**CONTRATO DE PRESTAÇÃO DE SERVIÇOS**

Pelo presente instrumento particular de Contrato de Prestação de Serviços, temos:

De um lado a empresa KOPI EDUCACIONAL, sob razão social MD SANTOS INTERNET E SISTEMAS, inscrita no CNPJ sob n.º 00.000.000/0001-00, com Inscrição Estadual n.º 000.000.000.000 e na prefeitura de Caraguatatuba sob n.º 000000, situada à Rua \_\_\_\_\_\_\_\_\_, \_\_\_\_ - \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, na cidade \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/SP, neste ato, representada por Márcio dos Santos, brasileiro, maior, Analista de Sistemas, CPF 000.000.000-00, doravante denominada simplesmente PRESTADOR.

Do outro lado a empresa \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, CNPJ 00.000.000/0001-00, Inscrição Estadual: 000.000.000.000, inscrição Municipal: 000000, situada à Rua \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_ , \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_\_\_/SP, CEP 00000000, representada por seu proprietário, o Sr. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, CPF 000.000.000-00, doravante denominada simplesmente CLIENTE.

**1. Objeto do Contrato**

Criação de sistema web voltado a [OBJETIVO DO PROJETO]. A URL (endereço do site) do projeto é: [URL DO PROJETO SE FOR WEB]. A alteração do nome do domínio futuramente, não impactará na natureza e escopo do projeto, logo, se a URL for modificada por qualquer motivo que seja, este contrato e seus anexos permanecem válidos, cabendo somente a alocação do conteúdo desenvolvido dentro da nova URL. O projeto inclui:

1. Criação de Layout personalizado, único e exclusivo, dentro dos padrões atuais de design;
2. Estruturação em HTML e CSS
3. Design responsivo, ou seja, adaptável a dispositivos móveis
4. O site já será desenvolvido com otimização On‐page, de forma que consiga atingir boas colocações no ranking do Google, de acordo com as palavras‐chave utilizadas nas buscas orgânicas dos usuários;
5. O site contará com um painel de controle para gestão dos conteúdos:
   1. Conteúdo a
   2. Conteúdo b
   3. Conteúdo c
   4. Conteúdo d
   5. Conteúdo f
   6. Conteúdo g
   7. …

**2. URL e conteúdo, data de início e término do contrato, horários de atendimento, host, e‐mail's, propriedade de conteúdo e direitos de cópia**

1. A URL registrada no Registro.br está descrita na cláusula 1. O Registro (que é realizado pelo site Registro.br) tem custo de R$ 40,00 anuais - cujo pagamento deverá ser feito diretamente ao próprio REGISTRO, pelo CLIENTE. O processo de registro do domínio (URL) poderá ser feito pelo PRESTADOR, em nome do CLIENTE, se este assim solicitar. O conteúdo do site será de acordo com *briefing* acordado entre as partes. Considera-se como *briefing* toda e qualquer mensagem de WhatsApp, Telegram, reuniões virtuais, mensagens de texto, e-mails, arquivos digitais ou físicos que tratem a respeito do projeto. O layout, após aprovado pelo CLIENTE, não poderá sofrer alterações. Em caso de alterações do layout, deverá ser solicitado um orçamento específico. Qualquer mudança ANTES da aprovação do layout será realizada pelo PRESTADOR como parte integrante deste contrato. A estrutura base do site poderá ser administrada pelo painel do cliente. Desta forma, o site será entregue sem conteúdos cadastrados. O CLIENTE definirá a data de lançamento do site após a finalização do projeto.

Para cada reunião, serão protocoladas as conversas, áudios e arquivos trafegados, que, em comum acordo, comporão anexos adicionais deste contrato e servirão de base oficial para direcionamento do projeto.

As ferramentas de acompanhamento (Kanban, Burndown Chart e outros artefatos do framework Scrum) também são parte integrante deste documento.

