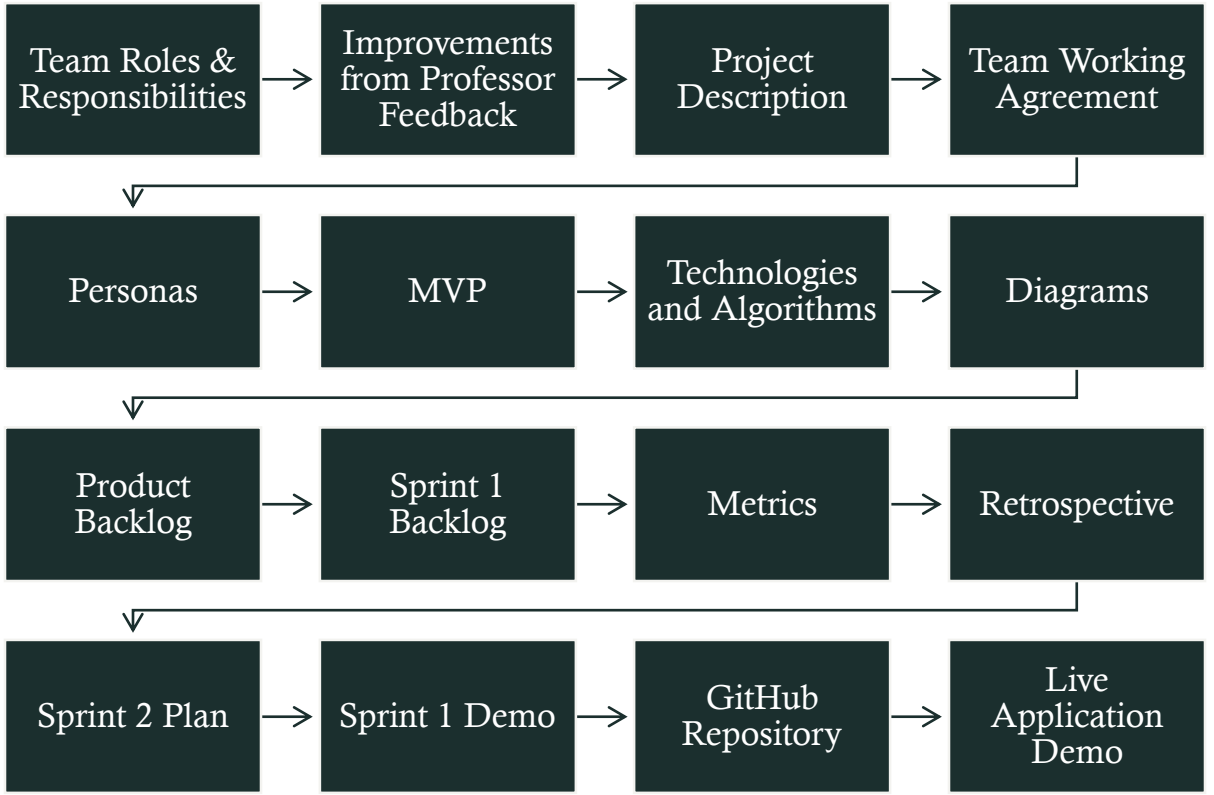
IMAGEMEDIX

By Team 5 The Minions

Sprint 1

AGENDA


Team Roles & Responsibilities
Improvements from Professor Feedback
Project Description
Team Working Agreement
Personas
MVP
Technologies and Algorithms
Diagrams
Product Backlog
Sprint 1 Backlog
Metrics
Retrospective
Sprint 2 Plan
Sprint 1 Demo
GitHub Repository
Live Application Demo

TEAM MEMBERS – MACHINE LEARNING

Vaishnavi Chandrasekar

Christiana Heden Kothuru

Deepika Mothkuri

Greeshma Kenche

TEAM MEMBERS – BACKEND

Yucong Hu

Lingyi Luan

TEAM MEMBERS – FRONTEND

Yash Bhanubhai Panchani

Rameez Ahmed Shaik

IMPROVEMENTS

- Improve team coordination

PROJECT DESCRIPTION

Project Name:	**ImageMedix**
Team:	The Minions
Project Description:	For **healthcare professionals** who **need to diagnose pneumonia and brain tumors efficiently**, the **Dual-Stage Medical Image Classification System** is a **two-step image analysis tool** that **automatically classifies medical images as lung X-rays or brain MRIs and provides a diagnosis**. Unlike **separate diagnostic systems that handle only one type of image**, our application **streamlines diagnosis by analyzing both image types within a single system, saving time and improving diagnostic accuracy**.
Benefit Outcomes:	• **Faster Diagnosis:** Automates image classification and diagnosis, reducing the time needed for manual analysis. • **Improved Accuracy:** Provides consistent and reliable preliminary diagnoses, minimizing human errors. • **Resource Optimization:** Assists medical facilities with limited access to specialists, enabling quicker and more informed decisions.
Github Link:	https://github.com/htmw/2025S-The-Minions/wiki

Team Working Agreement - The Minions

1. **Responsibilities:**
   - Each member is responsible for completing their tasks on time and maintaining the quality of their work.
   - Members must inform the group if they encounter any blockers that could delay the project.
2. **Communication:**
   - We will use common platforms like WhatsApp, Discord, or email for quick communication and updates.
   - Weekly meetings will be scheduled to review progress and discuss tasks.
3. **Meeting Attendance:**
   - All members are expected to attend scheduled meetings unless there are unavoidable circumstances.
   - Absentees should catch up on meeting notes to stay informed.
4. **Collaboration:**
   - All members are expected to contribute to their assigned roles (frontend, backend, machine learning).
   - Teamwork will be encouraged by helping each other when someone faces challenges or needs feedback.
5. **Deliverables:**
   - Each sprint will include presentations and documentation updates as required.
   - Work will be reviewed by peers to ensure quality before submission.
6. **Code and Documentation:**
   - Code should follow basic best practices for readability and functionality.
   - Documentation (tech papers, wikis) will be updated regularly by the respective team members.
7. **Decision-Making:**
   - Major decisions will be made collectively during meetings.
   - In case of disagreements, majority voting will determine the final decision.
8. **Conflict Resolution:**
   - Conflicts will be discussed openly during meetings, with all members encouraged to share their views.
   - If unresolved, the issue will be escalated to the **professor** for guidance.

Team Members
Chandrasekar Vaishnavi
Hu Yucong
Kenche Greeshma
Kothuru Christiana Heden
Luan Lingyi
Mothkuri Deepika
Panchani Yash Bhanubhai
Shaik Rameez Ahmed

TEAM WORKING AGREEMENT

PERSONAS

Dr. James Patel (Radiologist)

- **Age:** 45
- **Occupation:** Senior Radiologist at a metropolitan hospital
- **Background:** Over 20 years of experience in diagnostic imaging, specializing in lung and brain disorders. Known for mentoring junior doctors in radiology.
- **Goals:** To reduce the time spent analyzing large volumes of medical images while maintaining high diagnostic accuracy.
- **Challenges:** Overloaded with image analysis requests and administrative duties, leading to delays in diagnosis and increased stress.
- **How the System Helps:** Automates initial image classification and diagnosis, allowing Dr. Patel to prioritize critical and complex cases efficiently.

PERSONAS

Dr. Suzen Chen (General Practitioner)

- **Age:** 38
- **Occupation:** General Practitioner at a rural healthcare clinic
- **Background:** Has served in underserved regions for over a decade, often working without the immediate support of medical specialists. Committed to improving healthcare access in remote areas.
- **Goals:** To provide fast, accurate diagnoses for patients despite limited access to specialists.
- **Challenges:** She m ust handle a wide variety of cases on her own, making it difficult to diagnose complex conditions such as brain tumors and pneumonia accurately.
- **How the System Helps:** The AI system offers reliable preliminary diagnoses for brain and lung conditions, helping Dr. Chen make timely and informed treatment decisions.

PERSONAS

Dr. Raj Aryan (Healthcare Director)

- **Age:** 50
- **Occupation:** Director of a large hospital chain in India
- **Background:** A visionary healthcare leader who has overseen the expansion of multiple hospitals across the country. Inspired by challenges faced in rural and semi-urban healthcare delivery.
- **Goals:** To improve diagnostic efficiency, reduce patient wait times, and implement AI-based healthcare solutions across all branches.
- **Challenges:** Delays in diagnosis caused by a shortage of skilled radiologists, leading to overcrowded hospitals and slow patient care.
- **How the System Helps:** The automated system accelerates diagnosis processes across multiple hospitals, enabling faster and more efficient patient care, particularly in resource-constrained facilities.

MVP

Core Features	
Two-Step Image Analysis	Classifies medical images as either lung X-rays or brain MRIs.
	Provides automated diagnosis for pneumonia and brain tumors.
Automated Diagnosis System	Uses machine learning models to detect abnormalities.
	Outputs a diagnostic result with confidence scores.
Unified Platform	Supports both lung X-rays and brain MRIs in a single system.
	Eliminates the need for separate diagnostic tools.
User-Friendly Interface	Simple image upload functionality for healthcare professionals.
	Displays classification results and diagnosis in an intuitive format.
Performance Metrics	Ensures **accuracy** through AI-driven predictions.
	Optimized for **faster diagnosis** to reduce manual analysis time.
Basic Report Generation	Generates a preliminary diagnostic report.
	Includes confidence levels and possible next steps for further medical review.

TECHNOLOGIES

Frontend: React.js

- Used to create a user-friendly interface that allows healthcare professionals to upload and view medical images easily.

Backend: Node.js (API), Flask (Model Inference), MongoDB

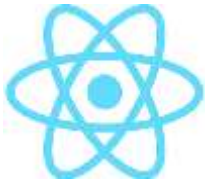
- **Node.js:** Handles API requests and manages communication between the frontend and backend services.
- **Flask:** Supports model inference by running machine learning models for image classification and diagnosis.
- **MongoDB:** Stores user data, images, and diagnostic results, enabling secure and scalable data management.

