Desenvolva uma tela de controle de documentos com as seguintes funcionalidades:

1. \*\*Upload de Arquivos:\*\*

- Permitir o upload de documentos (por exemplo, PDF, DOCX, XLSX, etc.).

- Realizar validação do tipo de arquivo e tamanho máximo permitido.

- Marcar com dropdown os usuários que podem acessar o documento (deixar opção todos como padrão), somente o empregador pode alterar quem terá acesso ao documento

- Exibir uma prévia ou o nome do arquivo após o upload, com feedback visual (ex: mensagem de sucesso ou erro).

2. \*\*Controle de Datas:\*\*

- Incluir campos para a inserção de datas importantes, como “Data de Criação”, “Data de Validade” e outras, se necessário.

- Utilizar componentes de seletor de data (date picker) para facilitar o preenchimento e garantir a integridade dos dados.

3. \*\*Responsáveis:\*\*

- Adicionar um campo para seleção ou inserção do responsável pelo documento.

- Esse campo pode ser implementado como um dropdown e com seleção de usuários registrados.

4. \*\*Interface e Experiência do Usuário:\*\*

- A interface deve ser responsiva, limpa e intuitiva, garantindo uma boa experiência mesmo em dispositivos móveis.

- Implementar feedback visual para operações críticas (ex: upload bem-sucedido, erros de validação).

- considerar o layout da tela de login

5. \*\*Funcionalidades Adicionais:\*\*

- Exibir uma listagem dos documentos já cadastrados, incluindo informações resumidas (nome do documento, data, responsável).

- Permitir ações como a visualização detalhada, edição e remoção dos documentos.

- Incluir filtros de pesquisa por data, responsável e nome do documento para facilitar a navegação e o gerenciamento.

6. \*\*Boas Práticas e Estruturação:\*\*

- Garanta a correta validação dos campos e tratamento de erros tanto no front-end quanto na comunicação com o backend.

- Considere a integração com o firebase/firestore.

- use os elementos reutilizáveis

Desenvolva uma gestão de comunicação focando na interação entre empregador, empregados e familiares, por grupo ou individual e com alertas e notificações. Deve contemplar as seguintes funcionalidades:

1. \*\*Comunicação em Grupo:\*\*

- Permitir o envio de mensagens de texto, imagens, vídeos e documentos nesses canais. Remover anexos após x dias

(o x será de acordo com o plano contratado)

- Implementar um sistema de notificações para alertar os membros sobre novas mensagens.

- Adicionar funcionalidades de moderação para garantir o bom uso dos canais.

2. \*\*Comunicação Individual:\*\*

- Permitir a troca de mensagens privadas entre usuários (empregador-empregado, empregado-familiar, empregador-familiar, etc).

- Implementar notificações em tempo real para novas mensagens.

- Possibilitar o envio de anexos (documentos, imagens, vídeos etc) nas mensagens privadas. Remover anexos após y dias. (o y será de acordo com o plano contratado)

3. \*\*Gestão de Alertas:\*\*

- Criar um sistema de alertas configuráveis para eventos importantes (ex: aniversários, prazos de entrega, comunicados internos).

- Permitir que os alertas sejam enviados por SMS e notificações push.

- Implementar cards no dashboard alertas não lidos

- crud para alertas.

4. \*\*Integração com Dados do Usuário:\*\*

- Integrar o sistema de comunicação com os dados dos usuários (empregados, familiares) para facilitar a identificação e o envio de mensagens.

5. \*\*Interface e Experiência do Usuário:\*\*

- A interface deve ser intuitiva e responsiva, adaptando-se a diferentes dispositivos (web e mobile)

- Implementar um sistema de busca eficiente para encontrar mensagens e alertas.

- Utilizar design moderno e agradável para melhorar a experiência do usuário.

- Considere usar o layout do login

Desenvolva o gerenciamento de tarefas com foco nas atividades domésticas, compras e compromissos financeiros com acompanhamento de tarefas por usuário responsável ou equipe, com as funcionalidades:

1. \*\*Crud de Tarefas:\*\*

- Permitir a criação de tarefas com título, descrição detalhada, data de vencimento, prioridade (alta, média, baixa) e status (pendente, em andamento, concluída, cancelada).

- Possibilitar a atribuição de tarefas a um ou mais usuários.

- indicação de concluída, pendente, iniciada, não iniciada, cancelada etc

- Adicionar campos personalizados para informações adicionais (ex: custo estimado, tempo estimado, foto, audio ).

