PROJETO - LabSecurity

Rui Paredes

RELATÓRIO PARA A OBTENÇÃO DO GRAU DE LICENCIADO

EM ENGENHARIA INFORMÁTICA

Jul/2018

## Elementos Identificativos

**Nome:** Rui Miguel Andrês Paredes

Número: 1011670

**Curso:** Licenciatura em Engenharia Informática

Ano Letivo: 2017/2018

**Escola:** Escola Superior de Tecnologia e Gestão – Instituto Politécnico da Guarda

**Orientador:** Professor José Carlos Coelho Martins Fonseca

Projeto realizado entre: maio e julho de 2018

## Agradecimentos

## Resumo

## Abstract

## Índice

## Siglário

# Introdução

Este documento desenvolvido pelo aluno Rui Paredes no âmbito da unidade curricular Projeto de Informática, da Licenciatura de Engenharia Informática, descreve o projeto realizado para aprovação à mesma.

O Projeto desenvolvido consiste numa plataforma web de programação e segurança, onde podem ser obtidas informações sobre vários ciberataques, realizar *quizzes* e competir contra outros utilizadores em competições de *hacking*.

## Motivação

Todos os anos, alunos da UC de Programação e Segurança desenvolvem exemplos que ilustram vulnerabilidades, ataques e defesas na área da cibersegurança. Porém, após a apresentação dos mesmos para avaliação, estes exemplos provavelmente não voltam a ser usados para o estudo de determinadas vulnerabilidades.

Por estas razões surgiu a ideia de desenvolver um laboratório de programação e segurança, que consiste numa plataforma web, onde se poderá aprender mais acerca de vulnerabilidades, ataques e defesas ao nível da segurança informática, contando com vários exemplos desenvolvidos por alunos da UC de Programação e Segurança, servindo ainda desta forma como um repositório.

Este laboratório de programação e segurança para lá de disponibilizar informações acerca de várias vulnerabilidades, ciberataques existentes e como defender dos mesmos, permite ainda realizar *quizzes* e competições de *hacking* contra outros utilizadores, servindo assim como uma plataforma de aprendizagem e de desenvolvimento das capacidades dos seus utilizadores.

## Objetivos da Plataforma

Esta plataforma tem como principais objetivos:

* Realizar desafios de cibersegurança
* Participar em competições de cibersegurança
* Verificação automática das respostas a quizzes ou desafios.
* Sistema de Ranking por pontos
* Plataforma online
* Realizar Quizzes
* Ver lições teóricas sobre as vulnerabilidades/ataques
* Inserção de Propostas

## Fases do Trabalho

## Estrutura do Documento

# Estado da Arte

Definidos os objetivos propostos para o desenvolvimento do projeto, foi realizada uma pesquisa de forma a saber quais as soluções atualmente existentes.

Desta pesquisa foram encontradas várias plataformas em que alguns dos objetivos são idênticos aos definidos inicialmente para este projeto, e de onde foram retiradas informações acerca do seu funcionamento que poderão servir de apoio para o desenvolvimento da plataforma web do LabSecurity.

Estas plataformas web, embora com alguns objetivos idênticos aos propostos, não chegam a completar todos os objetivos propostos ou divergem em alguns pontos importantes dos mesmos. Algumas delas focam -se em jogos CTF (*Capture the flag*) entre equipas ou indivíduos, ou apenas no ataque a cada vulnerabilidade sem competição direta com outros utilizadores.

## Plataformas existentes

Das várias aplicações encontradas na fase de pesquisa foram selecionadas para análise duas que demonstraram ter um maior reconhecimento na comunidade, sendo estas a Hacking Lab e a CTF365 e ainda três sugeridas pelo Orientador, sendo estas o Reversing.kr, RingZer0Team e WebGoat.

Na Tabela 1 são apresentados os objetivos definidos inicialmente para a plataforma web do LabSecurity e se cada solução existente contém essas características ou não.

