# MAPA DA APLICAÇÃO

## TELA INICIAL (LOGIN)

USUÁRIO
Tipo: POST -path=login-
Uso: verifica a existência do usuário no banco de dados e faz login
Params: não tem
Body: string email_nickName, string password
Return: user:{id,nickName:'string',email:'string'}, token:'string' OU error='string'

Apelido ou Email

Senha

MENSAGEM DE ERRO

ENTRAR - 1:

Não tenho cadastro

ESQUECI A SENHA

MENSAGEM DE ERRO
1.Usuário ou senha invalida
2.Problemas Internos, entrar em contato com equipe de suporte

## TELA DE CADASTRO

Nome do usuário

Apelido

Data

Email

Senha

Repetir Senha

MENSAGEM DE ERRO

USUÁRIO
Tipo: POST -path=Register-
Uso: Executa o registro do cliente no banco de dados
Params: não tem
Body: all string (name,nickName, email, date,passoword, passwordConfir
Return: string 'success' OU error='o que ocorreu'

Cadastrar

Volte para a página de login

MENSAGEM DE ERRO
1.Nome de usuário invalido
2.Apelido já utilizado
3.Email já Cadastrado
4.Campos senha não conferem
5.Problemas Internos, entrar em contato com equipe de suporte

## CONVITES

LISTA DE CONTATOS

MOLDE DA LISTA

APELIDO ACEITAR FECHAR

CONTATOS - CONVITES
Tipo: GET -path=contactInvitations-
Uso: Recupera uma lista de contatos que enviaram convite
Params: user:id
Body: não tem
Return: ListaContatos[{id,nickName,name}] OU error='o que ocorreu'

Tipo: POST -path=contactAccept-
Uso: Aceita o convite do contato
Params: não tem
Body: user: id, contact:id
Return: 'success' OU error='o que ocorreu'

LISTA DE REUNIÕES

MOLDE DA LISTA

MOTIVO
DATA ACEITAR FECHAR

MENSAGEM DE ERRO

FECHAR

REUNIÕES - CONVITES
Tipo: GET -path=meetingInvitations-
Uso: Recupera uma lista de reuniões que enviaram convite
Params: user:id
Body: não tem
Return: ListaReuniões[{id,theme,time,adm{id,name}}] OU error='o que ocorreu'

Tipo: POST -path=meetingAccept-
Uso: Aceita o convite de reunião
Params: não tem
Body: user: id, meeting:id
Return: 'success' OU error='o que ocorreu'

## FRES

USER - NOVA SENHA - 1
Tipo: POST -path=forgotPassword-
Uso: Envia para o usuario um email com um codigo aleatorio na url para reformulação de senha
link: 'linkfrontend.../newpassword/:(user:id)/:(codigo aleatorio)' armazenar o codigo por um tempo depois descarta
Params: não tem
Body: string email_nickName
Return: 'success' OU error='o que ocorreu'

Apelido ou Email

MENSAGEM DE ERRO

ENVIAR

ENTRA NA PRÓXIMA TELA POR UM LINK NO EMAIL

Vá para seu email para configurar uma nova senha

## FRES

USER - NOVA SENHA - 2
Tipo: POST -path=setNewPassword-
Uso: Envia e troca senha caso o codigo enviado no email ainda esteja ativo
Params: não tem
Body: string:newPassword, string:code
Return: 'success' OU error='o que ocorreu'

Senha

Nova Senha

MENSAGEM DE ERRO

ENVIAR

## TELA PRINCIPAL

NOME DO PROGRAMA

APELIDO DO USUARIO

Nome do usuário
Id do usuário
Opções do usuário
Convites
sair

Alterar informações
Excluir conta

USUÁRIO
Tipo: POST -path=deleteUser-
Uso: Apllica um soft delete no usuário
Params: não tem
Body: user: id
Return: string 'success' OU error='o que ocorreu'

CERTEZA DE EXCLUSÃO DE CONTA
S
N

CONTATO DELETANDO
Tipo: POST -path=deleteContact-
Uso: deleta um contato
Params: não tem
Body: user: id, contact:id
Return: 'success' OU error='o que ocorreu'

CERTEZA DAS EXCLUSÕES
S
N

CONTATOS
NOVO DELETAR

LISTA DE CONTATOS

CONTATOS - STATUS
Tipo: GET -path=contactStatus-
Uso: Recupera uma lista de contatos que pertençam ao usuário
Params: user:id
Body: não tem
Return: ListaContatos[{id,nickName,name,online,sendMessage}] OU error='o que ocorreu
sendMessage(é usado caso o contato tenha enviado mensagem que o user ainda não viu)

CONTATOS - CONVERSAS
Tipo: GET -path=getContactChat-
Uso: Recupera o referencia da conversa
Params: user:id, contact:id
Body: não tem
Return: Chat:id OU error='o que ocorreu'

CONVERSAS - ENVIA MENSAGEM E RECEBIMENTO DE MESSAGENS
a conversa será feita pelo ratchet framework (php)
URL: 'ws://localdobackend/chat'

ENVIO será enviado um json para o ratchet
data:{user:id,user:name,chat:id,message:string}
a messagem do usuario deve passar primeiro pelo backend antes de aparecer no frontend.

