



**Politécnico
Castelo Branco**

Escola Superior
de Tecnologia

EncryVault

Licenciatura em Informática e Multimédia - Projeto II

Miguel Fernandes Barata Gonçalves

Aluno nº 20170996

Orientador

Professor Doutor Alexandre José Pereira Duro da Fonte

Trabalho de Projeto apresentado à Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco para cumprimento dos requisitos necessários à obtenção do grau de Licenciado em Informática e Multimédia, realizada sob a orientação científica do Professor Doutor Alexandre José Pereira Duro da Fonte, do Instituto Politécnico de Castelo Branco.

Junho 2026

Composição do júri

Presidente do júri

Professor Doutor, Fernando Sérgio Rodrigues de Brito da Mota Barbosa

Professor Adjunto, Escola Superior de Tecnologia do Instituto Politécnico
de Castelo Branco

Vogais

Professor Doutor, Alexandre José Pereira Duro da Fonte

Professor Adjunto, Escola Superior de Tecnologia do Instituto Politécnico
de Castelo Branco

Professor Doutor, Osvaldo Arede dos Santos

Professor Adjunto, Escola Superior de Tecnologia do Instituto Politécnico
de Castelo Branco

Dedicatória

Dedico este trabalho à minha família, por ter sido a minha base em todas as fases do curso, com apoio, paciência e compreensão, sobretudo nos momentos em que tudo parecia mais pesado e difícil de conciliar.

Dedico também aos meus amigos, pela força, pelas conversas e pelos incentivos na altura certa, que me ajudaram a manter a motivação e a não desistir quando o cansaço e a pressão apertaram.

Por fim, dedico este projeto a todos os que, de forma direta ou indireta, contribuíram para que eu chegasse até aqui, porque este resultado é feito de esforço, mas também do apoio de quem esteve ao meu lado.

Agradecimentos

Agradeço ao meu orientador pela disponibilidade, acompanhamento e sentido crítico ao longo de todo o projeto, pela forma como foi apontando caminhos quando eu estava preso em dúvidas, e por me ter ajudado a transformar ideias soltas num trabalho mais sólido, mais coerente e mais bem estruturado.

Agradeço aos professores que fizeram parte do meu percurso acadêmico, porque cada aula, cada avaliação, cada correção e cada exigência contribuíram para eu ganhar bases, disciplina e responsabilidade, e porque foi esse conjunto de aprendizagens que me permitiu chegar a esta fase com mais maturidade e mais capacidade de enfrentar desafios reais.

Agradeço aos meus colegas e amigos pela entreajuda e pelo apoio nos momentos em que o cansaço apertou, pelas conversas que me voltaram a dar motivação, e pelas trocas de ideias que tantas vezes desbloquearam problemas que eu já estava a ver como impossíveis.

Agradeço também a todos os que, de forma direta ou indireta, estiveram presentes durante este processo, porque mesmo quando não se apercebem, pequenas palavras, pequenos gestos e pequenas ajudas acabam por fazer diferença quando a pressão e o desgaste começam a pesar.

Agradeço aos funcionários e a todos os profissionais da instituição pela disponibilidade e simpatia ao longo destes anos, porque o dia a dia é feito de detalhes e ter um ambiente mais humano e acessível torna o percurso mais leve e mais possível.

Por fim, agradeço à minha família por ser a minha base, por acreditar em mim nos dias em que eu próprio duvidei, por respeitar o meu tempo e o meu esforço, e por me dar a força necessária para continuar até ao fim com foco e determinação.

Resumo

Este projeto tem como objetivo concluir o desenvolvimento da aplicação móvel offline EncryVault para gestão segura de credenciais e informação sensível, no âmbito da unidade curricular Projeto II da Licenciatura em Informática e Multimédia da Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco, dando continuidade ao trabalho iniciado no Projeto I.

A motivação prende-se com a necessidade de proteger dados críticos do utilizador de forma simples e prática, reduzindo riscos como reutilização de palavras-passe, armazenamento inseguro e exposição de informação, garantindo privacidade e controlo local dos dados.

Neste relatório apresenta-se a evolução do EncryVault entre 16 de março e 18 de junho de 2026, desde a versão demonstrativa do Projeto I até à versão final, incluindo o novo formato V3 do ficheiro do cofre com conteúdo dividido em blocos cifrados, o suporte a documentos sigilosos, os reforços de segurança implementados (política de palavra-passe mestra, penalizações por tentativas falhadas, pepper com biometria, PINs de proteção, cofre falso, auto-destruição, proteção de ecrã e modo privado), a área de saúde das palavras-passe, a lixeira com retenção configurável, o Autofill no Android, os temas claro/escuro e os dois modos visuais, terminando com a validação através de testes automatizados e manuais e com uma síntese do trabalho realizado.

Palavras chave

EncryVault, Passwords, Segurança, Cibersegurança, Privacidade.

Abstract

This project aims to complete the development of EncryVault, an offline mobile application for the secure management of credentials and sensitive information, within the curricular unit Project II of the degree in Informatics and Multimedia at the School of Technology of the Polytechnic Institute of Castelo Branco, continuing the work started in Project I.

The main motivation is to protect critical user data in a simple and practical way, reducing risks such as password reuse, insecure storage, and information exposure, while ensuring privacy and full local control of the data.

This report presents the evolution of EncryVault between March 16 and June 18, 2026, from the demonstrative version of Project I to the final version, including the new V3 vault file format with encrypted chunks, support for confidential documents, the implemented security reinforcements (master password policy, failed attempt penalties, pepper with biometrics, protection PINs, decoy vault, self-destruct, screen protection and private mode), the password health area, the trash with configurable retention, Android Autofill, light/dark themes and two visual modes, ending with validation through automated and manual tests and a summary of the work carried out.

Keywords

EncryVault, Passwords, Cybersecurity, Encryption, Privacy.

Índice geral

1. Introdução	1
1.1 Motivação.....	1
1.2 Objetivos	2
1.3 Planeamento e cronograma.....	2
1.4 Estrutura do Documento.....	3
2. Contexto e análise de soluções existentes.....	5
2.1 Enquadramento.....	5
2.2 Funcionalidades dos gestores de credenciais	5
2.3 Critérios de comparação	6
2.4 Soluções existentes	7
2.5 Comparação aprofundada: EncryVault e KeePass	9
3. EncryVault e gestão segura de credenciais	11
3.1 O que é o EncryVault.....	11
3.2 Princípios de conceção	12
3.3 Visão geral da aplicação final	13
4. Investigação	33
4.1 Boas práticas de segurança aplicáveis	33
4.1.1 Modelo de ameaça.....	34
4.1.2 Medidas de mitigação no EncryVault.....	34
4.2 Síntese da investigação e implicações para o EncryVault.....	35
5. Requisitos e arquitetura do sistema	37
5.1 Requisitos funcionais.....	37
5.1.1 Requisitos base consolidados do Projeto I.....	37
5.1.2 Requisitos implementados no Projeto II	37
5.1.3 Requisitos adaptados face ao planeado.....	39
5.1.4 Funcionalidades futuras opcionais.....	40
5.2 Requisitos não funcionais.....	40
5.2.1 Requisitos assegurados desde o Projeto I.....	40
5.2.2 Requisitos reforçados no Projeto II	41
5.3 Arquitetura proposta	41
5.4 Estrutura geral do cofre e das entradas.....	42
5.5 Fluxo geral de acesso e utilização do cofre.....	44
5.6 Formato V3 do ficheiro .vltx.....	46

6. Design	47
6.1 Ecrãs da aplicação	47
6.2 Elementos de interface	48
7. Desenvolvimento	50
7.1 Metodologia de desenvolvimento	50
7.2 Ferramentas utilizadas	50
7.3 Linguagens e tecnologias de desenvolvimento.....	51
7.4 Recursos	52
7.5 Modulação e funcionamento do EncryVault.....	53
7.5.1 Casos de uso principais.....	53
7.5.2 Proteção do cofre e fluxo de encriptação	54
7.5.3 Gestão de sessão e <i>auto-lock</i>	59
7.5.4 Gestão de entradas, pesquisa e organização	60
7.5.5 Importação e exportação do cofre.....	61
7.5.6 Documentos sigilosos	61
7.5.7 Proteções avançadas	61
7.5.8 <i>Autofill</i> no Android	62
7.6 Versão demonstrativa à versão final.....	62
7.7 Dificuldades e soluções adotadas	62
8. Testes e validação.....	64
8.1 Abordagem de validação	64
8.2 Contexto técnico relevante para os testes.....	64
8.3 Testes funcionais (casos principais)	64
8.4 Testes de sessão e <i>auto-lock</i>	66
8.5 Testes de Importação e Exportação	66
8.6 Testes de robustez e integridade do ficheiro.....	67
8.7 Testes das funcionalidades do Projeto II.....	67
8.8 Critérios de aceitação	70
9. Conclusão e trabalho futuro.....	71
Bibliografia.....	72
Anexo A - Especificação do formato V3 do ficheiro .vltx	74
A.1 Cabeçalho.....	74
A.2 Blocos de documentos	74
A.3 Manifesto	74
A.4 Rodapé	75

Anexo B - Mapa dos testes automatizados	76
B.1 Criptografia e formato do cofre	76
B.2 Palavra-passe mestra e saúde	76
B.3 <i>Pepper</i>	76
B.4 PINs de proteção	76
B.5 Cofre falso e auto-destruição	76
B.6 Biometria e ações sensíveis	76
B.7 Proteção de ecrã e <i>clipboard</i>	76
B.8 <i>Autofill</i> e preenchimento	77
B.9 Entradas, lixeira e interface	77
Anexo C - Manifesto Android e modelo de permissões	78
C.1 Permissões	78
C.2 Cópias de segurança do sistema	78
C.3 Componentes nativos	78

Índice de figuras

Figura 1 - Diagrama de Gantt parte do planeamento do Projeto II.	3
Figura 2 - Exemplo de gerador de palavras-passe numa solução existente. Fonte: Bitwarden Documentation.	6
Figura 3 - Ecrã de arranque (<i>splash screen</i>) do EncryVault.	13
Figura 4 - Ecrã de termos e condições do EncryVault.	14
Figura 5 - Ecrã inicial com opções de criar ou configurar o cofre.	14
Figura 6 - Importação do cofre e escolha de tema.	15
Figura 7 - Criação de cofre e definição da palavra-passe mestra.	15
Figura 8 - Desbloqueio do cofre através da palavra-passe mestra.	16
Figura 9 - Página principal do cofre vazia.	16
Figura 10 - Criação de entrada no EncryVault.	17
Figura 11 - Página principal do cofre com lista de entradas.	17
Figura 12 - Pesquisa e filtragem de entradas no cofre.	18
Figura 13 - Visualização do detalhe de uma entrada guardada no cofre.	18
Figura 14 - Edição de entradas no EncryVault.	19
Figura 15 - Documentos sigilosos guardados no cofre.	20
Figura 16 - Página de configurações e diversas opções no cofre.	20
Figura 17 - Menu de segurança e proteções sensíveis.	21
Figura 18 - Autenticação biométrica adicional ao entrar no cofre.	22
Figura 19 - Proteção contra força bruta (<i>pepper</i>)	22
Figura 20 - Modo privado, ocultação de dados sensíveis na interface.	23
Figura 21 - Configuração do cofre falso (<i>decoy</i>).	23
Figura 22 - Preenchimento automático (<i>autofill</i>) e gestão do <i>clipboard</i>	24
Figura 23 - PINs de proteção (entradas, documentos e lixeira).	25
Figura 24 - PINs de proteção das entradas.	25
Figura 25 - PINs de proteção dos documentos.	26
Figura 26 - PINs de proteção da lixeira.	26
Figura 27 - Auto-destruição do cofre (Zona de Risco).	27
Figura 28 - Lixeira do cofre (entradas e documentos).	28
Figura 29 - Exportação e importação do cofre.	28
Figura 30 - Links rápidos das entradas.	29
Figura 31 - Aparência: temas e modos visuais (Moderno e Clássico).	30
Figura 32 - Sessão e bloqueio automático (<i>auto-lock</i>).	30
Figura 33 - Saúde das palavras-passe: alertas e recomendações.	31
Figura 34 - Página de Ajuda/FAQ.	32
Figura 35 - Ecrã "Sobre EncryVault".	32
Figura 36 - Diagrama de componentes do EncryVault (organização por camadas e serviços).	42
Figura 37 - Diagrama de classes do modelo do EncryVault (Cofre, Entradas e entidades relacionadas).	43
Figura 38 - DFD e fluxo de dados do EncryVault - Nível 0.	44
Figura 39 - DFD e fluxo de dados do EncryVault - Nível 1.	44

Figura 40 - Diagrama de atividade do fluxo de arranque e desbloqueio do cofre.	45
Figura 41 - Ícone do EncryVault.	49
Figura 42 - Imagem de arranque do EncryVault.	49
Figura 43 - Diagrama de casos de uso do EncryVault.	54
Figura 44 - Diagrama de sequência do desbloqueio do cofre (<i>Unlock</i>).	55
Figura 45 - Diagrama de sequência da criação do cofre (<i>Create Vault</i>).	56
Figura 46 - Diagrama de atividade do fluxo de encriptação e desencriptação do cofre.	57
Figura 47 - Fluxograma de alto nível do processo de encriptação/desencriptação.	58
Figura 48 - Diagrama de estados do cofre e da sessão (<i>Locked/Unlocked/Auto-lock</i>).	59
Figura 49 - Diagrama de sequência da adição de uma entrada ao cofre.	60

Lista de tabelas

Tabela 1 - Comparação de soluções analisadas e implicações para o EncryVault.	8
Tabela 2 - Comparação aprofundada entre KeePass e EncryVault.	9

Lista de abreviaturas, siglas e acrónimos

.vltx - Extensão do ficheiro do cofre do EncryVault

AAD - Additional Authenticated Data (dados adicionais autenticados na cifragem)

AEAD - Authenticated Encryption with Associated Data (encriptação autenticada com dados associados)

AES - Advanced Encryption Standard (algoritmo de cifra simétrica)

AES-256 - Advanced Encryption Standard de 256 bits (algoritmo AES com chave de 256 bits)

AES-KDF - AES Key Derivation Function (função de derivação de chave baseada em AES)

API - Application Programming Interface (interface de programação de aplicações)

Argon2 - Família de funções de derivação de chave resistente a ataques por hardware especializado

Argon2d - Variante da função Argon2 orientada para resistência a ataques por GPU

Argon2i - Variante da função Argon2 orientada para resistência a ataques por canais laterais

Argon2id - Função de derivação de chave (KDF) resistente a ataques por GPU/ASIC, variante híbrida

arm64-v8a - Arquitetura Android de 64 bits para processadores ARM

armeabi-v7a - Arquitetura Android de 32 bits para processadores ARM

ASIC - Application-Specific Integrated Circuit (circuito integrado específico)

Auto-lock - Bloqueio automático da sessão após inatividade

Auto-Type - Funcionalidade de preenchimento automático utilizada no KeePass em ambiente desktop

Autofill - Serviço Android de preenchimento automático de credenciais

Base64 - Formato de codificação textual de dados binários

BIND_AUTOFILL_SERVICE - Permissão Android necessária para declarar um serviço de preenchimento automático

CBC - Cipher Block Chaining (modo de operação usado com AES)

ChaCha20 - Algoritmo de cifra simétrica de fluxo

CRUD - Create, Read, Update, Delete (criar, ler, atualizar, apagar)

DFD - Data Flow Diagram (Diagrama de Fluxo de Dados)

ESTCB - Escola Superior de Tecnologia de Castelo Branco

FAQ - Frequently Asked Questions (perguntas frequentes)

FLAG_SECURE - Proteção Android contra capturas de ecrã e visualização em ecrãs recentes sensíveis

GPU - Graphics Processing Unit (unidade de processamento gráfico)

HMAC - Hash-based Message Authentication Code (código de autenticação baseado em hash)

HMAC-SHA-256 - Código de autenticação HMAC baseado na função SHA-256

IETF - Internet Engineering Task Force (organização responsável por normas técnicas da Internet)

IPCB - Instituto Politécnico de Castelo Branco

JSON - JavaScript Object Notation (formato de dados usado na serialização do conteúdo do cofre)

KDBX - Formato de ficheiro/base de dados utilizado pelo KeePass

KDF - Key Derivation Function (função de derivação de chave)

MAC - Message Authentication Code (código de autenticação de mensagem)

MB - Megabyte (unidade de armazenamento de dados)

NIST - National Institute of Standards and Technology (instituto norte-americano de normas e tecnologia)

OWASP - Open Worldwide Application Security Project (projeto internacional dedicado à segurança de aplicações)

PC - Personal Computer (computador pessoal)

PDF - Portable Document Format (formato de documento portátil)

Pepper - Segredo adicional guardado no dispositivo e combinado na derivação da chave

PIN - Personal Identification Number (código numérico de proteção)

ProGuard - Ferramenta de otimização, redução e ofuscação de código Java/Android

R8 - Ferramenta Android de otimização, redução e ofuscação de código

RFC - Request for Comments (documento técnico usado em normas e especificações da Internet)

SDK - Software Development Kit (conjunto de ferramentas de desenvolvimento)

SHA-256 - Secure Hash Algorithm 256 bits (função de hash criptográfica)

TOTP - Time-based One-Time Password (palavra-passe temporária baseada no tempo)

UI - User Interface (Interface do Utilizador)

URL - Uniform Resource Locator (endereço de recurso na web)

USE_BIOMETRIC - Permissão Android para utilização de autenticação biométrica

V3 - Terceira versão do formato do ficheiro do cofre (.vltx)

XChaCha20-Poly1305 - Algoritmo AEAD usado para cifrar e autenticar o cofre

1. Introdução

No âmbito da unidade curricular de Projeto II da Licenciatura em Informática e Multimédia da Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco, foi dada continuidade ao desenvolvimento do EncryVault, uma aplicação móvel *offline* orientada para a proteção e gestão segura de credenciais e informação sensível, iniciada no Projeto I.

Atualmente, uma parte significativa dos utilizadores continua a guardar palavras-passe e dados críticos de forma insegura, recorrendo a notas, ficheiros não protegidos ou repetição de credenciais entre serviços, o que aumenta a probabilidade de comprometimento de contas e exposição de informação pessoal.

Face a esta realidade, o EncryVault surge como uma proposta centrada em privacidade e controlo local, permitindo ao utilizador armazenar os seus dados num cofre encriptado e aceder ao conteúdo de forma simples, rápida e organizada, sem dependência de serviços externos e reduzindo a superfície de exposição do sistema.

Nesta fase do Projeto II, o trabalho centrou-se na evolução da aplicação desde a versão demonstrativa apresentada no Projeto I até à versão final, decorrendo entre 16 de março e 18 de junho de 2026 e incluindo a conclusão das funcionalidades planeadas, a migração do cofre para um novo formato de ficheiro (V3) e a introdução de um conjunto alargado de reforços de segurança e melhorias de experiência de utilização.

O Projeto II foca-se assim na implementação completa do EncryVault, no redesenho do formato do cofre, no suporte a documentos sigilosos, em proteções avançadas como *pepper*, biometria, PINs, cofre falso e auto-destruição, no *Autofill* em Android e na validação da solução final com testes automatizados e manuais.

1.1 Motivação

A motivação para o desenvolvimento do EncryVault surge de um problema frequente em que, apesar de ser recomendado usar palavras-passe fortes e diferentes em cada serviço, muitos utilizadores continuam a reutilizar credenciais e a guardá-las em locais pouco seguros como notas, ficheiros sem proteção ou mensagens, aumentando o risco de comprometer contas e expor informação sensível em caso de perda do dispositivo ou fuga de dados de um serviço.

Além disso, várias soluções existentes dependem de conta, sincronização e serviços externos, o que pode levantar preocupações de privacidade e confiança para quem prefere manter os dados sob controlo direto, pelo que o EncryVault propõe uma alternativa centrada no funcionamento local com um cofre encriptado que não precisa de Internet para ser utilizado e que permite armazenar

credenciais de forma simples e mais segura, sendo que o Projeto II transformou essa proposta numa solução final completa, estável e validada.

1.2 Objetivos

Concluir a implementação dos requisitos funcionais e não funcionais definidos no Projeto I, evoluindo a versão demonstrativa do EncryVault para uma versão final completa e estável.

Redesenhar o formato do ficheiro do cofre (.vltx) para a versão V3, com conteúdo dividido em blocos (*chunks*) cifrados, *manifest* e *footer* autenticados, permitindo guardar documentos sigilosos de maior dimensão de forma eficiente e segura.

Reforçar a segurança da aplicação com política obrigatória de palavra-passe mestra, penalizações por tentativas falhadas, *pepper* com suporte biométrico, PINs de proteção, cofre falso, auto-destruição, proteção contra capturas de ecrã, modo privado e limpeza automática da área de transferência.

Validar a versão final com testes automatizados e manuais, garantindo robustez, integridade do cofre e qualidade de utilização, e preparar a apresentação e avaliação final do projeto.

1.3 Planeamento e cronograma

O trabalho do Projeto II decorreu entre 16 de março e 18 de junho de 2026 e foi organizado em iterações, começando pela consolidação da base desenvolvida no Projeto I, passando pelo redesenho do formato do cofre e pela implementação progressiva das funcionalidades e reforços de segurança, e terminando com a fase de testes, validação e preparação da entrega final.

A Figura 1 apresenta o cronograma do Projeto II, evidenciando a distribuição temporal das tarefas realizadas nesta fase, entre 16 de março e 18 de junho de 2026.

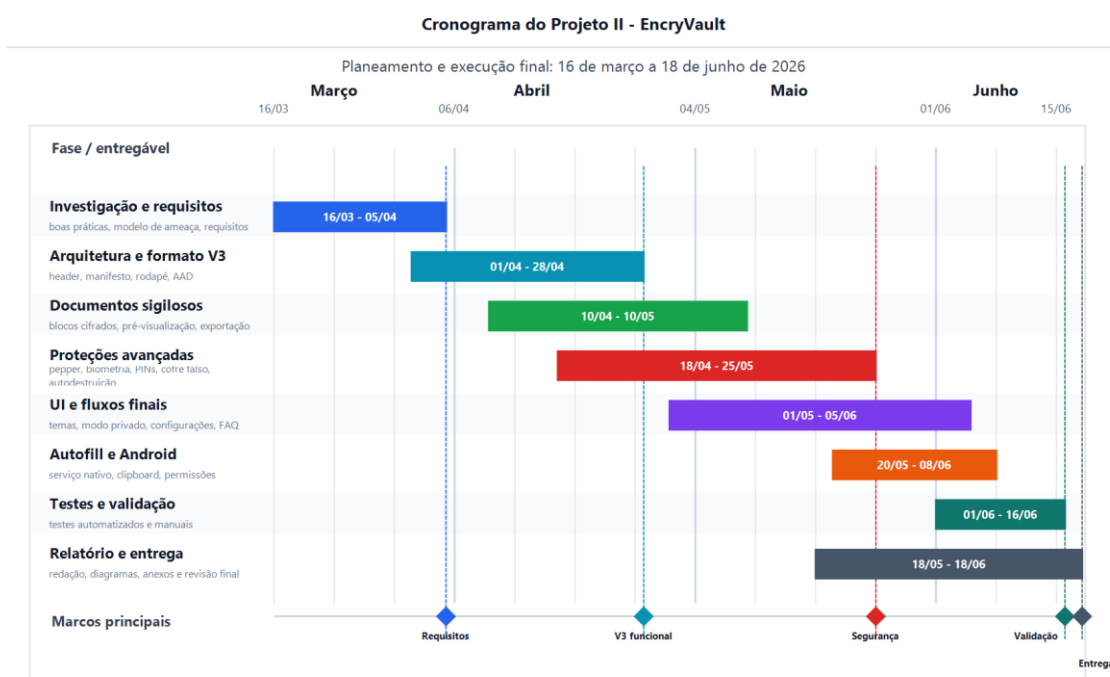


Figura 1 - Diagrama de Gantt parte do planeamento do Projeto II.

