# Methodology

## 1-Introduction

The primary purpose of this project is to provide reliable and accurate control over the robotic arm placed in the operation room to the doctor that controls this whole system remotely. The system will integrate multiple data inputs form the operation room and provide real time feedback for the doctor. The ai module will do some processing then offer some computed insights that help with the decision making and increase the overall accuracy and safety of the operation.

The problem we are trying to solve is that the doctor might be sometimes away from the hospital or where the operation is taking place. This might cause some critical and unwanted time waste due to the Doctor physical unavailability room, whether caused by travel or any other emergency.

In this system MQTT will be the bridge of connection between the multiple parts of the project, as it will be the utility we rely on to send the data form and to the doctor, considering that the Dr will send the IMU sensor data and will receive video feedback with Ai suggestions and vital signs added to it.

Like every computer system we have , this one also have key features like:  
-gesture control: the robotic arms will be controlled according to the readings caused by the movement of the IMU sensor.  
-patient monitoring: we spend a lot of efforts on the monitoring of the patient by using the Ai module and also displaying the video with the vital signs embedded in it.

In this project we have the flow of data as the following:  
- sensors → Data sent via ESP8266 → Received by Raspberry Pi.  
- Raspberry Pi → Processes data → Sends control signals to Stepper Motors.  
- Raspberry Pi → Outputs to Display and integrates object detection results.

We used this combination of connections, sensors, methods, and we analyzed the obstacles and benefits of each decision we made to reach the most optimal productivity of the system in a way that helps with saving lives.

## 2-System architecture

An ESP device or devices (depending on the need) will be connected to a set of IMU Sensors (10DOF GY91 MPU-925 BMP280). The data collected will then be sent to the Raspberry Pi device through MQTT. The Raspberry Pi will go on and analyze this data and move the physical hand connected to the Raspberry Pi directly using a set of stepper motors accordingly. Additionally, a camera will be connected to the raspberry pi device that will include object-detection features according to the use of the whole device (surgeries, industrial, etc.). And finally, a screen will be installed beside the raspberry Pi to monitor its live movement.

A diagram of a diagram

Description automatically generated with medium confidence

## 3-Key-compnents

### the controlling side

#### - ESP8266:

A microprocessor that can connect to Wi-Fi which we will use to send and receive data through wireless connections. This micro-processor has a clock-speed of 80 MHz, 160KB of RAM, and 1 MB Flash. These specifications are more than enough since we will be using it only to read data from the sensor and give PWM to the motors.

#### -IMU sensor:

The IMU sensor in a 10 degree of freedom sensor that detects and changes through the translational axis of the rotational axis. We use this to record any hand gesture the remote doctor makes. Stepper motors: Used to control robotic arm movement. We decided to use this motor since although it is more expensive than the servo motor, it will provide more accurate and precise movement which is needed in this kind of operation.

#### -Gloves:

A simple glove with IMU sensors attached to it. We will put a sensor on each joint to track the movement of every movement. This method will allow us to change any broken sensor with ease.

### Raspberry Pi 5

Will be the main brain of the operation that will receive images and vitals of the patient and through a designed knowledge graph fed by a well-organized medical and bio-medical databases to clear some work and analyze for the doctor. And the Raspberry pi will receive through MQTT connection data from the IMU sensors placed in the glove the remote doctor will be wearing to analyze these data and imitate the exact movement the doctor makes.

### The controlled side

#### -Stepper motors:

Used to control the robotic arm movement. We decided to use this motor since although it is more expensive than the servo motor, it will provide more accurate and precise movement which is needed in this kind of operation.

#### -Camera:

We will use the camera of the raspberry pi and then we will install it directly to the raspberry and then we will use it to transport the video data to the ESP on the Doctor’s side using the MQTT protocol, this whole process will for sure increase the accuracy of the work of the Doctor.

#### -Heart Rate (Pulse) Sensors

**MAX30102:**

It continuously tracks the heart rate of the patient while maintaining its output on the display, it Is widely used in ECG sensors for real time cardiac monitoring.

#### -Blood Pressure Sensors

**Omron HEM-7320:**it measures the blood pressure continuously and it is also automatic blood pressure cuffs or invasive arterial lines.

#### -Respiratory Rate Sensors

**Piezo Respiratory Belt:**

used to monitor breathing rate through chest movement, it can also be connected to the ventilators for precise measurements.

#### -Temperature Sensors

**MLX90614:**

Body temperature is gathered as data using the skin or core temperature to increase the accuracy of the readings.

#### -Oxygen Saturation (SpO₂) Sensors

**MAX30100:**

It is clipped to the finger, and it reads the concentration of oxygen in the blood to ensure adequate oxygenation during surgery.

#### -Glucose Sensors

**Freestyle Libre:**

For non-diabetic patients, we can use it indirectly by measuring it once or in a discrete way before the operation.  
For diabetic patients, we will measure the blood level continuously to monitor their sensitive reaction to multiple factors.

