**🔷 Phase 1 : Définition du projet (10 tâches)**

C'est une phase cruciale pour cadrer le projet et éviter des erreurs coûteuses plus tard. Un mauvais cadrage peut entraîner des fonctionnalités inutiles, un surcoût de développement ou une adoption limitée par les utilisateurs.

**1. Définir les objectifs du chatbot**

L’objectif principal doit être clairement défini pour s’assurer que le développement répondra aux besoins spécifiques des utilisateurs. Les objectifs possibles incluent :

* **Support client 24/7**
  + Répondre aux questions fréquentes (FAQ).
  + Assister les clients sur des problèmes techniques.
  + Gérer les tickets d’assistance (connexion avec un CRM).
* **Assistant interne pour employés**
  + Accès rapide aux ressources et documents internes.
  + Aide à l’automatisation des tâches répétitives.
  + Gestion des demandes internes (RH, IT).
* **Chatbot marketing**
  + Génération de leads avec qualification automatique.
  + Recommandation de produits ou services.
  + Engagement sur les réseaux sociaux via chatbot interactif.
* **Assistant conversationnel intelligent**
  + Utilisation de la mémoire contextuelle pour des interactions naturelles.
  + Apprentissage et amélioration des réponses en fonction du feedback utilisateur.
  + Intégration avec GPT-4 pour générer des réponses riches et dynamiques.

**2. Identifier les utilisateurs cibles**

Un bon chatbot est conçu pour ses utilisateurs. Il faut répondre aux questions suivantes :

* **Quels types d’utilisateurs vont interagir avec le chatbot ?**
  + **Clients** (B2C, B2B) → Besoin d’un support rapide et efficace.
  + **Partenaires** (fournisseurs, distributeurs) → Support spécifique pour leurs interactions.
  + **Employés** → Assistance pour leurs tâches quotidiennes.
* **Quel est leur niveau technique ?**
  + **Grand public** → Interface simple, vocabulaire accessible, navigation fluide.
  + **Experts / Professionnels** → Réponses détaillées, capacités d’analyse avancées, intégration avec outils métiers.
* **Quel canal est le plus pertinent pour eux ?**
  + Les **clients** préfèrent souvent **WhatsApp, Messenger** ou un widget sur un site web.
  + Les **employés** utilisent **Slack, Microsoft Teams** ou un intranet.
  + Les **professionnels** préfèrent souvent une **API intégrée à leurs outils**.

**3. Lister les cas d’usage prioritaires**

Définir les interactions types et scénarios majeurs du chatbot :

* **Recenser les principales questions et interactions attendues**
  + FAQ : horaires, tarifs, services.
  + Résolution de problèmes techniques.
  + Informations sur les produits et services.
* **Prioriser les scénarios critiques**
  + Urgences (ex. problème de connexion, paiement refusé).
  + Transactions (achat, prise de rendez-vous).
  + Assistance en cas d’escalade vers un agent humain.
* **Définir les réponses souhaitées**
  + **Réponses simples** → "Voici notre numéro de contact".
  + **Réponses guidées** → Proposer des boutons avec des options.
  + **Réponses interactives** → Formulaires, messages multimédias, etc.

**4. Choisir les canaux de diffusion**

Le choix des canaux dépend des utilisateurs :

* **Web** : Intégration sur un site via un **widget de chat** (React.js, Vue.js).
* **Messagerie instantanée** : **WhatsApp (Twilio API)**, **Messenger (Meta API)**.
* **Slack / Teams** : Pour un usage **interne en entreprise**.
* **Application mobile** : Intégration avec **Flutter ou React Native**.

**5. Déterminer le ton et la personnalité du chatbot**

La personnalité influe sur l’expérience utilisateur :

* **Formel** : Sérieux, professionnel, précis. Idéal pour la finance, la santé.
* **Amical** : Chaleureux, engageant, pour une marque jeune et dynamique.
* **Expert** : Données chiffrées, analyses détaillées, pour un public technique.

**6. Lister les intégrations nécessaires**

Le chatbot doit interagir avec plusieurs outils :

* **CRM** : HubSpot, Salesforce pour la gestion des clients.
* **Base de données clients** : PostgreSQL, Firebase Firestore.
* **APIs tierces** : Paiement (Stripe), agenda (Google Calendar), gestion documentaire (Notion).

**7. Sélectionner la technologie backend**

Le backend gère la logique métier du chatbot :

* **FastAPI** (Python) : Rapide, supporte OpenAPI, idéal pour les APIs.
* **Flask** (Python) : Léger, flexible, parfait pour des projets modulaires.
* **Django** (Python) : Plus structuré, idéal pour les gros projets.
* **Node.js** (JavaScript) : Si besoin d’un écosystème full-stack JS.

**8. Sélectionner la technologie frontend**

Le frontend affiche le chatbot et gère les interactions :

* **React.js** : Flexible, maintenu par Facebook, grande communauté.
* **Next.js** : Optimisé pour le SEO et les performances.
* **Vue.js** : Simple, rapide à prendre en main.

**9. Déterminer l’infrastructure d’hébergement**

Le choix dépend de la scalabilité et du budget :

* **Backend** :
  + **AWS Lambda** (serverless, coût à l’usage).
  + **Google Cloud Run** (scalable, géré par Google).
  + **Railway** (facile à déployer, supporte PostgreSQL).
* **Frontend** :
  + **Vercel** (Next.js optimisé).
  + **Netlify** (déploiement facile).
  + **Firebase Hosting** (si besoin d’un backend Firebase).
* **Base de données** :
  + **PostgreSQL** (relationnel, robuste).
  + **Firebase Firestore** (NoSQL, en temps réel).
  + **MongoDB** (NoSQL, flexible).

**10. Estimer les coûts et définir un budget**

Une estimation budgétaire est cruciale pour anticiper les dépenses :

* **OpenAI API**
  + Facturation en fonction du nombre de requêtes.
  + Estimer le volume mensuel d’appels API.
* **Hébergement**
  + **Backend** : AWS, Google Cloud → coûts variables selon la charge.
  + **Base de données** : PostgreSQL ou Firebase → coût stockage et requêtes.
* **Maintenance et mises à jour**
  + Temps de développement initial : 2-3 mois.
  + Coût de mise à jour (correction de bugs, optimisation).
  + Support technique (équipe dédiée ou automatisée).