

Ver Detalhes da Tarefa

Nº:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
16	[Projeto Biblioteca] General	grande	sempre	2022-10-23 00:14	2022-10-23 00:14
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	normal	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			
		Previsto para a Versão:			
Resumo:	Estilização (melhoria)				
Descrição:	Foi incluída a cor vermelha para enfatizar o tempo decorrido do empréstimo				
Marcadores:					
Passos para Reproduzir:	1 - em Listaem.cshtml de "emprestimo" foi incluido o comando "@foreach(var e in Model) { @if((System.DateTime.Compare(System.DateTime.Now, e.DataDevolucao)>0)) { Classe="text-danger"; } else { Classe=""; } <tr class="@Classe"> <td>@e.Id</td> <td>@e.NomeUsuario</td> <td>@e.DataEmprestimo.ToString("dd/MM/yyyy")</td> <td>@e.DataDevolucao.ToString("dd/MM/yyyy")</td> <td>@e.Livro.Titulo</td> <td>@e.Livro.Autor</td> <td>/Emprestimo/Edicao/@e.Id</td> </tr> } </tbody>"				
Informações Adicionais:					
Arquivos Anexados:					
Não há anotações anexadas a esta tarefa.					

Ver Detalhes da Tarefa

Nº:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
15	[Projeto Biblioteca] General	pequeno	sempre	2022-10-23 00:08	2022-10-23 00:09

Relator:	administrator	Plataforma:		
Atribuído a:	administrator	SO:		
Prioridade:	baixa	Versão SO:		
Estado:	resolvido	Versão do Produto:		
Compilação do Produto:		Resolução:	aberto	
Projeção:	nenhuma			
Tempo Estimado:	nenhum	Corrigido na Versão:		
		Previsto para a Versão:		
Resumo:	Paginação (melhoria)			
Descrição:	Foi incluso funcionalidade de paginação nas listagens de empréstimo e de livros			
Marcadores:				
Passos para Reproduzir:	<p>1 - LivroController</p> <pre>2 - public IActionResult Listagem(string tipoFiltro, string filtro, string itensPorPagina, int NumDaPagina, int PaginaAtual) { Autenticacao.CheckLogin(this); FiltrosLivros objFiltro = null; if(!string.IsNullOrEmpty(filtro)) { objFiltro = new FiltrosLivros(); objFiltro.Filtro = filtro; objFiltro.TipoFiltro = tipoFiltro; } ViewData["livrosPorPagina"] = (string.IsNullOrEmpty(itensPorPagina) ? 10 : int.Parse(itensPorPagina)); ViewData["PaginaAtual"] = (PaginaAtual != 0 ? PaginaAtual : 1); LivroService livroService = new LivroService(); return View(livroService.ListarTodos(objFiltro)); }</pre> <p>3 - Além disso, ocorreram outras alterações em "Listagem.cshtml" em "Livro"</p>			
Informações Adicionais:				
Arquivos Anexados:				
Não há anotações anexadas a esta tarefa.				

Ver Detalhes da Tarefa					
Nº:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
14	[Projeto Biblioteca] General	grande	sempre	2022-10-23 00:02	2022-10-23 00:03
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	urgente	Versão SO:			

Estado:	resolvido	Versão do Produto:		
Compilação do Produto:		Resolução:	aberto	
Projeção:	nenhuma			
Tempo Estimado:	nenhum	Corrigido na Versão:		
		Previsto para a Versão:		
Resumo:	Criptografia (melhoria)			
Descrição:	Foi incluída a função de Criptografia			
Marcadores:				
Passos para Reproduzir:	1 - Controller/Autenticacao.cs 2 - Criação das public statics: "verificaLoginSenha"; "verificaSeUsuarioAdminExiste" e "verificaSeUsuarioEAdmin" 3 - Inclusão da model "Criptografo.cs"			
Informações Adicionais:				
Arquivos Anexados:				
Não há anotações anexadas a esta tarefa.				

