



Universidade do Minho
Escola de Engenharia

DESENVOLVIMENTO DE SISTEMAS DE SOFTWARE

Trabalho Prático - Fase Final

Grupo 25

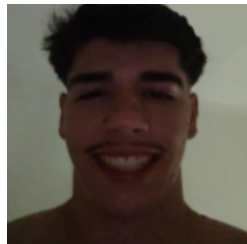
Link Github: <https://github.com/LEI-DSS/DSS2425-Grupo-25>

Departamento de Informática
Engenharia Informática 2024/25

Equipa de Trabalho:



Fábio Magalhães
A104365



Filipe Fernandes
A104185



João Macedo
A104080



André Pinto
A104267

Índice

1. Introdução	1
2. Modelo de Domínio	2
3. Diagrama de Use Case	3
4. Diagramas de Classes	4
5. Diagramas de Sequência Pós Implementação	7
6. Diagrama de Componentes	9
7. Estrutura da Base de Dados	10
7.1. Detalhes Técnicos	10
7.2. Operações Importantes	10
8. Utilização exemplo do programa	11
9. Integração CSV	16
9.1. Processo de Importação	16
9.2. Melhorias Implementadas	16
10. Conclusão	17

1. Introdução

Neste relatório, apresenta-se o resultado final do desenvolvimento de um sistema de software para a gestão de turnos no curso de Engenharia Informática. Ao longo do projeto, procuramos abordar de forma eficiente os desafios associados à gestão de turnos, considerando as particularidades de cada Unidade Curricular (UC), as necessidades dos alunos e a não sobreposição de horários. Este sistema foi concebido para lidar com a complexidade de cursos de grande dimensão, onde a gestão de recursos e horários representa um desafio significativo.

Com base nos requisitos inicialmente definidos, foi desenvolvido um sistema completo, desde a análise e modelação inicial até à implementação das funcionalidades e integração com a base de dados. Este relatório reflete o progresso do projeto, incluindo a descrição detalhada do sistema, os modelos desenvolvidos, as soluções implementadas e os resultados obtidos.

O trabalho realizado destaca a importância de uma abordagem estruturada ao desenvolvimento de software, desde a análise inicial de requisitos até à entrega de um produto funcional e alinhado com os objetivos estabelecidos.

2. Modelo de Domínio

Visual Paradigm Standard (Ricardo Universidade do Minho)

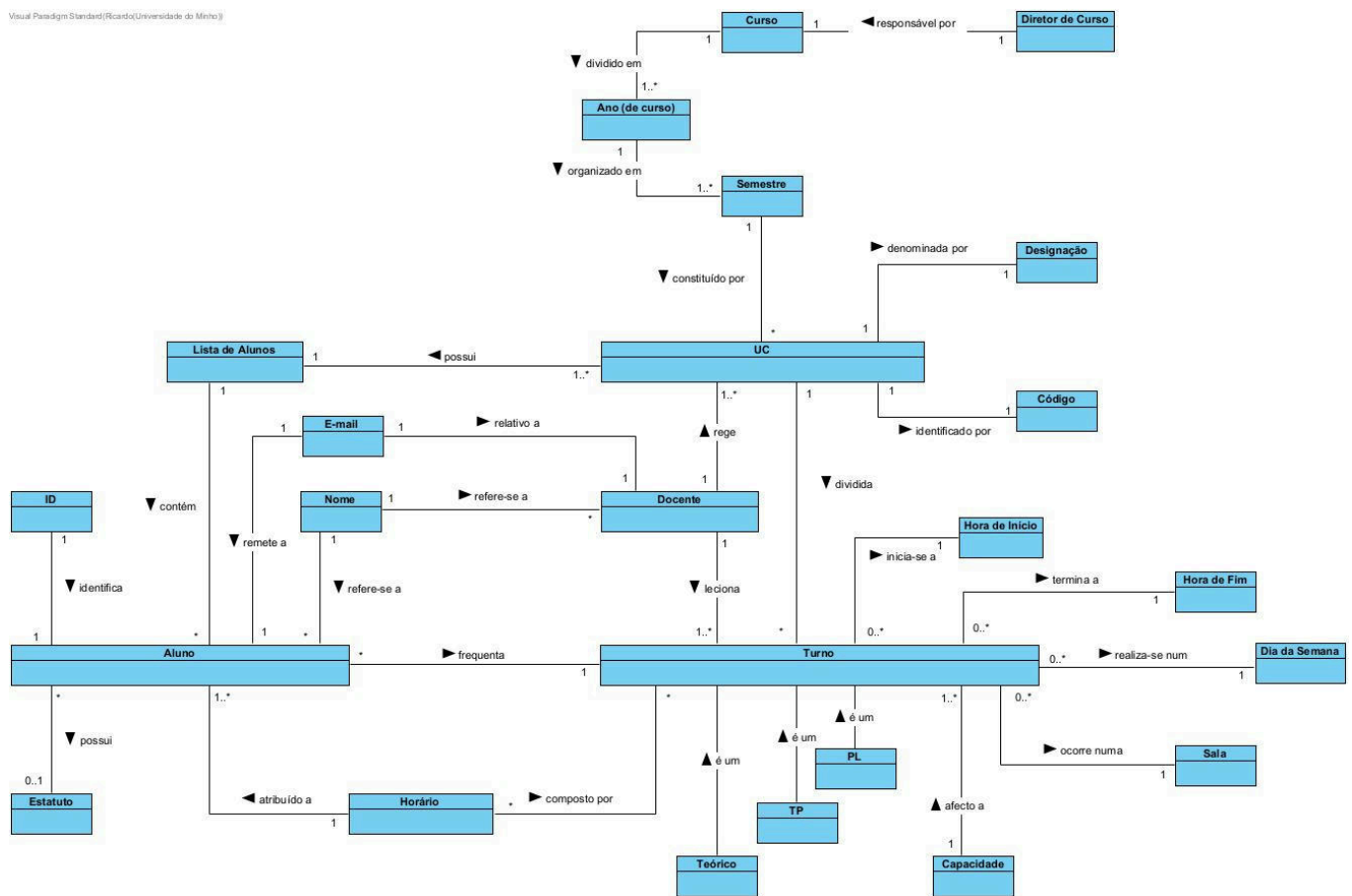


Figura 1: Modelo de Domínio.

3. Diagrama de Use Case

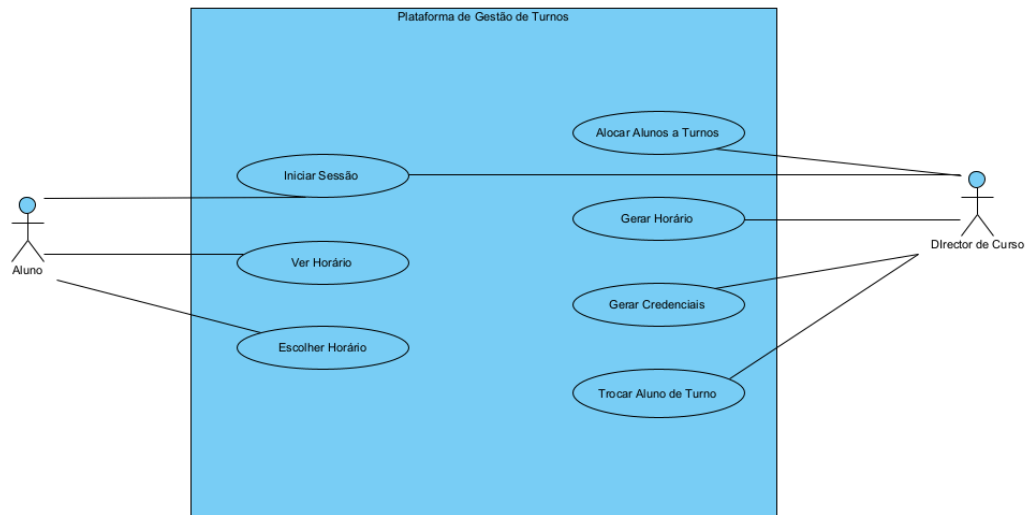


Figura 2: Diagrama de Use Case.

