**KẾ HOẠCH PHÁT TRIỂN HỆ THỐNG AI**

## I. CẢI TIẾN DOLPHIN CHATBOT AI

### 1. Thực trạng vấn đề

**Hạn chế của chatbot hiện tại:**

* Chỉ có khả năng xử lý một số câu hỏi cơ bản về chứng khoán, chưa linh hoạt với câu hỏi phức tạp.
* Khó mở rộng tính năng chuyên sâu (phân tích kỹ thuật, phân tích cơ bản, tư vấn danh mục,...) do chưa có kiến trúc phân lớp hoặc mô hình NLP chuyên biệt.
* Hiệu suất xử lý chưa tối ưu, phản hồi chậm khi lượng người dùng tăng.
* Độ chính xác chưa cao, thiếu dẫn chứng và trích nguồn rõ ràng.
* Khả năng mở rộng hạn chế do chưa áp dụng microservice, chưa có pipeline MLOps hoàn chỉnh.

**Vấn đề về dữ liệu:**

* Chưa có quy trình tiền xử lý dữ liệu để tạo chỉ số sẵn có (precomputed stock data).
* Thiếu kho dữ liệu vector (vector database) phục vụ tìm kiếm ngữ nghĩa (semantic search).
* Dữ liệu người dùng và lịch sử truy vấn chưa được quản lý tập trung, hạn chế khả năng cá nhân hóa.

**Hạn chế về bảo mật:**

* Chưa có lớp bảo mật & tuân thủ (security & compliance layer) đầy đủ.

### 2. Giải pháp và phạm vi áp dụng

#### 2.1. Giải pháp tổng quan

* Xây dựng kiến trúc **microservice** với các module tách biệt: Data Pipeline, LLM Backend, Database, Frontend/Voice.
* Triển khai **MLOps Pipeline** để tự động hóa huấn luyện, triển khai, giám sát mô hình.
* **Nâng cao độ chính xác** bằng cách cập nhật dữ liệu chứng khoán thường xuyên, thêm nguồn dẫn chứng, áp dụng validator và feedback loop.
* **Tối ưu hiệu suất** bằng cách chia chatbot thành nhiều mô hình chuyên biệt, sử dụng caching (Redis), streaming (Websocket), và scale hạ tầng ngang/dọc.
* **Tăng cường bảo mật** với lớp Security & Compliance, kiểm soát quyền truy cập, mã hóa dữ liệu quan trọng.

#### 2.2. Chi tiết tính năng chatbot

* **Smart Data Query Assistant:** Hỗ trợ truy vấn dữ liệu chứng khoán, chỉ số thị trường, tin tức.
* **Technical Stock Analysis Support:** Phân tích kỹ thuật, vẽ biểu đồ chỉ báo (RSI, MACD,...).
* **Fundamental Stock Analysis Support:** Truy vấn & phân tích chỉ số cơ bản (EPS, P/E, ROE...).
* **Market News Summary & Analysis:** Tóm tắt tin tức thị trường, phân tích sentiment.
* **Stock Filtering:** Cho phép người dùng lọc cổ phiếu theo tiêu chí cụ thể.
* **Personal Portfolio Analysis:** Phân tích danh mục cá nhân, đề xuất tái cân bằng danh mục.
* **Customer Service Assistant:** Hỗ trợ câu hỏi về thủ tục mở tài khoản, giao dịch.
* **Legal Supporter:** Cung cấp thông tin về quy định pháp lý, cảnh báo nội dung nhạy cảm.
* **Investment Knowledge Advisor:** Hướng dẫn về đầu tư, thuật ngữ tài chính, case study.

#### 2.3. Phạm vi áp dụng

* **Nền tảng:** Website, mobile app, API tích hợp (Zalo OA, Facebook Messenger).
* **Ngôn ngữ:** Tiếng Việt (có thể mở rộng sang tiếng Anh).
* **Đối tượng:** Nhà đầu tư cá nhân, tư vấn viên nội bộ, CSKH.

#### 2.4. Mục tiêu

* **Thời gian phản hồi trung bình:** < 2 giây.
* **Độ chính xác câu trả lời:** > 80%.
* **Giảm 50% khối lượng công việc CSKH** (tính theo số ticket xử lý thủ công).
* **Tăng chỉ số hài lòng khách hàng (NPS/CSAT)** so với chatbot cũ.

### 3. Kế hoạch phát triển

#### 3.1. Lộ trình triển khai (5 tháng)

**Giai đoạn 1: Khởi tạo & Thiết kế (Tháng 1)**

* Phân tích hệ thống cũ, xác định yêu cầu kỹ thuật & kinh doanh.
* Thiết kế kiến trúc microservice, lựa chọn công nghệ phù hợp.
* Thiết lập môi trường phát triển, triển khai MLOps pipeline.

**Giai đoạn 2: Xây dựng & Tích hợp tính năng cơ bản (Tháng 2 - 3)**

* Xây dựng **data pipeline** để thu thập & xử lý dữ liệu chứng khoán, tin tức, logs người dùng.
* Tích hợp **API Gateway, caching (Redis), Websocket** để tối ưu hiệu suất.
* Phát triển **LLM backend** và các module chính: truy vấn dữ liệu, phân tích kỹ thuật/cơ bản, tóm tắt tin tức.

