IoTSuite: A Toolkit for Prototyping Internet of Things Applications

IoTSuite CoreTeam
Aim of IoTSuite

Make IoT application development easy, given IoT applications exhibit heterogeneity
Smart Home Application
SmartHome Application

Resident

- Safety
SmartHome Application

Resident
• Safety

- Fire Notifier
- Calculate AvgTemp
- Temperature Sensor
- Smoke Sensor

EndUserApp
SmartHome Application

Resident
• Safety
SmartHome Application

- Fire Notifier
- Calculate AvgTemp
- Temperature Sensor
- Smoke Sensor
- Alarm
- EndUserApp

Resident
- Safety
- Comfort
SmartHome Application

- Resident
  - Safety
  - Comfort

- Calculate AvgTemp
- Fire Notifier
- EndUserApp
- Alarm
- DataBase Server
- Temperature Sensor
- Smoke Sensor
SmartHome Application

- Resident
  - Safety
  - Comfort
SmartHome Application

- Resident
  - Safety
  - Comfort

Diagram:
- Heater
- Temperature Regulator
- Proximity
- EndUserApp
- Fire Notifier
- Calculate AvgTemp
- Temperature Sensor
- Smoke Sensor
- RFID BadgeReader
- Database Server
SmartHome Application

- Resident
  - Safety
  - Comfort
  - Awareness

Diagram:
- Heater
- Temperature Regulator
- Proximity
- RFID
- Badge Reader
- DataBase Server
- Temperature Sensor
- Smoke Sensor
- Calculate AvgTemp
- Fire Notifier
- Alarm
- EndUserApp
SmartHome Application Development using IoTSuite

IoTSuite CoreTeam
Application Development using IoTSuite - Steps

- **Step 1**: Creating an IoTSuite Project
- **Step 2**: Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3**: Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4**: User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5**: Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6**: Implementing User Interface
Application Development using IoTSuite - Steps

- **Step 1: Creating an IoTSuite Project**
- **Step 2: Domain specification**
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3: Functional specification**
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4: User Interaction Specification**
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5: Deployment Specification**
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6: Implementing User Interface**
Application Development using IoTSuite - Steps

- **Step 1: Creating an IoTSuite Project**
- **Step 2: Domain specification**
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3: Functional specification**
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4: User Interaction Specification**
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5: Deployment Specification**
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6: Implementing User Interface**
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6:** Implementing User Interface
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6:** Implementing User Interface

This step involves the specification of concepts that are responsible for interacting with Entities of Interests (EoI).

- **Sensors** (Sense the EoI)
SmartHome Application: Sensors

- Heater
- Temperature Regulator
- Proximity
- RFID Badge Reader
- Database Server
- RFID
- Data Visualizer
- DashBoard
- EndUserApp
- Fire Notifier
- Calculate AvgTemp
- Temperature Sensor
- Smoke Sensor
- Humidity Sensor
- Yahoo Weather Service
- SmartHome Application: Sensors
Application Development using IoTSuite - Steps

- **Step 1**: Creating an IoTSuite Project
- **Step 2**: Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3**: Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4**: User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5**: Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6**: Implementing User Interface

This step involves the specification of concepts that are responsible for interacting with Entities of Interests (EoI).

- Sensors (Sense the EoI)
- Actuators (Affect the EoI)
SmartHome Application: Actuators

- Heater
- Alarm
- EndUserApp
- DashBoard
- Temperature Regulator
- Fire Notifier
- Calculate AvgTemp
- Proximity

Related Devices:
- RFID BadgeReader
- Database Server
- Temperature Sensor
- Smoke Sensor
- Humidity Sensor
- Yahoo Weather Service
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6:** Implementing User Interface

This step involves the specification of concepts that are responsible for interacting with Entities of Interests (EoI).

- Sensors (Sense the EoI)
- Actuators (Affect the EoI)
- **Storage** (Store information about the EoI)
SmartHome Application: Storage

- Heater
- Temperature Regulator
- Proximity
- RFID BadgeReader
- Database Server
- Temperature Sensor
- Smoke Sensor
- Humidity Sensor
- Fire Notifier
- Alarm
- Calculate AvgTemp
- EndUserApp
- DashBoard
- Data Visualizer
- Yahoo Weather Service
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6:** Implementing User Interface

This step involves the specification of concepts that are responsible for interacting with Entities of Interests (EoI).

