

Projeto de PTI/PTR

Grupo 04

Tiago Correia	45680
Núria Santos	48305
Mykhaylo Levytskyy	49994
Rui Pereira	51623
Tiago Sousa	51648
Daniel Gomes	51649
Sandro Lopes	51810

Objetivo

No âmbito das cadeiras de PGP, PTI e PTR, iremos criar um sistema de gestão de trabalhos de grupo em ambiente universitário. Este será principalmente usado nas instituições da Universidade de Lisboa e terá funções semelhantes ao Moodle.

Requisitos

Requisitos funcionais

Principais

Administradores do Sistema:

RF-1: Possibilidade de importação de dados em ficheiros .txt Os admins irão assegurar a existência de dados de:

RF-2: Alunos

RF-3: Unidades curriculares

RF-4: Turmas

RF-5: Professores

RF-6: Grupos

RF-7: Mensagens

RF-8: Posts de fóruns

RF-9: Avaliações

RF-10: Pontuações de alunos

RF-11: Etapas de projetos

RF-12: Regras de constituição de grupos

Requisitos

Requisitos funcionais

Principais

Professores:

RF-13: Definição de regras para a constituição de grupos para os projetos das suas unidades curriculares

RF-14: Estipular as etapas para os projetos, com objetivos, entregáveis, e prazos

RF-15: Consultar os documentos e outros artefactos que vão sendo produzidos por cada grupo em cada projeto

RF-16: Responder a dúvidas dos grupos e dar feedback sobre o que cada grupo entrega em cada etapa

Requisitos

Requisitos funcionais

Principais

Alunos:

RF-17: Criar grupo

RF-18: Entrar num grupo

RF-19: Comunicar entre si ou com o professor através de mensagens (chat)

RF-20: Guardar documentos e outros artefactos na área de trabalho do grupo a que pertence

RF-21: Sugerir períodos para todos os membros se reunirem ou aceitar um período já sugerido

RF-22: Registar o tempo que dedicam a cada tarefa do projeto

RF-23: Atribuir pontuações às prestações dos colegas no final do trabalho de grupo

Requisitos

Requisitos funcionais

Complementares

Professores:

RF-24: Publicar as informações da cadeira na página da disciplina e outras informações adicionais (usando o fórum da disciplina)

RF-25: Definir o sistema de avaliações da cadeira (testes, presenças, exames, projetos)

RF-26: Dar notas aos projetos dos alunos.

Requisitos

Requisitos funcionais

Complementares

Alunos:

RF-27: Acesso à página da disciplina (fóruns, etapas do projeto, contacto com o professor).

RF-28: Ver notas das cadeiras

RF-29: Ver os perfis de outros alunos (ex: ver as pontuações e comentários de projetos passados)

RF-30: Opção de gerar um grupo aleatório

RF-31: Agendar uma hora para submissão automática do trabalho

Requisitos

Requisitos funcionais

Complementares

Alunos:

RF-32: Ver a lista de tarefas e o seu estado

RF-33: Criar ou eliminar uma tarefa

RF-34: Auto-nomear-se como responsável por uma tarefa específica

RF-35: Registrar uma tarefa como concluída

RF-36: Comunicar com os vários membros do grupo através de um chat de grupo

RF-37: Marcar um horário pessoal com o professor.

Requisitos

Requisitos não funcionais

Escalabilidade

RNF-1: O Sistema tem que funcionar da mesma forma independentemente do número de utilizadores que usa simultaneamente.

Manutenção

RNF-2: O sistema tem que ser modular de forma a facilitar a adição e remoção de hardware.

Desempenho

RNF-3: O sistema tem que ter respostas rápidas.

RNF-4: Diferentes plataformas/aplicações não devem afetar o desempenho do sistema.

Requisitos

Requisitos não funcionais

Fiabilidade

RNF-5: O sistema tem que armazenar os dados de uma forma fiável.

RNF-6: No caso de falha no hardware, o sistema deve afetar o mínimo possível de desempenho.

RNF-7: Tolerância a falhas catastróficas com duração não superior a um dia, sendo apenas admissível a perda de dados registados nessas últimas 24 horas.

Segurança

RNF-8: O sistema tem que garantir a confidencialidade dos dados.