4. Diagramas de Classes

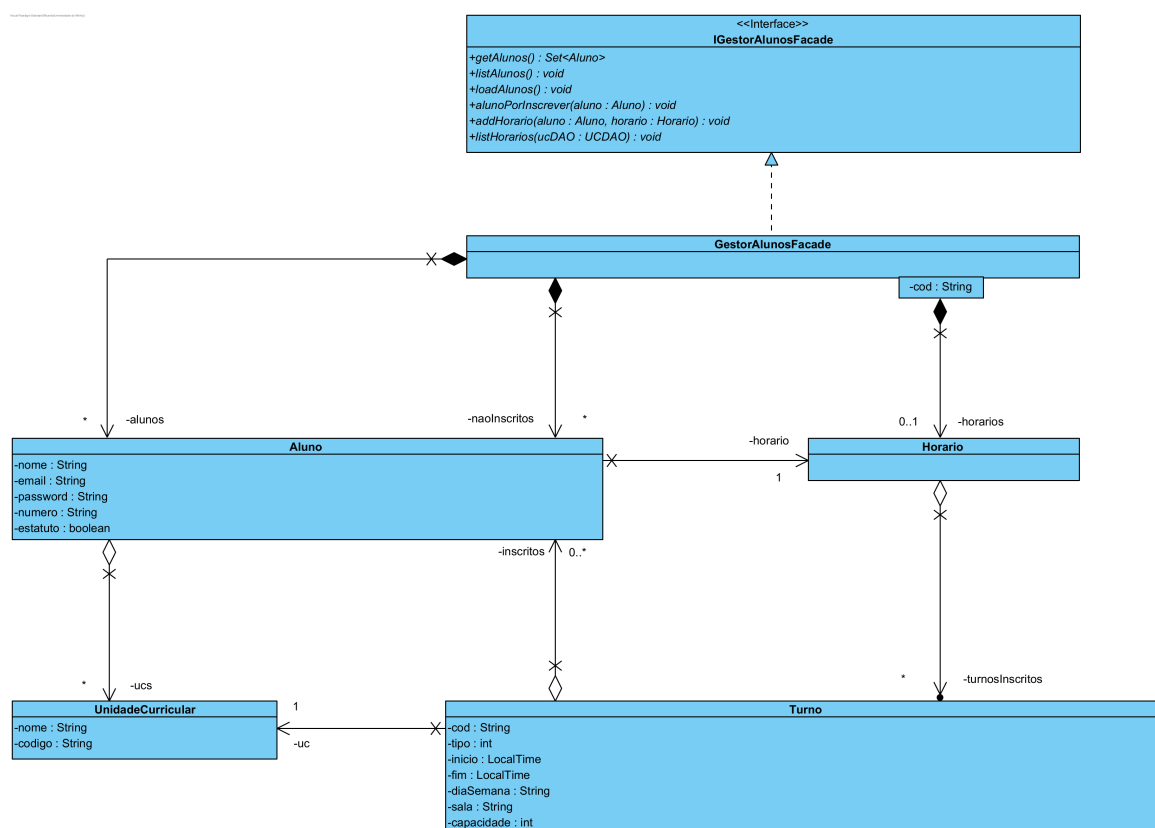


Figura 3: Diagrama de Classes de subsistema Alunos

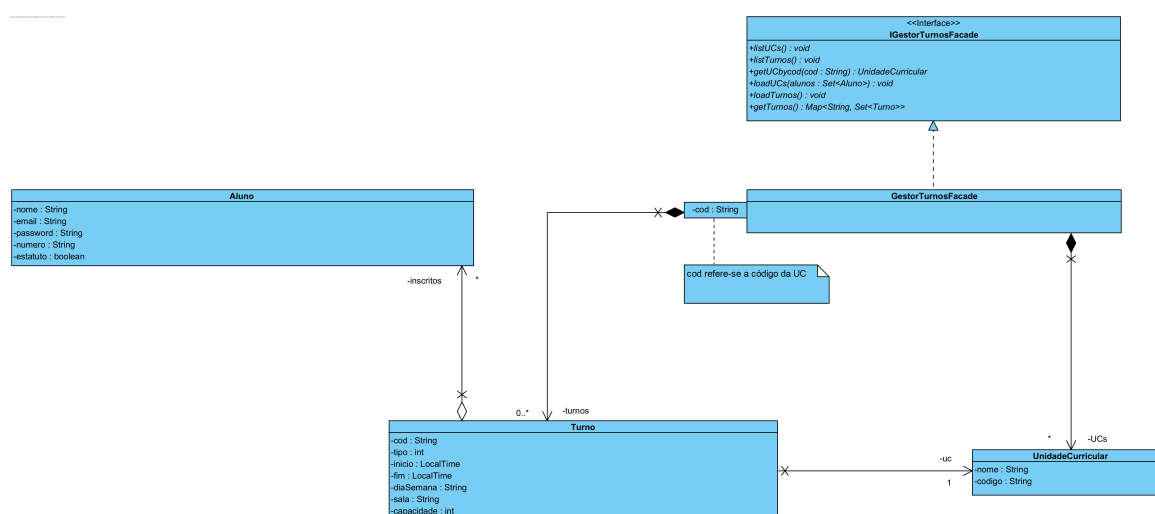


Figura 4: Diagrama de Classes de subsistema Turnos

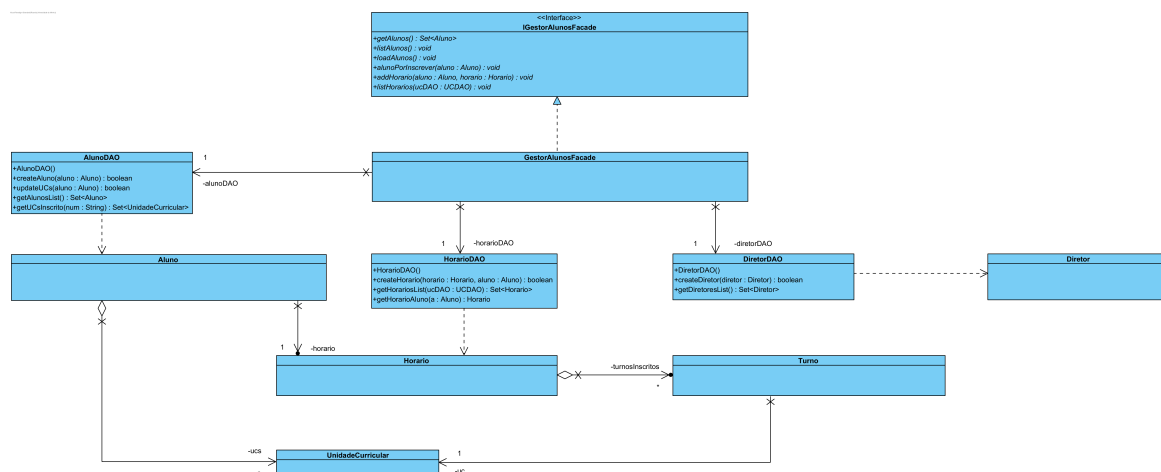


Figura 5: Diagrama de Classes DAO de subsistema Alunos

