**Assumption**

* ~~A1 : No concurrent user accesses. e.g.~~ 
  1. ~~Access through a control panel and access through a web page are exclusive; both cannot happen at the same time.~~
  2. ~~Only one control panel is installed in a house~~
* ~~A2 : Internet connection between a homeowner (i.e., homeowner’s smartphone or notebook) and the SafeHome system is always available.~~
* ~~A3 : All devices including cameras, sensors, and the SafeHome main system communicate using IEEE 802.11x protocol.~~

**Functional Requirements**

1. SafeHome bootup/shutdown service
   1. ~~Self-diagnosis – check whether connected sensors and cameras are ok.~~
   2. Initialize/finish sensors
   3. Initialize/finish cameras
2. Configuration service
   1. Password
      1. Type of password
         1. Control panel - digits
         2. ~~Web access~~ Java application GUI – alphabets and digits
      2. Length of password
         1. Control panel – 4 digits
         2. ~~Web access~~ Java application GUI – at least 8 characters
   2. ~~Floor plan~~
      1. ~~Floor structure~~
         1. ~~Distance scale~~
         2. ~~Wall~~
         3. ~~Door~~
      2. ~~Registering new devices~~
      3. ~~Relocating devices~~
   3. Security zone – a set of devices (e.g., sensors and cameras) which are manipulated as a group
   4. ~~Backup plan for power off – Which devices could be turned off? Which devices should be turned on?~~
   5. ~~Session timeout – See~~ [~~5.A Athentication.session~~](#_Hlk418156511)~~.~~
3. Real-time security service
   1. Arming the entire SafeHome system
   2. Disarming the entire SafeHome system
   3. Arming device(s)
   4. Disarming device(s)
   5. Alarming methods – Sound alarm, SMS, smartphone push alarm, and so on.
   6. ~~Enabling travel mode – Turning lights on/off~~
   7. Sensor
      1. Types
         1. Motion detector
         2. Window sensor
         3. ~~CO2 sensor~~
         4. ~~Fire sensor~~
         5. ~~Doggie angst sensor~~
      2. Output data format – ~~binary, real number, or complex data?~~ (API is provided.)
   8. Camera
      1. ~~Recording~~ 
         1. ~~Format of movie – H264, MPEG, and so on~~
         2. ~~Quality of movie – minimum required FPS issue~~
      2. Viewing
         1. A single camera
         2. All cameras through thumbnail view
      3. Controlling
         1. Zooming
         2. Panning
      4. ~~Movie storage~~
4. ~~Us~~~~er-requested information retrieval service~~
   1. ~~Finding ID/Password~~
   2. ~~Reporting system usage pattern~~
   3. ~~Reporting web page access history~~
   4. ~~Replaying recorded movie~~
5. System status reporting service
   1. Current status e.g. System is idle, armed, turned off, and so on.
6. Authentication
   1. ~~Session~~
      1. ~~Closing a session after certain timeout; See~~ [~~2.E Session timeout~~](#_Hlk418156781)~~.~~
   2. Control panel
      1. Input control panel password e.g. 4-digits password
      2. ~~Access permission is granted to only one session. See~~ [~~Assumption A1~~](#_Hlk418156474)
   3. ~~Webpage~~
      1. Input ID and password(s) e.g. ID and two levels of passwords
      2. ~~Access permission is granted to only one session. See~~ [~~Assumption A1~~](#_Hlk418156474)
7. User interface
   1. Control panel
      1. Input device – Keypad (12 keys)
      2. Output device
         1. LED notification (arm on/off and power on/off status)
         2. LCD display description
   2. ~~Web page~~ Java application GUI
      1. Configuration features. See [2. Configuration service](#_Hlk418157040).
      2. Real-time security features; See [3. Real-time security service](#_Hlk418157017).
      3. ~~Reporting service features; See~~ [~~4. User-requested reporting service~~](#_Hlk418156992)~~.~~
      4. ~~Web page should be designed mobile-friendly.~~

**~~Nonfunctional Requirements~~**

* ~~SafeHome system should respond to sensor events in real-time.~~