**Giai đoạn 3: Hoàn thiện & Tối ưu (Tháng 4)**

* Triển khai các tính năng nâng cao: **Legal Supporter, Investment Knowledge Advisor**.
* Tối ưu hiệu suất: chia nhỏ mô hình, cải thiện độ chính xác, tối ưu hạ tầng (Kubernetes scaling).
* Cải thiện bảo mật: mã hóa dữ liệu, kiểm soát quyền truy cập.

**Giai đoạn 4: Triển khai thử nghiệm & Go-live (Tháng 5)**

* Chạy thử nghiệm (UAT), đo lường hiệu suất & độ chính xác.
* A/B testing với chatbot cũ để đánh giá cải tiến.
* Fix lỗi, tối ưu lần cuối trước khi triển khai chính thức.
* Giám sát chatbot, thu thập phản hồi người dùng để tiếp tục cải tiến.

#### 3.2. Kiểm thử và quản lý chất lượng

* **Kiểm thử chức năng:** Đảm bảo từng module hoạt động đúng.
* **Kiểm thử hiệu suất:** Stress test trên Kubernetes, đo độ trễ, throughput.
* **Kiểm thử bảo mật:** Kiểm tra lỗ hổng bảo mật, tuân thủ chính sách an toàn.
* **A/B Testing:** So sánh chatbot mới với phiên bản cũ.

#### 3.3. Quản lý rủi ro

* **Rủi ro dữ liệu:** Thiếu dữ liệu hoặc cập nhật chậm → Xây dựng pipeline thu thập đáng tin cậy.
* **Rủi ro hiệu năng:** Tăng đột biến lượng người dùng → Kubernetes autoscale.
* **Rủi ro bảo mật:** Rò rỉ dữ liệu khách hàng → Thực thi mã hóa dữ liệu nhạy cảm.
* **Sai lệch mô hình:** Chatbot cung cấp thông tin sai → Thêm validator, feedback loop.

#### 3.4. Phương hướng cải tiến (Sau triển khai 5 tháng)

* **Hỗ trợ đa ngôn ngữ (Anh - Việt).**
* **Bổ sung tính năng Voice Assistant.**
* **Gợi ý câu hỏi liên quan để giúp người dùng tìm kiếm thông tin dễ dàng hơn.**
* **Smart Actions:** Thực hiện hành động theo yêu cầu người dùng (tạo watchlist, cảnh báo thông minh).
* **Mở rộng hệ thống MLOps:** Tự động retrain, giám sát drift.
* **Tích hợp chatbot với hệ thống bên ngoài (Zalo OA, Messenger, Email).**

## **KẾ HOẠCH PHÁT TRIỂN HỆ THỐNG AI NEWS**

#### **1. Thực trạng vấn đề**

**Hạn chế của hệ thống hiện tại:**

* **Lượng tin tức lớn và đa nguồn:** Công ty nhận hàng trăm đến hàng nghìn bài viết mỗi ngày từ API hoặc crawler, gây quá tải khi phân tích thủ công.
* **Thiếu hệ thống phân tích tự động:** Chưa có phân loại chủ đề, phân tích sentiment hoặc trích xuất mã cổ phiếu.
* **Khó theo dõi xu hướng thị trường:** Thiếu công cụ phân tích theo ngữ cảnh khiến việc nắm bắt tin nóng, tâm lý thị trường chưa hiệu quả.
* **Thiếu hỗ trợ ra quyết định:** Không có dashboard hay mô hình ML cung cấp insight cho bộ phận phân tích và đầu tư.
* **Tính năng “Tin tức thành Podcast” chưa có:** Người dùng muốn nghe tin tức nhưng hệ thống chưa hỗ trợ text-to-speech (TTS) chuyên biệt.

#### **2. Giải pháp và phạm vi áp dụng**

##### **2.1. Giải pháp tổng quan**

* **Xây dựng pipeline NLP tự động:**
  + Thu nhận tin tức từ API/crawler → Tiền xử lý → Phân tích sentiment, phân loại topic → Trích xuất mã cổ phiếu → Tóm tắt tin tức → Tính điểm cảm xúc cổ phiếu → Lưu trữ và hiển thị.
* **Ứng dụng mô hình ngôn ngữ hiện đại:**
  + Sentiment Analysis: Phân tích cảm xúc (BERT, PhoBERT, GPT…).
  + Topic/Sector Segmentation: Phân loại tin tức theo ngành.
  + Named Entity Recognition (NER): Trích xuất tên công ty, mã cổ phiếu.
  + Summarization: Tóm tắt tin tức tự động.
  + Text-to-Speech: Chuyển tin tức thành giọng nói (TTS).
* **Mở rộng Social Sentiment:**
  + Tích hợp dữ liệu mạng xã hội để đánh giá quan điểm cộng đồng với cổ phiếu/ngành.
* **Triển khai hạ tầng MLOps:**
  + Cập nhật, retrain mô hình khi có dữ liệu mới, giám sát hiệu năng hệ thống.

