

IoT System Design Methodology

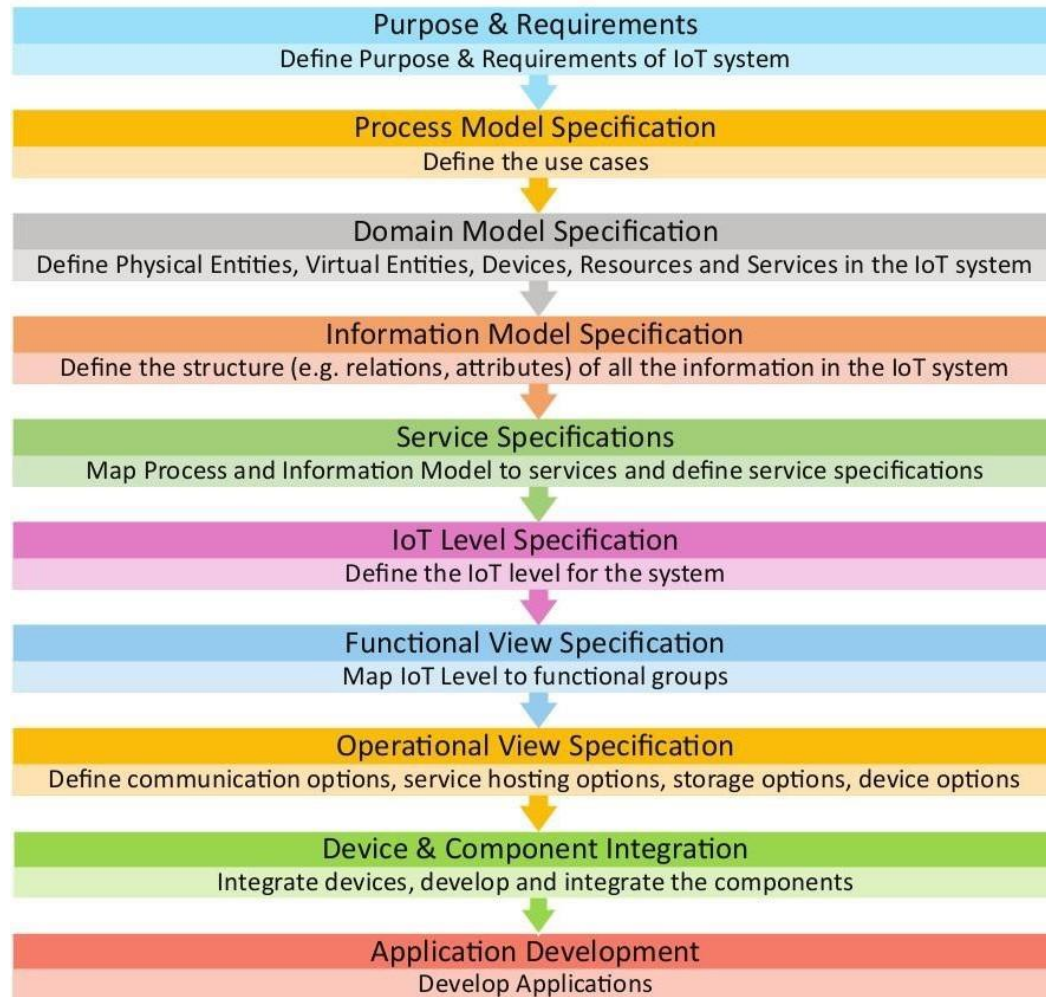
Prof.S.GOPALAKRISHNAN,

ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

SNS COLLEGE OF TECHNOLOGY-COIMBATORE

IoT Design Methodology



IoT System Design : Home Automation

Step:1 - Purpose & Requirements

- Purpose : A system that allows controlling of the lights in a home remotely using a web application
- Behaviour : The home automation system should have auto and manual modes.

