDE tornando o processo de envio mais conveniente.

3.4 Classroom for GitHub

A vertente *classroom* do GitHub habilita aos professores criarem a sua turma virtual e disponibilizar as suas tarefas para essa turma. Esta ferramenta, fazendo parte do GitHub, aproveita o seu mecanismo de envio e gerenciamento de versões, utilizando o plugin do GitHub

disponível para os IDE mais populares. Assim, o aluno consegue submeter os seus projetos às destinadas tarefas.

3.5 CodeChef

Estabelecido em 2009, O CodeChef é uma plataforma online e uma comunidade popular para programadores, especialmente para os interessados em programação competitiva e em aprimorar as suas habilidades de programação. Permite os participantes resolverem problemas algorítmicos em várias áreas e linguagens de programação, através de concursos competitivos. À parte das competições, esta plataforma também oferece uma coleção de puzzles lógicos e tutoriais, de maneira a enfatizar a aprendizagem. Desta forma, esta ferramenta estabelece um ambiente sinérgico que engloba tanto a avaliação automática dos desafios propostos quanto a disponibilização de um IDE para permitir que o utilizador possa programar. No entanto, esta ferramenta oferece um conjunto limitado de funcionalidades em comparação com os IDEs (como o IntelliJ IDEA), incluindo um editor de código com um fraco suporte de conclusão de código.

4 Engenharia

O primeiro passo na engenharia de uma ferramenta de avaliação automatizada é a identificação dos requisitos. Isso inclui entender a tarefa a ser avaliada e os critérios de avaliação, bem como os requisitos não funcionais. Dado que se trata de um projeto já começado, foi feita uma análise inicial do estado do projeto para avaliá-lo e identificar as suas fraquezas, de modo a reunir as ideias e os objetivos principais a alcançar.

Após avaliação foram delineados os objetivos macro principais a atingir, do qual resultaria um resultado concreto testável. Sendo elas:

- Compatibilidade com as versões mais recentes da plataforma IntelliJ:
 - Adaptação do código dos ficheiros de configuração do plugin, de modo a adaptá-lo às novas versões do IDE.
- Compatibilidade com a versão mais recente do Drop Project:
 - Novos *endpoints* da API;
 - Adaptar os novos formatos das respostas vindas do serviço;
 - Novo tipo de autenticação.
- Melhoramento da integração do plugin no ambiente IDE, nomeadamente em termos de interface gráfica;
- Habilitação para testes com alunos.

Estava definido, à priori, que se iriam antecipar os testes de usabilidade (em aula com alunos reais), e que esses testes iriam levar a uma segunda fase de especificação de requisitos, na qual seriam definidos requisitos mais concretos do projeto.

4.1 Levantamento e análise dos Requisitos

Este processo é contínuo e incremental, de tal maneira que ao ter concluído uma primeira versão, que dava resposta aos objetivos macro identificados no ponto anterior. Essa versão foi testada por alunos e o aparecimento de novos requisitos surgiu com o intuito de combater novas adversidades identificadas na testagem desta versão do produto. Dito isto, são apresentadas duas tabelas: Requisitos Principais/iniciais e requisitos pós-testagem, em que, respetivamente são mostrados os requisitos bem definidos, propostos inicialmente, que continuarão a ter o mesmo impacto para o futuro do projeto, podendo-se tratar por “requisitos base”; e os requisitos que surgiram após os testes com os alunos da unidade curricular Linguagem de Programação II, que vieram a corrigir erros e resolver necessidades identificadas. Para que o plugin funcionasse de forma esperada teriam de ser feitas mudanças, não só no próprio plugin, como no serviço do Drop Project, justificando então a existência da coluna “Âmbito”.

4.1.1 Requisitos Principais/iniciais

- **Requisitos funcionais**

ID	Título	Descrição	Âmbito
----	--------	-----------	--------

F001	Autenticação de sessão de utilizador	O sistema deve fornecer uma opção de início de sessão para armazenar as credenciais do utilizador, a fim de autenticá-lo automaticamente em futuras sessões.	Plugin
F002	Manutenção de Credenciais	O sistema deve implementar uma funcionalidade de armazenamento de credenciais para o utilizador, para que o utilizador não precise de inseri-las novamente a cada interação com o Drop Project.	Plugin
F003	Listagem de Tarefas	O sistema deve exibir uma lista/tabela das tarefas atribuídas ao utilizador, após o início de sessão, permitindo que o utilizador possa selecionar e futuramente submeter numa tarefa específica.	Plugin
F004	Instruções da Tarefa	O utilizador deve ser capaz de consultar as instruções da tarefa pretendida, de modo a não ter que aceder ao website.	Plugin
F005	Submissão do trabalho	A funcionalidade de submissão de projetos deve ser disponibilizada através de um botão simples, facilitando o processo de submissão para o utilizador e evitando a necessidade de criação e envio de ficheiros zip através do website.	Plugin
F006	Acesso à avaliação de submissões	A ferramenta deve permitir que o utilizador consulte a avaliação do seu trabalho submetido, sempre que disponível, diretamente na plataforma.	Plugin
F007	Autenticação básica para acesso à API	A ferramenta deve suportar a autenticação básica para aceder à API, incluindo o envio de nome de utilizador e <i>token</i> respetiva no cabeçalho de cada pedido REST.	Plugin
F008	Acesso a instruções de tarefas via plugin	A API deve disponibilizar as instruções de uma tarefa em HTML, através de uma requisição para o serviço, e deve ser acessível através do plugin.	Drop Project
F009	"Controle de acesso às tarefas	O aluno não deve aceder às tarefas que estão invisíveis ou que não lhe estão atribuídas, a menos que tenha sido dada autorização pelo administrador ou professor.	Plugin e Drop Project
F010	Integração de componentes do plugin no IDE	Os componentes do plugin devem ser facilmente acessíveis através da interface do utilizador do IDE, como por exemplo, através de menus ou barras de ferramentas.	Plugin

F011	Filtro de tarefas inativas	A API deve filtrar as tarefas inativas ao devolver a lista de tarefas atribuídas ao utilizador autenticado.	Drop Project
-------------	----------------------------	---	--------------

- **Requisitos não-funcionais**

ID	Título	Descrição	Âmbito
NF001	Segurança de Credenciais do Utilizador	A segurança das credenciais do utilizador deve ser garantida através da cifragem dos dados, para evitar exposição de informação sensível.	Plugin
NF002	Garantia de estabilidade do plugin no IDE	O IDE não deve lançar runtime exceptions (p.e. “null pointer exception”) relacionadas ao plugin.	Plugin
NF003	Compatibilidade do plugin com o IDE	O plugin requer uma versão do IDE (IntelliJ IDEA) superior a 2021.3.3 para funcionar corretamente.O plugin tem de ser compatível com a última versão do IntelliJ (2023)	Plugin
NF004	Limite de requisições	O sistema deve ter um limite máximo de requisições simultâneas para evitar sobrecarga no serviço do Drop Project.	Plugin

4.1.2 Requisitos Pós-testagem

De seguida, seguem-se os requisitos que foram identificados após os testes realizados em aula com alunos. Requisitos esses que provieram dos alunos e professores, justificando a adição da coluna “Origem” na tabela seguinte.

