

ToDo.It App

**Relatório Laboratório De Planeamento e
Desenvolvimento de Software**

Realizado Por:

- Bruno Costa A179895
- Carolina Machado A179359
- Luís Lemos A178644
- Pedro Pereira A178686

Índice

Introdução.....	3
Objetivos.....	4
Mockup	5
Modelo de Dados	13
Descrição Funcional	19
Diagrama de Classes	20
Diagrama de Casos de Uso.....	21
Conclusão	22

Introdução

Este relatório destaca a fase inicial do desenvolvimento de uma aplicação TodoList em WPF. Começámos por criar Mockups detalhados das interfaces necessárias e implementar o modelo de dados em C# utilizando o Visual Studio.

Esta etapa foi crucial para estabelecer uma base sólida, compreendendo os requisitos do sistema e traduzindo-os em elementos visuais e estrutura de dados coesos.

Este relatório oferece uma visão geral dos procedimentos adotados, decisões de design e características do modelo de dados implementado até ao momento.

Objetivos

- **Organização Pessoal:** O propósito primordial desta aplicação é auxiliar os utilizadores na organização das suas tarefas diárias, semanais ou mensais de forma eficaz.
- **Gestão do Tempo:** Capacitar os utilizadores a gerir o seu tempo de forma mais eficiente ao priorizarem tarefas e estabelecerem prazos.
- **Lembretes de Compromissos:** Facultar lembretes e notificações para assegurar que os utilizadores não se esqueçam das suas tarefas e compromissos importantes.
- **Interface Intuitiva:** Apresentar uma interface amigável e intuitiva, com funcionalidades como categorização de tarefas, visando proporcionar uma experiência de utilização agradável.
- **Personalização:** Permitir que os utilizadores personalizem as suas listas de tarefas de acordo com as suas preferências, adicionando etiquetas, cores, prioridades, entre outras opções.

Mockup

Páginas de Login/Registo

Entrar

G+ f t

ou usa o teu e-mail e password

E-mail

Password

Esqueceste-te da Password?

ENTRAR

Bem-vindo!

Regista os teus dados pessoais e começa uma jornada connosco !

REGISTAR

Figura 1 – Página de Login

Bem-Vindo de Volta!

Para continuar ligado connosco, por favor entra com as tuas credenciais

ENTRAR

Criar Conta

G+ f t

ou usa o teu e-mail para o registo

Nome

E-mail

Password

REGISTAR

Figura 2 – Página de Registo

Página Inicial

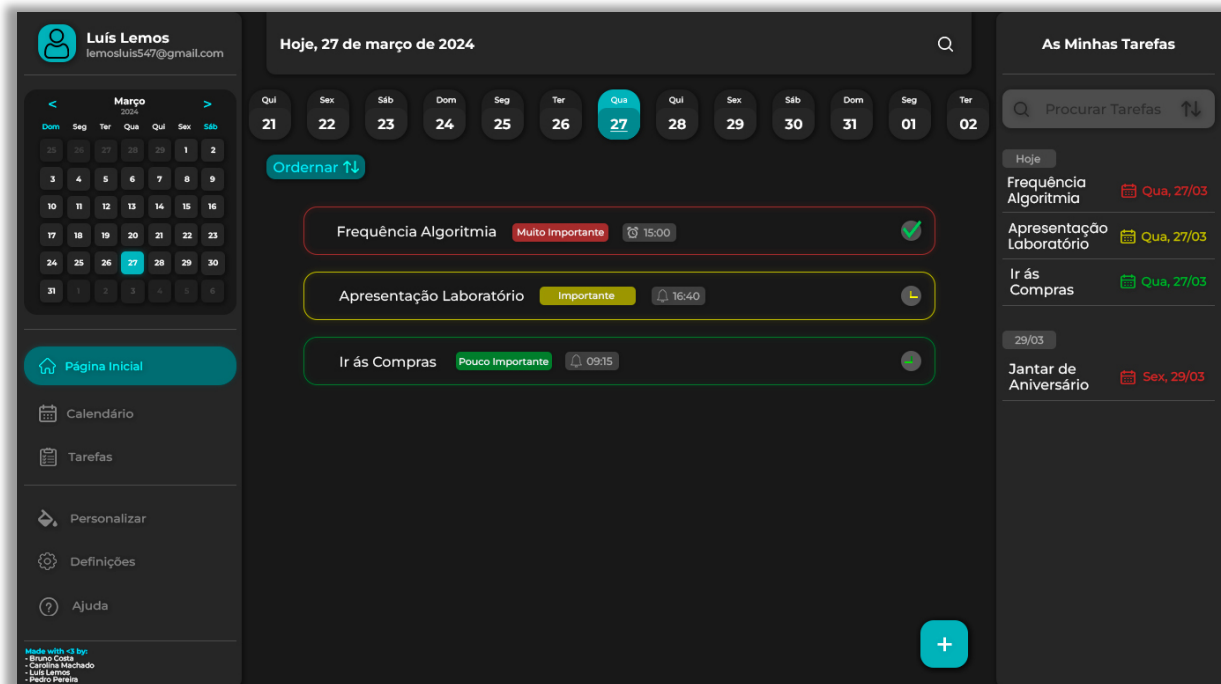


Figura 3 – Página Inicial

A Mockup da **“Página Inicial”** da aplicação ToDo.It apresenta um layout organizado em três secções principais:

- **Aba Esquerda:** Mostra o perfil do usuário, um calendário reduzido e opções de navegação.
- **Aba Direita:** Exibe uma versão resumida das tarefas pendentes.
- **Secção Central:** Destaca a data seleccionada pelo usuário e as tarefas associadas a esse dia.