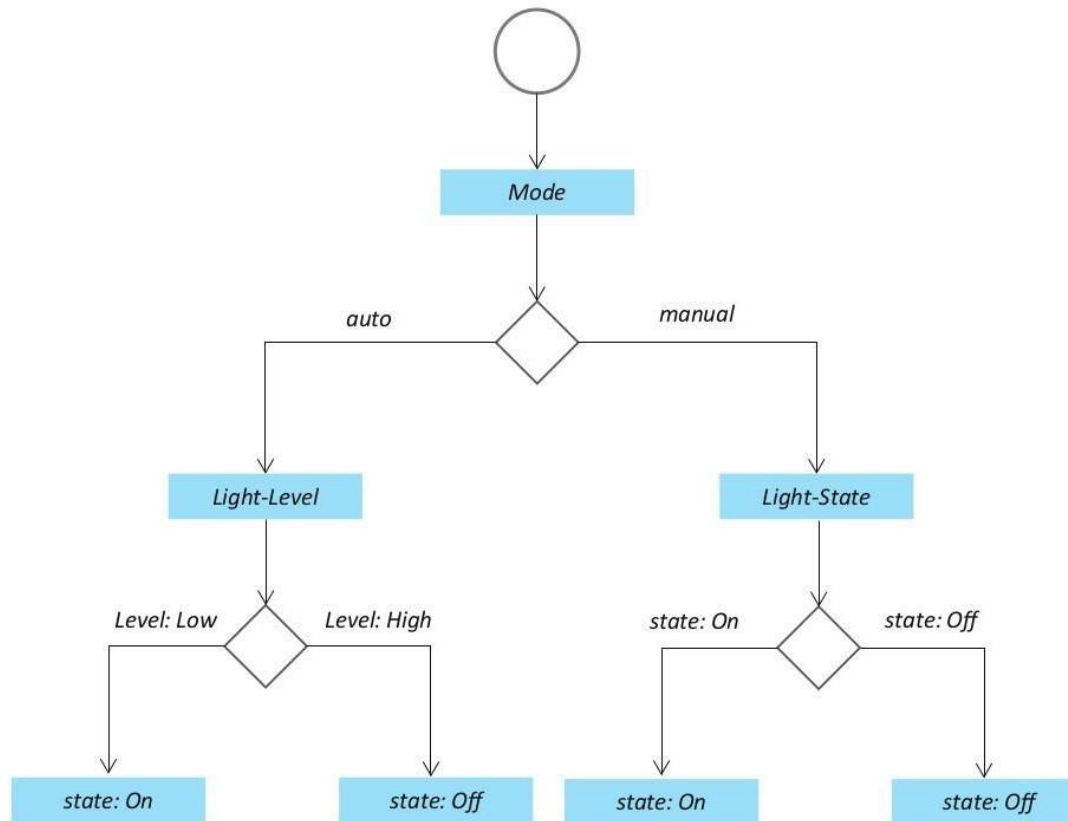
Auto mode - the system measures the light level in the room, switches light when dark.

Manual mode - the system provides the option of manually and remotely switching on/off the light.

- System Management Requirement : The system should provide remote monitoring and control functions.
- Data Analysis Requirement : system should perform local analysis of data
- Application Deployment Requirement : The application should be deployed locally on the device, but should be accessible remotely.
- Security Requirement : The system should have basic user authentication capability.

