



Upskill - Java

Laboratório de Programação

T4J – TASKS FOR JOE

A *startup Tasks for Joe* (T4J) dedica-se a facilitar e promover o contacto entre pessoas que trabalham por conta própria (*freelancers*) e organizações que pretendem contratar alguém externo (*outsourcing*) para a realização de determinadas tarefas. Para promover e suportar o seu negócio, a *T4J* pretende desenvolver uma plataforma informática que, por um lado, permita que qualquer organização interessada possa registar-se na plataforma de forma a poder publicar tarefas e gerirem elas próprias o processo de adjudicação dessas tarefas a *freelancers*; e por outro lado, permita que os *freelancers* acedam facilmente a essas tarefas e possam candidatar-se à realização das mesmas.

Desde já, perspetiva-se que a plataforma seja acedida por vários utilizadores com diferentes papéis, tais como:

- **Administrativos:** são colaboradores da *T4J* afetos à gestão da plataforma e, em particular, por realizar na plataforma várias atividades de suporte ao negócio, tais como, definir áreas de atividade (e.g. IT, Marketing, Design), definir categorias de tarefas (e.g. desenvolvimento aplicações web, desenvolvimento de aplicações móveis) e especificar competências técnicas requeridas para a realização de tarefas.
- **Gestor de Organização:** pessoa indicada como gestor da organização aquando do registo da organização na plataforma. Assume-se que é um colaborador dessa organização, sendo responsável por especificar na plataforma outros colaboradores dessa mesma organização;
- **Colaborador de Organização:** pessoa registada na plataforma como sendo alguém que atua em representação de uma determinada organização. Entre outras responsabilidades, cabe-lhe especificar tarefas para posterior publicação pela organização respetiva;
- **Freelancers:** pessoas que se propõem a realizar as tarefas publicadas pelas organizações.

As interações dos utilizadores supra mencionados devem ser precedidas de um processo de autenticação. A utilização da plataforma por outras pessoas é restrita ao registo de organizações. Aquando deste registo é obrigatório requerer o nome da organização, o seu número de identificação fiscal (NIF), o endereço postal, um contacto telefónico, um endereço web, um endereço de correio eletrónico (email) e os dados do colaborador responsável pelo registo (nome, função, contacto telefónico, endereço de email). Após o registo da organização, os seus colaboradores podem aceder imediatamente à plataforma.

Tanto uma área de atividade como uma competência técnica caracterizam-se através de um código único, uma descrição breve e outra mais detalhada respetivamente. Uma competência técnica caracteriza-se ainda por ser referente a uma dada área de atividade. Por outro lado, cada categoria de tarefa caracteriza-se por um identificador interno (automático), uma descrição, a área de atividade em que se enquadra (apenas uma) e uma lista de competências técnicas tipicamente requeridas para a realização de tarefas dessa categoria. Algumas destas competências têm carácter obrigatório e outras apenas são desejáveis.

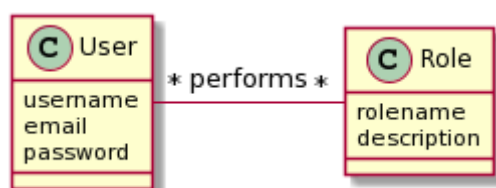
Cada tarefa caracteriza-se por ter uma referência única por organização, uma designação, uma descrição informal e outra de carácter técnico, uma estimativa de duração e custo bem como a categoria em que a mesma se enquadra. Enquanto a mesma não for publicada, o acesso à tarefa é exclusivo aos colaboradores da organização respetiva.

A designação comercial da plataforma e outros dados que venham a ser relevantes devem ser especificados por configuração aquando da sua implantação.

GESTÃO DE UTILIZADORES

O componente de gestão de utilizadores irá ser desenvolvido por outra equipa de desenvolvimento que também é responsável pela sua manutenção/evolução. Este componente visa satisfazer um conjunto de requisitos/funcionalidades genéricas e comuns a várias aplicações. A autenticação dos utilizadores será realizada com um identificador de utilizador e palavra-passe.

Os conceitos principais abordados neste componente são apresentados de seguida.



De acordo com o modelo do domínio, um “User” desempenha vários “Role” (i.e. funções), sendo que o mesmo “Role” também pode ser desempenhado por vários “User”.

Este componente disponibiliza as suas funcionalidades através de um *Web Service* que está em conformidade com o estilo arquitetural *REST* (*Representational State Transfer*). A utilização do serviço apenas é possível a quem conhecer uma chave que é obtida após registo. A T4J

Laboratório de Programação

2º SPRINT

já efetuou esse registo e já possui a respetiva chave, que fornecerá posteriormente à equipa de desenvolvimento.

Assim sendo, a documentação relevante com vista à sua utilização consiste na descrição dos métodos, *endpoints*, recursos, parâmetros e resultados de output esperados.

https://multilogus.dynns.com:8001			
Resource	Method	Parameters	Output example (Json format)
/context	GET	app_key : String	[{"app_context": "{7E19F342-A903-4C3B-806A-CF771120B9D0}"}]
/registerUser	POST	app_context : String username : String email : String password : String	
/registerUserWithRole	POST	app_context : String username : String email : String password : String role : String rolename	
/login	POST	app_context : String username : String password : String	
/logout	POST	app_context : String	
/roles	GET	app_context : String	[{"rolename": "administrativo", "description": "Administrativo"}, {"rolename": "colaborador", "description": "Colaborador de Organização"}, {"rolename": "freelancer", "description": "Freelancer"}, {"rolename": "gestor", "description": "Gestor de Organização"}]
/roles	POST	app_context : String name : String description : String	
/session	GET	app_context : String	[{"username": "John Doe", "email": "jdoe@company.com", "rolename": null, "logindate": "2021-01-23T18:33:27.000Z"}]

O Web Service disponibiliza os seguintes recursos:

- **"GET /context(String app_key)":** permite obter uma chave de contexto que identifica uma sessão de utilizador. Esta chave de contexto será necessária para executar todos os restantes métodos disponíveis.
- **"POST /registerUser(String app_context, String username, String email, String password)":** permite definir a existência de um novo utilizador que pode usar a aplicação;
- **"POST /registerUserWithRole(String app_context, String username, String email, String password, String role)":** semelhante ao método anterior, com a vantagem de associar imediatamente um papel/função a esse utilizador;

- **"POST login(String app_context, String user_id, String password)"**: permite autenticar um utilizador com vista à utilização da aplicação. O parâmetro *user_id* pode ser o *username* ou o *email* do utilizador;
- **"POST logout(String app_context)"**: termina a sessão de utilizador que estiver ativa no momento;
- **"POST /roles(String app_context, String name, String description)"**: permite definir a existência de um novo papel/função de interesse para a aplicação;
- **"GET /roles(String app_context)"**: permite obter todos os papéis/funções existentes no sistema;
- **"GET /session(String app_context)"**: permite obter os dados da sessão ativa.

O *Web Service* associa à resposta a todos os pedidos realizados um *HTTP response status code* em concordância com a publicação RFC 2616, de junho de 1999, que definiu o HTTP/1.1. Por exemplo, serão devolvidos os códigos HTTP 200 (OK) a cada pedido corretamente processado, 401 (*unauthorized*) a cada pedido sem autorização (i.e. contexto inválido, credenciais inválidas, etc.) e 500 (*internal error*) para todas as situações representativas de erros internos do serviço.

CASOS DE TESTE

Devem ser desenvolvidos cenários de teste para o software desenvolvido, nomeadamente testes de integração e testes unitários.