Esta estrutura proporciona uma visão geral das informações pessoais, tarefas e opções de navegação, facilitando o planeamento e a organização das atividades diárias.

Página “Calendário”

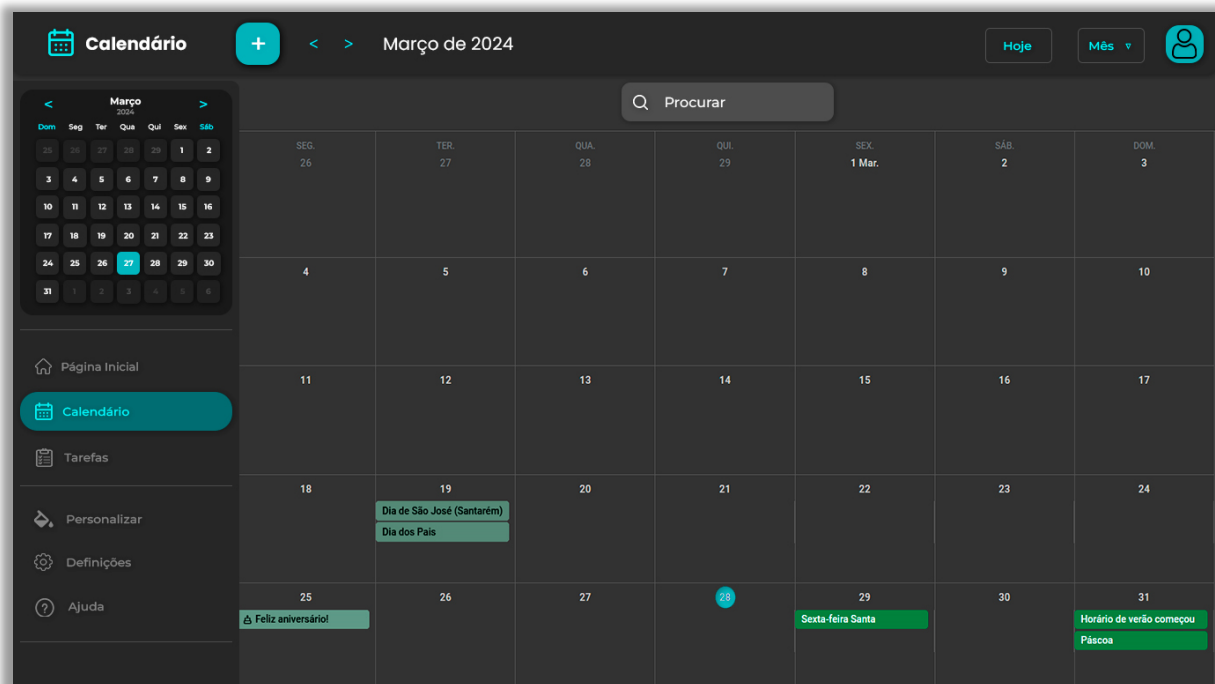


Figura 4 – Página “Calendário”

A Mockup da página “**Calendário**” da aplicação ToDo.It apresenta um layout organizado em duas secções principais:

- **Aba Esquerda:** Mostra o perfil do usuário, um calendário reduzido e opções de navegação.
- **Calendário:** Exibe o calendário numa versão aumentada permitindo ao utilizador ver com eficácia e clareza as suas tarefas e os seus respetivos dias.

Esta estrutura proporciona uma visão clara ao utilizador das suas tarefas e dos seus dias livres ao longo do tempo, permitindo assim uma gestão eficaz.

Página “Tarefas”