1.4 Estrutura do Documento

Este documento encontra-se organizado da seguinte forma:

O Capítulo 1 (Introdução) apresenta o enquadramento do problema, a motivação do projeto, os objetivos desta fase (Projeto II) e o planeamento/cronograma do trabalho.

O Capítulo 2 (Contexto e análise de soluções existentes) apresenta as funcionalidades típicas dos gestores de credenciais e analisa soluções existentes segundo critérios de comparação, incluindo uma comparação aprofundada com o KeePass, identificando pontos fortes e pontos fracos relevantes para o EncryVault.

O Capítulo 3 (EncryVault e gestão segura de credenciais) apresenta o que é o EncryVault, os princípios que orientaram a sua conceção e uma visão geral da aplicação na sua versão final, nas perspetivas do utilizador e do programador.

O Capítulo 4 (Investigação) reúne as boas práticas de segurança aplicáveis, incluindo o modelo de ameaça considerado e as medidas de mitigação efetivamente implementadas, e termina com a síntese da investigação e as suas implicações para o EncryVault.

O Capítulo 5 (Requisitos e arquitetura do sistema) define os requisitos funcionais e não funcionais e o respetivo estado de implementação no final do

Projeto II, descreve a arquitetura final e apresenta a estrutura do cofre, o fluxo de acesso e o novo formato V3 do ficheiro do cofre.

O Capítulo 6 (Design) apresenta as decisões principais de interface e navegação da versão final, incluindo os modos visuais Moderno e Clássico e o suporte a tema claro e escuro, com foco numa utilização simples, consistente e adequada a uma aplicação de cofre.

O Capítulo 7 (Desenvolvimento) descreve a metodologia adotada, as ferramentas, linguagens e tecnologias utilizadas, a modulação e funcionamento do EncryVault, a evolução da versão demonstrativa para a versão final e as principais dificuldades encontradas.

O Capítulo 8 (Testes e validação) apresenta a abordagem de validação, o contexto técnico relevante e os testes funcionais, de robustez e das funcionalidades do Projeto II, juntamente com os critérios de aceitação.

O Capítulo 9 (Conclusão e trabalho futuro) sintetiza o trabalho realizado no Projeto II e identifica as evoluções possíveis para o futuro do EncryVault.

O relatório ainda contém os Anexos que reúnem material de apoio: a especificação do formato V3 do ficheiro .vltx (Anexo A), o mapa dos testes automatizados (Anexo B) e o detalhe do manifesto Android e do modelo de permissões (Anexo C).

2. Contexto e análise de soluções existentes

Este capítulo apresenta o contexto atual no domínio dos gestores de credenciais: o enquadramento do problema, as funcionalidades típicas destas soluções, os critérios de comparação considerados e a análise das soluções existentes, incluindo uma comparação aprofundada com o KeePass. O objetivo é identificar padrões, pontos fortes e limitações das abordagens atuais, retirando implicações diretas para as decisões adotadas no EncryVault.

2.1 Enquadramento

Os gestores de credenciais (*password managers*) surgem como resposta à dificuldade do utilizador em memorizar e manter palavras-passe fortes e diferentes para múltiplos serviços. Na prática, estas soluções centralizam credenciais e outros dados sensíveis num “cofre” protegido por uma palavra-passe mestra (e, por vezes, por fatores adicionais), reduzindo a tendência para reutilização de palavras-passe e armazenamento inseguro.

Além da segurança, a componente de usabilidade é crítica: quanto mais simples for guardar, pesquisar e atualizar entradas, maior a probabilidade de adoção contínua. Assim, as soluções atuais combinam normalmente mecanismos de proteção (encriptação e controlo de sessão) com funcionalidades de organização e suporte ao utilizador (pesquisa, categorias/*tags*, geradores de palavras-passe e opções de *backup*) [14]-[19].

2.2 Funcionalidades dos gestores de credenciais

Conforme apresentado anteriormente, um gestor de credenciais é uma solução que permite guardar e consultar credenciais e outros dados sensíveis de forma organizada, reduzindo riscos comuns de armazenamento inseguro e de reutilização de palavras-passe [14]-[19].

As funcionalidades podem variar bastante consoante a solução, e cabe ao utilizador escolher aquela que melhor se adequa ao seu contexto de uso, sendo que de forma geral estes sistemas podem disponibilizar:

- Cofre encriptado para guardar credenciais e notas sensíveis.
- Criação, consulta, edição e remoção de entradas.
- Pesquisa e organização por categorias, etiquetas ou favoritos.
- Geração de palavras-passe fortes para novos registos.
- Cópia rápida de campos (utilizador, palavra-passe) para facilitar o uso.
- Bloqueio automático (*auto-lock*) e reautenticação após inatividade.
- Exportação e importação para *backup* e restauro do cofre.

A Figura 2 exemplifica a funcionalidade de geração de palavras-passe do Bitwarden, permitindo criar palavras-passe fortes com critérios configuráveis, sendo uma função comum em gestores de credenciais modernos [14], [15].

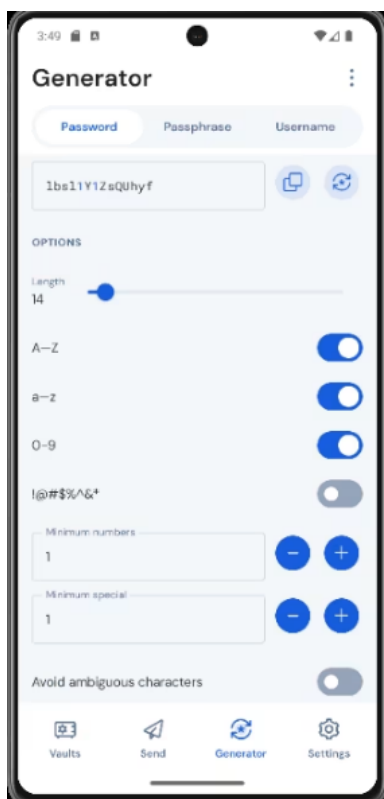


Figura 2 - Exemplo de gerador de palavras-passe numa solução existente. Fonte: Bitwarden Documentation.

No contexto do EncryVault, esta funcionalidade é aplicada com foco num modelo *offline*, mantendo o cofre local no dispositivo e garantindo que o utilizador mantém controlo direto sobre os seus dados, sendo que nos capítulos seguintes são detalhadas as boas práticas e as decisões técnicas que suportam esta abordagem [14]-[19].

2.3 Critérios de comparação

Para analisar as soluções existentes, consideraram-se critérios que afetam diretamente segurança, privacidade e experiência de utilização:

- Modelo de armazenamento: local/*offline* ou dependente de conta/serviço.
- Proteção do cofre: conceito de palavra-passe mestra, bloqueio automático e reautenticação.
- Organização e pesquisa: categorias, grupos, *tags* e rapidez de consulta.
- Suporte ao utilizador: geração de palavras-passe e, quando aplicável, preenchimento automático.

- Portabilidade: exportação/importação para *backup* e migração entre soluções.
- Controlo do utilizador: nível de dependência de serviços externos e gestão local dos dados.

2.4 Soluções existentes

Atualmente existem várias soluções consolidadas para gestão de credenciais com abordagens diferentes ao nível do armazenamento e funcionalidades. Entre as soluções analisadas destacam-se:

Bitwarden - disponibiliza um cofre encriptado, organização de entradas, geração de palavras-passe e funcionamento multiplataforma. Como referência, é relevante pela variedade de funcionalidades e por representar uma abordagem comum em gestores modernos [14], [15].

1Password - disponibiliza cofres encriptados e mecanismos de organização por “*vaults*”, destacando-se pela forma como complementa a proteção do acesso com fatores adicionais e uma abordagem orientada à segurança do utilizador [16].

KeePass - solução focada em armazenamento local, onde as credenciais são guardadas numa base de dados encriptada e acedidas através de uma chave mestra, sendo frequentemente utilizada em cenários *offline* [17].

LastPass - disponibiliza um cofre encriptado e funcionalidades comuns como organização, preenchimento automático (dependendo do contexto) e gerador de palavras-passe configurável, sendo uma solução bastante conhecida no domínio [18].

De forma geral, as soluções analisadas confirmam o conjunto de funcionalidades recorrentes apresentado na secção 2.2, com diferenças sobretudo no modelo de armazenamento e no grau de dependência de serviços externos [14]-[19].

Esta análise orienta a seleção dos requisitos e decisões do EncryVault, privilegiando um modelo *offline* e funcionalidades essenciais para um cofre local.

A Tabela 1 apresenta uma comparação entre Bitwarden, 1Password, KeePass e LastPass, analisando o modelo de utilização, os pontos fortes e os pontos fracos de cada solução. Esta comparação permite identificar diferenças entre abordagens mais dependentes de conta/ecossistema e alternativas orientadas para controlo local. Com base nesta análise, é também incluído o EncryVault, de forma a enquadrar as opções do projeto num modelo *offline* e autocontido, suportado por um cofre encriptado em ficheiro .vltx.

Tabela 1 - Comparação de soluções analisadas e implicações para o EncryVault.

Solução	Modelo / Foco	Pontos Fortes	Pontos Fracos
Bitwarden	Multiplataforma	• Conjunto funcional completo (cofre, organização, gerador) • Boa usabilidade no dia a dia • Referência moderna do “mercado”	• Pode implicar dependência de conta/serviço em cenários comuns • Controlo local absoluto nem sempre é o foco
1Password	Usabilidade + proteção da conta	• Organização por “<i>vaults</i>” bem definida • Forte integração de segurança na experiência • Fluxos maduros e consistentes	• Dependência de ecossistema/conta típica destas soluções • Menos alinhado com um objetivo estritamente <i>offline</i>
KeePass	<i>Offline</i> / armazenamento local	• Controlo local do cofre (ficheiro) • Bom encaixe para cenários <i>offline</i> • Portabilidade simples via ficheiro	• Experiência pode exigir mais configuração/gestão manual • Conveniências variam com integrações/extensões
LastPass	Conveniência / <i>features</i> “ <i>standard</i> ”	• Funcionalidades comuns bem cobertas (organização, gerador, etc.) • Orientado a facilidade de uso	• Tende a depender de conta/serviço para comodidades • Menos alinhado com um modelo <i>offline</i> autocontido
EncryVault (projeto)	<i>Offline</i> / controlo local	• 100% <i>offline</i>, sem dependência de serviços externos • Cofre local em ficheiro encriptado (.vltx, formato V3) • CRUD de entradas + pesquisa/filtros/ordenação + categorias e <i>tags</i> • Documentos sigilosos cifrados dentro do cofre • <i>Auto-lock</i>, biometria, <i>pepper</i>, PINs de proteção e bloqueio manual • Lixeira com retenção configurável e <i>Autofill</i> local em Android • Exportação/importação para <i>backup</i>	• Sem sincronização automática (por opção de modelo)

2.5 Comparação aprofundada: EncryVault e KeePass

Das soluções analisadas, o KeePass [25] é a mais diretamente comparável ao EncryVault, por ser igualmente um gestor de credenciais totalmente *offline*, baseado num cofre guardado num ficheiro local. A Tabela 2 compara as duas soluções em maior profundidade.

Tabela 2 - Comparação aprofundada entre KeePass e EncryVault.

Dimensão	KeePass	EncryVault
Modelo de funcionamento	<i>Offline</i> , cofre num ficheiro local.	<i>Offline</i> , cofre num ficheiro local.
Plataforma principal	<i>Desktop</i> (Windows); Android através de clientes de terceiros.	Android (<i>mobile-first</i>), com componente nativa integrada.
Formato do ficheiro	KDBX (formato aberto, suportado por múltiplas aplicações).	.vltx V3 (formato próprio com blocos, manifesto e rodapé).
Derivação de chave	AES-KDF ou Argon2 (KDBX 4).	Argon2id, com <i>pepper</i> opcional.
Cifra do conteúdo	AES-256 (predefinição) ou ChaCha20, com autenticação por HMAC.	XChaCha20-Poly1305 (cifragem autenticada, com AAD).
Documentos/anexos	Anexos binários associados às entradas.	Documentos em blocos cifrados, com limites, pré-visualização e lixeira própria.
Preenchimento automático	<i>Auto-Type</i> no <i>desktop</i> ; <i>autofill</i> no Android via clientes de terceiros.	Serviço de <i>autofill</i> nativo integrado.
Proteções para coação	Sem cofre falso ou auto-destruição de origem.	Cofre falso e auto-destruição nativos.
Maturidade e ecossistema	Projeto de longa data, código aberto, <i>plugins</i> e amplo escrutínio público.	Projeto académico recente, com base de código própria.

O KeePass tem vantagens claras decorrentes da sua maturidade: é um projeto de código aberto com largos anos de utilização e escrutínio público, o formato KDBX é aberto e suportado por múltiplas aplicações em várias plataformas, e o ecossistema de *plugins* permite estender funcionalidades [25]. Desde o KDBX 4, o formato suporta derivação de chave com Argon2 e cifra ChaCha20, com autenticação reforçada do cabeçalho e dos dados [26].

O EncryVault distingue-se sobretudo na experiência móvel e nas proteções para cenários extremos: no Android, o KeePass é utilizado através de clientes de terceiros (por exemplo, KeePassDX ou Keepass2Android) construídos sobre o formato KDBX, enquanto o EncryVault oferece uma experiência integrada, com *autofill* nativo, biometria, PINs e penalizações. Inclui ainda, de origem,

mecanismos que o KeePass não disponibiliza por omissão, como o cofre falso e a auto-destruição, e trata documentos sigilosos em blocos cifrados individualmente, com os metadados estruturais do ficheiro autenticados como AAD.

Esta comparação não posiciona o EncryVault como substituto do KeePass em ambiente *desktop*, onde a maturidade e o ecossistema deste último são difíceis de igualar. O EncryVault posiciona-se como uma solução *mobile-first* para Android, com foco em proteções locais avançadas e numa experiência unificada, servindo de base de estudo e de evolução no contexto deste projeto.

Quanto à cifra do conteúdo, a comparação relevante para o EncryVault é com os clientes do KeePass para Android (como o KeePassDX ou o KeePass2Android), que utilizam os mesmos algoritmos do formato KDBX. Ambas as abordagens recorrem a cifragem autenticada, mas de formas diferentes: o formato KDBX utiliza, por omissão, o AES-256 (em modo CBC) ou, em alternativa, o ChaCha20, assegurando a autenticação dos dados através de um HMAC-SHA-256 aplicado separadamente (esquema *Encrypt-then-MAC*). O EncryVault adota o XChaCha20-Poly1305, uma construção AEAD que integra cifragem e autenticação numa única primitiva. No contexto móvel, esta escolha tem duas vantagens práticas: o ChaCha20 tem bom desempenho em dispositivos sem aceleração de *hardware* para AES (comum em telemóveis), e o *nonce* alargado (192 bits) permite gerar um *nonce* aleatório por bloco e para o manifesto sem risco prático de colisão, o que simplifica a gestão de documentos guardados em blocos cifrados independentes. Importa sublinhar que todas estas cifras são seguras e modernas, a diferença não está na robustez do algoritmo, mas na sua integração e adequação à utilização móvel.

Quanto à derivação da chave a partir da palavra-passe mestra, tanto o EncryVault como os clientes KeePass para Android podem utilizar a família Argon2, vencedora da *Password Hashing Competition* e concebida para resistir a ataques por *hardware* especializado (GPU e ASIC) através de elevado consumo de memória. No formato KDBX, a Argon2 está disponível a partir da versão 4 (tipicamente na variante Argon2d), coexistindo com a alternativa mais antiga AES-KDF. O EncryVault utiliza sempre a variante Argon2id, que combina as propriedades da Argon2i e da Argon2d e acrescenta resistência a ataques por canais laterais (*side-channel*), particularmente relevante em dispositivos móveis. A diferença mais significativa para o utilizador de telemóvel, contudo, não está no algoritmo mas na configuração, no EncryVault os parâmetros da Argon2id estão fixados em valores seguros por omissão (perfis *sensitive* e *moderate* da biblioteca libsodium), ao passo que nos clientes KeePass a robustez efetiva depende de o utilizador configurar manualmente a memória, as iterações e o paralelismo, algo que, num telemóvel, a maioria dos utilizadores não ajusta.

3. EncryVault e gestão segura de credenciais

Neste capítulo é apresentada uma visão geral do EncryVault na sua versão final, enquanto aplicação móvel offline orientada para a gestão segura de credenciais e informação sensível. O objetivo é permitir ao leitor compreender, de forma global, o produto desenvolvido, o problema que procura resolver, os princípios que orientaram a sua conceção e as principais funcionalidades disponibilizadas.

Ao longo deste capítulo pretende-se responder a questões fundamentais relativamente ao projeto, nomeadamente:

- O que é o EncryVault e que problemas procura resolver?
- Que princípios orientaram a sua conceção?
- Que funcionalidades compõem a versão final da aplicação?
- Como se apresenta a aplicação ao utilizador final?

É ainda importante referir que uma solução como o EncryVault pode ser analisada em duas perspetivas complementares. A primeira corresponde à perspetiva do utilizador final, que procura uma aplicação simples, rápida e de confiança para guardar e consultar credenciais sem complicações. A segunda corresponde à perspetiva do programador, centrada na forma como a aplicação foi construída para garantir confidencialidade, integridade, proteção das chaves, armazenamento local seguro e robustez face a cenários de perda, roubo ou acesso indevido ao dispositivo.

Desta forma, este capítulo apresenta o EncryVault enquanto produto final, remetendo os detalhes técnicos, de investigação, requisitos, arquitetura, design, desenvolvimento e testes para os capítulos seguintes, onde são aprofundadas as decisões tomadas e o processo de consolidação da solução ao longo do Projeto II.

3.1 O que é o EncryVault

O EncryVault é uma aplicação móvel 100% *offline* que funciona como um cofre digital encriptado para guardar e organizar credenciais e informação sensível, permitindo ao utilizador proteger os seus dados localmente através de uma palavra-passe mestra.

Uma solução deste tipo procura reduzir práticas inseguras comuns, como guardar palavras-passe em notas ou ficheiros sem proteção, e reutilizar credenciais em vários serviços, mantendo o utilizador com controlo local sobre o ficheiro do cofre.

De forma geral, o EncryVault providencia:

- Acesso ao cofre através de palavra-passe mestra, com biometria opcional e proteções adicionais como *pepper* e penalizações por tentativas falhadas.
- Criação, consulta, edição e remoção de entradas, com categorias, *tags*, favoritos e histórico de palavras-passe.
- Pesquisa, filtragem e ordenação de entradas por vários critérios.
- Proteção de sessão com *auto-lock* configurável, bloqueio manual e bloqueio em eventos do ciclo de vida da aplicação.
- Ferramentas como gerador de palavras-passe, saúde das palavras-passe e exportação/importação para *backup*.
- Documentos sigilosos guardados dentro do cofre em blocos cifrados, com pré-visualização e exportação.
- Lixeira com retenção configurável, restauro e eliminação definitiva com confirmação.
- *Autofill* em Android para preenchimento de credenciais noutras aplicações.
- Modo privado, proteção contra capturas de ecrã e limpeza automática da área de transferência.
- Proteções avançadas opcionais como PINs, cofre falso e auto-destruição.

3.2 Princípios de conceção

A conceção do EncryVault foi orientada por um conjunto de princípios que se mantiveram estáveis do Projeto I ao Projeto II e que explicam grande parte das decisões descritas nos capítulos seguintes:

- Funcionamento totalmente *offline* - a aplicação não depende de servidores, contas ou serviços externos; todo o processamento e armazenamento acontecem no dispositivo.
- Controlo local dos dados - o cofre é um único ficheiro (.vltx) que o utilizador pode copiar, exportar e guardar onde entender, sem dependência de sincronizações automáticas.
- Segurança por omissão e em camadas - o conteúdo está sempre cifrado em repouso e as proteções (palavra-passe mestra, biometria, *pepper*, PINs e penalizações) complementam-se, de modo a que a falha de uma camada não comprometa, por si só, o cofre.
- Proteção em cenários extremos - mecanismos como o cofre falso e a auto-destruição foram pensados para situações de coação ou de perda de controlo do dispositivo.

- Robustez e previsibilidade - o formato do ficheiro é verificável (cifragem autenticada com AAD) e as gravações são atómicas, evitando estados intermédios mesmo em caso de falha durante a escrita.

Estes princípios concretizam-se nas medidas de mitigação do Capítulo 4 e nos requisitos definidos no Capítulo 5.

3.3 Visão geral da aplicação final

O EncryVault é composto por um conjunto de ecrãs e elementos principais:

- Ecrã de criação e desbloqueio do cofre.
- Página principal do cofre com lista de entradas, pesquisa, filtros, ordenação e alertas de saúde.
- Página de criar/editar entrada com campos essenciais e geração de palavra-passe.
- Página de configurações com opções de segurança, aparência, lixeira e *backup*.
- Página de documentos sigilosos, lixeira do cofre e página de Ajuda/FAQ.

As Figuras 3 a 35 apresentadas nas páginas seguintes exemplificam os pontos referidos anteriormente.

Começando pela Figura 3, a qual apresenta o ecrã de arranque da aplicação, visível no início da execução do EncryVault.



Figura 3 - Ecrã de arranque (*splash screen*) do EncryVault.

A Figura 4 apresenta o ecrã de termos e condições, onde o utilizador toma conhecimento das regras de utilização antes de prosseguir.

