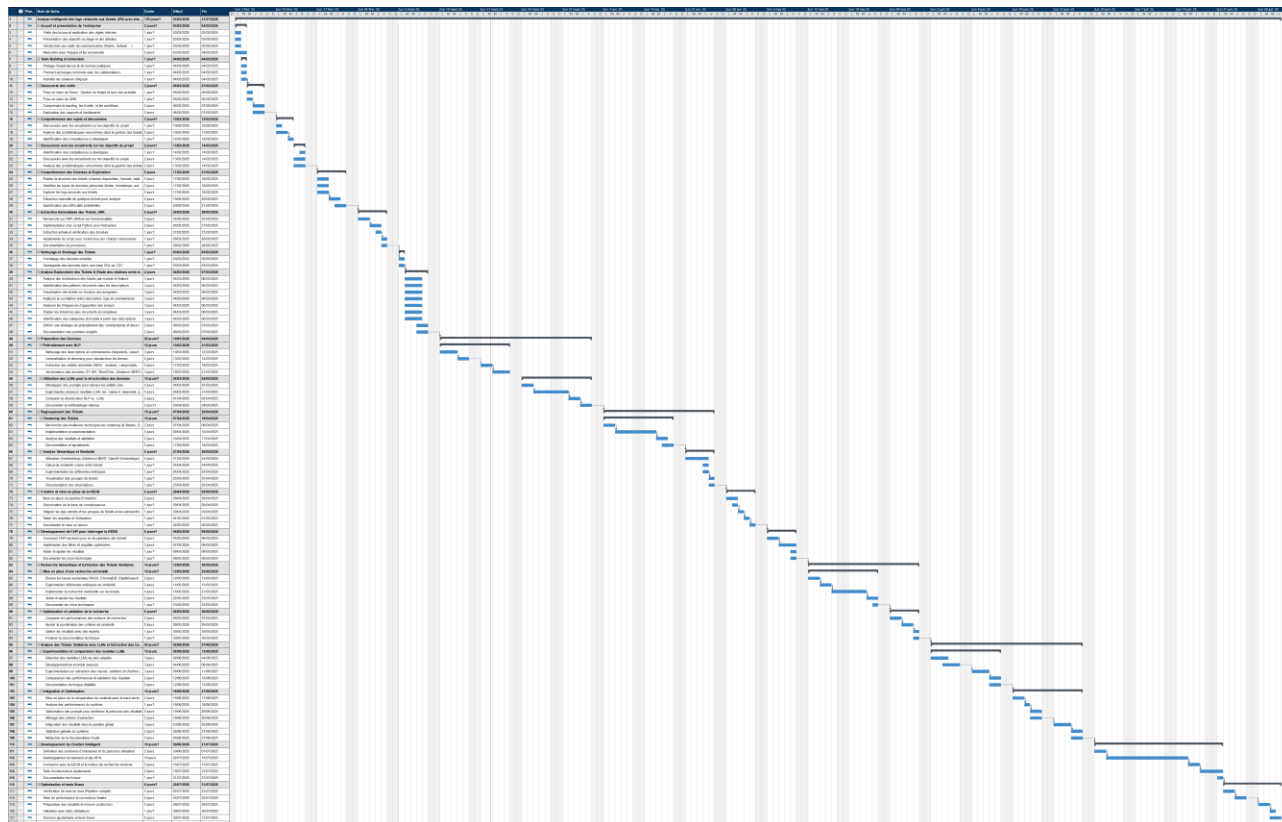


Analyse intelligente des logs rattachés aux tickets JIRA avec interaction d'une base d'erreurs connues et génération de rapports d'aide à la décision



Accueil et présentation de l'entreprise.....	6
Visite des locaux et explication des règles internes	6
Présentation des objectifs du stage et des attentes	6
Introduction aux outils de communication (Teams, Outlook ...).....	6
Rencontre avec l'équipe et les encadrants	6
Team Building et immersion	6
Partage d'expériences et de bonnes pratiques	6
Premiers échanges informels avec les collaborateurs.....	6
Activités de cohésion d'équipe	6
Découverte des outils	6
Prise en main de Timmi : Gestion du temps et suivi des activités	6

