

Documentação de um Produto de Software

TEMIFICADOR

Nome dos Alunos:

Brenda Alves Ferreira - RA: 820141463

Diego Assis Grzebieniak Beccari Borges - RA: 820140003

Julio Dantas Araujo - RA: 820146367

Lucas Mendonça Santos - RA: 819222510

Yuri Rafael Micheloni Rodrigues Viana - RA 820110205

Anderson Sousa Oliveira - RA:819225279

2020

1. Introdução

O presente documento tem por objetivo apresentar funcionalmente um projeto pensado, por nosso grupo, que ajuda a gerar Temas para projetos utilizando Inteligência Artificial.

Estávamos reunidos a pensar um tema para o projeto e encontramos grande dificuldade, e aí surgiu a ideia de acabar com essa dificuldade, criar um software que gera/busca Temas para projetos acadêmicos.

1.1. Temificador

O Temificador é um projeto que tem por objetivo gerar, automaticamente, um tema aleatório para cada grupo de projeto acadêmico, de acordo com seu curso, reduzindo o tempo que os alunos teriam para ter uma ideia.

1.2. Objetivos a serem alcançados

O Temificador tem como objetivo economizar tempo e auxiliar pessoas, principalmente alunos, que tem algum bloqueio ou dificuldade para a escolha de um tema, dando opções aleatórias de áreas determinadas pelo usuário.

1.3. Escopo principal

O Temificador é um software web que será desenvolvido para alunos, com foco nos cursos superiores de primeira formação, porém podendo ser utilizado por todos.

O software será grátis para todos, sendo monetizado por meio de anúncios e exigindo a criação de uma conta para uso.

Para criar a conta deverá obrigatoriamente ser inserido pelo usuário o nome completo, data de nascimento, curso, matérias que terá ao longo do curso, instituição de ensino, semestre atual (pedindo uma atualização do usuário semestral após a criação da conta), CPF, e-mail e telefone. Alunos abaixo de 18 anos será realizado um termo no qual o responsável terá que aceitá-lo, responsabilizando-o pelas ações do usuário. Validadores serão utilizados para conferir os dados inseridos. Os usuários deverão aceitar os termos de uso, protegendo tanto o usuário quanto o software.

Na geração dos temas, utilizaremos de inteligência artificial juntamente com um Big Data, que terá seus dados alimentados pelos cursos e matérias inseridos pelos usuários, assim como os robôs que realizarão pesquisas sobre os mesmos de forma global. A escolha do tema poderá ser feita de maneira simples pelo usuário, selecionando um curso ou poderá também utilizar filtros avançados, determinado qual matéria específica daquele curso quer que seja gerado o tema. Caso o tema gerado não agrada o usuário, ele pode solicitar que seja gerado um novo tema, tanto para gerar ou para trocar um tema será feito por meio de botões.

Não será possível que ocorra duplicidade de temas com alunos que sejam da mesma instituição, curso, semestre e matéria, mostrando para o usuário que há temas já em uso desses mesmos filtros já realizados anteriormente por outro aluno das especificações (instituição, curso, semestre e matéria).

Devido a inteligência artificial, aprendendo com projetos públicos sobre determinados temas, será possível também ter um orçamento sobre o projeto, não com 100% de exatidão, mas próximo da realidade atual.

