

MAC0350

Introdução ao desenvolvimento de sistemas de software

Mensagens importantes

- Eu não sou o JEF...
 - Alfredo
 - Fabio
 - JEF
- Primeiro oferecimento
- O quê é um sistema de software (SS)? Por que a disciplina precisa de três professores?
 - Sistemas serão usados por pessoas e possivelmente por muito tempo

Mensagens importantes

- Boa parte dos alunos chega ao quinto ano fazendo apenas algoritmos e EPs
 - Desenvolve, monitor corrige e programa morre
- Sistemas
 - Algo durável: quando você acaba, outras pessoas continuam o trabalho
 - Mais de uma pessoa envolvida
 - Maior complexidade
 - Persistência
 - Web / App
 - Exceções: sistemas de loja; sistemas operacionais; ERP.

O que precisamos saber para fazer sistemas?

- ~~Editor de texto: VI / EMACS~~
- IDE
 - Auto completar
 - Identação
 - Debug - Integrado
 - Eclipse / PyCharm / VisualStudio / AndroidStudio / Netbeans
- Controle de versão
 - Exemplo de uso correto
 - fazer merge
 - Não fazer commit com código comentado
 - Mensagens relevantes para o futuro
 - Commits pequenos (mudanças atômicas)
 - Git (git[hub|lab], bitbucket); HG; SVN; CVS

Mais ferramentas

- Testes automatizados
 - RSpec / *unit / selenium / cucumber (BDD) / state chart / MochaJS
- Linguagem
 - C / C++ / Java / Python
- Integração contínua
 - Repositório é um gatilho para diversas tarefas automatizadas
 - Executar testes
 - Gerar versão
 - Travis / Jenkins / GitlabCI / CodeCov / AppVeyer
- Docker
 - Ambiente onde as dependências são estáveis

Mais ferramentas

- Métricas de código
 - Código cheira mal
 - Rubocop (estilo de código) / Code climate / Mezuro / Sonar Qube

Próxima aula

- Integração contínua
- Docker
- Métricas