RNF-9: O sistema não pode permitir o acesso a utilizadores não autorizados.

RNF-10: O sistema tem que encriptar os dados privados de todos os utilizadores.

RNF-11: O sistema tem que ter um mecanismo de defesa para evitar a injeção de scripts.

Tecnologias

Software para o sistema

- ASW

Ferramentas de desenvolvimento

Linguagens de Programação:

- Python
- Javascript
- CSS
- HTML

Frameworks/bibliotecas:

- Django
- W3CSS
- jQuery

VCS:

- Git

Tecnologias

Ferramentas de desenvolvimento

Sistema de virtualização:

- Docker

Edição de código:

- Visual Studio Code

Gestão:

- Microsoft Project

Papéis

	T.Correia	Núria	Mykhaylo	Rui	T.Sousa	Daniel	Sandro
Programador back end		✓	✓				✓
Tratamento de dados		✓	✓				✓
Gestão da BD		✓	✓				✓
Arquitetura do sistema		✓	✓				✓
Programador front end			✓	✓			✓
Arquitetura do site			✓	✓			✓
Programação Javascript			✓	✓			✓
Designer	✓			✓			
Desenho da aplicação	✓			✓			
Criação de PBF e PAF	✓			✓			
Estilo do site	✓			✓			
Analista	✓				✓		
Análise de dados de inquéritos/entrevistas	✓				✓		
Entrevistador	✓				✓		
Criação e inquéritos/entrevistas	✓				✓		
Testes com utilizadores	✓				✓		
Técnico de redes		✓				✓	
Escalabilidade		✓				✓	
Segurança		✓				✓	
Backup		✓				✓	
Gestor					✓		
Planeamento e gestão do projeto					✓		
Comunicação com o grupo					✓		
Comunicação com professores					✓		

Matriz RACI

	T.Correia	Núria	Mykhaylo	Rui	T.Sousa	Daniel	Sandro
Tratamento de dados		R	R		A	CI	R
Gestão da BD		R	R		A	CI	R
Arquitetura do sistema		R	R	CI	A	CI	R
Arquitetura do site	CI		R	R	A		R
Programação Javascript	CI		R	R	A		R
Desenho da aplicação	R		CI	R	A		CI
Criação de PBF e PAF	R		CI	R	A		CI
Estilo do site	R		CI	R	A		CI
Análise de dados de inquéritos/entrevistas	R	I	I	I	RA	I	I
Criação e inquéritos/entrevistas	R	C	C	C	RA	C	C
Testes com utilizadores	R	C	C	C	RA	C	C
Escalabilidade		R	CI		A	R	CI
Segurança		R	CI		A	R	CI
Backup		R	CI		A	R	CI
Planeamento e gestão do projeto	CI	CI	CI	CI	RA	CI	CI
Comunicação com o grupo	I	I	I	I	RA	I	I
Comunicação com professores	I	I	I	I	RA	I	I

Plano semanal 1

Iteração: 1, Semana 1

Objetivo principal: Iniciar a primeira iteração do projeto. Focar na **modelação** do sistema e início de algumas **implementações técnicas**. No fim da semana espera-se ter um conceito geral do **desenho da aplicação**, ter configurado a **base de dados** ter iniciado a implementação de pelo menos 3 casos de uso. Estes casos de uso serão:

Admin: inserir dados de um aluno em específico na BD (por input de texto)

Aluno: visualizar notas de cadeiras passadas

Professor: publicar uma mensagem no fórum

	Tarefas	Artefactos a produzir
Programadores backend	Configurar a ligação à BD. Explorar as tecnologias backend. Inserir na BD dados suficientes para implementar os 3 casos de uso.	Estrutura da BD (PNG)
Programadores frontend	Com a assistência dos designers, iniciar a estrutura do site. Começar a implementação dos 3 casos de uso.	Início da aplicação (HTML, CSS, JS)
Designers	Criar PAF's e PBF's.	PBF (fotos); PAF (HTML,CSS)
Analistas	Analisar os dados dos testes com utilizadores.	Lista de conclusões e feedback dos testes de utilizadores.
Entrevistadores	Criar inquérito para utilizadores do PAF, fazer testes com utilizadores.	Inquérito de utilizador
Técnicos de redes	Explorar a tecnologia Django, investigar métodos de escalabilidade e segurança. Assistir os programadores Backend.	
Gestor	Fazer um planeamento aprofundado das várias fases futuras do projeto. Dividir o projeto em etapas separadas por iteração e por semana. Criar uma lista de casos de uso que exploram todos os requisitos funcionais.	Lista de casos de uso (DOCX)