ANÁLISE E DESIGN

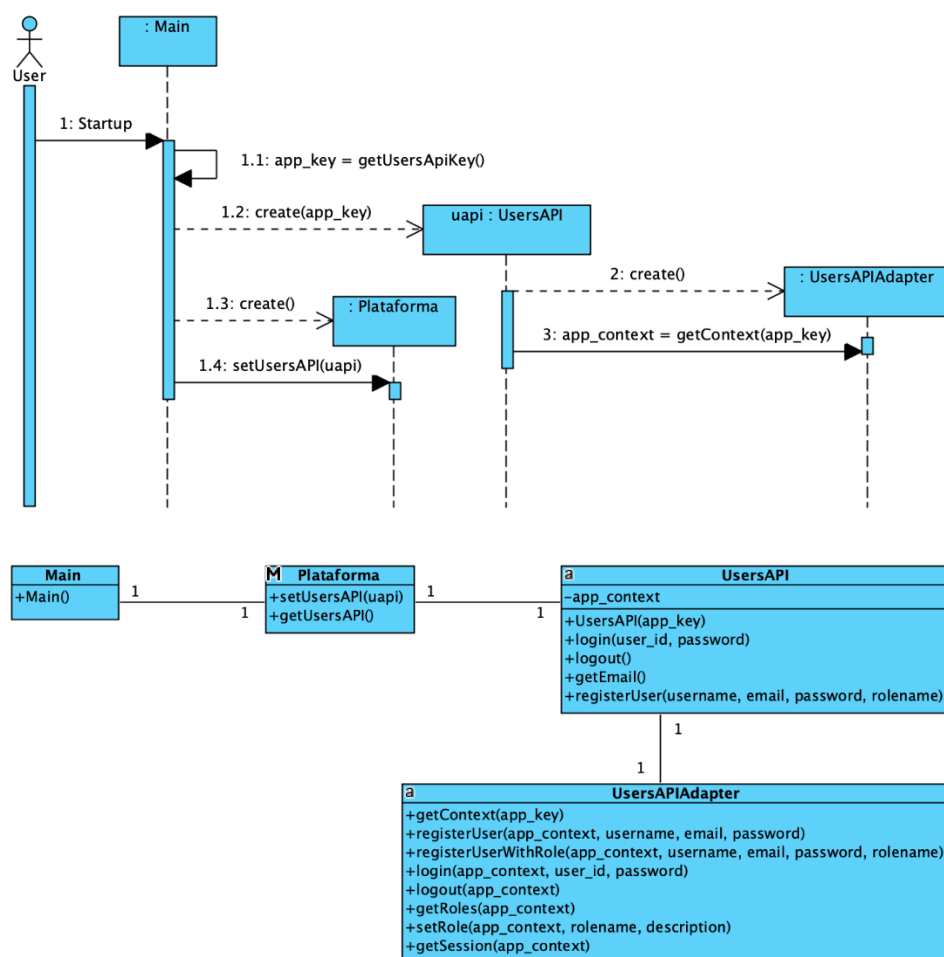
Foi realizado, parte do levantamento das especificações dos requisitos funcionais e não funcionais, bem como a análise OO e design no contexto de uma aplicação de consola, que se apresentam de seguida. Somente parte dos artefactos são apresentados, pretendendo-se a alteração destes (para repercutirem o uso de uma interface gráfica) e a produção dos restantes neste Sprint.

API PARA GESTÃO DE UTILIZADORES

Apresenta-se de seguida uma sugestão de design para a utilização do serviço de gestão de utilizadores.

Arranque da aplicação

No arranque da aplicação são criados os objetos necessários para o consumo do *webservice*.



Consumo do *webservice* (exemplo: autenticação)

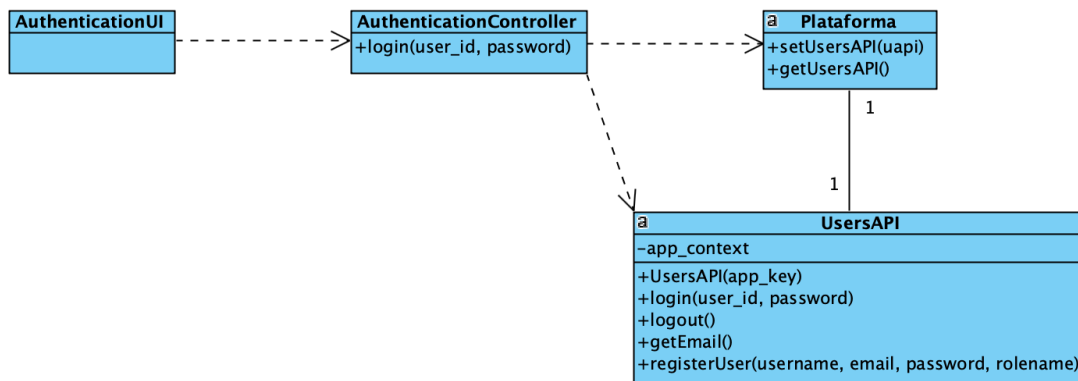
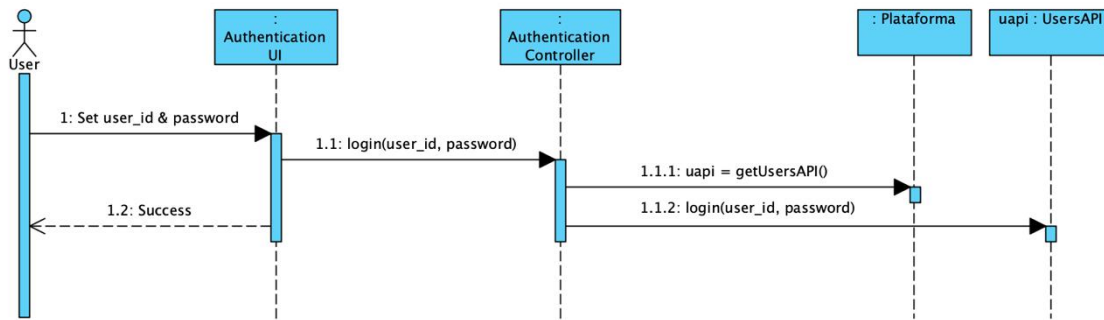
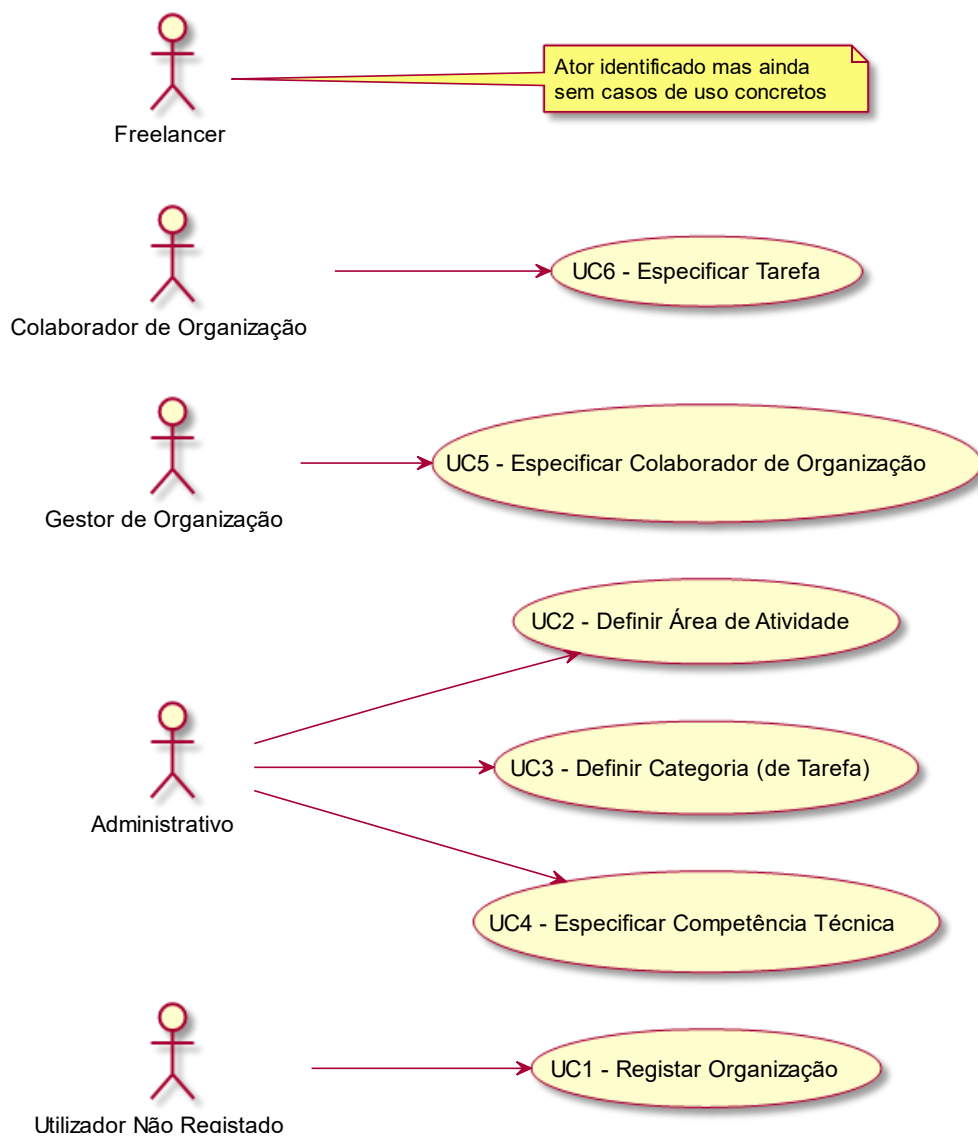