Prise en main de JIRA	6
Comprendre le backlog, les tickets, et les workflows.....	6
Exploration des rapports et dashboards.....	6
Compréhension des sujets et discussions	6
Discussions avec les encadrants sur les objectifs du projet	6
Analyse des problématiques rencontrées dans la gestion des tickets	6
Identification des compétences à développer	6
Discussions avec les encadrants sur les objectifs du projet	6
Identification des compétences à développer	6
Discussions avec les encadrants sur les objectifs du projet	6
Analyse des problématiques rencontrées dans la gestion des tickets.....	6
Compréhension des Données et Exploration	6
Étudier la structure des tickets (champs disponibles, formats, relations)	7
Identifier les types de données présentes (textes, timestamps, auteurs, etc.)	7
Explorer les logs associés aux tickets.....	7
Extraction manuelle de quelques tickets pour analyse	7
Identification des difficultés potentielles	7
Extraction Automatisée des Tickets JIRA	7
Recherche sur l'API JIRA et ses fonctionnalités	7
Implémentation d'un script Python pour l'extraction	7
Extraction initiale et vérification des données	7
Ajustements du script pour inclure tous les champs nécessaires	7
Documentation du processus	7
Nettoyage et Stockage des Tickets	7
Formatage des données extraites.....	7
Sauvegarde des données dans une base SQL ou CSV.....	7
Analyse Exploratoire des Tickets & Étude des relations entre descriptions, logs et commentaires.....	7
Analyse des distributions des tickets par module et feature	7
Identification des patterns récurrents dans les descriptions	7

Visualisation des tickets en fonction des assignees	7
Analyser la corrélation entre description, logs et commentaires	7
Analyser les fréquences d'apparition des erreurs	7
Étudier les tickets les plus récurrents et complexes.....	7
Identification des catégories de tickets à partir des descriptions	7
Définir une stratégie de prétraitement des commentaires et descriptions	8
Documentation des premiers insights	8
Préparation des Données	8
Prétraitement avec NLP	8
Nettoyage des descriptions et commentaires (stopwords, caractères spéciaux, normalisation). ..	8
Lemmatisation et stemming pour standardiser les termes.	8
Extraction des entités nommées (NER) : modules, composants, erreurs	8
Vectorisation des données (TF-IDF, Word2Vec, Sentence-BERT).	8
Utilisation des LLMs pour la structuration des données	8
Développer des prompts pour extraire les entités clés.....	8
Expérimenter plusieurs modèles LLMs (ex. Llama-2, deepseek, gemma, ;;;)	8
Comparer la structuration NLP vs. LLMs.....	8
Documenter la méthodologie retenue	8
Regroupement des Tickets	8
Clustering des Tickets.....	8
Recherche des meilleures techniques de clustering (K-Means, DBSCAN)	8
Implémentation et expérimentation	8
Analyse des résultats et validation	8
Documentation et ajustements	8
Analyse Sémantique et Similarité	8
Utilisation d'embeddings (Sentence-BERT, OpenAI Embeddings)	8
Calcul de similarité cosinus entre tickets	8
Expérimentation de différentes métriques	9
Visualisation des groupes de tickets.....	9
Documentation des observations.....	9
Création et mise en place de la KEDB	9

Mise en place du pipeline d'insertion	9
Structuration de la base de connaissances	9
Intégrer les logs extraits et les groupes de tickets et les autres informations	9
Tester les requêtes et l'indexation	9
Documenter la mise en œuvre	9
Développement de l'API pour interroger la KEDB	9
Concevoir l'API backend pour la récupération des tickets	9
Implémenter des filtres et requêtes optimisées	9
Tester et ajuster les résultats	9
Documenter les choix techniques	9
Recherche Sémantique et Extraction des Tickets Similaires	9
Mise en place d'une recherche vectorielle	9
Étudier les bases vectorielles FAISS, ChromaDB, ElastikSearch	9
Expérimenter différentes métriques de similarité	9
Implémenter la recherche vectorielle sur les tickets	9
Tester et ajuster les résultats	9
Documenter les choix techniques	9
Optimisation et validation de la recherche	9
Comparer les performances des moteurs de recherche	9
Ajuster la pondération des critères de similarité	9
Valider les résultats avec des experts	10
Finaliser la documentation technique	10
Analyse des Tickets Similaires avec LLMs et Extraction des Caractéristiques	10
Expérimentation et comparaison des modèles LLMs	10
Sélection des modèles LLMs les plus adaptés	10
Développement de prompts avancés	10
Expérimentation sur extraction des causes, solutions et d'autres insights	10
Comparaison des performances et validation des résultats	10
Documentation technique détaillée	10
Intégration et Optimisation	10
Mise en place de la récupération de contexte avec la base vectorielle	10

