|  |  |
| --- | --- |
| CURSO DE CIÊNCIA DA COMPUTAÇÃO – TCC | |
| (  x  ) PRÉ-PROJETO     (     ) PROJETO | ANO/SEMESTRE: 2023/1 |

ASSISTENTE DE ATUALIZAÇÃO DE PROJETOS ANGULAR

Nathan Reikdal Cervieri

Marcel Hugo

# Introdução

[A introdução deve despertar no leitor o interesse pelo texto, apresentando os assuntos que serão tratados e o enfoque que será dado ao tema central. Deve iniciar com uma **contextualização** do estudo a ser realizado, explicando claramente sua origem/motivação. Deve finalizar com a **formulação do problema** (pergunta de pesquisa) a ser investigado.

O tema da pesquisa deve ser abordado de forma clara e sucinta, identificando a situação ou o contexto no qual o problema está inserido. A visão geral do tema deve então ser afunilada até se chegar ao problema a ser pesquisado. Após o problema ter sido identificado, deve-se delimitar que aspectos ou elementos serão tratados. Em resumo, na introdução deve-se deixar bem claro o problema que se quer resolver com o desenvolvimento do trabalho.

**O pré-projeto deve ter no máximo 10 (dez) páginas e o projeto deve ter no máximo 12 (doze) páginas.**

## OBJETIVOS

O objetivo é simplificar o processo de atualização entre versões do framework Angular para diminuir o tempo e trabalho necessário através da automatização da análise do projeto e alterações a serem feitas para adequar à nova versão.

Os objetivos específicos são:

1. analisar projetos angular;
2. apontar mudanças necessárias para atualização;
3. Realizar mudanças encontradas automaticamente;

# trabalhos correlatos

[Devem ser descritos os TRABALHOS CORRELATOS identificados, cada um em uma seção específica. Deve-se apresentar as **características** destes trabalhos, incluindo principais funcionalidades, pontos fortes e fracos, bem como resultados descritos pelo autor, entre outros elementos que permitam compreender o trabalho correlato e sua relação com o problema formulado na introdução.

Observa-se que, antes da primeira seção, deve-se descrever o que o leitor vai encontrar nesse capítulo (preâmbulo), ou seja, quais trabalhos correlatos são descritos. Não devem ser descritos os títulos dos trabalhos tal como aparecem nos documentos consultados, mas o tipo de trabalho com a devida citação ao(s) autor(es) do mesmo, como no exemplo, a seguir:]

São apresentados trabalhos com características semelhantes aos principais objetivos do estudo proposto. O primeiro é uma linguagem de programação conhecida por Agda (NORELL, 2007), o segundo é o assistente de provas Coq (THE COQ DEVELOPMENT TEAM, 2014) e o terceiro é um interpretador para uma linguagem com tipos dependentes (LÖH; MCBRIDE; SWIERSTRA, 2010).

[Quando o projeto propuser uma continuação ou extensão de um Trabalho de Conclusão de Curso (TCC) anterior, através de correções ou adaptações, o trabalho anterior não deve ser considerado um trabalho correlato. Deve-se descrever o trabalho anterior em um novo capítulo (capítulo 3), incluindo a relação dos requisitos do trabalho atual. Sugere-se que o título desse capítulo seja: “SOFTWARE (ou FERRAMENTA, ou PROTÓTIPO, ou BIBLIOTECA) ATUAL”.]

Abaixo segue um exemplo de Figura (Figura 1). Observa-se que as figuras devem ter moldura e a legenda e fonte devem estar centralizadas. A referência no texto da figura ou quadro deve ser antes da sua inserção.

Figura 1 – Exemplo da aplicação



Fonte: Lyu *et al*. (2017).

## DOTNET UPGRADE ASSISTANT

O Upgrade Assistant do .NET é uma ferramenta fornecida pela Microsoft que ajuda desenvolvedores a atualizar suas aplicações .NET para versões mais recentes do framework .NET. Ele automatiza o processo de identificação de problemas potenciais e fornece recomendações sobre como corrigi-los. Esta ferramenta tanto para aplicações web quanto para aplicações desktop e suporta a atualização de aplicações construídas usando o .NET Framework ou o .NET Core.