##### **2.2. Chi tiết tính năng**

* **Sentiments:** Xác định cảm xúc của bài viết (tích cực / tiêu cực / trung tính).
* **Topic/Sector Segmentation:** Phân loại tin tức theo ngành (chứng khoán, ngân hàng…).
* **Mentioned Stock Detection:** Trích xuất mã cổ phiếu, theo dõi tần suất xuất hiện và sentiment liên quan.
* **Summary:** Tóm tắt tin tức giúp người dùng nắm bắt nội dung nhanh hơn.
* **Sentiment Stock Score:** Tính điểm sentiment trung bình cho cổ phiếu theo ngày/tuần/tháng.
* **Social Sentiment:** Phân tích cảm xúc và mức độ bàn luận trên mạng xã hội.

##### **2.3. Phạm vi áp dụng**

* **Nguồn dữ liệu:** API tin tức, dữ liệu crawler, mạng xã hội (tùy chọn).
* **Đối tượng sử dụng:** Bộ phận phân tích, tư vấn đầu tư, marketing, nhà đầu tư cá nhân.
* **Kênh xuất bản kết quả:** Web/Mobile App, API tích hợp với chatbot, dashboard phân tích real-time.

##### **2.4. Mục tiêu**

* **Tăng tốc độ xử lý:** Tin tức phân tích trong vòng 10s sau khi nhận.
* **Độ chính xác mô hình:**
  + Sentiment đạt 80-85% F1-Score.
  + Phân loại ngành đạt ít nhất 85% độ chính xác.
  + Tóm tắt đạt ROUGE/BLEU ≥ 0.6-0.7.
* **Giảm tải cho đội phân tích:** Ít nhất 50% tin tức được phân loại/tóm tắt tự động.
* **Khả năng mở rộng:** Xử lý ≥ 1.000 bài/ngày mà không quá tải.
* **Thời gian triển khai:** 2 tháng để có phiên bản hoàn thiện.

### **3. Kế hoạch phát triển**

#### **3.1. Lộ trình triển khai (2 tháng)**

##### **Giai đoạn 1: Thiết lập hạ tầng & Xây dựng pipeline cơ bản**

* Thiết kế kiến trúc tổng quan và lựa chọn công nghệ phù hợp.
* Thiết lập môi trường phát triển và hệ thống lưu trữ dữ liệu.
* Xây dựng data pipeline để thu thập tin tức từ API và crawler.
* Triển khai cơ chế tiền xử lý dữ liệu: loại bỏ trùng lặp, làm sạch dữ liệu.
* Thiết lập cơ sở dữ liệu lưu trữ tin tức và metadata.

##### **Giai đoạn 2: Phát triển mô hình phân tích dữ liệu**

* Huấn luyện mô hình phân tích cảm xúc (Sentiment Analysis).
* Xây dựng mô hình phân loại chủ đề (Topic/Sector Segmentation).
* Tích hợp Named Entity Recognition (NER) để trích xuất mã cổ phiếu.
* Xây dựng hệ thống tóm tắt tin tức tự động.
* Phát triển hệ thống tính điểm cảm xúc cổ phiếu (Sentiment Stock Score).
* Kiểm thử mô hình trên tập dữ liệu thực tế.

##### **Giai đoạn 3: Tích hợp, tối ưu hóa và kiểm thử**

* Tích hợp các mô hình AI vào hệ thống backend.
* Phát triển giao diện dashboard để hiển thị kết quả phân tích.
* Tối ưu hiệu suất pipeline: caching, batch processing, cải thiện tốc độ xử lý.
* Kiểm thử tổng thể hệ thống: đo tải, đánh giá độ chính xác và phản hồi của người dùng.
* Triển khai hệ thống cảnh báo sự kiện khi sentiment thay đổi đột ngột.

##### **Giai đoạn 4: Hoàn thiện và triển khai chính thức**

* Tích hợp tính năng chuyển đổi tin tức thành podcast (Text-to-Speech).
* Kiểm tra bảo mật và quyền truy cập dữ liệu.
* Tinh chỉnh mô hình AI dựa trên phản hồi từ thử nghiệm thực tế.
* Chạy thử nghiệm giới hạn với nhóm người dùng nhỏ.
* Triển khai hệ thống chính thức và giám sát vận hành.

#### **3.2. Kiểm thử và quản lý chất lượng**

* **Kiểm thử chức năng:** Đảm bảo pipeline thu thập, phân tích, tóm tắt hoạt động đúng.
* **Kiểm thử hiệu suất:** Đo throughput, latency, khả năng mở rộng.
* **Kiểm thử mô hình:** Đánh giá độ chính xác sentiment, phân loại topic, summarization.
* **Kiểm thử bảo mật:** Kiểm tra quyền truy cập, ẩn danh dữ liệu (anonymization).
* **A/B testing:** So sánh với hệ thống cũ để đánh giá cải tiến.

#### **3.3. Quản lý rủi ro**

* **Chất lượng dữ liệu không ổn định:** Áp dụng quy trình làm sạch & kiểm soát chất lượng.
* **Khối lượng tin tức lớn:** Dùng batch & streaming processing, scale hạ tầng.
* **Sai sót mô hình:** Giám sát độ chính xác, tự động retrain khi cần.
* **Thiếu nhân sự chuyên môn:** Cân đối nguồn lực, phân chia task hợp lý.