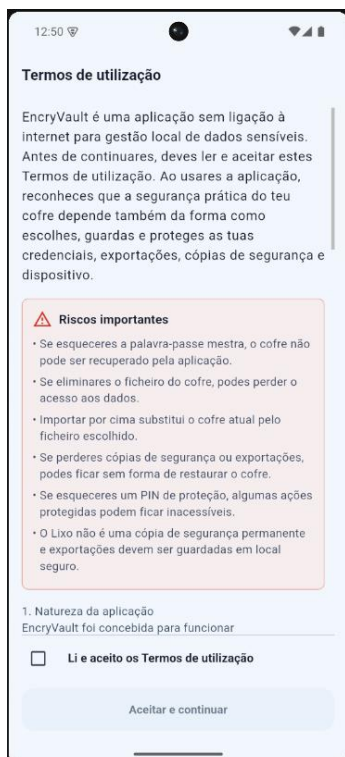


Figura 4 - Ecrã de termos e condições do EncryVault.

A Figura 5 mostra o ecrã inicial com as opções principais para criar um cofre ou importar um cofre existente.



Figura 5 - Ecrã inicial com opções de criar ou configurar o cofre.

A Figura 6 exemplifica o processo de importação do cofre, incluindo o caminho da pasta para ser seleccionado o ficheiro.



Figura 6 - Importação do cofre e escolha de tema.

A Figura 7 exemplifica o processo de criação do cofre, incluindo a definição da palavra-passe mestra para protecção do conteúdo.



Figura 7 - Criação de cofre e definição da palavra-passe mestra.

A Figura 8 apresenta o ecrã de desbloqueio do cofre, onde é necessária a palavra-passe mestra para aceder às entradas guardadas.



Figura 8 - Desbloqueio do cofre através da palavra-passe mestra.

A Figura 9 mostra a página principal do cofre ainda sem entradas, a partir da qual o utilizador tem acesso rápido às funcionalidades principais.

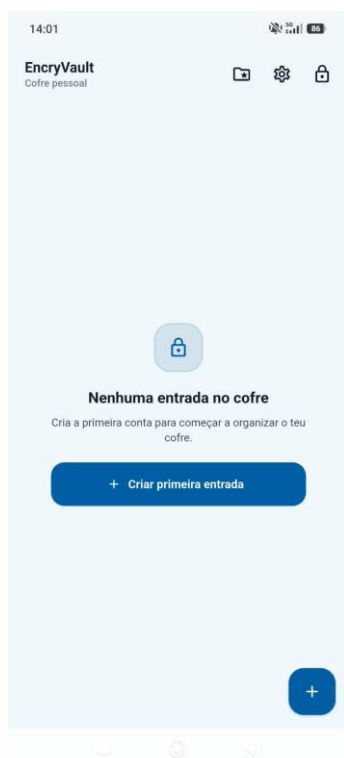


Figura 9 - Página principal do cofre vazia.

A Figura 10 mostra o formulário de criação de entrada, permitindo guardar a informação de forma estruturada.

The figure consists of three side-by-side screenshots of the 'Nova entrada' (New entry) form in the EncryVault app. The first screenshot shows the top section with fields for 'Título', 'Utilizador/correio eletrónico', and 'URL / site', along with 'Links rápidos' (Instagram, Facebook, Mais) and a 'Palavra-passe' field with a strength indicator. The second screenshot shows the 'Gerador de palavra-passe' (Password generator) section with options for length (20), complexity (Mixed case, Lowercase, Numbers, Symbols), and a 'Gerar' button. The third screenshot shows the 'Categoria' (Category) and 'Etiquetas' (Tags) sections, with a 'Criar' button at the bottom.

Figura 10 - Criação de entrada no EncryVault.

A Figura 11 mostra a página principal do cofre, onde estão apresentadas as entradas guardadas e o acesso rápido às funcionalidades principais.



Figura 11 - Página principal do cofre com lista de entradas.

A Figura 12 ilustra a funcionalidade de pesquisa e filtragem, permitindo localizar entradas de forma mais rápida e organizada.



Figura 12 - Pesquisa e filtragem de entradas no cofre.

A Figura 13 apresenta o detalhe de uma entrada, onde o utilizador pode visualizar os campos guardados e executar ações como copiar, editar e eliminar.



Figura 13 - Visualização do detalhe de uma entrada guardada no cofre.

A Figura 14 mostra o formulário de edição de entradas, permitindo atualizar a informação de forma estruturada.

The figure displays two side-by-side screenshots of the 'Editar entrada' (Edit entry) form in the EncryVault application. Both screenshots show the time 22:06 and signal strength indicators at the top.

Left Screenshot:

- Título:** Input field containing 'aaa'.
- Utilizador/correio eletrónico:** Input field containing 'bbb'.
- URL / site:** Input field with a link icon.
- Links rápidos:** A section with buttons for 'Instagram', 'Facebook', and 'Mais'.
- Palavra-passe:** Input field with a strength indicator (red bar) and a list of tips:
 - Força: Fraca
 - Mínimo 12 caracteres
 - Pelo menos 1 letra maiúscula
 - Pelo menos 1 letra minúscula
 - Pelo menos 1 número
 - Pelo menos 1 carácter especial permitido
 - Recomendação: usa uma palavra-passe forte, mas podes guardar qualquer palavra-passe.
 - Muito curta.
 - Sem letras maiúsculas.
 - Sem números.
 - Sem símbolos.
 - Tem caracteres demasiado repetidos.

Right Screenshot:

- Notas:** Input field.
- Categoria:** A list of categories with icons:
 - Social:** Selected (indicated by a checkmark).
 - Correio eletrónico
 - Banco
 - Jogos
 - Trabalho
 - Outros
- Etiquetas:** Input field containing 'Facebook'.
- Separadas por vírgula.** Text label.
- Guardar:** A large blue button at the bottom.

Figura 14 - Edição de entradas no EncryVault

As figuras seguintes completam a apresentação da versão final, cobrindo as restantes áreas da aplicação - configurações, segurança, gestão de documentos e demais funcionalidades introduzidas no Projeto II. Todas as capturas de ecrã incluídas correspondem a esta versão final, cujo design e funcionalidades evoluíram de forma assinalável relativamente ao protótipo apresentado no Projeto I.

A gestão de documentos sigilosos é feita no ecrã da Figura 15, onde o utilizador guarda, pré-visualiza e exporta ficheiros sensíveis cifrados dentro do próprio cofre, com limites de tamanho e lixeira própria.



Figura 15 - Documentos sigilosos guardados no cofre.

A Figura 16 apresenta a área de configurações, onde se concentram as opções de segurança, preenchimento, PINs de proteção, cofre, lixo, *backup* do cofre, links rápidos, aparência, sessão, auditoria/saúde, termos de utilização, FAQ e sobre a aplicação.

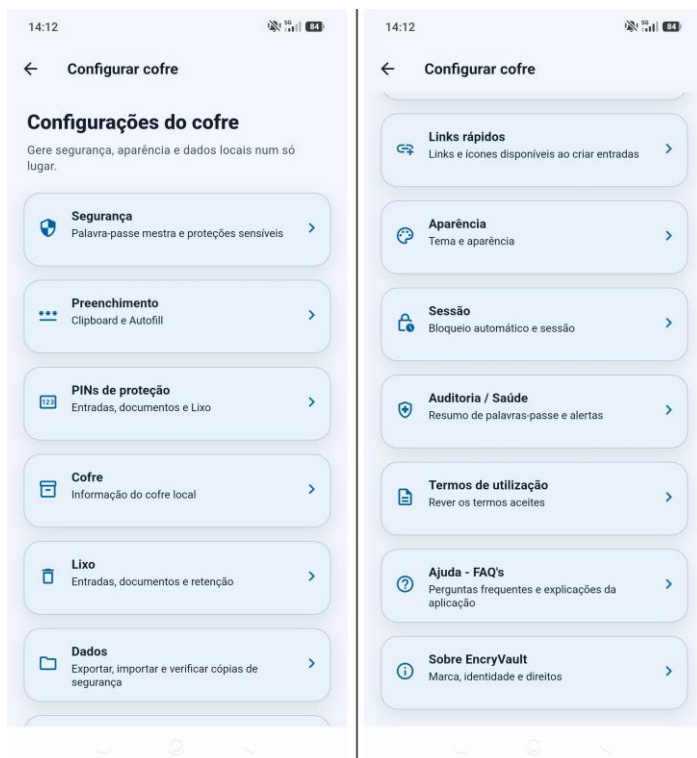


Figura 16 - Página de configurações e diversas opções no cofre.

O centro das proteções do cofre encontra-se no menu de segurança, apresentado na Figura 17. É a partir daqui que se ativam e configuram mecanismos como a biometria, o *pepper*, alguns PINs, o modo privado e as proteções de emergência.

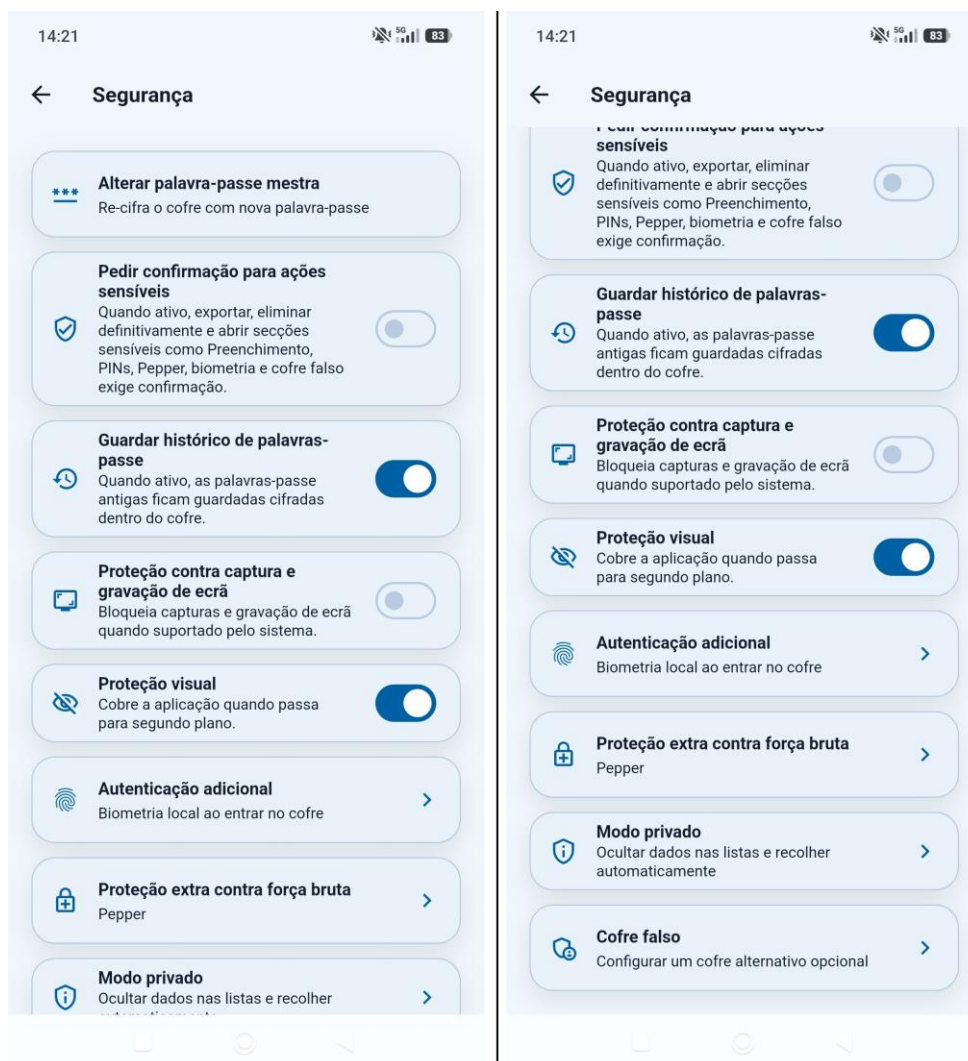


Figura 17 - Menu de segurança e proteções sensíveis.

Como reforço do desbloqueio, a autenticação biométrica pode ser ativada no ecrã da Figura 18, passando a exigir a impressão digital do utilizador ao entrar no cofre e na confirmação de ações sensíveis.

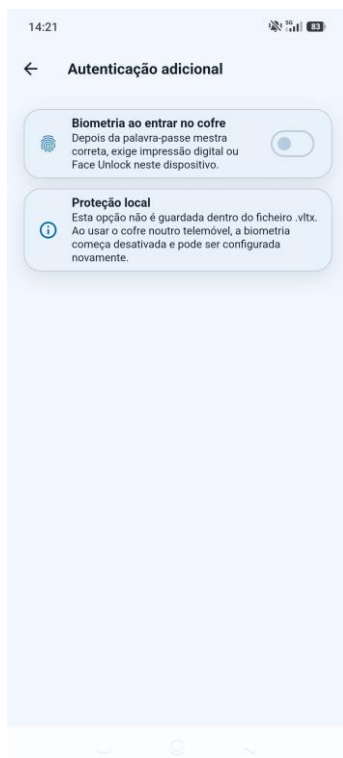


Figura 18 - Autenticação biométrica adicional ao entrar no cofre.

Para dificultar ataques por tentativa e erro, a Figura 19 mostra a proteção adicional baseada num *pepper*, um segredo guardado no armazenamento seguro do dispositivo e combinado na derivação da chave, acompanhado de um código de recuperação.

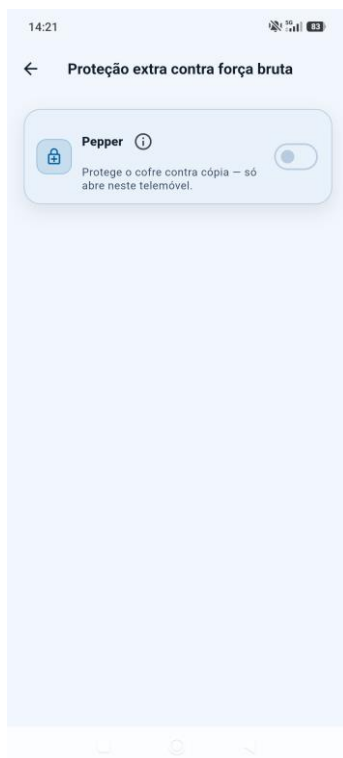


Figura 19 - Proteção contra força bruta (*pepper*)

O modo privado, ilustrado na Figura 20, destina-se a contextos em que o ecrã pode ser observado por terceiros: os nomes e os dados sensíveis ficam ocultos na lista de entradas, sendo revelados apenas por toque.

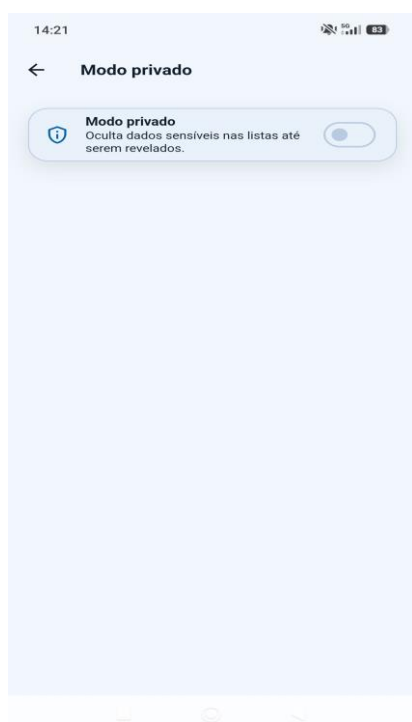


Figura 20 - Modo privado, ocultação de dados sensíveis na interface.

A Figura 21 corresponde à configuração do cofre falso, um mecanismo de proteção em situações de coação: uma palavra-passe alternativa abre um cofre com conteúdo fictício, mantendo o cofre real inacessível e indistinguível.

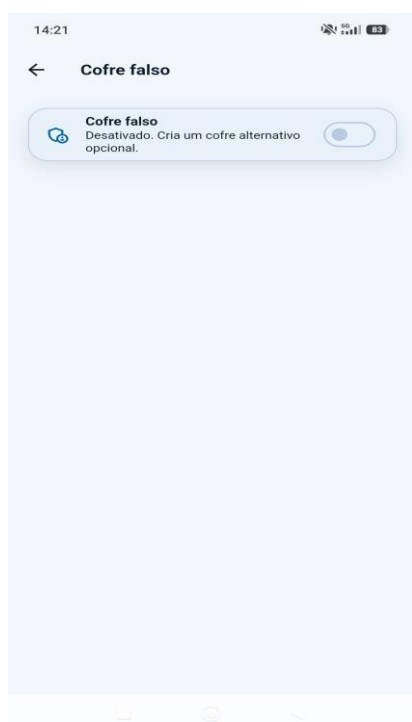


Figura 21 - Configuração do cofre falso (*decoy*)

O preenchimento automático é configurado na área da Figura 22, que permite ativar o serviço de *autofill* no Android e definir o comportamento de limpeza do *clipboard* após a cópia de credenciais.

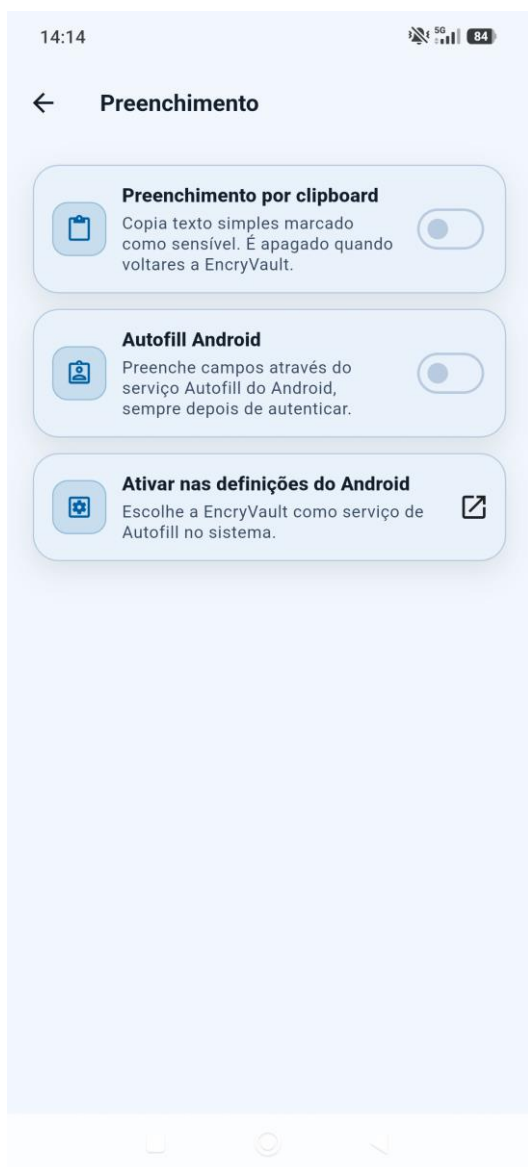


Figura 22 - Preenchimento automático (*autofill*) e gestão do *clipboard*.

O controlo de acesso por PIN é apresentado a partir da Figura 23, que mostra a vista geral dos PINs de proteção e o ponto de entrada para a sua configuração.

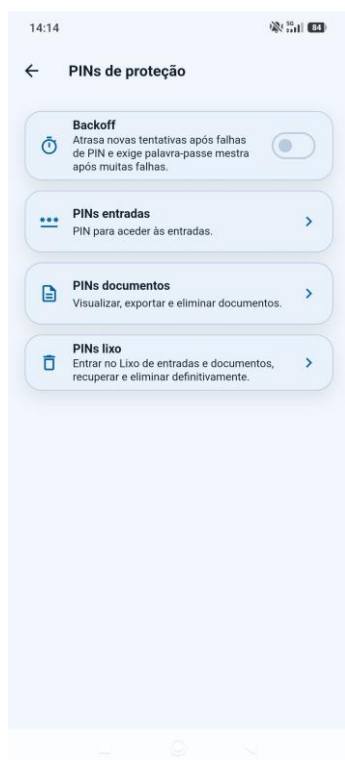


Figura 23 - PINs de proteção (entradas, documentos e lixeira).

A Figura 24 detalha o PIN de proteção das entradas, que pode ser exigido ao aceder à lista de entradas do cofre.

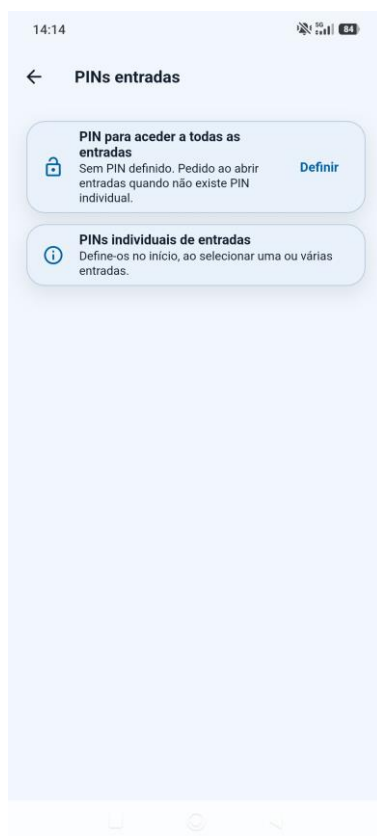


Figura 24 - PINs de proteção das entradas.

A Figura 25 apresenta o PIN de proteção dos documentos, aplicável ao acesso aos documentos sigilosos guardados no cofre.



Figura 25 - PINs de proteção dos documentos.

A Figura 26 mostra o PIN de proteção da lixeira, que restringe o acesso às entradas e documentos eliminados.



Figura 26 - PINs de proteção da lixeira.

Em complemento, a auto-destruição é configurada na área apresentada na Figura 27 (Zona de Risco), permitindo eliminar de forma irreversível o cofre real e os respectivos *backups* através de um PIN dedicado ou de um limite de tentativas falhadas.

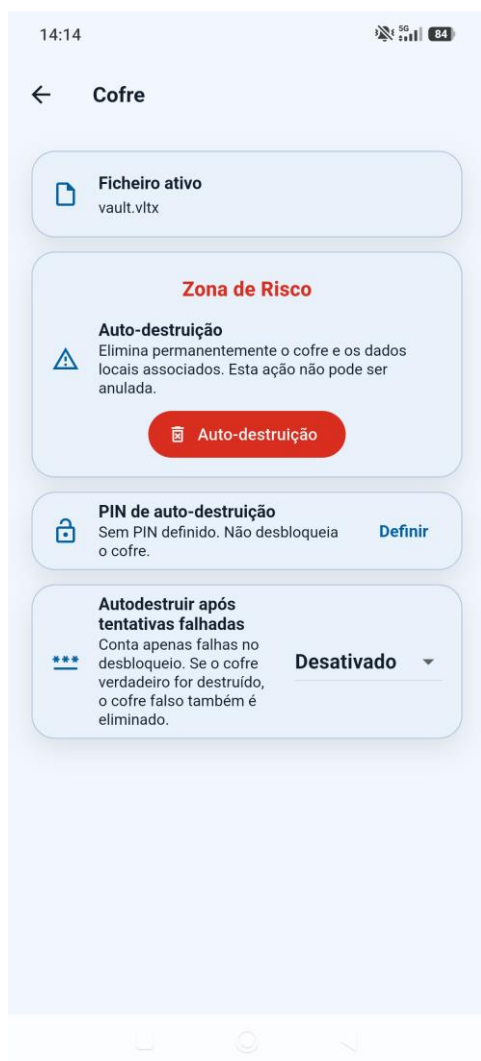


Figura 27 - Auto-destruição do cofre (Zona de Risco).