O Upgrade Assistant analisa o código da aplicação e gera um relatório que detalha as mudanças necessárias para atualizar para a versão mais recente do .NET. Ele detecta APIs obsoletas ou removidas e sugere APIs atualizadas para substituí-las. A ferramenta pode modificar o código aplicando refatoração de código automatizada ou fornece recomendações para o desenvolvedor sobre como modificar o código manualmente.

O Upgrade Assistant pode atualizar os pacotes NuGet para suas últimas versões compatíveis com o framework de destino. Ele pode converter arquivos de configuração para o novo formato usado em versões mais recentes do .NET. Além disso, pode modificar o código para usar novos recursos e APIs introduzidos no framework de destino. A ferramenta fornece recomendações sobre as melhores práticas para mudanças e melhorias no código.

O Upgrade Assistant fornece um modo interativo que permite aos desenvolvedores revisar e modificar as alterações sugeridas antes de aplicá-las ao código. Isso dá ao desenvolvedor total controle sobre as modificações no código. Ele também fornece documentação detalhada que inclui informações sobre como instalar e usar a ferramenta.

O Upgrade Assistant do .NET não é uma ferramenta totalmente automatizada, e alguma intervenção manual pode ser necessária para modificar o código corretamente. No entanto, ele simplifica significativamente o processo de atualização, automatizando muitas das tarefas tediosas e propensas a erros envolvidas na atualização de uma aplicação para uma nova versão do .NET. Esta ferramenta ajuda a simplificar e acelerar o processo de atualização de aplicações .NET, reduzindo o tempo e o esforço necessários para os desenvolvedores migrarem para versões mais recentes do framework.

Em resumo, o Upgrade Assistant do .NET é uma ferramenta poderosa que ajuda os desenvolvedores a atualizar suas aplicações .NET para versões mais recentes do framework .NET. Ele automatiza muitas das tarefas tediosas e propensas a erros envolvidas no processo de atualização, fornecendo recomendações sobre como corrigir problemas potenciais. Ele pode modificar o código usando refatoração de código automatizada, atualizar pacotes NuGet, converter arquivos de configuração e modificar o código para usar recursos e APIs mais recentes. A ferramenta fornece um modo interativo que permite aos desenvolvedores revisar e modificar as alterações sugeridas antes de aplicá-las ao código, dando-lhes total controle sobre as modificações no código.

## SEGUNDO TRABALHO CORRELATO

...

## n-ésimo TRABALHO CORRELATO

...

# proposta DA FERRAMENTA

[O título “PROPOSTA” deve ser complementado com “DO SOFTWARE”, “DA FERRAMENTA”, “DO PROTÓTIPO”, “DA BIBLIOTECA” ou de outro texto que caracterize o objeto do estudo. Esse capítulo deve descrever a justificativa para o desenvolvimento do estudo proposto, os requisitos principais que serão trabalhados e a metodologia de desenvolvimento que será seguida. Observa-se que, antes da primeira seção, deve-se descrever o que o leitor vai encontrar nesse capítulo (preâmbulo).]

Será feito neste capítulo uma análise dos trabalhos correlatos e como eles se relacionam com a proposta definida nos objetivos.

## JUSTIFICATIVA

[A pergunta essencial a ser respondida nessa seção é **por que** este estudo será feito. Para tanto, deve-se:

1. apresentar um quadro relacionando os trabalhos correlatos descritos no capítulo anterior e suas principais características / funcionalidades;
2. discutir textualmente o quadro apresentado;
3. relacionar e justificar os argumentos que determinam que a proposta é significativa ou importante, isto é, que não é algo trivial ou corriqueiro. Os argumentos podem ser científicos (em que o estudo melhora o conhecimento sobre o tema) ou metodológicos/técnicos (por que a metodologia ou as técnicas a serem utilizadas são essenciais para o contexto do estudo), ou ambos;
4. apresentar as contribuições que o estudo pode proporcionar. As contribuições podem ser teóricas (como o estudo pode avançar a teoria sobre o tema) ou práticas/sociais (como o estudo pode melhorar os elementos do contexto ao qual será aplicado) ou ambas.]