Machine Learning:

- Built using **PyTorch** and trained on datasets for lung X-rays and brain MRIs.

Cloud Infrastructure:

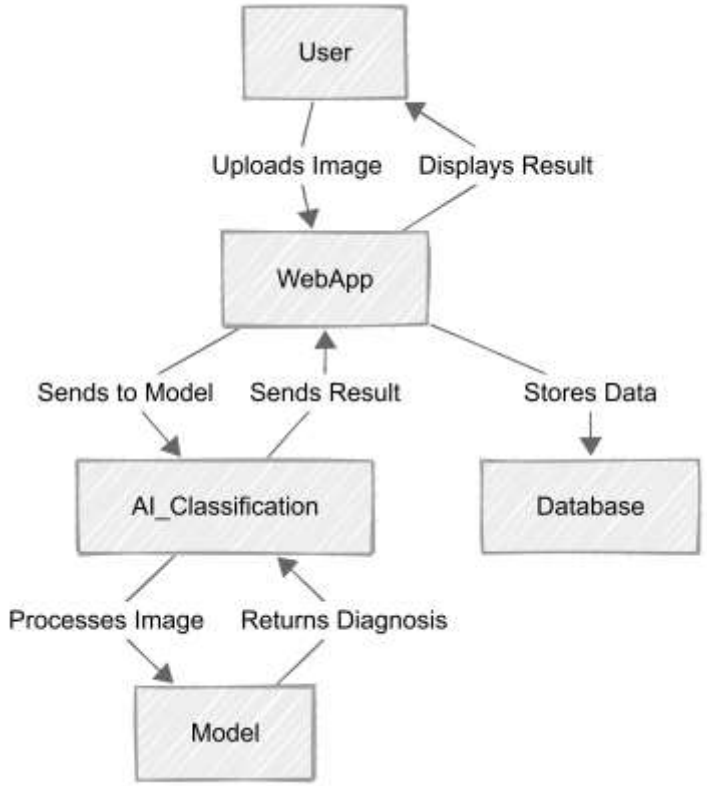
- Deployed using **AWS.**

ALGORITHMS

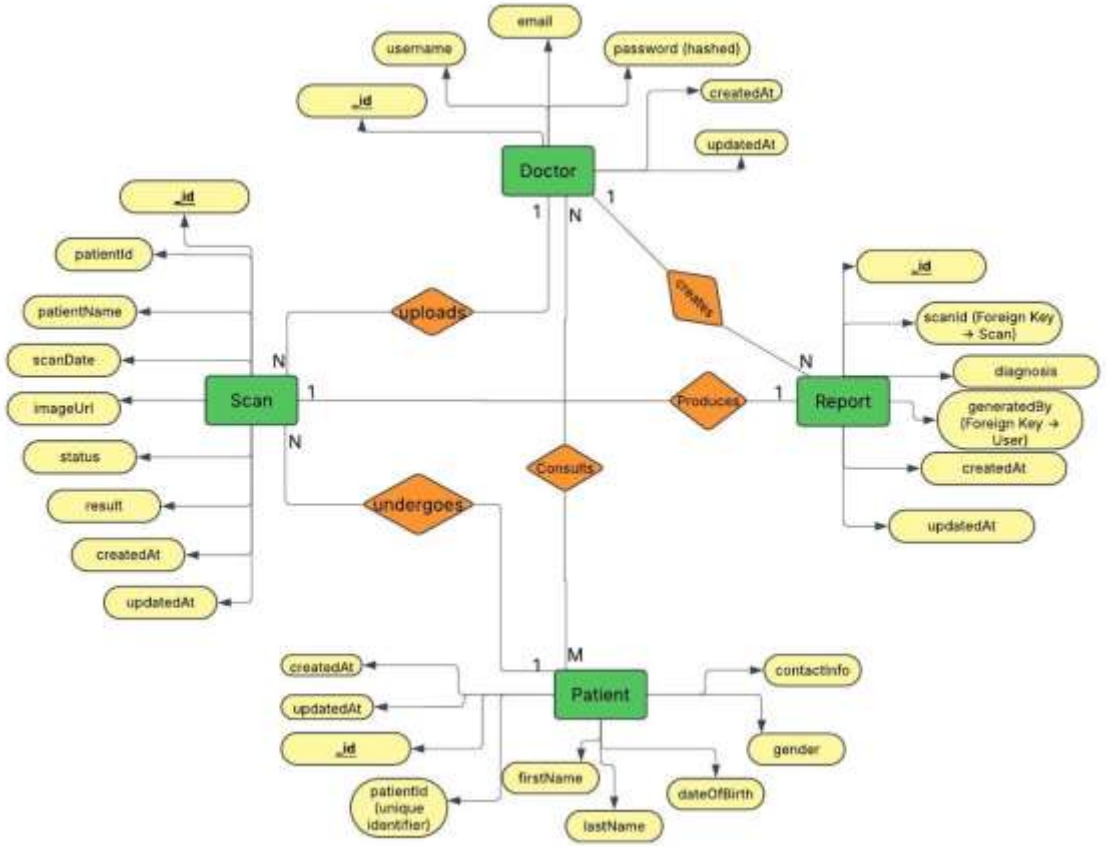
The system uses two core algorithms based on pre-trained deep learning models. A **ResNet** model, fine-tuned on a dataset of medical images, is used to classify whether the uploaded image is a lung X-ray or a brain MRI. Depending on the classification result, the system proceeds to a second stage. For lung X-rays, a **fine-tuned EfficientNet** model detects and classifies pneumonia into normal, viral, or bacterial categories by analyzing lung patterns. For brain MRIs, the same EfficientNet model is used to detect and classify brain tumors into glioma, meningioma, or no tumor by identifying abnormalities in brain structures. These models provide high accuracy and efficiency, ensuring reliable diagnoses for both image types.

ARCHITECTURE DIAGRAM

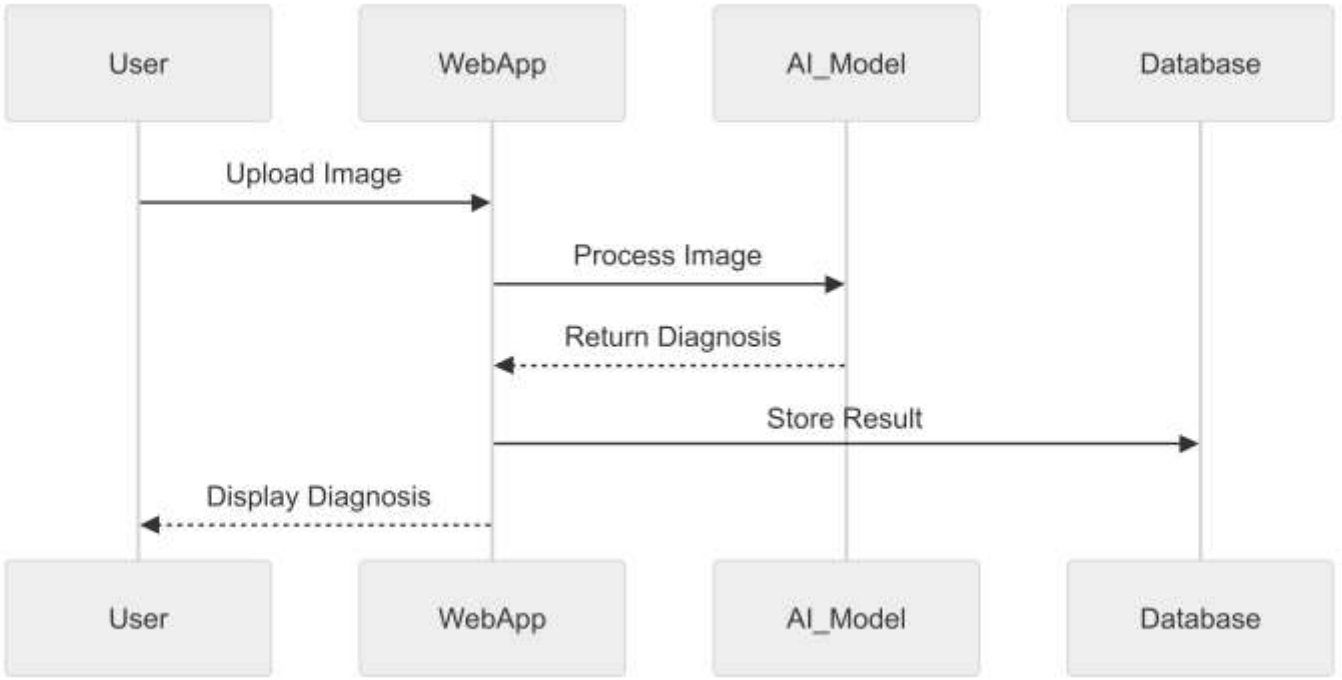


CONTEXT DIAGRAM

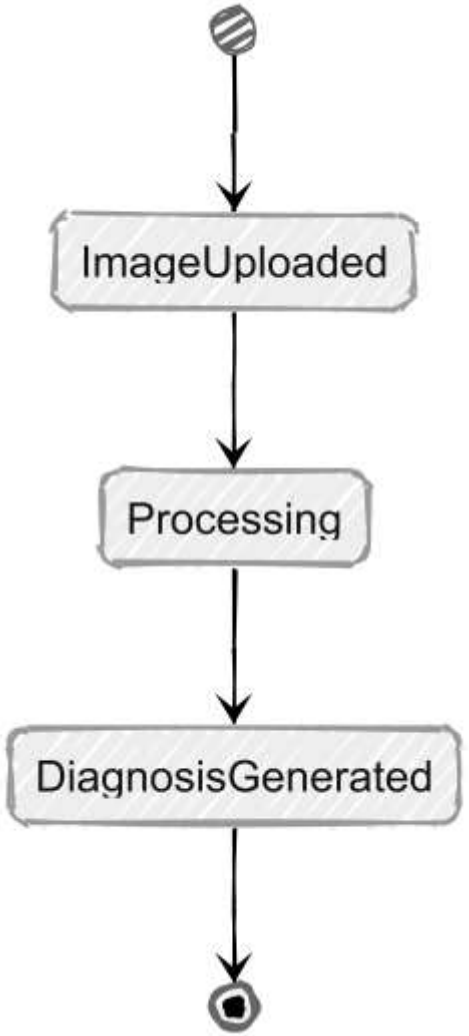
ER DIAGRAM


email
username
password (hashed)
_id
createdAt
updatedAt
Doctor
1
N
1
_id
patientId
patientName
scanDate
imageUrl
status
result
createdAt
updatedAt
uploads
N
Scan
1
N
Creates
N
Produces
1
Report
_id
scanId (Foreign Key → Scan)
diagnosis
generatedBy (Foreign Key → User)
createdAt
updatedAt
Consults
undergoes
1
M
Patient
createdAt
updatedAt
_id
patientId (unique identifier)
firstName
lastName
dateOfBirth
gender
contactInfo

