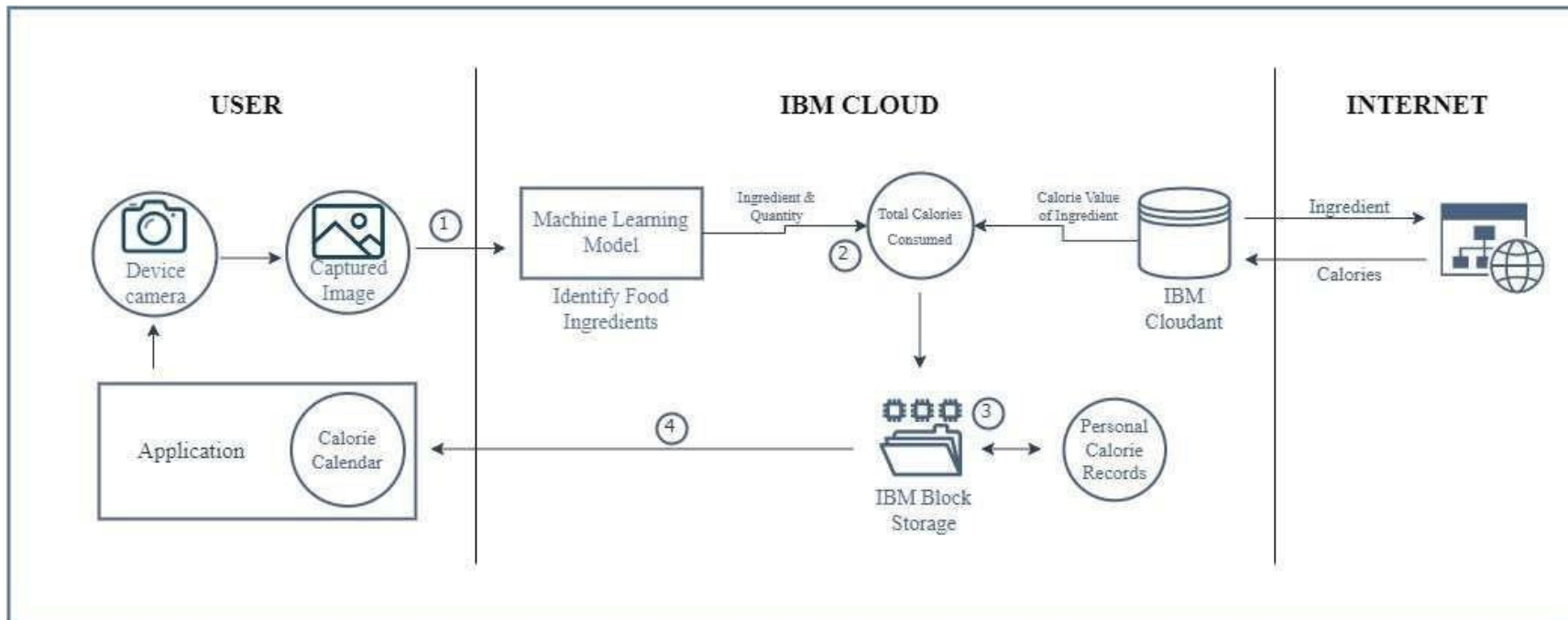


**Project Design Phase-II**  
**Technology Stack (Architecture & Stack)**

Date	14 October 2022
Team ID	PNT2022TMID23989
Project Name	Project - AI-powered Nutrition Analyzer for Fitness Enthusiasts
Maximum Marks	4 Marks

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2



**Table-1 : Components & Technologies:**

S. No.	Component	Description	Technology
1.	User Interface	An app that lets users make profiles, upload photos of the components they use in their food, and obtain a personalized nutrition calendar.	HTML, CSS, JavaScript
2.	Image Capture	Users are required to take a photo of the ingredient(s) they eat.	IBM Maximo Image Inspection
3.	Ingredient Detection Model	The ingredients used must be identified from the captured image.	Machine Learning & Image Processing using Python
4.	Calorie Consumption Monitoring	The software monitors the user's daily calorie intake and alerts them when there is an excess.	IBM Push Notifications
5.	Database of Ingredients	Ingredient information and the relevant calories are kept on file.	MySQL
6.	Cloud Database for Back-up	Here, backup copies of the application's data are kept, and consolidated reports of monthly calendars are also kept.	IBM Cloudant
7.	File Storage	A file system is used to keep track of the products consumed each day as well as the daily caloric intake. Additionally, a customized calorie calendar is created using this.	IBM Block Storage
8.	Calorie Value Consolidation	To determine the calorie counts of components that are saved in the database, a web-scraping API is used.	Beautiful Soup
9.	Machine Learning Model	To detect substances, captured photos are analyzed using machine learning algorithms.	Object Recognition Model to Label Ingredients
10.	Infrastructure (Server / Cloud)	The program is deployed to the cloud for use. Configuration of the cloud server:	Cloud Foundry

**Table-2: Application Characteristics:**

S. No.	Characteristics	Description	Technology
1.	Open-Source Frameworks	Google Colab, VS Code, Online Websites	Python, HTML, CSS, JavaScript
2.	Security Implementations	Email-based data access authentication and text encryption before file storage	SMTP, Encryption Algorithms
3.	Scalable Architecture	Applications are updated, bugs are fixed, and new features are added in response to user experience and input.	Customer feedback, reviews, and ratings
4.	Availability	Users should always be able to access the cloud-hosted application, and they shouldn't experience any problems like application crashes.	IBM Cloud
5.	Performance	The application should be able to process many requests without sacrificing the speed or quality of the results.	Testing - Black, White, and Beta Revise application in spiral model