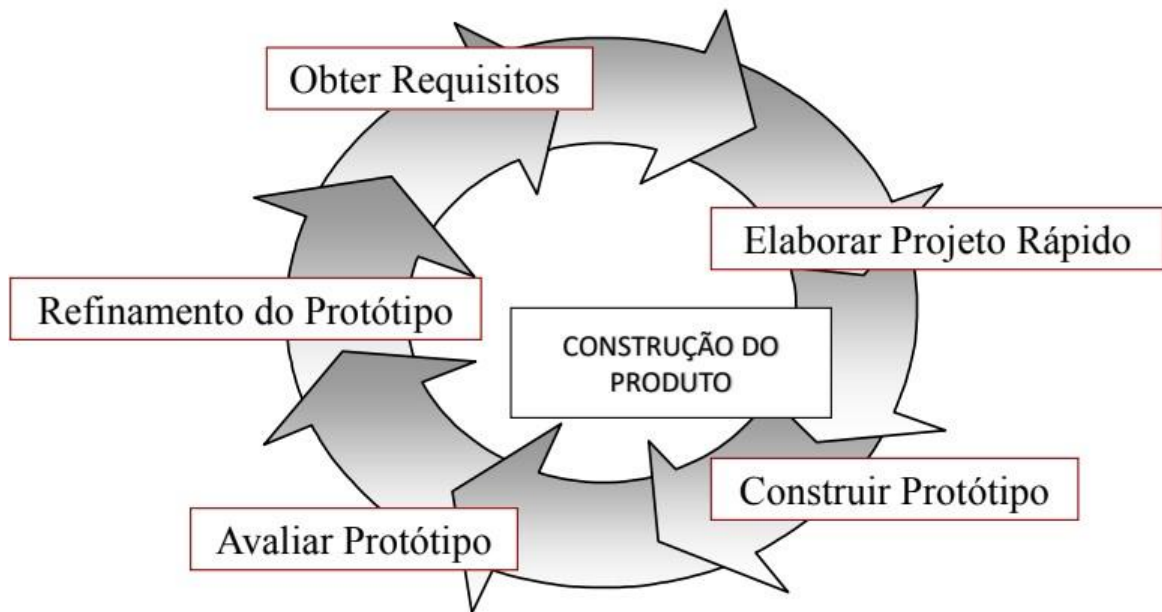
Com o uso do software, uma interface será gerada, podendo ser customizável pelo usuário e já mostrando pesquisas realizadas anteriormente, assim como os temas.

O suporte inicialmente será realizado pela equipe responsável do projeto, auxiliando os usuários novos com todas as dúvidas, via e-mail e chat. Futuramente, serão selecionados os usuários mais ativos e que tenham um bom conhecimento do software para que sejam do suporte, ganhando algumas regalias como por exemplo, prioridade em temas, brinds e retirada dos anúncios.

A segurança do software se dará as políticas de privacidade e cookies, aceitas pelos usuários na criação de sua conta. Respeitando as especificações da ABNT e da nova lei, LGPD.

2. Definição do Modelo de Processo

O modelo de processo que utilizaremos será o Modelo de Prototipação, que consiste em:



O processo de **obtenção requisitos** consiste na obtenção do material e conhecimento necessário para a **elaborar projeto rápido**, que consiste na montagem rápida de um modelo do nosso projeto.

Após a elaboração do modelo, será construído um **protótipo** que será **avaliado e aprimorado**.

Achamos o Modelo de Prototipação mais compatível, pois, com exceção da Inteligência Artificial, o nosso projeto consiste de fases menos complexas.

3. Requisitos do Sistema de Software

3.1. Requisitos Funcionais

Requisitos Funcionais.

Usuário:

Criar conta, estabelecendo uma senha de acesso criada pelo próprio usuário.

Armazenar os dados inseridos pelo cliente ao criar a conta num banco de dados (nome, sobrenome, e-mail, CPF, curso, instituição e materias do curso).

Editar seus dados citados anteriormente, com exceção do CPF.

Pesquisar tema sobre seu curso.

Adicionar filtros avançados a sua pesquisa.

Gera tema.

Alterar tema.

Verificar temas já escolhidos anteriormente.

Armazenar o tema no banco de dados.

Administrador.

Verificar todos os usuários.

Verificar temas gerados.

Editar e alterar contas.

Não verá a senha nem o CPF do cliente, estarão criptografados.

3.2. Requisitos Não-Funcionais

Requisitos de produtos:

Desempenho: o software efetuará em poucos segundos suas requisições.

Manutenabilidade: modificações e correções serão implementadas em menos de 24 horas.

Escalabilidade: o software conseguirá se adaptar as situações, caso tenha alta ou baixa demanda, evitando gastos desnecessários e utilizando apenas o necessário.

Disponibilidade: o software ficará disponível 24 horas por dia pelo menos 99% das vezes, isso não será respeitado apenas se ocorrer problemas ou alterações grandes.

Usabilidade (facilidade de uso): um usuário conseguirá utilizar o software em no máximo 30 minutos, isso auto-didata, caso tenha alguém com conhecimento sobre, esse tempo diminui pela metade. Os administradores demoram um pouco mais de tempo, cerca de 45 minutos, pois mexerá com possíveis dados sensíveis de usuários.

Portabilidade: o software deverá ser executado no Google Chrome, não tendo a totalidade das suas funções em outros navegadores, porém também podendo ser utilizado em outro navegador.

Flexibilidade: um programador com experiência mínima de desenvolvedor júnior conseguirá realizar ajustes no software caso haja mudanças na equipe.

