**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA**

**PAULA SOUZA**

**ETEC PROF. BASILIDES DE GODOY**

**M-TEC Desenvolvimento de Sistemas**

**João Paulo Amaral de Araujo**

**João Vitor dos Santos Nunes**

**Regular Expressions e Data Annotations**

**São Paulo**

**2025**

**João Paulo Amaral de Araujo**

**João Vitor dos Santos Nunes**

**Regular Expressions e Data Annotations**

Trabalho solicitado pela Etec Prof. Basilides de Godoy no 2ºM-tec Desenvolvimento de Sistemas como requisito parcial para aprovação no componente curricular de Desenvolvimento de Sistemas

Prof. Orientador: Nivia Maria Domingues

**São Paulo**

**2025**

*Dedico este trabalho à minha família, pelo apoio constante em minha jornada.*

**AGRADECIMENTOS**

A todos aqueles que contribuíram, de alguma forma, para a realização deste trabalho.

*"O verdadeiro conhecimento está em sabermos que nada sabemos."*

*Sócrates*

**RESUMO**

Este trabalho tem como objetivo apresentar de forma clara os conceitos de Regular Expressions (expressões regulares) e Data Annotations (anotações de dados), abordando suas aplicações, principais formatos utilizados, linguagens de programação compatíveis, diferenças entre frameworks e possíveis implicações no uso com bancos de dados. Inicialmente, são discutidas as expressões regulares, seguidas pelas anotações de dados, conforme a organização exigida. A pesquisa visa esclarecer as funcionalidades dessas ferramentas no desenvolvimento de software.

Palavras-chave: Regular Expression, Data Annotations, Frameworks.

**ABSTRACT**

This paper aims to clearly present the concepts of Regular Expressions and Data Annotations, addressing their applications, commonly used formats, supported programming languages, differences among frameworks, and possible implications when used with databases. Regular Expressions are discussed first, followed by Data Annotations, according to the required structure. The research seeks to clarify the functionalities of these tools in software development.

Keywords: Regular Expressions, Data Annotations, Frameworks.

**SUMÁRIO**

**1 INTRODUÇÃO.....................................................................................................7**

**2 REGULAR EXPRESSION E DATA ANNOTATION...........................................8**

**2.1 O que é Regular Expression?.......................................................................8**

**2.1.1 Como é utilizado?........................................................................................8**

**2.1.2 Principais expressões usadas...................................................................8**

**2.2 O que é Data annotation................................................................................9**

**2.2.1 Como é utilizado?.......................................................................................9**

**2.2.2 Principais annotations usadas..................................................................9**

**2.3 Linguagens que fazem uso dos dois...........................................................9**

**2.4 Diferença entre Regular Expression e Data Annotation...........................10**

**2.5 Uso com banco de dados............................................................................10**

**3 CONCLUSÃO....................................................................................................11**

**REFERÊNCIAS......................................................................................................**

**1 INTRODUÇÃO**

O desenvolvimento de sistemas modernos exige mecanismos robustos para garantir a integridade, segurança e organização dos dados. Entre as ferramentas mais relevantes nesse contexto, destacam-se as Regular Expressions e as Data Annotations. As Regular Expressions oferecem uma forma poderosa e flexível de identificar e manipular padrões em cadeias de texto, sendo muito utilizadas em linguagens como JavaScript, Python e C#. Já as Data Annotations, permitem a configuração direta de regras e validações sobre classes e propriedades, especialmente em ambientes .NET. Mais pra frente, exploraremos mais a fundo essas duas abordagens, analisando suas estruturas, formas de utilização, principais aplicações, diferenças entre linguagens e frameworks, além de analisar sua utilização com bancos de dados.

**2 REGULAR EXPRESSION E DATA ANNOTATION**

**2.1 O que é Regular Expression?**

A Regular Expression, também chamada de expressão regular, é uma forma de representar padrões de texto por meio de uma sintaxe própria. Essa técnica é usada para identificar, buscar, substituir ou validar sequências de caracteres dentro de uma string. Em vez de analisar caractere por caractere de maneira tradicional, o uso de uma expressão regular permite aplicar uma regra complexa de forma mais simples e eficiente. Por exemplo, é comum utilizar regex para verificar se um campo de e-mail está no formato correto, se um CPF tem os 11 dígitos, ou se uma senha atende a requisitos de segurança como letras maiúsculas, números e caracteres especiais.

**2.1.1 Como é utilizado?**

As expressões regulares são muito comuns em formulários e aplicações web para validar dados de entrada do usuário, como login, senhas e formulários de contato. Elas também podem ser usadas para filtrar dados em sistemas, localizar trechos de texto em documentos, extrair informações de arquivos, entre outras aplicações. Em programas de busca e edição de texto, como editores de código ou planilhas, é possível usar regex para fazer buscas mais avançadas.

