**MLOps\_UseCase\_API\_Design**

**API Endpoint Design**

| **Use Case** | **Service Name** | **Description** | **HTTP Command** | **Signature** | **Responses** | **Model Example (JSON)** | **Example HTTP Response (JSON)** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Clinical Diagnostics Enhancement IDC Detection | *DiagnosePatient* | *Diagnose a patient based on their medical data* | *POST* | *https://127.0.0.0:5000/diagnose-patient* | *200 = success, 400 = error, 500 = error* | ***{"patientId": "patient123","patientData": {"age": 45,"weight": 70,"height": 175,"symptoms": ["Swollen/painful breast, fever", "cough", "fatigue,"]}}*** | ***{"patientId": "patient123","diagnosis": [{"condition": "Flu", "likelihood": 0.75},{"condition": "breast cancer", "likelihood": 0.60}],"message": "Diagnosis completed successfully."}*** |
| Clinical Diagnostics Enhancement | *TrainDiagnosticModel* | *Train a new version of the diagnostic model with provided training data* | *POST* | *https://127.0.0.0:5000/train-model* | *200 = success, 400 = error, 500 = error* | ***{"trainingDataURL": "https://storage/training-data/diagnostic-data.csv","modelParameters": {"learningRate": 0.01,"epochs": 10},"description": "Training data for clinical diagnostic model."}*** | ***{"status": "Training completed","modelId": "diagnostic\_model\_v1.2.0","trainingMetrics": {"accuracy": 0.94,"loss": 0.06},"message": "The model was successfully trained."}*** |
| Clinical Diagnostics Enhancement | *EvaluateDiagnosticModel* | *Evaluate the performance of the diagnostic model with provided evaluation data* | *POST* | *https://127.0.0.0:5000/evaluate-model* | *200 = success, 400 = error, 500 = error* | ***{"evaluationDataURL": "https://storage/evaluation-data/diagnostic-evaluation-data.csv","modelId": "diagnostic\_model\_v1.2.0","evaluationMetrics": ["accuracy", "precision", "recall"]}*** | ***{"status": "Evaluation completed","modelId": "diagnostic\_model\_v1.2.0","evaluationResults": {"accuracy": 0.93,"precision": 0.92,"recall": 0.91},"message": "The model was successfully evaluated."}*** |
| Research and Development in Oncology | *PredictCancerRisk* | *Predict the risk of cancer based on patient genetic and clinical data* | *POST* | *https://127.0.0.0:5000/predict-cancer-risk* | *200 = success, 400 = error, 500 = error* | ***{"patientId": "patient456","patientData": {"age": 55,"geneticMarkers": ["BRCA1", "BRCA2"],"lifestyleFactors": ["smoking", "diet"]}}*** | ***{"patientId": "patient456","cancerRisk": {"breastCancer": 0.85,"lungCancer": 0.60},"message": "Cancer risk prediction completed successfully."}*** |
| Research and Development in Oncology | *TrainOncologyModel* | *Train a new version of the oncology model with provided training data* | *POST* | *https://127.0.0.0:5000/train-model* | *200 = success, 400 = error, 500 = error* | ***{"trainingDataURL": "https://storage/training-data/oncology-data.csv","modelParameters": {"learningRate": 0.01,"epochs": 10},"description": "Training data for oncology model."}*** | ***{"status": "Training completed","modelId": "oncology\_model\_v1.2.0","trainingMetrics": {"accuracy": 0.90,"loss": 0.10},"message": "The model was successfully trained."}*** |
| Research and Development in Oncology | *EvaluateOncologyModel* | *Evaluate the performance of the oncology model with provided evaluation data* | *POST* | *https://127.0.0.0:5000/evaluate-model* | *200 = success, 400 = error, 500 = error* | ***{"evaluationDataURL": "https://storage/evaluation-data/oncology-evaluation-data.csv","modelId": "oncology\_model\_v1.2.0","evaluationMetrics": ["accuracy", "precision", "recall"]}*** | ***{"status": "Evaluation completed","modelId": "oncology\_model\_v1.2.0","evaluationResults": {"accuracy": 0.89,"precision": 0.88,"recall": 0.87},"message": "The model was successfully evaluated."}*** |
| Educational and Training Tool for Pathologists | *AnalyzePathologyImage* | *Analyze a pathology image and provide diagnostic insights* | *POST* | *https://127.0.0.0:5000/analyze-image* | *200 = success, 400 = error, 500 = error* | ***{"imageId": "img789","imageURL": "https://storage/pathology-images/image789.jpg"}*** | ***{"imageId": "img789","diagnosticInsights": {"cancerCellsDetected": true,"cellCount": 250,"tumorType": "Adenocarcinoma"},"message": "Image analysis completed successfully."}*** |
| Educational and Training Tool for Pathologists | *TrainPathologyModel* | *Train a new version of the pathology image analysis model with provided training data* | *POST* | *https://127.0.0.0:5000/train-model* | *200 = success, 400 = error, 500 = error* | ***{"trainingDataURL": "https://storage/training-data/pathology-images.zip","modelParameters": {"learningRate": 0.01,"epochs": 10},"description": "Training data for pathology image analysis model."}*** | ***{"status": "Training completed","modelId": "pathology\_model\_v1.2.0","trainingMetrics": {"accuracy": 0.92,"loss": 0.08},"message": "The model was successfully trained."}*** |
| Educational and Training Tool for Pathologists | *EvaluatePathologyModel* | *Evaluate the performance of the pathology image analysis model with provided evaluation data* | *POST* | *https://127.0.0.0:5000/evaluate-model* | *200 = success, 400 = error, 500 = error* | ***{"evaluationDataURL": "https://storage/evaluation-data/pathology-evaluation-images.zip","modelId": "pathology\_model\_v1.2.0","evaluationMetrics": ["accuracy", "precision", "recall"]}*** | ***{"status": "Evaluation completed","modelId": "pathology\_model\_v1.2.0","evaluationResults": {"accuracy": 0.91,"precision": 0.90,"recall": 0.89},"message": "The model was successfully evaluated."}*** |

|  |  |
| --- | --- |
| **Field** | **Description** |
| Endpoint Name | ***Model Training Service*** |
| Endpoint URL | ***/trainModel*** |
| HTTP Method | ***POST*** |
| Description | *Endpoint for training machine learning models with provided data and parameters.* |
| Request Parameters | ***data*** *(string, required),* ***params*** *(dictionary, required)* |
| Response | *JSON object containing* ***model\_artifact*** *(string) and* ***training\_metrics*** *(dictionary)* |
| Example Request | ***curl -X POST http://127.0.0.1:5000/trainModel -H "Content-Type: application/json" -d '{"data": "sample\_data", "params": {"learning\_rate": 0.01}}'*** |
| Example Response | ***{"model\_artifact": "dummy\_model\_path", "training\_metrics": {"accuracy": 0.95}}*** |