A lixeira do cofre, apresentada na Figura 28, mantém durante um período configurável as entradas e os documentos eliminados, permitindo restaurá-los ou removê-los definitivamente, sempre com confirmação e com os dados sensíveis ocultos.

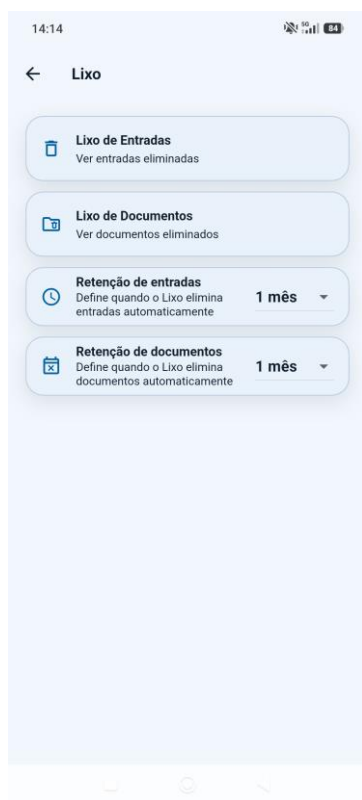


Figura 28 - Lixeira do cofre (entradas e documentos).

As operações de cópia de segurança são feitas no ecrã da Figura 29, que reúne a exportação e a importação do cofre, permitindo guardar e repor o ficheiro cifrado de forma controlada.



Figura 29 - Exportação e importação do cofre.

Para acesso rápido aos serviços associados a cada entrada, a Figura 30 mostra os links rápidos, que abrem diretamente o serviço correspondente a partir do detalhe da entrada.



Figura 30 - Links rápidos das entradas.

A personalização visual da aplicação é feita no ecrã da Figura 31, onde o utilizador escolhe entre tema claro, escuro ou automático e alterna entre os modos visuais Moderno e Clássico, que se aplicam de forma consistente a toda a aplicação.



Figura 31 - Aparência: temas e modos visuais (Moderno e Clássico).

A gestão da sessão é feita no ecrã da Figura 32, onde se define o tempo de inatividade até ao bloqueio automático (*auto-lock*), garantindo que o cofre não permanece acessível quando o dispositivo é deixado sem supervisão.

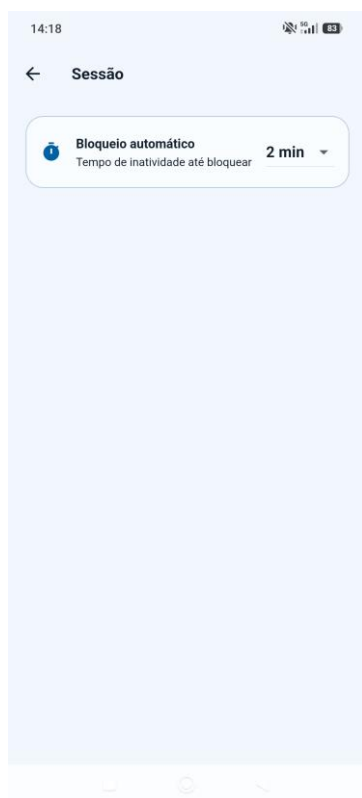


Figura 32 - Sessão e bloqueio automático (*auto-lock*).

A área de saúde das palavras-passe ajuda o utilizador a manter boas práticas. A Figura 33 apresenta a primeira parte desta área, com a deteção de palavras-passe fracas, reutilizadas ou antigas.

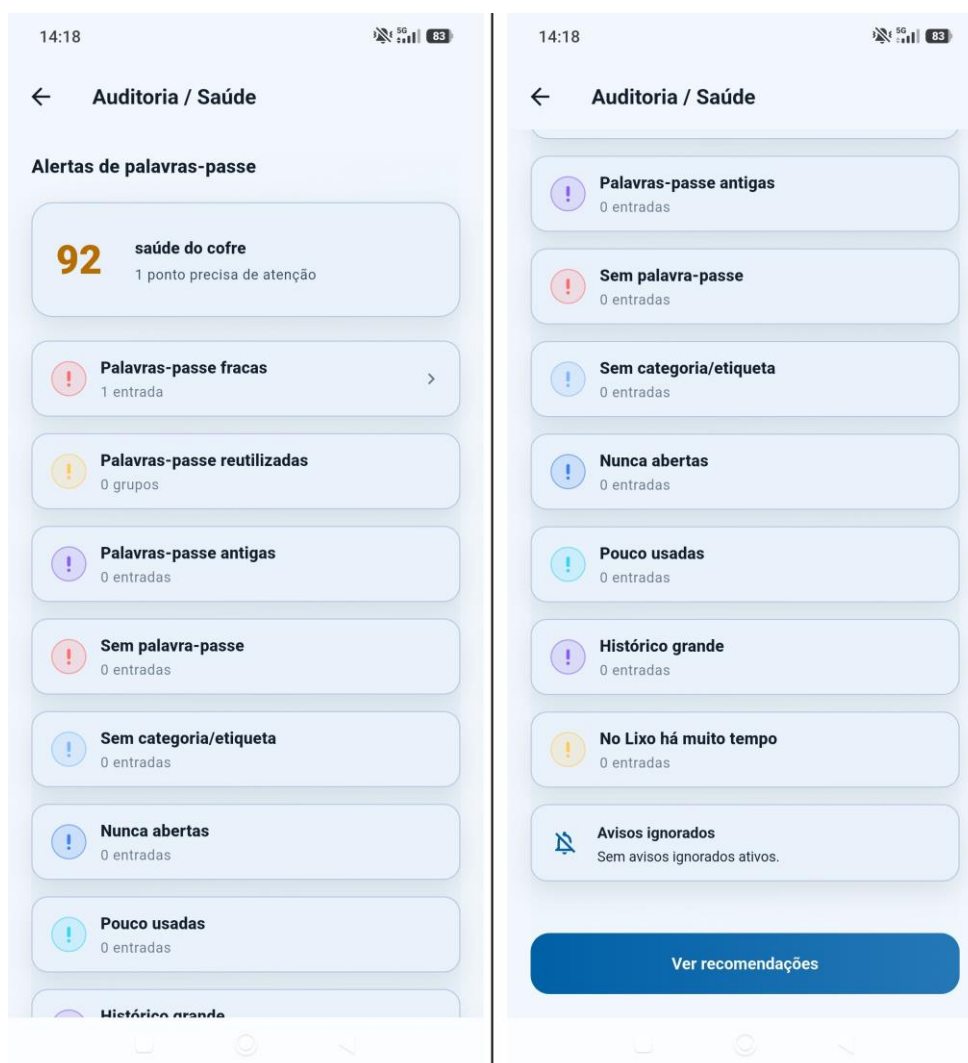


Figura 33 - Saúde das palavras-passe: alertas e recomendações.

A aplicação inclui ainda uma área de Ajuda/FAQ, apresentada na Figura 34, com respostas pesquisáveis sobre as funcionalidades e as opções de segurança do EncryVault.



Figura 34 - Página de Ajuda/FAQ.

Por fim, a Figura 35 mostra o ecrã “Sobre EncryVault”, que reúne a identidade da aplicação e informação sobre a sua natureza e finalidade.

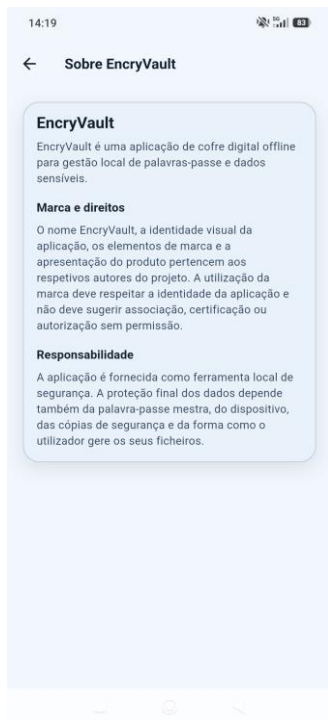


Figura 35 - Ecrã "Sobre EncryVault".

4. Investigação

Neste capítulo é apresentada a investigação realizada no âmbito do projeto, com o objetivo de reunir informação relevante sobre gestores de credenciais, boas práticas de segurança e aspetos técnicos essenciais para garantir que o EncryVault é uma solução coerente, segura e viável.

A investigação parte da análise comparativa de soluções existentes apresentada no Capítulo 2 e foca-se em dois pontos principais: o levantamento de boas práticas de segurança aplicáveis a um gestor de credenciais *offline* e a definição de um modelo de ameaça com as respetivas medidas de mitigação, terminando com uma síntese das implicações para o desenvolvimento do EncryVault.

Desta forma, o conteúdo deste capítulo serve de base para as decisões tomadas nos capítulos seguintes, nomeadamente na definição de requisitos e na proposta de arquitetura do sistema.

As soluções existentes analisadas no Capítulo 2 (Bitwarden, 1Password, KeePass e LastPass) evidenciam diferentes abordagens ao nível do armazenamento e das funcionalidades, servindo como referência para identificar padrões comuns em gestores de credenciais [14]-[18].

Com base nessa análise comparativa, a investigação centra-se agora nas boas práticas de segurança aplicáveis ao modelo *offline* do EncryVault e nas medidas necessárias para proteger o cofre local.

4.1 Boas práticas de segurança aplicáveis

Uma aplicação de gestão de credenciais deve garantir confidencialidade e integridade dos dados, reduzindo a probabilidade de exposição mesmo quando existe acesso indevido ao armazenamento do dispositivo [4]-[6].

Uma boa prática essencial é a utilização de derivação de chave a partir da palavra-passe mestra, evitando o uso direto da palavra-passe como chave de encriptação, e em conjunto com encriptação moderna com autenticação garante que o conteúdo do cofre não pode ser lido nem alterado sem deteção, reduzindo também riscos de manipulação do ficheiro do cofre [1]-[3], [4]-[6].

No contexto de cofres encriptados, a gestão correta de parâmetros como *salt* e *nonce* é importante para manter segurança criptográfica, e do ponto de vista de utilização é comum aplicar medidas como bloqueio automático (*auto-lock*) após inatividade e bloqueio manual, bem como minimizar a exposição de dados sensíveis evitando *logs*, texto simples e controlando a cópia para a área de transferência com limpeza automática [2]-[5].

Por fim, operações de *backup* devem ser feitas de forma consciente, garantindo que exportações e ficheiros gerados não ficam expostos em locais

inseguros e que o utilizador compreende o impacto de partilhar ou armazenar esses ficheiros [4], [5].

4.1.1 Modelo de ameaça

Para garantir a robustez do EncryVault, foram considerados cenários realistas de ataque e perda de controlo do dispositivo/dados:

- Roubo ou cópia do ficheiro do cofre (.vltx) - Um atacante obtém o ficheiro (*backup*, extração via PC, *cloud* não autorizada, partilha indevida, etc.) e tenta recuperar os dados.
- Tentativas de força-bruta à palavra-passe mestra - Um atacante tenta adivinhar a palavra-passe por tentativa/erro, ou com dicionários e listas de credenciais.
- Cofre aberto em dispositivo desbloqueado (acesso oportunista) - O utilizador deixa o telemóvel desbloqueado e outra pessoa tenta aceder ao conteúdo enquanto o cofre está desbloqueado.
- Exposição acidental por captura de ecrã/recentes - Conteúdo sensível pode aparecer em *screenshots*, gravações, ou no ecrã de aplicações recentes.
- Corrupção ou adulteração do ficheiro do cofre - O ficheiro pode ser alterado, truncado, ou corrompido para provocar falhas, perda de dados, ou exploração de erros.
- Extração de segredos em memória/*clipboard* - Dados podem ficar temporariamente expostos em memória, *logs*, ou na área de transferência.

4.1.2 Medidas de mitigação no EncryVault

Com base nas ameaças anteriores, o EncryVault aplica as seguintes medidas:

- Encriptação autenticada (AEAD) - O cofre é cifrado usando um algoritmo AEAD (ex.: XChaCha20-Poly1305), garantindo confidencialidade e integridade. Qualquer alteração no ficheiro (adulteração/corrupção) é detetada durante a decifragem [2], [3].
- Derivação de chave resistente a força-bruta (KDF) - A chave de encriptação é derivada da palavra-passe mestra com Argon2id, incluindo *salt* aleatório e parâmetros configuráveis (memória/operações/paralelismo), aumentando significativamente o custo de ataques *offline* [1], [4]-[6].
- *Pepper* com armazenamento seguro - Opcionalmente, é combinado na derivação da chave um segredo adicional (*pepper*) guardado no armazenamento seguro do dispositivo, com confirmação biométrica, fazendo com que a posse do ficheiro .vltx deixe de ser suficiente para

ataques *offline*; é gerado um código de recuperação para repor o acesso noutro dispositivo.

- Penalização por tentativas falhadas - Após 3 tentativas incorretas é aplicado um bloqueio de 10 minutos, sendo adicionados 15 minutos por cada conjunto adicional de 2 falhas, com reposição após autenticação bem-sucedida, reduzindo a eficácia de ataques por tentativa/erro no próprio dispositivo.
- *Auto-lock* por inatividade e bloqueio ao sair da app - O cofre regressa ao estado bloqueado após um período configurável de inatividade, e sempre que a aplicação passa para segundo plano (quando aplicável) [4], [6].
- Proteção contra captura de ecrã e “*recents*” - Ativação de *FLAG_SECURE* em Android quando o cofre está desbloqueado, mitigando exposição em *screenshots*, gravações e no ecrã de aplicações recentes [7].
- Redução de exposição no *clipboard* - Quando o utilizador copia credenciais, a aplicação limpa automaticamente a área de transferência ao sair do ecrã da entrada e, quando a aplicação regressa de segundo plano, volta a limpá-la; a limpeza ocorre apenas se o conteúdo copiado ainda pertencer à aplicação, minimizando o tempo de exposição dos dados [5].
- PINs de proteção - Áreas e ações sensíveis (entradas, documentos e lixeira) podem ser protegidas por PINs globais ou individuais, com atraso progressivo (*backoff*) em caso de falhas e recuperação através da palavra-passe mestra.
- Cofre falso e auto-destruição - Como proteção em cenários de coação, uma palavra-passe alternativa pode abrir um cofre falso com conteúdo fictício, e um PIN de auto-destruição (ou um limite de tentativas falhadas) pode eliminar de forma irreversível o cofre real e os respetivos *backups*.
- Sem dependência de servidores (*offline-first*) - Ao não existir *backend*, reduz-se a superfície de ataque associada a APIs, sessões remotas, vazamentos de base de dados e ataques de rede [4]-[6].

Nenhuma aplicação consegue eliminar todos os riscos, por exemplo o *malware* com privilégios elevados. O objetivo é reduzir a probabilidade e o impacto de ataques típicos contra gestores de palavras-passe.

4.2 Síntese da investigação e implicações para o EncryVault

A análise de mercado realizada no Capítulo 2 suporta a direção do EncryVault enquanto aplicação *offline*, com foco em controlo local e privacidade. As funcionalidades mais valorizadas nas soluções analisadas (cofre encriptado, organização, pesquisa, geração de palavras-passe e exportação/importação) foram consideradas requisitos centrais. Em paralelo, procurou-se reduzir dependências externas e manter um modelo simples e coerente, adequado a um

cofre local protegido por palavra-passe mestra e mecanismos criptográficos robustos [14]-[19].

Esta opção afasta deliberadamente qualquer dependência de servidores externos ou de serviços online, como mecanismos de sincronização automática entre dispositivos, privilegiando em troca um modelo totalmente local, com maior controlo sobre o ficheiro do cofre e uma superfície de ataque remota mais reduzida.

Com base na análise de soluções existentes e nas boas práticas identificadas, é possível definir um conjunto de requisitos e decisões técnicas que suportam o desenvolvimento do EncryVault, mantendo o foco em funcionamento *offline*, controlo local e proteção adequada do cofre [1]-[7], [14]-[19].

Em particular, a combinação de Argon2id para derivação de chave, XChaCha20-Poly1305 como cifra autenticada, *pepper* opcional com suporte biométrico e mecanismos de *auto-lock* e penalização de tentativas falhadas coloca a versão final do projeto alinhada com recomendações de OWASP e NIST para armazenamento seguro de credenciais em dispositivos do utilizador.

5. Requisitos e arquitetura do sistema

Neste capítulo são apresentados os requisitos do sistema e a arquitetura do EncryVault, com o objetivo de definir de forma clara o que a aplicação faz na sua versão final e como está organizada internamente.

A definição destes pontos foi importante para garantir consistência ao longo do Projeto II, permitindo evoluir a base do Projeto I de forma controlada e sem reestruturações profundas, mesmo com a introdução de funcionalidades de maior complexidade, como documentos sigilosos e o novo formato de ficheiro.

5.1 Requisitos funcionais

Os requisitos funcionais descrevem as funcionalidades que o sistema disponibiliza ao utilizador na versão final do Projeto II, distinguindo a base consolidada do Projeto I das funcionalidades implementadas nesta fase.

5.1.1 Requisitos base consolidados do Projeto I

De forma geral, a base transitada do Projeto I e consolidada no Projeto II permite:

- Criar um cofre e definir uma palavra-passe mestra.
- Desbloquear o cofre com a palavra-passe mestra.
- Guardar entradas no cofre, criando, consultando, editando e removendo registos.
- Pesquisar entradas e aplicar filtros (por exemplo, por *tags*).
- Gerar palavras-passe para utilização nas entradas.
- Copiar campos de uma entrada (por exemplo, utilizador e palavra-passe).
- Bloquear o cofre manualmente e bloquear automaticamente após inatividade (*auto-lock*).
- Exportar o cofre para ficheiro de *backup* e importar um cofre a partir de ficheiro.
- Apresentar ecrãs informativos necessários (por exemplo, termos de utilização na primeira execução).

5.1.2 Requisitos implementados no Projeto II

- Tema claro, tema escuro e modo automático (segue o sistema), aplicados de forma consistente em toda a aplicação, dentro e fora do cofre.
- Dois modos visuais alternativos (Moderno e Clássico), baseados em *design tokens*, configuráveis nas definições e aplicados a todos os ecrãs.

- Política de palavra-passe mestra na criação do cofre, com indicador de força e regras mínimas obrigatórias (12 caracteres, maiúsculas, minúsculas, números e caracteres especiais), bloqueando a criação de palavras-passe fracas.
- Autenticação biométrica (impressão digital) no desbloqueio do cofre e na confirmação de ações sensíveis, com base nas capacidades do dispositivo.
- Penalizações por tentativas falhadas no desbloqueio do cofre: bloqueio de 10 minutos após 3 falhas e mais 15 minutos por cada conjunto adicional de 2 falhas, com reposição após autenticação bem-sucedida.
- Proteção contra capturas e gravação de ecrã (*FLAG_SECURE*) enquanto o cofre está desbloqueado, incluindo ocultação no ecrã de aplicações recentes.
- Área de “saúde das palavras-passe” que identifica palavras-passe fracas, reutilizadas, antigas, vazias, sem categoria, nunca abertas ou raramente utilizadas, com alertas resumidos na página principal do cofre e possibilidade de dispensar alertas específicos.
- Recomendação de alteração da palavra-passe ao fim de um determinado tempo, com intervalo ajustado automaticamente consoante a força da palavra-passe da entrada.
- Organização das entradas por categorias predefinidas (Social, Correio eletrónico, Banco, Jogos, Trabalho e Outros), em complemento das *tags* e dos favoritos.
- Histórico de palavras-passe por entrada, guardado de forma cifrada dentro do cofre, permitindo consultar valores anteriores.
- Ecrã de detalhes de cada entrada com data de criação, última edição, última abertura, número de aberturas e tempo desde a última alteração da palavra-passe.
- Lixeira do cofre com retenção configurável (7 dias, 1 mês, 3 meses, 6 meses, 1 ano ou nunca) e eliminação automática ao fim do período definido.
- Alteração da palavra-passe mestra nas configurações do cofre, com recifragem completa do conteúdo.
- *Pepper* opcional combinado na derivação da chave, guardado no armazenamento seguro do dispositivo com confirmação biométrica e código de recuperação para reposição de acesso noutro dispositivo.
- Ordenação das entradas por ordem alfabética (A-Z e Z-A), data de criação (mais recentes e mais antigas), edição recente, abertura recente, mais/menos utilizadas e por categoria.

- Na lixeira, restauro de entradas e eliminação definitiva, individual ou em seleção múltipla, sempre com confirmação explícita.
- Na lixeira, visualização limitada do conteúdo, mantendo dados sensíveis ocultos e a palavra-passe mascarada.
- Documentos sigilosos: importação, pré-visualização (incluindo PDF), exportação e eliminação de ficheiros guardados de forma cifrada dentro do cofre, com lixeira própria e limites de tamanho definidos.
- Novo formato de ficheiro do cofre (V3, extensão .vltx), com suporte a documentos em blocos cifrados (*chunks*), manifesto cifrado e verificação de integridade, descrito na secção 5.6.
- Preenchimento automático (*autofill*) no Android, através de serviço nativo opcional, com correspondência por aplicação ou URL e autenticação antes de preencher.
- PINs de proteção globais (entradas, documentos e lixeira) e PINs individuais por entrada ou documento, com atraso progressivo (*backoff*) em caso de falhas e recuperação através da palavra-passe mestra.
- Cofre falso (*decoy*): uma palavra-passe alternativa abre um cofre com conteúdo fictício, como proteção em cenários de coação.
- Auto-destruição: um PIN dedicado, ou um limite configurável de tentativas falhadas, elimina de forma irreversível o cofre real, os *backups* e os segredos guardados no armazenamento seguro.
- Modo privado: ocultação de nomes e dados sensíveis na lista de entradas, com revelação por toque e ocultação automática.
- Limpeza automática do *clipboard* ao nível nativo após copiar credenciais, apenas se o conteúdo copiado ainda for o da aplicação.
- Links rápidos com ícones nas entradas, permitindo abrir diretamente o serviço associado (URL) a partir do detalhe da entrada.
- Página de Ajuda/FAQ integrada, com pesquisa, cobrindo as funcionalidades e as opções de segurança da aplicação.

5.1.3 Requisitos adaptados face ao planeado

Durante o Projeto II, alguns requisitos planeados no Projeto I foram adaptados em função da investigação efetuada e da experiência de utilização pretendida:

- O 2FA por código (TOTP) foi substituído por um conjunto de mecanismos locais considerados mais adequados a uma aplicação *offline*: autenticação biométrica, *pepper* com armazenamento seguro e PINs de proteção.
- As pastas foram substituídas por categorias predefinidas, combinadas com *tags* e favoritos, simplificando a organização sem estruturas hierárquicas.