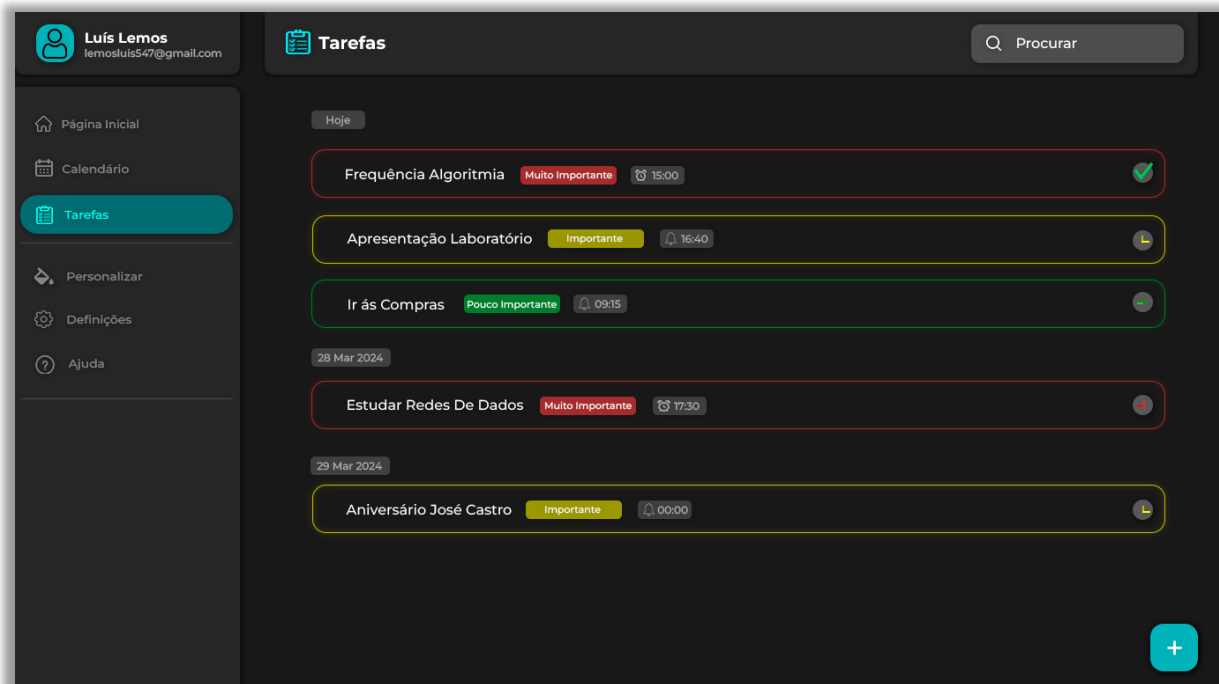


Figura 5 – Página “Tarefas”

A Mockup da página “**Tarefas**” da aplicação ToDo.It apresenta um layout organizado em duas secções principais:

- **Aba Esquerda:** Mostra o perfil do usuário e opções de navegação.
- **Tarefas:** Exibe no centro da tela uma lista com todas as tarefas organizadas da mais próxima para a mais distante em termos de data. A cor representa a sua importância e no lado mais direito da tela o utilizador tem acesso a botões relativos ao estado da tarefa.

Proporciona uma visão geral e clara das tarefas do utilizador.

Página “Criar Tarefa”

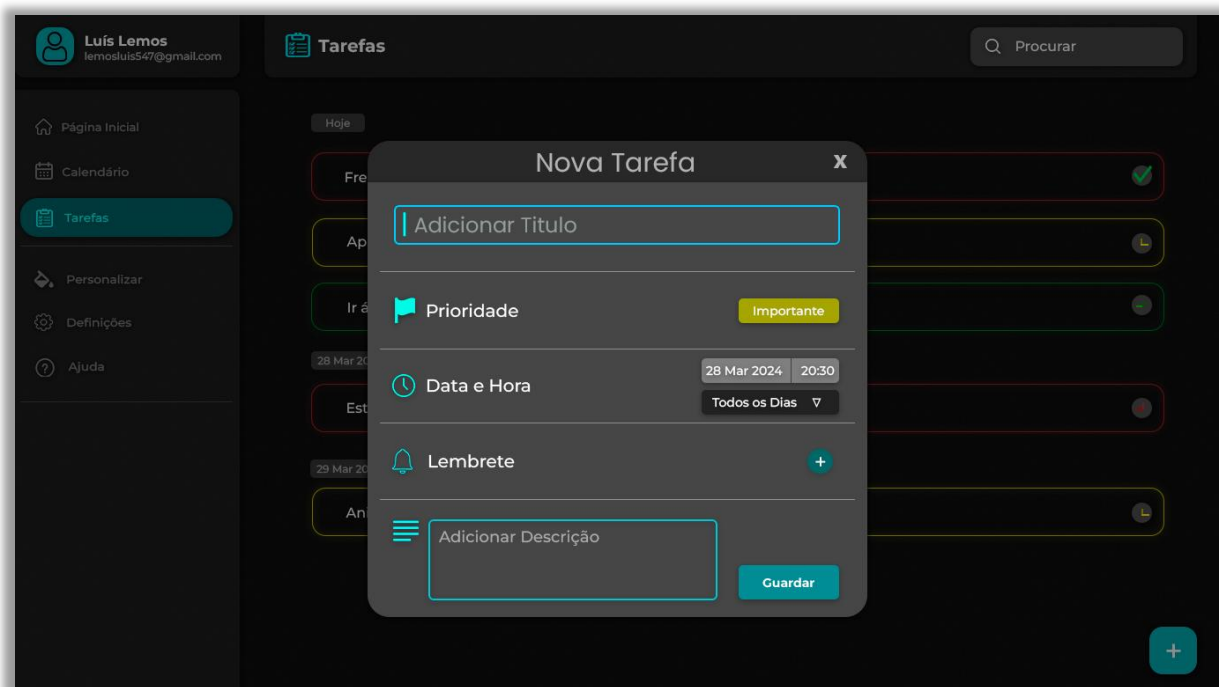


Figura 6 – Página “Criar Tarefa”

A Mockup da página **“Criar Tarefa”** da aplicação ToDo.It apresenta um layout organizado em duas secções principais numa secção:

– **Janela “Nova Tarefa”**: Abre uma janela pop-up onde o utilizador cria a nova tarefa e selecciona as características que pretende ter como título, prioridade, etc.

