

Universidade do Minho
Escola de Engenharia
Licenciatura em Engenharia Informática

Unidade Curricular de Interface Pessoa Máquina

Ano Letivo de 2024/2025

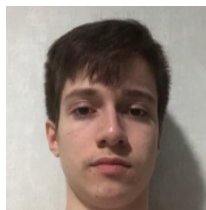
https://github.com/UMinho-ENGINF-IPM/trabalho-pr-tico-gp25_28

<https://www.figma.com/design/tlZRAiuMmjEUCaUK5fmZPd/IPM-Trabalho?node-id=0-1&t=1CskxUj0VysLg9Fd-1>

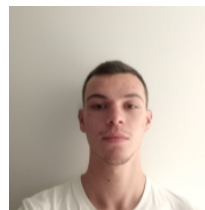
2ª Fase - Interface Final



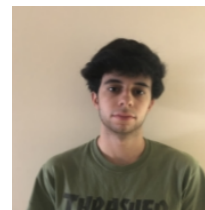
Pedro Ferreira
A97646



Enzo Vieira
A98352



Luís Figueiredo
A100549



Gustavo Barros
A100656



Miguel Gramoso
A100835

May 05, 2025

IPM

Índice

1. Introdução	2
2. Objetivos	3
3. Alterações do Protótipo	4
3.1. Apresentação de horários	4
3.2. Perfil de aluno	4
3.3. Apresentação dos conflitos	4
3.4. Gestão de UC's	4
3.5. Pedidos	4
4. Implementação	5
4.1. Estrutura	5
4.1.1. Componentes	5
4.1.2. Navegação	5
4.1.3. Services	6
4.2. Bibliotecas Utilizadas	6
4.2.1. FullCalendar	6
4.2.2. Axios	6
4.2.3. pa11y	6
4.2.4. Vue Router	6
4.2.5. Pinia	6
5. Reflexão sobre a aplicação	7
5.1. Testes	7
5.1.1. PA11Y	7
5.2. Cenários de Utilização	7
5.3. Pontos Fortes vs. Pontos Fracos	8
5.3.1. Pontos Fortes	8
5.3.2. Pontos Fracos	8
6. Manual de Utilização	9
6.1. Instalação e Execução da Aplicação	9
6.2. Diretor de Curso	9
6.3. Aluno	18
7. Reflexão sobre o Trabalho	23
7.1. Dificuldades sentidas	23
7.2. Aspetos positivos do Vue	23
8. Conclusão	24

1. Introdução

Este relatório foi desenvolvido para a 2ª Fase do trabalho prático da Unidade Curricular de Interface Pessoa Máquina. Neste trabalho, foi nos pedido que desenvolvamos uma interface para apoiar o Diretor de Curso na criação, visualização e ajuste dos turnos de aulas do curso.

Ao longo deste projeto, fomos recolhemos feedback da equipa docente e de utilizadores de teste, de maneira a obter uma aplicação final da qual ficamos satisfeitos, e sem infringir os princípios estudados ao longo do semestre.

O trabalho foi estruturado em duas fases: um protótipo em Figma, seguido da implementação em Vue.js, recorrendo a um backend simulado com json-server.

2. Objetivos

O principal objetivo deste trabalho prático consiste em conceber e implementar uma interface de utilizador que facilite a gestão de turnos para um curso de grande dimensão.

Pretende-se entregar uma interface que seja intuitiva, fácil de usar, robusta e que facilite a gestão dos turnos de uma eventual direção de curso.

Em geral, pretende-se:

- Permitir uma alocação automática dos alunos e turnos, de modo a reduzir o esforço burocrático (envolvido na distribuição manual) e atribuir prioridades conforme o perfil dos alunos (ex.: primeira inscrição, média, IDs).
- Disponibilizar mecanismos de controlo e intervenção manual, permitindo ao Diretor de Curso ajustar alocações em caso de conflitos.
- Funcionalidades de consulta e edição que suportem os cenários descritos no enunciado: gestão de pedidos de alteração de turno por alunos, mudança de sala para turnos existentes e consulta de horário dos alunos.
- Assegurar que a interface responde às necessidades dos dois perfis de utilizador identificados (Diretor de Curso e Estudante), fornecendo vistas claras de horários, alertas de conflito e fluxos de interação intuitivos e práticos.
- Validar a proposta de design perante heurísticas de usabilidade, garantindo consistência, prevenção de erros e eficiência de uso.

3. Alterações do Protótipo

3.1. Apresentação de horários

Após testar o protótipo inicial, concluímos que apresentar os turnos em lista — tal como nas aplicações atuais do curso de Engenharia Informática — era pouco intuitivo e dificultava a identificação e gestão de conflitos, podendo levar a erros por parte de alunos e professores. Por isso, na versão final adotaremos uma interface em formato de calendário, inspirada nos horários da UMinho e no Calendarium do CeSIUM, que é mais familiar e está alinhada às preferências dos utilizadores.

3.2. Perfil de aluno

Seguindo a alteração anterior, o diretor de curso também deverá ter acesso rápido e facilitado ao horário de alunos de interesse, seja devido a alunos possuírem estatuto, ou por pedidos pessoais de troca de turnos.

O novo perfil de aluno permitirá analisar horários atuais, estudar possibilidades de troca, consultar turnos inscritos, etc...