Tabela 1- Aplicações existentes e objetivos propostos

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Aplicação** | **Ranking** | **Uso de Flags** | **Auto- Verificação da Resposta** | **Ataques na Plataforma Web** | **Propostas de Ataques** | **Vulnerabilidades - Lessons** | **Quizzes** |
| Hacking-Lab | Sim | Não | Não | Não | Não | Não | Não |
| CTF365 | Sim | Não | Não | Sim | Não | Não | Não |
| Reversing.kr | Sim | Sim | Sim | Não | Não | Não | Não |
| RingZeroTeam | Sim | Sim | Sim | Sim | Não | Não | Não |
| WebGoat | Sim | Sim | Sim | Sim | Não | Sim | Não |

### Hacking Lab

O Hacking-Lab [1] consiste numa plataforma de *hacking* ético onde podem ser realizados desafios de segurança que contem várias competições que vão desde criptografia, a engenharia-reversa, entre outras, e que variam ainda dentro de cada categoria em dificuldade, podendo se observar na Figura 1 como estes se encontram apresentados.

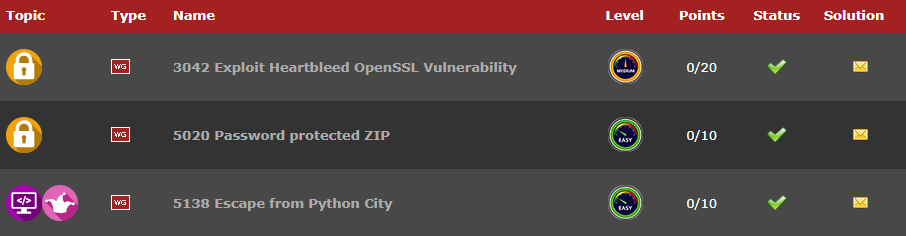
Ao nível dos desafios, o Hacking-Lab apresenta uma variedade de desafios. Algumas destas competições requerem que o utilizador use software adicional para resolver determinados desafios.

O Hacking-Lab dispõe ainda de uma secção de divulgação de conferências de cibersegurança.

De forma a ser possível aos utilizadores resolver os desafios é necessário que estes instalem uma máquina virtual com um sistema fornecido pela plataforma e se conectem através de VPN a esta.

Após resolvido um desafio e submetido, a solução apresentada pelo utilizador é enviada para ser analisada por uma equipa, e caso esta se encontre correta, o mesmo obtém pontos, sendo estes adicionados ao *rank* do utilizador.

Figura 1- Desafios Hacking- Lab

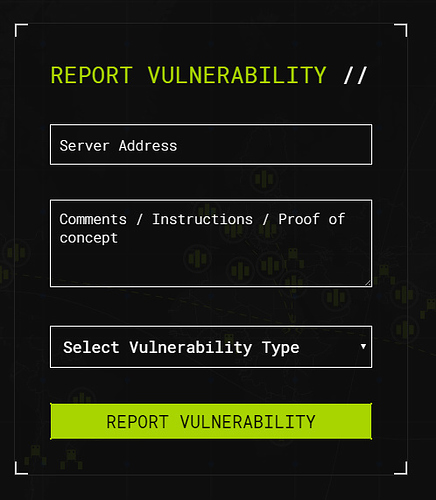


### CTF365

O CTF365 [2] consiste numa plataforma com jogos CTF (*Capture the flag*), em que são simulados ataques a vulnerabilidades aos serviços que cada equipa possui, e em que o objetivo principal consiste nas equipas tornarem os seus serviços o mais invulneráveis que conseguem enquanto atacam os serviços de equipas inimigas. Os serviços de cada equipa encontram -se nos vários “*Fortress*” que estas possuem, sendo que um *fortress* corresponde a uma máquina virtual atribuída para a equipa. Na realização de ataques a um *fortress* de outras equipa, os utilizadores podem usar uma grande variedade de ferramentas, não havendo uma grande quantidade de regras que os utilizadores devem respeitar nos seus ataques. A criação de um *fortress* é exclusiva a equipas que façam parte de um plano que requer pagamento, porém, qualquer utilizador que esteja registado na plataforma, com ou sem equipa ou plano, consegue realizar ataques.