–**Fundo**: Permanece na aba da “Tarefas”

Esta estrutura proporciona ao utilizador uma forma eficaz de criar a sua nova tarefa e moldá-la com as características que pretende.

Página “Personalizar”

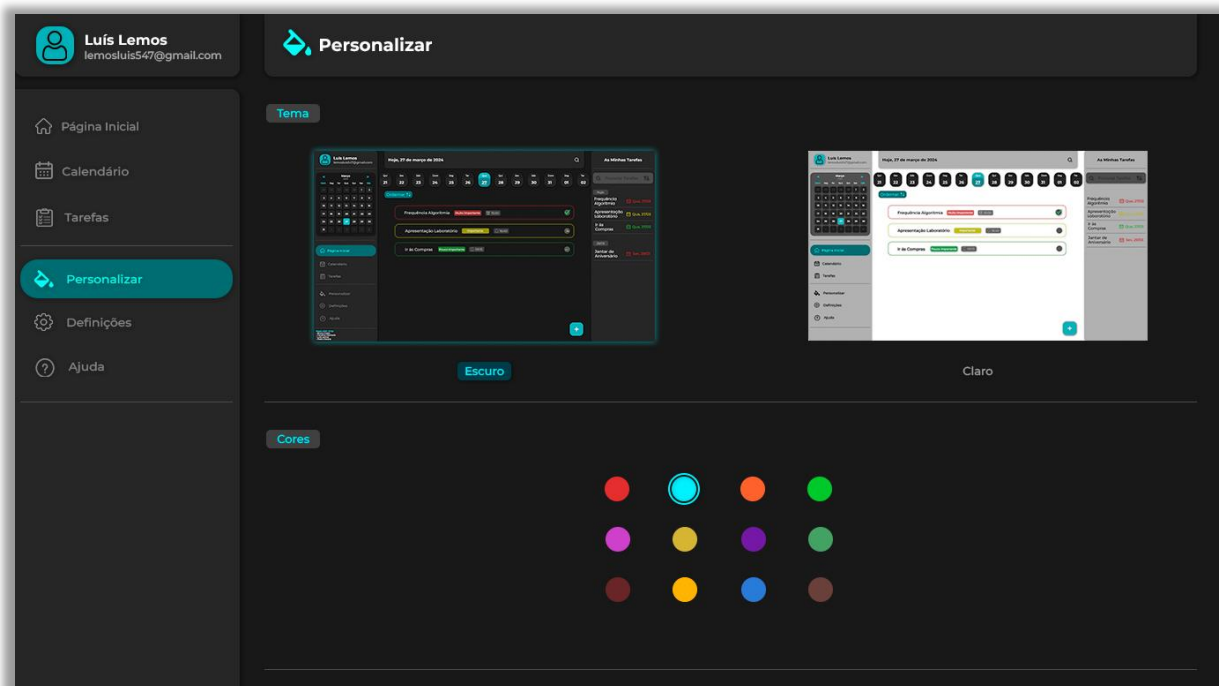


Figura 7 – Página “Personalizar”

A Mockup da página “**Personalizar**” da aplicação ToDo.It apresenta um layout organizado em duas secções principais:

- **Aba Esquerda:** Mostra o perfil do usuário e opções de navegação.
- **Secção Central:** Destaque das únicas duas características personalizáveis. Possibilidade de ajustar as cores principais da aplicação (Modo Escuro ou Modo Claro) e a personalização das cores secundárias (com várias opções incluídas).

Esta estrutura proporciona uma de forma clara e eficaz do utilizador poder personalizar a sua aplicação de acordo com as suas preferências

Página “Definições”