**2.1.2 Principais expressões usadas**

“.” representa qualquer caractere

“\*” repete o caractere anterior zero ou mais vezes

“+” repete o caractere anterior uma ou mais vezes

“?” torna o caractere anterior opcional

“\d” representa um dígito de 0 a 9

“[abc]” representa a ocorrência de “a”, “b” ou “c”

“^” indica o início da string

“$” indica o final da string

**2.2 O que é Data Annotation**

Data Annotation é um recurso usado principalmente em linguagens como C# (no framework .NET) para adicionar regras e instruções diretamente em atributos de uma classe. Essas anotações são colocadas entre colchetes e têm a função de definir validações, limites ou comportamentos para os dados que serão inseridos, processados ou armazenados. Por exemplo, ao utilizar [Required], o programador define que aquele campo é obrigatório, e com [StringLength(50)], que a string deve ter no máximo 50 caracteres.

**2.2.1 Como são utilizadas**

Elas são aplicadas diretamente no modelo de dados da aplicação, normalmente em sistemas que utilizam frameworks como Entity Framework, ASP.NET MVC ou Blazor. Com isso, evita-se a necessidade de criar funções de validação separadas, pois o próprio sistema interpreta essas anotações e aplica as regras automaticamente. Isso facilita a manutenção do sistema, pois as regras ficam centralizadas no código do modelo.

**2.2.2 Principais annotations usadas**

Entre os mais comuns estão [Required], [StringLength], [Range], [EmailAddress], [Key] e [RegularExpression]. Este último permite incluir uma expressão regular como parte da validação dentro da anotação.

**2.3 Linguagens que fazem uso dos dois**

Expressões regulares são usadas em diversas linguagens como Python, JavaScript, Java, C#, PHP, Ruby e Perl. Já as data annotations são mais comuns em C# e .NET, mas também existem variações em linguagens como Java (usando anotações do Bean Validation) e frameworks como o Spring.

**2.4 Diferença entre Regular Expression e Data Annotation**

Regular expression é uma ferramenta para descrever padrões de texto. Ela serve para identificar, validar ou substituir partes de uma string. Já data annotation é uma forma de declarar regras dentro do código-fonte, normalmente associada à estrutura dos dados de uma classe. Enquanto uma expressão regular diz o que deve ser validado, a data annotation diz onde e quando essa validação deve acontecer.

**2.5 Uso com banco de dados (regular expression)**

Expressões regulares podem ser usadas diretamente em algumas consultas SQL usando o operador REGEXP, presente em bancos como MySQL e PostgreSQL. No entanto, seu uso pode causar lentidão em grandes volumes de dados, pois essas operações costumam exigir mais do sistema de busca. Além disso, a sintaxe e o nível de suporte à regex podem variar de um banco para outro. Em sistemas que usam ORMs, como o Entity Framework, é comum deixar as validações de regex no código da aplicação, antes mesmo de os dados chegarem ao banco, o que evita problemas de desempenho e aumenta a segurança do sistema.

**3 CONCLUSÃO**

As expressões regulares e as data annotations são recursos poderosos que facilitam a validação e o controle de dados em sistemas de software. Enquanto as expressões regulares são focadas em identificar padrões dentro de textos, as data annotations organizam e centralizam regras diretamente nos modelos de dados, promovendo uma estrutura mais limpa e eficiente no desenvolvimento. Ambas as ferramentas são amplamente utilizadas em linguagens e frameworks modernos, e seu uso correto contribui para a criação de aplicações mais seguras, confiáveis e de fácil manutenção. Compreender bem a diferença entre elas e saber como aplicá-las, especialmente em conjunto com bancos de dados e frameworks, é essencial para qualquer desenvolvedor que deseja construir sistemas robustos e profissionais.

REFERÊNCIAS

ALURA. *O que são expressões regulares (Regex)?* Disponível em: https://www.alura.com.br/artigos/o-que-sao-expressoes-regulares-regex. Acesso em: 10 jun. 2025.

MICROSOFT DOCS. *Data Annotations - C#*. Disponível em: https://learn.microsoft.com/en-us/dotnet/api/system.componentmodel.dataannotations. Acesso em: 5 jun. 2025.

MOZILLA DEVELOPER NETWORK. *Regular expressions - JavaScript*. Disponível em: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular\_Expressions. Acesso em: 10 jun. 2025.

POSTGRESQL DOCS. *Pattern Matching*. Disponível em: https://www.postgresql.org/docs/current/functions-matching.html. Acesso em: 7 jun. 2025.

W3SCHOOLS. *C# Data Annotations*. Disponível em: https://www.w3schools.com/asp/aspnet\_dataannotations.asp. Acesso em: 7 jun. 2025.