

**PLANO ANALÍTICO DA UNIDADE CURRICULAR**

Faculdade/Escola/Centro*: Faculdade de Ciências*

Departamento de Matemática e Informática Ano lectivo de 2017 Semestre: II

Curso (s) : Licenciatura em Informática

Unidade Curricular: **Desenvolvimento de Aplicações Web**

**IV. TEMÁTICAS**

| **No** | **Temas** | **Observações** |
| --- | --- | --- |
|
|  | **Sistemas de Gestão de Conteúdos: CMS*** Conceito
* Tipos e Exemplos de CMS
* Critério de escolha
* Instalação, configuração e utilização
 |  |
|  | **Frameworks web** * Conceito
* Diferenciação (versus biblioteca)
* Características
* Critérios de seleção
 |  |
|  | **Frameworks Front End*** Tipos, critérios de seleção
* Design patterns: MVC, MVVM
* Desenvolvimento por Componentes
* UX e UI Design systems
 |  |
|  | **Frameworks Backend*** Tipos, critérios de seleção
* Design patterns: MVC, Dependency injection
* Base de dados e a Aplicação de ORMs
* Desenvolvimento de APIs e Web services
	+ REST, SOAP
 |  |
|  | **Controlo de Versão em projetos de Software*** Conceito e Importância
* Funcionamento
* Controlo de versão Centralizado e Distribuído
* Instalação, Configuração e Utilização
* Práticas de desenvolvimento com controlo de versão
	+ Feature branch development
	+ Trunk-based development, feature-toggles
	+ Mono-repo, multi-repo
 |  |
|  | **DevOps*** Conteinerização
	+ Conceito e funcionamento (ex: docker)
	+ Container orchestration (kubernetes, openshift etc…)
* Integração contínua e entrega contínua
* Software monitoring / Monitoria de Software
	+ Ferramentas de Logging (ex. Splunk
	+ Ferramentas de coleta de métricas
 | **Promover seminários e workshops** |
|  | **Automated Testing** * Tipos de teste: unit, integration, system, smoke, performance, end-to-end
* Implementação prática: unit e end-to-end
 |  |
|  | **Segurança em Aplicações Web Avançado*** Security by Design
* Vulnerabilidades e Ameaças na web
* Scan de Vulnerabilidades e Code smells
* Testes de Penetração / Pentesting
* Frameworks de controlo de segurança
* Hacking de aplicações
 | **SQL injection, XSS, CSRF, CORs,**  |
|  | **Deployment de Aplicações** * Processo e ferramentas de deploy
* On-premise e Cloud
* Configurações de Servidores web
* Aplicações Serverless
 | **Ambientes:** desenvolvimento, teste, QA, alpha, produçãoon-premise: deploy num data-center ou VPScloud: GCP, AWS, Azure, Heroku etc… |
|  | **Escalabilidade e Performance de Aplicações web*** Estratégias, fatores e ferramentas
* Arquitecturas: monolith e microservices
* Caching
* Load balancing
* Cloud Storage
 |  |
|  | **Tendências do Desenvolvimento de Aplicações*** Deep web
* Progressive Web App
 |  |
|  |  |  |
|  | **Preparação e realização de Avaliações (Testes, Defesas, Apresentações)** |  |
|  |  |

**VI. BIBLIOGRAFIA E RECURSOS**

1. J. Moscovitz, M., Lewis, J. R..(2008). Foundation Website Creation with CSS, XHTML, and JavaScript.FriendSof – Design to Designer, United States of America;
2. S. Krug (2014). Don't Make Me Think, Revisited: *A Common Sense Approach to Web Usability*, 3rd Edition
3. [G. Alonso](http://link.springer.com/search?facet-creator=%22Gustavo+Alonso%22), [F. Casati](http://link.springer.com/search?facet-creator=%22Fabio+Casati%22), [H. Kuno](http://link.springer.com/search?facet-creator=%22Harumi+Kuno%22) and [V. Machiraju](http://link.springer.com/search?facet-creator=%22Vijay+Machiraju%22) (2004). Web Services: *Concepts, Architectures and Applications*
4. *IEEE(2009).* Why are software projects moving from centralized to decentralized version control systems? Vancouver, BC (Disponível em [10.1109/CHASE.2009.5071408](http://dx.doi.org/10.1109/CHASE.2009.5071408))
5. Diomidis Spinellis (2005). Version Control Systems. Athens University of Economics and Business (disponível em <http://www.academia.edu/download/30699854/01504674-VCS.pdf> )
6. OWASP, T. (2013). Top 10–2013. *The Ten Most Critical Web Application Security Risks*. Papapanagiotou, K. (2013, November). OWASP Hackademic: a practical environment for teaching application security. In *AppSec USA 2013*. Owasp.