Progresso semanal 1

Iteração: 1, Semana 1

Objetivos cumpridos:

- Explorar tecnologias backend
- Início da estrutura da BD
- Início da estrutura do site
- Criar PBF's e PAF's
- Testes de PBF's e PAF's produzidos com utilizadores
- Criação da lista de casos de uso principais

Progresso semanal 1

Iteração: 1, Semana 1

Objetivos falhados:

- Configurar a ligação à BD
- Inserir na BD dados suficientes para implementar os 3 primeiros casos de uso
- Implementação frontend dos 3 casos de uso
- Analisar os dados dos testes com utilizadores
- Concluir a estrutura da BD
- Melhorar os work packages

Progresso semanal 1

	Tarefas feitas	Problemas encontrados
Tiago Correia	-Criou um PBF -Estudou Django	-Falta de tempo -Grande falta de conhecimento de Django (tecnologia backend)
Núria	-Estudou Django	-Grande falta de conhecimento de Django (tecnologia backend)
Mykhaylo	-Estudou Django	-Grande falta de conhecimento de Django (tecnologia backend)
Rui	-Criou um PBF e um PAF -Estudou Django	-Grande falta de conhecimento de Django (tecnologia backend)
Tiago Sousa	-Criou o início da estrutura da BD -Criou um PBF e um PAF -Fez testes com utilizadores para os PBF's e os PAF's -Criou a lista dos casos de uso principais -Estudou Django	-Não teve tempo para analisar os resultados dos PBF's e dos PAF's porque só recebeu os PBF e PAF do Rui no fim da semana. Apenas teve tempo para fazer os testes com os utilizadores. Quanto ao planeamento dos work packages, não conseguiu abrir o ficheiro mpp, também não soube como conciliar a lógica das dependências entre os casos de uso, e o ficheiro dos work packages original.
Daniel	-Estudou Django	-Grande falta de conhecimento de Django (tecnologia backend)
Sandro	-Modelo da estrutura da BD -Estudou Django	-Grande falta de conhecimento de Django (tecnologia backend)

Casos de uso

Admin:

- Importar dados de alunos
- Importar dados de cadeiras
- Importar dados de turmas
- Importar dados de professores
- Importar dados de avaliações
- Importar dados de grupos

Casos de uso

Aluno:

- Criar grupo
- Aderir a grupo
- Comunicar com professor ou aluno
- Enviar documento/submeter projeto
- Enviar mensagem no grupo
- Guardar ficheiro no grupo
- Marcar período de encontro com o grupo
- Criar lista de tarefas para o projeto
- Atribuir tarefas
- Registrar tempo de tarefa
- Atribuir pontuação a elemento do grupo
- Concluir tarefa