Ver Detalhes da Tarefa

Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
13	[Projeto Biblioteca] General	grande	sempre	2022-10-22 23:57	2022-10-22 23:57
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	normal	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			
		Previsto para a Versão:			
Resumo:	Criação da tabela "Usuários" (melhoria)				
Descrição:	Uma nova tabela foi incluída				
Marcadores:					
Passos para Reproduzir:	1 - criação das models "Usuario.cs" e "UsuarioService.cs" 2 - criação do controller "UsuarioController.cs" 3 - inclusão no banco de dados				
Informações Adicionais:					
Arquivos Anexados:					

Não há anotações anexadas a esta tarefa.

Ver Detalhes da Tarefa

Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
11	[Projeto Biblioteca] General	grande	sempre	2022-10-22 23:51	2022-10-22 23:54
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	normal	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			
		Previsto para a Versão:			
Resumo:	Livro cadastrado que já foi devolvido não aparece mais para empréstimo (melhoria)				
Descrição:	É necessário não ter sido devolvido para permanecer no sistema				
Marcadores:					
Passos para Reproduzir:	1 - "EmprestimoController.cs" 2 - ir em "public IActionResult Cadastro()" 3 - trocar "listartodos" por "listardisponiveis"				
Informações Adicionais:					
Arquivos Anexados:					
Não há anotações anexadas a esta tarefa.					

Ver Detalhes da Tarefa

Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
12	[Projeto Biblioteca] General	pequeno	sempre	2022-10-22 23:54	2022-10-22 23:54
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	baixa	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			

		Previsto para a Versão:		
Resumo:	Formatação de Menu (melhoria)			
Descrição:	Possibilidade de um acesso mais dinâmico com opções na aba "livros" e a inclusão da aba "usuário"			
Marcadores:				
Passos para Reproduzir:	<p>1 - Shared - Layout</p> <p>2 - <div class="navbar-collapse collapse d-sm-inline-flex flex-sm-row-reverse"></p> <pre><ul class="navbar-nav flex-grow-1"> <li class="nav-item dropdown"> # <div class="dropdown-menu" aria-labelledby="menuLivros"> /Livro/Cadastro /Livro/Listagem </div> <li class="nav-item dropdown"> # <div class="dropdown-menu" aria-labelledby="menuLivros"> /Emprestimo/Listagem /Emprestimo/Cadastro </div> <li class="nav-item dropdown"> # <div class="dropdown-menu" aria-labelledby="menuLivros"> /Usuario/RegistrarUsuarios </div> </div> </div></pre>			
Informações Adicionais:				
Arquivos Anexados:				
Não há anotações anexadas a esta tarefa.				

Ver Detalhes da Tarefa					
Nº:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
5	[Projeto Biblioteca] General	grande	sempre	2022-10-22 22:50	2022-10-22 23:46
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	urgente	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			

		Previsto para a Versão:		
Resumo:	A atualização do empréstimo deve permitir apenas a edição do campo de livro e dos campos de data.			
Descrição:	Os campos "Nome" e "Telefone" não são editáveis			
Marcadores:				
Passos para Reproduzir:	1 - View - Emprestimo - "Edicao.cshtml" 2 - incluir readonly="readonly"			
Informações Adicionais:				
Arquivos Anexados:				
Não há anotações anexadas a esta tarefa.				

Ver Detalhes da Tarefa					
Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
7	[Projeto Biblioteca] General	grande	sempre	2022-10-22 22:51	2022-10-22 23:44
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	normal	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			
		Previsto para a Versão:			
Resumo:	"Edição de empréstimos" com o campo "Telefone" em "somente leitura"				
Descrição:	O campo não pode ser editado				
Marcadores:					
Passos para Reproduzir:	1 - View - Emprestimo - "Edicao.cshtml" 2 - incluir readonly="readonly"				
Informações Adicionais:					
Arquivos Anexados:					
Não há anotações anexadas a esta tarefa.					