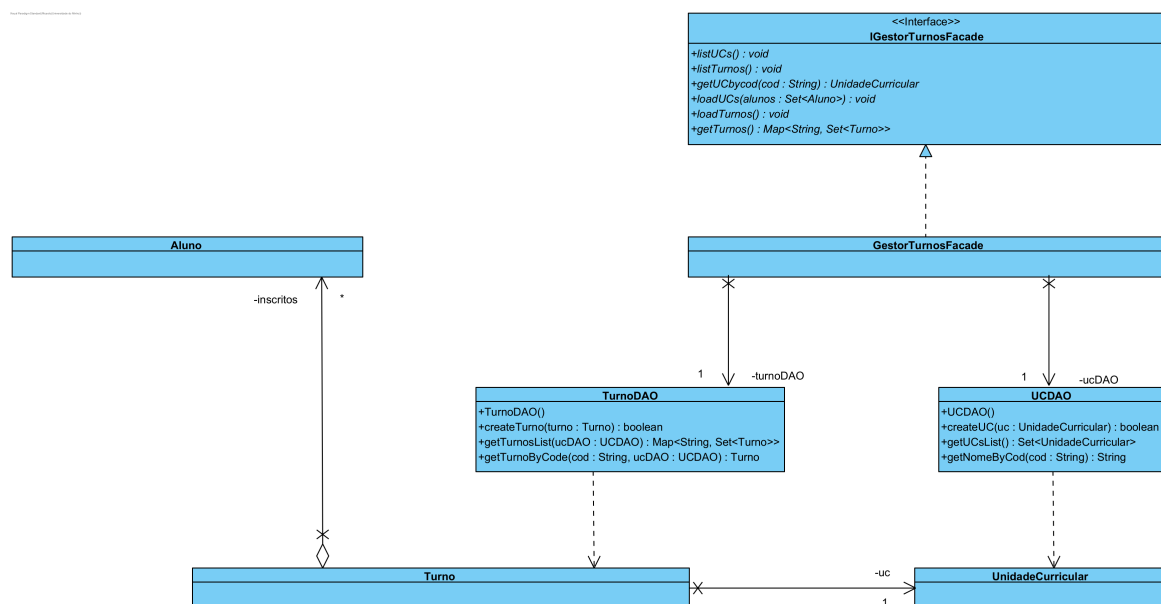


Figura 6: Diagrama de Classes DAO de subsistema Turnos

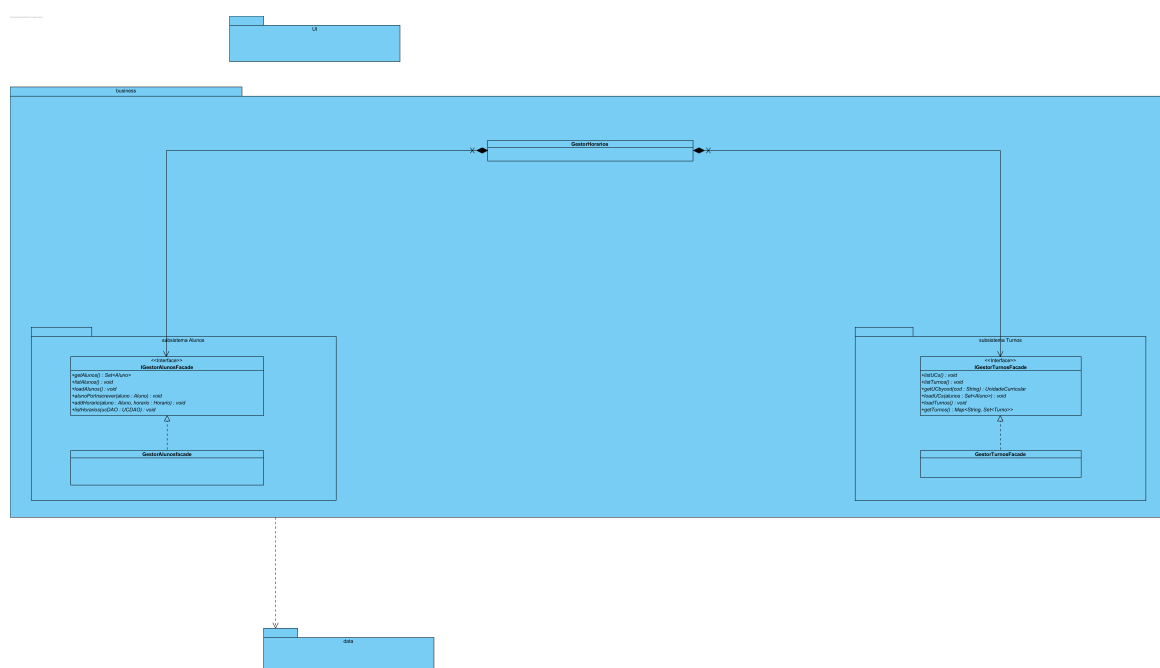


Figura 7: Diagrama de Classes de Arquitetura Geral

5. Diagramas de Sequência Pós Implementação

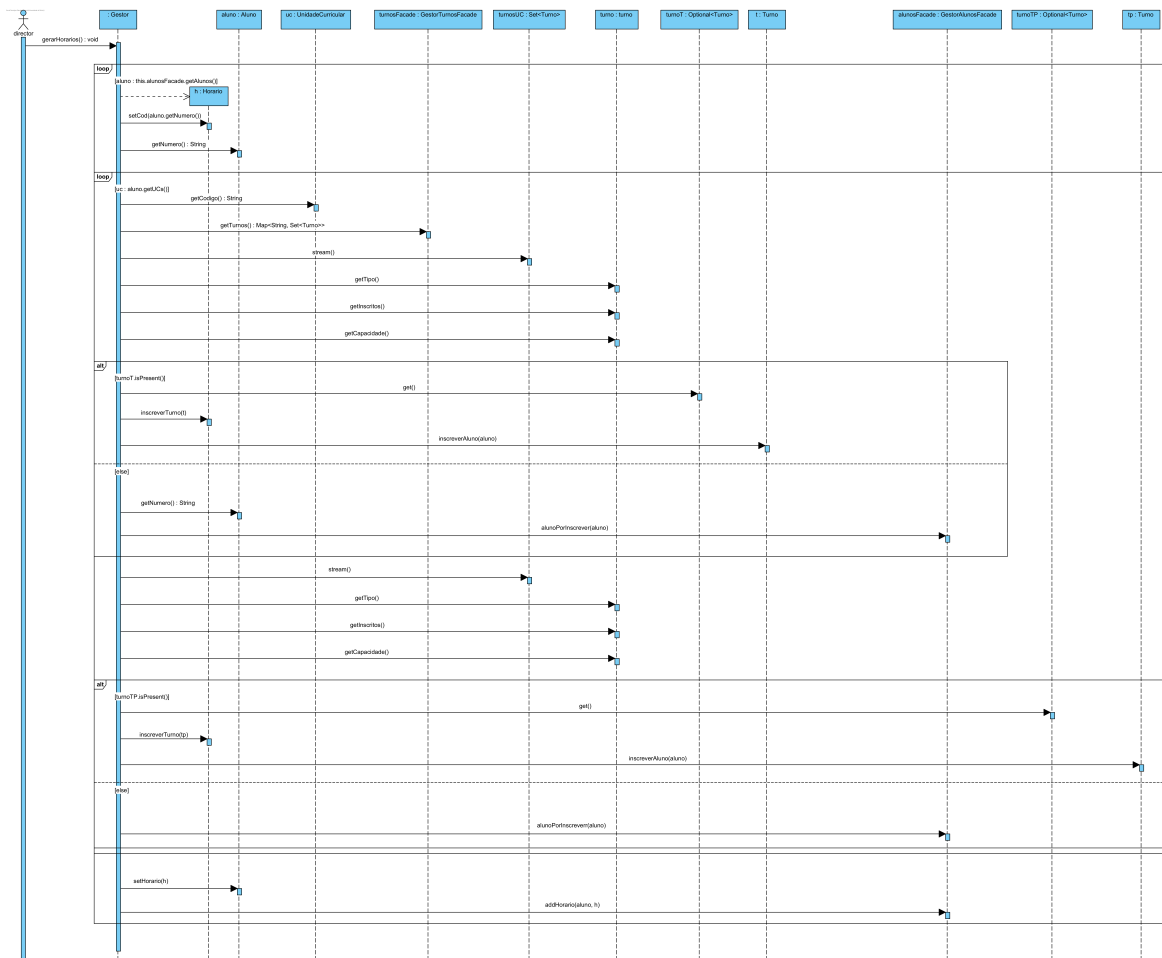


Figura 8: Diagrama de Sequência de Gerar Horários