3.3. Apresentação dos conflitos

O principal foco da aplicação deve ser a gestão dos alunos por turnos, tentando garantir o máximo possível de horários completos. Por esse motivo, decidimos dar maior destaque aos conflitos existentes. O diretor de curso deve ter sempre maior foco na resolução dos conflitos existentes, e a nossa interface necessita de demonstrar isso mesmo.

3.4. Gestão de UC's

O Diretor de Curso poderá receber pedidos sem ser por parte da sua equipa docente. Este tipo de pedidos é focado em pequenas alterações aos turnos de uma específica Unidade curricular, revelando assim mais uma necessidade para os utilizador, a possibilidade de editar um turno, alterando, como por exemplo, a sala associada.

3.5. Pedidos

Solicitar trocas de turno através de um único campo de texto revelou-se impraticável para alunos e Diretor, além de não corresponder ao objetivo do sistema. Para otimizar o processo, os pedidos são agora submetidos via formulário, onde o aluno seleciona a UC para troca de turno e visualiza simultaneamente no seu calendário a alteração proposta. De modo idêntico, o Diretor, ao aceder aos detalhes do pedido, vê também as modificações no calendário do aluno.

4. Implementação

O projeto foi desenvolvido com Vue 3, Vite, TypeScript para tipagem e utilizando maioritariamente a Options API. A estrutura do projeto está bem organizada, com separação de responsabilidades em diferentes pastas e ficheiros.

4.1. Estrutura

4.1.1. Componentes

De forma a manter a consistência da interface e criar componentes reutilizáveis, alguns componentes criados foram, dentre eles, os mais simples foram:

- `Link.vue`: componente de navegação integrado com o Vue Router (Secção 4.2.4), estilizado para manter a consistência da interface.
- `Button.vue`: botão genérico com variantes de estado (loading, disabled, holding) e estilos configuráveis (primary, secondary, destructive, etc.).
- `Card/`: conjunto de componentes (`Card.vue`, `CardTitle.vue`, `CardContent.vue`) para agrupar e destacar secções de conteúdo.
- `Table/`: elementos para construção de tabelas (`TableHeader.vue`, `TableBody.vue`, `TableRow.vue`).
- `Skeleton.vue`: placeholder visual que simula carregamento de conteúdos durante chamadas à API.

Com o mesmo propósito, visando encapsular o uso de componentes mais complexos, outros componentes criados foram:

- `Calendar.vue`: Componente para exibição de eventos em formato de calendário interativo, através do FullCalendar (Secção 4.2.1).
- `Combobox.vue`: campo de seleção com filtragem dinâmica em listas extensas, otimizado para pesquisa rápida.
- `Dropdown.vue`: Visualização e seleção de listas menores, sem opção de filtragem.
- `LogoutButton.vue`: botão que encapsula a lógica de limpeza do estado do utilizador em Pinia (Secção 4.2.5) e redirecciona para a página de login.

4.1.2. Navegação

Todas as páginas completas da aplicação residem em `src/pages/`, cada uma implementada como um componente Vue que corresponde a uma vista distinta (e.g., Login, Lista de UCs, Detalhes de Curso, Caixa de Pedidos). A configuração das rotas encontra-se em `src/router/index.ts`, onde cada rota mapeia um path para um destes componentes. Utilizam-se navigation guards do Vue Router (Secção 4.2.4) para validar autenticação e autorização antes de conceder acesso a áreas restringidas e para gerir redirecionamentos.

4.1.3. Services

Em `/services` estão encapsuladas todas as funcionalidades de chamada à API com Axios (Secção 4.2.2) e as regras de negócios.

4.2. Bibliotecas Utilizadas

4.2.1. FullCalendar

O componente mais reutilizado na aplicação foi construído sobre a biblioteca FullCalendar. Graças a ele, conseguimos apresentar aos utilizadores uma visualização familiar dos horários dos alunos e dos turnos de cada UC, num formato reconhecível e de fácil compreensão. Além disso, o componente permitiu implementar interações essenciais, como a seleção direta de um turno no calendário e a identificação visual dos estados dos turnos — em conflito, selecionados, inscrito, disponíveis ou lotados — destacados de acordo com o seu estado.

4.2.2. Axios

Juntamente com o `json-server`, a biblioteca do Axios nos permitiu fazer chamadas à API do backend para visualização, edição, remoção e criação de dados.

4.2.3. pa11y

Para assegurar a acessibilidade em todas as páginas da aplicação, recorremos à biblioteca `pa11y`, que se revelou imprescindível para automatizar os testes em todos os componentes da interface.

4.2.4. Vue Router

Para estruturar a navegação entre as várias páginas da aplicação, adotou-se o Vue Router. A configuração de rotas encontra-se em `src/router/index.ts`, utilizando parâmetros dinâmicos para aceder aos detalhes de cada Unidade Curricular e pedido por exemplo. Implementaram-se ainda guardas de rota (`navigation guards`) para proteger páginas que exigem autenticação ou permissão, assegurando que apenas utilizadores autenticados e autorizados podem aceder às funcionalidades internas.

4.2.5. Pinia

Para gerir o estado global da aplicação, recorreu-se ao Pinia como store. Foi definida uma entidade principal — o utilizador autenticado — que o seu próprio store em `src/stores/auth.ts`, com definições de `state`, `getters` e `actions` que encapsulam verificações para lógica de negócio.