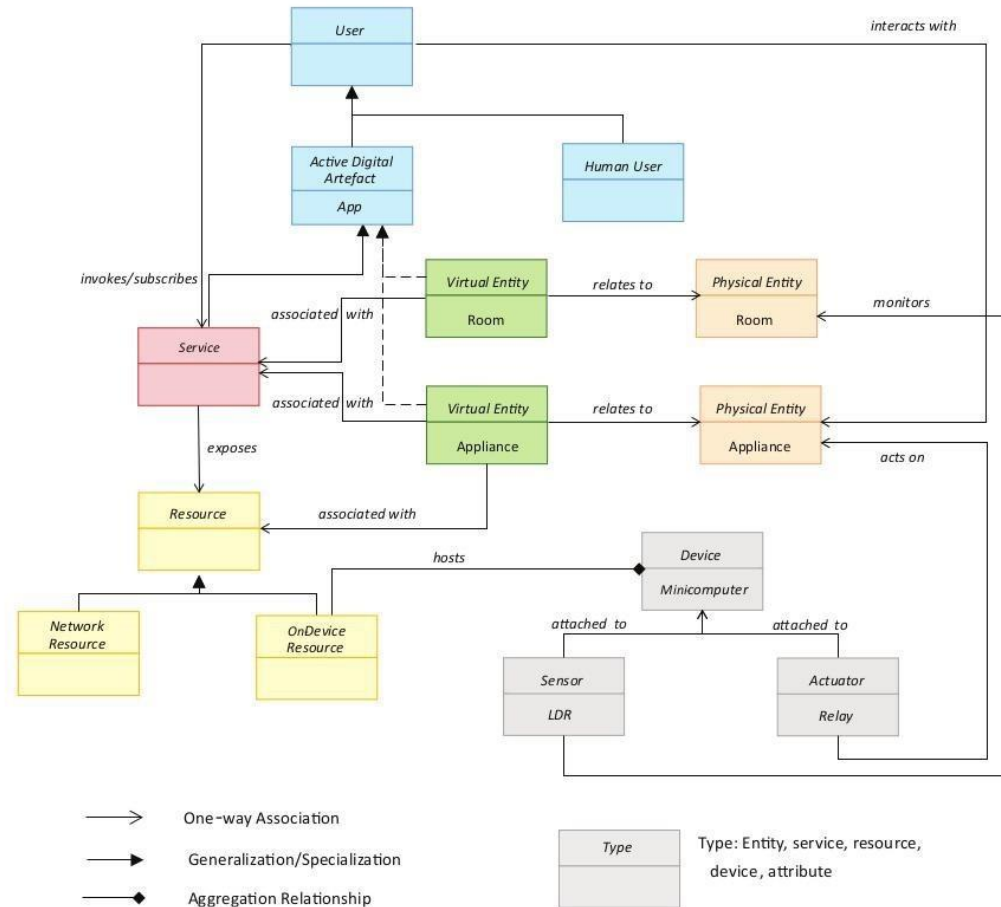
IoT System Design : Home Automation

Step : 2 - Process Specification



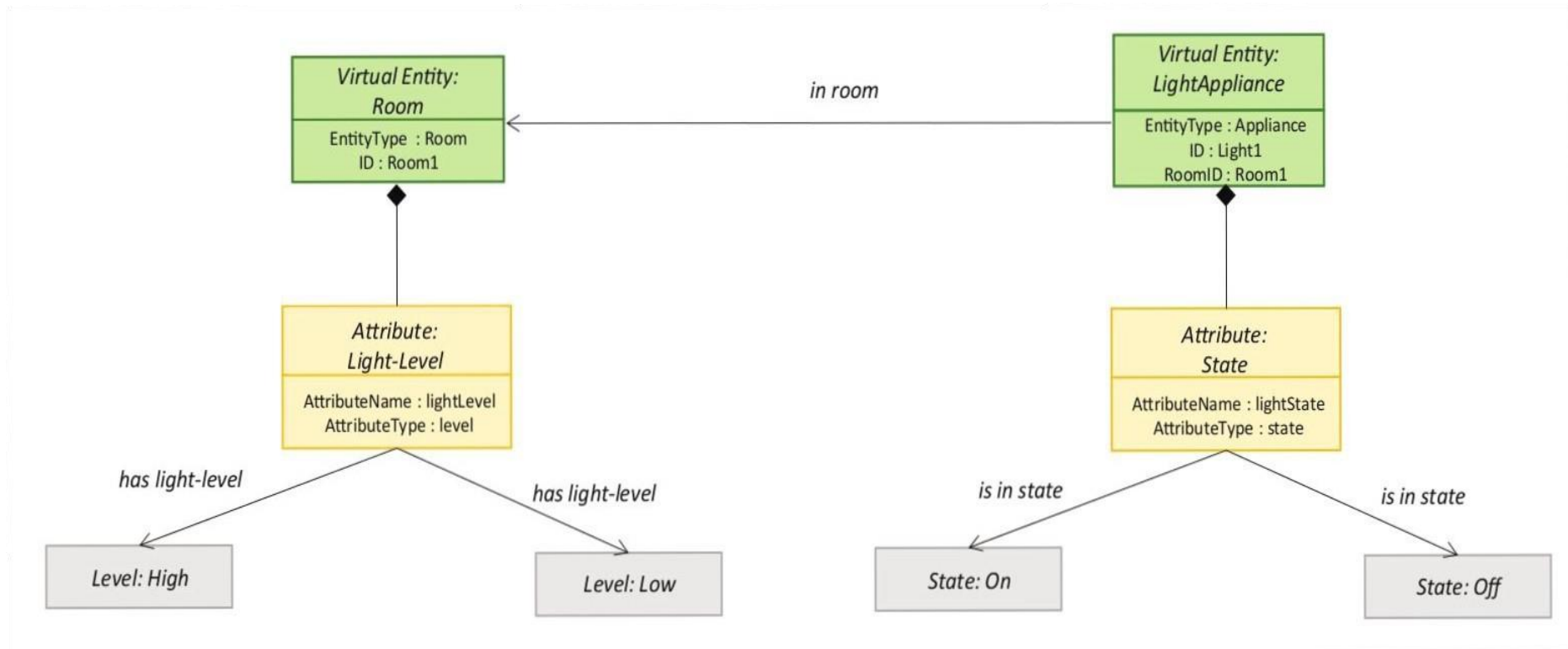
IoT System Design : Home Automation

Step : 3 - Domain Model Specification