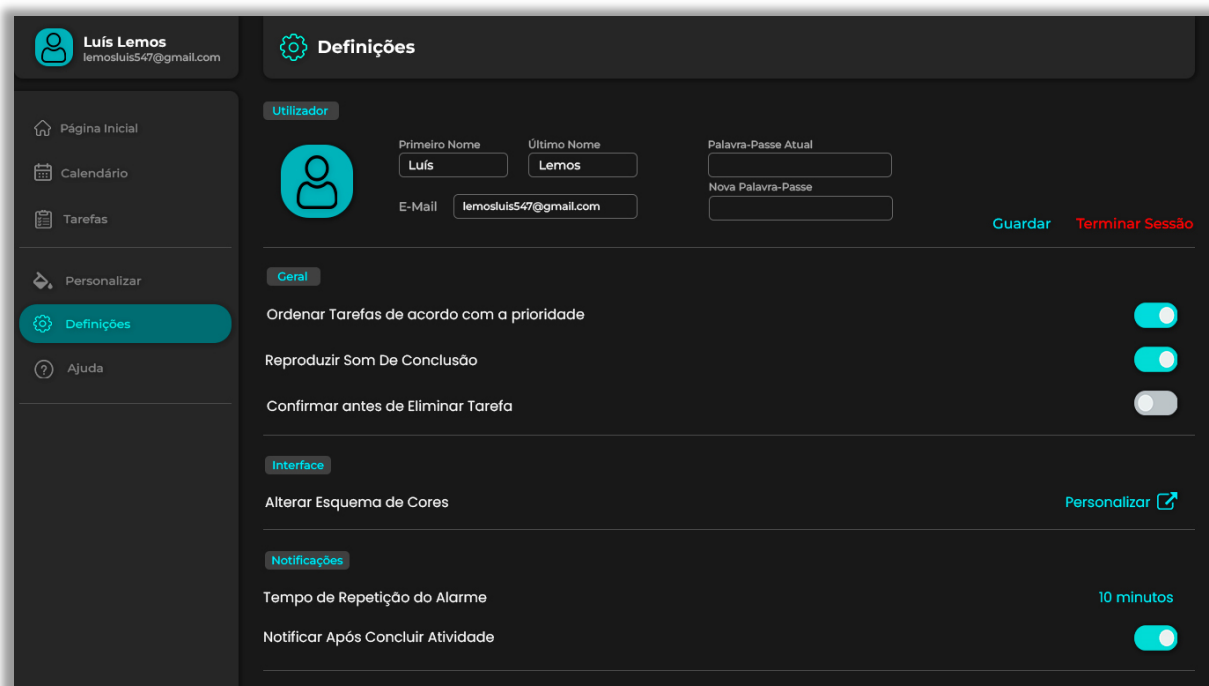


Figura 8 – Página “Definições”

A Mockup da página “**Definições**” da aplicação ToDo.It apresenta um layout organizado em duas secções principais:

- **Aba Esquerda:** Mostra o perfil do usuário e opções de navegação.
- **Secção central:** Proporciona ao utilizador mudar o seu perfil e as definições gerais da aplicação.

Esta estrutura proporciona uma visão clara ao utilizador do seu perfil e das definições que ele deseja utilizar.

Página “Ajuda”



Figura 9 – Página “Ajuda”

A Mockup da página **“Ajuda”** da aplicação ToDo.It apresenta um layout organizado em duas secções principais:

- **Aba Esquerda:** Mostra o perfil do usuário e opções de navegação.
- **Secção central:** Permite ao utilizador esclarecer as suas dúvidas sobre a aplicação.

Esta estrutura proporciona uma visão clara ao utilizador das dúvidas que pode ter e ainda proporciona uma aba de contacto para uma ajuda mais detalhada.

Modelo de Dados

VS Solution e Classe "BaseModel"

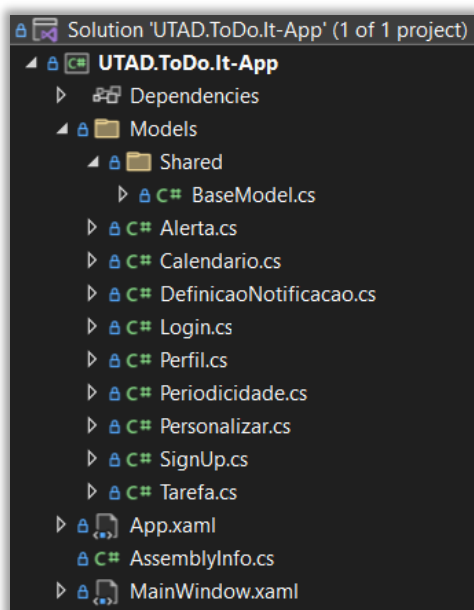


Figura 10 – VS Solution

Esta classe serve como base para outros modelos na aplicação, garantindo que cada um tenha um identificador único.

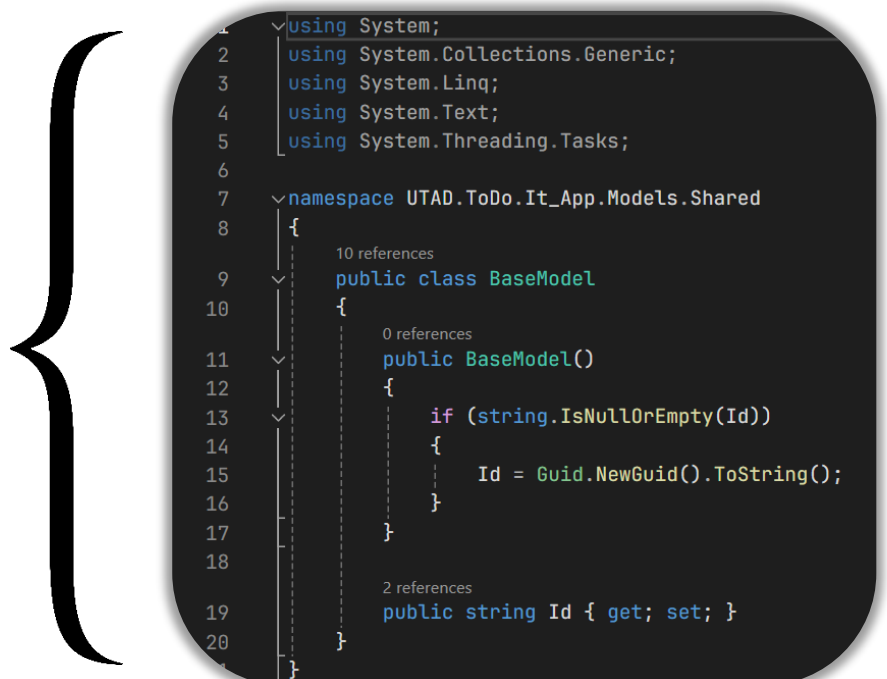


Figura 11 – Classe "BaseModel"