5. Reflexão sobre a aplicação

5.1. Testes

Durante o processo de desenvolvimento da nossa aplicação, optamos por testar com recurso a software.

5.1.1. PA11Y

Para confirmar que a nossa interface respeita as Web Content Accessibility Guidelines (WCAG), criamos um *script* que analisa cada página para possíveis erros ou incompatibilidades que possam surgir.

A análise automática com Pa11y identificou inicialmente três categorias de problemas:

1. Contraste insuficiente em componentes coloridos;
2. Uso indevido da role presentation no FullCalendar, que continha tags semânticas internas desencadeando avisos;
3. Inputs sem labels explícitas.

A versão mais atual da nossa aplicação passa no testes do Pa11y com 0 erros a reportar. O erro no calendário exigiu uma solução provisória: remoção manual do atributo role="presentation" via JavaScript, pois o mesmo erro foi indicado por outros utilizadores da biblioteca que sugeriram a remoção manual, visto que a própria biblioteca não possui um bom suporte a acessibilidade e não segue as Guidelines do Pa11y. O relatório HTML do Pa11y, incluído no repositório, demonstra que nenhum alerta permanece.

5.2. Cenários de Utilização

Dos quatro cenários descritos no enunciado, todos foram implementados com sucesso, exceto o pedido de troca de sala por docentes. A alocação automática é simulada por uma *fake promise*, simplificando a lógica sem fugir ao escopo do projeto. A alocação manual e mudança de turno é totalmente funcional, com atualização das informações no json-server e reflexo no protótipo interativo. Os pedidos de alteração de turno por alunos possuem também a funcionalidade completa, com o fluxo minimizado ao máximo para otimizar o processo de criação e aceitação/rejeição dos pedidos por parte do Aluno/Diretor.

5.3. Pontos Fortes vs. Pontos Fracos

5.3.1. Pontos Fortes

A simplicidade e intuitividade como ponto forte em interfaces limpas, com apenas as informações essenciais em cada ecrã, minimizando as navegações desnecessárias por parte do utilizador.

A adaptação automática a ecrãs menores é outro ponto forte da aplicação, sendo ela inteiramente responsiva a qualquer tamanho de ecrã, seja Desktop ou Mobile.

Feedback ao utilizador foram considerados na implementação e os componentes exibem estados de carregamento, sucesso e erro, promovendo transparência em cada interação.

Para diminuir a carga cognitiva do utilizador, foi fundamental manter a consistência em toda a aplicação por meio de componentes reutilizáveis, de uma paleta de cores simplificada e de uma tipografia cuidadosamente escolhida; esses elementos reforçam a identidade e tornam o uso mais intuitivo.

5.3.2. Pontos Fracos

A atual configuração dos dados de teste apresenta inconsistências, dado que o ficheiro JSON inclui registos de alunos inscritos em Unidades Curriculares nas quais não estão matriculados. Esta falta de correspondência entre dados limita a fiabilidade dos testes, especialmente quando se pretende avaliar a aplicação com cenários de dados mais diversificados. Durante os testes realizados, foram utilizados sempre os mesmos alunos, preferencialmente aqueles com os menores IDs (Mafalda, ID 1, p.e.), o mesmo acontece para as Unidades Curriculares, na qual a de Matemática foi preferencialmente testada.

A cobertura de casos de uso mantém um hiato relevante na funcionalidade de pedidos de troca de sala por parte dos docentes, ainda por implementar. Esta lacuna impede o sistema de suportar integralmente um dos cenários previstos, reduzindo a completude da solução.

Por fim, até ao momento não se realizaram testes com utilizadores reais. Esta ausência de feedback direto de utilizadores compromete a capacidade de ajustar a interface a problemas de usabilidade que apenas emergem em contextos de uso reais.

6. Manual de Utilização

Este manual descreve como utilizar a aplicação desenvolvida para gestão de turnos práticos nas Unidades Curriculares (UC) do curso. A aplicação é dirigida a Diretores de Curso, Docentes e Alunos.

6.1. Instalação e Execução da Aplicação

1. Instalar dependências: `'npm install'`
2. Iniciar o servidor através do comando: `'npm run server'`
3. Iniciar a aplicação através do comando: `'npm run dev'`

6.2. Diretor de Curso

O *login* é realizado através do email institucional e a password do diretor de curso.