#### **3.4. Phương hướng cải tiến**

* **Mở rộng nguồn tin tức:** Tích hợp thêm mạng xã hội, các trang tài chính.
* **Thêm tính năng gợi ý thông tin (recommendation):** Dự đoán tin tức quan trọng.
* **Mở rộng hệ thống Podcast:** Cho phép người dùng nghe tin tức dễ dàng hơn.
* **Phân tích mối quan hệ giữa các sự kiện tài chính.**
* **Tích hợp chatbot AI:** Trả lời câu hỏi về tin tức theo thời gian thực.
* **Cảnh báo sự kiện:** Gửi thông báo khi sentiment cổ phiếu thay đổi đột ngột.

## **KẾ HOẠCH PHÁT TRIỂN MÔ HÌNH PHÂN TÍCH KHÁCH HÀNG PHỤC VỤ BUSINESS**

(Phân loại khách hàng, xác định khả năng churn, tìm kiếm khách hàng tiềm năng - Thời gian: 3 tháng)

### **I. CẢI TIẾN HỆ THỐNG PHÂN TÍCH KHÁCH HÀNG**

#### **1. Thực trạng vấn đề**

* **Thiếu phân khúc khách hàng rõ ràng:** Khó đưa ra chiến dịch marketing nhắm đúng đối tượng.
* **Tỷ lệ churn (rời bỏ dịch vụ) tăng:** Không có công cụ dự đoán để can thiệp sớm.
* **Nguồn khách hàng tiềm năng lớn từ công ty mẹ chưa được khai thác tối ưu:** Chưa có mô hình lead scoring để đánh giá.
* **Dữ liệu khách hàng phân tán, chưa được chuẩn hóa:** Gây khó khăn khi xây dựng mô hình thống nhất.

### **II. GIẢI PHÁP VÀ PHẠM VI ÁP DỤNG**

#### **2.1. Giải pháp tổng quan**

* **Xây dựng mô hình phân loại khách hàng (Customer Segmentation):**
  + Dựa trên RFM (Recency, Frequency, Monetary), nhân khẩu học và hành vi giao dịch.
  + Sử dụng các thuật toán như K-Means, hierarchical clustering.
* **Mô hình dự đoán churn (Churn Prediction):**
  + Áp dụng supervised learning (Logistic Regression, XGBoost…) để phát hiện khách có nguy cơ rời bỏ.
* **Lead Scoring:**
  + Xếp hạng khách hàng từ công ty mẹ theo xác suất chuyển đổi.
  + Sử dụng dữ liệu hồ sơ tài chính, mức độ tương tác và hành vi giao dịch.
* **Tận dụng Feature Store:**
  + Đồng bộ dữ liệu từ dự án Feature Store để tránh trùng lặp, cải thiện chất lượng mô hình.

#### **2.2. Chi tiết tính năng**

* **Phân loại khách hàng (Segmentation):**
  + Chia khách hàng thành các nhóm như VIP, tiềm năng, ngủ đông,…
  + Xây dựng dashboard hiển thị quy mô và đặc điểm từng nhóm.
* **Churn Prediction:**
  + Xác định khách có nguy cơ rời bỏ, gửi cảnh báo cho đội CSKH, marketing.
  + Cập nhật chỉ số churn risk theo tuần/tháng.
* **Lead Scoring:**
  + Xếp hạng khách hàng theo khả năng chuyển đổi từ danh sách công ty mẹ.
  + Giúp đội marketing/sales ưu tiên tiếp cận nhóm tiềm năng.

#### **2.3. Phạm vi áp dụng**

* **Dữ liệu chính:**
  + CRM, lịch sử giao dịch, event hành vi (log web/app), dữ liệu khách hàng từ công ty mẹ.
* **Bộ phận sử dụng:**
  + Marketing, CSKH, Sales.
* **Mô hình dự đoán:**
  + Hỗ trợ batch inference (dự báo churn, lead scoring).
  + Có thể mở rộng real-time nếu tích hợp pipeline streaming.

#### **2.4. Mục tiêu**

* **Giảm 20% tỷ lệ churn** sau khi áp dụng mô hình.
* **Tăng 15% tỷ lệ chuyển đổi khách hàng tiềm năng** so với trước.
* **Phân khúc khách hàng rõ ràng**, hỗ trợ tối ưu chiến dịch marketing.
* **Thời gian triển khai:** 3 tháng để có mô hình chính thức và tích hợp vào quy trình kinh doanh.

### **III. KẾ HOẠCH PHÁT TRIỂN**

#### **3.1. Lộ trình triển khai (3 tháng)**

##### **Giai đoạn 1: Thu thập & Chuẩn hóa dữ liệu**

* Kết nối và đồng bộ dữ liệu từ CRM, lịch sử giao dịch, logs hành vi khách hàng.
* Chuẩn hóa dữ liệu: xử lý trùng lặp, dữ liệu thiếu, tạo chỉ số đặc trưng (feature engineering).
* Xác định KPI churn và lead scoring phù hợp với business.

