**UNIVERSIDADE FEDERAL DE GOIÁS**

ENGENHARIA DE SOFTWARE

**Integração II**

**LIÇÕES APRENDIDAS – FRONT-END**

**LIÇÃO 1**

O planejamento, estruturação de atividades e a escolha de tecnologias a serem empregadas em um determinado projeto deve levar em consideração, entre outros aspectos, o nível técnico da própria equipe envolvida, algo que não foi percebido na escolha do AngularJS para o projeto em questão, posto que a maioria dos colaboradores não tinha nenhuma experiência com a tecnologia adotada. Para que tal impacto fosse amenizado, tomou-se a decisão de tratar o emprego do AngularJS como uma evolução do sistema atual, que foi desenvolvido com o emprego do jQuery.

**LIÇÃO 2**

Nas reuniões da equipe de requisitos com os stakeholders, nenhum colaborador do desenvolvimento estava presente, isso acabou gerando somente requisitos de mais alto nível e o desenvolvimento acabou ficando desamparado nestas questões. Deste modo, para reuniões futuras, é preponderante que pelo menos um membro do desenvolvimento e/ou arquitetura participe destes encontros até mesmo para delimitar aspectos de mais baixo nível.

**LIÇÃO 3**

As tecnologias de versionamento empregadas, ou mesmo a forma de se trabalhar com as mesmas deve ser de total entendimento e controle de cada um dos membros do projeto pois tivemos situações em que erros na forma de trabalho (commits em branchs errados ou mesmo recommit de arquivos ultrapassados) com a ferramenta do git propiciou atrasos na entrega e deploy da aplicação para que se pudesse ser demonstrado o funcionamento para o stakeholders.

**LIÇÃO 4**

A adoção de sprints semanais também configurou um ponto a ser revisto para próximas construções até mesmo porque inerente à própria abertura e encerramento das sprints existem artefatos documentais que necessitam ser criados e isto acaba gerando um overhead muito grande pela periodicidade dos acontecimentos e muitas vezes para conseguirmos cumprir com todos os compromissos documentais firmados, necessitamos alocar um membro somente para fazer este acompanhamento, o que acaba sobrecarregando atividades de desenvolvimento propriamente ditas. Um modelo adotado poderia utilizar sprints de duas semanas, seria um padrão válido a ser utilizado o que acarretaria na diminuição da periodicidade da geração de tais artefatos.