

Project Title	MYNA For Myntra - “See It. Style It. Shop It.”
Skills take away From This Project	Computer Vision, Deep Learning, Generative AI, Image Segmentation, Object Detection, Recommendation Systems, Natural Language Processing, Conversational AI, LangChain, Retrieval-Augmented Generation (RAG), Streamlit, API, AWS S3, AWS RDS, Vector Databases, SQL, Python, PyTorch, TensorFlow, OpenCV, Stable Diffusion, YOLO, Hugging Face Transformers, API Integration, Cloud Deployment, MLOps, Nano Banana 🍌
Domain	E-Commerce, Fashion & Retail,

Problem Statement:

The rapid growth of **online fashion retail** has transformed how customers browse and purchase outfits. However, shoppers on platforms like **Myntra** still face critical challenges:

- Visualization Gap** – Customers struggle to imagine how a particular outfit or fabric (such as *Nano-Banana styled designs*) would look on them before purchase. Static product images and generic catalogs fail to deliver a personalized experience.
- Lack of Intelligent Styling Guidance** – Current recommendation engines focus on “similar items” but do not act as a **personal stylist** who can understand the customer’s preferences, detect outfit components (top, bottom, accessories), and suggest **curated looks** tailored to the user.
- Disconnected Purchase Journey** – While Myntra offers vast product catalogs, there is no **conversational AI agent** that can seamlessly take the customer from **outfit design → personalized recommendations → product discovery → purchase** in a single interactive flow.

To address these gaps, Myntra requires an **AI-powered Outfit Design and Recommendation Agent** that can:

- Allow customers to **upload their photo** and apply AI-generated outfit designs (e.g., Nano-Banana styling).
- Use **deep learning** to detect clothing parts (tops, bottoms, accessories) and recommend **3–5 best matches** per category from Myntra's product catalog with source links.
- Provide a **conversational AI experience** (chat-based history, contextual re-styling) to make the shopping journey interactive and human-like.
- Support a **purchase-ready cart system**, integrating recommendations into final checkout.
- Store generated images, preferences, and metadata securely on cloud infrastructure (e.g., AWS S3, RDS, Vector DB).

This solution will empower Myntra to deliver a **next-generation personalized shopping experience**, reduce product returns due to poor visualization, increase cart conversions, and strengthen its position as a **fashion-tech leader** in the Indian e-commerce industry.

Business Use Cases (Myntra):

1. AI-Powered Personal Stylist

- Customers upload their photo and interact with a conversational AI stylist that generates **Nano-Banana or other AI-style outfits**.
- Provides personalized outfit design suggestions based on body type, occasion, and fashion preferences.

2. Smart Product Recommendations

- Deep learning detects outfit components (tops, bottoms, footwear, accessories) and recommends **3–5 curated items** per category directly from Myntra's catalog.
- Improves product discoverability and drives cross-selling opportunities.

3. Virtual Try-On & Visualization

- Customers can preview AI-generated outfits with different colors, fabrics, and designs before purchasing.
- Reduces **returns/exchanges** caused by unmet expectations.

4. Conversational Shopping Journey

- Integrated chat-based environment maintains **conversation history** and allows iterative refinements (“Make it red with floral print”).
- Customers move from **design → recommendation → cart → purchase** within a single interactive flow.

Skill Up. Level Up

5. Cart Optimization & Upselling

- Once a user adds items to the cart, the system suggests **relevant complementary products** (e.g., matching bags, shoes, or jewelry).
- Boosts **average order value (AOV)** for Myntra.

6. Customer Insights & Analytics

- AI agent logs fashion preferences, trending designs, and frequently paired items.
- Myntra gains **data-driven insights** to refine marketing, trend forecasting, and personalized campaigns.