##### **Giai đoạn 2: Xây dựng & Huấn luyện mô hình**

* Phát triển mô hình phân loại khách hàng (segmentation) dựa trên K-Means, RFM clustering.
* Xây dựng mô hình churn prediction: thử nghiệm Logistic Regression, XGBoost, chọn mô hình tốt nhất dựa trên ROC-AUC, F1-score.
* Phát triển mô hình lead scoring, sử dụng dữ liệu từ công ty mẹ để xác định xác suất chuyển đổi.

##### **Giai đoạn 3: Tích hợp & Kiểm thử**

* Tích hợp mô hình churn vào quy trình CSKH và dashboard theo dõi churn risk.
* Đưa lead scoring vào quy trình marketing/sales, giúp ưu tiên tiếp cận nhóm khách hàng tiềm năng.
* Kiểm thử mô hình trên tập dữ liệu thực tế, đo lường độ chính xác.
* Thu thập phản hồi từ bộ phận kinh doanh và tinh chỉnh mô hình.

##### **Giai đoạn 4: Tối ưu hóa & Triển khai chính thức**

* Tối ưu mô hình segmentation, churn prediction và lead scoring.
* Xây dựng pipeline MLOps để tự động cập nhật mô hình khi có dữ liệu mới.
* Thiết lập lịch retrain hàng tháng/quý, giám sát drift của mô hình.
* A/B testing: chạy thử nghiệm trên một phần khách hàng để so sánh hiệu quả trước/sau.
* Chính thức go-live và đo lường kết quả: churn giảm, lead conversion tăng.

#### **3.2. Kiểm thử & Quản lý chất lượng**

* **Kiểm thử mô hình:**
  + **Segmentation:** Đánh giá trực quan (bản đồ phân khúc), kiểm tra tính ổn định của cluster.
  + **Churn Prediction:** Đánh giá bằng ROC-AUC, F1-score, Precision/Recall.
  + **Lead Scoring:** Đánh giá qua tỷ lệ chuyển đổi (conversion rate).
* **Thử nghiệm triển khai:**
  + Chạy thử nghiệm trên một phân khúc nhỏ khách hàng, so sánh kết quả trước/sau.
* **Kiểm thử bảo mật:**
  + Đảm bảo dữ liệu khách hàng được bảo vệ trong suốt quá trình xử lý.

#### **3.3. Quản lý rủi ro**

* **Thiếu nhãn churn:** Chưa có định nghĩa rõ ràng về thời gian không giao dịch để xác định churn → Cần thống nhất cách tính.
* **Dữ liệu từ công ty mẹ không đầy đủ:** Phải xử lý missing data và chuẩn hóa.
* **Khó khăn phối hợp giữa Data & Marketing:** Cần một Business Analyst làm trung gian hỗ trợ trao đổi.
* **Hiệu quả mô hình không như kỳ vọng:** Cần theo dõi, tối ưu hyperparameters, cập nhật dữ liệu liên tục.

#### **3.4. Phương hướng cải tiến**

* **Mở rộng mô hình sang upsell/cross-sell:** Gợi ý sản phẩm phù hợp cho từng nhóm khách hàng.
* **Tự động gợi ý chiến dịch marketing cá nhân hóa (Personalized Campaign).**
* **Online learning:** Cập nhật mô hình churn/lead scoring theo thời gian thực.
* **Phân tích xu hướng khách hàng:** Thêm yếu tố thời gian, seasonal patterns, deep learning để dự đoán hành vi tốt hơn.

## **KẾ HOẠCH PHÁT TRIỂN HỆ THỐNG PHÂN TÍCH VOC (VOICE OF CUSTOMER) VÀ TICKET**

(Phân loại và đánh nhãn sentiment - Thời gian: 2 tháng)

### **I. CẢI TIẾN HỆ THỐNG PHÂN TÍCH VOC & TICKET**

#### **1. Thực trạng vấn đề**

* **Số lượng VOC & ticket lớn:** Bộ phận CSKH nhận nhiều phản hồi mỗi ngày qua email, call, chat, form online, gây quá tải khi phân loại thủ công.
* **Thiếu gán nhãn sentiment:** Hiện tại chưa có cơ chế tự động đánh giá mức độ hài lòng hay tiêu cực của khách hàng, gây khó khăn trong việc ưu tiên xử lý.
* **Dữ liệu thô:** VOC/ticket chưa được phân loại theo chủ đề (phí, hỗ trợ kỹ thuật, khiếu nại…), làm giảm hiệu suất làm việc của CSKH.
* **Định hướng muốn sử dụng LLM API:** Dự kiến ứng dụng OpenAI GPT hoặc các LLM khác để tự động phân loại, phân tích nội dung.