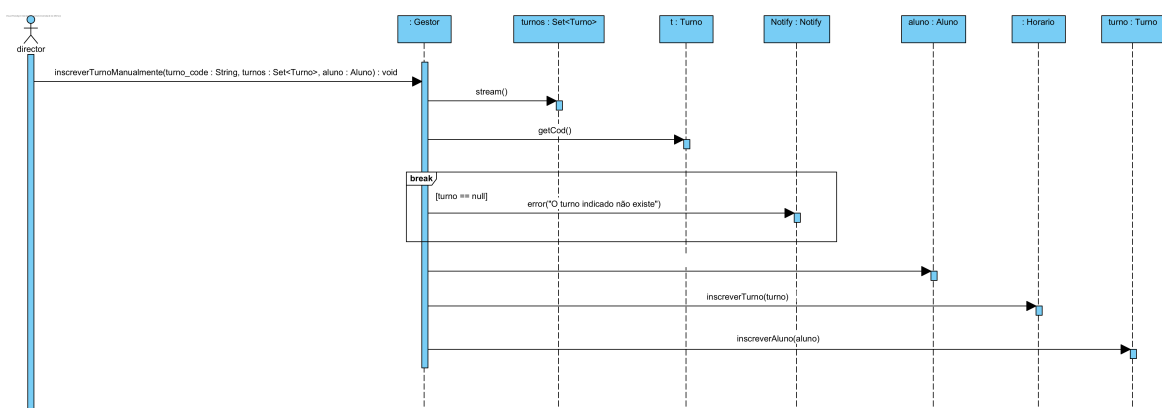


Figura 9: Diagrama de Sequência de Inscrever Turno Manualmente

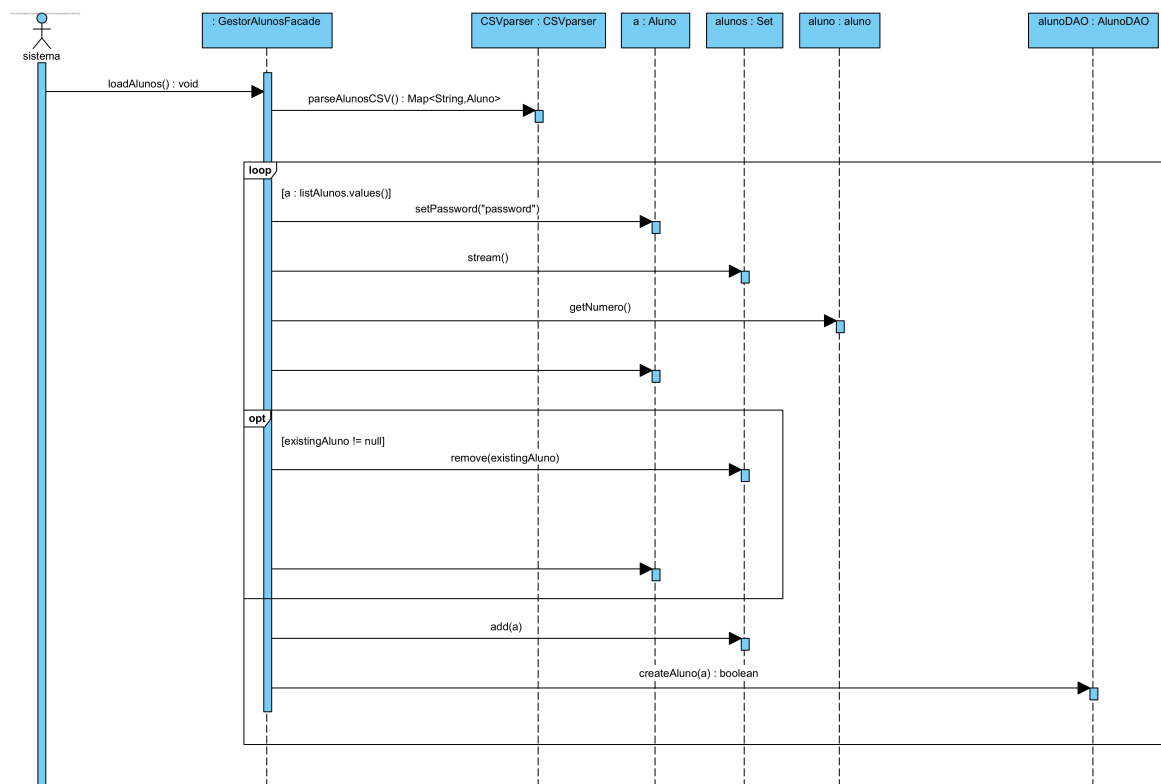


Figura 10: Diagrama de Sequência de Load Alunos

6. Diagrama de Componentes

Visual Paradigm Standard (Ricardo) (Universidade do Minho)

