# CHAPTER 5

**REQUIRMENT ANALYSIS**

* **INTRODUCTION REQUIRMENT DESIGN** :

Introduces the systems development life cycle (SDLC), the fundamental four-phase model (planning, analysis, design, and implementation) common to all information systems development projects.

It describes the evolution of system development methodologies and discusses the roles and skills required of a systems analyst.

The chapter then overviews the basic characteristics of object-oriented systems and the fundamentals of object-oriented systems analysis and design and closes with a description of the Unified Process and its extensions and the Unified Modeling Language.

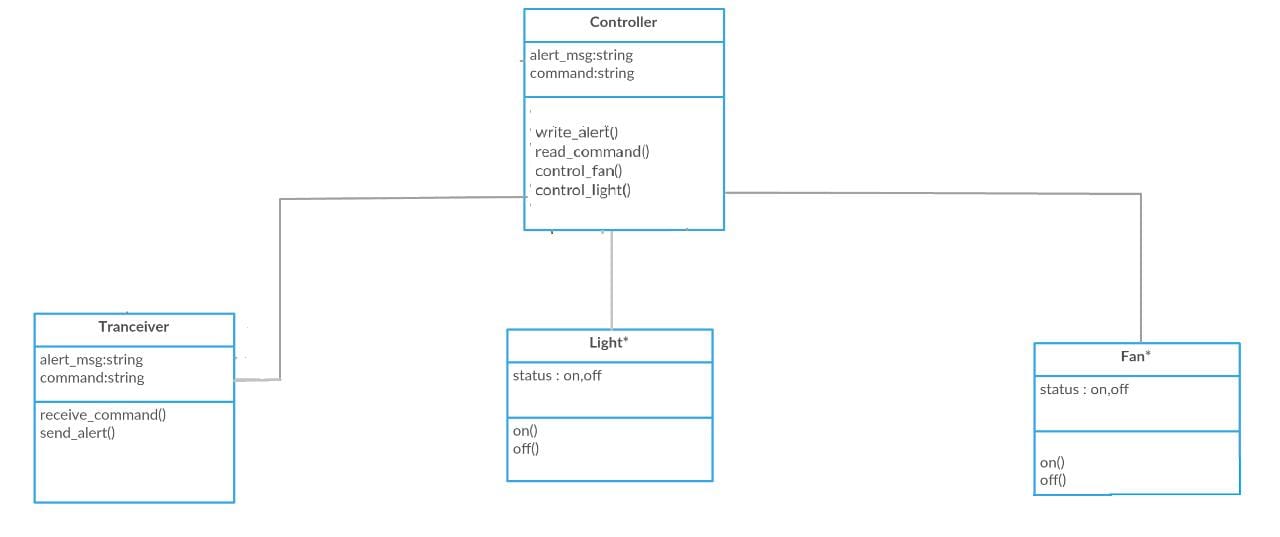
The main goal of systems analysis and design is to improve organizational systems, typically through applying software that can help employees accomplish key business tasks more easily and efficiently. As a systems analyst, you will be at the center of developing this software.

**# CLASS DIAGRAM:**

The [class diagram](https://t4tutorials.com/class-diagram-and-domain-model-of-store-management-system-project/) is a model for creating the classes, their attributes, and functions. In [software engineering](http://en.wikipedia.org/wiki/Software_engineering), a class diagram in the [Unified Modeling Language](http://en.wikipedia.org/wiki/Unified_Modeling_Language) (UML) is a type of static structure diagram that describes the structure of a system by showing the system's [classes](http://en.wikipedia.org/wiki/Class_%28computer_science%29), their attributes, operations (or methods), and the relationships among objects. The class diagram is the main building block of [object oriented](http://en.wikipedia.org/wiki/Object_oriented) modelling. It is used both for general [conceptual modelling](http://en.wikipedia.org/wiki/Conceptual_model) of the systematic of the application, and for detailed modelling translating the models into [programming code](http://en.wikipedia.org/wiki/Programming_code). Class diagrams can also be used for [data modelling](http://en.wikipedia.org/wiki/Data_modeling). The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed.

