

Aplicação de Domain-Driven Design no gerenciamento de GRUs de Cronotacógrafo no Inmetro/RS

Tiago O. de Farias¹

¹UniRitter Laureate International Universities

Rua Orfanatrópio, 555 - Alto Teresópolis - 90840-440 - Porto Alegre - RS - Brasil

tiago.farias.poa@gmail.com

Abstract.

Resumo. Desde 1997, quando foi instituído o Código de Trânsito Brasileiro, veículos de carga com peso bruto superior a 4.536 kg e de passageiros com mais de 10 lugares devem possuir cronotacógrafo. A partir de 2009, os instrumentos também devem ser verificados periodicamente pelo Inmetro (Instituto Nacional de Metrologia, Qualidade e Tecnologia), o que aumenta a confiabilidade das medições. Todo o processo de verificação do cronotacógrafo é gerenciado através de um sistema web. Este trabalho apresenta uma proposta de melhoria do atual sistema web utilizando algumas técnicas de Domain-Driven Design. O principal objetivo é fornecer uma estrutura organizada, altamente reutilizável e produtiva; onde o ciclo de vida da aplicação pode ser prolongado. Ao longo deste trabalho é feito um estudo sobre o conceito de Domain-Driven Design, além do estudo de caso de melhoria da atual aplicação.

1. Introdução

O Brasil está entre os países que tornaram o uso do cronotacógrafo obrigatório em ônibus e caminhões, o instrumento inibe os excessos e ajuda a reduzir os acidentes, uma vez que registra o histórico das velocidades desenvolvidas, distâncias percorridas e tempos de movimento e paradas do veículo.

O processo de verificação tem basicamente três grandes etapas: emissão e pagamento da GRU (Guia de Recolhimento da União), realização da selagem e realização do ensaio metrológico. O sistema de gerenciamento de verificação do cronotacógrafo existe há pelo menos oito anos e atende a todos os estados da federação. Inicialmente foi concebido para simples emissão de GRU onde o proprietário do veículo acessava o site do cronotacógrafo preenchia seus dados, emitia e pagava GRU, após se dirigia a um posto de selagem para dar início ao processo, ao finalizar a selagem e dentro de um período determinado deveria procurar um posto de ensaio para realizar o ensaio metrológico e assim recebia o certificado do Inmetro que garante que o instrumento atende aos padrões vigentes sob penalização de ser autuado pela polícia federal.

No dia 01 de janeiro de 2016 houve uma grande mudança no processo de emissão de GRU, hoje os postos de selagem e de ensaio emitem a GRU, e semelhante a um plano pré-pago de celular, o posto compra créditos para cada GRU emitida, cada crédito permite ao posto acessar o site do Inmetro e registrar os dados do serviço realizado, seja serviço de selagem ou de ensaio metrológico.

Este artigo está organizado da seguinte forma. A seção 2 descreve o que é a Linguagem Ubíqua e sua importância. A seção 3 introduz o conceito de DomainDriven Design. A seção 4 descreve como isolar o domínio das demais partes do sistema. A seção 5 fala sobre representação do modelo no software e a diferença entre entidade e objeto de valor. A seção 6 trata de padrões para se manter o ciclo de vida de um objeto de domínio. A seção 7 demonstra algumas técnicas descritas pelo DDD em um caso de uso.

2. Referencial Teórico

2.1. Domain-Driven Design

Domain-Driven Design é uma abordagem de desenvolvimento de software desenhado para gerir a complexa e grande escala de produtos de software.[Evans 2003]

Segundo Evans é um processo que alinha o código do desenvolvedor com o problema real, um conjunto de técnicas de desenvolvimento de software, usadas principalmente em projetos complexos que provê conceitos e regras para ajudar no ciclo do desenvolvimento de software, essas técnicas também tem o objetivo de ajudar clientes, gestores e todas as pessoas envolvidas no processo de desenvolvimento. Seguir a filosofia DDD dará aos desenvolvedores o conhecimento e as habilidades que eles precisam para enfrentar sistemas de negócios grandes e complexos de maneira eficaz.[Millett and Tune 2015]

