Relatório de issues

1. Inicialmente o sistema começou a ser desenvolvido usando funções e não classes, assim quando houve a migração para classes em alguns pontos não foi feito o objeto da classe importada, assim impossibilitando usar funções de outros arquivos.
2. A não familiaridade com o Sequelize nos levou a cometer alguns erros que em algumas documentações de versões mais antigas indicavam ser correto como a utilização direta do model e na versão atual é usado um index para acessar o model especifico.
3. Inicialmente foi pensado em gerenciar sessões pelo backend mas a medida que um pequeno progresso ocorreu percebemos a viabilidade de tal controle ser feito no frontend e o back gerenciava os acessos via token autenticando-o.
4. O sequelize tende a colocar as tabelas no banco no plural, como não sabíamos disso houve inicialmente alguns problemas na identificação das tabelas dando uma serie de erros que foram corrigidos posteriormente.
5. Quando começamos a modelar o banco não existia a coluna “createdAt” ou “updatedAt” nas tabelas, isso gerou problemas pois o sequelize puxa automaticamente tais colunas e como não havia dava erros impossibilitando interação com o banco.
6. Erro no arquivo migrate, ao testar foi verificado tags em aberto que comprometeu a sequencia do código seguinte.
7. Erro no datatype do migrate pois como parâmetro tava sendo passado “sequelize” e em alguns pontos foi buscado “datatype”, não existindo tal variável travava a execução.
8. Backend estava redirecionando usuários para paginas como se a navegação não fosse independente, após isso ser observado refizemos tais rotas para isolar as partes do código e manter o comportamento esperado.
9. Comando usado para migrar a estrutura do banco não estava adicionando campos a um banco já existente, sendo necessária a recriação do banco quando alguma alteração era feita.
10. Alguns campos de formulários no front foram removidos por não ter sido documentados previamente e o back ter preparado o request sendo diferente pois os dados que o banco aceitava eram outros, mas o problema foi logo identificado e corrigido.
11. Cors não foi configurado inicialmente e na hora da integração a api não fornecia o dados quando solicitado, após isso a configuração foi feita.
12. Banco não guardando dados de submissão\_respostas (as respostas nas perguntas), foi necessário criar outra tabela para poder criar a relação many to one relacionada a submissão.
13. Função para verificar caracteres especiais não funcionou inicialmente e após algumas analises e refatoração do código passou a cumprir o papel esperado.
14. Mais uma vez o problema do sequelize passar para o plural na hora de enviar a query para o banco ocorreu e transformava uma query para a tabela “Submissao” em “Submissaos”, código de criação das tabelas teve q ser revisto para evitar coisas assim.
15. Front precisou de uma rota para identificar o tipo de conta de usuário pra modificar o menu dependendo do tipo de conta, assim foi necessário a solicitação para a criação de tal rota não prevista.
16. Corrigido para aceitar múltiplas deficiências (inicialmente só aceitava uma).
17. Solicitado a criação de tabelas não prevista para armazenar o modelo do certificado, também algumas modificações na tabela de certificado como adicionar a coluna de qual evento ele pertence com seu relacionamento devidamente configurado.
18. Transformar odt para pdf estava dando muitos erros na biblioteca e não havia como resolver, até que foi percebido que ela só funciona em ambiente Unix. Então tivemos que buscar uma como alternativa para Windows.
19. Parte da equipe de desenvolvimento tinha pouco conhecimento em programação web, assim como pouca familiaridade com a linguagem usada, sendo necessário dedicar mais tempo no aprendizado que no desenvolvimento.
20. Problema com o endereço localhost do servidor e do site ao executar a WebView da aplicação Nativa.
21. Problema com Assincronicidade do react Js onde ele tentava ler certas variáveis que ainda não haviam sido preenchidas com as informações do Backend ocasionando em erros de valores não encontrados, ou tipo underfined.
22. Ao migrar o frontend para o mobile houve problemas com múltiplas renderizações, pois era comum durante o desenvolvimento do frontend, observar valores de variáveis para definir oque deveria ser exibido em tela, no mobile houve uma limitação maior, do numero de variáveis que poderia estar controlando oque seria exibido em tela.
23. Problemas de configuração do git. Ao usar os comandos padrões para se iniciar um projeto, tanto de reactJs quanto de react-native, eles geravam além dos documentos de estruturas para essas tecnologias, documentos de controle de versão do git e como todo o código, backend, banco de dados, front web e mobile eram trabalhados dentro de um único projeto git houveram certos problemas de sub repositório.
24. Foi necessária a utilização de bibliotecas de terceiros e desenvolvidas pela comunidade para a criação de componentes que não vem por padrão no React, como o Custom-select, para navegação entre telas, armazenamento de dados na memoria do navegador ou celular e a WebView.