DIAGRAMA DE CASOS DE USO

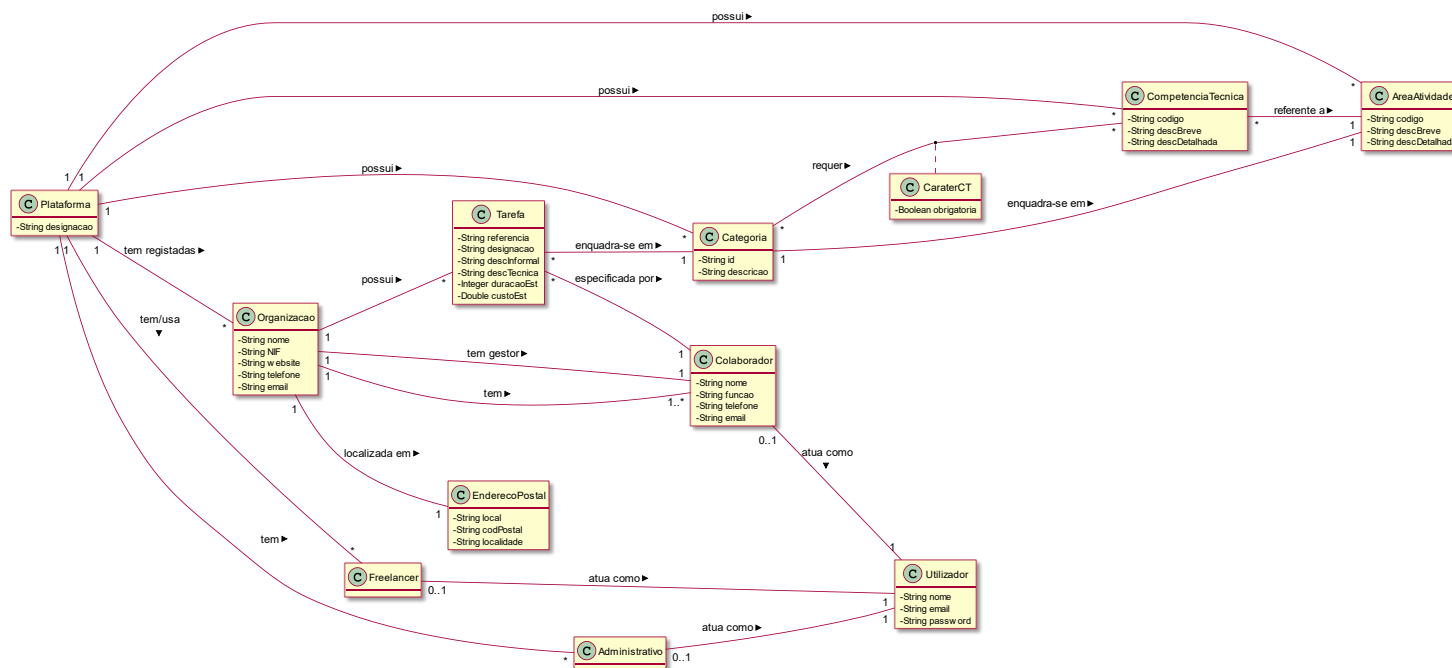


GLOSSÁRIO

<u>Termo</u>	<u>Descrição</u>
Administrativo	Pessoa responsável por realizar na plataforma várias atividades de suporte ao negócio.
ADM	Acrónimo para Administrativo.
Área de Atividade	Representa a área de negócio/mercado em que se atua como, por exemplo, IT (Information Technologies), Marketing e Design.
AT	Acrónimo para Área de Atividade.
Categoria (de Tarefa)	Corresponde a uma descrição usada para catalogar uma ou mais tarefas (semelhantes).
Colaborador de Organização	Pessoa que exerce funções (e.g. especificar tarefas) no âmbito de uma Organização.
Competência Técnica	Refere-se a um conjunto de conhecimentos e competências (habilidades) necessárias à execução de tarefas de uma determinada categoria.
CT	Acrónimo de Competência Técnica.
Freelancer	Pessoa individual que trabalha por conta própria e não está necessariamente comprometida com uma entidade empregadora (organização) específica a longo prazo.
Gestor de Organização	Corresponde ao papel desempenhado por um Colaborador dessa Organização na própria Organização.
Implantação (de Software)	Processo que visa colocar um determinado software em produção. Nos casos mais simples, pode corresponder simplesmente à atividade de

<u>Termo</u>	<u>Descrição</u>
	instalação desse software. Contudo, este processo pode implicar também outras atividades como ativação, configuração, etc.
NIF	Acrónimo para Número de Identificação Fiscal.
Organização	Pessoa coletiva que pretende contratar <i>freelancers</i> para a realização de tarefas necessárias à atividade da mesma.
Processo de Autenticação	Meio através do qual se procede à verificação da identidade da pessoa que pretende/está a utilizar a plataforma informática.
Utilizador	Pessoa que interage com a aplicação informática.
Utilizador Não Registado	Utilizador que interage com a plataforma informática de forma anónima, i.e. sem ter realizado previamente o processo de autenticação previsto.
Utilizador Registado	Utilizador que interage com a plataforma informática após ter realizado o processo de autenticação previsto e, portanto, a aplicação conhece a sua identidade. Tipicamente, este assume o papel/função de Administrativo ou Gestor de Organização ou Colaborador de Organização ou <i>Freelancer</i> .

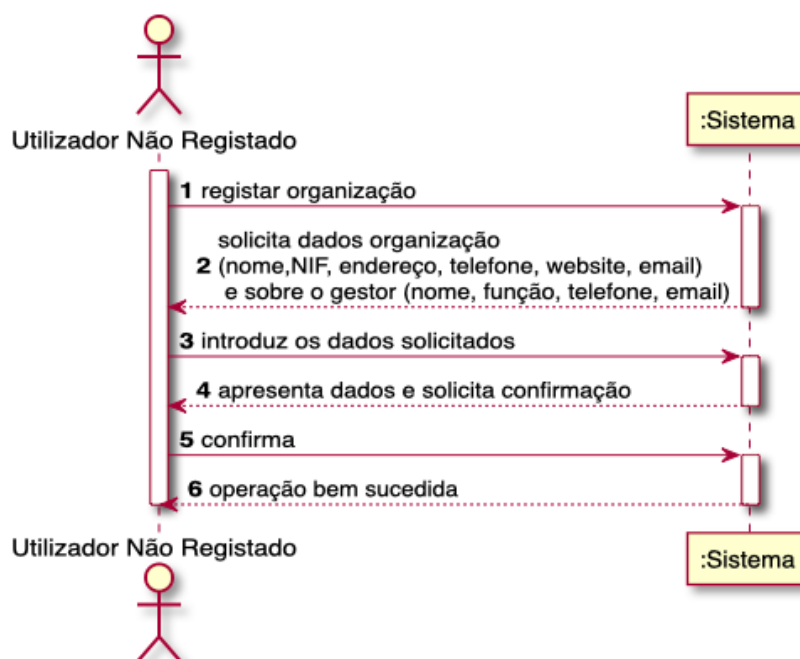
MODELO DO DOMÍNIO



UC1 – REGISTAR ORGANIZAÇÃO

O utilizador não registado inicia o registo de uma organização. O sistema solicita os dados necessários sobre a organização (i.e. nome da organização, o NIF, o endereço postal, um contacto telefónico, o endereço web, email) e sobre o colaborador (i.e. gestor) que está a proceder ao registo (i.e. nome, função, contacto telefónico e email). O utilizador não registado introduz os dados solicitados. O sistema valida e apresenta os dados, pedindo que os confirme. O utilizador não registado confirma. O sistema regista os dados da organização e do seu colaborador/gestor, tornando este último um utilizador registado, **remetendo-lhe a sua palavra-passe (gerada pelo sistema) por email** e informa o utilizador não registado do sucesso da operação.