A solução está em resolver o problema do domínio, focado no domínio do problema, falar a mesma linguagem dos especialistas do domínio. Pode ser aplicado em qualquer projeto desde que não torne complicado o desenvolvimento, algumas vezes o desenvolvimento não é tão complexo que necessite do DDD. No seu coração ele se concentra na criação de uma linguagem comum conhecida como a linguagem ubíqua para descrever de forma eficiente e eficaz um domínio de problemas.[Millett and Tune 2015] De acordo com o conceito do DDD o mais importante em um software não é somente o código, não é somente a arquitetura, tampouco a tecnologia sobre o qual foi desenvolvido, mas sim o problema que o mesmo se propõe a resolver, a regra de negócio. Ela é a razão do software existir. DDD é sobre o desenvolvimento de conhecimento em torno do negócio e de utilizar a tecnologia para fornecer valor.[Carlos Buenosvinos and Akbary 2014]

3. Estado da Arte

4. Solução Implementada

4.1. Subsections

The subsection titles must be in boldface, 12pt, flush left.

5. Considerações Finais

6. Figures and Captions

Figure and table captions should be centered if less than one line (Figure 1), otherwise justified and indented by 0.8cm on both margins, as shown in Figure 2. The caption font must be Helvetica, 10 point, boldface, with 6 points of space before and after each caption.

In tables, try to avoid the use of colored or shaded backgrounds, and avoid thick, doubled, or unnecessary framing lines. When reporting empirical data, do not use more



Figure 1. A typical figure

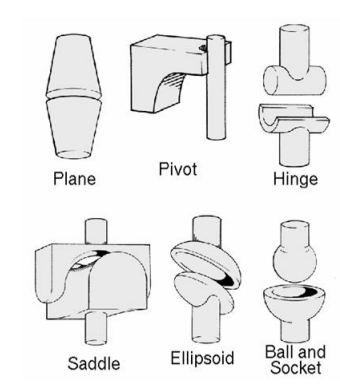


Figure 2. This figure is an example of a figure caption taking more than one line and justified considering margins mentioned in Section 6.

decimal digits than warranted by their precision and reproducibility. Table caption must be placed before the table (see Table 1) and the font used must also be Helvetica, 10 point, boldface, with 6 points of space before and after each caption.

7. Referencias

Bibliographic references must be unambiguous and uniform. We recommend giving the author names references in brackets, e.g. [Knuth 1984], [Boulic and Renault 1991], and [Smith and Jones 1999].

The references must be listed using 12 point font size, with 6 points of space before each reference. The first line of each reference should not be indented, while the subsequent should be indented by 0.5 cm.

References

Boulic, R. and Renault, O. (1991). 3d hierarchies for animation. In Magnenat-Thalmann, N. and Thalmann, D., editors, *New Trends in Animation and Visualization*. John Wiley & Sons Ltd.

Table 1. Variables to be considered on the evaluation of interaction techniques

	Chessboard top view	Chessboard perspective view
Selection with side movements	6.02 ± 5.22	7.01±6.84
Selection with in- depth movements	6.29±4.99	12.22±11.33
Manipulation with side movements	4.66± 4.94	3.47±2.20
Manipulation with in- depth movements	5.71 ±4.55	5.37 ±3.28

Carlos Buenosvinos, C. S. and Akbary, K. (2014). *Domain-Driven Design in PHP*. Lean-pub. Real examples written in PHP showcasing DDD Architectural Styles, Tactical Design, and Bounded Context Integration.

Evans (2003). *Domain-Driven Design: Tacking Complexity In the Heart of Software*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Knuth, D. E. (1984). *The T_EX Book*. Addison-Wesley, 15th edition.

Millett, S. and Tune, N. (2015). Patterns, principles, and practices of domain-driven design.

Smith, A. and Jones, B. (1999). On the complexity of computing. In Smith-Jones, A. B., editor, *Advances in Computer Science*, pages 555–566. Publishing Press.