RETORNO será recebido um json similar ao enviado
data:{sentMessageId:id,sentMessageName:string,
date: data e hora do envio, message:string}

TELA DE CHAT DURANTE CONVERSAS 1-1

ZONA DE DIGITAÇÃO

REUNIÃO DELETANDO
Tipo: POST -path=deleteMeeting-
Uso: caso seja adm deleta a reunião ou caso seja participante se retira de uma reunião
Params: não tem
Body: user: id, reunião:id
Return: 'success' OU error='o que ocorreu'

CERTEZA DAS EXCLUSÕES
S
N

REUNIÕES
CRIAR DELETAR

LISTA DE REUNIÕES

REUNIÃO - STATUS
Tipo: GET -path=meetingStatus-
Uso: Recupera uma lista de reuniões que o usuário aceitou comparecer
Params: user:id
Body: não tem
Return: ListaMeeting[{id,theme,date,adm{id,name},occurring}] OU error='o que ocorreu'
occurring: boolean (caso a reunião esteja acontecendo naquele momento)

Reunião - CONVERSAS
Tipo: GET -path=getReuniãoChat-
Uso: Recupera o referencia da conversa
Params: user:id, meeting:id
Body: não tem
Return: Chat:id OU error='o que ocorreu'

FILHO

ALTERAR INFORMAÇÕES

Nome do usuário
Apelido
Email
Senha Atual
Nova Senha
Repetir Senha

MESMOS ERROS DE CADASTRO

ALTERAR

CERTEZA DAS ALTERAÇÕES
S
N

USUÁRIO
Tipo: POST -path=updateUser-
Uso: Executa uma atulização no registro do cliente no banco de dados
Params: não tem
Body: all string (name,nickName, email, date,password,newpassoword, passwordConfir)
Return: string 'success' OU error='o que ocorreu'

LEGENDAS
MENUS DE ABA
FIM SAI DA JANELA
FILHO JANELA FILHO
MOTIVO S N JANELA CONDICIONAL SIM E NÃO
JANELA DE SELEÇÃO
JANELA DE TEXTO

FILHO

MANDAR CONVITE

APELIDO

MENSAGEM DE ERRO

MANDAR CANCELAR

MENSAGEM DE ERRO
1.Apelido inválido
5.Problemas Internos, entrar em contato com equipe de suporte

CONTATO - ADICIONANDO
Tipo: POST -path=sendContactInvitation-
Uso: Envia um convite ao contato
Params: não tem
Body: user: id, contact:nickName
Return: 'success' OU error='o que ocorreu'

FILHO

CRIAR / MODIFICAR REUNIÃO

DIA MÊS ANO

HORA MINUTO

MOLDE DA LISTA

APELIDO

MENSAGEM DE ERRO

CRIAR CANCELAR

MENSAGEM DE ERRO
1.Data Inválida
2.Nenhum usuário selecionado
5.Problemas Internos, entrar em contato com equipe de suporte

REUNIÃO ADICIONANDO E ATUALIZANDO
Tipo: GET -path=getMeeting-
Uso: Recupera uma reunião caso o user seja o adm da reunião
Params: user:id, meeting:id
Body: não tem
Return: meeting{id,theme,date,listContact[id,name,nickname,isThisMeeting] } OU error='o que ocorreu
isThisMeeting: boolean - (se o contato ja foi convidado ou aceitou aquela reunião - )

Tipo: POST -path=setMeeting-
Uso: Cria um nova reunião ou altera e envia os convites para os convidados ou retira eles
se meeting:id = -1 logo está sendo criada uma reunião
Params: não tem
Body: user: id, meeting:id, date: string data e hora, listContact[id]
Return: 'success' OU error='o que ocorreu'

## TELA DE CHAT DURANTE REUNIÕES

ZONA DE DIGITAÇÃO

REUNIÃO

PARTICIPANTE

MOTIVO DA REUNIÃO
HORA DE INICIO
ADM DA REUNIÃO

MOLDE DA LISTA

APELIDO STATUS

STATUS:
NA REUNIÃO
AUSENTE

RESTANTE DA LISTA

FECHAR

ADMINISTRADOR

MOTIVO DA REUNIÃO
HORA DE INICIO

MOLDE DA LISTA

APELIDO STATUS CONVIDAR

RESTANTE DA LISTA

APELIDO STATUS REMOVER

RESTANTE DA LISTA

FINALIZAR REUNIÃO

FECHAR

CONTATOS NA REUNIÃO
Tipo: GET -path=contactInThisMeeting-
Uso: Recupera um lista de contatos que foram convidados para a reunião e aceitaram o convite e estados na reunião ex: ausente - na reunião
Params: user:id, meeting:id
Body: não tem
Return: ListaContatos[{id,nickName,name,status,admId}] OU error='o que ocorreu'
Status:String(se o está ausente ou presente na reunião)
AdmId: id( identificador do adm, caso o user seja o adm segunda tela)

REUNIÃO REMOVE O CONTATO
Tipo:POST -path=removeContactMeeting-
Uso: Retira um contato da reunião (somente se user:id == adm:id)
Params: não tem
Body: user:id, meeting:id, contact:id
Return: string 'success' OU error='o que ocorreu'

REUNIÃO ADICIONA O CONTATO
Tipo: POST -path=addContactMeeting-
Uso: manda um convite aos contatos em um reunião ocorrendo (somente se user:id == adm:id)
Params: não tem
Body: user:id, meeting:id, contact:id
Return: string 'success' OU error='o que ocorreu'

REUNIÃO ENCERRA A REUNIÃO
Tipo: POST -path=finishingMeeting-
Uso: finaliza a reunião encerrando o envio de mensagens
Params: não tem
Body: user:id, meeting:id
Return: string 'success' OU error='o que ocorreu'