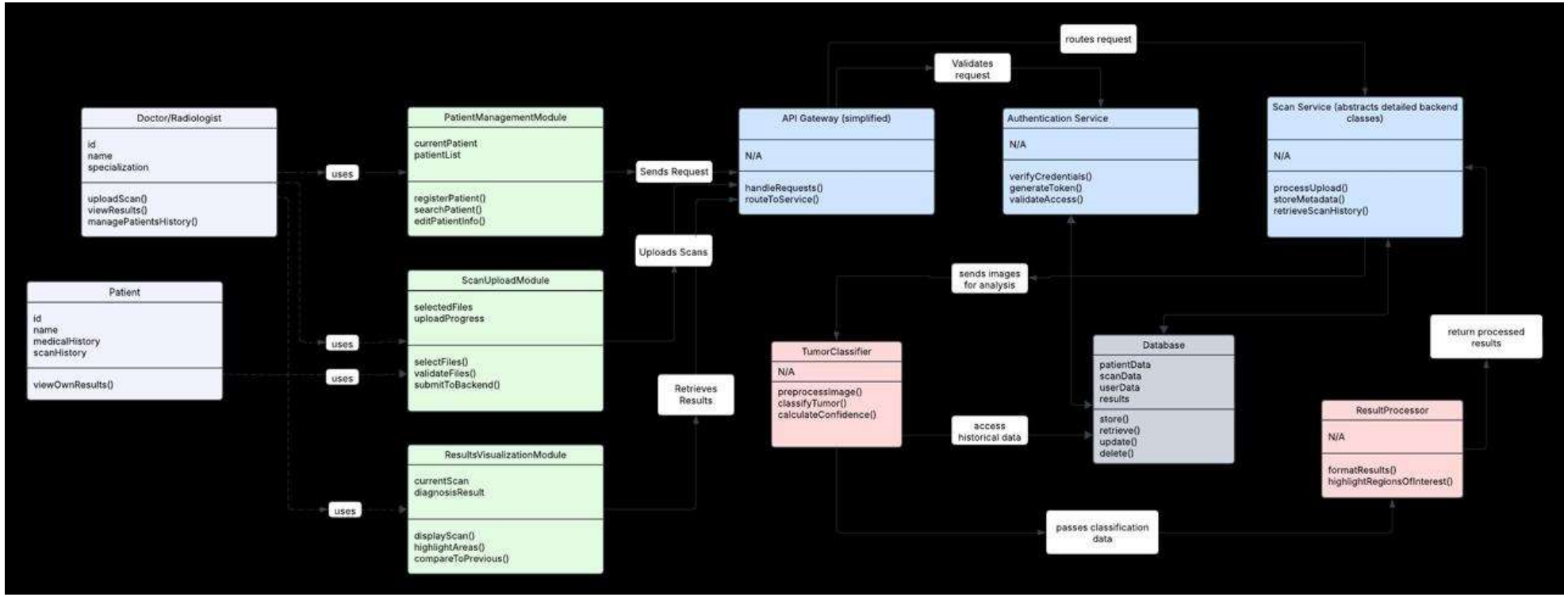
SEQUENCE DIAGRAM



STATE DIAGRAM



CLASS DIAGRAM



PRODUCT BACKLOG

ID	Sprint	User Story / Technical Story (TS)	Acceptance Criteria	Story Points (SP)
US-01	Sprint 1	As a user, I want a visually appealing landing page so I can understand the system at a glance.	Must include a header, project description, CTA button, and demo preview.	3
US-02	Sprint 1	As a user, I want an intuitive navigation menu so I can easily access different sections of the website.	Implement navbar with links to Home, Upload, Results, and About.	2
US-03	Sprint 1	As a user, I want a simple image upload interface so I can easily upload my medical images.	Create a file upload UI (Frontend only, no backend integration).	3
US-04	Sprint 1	As a user, I want a responsive design so I can access the site on mobile devices.	Ensure UI adapts well to different screen sizes.	2

ID	Sprint	User Story / Technical Story (TS)	Acceptance Criteria	Story Points (SP)
US-05	Sprint 1	As a user, I want to see a static results page so I understand how the diagnosis will be displayed.	Create a placeholder UI for displaying diagnostic results.	2
TS-01	Sprint 1	Set up frontend framework and project structure. As a developer, I want to set up the frontend framework and project structure so that the development process is organized and scalable.	Initialize React project and configure routing.	2
TS-02	Sprint 1	Implement UI components for buttons, inputs, and cards.	Ensure reusable and styled components are available.	3
US-06	Sprint 2	As a developer, I want to implement UI components for buttons, inputs, and cards so that the system has reusable and consistent elements.	Enable image uploads and store images in the database.	5

ID	Sprint	User Story / Technical Story (TS)	Acceptance Criteria	Story Points (SP)
US-07	Sprint 2	As a user, I want my medical image to be analyzed automatically so I can receive a diagnosis.	Integrate AI model for classification of lung X-rays and brain MRIs.	8
US-08	Sprint 2	As a user, I want to see my diagnosis displayed clearly so I can understand the results.	Display AI-generated results on the frontend.	3
US-09	Sprint 2	As a user, I want my diagnosis results to be stored so I can access them later.	Store diagnosis and image data in a database.	5
US-10	Sprint 2	As a user, I want basic account authentication so I can securely log in and access my past diagnoses.	Implement login/sign-up functionality.	5

ID	Sprint	User Story / Technical Story (TS)	Acceptance Criteria	Story Points (SP)
TS-03	Sprint 2	As a developer, I want to set up a backend API for image processing so that the system can analyze uploaded images efficiently.	Develop Flask/Django API to handle image uploads and analysis.	8
TS-04	Sprint 2	As a developer, I want to connect the frontend with the backend API so that data flows seamlessly between the client and server.	Ensure API endpoints are properly integrated with React.	5
US-11	Sprint 3	As a user, I want my diagnosis report to be downloadable so I can share it with doctors.	Generate PDF reports for diagnosis results.	5
US-12	Sprint 3	As a user, I want a faster processing time so I don't have to wait long for results.	Optimize AI model inference time to under 2 seconds.	8

ID	Sprint	User Story / Technical Story (TS)	Acceptance Criteria	Story Points (SP)
US-13	Sprint 3	As a user, I want role-based access (doctor, patient, admin) so I can have personalized features.	Implement RBAC with different user permissions.	5
US-14	Sprint 3	As a user, I want the system to be highly accurate so I can trust the diagnosis.	Improve AI model accuracy to at least 90%.	8
US-15	Sprint 3	As a user, I want to access the system online so I can use it from anywhere.	Deploy the web application and make it publicly accessible.	5
TS-05	Sprint 3	As a developer, I want to implement logging and error handling so that system failures can be monitored and resolved efficiently.	Ensure system logs errors and alerts admin for failures.	3

SPRINT 1 BACKLOG

ID	User Story / Task (TS)	Acceptance Criteria (AC)	Story Points (SP)
US-01	As a user, I want a visually appealing landing page so I can understand the system at a glance.	Must include a header, project description, CTA button, and demo preview.	3
US-02	As a user, I want an intuitive navigation menu so I can easily access different sections of the website.	Implement navbar with links to Home, Upload, Results, and About.	2
US-03	As a user, I want a simple image upload interface so I can easily upload my medical images.	Create a file upload UI (Frontend only, no backend integration).	3
US-04	As a user, I want a responsive design so I can access the site on mobile devices.	Ensure UI adapts well to different screen sizes.	2
US-05	As a user, I want to see a static results page so I understand how the diagnosis will be displayed.	Create a placeholder UI for displaying diagnostic results.	2
TS-01	Set up frontend framework and project structure. As a developer, I want to set up the frontend framework and project structure so that the development process is organized and scalable.	Initialize React project and configure routing.	2
TS-02	Implement UI components for buttons, inputs, and cards.	Ensure reusable and styled components are available.	3