### **II. GIẢI PHÁP VÀ PHẠM VI ÁP DỤNG**

#### **2.1. Giải pháp tổng quan**

* **Thu thập & tiền xử lý VOC/ticket:**
  + Trích xuất nội dung text từ các kênh (email, chat, ghi âm call → chuyển thành văn bản bằng STT).
* **Phân loại chủ đề (Topic Classification):**
  + Dùng LLM API hoặc fine-tuned model để gán nhãn như “Phàn nàn về phí”, “Vấn đề kỹ thuật”, “Yêu cầu thông tin”…
* **Phân tích sentiment:**
  + Xác định mức độ “Tiêu cực”, “Trung tính”, “Tích cực” để đánh giá cảm xúc khách hàng.
* **Sử dụng LLM API:**
  + Áp dụng prompt engineering và custom instructions để LLM trả về kết quả dạng structured JSON (chứa topic, sentiment, độ tin cậy…).
  + Xây dựng feedback loop để cải thiện độ chính xác.

#### **2.2. Chi tiết tính năng**

* **Phân loại ticket theo chủ đề:**
  + Tự động điền thông tin “Loại vấn đề” (issue type) và “Mức độ ưu tiên” (priority).
* **Sentiment Analysis:**
  + Đánh giá mức độ hài lòng của khách hàng, highlight các ticket có mức độ bức xúc cao cần xử lý gấp.
* **Dashboard giám sát:**
  + Thống kê theo thời gian, tỷ lệ VOC tiêu cực/tích cực, các chủ đề được hỏi nhiều nhất.
* **Cảnh báo real-time (nếu cần):**
  + Nếu hệ thống phát hiện sentiment quá tiêu cực, tự động gửi cảnh báo đến quản lý CSKH.

#### **2.3. Phạm vi áp dụng**

* **Dữ liệu VOC:**
  + Email, chat, phản hồi trên cổng feedback, ghi âm cuộc gọi (được chuyển thành text).
* **Dữ liệu ticket:**
  + Tích hợp với hệ thống ticketing (Zendesk, ServiceNow, Jira, hoặc hệ thống nội bộ).
* **Đối tượng sử dụng:**
  + Đội CSKH, chăm sóc khách hàng VIP, quản lý chất lượng.

#### **2.4. Mục tiêu**

* **Tự động phân loại ≥ 80% ticket/VOC** mà không cần con người can thiệp ban đầu.
* **Độ chính xác sentiment đạt trên 85% F1-score** (đo trên tập dữ liệu gắn nhãn thủ công).
* **Tiết kiệm 50% thời gian đội CSKH** dành cho việc sàng lọc, phân loại VOC/ticket.
* **Thời gian triển khai:** 2 tháng để hoàn thiện hệ thống MVP.

### **III. KẾ HOẠCH PHÁT TRIỂN**

#### **3.1. Lộ trình triển khai (2 tháng)**

##### **Giai đoạn 1: Phân tích & Chuẩn bị dữ liệu**

* Xác định danh sách chủ đề chính (phí, khiếu nại, vấn đề kỹ thuật…) và khung sentiment.
* Kết nối và tích hợp dữ liệu VOC/ticket từ các nguồn email, chat, call logs.
* Xây dựng pipeline ETL để làm sạch, chuẩn hóa dữ liệu text.

##### **Giai đoạn 2: Xây dựng & Huấn luyện mô hình**

* Thiết kế prompt/API call cho LLM, xác định format output JSON chứa topic, sentiment.
* Thử nghiệm các cách tiếp cận: Prompt Engineering hoặc Fine-Tuning (nếu mô hình hỗ trợ).
* Xây dựng prototype, chạy test trên một phần dữ liệu VOC/ticket và đánh giá F1-score.

##### **Giai đoạn 3: Tích hợp & Tối ưu hóa**

* Triển khai dashboard giám sát: hiển thị tỷ lệ chủ đề, sentiment, cảnh báo quan trọng.
* Tối ưu hóa hiệu suất xử lý: điều chỉnh batch processing vs. real-time API call.
* Thiết lập feedback loop để CSKH có thể sửa nhãn, cải thiện độ chính xác mô hình.

##### **Giai đoạn 4: Kiểm thử & Triển khai chính thức**

* Kiểm thử tổng thể trên tập dữ liệu thực tế, đo lường độ chính xác của hệ thống.
* Tích hợp vào hệ thống ticketing chính thức, đảm bảo khả năng hoạt động ổn định.
* Đào tạo đội CSKH sử dụng hệ thống, chính thức go-live và theo dõi vận hành.

#### **3.2. Kiểm thử & Quản lý chất lượng**

* **Kiểm thử độ chính xác:**
  + So sánh nhãn LLM với nhãn do CSKH/QA gắn thủ công, tính toán F1-score, Precision, Recall.
* **Kiểm thử hiệu suất:**
  + Đo lường thời gian xử lý/truy vấn LLM API.
  + Đánh giá khả năng hệ thống khi nhiều ticket đổ vào cùng lúc.
* **Kiểm thử bảo mật:**
  + Đảm bảo dữ liệu khách hàng được ẩn danh hoặc mã hóa trước khi gửi đến LLM API.
* **Pilot triển khai:**
  + Chạy thử nghiệm trên một số phòng ban trước khi mở rộng toàn công ty.