- Sensors (Sense the EoI)
- Actuators (Affect the EoI)
- Storage (Store information about the EoI)
- Tags (Identify the EoI)
SmartHome Application: Tags

- Heater
- Temperature Regulator
- Proximity
- RFID BadgeReader
- Database Server
- Temperature Sensor
- Smoke Sensor
- Humidity Sensor
- Fire Notifier
- Alarm
- EndUserApp
- Data Visualizer
- Yahoo Weather Service
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
  - **Step 2:** Domain specification
    - Writing `vocab.mydsl` file
    - Compiling `vocab.mydsl` file
  - **Step 3:** Functional specification
    - Writing `arch.mydsl` file
    - Compiling `arch.mydsl` file
    - Implementing application logic
  - **Step 4:** User Interaction Specification
    - Writing `userinteraction.mydsl` file
    - Compiling `userinteraction.mydsl` file
  - **Step 5:** Deployment Specification
    - Writing `deploy.mydsl` file
    - Compiling `deploy.mydsl` file
  - **Step 6:** Implementing User Interface

This step involves the specification of concepts that are responsible for interacting with Entities of Interests (EoI).

- **Sensors** (Sense the EoI)
- **Actuators** (Affect the EoI)
- **Storage** (Store information about the EoI)
- **Tags** (Identify the EoI)

**Specifying the resources using IoTSuite**
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6:** Implementing User Interface
Application Development using IoTSuite - Steps

- **Step 1: Creating an IoTSuite Project**
- **Step 2: Domain specification**
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3: Functional specification**
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4: User Interaction Specification**
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5: Deployment Specification**
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6: Implementing User Interface**
Application Development using IoTSuite - Steps

- **Step 1**: Creating an IoTSuite Project

- **Step 2**: Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file

- **Step 3**: Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic

- **Step 4**: User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file

- **Step 5**: Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file

- **Step 6**: Implementing User Interface

This step involves the specification of concepts that are responsible for computation or processing.

- Computational Service
SmartHome Application: Computational

- Heater
- Temperature Regulator
- Proximity
- RFID Badge Reader
- Database Server
- Temperature Sensor
- Smoke Sensor
- Humidity Sensor
- Fire Notifier
- Calculate AvgTemp
- Data Visualizer
- EndUserApp
- DashBoard
- Yahoo Weather Service

Diagram shows the integration of various sensors, regulators, and services to create a comprehensive smart home application.
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project

- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file

- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic

- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file

- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file

- **Step 6:** Implementing User Interface

This step involves the specification of concepts that are responsible for computation or processing.

- Computational Service

Specifying computational Services using IoTSuite
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6:** Implementing User Interface
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6:** Implementing User Interface
Application Development using IoTSuite - Steps

- **Step 1**: Creating an IoTSuite Project
- **Step 2**: Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3**: Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4**: User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5**: Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6**: Implementing User Interface
Application Development using IoTSuite - Steps

- **Step 1**: Creating an IoTSuite Project
- **Step 2**: Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3**: Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4**: User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5**: Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6**: Implementing User Interface

This step involves the specification of concepts that denotes information exchange between an application and a user. We define the following abstractions:

- **Command**: denotes the information flow from a user to an application.
- **Notify**: denotes the information flow from an application to a user.
- **Request**: denotes the information flow round-trip between an application and a user.
SmartHome Application: Computational

- Heater
- Temperature Regulator
- Proximity
- RFID BadgeReader
- DataBase Server
- Temperature Sensor
- Smoke Sensor
- Humidity Sensor
- Fire Notifier
- Calculate AvgTemp
- Alarm
- EndUserApp
- DashBoard
- Data Visualizer
- Yahoo Weather Service
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6:** Implementing User Interface
Application Development using IoTSuite - Steps

- **Step 1: Creating an IoTSuite Project**
- **Step 2: Domain specification**
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3: Functional specification**
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4: User Interaction Specification**
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5: Deployment Specification**
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6: Implementing User Interface**
Application Development using IoTSuite - Steps

- **Step 1:** Creating an IoTSuite Project
- **Step 2:** Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3:** Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4:** User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5:** Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6:** Implementing User Interface

This step involves the specification of devices and its properties in a target deployment.
Application Development using IoTSuite - Steps

- Step 1: Creating an IoTSuite Project
- Step 2: Domain specification
  - Writing vocab.mydsl file
  - Compiling vocab.mydsl file
- Step 3: Functional specification
  - Writing arch.mydsl file
  - Compiling arch.mydsl file
  - Implementing application logic
- Step 4: User Interaction Specification
  - Writing userinteraction.mydsl file
  - Compiling userinteraction.mydsl file
- Step 5: Deployment Specification
  - Writing deploy.mydsl file
  - Compiling deploy.mydsl file
- Step 6: Implementing User Interface
Application Development using IoTSuite - Steps

