

Nome do projeto: central de atendimento da uby

Autor(a): goimy rondonelli da silveira

E-mail: goimyrondinellis@gmail.com

CANVAS

Problema

Descreva o problema do negócio

Muitos clientes não têm familiaridade com o aplicativo

Clientes preferem testar o serviço primeiro pelo WhatsApp

O processo atual é 100% manual

Atendentes precisam:

Ler mensagens no WhatsApp

Identificar endereço de origem

Identificar endereço de destino

Cadastrar a corrida no aplicativo

Isso gera:

Demora no atendimento

Risco de erro humano

Custo alto com equipe

Dificuldade de escalar o atendimento

Como resolver SEM AI

Descreva o processo atual e solução

Fluxo operacional:

Canal: WhatsApp Business (número oficial). Atendente humano recebe mensagem, segue script padrão:

Pergunta 1: "Onde você está? (endereço ou enviar localização)"

Pergunta 2: "Para onde deseja ir?"

Criar corrida via API do app.

Informar motorista ao cliente.

Padrões: máximo 2 minutos SLA de resposta inicial , máximo 5 minutos para confirmação .

Registro: cada atendimento logado (mensagens, tempo, resultado).

Recursos necessários (MVP sem IA):

Equipe de atendimento: 12 atendentes

Como resolver COM AI

Descreva a solução AI by design

Principais decisões de design:

Não permitir que LLM "decida" dispatch crítico sem validação e de endereço.

Human-in-the-loop: bot resolve fluxos simples; atendente recebe histórico e assume exceções.

Fluxo técnico (MVP recomendado):

Cliente envia mensagem no WhatsApp → provider (Twilio / 360dialog / Zenvia).

Bot low-code responde com perguntas estruturadas (origem, destino). Usa NLU leve (regex + modelo leve/LLM para entropia).

Bot chama APP para validar endereço

Se endereço validado:

APP criar corrida e obter motorista/ETA.

Bot informa nome do motorista, ETA, modelo do carro e número de placa parcial.

Se não encontrar motorista, não encontrar endereço ou tiver qualquer dúvida do que está fazendo (nível de confiança inferior a 70%) chama o atendente.

Para quem?

Descreva as pessoas

Persona primária (alvo do MVP):

"João" — 67 anos, mora em cidade pequena, pouco uso de smartphone, liga para a central ou manda mensagem pedindo corrida. Objetivo: pedir corrida sem instalar/uso o app.

Personas secundárias:

Novos usuários do app (sem familiaridade) porque na maioria das vezes nunca usaram um transporte por aplicativo, depois de se acostumarem a usar vão chamar no app porque é mais rápido e mais barato.

Atendentes da central (usuários do painel de handoff).

Gestores de operação (monitoram KPIs). Motoristas (afetados indiretamente: recepção de corridas mais corretas).

Ações recomendadas:
Realizar 10 entrevistas com João-type por cidade piloto para ajustar linguagem.
Testes de usabilidade do fluxo WhatsApp (observacional).

Está de acordo com esse público? Quer segmentar por faixa etária ou cidade específica?

1

3

4

5

Dados

Liste dados e fontes

Ferramentas

Descreva as ferramentas

Indicadores de sucesso

Descreva o(s) indicador(es)

Fontes de dados necessárias:

Mensagens do WhatsApp (texto, mídia, áudio, timestamps).

API do app: validação de endereço / busca de motorista / criação de corrida / ETA.

Logs da central (atendimentos por operador, tempo médio).

Dados de motoristas disponíveis (localização, status).

6

Provedores WhatsApp Business:

360dialog / Twilio / Zenvia — escolha depende de custo, SLA e integração local.

Flow builder / Chatbot:

Typebot / Landbot (low-code, rápido para MVP) — prós: velocidade; contras: flexibilidade para NLU avançada.

Botpress / Rasa (mais técnico) — prós: controle; contras: tempo de

Indicadores propostos:

Tempo médio do primeiro contato até criação da corrida (Tempo de pedido → dispatch)

Baseline: 5 minutos

Meta MVP: reduzir em 40%

Erros operacionais por 100 corridas (ex.: corrida com endereço inválido,

7

.StartSe