- **Requisitos funcionais**

ID	Título	Descrição	Âmbito	Origem
F011	Lidar com o cool-off	O plugin deve tratar a resposta do serviço quando o utilizador se encontra em período de “cool-off” de maneira adequada, incluindo a exibição de um aviso ao utilizador	Plugin	Professores
F012	Perceber a tarefa ativa	O plugin deve ser capaz de identificar e destacar a tarefa selecionada na lista de tarefas, exibida no IDE, permitindo ao utilizador facilmente ver e gerenciar a tarefa selecionada	Plugin	Alunos

F013	Facilitar o acesso ao relatório mais recente	O plugin deve prover um mecanismo integrado na interface do IDE para que o utilizador possa visualizar o relatório mais recente. Isso deve incluir, mas não se limitar a, a possibilidade de aceder ao relatório mais recente diretamente através da interface principal do IDE, como por exemplo, por meio de um ícone ou botão específico. Além disso, o sistema deve ser capaz de armazenar e manter a disponibilidade dos relatórios anteriores, para que o utilizador possa ter acesso, quando necessário.	Plugin	Professores
F014	Atualização da lista de tarefas	O plugin deve incluir uma funcionalidade que permite ao utilizador atualizar a lista de tarefas de forma rápida e eficiente. Essa funcionalidade deve ser implementada como um botão específico, localizado de forma intuitiva e de fácil acesso, que ao ser pressionado, atualiza a lista de tarefas automaticamente. Além disso, o sistema deve incluir uma indicação visual, como por exemplo uma notificação, para informar ao utilizador que a lista de tarefas foi atualizada com sucesso	Plugin	Professores
F015	Adição manual de uma tarefa	O plugin deve prover uma funcionalidade para que o utilizador possa adicionar na lista de tarefas uma tarefa específica através do seu identificador único (ID). Essa funcionalidade deve incluir, pelo menos, a possibilidade de inserir o ID da tarefa desejada através da interface do IDE, e a API do Drop Project deve retornar os detalhes da tarefa correspondente [F016]. De referir que o plugin deve ser capaz de reagir adequadamente à resposta fornecida pela API, sendo o ID válido ou não.	Plugin	Professores
F016	Obter uma tarefa através do seu ID	A API deve fornecer uma funcionalidade para devolver os detalhes de uma tarefa específica (incluindo informações como instruções, data de vencimento, linguagem de programação, etc.)	Drop Project	Programador

		através de uma requisição para o serviço com o ID da tarefa. A funcionalidade deve ser acessível através de pedidos REST		
F017	Auto-save ao submeter	O plugin deve incluir uma funcionalidade de <i>auto-save</i> automático para os ficheiros que foram alterados pelo utilizador, antes de serem submetidos	Plugin	Alunos
F018	Armazenamento de credenciais	O plugin deve ter a capacidade de armazenar as credenciais de login do utilizador no disco local de forma segura e criptografada, permitindo que o utilizador não precise inserir manualmente suas credenciais toda vez que abrir o IDE	Plugin	Alunos
F019	Formatação dos resultados da avaliação	O plugin deve fornecer uma formatação semelhante à dos resultados da avaliação de uma submissão no website (Drop Project), na apresentação dos detalhes da avaliação submetida pelo utilizador através do plugin, com o objetivo de familiarizar o utilizador e, consequentemente, facilitar a sua leitura.	Plugin	Alunos
F020	Segurança das instruções de uma tarefa não atribuída	O serviço deve garantir a proteção contra possíveis acessos não autorizados a tarefas. Quando o utilizador introduz o ID de uma tarefa que não lhe está diretamente atribuído, o acesso às instruções fica bloqueado. Desta forma, garante-se que apenas os utilizadores com acesso autorizado possam visualizar as instruções da tarefa em questão. Impossibilitando assim, a revelação de informação ainda não anunciada.	Drop Project	Professores

4.2 Modelos relevantes

Como projeto de continuidade, foi adotado o mesmo modelo conceitual do projeto anterior, garantindo a consistência e coerência dos aspetos funcionais e estéticos, bem como a compatibilidade com os objetivos estabelecidos. Desta forma, garante-se a continuidade e a evolução do projeto original, assegurando a sua eficácia e eficiência na consecução dos objetivos propostos. Assim sendo, para proporcionar uma compreensão precisa e detalhada sobre a estruturação da engenharia deste plugin, é apresentado o seguinte modelo conceitual:

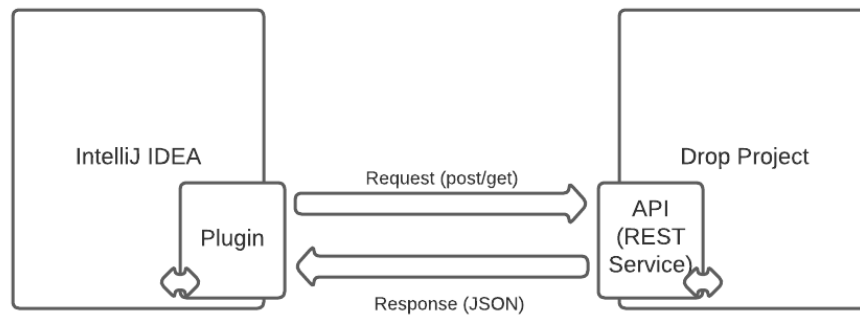


Figura 4 - Modelo conceptual da solução

De maneira a entender o funcionamento da solução, são demonstrados os processos principais da solução, através de diagramas de sequência.

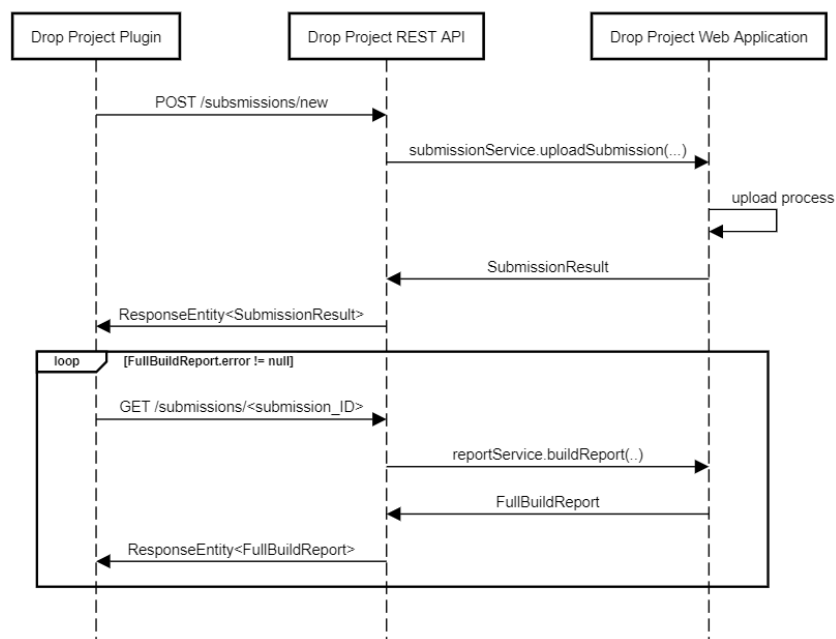


Figura 5 - Diagrama de Sequência de uma submissão do plugin.