SPRINT 1 TEST CASES

ID	Test Case	Expected Result	Actual Result	Pass/Fail
US-01-TC01	Load the landing page.	The page should display project description, CTA button, and a demo preview.	The landing page loads with all required elements correctly displayed.	Pass
US-02-TC01	Click on each navbar link (Home, Upload, Results, About).	The user should be navigated to the correct section/page.	Navbar links navigate to the correct pages smoothly.	Pass
US-03-TC01	Try uploading an image (PNG, JPG).	The file should be accepted and UI should show a preview.	Image upload works, but preview does not appear.	Fail
US-03-TC02	Try uploading a non-image file (PDF, TXT).	The UI should display an error message.	The UI correctly blocks non-image files and shows an error message.	Pass
US-04-TC01	Resize the browser window to mobile size.	The UI layout should adjust properly.	The layout adapts correctly to different screen sizes.	Pass
US-05-TC01	Open the static results page.	The placeholder UI should be visible.	Placeholder UI is visible, but the styling needs adjustments.	Fail
TS-01-TC01	Check if the React project runs successfully.	The application should compile and load without errors.	The React project runs successfully with no build errors.	Pass
TS-02-TC01	Verify button and input components.	UI components should render correctly with proper styles.	Buttons and inputs display correctly and are functional.	Pass

SPRINT 1 STORIES COMPLETED

ID	User Story / Task (TS)	Acceptance Criteria (AC)	Story Points (SP)
US-01	As a user, I want a visually appealing landing page so I can understand the system at a glance.	Must include a header, project description, CTA button, and demo preview.	3
US-02	As a user, I want an intuitive navigation menu so I can easily access different sections of the website.	Implement navbar with links to Home, Upload, Results, and About.	2
US-04	As a user, I want a responsive design so I can access the site on mobile devices.	Ensure UI adapts well to different screen sizes.	2
TS-01	Set up frontend framework and project structure. As a developer, I want to set up the frontend framework and project structure so that the development process is organized and scalable.	Initialize React project and configure routing.	2
TS-02	Implement UI components for buttons, inputs, and cards.	Ensure reusable and styled components are available.	3

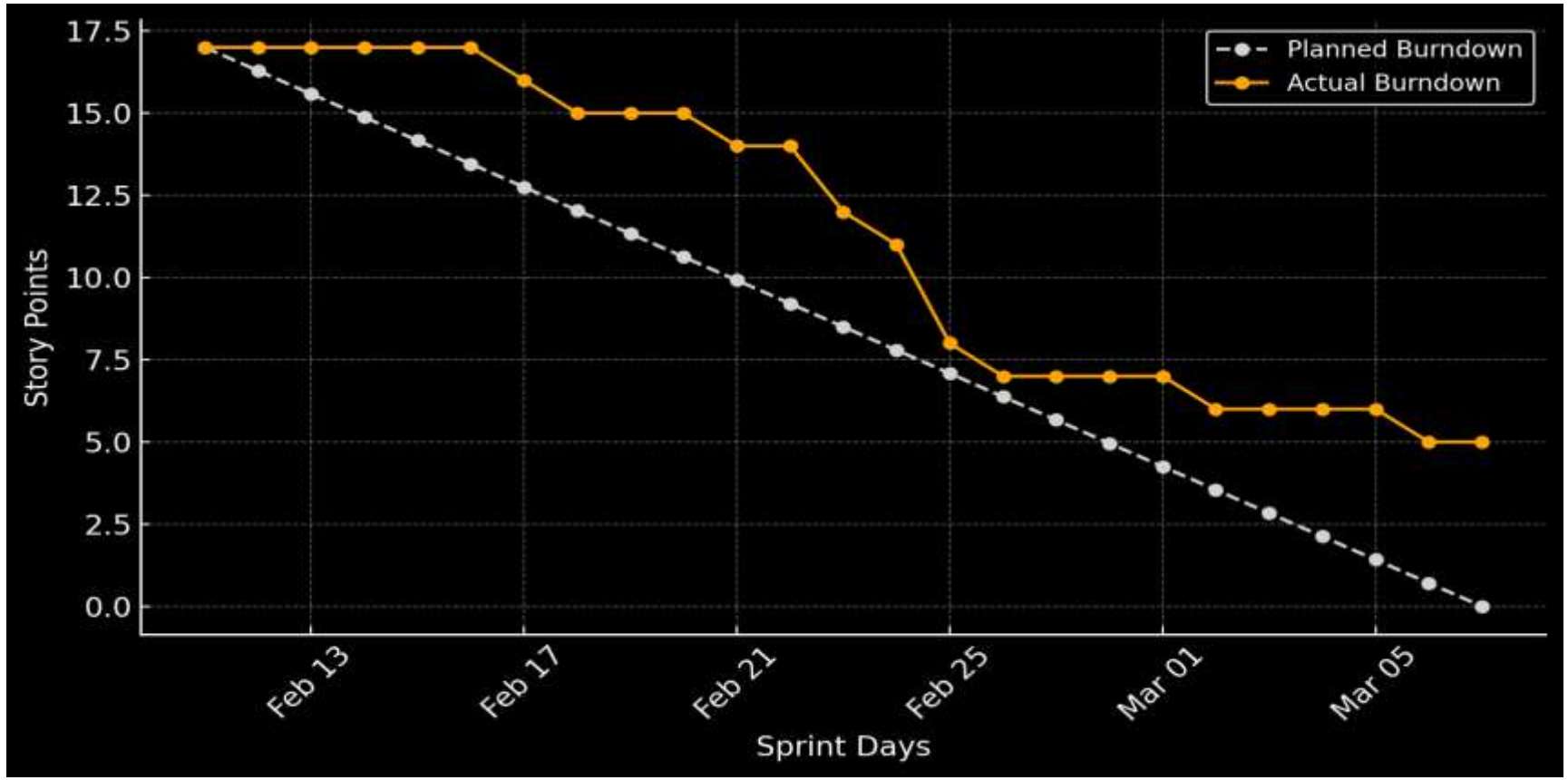
SPRINT 1 STORIES NOT COMPLETED

ID	User Story / Task (TS)	Acceptance Criteria (AC)	Story Points (SP)	Reason for Incompletion
US-03	As a user, I want a simple image upload interface so I can easily upload my medical images.	Create a file upload UI (Frontend only, no backend integration).	3	Image upload preview not displaying properly.
US-05	As a user, I want to see a static results page so I understand how the diagnosis will be displayed.	Create a placeholder UI for displaying diagnostic results.	2	UI styling needs adjustments to match design.

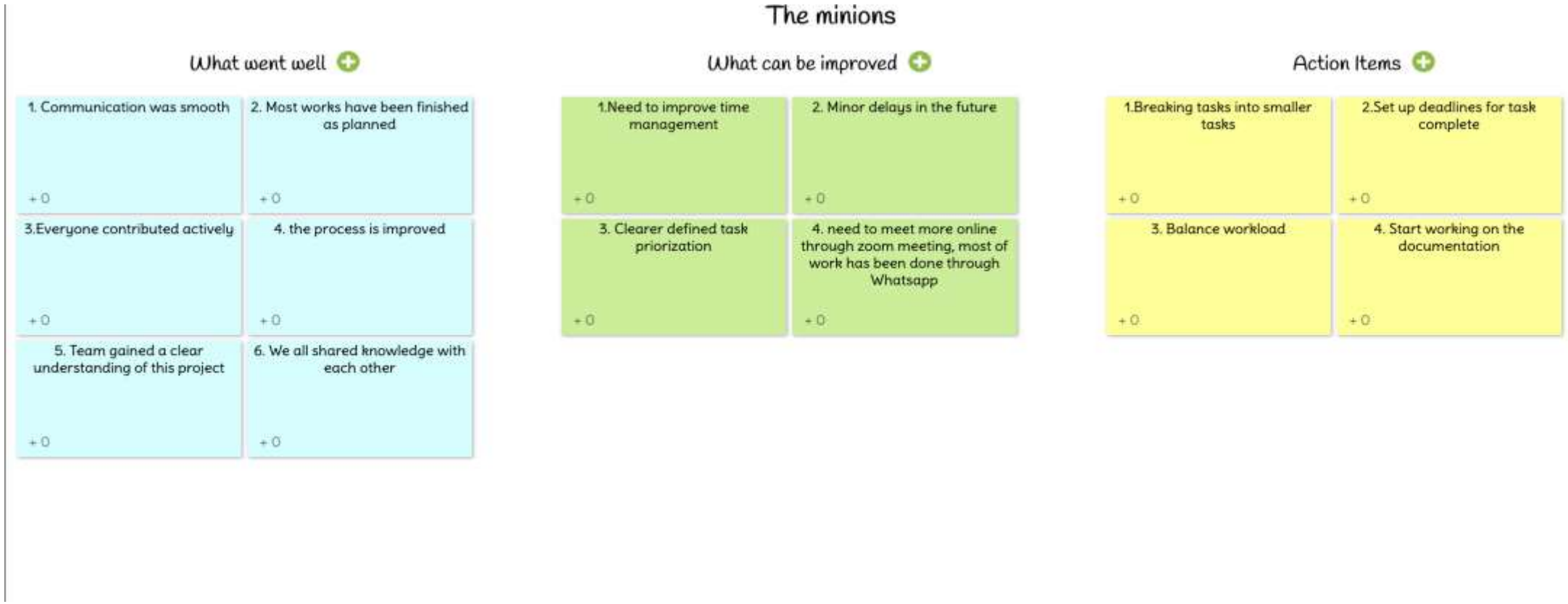
METRICS

- **Team Velocity**
  - **Total Story Points Completed: 12 SP**
  - **Team Velocity: 12 SP**
- **Completed/Committed Ratio**
  - **Total Story Points Committed: 17 SP**
  - **Total Story Points Completed: 12 SP**
  - **Completed/Committed Ratio: (12/17) = 70.59%**