- A personalização por cor/emoji de cada entrada foi substituída por ícones de categoria e links rápidos com ícones, mantendo identificação visual consistente.
- O histórico de acessos ao cofre foi adiado para trabalho futuro, por ter menor prioridade face às proteções implementadas.

5.1.4 Funcionalidades futuras opcionais

- Implementar um teclado interno na aplicação para reduzir riscos associados a *keyloggers*.
- Implementar um teclado aleatório para introdução da palavra-passe mestra ou PIN, alterando a posição das teclas a cada utilização.
- Definir um limite máximo configurável para o histórico de palavras-passe de cada entrada, controlando o crescimento do cofre.
- Implementar um histórico de acessos ao cofre com data/hora, permitindo consulta pelo utilizador.
- Reforçar a limpeza de dados sensíveis em memória (zeroização) dentro das limitações da plataforma e estudar o suporte a outras plataformas.

5.2 Requisitos não funcionais

Os requisitos não funcionais descrevem características de qualidade e restrições do sistema, relacionadas com segurança, desempenho, usabilidade e manutenção.

5.2.1 Requisitos assegurados desde o Projeto I

- A aplicação deve funcionar em modo *offline*, sem dependência de serviços externos para operar.
- O cofre é guardado localmente num ficheiro encriptado, sem armazenamento de conteúdo em texto simples no dispositivo.
- O acesso ao cofre exige autenticação através de palavra-passe mestra.
- A aplicação mantém uma utilização simples e organizada, com navegação clara entre ecrãs principais.
- A solução manteve uma estruturada modular que permitiu a evolução ao longo do Projeto II sem reestruturações profundas.

5.2.2 Requisitos reforçados no Projeto II

- A proteção de dados sensíveis durante a utilização foi reforçada, minimizando a exposição de conteúdo em memória, em operações de cópia e na própria interface (modo privado e dados mascarados).
- O cofre passou a aplicar penalizações progressivas a tentativas repetidas de acesso e controlos adicionais como PINs com *backoff* e auto-destruição opcional.
- A aplicação impede capturas e gravação de ecrã quando o cofre está desbloqueado, reduzindo o risco de exposição visual.
- A consistência visual e a acessibilidade foram melhoradas com suporte completo a modo claro/escuro, modo automático e dois modos visuais (Moderno e Clássico).
- A aplicação foi testada e validada em cenários reais e casos limite, incluindo uma bateria de testes automatizados, garantindo robustez e estabilidade.

5.3 Arquitetura proposta

A arquitetura do EncryVault está organizada por camadas, separando interface, lógica e armazenamento, de modo a tornar o projeto mais simples de manter e evoluir.

De forma geral, a aplicação é composta por:

- Interface (UI): ecrãs de criação e desbloqueio do cofre, lista de entradas, pesquisa/filtros, detalhe, criação/edição, documentos sigilosos, lixeira, Ajuda/FAQ e configurações.
- Lógica da aplicação: regras de navegação, validações, gestão de sessão, ordenação e filtragem, e gestão de estados do cofre.
- Camada de segurança: derivação de chave a partir da palavra-passe mestra (com *pepper* opcional), encriptação e desencriptação do cofre, e controlos adicionais como biometria, PINs, penalizações, cofre falso e auto-destruição.
- Armazenamento local: leitura e escrita do ficheiro do cofre no formato V3, gestão de blocos de documentos, exportação, importação e escrita atómica para evitar estados inconsistentes.

A Figura 36 representa a separação entre interface, serviços, modelos de domínio e persistência local do ficheiro do cofre.

Esta separação permitiu que alterações na interface não obrigassem a mexer diretamente na parte de segurança e armazenamento, e revelou-se essencial para acomodar o crescimento significativo do projeto ao longo do Projeto II.

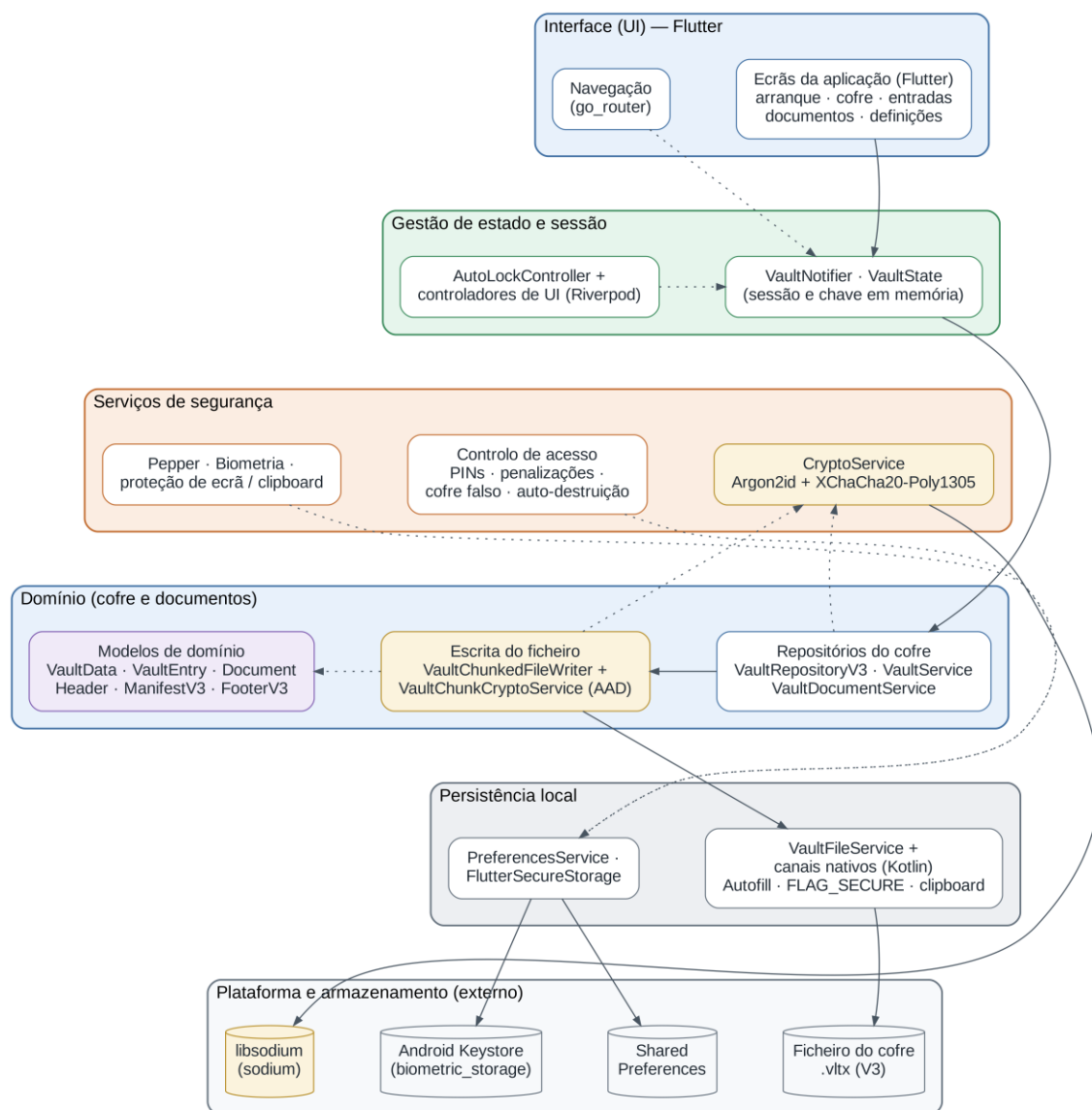


Figura 36 - Diagrama de componentes do EncryVault (organização por camadas e serviços).

5.4 Estrutura geral do cofre e das entradas

O cofre guarda um conjunto de entradas e de documentos sigilosos, sendo cada entrada composta por campos que permitem armazenar credenciais e informação adicional.

Cada entrada pode conter campos essenciais como:

- Título.
- Utilizador/Email.
- Palavra-passe.
- URL.
- Notas.

- *Tags*.
- Categoria e favorito.
- Datas e estatísticas de utilização (criação, edição, última abertura e número de aberturas) e histórico de palavras-passe.

O cofre é guardado localmente num ficheiro encriptado, sendo este ficheiro o elemento principal de persistência do sistema.

No Projeto II, a estrutura foi efetivamente estendida com estes campos adicionais, com a lixeira integrada no cofre (entradas e documentos com data de eliminação e retenção configurável) e com o suporte a documentos sigilosos guardados em blocos cifrados.

A Figura 37 representa o modelo de dados do cofre e das entradas.

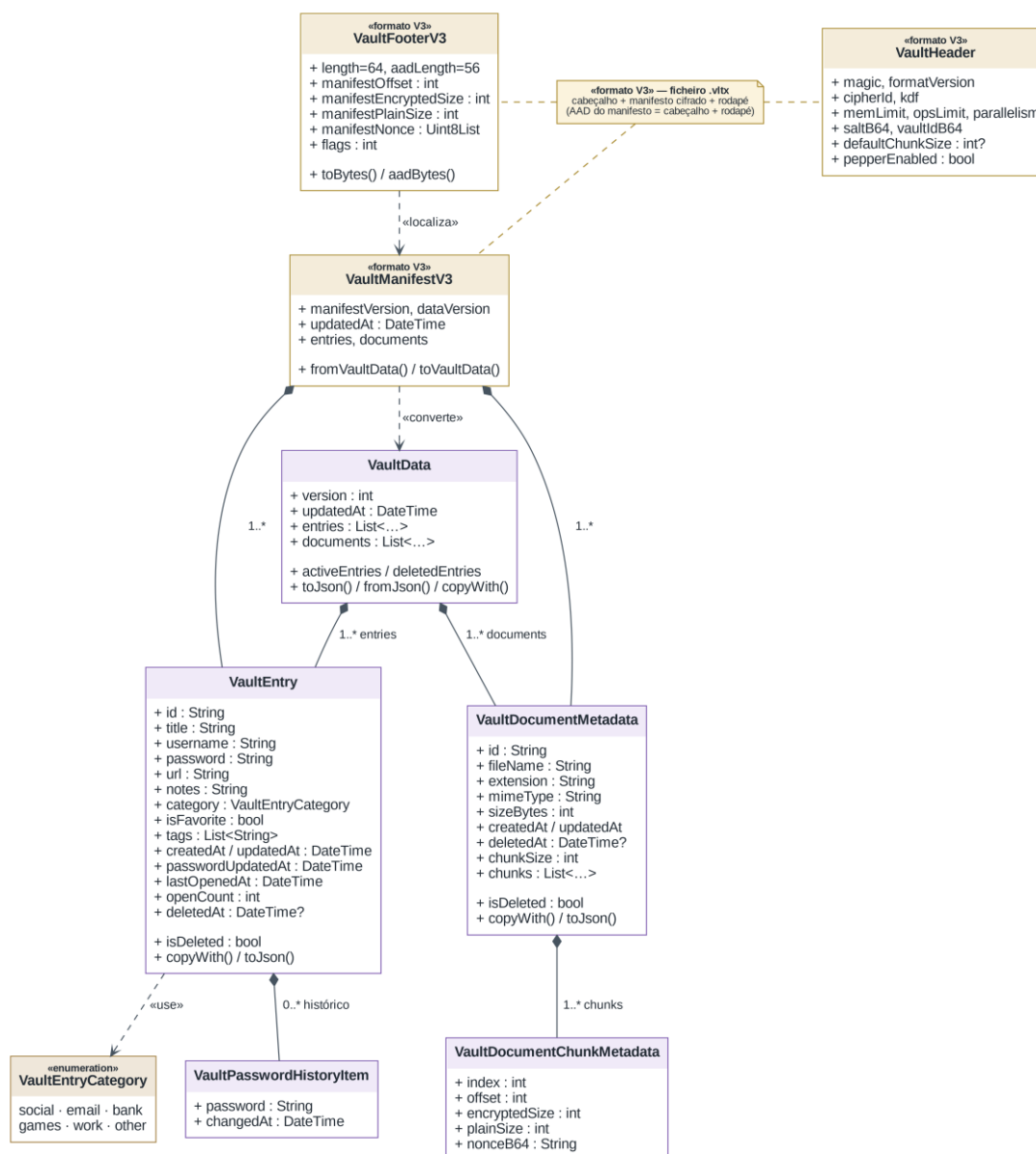


Figura 37 - Diagrama de classes do modelo do EncryVault (Cofre, Entradas e entidades relacionadas).

5.5 Fluxo geral de acesso e utilização do cofre

O fluxo de utilização do EncryVault inicia-se com a criação do cofre e definição da palavra-passe mestra, sendo o acesso futuro feito através do desbloqueio do cofre com essa palavra-passe.

A Figura 38 apresenta o fluxo de dados a alto nível entre o utilizador, a aplicação e o ficheiro do cofre. Esta visão resume entradas e saídas do sistema sem detalhar processos internos.

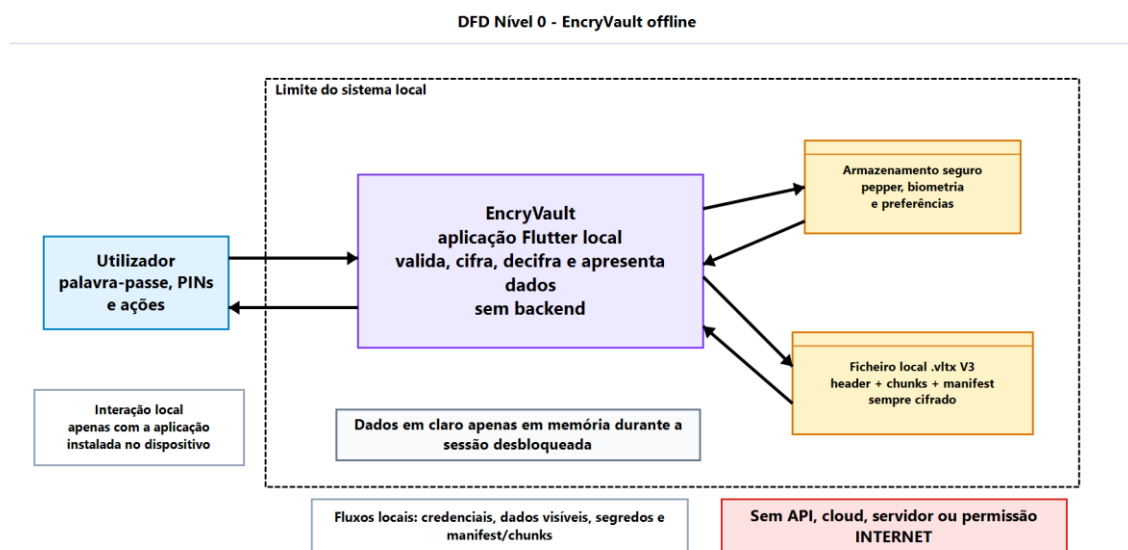


Figura 38 - DFD e fluxo de dados do EncryVault - Nível 0.

A Figura 39 detalha o fluxo de dados apresentado no nível 0, decompondo-o em processos internos principais. O diagrama clarifica onde ocorrem operações como desbloqueio, gestão de entradas e persistência do cofre.

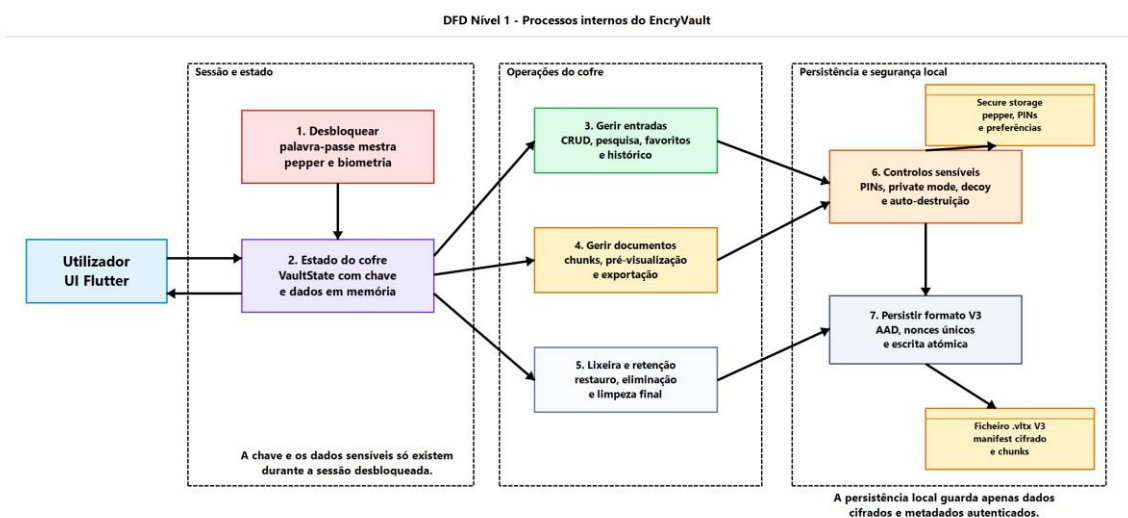


Figura 39 - DFD e fluxo de dados do EncryVault - Nível 1.

A Figura 40 ilustra o fluxo principal da aplicação, desde o arranque até ao acesso ao cofre, incluindo criação/importação e validação da palavra-passe mestra.

Após o desbloqueio, o utilizador pode consultar a lista de entradas, pesquisar, filtrar, criar entradas e aceder ao detalhe de cada registo, ficando o conteúdo disponível apenas durante a sessão ativa.

Quando ocorre inatividade, ou quando o utilizador bloqueia manualmente o cofre, a aplicação volta a exigir autenticação para permitir acesso ao conteúdo.

No Projeto II, este fluxo foi reforçado com autenticação biométrica, penalizações por tentativas falhadas, PINs de proteção em áreas e ações sensíveis, e mecanismos adicionais contra exposição visual e operacional do conteúdo (FLAG_SECURE, modo privado e limpeza automática do *clipboard*).

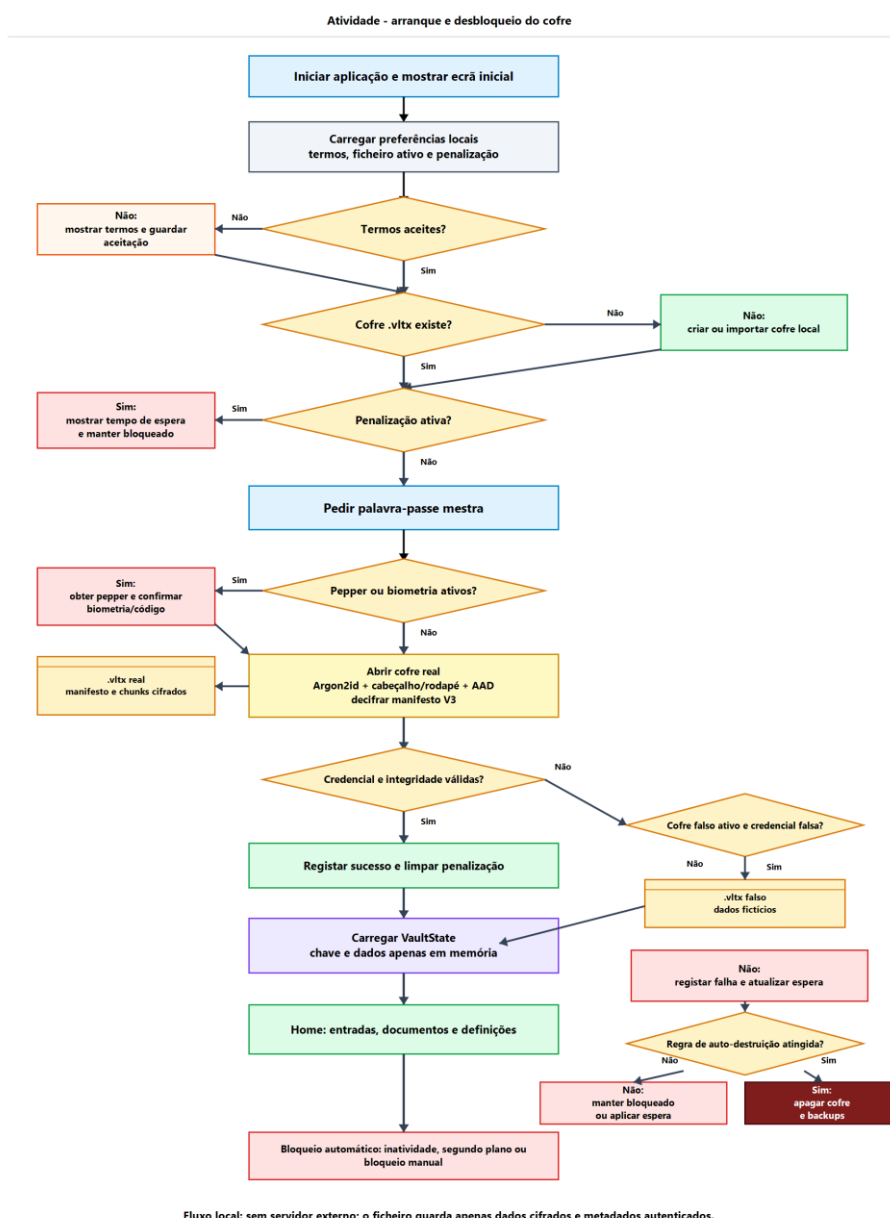


Figura 40 - Diagrama de atividade do fluxo de arranque e desbloqueio do cofre.

5.6 Formato V3 do ficheiro .vltx

A versão final do EncryVault utiliza um formato próprio de ficheiro, designado V3, desenvolvido no Projeto II para suportar documentos sigilosos de forma eficiente e garantir integridade verificável de todo o conteúdo.

O ficheiro .vltx está organizado da seguinte forma:

- Cabeçalho em claro (JSON) com metadados do formato: identificador (magic), versão, tipo de contentor, cifra, parâmetros da KDF (Argon2id), *salt*, identificador do cofre, tamanho por omissão dos blocos e indicação de utilização de *pepper*.
- Blocos (*chunks*) de documentos, cifrados individualmente com XChaCha20-Poly1305, permitindo ler e escrever documentos sem recifrar o cofre inteiro.
- Manifesto cifrado com os metadados das entradas, dos documentos e da lista de blocos.
- Rodapé fixo de 64 *bytes* com *offsets*, tamanhos, *nonce* do manifesto e *flags*, permitindo localizar as secções do ficheiro de forma determinística.
- *Nonces* únicos por bloco cifrado, gerados aleatoriamente.

O cabeçalho, o rodapé e os metadados de cada bloco são autenticados como dados associados (AAD) na cifragem, pelo que qualquer adulteração destes campos, mesmo estando em claro, é detetada e o ficheiro é rejeitado.

As operações de gravação utilizam escrita atómica (ficheiro temporário, flush e substituição), evitando estados intermédios em caso de interrupção durante a escrita.