7. Brand Differentiation in Fashion-Tech

- Establishes Myntra as a **pioneer in AI-driven e-commerce** by blending **Generative AI, Recommendation Engines, and Conversational Agents.**
 - Enhances brand positioning against competitors like Amazon Fashion, Ajio, and Flipkart.
-

Approach:

1. User Interaction & Photo Upload

- Customers upload their **photo** into the Myntra app/web portal.
- Image preprocessing ensures the photo is cleaned, cropped, and optimized for AI processing.

Skill Up. Level Up

- Chat-based conversational interface (powered by **Conversational AI Agent**) guides the user with options:
“Would you like a casual outfit, formal wear, or party look?”
-

2. AI-Based Outfit Generation (Nano-Banana Styling)

- Use **Generative AI models** (e.g., Stable Diffusion, ControlNet, GANs) to apply **Nano-Banana fabric style** or similar AI-based outfit design to the uploaded photo.
- Image segmentation with **Deep Learning (YOLOv8 / Detectron2)** isolates clothing parts (top, bottom, footwear, accessories).
- AI regenerates outfits based on **conversation feedback**:

- Example: “*Make the dress sleeveless*”, “*Add floral prints*”, “*Change color to red*”.
 - Each iteration preserves **conversation history** and stores the generated image in **AWS S3**.
-

3. Outfit Component Detection

- A **Deep Learning agent** analyzes the AI-generated outfit and classifies components:
 - **Top** (shirt, kurti, t-shirt, blouse)
 - **Bottom** (jeans, trousers, skirts, saree drape)
 - **Footwear**
 - **Accessories** (belt, watch, jewelry, handbag)
 - Detected labels are stored in **metadata (AWS RDS)** for linking with the recommendation engine.
-

4. Recommendation Engine

- Hybrid Recommendation Approach:
 - **Content-Based Filtering** → Matches outfit features (color, category, fabric, occasion) with Myntra’s product catalog.
 - **RAG (Retrieval-Augmented Generation)** → Provides conversational explanations of why products were suggested.

- For each outfit part, **3–5 recommended products** are displayed with:

- Product image
 - Price & description
 - Direct clickable **Myntra product link**
-

5. Conversational AI Integration

- A chat-based **AI Agent** (built using LangChain + Vector DB like Pinecone/ChromaDB) manages the flow:
 - Remembers user preferences during the session.
 - Explains recommendations in natural language (“*This handbag matches your Nano-Banana styled kurti because of its color palette*”).
 - Allows iterative updates (“*Show me alternatives under ₹1500*”).
-

6. Cart Management & Purchase Flow

- Users can click “**Add to Cart**” for recommended items.
 - Cart is updated in real-time with product IDs linked to Myntra’s backend system.
 - Before checkout, AI suggests **cross-sell/upsell products** (shoes, jewelry, bags) to increase basket size.
 - Cart details and transaction logs are stored in **RDS** for analytics.
-

7. Data Storage & Infrastructure

- **AWS S3** → Stores all generated images and outfit visuals.
- **AWS RDS (PostgreSQL/MySQL)** → Stores metadata, cart details, user interactions.
- **Vector DB (Pinecone/Chroma)** → Stores embeddings for RAG-based conversational memory.
- **APIs (FastAPI / Flask)** → Middleware for image generation, detection, recommendation, and cart integration.
- **Streamlit / React Frontend** → Provides interactive UI for testing and deployment.



8. Analytics & Insights

- Skills Up. Level Up
- Track **user engagement** (time spent, conversations per session).
 - Monitor **CTR (Click-Through Rate)** on recommended products.
 - Analyze **conversion rates** from AI-generated outfits to final purchases.
 - Feed insights back to Myntra's marketing & product teams for **trend forecasting**.

9. Deployment & Scaling

- Deploy as a **Myntra in-app feature** ("MYNA by Myntra").
 - Scalable microservice architecture with **Docker** for AI services.
-

DataBase Schema:

1. Users Table

Stores user info.

Column Name	Data Type	Description
user_id	UUID / PK	Unique user identifier
name	VARCHAR(100)	User full name
email	VARCHAR(100)	Email
created_at	TIMESTAMP	Account creation date/time
last_login	TIMESTAMP	Last login timestamp

2. Outfits Table

Stores uploaded and AI-generated outfits along with detected components.