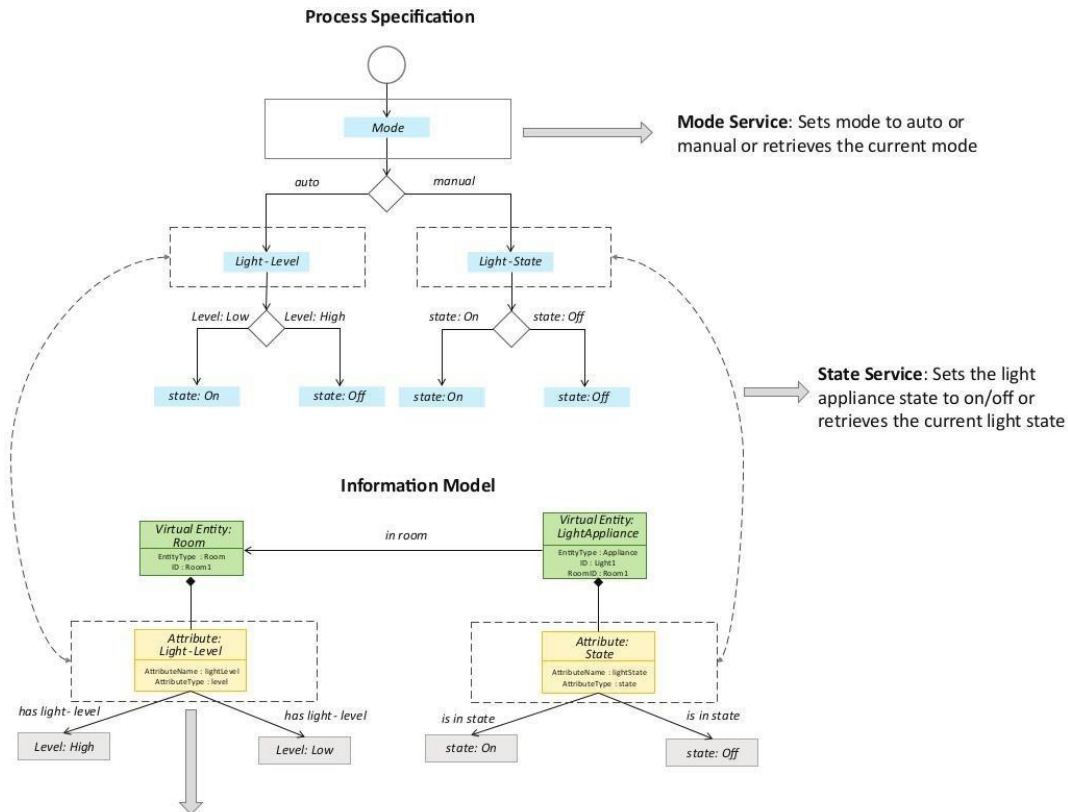
IoT System Design : Home Automation

Step : 4 - Information Model Specification



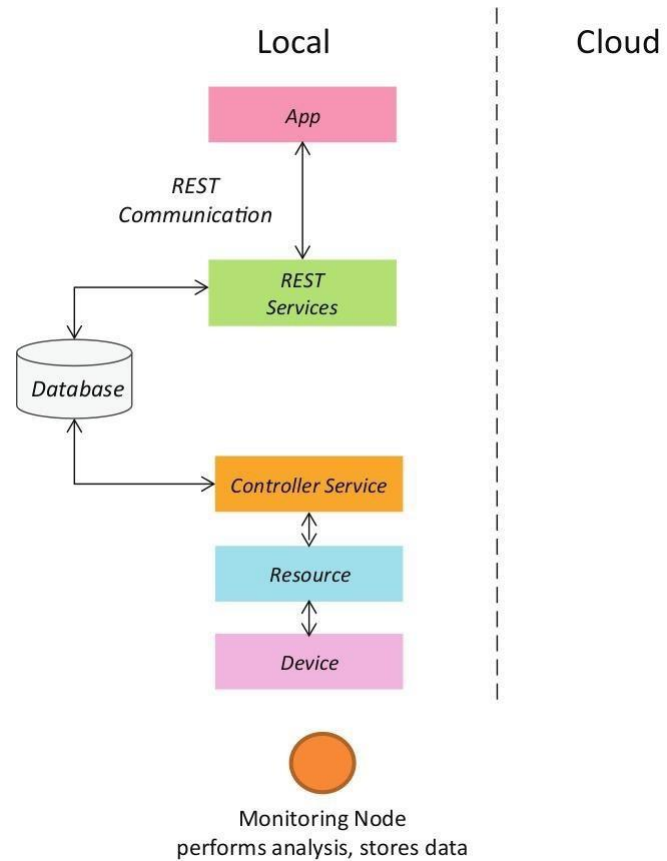
IoT System Design : Home Automation