As páginas do sistema serão:

**VISÃO DE USUÁRIO DO SISTEMA**

* 1. A visão do usuário quanto ao sistema
  2. Continua…
  3. Continua…
  4. Continua…
  5. Continua…
  6. Continua…
  7. Continua…

**VISÃO DE ADMINISTRADOR**

* 1. Os poderes do administrador quanto ao sistema
  2. Os poderes do administrador quanto ao sistema
  3. Os poderes do administrador quanto ao sistema
  4. Os poderes do administrador quanto ao sistema
  5. Os poderes do administrador quanto ao sistema
  6. Os poderes do administrador quanto ao sistema
  7. Detalhes extras e especificações do sistema, constam no ANEXO I
  8. Outros detalhes podem ser adicionados desde que se mantenha o escopo global apontado no ANEXO I e registrado em texto por:
     1. E-mail
     2. Mensagens de comunicadores instantâneos de qualquer tipo

AQUI, RECOMENDO QUE USE O SEU ORÇAMENTO COMO ANEXO, OU SEJA, VOCÊ JÁ DEVE TER ESPECIFICADO TUDO NO ORÇAMENTO ENTÃO ELE ENTRA COMO ANEXO DO CONTRATO E VOCÊ APENAS O CITA NA HORA DE APROFUNDAR ALGUMA COISA SOBRE O SISTEMA. NO FINAL DESTE CONTRATO VOCÊ VÊ O ANEXO DESTE DOCUMENTO 9QUE FOI A PROPOSTA APRESENTADA AO CLIENTE).

Os apontamentos combinados por qualquer um dos meios descritos no item H (vai mudar de acordo com a sua lista de itens), desde que de comum acordo entre as partes, são válidos como acordo formal e parte integrante deste contrato, ainda que sejam combinados posteriormente à assinatura deste instrumento.

1. Este contrato se inicia na data de 31.01.2022 e será vigente pelo prazo do projeto, descrito na cláusula 5 e no ANEXO I, respeitando-se os devidos períodos extras previamente descritos na cláusula citada.
2. O horário de atendimento por parte do PRESTADOR é das 9:00h às 17:00h, de Segunda à Sexta‐feira (exceto feriados) através de e‐mail, comunicadores instantâneos, telefone ou pessoalmente caso necessário. O envio de documentos deverá ser impreterivelmente realizado via e‐mail ou comunicadores instantâneos.
3. A hospedagem do site poderá ser feita a encargo do PRESTADOR em VPS ou Hospedagem Compartilhada, conforme apontado no ANEXO I.
4. O endereço de e‐mail para contato entre o CLIENTE e o PRESTADOR é: email@docliente.com.br.
5. A propriedade sobre o conteúdo do site e toda a documentação gerada é do CLIENTE, bem como o direito de cópias e reproduções.

**3. O valor**

O valor fixado para o projeto é de R$ 00.000,00 (Preço por extenso) - já com desconto - e será liquidado da seguinte forma:

1. Pagamento ao início de cada Sprint, conforme especificado no ANEXO I. Ao todo serão 5 Sprints (5 pagamentos de R$ 00.000,00 (Preço por extenso), sendo o primeiro pagamento para o dia 31.01.2022).
2. O último pagamento ocorrerá ao final da quinta Sprint.
3. Os pagamentos ocorrerão nas seguintes datas:
   1. 31.01.2022 (R$ 00.000,00)
   2. 15.02.2022 R$ 00.000,00)
   3. 28.02.2022 (R$ 00.000,00)
   4. 15.03.2022 (R$ 00.000,00)
   5. 30.03.2022 (R$ 00.000,00)

**4. O direito das partes envolvidas**

O PRESTADOR compromete‐se a entregar o objeto deste contrato em tempo hábil, de acordo com os prazos fixados (desde que todo o material solicitado tenha sido devidamente entregue), bem como o CLIENTE compromete‐se a realizar quaisquer quitações financeiras que este mesmo objeto do contrato venha a onerar. O descumprimento de qualquer item deste contrato por qualquer uma das partes nele envolvidas acarretará na tomada de medidas cabíveis, bem como (se for o caso), cobranças extrajudiciais, judiciais e registro do débito junto ao Cartório de Protesto e órgãos de proteção ao crédito.

