Introducing the Case Study: Pizza Delivered Quickly (PDQ)

Pizza Delivered Quickly (PDQ) is a local chain (40 stores) of eat-in and home delivery pizza stores. Recently PDQ has lost 30 percent of sales revenue due mostly to a drop in their home delivery business. They attribute this solely to their major competitor who recently promoted a program that guarantees 45-minute delivery service from order entry to home delivery. PDQ advertises one-hour delivery. PDQ currently uses computers for in-store operations and the usual business functions, but otherwise is not heavily dependent upon software systems to help them receive, process, and home deliver their customers' orders. Pepe Ronee, their Supervisor of Computer Operations, has been charged with developing a software application to identify "pizza factory" locations and create the software system needed to operate them. In commissioning this project, Dee Livery, their president, said to pull out all the stops. She further stated that the future of PDQ depends on this project. She wants the team to investigate an option to deliver the pizza unbaked and "ready for the oven" in 30 minutes or less or deliver it pre-baked in 45 minutes or less.

These pizza factories would not have any retail space. Their only function will be to receive orders, prepare, and deliver the pizzas. The factory location nearest the customer's location will receive the order from a central ordering facility, process, and deliver the order within 30 or 45 minutes of order entry depending on whether the customer orders their pizza ready for the oven or already baked. There are six software applications that Pepe has identified for the solution.

Pizza Factory Locator Subsystem

The first is a software subsystem to find pizza factory locations. It is not known how many such factories will be needed nor where they should be located. The software subsystem will have to determine that. Clearly this subsystem is a very complex application. The goal can be clearly defined, but even at that the solution will not be at all obvious. This subsystem will have to use a very sophisticated modeling tool. The requirements, functionality, and features are not at all obvious. Some of the solution can probably be envisioned, but clearly the whole solution is elusive at this early stage. Exactly how to model it is not known at the outset. It will have to be discovered as the development project is underway.

Order Entry Subsystem

The second is an order entry subsystem to support store and factory operations. Telephone orders will come to a single location, be taken there and then routed to the appropriate store or factory electronically. This system focuses on routine business functions and should be easily defined. Off the shelf commercial software may be a big part of the final solution to support store and factory operations. This subsystem can utilize COTS (commercial off the shelf) order entry software.

Order Submit Subsystem

This subsystem will direct the order to a store, factory or pizza van. The logistics for making this assignment are not at all clear, and subsystem design will be complex.

Logistics Subsystem

This subsystem is the most complex of the six subsystems. It will require a holistic view of the entire PDQ system. Its complexity arises from the fact that the pizza vans are a mobile production and delivery facility. So the assignment of an order to a pizza van must take into account where the van is likely to be when it is time for order delivery.

Routing Subsystem

This software application will be a routing subsystem for the delivery trucks. This application is straightforward and will probably involve having GPS systems installed in all the delivery trucks.

Inventory Management Subsystem

The final application will be an inventory control system to manage inventories at all stores and factories and automatically reorder from the single vendor that PDQ has been using since it first started in the business. PDQ has been informed by their vendor that they can earn discounts by using the automatic reordering feature. This application should also be a COTS application. These applications are obviously very different software development projects requiring very different approaches. The Pizza Factory Locator subsystem will be a very sophisticated modeling tool. The requirements, functionality, and features are not at all obvious. Some of the solution can probably be envisioned, but clearly the whole solution is elusive at this early stage. Exactly how it will do modeling is not known at the outset. It will have to be discovered as the development project is underway. The Order Entry subsystem can utilize COTS order entry software that will have to be enhanced at the front end to direct the order to the closest factory and provide driving directions for delivery and other fulfillment tasks on the back end. The requirements, functionality, and features of this subsystem may be problematic. The six subsystems that make up the PDQ solution may each require a different project management approach. There will be a number of exercises incorporated in each chapter that require strategy formation and other decisions in order to find and maintain a best-fit project management approach.

Wysocki, Robert K. (2011-02-25). Effective Project Management: Traditional, Agile, Extreme (Kindle Locations 736-743). Wiley. Kindle Edition.