



## Escopo do Projeto

O presente trabalho consiste na definição, especificação, implementação, teste, documentação e apresentação do seguinte projeto:

Sistema de cadastramento de pets disponíveis para adoção, contendo:

- **Cadastro dos pets:** características dos pets, necessidades de saúde, tratamentos contínuos e necessidades especiais. Status do pet (disponível para adoção, adotado, perdido ou falecido).
- **Histórico de Rastreabilidade:** registro de localização e por quanto tempo, desde o resgate do animal (que pode ter ocorrido em data anterior ao cadastro deste no site).
- **Cadastro dos tutores:** dados de contato do tutor, que pode ter um ou mais pets cadastrados no sistema.
- **Cadastro de possíveis adotantes:** dados de contato do adotante, pets que já possuem (cadastrados no sistema) e características desejadas do pet a ser adotado. Apenas um pet pode ser buscado para adoção por vez.
- **Acompanhamento pós-adoção:** após a adoção, o sistema deve ser capaz de permitir o acompanhamento do pet no novo lar, por pelo menos 6 meses, disponibilizando uma funcionalidade de registro de visitas e interações com o tutor, bem como incidentes de saúde (como vacinação, castração, alguma doença, etc.) com emissão de relatório.
- **Quero adotar:** funcionalidade implementada separada do restante do sistema, permitindo que seja realizada uma busca pela lista de pets disponíveis para adoção, levando em consideração o cadastro dos pets e adotantes e utilizando o grau de compatibilidade entre pet e adotante. Esta funcionalidade deverá ser implementada como sistema web ou celular, a escolha do aluno.

O sistema deverá, ainda, emitir um *relatório de compatibilidade*, gerando uma lista de adotantes e pets com maior chance de combinação ao adotante. Para calcular a compatibilidade entre adotante e pet, deverá ser utilizada a seguinte metodologia:

**Opção 1:** compatibilidade é igual à soma dos seguintes pontos:

Categoria	Característica do pet	Característica desejada pelo adotante	Pontos
Espécie	Mesma desejada pelo adotante		+20
Espécie <sup>1</sup>	Diferente da desejada pelo adotante		-20
Raça	Mesma desejada pelo adotante		+10
Raça	Diferente da desejada pelo adotante		-10
Porte	Mesmo desejado pelo adotante		+10
Porte	Diferente do desejado pelo adotante		-10
Sexo	Igual ao desejado pelo adotante		+5
Sexo	Diferente do desejado pelo adotante		-5
Saúde	Com necessidades especiais ou com tratamento contínuo	Aceita necessidades especiais ou tratamento contínuo	+10
Saúde	Sem necessidades especiais e sem tratamento contínuo	Aceita necessidades especiais ou tratamento contínuo	+5
Saúde	Sem necessidades especiais e sem tratamento contínuo	Somente aceita sem necessidades especiais e sem tratamentos contínuos	+10
Saúde	Sem nenhuma doença crônica ou incurável	aceita somente sem nenhuma doença crônica ou incurável	+5
Saúde	Com doença crônica ou incurável	Aceita com doenças crônicas ou incuráveis	+10
Social	Animal se dá bem com outros animais, inclusive de outras espécies	Já possui outros animais	+5
Social	Animal não se adapta a convivência com outros animais	Já possui animais	Fator impeditivo!
Social	Animal exige cuidados constantes	Possui disponibilidade de tempo para dedicar ao animal	+5
Social	Animal exige cuidados constantes	Não possui disponibilidade de tempo para dedicar ao animal	Fator impeditivo!

**Importante:** esta tabela é apenas uma *sugestão* e deve servir como ponto de partida para uma análise por parte do aluno, que deverá avaliar a necessidade utilização de outros fatores e testar o respectivo impacto no relatório de compatibilidade.

**Opção 2:** integração a alguma plataforma de inteligência artificial via API para verificação do grau de compatibilidade entre potencial adotante e pet. Por exemplo, [essa consulta ao chatGPT](#), ou ainda através de uma plataforma como <https://aimlapi.com/>, [OpenAI](#), ou [Puter.JS](#).

<sup>1</sup> [https://pediaa.com/difference-between-breed-and-species/#google\\_vignette](https://pediaa.com/difference-between-breed-and-species/#google_vignette)

### Observações:

- O foco do projeto deverá ser o *backend* da solução. Entretanto, o *frontend* deverá permitir ao usuário o uso facilitado da solução. Caso o aluno assim deseje, poderão ser utilizados bibliotecas prontas ou modelos (HTML5 responsivos, por exemplo) para a implementação do *frontend*.
- Todos os integrantes do grupo deverão ser capazes de responder a qualquer questionamento no momento da apresentação.