Em caso de descumprimento do contrato por parte do PRESTADOR:

1. **Por atraso da entrega:** Multa diária de 0,01% do valor do projeto.
2. **Por Entrega fora dos padrões:** Readequação do conteúdo de acordo com o briefing enviado até que esteja de acordo com o solicitado pelo CLIENTE. Caso o PRESTADOR não atinja os objetivos solicitados, o CLIENTE poderá solicitar os ajustes necessários, respeitando o escopo de cada área do site dentro do briefing previsto.
3. Em caso de descumprimento das cláusulas deste contrato, o PRESTADOR deverá ressarcir o CLIENTE sobre o valor do projeto pago, com desconto proporcional do material previamente desenvolvido, cujo montante não poderá ultrapassar 60% do valor total do projeto, independente do estágio em que esteja. Para tanto, haverá uma análise prévia de conteúdos para aferir a falta da entrega dentro dos padrões contratados.

Em caso de descumprimento do contrato por parte da CLIENTE:

1. **Por atraso no pagamento:** Acréscimo de 0,01% por dia de atraso sobre o valor do projeto.
2. **Por atraso no envio de materiais:** 2 dias úteis adicionados ao prazo de entrega.

Em caso de rescisão de contrato sem justificativa prévia, por qualquer uma das partes: Multa de 50% do valor do projeto (fixado na cláusula 3), com entrega dos materiais desenvolvidos até o momento da rescisão.

**5. Prazo para entrega dos serviços**

Os serviços serão entregues de forma gradativa (*wireframe*, *layout*, painel de controle, versão responsiva, etc). O prazo para entrega de cada um dos serviços será: dividido proporcionalmente dentro do prazo total do projeto, que é de 80 dias úteis. O PRESTADOR disponibilizará uma “burndown chart” - um gráfico de dias do projeto. Por meio deste gráfico, ambas as partes poderão acompanhar a evolução do projeto, bem como as entregas parciais. Eventuais aumentos ou diminuições de prazo serão também lançados neste gráfico. O envio do link para acesso ao Burndown Chart será feito via comunicadores instantâneos.

**6. Atualizações, publicação, otimização e modificações durante o processo de desenvolvimento**

1. As atualizações do site deverão ser realizadas pelo CLIENTE por meio de painel de controle disponibilizado pelo PRESTADOR (parte integrante do objeto do contrato).
2. b) A publicação do site se dará a partir do momento em que o CLIENTE aprovar a apresentação final dos sistemas e o layout de cada sessão do site. O CLIENTE poderá, ainda, solicitar a publicação a qualquer momento, mesmo que o site ainda não tenha conteúdos publicados. Para as modificações durante o processo de desenvolvimento, será acrescido ao prazo (cláusula 5), mais 2 dias úteis (para cada modificação). Esta extensão de prazos não representará atrasos na entrega.

**7. Do Foro**

Fica eleito o Foro da Comarca de \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / SP para dirimir quaisquer causas oriundas deste contrato, descartando‐se qualquer outro por mais privilegiado que seja. E por estarem justos e contratados, firmam o presente contrato em duas vias de igual teor.

Caraguatatuba, 31 de Janeiro de 2022.

| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  NOME DA EMPRESA CNPJ: 00.0000.000/000-00 CLIENTE | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Kopi Educacional - MD Santos Internet e Sistemas CNPJ:00.000.000/0001-00 PRESTADOR |
| --- | --- |
|  |
|  |  |

**ANEXO I**

**SOBRE O SISTEMA**

Plataforma web para\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**COMO LER ESTE DOCUMENTO**

Alguns termos técnicos serão utilizados durante a explanação desta proposta. Após a explanação, é possível acessar o Dicionário Técnico para compreender mais detalhes. Outros itens serão explicados dentro dos próprios tópicos.

**DOCUMENTAÇÃO**

A documentação do sistema será feita de forma incremental, ou seja, conforme as sprints forem sendo executadas, a documentação será atualizada (mais detalhes sobre “sprints” no dicionário técnico).