Step : 5 - Service Specification



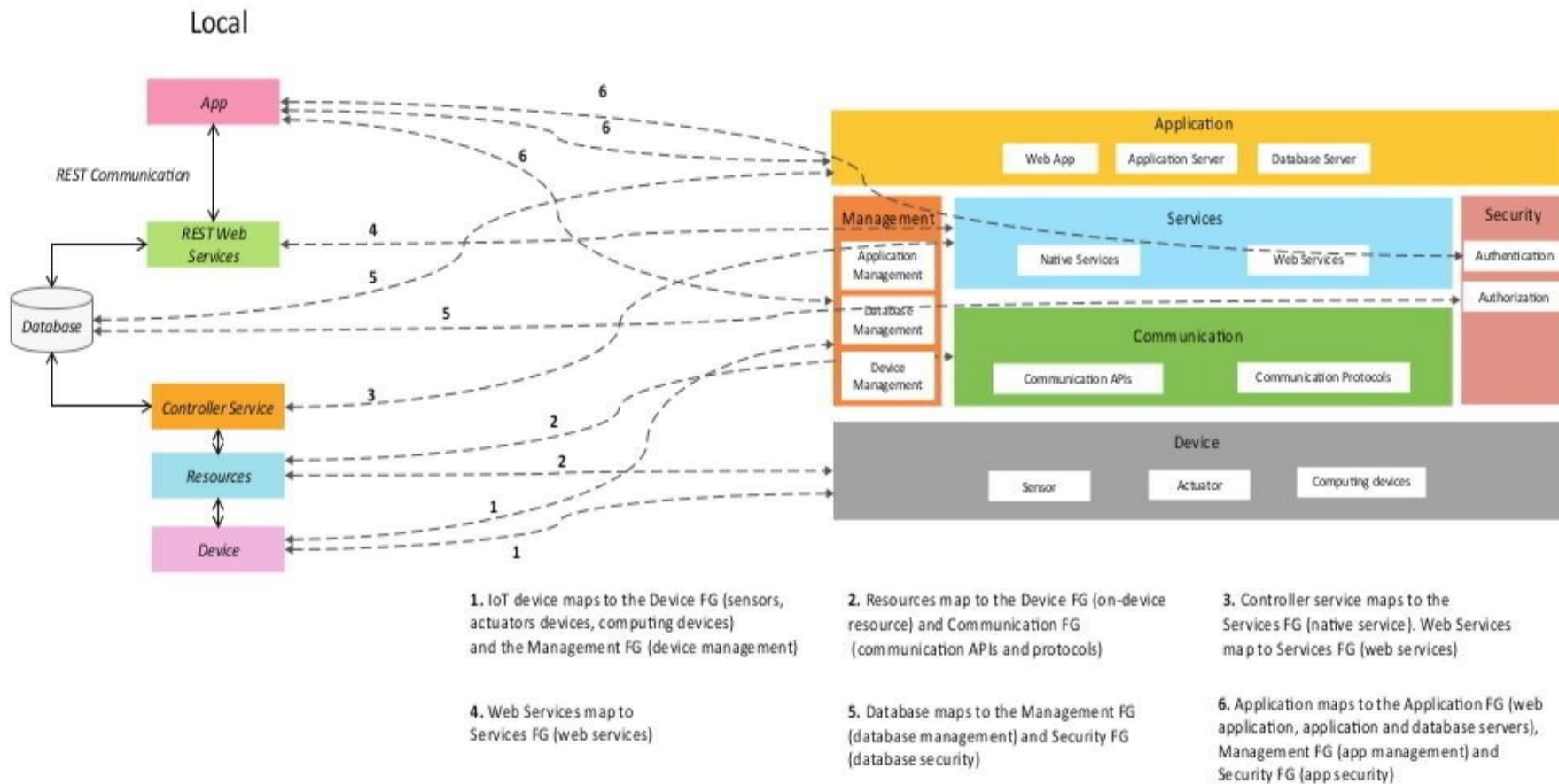
IoT System Design : Home Automation

Step : 6 – IoT Level Specification



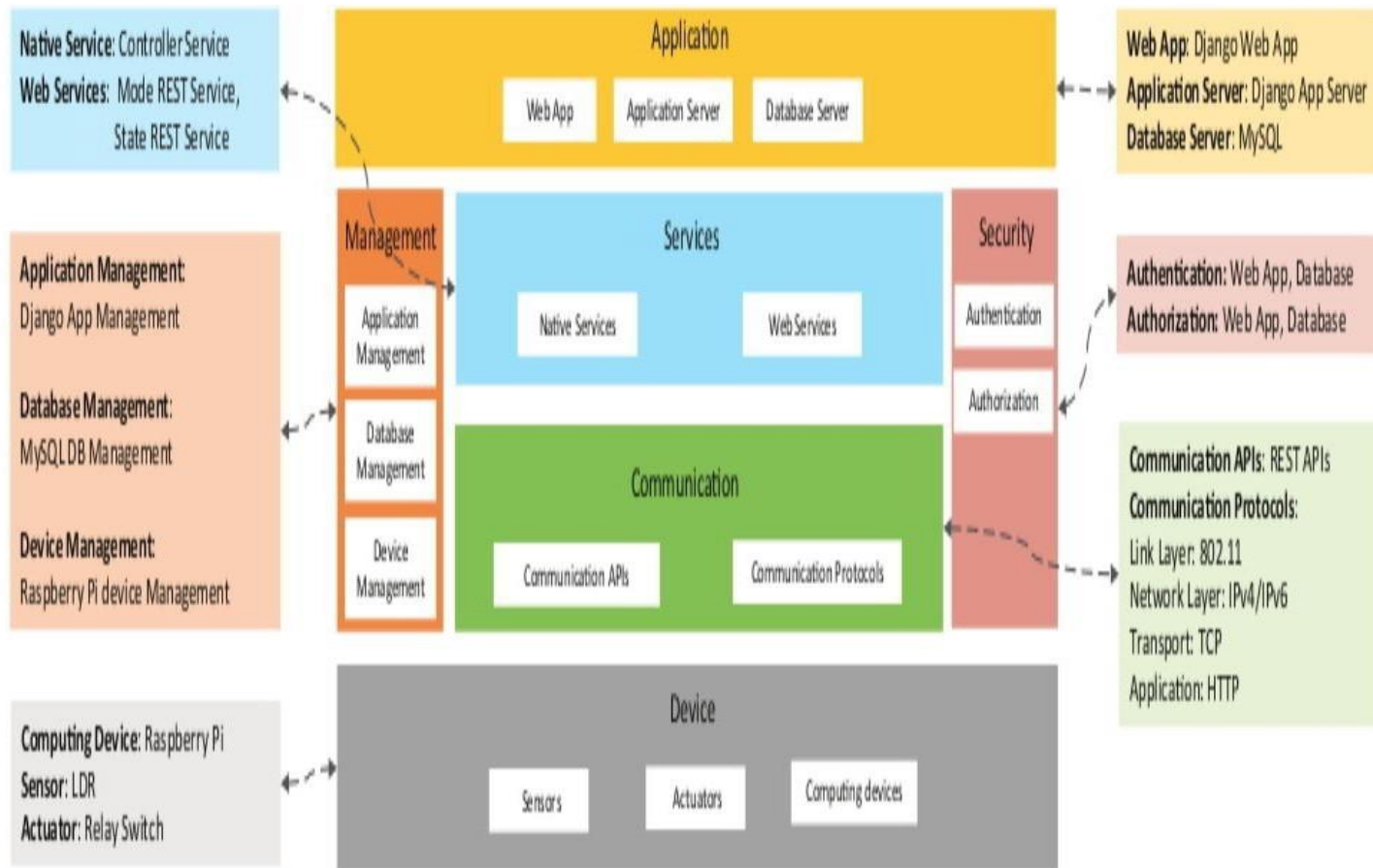
IoT System Design : Home Automation

