Listes de Taches

**1. Analyse des besoins**

* Identifier les utilisateurs cibles du système (agents support, managers, etc.)
* Recueillir les attentes fonctionnelles (ex. : rapidité, précision, interface)
* Lister les types de tickets à traiter (bugs, incidents, demandes, etc.)
* Définir les catégories finales de classification
* Rédiger le cahier des charges

**2. Collecte et préparation des données**

* Collecter un corpus de tickets existants (avec catégories)
* Nettoyer les données (suppression des doublons, anonymisation)
* Prétraiter les textes (tokenisation, lemmatisation, suppression des stopwords)
* Annoter manuellement un échantillon si besoin (jeu de validation)

**3. Conception du système**

* Choisir l’approche NLP (embeddings, BERT)
* Sélectionner un ou plusieurs algorithmes de classification (Ticket)
* Définir l’architecture globale : pipeline de traitement + interface
* Évaluer les outils à intégrer (spaCy)

**4. Développement**

* Implémenter le pipeline de classification automatique
* Intégrer la détection automatique de la langue si nécessaire
* Ajouter un mécanisme de feedback utilisateur (correction manuelle de la catégorie)
* Créer une API pour exposer le système de classification
* Développer une interface web simple (formulaire de soumission de tickets + résultat)

**5. Tests et évaluation**

* Évaluer les performances du modèle (accuracy)
* Tester les cas limites (tickets ambigus, langage familier, fautes de frappe)
* Impliquer des utilisateurs réels pour la recette fonctionnelle
* Corriger les bugs et ajuster le modèle

**6. Déploiement**

* Déployer sur un serveur de test
* Configurer les droits d’accès, la sécurité et les logs
* Rédiger la procédure de mise en production

**7. Documentation**

* Documenter le code, les API, les modèles utilisés
* Rédiger un guide d’utilisation pour les utilisateurs finaux
* Fournir une documentation technique pour les développeurs ou mainteneurs

**8. Suivi et maintenance**

* Surveiller les performances en production (temps de réponse, taux d’erreur)
* Mettre à jour le modèle avec de nouveaux tickets (réentraînement)
* Ajouter des nouvelles catégories si besoin
* Prévoir un plan de sauvegarde et de reprise après incident

**Distribution :**

* **Tâche**
* **Phase**
* **Priorité**
* **Échéance estimée**
* **Responsable**
* **Commentaires / Avancement**

| **Tâche** | **Phase** | **Priorité** | **Échéance estimée** | **Responsable** | **Commentaires / Avancement** |
| --- | --- | --- | --- | --- | --- |
| Identifier les utilisateurs et leurs besoins | Analyse des besoins | Haute | 10/06/2025 | Ulrich | Interviews utilisateurs à prévoir |
| Lister les types de tickets à classifier | Analyse des besoins | Haute | 11/06/2025 | Angea | Voir base de données historique |
| Collecter un corpus de tickets | Données | Haute | 13/06/2025 | Anea | Accès à la base XAMPP |
| Nettoyer et anonymiser les données | Données | Haute | 15/06/2025 | Angea | Supprimer données sensibles |
| Prétraiter les textes (tokenisation, etc.) | Données | Moyenne | 16/06/2025 | Angea | Utiliser spaCy ou NLTK |
| Choisir une approche NLP | Conception | Haute | 17/06/2025 | Angea | Comparer TF-IDF vs BERT |
| Sélectionner les algorithmes de classification | Conception | Haute | 18/06/2025 | Naomi | Tester plusieurs modèles |
| Développer le pipeline NLP | Développement | Haute | 21/06/2025 | Jean-Claude | À modulariser pour évolutivité |
| Créer une API de classification | Développement | Moyenne | 23/06/2025 | Angea | FastAPI recommandé |
| Développer une interface web simple | Développement | Moyenne | 25/06/2025 | Ulrich | HTML + JS / React minimal |
| Évaluer les performances du modèle | Tests | Haute | 26/06/2025 | Ulrich | F1-score + confusion matrix |
| Tester avec des utilisateurs réels | Tests | Moyenne | 27/06/2025 | ULrich | Prévoir un retour structuré |
| Corriger bugs et ajuster modèle | Tests | Haute | 28/06/2025 | Jean-Claude | En fonction des retours |
| Conteneuriser le système (Docker) | Déploiement | Moyenne | 29/06/2025 | Jean-Claude | Dockerfile + docker-compose |
| Déployer le système | Déploiement | Moyenne | 30/06/2025 | Ulrich | Serveur Heroku ou local |
| Documenter le projet | Documentation | Haute | 01/07/2025 | Naomi | Inclure code, usage, modèle |
| Mettre en place un suivi post-prod | Suivi & maintenance | Moyenne | 05/07/2025 | Naomi | Logs, feedback, MAJ modèle |