Ao final do projeto, a documentação será revista, atualizada e entregue como parte integrante dos documentos técnicos do projeto.

**LINGUAGENS ENVOLVIDAS**

As linguagens estão divididas em duas vertentes: front end e back end.

* Linguagens Front end
* HTML5
* CSS3 (Framework de estilização - Bootstrap)
* JavaScript (React / Vue / Angular)

* Linguagens Back end
* JavaScript (com interpretador Node.js / React / Vue / Angular)
* SQL
* jSon (API, caso seja necessário em algum momento)

**Porque JavaScript? (POC - Proof of Concept)**

Em 2018, em um teste de performance realizado no Conselho de Medicina do Estado de São Paulo, onde eu atuava como servidor público federal no cargo de analista de tecnologia da informação, fizemos uma prova de conceito para eleger a linguagem que seria adotada no desenvolvimento do novo sistema. As linguagens utilizadas nesta POC foram:

* JavaScript com Node.js
* Python
* PHP
* Java
* C#
* C++

Desenvolvemos um algoritmo com crescimento exponencial baseado em função recursiva e matriz multidimensional. O foco era justamente analisar o tempo de execução.

Montamos máquinas virtuais para que o ambiente de execução fosse igual para todas as situações.

Ao executar o algoritmo desenvolvido em linguagens diferentes (sem uso de bibliotecas ou frameworks), o JavaScript com Node.js teve desempenho expressivo em relação às outras linguagens. Em uma escala de tempo, em que **0** representa “**muito rápido**” e **10** representa “**muito lento**”, enquanto a maioria das linguagens ficou com desempenho acima de **5**, o javaScript ficou com desempenho na casa dos **0,5** (meio).

Por este motivo, visando a performance da aplicação, recomendo o uso de JavaScript com Node.js para que o sistema tenha uma estrutura ágil, robusta, leve e ao mesmo tempo escalável.

**RESPONSIVIDADE**

A aplicação será desenvolvida com o conceito de “site responsivo”, ou seja, ele será automaticamente adequado às plataformas móveis (celulares e tablets).

**FRAMEWORKS DE PROGRAMAÇÃO**

* ExpressJS

**BIBLIOTECAS**

* Serão utilizadas as bibliotecas nativas do Node.js

**BIBLIOTECAS DE IA (INTELIGÊNCIA ARTIFICIAL)**

* TensorFlow

Visando projeções futuras, o Node.js trabalha com uma biblioteca específica para aprendizado de máquina (machine learning), a TensorFlow. Em um cenário de futura escalabilidade do sistema para uma inteligência própria, a linguagem já estaria totalmente adequada para suportar a expansão.

**BANCO DE DADOS RELACIONAL**

A abordagem de um banco relacional é o ideal para este início, mas é importante entender que em caso de expansão massiva do projeto, um banco Não Relacional (NoSQL - Not Only SQL) seria a abordagem mais adequada.

Desta forma, o projeto do Banco de Dados Relacional será desenvolvido já pensando em uma futura expansão, caso ela ocorra.

**AMBIENTE DE DESENVOLVIMENTO**

Será cedido um endereço virtual para acompanhamento do projeto. Esta URL deverá ser acessada conforme as demandas forem sendo publicadas. Os endereços serão fornecidos ao final da primeiro Sprint e serão utilizados até o final do projeto.

**BACKUP**

Será disponibilizada uma cópia de segurança (Backup) a ser sincronizada diariamente com o projeto em desenvolvimento, de forma que uma cópia incremental esteja sempre disponível para download por parte do Stakeholder.

**FRAMEWORK DE GESTÃO DE PROJETO**

Scrum. O Scrum é framework mas é tratado como metodologia de desenvolvimento de projetos (não apenas na área de TI). Sua versatilidade permite que ele seja adequado a cenários diversos, embora ele tenha surgido com foco em cenários de computação.

O Scrum é rápido e visa contato praticamente diário com os Stakeholders (aqui, leia-se “cliente”) para posicionamento constante do status do projeto. Isso faz com que o projeto não “saia dos trilhos”, uma vez que o cliente estará acompanhando cada evolução e opinando sobre os passos seguintes.