Casos de uso

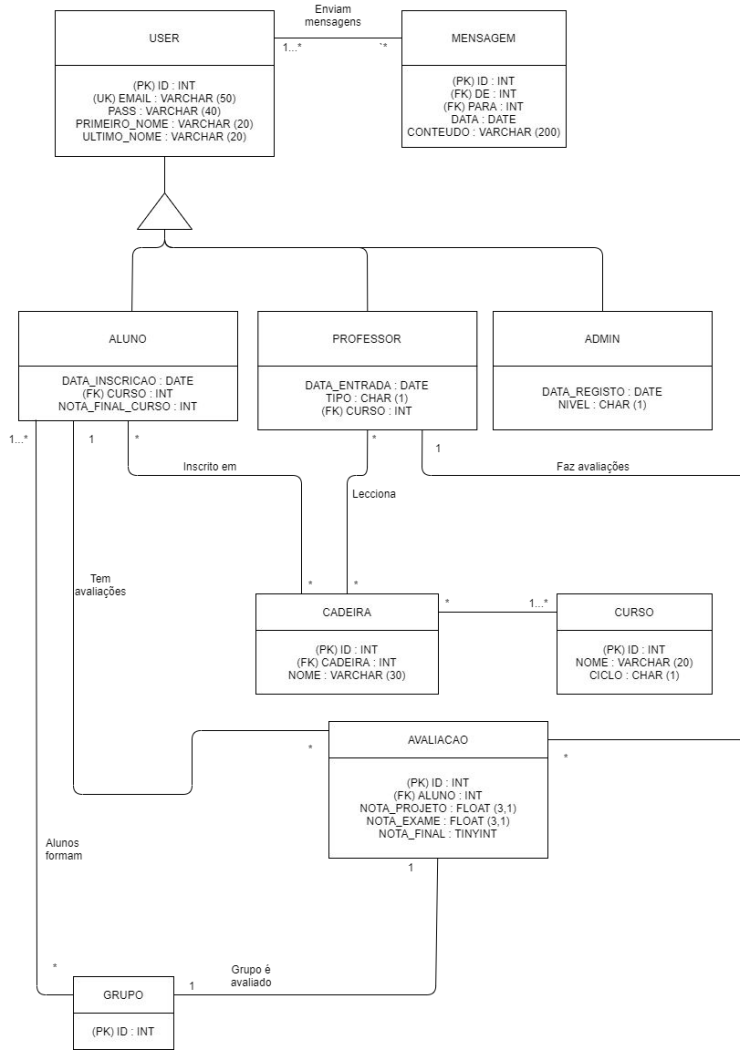
Professor:

- Definir regras de constituição de grupos
- Definir etapas para projetos
- Abrir documento de grupo de alunos
- Dar feedback a grupo

Casos de uso (exemplo)

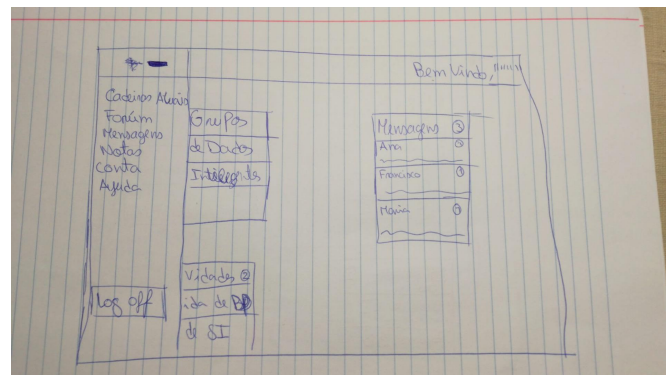
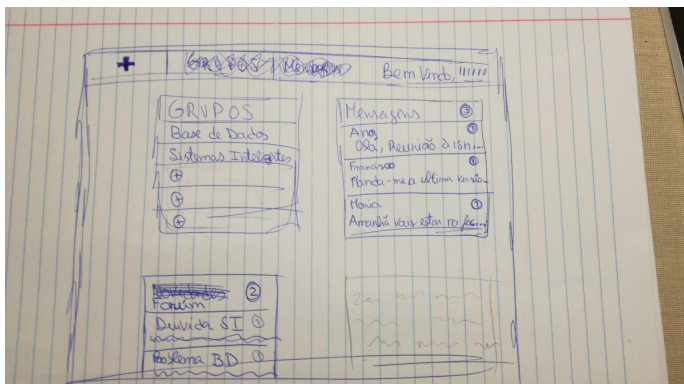
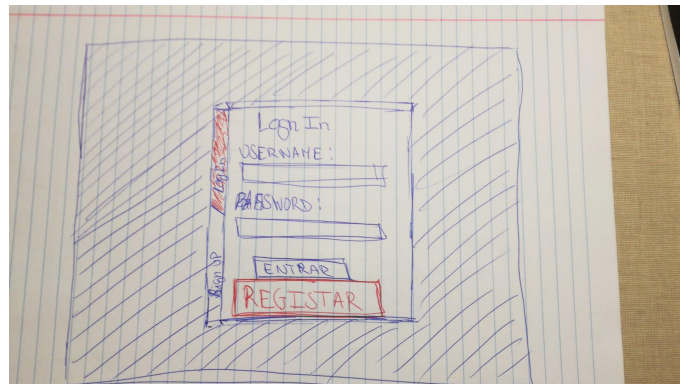
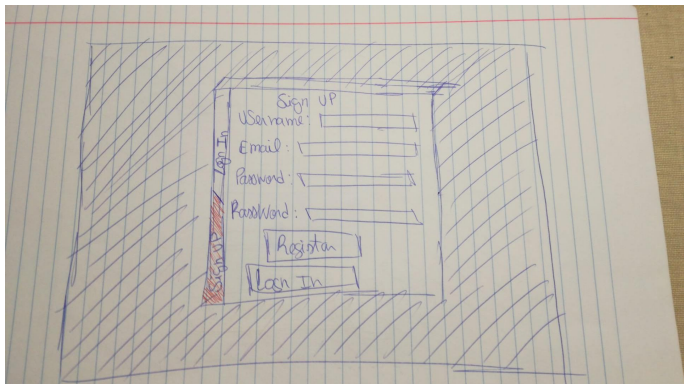
Nome do caso de uso	Dar feedback a grupo
Ator	Professor
Descrição	O professor dá feedback a um grupo sobre o que entregou numa determinada etapa do projeto.
Trigger	O professor acede a área da sua cadeira.
Pré-condição	O professor está registado e fez login no sistema. Já existem etapas definidas. Os grupos já estão formados. Pelo menos 1 grupo já fez upload do seu projeto.
Caminho principal	1- Clica no botão "feedback". 2- Aparece uma lista com todos os grupos. Clica no grupo a quem quer dar feedback. 3- Aparece um formulário simples (título, nº da etapa, descrição) que o professor preenche para dar o feedback. Clica em "submit". 4- Aparece uma mensagem de sucesso. 5- O professor, ao aceder a esta área, pode ver o feedback que deu a este grupo, e pode modificá-lo.
Pós-condição	Os alunos, ao acederem a área de feedback do projeto, vêem o que o professor escreveu.

