

# **Service Implementation**

## **Background**

As business needs raise at different level and different timings, product/development teams will be established in silos to deliver the results. In these situations where some common functionalities could have been developed by many teams. In other words, the effort to deliver functionalities could have been duplicated. The pros and cons of developing common vs customized platform for common functionalities can be discussed below.

### **Proprietary or Custom approach:**

The development process derived by the product requirements and this would be continued in multiple sprints. The requirements would be identified in each sprint phase then goals will be achieved. This promotes the independence of each sprint goals and overall deliverables agreed upon project initiation or enhancement phases. There won't be any interdependency with other products/projects. However the ideas are limited to team needs and not looked at benefits achieved by other teams implemented the similar functionality.

### **Common approach:**

The development process initiated by teams coming together as single team to define requirements as deliverable product. In this process, each team contributes their ideas and all these ideas were put for discussion. The outcome of all parties agreement will be a common approved design which will be developed, monitored, tailored and delivered to all parties.

### **Conclusion:**

The above approaches outline how a functionality development works where ideas are strictly limited to single team vs multiple teams. Given the benefits provided by common functionality development, new ideas and latest technologies can be pulled into final deliverable product. Definitely there is a dependency will be injected which forces the delivery schedule of each individual team. However, benefits outweighs the schedule. Based on the benefits provided the common functionality approach is recommended.