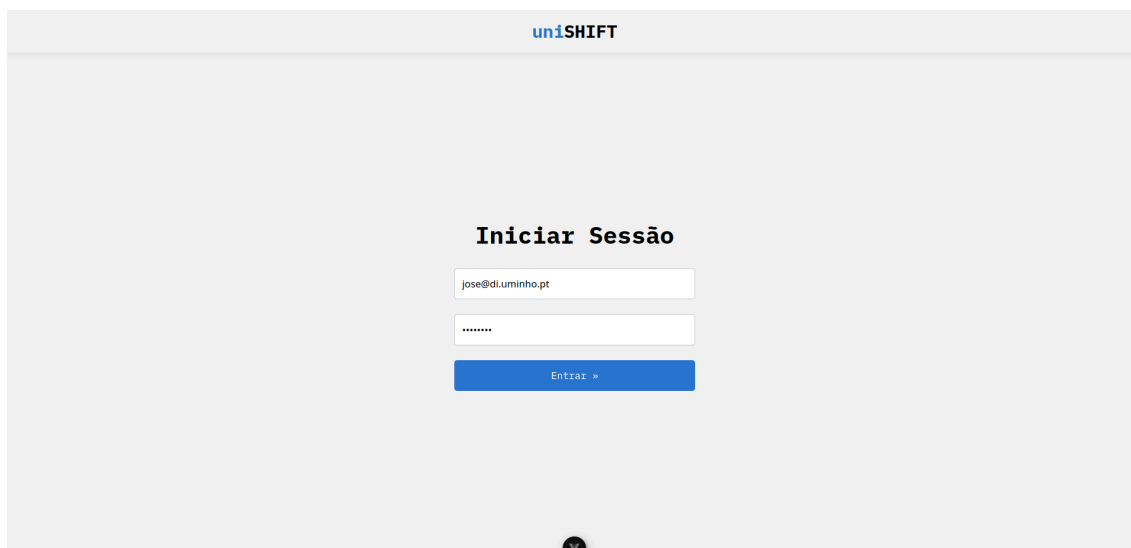


Figura 1: Login como Diretor de Curso

O Diretor de curso, após o *login*, é encaminhado para a página de gestão das UC's do Curso. Nesta página é possível navegar para outras páginas através da *Navbar* ou aceder aos detalhes de cada UC listada.

uniSHIFT

Unidades Curriculares

Podidos 2

Conflitos 4

Alocação

jose@di.uninho.pt

SAIR

Unidades Curriculares

1º ANO

Semestre	Nome	Alunos	Alocações Conflituosas	
1ª	Matemática	5	20%	Detalhes
1ª	Fundamentos de Programação	5	20%	Detalhes
1ª	Introdução à Computação	5	0%	Detalhes
1ª	Álgebra Linear	5	0%	Detalhes
1ª	Lógica para Computação	5	0%	Detalhes
2ª	Cálculo	5	20%	Detalhes
2ª	Programação Orientada a Objetos	5	0%	Detalhes
2ª	Estruturas de Dados	5	0%	Detalhes
2ª	Arquitetura de Computadores	5	0%	Detalhes
2ª	Sistemas Operativos	5	0%	Detalhes

2º ANO

Semestre	Nome	Alunos	Alocações Conflituosas	
1ª	Bases de Dados	5	20%	Detalhes
1ª	Redes de Computadores	5	0%	Detalhes
1ª	Engenharia de Software	5	0%	Detalhes
1ª	Inteligência Artificial	5	0%	Detalhes
1ª	Sistemas Distribuídos	5	0%	Detalhes
2ª	Compiladores	5	0%	Detalhes
2ª	Segurança Informática	5	0%	Detalhes
2ª	Interação Humano-Computador	5	0%	Detalhes
2ª	Grafos e Algoritmos	5	0%	Detalhes
2ª	Processamento de Linguagens	5	0%	Detalhes

3º ANO

Semestre	Nome	Alunos	Alocações Conflituosas	
1ª	Aprendizagem de Máquina	5	0%	Detalhes
1ª	Sistemas de Informação	5	0%	Detalhes
1ª	Computação Gráfica	5	0%	Detalhes
1ª	Redes de Nova Geração	5	0%	Detalhes
1ª	Sistemas Embarcados	5	0%	Detalhes
2ª	Projeto de Engenharia Informática	5	0%	Detalhes
2ª	Tópicos Avançados em Computação	5	0%	Detalhes
2ª	Gestão de Projetos de Software	5	0%	Detalhes
2ª	Computação Móvel	5	0%	Detalhes
2ª	Sistemas de Tempo Real	5	0%	Detalhes

© Interface Pessoa-Máquina 2025

Figura 2: Gestão de UC's

Ao clicar em “**Detalhes**” de uma UC na página de Gestão de UC’s, entramos na página de detalhes da UC selecionada.

unishift

Unidades CurricularesPedidos2Conflitos3Alocação

Jose@SL.uninho.ptSAIR

UC: Matemática

DETALHES

Época: 1º ano / 1º semestre

Alunos: 5 (60% alocados em turno)

HORÁRIO

8h					
9h					
10h					
11h	MAT/TP2 3/18 ocupado E1-1.01		MAT/TP1 3/18 ocupado E1-1.01		MAT/TP3 3/18 ocupado E1-1.01
12h					
13h					
14h					
15h					
16h					
17h					
	Mon	Tue	Wed	Thu	Fri

TURNOS

MAT-T2 (Seg 11:00h-13:00h)

Docente: Pedro

3 alunos (75% ocupado)

E1-1.02: 4 lugares

MAT-TP5 (Ter 9:00h-11:00h)

Docente: Pedro

2 alunos (4% ocupado)

E1-1.01: 50 lugares

MAT-TP4 (Qua 9:00h-11:00h)

Docente: Pedro

4 alunos (8% ocupado)

E1-1.01: 50 lugares

MAT-TP1 (Qua 11:00h-13:00h)

Docente: Pedro

3 alunos (6% ocupado)

E1-1.01: 50 lugares

MAT-TP3 (Sex 9:00h-11:00h)

Docente: Pedro

3 alunos (6% ocupado)

E1-1.01: 50 lugares

MAT-TP2 (Sex 11:00h-13:00h)

Docente: Pedro

3 alunos (6% ocupado)

