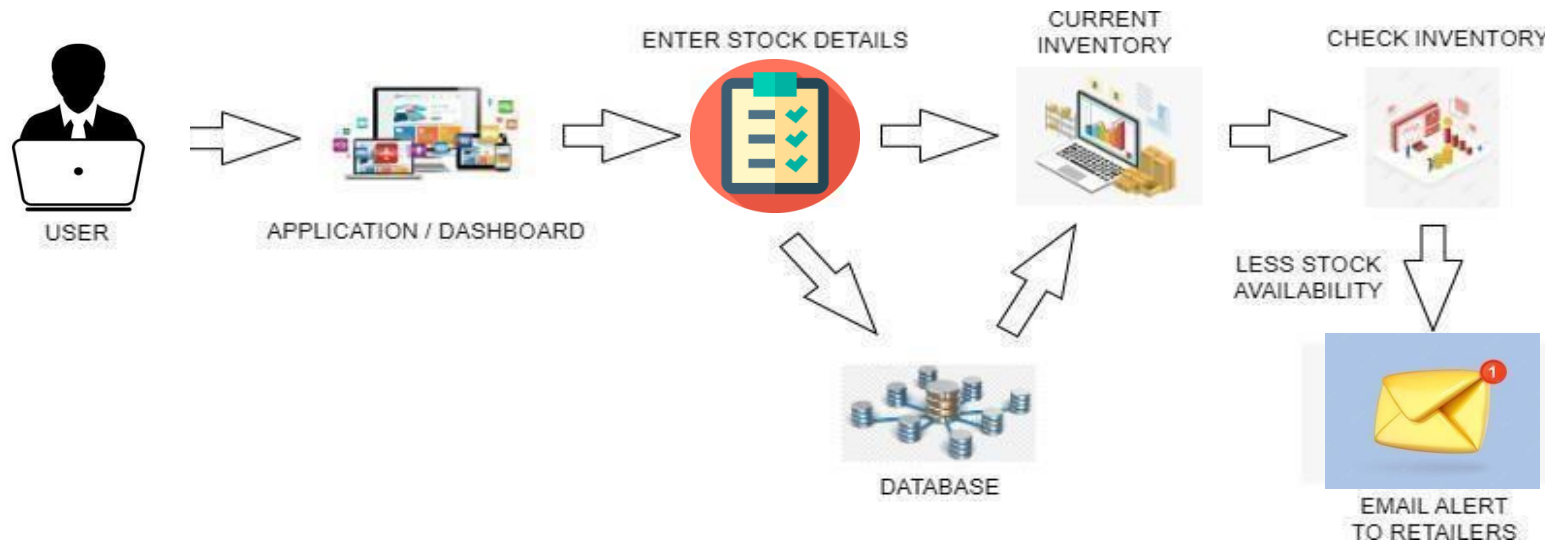


## Project Design Phase-II Technology Stack (Architecture & Stack)

|               |   |
|---------------|---|
| Date          | 1 November 2022                           |
| Team ID       | PNT2022TMID05629                          |
| Project Name  | Inventory Management System for Retailers |
| Maximum Marks | 4 Marks                                   |

### Technical Architecture:



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

| S.No | Component           | Description   | Technology   |
|------|---------------------|---|--|
| 1.   | User Interface      | How user interacts with application e.g. Web UI, Mobile App, Chatbot etc. | HTML, CSS, JavaScript  |
| 2.   | Application Logic-1 | Logic for a process in the application                                    | Python   |
| 3.   | Application Logic-2 | Logic for a process in the application                                    | IBM Watson Assistant   |
| 4.   | Database            | Data Type, Configurations etc.  | MySQL  |
| 5.   | Cloud Database      | Database Service on Cloud   | IBM DB2, IBM Cloudant etc.                                     |
| 6.   | File Storage        | File storage requirements   | IBM Block Storage or Other Storage Service or Local Filesystem |
| 7.   | External API        | Purpose of External API used in the application                           | SendGrid   |

|    |                                 |   |  |
|----|---------------------------------|---|--|
| 8. | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud<br>Local Server Configuration:<br>Cloud Server Configuration : | Local, Cloud Foundry, Kubernetes, etc. |
|----|---------------------------------|---|--|

**Table-2: Application Characteristics**

| S.No | Characteristics          | Description  | Technology                              |
|------|--------------------------|--|---|
| 1.   | Open-Source Frameworks   | Micro web framework based on python  | Flask                                   |
| 2.   | Security Implementations | Cloud access, user authentication and authorization  | Bcrypt, Encryptions, IAM Controls, etc. |
| 3.   | Scalable Architecture    | Kubernetes is an open source container orchestration engine for automating deployment, scaling, and management of containerized applications. The open source project is hosted by the Cloud Native Computing Foundation | Kubernetes                              |
| 4.   | Availability             | To customize settings for the docker CLI. The configuration file uses JSON.  | Docker CLI                              |

|    |             |  |                 |
|----|-------------|--|-----------------|
| 5. | Performance | Database caching allows you to dramatically increase throughput and lower the data retrieval latency associated with backend databases, which as a result, <b>improves the overall performance of your applications.</b> | Browser caching |
|----|-------------|--|-----------------|