A ideia do Scrum é trabalhar com o método de “time-boxed”, ou seja, tempo fechado. Depois de definido um prazo, ele pode ser reduzido, mas nunca ampliado. Isso garante a qualidade da entrega em tempo ágil.

Cada ciclo de entregas ao Stakeholder é feito em períodos de 15 dias (chamado de Sprint).

**FERRAMENTAS E EVENTOS DE ACOMPANHAMENTO**

* Kanban (para acompanhamento das tarefas diárias)
* Burndown chart (para uma visão holística do projeto em detrimento ao prazo
* User Stories (para descrição de funcionalidades do sistema)
* Daily Meet (apenas para a equipe de desenvolvimento, para acompanhamento diário do desenvolvimento da aplicação)
* Sprint Review (apresentação do conteúdo desenvolvido em 15 dias)
* Sprint Retrospective (fechada à equipe de desenvolvimento para alinhamento dos gargalos da Sprint atual)
* Sprint Planning (planejamento da próxima Sprint com participação opcional do Stakeholder)

**PARADIGMA UTILIZADO**

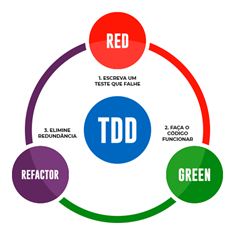
* POO - Programação orientada a Objetos.

**METODOLOGIA DE DESENVOLVIMENTO**

* TDD - Test Driven Development

É uma metodologia de desenvolvimento orientada a testes. Geralmente o sprogramadores usam uma abordagem mais convencional de criação de código: Eles criam o algoritmo e depois testam.

Com TDD, a abordagem é inversa: Criam-se cenários de testes e em cima desses cenários, são criados os algoritmos.



Um dos pontos interessantes desta abordagem é a forma que ela funciona:

* Primeiro, faz-se um algoritmo com o objetivo de não passar no teste (RED)
* Em seguida, cria-se um algoritmo com o objetivo de passar no teste (GREEN)
* Por fim, o código é refatorado para atender ao requisito

Esta abordagem diminui drasticamente os testes não realizados e situações de falhas ou erros.

**TESTES UNITÁRIOS**

Embora os testes sejam feitos durante o desenvolvimento (TDD), após o fechamento de cada módulo, será executado o teste de unidade. Este teste, agrupa funcionalidades que compõem um mesmo rol de tarefas e as testas de maneira individual e depois como um elemento único, de forma a ver se entre aquelas micro partições, existem pontos de falha. Estes testes serão executados periodicamente, ao final de cada novo conjunto de tarefas.

Tanto a abordagem TDD quanto a de teste unitário envolve uma razão matemática entre Custo e Tempo.

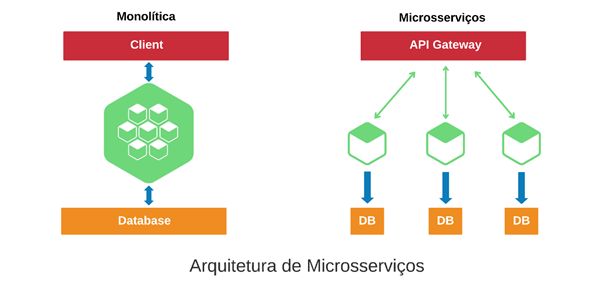
Por ser uma abordagem mais demorada no desenvolvimento, consome mais tempo no início do projeto, gerando a impressão de o projeto estar mais oneroso no início.

Após 60% do projeto feito, o prazo de integração das unidades que compõem o sistema torna-se mais rápido, pois já foram previamente testados, ocasionando um menor tempo de produção e consequentemente, um custo mais baixo com refatoração de código.

**MICROSSERVIÇOS**

O sistema será desenvolvido com uma visão mais holística, ou seja, enxergando o todo da aplicação, mas terá também uma visão específica sobre cada micro segmento. O motivo disso é prover a escalabilidade do projeto no futuro, caso seja necessário.