Column Name	Data Type	Description
outfit_id	UUID / PK	Unique outfit identifier

user_id	UUID / FK	References Users.user_id
photo_url	VARCHAR(500)	S3 URL of uploaded/generated photo
style_type	VARCHAR(100)	e.g., Nano-Banana, Casual, Party
top_label	VARCHAR(50)	Detected top type
bottom_label	VARCHAR(50)	Detected bottom type
accessory_labels	TEXT	Comma-separated accessories detected
created_at	TIMESTAMP	Timestamp of outfit generation

3. Product_Catalog Table

Stores Myntra product details for recommendation.

Column Name	Data Type	Description

product_id	VARCHAR(50) PK	Unique product identifier
name	VARCHAR(200)	Product name
category	VARCHAR(50)	Top, Bottom, Footwear, Accessories
price	DECIMAL(10,2)	Product price
image_url	VARCHAR(500)	Product image URL
product_url	VARCHAR(500)	Myntra product link
created_at	TIMESTAMP	Timestamp when product added

4. Cart_Recommendations Table

Combines **recommendations** and **cart tracking** in one table for simplicity.

Column Name	Data Type	Description
entry_id	UUID / PK	Unique entry ID

user_id	UUID / FK	References Users.user_id
outfit_id	UUID / FK	References Outfits.outfit_id
product_id	VARCHAR(50) FK	References Product_Catalog.product_id
component_type	VARCHAR(50)	Top, Bottom, Accessories, Footwear
added_to_cart	BOOLEAN	True if added to cart
recommended_at	TIMESTAMP	Timestamp of recommendation
purchased	BOOLEAN	True if purchased

Results / Expected Outcomes:

1. AI-Generated Personalized Outfits

- Users can upload their photos and receive **Nano-Banana styled or custom AI-generated outfits.**

- Each generated outfit preserves **conversation history** and can be iteratively refined based on user preferences.
- High-quality visualizations allow users to **preview outfits realistically** before purchase.

2. Intelligent Outfit Recommendations

- Deep learning models detect **top, bottom, footwear, and accessories** in generated outfits.
- For each detected component, the system provides **3–5 relevant product recommendations** from Myntra's catalog with images, details, and direct purchase links.
- Recommendations are personalized based on **user style, previous interactions, and trending fashion data**.

3. Enhanced Conversational Shopping Experience

- Users interact with a **chat-based AI agent** that guides styling choices, suggests alternative options, and explains recommendations.
- Iterative, contextual conversations ensure **high engagement and satisfaction**.

4. Seamless Cart & Purchase Integration

- Recommended products can be **added to the cart directly** from the AI interface.
- AI suggests **complementary items** (cross-sell & upsell) to increase basket size.
- Tracks items added and purchased, enabling Myntra to **analyze conversion rates**.

5. Data-Driven Insights for Myntra

- Captures **user fashion preferences, popular styles, and frequently paired items.**
- Provides Myntra with actionable analytics for **marketing, trend forecasting, and inventory planning.**

6. Business Impact

- Reduced **returns/exchanges** by improving outfit visualization before purchase.
- Increased **customer engagement, average order value (AOV), and retention.**
- Strengthened Myntra's positioning as a **fashion-tech leader with AI-driven personalization.**



Deliverable	Description
AI-Generated Outfits	Personalized outfits generated from user-uploaded photos using Nano-Banana or similar AI styling.
Product Recommendations	3–5 curated product suggestions per outfit component (top, bottom, accessories) with clickable Myntra links.
Conversational AI Agent	Chat-based assistant guiding users through styling choices, recommendations, and checkout.