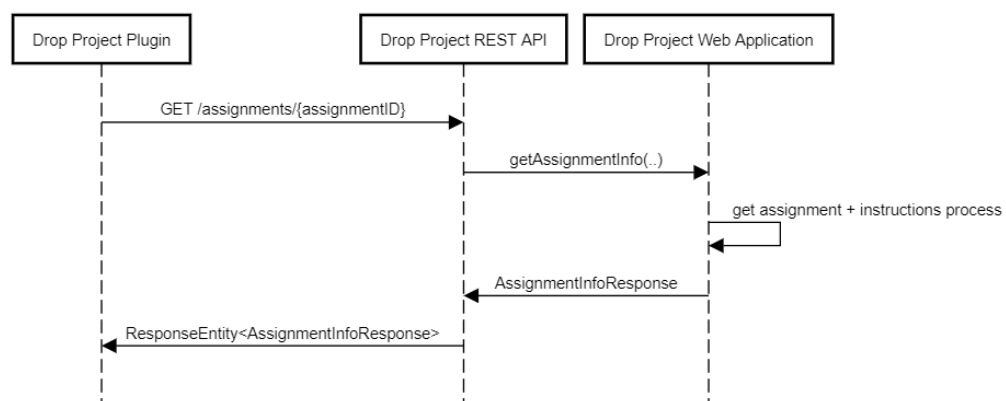


Figura 6 - Diagrama de Sequência de obtenção de detalhes de uma tarefa.

5 Solução Desenvolvida

Esta secção dedica-se à apresentação detalhada da solução desenvolvida, que visa enfrentar os desafios referidos na identificação do problema. O foco principal desta solução é proporcionar aos alunos um plugin que agregue valor e, posteriormente, seja adotado nas unidades curriculares onde a sua utilização seja relevante. Esta apresentação é composta pela sua introdução/descrição, pelas tecnologias e ferramentas empregadas, bem como pela sua implementação e abrangência.

5.1 Introdução

O novo plugin do Drop Project visa fornecer soluções de eficiência e usabilidade face à sua versão anterior. O plugin é desenvolvido para funcionar no ambiente de desenvolvimento integrado (IDE) IntelliJ IDEA, ou seja, é compatível com qualquer sistema operativo que execute o IDE. O objetivo desta versão é oferecer uma interface intuitiva e fácil de usar, para que os alunos tenham um ambiente produtivo e facilitado de trabalho, através da consulta dos detalhes das tarefas, submissão de trabalhos e verificação dos seus respetivos resultados de avaliação de forma intuitiva.

Através do kit de desenvolvimento de plugins, foi possível a implementação da integração da interface gráfica total com o IDE, nesta versão. O que trouxe um novo e moderno visual ao plugin. Podendo-se observar nos seguintes links:

- Vídeo demonstrativo
 - v0.9
 - <https://youtu.be/fkqXJ04pUjs>
- Repositório Git
 - <https://github.com/bernardovlbaltazar/Drop-Project-for-IntelliJ-Idea>
- Atualmente, disponível no Marketplace:
 - [Drop Project - IntelliJ IDEs Plugin | Marketplace \(jetbrains.com\)](#)

5.2 Tecnologias e Ferramentas Utilizadas

A tecnologia utilizada no desenvolvimento desta versão do plugin é baseada em Kotlin, utilizando o kit de desenvolvimento de software (SDK) do IntelliJ IDEA chamado IntelliJ Platform Plugin SDK. O desenvolvimento de plugins para o IntelliJ IDEA também pode utilizar outras tecnologias como XML, Groovy e outras linguagens de programação com suporte ao SDK (Java). A versão anterior deste plugin utilizava Groovy e Kotlin, tendo passado para 100% Kotlin com a finalidade de uniformizar a linguagem de programação do projeto. Para o gerenciamento de dependências e automatização do processo de construção foi mantida a ferramenta Gradle, dado que é mais flexível e configurável comparando com Maven, a outra única opção.

Gradle

Gradle é uma ferramenta *open source* de automação de compilação baseado em Java. Ferramenta esta, que é utilizada neste projeto, permite aos programadores criar e gerenciar projetos de software de maneira flexível e escalável. É projetada para ser compatível com outras

ferramentas de construção e pode ser integrada com outras ferramentas de desenvolvimento, sendo uma delas o IntelliJ IDEA. Uma das principais vantagens do Gradle é sua capacidade de lidar com projetos complexos.

O Gradle utiliza um modelo de projeto baseado em arquivos de configuração, que permite aos programadores definir facilmente dependências, configurações de compilação e outras informações do projeto, deste modo é possível fazer mudanças de maneira rápida que permitem compatibilidade com outras ferramentas (por exemplo, compatibilidade às versões da plataforma IntelliJ). Mais importante, a ferramenta oferece suporte para criação de plugins, o que permite aos utilizadores desta estenderem as funcionalidades do Gradle com funcionalidades adicionais.

IntelliJ Platform Plugin SDK

O IntelliJ Platform Plugin SDK é uma ferramenta de desenvolvimento que permite a criação de plugins personalizados para o IDE IntelliJ. Fornece todas as ferramentas e recursos necessários para criar, testar e publicar plugins para o IntelliJ IDEA e outras IDEs baseadas na plataforma IntelliJ como PyCharm, CLion, etc..

Este SDK é baseado na linguagem de programação Java e utiliza a biblioteca de desenvolvimento do IntelliJ Platform, que fornece uma API para aceder a recursos e funcionalidades da plataforma IntelliJ, dando acesso à criação de componentes para a interface do IDE: como janelas, menus, barras de ferramentas, entre outros.

Kotlin

Conhecida por ser uma linguagem mais segura, concisa e fácil de ler, a linguagem de programação de alto nível, Kotlin, é baseada na linguagem Java e possui muitas características semelhantes. Por ser compatível com a plataforma Java, as ferramentas anteriormente enunciadas funcionam na perfeição com esta tecnologia. Kotlin foi desenvolvida pela JetBrains (criadora da plataforma IntelliJ), uma empenhada companhia que foi fundada em 2010. Esta linguagem de programação ganhou popularidade em 2018 com o lançamento da versão 1.3, que anunciava suporte oficial para desenvolvimento multiplataforma, o que permitiu aos programadores escrever código que podia ser executado tanto em ambientes Java quanto noutras plataformas, como JavaScript.

Capacidade de integração das ferramentas

A escolha das ferramentas Gradle e IntelliJ Platform Plugin SDK, utilizando Kotlin, para o desenvolvimento de um plugin para o IntelliJ IDEA é baseada na capacidade destas ferramentas trabalharem de forma integrada e eficiente, permitindo aos programadores criar plugins de alta qualidade. O Gradle fornece uma estrutura de construção flexível e escalável, enquanto o SDK permite aos programadores criar plugins que se integram perfeitamente com o IDE. A utilização do Kotlin como linguagem de programação fornece aos programadores uma linguagem moderna e segura, com recursos avançados, e a capacidade de manter a compatibilidade com a plataforma Java.