Tolerância a falha: caso haja alguma falha grave com a IA de geração de temas, utilizaremos o banco de dados e atuaremos o mais rápido possível.

Testabilidade: os testes primeiramente serão realizados manualmente, a medida de que forem aprovados os protótipos, serão passados para automatizados.

Requisitos organizacionais.

Entrega: serão realizados diversos protótipos e serão apresentados ao cliente a cada 15 dias, junto com toda sua documentação até o momento.

Implementação: o sistema deverá ser desenvolvido na linguagem NodeJS

Requisitos externos.

Interoperabilidade: o sistema deverá se comunicar com o banco MongoDB.

Éticos: o sistema não apresentará aos usuários quaisquer dados de cunho privativo. Requisitos legais: o sistema deverá atender às normas legais, tais como padrões, leis, etc.

Integridade/segurança: apenas usuários administradores poderão verificar as contas dos usuários, desde seus dados até as pesquisas realizadas.

4. Projeto

4.1. Arquitetura Lógica

Arquitetura de duto e filtro.

Vantagens:

- Útil para aplicações de processamento de informação que interagem pouco com usuários;
- Rápida prototipação;
- Apoia reuso de transformações (filtros)
- É fácil adicionar, recombinar, ou trocar, novas transformações (flexibilidade);
- É relativamente simples implementar como sistema concorrente (vários filtros em paralelo) ou sequencial;
- Eficiência em processamento

As vantagens de utilizar o modelo arquitetural de Pipes e Filtros é que estes podem ser vistos como caixas pretas, onde há encapsulamento, alta coesão, recombinação e reuso dos dados. Isso implica afirmar que esse estilo tem suporte à reutilização. Além disso, os filtros interagem com os outros componentes de forma limitada - o que leva a um baixo acoplamento. O mais relevante é que o sistema pode ser facilmente estendido e modificado, facilitando a implementação em processadores paralelos ou em múltiplas threads em processadores simples.

A linguagem utilizada para a realização do Software será NodeJS. Além de ser uma plataforma para construir aplicações WEB escaláveis, reduzindo custos com infraestrutura e aumentar o desempenho do projeto. Ele usa uma arquitetura voltada a eventos, o que se integra muito bem com JavaScript. Usando um loop de eventos, o Node interpreta, em uma única thread, as requisições de forma assíncrona em vez de sequenciais, e não permitindo bloqueios. Isso o torna incrivelmente rápido... perfeito para lidar com um número muito alto de requisições.

MongoDB é um banco de dados orientado direcionado a Documentos, a sua característica principal é ser muito simples e conseguirá conter as informações mais importantes em um mesmo documento, e sendo também totalmente livre de esquemas, ou seja, é muito mais fácil, o MongoDB tem características ser um código-fonte que aberto e licenciado pela Affero General Public License (AGPL);

Outro motivo de utilizarmos o MongoDB seria por também ser capaz de fazer queries velozes e fornecer informações solicitadas com baixa latência, o MongoDB é bastante recomendado para aplicações web e mobile, com muitas requisições simultâneas.

4.2. Arquitetura Física

O temificador pode ser acessado por quaisquer sistemas operacionais de celulares e computadores, é preciso ter a versão mais recente do navegador, além de uma boa conexão de Internet.

5. Protótipo de Interface

Neste item deve ser apresentado o protótipo do projeto. O protótipo é um recurso que deve ser adotado como estratégia para levantamento, detalhamento, validação de requisitos e modelagem de interface com o usuário (usabilidade).

Referência: UC Modelos, métodos e técnicas da engenharia de software

6. Critérios de Qualidade de Software

Neste item devem ser listados e descritos os critérios de garantia da qualidade do processo que serão considerados no Projeto.

Referência: UC Gestão e Qualidade de software

7. Testes

7.1. Plano de Testes

Neste item deve ser criado o plano de testes do sistema, permitindo a validação do sistema por parte do desenvolvedor, através da verificação dos requisitos do sistema desenvolvido. **Referência:** UC *Gestão e Qualidade de software*

7.2. Roteiro de Testes

Neste item devem ser registrados os testes realizados no sistema tendo como base o Plano de Testes do Sistema. O roteiro de testes deve ser elaborado com base nos casos de uso ou user stories (cartão de histórias) de forma manual ou automatizada.

Referência: UC *Gestão e Qualidade de software*

Anexo I

Neste item deve ser anexado o roteiro de entrevista ou questionário respondido, caso tenha sido aplicado.

Referência: UC Modelos, métodos e técnicas da engenharia de software