

# Planeamento e Gestão de Projeto

Enunciado do Projeto — 2023/2024

## Introdução

Pretende-se fazer a análise, desenho inicial, e planeamento de um projeto de desenvolvimento de um sistema de informação sobre o tema de **perdidos e achados**.

São valorizados trabalhos cuja análise demonstre uma boa compreensão do problema, cujo desenho inicial tenha as bases para uma solução adequada e fazível com tecnologias existentes e com integração com sistemas externos, e cujo planeamento tenha em conta as pessoas da equipa, a análise e desenho inicial do produto, o processo de desenvolvimento, e a calendarização das tarefas do projeto.

O planeamento vai ser executado no próximo semestre, pelo mesmo grupo, em Projeto de Tecnologias de Informação (PTI) e Projeto de Tecnologias de Redes (PTR), pelo que se incluem aqui elementos essenciais ao sucesso nessas duas disciplinas.

## Descrição inicial do sistema

O sistema deve suportar três vertentes principais: a) o registo e pesquisa de objetos perdidos; b) o registo e pesquisa de objetos achados e entregues em postos da polícia; e c) o leilão de objetos entregues, mas que não foram reclamados após um longo período de tempo. Dois aspetos importantes a ter em conta são a necessidade de haver correspondências entre as descrições de objetos perdidos e achados, para evitar usos abusivos do sistema, bem como ser dada prioridade ao local onde os utilizadores julgam ter perdido os objetos, para focar as pesquisas.

## Atores

Os atores que interagem com o sistema são, pelo menos, os seguintes:

- **Dono:** regista e procura um ou mais objetos que perdeu.
- **Polícia:** regista objetos que alguém achou e entregou.
- **Licitante:** licita um objeto não reclamado em leilão.

Para além destes atores, são habitualmente considerados administradores de sistema bem como sistemas externos que oferecem serviços de autenticação de utilizadores, pagamentos *online*, mapas digitais, entre outros.

## Requisitos de informação

Alguns dados a registar são comuns aos **donos** e aos **licitantes**, nomeadamente: nome, género, ano de nascimento, morada (e respetivas coordenadas geográficas), identificação fiscal e civil, telemóvel de contacto, e endereço de correio eletrónico. No caso do **polícia**, é preciso saber o nome, o identificador interno, e o posto de

polícia onde trabalha, que tem uma morada. Quando alguém entrega um objeto que achou a um polícia, é necessário registar os mesmos dados sobre essa pessoa que para os donos e licitantes.

Todos os **objetos**, perdidos ou achados, têm diversos atributos: a) uma descrição, que permite fazer pesquisas genéricas por texto; b) campos específicos que dependem da categoria do objeto (por exemplo, se é um equipamento informático) como a marca, o modelo, o número de série, a cor, e outros; c) um local, onde o dono julga ter perdido o objeto ou onde alguém encontrou um objeto; e d) uma data, que pode ser a da perda ou do achamento, conforme o caso.

O conceito de local mencionado no parágrafo anterior deve adaptar-se a diferentes situações, suportando vários níveis hierárquicos, tais como rua, freguesia, município, distrito, e país. Por exemplo, quando um objeto é achado, o local pode corresponder a uma rua específica e até um dado número de porta; mas, o dono de um objeto perdido pode ter dúvidas sobre o local exato, pelo que deve poder indicar que foi algures numa freguesia inteira.

De igual forma, uma data pode ser específica, tal como um certo dia por volta de uma hora em que um objeto foi encontrado; mas também pode ser um intervalo de dias em que um dono admite que possa ter perdido um objeto.

Os **objetos achados** podem vir a ser leiloados se não forem reclamados pelos legítimos donos dentro de um prazo, que tem de ficar registado.

Para um objeto ir a **leilão** tem de ter um valor monetário atribuído, para servir de base de licitação, sendo que os licitantes podem fazer lances desde que o leilão começa até que termina. O período de tempo em que um leilão está ativo tem de ser conhecido antes de este começar. O licitante que fizer o lance mais elevado passa a ser o novo dono do objeto e faz um pagamento à entidade que gere o sistema.

Naturalmente, podem ser considerados mais dados, especialmente se forem suportados por resultados de pesquisas sobre este tema.

## **Requisitos funcionais**

O sistema deve suportar os seguintes requisitos, sendo que praticamente todos estão disponíveis apenas para utilizadores autenticados. O símbolo ‘\*’ no final do nome de um requisito indica menor prioridade face aos restantes.