Diagrama de sequência de sistema



Excerto do modelo do domínio

Laboratório de Programação

2º SPRINT

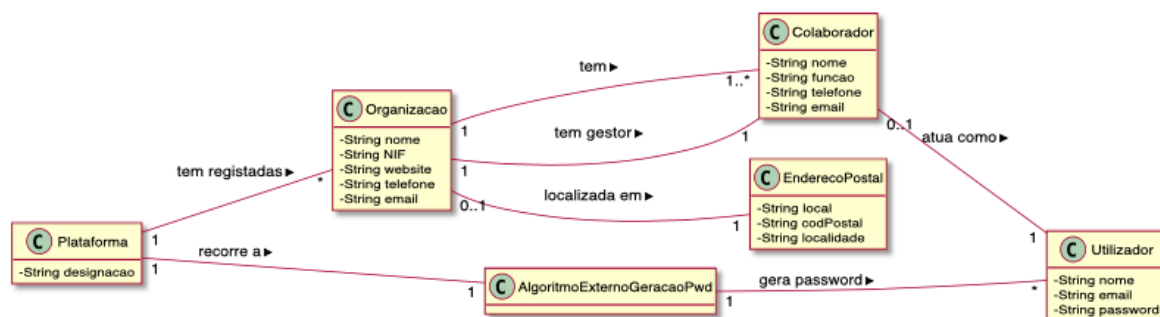


Diagrama de sequência

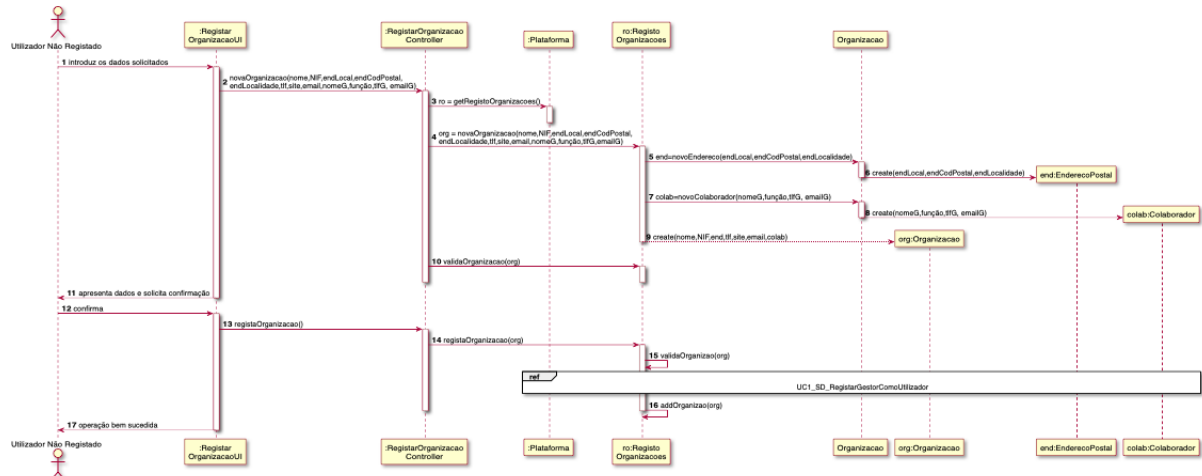
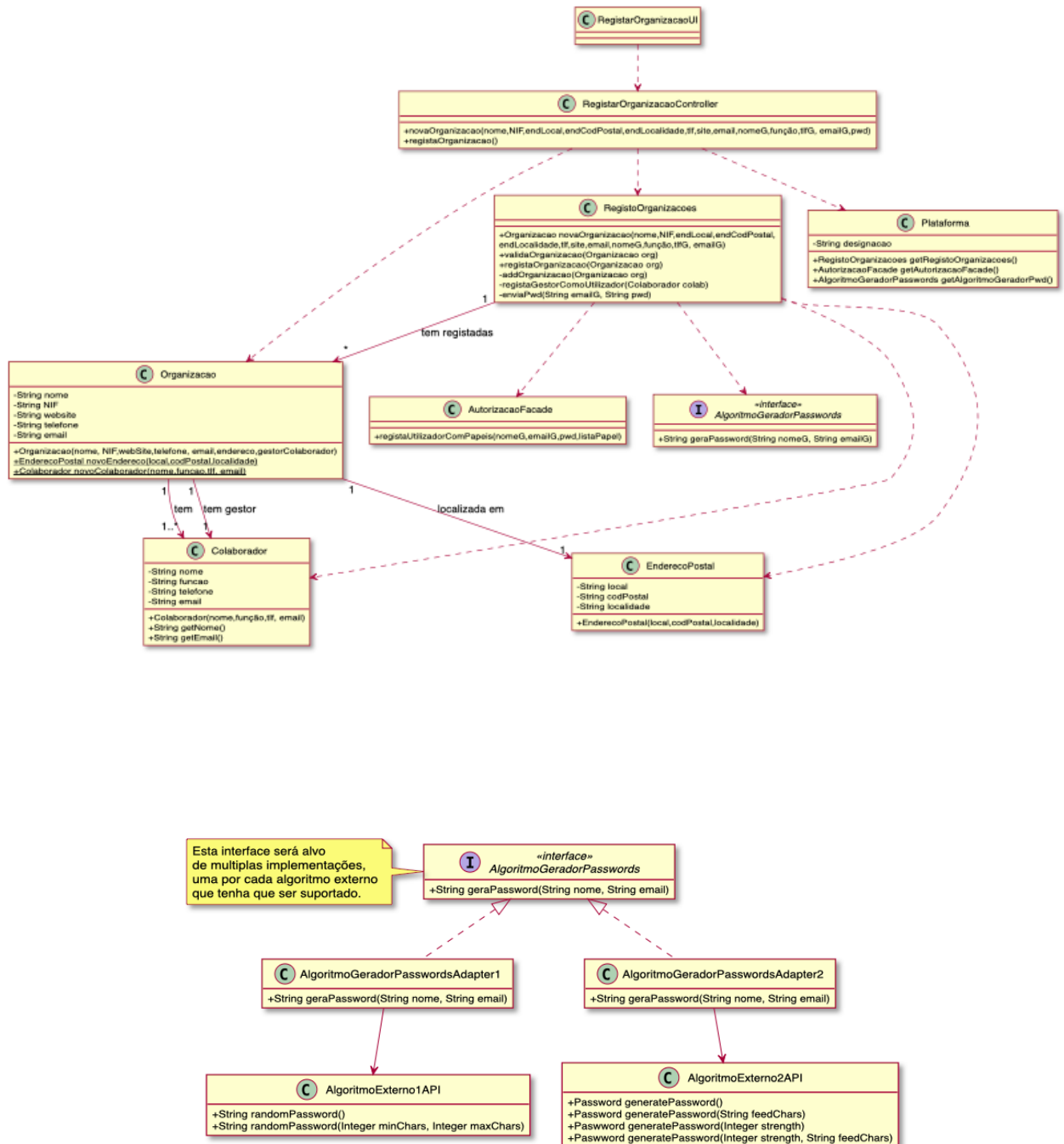