Desta maneira, embora o sistema trabalhe como um todo, será utilizada uma arquitetura voltada a micros serviços, que consiste basicamente em utilizar o conceito de “alta coesão e baixo acoplamento”, ou seja, o sistema funcionará como um todo, mas poderá ser desmembrado para que cada partícula funcione de forma individual, inclusive, com linguagens diferentes (que é o foco do micros serviço).



Para o caso de utilizar micros serviços no futuro, o uso de APIs em jSon será necessário.

**ENTREGÁVEIS**

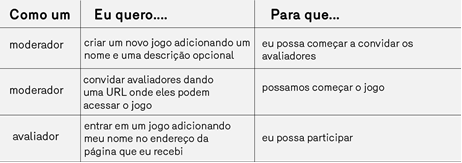
Cada entregável será feito ao final de cada Sprint (15 dias). Entenda como “entregável”, algo utilizável dentro do sistema, seja no front ou back.

**DATAS E PRAZOS**

Cada Sprint tem, por padrão, 15 dias. Caso haja exceção, expandido para 30 dias, isso será informado ao Stakeholder no início de cada Sprint.

**ENTREGAS PARCIAIS**

O início do projeto se dá com a elicitação de requisitos (User Stories). Essas Histórias do usuário são o guia que orientará a equipe de desenvolvimento nas etapas de programação. O Stakeholder deverá, juntamente com o Scrum Master (Marcio Santos), apontar todas as histórias do usuário, que descreverão o sistema como um todo. Paralela a essa etapa, o Stakeholder deverá apontar a prioridade de cada item, isso guiará a equipe de desenvolvimento na hora de priorizar tarefas. Abaixo, segue um modelo de User Stories (em reunião oficial, será mostrado um modelo similar a este que deverá ser preenchido pelo Stakeholder e Scrum Master).



Cada item da lista será priorizado com números de 1 a 10. A equipe de desenvolvimento irá sempre priorizar as tarefas com os números mais altos.

**PRAZO ESTIMADO PARA FINALIZAÇÃO DO PROJETO**

**000 dias**, sendo **5 sprints** sequenciais e um **adicional de 5 dias, que é o intervalo entre uma Sprint e outra**. O prazo é compatível com a estrutura oferecida, todavia, caso o escopo do projeto seja expandido (o que é aceito pelo framework Scrum), aditivos deverão ser anexados ao contrato de prestação de serviços, de forma a explicitar o acréscimo de prazo independente do serviço inicialmente contratado.

**VALOR DO PROJETO**

* O valor inicial¹ orçado é de R$ 00.000,00
* Sinal de R$ R$ 00.000,00 para iniciar a primeira Sprint (após assinatura do contrato)
* As datas de pagamento definidas são:
  + 31.01.2022 (R$ 00.000,00)
  + 15.02.2022 R$ 00.000,00)
  + 28.02.2022 (R$ 00.000,00)
  + 15.03.2022 (R$ 00.000,00)
  + 30.03.2022 (R$ 00.000,00)

Forma de pagamento: ao final de cada Sprint (excetuando-se a primeira, em que ocorre o pagamento inicial para o *start* do projeto), pois sempre ao final de uma Sprint, existe um entregável de valor ao Stakeholder.

Com o Framework Scrum, o projeto torna-se mais maleável, o que significa que ele aceita incrementos de recursos. Esses incrementos resultarão em um aditivo contratual a ser negociado no momento de adesão.1

**MANUTENÇÃO MENSAL**

* Valor **0000**% de um salário-mínimo, com reajuste anual conforme percentual proporcional do salário mínimo.
* Engloba
* Manutenção funcional do sistema
* Adição, remoção ou refatoração de recursos
* É importante ressaltar que uma manutenção não simboliza a criação de novos sistemas totalmente independentes da estrutura atual. Para tanto, considera-se como manutenção, toda e qualquer manipulação da estrutura publicada com limitação de até 20 horas/mês