5.3 Implementação

O ambiente produtivo da solução a desenvolver é o próprio IntelliJ IDEA. O plugin será implantado como uma extensão dentro do ambiente de desenvolvimento, permitindo que os

utilizadores acessem e utilizem as suas funcionalidades diretamente no IDE. O plugin será construído usando as ferramentas Gradle, IntelliJ Platform Plugin SDK e Kotlin (como dito anteriormente), e será testado dentro do ambiente do IDE antes de ser disponibilizado para os utilizadores finais. Como produto final, o plugin será distribuído através da plataforma de plugins oficial do IntelliJ IDEA, permitindo aos utilizadores instalarem-no e atualizarem-no facilmente.

5.4 Abrangência

A solução desenvolvida abrange as seguintes disciplinas e áreas científicas do curso:

- Programação orientada a objetos:
 - Utilização da linguagem de programação Kotlin, que é fortemente orientada a objetos.
- Arquitetura de software:
 - Desenvolvimento do plugin de forma modular, permitindo uma fácil manutenção e escalabilidade.
- Engenharia de software:
 - Garantir a qualidade do software, através de técnicas de teste e verificação;
 - Metodologias ágeis:
 - Garantir entrega do plugin no prazo previsto;
 - Garantir que as necessidades do utilizador sejam atendidas.
- Segurança de dados:
 - Garantir a segurança do plugin, através de ações que dificultem os acessos não autorizados.

6 Método e Planeamento

Esta secção descreve o método de trabalho seguido no desenvolvimento do projeto, levando em consideração os calendários propostos nos relatórios anteriores. Além disso, também farei uma análise crítica do cumprimento desses calendários, com base nas apreciações de progresso realizadas em cada avaliação, proporcionando uma avaliação geral do planeamento e execução do projeto ao longo do ano.

Durante a fase inicial do projeto, estabelecemos um plano detalhado, identificando as principais etapas, prazos e recursos necessários. O calendário proposto foi usado como ponto de partida para o desenvolvimento do plano de trabalho. No entanto, ao longo do ano, enfrentámos desafios inesperados e mudanças de objetivos que impactaram o cumprimento do calendário. Nesse calendário, estava previsto o desenvolvimento de um plugin adicional destinado aos professores, para além da vertente alunos. No entanto, essa meta foi interrompida quando o plugin dos alunos entrou em fase de testes, tendo necessitado atenção adicional para corrigir erros e incorporar sugestões fornecidas pelos utilizadores.

Após a reavaliação, o calendário foi reajustado consoante a situação, levando em consideração a priorização da vertente dos alunos como foco principal de atenção. Sendo assim, o calendário foi reconfigurado da seguinte maneira:

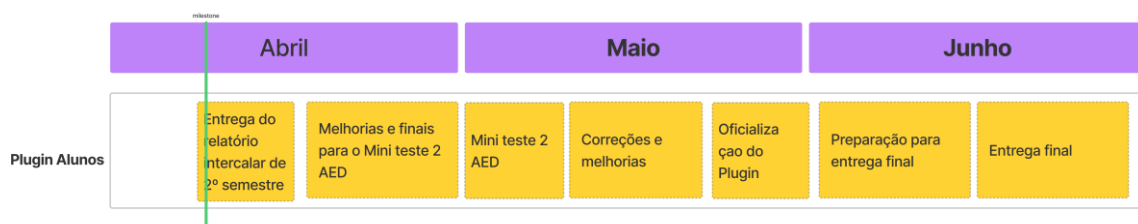


Figura 7 - Timeline do planeamento da avaliação anterior

Por meio da adoção de uma abordagem flexível, revisão periódica dos prazos conforme necessário e priorização de tarefas críticas, o planeamento foi executado de acordo com as expectativas estabelecidas. Como resultado, alcançamos a meta mais significativa do projeto, que era o lançamento da versão oficial do plugin destinado aos alunos. Através dessas medidas, pudemos assegurar que os recursos foram devidamente alocados, permitindo o cumprimento dos prazos estabelecidos e a conclusão bem-sucedida do planeamento.



Figura 8 - Planeamento e execução do projeto ao longo do ano

7 Resultados

Esta secção tem como objetivo apresentar os resultados obtidos por meio da solução desenvolvida, com foco na continuidade dos resultados ao longo do segundo semestre. Para uma melhor compreensão, esta secção é dividida em duas subsecções: Enquadramento, Resultados e Sugestões e finalmente Inquérito Final e Adesão.

Na subsecção do enquadramento, são fornecidas informações contextualizadas que permitem compreender o contexto em que os resultados foram alcançados. Aborda aspetos relevantes, como o ambiente de testes e os critérios de avaliação adotados. Esta contextualização serve para a compreensão dos resultados subsequentes.

Na subsecção dos Resultados e Sugestões, são apresentados os resultados obtidos ao longo do segundo semestre, demonstrando a continuidade e evolução dos *outcomes* gerados pela solução implementada. Detalha-se métricas utilizadas, incluindo os pontos melhorados e pontos a melhorar.

Na subsecção do Inquérito Final e Adesão, são demonstrados a adesão do plugin por parte dos alunos e os resultados de um inquérito feito com intuito final, após uma longa fase de experimentação e testes. Nesta etapa conclusiva, procurou-se obter informações atualizadas relativamente à opinião relativa ao plugin e fazer análises não só comparativas como de desenvolvimento.

Com base nesta estrutura, espera-se que esta secção proporcione uma visão abrangente dos resultados alcançados, destacando a sua continuidade e relevância ao longo do segundo semestre.

7.1 Enquadramento

No início de 2023, foi definido um plano de trabalho para o projeto que incluía a entrada na fase de testes. O objetivo desta fase era detetar quaisquer erros e melhorar a usabilidade do plugin de maneira geral. Este plano de trabalho foi desenvolvido para garantir que o plugin estivesse completamente funcional e otimizado para atender às necessidades dos alunos.

A fase de testes teve como objetivo principal a apresentação da ferramenta aos alunos da unidade curricular algoritmia e estrutura de dados e o incentivo da sua utilização, de modo a garantir o seu funcionamento em todas as situações. Graças a isso, foram realizados testes de configuração, funcionalidade, desempenho, compatibilidade e usabilidade. Nesta fase, as aulas desta unidade curricular foram atendidas pelo orientador e principalmente pelo programador, de modo a analisar e detetar quaisquer erros ou problemas que o plugin pudesse vir a ter. Qualquer problema encontrado durante as aulas foi corrigido o mais breve possível, para garantir que o plugin funcionasse corretamente e não perdesse o envolvimento dos alunos. A melhoria da usabilidade do plugin também foi um objetivo chave durante a fase de testes. Com base nos resultados dos testes e feedback dos alunos, foram feitas melhorias na interface do utilizador e na sua experiência para tornar o plugin mais fácil e eficiente de usar.

7.2 Resultados e Sugestões

Durante este período, foram identificados problemas e sugestões de alunos e professores, que foram levados em consideração de modo a melhorar o plugin. Para facilitar o controlo de correção de erros e melhorias, foram sendo definidas diferentes versões de lançamento o do plugin.