E1-1.01: 50 lugares

MAT-T1 (Sex 11:00h-13:00h)

Docente: Pedro

3 alunos (6% ocupado)

E1-1.01: 50 lugares

MAT-TP6 (Sex 16:00h-18:00h)

Docente: Pedro

3 alunos (6% ocupado)

E1-1.01: 50 lugares

ALUNOS MATRICULADOS

FiltrosNome/Email do aluno

Mafalda

Email: a32324@alunos.uninho.pt

Matriculas prévias: 0

Turnos: T2, TP4

Perfil

Editar turnos

Inês

Email: i32324@alunos.uninho.pt

Matriculas prévias: 0

Turnos:

Perfil

Editar turnos

Ana

Email: a32325@alunos.uninho.pt

Matriculas prévias: 0

Turnos: T2, TP5

Perfil

Editar turnos

Gabriela

Email: g32325@alunos.uninho.pt

Matriculas prévias: 0

Turnos:

Perfil

Editar turnos

Nuno

Email: n32325@alunos.uninho.pt

Matriculas prévias: 0

Turnos: T2, TP5

Perfil

Editar turnos

© Interface Pessoa-Máquina 2025

Figura 3: Detalhes de uma UC

Na *Navbar*, a segunda página disponível é a página de Pedidos. Esta página permite o Diretor de Curso consultar todos os pedidos que lhe chegaram por parte dos alunos ou dos professores.

The screenshot shows the 'Pedidos' page in the uniSHIFT system. The top navigation bar includes 'Unidades Curriculares', 'Pedidos' (highlighted with a red '2'), 'Conflitos' (with a red '4'), and 'Alocação'. The user 'jose@di.uminho.pt' is logged in, with a 'SAIR' button. The main heading is 'Pedidos'. Below it, there are two sections: 'NOVOS PEDIDOS' and 'PEDIDOS FECHADOS'. Each section contains a table with columns 'AUTOR', 'DATA/HORA', 'CURSO', and a 'Detalhes' link.

AUTOR	DATA/HORA	CURSO	
Mafalda	2025/05/01 02:22	Matemática	Detalhes
Mafalda	2025/02/10 13:35	Fundamentos de Programação	Detalhes

AUTOR	DATA/HORA	CURSO	
Mafalda	2025/05/01 02:22	Matemática	Detalhes
Mafalda	2025/02/10 13:35	Matemática	Detalhes

© Interface Pessoa-Máquina 2025

Figura 4: Lista de Pedidos

Ao investigar os “**Detalhes**” de cada pedido, conseguimos investigar melhor o pedido feito, como os detalhes da mensagem, ou o impacto que a troca terá no horário do aluno.

uniSHIFT

Unidades Curriculares

Pedidos 2

Conflitos 4

Alocação

jose@di.uminho.pt

SAIR

Pedido Troca de Turno (MAT)

DETALHES

Pedido pendente

AUTOR: Filipe

EMAIL: f32325@alunos.uminho.pt (Aluno, Estatuto Normal)

ABERTURA: 2025/05/01 02:22

TURNO ATUAL: MAT-TP4

TURNO PEDIDO: MAT-TP6

Sem mensagem do aluno.

HORÁRIO DO ALUNO

8am					
9am			MAT/TP4		
10am			14:00 ocupado		
11am			15-1.01		
12pm	MAT/TP2			FP/TP1	IC/TP1
1pm	14:00 ocupado			15:00 ocupado	16-1.01
2pm	Mon	Tue	Wed	Thu	Fri
3pm					
4pm					
5pm					
6pm					
7pm					

OBSERVAÇÕES (OPCIONAL)

Escreva observações que ache pertinente para o Aluno

Escreva as observações...

Rejeitar

Aceitar

© Interface Pessoa-Máquina 2025

Figura 5: Detalhes de Pedido

A terceira opção presente na *Navbar*, é os conflitos. Esta página lista todos os conflitos existentes, para que o Diretor de Curso consiga acompanhar o estado da alocação atual.

Conflitos de Turno

ALUNOS

Filtros Ordenar Nome/Email do aluno

Mafalda a32324@alunos.uminho.pt	Ver aluno
Conflitos (1): MAT/T1 com FP/TP4 Resolver	
Carlos 33232@alunos.uminho.pt	Ver aluno
Conflitos (1): CAL/T1 com COMP/TP4 Resolver	
Joana j32324@alunos.uminho.pt	Ver aluno
Conflitos (1): BD/T1 com AM/TP4 Resolver	
Ricardo r32324@alunos.uminho.pt	Ver aluno
Conflitos (1): MAT/T1 com MAT/TP4 Resolver	

© Interface Pessoa-Máquina 2025

Figura 6: Lista de Conflitos

Ao clicar em “**Detalhes**” de um Aluno na página de Conflitos entramos na página de detalhes do aluno em conflito. Esta página permite consultar os turnos em qual o estudante está inscrito, e visualizar os conflitos existentes.

uniSHIFT
Unidades Curriculares
Pedidos 2
Conflitos 4
Alocação
jose@di.uminho.pt
SAIR

