SpotMicro Minimalist Project - Egypt (Detailed)

Executive Summary

This proposal outlines a simplified SpotMicro quadruped robot focused on three core features: obstacle avoidance, movement primitives, and a remote controller interface. The project will be built in Cairo, Egypt, using locally available components, with a total hardware budget of **6,814 EGP**.

Core Features

1. **Autonomous steering:** Utilizes an ultrasonic sensor as well as stereo cameras to be able to steer autonomously while dodging obstacles
2. **Controller Interface:** Allows for remote operation via a physical RC controller or MQTT commands Web.

Hardware Components

Mechanical Platform

- **Model:** SpotMicro V4 (Community Model from CrealityCloud: https://www.crealitycloud.com/model-detail/spotmicro-robotic-dog-v4)
- **Materials:** PLA 3D-printed parts, to be fabricated by a local print shop.
- **Printer:** Creality Ender 3
- **Infill:** 15%
- **Colors:** White, Black, Gray
- **Actuation:** Twelve high-torque MG996R servo motors (three per leg) provide comprehensive control over the robot's locomotion.

Electronics & Control

Component	Model	Price (EGP)	Function & Description	Source (Link)
Microcontroller	Arduino Nano RP2040 Connect	1,750	Main control board with onboard Wi-Fi for MQTT communication and an IMU for balance.	RAM Electronics: Link
Servo Driver	PCA9685 16-Channel PWM	210	I²C-controlled driver for precise, synchronized	MakerSelectronics: Link

			control of all twelve servos.	
Servos	MG996R Metal-Gear (×12)	2,520	High-torque servos providing powerful and precise leg actuation for dynamic movement.	MakerSelectronics: Link
Ultrasonic Sensor	HC-SR04	60	Provides distance measurement for forward obstacle detection and avoidance.	MakerSelectronics: Link
Stereo Cameras	OV9655 (×2)	700	A pair of cameras used for depth perception	RAM Electronics: Link
Battery	12V Li-ion 7000mAh	1,200	Powers the entire system, including the microcontroller and all twelve servos.	(Generic Estimate)
Display	2x16 line LCD display I2C	170	Small screen for providing real-time debug information and status feedback.	
DC to DC step down BUCK converter	LM2596S	50	Power the logic microcontroller 12V down to	https://www.ram-e-shop.com/shop/dc-dc-100-dc-dc-step-d

			5V	own-voltage-converter-adjustable-3a-lm2596s-4-38vdc-to-1-5-35vdc-sku-dc100-7257
Step Down Buck Converter Power Supply XL4016 PWM Adjustable 4-40V To 1.25-36V Step-Down Board Module XH-M401	XL4016 PWM	150	12V down to 6V for your 12 servos.	https://www.ram-e-shop.com/shop/dc-dc-105-dc-dc-step-down-voltage-converter-adjustable-8a-xl4016-4v-40vdc-to-1-25v-36vdc-sku-dc105-7670
Rocker Switch (10A)	Switch	4	To switch on and off the robot	https://makerselectronics.com/product/rocker-switch-spdt-2-position-on-off/
Small 6V DC Motor (130-size)	Micro Metal 6V DC geared motor 30RPM GA12-N20	50	Used to actuate a simple "wagging tail" for feedback.	
H-Bridge L298N Dual H-Bridge	L298N Dual H-Bridge Motor Driver			

Motor Driver				
Total		**6,814 EGP**		

3. Subsystems & Operational Logic

This project is broken into four main subsystems, designed to meet the course requirements (3 subsystems, 2 parallel).

- **Subsystem 1:** Core Balance & Locomotion (Sequential)
- **Subsystem 2:** Autonomous Navigation (Parallel)
- **Subsystem 3:** Remote MQTT Control (Parallel)

Subsystems 2, and 3 are designed to run in parallel. The robot can be autonomously navigating (Sub 2) while simultaneously listening for MQTT command (Sub 3).