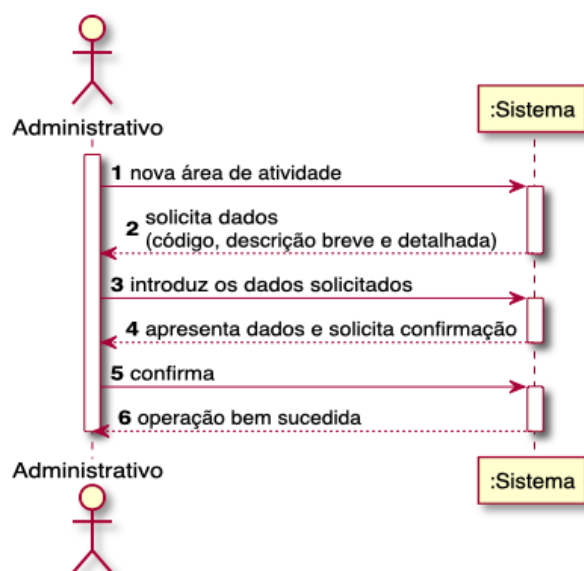
Diagrama de classes



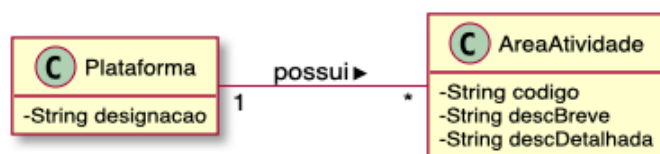
UC2 - DEFINIR ÁREA DE ATIVIDADE

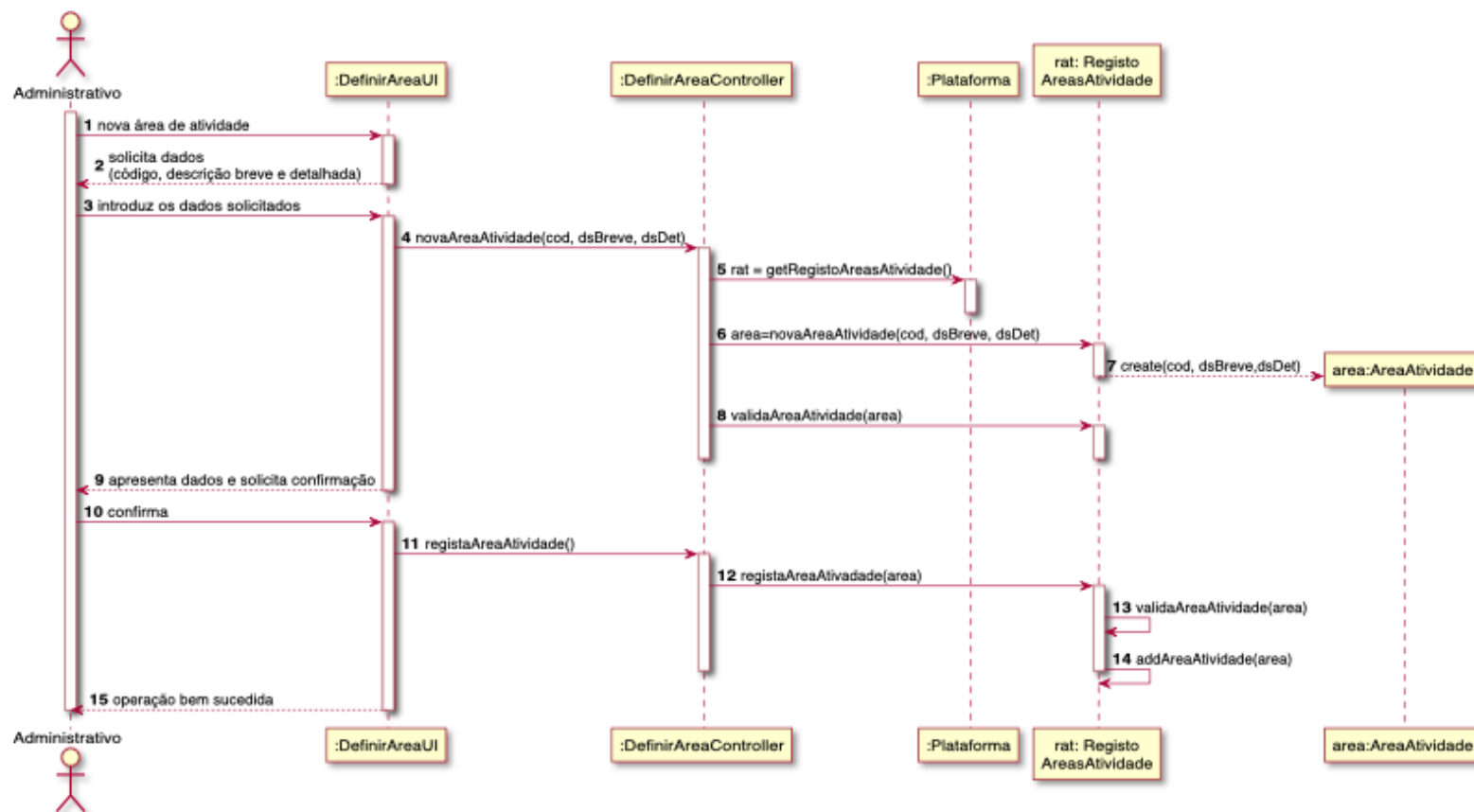
O administrativo inicia a definição de uma nova área de atividade. O sistema solicita os dados necessários (i.e. código único e descrição breve e detalhada). O administrativo introduz os dados solicitados. O sistema valida e apresenta os dados ao administrativo, pedindo que os confirme. O administrativo confirma. O sistema regista os dados e informa o administrativo do sucesso da operação.

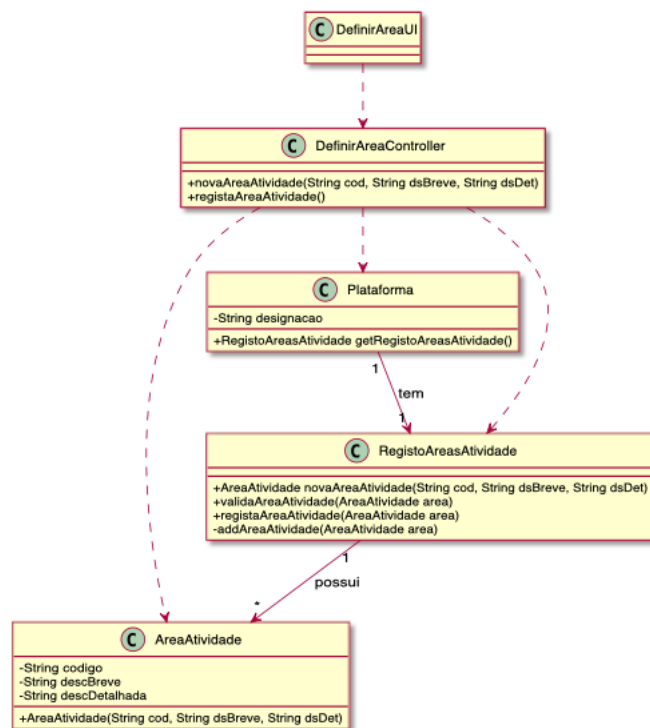
Diagrama de sequência de sistema



Excerto do modelo do domínio



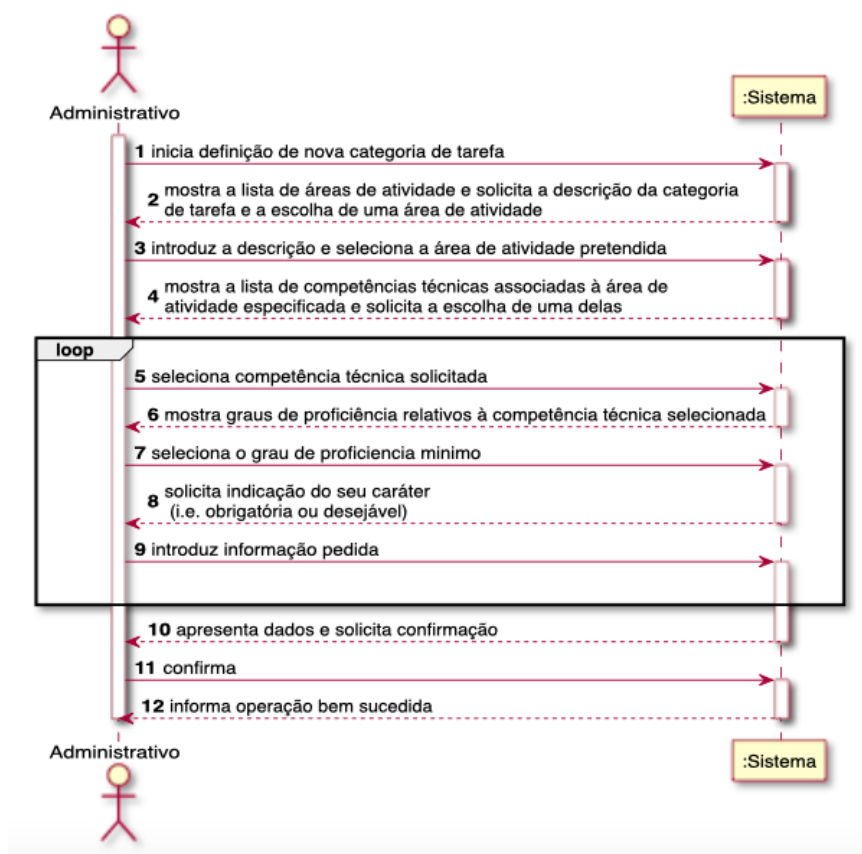




UC3 - DEFINIR CATEGORIA DE TAREFA

O administrativo inicia a definição de uma nova categoria de tarefa. O sistema solicita os dados necessários (i.e. descrição, área de atividade e **lista de competências técnicas requeridas e respetiva caracterização**). O administrativo introduz os dados solicitados. O sistema valida e apresenta os dados ao administrativo, pedindo que os confirme. O administrativo confirma. O sistema regista os dados e informa o administrativo do sucesso da operação.

