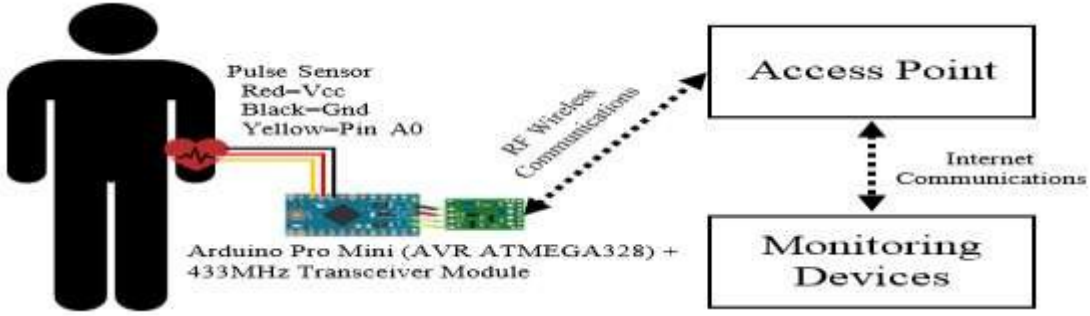
**Project Design Phase-II**
**Technology Stack (Architecture & Stack)**

| | |
|---|---|
| Date | November 2022 |
| Team ID | PNT2022TMID33743 |
| Project Name | Project -Virtual Eye - Lifeguard for Swimming Pools To Detect Active Drowning |
| Maximum Marks | 4 Marks |

# Technical Architecture:

**PULSE DETECTION**



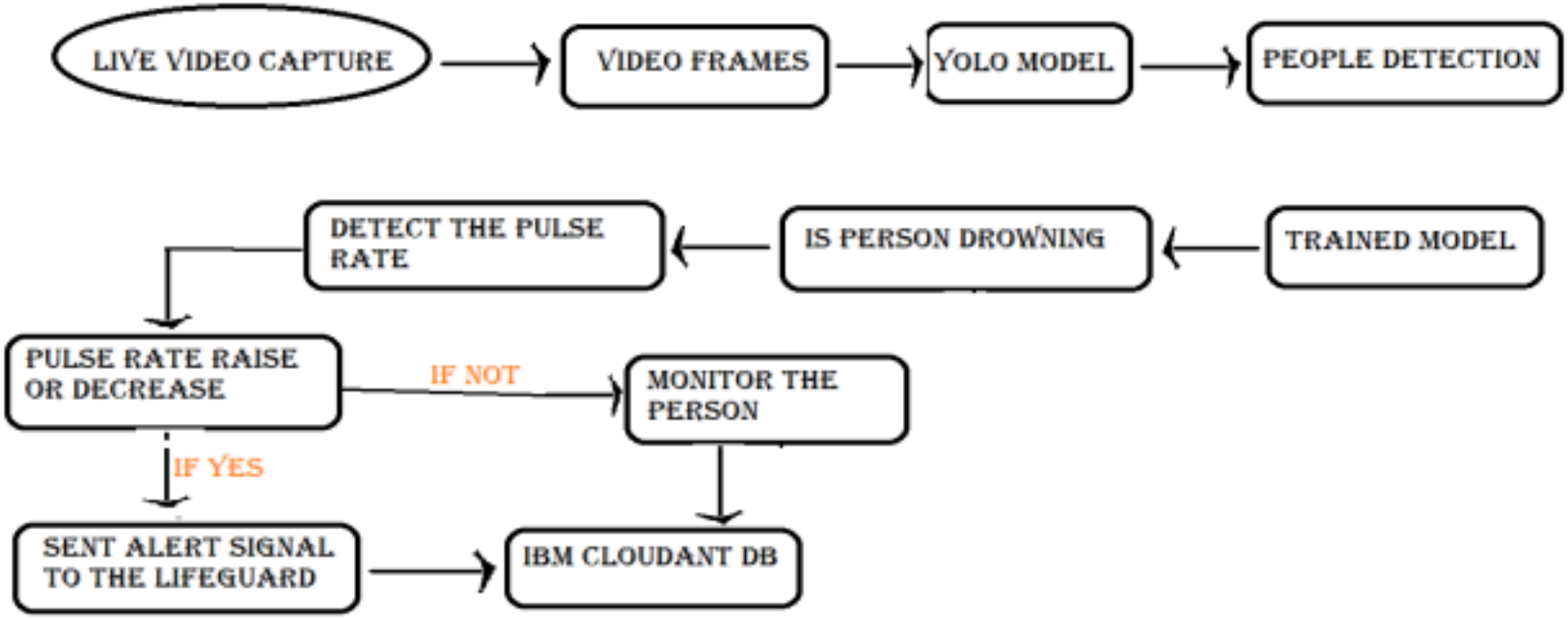

LIVE VIDEO CAPTURE
VIDEO FRAMES
YOLO MODEL
PEOPLE DETECTION
DETECT THE PULSE RATE
IS PERSON DROWNING
TRAINED MODEL
PULSE RATE RAISE OR DECREASE
IF NOT
MONITOR THE PERSON
IF YES
SENT ALERT SIGNAL TO THE LIFEGUARD
IBM CLOUDANT DB

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

| S.No | Component | Description | Technology |
|---|---|---|---|
| 1. | User Interface | An web UI provided to the admin who installed the software on the swimming pool | HTML, CSS, JavaScript / React J etc. |
| 2. | Application Logic-1 | Pre-processing the model using datasets | Java / Python |
| 3. | Application Logic-2 | Image extraction | Python |
| 4. | Application Logic-3 | A Deep learning model to find and detect for a drowning person | Machine Learning & Image Processing using Python |
| 5. | Database | Data Type, Configurations etc. | MySQL, NoSQL, etc. |
| 6. | Cloud Database | All the drowning instances with timestamp and segmented images are logged in for future uses. Database Service on Cloud | IBM DB2, IBM Cloudant etc. |
| 7. | File Storage | File storage requirements. User details pertaining to the apartment( private use case) | IBM Block Storage or Other Storage Service or Local Filesystem |

| | | | |
|---|---|---|---|
| 8. | Deep Learning Model | Captured images are processed using machine learning models to identify for drowning | Aadhar API, etc. |
| 9. | Infrastructure (Server / Cloud) | The application is deployment on cloud for use | Local, Cloud Foundry, Kubernetes, etc. |

**Table-2: Application Characteristics:**

| S.No | Characteristics | Description | Technology |
|---|---|---|---|
| 1. | Open-Source Frameworks | Python (Anaconda) open-source frameworks used | Python |
| 2. | Security Implementations | Trigger once drowning is detected and send notifications to pertaining authorities | SMTP |
| 3. | Scalable Architecture | Application is highlight scalable and multiple cameras can be introduced in further updates | Customer feedback, reviews, and ratings |
| 4. | Availability | All the time persons are under surveillance | AI |
| 5. | Performance | Many persons in the swimming pool will be detected whether the person is drowning or not | Testing - Black, White, and Beta Revise application in a iterative manner |