Cart Integration & Purchase	Users can directly add recommended items to cart and complete purchase with upsell suggestions.
Stored Images	Uploaded photos and AI-generated outfit images stored securely in AWS S3.
Recommendation Metadata	Stores component labels, recommended products, timestamps, and interactions in RDS.
Analytics Reports	Summary of user engagement, conversions, and popular styles in PDF/CSV/HTML formats.
RAG-Powered Queries	Interactive insights via AI agent to query past sessions, recommendations, and user preferences.
Aggregated Insights / Summary	Consolidated report for Myntra stakeholders showing trend analysis, engagement ratios, and business impact.

Project Evaluation – Myntra AI Stylist Project:

AI Outfit Generation Accuracy



- Ensures AI-generated outfits accurately reflect the user's style preferences, selected style type, and body features.
- Focuses on **high visual realism and personalized styling** that appeals to the user.

Recommendation Relevance

- Provides highly relevant product suggestions for each outfit component (top, bottom, accessories, footwear).
- Recommendations are tailored to the user's preferences, occasion, and trending fashion styles.

Conversational AI & Styling Guidance

- The AI agent interprets user queries and provides **contextually appropriate styling advice**.
- Maintains conversation history for **multi-turn interactions**, allowing iterative outfit refinements.

Cart Integration & Purchase Flow

- Users can seamlessly **add recommended products to the cart and complete purchases**.
- Ensures a smooth and intuitive end-to-end shopping experience from AI styling to checkout.

Data Storage & Retrieval

- Uploaded photos and generated outfits are securely stored, and associated metadata and recommendations are properly managed.
- Ensures **data integrity, traceability, and easy retrieval** for further processing.



RAG-Powered Insights & Interactive Queries

- The AI agent delivers **interactive and contextual insights** from past sessions, recommendations, and user preferences.
- Supports decision-making and **enhances the overall personalization experience**.

System Performance & Reliability

- Outfit generation and recommendation processes are optimized for **real-time interactions**.
- The system reliably handles multiple users simultaneously, ensuring **a responsive user experience**.



- Enhances user engagement, personalization, and confidence in purchases.



- Reduces product returns through better outfit visualization and AI guidance.
- Provides actionable insights for fashion trends and strengthens Mynta's **position as a leader in AI-driven e-commerce**.

Technical Tags:

Generative AI, Deep Learning, Computer Vision, Image Segmentation, YOLOv8, ControlNet, GANs, Style Transfer, Recommendation System, Content-Based Filtering, RAG, Conversational AI, LangChain, Vector Database, Pinecone, ChromaDB, AWS S3, AWS RDS, PostgreSQL, MySQL, API, Flask, Streamlit, React, Microservices, Docker, MLOps, MLflow

Data Set:

1. Myntra Products Dataset

- Link: [Myntra Products Dataset](#)
- Description: This dataset contains over 1 million product listings from Myntra as of May 2023, including details such as price, discount, and MRP. ([kaggle.com](#))

2. E-commerce Products Images

- Link: [E-commerce Products Images](#)
- Description: This dataset comprises high-resolution images of fashion products, with multiple category labels and descriptions, sourced from various e-commerce platforms. ([kaggle.com](#))

Data Set Explanation:

1. Myntra Products Dataset (by Ronak Bokaria)

Purpose:

- Provides **structured product data** from Myntra, useful for training recommendation systems, analyzing trends, and mapping AI-generated outfit components to real products.

Contents:

- **Product ID:** Unique identifier for each item.
- **Product Name & Description:** Helps in **content-based recommendation**.
- **Brand:** Useful for filtering by brand or styling preferences.
- **Category & Subcategory:** E.g., Tops, Bottoms, Footwear, Accessories — critical for mapping AI-detected outfit components.
- **Price, Discount, MRP:** Supports **personalized pricing recommendations**.
- **Ratings & Reviews (if available):** Optional for enhancing recommendation relevance.

Use in Project:

- Helps AI match generated outfits to **actual Myntra products**.

Skills Required:

- Enables filtering recommendations based on **category, price range, and brand**.
- Can be combined with image datasets for **multi-modal recommendations**.

