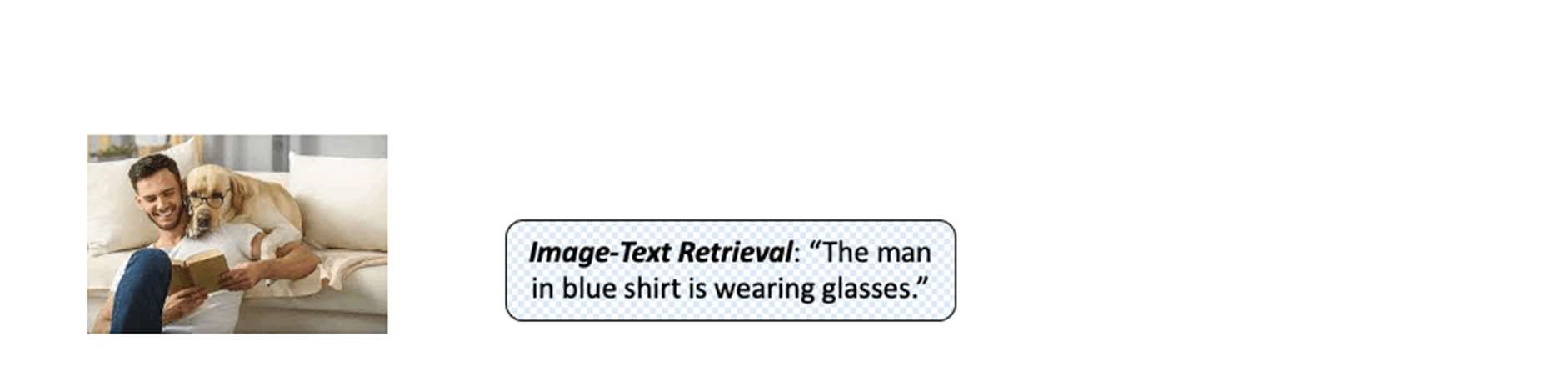
# Project Design Phase-II Technology Stack (Architecture & Stack)

|  |  |  |
| --- | --- | --- |
| Date | 22 November 2023 |  |
| Team ID | Team-591664 |  |
| Project Name | Image caption generation | |
| Maximum Marks | 4 Marks |  |

Technical Architecture for Image Cap on Genera on:



Guidelines:

* Include all the processes as application on logic or technology blocks.
* Specify whether the system is deployed locally or in the cloud.
* Indicate external interfaces, such as third-party APIs.
* Identify data storage components or services.
* Indicate the integra on with machine learning models, if applicable.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table-1: Components & Technologies: | | | |  | |
| S.No | Component | Description | | Technology | |
| 1 | User Interface | How users interact with the application (if any) | | Web UI, Mobile App, etc. | |
| 2 | Image Processing | Logic for image preprocessing and feature extraction | | Python, OpenCV, etc. | |
| 3 | Natural Language  Processing | Logic for generating captions from features | | Python, TensorFlow, PyTorch, etc. | |
| 4 | Image Dataset | Data type and configurations for image storage | | Local storage, cloud storage, etc. | |
| 5 | Caption Database | Storage for generated captions | | SQL database, NoSQL database, etc. | |
| 6 | Cloud Infrastructure | Deployment on cloud or local system | | AWS, Azure, Google Cloud, local servers, etc. | |
| 7 | External API-1 | External services for additional data  (if any) | | Image recognition API, Language translation API, etc. | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 8 | Machine Learning  Models | | Models used for image caption generation | | Pretrained image captioning models, etc. | |
| 9 | File Storage | | Storage requirements for model checkpoints, data | | Cloud storage, local filesystem, etc. | |
| Table-2: Application Characteristics: | | | | | | |
| S.No | Characteristics | Description | | Technology | | |
| 1 | Open-Source  Frameworks | List open-source frameworks used in the system | | TensorFlow, PyTorch, OpenCV, etc. | | |
| 2 | Security  Implementations | Describe security and access controls implemented | | Encryption, authentication, authorization, etc. | | |
| 3 | Scalable Architecture | Justify the architecture's scalability | | Distributed computing, microservices, etc. | | |
| 4 | Availability | Describe how the application ensures availability | | Load balancing, redundancy, failover mechanisms, etc. | | |
| 5 | Performance | Design considerations for application performance | | Caching, content delivery networks (CDNs), etc. | | |