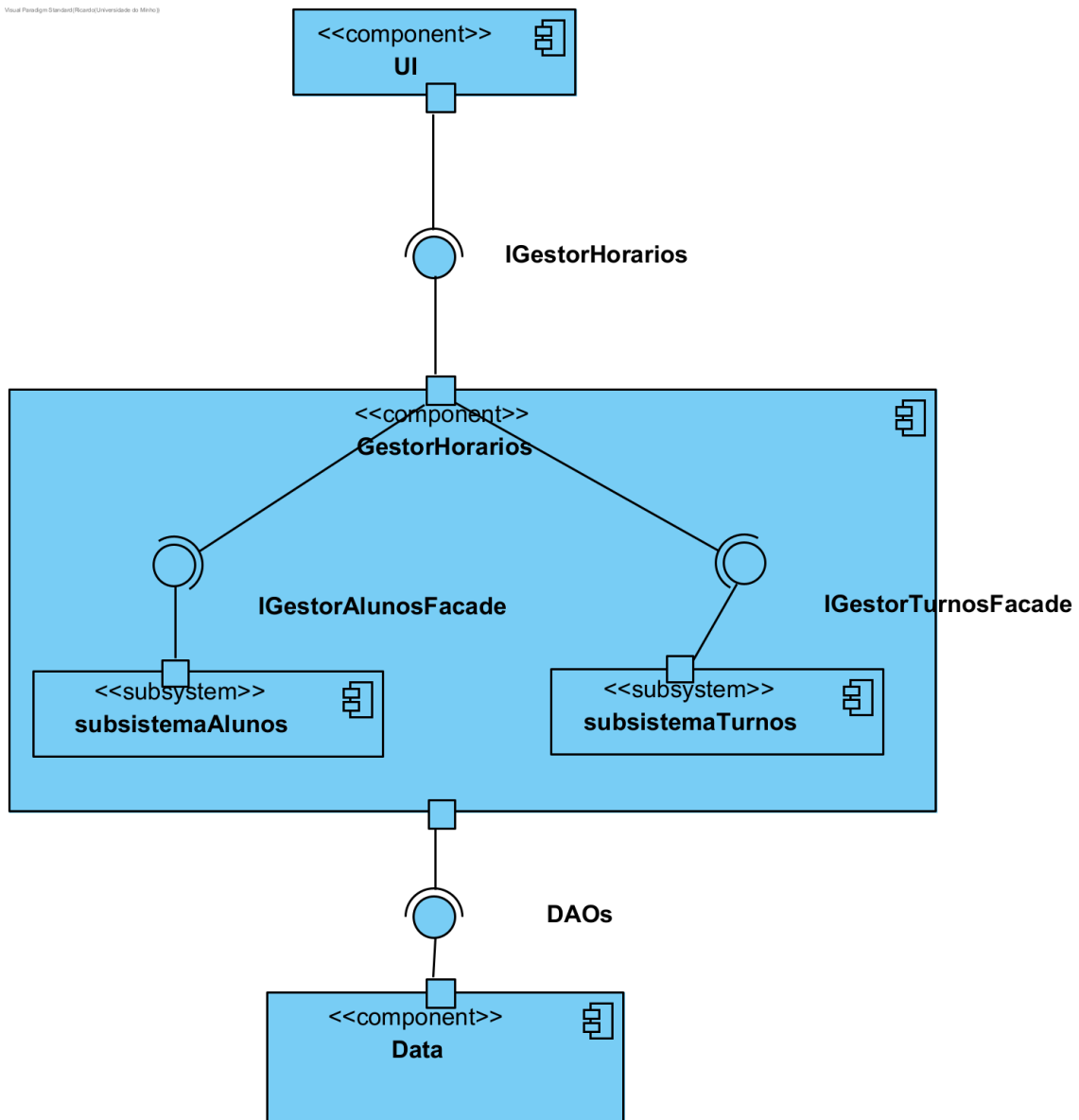


Figura 11: Diagrama de Componentes

7. Estrutura da Base de Dados

A estrutura da base de dados utilizada pelo sistema é representada pelas seguintes tabelas:

- **Alunos:** Armazena informações dos alunos, incluindo nome, número, e-mail e estado. A tabela também contém as relações entre os alunos e as Unidades Curriculares (UCs) a que estão inscritos.
- **Turnos:** Contém informações sobre turnos, como código, tipo, horários, sala e capacidade.
- **Horários:** Regista as associações de alunos a turnos, garantindo que as inscrições respeitem as regras de não sobreposição.
- **Unidades Curriculares (UCs):** Guarda detalhes das UCs, incluindo nome e código, e estabelece relações com os turnos disponíveis.

7.1. Detalhes Técnicos

- As tabelas foram criadas com base no script `SQLCreate.sql` e são povoadas inicialmente através de `SQLPopulate.sql` e dos ficheiros CSV.
- Os DAOs (`AlunoDAO`, `TurnoDAO`, etc.) interagem diretamente com estas tabelas para garantir a consistência dos dados entre a aplicação e a base de dados.

7.2. Operações Importantes

- **Inscrição de Alunos em Turnos:** O sistema verifica automaticamente a capacidade dos turnos e evita conflitos de horário antes de adicionar um aluno.
- **Consulta de Dados:** Métodos como `getAlunosList` e `getTurnosList` permitem extrair informações completas e bem estruturadas para exibição e manipulação.

8. Utilização exemplo do programa

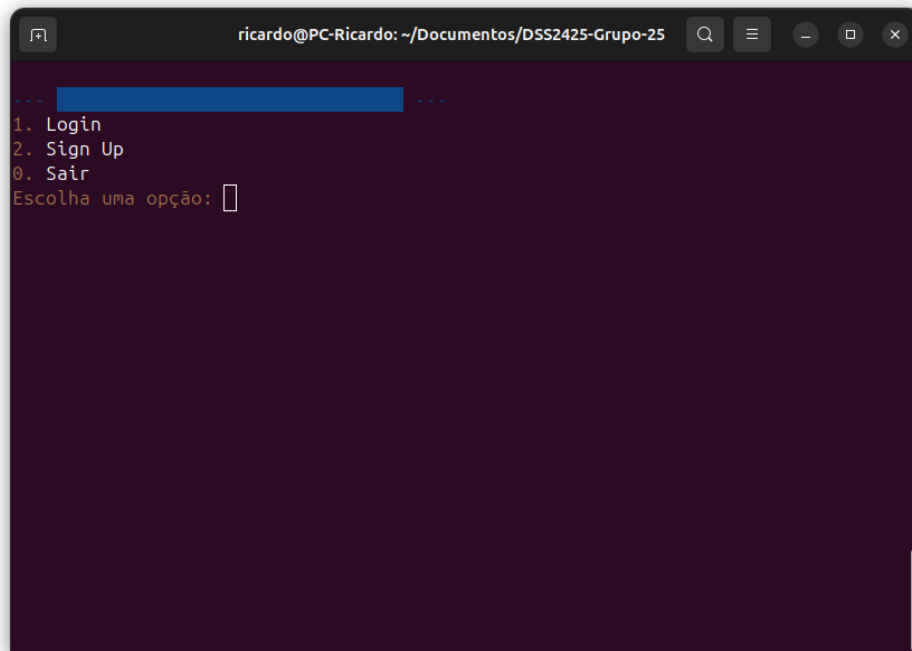


Figura 12: Menu inicial.

1 - O menu inicial aparecerá ao iniciar a aplicação e permite o início de sessão de um aluno ou um diretor de curso/regente de UC. A opção de registo/“Sign Up” foi mantida para efeitos de testagem da aplicação, dado que, e segundo os requisitos enunciados para o projeto. Apenas o Diretor de Curso poderá realizar este registo.

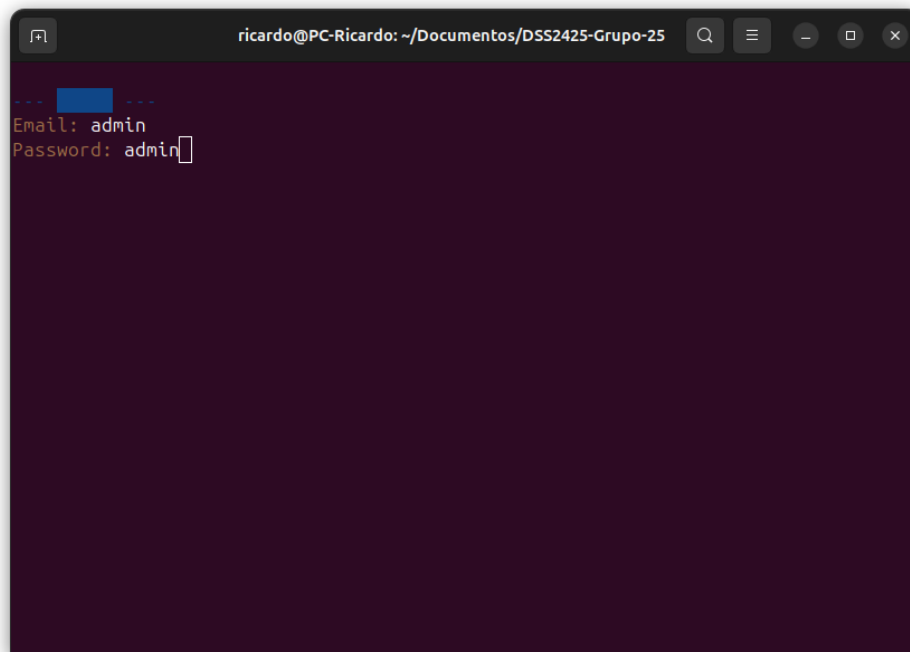


Figura 13: Página de Login.

2 - Tela de login do sistema, permitindo o acesso de administradores e alunos. Aqui vemos o login como admin (utilizador colocado através de um script de pré-povoamento SQL).