Ver Detalhes da Tarefa					
Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:

6	[Projeto Biblioteca] General	grande	sempre	2022-10-22 22:51	2022-10-22 23:40
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	imediato	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			
		Previsto para a Versão:			
Resumo:	Páginas com acesso impossibilitado sem login				
Descrição:	Todas as páginas, exceto a de log in, tem acesso restrito a usuários logados.				
Marcadores:					
Passos para Reproduzir:	1 - Ir em todas as páginas e colocar o comando 2 - colocar "Autenticacao.CheckLogin(this);"				
Informações Adicionais:					
Arquivos Anexados:					
Não há anotações anexadas a esta tarefa.					

Ver Detalhes da Tarefa					
Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
1	[Projeto Biblioteca] General	pequeno	sempre	2022-10-22 22:45	2022-10-22 23:37
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	alta	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			
		Previsto para a Versão:			
Resumo:	Listagem de empréstimo com funcionalidade de filtro				
Descrição:	A listagem deve contém a funcionalidade de filtro por usuário e título do livro. Os filtros consideraram texto inteiro (título completo de um livro, por exemplo) ou textos parciais (parte do título de um livro, por exemplo), tanto para título quanto para autor.				

Marcadores:	
Passos para Reproduzir:	<pre> 1 - Models - "EmprestimoService.cs" 2 - public ICollection<Emprestimo> ListarTodos(FiltrosEmprestimos filtro = null) { using(BibliotecaContext bc = new BibliotecaContext()) { IQueryable<Emprestimo> query; if(filtro != null) { //definindo dinamicamente a filtragem switch(filtro.TipoFiltro) { case "Usuario": query = bc.Emprestimos.Where(e => e.NomeUsuario.Contains(filtro.Filtro.ToLower())); break; case "Livro": query = bc.Emprestimos.Where(e => e.Livro.Titulo.Contains(filtro.Filtro.ToLower())); break; default: query = bc.Emprestimos; break; } } else { // caso filtro não tenha sido informado query = bc.Emprestimos; } //ordenação padrão return query.Include(e => e.Livro).OrderByDescending(e => e.DataDevolucao).ToList(); } } </pre>
Informações Adicionais:	
Arquivos Anexados:	
Não há anotações anexadas a esta tarefa.	

Ver Detalhes da Tarefa					
Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
8	[Projeto Biblioteca] General	pequeno	sempre	2022-10-22 22:52	2022-10-22 23:26
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	baixa	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			

		Previsto para a Versão:		
Resumo:	Listagem de empréstimo com aviso de nulidade			
Descrição:	Quando a listagem não trouxer nenhum elemento, aparece na página a mensagem uma mensagem dizendo que nada foi encontrado			
Marcadores:				
Passos para Reproduzir:	<p>1 - View - "Emprestimo" - "Listagem.cshtml"</p> <pre>2 - <tbody> @foreach(var e in Model) { @if((System.DateTime.Compare(System.DateTime.Now, e.DataDevolucao)>0)) { Classe="text-danger"; } else { Classe=""; } <tr class="@Classe"> <td>@e.Id</td> <td>@e.NomeUsuario</td> <td>@e.DataEmprestimo.ToString("dd/MM/yyyy")</td> <td>@e.DataDevolucao.ToString("dd/MM/yyyy")</td> <td>@e.Livro.Titulo</td> <td>@e.Livro.Autor</td> <td>/Emprestimo/Edicao/@e.Id</td> </tr> } </tbody> </table> } else { <h2>Nenhum registro encontrado</h2> }</pre>			
Informações Adicionais:				
Arquivos Anexados:				
Não há anotações anexadas a esta tarefa.				