7.2.1 Versões de lançamento:

A tabela abaixo apresenta as diferentes versões de lançamento do plugin, juntamente com a informação sobre as melhorias e correções incluídas em cada versão, como também os seus problemas e sugestões:

Versão	Melhorias e correções	Problemas e sugestões
1.3 alpha	<ul style="list-style-type: none"> • Adicionada funcionalidade de refrescar a lista de tarefas; • Estado de início de sessão preservado; • Formatação inicial das instruções da tarefa; • Identificador de tarefa selecionada (Radio Button). 	<ul style="list-style-type: none"> • Problema na seleção das tarefas, por vezes apareciam duas tarefas selecionadas; • Pouca usabilidade na instruções e relatórios de resultados, pois estão em janelas separadas do IDE.
1.4 alpha	<ul style="list-style-type: none"> • Integração da página de instruções e do relatório de resultados na interface gráfica do IDE; • Adicionada funcionalidade de verificar o último relatório de resultados da tarefa; • Problema de seleção múltipla de tarefas resolvido; • Tarefa selecionada mantida entre sessões. 	<ul style="list-style-type: none"> • Necessidade de adicionar tarefas/fichas de aula manualmente; • Tarefa selecionada partilhada com outros projetos gerava confusão; • Pouca usabilidade no relatório de resultado da tarefa, mesmo estando integrado no IDE (não responsivo).
1.5 alpha	<ul style="list-style-type: none"> • Adicionada funcionalidade de adição manual de uma tarefa; • Alteração da estrutura das instruções (API); 	<ul style="list-style-type: none"> • Erro de múltiplos projetos abertos: Partilha de funcionalidades e informações do plugin causavam interferências quando outro

	<ul style="list-style-type: none"> Tarefa selecionada ligada unicamente a um projeto. 	<p>projeto era aberto em simultâneo;</p> <ul style="list-style-type: none"> Contagem mal feita de erros no relatório de resultados; Desaparecimento da tarefa adicionada manualmente quando refrescava a lista.
1.6 alpha	<ul style="list-style-type: none"> Reformatação e correção de erros nos relatórios de resultados da tarefa; Adicionada funcionalidade de desselecionar tarefa. 	<ul style="list-style-type: none"> Não funcionamento do botão para procurar a tarefa a adicionar.
1.7 alpha	<ul style="list-style-type: none"> Resolução do problema de múltiplos projetos identificada na versão 1.5; Resolução do problema do desaparecimento da tarefa adicionada manualmente, ao refrescar a lista; Resolução do problema do botão de procurar tarefa; Adicionado redireccionamento direto para consultar a token, no momento de início de sessão; Remoção da funcionalidade de desselecionar tarefa. 	<ul style="list-style-type: none"> Desaparecimento das tarefas não selecionadas e adicionadas manualmente, ao refrescar a lista de tarefas; Pedido de melhoria da interface de início de sessão; Data de vencimento errada de algumas tarefas; Pedido de adição de configurações customizáveis do plugin.
1.8 alpha	<ul style="list-style-type: none"> Nova Interface de início de sessão; Adicionada funcionalidade de configurações do plugin; Funcionalidade de preenchimento automático do início de sessão através 	<ul style="list-style-type: none"> Tarefas adicionadas num projeto não são passadas para outros projetos

	das configurações do plugin; <ul style="list-style-type: none"> • Botão para o último relatório de resultados da tarefa desabilitado ao submeter o projeto. 	
0.9 oficial	<ul style="list-style-type: none"> • Melhoramento da compatibilidade nas versões do IDE • Nova funcionalidade de esquecer tarefa • Tarefas adicionadas partilhadas entre sessões e projetos • Disponível no Marketplace oficial da JetBrains 	

7.2.2 Alterações e correções extra

Além das melhorias e correções das funcionalidades do plugin em si, houve a necessidade de corrigir e aprimorar a API do Drop Project. Essa melhoria foi necessária para permitir que o plugin comunique de maneira mais eficiente com a API, a fim de atender às necessidades dos alunos. As melhorias na API do Drop Project foram um fator essencial para a melhoria global do plugin, permitindo, assim, oferecer aos alunos uma experiência mais eficiente, tendo sido principalmente estas:

- Novo *endpoint* criado, que recebe um identificador de uma tarefa e é devolvida a informação do mesmo;
- Alteração da estrutura das Instruções de modo a suportar diferentes tipos de ficheiro;
- Atributo modo de submissão criado de maneira a entender a adesão do plugin;
- Implementação de funcionalidades de acessibilidade complementares ao plugin: botões de fácil acesso para cópia do id da tarefa e da *token*;
- Resolução de certos erros, que afetavam o plugin.

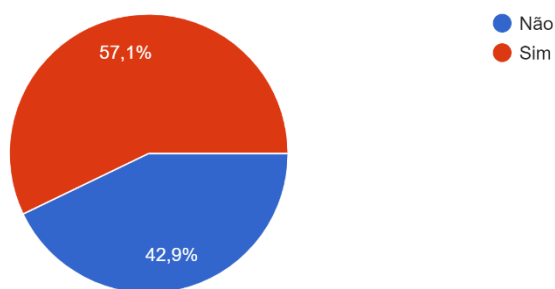
7.3 Inquérito Final e Adesão

De acordo com os resultados deste inquérito final, verificou-se que, em geral, os alunos reconheceram a utilidade do plugin no combate aos problemas identificados. No entanto, alguns desafios foram identificados em relação à sua utilização, como a falta de informação atualizada sobre o plugin (a falta de adoção da versão mais recente do plugin, pelo qual gerou feedback que já havia sido tratado previamente) ou a necessidade de se familiarizar com a interface gráfica. Estes resultados indicam que, embora o plugin tenha uma base sólida, ainda são necessários ajustes para atender às necessidades dos alunos que podem vir a enfrentar estes mesmos problemas. Como estratégia para combater essa questão, foi criado um [vídeo](#) com uma explicação detalhada do fluxo de trabalho do plugin, que visa fornecer um suporte adicional aos alunos. Apesar destes desafios, é importante ressaltar que os resultados obtidos não foram

inteiramente negativos. Esta constatação justifica ainda mais a pertinência do projeto e sinaliza a viabilidade do seu futuro, incluindo a possibilidade de ser adotado pelos alunos nos próximos anos letivos. Através do inquérito, foi possível obter *insights* valiosos sobre a percepção dos utilizadores em relação ao plugin e identificar áreas de melhoria, o que contribuirá para o melhoramento futuro do projeto, o que poderá levar a uma maior aceitação por parte dos alunos, abrindo caminho para um futuro promissor no qual o plugin poderá ser amplamente adotado pelos alunos nos próximos anos letivos. De seguida, apresentam-se os resultados.

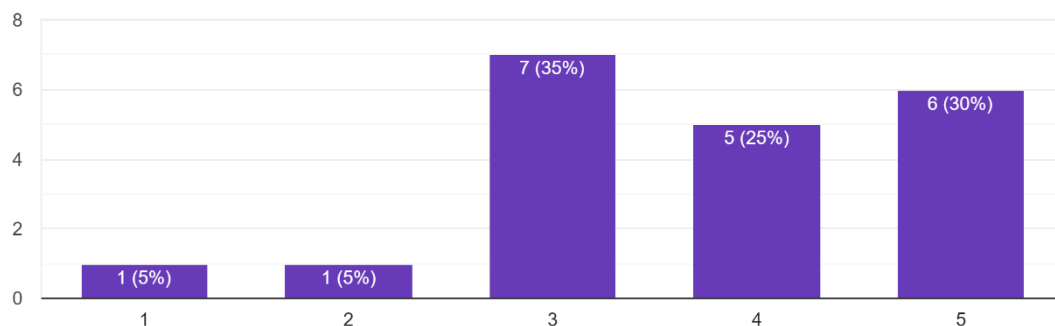
Já usaste o Plugin DP?