De forma às equipas submeterem as vulnerabilidades encontradas durante os ataques e ganharem pontos é necessário na secção de submissão da resposta indicar qual o tipo de vulnerabilidade, a quem pertence o *fortress* inimigo, e qual o endereço do mesmo, podendo este processo ser verificado na Figura 2. Após a vulnerabilidade ser submetida é analisada por uma equipa do CTF365 que verificará se a vulnerabilidade realmente existe ou não no *fortress* indicado, adicionando pontos ao *rank* da equipa que detetou a vulnerabilidade.

Figura 2- CTF365 Submeter Vulnerabilidade



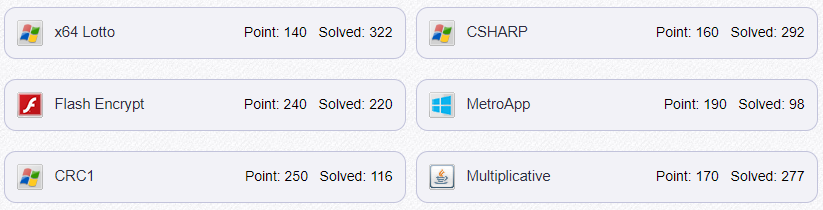
### Reversing.kr

O Reversing.kr [3] é uma plataforma de desafios de *cracking* e de uso de engenharia reversa.

Os desafios que esta plataforma possuí consistem em aplicações para diferentes ambientes (Windows, Linux, Windows Phone, …), podendo ser alguns destes observados na Figura 3, e em que o ataque é executado nesses mesmos ambientes. Quando um utilizador resolve um desafio e chega a uma solução correta, de forma a ganhar pontos na plataforma, necessita submeter a mesma numa secção própria na plataforma do Reversing.kr. Após obtenção dos pontos, estes passam a ser visíveis na tabela de *Ranking* da plataforma.

Possuí ainda uma secção onde se podem encontrar plataformas semelhantes a esta.

Figura 3- Reversing.kr - Challenges

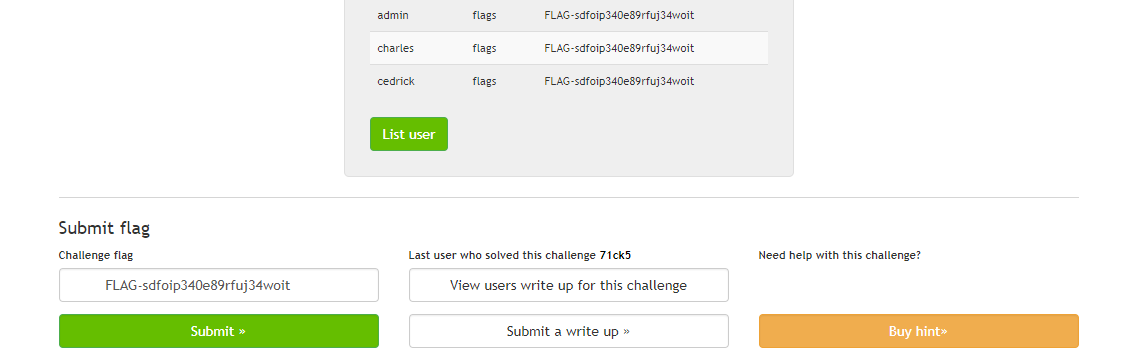


### RingZer0 Team Online CTF

O RingZer0 Team Online CTF [4] consiste numa plataforma onde se encontram disponíveis uma grande variedade de desafios de cibersegurança em jogos *capture the flag*.

Estes desafios vão desde criptografia, análise de *malwares*, SQLi, entre outros, onde alguns deles podem ser resolvidos diretamente na plataforma, enquanto que outros a resolução é feita no sistema do utilizador, por exemplo, uma aplicação com vulnerabilidade, ou então através de SSH para uma máquina onde a vulnerabilidade estará presente. Independentemente do tipo de vulnerabilidade, após realizado o ataque com sucesso, é devolvida uma *flag* que pode ser usada seguidamente na página onde o desafio realizado se encontra, obtendo pontos de acordo com o desafio realizado. A plataforma disponibiliza ainda algumas ferramentas que poderão ser usadas de forma a resolver certos desafios. Possuí ainda um *Scoreboard* onde são apresentados os 50 utilizadores com mais pontos na plataforma.