2.1 - Após o início de sessão estes são redirecionados para dois tipos de menus diferentes. Um caso este seja um Diretor de Curso ou outro caso seja um aluno.

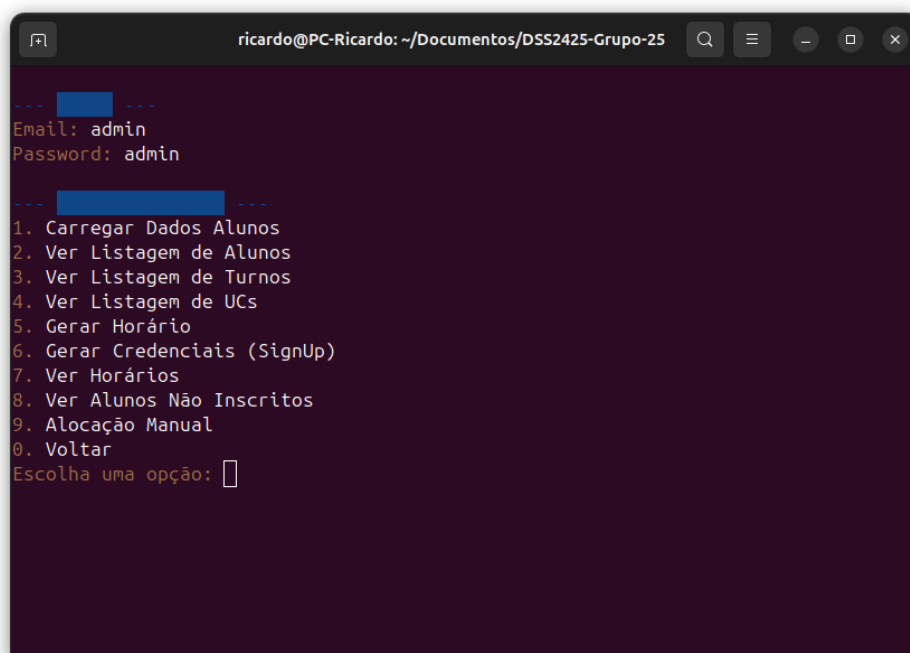
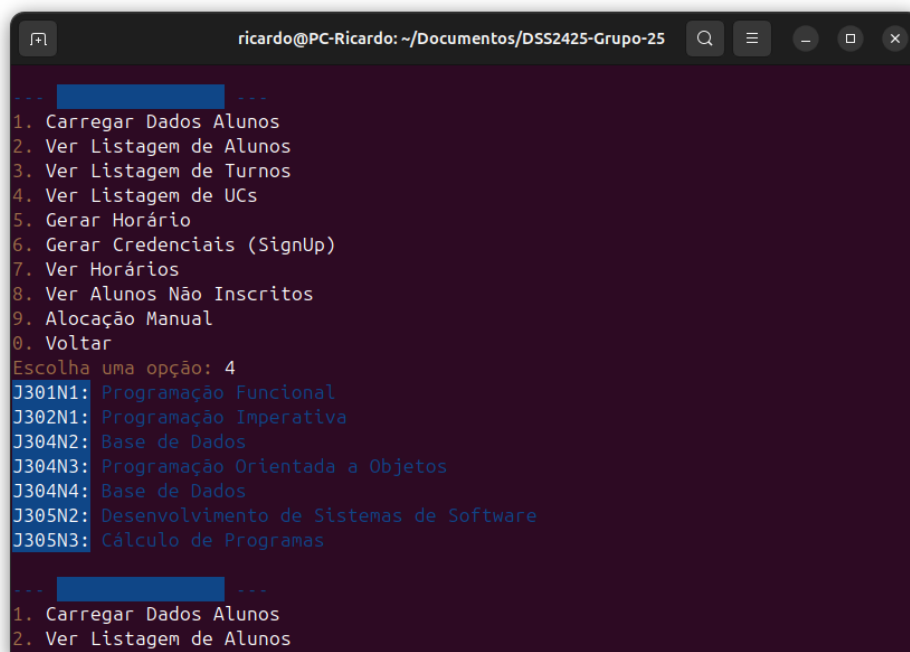


Figura 14: Menu Diretor Curso.

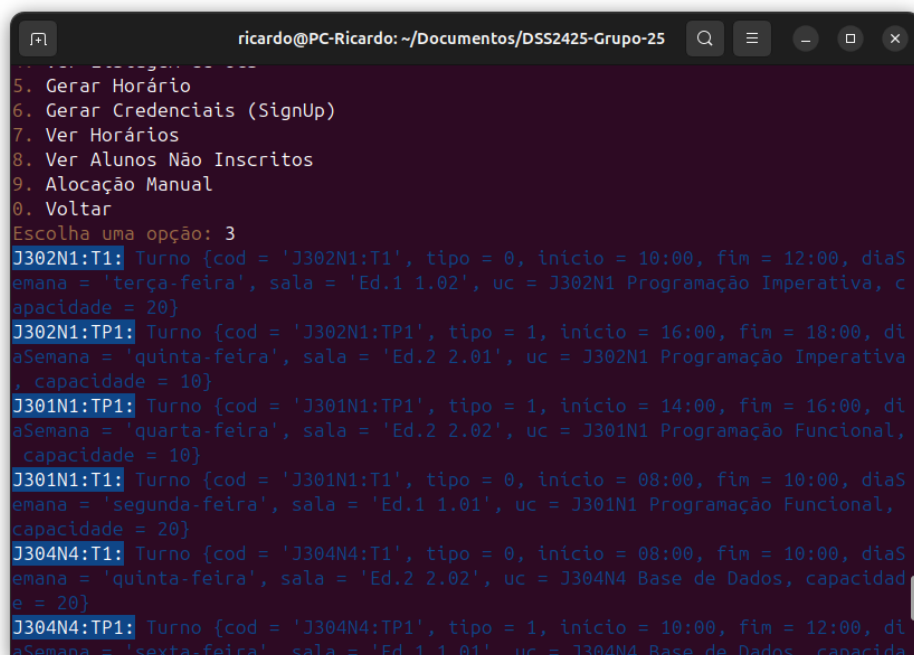
3 - No menu de diretor o utilizador terá acesso a opções relativas à manipulação da informação e dados da aplicação. De entre outras opções do menu, este deverá (num primeiro início da aplicação) realizar a leitura/carregamento dos dados de um ficheiro CSV para a base de dados do programa através da opção 1.



```
ricardo@PC-Ricardo: ~/Documentos/DSS2425-Grupo-25
---
1. Carregar Dados Alunos
2. Ver Listagem de Alunos
3. Ver Listagem de Turnos
4. Ver Listagem de UCs
5. Gerar Horário
6. Gerar Credenciais (SignUp)
7. Ver Horários
8. Ver Alunos Não Inscritos
9. Alocação Manual
0. Voltar
Escolha uma opção: 4
J301N1: Programação Funcional
J302N1: Programação Imperativa
J304N2: Base de Dados
J304N3: Programação Orientada a Objetos
J304N4: Base de Dados
J305N2: Desenvolvimento de Sistemas de Software
J305N3: Cálculo de Programas
---
1. Carregar Dados Alunos
2. Ver Listagem de Alunos
```

Figura 15: Menu Diretor Curso - Ver UCs.