35 respostas



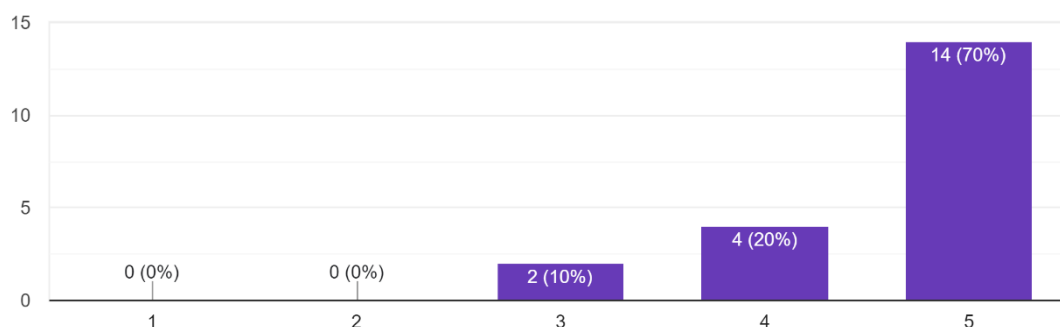
Como classifica o seu uso do Plugin DP? 1 - Nunca uso 2 - Experimentei, mas não gostei da experiência 3 - Uso às vezes 4 - Uso quase sempre 5 - Uso sempre que possível

20 respostas



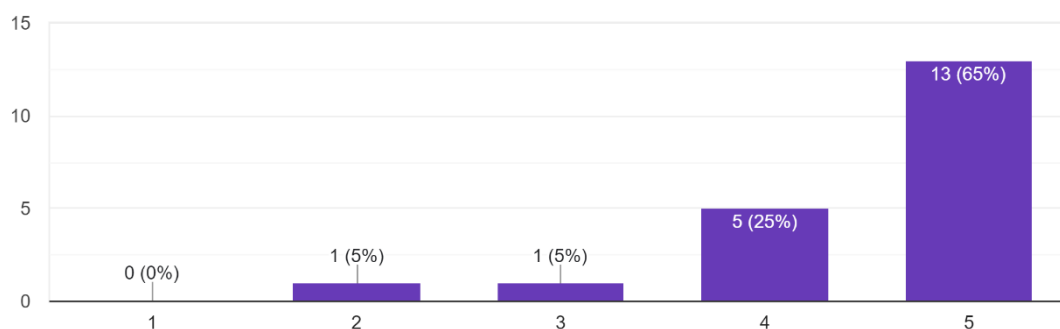
Como classifica a eficiência do plugin na simplificação do processo de submissão para o DP?

20 respostas



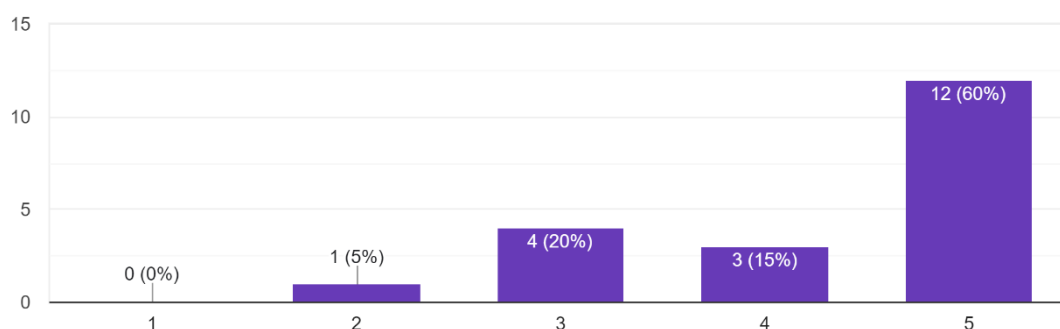
O uso do Plugin DP resulta numa economia de tempo e/ou esforço quando comparado com o método tradicional de submissão (upload web) ?

20 respostas



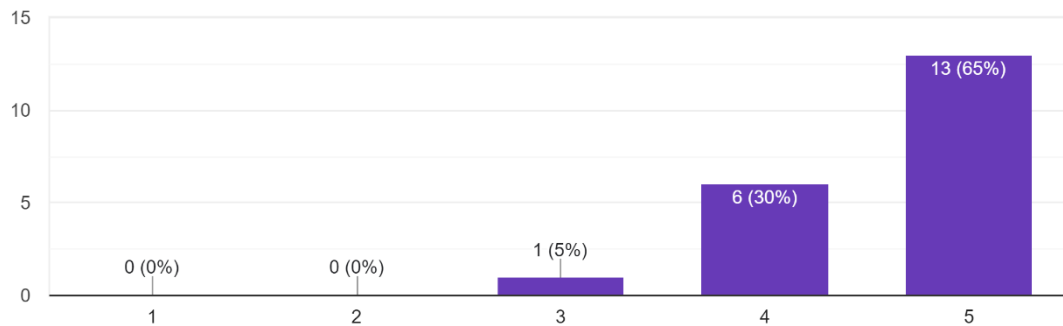
O quão relevante é para si a criação automática do ficheiro AUTHORS.txt e o envio automático do zip com a pasta "/src" e o AUTHORS.txt

20 respostas



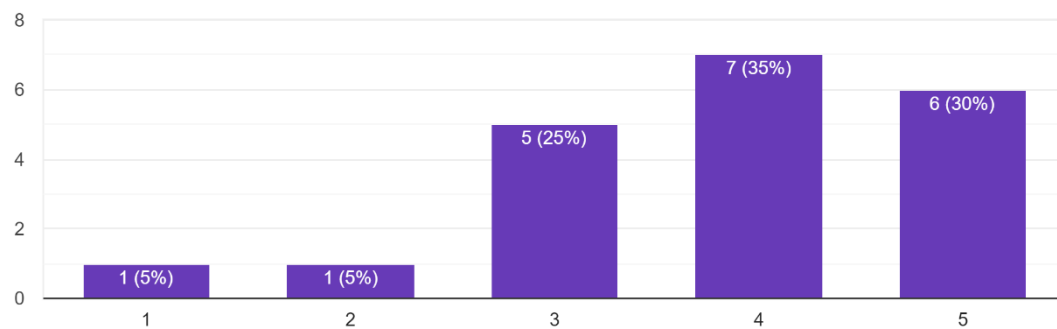
Como classifica a sua confiança na submissão de exercícios através do plugin?