Figura 4- RingZer0Team - Desafio SQLi após ataque

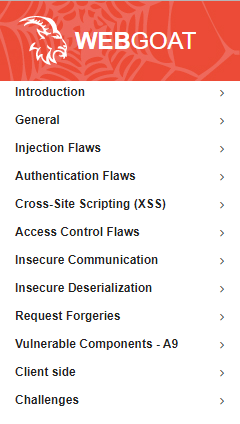


### WebGoat

A WebGoat [5] consiste numa aplicação onde se pode encontrar uma variedade de vulnerabilidades comuns em aplicações desenvolvidas em Java e componentes *open source*. Esta plataforma, para lá de ter vários desafios sobre diversas vulnerabilidades, que vão desde falhas que podem ser exploradas através de injeção de código malicioso, ataques de SQLi por exemplo, a ataques de CSRF, entre outros que podem ser observados na Figura 5, tem ainda lições teóricas acerca dos vários ataques que podem ser testados na plataforma. Todos os desafios presentes na WebGoat não necessitam de ferramentas adicionais para a sua resolução, e encontram -se divididas em dois tipos, desafios para testes das vulnerabilidades, nos quais, durante a sua resolução, os utilizadores podem obter dicas de como realizar o ataque, sendo que da sua resolução não resulta pontuação adicional ao utilizador, e desafios competitivos, nos quais não há qualquer tipo de dicas para chegar à resposta e de onde resulta uma *Flag* usada para obter pontos pelo desafio realizado.

O servidor para a plataforma WebGoat corre como servidor local, pelo qual se encontra disponível para download na internet.

Figura 5- WebGoat - Menu Principal



## Análise crítica das soluções existentes

Todas as aplicações apresentadas reúnem um conjunto de características que vão ao encontro dos objetivos do LabSecurity.

O Hacking-Lab, tem pontos fortes que devem ser referidos, como uma grande variedade de desafios a explorar e que analisam uma grande diversidade de vulnerabilidades, que vão desde desafios fáceis a difíceis. Para lá disso, a plataforma online é de fácil uso e possui as informações necessárias à realização de cada desafio. A plataforma apresenta ainda uma secção onde são divulgados eventos de cibersegurança, que apesar de não ir ao encontro dos objetivos especificados para o LabSecurity, não deixa de ser um ponto forte já que esses eventos são uma forma de divulgar informação na área da cibersegurança.

Como pontos fracos o Hacking-Lab apresenta características como a necessidade de ter de instalar a máquina virtual disponibilizada pela plataforma do Hacking-Lab e o acesso através de VPN à plataforma de forma a ter entrada a grande parte dos desafios. Para lá disso existem ainda características como o uso de ferramentas adicionais e a forma como a solução encontrada para cada desafio é avaliada.

Ao nível do CTF365, esta plataforma tem como principais pontos fortes a ótima apresentação da plataforma e a variedade de vulnerabilidades que podem ser exploradas devido aos ataques serem executados em *sandboxes*. Tem ainda características positivas como a liberdade na criação dos *fortress* de cada equipa e da defesa dos mesmos, bem como a situação de ataque ao *fortress* de outras equipas. Como pontos fracos apresentam -se características como a falta de informação acerca das várias vulnerabilidades a serem exploradas, a dificuldade que a plataforma apresenta a utilizadores com pouco conhecimento na área da programação e segurança e o método de verificação da vulnerabilidade que exige verificação por parte da gestão da plataforma.