- Step 1: Creating an IoTSuite Project
- Step 2: Domain specification
  - Writing vocab.mydsl file
  - Compiling vocab.mydsl file
- Step 3: Functional specification
  - Writing arch.mydsl file
  - Compiling arch.mydsl file
  - Implementing application logic
- Step 4: User Interaction Specification
  - Writing userinteraction.mydsl file
  - Compiling userinteraction.mydsl file
- Step 5: Deployment Specification
  - Writing deploy.mydsl file
  - Compiling deploy.mydsl file
- Step 6: Implementing User Interface
Backup slides
Aim and Objectives

**Aim:**
- Make IoT application development easy, given IoT applications exhibit heterogeneity

**Objectives:**
- Integrate high-level Domain-Specific Languages (DSLs) to specify IoT applications
  - To hide development complexity
- Provide automation techniques
  - To parse the DSLs
  - To reduce application development efforts at different life-cycle phases
Application Development using IoTSuite - Steps

- **Step 1: Creating an IoTSuite Project**
  - **Step 2: Domain specification**
    - Writing `vocab.mydsl` file
    - Compiling `vocab.mydsl` file
  - **Step 3: Functional specification**
    - Writing `arch.mydsl` file
    - Compiling `arch.mydsl` file
    - Implementing application logic
  - **Step 4: User Interaction Specification**
    - Writing `userinteraction.mydsl` file
    - Compiling `userinteraction.mydsl` file
  - **Step 5: Deployment Specification**
    - Writing `deploy.mydsl` file
    - Compiling `deploy.mydsl` file
  - **Step 6: Implementing User Interface**

This step involves the specification of resources that are responsible for interacting with Entities of Interests (EoI).

- **Sensors** (Sense the EoI)
- **Actuators** (Affect the EoI)
- **Storage** (Store the information about the EoI)
- **Tags** (Identify the EoI)
- **Third-party service** (Provide information about EoI)
Application Development using IoTSuite - Steps

- Step 1: Creating an IoTSuite Project
- Step 2: Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- Step 3: Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- Step 4: User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- Step 5: Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- Step 6: Implementing User Interface

This step involves the specification of resources that are responsible for interacting with Entities of Interests (EoI).

- Sensors (Sense the EoI)
- Actuators (Affect the EoI)
- Storage (Store information about the EoI)
- Tags (Identify the EoI)
- Third-party service (Provide information about EoI)
SmartHome Application: Third-party services

- Heater
- Temperature Regulator
- Proximity
- RFID BadgeReader
- DataBase Server
- Temperature Sensor
- Smoke Sensor
- Humidity Sensor
- Yahoo Weather Service
- Fire Notifier
- DashBoard
- Data Visualizer
- EndUserApp
- Calculate AvgTemp

Third-party services:
- DataBase Server
- Yahoo Weather Service
- Temperature Sensor
- Smoke Sensor
- Humidity Sensor
- RFID BadgeReader
- Heater
- Temperature Regulator
- Proximity
- Fire Notifier
- DashBoard
- Data Visualizer
- EndUserApp
- Calculate AvgTemp
Steps

1. Creating an IoTSuite Project
2. Domain specification
   - Writing `vocab.mydsl` file
   - Compiling `vocab.mydsl` file
3. Functional specification
   - Writing `arch.mydsl` file
   - Compiling `arch.mydsl` file
   - Implementing application logic
4. User Interaction Specification
   - Writing `userinteraction.mydsl` file
   - Compiling `userinteraction.mydsl` file
5. Deployment Specification
   - Writing `deploy.mydsl` file
   - Compiling `deploy.mydsl` file
6. Implementing User Interface

This step involves the specification of resources that are responsible for interacting with Entities of Interests (EoI).

- Sensors (Sense the EoI)
- Actuators (Affect the EoI)
- Storage (Store information about the EoI)
- Tags (Identify the EoI)
- Third-party service (Provide information about EoI)
Application Development using IoTSuite - Steps

- **Step 1**: Creating an IoTSuite Project
- **Step 2**: Domain specification
  - Writing `vocab.mydsl` file
  - Compiling `vocab.mydsl` file
- **Step 3**: Functional specification
  - Writing `arch.mydsl` file
  - Compiling `arch.mydsl` file
  - Implementing application logic
- **Step 4**: User Interaction Specification
  - Writing `userinteraction.mydsl` file
  - Compiling `userinteraction.mydsl` file
- **Step 5**: Deployment Specification
  - Writing `deploy.mydsl` file
  - Compiling `deploy.mydsl` file
- **Step 6**: Implementing User Interface