Step : 7 – Functional View Specification



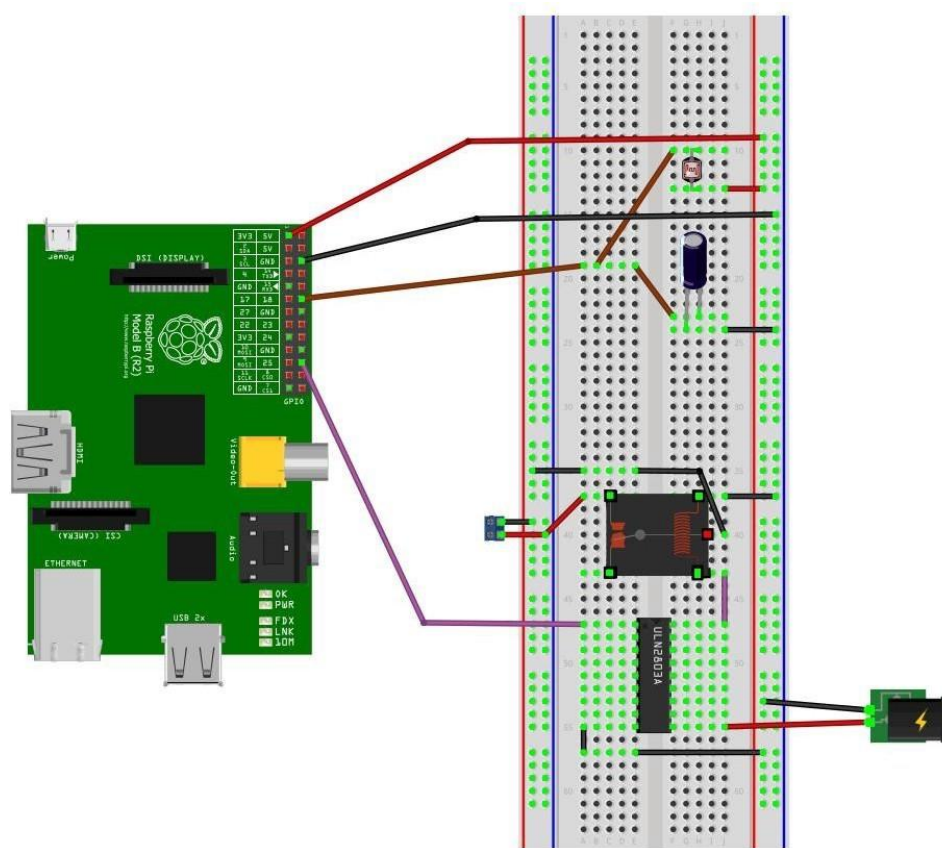
IoT System Design : Home Automation

Step : 8 – Operational View Specification



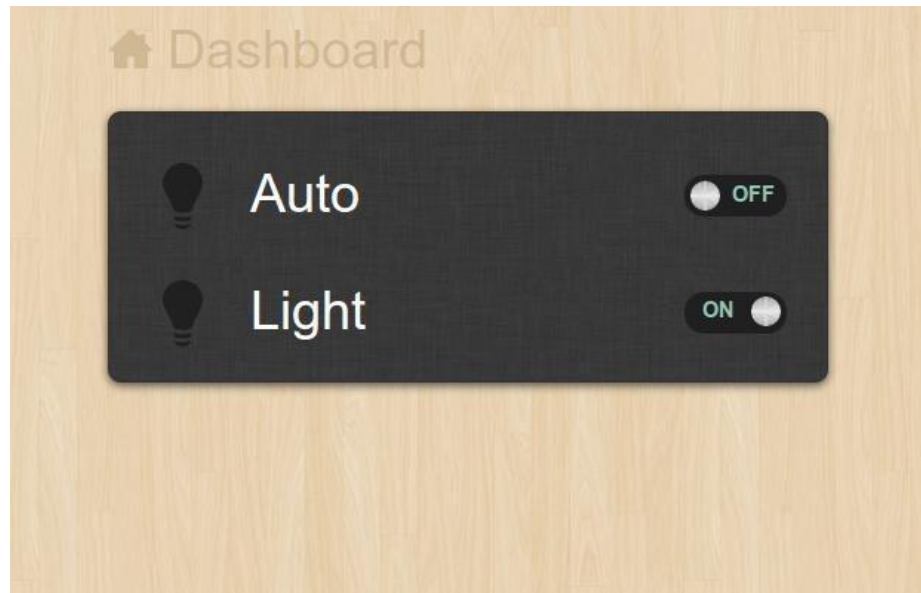