20 respostas



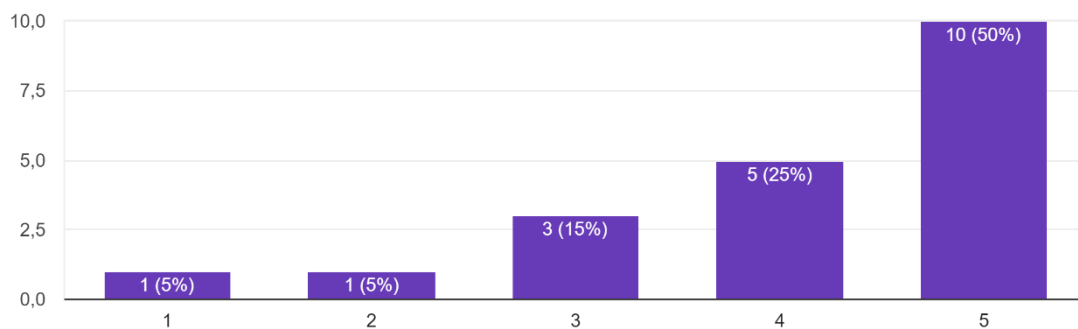
De 1 a 5, indique o quanto concorda com a seguinte afirmação: "A capacidade de ver os enunciados das tarefas diretamente no IDE tem efeitos positivos na minha produtividade."

20 respostas



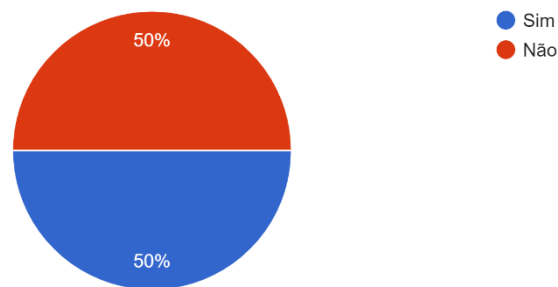
De 1 a 5, indique o quão concorda com a seguinte afirmação: "A capacidade de visualizar o "build report" de uma submissão diretamente no IDE tem efeitos positivos na minha produtividade."

20 respostas



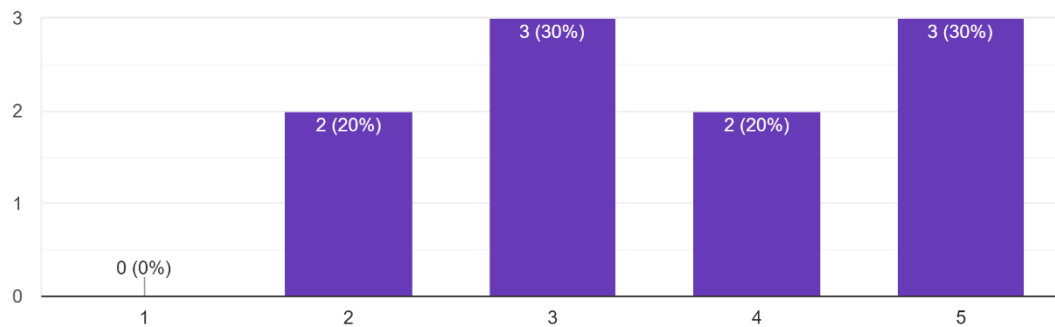
Usaste o Plugin DP no Mini-Teste 2?

20 respostas



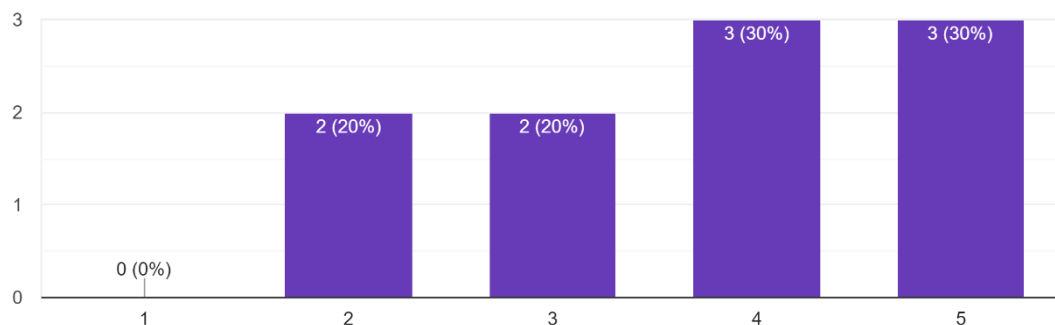
Se usaste o Plugin DP no Mini-Teste 2: Classifica, de 1-a-5, o quanto concordas com esta afirmação: O uso do Plugin contribuiu para uma r...do nervosismo durante a realização do mini-teste.

10 respostas



Se usaste o Plugin DP no Mini-Teste 2: Classifica, de 1-a-5, o quanto concordas com esta afirmação: O uso do Plugin contribuiu para uma ...eu desempenho durante a realização do mini-teste.

10 respostas



7.3.1 Adesão do plugin

A verificação dos dados da utilização do plugin é um aspeto fundamental para avaliar o impacto do mesmo. A fim de realizar essa verificação, é importante definir uma métrica de avaliação adequada que permita agregar dados precisos e relevantes. No caso específico do

plugin em questão, uma das métricas de avaliação definidas foi a distinção do modo de submissão entre submeter pela Web e pela API. Esta métrica é especialmente importante, uma vez que o plugin é único a utilizar a API.

Ao avaliar a distinção do modo de submissão, foi possível identificar que 17% das submissões totais foram feitas através do plugin na realização de mini teste, fichas de aula, e o projeto prático como mostra a figura seguinte.

count(*)	submission_mode
13232	0
2713	2

Figura 9 - Submissões totais (0=UPLOAD; 2=API)

Foi também possível obter as submissões realizadas apenas para o projeto prático, o que ajuda para uma análise mais aprofundada de maneira a interpretar a evolução da adesão do plugin.

3005	aed-2223-projeto-p1
119	aed-2223-projeto-p1
3466	aed-2223-projeto-p2
319	aed-2223-projeto-p2

Figura 10 - Submissões projeto prático parte 1 e parte 2 (linhas escuras = UPLOAP; linhas claras = API)

Os resultados obtidos revelam uma notável discrepância em relação às submissões do projeto. Essa disparidade pode ser atribuída a várias razões possíveis, sendo a principal delas a falta de informação por parte dos alunos sobre a possibilidade de submeter o projeto por meio do plugin. Embora essa razão seja válida e tenha sido mencionada pelos alunos, é importante ressaltar que não podemos generalizar e considerá-la como a única causa. É possível que muitos alunos simplesmente não estivessem cientes dessa opção e, portanto, optaram pelo método tradicional de submissão. No entanto, é crucial reconhecer que existem outras possíveis razões para a discrepância observada. Fatores como preferência pessoal dos alunos, familiaridade com os métodos tradicionais de submissão ou resistência à adoção de uma nova ferramenta também podem ter influenciado a decisão de não utilizar o plugin para submeter os projetos. É importante considerar essas diferentes perspectivas para obter uma compreensão abrangente dos motivos por trás dos resultados obtidos.

