

Universidade da Beira Interior

Departamento de Informática



Departamento de
Informática

C-Team: *Challenge Accepted*

Elaborado por:

38950 — Diogo José Real Lavareda
39392 — Joana Elias Almeida
41266 — Diogo Castanheira Simões
41358 — Beatriz Tavares da Costa
41381 — Igor Cordeiro Bordalo Nunes

Orientador:

Professor Doutor Pedro Ricardo Morais Inácio

22 de maio de 2021

Conteúdo

Conteúdo	i
Lista de Figuras	ii
Lista de Tabelas	iii
Lista de Excertos de Código	iv
1 Introdução	1
1.1 Descrição da proposta	1
1.2 Constituição do grupo	1
1.3 Organização do Documento	1
2 Engenharia de Software	3
2.1 Introdução	3
2.2 Ferramentas e tecnologias utilizadas	3
2.3 Requisitos	4
2.3.1 Requisitos funcionais	4
2.3.2 Requisitos não-funcionais	5
2.4 Casos de Uso	5
2.5 Diagrama do sistema	6
2.6 Conclusões	6
Bibliografia	7

Lista de Figuras

2.1	Diagrama de casos de uso: processo de registo e <i>Login</i>	5
2.2	Diagrama de casos de uso: <i>homepage</i>	5
2.3	Diagrama de casos de uso: processo de propor desafio e cifra de mensagem	5
2.4	Diagrama de casos de uso: processo de responder a desafios . . .	5
2.5	Diagrama de casos de uso: processo de propor desafios do tipo valor de hash	6
2.6	Diagrama da arquitetura do sistema	6

Lista de Tabelas

1.1	Constituição da equipa <i>C-Team</i>	1
2.1	Ferramentas utilizadas	4

Lista de Excertos de Código

Acrónimos

RSA *Rivest-Shamir-Adleman*

Capítulo 1

Introdução

1.1 Descrição da proposta

A plataforma *Challenge Accepted* tem como **objetivo** permitir aos seus utilizadores publicar e resolver vários desafios de criptografia.

1.2 Constituição do grupo

O presente projeto foi realizado pela equipa *C-Team*, constituída pelos elementos listados na Tabela 1.1.

Nº	Nome	Alcunha
38950	Diogo José Real Lavareda	<i>Lavareda</i>
39392	Joana Elias Almeida	<i>Joaninha</i>
41266	Diogo Castanheira Simões	<i>Ash</i>
41358	Beatriz Tavares da Costa	<i>Bea</i>
41381	Igor Cordeiro Bordalo Nunes	<i>Etileno</i>

Tabela 1.1: Constituição da equipa *C-Team*.

1.3 Organização do Documento

De modo a refletir o projeto realizado, este relatório encontra-se estruturado em cinco capítulos, nomeadamente:

1. No primeiro capítulo — **Introdução** — são apresentados o projeto, os seus objetivos, a equipa desenvolvedora e a respetiva organização do relatório.

2. No segundo capítulo — **Engenharia de Software** — são elaborados os diagramas de casos de uso da aplicação que orientam a respetiva implementação.
3. No terceiro capítulo — **Implementação** — são descritas as escolhas e os detalhes de implementação da aplicação, bem como as tecnologias utilizadas durante o seu desenvolvimento.
4. No quarto capítulo — **Reflexão Crítica e Problemas Encontrados** — são indicados os objetivos alcançados, quais as tarefas realizadas por cada membro do grupo, assim como são expostos os problemas enfrentados e é feita uma reflexão crítica sobre o trabalho.
5. No quinto capítulo — **Conclusões e Trabalho Futuro** — são analisados os conhecimentos adquiridos ao longo do desenvolvimento do projeto e, em contrapartida, o que não se conseguiu alcançar e que poderá ser explorado futuramente.