Ao analisar a plataforma Reversing.kr verificou -se características positivas como a variedade de desafios disponíveis, o método de obtenção automática de uma *flag* após chegar a uma solução correta, e o uso das *flags* para confirmação que a resposta encontrada está correta e obtenção de pontos. Como características negativas verificou -se que apesar de haver uma boa variedade de desafios disponíveis, todos eles estão relacionados com vulnerabilidades de aplicações, sendo que encontram -se estão ainda limitadas a diversos tipos de sistemas operativos e linguagens de programação o que faz com que seja necessário executar essas aplicações nos sistemas correspondentes. Existe ainda a falta de informação acerca dos vários desafios disponíveis.

O RingZer0 Team Online CTF é uma plataforma que apresenta alguns aspetos positivos como uma vasta variedade de desafios disponíveis, o uso de *flags* para validação e obtenção de pontos por desafio realizado com sucesso, a disponibilização de ferramentas adicionais que podem ser usadas na resolução de certos desafios e a facilidade de uso da plataforma. Como aspetos negativos verificou -se a falta de informação acerca dos desafios disponíveis e vulnerabilidades a explorar e a necessidade de usar SSH para certas máquinas de forma a poderem ser realizados determinados desafios.

O WebGoat apesar de funcionar como um servidor local, mostra ser uma plataforma bastante completa que apresenta características positivas como a variedade de ataques disponíveis, constituídos por informação acerca dos mesmos, desafios de treino e algumas competições. Como ponto positivo existe ainda o uso de *flags* para a validação e obtenção de pontos para o *ranking*.

Esta plataforma apresenta ainda como pontos negativos a característica de esta não ser uma plataforma online e de ter muitas poucas competições disponíveis.

No geral, todas as plataformas analisadas têm alguns pontos que a plataforma web a desenvolver tem como objetivos, no entanto nenhuma delas tem um sistema de *Quizzes* ou de Propostas de ataques feitas pelos utilizadores e sendo que a única que fornece informação teórica acerca das várias vulnerabilidades a serem exploradas é a plataforma WebGoat que apresenta ser a mais completa para um ambiente de aprendizagem de todas as exploradas. Apesar da variedade entre as várias plataformas analisadas, todas elas têm funcionalidades a ter em conta no desenvolvimento da plataforma web LabSecurity.

Seguidamente apresenta -se a metodologia escolhida e que se pretende usar no processo de desenvolvimento da plataforma web.

# Metodologia

De forma a minimizar os riscos de erros e falhas no decorrer do desenvolvimento de uma plataforma web, é aconselhado usar metodologias de desenvolvimento de software. Apesar de existir uma grande variedade de metodologias, estas encontram -se divididas em dois tipos: metodologias tradicionais e metodologias ágeis, tendo como tradicionais a metodologia em espiral, RUP, entre outras, e como metodologias ágeis o SCRUM e o eXtreme Programming (XP).

Para o desenvolvimento deste projeto foi definida como metodologia de desenvolvimento de software o SCRUM devido a ser uma metodologia focada em desenvolvimento rápido e iterativo de software. Além isso, o SCRUM é fácil de usar e cria a oportunidade de recolher de forma mais flexível os requisitos necessários para o desenvolvimento da plataforma web e permitir que esse processo se desperte ainda numa fase inicial do desenvolvimento do projeto.

## Metodologia de desenvolvimento ágil: SCRUM

O SCRUM [6] é uma estrutura da metodologia de desenvolvimento de software ágil conhecida por ser focada em produzir e entregar produtos de grande qualidade. É um *framework* simples de entender, apesar de ser complexo de colocar em prática, e é usado em trabalhos complexos onde não se consegue prever futuras dificuldades.

O SCRUM é constituído por um conjunto de cargos, eventos e regras interligados que juntos permitem abordar problemas complexos e desenvolver produto de qualidade ao mesmo tempo. Para lá disso, permite ainda saber se as técnicas e método de manutenção do projeto são eficazes, fazendo com que seja possível fazer melhorias continuas ao nível do produto a entregar, ao nível da equipa que se encontra a trabalhar no projeto e no ambiente de trabalho.

O Scrum é baseado no controlo empírico de processos e usa um método iterativo e incremental de forma a tentar aumentar a previsão de erros e controlo de riscos.