* **Purpose Of Class Diagram:**
* In software engineering, a **class diagram** in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

­­­­



**Fig 1 .Class Diagram For Home Automation System**

In the above figure we have the class diagram which has controller ,tranceiver ,light and fan. The controller has the attribute of alert\_msg which is in the from string and command.It has functions to write\_alert() messages to read\_command() and to control appliances of fan and light control \_fan(),control\_light(). The tranceiver has alert\_msg and command as attribute , If has function to receiver\_command and send\_alert. The light and fan has the status attribute which shows the status of appliances whether it is on or off .i.e. status :on,off and the function on() and off().

**# DATA FLOW DIAGRAM :**

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an [information system](http://en.wikipedia.org/wiki/Information_system), modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the [visualization](http://en.wikipedia.org/wiki/Data_visualization) of [data processing](http://en.wikipedia.org/wiki/Data_processing) (structured design). A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a [flowchart](http://en.wikipedia.org/wiki/Flowchart)).

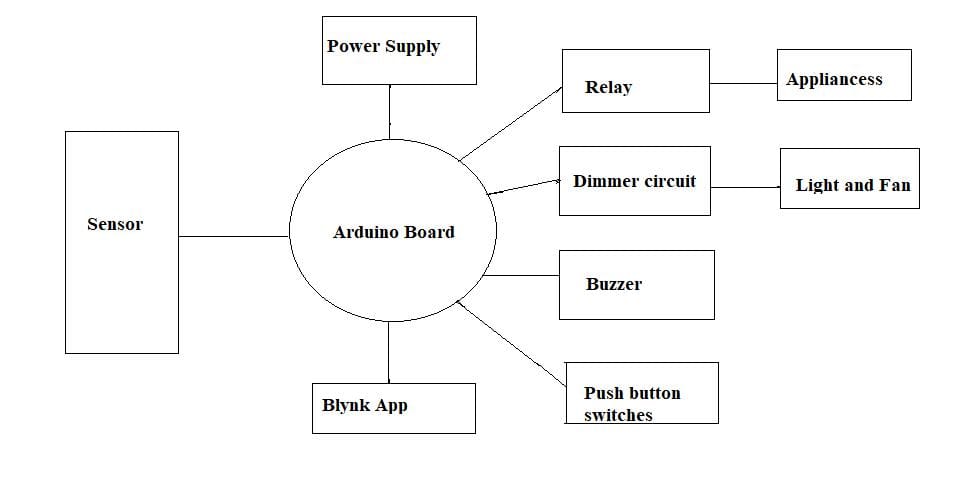
A data flow diagram (DFD) is a graphical representation of the "flow" of data through an [information system](http://en.wikipedia.org/wiki/Information_system), modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the [visualization](http://en.wikipedia.org/wiki/Data_visualization) of [data processing](http://en.wikipedia.org/wiki/Data_processing) (structured design).

Data flow diagrams only involve four symbols. They are:

* Process
* Data Object
* Data Store
* External entity
* **Purpose OF Data Flow Diagram:**
* Data flow diagrams provide a graphical representation of how information moves between processes in a system. Data flow diagrams

follow a hierarchy; that is, a diagram may consist of several layers, each unique to

specific process or data function.



## 

**Fig 2 . Data Flow Diagram For Home Automation System**

The data flow diagram consist of sensor which given as input to arduino Board .The arduino board is connected to the power supply and blynk app it has relay, dimmer circuit has and buzzer connected to it. The relay is connected to all the appliances and the dimmer circuit has the light and fan appliances connected to it. The push button switches is also connected to the arduino board which has esp-32 in it . the changes of arduino reflect in the blynk app which is connected to the cloud interface.

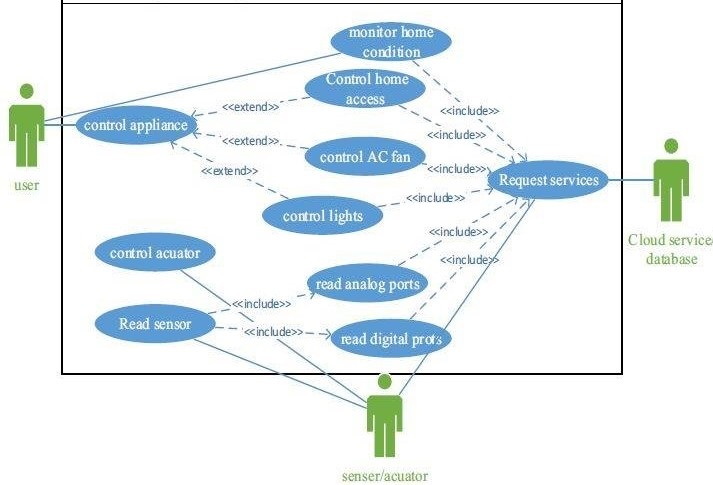
**#USE CASE DIAGRAM :**

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system.