**FORMA DE PAGAMENTO E QUESTÕES ADMINISTRATIVAS**

* Pagamento realizado via PIX ou Boleto bancário;
* Nota Fiscal emitida sempre no 1º dia útil do mês subsequente à(s) Sprints realizada(s); será emitida apenas uma nota para cada mês anterior, independentemente da quantidade de Sprints que aquele mês tenha.
* Ex.: Pressuponha que a Sprint 1 iniciou no dia 05/12. Ela durará até o dia 20 (15 dias). A próxima Sprint iniciará no dia 21/12 e vai finalizar apenas no dia 06/01. Neste caso, no dia 01/01\*, será emitida uma Nota Fiscal de Prestação de Serviços referente à Sprint finalizada em 20/12.

*\* Vale a ressalva que é sempre no 1º dia útil, e não propriamente no dia 01 do mês.*

Quebra de Página

**DICIONÁRIO TÉCNICO**

**API - Application Programming Interface** - É um elo entre duas tecnologias distintas. Serve para conectar uma aplicação A a uma aplicação B, ainda que essas estejam desenvolvidas em linguagens diferentes.

**Biblioteca** - executa tarefas específicas dentro da aplicação. Um exemplo: a validação de um CPF

**Back end** -São aquelas utilizadas para construir a lógica das regras de negócio, banco de dados, APIs, etc.

**Burndown chart** - É um gráfico em que o cliente acompanha o ponto em que estamos no projeto, em relação ao todo. O eixo X é a quantidade de entregas. O eixo Y é o prazo.

**Daily Meet** - É uma reunião diária, com duração máxima de 15 minutos, entre a equipe de desenvolvimento, guiada pelo Scrum Master (Marcio Santos).

**Framework** - É um conceito para ser utilizado. pode ser um conjunto de recomendações (como o SCRUM) ou um conjunto de códigos (como o ExpressJS)

**Front end** - As linguagens front, são aqueles que lidam com os aspectos visuais e geralmente interagem com o usuário. Em suma, em um site, as linguagens de front são responsáveis por exibir em tela tudo o que o usuário vê e interage.

**Kanban** - É um quadro de tarefas em que se organizam as atividades em, pelo menos, 3 colunas: Coisas a Fazer, Coisas sendo Feitas, Coisas Feitas.

**POO - Programação Orientada a Objetos** - É um paradigma que reduz consideravelmente o tempo de codificação, uma vez que um mesmo código poderá ser aproveitado em outras etapas do sistema.

**Refatoração** - Processo de reescrita de código visando uma melhor performance. Um mesmo algoritmo (instruções computacionais) pode ser escrito de N formas diferentes, entretanto, dependendo da estrutura escolhida, o desempenho (velocidade de processamento e alocação de memória) pode mudar. A refatoração preza sempre pela melhor performance absoluta.

**Scrum Master** - É a pessoa que mais entende do framework Scrum dentro do projeto. Ela lida com a equipe de desenvolvimento e mantém o time dentro dos prazos, escopo, qualidade do projeto e nas normas do Scrum. Existe um outro papel no Scrum, denominado PO (Product Owner). Esse papel de P.O. será exercido pelo Sr. José Pacheco.

**Sprint Review** - Ocorre ao final do 15º dia de desenvolvimento daquela Sprint. Nesta reunião o Stakeholder precisa participar para fazer as aprovações e ponderações para a próxima Sprint. Ocorre ao final de cada Sprint.

**Sprint Retrospective** - Momento em que a equipe de desenvolvimento analisa o desempenho da Sprint atual para fazer ajustes e traçar estratégias para a próxima Sprint. Ocorre ao final de cada Sprint.

**Sprint Planning** - Momento de planejamento da próxima Sprint. Necessária a participação do Stakeholder. Ocorre ao final de cada Sprint, antes de iniciar a próxima.

**Stakeholders** - são os interessados pelo projeto. Pode ser o cliente, o investidor, o patrocinador.

**User Stories** - É a coleta de requisitos, onde o cliente define: o que ele quer, para que ele quer e como aquele recurso irá se comportar.