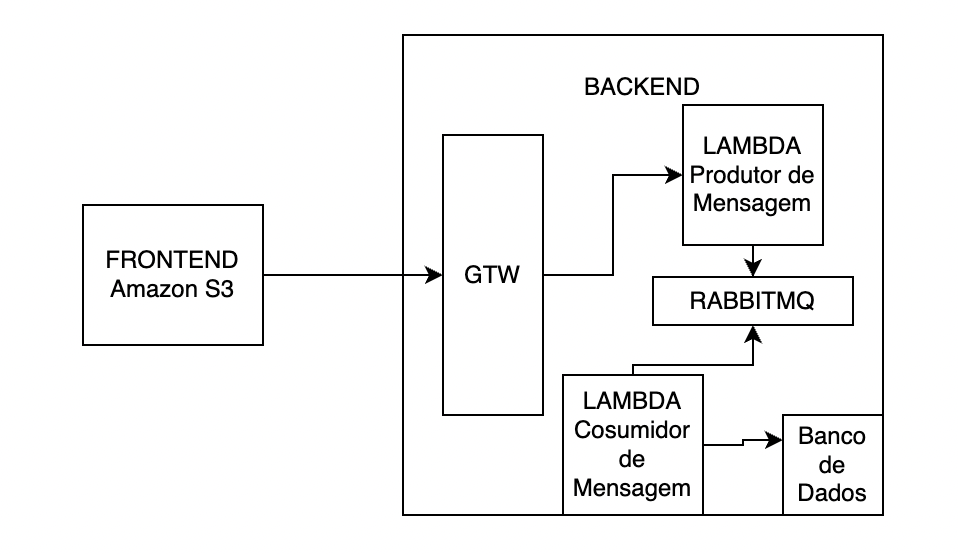
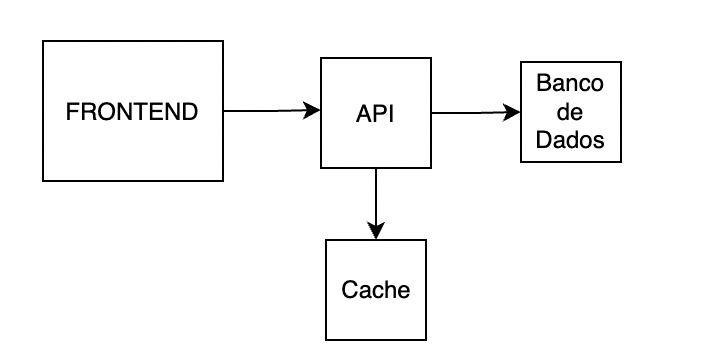
1 - Vou utilizar para essa respostas os serviços da AWS, mas a solução se aplica a qualquer outro provedor de nuvem. Como ficou definido que não poderá ser mais utilizado arquivos para a comunicação, minha ideia é criar um sistema onde os Hospitais cadastram os exames a medida que eles forem sendo realizados. O sistema será composto por um frontend hospedado na Amazon S3, onde os hospitais registram os exames, como iremos utilizar. O front chamará a rota de cadastro de exames expostas em um API Gateway que vai direcionar as chamadas para uma Lambda Function. A Lambda Function irá publicar as mensagens em um sistema de filas, que no caso será o RabbitMQ. Uma outra Lambda Function será a consumidora da fila, que vai ler a mensagem e gravar em um banco de dados.



2 - Para reduzir o número de consultas no banco de dados e visto que não será possível nenhuma alteração de hardware, eu implementaria um sistema cache rodando dentro do servidor da API. Atualizações nos exames serão sempre refletidas no cache e os exames permanecerão no cache em um período condizente com as características do negócio. Todas as requisições dos usuários serão direcionadas primeiro para o cache e caso não encontre dai sim irá chamar o banco de dados.



3 - A primeira solução por ser assíncrona e auto escalável será altamente disponível, para ficar indisponível seria apenas se os próprios serviços do fornecedor caíssem, mesmo assim é interessante algum serviço de monitoramento da saúde das aplicações deve ser implementado.. Sobre o aspecto da qualidade, não podemos correr o risco de perder nenhuma mensagem, por isso mecanismos de retry e fila hospital devem ser implementados. Para segurança implementarmos uma camada de autenticação que assegurem que o usuário está autenticado e tenha permissão para fazer o processo.

A disponibilidade da segunda solução estará assegurada com a utilização de servidores de tamanhos e quantidade condizentes com o negócio, além disso um sistema de monitoramento da saúde da aplicação deve ser implementado. Sobre a qualidade é fundamental assegurar que o cache esteja sempre atualizado com o estado atual de cada exame, para não termos furos nas informações fornecidas aos usuários. Para segurança implementarmos uma camada de autenticação que assegurem que o usuário está autenticado e tenha permissão para fazer o processo.