#### **3.3. Quản lý rủi ro**

* **Chi phí API:**
  + Gọi LLM API (ChatGPT, GPT-4) có thể tốn phí cao nếu volume ticket lớn → cần tối ưu cache, batch processing, hoặc thương lượng pricing.
* **Độ chính xác trên chủ đề hiếm (rare topic):**
  + Nếu chủ đề quá ít dữ liệu, cần thu thập thêm ví dụ hoặc tinh chỉnh prompt/model.
* **Bảo mật dữ liệu khách hàng:**
  + Đảm bảo dữ liệu nhạy cảm được ẩn danh hoặc truncate trước khi gửi đến API.
* **Chậm trễ do mạng hoặc API downtime:**
  + Xây dựng fallback solution, sử dụng queue processing khi API gặp sự cố.

#### **3.4. Phương hướng cải tiến**

* **Tự động gợi ý đáp án (Auto-reply):**
  + Phát triển tính năng phản hồi tự động cho các ticket đơn giản (FAQ).
* **Hỗ trợ đa ngôn ngữ:**
  + Mở rộng phân tích VOC/ticket bằng nhiều ngôn ngữ khác nhau.
* **Triển khai “voice sentiment analysis”:**
  + Nếu call center có ghi âm, tích hợp STT (speech-to-text) để phân tích cảm xúc khách hàng trong cuộc gọi.
* **Nâng cấp mô hình:**
  + Nếu dữ liệu nhạy cảm, có thể sử dụng mô hình LLM nội bộ (on-prem) hoặc fine-tune một mô hình chuyên biệt.

## **KẾ HOẠCH PHÁT TRIỂN HỆ THỐNG FEATURE STORE PHỤC VỤ CÁC BÀI TOÁN BUSINESS**

(Thời gian triển khai: 3 tháng)

### **I. CẢI TIẾN HỆ THỐNG QUẢN LÝ FEATURE**

#### **1. Thực trạng vấn đề**

* **Dữ liệu rời rạc:**
  + Dữ liệu từ nhiều hệ thống (CRM, giao dịch, dữ liệu công ty mẹ, dữ liệu bên ngoài…) được lưu trữ riêng rẽ, gây khó khăn trong việc xử lý và sử dụng.
  + Các nhóm phân tích phải lặp lại quá trình làm sạch, chuẩn hóa, biến đổi dữ liệu cho từng dự án.
* **Thiếu “single source of truth” cho feature:**
  + Mỗi dự án ML/DS thường tự tạo và tính toán feature riêng, dẫn đến thiếu thống nhất, khó chia sẻ, sai khác logic.
* **Khó mở rộng mô hình:**
  + Khi bổ sung tính năng mới hoặc thay đổi quy trình, việc cập nhật feature cho nhiều mô hình trở nên phức tạp, dễ phát sinh lỗi.
* **Chậm đưa mô hình vào sản xuất:**
  + Thời gian đưa mô hình vào production kéo dài do phải xử lý ETL thủ công và cập nhật dữ liệu không đồng nhất.

### **II. GIẢI PHÁP VÀ PHẠM VI ÁP DỤNG**

#### **2.1. Giải pháp tổng quan**

* **Thiết lập Feature Store:**
  + Xây dựng kho trung tâm quản lý toàn bộ feature, hỗ trợ cả batch (offline) và near real-time (online).
  + Quản lý version của feature, lưu lại logic tính toán và thời gian hiệu lực (time-travel).
  + Cung cấp API hoặc SDK để Data Scientist, ML Engineer truy cập feature đồng nhất.
* **Xây dựng kiến trúc pipeline:**
  + **Data ingestion:** Kết nối với CRM, lịch sử giao dịch, dữ liệu công ty mẹ, dữ liệu bên ngoài…
  + **Data processing & feature transformation:** Làm sạch, tổng hợp, encoding, chuẩn hóa dữ liệu.
  + **Serving layer:** Cung cấp feature cho mô hình ML ở cả batch và near real-time, đồng bộ giữa môi trường training và inference.
* **Kết hợp MLOps:**
  + Tích hợp CI/CD để tự động triển khai pipeline.
  + Giám sát chất lượng dữ liệu (data quality) và drift của feature theo thời gian.
  + Tự động ghi log, cảnh báo khi có sai lệch hoặc sự cố.

#### **2.2. Chi tiết tính năng**

* **Feature Catalog:**
  + Danh sách tất cả feature (transaction\_count\_30days, avg\_balance\_90days, customer\_segment…).
  + Mô tả chi tiết logic tính toán, ngày cập nhật và mục đích sử dụng.
* **Versioning & Time Travel:**
  + Giúp quay lại phiên bản feature đúng thời điểm mô hình được huấn luyện, đảm bảo tính tái lập kết quả.
* **Real-time/Batch Serving:**
  + **Batch:** Xử lý dữ liệu định kỳ (hằng ngày, hằng tuần) cho các mô hình offline.
  + **Online/Near real-time:** Cập nhật nhanh feature phục vụ các mô hình cần phản hồi tức thời.
* **Metadata & Access Control:**
  + Lưu trữ thông tin về creator, mục đích sử dụng, kiểu dữ liệu.
  + Xác định chính sách phân quyền cho các feature nhạy cảm (dữ liệu cá nhân, tài chính…).