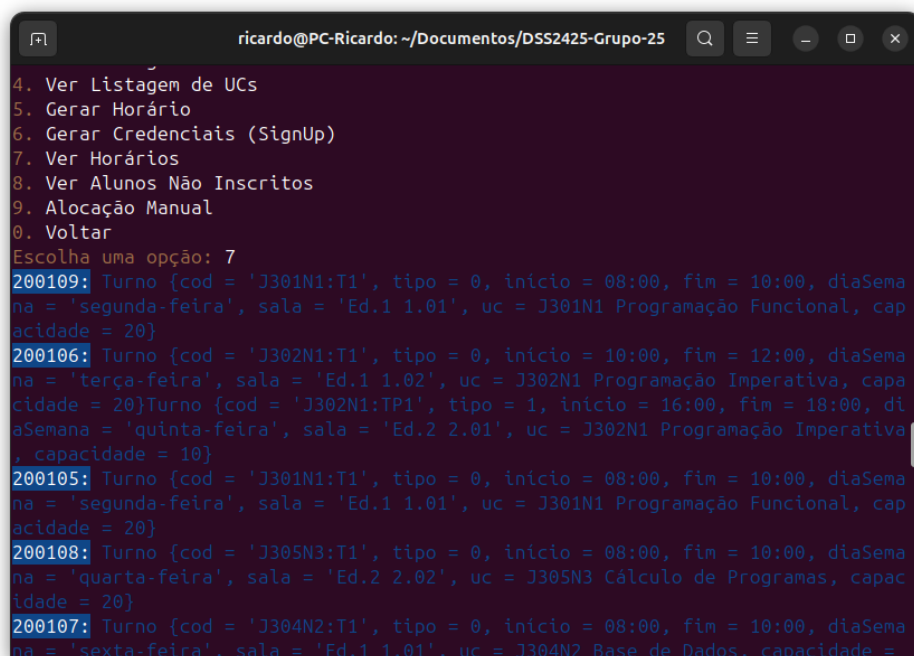
4 - Nesta opção do menu do Diretor de Curso é exibida uma listagem de Unidades Curriculares disponíveis, com os respetivos códigos e descrições.



```
ricardo@PC-Ricardo: ~/Documentos/DSS2425-Grupo-25
5. Gerar Horário
6. Gerar Credenciais (SignUp)
7. Ver Horários
8. Ver Alunos Não Inscritos
9. Alocação Manual
10. Voltar
Escolha uma opção: 3
J302N1:T1: Turno {cod = 'J302N1:T1', tipo = 0, inicio = 10:00, fim = 12:00, diaS
emana = 'terça-feira', sala = 'Ed.1 1.02', uc = J302N1 Programação Imperativa, c
apacidade = 20}
J302N1:TP1: Turno {cod = 'J302N1:TP1', tipo = 1, inicio = 16:00, fim = 18:00, di
aSemana = 'quinta-feira', sala = 'Ed.2 2.01', uc = J302N1 Programação Imperativa
, capacidade = 10}
J301N1:TP1: Turno {cod = 'J301N1:TP1', tipo = 1, inicio = 14:00, fim = 16:00, di
aSemana = 'quarta-feira', sala = 'Ed.2 2.02', uc = J301N1 Programação Funcional,
capacidade = 10}
J301N1:T1: Turno {cod = 'J301N1:T1', tipo = 0, inicio = 08:00, fim = 10:00, diaS
emana = 'segunda-feira', sala = 'Ed.1 1.01', uc = J301N1 Programação Funcional,
capacidade = 20}
J304N4:T1: Turno {cod = 'J304N4:T1', tipo = 0, inicio = 08:00, fim = 10:00, diaS
emana = 'quinta-feira', sala = 'Ed.2 2.02', uc = J304N4 Base de Dados, capacida
de = 20}
J304N4:TP1: Turno {cod = 'J304N4:TP1', tipo = 1, inicio = 10:00, fim = 12:00, di
aSemana = 'sexta-feira', sala = 'Ed.1 1.01', uc = J304N4 Base de Dados, capacida
```

Figura 16: Menu Diretor Curso - Ver Turnos

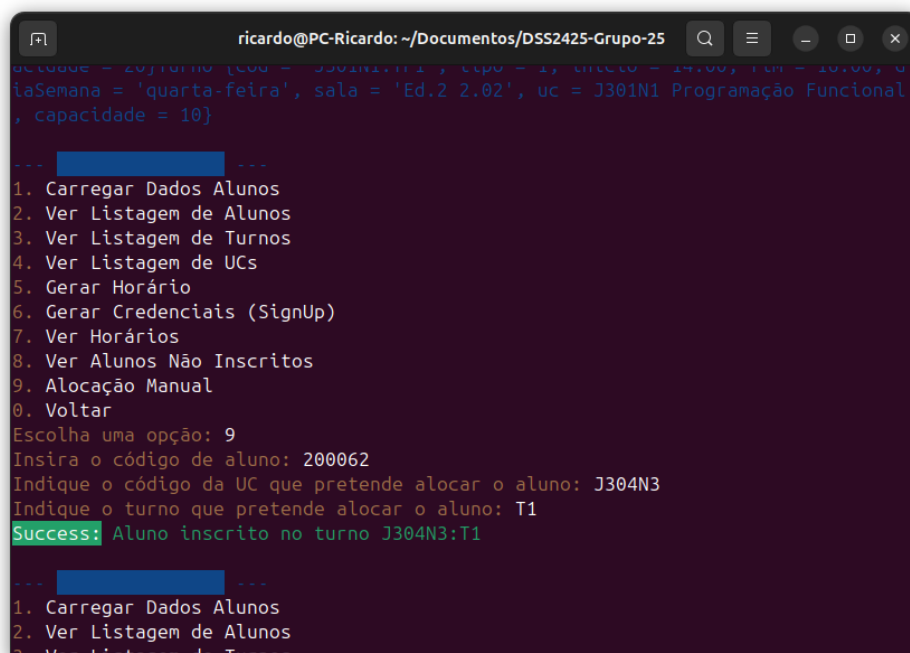
5 - Nesta opção do menu do Diretor de Curso é exibida uma listagem detalhada dos turnos disponíveis, incluindo informações de horário, sala, e capacidade.



```
ricardo@PC-Ricardo: ~/Documentos/DSS2425-Grupo-25
4. Ver Listagem de UCs
5. Gerar Horário
6. Gerar Credenciais (SignUp)
7. Ver Horários
8. Ver Alunos Não Inscritos
9. Alocação Manual
10. Voltar
Escolha uma opção: 7
200109: Turno {cod = 'J301N1:T1', tipo = 0, inicio = 08:00, fim = 10:00, diaSema
na = 'segunda-feira', sala = 'Ed.1 1.01', uc = J301N1 Programação Funcional, cap
acidade = 20}
200106: Turno {cod = 'J302N1:T1', tipo = 0, inicio = 10:00, fim = 12:00, diaSema
na = 'terça-feira', sala = 'Ed.1 1.02', uc = J302N1 Programação Imperativa, capa
cidade = 20}
200105: Turno {cod = 'J302N1:TP1', tipo = 1, inicio = 16:00, fim = 18:00, di
aSemana = 'quinta-feira', sala = 'Ed.2 2.01', uc = J302N1 Programação Imperativa
, capacidade = 10}
200108: Turno {cod = 'J301N1:T1', tipo = 0, inicio = 08:00, fim = 10:00, diaSema
na = 'segunda-feira', sala = 'Ed.1 1.01', uc = J301N1 Programação Funcional, cap
acidade = 20}
200107: Turno {cod = 'J305N3:T1', tipo = 0, inicio = 08:00, fim = 10:00, diaSema
na = 'quarta-feira', sala = 'Ed.2 2.02', uc = J305N3 Cálculo de Programas, capac
idade = 20}
200107: Turno {cod = 'J304N2:T1', tipo = 0, inicio = 08:00, fim = 10:00, diaSema
na = 'sexta-feira', sala = 'Ed.1 1.01', uc = J304N2 Base de Dados, capacidade =
```

Figura 17: Menu Diretor Curso - Ver Horarios.

