**Avaliação Técnica de Engenharia de SW - Estágio**

**Rede D’Or - Arquitetura e Engenharia de Software Corporativas**

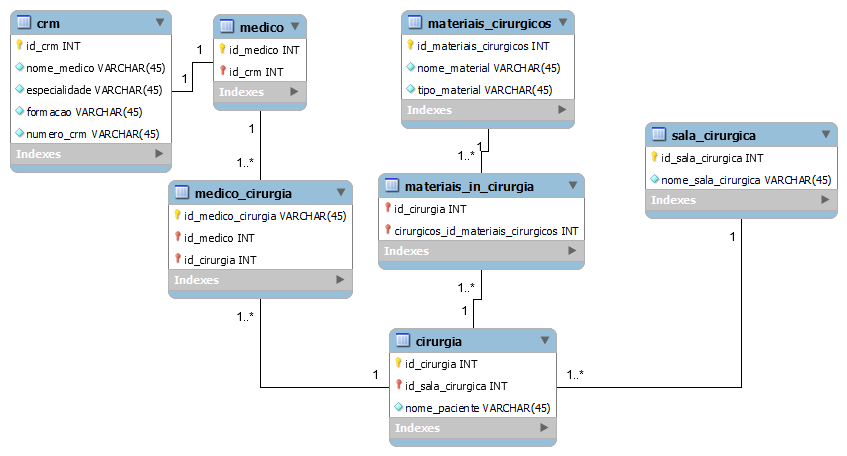
Responda cada questão abaixo de forma a expressar seu conhecimento técnico, adicionando imagens, trechos de códigos e demais artefatos que julgar necessário.

**Questão 1: Modelagem de Dados** Considere a seguinte situação:

1. Um hospital possui múltiplas salas cirúrgicas.
2. Um médico realiza várias cirurgias. Uma cirurgia é realizada por um ou mais médicos em uma única sala cirúrgica.
3. Uma cirurgia pode conter um ou mais materiais.

Como você abordaria a modelagem de dados nesta situação?

**Resolução:**



# Questão 2: Automação de Processos

Elabore um processo em que você usaria uma ferramenta low-code (*Ex.: ServiceNow App Engine, MS Power Platform e Outsystems*) para automatizá-lo. Explique o porquê este processo pode ser automatizado com ferramenta low-code e quais seriam os principais passos automatizados do processo.

**Resolução:**

Considere um restaurante onde um cliente entra no restaurante, senta-se em uma mesa, solicita um garçom um cardápio e realiza seu pedido. Uma solução para automatização desse processo seria uma aplicação onde o cliente senta-se a mesa e já tem disponível um dispositivo integrado com o cardápio, acompanhando os pratos disponíveis e seus ingredientes, o próprio cliente faz o pedido via tablet, onde durante o próprio dispositivo é possível acompanhar o status de desenvolvimento de seu prato.

Dado esse processo de automatização, uma ferramenta low-code pode contribuir para esse processo da seguinte forma:

**1° Agilidade no desenvolvimento e manutenção do aplicativo:**

Com uma plataforma low-code seria reduzido drasticamente o tempo de desenvolvimento do sistema e teria uma manutenção facilitada. Com isso o tempo de disponibilidade da aplicação é elevada, pois o processo para uma atualização, manutenção de rotina, ou manutenção de emergência terão o tempo reduzido, garantindo assim para o restaurante e seus clientes que o serviço estará disponível por mais tempo.

**2° Integração de Dados e Sistema:**

Uma plataforma low-code pode ser integrada facilmente com diversos sistema, e isso seria um benefício para integração do sistema com o sistema de gestão de um restaurante. Garantindo assim ao restaurante fácil acesso as informações de estoque e garantindo ao cliente um sistema com confiabilidade dos dados recebidos de seus pratos, ingredientes e status de seus pedidos.

**3° Automatização do pedido:**

Com a plataforma low-code o cliente não ficará a mercer da disponibilidade dos garçons disponibilizados para o atendimento, com o processo de automatização o cliente realiza seu pedido pelo próprio aplicativo que confirmando o pedido é encaminhado para um sistema integrado com a cozinha onde receberam o pedido e iniciaram a elaboração do prato desejado.