Aluno: Mafalda

DETALHES

NOME: Mafalda
EMAIL: a32324@alunos.uminho.pt
CARGO: Aluno
ESTATUTO: Especial

HORÁRIO

	Mon	Tue	Wed	Thu	Fri
8am					
9am			MAT/TP4 4/50 ocupado E1-1.01		
10am			FP/T1 4/50 ocupado E1-1.01		IC/T1 3/50 ocupado E1-1.01
11am	MAT/T2 3/4 ocupado E1-1.02				
12pm					
1pm					
2pm				FP/PL1 3/50 ocupado E1-1.01	
3pm					
4pm					
5pm					
6pm					
7pm					

METRÍCULA ATUAL

Matemática
Época: 1º ano / 1º semestre
Matriculas Prévias: 1
Turnos: T2, TP4

Fundamentos de Programação
Época: 1º ano / 1º semestre
Matriculas Prévias: 1
Turnos: T1, PL1

Introdução à Computação
Época: 1º ano / 1º semestre
Matriculas Prévias: 1
Turnos: T1

Álgebra Linear
Época: 1º ano / 1º semestre
Matriculas Prévias: 1
Turnos:

Lógica para Computação
Época: 1º ano / 1º semestre
Matriculas Prévias: 1
Turnos:

© Interface Pessoa-Máquina 2025

Figura 7: Detalhes de um Aluno

A quarta opção presente na *Navbar*, é a Alocação. A alocação é dividida em 2 opções, alocação manual e alocação automática. A alocação manual, permite ao diretor alocar turnos a alunos específicos, os seus turnos individualmente.

uniSHIFT

Unidades Curriculares Pedidos 2 Conflitos 4 Alocação ▾

jose@di.uminho.pt

SAIR »

Alocação Manual em Turno

CONFIGURAÇÃO

Indique um aluno e uma UC da sua matricula

Carlos ▾

Matemática ▾

Indique para que turnos mover o aluno

T

☐ MAT-T2: 3/4

☒ MAT-T1: 3/50

TP

☒ MAT-TP5: 2/50 (CONFLITO)

☐ MAT-TP4: 4/50

☐ MAT-TP1: 3/50

☐ MAT-TP3: 3/50

☐ MAT-TP2: 3/50

☐ MAT-TP6: 3/50

HORÁRIO DO ALUNO

	Mon	Tue	Wed	Thu	Fri
8am					
9am		MAT/TP5 2/50 ocupado E1-1.01	MAT/TP4 4/50 ocupado E1-1.01		MAT/TP3 3/50 ocupado E1-1.01
10am		AL/T1 3/50 ocupado E1-1.01		LC/T1 3/50 ocupado E1-1.01	
11am	MAT/T2 3/4 ocupado E1-1.02		MAT/TP1 3/50 ocupado E1-1.01		MAT/T1 4/50 ocupado E1-1.01
12pm					MAT/TP2 3/50 ocupado E1-1.01
1pm					
2pm	LC/PL1 3/50 ocupado E1-1.01		AL/PL1 3/50 ocupado E1-1.01		LC/PL1 3/50 ocupado E1-1.01
3pm					
4pm					MAT/TP6 3/50 ocupado E1-1.01
5pm					
6pm					
7pm					

Aplicar

© Interface Pessoa-Máquina 2025

Figura 8: Alocação Manual de Alunos

A alocação automática, como o nome indica, é a página que permite ao diretor de curso realizar a alocação automática, usando o algoritmo do backend.

un1SHIFT

Unidades Curriculares Pedidos 2 Conflitos 4 Alocação

JoseRei.uninho.pt

Sair

Alocação Automática em Turno

CONFIGURAÇÃO

0 Max TP/PL é limitado pela sala mais pequena em uso.

Unidade Curricular	MAX TP/PL	Prioridade de Alunos
Matemática	15 Lugares	id (predefinido)
Fundamentos de Programação	15 Lugares	id (predefinido)
Introdução à Computação	15 Lugares	id (predefinido)
Álgebra Linear	15 Lugares	id (predefinido)
Lógica para Computação	15 Lugares	id (predefinido)
Cálculo	15 Lugares	id (predefinido)
Programação Orientada a Objetos	15 Lugares	id (predefinido)
Estruturas de Dados	15 Lugares	id (predefinido)
Arquitetura de Computadores	15 Lugares	id (predefinido)
Sistemas Operativos	15 Lugares	id (predefinido)
Bases de Dados	15 Lugares	id (predefinido)
Redes de Computadores	Lugares	id (predefinido)
Engenharia de Software	Lugares	id (predefinido)
Inteligência Artificial	Lugares	id (predefinido)
Sistemas Distribuídos	Lugares	id (predefinido)
Compiladores	Lugares	id (predefinido)
Segurança Informática	Lugares	id (predefinido)
Interação Humano-Computador	Lugares	id (predefinido)
Grafos e Algoritmos	Lugares	id (predefinido)
Processamento de Linguagens	Lugares	id (predefinido)
Aprendizagem de Máquina	Lugares	id (predefinido)
Sistemas de Informação	Lugares	id (predefinido)
Computação Gráfica	Lugares	id (predefinido)
Redes de Nova Geração	Lugares	id (predefinido)
Sistemas Embarcados	Lugares	id (predefinido)
Projeto de Engenharia Informática	Lugares	id (predefinido)
Tópicos Avançados em Computação	Lugares	id (predefinido)
Gestão de Projetos de Software	Lugares	id (predefinido)
Computação Móvel	Lugares	id (predefinido)
Sistemas de Tempo Real	Lugares	id (predefinido)

Adicionar

© Interface Pessoa-Máquina 2025

Figura 9: Alocação Automática de Alunos

A qualquer altura é possível fazer logout e ir para a página de login ao clicar em “Sair” na *Navbar*.

