

PetSpace

HISTÓRICO DE REVISÃO

Versão	Data	Descrição	Nome
1.0	18/02/2021	Criação do documento	Equipe Void
2.0	10/03/2021	Revisão do documento perante ao feedback do professor	Equipe Void

1. INTRODUÇÃO

1.1 DECLARAÇÃO DO PROBLEMA

O problema	A não praticidade e facilidade de obter produtos e serviços voltados aos pets
Afeta	Todos os pets e seus donos
Cujo impacto é	A pouca diversidade de interação dos donos e seus animais
Uma solução de sucesso seria	Um e-commerce para a venda de produtos, serviços e adoção de pets

1.2 OBJETIVO DO PROJETO

Criar um e-commerce voltados às necessidades e comodidade dos donos de pets. No meio da pandemia da COVID-19, vários aspectos foram afetados, um deles foi a livre circulação, tornando a compra pela internet, além de uma medida de segurança, prático. Com isso percebemos a oportunidade de criar uma loja virtual para que pessoas consigam comprar produtos e serviços focados para seus pets.

2. STAKEHOLDERS

Nome	Descrição	Responsabilidade
Usuário	Donos dos animais de estimação	Consumir os produtos, serviços e adotar animais

Nome	Descrição	Responsabilidade
Equipe de Desenvolvimento	Estudantes da disciplina de Requisitos de Software da UnB	Documentar, desenvolver e implementar a aplicação elaborada
Administrador	Membro da administração	Responsável por gerenciar a plataforma
Professor Orientador	Professor da disciplina Requisitos de Software	Avaliar e gerar feedback sobre projeto

3. VISÃO GERAL DO PRODUTO

3.1 DECLARAÇÃO DE POSIÇÃO DO PRODUTO

Para	Donos de Pets
Quem	Necessitam comprar produtos, agendar serviços voltado aos pets, adotar um novo pet
O PetSpace	É um e-commerce
Que	Busca fornecer praticidade e facilidade na compra de produtos, serviços e adoção de animais
Ao contrário	Da Petz, que somente faz venda de produtos
Nosso produto	Além da venda de produtos, fornece a oferta de serviços e adoção de pets

3.2 MÍNIMO PRODUTO VIÁVEL (MVP)

- Visualizar lista de produtos;
- Ver detalhes do produto;
- Adicionar produto ao carrinho;
- Autenticação do usuário no site.
- Finalizar compra.
- Avaliar produto.

4. VISÃO GERAL DO PROJETO

4.1 ORGANIZAÇÃO DO PROJETO

Papel	Atribuição	Responsável
Professor	Orientar e resolver problemas em última instância.	George
Scrum Master	Membro responsável por garantir o bom andamento e uso do framework Scrum.	Bruno Félix
Product Owner	Membro responsável por definir histórias e priorizar as funcionalidades do backlog que serão desenvolvidas no projeto	Daniel Primo
DevOps	Membro que entende o ciclo de vida de desenvolvimento de software e traz ferramentas e processos de engenharia de software para resolver desafios clássicos de operações	Francisco Ferreira
Time de Desenvolvimento	É a equipe de profissionais responsável por transformar o Product Backlog em um produto funcional.	Bruno Félix, Francisco Ferreira, Hércules Ismael, Daniel Primo e Pedro Henrique

5. FERRAMENTAS, AMBIENTE E INFRAESTRUTURA

Ferramentas	Descrição
WordPress	É um sistema livre e aberto de gestão de conteúdo para internet, baseado em PHP com banco de dados MySQL, executado em um servidor interpretador
MySQL	É um sistema de gerenciamento de banco de dados, que utiliza a linguagem SQL como interface.
phpMyAdmin	É um aplicativo web livre e de código aberto desenvolvido em PHP para administração do MySQL pela Internet.
Docker	É um conjunto de produtos de plataforma como serviço que usam virtualização de nível de sistema operacional para entregar software em pacotes chamados contêineres.
Heroku	É uma plataforma em nuvem como um serviço que suporta várias linguagens de programação.

Heroku	É uma plataforma em nuvem como um serviço que suporta várias linguagens de programação.
Travis CI	É um serviço de integração contínua hospedado usado para construir e testar projetos de software hospedados no GitHub e Bitbucket.
Dockerhub	É um repositório público onde empresas podem publicar suas soluções em forma de imagem
GitHub	É uma plataforma de hospedagem de código-fonte e arquivos com controle de versão usando o Git.

6. PROCESSO DE GERÊNCIA DE PROJETO

6.1 PLANEJAMENTO DAS FASES E ITERAÇÕES DO PROJETO

Sprint	Data	Descrição
Sprint 1	04/fev a 18/fev	Definição do projeto Visão do projeto
Sprint 2	23/fev a 11/mar	Organização estrutural dos requisitos Backlog do Produto
Sprint 3	16/mar a 01/abr	Backlog da Sprint Gerenciamento de requisitos - implementação

6.2 PROCESSOS DE DESENVOLVIMENTO E MENSURAÇÃO

O processo de desenvolvimento da aplicação será conduzido por meio do uso de certas práticas da metodologia SCRUM, tendo seus ritos adaptados à realidade do projeto, considerando o contexto de home office por conta da pandemia. O projeto será acompanhado por meio de reuniões semanais, para o planejamento da sprint e revisão da sprint anterior. Além disso, serão realizadas reuniões rápidas, nos moldes de dailys, da metodologia SCRUM, porém realizadas 3 vezes por semana, para rastrear eventuais impedimentos e ter uma boa perspectiva do que está sendo feito. As datas finais das fases do projeto serão as bases para as Releases do produto.