**4° Notificação em tempo real:**

Feito o pedido o cliente poderá ir acompanhando a elaboração de seu prato via aplicativo. Dentro do próprio aplicativo poderá receber notificações como: pedido recebido, em preparo, pronto para servir, e demais notificações que o restaurante compreenda ser necessário para deixar seus clientes informados.

**5° Análise de Dados:**

O aplicativo através de uma plataforma low-code pode disponibilizar ao restaurante feedbacks necessários sobre a sazonalidade de seus pratos, padrões de consumo, horários e dias de picos para o restaurante se preparar se fornecer aos seus clientes um prato com maior padrão de qualidade.

# Questão 3: Interoperabilidade

Uma clínica de saúde deseja permitir que pacientes agendem consultas em um website parceiro. Especifique uma API que realize o agendamento e receba os dados do paciente, médico, convênio e plano.

# Questão 4: Lógica de Programação

Desenvolva um trecho de código (Em qualquer linguagem, mas preferencialmente JavaScript) para verificar se pacientes devem ser encaminhados para urgência ou emergência.

Requisitos:

1. Defina uma coleção de objetos representando pacientes, com os seguintes atributos: “nome”, “idade” e “condição”
2. Escreva um método que identifique na coleção, segundo o critério a seguir, se o paciente deve ser encaminhado para urgência ou emergência:
   1. Pacientes com dores agudas no peito, fraturas expostas, acidentes de trânsito, ferimentos à bala ou com idade superior a 70 anos devem ser encaminhados para emergência.
   2. Pacientes com fraturas não expostas, cólicas renais, luxações e torções devem ser encaminhados para urgência.

# Questão 5: Qualidade, Agilidade e Observabilidade

Descreva quais práticas você adotaria para garantir a qualidade do código e dos componentes low-code, agilidade em entregar mudanças para os usuários e capacidade de prevenir incidentes em um ambiente low-code.

**Resolução:**

Considerando as boas práticas de programação e com o objetivo de garantir a qualidade do código e componentes temos que pensar em inúmeras questões que podem contribuir para o processo. O princípio de tudo deve ser o planejamento, elaborar um bom planejamento pensando claramente nos requisitos, integrações que serão necessárias com outras plataformas, funcionalidades que a aplicação necessita e nos recursos que a ferramenta low-code poderá contribuir para o desenvolvimento da aplicação.

Com tudo planejado devemos iniciar o desenvolvimento, não esquecendo de seguir o planejamento pré-definido, com isso em mente para um bom desempenho no desenvolvimento seria conhecer a plataforma low-code e recursos que a plataforma pode nos oferecer com o objetivo de utilizar o máximo do que a plataforma tem a nos entregar, com isso devemos sempre ficar atentos com os componentes á disponibilizados pela plataforma.

Com tantas possibilidades fornecidas pelas plataformas low-code há uma tentação de criar funcionalidades excessivas que fogem do escopo do projeto, ou até mesmo incrementar uma funcionalidade necessária de forma excessiva. Uma questão que deve ser avaliada durante o desenvolvimento da aplicação seria a simplicidade do código, pensando em entregar bem-feito tudo o que o escopo do projeto já definiu na etapa de planejamento, com isso em mente devemos compreender que mais vale uma função simples e eficiente que cumpre os requisitos do escopo do que uma funcionalidade complexa, com inúmeras dependências que não foi prevista e que poderá causar maiores dificuldade na etapa de manutenção de um código.

Durante o processo de desenvolvimento não podemos deixar de pensar na documentação da aplicação, uma documentação bem-feita pela equipe de desenvolvimento contribui para a eficiência do desenvolvimento das aplicações, pois assim, todos integrantes da equipe sabem exatamente a importância de cada funcionalidade e como ela funciona. Além de garantir a eficiência durante o processo de desenvolvimento a documentação contribui para manutenibilidade do código, garantindo um maior tempo de vida ao código.

Com todas essas práticas sendo seguidas fielmente, por último uma boa prática que deve ser realizada seria os testes regulares da aplicação. Durante o processo de testes serão encontradas questões a serem discutidas, bugs nas funcionalidades das aplicações, além de garantir que os requisitos definidos pelo escopo do projeto estão sendo cumpridos conforme definidos pelas regras de aceitação.

Considerando todas essas boas práticas quando adotadas pela equipe, a eficiência da equipe tende a crescer muito, durante o processo de desenvolvimento e durante o processo de manutenção da aplicação.