Em todas as implementações de processos de controlo empírico existem três pilares essenciais a seguir: Transparência, Inspeção e Adaptação.

A Transparência consiste num conjunto de aspetos relevantes no processo de controlo que devem ser visíveis aos responsáveis pelos resultados obtidos.

Este conjunto de aspetos devem ser definidos de forma a que os participantes partilhem um mesmo conhecimento do que lhes é apresentado. Um exemplo desses aspetos é saber quando se pode dizer que determinada funcionalidade é dada como terminada.

A Inspeção consiste em análises frequentes ao trabalho que se está a desenvolver de forma a verificar se não existem variações do que é pedido e do que se está a desenvolver, e são normalmente efetuadas por alguém com experiência na execução dessas análises

A Adaptação consiste na reorganização do projeto caso se detetem desvios nos objetivos do mesmo, sendo que esta deve ser realizada o mais breve possível de forma a minimizar os riscos causados.

O Scrum é constituído por quatro fases que definem o decorrer de uma Sprint, sendo estas, por ordem de realização, *Sprint Planning*, *Daily Scrum*, *Sprint Review* e *Sprint Retrospective*.

Uma *Sprint* consiste numa fase do projeto que é iniciada e que tem um espaço de tempo já definido para a conclusão da mesma, não podendo este exceder um mês. Uma nova *Sprint* só começa após a finalização da *Sprint* que se encontra a decorrer.

O uso de *sprints* pelo Scrum é bastante vantajoso já que estas permitem fazer uma previsão do estado do projeto através da inspeção e adaptação pelo menos a cada mês.

Uma *sprint* inicia -se por uma *Sprint Planning* que consiste numa reunião em que participam o proprietário do produto, o gestor do projeto e a equipa de desenvolvimento, em que é decidido o que será entregue no final da sprint e como é que o trabalho realizado vai ser gerido. Nesta reunião é analisado um documento, designado de *Product Backlog*, que contém todas as funcionalidades que se pretende que o produto tenha no final do seu desenvolvimento, e no final da mesma deve resultar um documento, designado de *Sprint Backlog*, que consiste numa lista de todas as funcionalidades a que a equipa de desenvolvimento, juntamente com o gestor do projeto se comprometem a entregar no final da *sprint*.

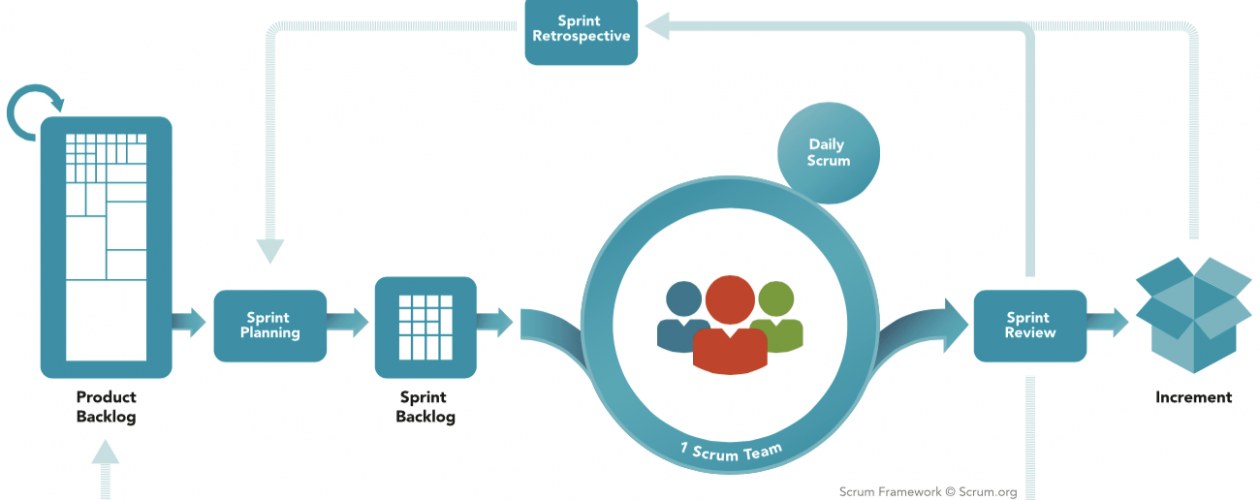
Ao longo do desenvolvimento do produto são realizadas várias *Daily Scrum*, que consistem em reuniões diárias onde se verifica que trabalho foi realizado desde a última *Daily Scrum* e tentar prever que trabalho será feito até à próxima, ajudando a perceber qual a probabilidade da equipa de desenvolvimento alcançar o objetivo da *sprint*, e até mesmo aumentar a mesma através de ajustes a vários níveis.