2. \*\*Organização e Priorização:\*\*

- Implementar um sistema de listas ou quadros Kanban para organizar as tarefas por status.

- Permitir a criação de categorias ou tags para classificar as tarefas.

- Implementar filtros e ordenação para facilitar a busca e a priorização de tarefas.

- incluir conceito de tarefa recorrente por n períodos (DIAS, MESES ETC)

3. \*\*Acompanhamento e Notificações:\*\*

- Enviar notificações por sms e push quando uma tarefa é atribuída, tem sua data de vencimento próxima ou é atualizada.

- Permitir que os usuários comentem nas tarefas para facilitar a comunicação e o acompanhamento.

- Implementar um histórico de atividades para registrar as mudanças realizadas nas tarefas.

4. \*\*Colaboração em Equipe:\*\*

- Implementar um sistema de compartilhamento de tarefas entre os membros da equipe.

5. \*\*Interface e Experiência do Usuário:\*\*

- A interface deve ser intuitiva e responsiva, adaptando-se a web e mobile

- Utilizar drag-and-drop para facilitar a organização das tarefas.

- Implementar um sistema de busca eficiente para encontrar, ordenar e filtrar tarefas e responsáveis.

6. \*\*Relatórios e Métricas:\*\*

- Gerar relatórios sobre o progresso das tarefas, o tempo gasto em cada tarefa e a produtividade dos empregados

7. \*\*Integração com Outras Ferramentas:\*\*

- Permitir a integração com calendários (ex: Google Calendar, Outlook Calendar) para sincronizar as datas de vencimento das tarefas.

- Integrar com ferramentas interna de comunicação para enviar notificações e alertas.

- considerar usar layout da tela de login

- incluir no menu do dashboard

Prompt Completo para Sistema de Gestão de Ponto Eletrônico com Firebase

1. Visão Geral

Desenvolver uma funcionalidade para gestão de ponto eletrônico completo, utilizando Firebase como backend (autenticação, banco de dados - Firestore, e Cloud Functions) e interface (web e mobile) responsiva. Deve permitir o registro de ponto por funcionários, gestão de funcionários e horários por empregadores, e geração de relatórios.

2. Funcionalidades Principais

Registro de Ponto:

Registro de entrada e saída, intervalo para descanso e entrada e saída para horas extras com data e hora precisas.

Inclusão da localização e wifi utilizado do em cada registro (geolocalização) e

Opção de adicionar observações ao registro com vínculo aos documentos de upload

3. Gestão de Funcionários:

crud de funcionários (nome, e-mail, cargo, etc.). antes criar o cadastramento do empregado pelo empregador nos moldes de “criar conta”

4. Atribuição de horários de trabalho.

Definição de horários de trabalho padrão (entrada, saída, pausas, total de horas semanais, regime, folgas, etc).

Criação de horários flexíveis.

5. Associação de horários a funcionários.

6. Relatórios:

Geração de relatórios de ponto por funcionário e período.

Cálculo de horas trabalhadas, horas extras e atrasos.

Exportação de relatórios em formato CSV, PDF e xlsx.

7. Notificações:

Alertas para funcionários sobre horários de entrada, saída e intervalos.

Notificações para empregadores sobre atrasos e faltas.

Alertas para empregador sobre registro de horários fora do esperado  
  
8. Inclua a funcionalidade no menu do dashboard  
9. Considere usar o layout do login  
10. Crie e use elementos reutilizáveis

11. lembre-se as credenciais estão no arquivo serviceAccount.json na pasta src/secrets  
12. Veja se há necessidade de 'use client';

imagine o fluxo que após o empregador criar sua conta ele deve complementar suas informações de modo a possibilitar o envio completo para o esocial doméstico. (precisa pesquisar na internet quais são todas as informações obrigatórias e opcionais) inclua a questão do upload e senha do certificado digital (linka para gestão de documento).

Depois considere o cadastramento dos tipos: empregados (também com pesquisa das informações) e dos familiares (bem simples, somente para acesso ao aplicativo). Os tipos só podem ter crud pelo próprio ou pelo empregador.

Atenção para as gravações no firebase/firestore

as credenciais estão no arquivo serviceAccount.json na pasta src/secrets

Veja se há necessidade de 'use client';

Considere o uso das boas práticas e

criação e uso de elementos reutilizáveis

integre com os demais módulos

 Use o <select> HTML puro, com os mesmos atributos ao invés do <Select> do Chakra (porque está apresentando erro)