Subsystem 1: Core Balance & Locomotion (Sequential)

- ***Story Overview:*** *The robot's core function is to maintain balance while standing or walking, using an IMU to correct its posture.*
- ***Inputs:*** `LSM6DSOXTR 6-axis IMU` *(Analog-style data stream).*
- ***Outputs:*** `12x MG996R Servo Motors`, `SSD1306 OLED Display`.
- ***Overall Logic:***
  1. ***Initiation:*** *Robot is powered on via the* `Rocker Switch`*. The OLED Display shows "BOOTING...".*
  2. ***Calibration:*** *The system initializes the* `LSM6DSOXTR IMU`*. The OLED displays "CALIBRATING IMU... DO NOT MOVE".*
  3. ***Ready State:*** *After 3 seconds, calibration is complete. The robot moves to a default "STAND" pose. The OLED displays "STATUS: IDLE".*
  4. ***Sequential Loop (Balancing):*** *The system continuously runs this loop:*
     - *Read the current* `pitch` *and* `roll` *from the IMU.*
     - ***IF*** `|pitch| > 10` ***OR*** `|roll| > 10` ***degrees:*** *The system calculates the micro-adjustments needed for the 12 leg servos to counteract the tilt and restore balance.*
     - *The* `PCA9685` *sends the new commands to the servos.*
  5. ***Command Execution:*** *When a command is received from another subsystem (e.g., "WALK_FWD"), this subsystem executes the gait cycle, while the "Sequential Loop (Balancing)" continues to run to ensure stability during*

movement.

Subsystem 2: Autonomous Navigation (Parallel)

- ***Story Overview:*** *In "Patrol Mode," the robot uses its sensors to navigate a space, avoiding simple (walls).*
- ***Inputs:*** `HC-SR04 Ultrasonic Sensor`, `2x OV9655 Stereo Cameras`, `LSM6DSOXTR 6-axis IMU`
- ***Outputs:*** `SSD1306 OLED Display`, *(Sends commands to Subsystem 1).*
- ***CV:*** *Stereo vision depth mapping for complex obstacle/drop-off detection.*
- ***Overall Logic:***
    1. ***Activation:*** *The robot receives a "PATROL" command (from Subsystem 3 or 4). The OLED displays "STATUS: PATROLLING".*
    2. ***Default State:*** *The system sends a "WALK_FWD" command to Subsystem 1.*
    3. ***Sensing Loop (Parallel):*** *The system continuously runs two sensing tasks at the same time:*
        - ***Task A (Ultrasonic):*** *The* `HC-SR04` *continuously pings for immediate frontal obstacles.*
        - ***Task B (Stereo Vision):*** *The* `2x OV9655 Cameras` *continuously process frames to build a basic depth map of the environment,*
        - ***Task C (IMU):*** *state estimation to check where the robot is*
    4. ***Conditional Logic (Obstacle Detected):***
        - ***IF*** **`Stereo Vision detects a drop-off`*****:*** *This is a high-priority interrupt. The system sends "EMERGENCY_STOP" then "TURN_180" to Subsystem 1. The OLED displays "DANGER: DROP-OFF!".*
        - ***IF*** **`HC-SR04 distance < 20cm`** ***(Simple Wall):*** *The system sends "STOP" then "TURN_LEFT_90" to Subsystem 1. The OLED displays "STATUS: AVOIDING".*
    5. ***Resumption:*** `After a "TURN_LEFT_90" command, the IMU (Task C) verifies the robot has turned ~90 degrees before returning to "WALK_FWD".`

Subsystem 3: Remote MQTT Control (Parallel)

- ***Story Overview:*** *The robot can be fully controlled and monitored remotely over Wi-Fi via an MQTT dashboard, allowing for high-level commands and status checks. WEB0*
- ***Inputs:*** `Arduino Nano RP2040 Connect` *(Wi-Fi Module).*
- ***Outputs:*** *(Sends commands to all other Subsystems).*
- ***Web Interface:*** *Fulfills the "Web Interface" requirement via MQTT protocol.*
- ***Overall Logic:***
    1. ***Initiation:*** *On boot (after Subsystem 1 is "IDLE"), the* `RP2040` *connects to the designated Wi-Fi network and MQTT broker. The OLED displays "MQTT CONNECTED".*

2. ***Listening Loop (Parallel):*** *The system subscribes to the* `spotmicro/command` *topic and listens in the background.*
3. ***Command Execution:***
   - ***IF*** ***`Message == "PATROL"`:*** *Activates Subsystem 2.*
   - ***IF*** ***`Message == "STOP"`:*** *Sends "STOP" to Subsystem 1.*
   - ***IF*** ***`Message == "TAIL_WAG"`:*** *Triggers the DC motor.*
   - ***IF*** ***`Message == "STATUS_REQUEST"`:*** *The robot gathers data (IMU status, Obstacle status) and publishes it to the* `spotmicro/status` *topic.*
   - *IF Message == "TURN_LEFT_X": The robot will turn left with the value X degrees*
   - *IF Message == "TURN_RIGHT_X" The robot will turn right with the value X degrees*
   - *IF Message == "WALK_FWD" The robot will move forward*
   - *IF Message == "WALK_BWD" The robot will move backward*