BURNDOWN CHART



RETROSPECTIVE



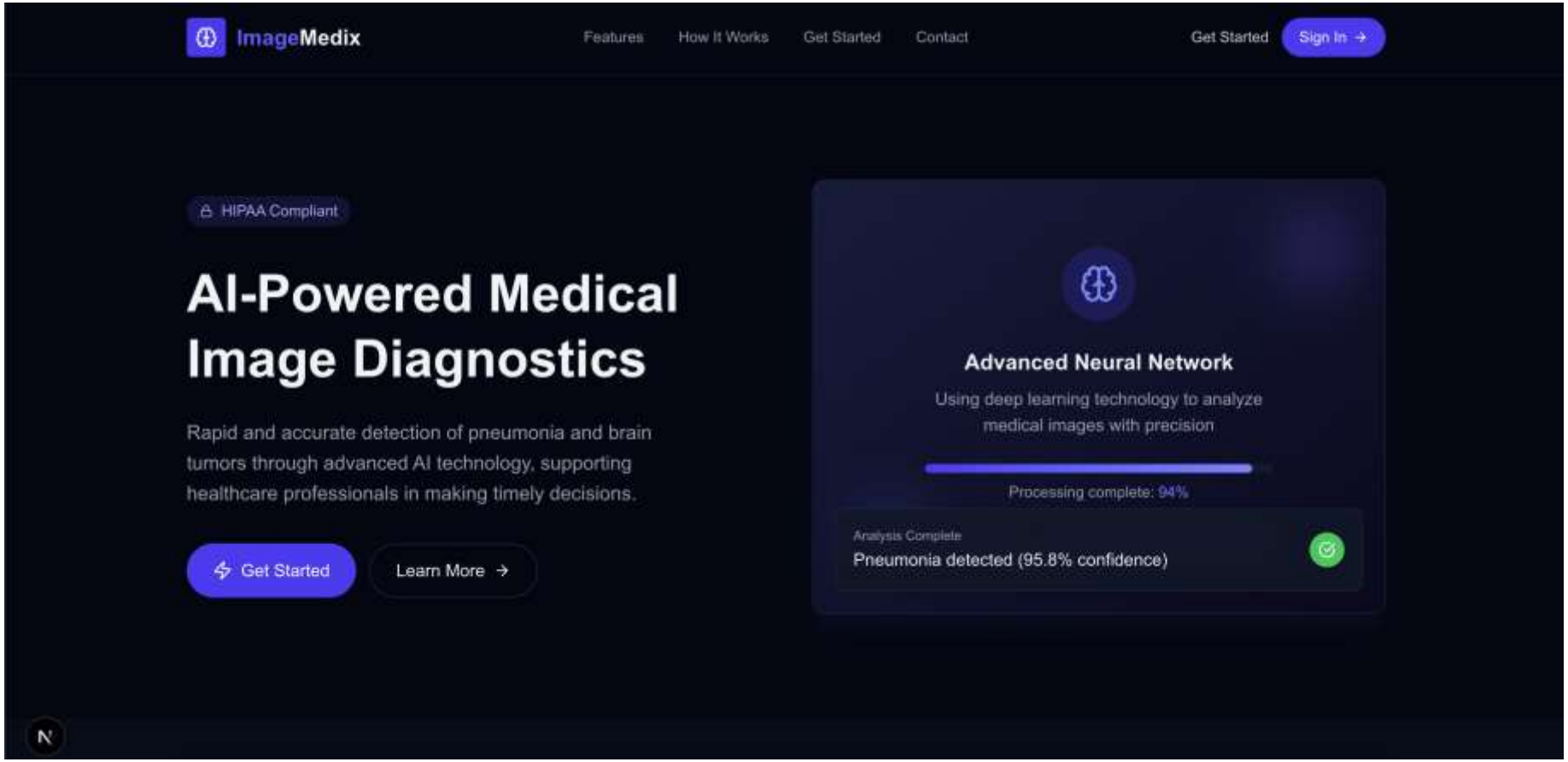
SPRINT 2 PLANNING

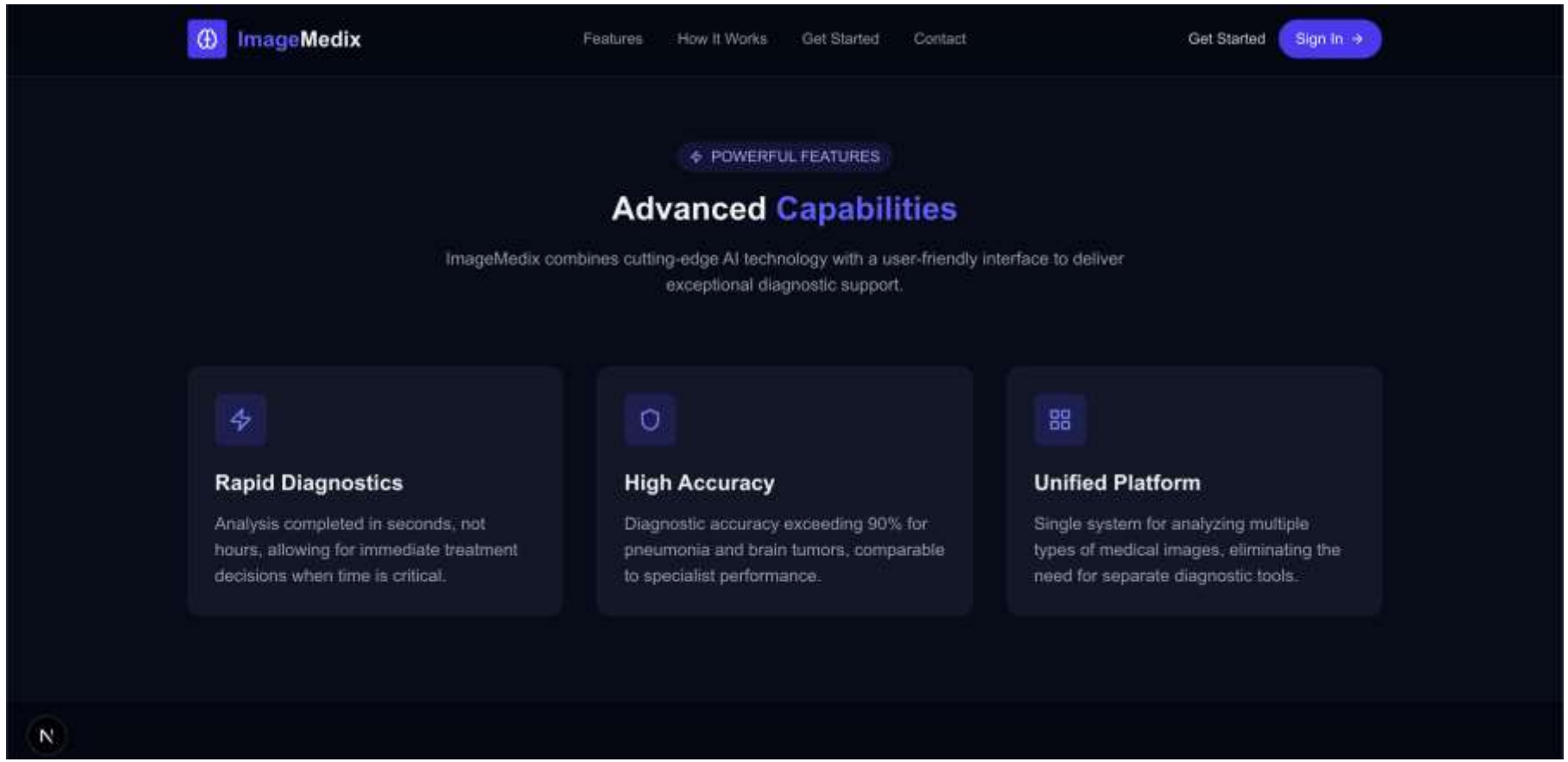
ID	User Story / Task (TS)	Acceptance Criteria (AC)	Story Points (SP)
US-06	As a user, I want to upload an image and get it processed so I can receive a diagnosis.	Enable image uploads and store images in the database.	5
US-07	As a user, I want my medical image to be analyzed automatically so I can receive a diagnosis.	Integrate AI model for classification of lung X-rays and brain MRIs.	8
US-08	As a user, I want to see my diagnosis displayed clearly so I can understand the results.	Display AI-generated results on the frontend.	3
US-09	As a user, I want my diagnosis results to be stored so I can access them later.	Store diagnosis and image data in a database.	5
US-10	As a user, I want basic account authentication so I can securely log in and access my past diagnoses.	Implement login/sign-up functionality.	5