Modelo da BD



PBF

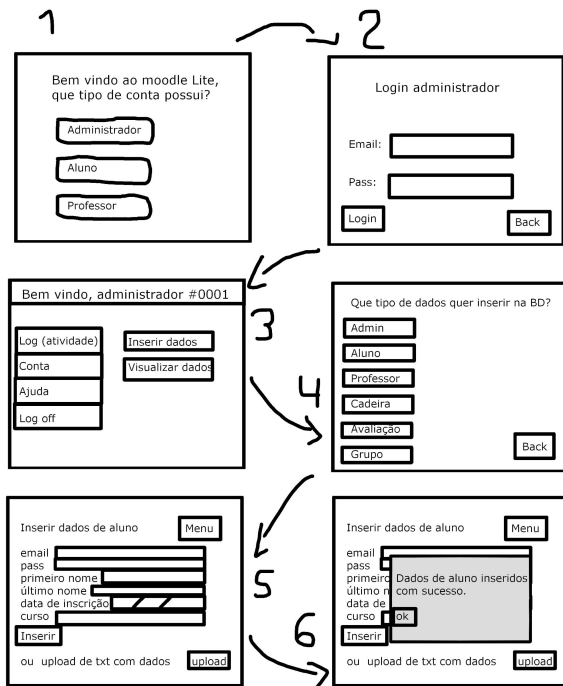
Rui



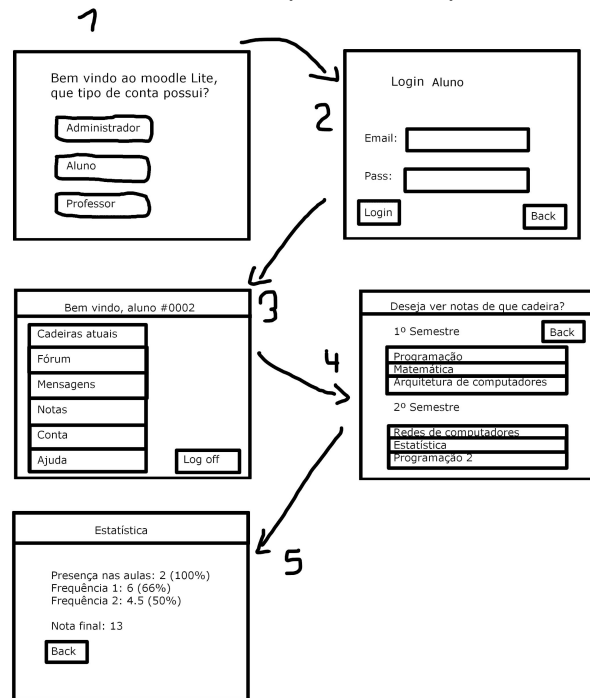
PBF

Tiago

Storyboard: Admin (inserir dados)