Classe “Alerta” e “Calendário”

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using UTAD.ToDo.It_App.Models.Shared;
7
8 namespace UTAD.ToDo.It_App.Models
9 {
10     1 reference
11     public enum TipoAlerta
12     {
13         MuitoImportante,
14         Importante,
15         PoucoImportante
16     }
17
18     0 references
19     public class Alerta : BaseModel
20     {
21         0 references
22         public string Mensagem { get; set; }
23
24         0 references
25         public DateTime DataHora { get; set; }
26
27         0 references
28         public bool Desligado { get; set; }
29
30         0 references
31         public TipoAlerta Tipo { get; set; }
32     }
33 }

```

Esta classe serve para definir o nível de importância de cada tarefa.

Figura 12 – Classe “Alerta”

Esta classe é usada para definir o calendário na aplicação.

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using UTAD.ToDo.It_App.Models.Shared;
7
8 namespace UTAD.ToDo.It_App.Models
9 {
10     0 references
11     public class Calendario : BaseModel
12     {
13         0 references
14         public int Dia { get; set; }
15
16         0 references
17         public int Semana { get; set; }
18
19         0 references
20         public int Mes { get; set; }
21
22         0 references
23         public int Ano { get; set; }
24     }
25 }

```

Figura 13 – Classe “Calendario”

Classe “DefinicaoNotificacao” e “Login”

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using UTAD.ToDo.It_App.Models.Shared;
7
8 namespace UTAD.ToDo.It_App.Models
9 {
10     0 references
11     public class DefinicaoNotificacao : BaseModel
12     {
13         0 references
14         public string Som { get; set; }
15
16         0 references
17         public string Volume { get; set; }
18
19         0 references
20         public string Repetição { get; set; }
21     }
22 }

```

Esta classe é possui três propriedades que representam as configurações de som, volume e repetição para as notificações da aplicação.

Figura 14 – Classe “DefinicaoNotificacao”

Esta classe é usada para modelar os dados de login na aplicação.

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using UTAD.ToDo.It_App.Models.Shared;
7
8 namespace UTAD.ToDo.It_App.Models
9 {
10     0 references
11     public class Login : BaseModel
12     {
13         0 references
14         public string id { get; set; }
15
16         0 references
17         public string Password { get; set; }
18     }
19 }

```

Figura 15 – Classe “Login”

Classe “Perfil” e “Periodicidade”

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using UTAD.ToDo.It_App.Models.Shared;
7
8 namespace UTAD.ToDo.It_App.Models
9 {
10     0 references
11     public class Perfil : BaseModel
12     {
13         0 references
14         public string Nome { get; set; }
15
16         0 references
17         public string Email { get; set; }
18
19         0 references
20         public string Fotografia { get; set; }
21
22         0 references
23         public string PalavraPasseAtual { get; set; }
24
25         0 references
26         public string PalavraPasseNova { get; set; }
27     }
28 }

```

Esta classe é usada para definir o perfil do utilizador na aplicação.

Figura 16 – Classe “Perfil”

Esta classe é usada para definir a periodicidade da tarefa na aplicação.

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using UTAD.ToDo.It_App.Models.Shared;
7
8 namespace UTAD.ToDo.It_App.Models
9 {
10     0 references
11     public class Periodicidade : BaseModel
12     {
13         0 references
14         public string Tipo { get; set; }
15
16         0 references
17         public string DiaDaSemana { get; set; }
18     }
19 }

```

Figura 17 – Classe “Periodicidade”

Classe “Personalizar” e “SignUp”

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using UTAD.ToDo.It_App.Models.Shared;
7
8 namespace UTAD.ToDo.It_App.Models
9 {
10     0 references
11     public class Personalizar : BaseModel
12     {
13         0 references
14         public string Tema { get; set; }
15
16         0 references
17         public string Cores { get; set; }
18     }
19 }

```

Esta classe contém as propriedades que representam as opções de personalização do tema e das cores da aplicação.

Figura 18 – Classe “Personalizar”

Esta classe é usada para modelar os dados de registo de usuários na aplicação.

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using UTAD.ToDo.It_App.Models.Shared;
7
8 namespace UTAD.ToDo.It_App.Models
9 {
10     0 references
11     public class SignUp : BaseModel
12     {
13         0 references
14         public string id { get; set; }
15
16         0 references
17         public string Email { get; set; }
18
19         0 references
20         public string Password { get; set; }
21     }
22 }

```

Figura 19 – Classe “SignUp”

Classe “Tarefa”

```

3 using System;
4 using System.Collections.Generic;
5 using System.Linq;
6 using System.Text;
7 using System.Threading.Tasks;
8 using UTAD.ToDo.It_App.Models.Shared;
9
10 namespace UTAD.ToDo.It_App.Models
11 {
12     public class Tarefa : BaseModel
13     {
14         public int id { get; set; }
15
16         public string Titulo { get; set; }
17
18         0 references
19         public string Descricao { get; set; }
20
21         0 references
22         public int DataInicio { get; set; }
23
24         0 references
25         public int DataFim { get; set; }
26
27         0 references
28         public int HoraInicio { get; set; }
29
30         0 references
31         public int HoraFim { get; set; }
32
33         0 references
34         public string NivelImportancia { get; set; }
35
36         0 references
37         public string Estado { get; set; }
38
39         0 references
40         public string Periodicidade { get; set; }
41
42         0 references
43         public string AlertaAntecipacao { get; set; }
44
45         0 references
46         public string AlertaExecucao { get; set; }
47     }
48 }

```

Esta classe é usada
para definir a tarefa e
todos os seus
atributos.

