|  |  |
| --- | --- |
| **ATIVIDADE** | **SITUAÇÃO PROBLEMA FORMATIVA** |
| **CONTEXTO** | |
| Resultado de imagem para desenho professor Resultado de imagem para desenho serviço social  Chega até você um novo projeto a ser realizado, desta vez de um sistema escolar, mais abrangente e que deverá substituir os procedimentos já utilizados pela escola, exigindo de você uma maior polivalência.  Seu novo cliente é a ***Hogwarts University***, que almeja uma troca em seu sistema escolar objetivando a adição de novos recursos e também sobretudo a obtenção de uma solução de software mais performática, visto que a plataforma atual é muito caracterizada pela sua lentidão.  Neste projeto, será necessário ter um cuidado apurado na criação da modelagem do Banco de Dados (camada de persistência da aplicação) para que possibilite o sucesso no desenvolvimento Backend e Frontend que será feito por outro time.  Hogwarts School of Witchcraft and Wizardry letters photo – Free Certificate  Image on Unsplash  Como pode perceber, este projeto é mais desafiador, portanto, você precisará desenvolvê-lo usando ferramentas e tecnologias atuais de desenvolvimento, como GitHub, Docker e Azure Cloud.  **ETAPA 1 - O PROJETO**  ***AGENDAMENTO DE AMBIENTES E CONFERÊNCIAS***  Hogwarts University é uma escola muito grande, com inúmeros alunos, cursos, eventos, etc, portanto, atualmente há uma necessidade de criação de um sistema online de agendamentos do auditório da escola, evitando colisões de horários de usuários que necessitem utilizar tal ambiente.  Do ponto de vista do projeto como um todo, este sistema deve conter as seguintes funcionalidades:  ***- Adicionar, Alterar ou Excluir Agendamentos, respeitando os privilégios de cada usuário;***  ***- Consultar Agendamentos existentes;***  ***- Impressão de Relatórios e Gráficos de Ocupação;***  ***- Histórico de Acesso / Agendamentos;***  ***- Envio Automático de Email após agendamento, como lembretes;***  ***- Sistema online;***  As telas abaixo referem-se a uma ideia de layout desejada pelo cliente:        De forma mais clara, sua missão é criar a estrutura mais adequada de banco de dados que permita que os dados que estão contidos atualmente neste escopo possam ser utilizados pela equipe de desenvolvimento de backend + frontend poderem codificar suas respectivas partes.  Abaixo se encontram alguns requisitos importantes para a aplicação, aos quais você deve analisar e verificar se o seu projeto de banco de dados está preparado para contemplá-los:  ·       Na aplicação devem existir cadastramento de usuários, com dados principais dos mesmos, como nome, email, data de nascimento, senha, data de cadastro e mais alguns dados que julgar pertinente;  ·       Deve existir distinções entre as ocupações dos usuários na escola, ou seja, coordenador, orientador, assistente social, secretária, etc…;  ·       Também deve existir na aplicação algumas regras de acesso, como:  “Admin” – pode alterar tudo na aplicação, possui o maior nível de acesso;  “Gestor” – nível inferior ao admin, mas com acesso pleno ao cadastro de cadastro de usuarios, cadastro de ambientes, pode cancelar um evento, emitir relatório, etc.  “Usuário” – pode fazer reservas, mas não cadastrar usuários e nem cancelar eventos  “Visitante” – pode se inscrever em um evento participante;  Observe que as regras de acesso acima devem ser possíveis de ser atreladas ao tipo de ocupação de um usuário de forma flexível em seu projeto.  ·       Cada local / ambiente cadastrado deve possuir algumas informações pertinentes, dentre elas o nome do local, bloco da universidade (blocos existentes são A,B,C,D), e também a ocupação do local em quantidade de pessoas;  ·       Além das informações mencionadas anteriormente, cada local deve ter um check list do que é presente ou não no mesmo, porém este check list deve ser flexível, podendo ser adicionado ou não na lista de perguntas dentro do sistema, por exemplo:  Laboratório de Eletrônica 1:   * Projetor * TV Smart * Ar condicionado * WorkStation   ·       Este projeto, além do agendamento de ambientes, também deve contar com a possibilidade de check-in dos participantes do evento, ou seja, os usuários cadastrados podem entrar na aplicação e escolher qual evento desejam participar, reservando seu bilhete de entrada;  ·       Quando o usuário realizar o check-in em um evento, deve-se gerar um ticket para o mesmo contendo a informação do usuário e do evento, mantendo o histórico dessa informação (não está no escopo reserva de assentos, apenas da presença do participante).;  ·       Após a realização do check-in, a quantidade disponível de vagas dentro do evento deve ser diminuida (isso deve ser feito via procedure, não se preocupe com este requisito, mas planeje com que o evento possa guardar essa informação);  ·       Ao se cadastrar um evento, deve ser possível também programar quanto tempo antes do início do evento pode-se realizar o check-in e o tempo máximo para o mesmo;  Problemática a ser considerada: Nesta aplicação os usuários cadastrados podem um dia ser também excluídos, porém, ao se excluir um usuário que já participou de eventos, seu ID está na base de dados de eventos e portanto haverá problemas ao tentar realizar esta exclusão visto que este relacionamento entre chaves ficará faltante. Desse modo, como você resolveria este problema? Ou seja, como poder eliminar com que usuários não mais autorizados usem seu sistema sem no entanto eliminar todo o o histórico que eles possuem em sua aplicação??? Implemente sua solução já no projeto.  Atente-se aos detalhes e dicas abaixo:  ·       Crie este banco de dados localmente em um container Docker;  ·       Após o desenvolvimento completo da atividade, faça a criação deste banco usando sua conta na Azure Cloud e se conecte no banco, colando os comandos de criação que você utilizou para deixar a estrutura do banco já criada no ambiente em cloud (conecte-se no banco usando o bastion);  ·       Após finalizado o código de criação, coloque-o em um repositório git para disponibilizar para o time de desenvolvimento;  Etapa 2:  Antes de finalizar por completo o seu projeto, você precisa garantir que sua estrutura de banco de dados está devidamente elaborada permitindo com que se faça algumas consultas que o backend vai necessitar obter, portanto, adicione dados fictícios em suas tabelas e crie as seguintes consultas:  ·       Crie uma consulta que mostre todos os usuários que foram cadastrados em um determinado período mas que contenham ao mínimo dois check-ins;  **ETAPA 2 – BASE DE DADOS DUMP IDESP**  ·       Faça a importação do arquivo .zip DumpIDESP, referente ao Indice de desenvolvimento das escolas de SP;  ·       Crie uma consulta para mostrar qual a média por municipio da nota idesp\_ef\_iniciais, nota\_idesp\_ef\_finais e nota\_idesp\_em;  ·       Crie uma consulta para mostrar qual a média por municipio e por ano da nota idesp\_ef\_iniciais, nota\_idesp\_ef\_finais e nota\_idesp\_em;  ·       Crie uma consulta para mostrar qual a média da composição conjunta por municipio da nota idesp\_ef\_iniciais, nota\_idesp\_ef\_finais e nota\_idesp\_em; | |
|  | |

|  |  |  |  |
| --- | --- | --- | --- |
| ELABORAÇÃO | DATA | APROVAÇÃO | DATA |
| **André Felipe Savedra Cruz** | **15 / 05 / 2023** |  | **/ /** |