Project Design Phase-II
Solution Requirements (Functional & Non-functional)

Date	28 june 2025
Team ID	LTVIP2025TMID45999
Project Name	Transfer learning based classification of poultry diseases for enhanced health management.
Maximum Marks	4 Marks

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Uploading the image	Know the kind of disease that the one is suffering with
FR-2	User Confirmation	Know the medicine
FR-3	Image uploading	Disease known
FR-4	Image detection	Type of disease

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The system should be intuitive and user-friendly, ensuring that users can easily navigate and interact with the application with minimal training or support. The interface should be consistent and accessible to all user types.
NFR-2	**Security**	The solution must ensure data protection and system integrity by implementing proper authentication, authorization, encryption, and regular security audits. It should comply with relevant data protection regulations and standards.
NFR-3	**Reliability**	The system should perform consistently and correctly under defined conditions without failures. It must handle unexpected conditions gracefully and ensure data integrity during failures or interruptions.
NFR-4	**Performance**	The system should respond to user inputs and process requests within acceptable time limits under normal and peak load conditions. This includes fast load times, efficient processing, and minimal latency.
NFR-5	**Availability**	The system should be available and operational 24/7 or as per the defined service-level agreement (SLA), with minimal downtime for maintenance or unexpected outages.

NFR-6	**Scalability**	The solution must be able to accommodate growth in users, data volume, and transaction load without compromising performance, by scaling up (hardware) or out (adding nodes/resources).