IoT System Design : Home Automation

Step : 9 – Device and Component Integration



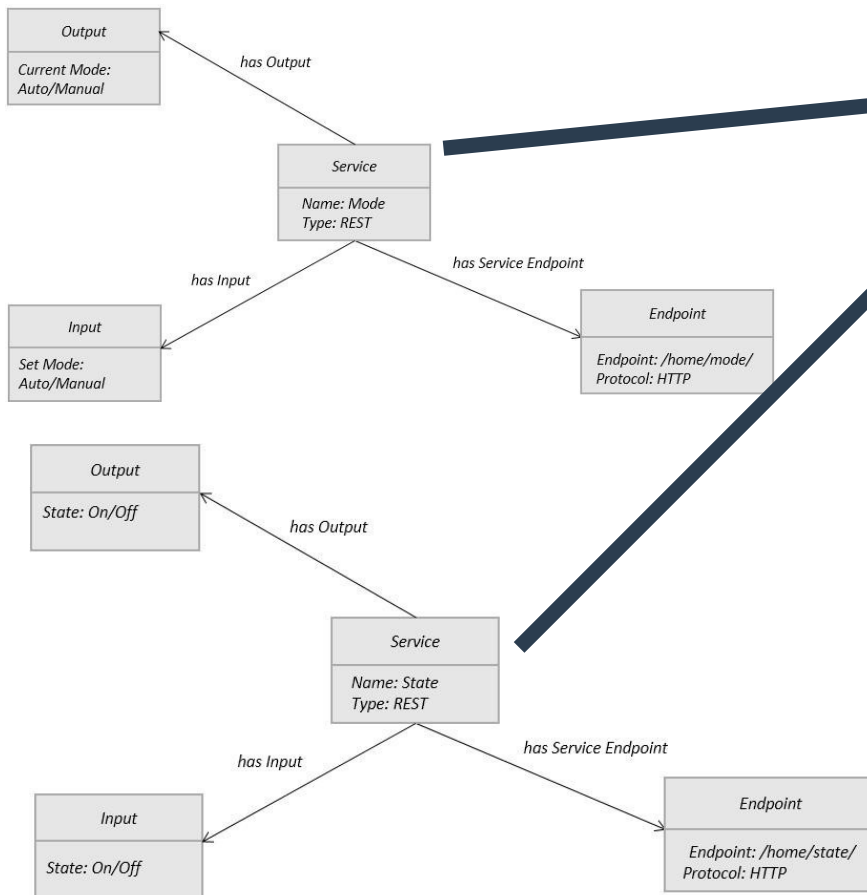
IoT System Design : Home Automation

Step : 10 – Application Development (Interface)