Ver Detalhes da Tarefa					
Nº:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
9	[Projeto Biblioteca] General	mínimo	sempre	2022-10-22 22:53	2022-10-22 23:22
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	baixa	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		

Projeção:	nenhuma			
Tempo Estimado:	nenhum	Corrigido na Versão:		
		Previsto para a Versão:		
Resumo:	funcionalidade na ordem da listagem de empréstimo			
Descrição:	Os registros aparecem em ordem decrescente de data de devolução.			
Marcadores:				
Passos para Reproduzir:	1 - Models - "EmprestimoService.cs" 2 - public ICollection<Emprestimo> ListarTodos(FiltrosEmprestimos filtro = null) 3 - return query.Include(e => e.Livro).OrderByDescending(e => e.DataDevolucao).ToList();			
Informações Adicionais:				
Arquivos Anexados:				
Não há anotações anexadas a esta tarefa.				

Ver Detalhes da Tarefa								
Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:			
10	[Projeto Biblioteca] General	grande	sempre	2022-10-22 22:54	2022-10-22 23:17			
Relator:	administrator	Plataforma:						
Atribuído a:	administrator	SO:						
Prioridade:	normal	Versão SO:						
Estado:	resolvido	Versão do Produto:						
Compilação do Produto:		Resolução:	aberto					
Projeção:	nenhuma							
Tempo Estimado:	nenhum	Corrigido na Versão:						
		Previsto para a Versão:						
Resumo:	Todos os campos são obrigatórios no cadastro de empréstimos							
Descrição:	Não é mais possível clicar em "salvar" sem antes digitar em todos os campos							
Marcadores:								
Passos para Reproduzir:	1 - View "Cadastro.cshtml" de "Emprestimo" 2 - colocar "required"							
Informações Adicionais:								
Arquivos Anexados:								

Não há anotações anexadas a esta tarefa.

Ver Detalhes da Tarefa

Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
3	[Projeto Biblioteca] General	pequeno	sempre	2022-10-22 22:48	2022-10-22 23:10
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	normal	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			
		Previsto para a Versão:			
Resumo:	Aviso de "indisponibilidade" na listagem de livros				
Descrição:	Um aviso será executado se não houver livros				
Marcadores:					
Passos para Reproduzir:	1 - View "Listagem.cshtml" de "Livro" 2 - @for(int i =(PaginaAtual-1)*NumLivrosPorPag; i<(NumLivrosPorPag*PaginaAtual<Model.Count? NumLivrosPorPag*PaginaAtual:Model.Count); i++) { <tr> <td>@Model[i].Id</td> <td>@Model[i].Titulo</td> <td>@Model[i].Autor</td> <td>@Model[i].Ano</td> <td>Editar</td> </tr> } </tbody> </table> } else { <h2>Nenhum Registro Encontrado</h2> }				
Informações Adicionais:					
Arquivos Anexados:					
Não há anotações anexadas a esta tarefa.					

Ver Detalhes da Tarefa

Núm:	Categoria:	Gravidade:	Freqüência:	Data de Envio:	Última Atualização:
------	------------	------------	-------------	----------------	---------------------

2	[Projeto Biblioteca] General	grande	sempre	2022-10-22 22:47	2022-10-22 23:02
Relator:	administrator	Plataforma:			
Atribuído a:	administrator	SO:			
Prioridade:	urgente	Versão SO:			
Estado:	resolvido	Versão do Produto:			
Compilação do Produto:		Resolução:	aberto		
Projeção:	nenhuma				
Tempo Estimado:	nenhum	Corrigido na Versão:			
		Previsto para a Versão:			
Resumo:	Os campos estão obrigatórios em "Cadastro de Livros"				
Descrição:	Todos os campos estão obrigatórios: Livro, Ano e Autor				
Marcadores:					
Passos para Reproduzir:	1 - Ir na View "_FormCadastro.cshtml"; 2 - Colocar "required" nos campos.				
Informações Adicionais:					
Arquivos Anexados:					
Não há anotações anexadas a esta tarefa.					