Scrum e Kanban de forma prática e descomplicada

Aprenda de forma prática, como converter seus casos de Uso em Estórias de Usuário e seu Cronograma em Gestão de Demandas



**Luís Gabriel Nascimento Simas**   
Login: [gabrielsimas@gmail.com](mailto:gabrielsimas@gmail.com)   
Email: [gabrielsimas@gmail.com](mailto:gabrielsimas@gmail.com)   
Atua como Desenvolvedor de Software desde 1996 atualmente é Engenheiro de Software na Ci&T**,** é aluno da COTI Informática e da Universidade Estácio de Sá, trabalha com metodologias Ágeis em seus Projetos e é proficiente nas Linguagens Java e C#

[DIAGRAMADOR – Sugestão de ilustração – [Imagem de uma Caldeira de Metalúrgica com Ferro fundido líquido caindo e faíscas em torno]

|  |
| --- |
| Integrando Spring.NET com NHibernate e ASP.MVC - Parte 2 |
| No artigo anterior vimos um comparativo entre o NHibernate e o Spring Data, que internamente têm uma implementação do NHibernate até a versão 3.2, mostramos ao leitor as diferenças entre ambos no que faz menção à produtividade e controle da aplicação.  Nesta segunda e última parte, veremos uma aplicação de “mundo real“ utilizando esta integração, na qual tem muito a acrescentar em produtividade para o leitor. O Spring.NET tem seu nome modificado para Spring MVC quando seu módulo de nome *Spring.Web.Mvc* é utilizado. |
|  |
| **Em que situação o tema é útil** |
| O leitor terá a oportunidade de verificar como o Spring.NET pode atuar em mais de uma camada de forma estável e funcional, comprovando que não só para Injeção de Dependências - DI - este framework deve ser utilizado, a DI é apenas o atributo principal, o mínimo para que o Spring.NET funcione corretamente e daí então se acoplar a outras camadas da aplicação de forma exclusiva e extensiva, o que dá a ele uma grande flexibilidade para controlar todo o ciclo de vida da aplicação. Assim como na Camada de Dados, ele pode ser largamente utilizado na camada de Apresentação, o Spring Data, nós já sabemos do que se trata este nome, pois quando integrado ao ASP.NET MVC, o Spring.NET se torna Spring MVC. Desta forma, a Camada de Negócios, neste artigo ficará apenas com o acesso via Serviços. O leitor também terá a oportunidade de verificar mais dois padrões de projetos sendo utilizados neste artigo: Façade e Data Transfer Object, também conhecido como DTO. O Spring.NET, como o leitor pode notar, facilmente pode ser utilizado em ambientes corporativos nas “Fábricas de Software”, pois seus componentes sempre requerem o uso de Padrões de Projeto para que sua implementação seja minimamente funcional. Não teremos só, neste artigo, abordagem sobre o Spring MVC e Spring Data,também iremos abordar o uso do Razor, ASP.NET MVC, Arquitetura em Camadas e como bônus, veremos um cadastro sendo portado para dispositivos móveis. |

|  |
| --- |
| **Nota Devman** |
|  |

**Por que usar o Spring MVC?**

**Diferenças entre Spring MVC e ASP.NET MVC**

**Conclusão**

O uso do Spring bem como do NHibernate geram para um projeto um grande ganho de qualidade e produtividade muito grandes e neste artigo focamos justamente em demonstrar as diferenças do uso do NHibernate puro e do mesmo junto com o Spring.

Na próxima edição veremos um exemplo prático de uma aplicação completa unindo estes dois excelentes frameworks do mundo .NET ao ASP.NET MVC.

**Links**

**Site do Projeto Spring.NET:**

http://www.springframework.net

**Site do Projeto NHibernate:**

http://nhforge.org/