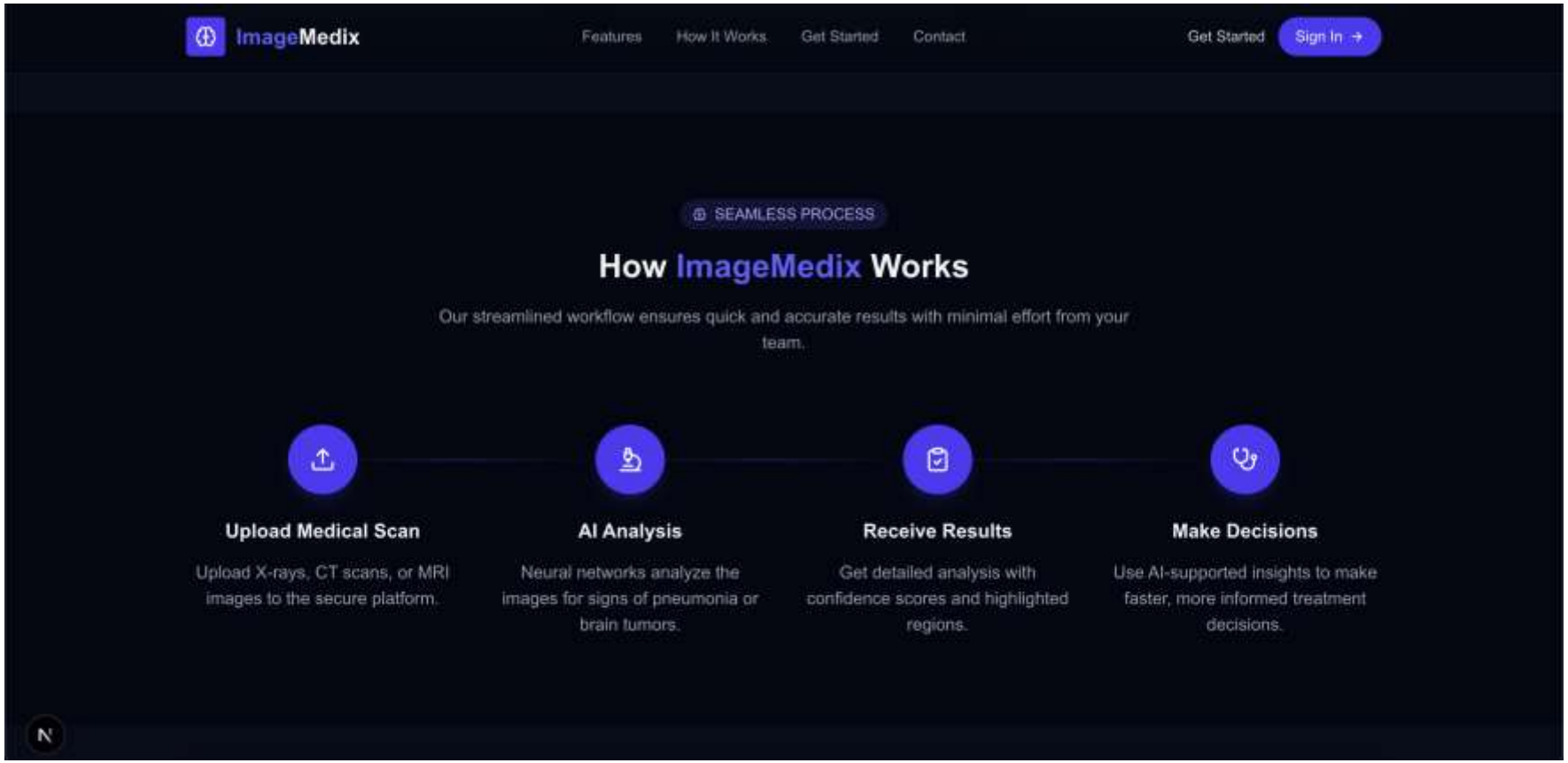
SPRINT 2 PLANNING

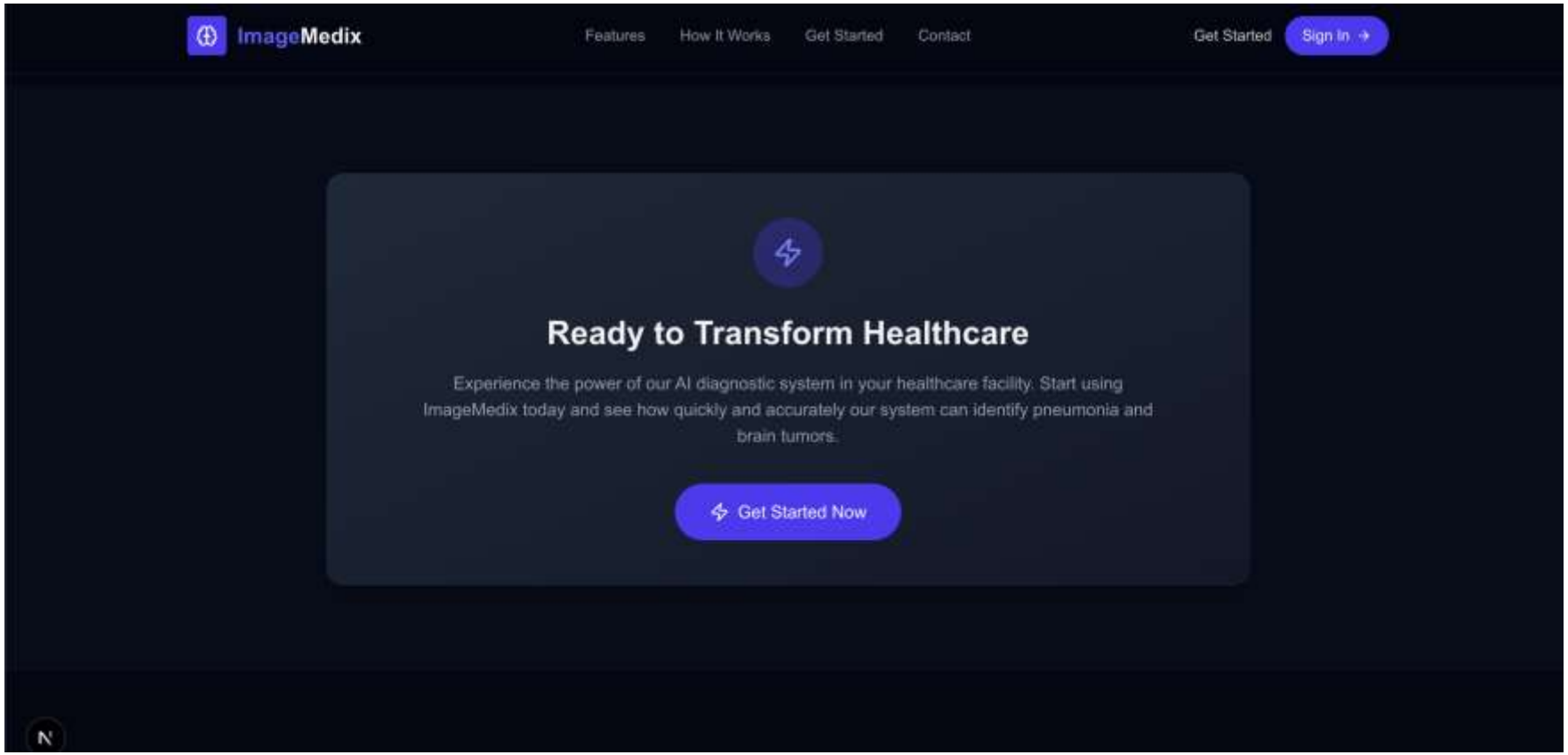
ID	User Story / Task (TS)	Acceptance Criteria (AC)	Story Points (SP)
US-03	(Carried from Sprint 1) As a user, I want a simple image upload interface so I can easily upload my medical images.	Fix image preview issue.	3
US-05	(Carried from Sprint 1) As a user, I want to see a static results page so I understand how the diagnosis will be displayed.	Fix UI styling issues.	2
TS-03	As a developer, I want to set up a backend API for image processing so that the system can analyze uploaded images efficiently.	Develop Flask/Django API to handle image uploads and analysis.	8
TS-04	As a developer, I want to connect the frontend with the backend API so that data flows seamlessly between the client and server.	Ensure API endpoints are properly integrated with React.	5

