



UNIVERSIDADE FEDERAL DA FRONTEIRA SUL-UFFS
CIÊNCIA DA COMPUTAÇÃO

REQUISITOS DO USUÁRIO

Botneer - Um Bot de Gerenciamento Central de Servidores no Discord

FEVEREIRO, 2022
CHAPECÓ-SC

1 IDENTIFICAÇÃO DA SOLUÇÃO E STAKEHOLDERS

1.1 IDENTIFICAÇÃO DA SOLUÇÃO

Com a pandemia, houve uma popularização no trabalho remoto e em soluções para ele. Nesse sentido, fiasse necessário a utilização de plataformas para suprir a falta de contato direto entre os membros da empresa. Neste sentido, o Discord tornasse uma forte ferramenta para integrar o time por permitir criação de múltiplos canais de comunicação, divisão de membros em grupo e restrições de ações por grupos. Entretanto, um de seus problemas são a falta de ferramentas nativas para realizar tarefas mais complexas ou repetitivas como realizar avisos de tempos em tempos, banir membros que desobedessão um determinado conjunto de regras, dar boas vindas personalizadas à novos membros etc.

Para suprir essa necessidade, o Discord possibilita cria robôs personalizados que são capazes de realizar tarefas de forma pré-programada, chamados de bots. Como uma empresa que utiliza o discord como plataforma principal de comunicação, a Keener.io, sofre desses problemas e vê necessária a criação de um bot capaz de gerenciar o servidor da empresa e automatizar tarefas repetitivas.

2 ELICITAÇÃO DOS REQUISITOS

2.1 DESCRIÇÃO DOS PROCESSOS DA PROPRIEDADE/FUNCIONAMENTO DA PROPRIEDADE

A empresa é subdividida nos setores de desenvolvimento, RH, vendas e administrativo. Por se tratar de uma empresa de desenvolvimento de software, o setor de desenvolvimento é o maior entre os 3 e subdivido por grupos de desenvolvimento, chamados de squads, e por função de cada funcionário dentro da sua respectiva squad.

Cada squad possui um nome escolhido pelos seus membros, uma imagem personalizada e um grupo de pessoas de diversas funções atreladas a ela, normalmente não ultrapassando às 11 pessoas, elas também possuem projetos atrelados a elas, que são representados no servidor por canais de texto com o nome do respectivo projeto. Membros de outras squads não podem acessar canais de uma squad que não façam parte e cada uma possui um cronograma de reuniões e eventos importantes específico.

Existem 8 funções dentro da empresa, sendo 4 apenas para desenvolvimento, dentro das de desenvolvimento temos: design, dev, QA e PO. Nesse caso, existem canais restritos e canais compartilhados, os restritos possuem o sufixo “-função” em seus nomes e os compartilhados não. As outras funções possuem uma organização semelhante, salvo Administradores e RH que possuem uma organização própria.

Dito isso, RH e Administração são os principais moderadores do servidor e por isso possuem privilégios superiores aos outros membros da organização. Eles possuem permissão de escrita nos canais de avisos do servidor e a capacidade de acessar qualquer canal, além de possuírem privilégios de administrador e organizarem os eventos da empresa.

2.2 DESCRIÇÃO DOS PRINCIPAIS PROBLEMAS ENFRENTADOS

- Dificuldade de enviar avisos aos membros da organização, hoje tudo é manual e existem avisos diários e semanais.