6. Design

Neste capítulo apresenta-se o design da aplicação móvel EncryVault na sua versão final, descrevendo a organização geral dos ecrãs e os fluxos principais de navegação.

São apresentados os ecrãs da aplicação e o estudo efetuado relativamente à interface, de forma a garantir uma UI simples, consistente e adequada a uma aplicação de segurança, incluindo as variações de tema (claro/escuro) e os dois modos visuais (Moderno e Clássico) introduzidos no Projeto II.

6.1 Ecrãs da aplicação

Neste subcapítulo pretende-se demonstrar, por intermédio de prints da aplicação, qual o aspeto geral da versão final e como se distribuem os ecrãs principais.

A aplicação apresenta o fluxo de utilização desde o arranque até ao acesso ao cofre, incluindo criação de cofre, desbloqueio, listagem de entradas, pesquisa, gestão de documentos sigilosos, lixeira e configurações.

De forma geral, os ecrãs principais da versão final são:

- Ecrã de arranque (*splash screen*).
- Termos e condições (primeira execução).
- Ecrã inicial (criar/importar/entrar).
- Criação de cofre e definição de palavra-passe mestra.
- Desbloqueio do cofre.
- Página principal do cofre (lista de entradas).
- Pesquisa e filtragem.
- Visualização de uma entrada.
- Criação/Edição de uma entrada.
- Configurações do cofre (incluindo exportação/importação).
- Documentos sigilosos (lista, pré-visualização e gestão).
- Lixeira do cofre (entradas e documentos).
- Página de Ajuda/FAQ com pesquisa.

Os prints incluídos neste capítulo permitem validar o layout e o fluxo de navegação, garantindo que a estrutura do EncryVault se manteve consistente ao longo da evolução do Projeto II.

As Figuras 3 a 35 apresentam os ecrãs principais da aplicação e exemplificam o fluxo geral de navegação, desde o arranque até à gestão de entradas no cofre.

Importa notar que todas as capturas de ecrã apresentadas neste relatório correspondem à versão final do Projeto II, refletindo o design e as funcionalidades atuais, que evoluíram significativamente face às do Projeto I.

6.2 Elementos de interface

Neste estudo pretende-se identificar os elementos gráficos que compõem a interface da aplicação, garantindo uma utilização simples e intuitiva.

Sendo o EncryVault uma aplicação de cofre digital, é natural que a interface privilegie clareza e rapidez, reduzindo elementos visuais desnecessários e mantendo ações principais bem identificadas, recorrendo a componentes familiares como botões, ícones, menus, *switches* e *sliders*.

De forma geral, os principais elementos gráficos da UI incluem:

- Ícones associados a ações rápidas, como copiar, editar, apagar e mostrar/ocultar palavra-passe.
- Campos de introdução de texto com validação (por exemplo, título, utilizador/email, palavra-passe e notas).
- Alternadores (*toggle*) e *sliders* em configurações (por exemplo, opções de segurança e tempo de *auto-lock*).
- Mensagens de confirmação e avisos para operações sensíveis (por exemplo, apagar, exportar e importar).
- *Feedback* visual curto (por exemplo, notificações) após ações como copiar e guardar.

No Projeto II, este estudo foi reforçado com melhorias de consistência visual e personalização, como modo claro/escuro/automático, dois modos visuais (Moderno e Clássico), modo privado e organização adicional (categorias, favoritos, lixeira e detalhes de utilização).

Os elementos visuais referidos (botões, ícones, campos de texto, listas, pesquisa e opções de configuração) são visíveis nas Figuras 3 a 35, garantindo consistência e simplicidade na utilização da aplicação.



Figura 41 - Ícone do EncryVault.



Figura 42 - Imagem de arranque do EncryVault.

A figura 41 e 42 foram utilizados como elementos de identidade visual da aplicação, reforçando a associação entre o nome do projeto e a ideia de cofre digital seguro. As suas utilizações contribuem para tornar a aplicação mais reconhecível e consistente, aparecendo associada à apresentação visual do produto e ao primeiro contacto do utilizador com a aplicação.

7. Desenvolvimento

Neste capítulo descrevem-se os aspetos principais do desenvolvimento do EncryVault ao longo do Projeto II: a metodologia adotada, as ferramentas e tecnologias utilizadas, os recursos necessários, a forma como a aplicação funciona internamente e as principais dificuldades encontradas, ficando a validação detalhada para o Capítulo 8.

Este capítulo serve também para clarificar como a versão demonstrativa do Projeto I evoluiu para a versão final, mantendo coerência com os requisitos e a arquitetura definidos no Capítulo 5.

7.1 Metodologia de desenvolvimento

A metodologia seguida para o EncryVault baseia-se numa abordagem iterativa, assente em prototipagem, implementação incremental e melhoria contínua.

No Projeto I, esta abordagem permitiu avançar com uma versão demonstrativa com funcionalidades nucleares, validar o fluxo de navegação e garantir uma estrutura preparada para crescer.

No Projeto II, a evolução foi orientada pela consolidação do que já estava desenvolvido, pela introdução progressiva das funcionalidades planeadas (e das adaptações descritas na secção 5.1.3) e pela realização contínua de testes e validações.

Na prática, cada iteração passou por três fases principais: ajuste dos requisitos imediatos, implementação de um conjunto pequeno e bem definido de funcionalidades e validação do resultado antes de avançar para a iteração seguinte.

7.2 Ferramentas utilizadas

Na elaboração deste projeto foram utilizadas as seguintes ferramentas:

- Visual Studio Code.
- Flutter SDK.
- Dart.
- Kotlin.
- Android Studio.
- Git e GitHub.
- Draw.io.
- Moqups.

Visual Studio Code: Foi utilizado como editor principal para desenvolvimento da aplicação, permitindo escrever e organizar o código, gerir o projeto e trabalhar de forma rápida com extensões úteis para Flutter e Dart.

Flutter SDK: Foi utilizado como *framework* para desenvolvimento da aplicação móvel, permitindo criar a interface e a lógica da aplicação numa base única, com foco em desempenho e consistência visual [8], [9].

Dart: Foi a linguagem usada para implementar a aplicação, sendo a linguagem principal do Flutter e permitindo criar toda a lógica do EncryVault de forma estruturada [10].

Kotlin: Foi utilizado na componente nativa Android, em particular no serviço de preenchimento automático (*autofill*) e na integração com funcionalidades do sistema (proteção de ecrã e limpeza do *clipboard*) através de *method channels*.

Android Studio: Foi utilizado para gestão do ambiente Android, em especial para emuladores, SDKs e testes em execução, facilitando também a validação da aplicação em contexto Android.

GitHub: Foi usado para controlo de versões do projeto, permitindo registar alterações, manter histórico do desenvolvimento e garantir organização e segurança do trabalho ao longo do tempo.

Draw.io: Foi utilizado para criar diagramas e esquemas simples que ajudam a explicar a estrutura do sistema e algumas partes do funcionamento do projeto.

Moqups: Foi utilizado para criar wireframes e mockups das principais páginas da aplicação, ajudando a definir o aspeto visual e os fluxos antes e durante a implementação.

7.3 Linguagens e tecnologias de desenvolvimento

O EncryVault é desenvolvido como uma aplicação móvel em Flutter, utilizando a linguagem Dart na aplicação e Kotlin na componente nativa Android, permitindo criar uma interface consistente e moderna e manter um funcionamento totalmente *offline* [8], [10].

A escolha do Flutter permite construir a UI com componentes reutilizáveis e manter uma separação entre interface, lógica e persistência. A gestão de estado é feita com Riverpod e a navegação com *go_router*, mantendo os ecrãs desacoplados dos serviços e facilitando a manutenção e a evolução do projeto ao longo do Projeto II.

A aplicação utiliza armazenamento local através de um ficheiro de cofre em formato próprio (V3, descrito na secção 5.6), suportando entradas, documentos sigilosos e lixeira, bem como operações de importação e exportação para *backup*, garantindo que o conteúdo nunca é guardado em texto simples.

Para proteção do cofre é aplicada uma abordagem baseada em palavra-passe mestra, com derivação de chave (Argon2id, com *pepper* opcional) e cifragem autenticada do conteúdo (XChaCha20-Poly1305), permitindo que o ficheiro do cofre permaneça ilegível e à prova de adulteração sem a credencial correta.

7.4 Recursos

Para o desenvolvimento do EncryVault foram necessários recursos de apoio à implementação, teste e validação, garantindo evolução estruturada ao longo do Projeto II.

A implementação foi suportada por ferramentas de desenvolvimento, ambiente Android para testes (emulador e/ou dispositivo físico) e controlo de versões para manter histórico do projeto e facilitar correções e melhorias.

Foram também utilizados recursos de prototipagem e documentação para preparar a organização dos ecrãs, fluxos principais e estrutura geral do sistema, de forma a garantir consistência desde cedo.

De forma geral, a aplicação recorre a bibliotecas/pacotes para operações como gestão de ficheiros locais, armazenamento seguro e autenticação biométrica (`local_auth` [21] e `biometric_storage` [22]), avaliação de força de palavras-passe (`password_strength_checker` [23]), criptografia (`sodium`), manipulação e validação de dados, e componentes de interface e localização.

Em termos de execução, a aplicação destina-se a dispositivos Android 7.0 (API 24) ou superior. Um aspeto relevante do ponto de vista da privacidade é o conjunto mínimo de permissões pedidas: o EncryVault declara apenas a permissão de utilização de biometria (`USE_BIOMETRIC`) e não declara a permissão de acesso à Internet, pelo que a aplicação é tecnicamente incapaz de comunicar com servidores externos. As cópias de segurança automáticas do sistema operativo estão também desativadas, evitando que o ficheiro do cofre seja incluído em *backups* externos sem controlo do utilizador.

O detalhe do manifesto Android e do modelo de permissões é apresentado no Anexo C.

Para distribuição, a aplicação é compilada num modo otimizado: o código é minificado e ofuscado e os recursos não utilizados são removidos (R8/ProGuard), o pacote é assinado com uma chave própria (*keystore*) e são geradas variantes apenas para as arquiteturas de processador mais comuns em dispositivos Android atuais (`armeabi-v7a` e `arm64-v8a`). O identificador da aplicação é `com.encryvault.app` e a versão mínima suportada corresponde ao Android 7.0 (API 24).

7.5 Modulação e funcionamento do EncryVault

Nesta secção descreve-se o funcionamento lógico do EncryVault, apresentando como os módulos se relacionam e como o cofre é gerido ao longo do ciclo de utilização.

De forma geral, a aplicação pode ser entendida em quatro blocos principais:

- Interface (UI): ecrãs de arranque, fluxo inicial, criação/desbloqueio do cofre, lista de entradas, pesquisa/filtros, detalhe de entrada, criação/edição, documentos sigilosos, lixeira, Ajuda/FAQ e configurações.
- Lógica da aplicação: regras de validação, criação/edição de entradas, pesquisa, filtragem, ordenação, e controlo de ações sensíveis (copiar, apagar, exportar, importar).
- Camada de segurança: proteção do cofre com palavra-passe mestra, derivação de chave (com *pepper* opcional) e encriptação/desencriptação do conteúdo, controlo de sessão, biometria, PINs, penalizações, cofre falso e auto-destruição.
- Persistência local: leitura e escrita do ficheiro do cofre no formato V3, gestão de blocos de documentos, exportação e importação, e escrita atómica do ficheiro.

7.5.1 Casos de uso principais

Os casos de uso mais relevantes no EncryVault incluem:

- Criar um cofre e definir a palavra-passe mestra.
- Desbloquear o cofre com a palavra-passe mestra.
- Criar, consultar, editar e remover entradas no cofre.
- Pesquisar entradas e aplicar filtros (por exemplo, por *tags*).
- Gerar palavra-passe durante a criação/edição de entradas.
- Exportar cofre para *backup* e importar cofre a partir de ficheiro.
- Bloquear o cofre manualmente e aplicar *auto-lock* após inatividade.
- Guardar, pré-visualizar e exportar documentos sigilosos.
- Proteger áreas e ações sensíveis com PINs (globais ou individuais) e biometria.
- Configurar cofre falso e auto-destruição.
- Utilizar o preenchimento automático (*autofill*) noutras aplicações Android.
- Consultar a saúde das palavras-passe e o histórico de palavras-passe de cada entrada.

A Figura 43 apresenta os casos de uso principais do EncryVault, evidenciando as ações centrais do utilizador sobre o cofre, desde a criação e desbloqueio até à gestão de entradas, pesquisa, bloqueio de sessão e operações de importação/exportação.

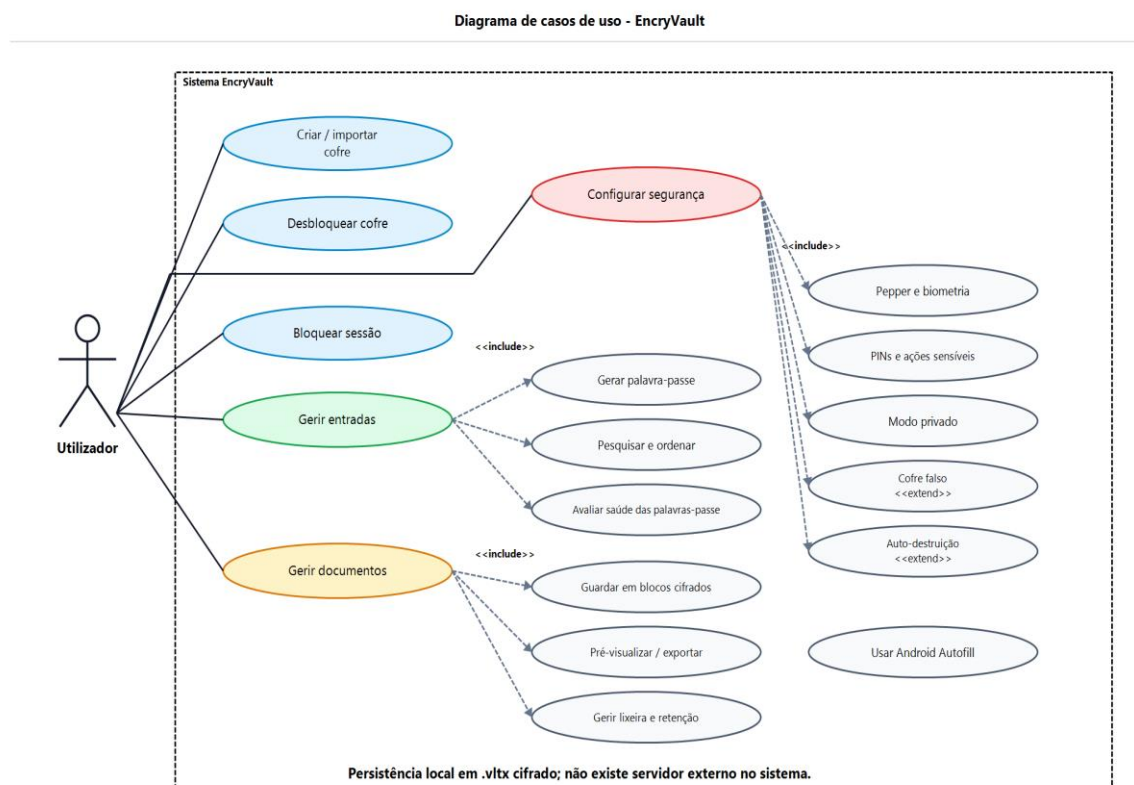


Figura 43 - Diagrama de casos de uso do EncryVault.

7.5.2 Proteção do cofre e fluxo de encriptação

O cofre é protegido para garantir que os dados armazenados localmente não ficam acessíveis em texto simples.

De forma geral, após a definição da palavra-passe mestra, a aplicação deriva uma chave através de Argon2id (combinando opcionalmente um *pepper* guardado no armazenamento seguro do dispositivo) e encripta o conteúdo do cofre com XChaCha20-Poly1305, sendo que em cada desbloqueio o processo é repetido para validar a credencial e decifrar os dados. Os parâmetros do Argon2id seguem os perfis recomendados do libsodium (limite de memória de nível *sensitive* e número de operações de nível *moderate*) e ficam registados no cabeçalho do ficheiro, garantindo que o cofre é sempre aberto com os mesmos parâmetros com que foi criado.

Este modelo permite que o ficheiro do cofre possa ser guardado no armazenamento do dispositivo com menor risco de exposição, mantendo o utilizador com controlo local sobre os seus dados.

No formato V3, o manifesto e cada bloco de documento são cifrados individualmente, e o cabeçalho, o rodapé e os metadados dos blocos são autenticados como dados associados (AAD), pelo que qualquer adulteração do ficheiro - mesmo nas partes em claro - é detetada na abertura.

A Figura 44 descreve a sequência do desbloqueio do cofre, desde a introdução da palavra-passe até ao carregamento dos dados. O diagrama evidencia os pontos críticos onde são aplicadas validações e onde pode ocorrer rejeição do desbloqueio.

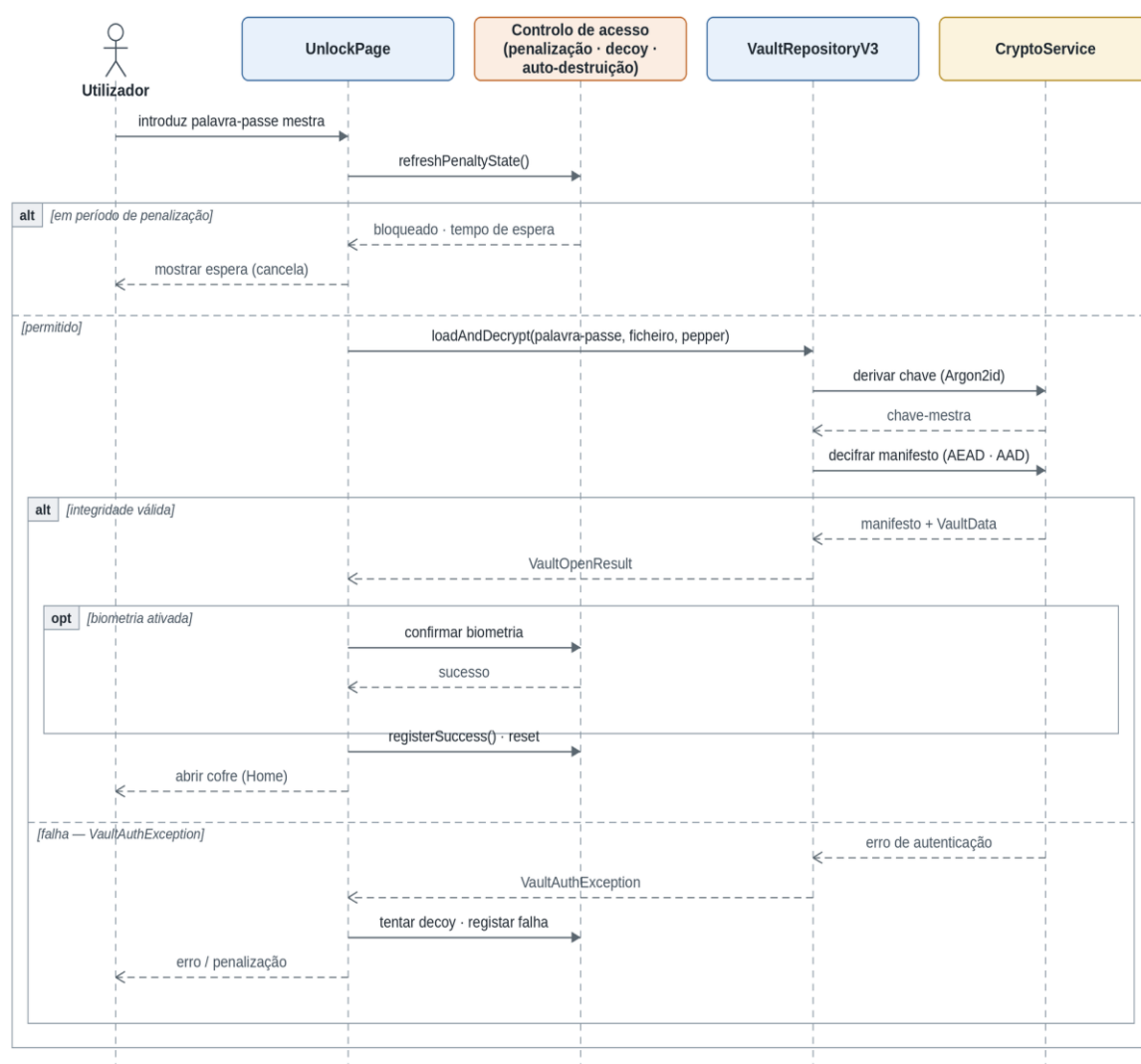


Figura 44 - Diagrama de sequência do desbloqueio do cofre (*Unlock*).

A Figura 45 descreve a sequência de criação do cofre, incluindo a geração de valores aleatórios e o primeiro processo de cifragem. Este fluxo define a base de persistência segura do cofre no ficheiro .vltx.

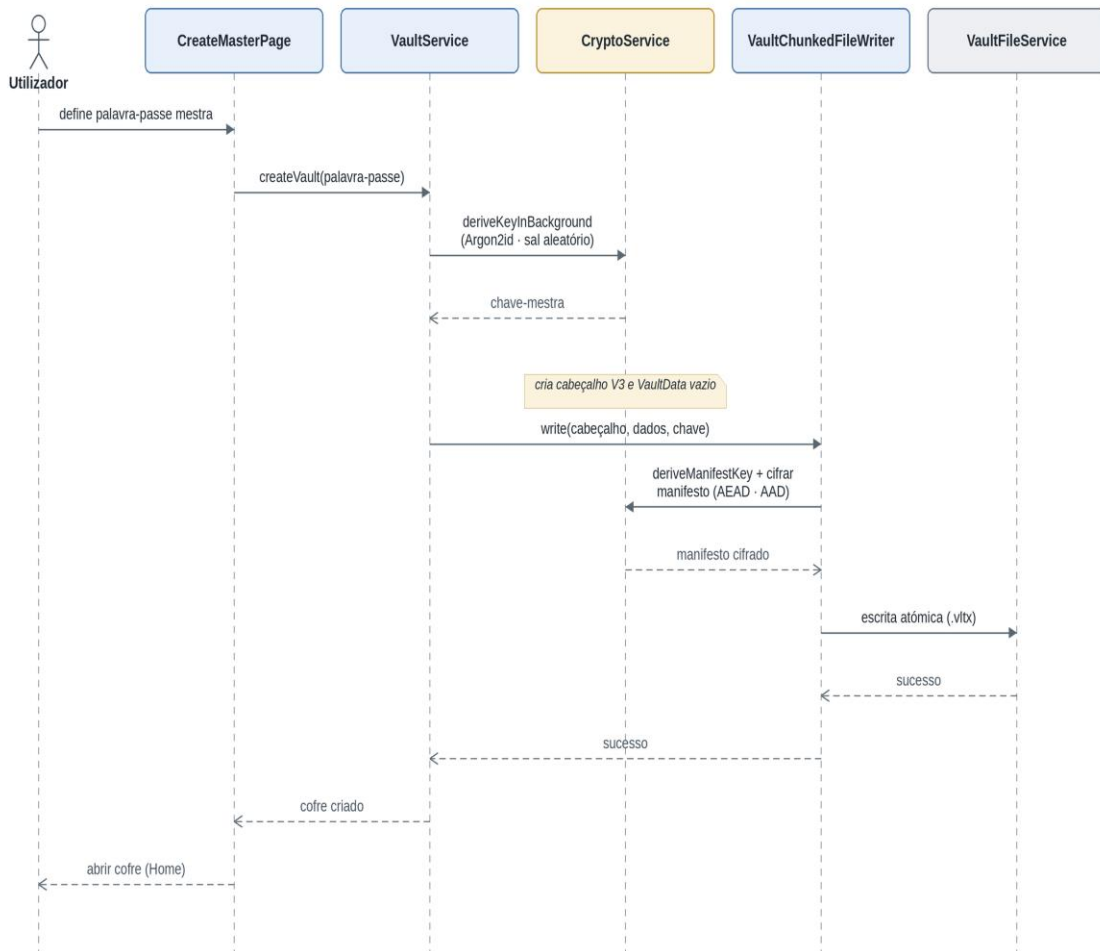


Figura 45 - Diagrama de sequência da criação do cofre (*Create Vault*).