Nosso processo de mensuração, no planejamento da Sprint, será baseado nos Burndown report e Velocity tracking, do recurso ZenHub da plataforma GitHub. Usaremos também uma pontuação, o Planning Poker, para definir a dificuldade de desenvolvimento de uma User Story, assim mensurando bem os esforços e o tempo gasto durante a Sprint.

6.3 MATRIZ DE COMUNICAÇÃO

Descrição	Área/ Envolvidas	Periodicidade	Produtos Gerados
Revisar todas as tarefas feitas na semana anterior; Planejar as tarefas para a próxima semana.	Equipe de Desenvolvimento	Baseado no fim das Sprints do tópico 6.1	Planning da Sprint seguinte; Revisão da Sprint anterior; Ata da Reunião.
Comunicar o que está feito, empecilhos, e o que planeja ainda fazer	Equipe de Desenvolvimento	Segunda, Quarta e Sexta	Ata da Reunião
Receber feedbacks das entregas do projeto	Equipe de Desenvolvimento e o Professor	Ao final de cada Sprint	Feedback

6.4 ESCALABILIDADE DO PROJETO

A necessidade de mudança e resolução de problemas é uma característica inerente ao desenvolvimento de software.

Caso ocorra algum problema no desenvolvimento o desenvolvedor deve tentar saná-lo por si só, caso não consiga, ficará a cargo do DevOps resolver o problema e garantir a continuidade do projeto. Se o problema persistir e estiver relacionado a integração do time, o Scrum Master será acionado e tentará solucionar o problema. Caso seja um problema de requisitos ou visão do produto, será acionado o Product Owner para reavaliar os requisitos e talvez fazer uma nova reunião com o cliente. Mas caso o problema seja em relação à matéria ou orientação de projeto, o professor será acionado como última instância.

6.5 GERENCIAMENTO DE RISCO

ID	Descrição	Ação
RS01	Dificuldades da equipe com as tecnologias inseridas	Realização de dojô técnico durante o desenvolvimento do projeto para a melhor performance
RS02	Erros durante o planejamento das atividades	Utilização de ferramentas de fluxo de trabalho para planejar
RS03	Atividades não entregues na Sprint	Revisar tarefas e metas no planejamento da Sprint

ID	Descrição	Ação
RS04	Má elicitação dos requisitos	Uso de estratégias de elicitação de requisitos que já estão consolidados, como rich picture e brainstorming
RS05	Falta de alinhamento entre a equipe	Encontros através de dailies e documentação dos ocorridos em uma plataforma de compartilhamento como Github
RS06	Conflito com outras atividades acadêmicas	Uso de planejamento de horário através de cada sprint, para o melhor aproveitamento das horas para a matéria
RS07	Desistência da disciplina	Incentivar a participação dos membros e a união do time. Redistribuir tarefas e refazer parte do planejamento do projeto
RS08	Problemas pessoais ou de saúde relacionados a pandemia da COVID-19	Seguir as recomendações de prevenção da Organização Mundial da Saúde e remanejamento das atividades e apoio ao(s) membro(s) afetado(s)
RS09	Problemas com a estação de trabalho	Remanejamento das atividades e revisão do cronograma
RS10	Alteração do escopo	Refinar constantemente os requisitos e manter as funcionalidades atualizadas e redefinir o escopo e redistribuir tarefas
RS11	Divergência de horários entre membros da equipe	Planejar os horários e reuniões baseado na planilha de horários
RS12	Falta de interação entre a equipe	Chamar os membros para alguma atividade divertida fora do contexto da disciplina

6.6 CRITÉRIOS DE REPLANEJAMENTO

Ter bem definido os critérios de replanejamento assegurará uma boa entrega do produto, levantamos 3 que serão cruciais no projeto.

- Mudança de visão do projeto: Ao debater sobre uma funcionalidade, ou conversar com o cliente e entender melhor suas necessidades, podem haver novas perspectivas sobre o que o projeto deveria ser, havendo assim, a necessidade de replanejar e realinhar a visão do projeto;
- Mudança de escopo do produto: Baseado numa mudança de visão, ou por um curto espaço de tempo hábil para o desenvolvimento, talvez se mostre necessário replanejar que será entregue;

- Poucas entregas: Ao final de cada sprint será feita uma reunião de revisão, pode-se notar uma diferença grande entre o planejado e o entregue. Isso pede que no planejamento da próxima Sprint seja levantado o que deu errado e o que pode ser feito para que as entregas da Sprint passadas sejam feitas e que as novas sejam bem desenvolvidas, dentro do prazo e do planejamento.

7. LIÇÕES APRENDIDAS

A falta de comunicação devido a poucas reuniões nos levaram a ter visões diferentes do projeto, com o passar disciplina, aumentar a frequência das reuniões e, principalmente, fazer uso do Lean inception, conseguimos alinhar bem melhor as ideias. Além disso, os conhecimentos relacionados a requisitos foram sendo solidificados e lapidados, fazendo com que cada vez mais tivéssemos uma ideia clara do projeto, conseguindo assim, levantar de forma bem mais clara e coerente os requisitos para o desenvolvimento das entregas de cada unidade e do produto final. Todo o fluxo de trabalho implementado pelo SAFe auxiliou e muito para o desenvolvimento robusto, descritivo e detalhado do que foi escolhido para ser implementado.

8. REFERÊNCIAS

- Schwaber, K. & Beedle, M. (2002). Agile software development with Scrum. New Jersey: Prentice Hall;
- Ambler, S. (2004). Modelagem Ágil: práticas eficazes para a Programação eXtrema e o Processo Unificado. Porto Alegre: Bookman;
- Beck, K. (2004). Programação eXtrema (XP) explicada: acolha as mudanças. Porto Alegre: Bookman.