#### **2.3. Phạm vi áp dụng**

* **Dữ liệu chính:**
  + CRM (thông tin khách hàng).
  + Lịch sử giao dịch (chứng khoán, ngân hàng…).
  + Dữ liệu công ty mẹ (khách hàng tiềm năng).
  + Dữ liệu từ đối tác/bên ngoài (nếu có).
* **Các bài toán ML/DS ứng dụng:**
  + Phân tích khách hàng (churn prediction, segmentation, lead scoring).
  + Mô hình rủi ro, khuyến nghị sản phẩm, dự báo hành vi.

#### **2.4. Mục tiêu**

* **Giảm 50% thời gian chuẩn bị và làm sạch dữ liệu** cho các dự án ML so với trước.
* **Nâng cao tính nhất quán:** Một feature chỉ được định nghĩa và quản lý tại một nơi, tránh sai lệch khi dùng ở nhiều mô hình.
* **Khả năng mở rộng:** Hỗ trợ cả batch và online serving khi nhu cầu kinh doanh thay đổi.
* **Thời gian triển khai:** 3 tháng để có phiên bản MVP đưa vào sử dụng thực tế.

### **III. KẾ HOẠCH PHÁT TRIỂN**

#### **3.1. Lộ trình triển khai (3 tháng)**

##### **Giai đoạn 1: Phân tích, Thiết kế & Thiết lập cơ bản**

* Thu thập yêu cầu từ các phòng ban (DS, Marketing, CSKH…) về feature cần thiết.
* Xác định danh mục feature cốt lõi, ưu tiên xây dựng trước.
* Thiết kế kiến trúc Feature Store, chốt công nghệ (Feast, Tecton, Databricks Feature Store hoặc custom).
* Thiết lập quy ước đặt tên, format dữ liệu, metadata.
* Xây dựng pipeline ETL batch cơ bản để ingest dữ liệu CRM, giao dịch.
* Phát triển bản mẫu (PoC) lưu một số feature vào Feature Store, thử nghiệm truy vấn.

##### **Giai đoạn 2: Hoàn thiện, Tích hợp & Versioning**

* Mở rộng ingestion pipeline để thu thập thêm dữ liệu công ty mẹ, logs hành vi khách hàng.
* Xây dựng cơ chế versioning & time travel để quản lý trạng thái feature theo thời gian.
* Tích hợp MLOps pipeline để tự động trích xuất feature khi training mô hình.
* Thiết lập CI/CD để cập nhật Feature Store định kỳ.
* Kiểm tra hiệu năng: đo thời gian truy xuất feature, kiểm thử latency online serving (nếu có).
* Tích hợp Feature Store với một số mô hình ML pilot (churn prediction, lead scoring).

##### **Giai đoạn 3: Kiểm thử, Tối ưu & Go-live MVP**

* Kiểm thử chất lượng feature: so sánh kết quả với logic tính toán truyền thống.
* Xây dựng dashboard giám sát data quality (tỷ lệ null, outlier…).
* Kiểm thử bảo mật: đánh giá an ninh, xác thực & phân quyền.
* Tinh chỉnh pipeline để giảm lỗi, alert khi dữ liệu đến chậm hoặc sai định dạng.
* Đào tạo đội ngũ Data Science, ML Engineer sử dụng Feature Store.
* Đánh giá hiệu quả (ROI): tiết kiệm thời gian xử lý dữ liệu, cải thiện quy trình phát triển ML.
* Chính thức go-live MVP Feature Store, hỗ trợ các mô hình business quan trọng.

#### **3.2. Kiểm thử & Quản lý chất lượng**

* **Kiểm thử tính chính xác:**
  + So sánh giá trị feature giữa pipeline mới và pipeline truyền thống.
* **Kiểm thử hiệu năng:**
  + Đo thời gian batch ingestion, thời gian query feature, khả năng mở rộng.
* **Kiểm thử bảo mật:**
  + Đảm bảo dữ liệu nhạy cảm được mã hóa, phân quyền phù hợp.
* **Pilot deployment:**
  + Tích hợp Feature Store với 1-2 mô hình ML để đo mức độ hiệu quả.

#### **3.3. Quản lý rủi ro**

* **Chậm trễ do nhiều bên liên quan:**
  + Điều phối ưu tiên phát triển giữa các phòng ban.
* **Sai sót dữ liệu:**
  + Thiết lập validation, logging & alert khi có bất thường.
* **Khó khăn về hạ tầng:**
  + Đảm bảo đội ngũ DevOps có đủ kinh nghiệm triển khai Feature Store.
* **Chi phí vận hành:**
  + Theo dõi chi phí nếu sử dụng nền tảng cloud hoặc phần mềm trả phí.

#### **3.4. Phương hướng cải tiến**

* **Mở rộng nguồn dữ liệu:**
  + Tích hợp thêm web/app logs, dữ liệu đối tác, social data.
* **Realtime streaming:**
  + Hoàn thiện pipeline streaming nếu business yêu cầu dự đoán tức thời.
* **Auto Feature Engineering:**
  + Xây dựng module tự động gợi ý feature mới.
* **Tích hợp chặt chẽ với MLOps:**
  + Tự động retrain mô hình khi feature thay đổi, theo dõi drift, degrade.