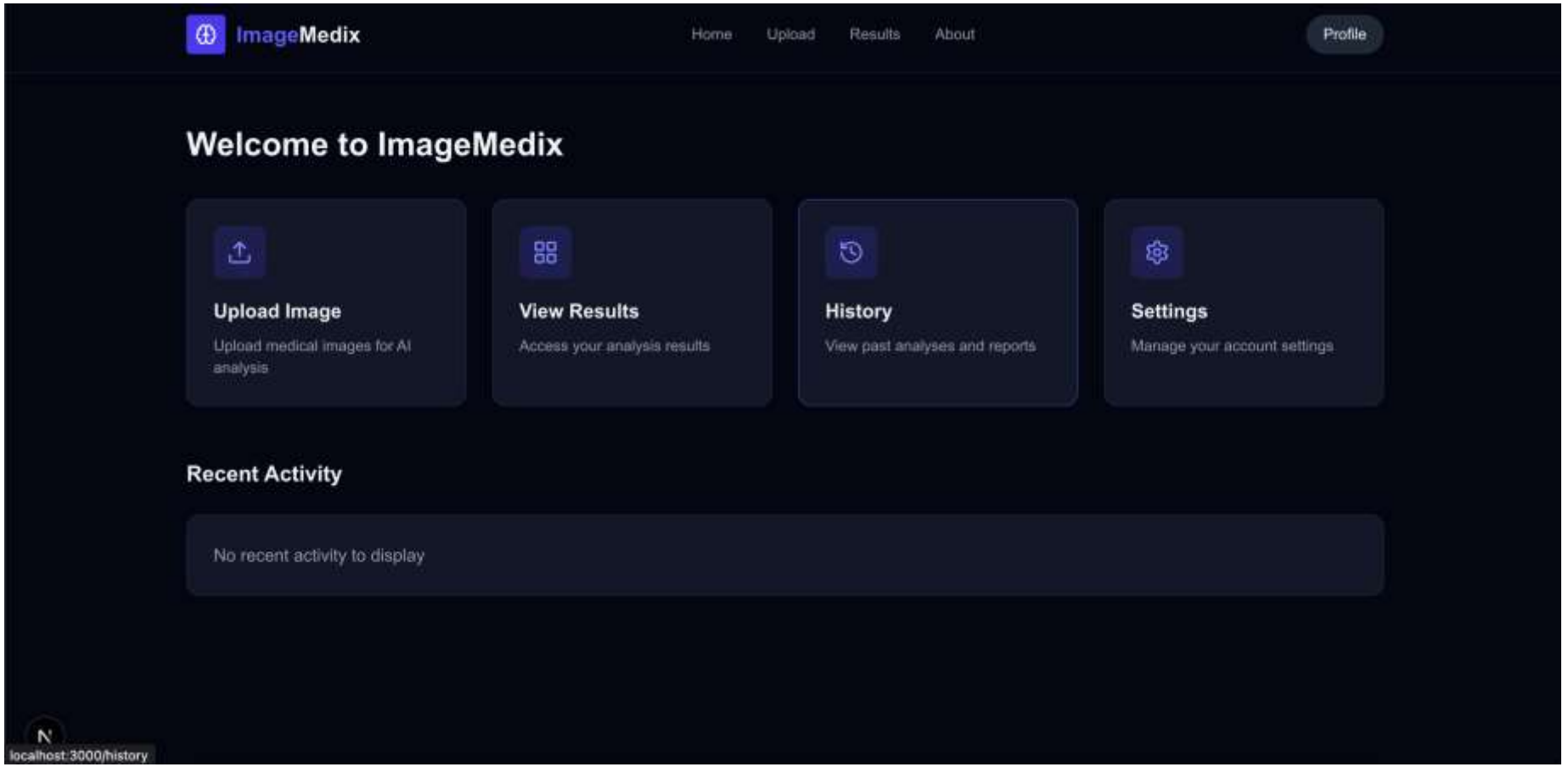
APPLICATION SCREENSHOTS

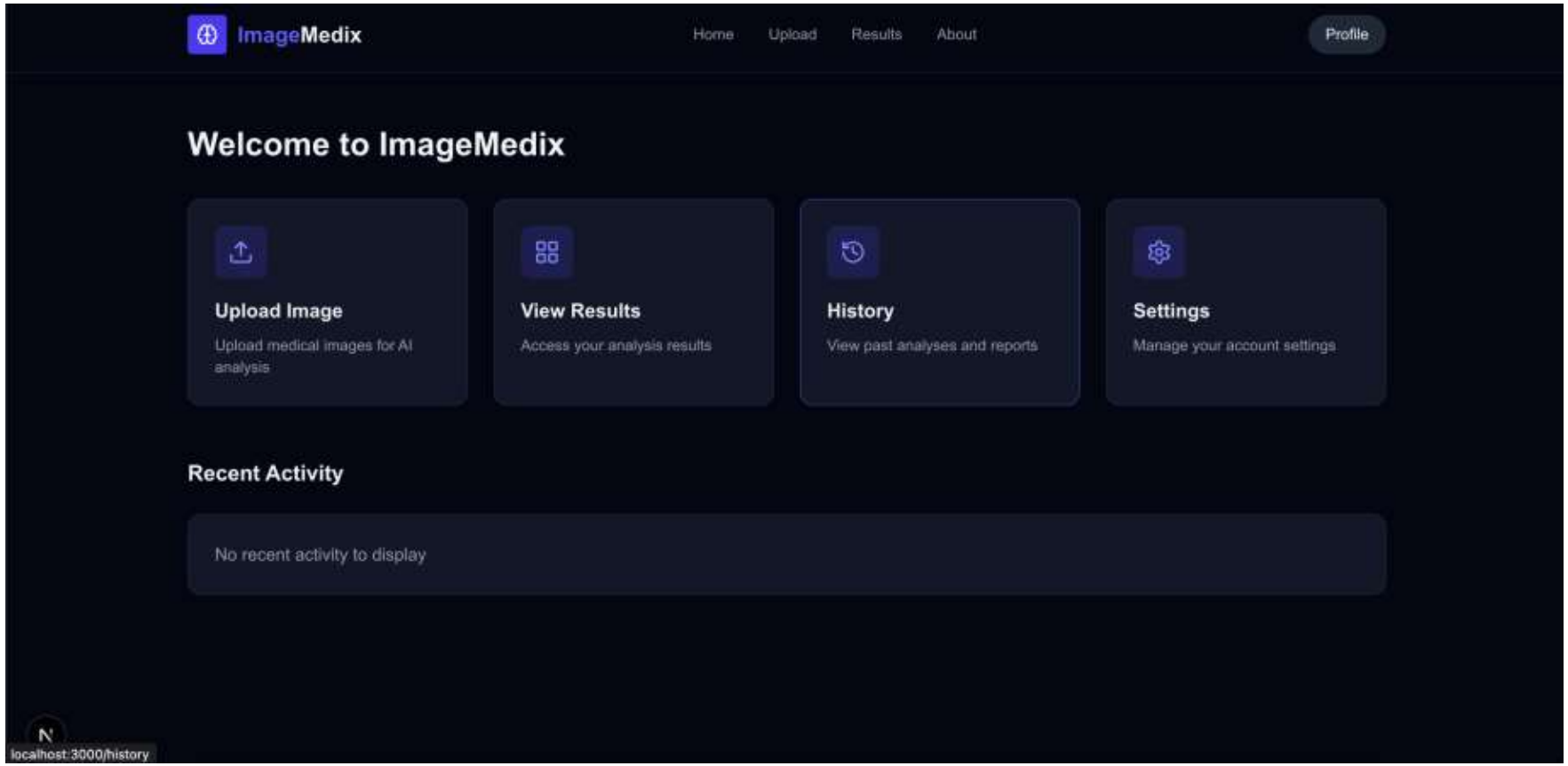


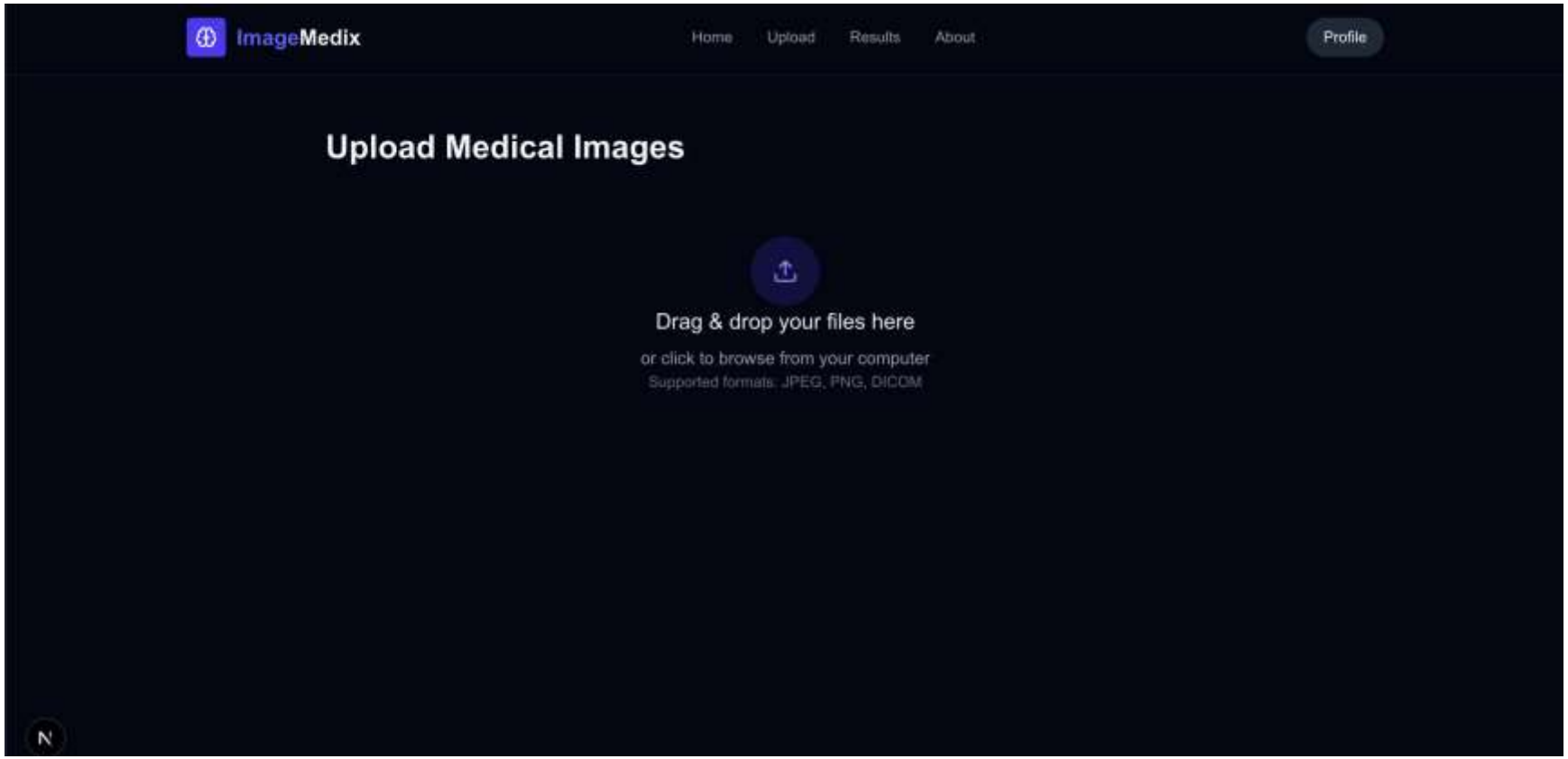

ImageMedix
Features
How It Works
Get Started
Contact
Get Started
Sign In
POWERFUL FEATURES
Advanced Capabilities
ImageMedix combines cutting-edge AI technology with a user-friendly interface to deliver exceptional diagnostic support.
Rapid Diagnostics
Analysis completed in seconds, not hours, allowing for immediate treatment decisions when time is critical.
High Accuracy
Diagnostic accuracy exceeding 90% for pneumonia and brain tumors, comparable to specialist performance.
Unified Platform
Single system for analyzing multiple types of medical images, eliminating the need for separate diagnostic tools.