- Necessidade de melhorar a recepção de novos membros a equipe, devido ao crescimento da empresa, há a necessidade de receber os novos membros de forma automática de forma a passar todas as informações necessárias para o entendimento do funcionamento da empresa
- Criar agendas personalizadas e mostrar os eventos no discord. Não há integração entre o calendário de eventos geral e individual(por squad e por função) com o discord, o que cria uma demanda de sempre fazer esse processo à mão

2.3 EXPECTATIVAS DO SISTEMA E QUAIS FUNCIONALIDADES DEVERÁ CONTER

- Permitir automação de tarefas
 - Criação de eventos
 - Boas vindas aos novos membros
 - Avisos no sistema
- Criação de boas vindas personalizadas que possibilitem ao novo membro compreender funcionamento da empresa e integrar ao todo
- Criação de sistema de aviso que será disparado em períodos específicos e será repetido diariamente
- Integrar caledários

3 ESPECIFICAÇÃO DOS REQUISITOS

3.1 REQUISITOS FUNCIONAIS DO SISTEMA

ID	USUÁRIO	NOME	DESCRIÇÃO
RF01	Bot(Sistema)	Gerar mensagem de alerta	O bot deve ser capaz de gerar mensagem de alerta personalizados
RF02	Bot(Sistema)	Gerar alerta de horas(Chat geral)	Bot deve ser gerar um alerta todo dia às 18h para lembrar as pessoas de colocar horas no portal
RF03	Bot(Sistema)	Gerar alerta de horas(DM)	Bot deve gerar alerta privada para as pessoas que não registraram horas no dia, lembrar todo dia às 09h
RF04	Usuário Geral	Registrar mensagens de alerta	Qualquer usuário deve ser capaz de adicionar novas mensagens de

			alerta ao bot
RF05	Usuário ADM	Remover mensagem de alerta	O usuário admin deve ser capaz de remover mensagens de alerta inapropriadas
RF06	Usuário ADM	Listar mensagens de alerta cadastradas	Listar todos as mensagem cadastrada no bot
RF07	Bot(Sistema)	Gerar boas vindas personalizadas	Criar boas vindas personalizadas para os novos participantes do server
RF08	Bot(Sistema)	Dar cargos aos novos integrantes do server	Ao entrar no servidor, usuário deve receber um cargo chamado "freshman"
RF09	Bot(Sistema)	Liberar acesso a canais de registro	Todo novo usuário no servidor deve ser capaz de interagir apenas com dois canais: 'bem-vindo'
RF10	Bot(Sistema)	Gerar canal de 'bem-vindo'	Gerar um canal de texto com uma mensagem de boas vindas ao novo integrante
RF11	Bot(Sistema)	Gerar canal de 'registre-se'	Gerar um canal onde o integrante irá escolher os seus cargos dentro do servidor.
RF12	Usuário ADM	Cadastrar eventos	O usuário admin deve ser capaz de cadastrar eventos mensais que serão direcionados a determinado grupo de usuários
RF13	Usuário Geral	Listar eventos	Todo usuário deve ser capaz de gerar uma lista de eventos direcionados ao grupo que ele

			pertence
RF14	Usuário Geral	Selecionar evento	Todo usuário deve ser capaz de ver mais detalhes sobre um determinado evento

3.1 REQUISITOS NÃO FUNCIONAIS DO SISTEMA

ID	REQUISITO NÃO FUN.	DESCRIÇÃO
RNF01	Tecnologia	Deve ser feito em python e possuir integração com as API do discord e do portal da empresa.
RNF02	Estabilidade	O bot deve ser capaz de ficar online 24h sem nunca, ou quase nunca, ser desligado ou reiniciado.
RNF03	Segurança	As datas e eventos devem ser protegidos para impedir, problemas de spot ou vazamento de dados.

3.2 DIAGRAMA DE CASOS DE USO

3.2.1 Atores do diagrama de casos de uso

Autor	Descrição
Bot(Sistema)	O bot em si, responsável por automatizar tarefas e permitir os outros usuários de realizar certas ações por intermédio dele
Usuário ADM	Usuário moderador do servidor, responsável por gerenciar os outros usuários, enviar avisos e organizar eventos
Usuário Geral	Usuário padrão do sistema, aqui se junto todos que não são do administrativo ou do RH