IoT System Design : Home Automation

Step : 11 – Web Services Development

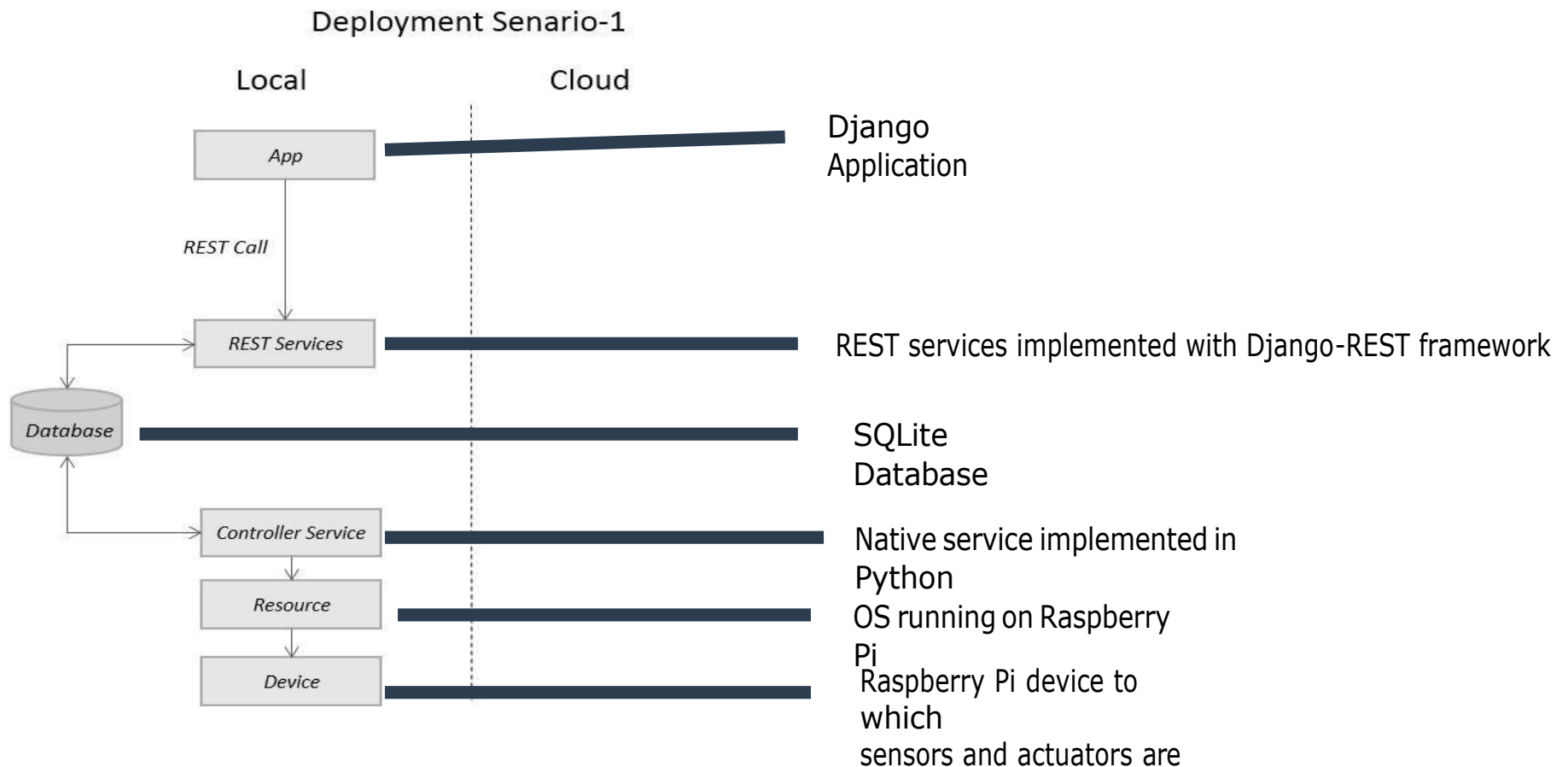


1. Map services to models. Model fields store the states (on/off, auto/manual)

2. Write Model serializers. Serializers allow complex data (such as model instances) to be converted to native Python datatypes that can then be easily rendered into JSON, XML or other content types.

IoT System Design : Home Automation

Step : 12 – Integrating the System



Source: Book website: <http://www.connected-net-of-things-book.com>