### Lista de entregáveis do projeto:

#### Avaliação E1

- Planejamento macro do projeto (*Project Charter*):
  - Escopo inicial de cada Sprint.
  - Definição do stack tecnológico.
  - Definição dos papéis de cada membro da equipe.
  - Quais ferramentas de acompanhamento serão utilizadas pelo time?
  - Repositório para trabalho em grupo? Por exemplo, GitHub?
  - Mapa de responsabilidades para os membros da equipe.
  - Cronograma macro.
  - Indicadores de qualidade a serem controlados.
  - Metodologia a ser aplicada.
- Definições de produto:
  - **Mapas de Personagens (Personas):** Perfis detalhados que representam os usuários finais, suas necessidades, comportamentos, dores e objetivos.
  - **Jornada do Usuário (Customer Journey Map):** Visualização do percurso do usuário ao usar o produto, destacando pontos de contato, emoções e possíveis obstáculos.
  - **Mapa de Empatia (Empathy Map):** Ferramenta para compreender melhor o que o usuário pensa, sente, vê, ouve, fala e faz.
  - **Propostas de Valor (Value Proposition Canvas):** Descrição clara das dores, ganhos e como o produto atende às necessidades do cliente.
  - **Requisitos de Produto (User Stories e Critérios de Aceitação):** Histórias de usuário detalhadas alinhadas às personas para orientar o desenvolvimento.
- Avaliação Individual.

#### Avaliação E2

- *User stories* (ou entregável de escopo técnico equivalente), já refinados e estimados, para a Sprint 2.
  - Estas devem conter não apenas tarefas de *backend*, como também de *frontend* e infraestrutura.
- Relatório de retrospectiva da Sprint 1.

- Definições detalhadas do stack tecnológico.
- Definição detalhada e definitiva da infraestrutura.
- Desejável: wireframes e/ou protótipos (por exemplo, no Figma).
- Avaliação Individual.

### **Avaliação E3**

- *User stories* (ou entregável de escopo técnico equivalente), já refinados e estimados, para a Sprint 4.
- Interfaces com o usuário implementadas e navegáveis (não necessária integração ao *backend* ainda).
- Relatórios de retrospectiva das Sprints 2 e 3.
- Avaliação individual.

### **Avaliação E4**

- Ferramenta 100% implementada e funcional.
- Apresentação da ferramenta com demonstração das funcionalidades.
  - **Todas** as funcionalidades exigidas no escopo inicial proposto pelo professor deverão ser demonstradas em estado 100% funcional.
- Apresentação dos indicadores de entrega e qualidade de todo o projeto (Sprints 0 até 4).
- Entrega dos códigos-fonte através de repositório público, bem como da base de dados populada, permitindo a realização de testes.

### **Sobre as Avaliações Individuais**

Serão disponibilizadas tarefas no Moodle da disciplina para que cada aluno realize a sua avaliação na data planejada. Além disso, deverá ser entregue uma “avaliação geral”, com a perspectiva do grupo. Esta será entregue por um dos membros do time diretamente ao professor.

### **Considerações gerais**

Ao fim de cada Sprint, deverá ser realizada uma reunião de demonstração, para toda a turma, mostrando o que foi planejado e o que foi entregue durante o funcionamento (de acordo com o requisitado para cada Sprint).

As tecnologias utilizadas para a implementação do projeto, bem como o SGBD, poderão ser escolha do aluno. No entanto, o professor somente dará suporte ao seguinte *stack*:

C, C++, C#, Python, CSS, HTML 5, React, Javascript, Typescript. Para o banco de dados, SQL Server, MySQL, PostgreSQL, MongoDB ou Oracle.

O aluno é responsável pela correta configuração do ambiente de desenvolvimento e também pelo ambiente de produção. No dia da entrega final e apresentação do projeto, este deverá estar funcionando corretamente para a demonstração.

#### **Sobre user stories:**

- ✓ <https://www.atlassian.com/agile/project-management/user-stories>
- ✓ <https://awari.com.br/template-de-user-story-no-scrum/>
- ✓ <https://miro.com/pt/modelos/user-story-mapping/>
- ✓ [https://youtu.be/m7I4f\\_1lzWE?si=ZlWH\\_-q0jOj6PSD1](https://youtu.be/m7I4f_1lzWE?si=ZlWH_-q0jOj6PSD1)

#### **Sobre SCRUM:**

- ✓ <https://www.devmedia.com.br/introducao-ao-scrum/33724>
- ✓ <https://scrumguides.org/scrum-guide.html>

Dúvidas: *filipo.mor* AT *pucrs* DOT *br*