3.2.2 Diagrama

Nome do Caso de Uso	Gerar alerta de horas(DM)
Casos de Uso Relacionados	UC01, UC02, UC04, UC05, UC06
Pré-Condição	Usuario não ter inseridos horas no portal no dia anterior e bot estar online
Descrição Caso de Uso	Bot deve gerar alerta privada para as pessoas que não registraram horas no dia, lembrar todo dia às 09h

ID Caso de Uso	UC04
Nome do Caso de Uso	Registrar mensagens de alerta
Casos de Uso Relacionados	UC01, UC02, UC03, UC05, UC06
Pré-Condição	Bot estar online
Descrição Caso de Uso	Qualquer usuário deve ser capaz de adicionar novas mensagens de alerta no bot

ID Caso de Uso	UC05
Nome do Caso de Uso	Remover mensagem de alerta
Casos de Uso Relacionados	UC01, UC02, UC03, UC04, UC06
Pré-Condição	Haver mensagens cadastradas no bot
Descrição Caso de Uso	O usuário admin deve ser capaz de remover mensagens de alerta inapropriadas

ID Caso de Uso	UC06
Nome do Caso de Uso	Listar mensagens de alerta cadastradas
Casos de Uso Relacionados	UC01, UC02, UC03, UC04, UC05
Pré-Condição	Existirem mensagens cadastradas no bot
Descrição Caso de Uso	Listar todas as mensagens cadastradas no bot

ID Caso de Uso	UC07
Nome do Caso de Uso	Gerar boas vindas personalizadas
Casos de Uso Relacionados	
Pré-Condição	Um novo usuário entrar no servidor
Descrição Caso de Uso	Criar boas vindas personalizadas para os novos participantes do server

ID Caso de Uso	UC08
Nome do Caso de Uso	Dar cargos aos novos integrantes do server
Casos de Uso Relacionados	UC07
Pré-Condição	Um novo usuário deve entrar no servidor
Descrição Caso de Uso	Ao entrar no servidor, usuário deve receber um cargo chamado “freshman”

ID Caso de Uso	UC09
Nome do Caso de Uso	Liberar acesso a canais de registro
Casos de Uso Relacionados	UC07, UC08
Pré-Condição	Usuário deve possuir cargo “freshman”
Descrição Caso de Uso	Todo novo usuário no servidor deve ser capaz de interagir apenas com dois canais: ‘bem-vindo’

ID Caso de Uso	UC10
Nome do Caso de Uso	Gerar canal de ‘bem-vindo’
Casos de Uso Relacionados	UC07, UC08, UC09
Pré-Condição	Canal ‘bem-vindo’ não deve existir
Descrição Caso de Uso	Gerar um canal de texto com uma mensagem de boas vindas ao novo integrante

ID Caso de Uso	UC11
Nome do Caso de Uso	Gerar canal de 'registre-se'
Casos de Uso Relacionados	UC07, UC08, UC09
Pré-Condição	Canal 'registre-se' não deve existir
Descrição Caso de Uso	Gerar um canal onde o integrante irá escolher os seus cargos dentro do servidor.

ID Caso de Uso	UC12
Nome do Caso de Uso	Cadastrar eventos
Casos de Uso Relacionados	
Pré-Condição	Bot deve estar online
Descrição Caso de Uso	O usuário admin deve ser capaz de cadastrar eventos mensais que serão direcionados a determinado grupo de usuários

ID Caso de Uso	UC13
Nome do Caso de Uso	Listar eventos
Casos de Uso Relacionados	UC12
Pré-Condição	Devem existir eventos para aquele período de tempo
Descrição Caso de Uso	Todo usuário deve ser capaz de gerar uma lista de eventos direcionados ao grupo que ele pertence

ID Caso de Uso	UC14
Nome do Caso de Uso	Selecionar evento
Casos de Uso Relacionados	UC12, UC13
Pré-Condição	Evento deve existir
Descrição Caso de Uso	Todo usuário deve ser capaz de ver mais detalhes sobre um determinado evento