Adicionalmente, é importante destacar que, embora a taxa de submissão através do plugin tenha sido baixa em geral, houve um aumento significativo de submissões da primeira parte do projeto para a segunda parte, chegando a mais do que o dobro. No entanto, é necessário ter cautela ao tirar a conclusão de que o uso do plugin foi duplicado nesse caso

específico, pois existem diversas variáveis que podem ter influenciado esse resultado. Uma dessas variáveis é o facto de que a segunda parte do projeto pode ter sido mais trabalhosa em comparação com a primeira, o que pode ter levado a um maior número de submissões por parte de cada aluno. Isso poderia explicar, em parte, o aumento observado nas submissões do plugin. No entanto, é necessário apresentar uma contra-argumentação em relação a este ponto: ao analisar os dados, nota-se que o método tradicional de submissão também registou um aumento de 15% entre a primeira e a segunda parte do projeto. Em contraste, a submissão pelo plugin apresentou um aumento muito mais expressivo, atingindo 168%. Essa disparidade sugere que, apesar das possíveis influências das características do projeto, o uso do plugin em projetos teve um crescimento substancial em comparação com o método tradicional de submissão. Portanto, embora seja necessário levar em consideração as variáveis que podem ter influenciado estes resultados, os dados indicam claramente que o plugin desempenhou um papel significativo no aumento das submissões ao longo do projeto. Essa tendência de crescimento ressalta a importância de continuar a promover e melhorar o uso do plugin como uma alternativa válida e eficiente para a submissão de projetos.

8 Conclusão e Trabalhos Futuro

Na conclusão deste projeto, foram obtidas diversas aprendizagens relevantes no processo de desenvolvimento de software. Nos métodos eficazes de implementação de forma estruturada e eficiente, ficou evidente a importância de realizar um plano detalhado e estabelecer metas alcançáveis ao longo do desenvolvimento. A adoção de abordagens iterativas também se mostrou valiosa, permitindo ajustes e melhorias contínuas à medida que novos desafios surgiam. A interação e reação com os utilizadores (alunos) desempenharam um papel crucial no melhoramento do projeto. Ao obter feedback direto dos alunos, foi possível compreender melhor as suas necessidades e expectativas, resultando em ajustes que proporcionavam uma experiência mais positiva. Outro ponto relevante foi a estruturação e o planeamento adequados do projeto. A definição clara dos requisitos e a implementação das funcionalidades em etapas viáveis foram essenciais para garantir um desenvolvimento organizado e eficiente. Adicionalmente, a implementação de funcionalidades em projetos já existentes, como o Drop Project, demonstrou a importância de integrar novas características de forma harmoniosa, preservando a integridade e a usabilidade da plataforma principal.

Com base nas experiências aprendidas, o objetivo visa o desenvolvimento e a manutenção contínua da ferramenta, de maneira a integrá-la no dia a dia dos alunos. Este projeto abriu portas para a inquirição de novas funcionalidades e melhorias. A partir do feedback dos alunos, será possível explorar e incorporar novas funcionalidades que agreguem valor e potencializem ainda mais a utilidade do plugin.

Bibliografia

- [AATs05] Douce C., Livingstone D., Orwell J., *Automatic test-based assessment of programming: A review*, *Journal on Educational Resources in Computing*, Volume 5, Issue 3, 2005
- [AnJJ12] André Silva, José Paulo Leal, José Carlos Paiva, [Raccode: an Eclipse Plugin for Assessment of Programming Exercises - DocsLib](#), 2012
- [Apac21] Apache Software Foundation, Apache Maven project, <https://maven.apache.org/>, 2021
- [Drop22] Bruno Pereira Cipriano, Nuno Fachada, Pedro Alves, Drop Project: An automatic assessment tool for programming assignments, <https://www.sciencedirect.com/science/article/pii/S2352711022000577>, 2022
- [InSi15] Insa D, Silva J, Semi-Automatic Assessment of Unrestrained Java Code: A Library, a DSL, and a Workbench to Assess Exams and Exercises, https://dl.acm.org/doi/abs/10.1145/2729094.2742615?casa_token=TkSXgSJONNgAAAAA:b0Z0KNe4UqgZRHINDuLH9pgjwY8jFSdA7N77-VRaRzkyoZuZXqMGrJi5AUYGW05tutTXTQX4Q6_i8KE, 2015
- [Kode19] Kodeco, Domain-Specific Languages In Kotlin: Getting Started <https://www.kodeco.com/2780058-domain-specific-languages-in-kotlin-getting-started>, 2019
- [Leht05] Lehtone T., Java-addictive e-learning of the Java programming language, https://www.researchgate.net/profile/Tomi-Suovuo/publication/31596693_Koli_Calling_2005_-_Proceedings_of_the_Fifth_Koli_Calling_Conference_on_Computer_Science_Education/links/5daeee4fa6fdccc99d92b54c/Koli-Calling-2005-Proceedings-of-the-Fifth-Koli-Calling-Conference-on-Computer-Science-Education.pdf#page=48,2005
- [Sdk00] JetBrains, IntelliJ Platform Plugin SDK, <https://plugins.jetbrains.com/docs/intellij/welcome.html#getting-started>
- [Grd00] Gradle, Document on “What is Gradle?”, https://docs.gradle.org/current/userguide/what_is_gradle.html

Anexos

Plano de trabalho resultante da fase de testes. De notar que a fase de testes também envolveu os alunos da unidade curricular de linguagens de programação 2, não tendo sido destacado anteriormente pois o foco maior seriam os alunos do primeiro ano da unidade curricular de algoritmia e estrutura de dados.

Cenário	Detalhes
Participação das últimas aulas de LP2 (dezembro, janeiro), de modo que os alunos usem o plugin.	O objetivo é tentar fazer alguns testes das funcionalidades entretanto melhoradas. Nestas aulas os alunos devem estar principalmente preocupados com o projeto, mas poderá existir a hipótese de alguém querer experimentar o plugin.
Início de 2 semestre	O objetivo é ter o plugin estável no início do 2º semestre: Neste caso, “estável” significa ter a confiança para sugerir aos alunos a utilização do plugin no mini teste. Também convém que a interface gráfica não mude muito a partir deste ponto, salvo correções de bugs.
Primeira aula de algoritmia e estrutura de dados do 2º semestre.	Apresentação do plugin aos alunos. Indicação dos requisitos para a utilização do plugin (versão mínima necessária do IDE).
Na segunda semana de aulas, os alunos de AED experimentam o plugin nas fichas da aula para ganhar experiência de usabilidade.	Participação em várias aulas das semanas 2 e 3: <ul style="list-style-type: none">• Acompanhamento dos alunos e anotação de eventuais erros/problemas.• Caso haja o aparecimento de erros/problemas, tentar corrigir o mais rapidamente possível. Neste momento, reúne-se e avalia-se quais os bugs/problemas que são críticos.
Quarta semana de aulas: o primeiro mini teste de AED	Nessa semana, os alunos fazem o teste, e podem escolher usar o plugin. No fim do teste, ou em casa, respondem a um questionário sobre a utilização do plugin.
Após o primeiro mini teste, fazem-se as restantes melhorias que já estão identificadas e que ainda estiverem pendentes	Depois desta fase, o plugin dos alunos seria oficializado

Glossário

LEI	Licenciatura em Engenharia Informática
TFC	Trabalho Final de Curso
AAT	Automated Assesment Tool
DP	Drop Project
UI	User Interface