To model a system the most important aspect is to capture the dynamic behaviour. To clarify a bit in details, dynamic behaviour means the behaviour of the system when it is running /operating. So only static behaviour is not sufficient to model a system rather dynamic behaviour is more important than static behaviour. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction.

The purpose of use case diagram is to capture the dynamic aspect of a system. But this definition is too generic to describe the purpose. Because other four diagrams (activity, sequence, collaboration and Statechart) are also having the same purpose.Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements.

* **Purpose Of Use Case Flow:**
* **Actors:** The users that interact with a system. An actor can be a person, an organization, or an outside system that interacts with your application or system. They must be external objects that produce or consume data.
* **System:** A specific sequence of actions and interactions between actors and the system. A system may also be referred to as a scenario.
* **Goals:** The end result of most use cases. A successful diagram should describe the activities and variants used to reach the goal.



**Fig 3. Use Case Diagram for Home Automation System**

The above figure is a use case diagram of the web server based home automation system.It has the entity as user ,sensor/actuator and cloud service database which is connected with the blynk app .The user can control all the appliances such as light,fan etc.The sensor has the control acuator read digital part and analog parts which includes read sensors in it .The cloud services includes all the request services of the appliances such as to control ac,fan and light . This all are connected with the blynk app and esp-32 node mcu which is connected.

**#SEQUENCE DIAGRAM:**

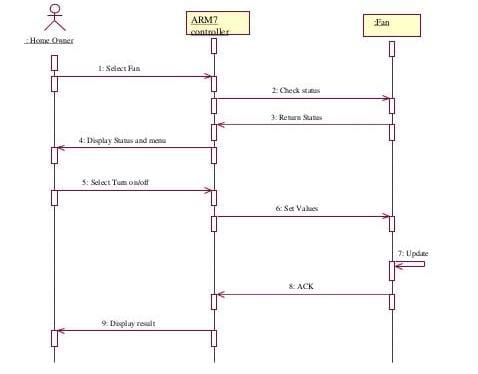
Sequence diagrams, commonly used by developers, model the interactions between objects in a single use case. They illustrate how the different parts of a system interact with each other to carry out a function, and the order in which the interactions occur when a particular use case is executed.

They capture the interaction between objects in the context of a collaboration. **Sequence Diagrams** are time focus and they show the order of the interaction visually by using the vertical axis of the diagram .

A sequence diagram is structured in such a way that it represents a timeline which begins at the top and descends gradually to mark the sequence of interactions. Each object has a column and the messages exchanged between them are represented by arrows.

* **Purpose Of Sequence Diagram :**

* The sequence diagram is used primarily to show the interactions between objects in the sequential order that those interactions occur. Much like the class diagram, developers typically think sequence diagrams were meant exclusively for them.
* One of the primary uses of sequence diagrams is in the transition from requirements expressed as use cases to the next and more formal level of refinement. Use cases are often refined into one or more sequence diagrams.



**Fig 4.Sequence Diagram For Home Automation System**

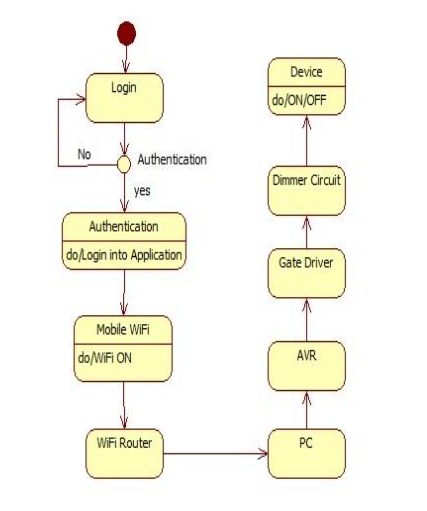
The above figure shows the sequence diagram of the home automation system .It has object as home owner controller and fan.The home owner can select the each object seperatly it select the fan then the status goes to the controller and it on the fan and also shows the status in the blynk app, then it can select turn on anf turn off option with the specific set values such as 1 for off and 0 for on.When the user select the particular status ON or OFF the token get generated and the status in the app get updated as it is connected to the cloud through internet and appliances works accordingly . The acknowledgment for all the appliances get back to the controller i.e. esp-32 node mcu and it display the result on the home owner app.