A Figura 46 descreve o fluxo de encriptação e descriptação do cofre. O processo inclui a derivação da chave a partir da palavra-passe mestra, a cifragem autenticada do conteúdo e a validação de integridade no momento da abertura. Desta forma, qualquer ficheiro adulterado é rejeitado e o cofre só é carregado quando a palavra-passe e os parâmetros são compatíveis.

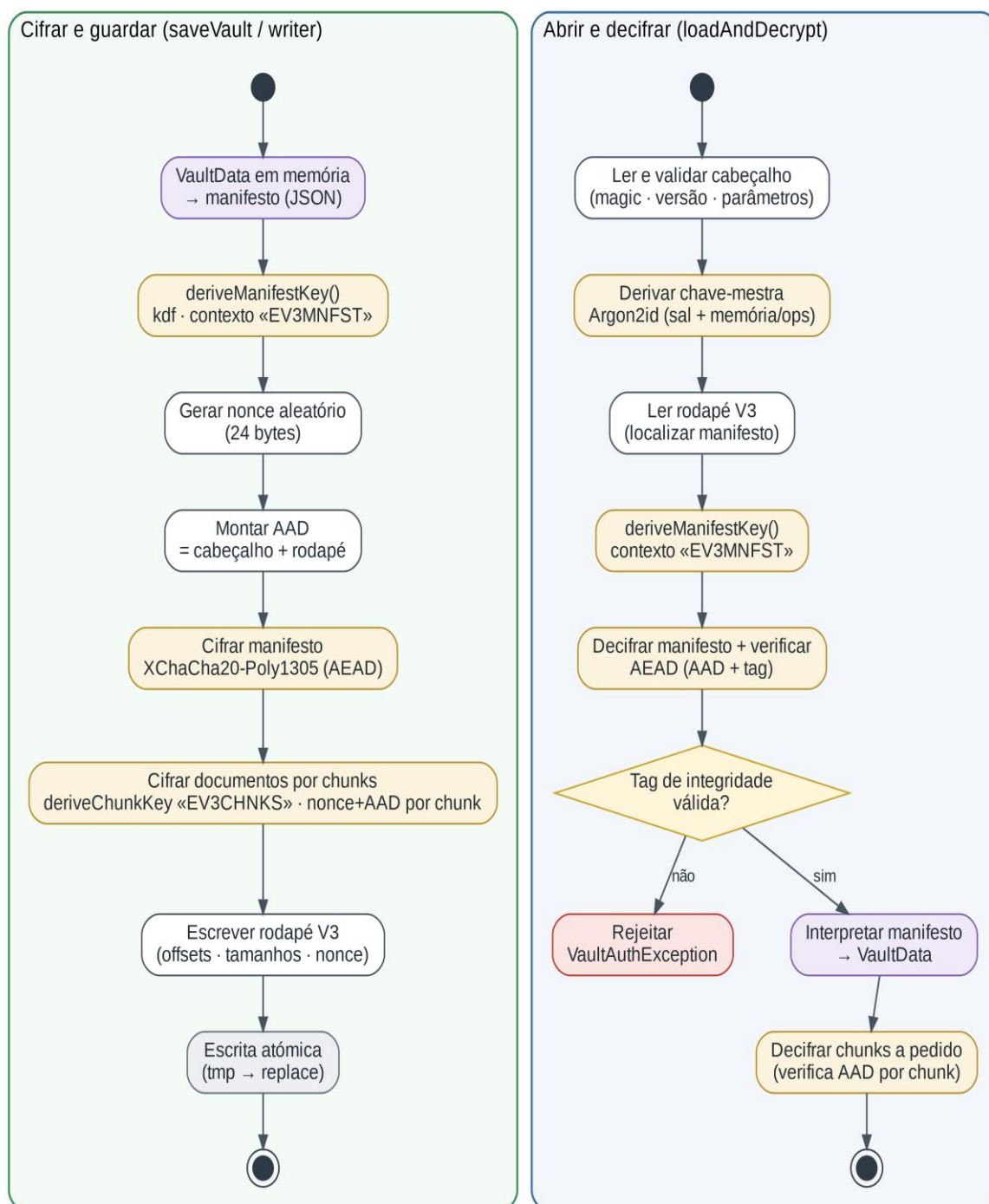
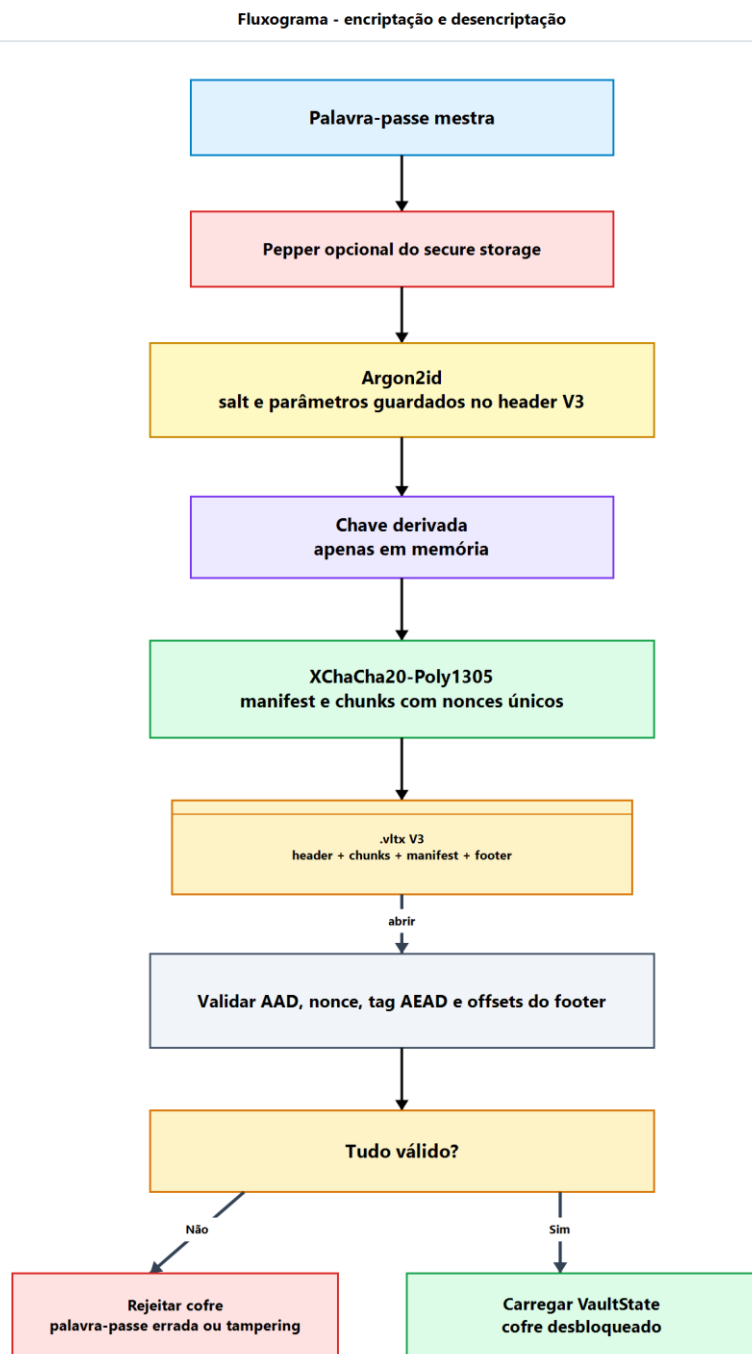


Figura 46 - Diagrama de atividade do fluxo de encriptação e descriptação do cofre.

A Figura 47 apresenta um fluxograma simplificado do processo de cifragem e decifragem do cofre. Esta representação resume as etapas principais e ajuda a visualizar a sequência geral antes de avançar para a gestão de sessão e *auto-lock*.



O modelo é totalmente local: sem servidor, sincronização remota ou dados em texto claro no ficheiro.

Figura 47 - Fluxograma de alto nível do processo de encriptação/desencriptação.

7.5.3 Gestão de sessão e *auto-lock*

Durante a utilização do cofre, o conteúdo deve permanecer acessível apenas enquanto a sessão está ativa.

Quando ocorre inatividade durante um intervalo definido, quando a aplicação vai para segundo plano, ou quando o utilizador escolhe bloquear manualmente, a aplicação volta a exigir desbloqueio para permitir acesso ao conteúdo, podendo o utilizador ajustar o tempo de *auto-lock* entre 1 e 30 minutos (predefinição de 2 minutos) ou desativar o bloqueio automático nas definições.

A existência deste mecanismo reforça a segurança sem comprometer demasiado a usabilidade, mantendo a experiência de utilização simples e direta.

A Figura 48 ilustra os estados do cofre e as transições entre bloqueado e desbloqueado. As transições são motivadas por ações do utilizador e por eventos como inatividade ou envio da aplicação para segundo plano. Esta representação ajuda a compreender o comportamento do *auto-lock* e o controlo de sessão.

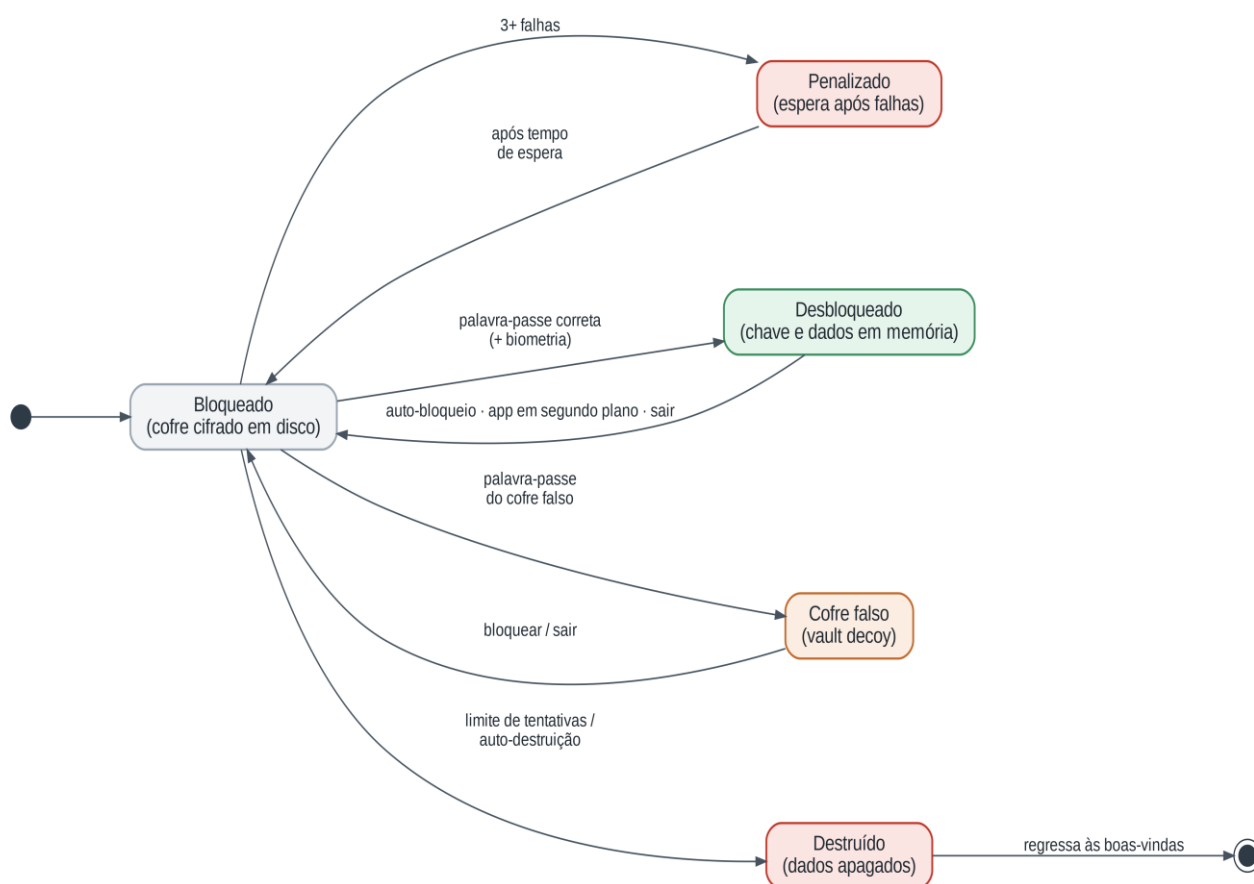


Figura 48 - Diagrama de estados do cofre e da sessão (*Locked/Unlocked/Auto-lock*).

7.5.4 Gestão de entradas, pesquisa e organização

As entradas representam a unidade base de informação do cofre, permitindo guardar credenciais e informação complementar.

A aplicação disponibiliza operações de criação, edição, remoção (para a lixeira) e consulta, bem como pesquisa para acesso rápido a registos, complementada por filtros (*tags* e categorias), favoritos, múltiplos critérios de ordenação e modo privado, garantindo organização mesmo com muitas entradas.

A funcionalidade de geração de palavra-passe apoia a criação de credenciais mais fortes, complementada na versão final pela área de saúde das palavras-passe, pelas recomendações de alteração e pelo histórico de palavras-passe de cada entrada, reduzindo os problemas associados a *passwords* fracas ou repetidas.

A Figura 49 apresenta a sequência de adição de uma entrada ao cofre. O fluxo evidencia a atualização dos dados em memória e a persistência segura após a operação.

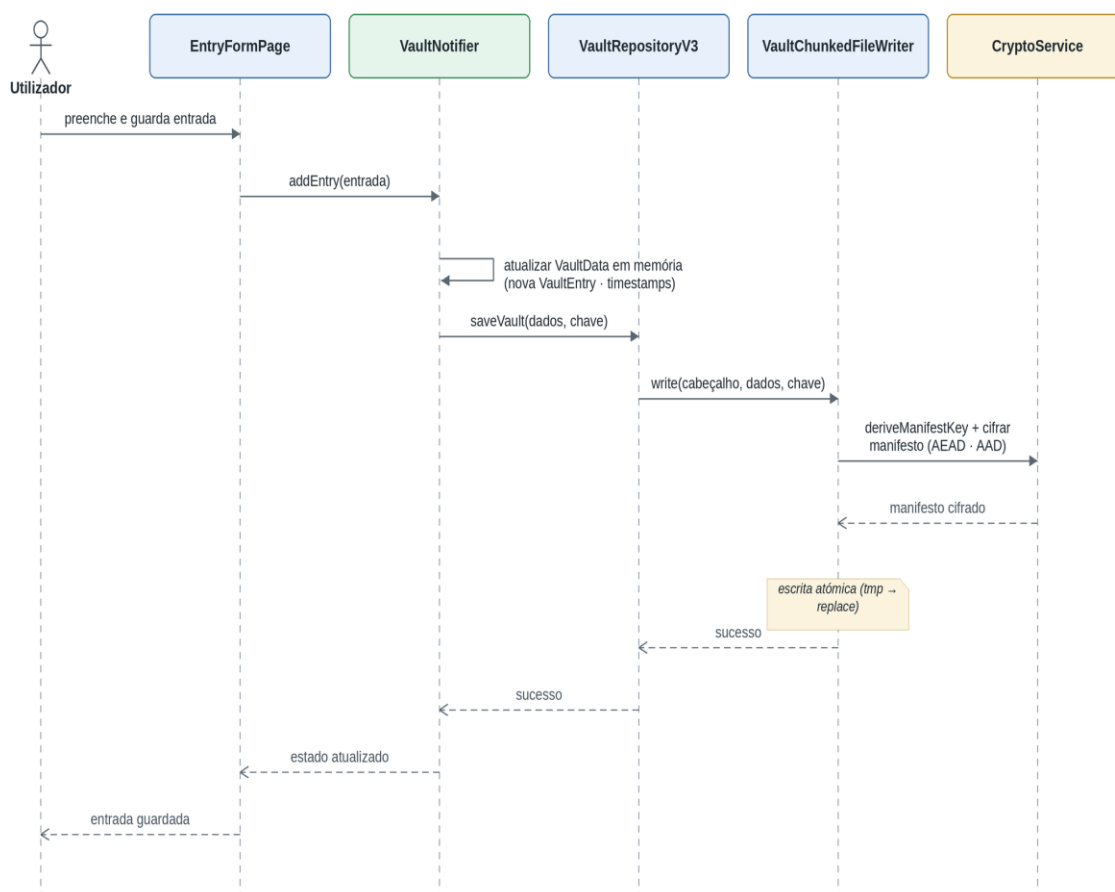


Figura 49 - Diagrama de sequência da adição de uma entrada ao cofre.

7.5.5 Importação e exportação do cofre

A exportação permite criar uma cópia do cofre para *backup*, garantindo capacidade de recuperação em caso de perda do dispositivo ou necessidade de transferência.

A importação permite restaurar um cofre previamente guardado, mantendo as entradas e as configurações associadas, desde que o utilizador tenha a credencial necessária para o desbloqueio do cofre.

Este mecanismo é essencial num gestor *offline*, porque a recuperação não depende de serviços externos, mas sim do controlo do próprio ficheiro de cofre.

Na importação, o ficheiro é validado antes de ser aceite: o cabeçalho e o rodapé são verificados (formato, versão e parâmetros), e a integridade do conteúdo é confirmada pela cifragem autenticada, sendo rejeitados ficheiros corrompidos, truncados ou adulterados, sem criar estados inconsistentes.

7.5.6 Documentos sigilosos

Os documentos sigilosos permitem guardar ficheiros (por exemplo, documentos de identificação ou comprovativos) de forma cifrada dentro do próprio cofre. Cada documento é dividido em blocos (*chunks*) de 2 MB por omissão, cifrados individualmente, com limites de 100 MB por documento e 500 MB no total.

A aplicação suporta pré-visualização de documentos até 10 MB (incluindo visualização de PDF), exportação para fora do cofre mediante confirmação, e eliminação com passagem por uma lixeira própria, sujeita à mesma política de retenção do cofre.

7.5.7 Proteções avançadas

Para além da proteção base do cofre, a versão final inclui um conjunto de proteções avançadas opcionais: PINs globais e individuais com atraso progressivo (*backoff*) e recuperação por palavra-passe mestra; *pepper* guardado no armazenamento seguro com confirmação biométrica e código de recuperação; e confirmação explícita de ações sensíveis.

Em cenários de coação, o cofre falso permite que uma palavra-passe alternativa abra um cofre com conteúdo fictício, e a auto-destruição (através de um PIN dedicado ou de um limite de tentativas falhadas) elimina de forma irreversível o cofre real, os *backups* e os segredos do armazenamento seguro.

Complementarmente, o modo privado oculta nomes e dados sensíveis na interface, o *FLAG_SECURE* impede capturas de ecrã e a limpeza nativa do *clipboard* remove as credenciais copiadas ao sair do ecrã da entrada e quando a

aplicação regressa de segundo plano, apenas quando o conteúdo ainda pertence à aplicação.

7.5.8 Autofill no Android

O EncryVault implementa um serviço nativo de preenchimento automático (*autofill*) para Android [20], desenvolvido em Kotlin, que sugere credenciais do cofre noutras aplicações e páginas *web*.

A correspondência é feita por identificador da aplicação ou por URL, e o preenchimento exige autenticação prévia (desbloqueio do cofre), sendo a funcionalidade totalmente opcional e ativada explicitamente pelo utilizador nas definições do sistema.

7.6 Versão demonstrativa à versão final

No final do Projeto I existia uma versão demonstrativa do EncryVault com as funcionalidades essenciais (cofre local, gestão de entradas, pesquisa, *auto-lock* e exportação/importação), composta por cerca de 30 ficheiros Dart, aproximadamente 2 mil linhas de código e 2 ficheiros de teste.

No Projeto II foi realizada a conclusão e consolidação do desenvolvimento, incluindo o novo formato V3, documentos sigilosos, proteções avançadas, *autofill* nativo e melhorias de experiência de utilização, resultando numa base de código com cerca de 115 ficheiros Dart, aproximadamente 28 mil linhas e 28 ficheiros de teste automatizado, além de componentes nativos em Kotlin.

7.7 Dificuldades e soluções adotadas

Ao longo do Projeto II surgiram dificuldades técnicas que obrigaram a iterar soluções e a rever decisões. As principais foram as seguintes:

Formato V3 e documentos em blocos - garantir a leitura e a escrita de documentos grandes sem carregar o ficheiro completo em memória e sem recifrar o cofre inteiro exigiu várias iterações ao desenho do formato. A solução passou por blocos de 2 MB cifrados individualmente, um rodapé fixo com *offsets* para localizar as secções e escrita atómica do ficheiro.

Integridade com dados associados (AAD) - aplicar corretamente o AAD na cifragem, abrangendo o cabeçalho, o rodapé e os metadados dos blocos, revelou-se sensível a pequenos erros de ordem e de serialização. A validação foi consolidada com testes dedicados de adulteração e corrupção do ficheiro.

Pepper com biometria e recuperação - o *pepper* introduz vários estados possíveis (ativado, biometria indisponível, dispositivo novo), e foi necessário garantir que o utilizador nunca fica permanentemente impedido de aceder ao

cofre. A solução foi um código de recuperação gerado na ativação e fluxos de confirmação explícitos.

Autofill nativo no Android - a integração entre o Flutter e a componente nativa em Kotlin, o ciclo de vida do serviço de *autofill* e a exigência de autenticação antes do preenchimento levaram à criação de uma atividade de autenticação dedicada e à comunicação por *method channels*.