Capítulo 2

Engenharia de Software

2.1 Introdução

Um projeto de sucesso envolve uma preparação prévia exímia. A plataforma *Challenge Accepted* desenvolvida pela *C-Team* não foi exceção. Foram delineados antecipadamente vários pontos fulcrais, nomeadamente:

- Ferramentas e tecnologias (Secção 2.2): perante uma equipa de 5 pessoas, foi vital determinar quais as tecnologias a utilizar não só para implementar a aplicação mas também para gerir o trabalho paralelo que iria decorrer;
- Requisitos funcionais e não-funcionais (Secção 2.3): a aplicação deve cumprir uma série de requisitos a fim de poder alcançar os objetivos propostos;
- Diagramas de casos de uso (Secção 2.4): a fim de se perceber as atividades a desenhar e o respetivo código-fonte que as interliga, diferentes casos de uso foram estudados;
- Diagrama de atividades (Secção 2.4): este diagrama resume o fluxo de funcionamento da aplicação.

2.2 Ferramentas e tecnologias utilizadas

As ferramentas utilizadas no âmbito da realização do projeto, sumariadas na Tabela 2.1, visam três componentes essenciais na sua gestão: 1) linguagem de programação, 2) servidor, 3) administração bd, 4) relatório, 5) controlo de versões.

<i>Software</i>	<i>Versão</i>
Linguagem de Programação	
<i>Python</i>	3.8.0
Servidor	
<i>Microsoft Windows Server</i>	2019
<i>MariaDB</i>	10.4.18
Administração BD	
<i>DBeaver</i>	21.0.5
Relatório	
<i>Overleaf</i>	
Controlo de versões	
<i>git</i>	2.17.1
<i>GitKraken</i>	7.6.1

Tabela 2.1: Ferramentas e tecnologias utilizadas, organizadas por categoria.

2.3 Requisitos

De forma a ir de encontro aos objetivos propostos do projeto (Secção 1.1), uma série de requisitos funcionais e não-funcionais foi delineada.

2.3.1 Requisitos funcionais

A plataforma deve:

1. Possuir um ecrã de boas-vindas e instruções de uso do sistema;
2. Ter um ecrã de registo e *login* de utilizadores;
3. Ter uma *homepage* com acesso direto às seguintes funcionalidades:
 - (a) resolução de desafios disponíveis,
 - (b) propor desafios,
 - (c) pedir ajuda,
 - (d) *scoreboard* dos utilizadores,
4. Permitir responder a desafios;
5. Permitir submeter dois tipos de desafios, em que um é de decifra de mensagem e outro de descoberta de mensagem.

2.3.2 Requisitos não-funcionais

A plataforma deve:

1. Permitir apenas uma tentativa a cada 15 segundos para os desafios cifrados;
2. Usar assinaturas digitais *Rivest-Shamir-Adleman* (RSA);
3. Ter uma interface gráfica minimalista e (*user-friendly*);
4. Ser segura em termos do armazenamento dos dados na base de dados (se possível cifrar os dados guardados duplamente no caso dos desafios).

2.4 Casos de Uso

Para a plataforma *CHALLENGE-ACCEPTED*, foram identificados os seguintes casos de uso:

1. Processo de Registo e *Login* (Figura 2.1);
2. *Homepage* (Figura 2.2);
3. Processo de propor desafio e cifra de mensagem (Figura 2.3);
4. Processo de responder a desafios (Figura 2.4);
5. Processo de propor desafios do tipo valor de hash (Figura 2.5);

Os diagramas foram elaborados com recurso ao serviço *Visual Paradigm*.

Figura 2.1: Diagrama de casos de uso: processo de registo e *Login*

Figura 2.2: Diagrama de casos de uso: *homepage*.

Figura 2.3: Diagrama de casos de uso: processo de propor desafio e cifra de mensagem.

Figura 2.4: Diagrama de casos de uso: processo de responder a desafios.

Figura 2.5: Diagrama de casos de uso: processo de propor desafios do tipo valor de hash.

Figura 2.6: Diagrama da arquitetura do sistema

2.5 Diagrama do sistema

2.6 Conclusões

Na posse de um plano delineado segundo as práticas comuns da área da Engenharia de *Software*, segue-se a fase de implementação, a qual deve seguir escrupulosamente os requisitos determinados e tem por base os casos de uso estudados. Por fim, o fluxo da aplicação deverá seguir o diagrama de atividades obtido por esta fase de estudo.

Bibliografia