**#ACTIVITY DIAGRAM :**

An **activity diagram** visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow **diagram**. **Activity diagrams** are often used in business process modeling. They can also describe the steps in a use case **diagram**. **Activities** modeled can be sequential and concurrent. The [Unified Modeling Language](https://www.lucidchart.com/pages/what-is-UML-unified-modeling-language) includes several subsets of diagrams, including structure diagrams, interaction diagrams, and behavior diagrams.

Activity diagrams, along with [use case](https://www.lucidchart.com/pages/uml-use-case-diagram) and [state machine diagrams](https://www.lucidchart.com/pages/uml-state-machine-diagram), are considered behavior diagrams because they describe what must happen in the system being modeled.Activity diagrams help people on the business and development sides of an organization come together to understand the same process and behavior. You'll use a set of specialized symbols—including those used for starting, ending, merging, or receiving steps in the flow—to make an activity diagram, which we’ll cover in more depth within this activity diagram guide.

* **Purpose Of Activity Diagram :**
* **Activity diagrams** describe parallel and conditional **activities**, use cases and system functions at a detailed level. An **activity diagram** is used to model a large **activity's** sequential work flow by focusing on action sequences and respective action initiating conditions.
* An Activity Diagram is a behavioral diagram that shows the flow or sequence of activities through a system.  The terms activity diagram and process flow are often used interchangeably.  However, the term activity diagram is typically more restrictive as it refers to one of thirteen standard Unified Model Language (UML) diagrams.  Activity Diagrams are one of the most commonly used diagrams since its notation and origin are based on the widely known flowchart notation

****

**Fig 5. Activity Diagram For Home Automation System**

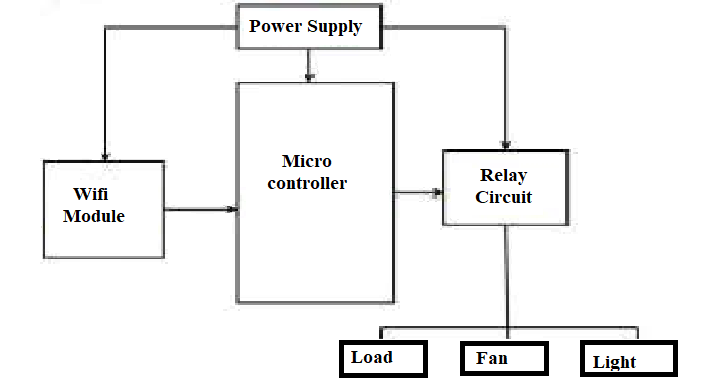
In the above figure has the activity diagram for home automation system.First get start and user has to login in the system it then get authenticated whether it is a valid user or not if it is valid then they have permission to use app and make the changes then it checks whether the phone is connected to the internet or not .it has wifi on then it get connected to pc and the AVR i.e ESP-32 the gate driver and the dimmer circuit which we have to control the appliances then the device get switch ON and OFF.

**#DEPLOYMENT DIAGRAM:**

A UML **deployment diagram** is a diagram that shows the configuration of run time processing nodes and the components that live on them. **Deployment diagrams** is a kind of structure diagram used in modeling the physical aspects of an object-oriented system. Deploymentdiagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. Deploymentdiagrams are typically used to visualize the physical hardware and software of a system.

**Deployment diagram** is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. **Deployment diagrams** are typically used to visualize the physical hardware and software of a system.

* **Purpose Of Deployment Diagram:**
* Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.
* Deploymentdiagrams are mainly used by system engineers. These diagrams are used to describe the physical components (hardware), their distribution, and association



**Fig 6.Deployment Diagram For Home Automation System**

The above figure shows the Deployement diagram based on home automation system. The deployment dosent have much component in it. It just has a arduino controller i.e micro controller circuit and which is connected to the wifi module power supply and relay circuit . The relay circuit has a appliances connected to it such as load , fan and light . It used to visualize the physical hardware of a software system.