#### -Electrolyte Sensors

**Medtronic Sodium Sensor:**

It is used to monitor gases in the blood during long or critical surgeries.

#### -Motion Sensors

**MPU-6050:**can be used to track equipment and surgeon movements during robotic operation.

## 4-Data transmission and processing

### MQTT

#### Overview of data flow

-The input data of the MQTT protocol will be the IMU data collection from the controller side and it will be a kinetic reading to measure if there’s a movement at the hand considering acceleration.

-the output data will be the live video feedback from the operation room, merged with the vital sensors’ readings and the integration of the Ai module.

#### Why MQTT is Ideal for This System

-it is lightweight and doesn’t require much processing which makes it a good option considering the small processor we have(raspberry).

- it is optimized for IOT.

-low latency with taking into consideration the local broker we can assign.

-It stands out from all the alternatives like “HTTP” and “WebSocket” in this specific use.

#### *Role of MQTT*

-the MQTT in this project will serve the role like the veins to the organs, it will send oxygen (Data) to the different parts of the system (organs).

#### Ensuring Reliable Communication

To make sure that the Data is received or sent correctly we can apply:

-error handling: we can double-check the received data to resolve any missing data or delayed messages.

-Encryption: we can later do an encryption protocol to the MQTT sent and received data to increase the security of the system.

#### Challenges and Mitigation Strategies

-latency issues.

-network consumption.

-system integration.

### Raspberry PI

#### Gesture replication

* We are going to receive data from the IMU sensors through the MQTT Protocol and then we will replicate the movements of the hands and fingers and project it to the stepper motors by mapping the data collected to fit the range of movement of each motor.
* While having multiple options to work with the data we chose the “numpy” and the “scipy” for motion data processing.
* For motor control and interfacing we chose the “RPI.GPIO” alongside “pigpio”.

#### Patient Data processing

* It will gather then vital sensors information and merge them will the Ai module that will be fed with all the data from medical books and research and it will analyze this data and predict illness or even warn the doctor before operating on the patient.
* We will also include an alert system that will for sure read the details of the data and notice any organ failure or any type of unwanted and abnormal bleeding during the operation.
* For better usage of these methods, we will use the “pandas” library in python for data handling and analysis. While we will use the “matplotlib” for the dynamic display of the vitals.

#### Object detection

* We are going to connect the camera to send us live video feedback, and to detect obstacles in the operation room.
* We haven’t yet chosen an object detection algorithm since we want to use the machine learning approach.

#### Knowledge Graph Integration

* The purpose of using it is to provide a context-aware insight of everything that happens in the operation room.
* We haven’t also reached any results considering this model since we don’t have any experience with Ai models.

5-Implementation Plan

*Prototype Construction:*

The arm will have stepper motors connected to its several joints, this will give the arm the closest range to a precise and accurate operation. Then a simple glove is assembled with the arms hand area connecting IMU sensors to each joint tracking every movement of the actual arm. The raspberry pi device will then be connected to the installed sensors, and through an ESP8266 microprocessor that’s going to be connected to the raspberry pi device, it will receive the data collected through the IMU sensors using MQTT connection to be analyzed by the raspberry pi resulting in the imitation of the doctor’s exact movement by the robotic arm. On the other hand, in the controlled part of the device, the stepper motors installed into the arms joints will give precise movement for the arm, and the raspberry pi’s camera will be directly connected to the raspberry transferring video data to the ESP on the doctor’s side through MQTT protocol. The raspberry device is connected to another ESP device connecting it to another set of sensors for the vitals. And finally, it is connected to an ai module used that includes several sensor types (Heart Rate (Pulse) Sensors, Blood Pressure Sensors, Respiratory Rate Sensors, Temperature Sensors, Oxygen Saturation (SpO₂) Sensors, Glucose Sensors, Electrolyte Sensors, Motion Sensors).

*Software Development:*

Write and test code for:

Sensor data collection.

Data transmission using MQTT.

Motor control algorithms.

Object detection models.

System Integration:

Combine hardware and software, ensuring seamless communication and synchronization.

Testing and Calibration:

Test each component individually, then test the entire system.

Calibrate sensors for accuracy (e.g., ensuring IMU tracks precise movements).

6-Challenges and Mitigation

*Precision and Accuracy:* In medical applications, it is essential to make sure the robotic arm executes tasks with a high degree of precision. Any deviation may result in issues.  
  
*Mitigation:*  
-Use sophisticated sensor and motor calibration procedures.  
-For increased accuracy, use stepper motors and high-resolution IMU sensors.  
-Before beginning real operations, carry out thorough testing in a simulated setting.

*Control Latency:* Responsiveness may be impacted by communication lags between the robotic arm and the control system (such as a Raspberry Pi).  
  
*Mitigation:*-Reduce latency by optimizing the devices' communication protocol.  
-To guarantee prompt replies, use real-time operating systems or give important processes in the software priority.