Analyse des performances du système	10
Optimisation des prompts pour améliorer la précision des résultats	10
Affinage des critères d'extraction	10
Intégration des résultats dans le pipeline global.....	10
Validation globale du système.....	10
Rédaction de la documentation finale	10
Développement du Chatbot Intelligent	10
Définition des scénarios d'interaction et du parcours utilisateur	10
Développement du backend et des APIs	10
Connexion avec la KEDB et le moteur de recherche vectoriel	10
Tests fonctionnels et ajustements	10
Documentation technique	10
Optimisation et tests finaux	10
Vérification de bout en bout (Pipeline complet)	11
Tests de performance et corrections finales	11
Préparation des résultats et mise en production	11
Validation avec tests utilisateurs	11
Derniers ajustements et tests finaux	11
Rédaction de la documentation finale.....	11

1. Accueil et présentation de l'entreprise

- 1.1. Visite des locaux et explication des règles internes
- 1.2. Présentation des objectifs du stage et des attentes
- 1.3. Introduction aux outils de communication (Teams, Outlook ...)
- 1.4. Rencontre avec l'équipe et les encadrants

2. Team Building et immersion

- 2.1. Partage d'expériences et de bonnes pratiques
- 2.2. Premiers échanges informels avec les collaborateurs
- 2.3. Activités de cohésion d'équipe

3. Découverte des outils

- 3.1. Prise en main de Timmi : Gestion du temps et suivi des activités
- 3.2. Prise en main de JIRA
- 3.3. Comprendre le backlog, les tickets, et les workflows
- 3.4. Exploration des rapports et dashboards

4. Compréhension des sujets et discussions

- 4.1. Discussions avec les encadrants sur les objectifs du projet
- 4.2. Analyse des problématiques rencontrées dans la gestion des tickets
- 4.3. Identification des compétences à développer

5. Discussions avec les encadrants sur les objectifs du projet

- 5.1. Identification des compétences à développer
- 5.2. Discussions avec les encadrants sur les objectifs du projet
- 5.3. Analyse des problématiques rencontrées dans la gestion des tickets

6. Compréhension des Données et Exploration

- 6.1. Étudier la structure des tickets (champs disponibles, formats, relations)
 - 6.2. Identifier les types de données présentes (textes, timestamps, auteurs, etc.)
 - 6.3. Explorer les logs associés aux tickets
 - 6.4. Extraction manuelle de quelques tickets pour analyse
 - 6.5. Identification des difficultés potentielles
-

7. Extraction Automatisée des Tickets JIRA

- 7.1. Recherche sur l'API JIRA et ses fonctionnalités
 - 7.2. Implémentation d'un script Python pour l'extraction
 - 7.3. Extraction initiale et vérification des données
 - 7.4. Ajustements du script pour inclure tous les champs nécessaires
 - 7.5. Documentation du processus
-

8. Nettoyage et Stockage des Tickets

- 8.1. Formatage des données extraites
 - 8.2. Sauvegarde des données dans une base SQL ou CSV
-

9. Analyse Exploratoire des Tickets & Étude des relations entre descriptions, logs et commentaires

- 9.1. Analyse des distributions des tickets par module et feature
 - 9.2. Identification des patterns récurrents dans les descriptions
 - 9.3. Visualisation des tickets en fonction des assignees
 - 9.4. Analyser la corrélation entre description, logs et commentaires
 - 9.5. Analyser les fréquences d'apparition des erreurs
 - 9.6. Étudier les tickets les plus récurrents et complexes
 - 9.7. Identification des catégories de tickets à partir des descriptions
-

9.8. Définir une stratégie de prétraitement des commentaires et descriptions

9.9. Documentation des premiers insights

10. Préparation des Données

10.1. Prétraitement avec NLP

- 10.1.1. Nettoyage des descriptions et commentaires (stopwords, caractères spéciaux, normalisation).
- 10.1.2. Lemmatisation et stemming pour standardiser les termes.
- 10.1.3. Extraction des entités nommées (NER) : modules, composants, erreurs
- 10.1.4. Vectorisation des données (TF-IDF, Word2Vec, Sentence-BERT).