6.3. Aluno

O *login* é realizado através do email institucional e a password do aluno.

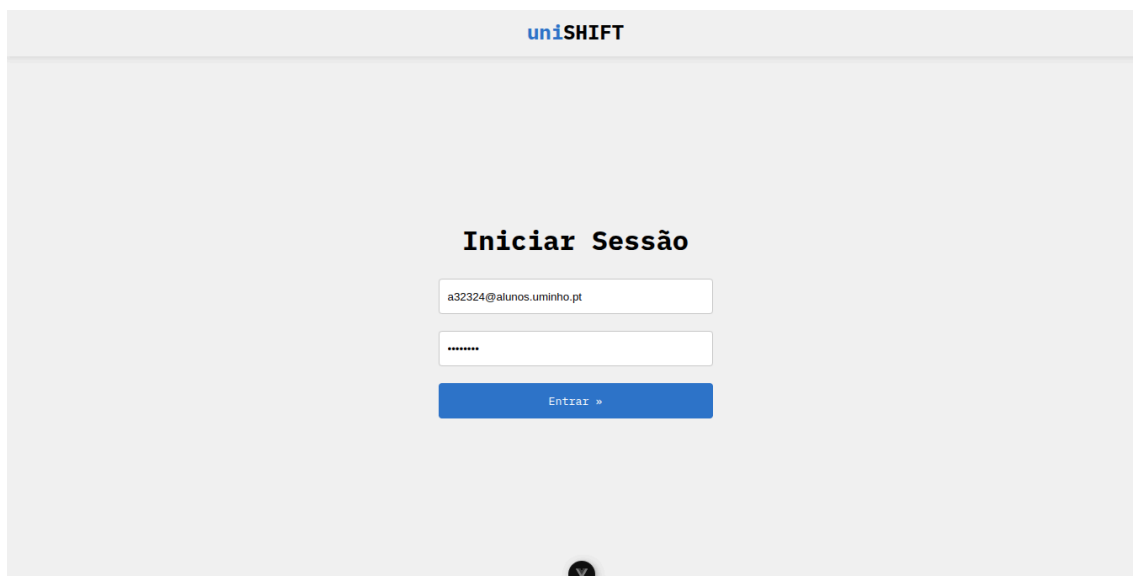
The image shows a web browser window with the uniSHIFT logo in the top header. The main content area is titled "Iniciar Sessão" (Log In). Below the title, there are two input fields: the first contains the email address "a32324@alunos.uminho.pt" and the second contains a masked password "*****". Below these fields is a blue button labeled "Entrar »" (Log In). At the bottom center of the page, there is a small, dark, circular icon.

Figura 10: Login como Aluno

Após o *login*, o aluno é encaminhado para a sua página pessoal, onde pode ver os seus detalhes, o seu horário e as UC's em que está matriculado. Nesta página é possível navegar para outras páginas através da *Navbar* ou aceder à página de novo pedido com a informação da respetiva UC já preenchida.

unishift

Horário

Pedidos

2

a32324@alunos.uminho.pt

SAIR »

Aluno: Mafalda

DETALHES

NOME: Mafalda

EMAIL: a32324@alunos.uminho.pt

CARGO: Aluno

ESTATUTO: Especial

HORÁRIO

	Mon	Tue	Wed	Thu	Fri
8am					
9am			MAT-TP4 4/50 al. E1-1.01		
10am			FP-T1 3/50 al. E1-1.01		IC-T1 3/50 alunos E1-1.01
11am	MAT-T2 3/50 alunos E1-1.01				
12pm					
1pm					
2pm				FP-PL1 3/50 alunos E1-1.01	
3pm					
4pm					
5pm					
6pm					
7pm					

MATRÍCULA ATUAL

Matemática

Época: 1º ano / 1º semestre

Matriculas Prévias: 1

Turnos: T2, TP4

Pedido de Turno

Fundamentos de Programação

Época: 1º ano / 1º semestre

Matriculas Prévias: 1

Turnos: T1, PL1

Pedido de Turno

Introdução à Computação

Época: 1º ano / 1º semestre

Matriculas Prévias: 1

Turnos: T1

Pedido de Turno

Álgebra Linear

Época: 1º ano / 1º semestre

Matriculas Prévias: 1

Turnos:

Pedido de Turno

Lógica para Computação

Época: 1º ano / 1º semestre

Matriculas Prévias: 1

Turnos:

Pedido de Turno

© Interface Pessoa-Máquina 2025

Figura 11: Horário

Ao clicar em “Pedidos” na *Navbar*, entramos na página onde estão os pedidos (pendentes e fechados). Aqui é possível aceder às páginas de Novo Pedido e de Detalhes de um Pedido.

uniSHIFT

Horário

Pedidos 2

a32324@alunos.uminho.pt

SAIR »

Pedidos

NOVOS PEDIDOS

NOVO PEDIDO »

AUTOR	DATA/HORA	UC	
Mafalda	2025/05/01 02:22	Matemática	Detalhes
Mafalda	2025/02/10 13:35	Fundamentos de Programação	Detalhes

