



7^a Conferência da Qualidade de Software e Serviços



Ensinando a programar com <coding dojo>

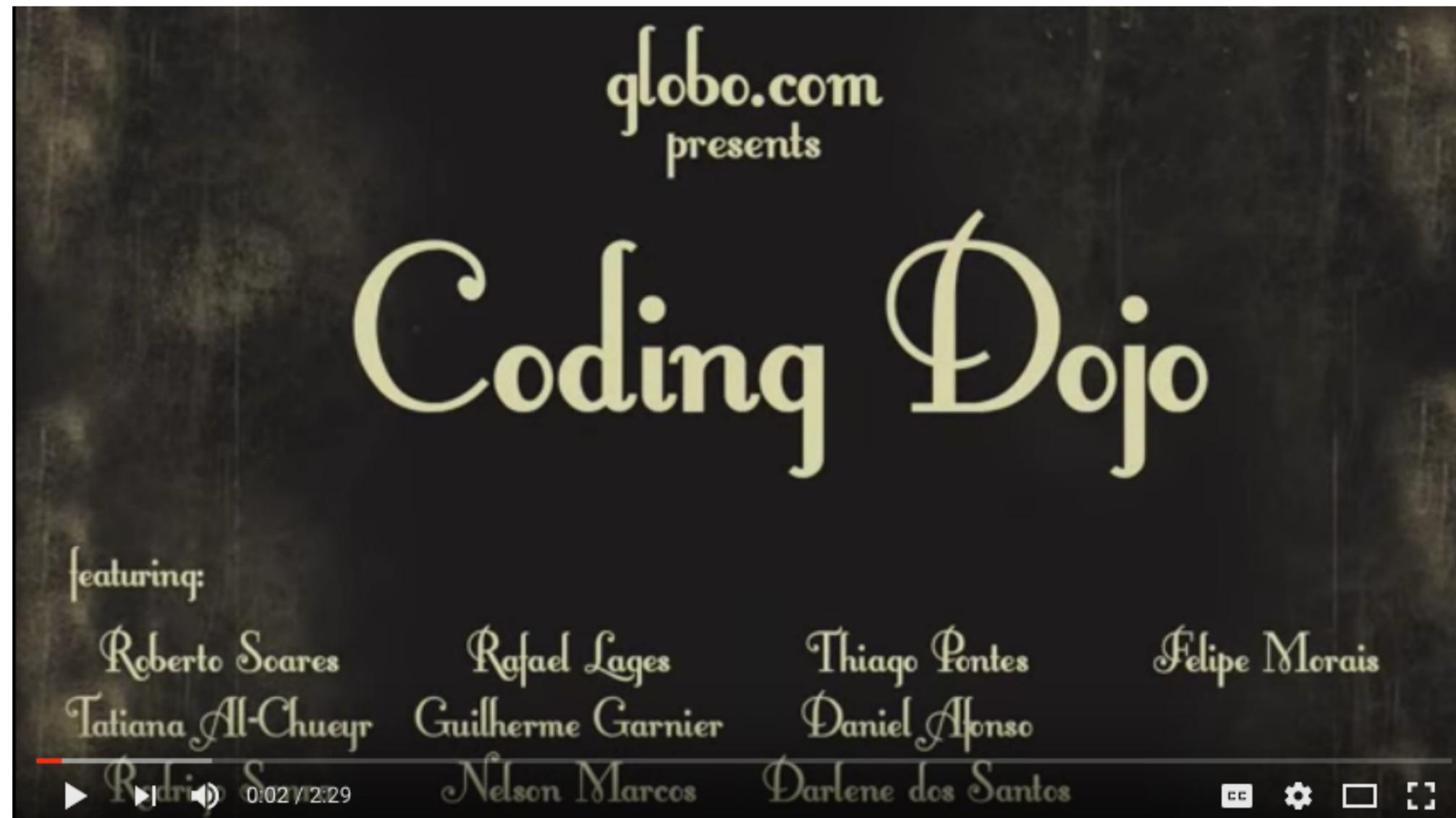
Prof. A. Sergio F. Bonato

Professor das Disciplinas de Programação dos cursos de Ciência da Computação, Sistemas de Informação e Análise e Desenvolvimento de Sistemas da USJT - Diretor de Desenvolvimento de Sistemas da SEFAZ-SP



[Dojo Fight - cena do filme The Matrix (1999)]

[coding dojo na globo.com]



<vídeo disponível em <https://www.youtube.com/watch?v=vqnrwQ3oVM1M>
consultado em 07/06/2016>

[motivação]

**Ensinar
POO é
difícil**

**USJT
ensina POO
+ 10 anos**

**Cursos
mudaram
para
semestrais**

**Nova
metodologia
de ensino**

dojo aplicado a uma turma de ADS em 2015/2 e este semestre 2016/1 aplicado às novas turmas de CC e SI