Figura 20 – Classe “Tarefa”

Descrição Funcional

Efetuar a manutenção do perfil do utilizador:

- Nome;
- Email;
- Fotografia;
- Redefinir Palavra-Passe;
- Ordenar Tarefas por periodicidade;
- Reproduzir som de conclusão;
- Confirmar antes de Eliminar Tarefa;
- Tempo de repetição do alarme;
- Notificar após concluir atividade;

Efetuar a manutenção dos alertas sobre as tarefas (operações CRUD):

- Id;
- Mensagem;
- Data/hora;
- Tipo(s) (Email, Alerta Windows);
- Desligado (usado para desativar um alerta definido);

Efetuar a manutenção das Periodicidades (operações CRUD):

- Id;
- Tipo (diária ou semanal);
- Dia(s) da semana;

Efetuar a manutenção da lista das tarefas registadas (operações CRUD):

- Id;
- Título;
- Descrição (opcional);
- Data/hora de início (hora opcional);
- Data/hora de término (hora opcional);
- Nível de importância;
- Estado (por iniciar, em execução, por terminar);
- Periodicidade;
- AlertaAntecipação (podem ser vários se a tarefa for periódica);
- AlertaExecução (podem ser vários se a tarefa for periódica);

Efetuar a manutenção da personalização com a edição dos campos:

- Tema (escuro/claro);
- Cores de ícones;

Consultar ajuda:

- Envio de email;
- Consulta de instruções sobre a aplicação;