Quadro 1 - Comparativo dos trabalhos correlatos

|  |  |  |  |
| --- | --- | --- | --- |
| Trabalhos Correlatos  Características | Correlato 01 | Correlato 02 | Correlato 03 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Fonte: elaborado pelo autor.

Ao buscar na internet, quando se vê a análise de um sistema legado, muitas vezes as fontes recomendam reescrever o sistema do início, a opção de atualizar o framework que estava sendo utilizado não é nem considerado como uma opção válida devido a grande dificuldade de lidar com as alterações que vem de anos de má manutenção em projetos.

Uma ferramenta como a que estamos propondo tem a possibilidade de semanas de desenvolvimento e retrabalho de migrar código para uma plataforma nova, dando vida nova a um sistema já esquecido e muito funcional.

## REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

* Permitir incluir projeto para análise (RF)
* Permitir análise de arquivos singulares (RF)
* Selecionar a versão de origem e versão objetivo (RF)
* Criar aplicação em angular cli para interface de usuário (RNF)
* Criar servidor C# para analise e subsituição (RNF)
* TODO: Decidir ferramenta para realizar análise léxica e sintática (RNF)
* TODO: Definir maneira de substituição (RNF)
* Disponibilizar projeto online (RNF)

## METODOLOGIA

[A metodologia refere-se à descrição dos procedimentos, métodos e recursos a serem utilizados no decorrer do trabalho. Podem ser arroladas tantas etapas quantas forem necessárias, tais como reavaliação de requisitos, especificação, projeto do sistema, implementação, testes, validação, entre outras. Observa-se que cada etapa deve ser descrita detalhadamente, incluindo os métodos e ferramentas a serem usados, conforme o caso.]

O trabalho será desenvolvido observando as seguintes etapas:

1. nome da etapa 01: descrever as atividades a serem realizadas, incluindo (quando for o caso) métodos e ferramentas a serem usados;
2. nome da etapa 02: descrever as atividades a serem realizadas, incluindo (quando for o caso) métodos e ferramentas a serem usados;
3. (...);
4. nome da etapa n: descrever as atividades a serem realizadas, incluindo (quando for o caso) métodos e ferramentas a serem usados.

[Para cada uma das etapas listadas na metodologia deve-se especificar o período necessário para a sua realização, lembrando que algumas delas são desempenhadas simultaneamente. Distribua as etapas num cronograma, conforme exemplo abaixo.]

As etapas serão realizadas nos períodos relacionados no Quadro 2.

Quadro 2 - Cronograma

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | ano | | | | | | | | | |
|  | mês. | | mês. | | mês. | | mês. | | mês. | |
| etapas / quinzenas | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 |
| nome da etapa 01 |  |  |  |  |  |  |  |  |  |  |
| nome da etapa 02 |  |  |  |  |  |  |  |  |  |  |
| ... |  |  |  |  |  |  |  |  |  |  |
| nome da etapa n |  |  |  |  |  |  |  |  |  |  |

Fonte: elaborado pelo autor.

# REVISÃO BIBLIOGRÁFICA

[No pré-projeto devem ser descritos brevemente os assuntos que fundamentarão o estudo a ser realizado, relacionando a(s) principal(is) referência(s) bibliográfica(s), a(s) qual(is) deve(m) constar nas REFERÊNCIAS. Cada assunto abordado deve ser descrito em um parágrafo.

No projeto deve ser apresentado estudo inicial sobre o tema escolhido, detalhando cada parágrafo, na forma de seções, os assuntos relacionados no pré-projeto. A revisão bibliográfica consiste na sistematização de ideias e fundamentos de autores que dão sustentação ao assunto estudado. Observa-se que, antes da primeira seção, deve-se descrever o que o leitor vai encontrar nesse capítulo (preâmbulo), ou seja, como a revisão bibliográfica está organizada.]

* Javascript
* Angular
* Desenvolvimento web
* Sistema legado

## Título da 1ª seção [inserir somente no projeto]

...