*Problems with the Power Supply:* Power supply disruptions may result in faults that put patients in danger.

*Mitigation:*

-Create a system with a redundant power source to guarantee ongoing functioning.

-Include battery backup devices to keep things running when the power goes out.

*Sensor Drift Challenge:* Over time, IMU sensors may drift, which could result in inaccurate arm positioning.

*Mitigation:*

-Use sensor fusion methods to reduce drift and aggregate data from several sensors.

-During operation, recalibrate sensors often or employ recurring recalibration procedures.

*Complexity of the User Interface:* Under pressure, the medical staff may find the control interface too complicated to utilise efficiently.

*Mitigation:*

Create a user interface that is easy to use and provides unambiguous visual feedback.To acquaint medical staff with the system, conduct training sessions.

*Safety and Reliability:* Making sure the patient or medical personnel are not harmed during the robotic arm's operation.

*Mitigation:*-Include safety measures such as fail-safes and emergency stop buttons.  
-Conduct comprehensive risk evaluations and adhere to medical device regulations.

*Environmental Interference:* Sensor readings may be distorted by physical obstructions or electromagnetic interference in the operation area.  
  
*Mitigation:*  
-Protect components from interference caused by electromagnetic waves.  
-To find and eliminate such barriers, do environmental assessments.

Integration with Current Frameworks Compatibility with current surgical instruments and technologies is a challenge.  
*Mitigation:*  
-Create a modular robotic system that is simple to integrate with other medical devices.  
-Work together with manufacturers of medical devices to guarantee interoperability.

*Resource and Cost Restraints:* It can be costly to develop and maintain sophisticated robotic systems.  
*Mitigation:*-For financial support, look into joint ventures with funding organizations or research organizations.  
-When feasible, use open-source software and inexpensive components.*The Challenge of Regulatory Compliance:* It might be intimidating to navigate the complicated regulatory environment for medical equipment.  
*Mitigation:*  
-Consult regulatory specialists at the outset of the development process.  
-Verify that all designs and procedures adhere to the applicable laws and guidelines for medical devices.

7-Ethical and Safety Considerations

*Ensure patient data is encrypted and stored securely:*

*Importance:* Ensuring patient confidentiality and confidence in medical processes depends on protecting patient data. Unauthorised access may result in privacy violations and possible sensitive data exploitation.  
*Method:*-To prevent unwanted access and data breaches, use robust encryption algorithms (such as AES-256) for both in-transit and at-rest data.  
-Make use of safe storage options, such encrypted databases, to guarantee that patient data is only accessible by authorised staff.  
-Update and patch software frequently to guard against vulnerabilities.  
-To find and fix any possible flaws in the system, do penetration tests and security audits.

*Validate the system rigorously to minimize risks during operations:*

*Importance:*Thorough validation is necessary to guarantee that the robotic arm performs as planned, reducing hazards that can jeopardise patient safety during operations.  
*Method:*-Use thorough testing techniques, such as system, integration, and unit testing, to ensure that every component operates as intended.  
-Perform controlled trials and simulations in a non-clinical environment to see how the system behaves in different scenarios.  
-Establish a feedback loop that analyses surgical results in order to continuously improve the system.  
-To make sure the system satisfies clinical requirements and safety standards, include medical specialists in the validation process.

*Comply with relevant regulations (e.g., medical device standards):*

*Importance:*Following legal requirements guarantees that the robotic arm satisfies safety, efficacy, and quality standards—all of which are essential for patient confidence and legal compliance.  
*Method:*-Learn about pertinent legislation, such as the Medical Device Regulation (MDR) in the EU or the FDA standards in the US, and make sure that all development lifecycles are compliant.  
-Early on in the design phase, work with regulatory consultants to understand compliance standards and get ready for required paperwork and submissions.  
-Perform risk assessments to find possible risks connected to the equipment and put mitigation plans in place.  
-To make inspections and audits easier, keep detailed records of your design procedures, testing outcomes, and compliance initiatives.

8-Conclude the Methodology

This methodology's goal is to offer a methodical framework for creating a medical robotic arm that surgeons may control during procedures. By tackling certain issues with accuracy, safety, and regulatory compliance, this strategy is essential for guaranteeing the project's success.  
  
*Important Functions of the Methodology:*-Structure and Guidance: It describes every stage of development and makes sure that parts like stepper motors and IMU sensors are successfully integrated.  
-Risk Mitigation: Priorities thorough validation to reduce surgical risks and guarantee the precise and secure operation of the robotic arm.  
-Clear Objectives: Helps the team stay focused and track progress by establishing clear goals for accuracy and data protection.  
-Improved Communication: Encourages cooperation between engineers, physicians, and regulators to match the arm's functionalities with clinical requirements.

-Iterative testing and feedback are incorporated into continuous improvement, enabling real-time adjustments based on surgical performance.