No Final da *sprint* é feita a *Sprint Review* que consiste numa reunião onde participa o proprietário do produto, o gestor de projeto e equipa de desenvolvimento, onde é revisto o trabalho realizado durante a *sprint* e apontadas informações valiosas abordadas durante a reunião.

Entre a *Sprint Review* e a nova *Sprint Planning*, realiza -se a *Sprint Retrospective*, que consiste numa reunião entre o gestor do projeto e a equipa de desenvolvimento com o objetivo de melhorar o rendimento da próxima *sprint* tendo em conta a última.

Todo este processo pode ser observado na Figura 6.

Figura 6- Processo de uma Sprint [6]



# Análise de Requisitos

## Atores e respetivos casos de uso

A plataforma terá como atores que usam a mesma:

* **Gestor da Plataforma –** Pessoa(as) responsável por gerir a plataforma.
* **Aluno –** Utilizadores que podem participar nas várias componentes que a plataforma disponibiliza, desde as competições, aos quizzes, entre outras.

De forma a que a plataforma funcione corretamente, existem funcionalidades que são específicas para o gestor da plataforma, sendo que todas as outras comuns para os alunos e para o gestor, sendo estas apresentadas na Tabela 2.

Tabela 2- Atores e respetivos casos de uso

|  |  |
| --- | --- |
| **Atores** | **Casos de Uso** |
| Gestor da Plataforma | * Criar, Editar, Eliminar Competição * Criar, Editar, Eliminar Desafio * Inserir Desafio para competição * Criar, Editar, Eliminar Pergunta * Criar, Editar, Eliminar Quizz * Inserir pergunta para Quizz * Ver Sugestão * Aceitar Sugestão * Rejeitar Sugestão |
| Aluno | * Criar, Editar, Eliminar Conta * Ver, Realizar Competição * Registar ou Desistir de competição * Ver Scoreboard * Ver, Realizar Desafio * Ver, Realizar Quizz * Ver, Realizar Pergunta * Criar Sugestão |

## Diagrama de casos de uso

Seguidamente pode -se observar o diagrama de casos de uso, onde se encontram representados os atores que interagem com a plataforma e os seus respetivos casos de uso generalizados. Todos os casos de uso que irão ser desenvolvidos no decorrer no projeto encontram -se dentro da fronteira.

Uma imagem com texto, mapa

Descrição gerada com confiança muito alta

# Bibliografia

|  |  |
| --- | --- |
| [1] | “Hacking-Lab,” [Online]. Available: https://www.hacking-lab.com/index.html. [Acedido em 12 Junho 2018]. |
| [2] | “CTF365,” [Online]. Available: https://ctf365.com. [Acedido em 12 Junho 2018]. |
| [3] | “Reversing.kr,” [Online]. Available: http://reversing.kr/. [Acedido em 13 Junho 2018]. |
| [4] | “RingZer0 Team Online CTF,” [Online]. Available: https://ringzer0team.com. [Acedido em 13 Junho 2018]. |
| [5] | “OWASP WebGoat Project,” [Online]. Available: https://www.owasp.org/index.php/Category:OWASP\_WebGoat\_Project. [Acedido em 13 Junho 2018]. |
| [6] | “SCRUM,” [Online]. Available: https://www.scrum.org/resources/what-is-scrum. [Acedido em 17 Junho 2018]. |