## Título da 2ª seção [inserir somente no projeto]

...

Referências

[Só podem ser inseridas nas referências os documentos citados no projeto. Todos os documentos citados obrigatoriamente tem que estar inserido nas referências.

As referências deverão ser apresentadas em ordem alfabética, de acordo com as normas da ABNT. Como padrão, o nome do autor deve ser apresentado da seguinte forma: sobrenome com todas as letras maiúsculas; primeiro nome por extenso com a primeira letra maiúscula e as demais em minúscula; os outros nomes abreviados (letra em maiúscula seguida de ponto).]

[parte de um documento:]

AMADO, Gilles. Coesão organizacional e ilusão coletiva. In: MOTTA, Fernando C. P.; FREITAS, Maria E. (Org.). **Vida psíquica e organização**. Rio de Janeiro: FGV, 2000. p. 103-115.

[trabalho acadêmico ou monografia (TCC/Estágio, especialização, dissertação, tese):]

AMBONI, Narcisa F. **Estratégias organizacionais**: um estudo de multicasos em sistemas universitários federais das capitais da região sul do país. 1995. 143 f. Dissertação (Mestrado em Administração) - Curso de Pós-Graduação em Administração, Universidade Federal de Santa Catarina, Florianópolis.

[norma técnica:]

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6023**: informação e documentação: referências - elaboração. Rio de Janeiro, 2002a. 24 p.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 10520**: informação e documentação: citações em documentos: apresentação. Rio de Janeiro, 2002b. 7 p.

[livro:]

BASTOS, Lília R.; PAIXÃO, Lyra; FERNANDES, Lúcia M. **Manual para a elaboração de projetos e relatórios de pesquisa, teses e dissertações**. Rio de Janeiro: Zahar, 1979.

[trabalho acadêmico ou monografia (TCC/Estágio, especialização, dissertação, tese):]

BRUXEL, Jorge L. **Definição de um interpretador para a linguagem Portugol, utilizando gramática de atributos**. 1996. 77 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

[verbete de enciclopédia em meio eletrônico:]

EDITORES gráficos. In: WIKIPEDIA, a enciclopédia livre. [S.l.]: Wikimedia Foundation, 2006. Disponível em: http://pt.wikipedia.org/wiki/Editores\_graficos. Acesso em: 13 maio 2006.

[artigo em evento:]

FRALEIGH, Arnold. The Algerian of independence. In: ANNUAL MEETING OF THE AMERICAN SOCIETY OF INTERNATIONAL LAW, 61, 1967, Washington. **Proceedings…** Washington: Society of International Law, 1967. p. 6-12.

[norma técnica:]

IBGE. **Normas para apresentação tabular**. 3. ed. Rio de Janeiro, 1993. 61 p. Disponível em: http://biblioteca.ibge.gov.br/visualizacao/monografias/GEBIS%20-%20RJ/normastabular.pdf. Acesso em: 27 ago. 2013.

[artigo em periódico:]

KNUTH, Donald E. Semantic of context-free languages. **Mathematical Systems Theory**, New York, v. 2, n. 2, p. 33-50, jan./mar. 1968.

[trabalho acadêmico ou monografia (TCC/Estágio, especialização, dissertação, tese):]

SCHUBERT, Lucas A. **Aplicativo para controle de ferrovia utilizando processamento em tempo real e redes de Petri**. 2003. 76 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

[página da internet com autor]

SCHULER, João P. S. **Tutorial de Delphi**. Porto Alegre, [2002]. Disponível em: http://www.schulers.com/jpss/pascal/dtut/. Acesso em: 27 ago. 2013.

[página da internet sem autor]

SCHRATCH. **Program, imagine, share**. [S.l.], [2013?]. Disponível em: <https://scratch.mit.edu/>. Acesso em: 27 maio 2013.

[relatório de pesquisa:]

VARGAS, Douglas N. **Editor dirigido por sintaxe**. 1992. Relatório de pesquisa n. 240 arquivado na Pró-Reitoria de Pesquisa, Universidade Regional de Blumenau, Blumenau.