

MC855 – Projetos em Sistemas de Computação

(2º semestre 2019)

Profa. Juliana Freitag Borin

juliana@ic.unicamp.br
www.ic.unicamp.br/~juliana

Prof. Breno B. N. de França

breno@ic.unicamp.br
www.ic.unicamp.br/~breno

Dia	Horário	Sala
Terça-feira	16-18h	
Quinta-feira	16-18h	

1 Ementa da Disciplina

Implementação de um projeto prático na área de Sistemas de Computação.

2 Objetivos Terminais

Ao final do curso, o aluno deve ser capaz de:

- ★ Aplicar corretamente as práticas de **desenvolvimento ágil de software** e conhecimentos de redes de computadores e sistemas distribuídos, no contexto de um projeto de um sistema envolvendo tecnologias da **Internet das Coisas** que possa ser integrado ao projeto Smart Campus Unicamp.

3 Conhecimentos Requeridos

- Programação modular (ou orientada a objetos) em linguagem com suporte a definição de interfaces e componentes (módulos ou classes), como C, C++ e Python;
- Análise e Projeto de Software, que compreende a especificação de requisitos, bem como as transformações de requisitos código fonte, passando por atividades de projeto e arquitetura;
- Noções de Teste de Software, sobretudo testes funcionais em nível unitário;
- Projeto e implementação de Banco de Dados;
- Conceitos de sistemas distribuídos;
- Conhecimento básico de protocolos de rede;
- Eletrônica básica.

4 Projeto Proposto

O(s) projeto(s) a ser(em) desenvolvido(s) pelos alunos estão elencados a seguir e devem utilizar tecnologias de IoT (*Internet of Things*) e ser conduzido com métodos ágeis de desenvolvimento (utilizando práticas como histórias de usuário, controle de versões, testes automatizados, integração contínua e fluxo Kanban):

- Monitoramento de ônibus internos da Cidade Universitária da UNICAMP - SmartCampus (Cliente: Prefeitura da Unicamp)
- Sistema para trancamento e abertura de portas automático (smartlock) na UNICAMP - SmartCampus (Cliente: Prefeitura da Unicamp)
- Outro projeto proposto pela equipe em acordo com os professores.

Procedimento pedagógico: aprendizado com base na execução de um projeto, onde os alunos devem se organizar em equipes multifuncionais, que são inteiramente responsáveis por módulos distintos do sistema, mas que devem ser integrados ao mesmo. As equipes deverão realizar entregas ao final de cada iteração, com uma demonstração do que foi realizado no decorrer do período. Durante as iterações, é esperado que as equipes se organizem em atividades de levantamento de requisitos (histórias de usuário), design, codificação e testes. Essas atividades devem ser apoiadas pela prática de integração contínua. Ainda, estão previstas exposições sobre a metodologia de desenvolvimento a ser seguida e uma visão geral sobre Internet of Things e o projeto Smart Campus.

5 Avaliação

A avaliação da disciplina será realizada com base nos critérios a seguir:

1. **Participação (10%):** este critério é individual. A atribuição da nota de participação é proporcional e considera frequência, envolvimento nas atividades em sala/laboratório e fora de sala, além de cumprimento de prazos relativos às entregas.
2. **Entregas (60%):** as entregas devem conter o escopo da sprint implementado, disponibilizado no repositório de controle de versão, em estado funcional, e com a suíte de testes executando com sucesso. Ainda, uma demonstração deve ser feita com o estado atual ao final da iteração.
3. **Estado Final do Produto (30%):** o sistema ao final deve estar funcional, com funcionalidades implementadas suficientemente para uso. Ainda, o resultado deverá ser apresentado no formato de *poster* no dia 27/11 no workshop do IC.

Observação: serão marcadas reuniões periódicas ao longo do semestre para apresentações do desenvolvimento do sistema proposto.

5.1 Informações Importantes

- A presença é obrigatória em todas as aulas (incluindo laboratório). Frequência inferior a 75% causa reprovação.
- Casos de plágio (cópia de texto, imagem ou ideia) entre os trabalhos ou de conteúdos externos serão tratados com rigor. A nota da avaliação em questão será anulada sem possibilidade de reposição e o caso será encaminhado à coordenação do curso.
- É esperado que o aluno desempenhe atividades do projeto fora do horário de aula (mínimo de 8 horas).
- Essa disciplina não prevê a realização de provas ou exames finais.

6 Bibliografia

COHN, Mike. **User stories applied: For agile software development**. Addison-Wesley Professional, 2004.

FOWLER, Martin; FOEMMEL, Matthew. Continuous integration. **Thought-Works**) [http://www.thoughtworks.com/Continuous Integration.pdf](http://www.thoughtworks.com/Continuous%20Integration.pdf), v. 122, 2006.

SCHWABER, Ken; BEEDLE, Mike. **Agile software development with Scrum**. Upper Saddle River: Prentice Hall, 2002.

GAMMA, Erich. **Design patterns: elements of reusable object-oriented software**. Pearson Education India, 1995.

BASS, Len. **Software architecture in practice**. Pearson Education India, 2007.

TANENBAUM, Andrew S; VAN STEEN, Maarten. **Distributed Systems: Principles and Paradigms**. Third edition, CreateSpace, 2017.

KUROSE, Jim; ROSS, Keith. **Computer Networking a Top Down Approach**. Addison-Wesley, 6th edition, 2012.

VASSEUR, Jean-Philippe; DUNKELS, Adam. **Interconnecting Smart Objects with IP: The Next Internet**. Morgan Kaufmann Publishers Inc., 2010.