Diagrama de sequência de sistema



Excerto do modelo do domínio

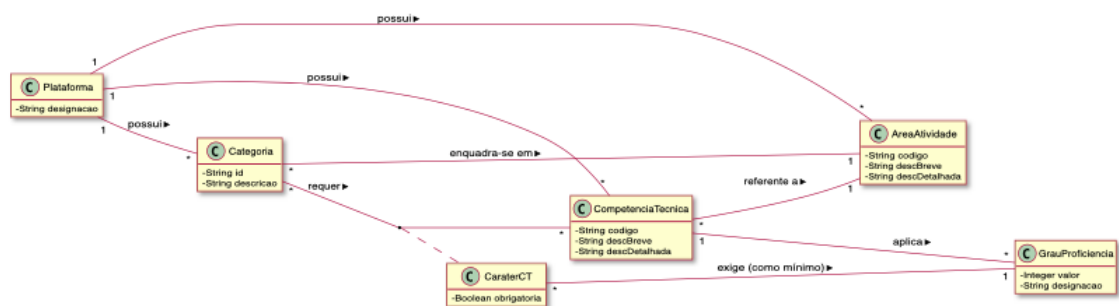
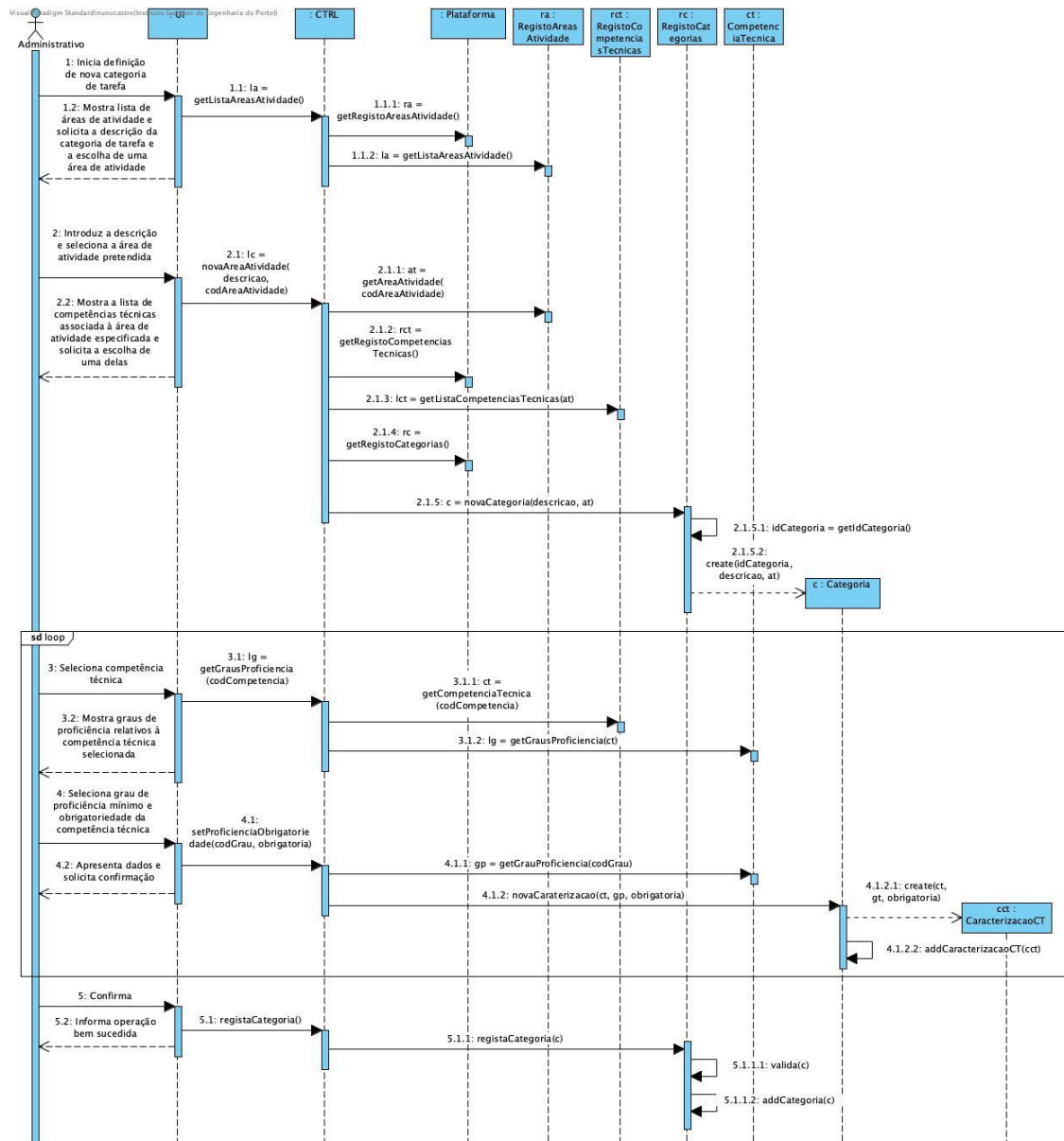


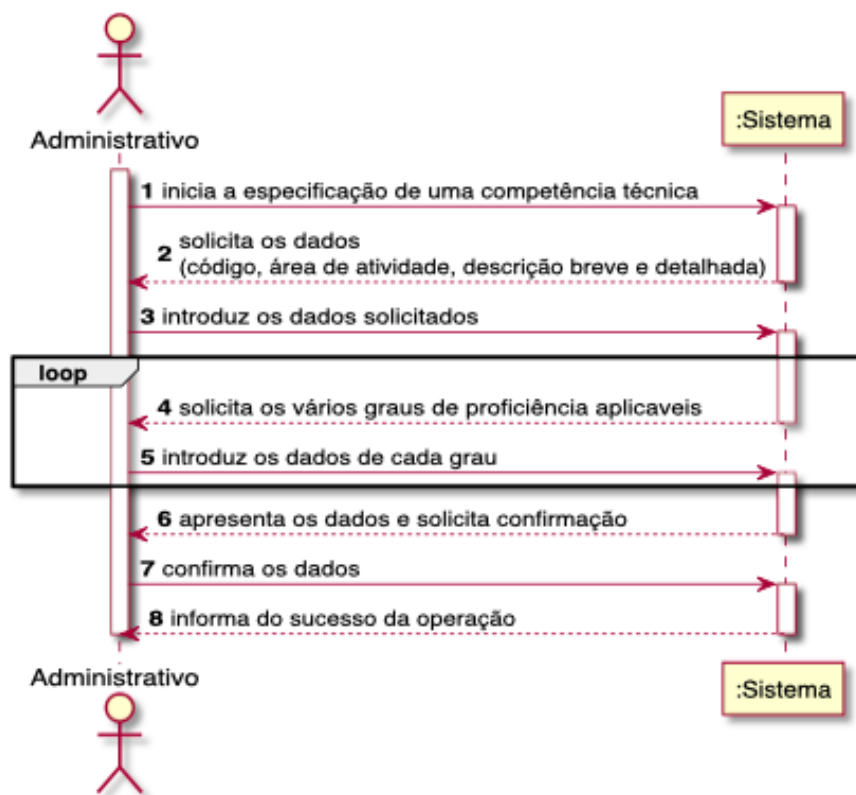
Diagrama de sequência



UC4 - ESPECIFICAR COMPETÊNCIA TÉCNICA

O administrativo inicia a especificação de uma competência técnica. O sistema solicita os dados necessários (i.e. código **único**, área de atividade e descrição breve e detalhada). O administrativo introduz os dados solicitados. O sistema solicita os vários graus de proficiência aplicáveis a esta CT. O administrativo introduz os dados de cada grau. O sistema **valida** e apresenta os dados ao administrativo, pedindo que os confirme. O administrativo confirma os dados. O sistema **regista os dados** e informa o administrativo do sucesso da operação.

Diagrama de sequência de sistema





Excerto do modelo do domínio

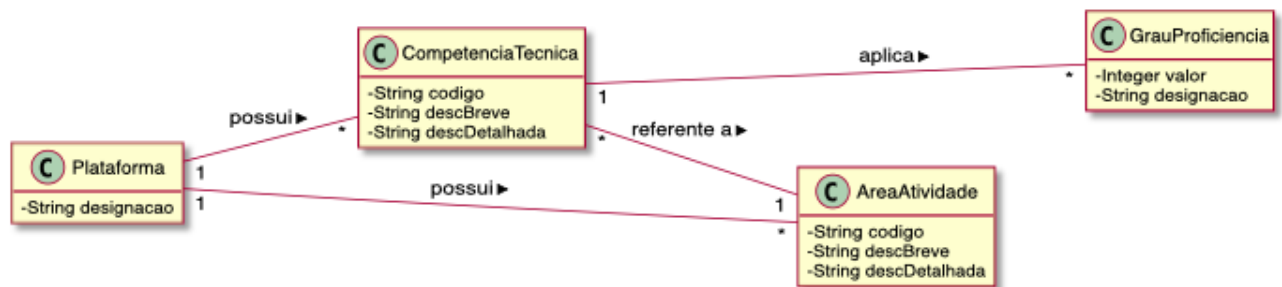
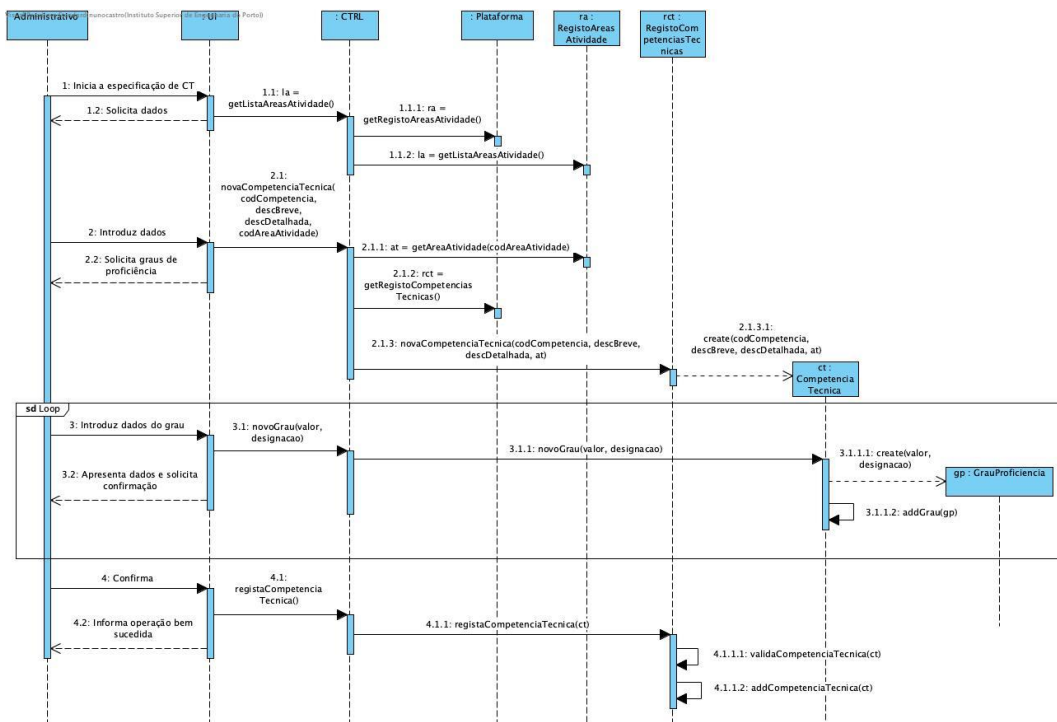


