**Smart Home Automation System**

Problem Statement: Design and implement an Object-Oriented Programming (OOP) based solution in Python to create a smart home automation system. The system should allow users to control and manage different appliances and devices in their home, such as lights, air conditioners, security systems, and entertainment systems. The solution should be scalable, modular, and support the addition of new devices or functionalities in the future.

Project Requirements:

1. Define classes and inheritance hierarchies for various devices and appliances in the smart home system. For example, you can have a base class 'Device' and derive specific classes like 'Light', 'AirConditioner', 'SecuritySystem', and 'EntertainmentSystem'.
2. Implement methods and properties for each class that allow the user to interact with and manipulate the devices. For instance, for the 'Light' class, you can have methods like 'turn\_on()', 'turn\_off()', and 'set\_brightness()' along with properties like 'status' and 'brightness'.
3. Create an interface (e.g., a command-line interface or a graphical user interface) that allows users to interact with the system, control devices, and view their status. You may want to consider using a popular Python GUI library, such as Tkinter or PyQt.
4. Implement a 'Room' class that represents a room in the house and contains a collection of devices. The 'Room' class should have methods to add and remove devices, as well as methods to control and manage the devices within the room.
5. Implement a 'House' class that represents the entire home and contains a collection of rooms. The 'House' class should have methods to add and remove rooms, as well as methods to control and manage the devices in all the rooms.
6. Design a mechanism to store and load the configuration of the smart home system, such as device settings and room arrangements, in a file. This will allow users to save their settings and load them when the program starts.
7. Write unit tests to ensure that the classes, methods, and properties are working as expected.
8. Provide thorough documentation on how to use the smart home automation system, including instructions for installing any necessary dependencies, running the program, and using the interface to control devices.

Stretch Goals:

1. Implement additional device classes, such as 'Thermostat', 'SmartLock', and 'SmartSpeaker', to expand the capabilities of the smart home system.
2. Integrate voice control functionality, utilizing a speech-to-text library to allow users to control their devices using voice commands.
3. Develop a mobile application or web interface that enables users to control their smart home system remotely.