



Curso Técnico em Desenvolvimento de Sistemas Online

PLANEJAMENTO DE TRABALHO DE CONCLUSÃO DE CURSO

GEEaD - Grupo de Estudo de Educação a Distância

Centro de Educação Tecnológica Paula Souza

Expediente

*GEEaD – CETEC GOVERNO DO ESTADO DE SÃO PAULO
EIXO TECNOLÓGICO DE INFORMAÇÃO E COMUNICAÇÃO
CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS
PLANEJAMENTO DO TRABALHO DE CONCLUSÃO DE CURSO - PTCC*

Autor: Eliana Cristina Nogueira Barion

Revisão Técnica: Lilian Aparecida Bertini

Revisão Gramatical: Juçara Maria Montenegro Simonsen Santos

Editoração e Diagramação: Flávio Biazim

São Paulo – SP, 2019

APRESENTAÇÃO

Este material didático do Curso Técnico em Desenvolvimento de Sistemas modalidade EaD foi elaborado especialmente por professores do Centro Paula Souza para as Escolas Técnicas Estaduais – ETECs.

O material foi elaborado para servir de apoio aos estudos dos discentes para que estes atinjam as competências e as habilidades profissionais necessárias para a sua plena formação como Técnicos em Desenvolvimento de Sistemas.

Esperamos que este livro possa contribuir para uma melhor formação e aperfeiçoamento dos futuros Técnicos.

AGENDA 6

PLANEJAMENTO
DAS SPRINTS -
EVENTOS DE
DURAÇÃO FIXA -
TIME BOXED





MERGULHANDO NO TEMA...

O que diferencia o Scrum de outras metodologias ágeis é o desenvolvimento de time-boxes: “caixas” de tempo que possuem capacidade previamente definidas e que não são negociáveis depois que uma iteração ou execução é iniciada, com foco nos processos prioritários, aqueles que uma vez finalizados, já poderão ser utilizados de alguma forma pelo usuário do projeto, agregando valor. Essas “caixas” de tempo chamamos de Sprint, durante esse intervalo a equipe desenvolve um ou mais requisitos do sistema e no final dele valida com o cliente o que foi produzido.

Essa divisão pode ser realizada de maneira bem simples, verificando por exemplo, quais tarefas podem ser realizadas por mais de um membro da equipe.

História	Prioridade (Valor de Negócio)	Tempo (dias)
O sistema deverá possuir um cadastro para armazenamento de dados de pacientes, médicos e funcionários.	1	34
Desenvolvimento de Estruturas de banco de dados para armazenamento de dados de pacientes, médicos e funcionários	1.1	5
Desenvolvimento de aplicações para cadastros dos dados	1.2	13

Vamos utilizar a história 1.2 do Backlog como exemplo. A primeira pergunta seria: Essa tarefa pode ser realizada por mais de um membro?

Se a resposta for positiva, a próxima pergunta seria, por quantos membros?

A quantidade respondida provavelmente será a quantidade de partes que essa tarefa poderá ser dividida. Por exemplo se você respondeu 3 (três), isso quer dizer que o Scrum Master poderá dividir a tarefa em três pequenas partes.

História	Prioridade (Valor de Negócio)	Tempo (dias)
O sistema deverá possuir um cadastro para armazenamento de dados de pacientes, médicos e funcionários	1	34
Desenvolvimento de Estruturas de banco de dados para armazenamento de dados de pacientes, médicos e funcionários	1.1	5
Desenvolvimento de aplicações para cadastros dos dados	1.2	13
Desenvolvimento de aplicações para cadastros dos dados de pacientes	1.2.1	5
Desenvolvimento de aplicações para cadastros dos dados de médicos	1.2.2	5
Desenvolvimento de aplicações para cadastros dos dados de funcionários	1.2.3	3

Após a divisão da história 1.2 podemos visualizar a quantidade de sprints da seguinte maneira:

História	Prioridade (Valor de Negócio)	Tempo (dias)	Sprint
O sistema deverá possuir um cadastro para armazenamento de dados de pacientes, médicos e funcionários	1	34	
Desenvolvimento de Estruturas de banco de dados para armazenamento de dados de pacientes, médicos e funcionários	1.1	5	1
Desenvolvimento de aplicações para cadastros dos dados	1.2	13	
Desenvolvimento de aplicações para cadastros dos dados de pacientes	1.2.1	5	1
Desenvolvimento de aplicações para cadastros dos dados de médicos	1.2.2	5	2
Desenvolvimento de aplicações para cadastros dos dados de funcionários	1.2.3	3	3