10.2. Utilisation des LLMs pour la structuration des données

- 10.2.1. Développer des prompts pour extraire les entités clés
- 10.2.2. Expérimenter plusieurs modèles LLMs (ex. Llama-2, deepseek, gemma, ;;)
- 10.2.3. Comparer la structuration NLP vs. LLMs
- 10.2.4. Documenter la méthodologie retenue

11. Regroupement des Tickets

11.1. Clustering des Tickets

- 11.1.1. Recherche des meilleures techniques de clustering (K-Means, DBSCAN)
- 11.1.2. Implémentation et expérimentation
- 11.1.3. Analyse des résultats et validation
- 11.1.4. Documentation et ajustements

11.2. Analyse Sémantique et Similarité

- 11.2.1. Utilisation d'embeddings (Sentence-BERT, OpenAI Embeddings)
- 11.2.2. Calcul de similarité cosine entre tickets

- 11.2.3. Expérimentation de différentes métriques
- 11.2.4. Visualisation des groupes de tickets
- 11.2.5. Documentation des observations

12. Création et mise en place de la KEDB

- 12.1. Mise en place du pipeline d'insertion
- 12.2. Structuration de la base de connaissances
- 12.3. Intégrer les logs extraits et les groupes de tickets et les autres informations
- 12.4. Tester les requêtes et l'indexation
- 12.5. Documenter la mise en œuvre

13. Développement de l'API pour interroger la KEDB

- 13.1. Concevoir l'API backend pour la récupération des tickets
- 13.2. Implémenter des filtres et requêtes optimisées
- 13.3. Tester et ajuster les résultats
- 13.4. Documenter les choix techniques

14. Recherche Sémantique et Extraction des Tickets Similaires

14.1. Mise en place d'une recherche vectorielle

- 14.1.1. Étudier les bases vectorielles FAISS, ChromaDB, Elasticsearch
- 14.1.2. Expérimenter différentes métriques de similarité
- 14.1.3. Implémenter la recherche vectorielle sur les tickets
- 14.1.4. Tester et ajuster les résultats
- 14.1.5. Documenter les choix techniques

14.2. Optimisation et validation de la recherche

- 14.2.1. Comparer les performances des moteurs de recherche
- 14.2.2. Ajuster la pondération des critères de similarité

14.2.3. Valider les résultats avec des experts

14.2.4. Finaliser la documentation technique

15. Analyse des Tickets Similaires avec LLMs et Extraction des Caractéristiques

15.1. Expérimentation et comparaison des modèles LLMs

15.1.1. Sélection des modèles LLMs les plus adaptés

15.1.2. Développement de prompts avancés

15.1.3. Expérimentation sur extraction des causes, solutions et d'autres insights

15.1.4. Comparaison des performances et validation des résultats

15.1.5. Documentation technique détaillée

15.2. Intégration et Optimisation

15.2.1. Mise en place de la récupération de contexte avec la base vectorielle

15.2.2. Analyse des performances du système

15.2.3. Optimisation des prompts pour améliorer la précision des résultats

15.2.4. Affinage des critères d'extraction

15.2.5. Intégration des résultats dans le pipeline global

15.2.6. Validation globale du système

15.2.7. Rédaction de la documentation finale

16. Développement du Chatbot Intelligent

16.1. Définition des scénarios d'interaction et du parcours utilisateur

16.2. Développement du backend et des APIs

16.3. Connexion avec la KEDB et le moteur de recherche vectoriel

16.4. Tests fonctionnels et ajustements

16.5. Documentation technique

17. Optimisation et tests finaux

- 17.1. Vérification de bout en bout (Pipeline complet)**
 - 17.2. Tests de performance et corrections finales**
 - 17.3. Préparation des résultats et mise en production**
 - 17.4. Validation avec tests utilisateurs**
 - 17.5. Derniers ajustements et tests finaux**
 - 17.6. Rédaction de la documentation finale**
-