[papéis]

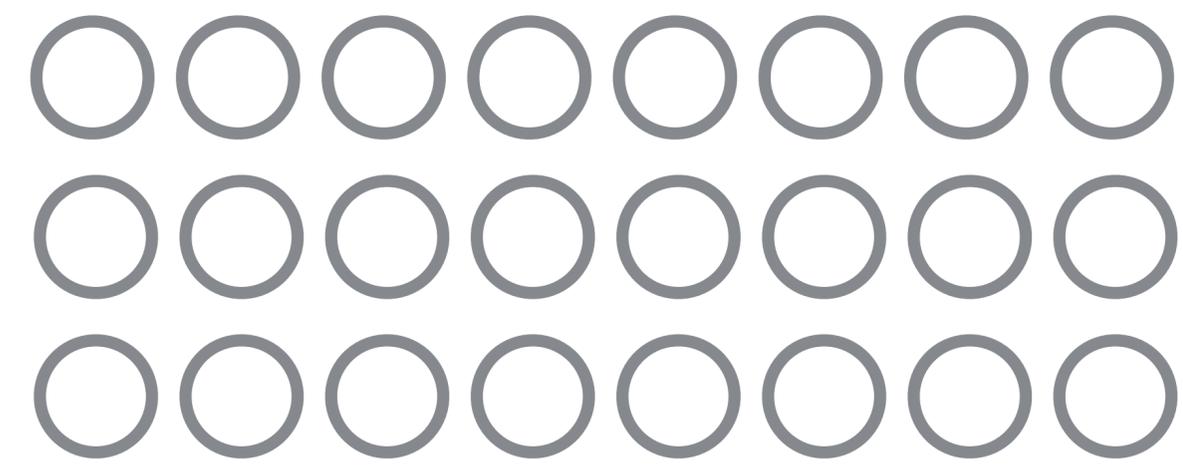


tela

sensei ○

teclado

piloto ○ ○ copiloto



plateia

[kata]

- O **sensei** apresenta a solução de um problema
- A plateia assiste
- Como usamos o laboratório, os alunos ligam as máquinas e vão copiando e executando a solução

[randori]

- O **sensei** (professor) propõe um problema
- O alunos são chamados em duplas para o teclado: **piloto** e **copiloto**
- **Piloto** tecla, **copiloto** ajuda na solução
- A plateia assiste e dá palpites mediante incentivo do **sensei**
- **Sensei** troca a dupla a cada 5 minutos
- Computadores **desligados** (código é disponibilizado no final da aula)
- No final o **sensei** faz uma retrospectiva

[kake]

- Formato de dojo ainda não utilizado na USJT
- Semelhante ao Randori, porém há várias duplas trabalhando simultaneamente.
- A cada turno as duplas são trocadas, promovendo a integração entre todos os participantes do evento.
- Um conhecimento mais avançado dos participantes é necessário.

[avaliação do aluno no dojo]

participação

comportamento

assiduidade

nunca pelo código produzido

[resultados qualitativos]

- maior envolvimento do aluno
- as dúvidas e dificuldades de cada aluno ficam evidentes para o professor
- o aluno aprende a trabalhar de forma colaborativa
- o aluno aprende a se expor e expor seu trabalho
- nível de aprendizagem igual ou superior ao dos cursos anuais

[resultados quantitativos]

- temos resultados fechados apenas de uma turma
- foi aplicada uma questão comum na prova final para as turmas anuais de CC e SI e para turma semestral de ADS
- Acertos:
 - **Ciência da Computação: 29%**
 - **Sistemas de Informação: 8%**
 - **Análise e Desenvolvimento de Sistemas: 13%**

[referências]

- Martin, R. (2011), The Clean Coder: A Code of Conduct for Professional Programmers, Prentice Hall Press, 1st edition.
- Beck, K. (2000), Extreme Programming Explained: Embrace Change, Addison-Wesley, 1st edition.
- Beck, K. (2002), Test Driven Development: By Example. Addison-Wesley, 1st edition.
- Heinonen, K. and Hirvikoski, K. and Luukkainen, M. and Vihavainen, A. (2013), "Learning Agile Software Engineering Practices Using Coding Dojo", In: Proceedings of the 14th Annual ACM SIGITE Conference on Information Technology Education, ACM Press, New York, NY, USA.
- Kolling, M. (1999). The Problem of Teaching Object-Oriented Programming. Part 1: Languages, Journal of Object-Oriented Programming, 11(8):8-15, 1999.

Obrigado!

A. Sergio F. Bonato
asbonato@gmail.com

Linked 

<http://www.linkedin.com/in/sbonato>