Diagrama de Classes

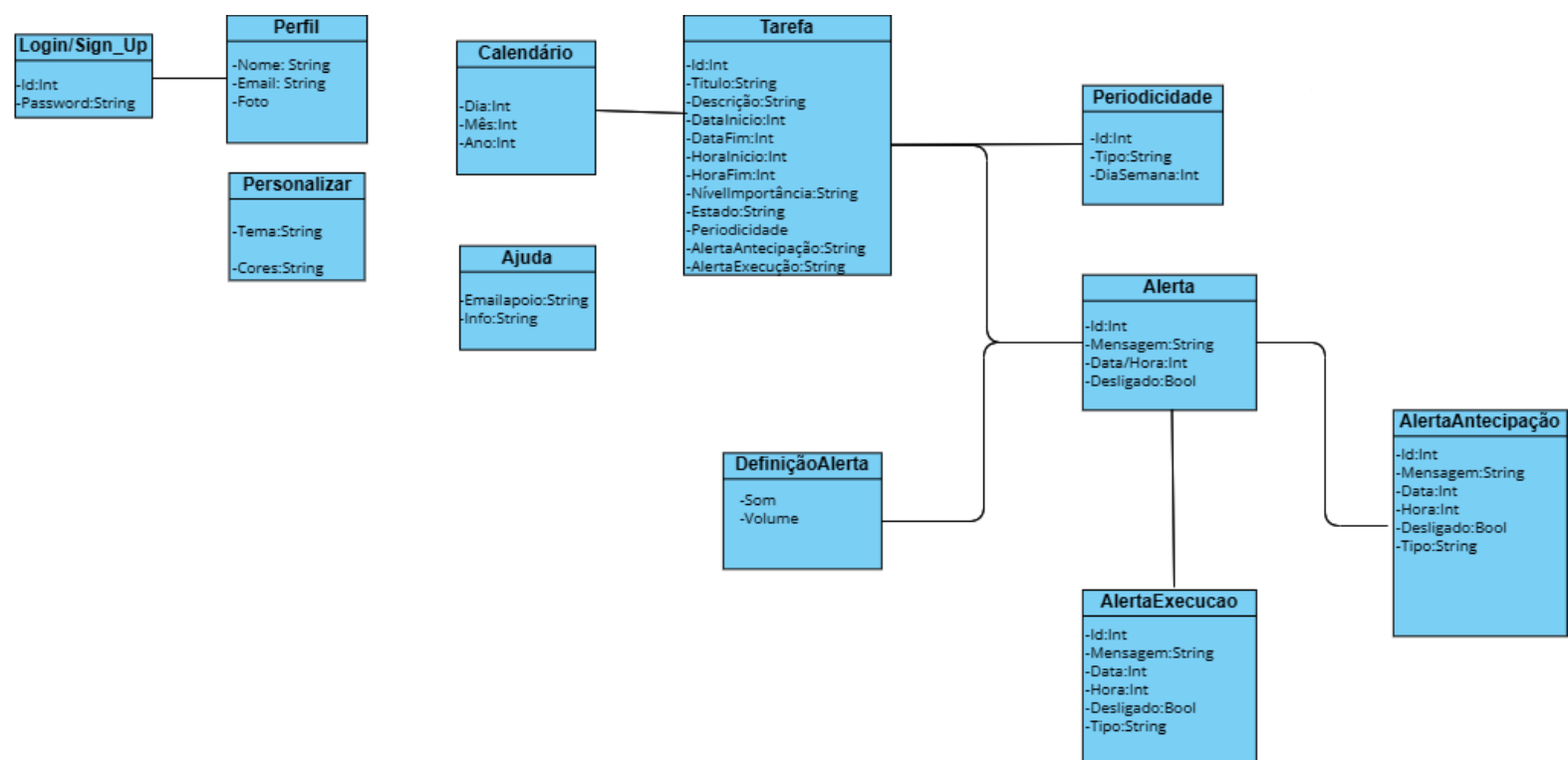


Figura 21 – Diagrama de Classes

Diagrama de Casos de Uso

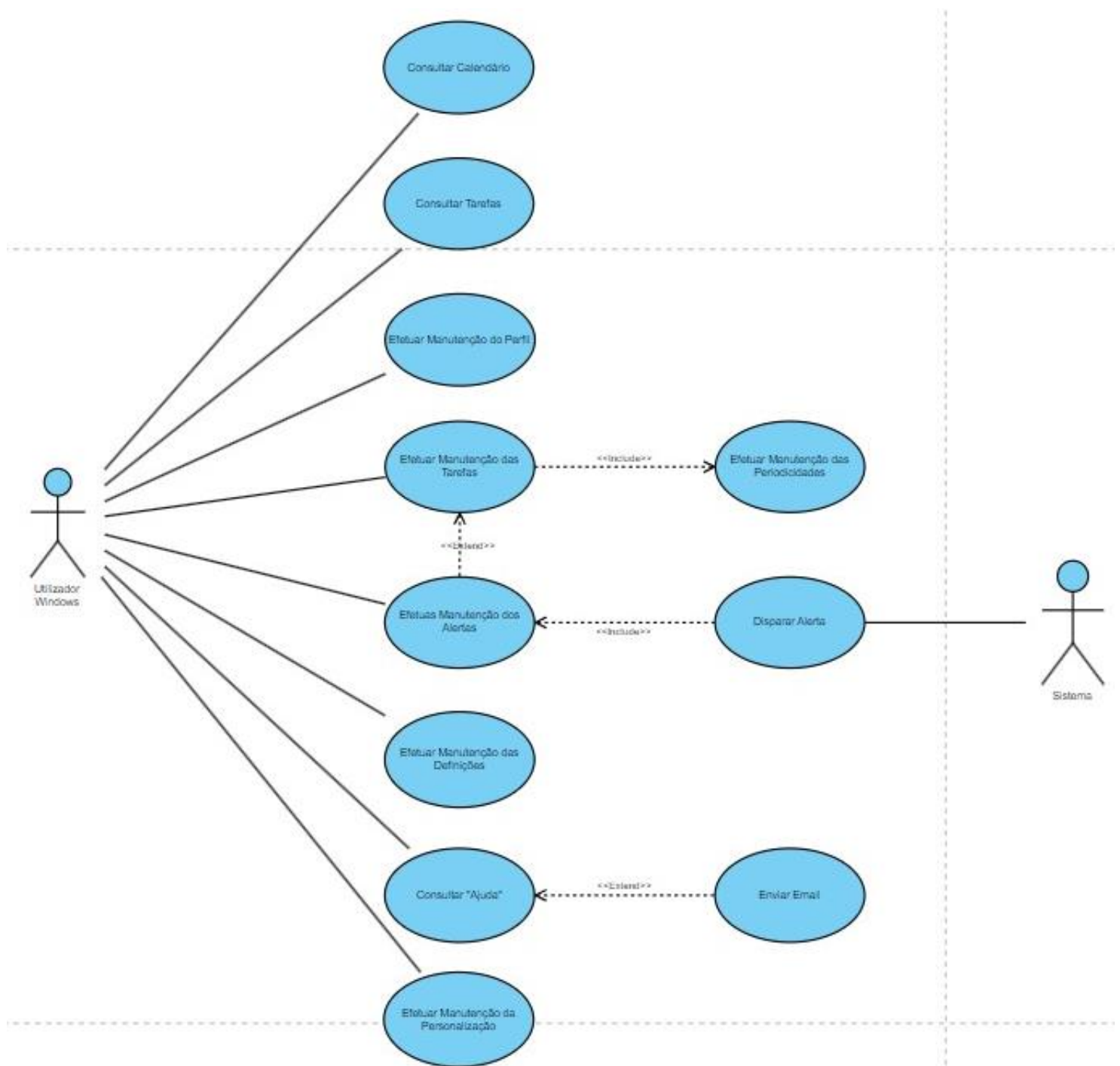


Figura 22 – Diagrama de Casos de Uso

Conclusão

Em suma, resumimos num único documento a fase inicial do desenvolvimento de uma aplicação ToDoList em **WPF**.

Explicamos com sucesso as **Mockups** detalhadas do projeto e os diagramas de caso-de-uso tais como os de classes.

Através deste **relatório** conseguimos oferecer uma visão geral do que tencionamos apresentar no final do projeto, tais como decisões de design e até mesmo características do modelo de dados implementado até à data de entrega deste relatório.

Estabelecemos esta etapa do projeto bastante **essencial** pois forneceu ao grupo uma base sólida para o resto do projeto de forma a compreendermos o que teremos de aplicar mais tarde durante a realização da aplicação.