Aluno (ver notas)



PAF

Tiago

[Voltar atrás](#)

Login de aluno

Nome

Pass

Login

[Log off](#)

Bem vindo, estudante

[Mensagens](#)

[Cadeiras atuais](#)

[Notas](#)

[Grupos](#)

[Menu](#)

Grupo - 004 - IC
Feedback da etapa 1

[Back](#)

Nota do professor:

O vosso código não funciona em Linux. É suposto o programa correr em todos os sistemas operativos.

Tirando isso, parece estar a funcionar bem em Windows. Apenas recomendo que limpem melhor o vosso código e adicionem mais comentários.

[Menu](#)

Tarefas

[Back](#)

Tarefas para o projeto:

Tarefa 1: Fazer lista de requisitos funcionais (Núria)

Tarefa 2: Fazer PBF (Tiago)

Tarefa 3: Fazer o modelo da base de dados

[Editar](#)

[Remover](#)

[+ Nova tarefa](#)

PAF - Testes

Casos de uso:

- 1- Aluno envia mensagem para o utilizador Tiago Sousa.
- 2- Aluno envia mensagem para o professor Mário Calha.
- 3- Aluno cria uma nova tarefa no grupo de Interação com Computadores. A nova tarefa será "Fazer o PAF"
- 4- Aluno vê feedback da etapa 1 do projeto de IC.
- 5- Aluno avalia o colega do projeto de IC, Tiago Sousa, dando 0 estrelas.
- 6- Professor muda a nota do segundo teste de arquiteturas de computadores, da aluna Andreia Batista, para 10.0.
- 7- Professor dá feedback da segunda etapa ao grupo 003 de Interação com Computadores.

PAF - Testes

Métrica	Valores
Eficácia	100% para todos os utilizadores
Eficiência	55, 57, 59, 60, 60, 65, 65, 69
Tempo	00:50, 01:05, 01:10, 01:20, 01:22, 01:25, 01:30, 01:55
Erros	0, 0, 0, 0, 1, 1, 1- Em "Ver feedback" uma pessoa não sabia onde ir, no menu principal, tentou ir a 'cadeiras atuais' o que seria possível aceder ao feedback, mas não está implementado no PAF. Em "Dar feedback" uma pessoa clicou em "Etapas" na área dos grupos. Em "Avaliar colega" uma pessoa não sabia onde era para ir, tentou sair da área de grupo. Achou que o botão "Pontuação" não era muito explícito.
Satisfação geral	3,3,4,4,4,5,5,5
Comentários	<p>- "Gosto da estrutura do site mas de aspeto está fraco."</p> <p>- "Tive dúvidas na parte de 'avaliar colega'."</p> <p>- "Foi fácil de usar."</p> <p>- "Dava jeito entrar nas mesmas páginas a partir de partes diferentes, por exemplo entrar na área de grupos a partir da página da disciplina ou assim."</p> <p>- "Acho que no geral está bom o site."</p> <p>- "Achei que todas as funções estavam bastante boas, menos a do professor dar feedback ao grupo, isso não achei muito óbvio de como fazer."</p> <p>Nota do entrevistador: as pessoas clicaram primeiro em "Menu" em vez de "back", isso adiciona clicks desnecessários à contagem e obriga a navegar a mais ecrãs.</p>

Plano semanal 2

Objetivo principal: Continuar a primeira iteração do projeto. Focar na implementação dos casos de uso.

Concluir os objetivos falhados da semana passada:

No fim da semana espera-se ter um conceito geral do **desenho da aplicação**, ter configurado a **base de dados** ter iniciado a implementação de pelo menos **3 casos de uso**. Estes casos de uso serão:

- Admin: inserir dados de um aluno em específico na BD (por input de texto)
- Aluno: visualizar notas de cadeiras passadas
- Professor: publicar uma mensagem no fórum