### **Contas e sessões de utilizadores**

- RF-01: Registar, editar, e remover uma conta de dono ou licitante.
- RF-02: Registar, editar, e remover um posto de polícia.
- RF-03: Registar, editar, e remover uma conta de utilizador polícia.
- RF-04: Iniciar e terminar uma sessão autenticada.
- RF-05: Desativar e reativar uma conta de utilizador. \*

### **Objetos perdidos e achados**

- RF-06: Registrar, editar, e remover um objeto perdido.
- RF-07: Ver histórico dos seus objetos perdidos.
- RF-08: Registrar, editar, e remover um objeto achado.
- RF-09: Ver histórico dos seus objetos achados.
- RF-10: Pesquisar objetos perdidos pela descrição.
- RF-11: Pesquisar objetos perdidos pelos campos específicos de categorias.
- RF-12: Pesquisar objetos achados correspondentes a objetos perdidos.
- RF-13: Comparar um objeto perdido com um achado, revelando as diferenças. \*
- RF-14: Registrar, editar, e remover possível dono de objeto achado.
- RF-15: Notificar dono que um seu objeto perdido pode ter sido achado. \*
- RF-16: Registrar entrega de objeto achado ao dono.
- RF-17: Ver relatório com estatísticas e mapa de objetos perdidos e achados.

### **Leilões de objetos não reclamados**

- RF-18: Registrar, editar, e remover um leilão de um objeto.
- RF-19: Ver leilões passados, ativos, e futuros.
- RF-20: Subscrever e cancelar notificações sobre um leilão. \*
- RF-21: Notificar evento ocorrido em leilão. \*
- RF-22: Iniciar e terminar um leilão.
- RF-23: Ver histórico de licitações num leilão.
- RF-24: Licitar um objeto num leilão.
- RF-25: Pagar objeto licitado.
- RF-26: Ver histórico de objetos comprados em leilão.

Esta lista de requisitos funcionais pode sofrer mudanças. Aliás, os grupos são incentivados a acrescentar requisitos para distinguirem o seu sistema dos restantes.

### **Requisitos não funcionais**

O desenho inicial do sistema e o planeamento do projeto devem ter em conta os requisitos não funcionais mencionados a seguir.

#### **Obrigatórios**

- RNF-01: **Arquitetura cliente-servidor *Web***. Para aceder ao sistema, os utilizadores devem usar um *browser Web* que comunica com um servidor aplicacional. Por sua vez, o servidor deve ter um *front end* e um *back end*, que são aplicações independentes que interagem na prestação de serviços ao exterior. O *front end*

implementa uma API, fornece conteúdos HTML, CSS e JavaScript, e trata da autenticação/autorização, enquanto o *back end* implementa a lógica de negócio e o modelo, que abstrai as operações disponibilizadas pela base de dados. Ambos podem ser decompostos em aplicações mais pequenas, dependendo do projeto.

- RNF-02: **Coerência dos dados.** O sistema deve garantir a coerência e integridade dos dados, de acordo com as regras do negócio.
- RNF-03: **Usabilidade.** As funcionalidades do sistema devem ser oferecidas de forma eficaz e eficiente, proporcionando uma experiência de utilização satisfatória. O sistema deve *adaptar as funcionalidades consoante o tipo de dispositivo* do utilizador, nomeadamente ecrãs de diferentes tamanhos e formas de interação.
- RNF-04: **Fonte aberta.** O sistema deve ser de código fonte aberto.

### Importantes

- RNF-05: **Ambiente de desenvolvimento.** Deve ter ferramentas de controlo de versões que suportem uma equipa de programadores. Adicionalmente devem ser usados mecanismos de automação que facilitem o desenvolvimento da aplicação.
- RNF-06: **Lançamento em produção.** O sistema deve ter um automatismo que permita lançá-lo em produção a partir de um repositório de código.
- RNF-07: **Tempo de resposta e desempenho.** O sistema deve ter um tempo de resposta e um desempenho adequados à expectativa dos utilizadores.
- RNF-08: **Interoperabilidade.** O sistema deve ter uma API REST que lhe permita a interação com outras aplicações.
- RNF-09: **Escalabilidade.** O sistema deve ter uma capacidade ajustável para atender pedidos. Pode ser horizontal, acrescentando nós, e/ou vertical, adicionando recursos aos nós existentes. Um aspeto complementar é a elasticidade com que o sistema adapta automaticamente a sua capacidade em função da carga, aumentando ou diminuindo o número de nós (dentro de limites pré-definidos).
- RNF-10: **Tolerância a falhas.** O sistema deve tolerar a falha de qualquer dos seus componentes, através de redundância.
- RNF-11: **Segurança.** O sistema e as redes devem estar cobertas por políticas de segurança que fazem uso de mecanismos de proteção de modo a impedir o acesso não autorizado a recursos e dados e também impedir a disrupção do serviço.
- RNF-12: **Testes.** O sistema deve ser desenvolvido com uma política integrada de testes que permita ir testando à medida que se adicionam novos componentes ou que se alteram componentes existentes.

### Outros

- RNF-13: **Salvaguarda de dados.** Os dados do sistema devem ser salvaguardados regularmente conforme indicado numa política de *backup*. Adicionalmente, deve estar definido um procedimento de restauro de qualquer parte dos dados.