2. E-commerce Products Images Dataset

Purpose:

- Provides **high-resolution fashion product images**, essential for **visual matching**, deep learning models, and recommendation systems.

Contents:

- **Images:** High-quality product images in multiple categories.

- **Category Labels:** Each image is labeled with categories like Tops, Bottoms, Footwear, Accessories, etc.
- **Descriptions:** Textual metadata for content-based matching.

Use in Project:

- Deep learning model can use these images to **recognize outfit components** in AI-generated images.
- Enables **visual similarity search** for recommending products that closely match AI-generated outfits.
- Supports **training image-based recommendation systems** and validation of AI-generated outfit realism.



Project Deliverables:

Skill Up. Level Up

Upon completion of the project, learners are expected to submit the following:

1. Source Code

- Python scripts or notebooks for:
 - Video ingestion and frame extraction
 - Brand detection using deep learning models
 - Placement and match moment classification
 - Video chunk extraction and upload to S3
 - RDS integration for storing match, brand, and analytics data
 - Dashboard/API implementation for visualization and query

2. Video Dataset

- Selected 1–2 sample cricket match videos used for training/testing.
- Any preprocessed frames or annotated samples used for detection and analysis.

3. Extracted Video Chunks

- Brand-specific clips stored in **S3** (or local folder for testing).
- Properly named and organized for auditing purposes (brand + timestamp).

4. Analytical Reports

- Automated reports (PDF/CSV/HTML) summarizing:
 - Brand visibility metrics (duration, ratio)
 - Placement distribution
 - Event-linked exposure (six, wicket, batting, bowling, fielding)
 - S3 links for video chunks

5. Dashboard / Visualization

- Interactive dashboard connected to **RDS** showing:
 - Brand-wise exposure
 - Placement breakdown
 - Event-linked performance
 - Video chunk access

6. RAG-powered Query Interface

- A conversational AI interface that allows stakeholders to ask natural language queries about brand visibility and analytics.

7. Documentation

- Detailed project report including:
 - Problem statement
 - Dataset explanation
 - Approach
 - Results and evaluation metrics
 - Instructions for running the code and using the dashboard

8. Deployment Instructions

- Guidelines for **local or cloud deployment** of the inference API, dashboard, and database connections.

Project Guidelines – Myntra AI Stylist & Recommendation:

To ensure successful completion of the project, learners should follow these best practices and guidelines:

1. Coding Standards

- Write clean, readable, and well-documented code.
- Use meaningful variable and function names reflecting their purpose.
- Include comments explaining key logic, workflow steps, and AI model decisions.
- Follow **PEP 8 standards** for Python code.

2. Version Control

- Use **Git** for version control.
 - Create separate branches for features, bug fixes, and experiments.
 - Commit code regularly with meaningful messages.
 - Use platforms like **GitHub** / **GitLab** / **Bitbucket** for repository management.
-

3. Data Handling

- Store user-uploaded images and AI-generated outfits in **AWS S3**.
 - Preprocess images efficiently (resize, normalize, format conversion).
 - Avoid storing large datasets directly in the repository; provide download links or instructions.
 - Use **Myntra Products Dataset** and **E-commerce Images Dataset** for recommendations.
-

4. Model Training & Inference

- Use **Nano-Banana** or similar generative AI models for outfit design.
- Use **YOLOv8** / **segmentation models** for detecting top, bottom, footwear, and accessories.
- Validate models with sample images before full-scale deployment.
- Keep detection confidence thresholds configurable to balance accuracy and recall.