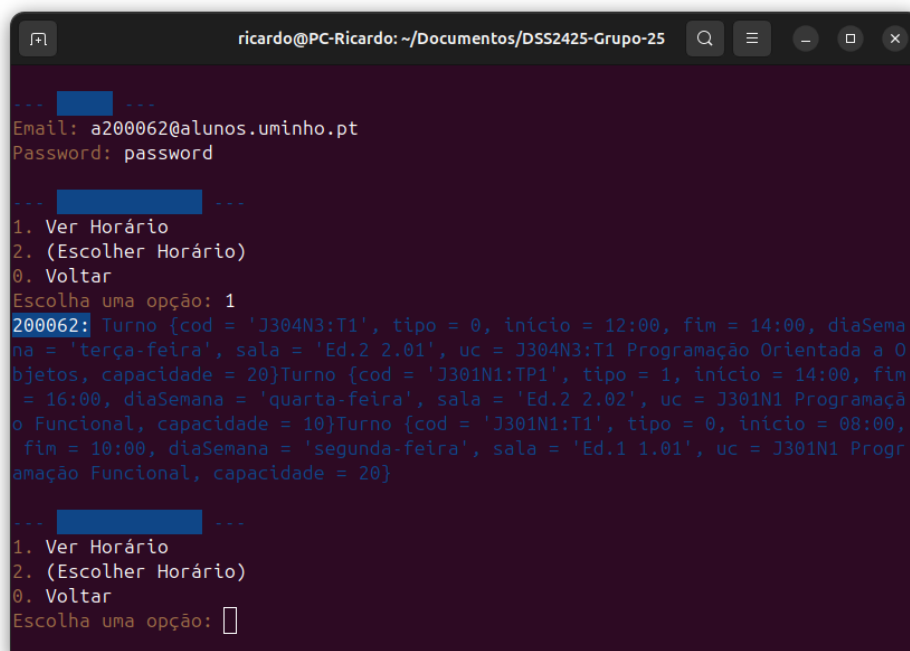
6 - Nesta opção do menu do Diretor de Curso é exibida uma listagem de horários disponíveis, com informações detalhadas dos turnos, incluindo código, tipo, horário, sala, e capacidade.



```
ricardo@PC-Ricardo: ~/Documentos/DSS2425-Grupo-25
---
1. Carregar Dados Alunos
2. Ver Listagem de Alunos
3. Ver Listagem de Turnos
4. Ver Listagem de UCs
5. Gerar Horário
6. Gerar Credenciais (SignUp)
7. Ver Horários
8. Ver Alunos Não Inscritos
9. Alocação Manual
0. Voltar
Escolha uma opção: 9
Insira o código de aluno: 200062
Indique o código da UC que pretende alocar o aluno: J304N3
Indique o turno que pretende alocar o aluno: T1
Success: Aluno inscrito no turno J304N3:T1
---
1. Carregar Dados Alunos
2. Ver Listagem de Alunos
3. Ver Listagem de Turnos
```

Figura 18: Menu Diretor Curso - Alocação Manual.

7 - Nesta opção é exibida a função de alocação manual, que permite a inscrição direta de alunos em turnos específicos.



```
ricardo@PC-Ricardo: ~/Documentos/DSS2425-Grupo-25
---
Email: a200062@alunos.uminho.pt
Password: password
---
1. Ver Horário
2. (Escolher Horário)
0. Voltar
Escolha uma opção: 1
200062: Turno {cod = 'J304N3:T1', tipo = 0, inicio = 12:00, fim = 14:00, diaSema
na = 'terça-feira', sala = 'Ed.2 2.01', uc = J304N3:T1 Programação Orientada a O
bjetos, capacidade = 20}Turno {cod = 'J301N1:TP1', tipo = 1, inicio = 14:00, fim
= 16:00, diaSemana = 'quarta-feira', sala = 'Ed.2 2.02', uc = J301N1 Programaçã
o Funcional, capacidade = 10}Turno {cod = 'J301N1:T1', tipo = 0, inicio = 08:00,
fim = 10:00, diaSemana = 'segunda-feira', sala = 'Ed.1 1.01', uc = J301N1 Progr
amação Funcional, capacidade = 20}
---
1. Ver Horário
2. (Escolher Horário)
0. Voltar
Escolha uma opção: 
```

Figura 19: Menu Aluno - Ver Próprio Horário.

8 - Nesta opção do menu do Aluno temos a visualização de horários de um aluno em específico, com detalhes dos turnos inscritos.

9. Integração CSV

Os ficheiros CSV fornecem uma fonte inicial de dados para o sistema:

- **alunos.csv**: Contém informações detalhadas dos alunos e as Unidades Curriculares associadas.
- **turnos.csv**: Inclui dados dos turnos, como horários, capacidade e relação com as UCs.

9.1. Processo de Importação

O CSVparser realiza as seguintes etapas:

1. **Leitura do Ficheiro**: As linhas são lidas sequencialmente.
2. **Validação de Dados**: Linhas malformadas (e.g. com colunas incompletas) são registadas em logs para futura análise.
3. **Inserção na Base de Dados**: Os dados válidos são inseridos utilizando métodos de DAOs, como `createAluno` e `createTurno`.

9.2. Melhorias Implementadas

- **Logs de Erros**: Registo detalhado de linhas inválidas para facilitar a depuração e correção.
- **Carga Inicial de Dados**: Automatização da integração com a base de dados para garantir consistência.

10. Conclusão

Neste projeto, foi possível desenvolver e apresentar um sistema funcional para a gestão de turnos em cursos de grande dimensão, como é o caso da Licenciatura em Engenharia Informática. Através da análise detalhada dos requisitos e da implementação cuidadosa, foi possível atingir os seguintes objetivos principais:

- **Gestão de Dados:** A integração de ficheiros CSV e a utilização de uma base de dados estruturada permitiram o carregamento e a gestão eficiente de informações relacionadas a alunos, turnos e unidades curriculares.
- **Interatividade:** O sistema oferece uma interface de menu clara e intuitiva, que suporta diversas funcionalidades como visualização de dados, geração de horários e alocação manual de alunos.
- **Consistência e Feedback:** As mensagens de sucesso e erro exibidas durante as operações garantem uma boa experiência ao utilizador, permitindo que ações incorretas sejam identificadas e corrigidas facilmente.

Apesar do progresso significativo, identificaram-se algumas áreas que podem ser melhoradas em fases futuras:

- **Escalabilidade:** Adicionar suporte para um maior número de utilizadores e turnos.
- **Otimização:** Melhorar a eficiência de consultas à base de dados, especialmente em cenários com muitos alunos e turnos.
- **Interface Gráfica:** Evoluir para uma interface gráfica que ofereça uma experiência mais moderna e visual aos utilizadores.

Acreditamos que o trabalho desenvolvido ofereceu uma base sólida no que toca ao desenvolvimento de sistemas de software, promovendo um progresso mais eficiente e alinhado com os objetivos definidos inicialmente. Através da implementação cuidadosa e da análise detalhada de requisitos, conseguimos criar um sistema funcional, intuitivo e adaptado às necessidades identificadas.

As aprendizagens obtidas ao longo deste projeto reforçaram a importância de uma abordagem estruturada, desde a modelação inicial até à implementação e validação das funcionalidades. Este processo não só permitiu consolidar os nossos conhecimentos em desenvolvimento de software, como também evidenciou o impacto positivo de boas práticas de organização, colaboração e gestão de projetos.

Consideramos que os resultados alcançados são um reflexo direto do empenho e da aplicação dos conhecimentos adquiridos ao longo do curso, oferecendo-nos uma visão prática e integrada sobre o processo de construção de sistemas de software.