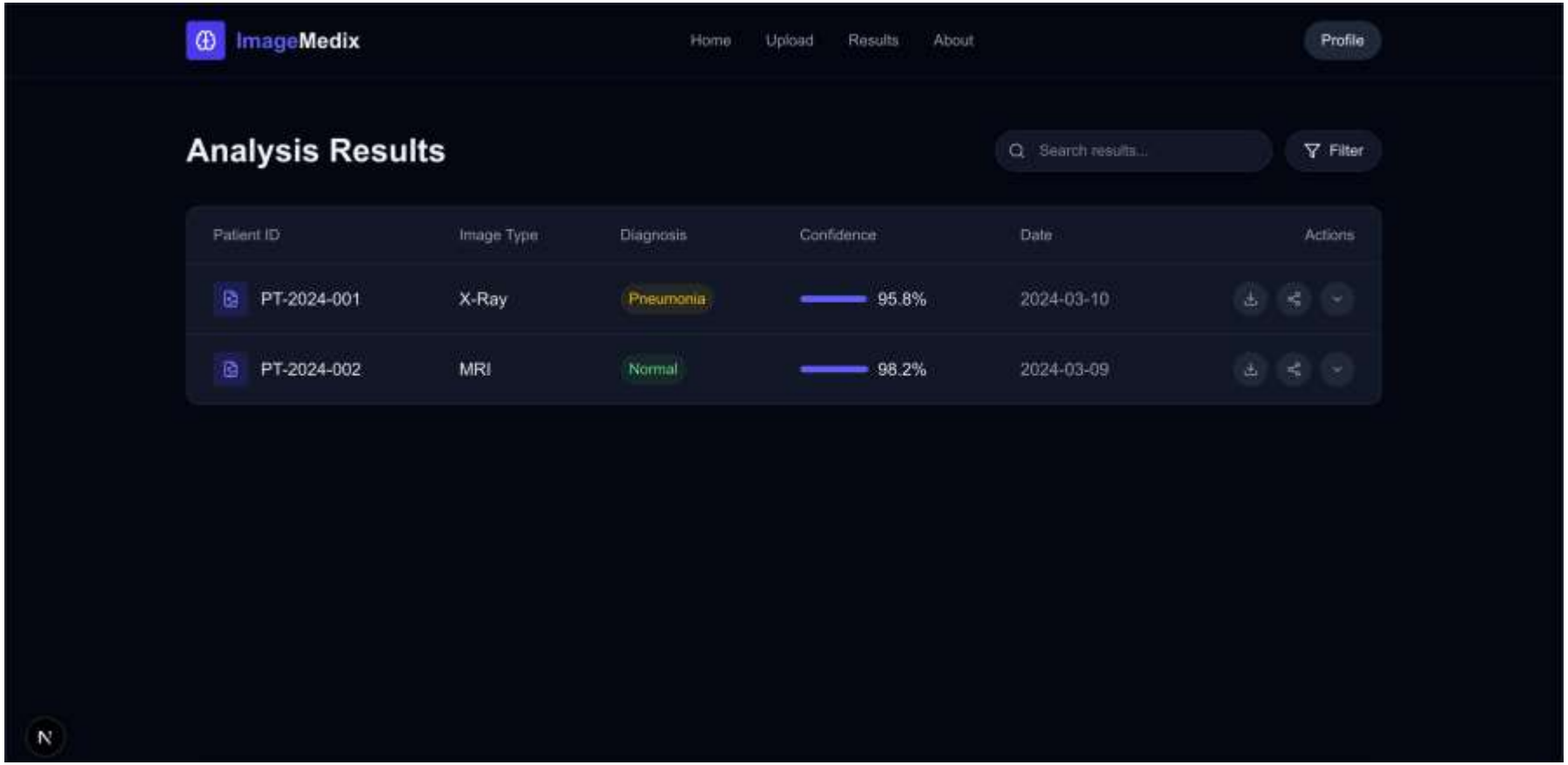

ImageMedix
Features
How It Works
Get Started
Contact
Get Started
Sign In →
SEAMLESS PROCESS
How ImageMedix Works
Our streamlined workflow ensures quick and accurate results with minimal effort from your team.
Upload Medical Scan
Upload X-rays, CT scans, or MRI images to the secure platform.
AI Analysis
Neural networks analyze the images for signs of pneumonia or brain tumors.
Receive Results
Get detailed analysis with confidence scores and highlighted regions.
Make Decisions
Use AI-supported insights to make faster, more informed treatment decisions.

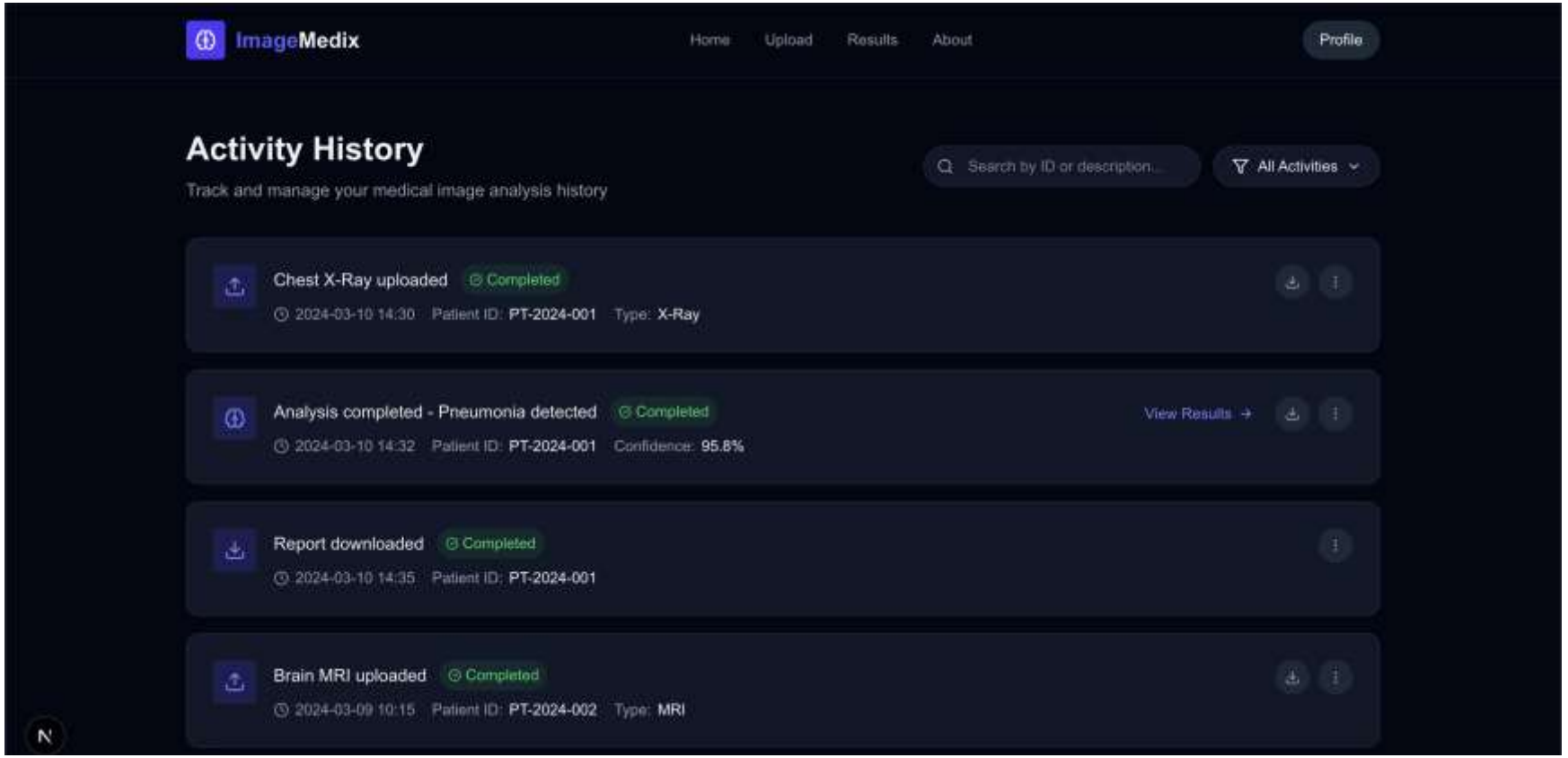

ImageMedix
Features
How It Works
Get Started
Contact
Get Started
Sign In
Ready to Transform Healthcare
Experience the power of our AI diagnostic system in your healthcare facility. Start using ImageMedix today and see how quickly and accurately our system can identify pneumonia and brain tumors.
Get Started Now


ImageMedix
Home
Upload
Results
About
Profile
Welcome to ImageMedix
Upload Image
Upload medical images for AI analysis
View Results
Access your analysis results
History
View past analyses and reports
Settings
Manage your account settings
Recent Activity
No recent activity to display
localhost:3000/history


ImageMedix
Home
Upload
Results
About
Profile
Welcome to ImageMedix
Upload Image
Upload medical images for AI analysis
View Results
Access your analysis results
History
View past analyses and reports
Settings
Manage your account settings
Recent Activity
No recent activity to display
localhost:3000/history


ImageMedix
Home
Upload
Results
About
Profile
Upload Medical Images
Drag & drop your files here
or click to browse from your computer
Supported formats: JPEG, PNG, DICOM


ImageMedix
Home
Upload
Results
About
Profile
Analysis Results
Search results...
Filter
Patient ID
Image Type
Diagnosis
Confidence
Date
Actions
PT-2024-001
X-Ray
Pneumonia
95.8%
2024-03-10
PT-2024-002
MRI
Normal
98.2%
2024-03-09


ImageMedix
Home
Upload
Results
About
Profile
Activity History
Track and manage your medical image analysis history
Search by ID or description...
All Activities
Chest X-Ray uploaded
Completed
2024-03-10 14:30 Patient ID: PT-2024-001 Type: X-Ray
Analysis completed - Pneumonia detected
Completed
View Results →
2024-03-10 14:32 Patient ID: PT-2024-001 Confidence: 95.8%
Report downloaded
Completed
2024-03-10 14:35 Patient ID: PT-2024-001
Brain MRI uploaded
Completed
2024-03-09 10:15 Patient ID: PT-2024-002 Type: MRI

WIKIPAGE LINK

https://github.com/htmw/2025S-The-Minions/wiki

APPLICATION DEMO

THANK YOU