- RNF-14: **Custo.** Deve ser escolhida uma plataforma de computação que preste o melhor serviço pelo preço mais baixo, idealmente de forma gratuita.
- RNF-15: **Proteção de dados.** O sistema deve disponibilizar informação para os utilizadores sobre a forma como os dados são tratados e a possibilidade de consulta e eliminação dos seus dados pessoais.

Cada grupo deve escolher os requisitos não funcionais a serem concretizados pelo seu sistema, especificando, para cada um, *o que deve ser satisfeito* pela solução. Por exemplo, a usabilidade pode ser avaliada com testes SUS, e a pontuação mínima desejada pode ser cerca de 85, correspondente a excelente.<sup>1</sup>

No final da implementação do sistema, a satisfação dos requisitos não funcionais deve ser comprovada, por exemplo através de testes à API, testes de carga, análise de vulnerabilidades, testes de usabilidade com utilizadores, e outros meios.

## Realização

O trabalho do projeto está organizado em duas etapas, com os objetivos indicados a seguir, sendo que, em geral, os diagramas seguem a notação UML.<sup>2</sup>

### Etapa 1: até 3 de novembro

Tem como propósito produzir artefactos relativos às seguintes atividades:

- **Modelação:** desenhar um diagrama de *classes* com o esquema de dados e restrições de integridade, diagramas de *atividades* para descrever processos de negócio, e um diagrama de *implementação* com a arquitetura do sistema. Também podem ser elaborados artefactos que o grupo achar relevantes, tais como diagramas de casos de uso, de estados, e protótipos da interface com o utilizador.
- **Pesquisa:** escolher, de forma justificada, uma *framework* de desenvolvimento de aplicações *Web*, e serviços externos para fazer autenticação de utilizadores, mostrar mapas e calcular distâncias, efetuar pagamentos *online*, entre outros que possam vir a ser necessários. A justificação de cada decisão deve incluir uma lista de vantagens, de preferência comparando com um produto concorrente.
- **Planeamento preliminar:** identificar as principais tarefas de desenvolvimento do sistema e fazer um esboço da calendarização, por exemplo ao mês, sem indicar quem faz o quê. Este planeamento serve para identificar lacunas e problemas, tais como a ausência de testes de usabilidade com utilizadores ou estes acontecerem demasiado tarde para poderem ser feitas melhorias ao sistema, ou as funcionalidades da interface com o utilizador serem desenvolvidas independentemente da implementação da API, com correspondente duplicação de esforço.

<sup>1</sup> Bangor, A., Kortum, P., & Miller, J. (2009). Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of Usability Studies*, 4(3), 114–123.

<sup>2</sup> Fowler, M. (2005). *UML distilled: A brief guide to the standard object modeling language* (3ª edição). Addison-Wesley.

## Etapa 2: até 30 de novembro

Tem como objetivo planejar o trabalho a realizar no próximo semestre nas disciplinas de PTI e PTR, tendo em conta aspetos relacionados com as pessoas, o produto, o processo, e o projeto. Assim, pretende-se:

- Identificação de **recursos** humanos, das ferramentas de desenvolvimento, e do *software* necessário ao funcionamento do sistema.
- **Estimação** do esforço disponível e do esforço necessário à realização do trabalho, usando dados históricos, decomposição, e um modelo empírico.
- Escolha do **modelo de processo** de desenvolvimento de *software*, com justificação da opção tomada, sendo o modelo Scrum uma possibilidade viável.
- Organização da equipa, e atribuição de **papéis**, tais como gestor de projeto, consultor de dados, programador de *front end* e de *back end*, administrador da base de dados, arquiteto de sistema, especialista de redes e de segurança, técnico de manutenção, e responsável de testes com utilizadores.
- Descrição dos **work packages**, suas tarefas, recursos associados, e principais resultados (*milestones*), e respetiva **calendarização** num mapa de Gantt.
- Gestão de **risco**, incluindo a preparação de uma tabela de riscos, e um plano de mitigação, monitorização, e gestão dos principais riscos.
- Uma versão preliminar da **API REST** que o sistema deve suportar, sendo recomendado o uso da especificação OpenAPI e do editor Swagger.

De notar que as datas das etapas estão sujeitas à aprovação da coordenadora da LTI e podem vir a ser alteradas.

## Entregas

Em cada etapa, os elementos pedidos devem ser colocados num arquivo ZIP com nome PGP-2324-GXX-EY.ZIP, onde XX é o número do grupo (ex. 01) e Y é o número da etapa, o qual deve ser entregue através de atividades existentes na página de entrada da disciplina no Moodle.

## Avaliações

Os projetos são avaliados na presença de cada grupo, sendo atribuídas notas qualitativas. Tendo em conta as avaliações presenciais e os ficheiros entregues, é posteriormente atribuída uma nota quantitativa.

Bom trabalho no projeto!