PEDIDOS FECHADOS

AUTOR	DATA/HORA	UC	
Mafalda	2025/05/04 19:14	Matemática	Detalhes
Mafalda	2025/05/01 02:22	Matemática	Detalhes
Mafalda	2025/02/10 13:35	Matemática	Detalhes

© Interface Pessoa-Máquina 2025

Figura 12: Pedidos

Ao clicar em “**Novo Pedido**” na página **Pedidos**, entramos na página onde é possível selecionar uma UC e um turno para realizar o pedido ao diretor de Curso.

uniSHIFT

Horário Pedidos 2

a32324@alunos.uminho.pt

SAIR »

Novo pedido

SELECIONAR UC

Indique a Unidade Curricular a qual pretende se inscrever a um novo turno

Escolha a UC:

-- Selecione uma UC --

SELECIONAR TURNO

Indique apenas um Turno ao qual pretende se inscrever

	Mon	Tue	Wed	Thu	Fri
8am					
9am			MAT-TP4 4/50 al. E1-1.01		
10am			FP-T1 3/50 al. E1-1.01		IC-T1 3/50 alunos E1-1.01
11am	MAT-T2 3/50 alunos E1-1.01				
12pm					
1pm					
2pm				FP-PL1 3/50 alunos E1-1.01	
3pm					
4pm					
5pm					
6pm					
7pm					

OBSERVAÇÕES (OPCIONAL)

Escreva observações que ache pertinente para o Diretor

Escreva as observações...

Cancelar

Enviar »

© Interface Pessoa-Máquina 2025

Figura 13: Pedidos

Ao clicar em “**Detalhes**” na página **Pedidos**, entramos na página onde é possível ver todas as informações sobre o respetivo pedido. Aqui é possível aceder à página de perfil do autor do pedido.

unISHIFT

Horário Pedidos 2

a32324@alunos.uminho.pt

SAIR »

Pedido Troca de Turno (MAT)

DETALHES

Pedido pendente

AUTOR: Filipe [Ver perfil](#)

EMAIL: f32325@alunos.uminho.pt (Aluno, Estatuto Normal)

ABERTURA: 2025/05/01 02:22

TURNO ATUAL: MAT-TP4

TURNO PEDIDO: MAT-TP6

Sem mensagem do aluno.

HORÁRIO DO ALUNO

	Mon	Tue	Wed	Thu	Fri
8am					
9am			MAT-TP4 4/50 al. E1-1.01		
10am			FP-T1 3/50 al. E1-1.01		IC-T1 3/50 alunos E1-1.01
11am	MAT-T2 3/50 alunos E1-1.01				
12pm					
1pm					
2pm				FP-PL1 3/50 alunos E1-1.01	
3pm					
4pm					MAT-TP6 3/50 alunos E1-1.01
5pm					
6pm					
7pm					

OBSERVAÇÕES (OPCIONAL)

Escreva observações que ache pertinente para o Aluno

Escreva as observações...

© Interface Pessoa-Máquina 2025

Figura 14: Detalhes de Pedido

A qualquer altura é possível fazer logout e ir para a página de login ao clicar em “**Sair**” na *Navbar*.

7. Reflexão sobre o Trabalho

7.1. Dificuldades sentidas

Durante o desenvolvimento desta aplicação, as principais dificuldades surgiram na integração de componentes externos e na gestão da reatividade das opções do calendário. Lidar com a atualização dinâmica de eventos no FullCalendar revelou-se mais complexo do que o esperado, exigindo diferentes abordagens (reatribuição de props, uso da API interna e, por fim, simplificação do binding). Adicionalmente, a necessidade de garantir acessibilidade num componente que não seguia as boas práticas native do framework, como o FullCalendar, obrigou a recorrer a uma solução provisória de remoção de atributos role via JavaScript, evidenciando a importância de avaliações automáticas e a limitação de algumas bibliotecas.

A passagem de propriedades a componentes e a emissão de eventos revelou-se igualmente desafiante, dado que a Options API exige uma sintaxe excessivamente verbosa e fragmentada. Pessoalmente, consideramos a Composition API mais simples e produtiva para estruturar a lógica, a reatividade e a comunicação entre componentes, apesar de utilizar maioritariamente a Options API.

7.2. Aspetos positivos do Vue

Por outro lado, o que mais nos agradou em Vue.js foi a sua simplicidade e produtividade. A introdução do Pinia mostrou-se particularmente produtiva para gerir estado global de forma tipada e modular, enquanto o Vue Router facilitou a construção de fluxos com guardas de autenticação. A reatividade declarativa acelerou o desenvolvimento de interfaces interativas, e o formato de Single File Components permitiu organizar HTML, CSS e JavaScript de modo coeso.

8. Conclusão

Em suma, este trabalho reforçou a nossa confiança no Vue.js como framework produtiva para criar aplicações complexas. Como trabalhos futuros, planeia-se:

- Realizar testes de usabilidade com perfis reais, obtendo consentimento formal e documentando as sessões.
- Implementar o cenário de pedidos por docentes e refinar a lógica de exceções para alunos com estatuto.
- Melhorar a gestão de dados de teste, criando fixtures mais fiáveis ou integrando um backend temporal.

Esta reflexão orienta futuras iterações, assegurando que a aplicação evolua tanto em robustez técnica quanto em experiência de utilizador.