

Resumo para *The Mythical Man-Month*

É muito comum projetos, no mundo da computação, terem sua entrega final adiada ou não atender às expectativas esperadas por diversos fatores como: técnica de estimativa mal desenvolvida, desentendimento entre esforço e progresso, mal supervisionamento do progresso no desenvolvimento, etc. E no caso quando ocorrem os famosos atrasos de entrega, a solução escolhida para resolver o problema é mal interpretada.

Por questões diversas, isso ocorre devido a impressão inicial que temos de que *tudo ocorrerá bem*, e é aí que começa toda a defasagem. É possível que tudo ocorra conforme planejado em uma dada tarefa, porém o desenvolvimento de software consiste em tantas tarefas, uma seguida da outra, que a chance de que *todas ocorram bem* diminuem muito.

Outro jeito errôneo, mas comum, de se pensar é a utilização de homens-mês como unidade de medida para um determinado trabalho. O processo de alocar mais pessoas para um projeto se aplica para tarefas que podem ser divididas entre mais *mãos* (como cortar trigo, colher algodão), mas não se aplica no mundo do software em boa parte dos casos, devido a interdependência entre as tarefas.

A tão esperada *alocação ótima* de tempo é, geralmente, mal escolhida quanto aos requisitos do desenvolvimento: planejamento, codificação, testes dos componente e teste final. Por causa do otimismo (de novo), esperamos que ocorram poucas falhas com o decorrer do projeto e alocamos pouco tempo para o planejamento e para os testes, e um tempo desnecessário para a parte mais fácil, o *coding* em si. O que não deve acontecer, mas é muito comum no nosso estilo de vida atual, é procurar atender às expectativas dos gerentes de projeto ao invés de defender estimativas baseadas em situações mais plausíveis para a entrega.

E quando um trabalho já está atrasado? Dentre algumas decisões que podemos tomar nessa hora, a que mais acontece é cortar a tarefa, entregar mal feito. A vaga ideia: quanto mais esforço colocamos num projeto menos tempo conseguimos concluir é falha e cai na lei de Brooke: "*Alocar mais mão-de-obra em um desenvolvimento que está atrasado, faz este ficar mais atrasado ainda*", devido a uma série de fatores como tempo de treinamento inicial e demora na comunicação entre eles.

A conclusão é que o número de meses de um projeto está ligado diretamente ao progresso sequencial dele, enquanto que o número máximo de pessoas permitido em um projeto depende da quantidade de tarefas independentes dentro do projeto. Assim, é possível trabalhar com uma equipe menor durante mais tempo, mas nunca o contrário.