Technology Selection Justification

Battery-Powered Relay Controller

1. Microcontroller Selection

Arduino Uno (ATmega328P)

I selected the Arduino Uno for this prototype implementation due to the following considerations:

- **Availability**: The Arduino Uno was immediately available in my inventory, which allowed for faster prototyping and development.
- **Development Environment**: The Arduino IDE offers a straightforward programming interface with extensive libraries.
- **5V Logic**: Naturally compatible with most relay modules without additional level shifters.
- **I/O Capability**: Sufficient pins for the required functions (button input, relay control, LED indicator, and analog battery monitoring).

Better Alternative for Production: Arduino Pro Mini

For a production version or further development, the **Arduino Pro Mini** would be a superior choice for these reasons:

- **Lower Power Consumption**: The Pro Mini consumes significantly less power (~3.5mA at 8MHz/3.3V compared to ~15mA for Uno).
- **Smaller Form Factor**: Much more compact, allowing for a smaller final device.
- **Simplified Power Supply**: The 3.3V version could potentially operate directly from the Li-ion battery in some configurations.
- **Cost Efficiency**: Generally less expensive than the Uno when considering production quantities.

I chose the Uno for the prototype due to immediate availability and simplified development, but acknowledge that optimizing for power consumption would require transitioning to the Pro Mini or a similar low-power microcontroller.

2. Battery Technology

18650 Li-ion Battery

I selected the 18650 lithium-ion battery for the following reasons:

- **High Energy Density**: Provides 2000-3500mAh capacity in a relatively small package.
- **Rechargeability**: Can be recharged hundreds of times, making the device more economical over time.
- **Widely Available**: Standard form factor with good availability and reasonable cost.
- **Voltage Characteristics**: 3.7V nominal voltage is suitable for boosting to the 5V required by most components.

TP4056 Charging Module

This module was chosen because it provides:

- **Complete Charging Solution**: Handles all aspects of Li-ion charging with proper CC/CV profiles.
- **Modern Connectivity**: USB-C interface for convenient charging from standard power sources.
- **Integrated Protection**: Built-in overcharge, over-discharge, and short-circuit protection circuits.
- **Status Indication**: Charging and full indicators for user feedback.

3. Voltage Regulation

MT3608 Step-up Converter

This boost converter was selected because:

- **Efficient Operation**: Up to 93% efficiency when converting from Li-ion voltages to 5V.
- **Wide Input Range**: Handles the full Li-ion discharge cycle (4.2V down to 3.0V).
- **Sufficient Current**: Capable of supplying enough current for both the Arduino and relay coil.
- **Small Footprint**: Compact size suitable for a portable device.

4. Relay Selection

5V 10A Relay Module

I chose a standard 5V relay module with these characteristics:

- **Sufficient Rating**: 10A capacity exceeds the required 6A for 110/240 VAC loads.
- **Optoisolated Input**: Provides electrical isolation between the control circuit and high-voltage switching.
- **Compatible Control Voltage**: 5V control signal matches the system voltage from the step-up converter.
- **Indicator LED**: Built-in LED for visual confirmation of relay state (in addition to our status LED).

5. Future Improvements and Alternatives

Potential Technology Enhancements

1. **Microcontroller Alternatives**:
    - **ATtiny85**: For even lower power consumption if fewer I/O pins are acceptable.
    - **ESP32-C3**: If wireless connectivity would be beneficial (though with higher power requirements).
    - **STM32L0 Series**: For advanced low-power modes and more processing capability.
2. **Power Management Improvements**:
    - **Load Switches**: To completely disconnect the relay when not in use.
    - **INA219**: Precision current/voltage monitoring for more accurate battery status.
    - **TPS61023**: More efficient boost converter with lower quiescent current.
3. **Interface Enhancements**:
    - **OLED Display**: To show battery level, selected frequency, and device status.
    - **Rotary Encoder**: For more intuitive frequency selection.
    - **Bluetooth LE**: For remote control and monitoring.

6. Development Limitations and Future Work

This project was developed to meet the assignment requirements within the available timeframe. Several hardware and testing improvements could be implemented in future iterations:

- **Additional Components**: The circuit could benefit from bypass capacitors near the microcontroller power pins and an RC snubber circuit across the relay contacts to reduce EMI and extend relay life.
- **Hardware Refinements**: A custom PCB instead of breadboard would improve reliability and reduce size. Adding a voltage supervisor IC would provide more reliable low-battery detection.
- **Testing Scope**: Testing was limited to basic functionality verification. More extensive testing would be valuable, particularly for battery life and relay reliability under various load conditions.

7. Conclusion

The technology choices for this prototype represent a balance between availability, functionality, and power efficiency. While the Arduino Uno served well for the prototype phase, transitioning to a more power-efficient microcontroller like the Arduino Pro Mini would be the next logical step for optimizing battery life in a production version. The prototype effectively demonstrates all required functionality, but with additional development time, both the hardware selection and firmware implementation could be further refined for optimal performance and power efficiency.