

Software Systems Architecture

MIEIC – 4th Year, 2nd Term

Exam – 4 May 2017 – With Consultation – Duration 90 minutes

Consider that you are the main **Architect** for a system that aims to transform a normal family house into a **Smart House**. For this purpose, your company has an integrated solution comprised of both **Software** and **Hardware**. A full pack includes sensors in every window and door, capable of detecting if it is open, **close**, or vibrating; temperature, humidity, **lux**, sound and air quality in every room; modified switches and power plugs; RGB led strips for ambient lighting in every room; emergency sirens; etc. On average, one should expect between 60 – 80 devices per house. There are two requirements considered paramount: (a) should one or several devices fail, all others should continue working normally; (b) everything should be controlled/monitored remotely via a secure connection over the **Internet**. Other requirements include plug and play (**discoverability**), 3rd party **integration**, **elasticity**, and **visual** configuration. You are the Software Architect, so you can't modify Hardware constraints; you can assume every device is independently powered and is **WiFi** connected, but not all devices are capable enough to run a full-blown operating system, and instead rely on firmware images, either proprietary, or written by you using the appropriate toolchains.

For each one of the following **eight** questions, select the correct answer(s), always justifying it/them. A correct answer without a suitable justification receives zero marks. A wrong answer does not receive negative marks.

1. Consider the concerns and goals of Software Architecture. Would you say that the primary concern when "establishing the architecture" of a given system is:
 - a. To establish the common paradigms and high-level relations of — and among — its components;
 - ~~b. To enumerate the system's **functional** requirements — by the architect — in a way that the developers can further implement it;~~
 - ~~c. To dictate the **programming** languages to be used — e.g. Java — and the technological infrastructure — e.g. Microsoft SQLServer;~~
 - ~~d. To organically structure the **team** and the appropriate development **methodology** for the given system.~~
2. Considering the above information, which of the following architectural styles deserve further study in relationship with this project:
 - a. Interpreter;
 - b. Event Systems;
 - c. Repository;
 - d. Call and Return.

3. Someone suggested the rules of the system (e.g., immediately after the front door opens, if no motion is detected for 1h, then activate the alarm), to be implemented following a Pipes and Filter architectural pattern. Would you say this is a good idea?
 - a. Yes;
 - b. No.

4. Consider the "Decorator" pattern as described by Gamma et al. (commonly known as GoF patterns). In the HoT context, would you say the Decorator might be a good solution:
 - a. To decouple the notion of actions and ~~states~~;
 - b. To support ~~Undo/ReDo~~ of operations;
 - c. To facilitate the development of bridges that interoperate with third 3rd party services;
 - d. To provide the notion of device capabilities (e.g. on/off, delayed action, etc.).

5. Consider the "Command" pattern as described by Gamma et al. (commonly known as GoF patterns). In the HoT context, would you say the Command might be a good solution:
 - a. To decouple the notion of actions and ~~states~~;
 - b. To support Undo/ReDo of operations;
 - c. To facilitate the development of ~~bridges~~ that interoperate with third 3rd party services;
 - d. To provide the notion of device capabilities (e.g. on/off, delayed action, etc.).

6. Consider Fault-Tolerance, discussed during the classes as part of the self-healing mechanisms, and enumerated as the first paramount requirement in the context. Which of the following Internet of Things patterns could help address such concern?
 - a. Heartbeat;
 - b. ~~Service Locator~~;
 - c. ~~Protocol Bridge~~;
 - d. ~~Broker~~.

7. Someone suggested the whole system could follow an Event-Driven architecture, with devices publishing messages and subscribing to topics in an Event Bus. You immediately recognize this has several advantages and disadvantages depending on how relevant certain requirements are. Pick **one** of the following concerns, and provide your opinion on the impact of using an Event-Driven architecture for:
 - a. Fault Tolerance;
 - b. Heterogeneity;
 - c. Elasticity;
 - d. Security.