Mas porque a história 1.1. não foi dividida? Nesse caso, Gustavo achou melhor não dividi-la, pois sozinha ela não agraga valor ao cliente, ou seja, não será útil, uma vez que a estrutura do banco de dados é uma camada transparente para os usuários, sua validação é mais técnica, deixando para ser entregue junto com a história 1.2.1., já que foram definidas como a Sprint 1, conforme quadro anterior.

Você também deve ter percebido que mesmo dividindo as tarefas, o tempo para entrega do projeto continua o mesmo, isso porque a definição da data final de entrega depende de alguns fatores, principalmente definindo como as sprints serão executadas, uma de cada vez ou mais de uma ao mesmo tempo. O importante agora é entender como é definido e como funciona o ciclo de execução dessas tarefas.

No exemplo anterior foi definido o tempo em poucos dias, mas como vimos na Agenda 1, uma Sprint pode e deve ser limitada a no máximo um mês de duração, fazendo com que todos os envolvidos no projeto não se distanciem. No final de cada Sprint, o time apresenta ao cliente tudo que foi produzido, com objetivo de receber um feedback do produto. Além de promover a comunicação entre as duas partes envolvidas, equipe e cliente, essa proximidade reduz muito o risco do projeto, evitando que grandes erros se propagem durante a execução.

Quanto ao tempo de uma Sprint, sabemos o tempo máximo, e que ele pode ser definido por meio de um Planning Poker entre os membros da equipe, conforme vimos na Agenda 1, mesmo assim é importante não deixar de levar em consideração alguns fatores, como por exemplo:

Disponibilidade do Cliente: se o cliente poderá estar presente apenas uma vez por semana, não há o que fazer, as sprints serão finalizadas nas datas de acordo com a disponibilidade dele.

Em relação à equipe de desenvolvimento podemos citar três fatores:

i) Tamanho: uma equipe grande conseguiria entregar mais sprints que uma pequena no mesmo período, mas é necessária uma avaliação muito criteriosa se isso valerá a pena, porque nesse caso, o cliente terá que validar um pacote muito grande de rotinas entregues.

ii) Conhecimento de tecnologia: se a equipe não tem conhecimento da tecnologia que será utilizada, é melhor a sprint ser maior, para que sua equipe possa adquirir conhecimento suficiente para agregar valor ao produto.

iii) Conhecimento de negócios: quando a equipe não tem domínio da natureza do negócio, há grandes chances de cometer erros de interpretação, neste caso, um feedback mais frequente do cliente, que é o que entende do negócio, é fundamental para melhoria da qualidade do produto. Geralmente, o que temos é que as sprints menores recebem feedbacks excelentes, fazendo com que os encontros com o cliente sejam para apresentar pequenos avanços. Já as sprints maiores, entrega-se mais ao cliente, mas devido à grande quantidade de rotinas entregues, a equipe corre o risco de esquecer a validação de algum requisito.

Cada projeto possui suas particularidades e essas especificidades devem ser tratadas pelo Scrum Master da melhor maneira possível. É necessário que a equipe tente encontrar um equilíbrio produtivo de acordo com as necessidades do projeto.



VOCÊ NO COMANDO

A seguir, tente identificar no Backlog do projeto Consultório Médico, as tarefas que podem ser divididas em pequenas partes, deixando bem claro a quantidade de sprints que o projeto terá.