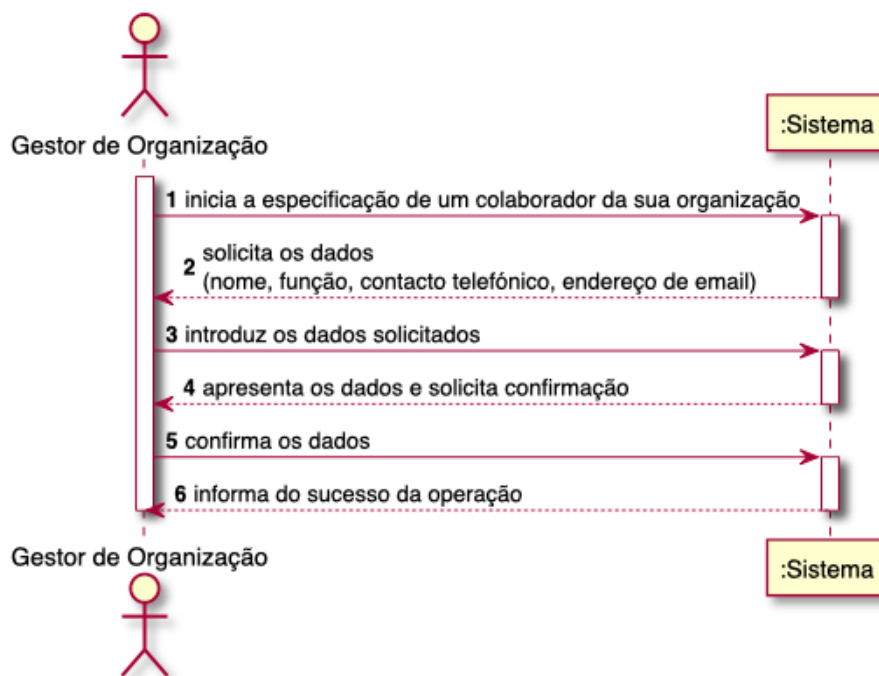
Diagrama de sequência



UC5 - ESPECIFICAR COLABORADOR DE ORGANIZAÇÃO

O gestor de organização inicia a especificação de um colaborador da sua organização. O sistema solicita os dados necessários (i.e. nome, função, contacto telefónico, endereço de email). O gestor de organização introduz os dados solicitados. O sistema valida e apresenta os dados ao gestor de organização, pedindo que os confirme. O gestor de organização confirma. O sistema regista os dados do colaborador, **gera uma password** para o colaborador, tornando este num utilizador registado, **envia, por email, os dados de acesso ao colaborador** e informa o gestor de organização do sucesso da operação.

Diagrama de sequência de sistema



Excerto do modelo do domínio

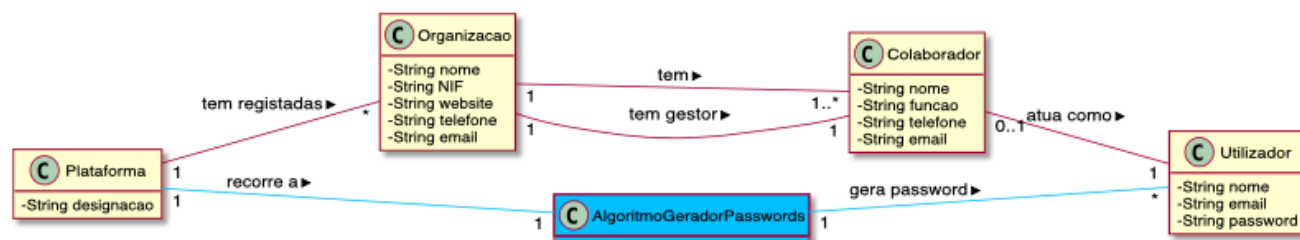
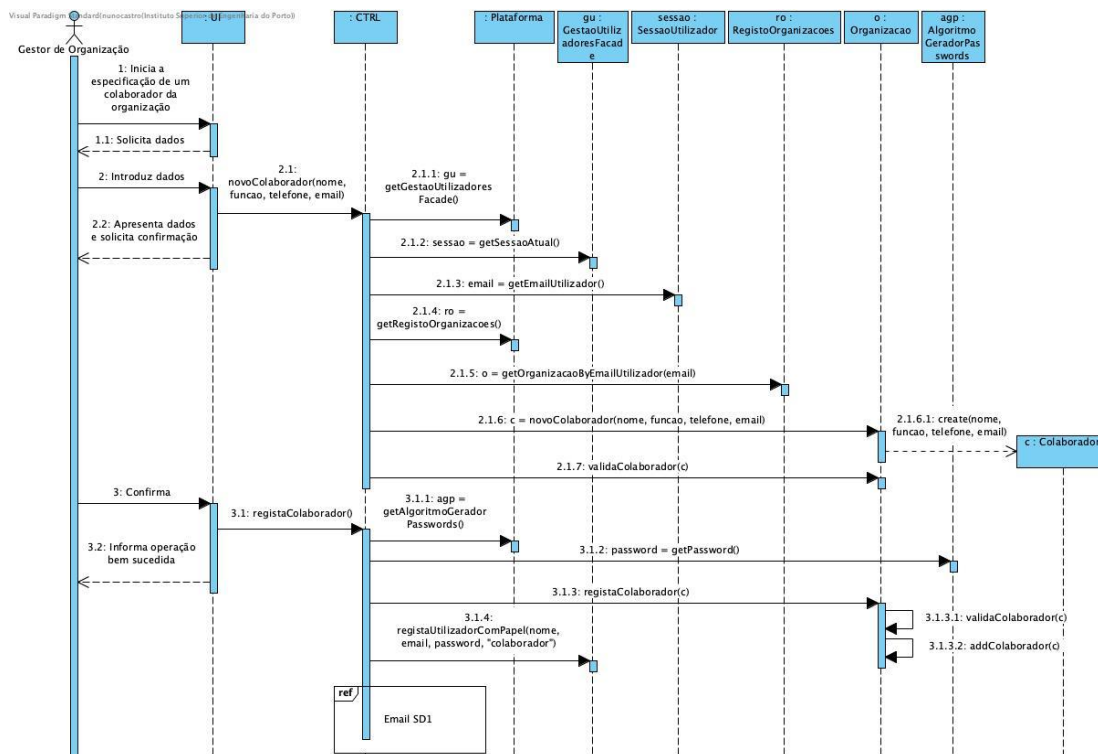


Diagrama de classes



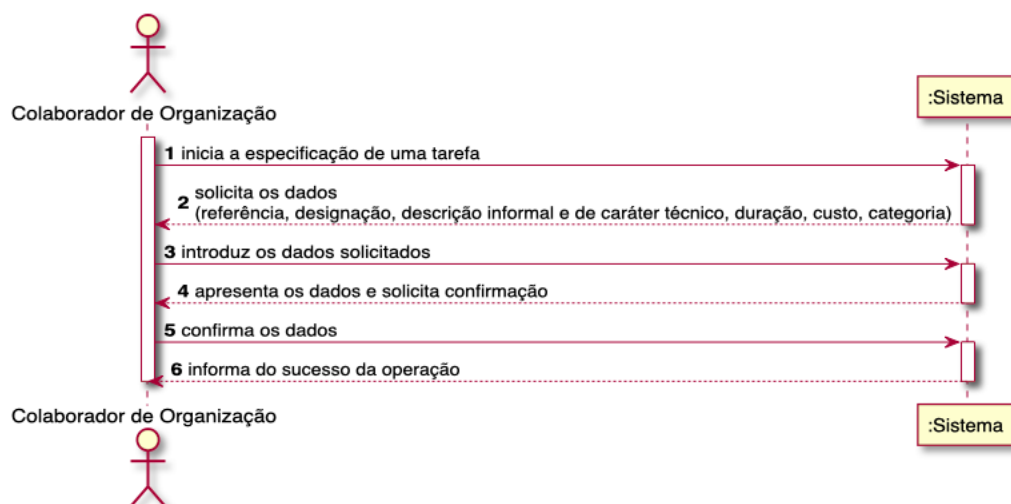
UC6 - ESPECIFICAR TAREFA

O colaborador de organização inicia a especificação de uma tarefa. O sistema solicita os dados necessários (i.e. uma referência única por organização, uma designação, uma descrição informal e outra de carácter técnico, uma estimativa de duração e custo, a categoria). O colaborador de organização introduz os dados solicitados. O sistema **valida** e apresenta os dados ao colaborador de organização, pedindo que os confirme. O colaborador de organização confirma. O sistema **registra os dados** e informa o colaborador de organização do sucesso da operação.