Cofre falso e auto-destruição sem riscos - foi necessário garantir que o cofre falso é indistinguível do real para um observador e que a auto-destruição elimina todos os dados de forma irreversível, sem nunca poder ser acionada por engano. Estas funcionalidades foram validadas com cautela em ambiente controlado e exigem confirmações explícitas na configuração.

Comportamento nativo entre versões do Android - funcionalidades como o FLAG_SECURE e a limpeza do *clipboard* variam entre versões do sistema; a limpeza foi implementada ao nível nativo e apenas atua quando o conteúdo copiado ainda pertence à aplicação.

8. Testes e validação

Este capítulo descreve como foi verificado o correto funcionamento do EncryVault, assegurando que as funcionalidades principais operam como esperado e que o cofre, no formato V3, mantém confidencialidade e integridade mesmo em cenários de falha ou adulteração.

8.1 Abordagem de validação

A validação do EncryVault é composta por:

- Testes automatizados (flutter test), com 28 ficheiros de teste que totalizam cerca de 188 casos de teste, cobrindo os serviços principais (formato do cofre, penalizações de desbloqueio, saúde das palavras-passe, lixeira, documentos, ordenação, entre outros).
- Testes funcionais manuais, executados através da interface da aplicação, cobrindo os fluxos principais e as funcionalidades novas do Projeto II.
- Testes de robustez, focados em corrupção/adulteração do ficheiro do cofre e cenários de falha, e testes de consistência e persistência após reinício.

Sempre que possível, os testes foram repetidos em:

- Emulador Android.
- Dispositivo físico.

8.2 Contexto técnico relevante para os testes

Os testes assumem o formato V3 do ficheiro .vltx, descrito em detalhe na secção 5.6: cabeçalho em claro com os metadados do formato, blocos de documentos e manifesto cifrados com XChaCha20-Poly1305, rodapé fixo de localização e elementos estruturais autenticados como dados associados (AAD), com derivação de chave Argon2id (com *pepper* opcional) e escrita atômica nas operações de atualização [24].

Estas propriedades tornam detetável, logo na abertura, qualquer corrupção ou adulteração do ficheiro, e enquadram diretamente os testes de robustez e integridade apresentados na secção 8.6.

8.3 Testes funcionais (casos principais)

Os testes seguintes (8.3 a 8.6) cobrem as funcionalidades base consolidadas do Projeto I, que é a criação e desbloqueio do cofre, gestão de entradas, pesquisa, sessão e *auto-lock*, importação/exportação e robustez do ficheiro, que são mantidas e revalidadas na versão final. As funcionalidades introduzidas no Projeto II são validadas separadamente na secção 8.7.

Teste 01 - Criação de cofre

Passos: criar cofre definindo palavra-passe mestra.

Resultado esperado: o ficheiro do cofre é criado; a aplicação confirma sucesso; é possível prosseguir para desbloqueio.

Teste 02 - Desbloqueio com palavra-passe correta

Passos: introduzir a palavra-passe correta no ecrã de desbloqueio.

Resultado esperado: o cofre abre; o ecrã principal carrega; as entradas são apresentadas.

Teste 03 - Desbloqueio com palavra-passe incorreta

Passos: introduzir palavra-passe errada.

Resultado esperado: o desbloqueio falha; mensagem clara; não existe acesso parcial a dados.

Teste 04 - Criação de entrada

Passos: criar uma entrada com título, utilizador e palavra-passe.

Resultado esperado: a entrada aparece na lista e mantém-se após reiniciar e desbloquear novamente.

Teste 05 - Edição de entrada

Passos: editar campos de uma entrada e gravar.

Resultado esperado: alterações persistem e são refletidas na lista/detalhe.

Teste 06 - Remoção de entrada

Passos: apagar uma entrada e confirmar.

Resultado esperado: a entrada deixa de existir e continua removida após reinício.

Teste 07 - Pesquisa

Passos: pesquisar por texto (ex.: título/utilizador).

Resultado esperado: filtragem correta; pesquisa não altera dados do cofre.

Teste 08 - Copiar campos (*clipboard*)

Passos: copiar utilizador/palavra-passe; aguardar o tempo definido.

Resultado esperado: o valor é copiado; a aplicação minimiza exposição e limpa o *clipboard* quando aplicável.

Teste 09 - Bloqueio manual

Passos: acionar opção “Bloquear” no cofre.

Resultado esperado: regressa ao ecrã de desbloqueio; não é possível aceder a dados sem desbloquear.

8.4 Testes de sessão e *auto-lock*

Teste 10 - *Auto-lock* por inatividade

Passos: ficar sem interação durante o tempo configurado.

Resultado esperado: o cofre bloqueia automaticamente e exige novo desbloqueio.

Teste 11 - *Auto-lock* ao ir para segundo plano

Passos: enviar a aplicação para segundo plano com o cofre desbloqueado.

Resultado esperado: ao regressar, o cofre está bloqueado.

Teste 12 - Alteração do tempo de *auto-lock*

Passos: alterar o valor nas definições e repetir Teste 10.

Resultado esperado: o tempo aplicado corresponde à configuração.

8.5 Testes de Importação e Exportação

Teste 13 - Exportação do cofre

Passos: exportar o cofre para um destino escolhido.

Resultado esperado: é criada uma cópia cifrada válida; o cofre original mantém-se íntegro.

Teste 14 - Importação do cofre

Passos: selecionar um ficheiro .vltx válido e importar.

Resultado esperado: o cofre passa a corresponder ao ficheiro importado; é necessário desbloquear com a palavra-passe desse cofre.

8.6 Testes de robustez e integridade do ficheiro

Teste 15 - Cabeçalho inválido ou incompleto

Passos: corromper/truncar o ficheiro (ex.: remover *bytes* iniciais).

Resultado esperado: a aplicação rejeita o ficheiro com erro; não crasha.

Teste 16 - Versão/formato não suportados

Passos: modificar campos do cabeçalho para valores inesperados.

Resultado esperado: ficheiro rejeitado com mensagem de formato não suportado.

Teste 17 - *Salt/nonce* inválidos

Passos: alterar valores para tamanhos errados ou Base64 inválido.

Resultado esperado: falha controlada; cofre não abre.

Teste 18 - Adulteração do conteúdo cifrado

Passos: alterar alguns *bytes* do *payload* cifrado.

Resultado esperado: decifragem falha por integridade; não há qualquer fuga de dados.

Teste 19 - Interrupção durante gravação

Passos: tentar interromper a aplicação durante uma operação que grava o cofre.

Resultado esperado: a aplicação evita ficheiros “meio gravados” e mantém o cofre consistente.

8.7 Testes das funcionalidades do Projeto II

Teste 20 - Política de palavra-passe mestra

Passos: tentar criar um cofre com palavra-passe fraca (curta ou sem classes de caracteres).

Resultado esperado: a criação é bloqueada; o indicador de força e as regras em falta são apresentados.

Teste 21 - Penalizações de desbloqueio

Passos: falhar o desbloqueio 3 vezes; verificar o bloqueio; falhar mais 2 vezes após o fim do bloqueio.

Resultado esperado: bloqueio de 10 minutos após 3 falhas; +15 minutos por cada 2 falhas adicionais; reposição após desbloqueio bem-sucedido.

Teste 22 - Lixeira e retenção

Passos: apagar entradas/documentos; restaurar uns e eliminar definitivamente outros; alterar o período de retenção.

Resultado esperado: itens vão para a lixeira com dados ocultos; restauro e eliminação definitiva funcionam com confirmação; a retenção configurada é respeitada.

Teste 23 - Documentos sigilosos

Passos: importar documentos de vários tipos e tamanhos; pré-visualizar; exportar; tentar exceder os limites.

Resultado esperado: documentos guardados cifrados em blocos; pré-visualização até 10 MB; limites de 100 MB/documento e 500 MB totais aplicados com mensagens claras.

Teste 24 - *Pepper* e código de recuperação

Passos: ativar o *pepper*; guardar o código de recuperação; simular reposição com o código.

Resultado esperado: o desbloqueio passa a exigir o segredo adicional; o código de recuperação repõe o acesso.

Teste 25 - Autenticação biométrica

Passos: ativar biometria; desbloquear com impressão digital; cancelar a biometria e usar palavra-passe.

Resultado esperado: a biometria desbloqueia quando confirmada; o cancelamento mantém o acesso por palavra-passe mestra.

Teste 26 - PINs de proteção e *backoff*

Passos: definir PINs (globais e individuais); falhar repetidamente; recuperar com a palavra-passe mestra.

Resultado esperado: o acesso é bloqueado com atraso progressivo; a recuperação por palavra-passe mestra funciona.

Teste 27 - Cofre falso

Passos: configurar palavra-passe de cofre falso; desbloquear com essa palavra-passe.

Resultado esperado: abre o cofre falso com conteúdo fictício; o cofre real permanece inacessível e indistinguível para um observador.

Teste 28 - Auto-destruição (ambiente controlado)

Passos: em cofre de teste, configurar PIN de auto-destruição e acioná-lo; repetir com limite de tentativas falhadas.

Resultado esperado: o cofre real, os *backups* e os segredos do armazenamento seguro são eliminados de forma irreversível.

Teste 29 - *Autofill* no Android

Passos: ativar o serviço de *autofill*; abrir uma aplicação/página com credenciais guardadas; preencher.

Resultado esperado: as sugestões corretas aparecem por aplicação/URL; o preenchimento exige autenticação prévia.

Teste 30 - Temas, modos visuais e modo privado

Passos: alternar tema claro/escuro/automático e modos Moderno/Clássico; ativar o modo privado.

Resultado esperado: a interface atualiza de forma consistente em todos os ecrãs; no modo privado os dados sensíveis ficam ocultos com revelação por toque.

Teste 31 - Limite total de documentos sigilosos

Passos: adicionar documentos até aproximar o limite total (500 MB) e tentar ultrapassá-lo.

Resultado esperado: a aplicação rejeita a operação que excede o limite total, sem corromper nem reescrever o cofre, apresentando mensagem clara.

Teste 32 - Rejeição de ficheiro de formato não suportado

Passos: tentar importar como cofre um ficheiro que não esteja no formato V3 (por exemplo, um ficheiro de versão anterior ou de outro tipo).

Resultado esperado: o ficheiro é rejeitado com uma mensagem de formato não suportado, sem afetar o cofre atual.

Teste 33 - Alteração da palavra-passe mestra com recifragem

Passos: alterar a palavra-passe mestra nas definições e voltar a desbloquear o cofre com a nova palavra-passe.

Resultado esperado: o cofre é recifrado com a nova chave, todos os dados são preservados e o desbloqueio só funciona com a nova palavra-passe.

8.8 Critérios de aceitação

Considera-se a versão final validada quando:

- O cofre é criado e desbloqueado de forma consistente.
- Palavra-passe incorreta nunca permite acesso ao conteúdo.
- Criar/editar/apagar entradas persiste após reinícios.
- *Auto-lock* funciona por inatividade e por mudança de estado da aplicação.
- Exportação e importação produzem ficheiros válidos e recuperáveis.
- Ficheiros corrompidos/adulterados são rejeitados sem crash e sem exposição de dados.
- Os testes automatizados (flutter test) passam na totalidade.
- As proteções avançadas (biometria, PINs, *pepper*, cofre falso e auto-destruição) funcionam como configurado, validadas em ambiente controlado.
- Os limites de documentos sigilosos são respeitados e a pré-visualização e exportação funcionam sem expor conteúdo em texto simples.

9. Conclusão e trabalho futuro

O EncryVault foi concebido como um gestor de credenciais *offline-first*, centrado na proteção do cofre local e na redução da superfície de ataque associada a serviços remotos. No final do Projeto II, a aplicação cumpre esse objetivo numa versão final funcional: o cofre no formato V3 guarda entradas e documentos sigilosos cifrados com Argon2id e XChaCha20-Poly1305, e o acesso é protegido por palavra-passe mestra, biometria, *pepper* opcional, PINs, penalizações progressivas e mecanismos de emergência (cofre falso e auto-destruição), complementados por funcionalidades de organização, saúde das palavras-passe e *autofill* nativo no Android.

Apesar do progresso realizado, existem limitações naturais num sistema deste tipo: a segurança global continua dependente da robustez da palavra-passe mestra escolhida pelo utilizador; a limpeza de dados sensíveis em memória (zeroização) é limitada pelas características da linguagem Dart e do seu *garbage collector*; e o histórico de palavras-passe não tem atualmente um limite máximo configurável, podendo fazer crescer o cofre ao longo do tempo.

Como trabalho futuro, pretende-se introduzir um limite configurável para o histórico de palavras-passe; implementar o histórico de acessos ao cofre; estudar o teclado interno e o teclado aleatório planeados como funcionalidades opcionais; reforçar a zeroização de memória dentro das limitações da plataforma; e avaliar o suporte a outras plataformas, mantendo o princípio de funcionamento totalmente *offline*.

Bibliografia

[1] IETF, “RFC 9106: Argon2 Memory-Hard Function for Password Hashing and Proof-of-Work Applications,” 2021. [Online]. Available: <https://datatracker.ietf.org/doc/rfc9106/>. [Acedido em 20 01 2026].

[2] IETF, “RFC 8439: ChaCha20 and Poly1305 for IETF Protocols,” 2018. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc8439>. [Acedido em 20 01 2026].

[3] Libsodium, “XChaCha20-Poly1305 (AEAD) — Construction,” [Online]. Available: https://libsodium.gitbook.io/doc/secret-key_cryptography/aead/chacha20-poly1305/xchacha20-poly1305_construction. [Acedido em 20 01 2026].

[4] OWASP, “Password Storage Cheat Sheet,” [Online]. Available: https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html. [Acedido em 20 01 2026].

[5] OWASP, “Cryptographic Storage Cheat Sheet,” [Online]. Available: https://cheatsheetseries.owasp.org/cheatsheets/Cryptographic_Storage_Cheat_Sheet.html. [Acedido em 20 01 2026].

[6] NIST, “Digital Identity Guidelines: Authentication and Lifecycle Management (SP 800-63B),” [Online]. Available: <https://pages.nist.gov/800-63-4/sp800-63b.html>. [Acedido em 20 01 2026].

[7] Android Developers, “Secure sensitive activities (FLAG_SECURE) | Fraud prevention,” [Online]. Available: <https://developer.android.com/security/fraud-prevention/activities>. [Acedido em 20 01 2026].

[8] Flutter, “Flutter documentation,” [Online]. Available: <https://docs.flutter.dev/>. [Acedido em 20 01 2026].

[9] Flutter, “Read and write files (Cookbook),” [Online]. Available: <https://docs.flutter.dev/cookbook/persistence/reading-writing-files>. [Acedido em 20 01 2026].

[10] Dart, “Dart language tour,” [Online]. Available: <https://dart.dev/guides/language/language-tour>. [Acedido em 20 01 2026].

[11] pub.dev, “file_picker | Flutter package,” [Online]. Available: https://pub.dev/packages/file_picker. [Acedido em 20 01 2026].

[12] pub.dev, “shared_preferences | Flutter package,” [Online]. Available: https://pub.dev/packages/shared_preferences. [Acedido em 20 01 2026].

[13] pub.dev, “flutter_secure_storage | Flutter package,” [Online]. Available: https://pub.dev/packages/flutter_secure_storage. [Acedido em 20 01 2026].

- [14] Bitwarden, “Bitwarden Security Whitepaper,” [Online]. Available: <https://bitwarden.com/help/bitwarden-security-white-paper/>. [Acedido em 20 01 2026].
- [15] Bitwarden, “Username & Password Generator,” [Online]. Available: <https://bitwarden.com/help/generator/>. [Acedido em 20 01 2026].
- [16] 1Password, “1Password Security Design (White Paper),” 2021. [Online]. Available: <https://1password.com/files/1password-white-paper.pdf>. [Acedido em 20 01 2026].
- [17] KeePass, “KDBX File Format Specification,” [Online]. Available: <https://keepass.info/help/kb/kdbx.html>. [Acedido em 20 01 2026].
- [18] LastPass, “Password Generator,” [Online]. Available: <https://www.lastpass.com/features/password-generator>. [Acedido em 20 01 2026].
- [19] Proton AG, “Proton Pass,” [Online]. Available: <https://proton.me/pass>. [Acedido em 20 01 2026].
- [20] Android Developers, “Autofill framework,” [Online]. Available: <https://developer.android.com/guide/topics/text/autofill>. [Acedido em 05 06 2026].
- [21] pub.dev, “local_auth | Flutter package,” [Online]. Available: https://pub.dev/packages/local_auth. [Acedido em 05 06 2026].
- [22] pub.dev, “biometric_storage | Flutter package,” [Online]. Available: https://pub.dev/packages/biometric_storage. [Acedido em 05 06 2026].
- [23] pub.dev, “password_strength_checker | Flutter package,” [Online]. Available: https://pub.dev/packages/password_strength_checker. [Acedido em 05 06 2026].
- [24] OWASP, “Mobile Application Security (MAS),” [Online]. Available: <https://mas.owasp.org/>. [Acedido em 05 06 2026].
- [25] D. Reichl, “KeePass Password Safe,” [Online]. Available: <https://keepass.info/>. [Acedido em 12 06 2026].
- [26] D. Reichl, “KDBX 4 - KeePass,” [Online]. Available: https://keepass.info/help/kb/kdbx_4.html. [Acedido em 12 06 2026].

Nota: Formatação inspirada no modelo de bibliografia em estilo IEEE.

Anexo A - Especificação do formato V3 do ficheiro .vltx

Este anexo descreve a organização do ficheiro de cofre no formato V3 (extensão .vltx), complementando a secção 5.6. O ficheiro é composto por quatro zonas: cabeçalho, blocos de documentos, manifesto e rodapé.

A.1 Cabeçalho

O cabeçalho é precedido por um inteiro de 4 *bytes* (ordem *big-endian*) que indica o tamanho do bloco seguinte, um documento JSON em claro com os metadados do formato. Os campos principais do cabeçalho são:

- *magic* - identificador do formato.
- *formatVersion* - versão do formato (3).
- *container* - tipo de contentor utilizado.
- *cipherId* - cifra do conteúdo (XChaCha20-Poly1305-IETF).
- *kdf*, *memLimit*, *opsLimit* e *parallelism* - função e parâmetros de derivação de chave (Argon2id).
- *saltB64* - *salt* da derivação de chave.
- *subkeyKdf* - derivação das subchaves utilizadas.
- *vaultIdB64* - identificador do cofre.
- *defaultChunkSize* - tamanho por omissão dos blocos de documentos.
- *pepperEnabled* - indica se está ativa a utilização de *pepper*.

A.2 Blocos de documentos

A seguir ao cabeçalho encontram-se os blocos (*chunks*) dos documentos sigilosos. Cada bloco é cifrado individualmente com XChaCha20-Poly1305 e utiliza um *nonce* próprio, permitindo ler e escrever documentos sem recifrar a totalidade do cofre.

A.3 Manifesto

O manifesto é um bloco cifrado que contém os metadados das entradas, dos documentos e a lista de blocos associada a cada documento, incluindo a sua localização e tamanho.

A.4 Rodapé

O ficheiro termina com um rodapé de tamanho fixo (64 *bytes*) que permite localizar as restantes zonas de forma determinística. O rodapé inclui:

- Identificador e versão do rodapé.
- *offset* e tamanho (cifrado e em claro) do manifesto.
- *nonce* do manifesto.
- Campo de *flags*.

O cabeçalho, o rodapé e os metadados de cada bloco são autenticados como dados associados (AAD) na cifragem. Desta forma, qualquer adulteração destas zonas, ainda que parcialmente em claro, é detetada na abertura e o ficheiro é rejeitado. As operações de gravação utilizam escrita atómica (ficheiro temporário, *flush* e substituição), com uma cópia de salvaguarda temporária que permite repor o ficheiro anterior caso a escrita seja interrompida.

Anexo B - Mapa dos testes automatizados

Este anexo complementa o Capítulo 8 e enumera os 28 ficheiros de teste automatizado (flutter test), que totalizam cerca de 188 casos de teste, agrupados por área funcional.

B.1 Criptografia e formato do cofre

- Cofre e cifragem: *crypto_vault_test*.

B.2 Palavra-passe mestra e saúde

- Política da palavra-passe mestra: *master_password_policy_test*.

B.3 Pepper

- Serviço e proteção: *pepper_service_test*, *pepper_protection_settings_section_test*, *pepper_unlock_service_test*.
- Código de recuperação: *pepper_recovery_code_test*, *pepper_recovery_code_lifecycle_test*.

B.4 PINs de proteção

- PINs, grupos e atraso progressivo: *pin_guard_service_test*, *protection_pin_groups_test*, *pin_backoff_service_test*.

B.5 Cofre falso e auto-destruição

- Cofre falso: *decoy_vault_service_test*.
- Auto-destruição: *self_destruct_service_test*.

B.6 Biometria e ações sensíveis

- Biometria: *biometric_auth_service_test*.
- Âmbito de ações sensíveis: *sensitive_action_scope_test*.

B.7 Proteção de ecrã e *clipboard*

- Proteção de ecrã: *screen_protection_settings_ui_test*.
- Limpeza do *clipboard*: *sensitive_clipboard_service_test*, *sensitive_clipboard_lifecycle_test*.

B.8 Autofill e preenchimento

- *Autofill* e preenchimento de credenciais: *autofill_match_service_test*, *credential_fill_preferences_test*, *credential_fill_settings_section_test*.

B.9 Entradas, lixeira e interface

- Lixeira de entradas: *vault_entry_trash_test*.
- Modo privado e aparência: *private_mode_preferences_test*, *app_appearance_test*.
- Ajuda/FAQ: *help_faq_page_test*, *help_faq_routing_test*.
- Texto da interface e manifesto: *accented_user_facing_text_test*, *android_manifest_test*, *widget_test*.

Anexo C - Manifesto Android e modelo de permissões

Este anexo detalha a configuração de segurança e privacidade declarada no manifesto Android da aplicação, referida na secção 7.4.

C.1 Permissões

A aplicação declara um conjunto mínimo de permissões:

- *USE_BIOMETRIC* - necessária para a autenticação biométrica.

Não é declarada a permissão de acesso à Internet, o que torna a aplicação tecnicamente incapaz de estabelecer ligações de rede e reforça, de forma verificável, a sua natureza totalmente *offline*.

C.2 Cópias de segurança do sistema

As cópias de segurança automáticas do Android estão desativadas (*allowBackup* e *fullBackupContent* a falso), garantindo que o ficheiro do cofre não é incluído em mecanismos de *backup* externos sem o controlo explícito do utilizador.

C.3 Componentes nativos

O manifesto declara os seguintes componentes:

- *MainActivity* - atividade principal que aloja a interface Flutter.
- *AutofillAuthActivity* - atividade de autenticação utilizada pelo serviço de preenchimento automático, excluída da lista de aplicações recentes.
- *EncryVaultAutofillService* - serviço de *autofill* protegido pela permissão de sistema *BIND_AUTOFILL_SERVICE*, ativado explicitamente pelo utilizador nas definições do Android.