História	Prioridade (Valor de Negócio)	Tempo (dias)
O sistema deverá possuir um cadastro para armazenamento de dados de pacientes, médicos e funcionários	1	34
Desenvolvimento da Estruturas de banco de dados para armazenamento de dados de pacientes, médicos e funcionários	1.1	5
Desenvolvimento das aplicações para cadastros dos dados	1.2	13
Desenvolvimento de regras para controle da entradas, alteração e recuperação dos dados.	1.3	13
O sistema deverá possuir rotina para agendamento (de consultas?).	2	39
Desenvolvimento de estruturas de banco de dados para armazenamento dos dados das programações das agendas e agendamentos.	2.1	5
Desenvolvimento de aplicações para cadastro de agendas e agendamento de pacientes	2.2	34
O sistema deverá gerenciar as receitas e despesas do consultório.	3	
Desenvolvimento de Estruturas de banco de dados para armazenamento dos dados de receita e despesa	3.1	5
Desenvolvimento de aplicação para cadastro de receitas e despesas e gerenciamento de fluxo de caixa	3.2	13
O sistema deverá gerar relatórios operacionais, estratégicos e gerenciais de atendimento.	4	24
Definição e desenvolvimento dos relatórios operacionais que o sistema deverá gerar	4.1	8
Definição e desenvolvimento dos relatórios estratégicos que o sistema deverá gerar	4.2	8
Definição e desenvolvimento dos relatórios gerenciais que o sistema deverá gerar	4.3	8
O usuário deverá possuir um login e senha para acessar ao sistema	5	26
Desenvolvimento da Estrutura de banco de dados para armazenamento dos dados de usuários	5.1	5
Desenvolvimento das aplicações para cadastros de usuários e gerenciamento de acesso.	5.2	21

Como foi, achou que seria mais fácil? O importante é identificar quando uma entrega agrega valor ao cliente, de nada vai adiantar a equipe terminar alguma parte do projeto que não fica diretamente em contato com o usuário, até porque ele não vai conseguir validar, mas isso não quer dizer que essas tarefas são menos importantes, pelo contrário, na maioria das vezes elas devem ser executadas antes de qualquer outra, é o caso da estrutura do banco de dados.

O mesmo não acontece com os relatórios que podem ficar por último, pois são visões de dados anteriormente cadastrados, ou seja, as aplicações de entrada de dados devem ter sido entregues para que os relatórios possam ser validados.

Com isso, após uma análise das tarefas, podemos obter como resultado o seguinte Backlog do projeto Consultório Médico:

História	Prioridade (Valor de Negócio)	Tempo (dias)	Sprint
O sistema deverá possuir um cadastro para armazenamento de dados de pacientes, médicos e funcionários	1	34	
Desenvolvimento de Estruturas de banco de dados para armazenamento de dados de pacientes, médicos e funcionários	1.1	5	1
Desenvolvimento de aplicações para cadastros dos dados de pacientes	1.2	13	
Desenvolvimento de aplicações para cadastros dos dados de pacientes	1.2.1	5	1
Desenvolvimento de aplicações para cadastros dos dados de médicos	1.2.2	5	2
Desenvolvimento de aplicações para cadastros dos dados de funcionários	1.2.3	3	3
Desenvolvimento de regras para controle da entradas, alteração e recuperação dos dados.	1.3	13	4
O sistema deverá possuir rotina para agendamento.	2		
Desenvolvimento da estruturação do banco de dados para armazenamento dos dados das programações das agendas e agendamentos.	2.1	5	
Desenvolvimento da estruturação do banco de dados para armazenamento dos dados das programações das agendas	2.1.1	3	5
Desenvolvimento da estruturação do banco de dados para armazenamento dos dados dos agendamentos	2.1.2	2	6
Desenvolvimento de aplicações para cadastro de agendas e agendamento de pacientes	2.2	34	

- Desenvolvimento de aplicações para cadastro de agendas	2.2.1	21	5
- Desenvolvimento de aplicações para agendamento de pacientes	2.2.2	13	6
O sistema deverá gerenciar as receitas e despesas do consultório.	3		
Desenvolvimento da estrutura do banco de dados para armazenamento dos dados de receita e despesa	3.1	5	7
Desenvolvimento de aplicação para cadastro de receitas e despesas e gerenciamento de fluxo de caixa	3.2	13	8
O sistema deverá gerar relatórios operacionais, estratégicos e gerenciais de atendimento.	4		
Definição e desenvolvimento dos relatórios operacionais que o sistema deverá gerar	4.1	8	9
Definição e desenvolvimento dos relatórios estratégicos que o sistema deverá gerar	4.2	8	10
Definição e desenvolvimento dos relatórios gerenciais que o sistema deverá gerar	4.3	8	11
O usuário deverá possuir um login e senha para acessar ao sistema	5		
Desenvolvimento da estrutura do banco de dados para armazenamento dos dados de usuários	5.1	5	12
Desenvolvimento de aplicações para cadastros de usuários e gerenciamento de acesso.	5.2	21	12