

Choose the most appropriate architectural pattern (one) for the 5 descriptions below. Motivate for your choice in one sentence (give reasons for choosing the pattern):

• Layered • Client-Server • Service-Oriented • Multi-tier • Serverless • Event-Driven • Microservices

- a) **Need software to encode text files. The software consists of several algorithms of encryption, which can be combined for better result.**

⇒ *Pipe and filter.*

- *The input file to be encoded will be moved through the pipe and be adapted and worked on until modification and transformation is done and output file is created.*

- b) **Need software for a digital speedometer for boats. The software should be flexible to support various number and types of displays (from simple LED-displays to large screens), and various ways of graphical designs of showing speed on the display.**

- c) **Need software for a smartphone game where players can compete against each other through Bluetooth (no need to access the Internet).**

- d) **Need software for managing and providing different kinds of information (weather, location, points-of-interest etc.) to app-and web-developers. The information will be gathered from various sources and processed by through the software. The software should provide open data access to anyone who wants to use the information through standard APIs.**

⇒ *Publish-Subscribe (Broker model), One client many servers.*

- *Broker acts as middle man between the servers different that will input weather and so on. The broker will then relay this to the client which has the Open API connection.*

- e) **Need software to very efficiently find the city in the world with a population above one million with the highest measured temperature. The temperatures measured from each city for 100 years are stored in separate files containing 36500 temperature entries. There are in total 457 cities with a population above 1 million.**

- *Map reduce that handles big data very well by splitting the data to different processes to faster find the right input.*

- f) **A system for video surveillance: in order to secure a particular building and its surroundings (e.g. military, key-infrastructure, country-border), a collection of video cameras is installed across the area to be monitored. In addition, there is a central control room where images from each of these cameras can be viewed – in near real-time**

⇒ *Publish-Subscribe.*

- *The cameras will stream continuous input to that will then be handled and stored in databases. will be easy to add more cameras..*

g) A system for tracking sports matches: Enthusiasts of sports would like to track the matches in which their favourite players (e.g. in tennis) or teams (e.g. in ice hockey) plays. To this end, a system allows a user to mark a sports-match. The system will then send updates about significant events to the user.

⇒ *Publish-Subscribe.*

- *Users are subscribes to what they are interested in as players etc and the system will act as publishers in that it will notify the users about the relevant info the user has subscribed to.*

h) Software for an electron-microscope. The driver for the architecture of this system is the continuous (real-time) stream of images that goes through a regular process of several subsequent image enhancement steps. There is little user-interaction with this system other than panning and zooming.

⇒ *Pipe and filter.*

- *The image will be input into the "pipe" and then be transformed through different processes such as img enhancement mentioned. After going through the pipe and filters it will be outputted as the final image.*

i) Supermarket loyalty system: customers of a (chain of) supermarket can register (make an account) to become a 'loyal customer' of the supermarket chain. With this card, the customer can collect points in return for buying groceries in the supermarket. Via a website, the customer can keep track of the points he has collected at any point in time. This system needs to be able to scale to a large number of customers. The frequency of interactions of users with the system is relatively low. Customers should not be able to see the points of other customers.

⇒ *Client server.*

- *The customers will be considered clients and they will generate info that will then be connected to the supermarkets server that will store and handle the information.*
- *This server will be used with the website to let the customers see how many points they have.*
- *Client server is a good fit because of the reasons above and how it scalable to new customers with high security.*

Khalid Shawki