Diagrama de sequência de sistema

Laboratório de Programação

2º SPRINT – Tasks for Joe (T4J)



Excerto do modelo do domínio

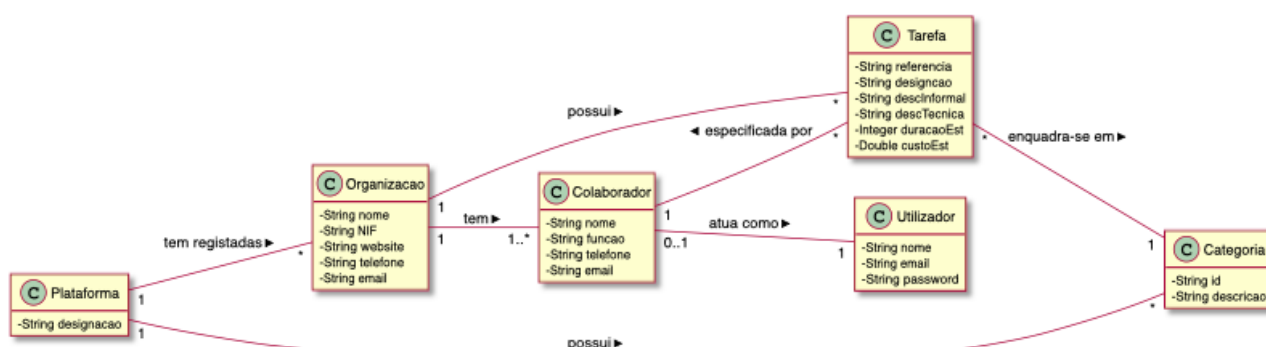
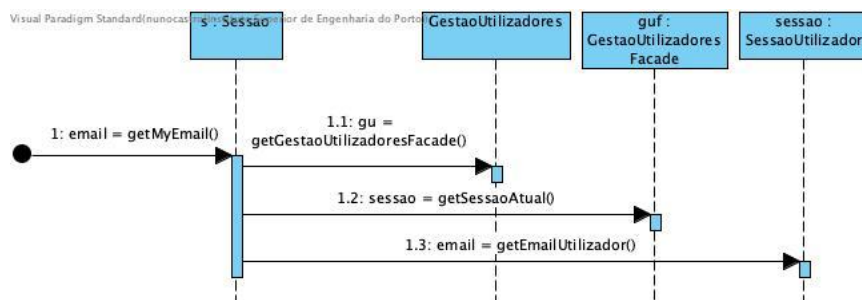
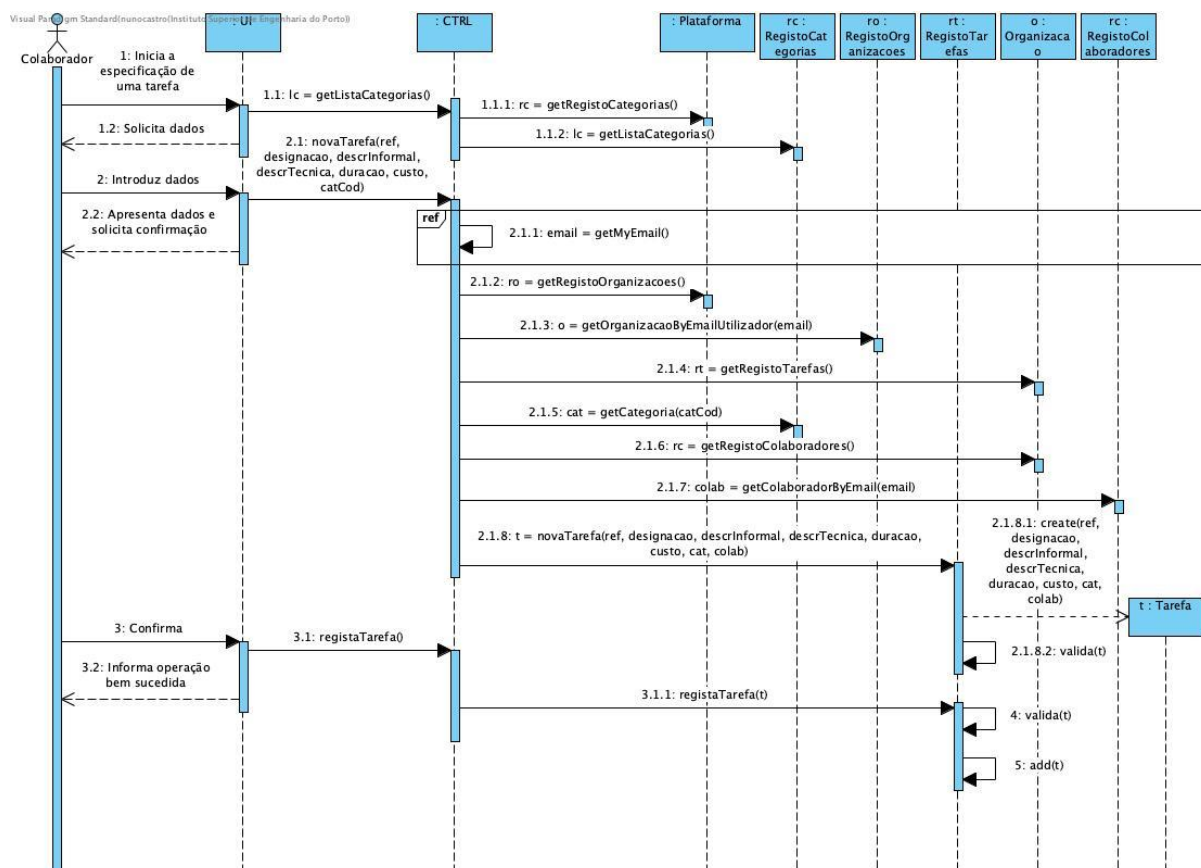


Diagrama de sequência



IMPLEMENTAÇÃO

Desenvolva uma aplicação que disponibilize os casos de uso apresentados acima.

Todo o código produzido deve ter sempre em consideração os principais princípios da programação orientada por objetos: abstração, encapsulamento, herança e polimorfismo.

O núcleo principal do software deve ser implementado em Java. Com o intuito de aumentar a manutenibilidade do software, devem ser adotadas boas práticas de análise e design de software OO. Aplique o processo de desenvolvimento de software designado por **Test Driven Development (TDD)** na implementação das classes.

Dever-se-á utilizar o *plugin* Maven JaCoCo (Java Code Coverage) no IDE Netbeans para verificar a cobertura de testes.

O código deverá conter os comentários necessários para que possa ser gerada a documentação usando a ferramenta Javadoc.

A implementação do software deve adotar normas de codificação (e.g. Camel case) e de controlo de versões. O controlo de versões será conseguido, usando o GitHub.

O trabalho deverá ser realizado por **um grupo de quatro formandos**.

Deverá ser submetido no Moodle do UPskill, um ficheiro ZIP com: o projeto Maven com o seguinte formato:

JavaNºTurma_PrimeiroUltimoNome_PrimeiroUltimoNome_PrimeiroUltimoNome_PrimeiroUltimoNome, como por exemplo, Java1_IsabelBras_ArturSilva_CarlaCosta_António_Sousa; e um pdf com a parte restante do design.

O projeto Maven deve ser implementado recorrendo a um repositório do GitHub, criado e configurado por um dos elementos do grupo. Na turma Java1, os professores Jorge Santos (ajs@upskill.pt) , Alexandre Gouveia (aas@upskill.pt), Nuno Morgado (nvm@upskill.pt), Marílio Cardoso (joc@upskill.pt) e Paulo Baltarejo (pbs@upskill.pt) têm de ser adicionados à lista de elementos com acesso ao repositório. Na turma Java2, os professores Nuno Melo e Castro (anc@upskill.pt), Jorge Duarte (fjd@upskill.pt), Alexandre Gouveia (aas@upskill.pt), Nuno Morgado (nvm@upskill.pt), Isabel

Laboratório de Programação

2º SPRINT – Tasks for Joe (T4J)

Sampaio (ais@upskill.pt) e Joaquim Santos (jpe@upskill.pt) têm de ser adicionados à lista de elementos com acesso ao repositório.

O trabalho deverá ser submetido no Moodle até às 17:30 do dia 8 de fevereiro de 2021 (segunda-feira).