5. Recommendation System & Cart Integration

- Match detected outfit components to real products using **metadata and visual similarity**.
 - Display **3–5 product recommendations per component** with image, price, and clickable Myntra link.
 - Integrate with **cart and checkout** to allow direct purchase and suggest complementary items.
-

6. Data Storage & Retrieval

- Store metadata, recommendations, conversation history, and cart actions in **AWS RDS** (PostgreSQL/MySQL).
 - Ensure proper table relations between users, outfits, products, and purchases.
 - Maintain data integrity, traceability, and efficient retrieval for analytics and RAG queries.
-

7. Conversational AI & RAG Integration

- Implement a **chat-based AI agent** to guide users through styling, recommendations, and checkout.
 - Maintain conversation history for **context-aware, multi-turn interactions**.
 - Provide **interactive insights** via RAG for past sessions, recommendations, and preferences.
-

8. System Performance & Deployment

- Optimize AI model latency for **real-time outfit generation and recommendations**.
 - Ensure the system can handle **multiple concurrent users** without errors.
 - Deploy the inference API and chat interface locally or on cloud platforms (**AWS/GCP/Azure**) using **Docker** for consistency.
-

9. Testing & Validation

- Test the system with **sample images** before scaling.
 - Validate AI outfit generation, component detection, and recommendation accuracy.
 - Verify **cart and purchase workflow**, ensuring recommendations map correctly to products.
 - Ensure **RDS storage** and retrieval match actual interactions.
-

10. Documentation

- Include code, dataset details, model architectures, and system workflow.
 - Provide instructions for **running code, deploying APIs, and using the chat interface**.
 - Clearly explain assumptions, limitations, and best practices for extending the project.
-

Project Timeline:

Week	Tasks / Milestones	Deliverables
Week 1	<ul style="list-style-type: none"> - Collect and preprocess sample user photos and product images. - Set up AWS S3 for storing uploaded and AI-generated images. - Set up RDS (PostgreSQL/MySQL) for metadata, recommendations, and user sessions. - Implement initial AI outfit generation prototype using Nano-Banana or similar model. - Develop initial deep learning pipeline for detecting top, bottom, footwear, and accessories. - Begin integration of product recommendation system. 	<ul style="list-style-type: none"> - Preprocessed image dataset- S3 and RDS setup- Prototype AI outfit generation model- Initial component detection pipeline- Basic recommendation system setup
Week 2	<ul style="list-style-type: none"> - Refine AI outfit generation for style accuracy and realism. - Complete component detection and recommendation integration. - Implement conversational AI agent with RAG integration for multi-turn styling queries. - Integrate cart and checkout workflow with recommended products. - Conduct end-to-end testing with sample users. - Finalize data storage, retrieval, and analytics pipeline. - Prepare complete project documentation. 	<ul style="list-style-type: none"> - Refined AI outfit generation model- Fully integrated component detection & recommendation system- Chat-based AI agent with RAG- Cart & purchase workflow functional- Sample user test results- Complete RDS & S3 data management- Final project documentation

PROJECT DOUBT CLARIFICATION SESSION (PROJECT AND CLASS DOUBTS)

About Session: The Project Doubt Clarification Session is a helpful resource for resolving questions and concerns about projects and class topics. It provides support in understanding project requirements, addressing code issues, and clarifying class concepts. The session aims to enhance comprehension and provide guidance to overcome challenges effectively.

Note: Book the slot at least before 12:00 Pm on the same day

Timing: Monday-Saturday (4:00PM to 5:00PM)

Booking link :<https://forms.gle/XC553oSbMJ2Gcfug9>



For DE/BADM project/class topic doubt slot clarification session:

Booking link : <https://forms.gle/NtkQ4UV9cBV7Ac3C8>

Session timing:

For DE: 04:00 pm to 5:00 pm every saturday

For BADM 05:00 to 07:00 pm every saturday

LIVE EVALUATION SESSION (CAPSTONE AND FINAL PROJECT)

About Session: The Live Evaluation Session for Capstone and Final Projects allows participants to showcase their projects and receive real-time feedback for improvement. It assesses project quality and provides an opportunity for discussion and evaluation.

Note: This form will Open only on Saturday (after 2 PM) and Sunday on Every Week

Timing:

For BADM and DE

Monday-Saturday (11:30AM to 1:00PM)

For DS and AIML

Monday-Saturday (05:30PM to 07:00PM)

Booking link : <https://forms.gle/1m2Gsro41fLtZurRA>