WHEN AND HOW SOFTWARE DEVELOPMENT BECOMES R&D?

In tough economic times, if product companies or independent vendors don't get time or funds for innovation, who will spend time and energy for that activity? It has been a million dollar question for product firms as product owners often spend their time and energy to think about new version releases, retaining customers, maintain older versions and thinking of staying ahead of the competition. But they tend to forget one important thing - Increased R&D spending is vital to product firms.

There are couple of unanswered questions for CxOs of product firms:

1. What is the appropriate rate of investment in R&D for software development companies in a bear market?
2. As budgets tighten across the software industry and beyond, how can they ensure that the software remains one of the hotbeds of innovation in the global economy?

Independent Software Vendors tend to reduce the cost of software projects and invest that money into software R&D without knowing how Software Development can become "technically" R&D. One thing is pretty clear: R&D budget should be kept separate

R&D = experimenting with ideas/technology with the probability that the ideas may never become a product/ prototype.

Vs.

Software development = working on a product/service desired by the organization’s end customers i.e a working product/ solution that is giving product firms the bread and butter.

There are two primary models:

- In the development model, the primary function is to develop new products/ improve existing products
- In the R&D model, the primary function is to discover and uncover development of valuable new products, processes, and services.

Organizations are so concerned about delays and quality problems in software development projects, both in-house and outsourced, that when they find a reliable, timely, high-quality software R&D arm, they are willing to pay competitive rates. It is important to distinguish between ‘Development’ and ‘R&D.’

Expedux addresses the all-important **WHEN AND HOW SOFTWARE DEVELOPMENT BECOMES R&D?** question with 6 points:

Point 1
R&D is all about developing new solutions for a specific problem domain. The end result is “research toys” or “Product/ idea prototypes/ proof of concepts/ pilot projects.

To be a software product, the proto type has to be completely re-implemented and it has to be accessible and loved by the end customer.

Point 2
Software R&D has different motivations and outcomes but it should be driven by a single minded focus - potential for profit/ dollar.

Point 3
All development of new products has to be inducted into R&D. It is important to create a product to sell and not just guys sitting around messing about with whatever they feel like.

Point 4
R&D in the technical world = finding ways to do something interesting or important, using known techniques and technology as a starting point.

Point 5
Managing software development is different for various sized companies and R&D takes on different meanings depending on the size of the company, customer base etc.

In a small software company, with only a hand full of employees, the line between R&D software and Production software is usually very small. What one day is a software R&D project, may the next day be shipping as production software to customers.

As software companies grow and have one or more production software lines, they tend to create greater separation between R&D software projects and Production software products (for obvious reasons). This R&D gap is typically created to create greater diversification in their software products for tomorrow, while allowing the production software development to continue to produce today.

Point 6
It is a wrong perception that R&D is the luckiest team in the planet who get to do anything they want without accountability.

Identifying R&D in software development

Product firms should take R&D seriously if:

- The solution is unique to a larger set of problems for a particular industry/ domain
- There is a potential for the prototype to become a cash cow within 2-3 years.

Expedux has gone through this process and we have developed a product for Omnichannel strategies and retail back end operations. Here are our top three tips for improving R&D activity descriptions:

- Discuss the state of the art and available knowledge, emphasizing how technical knowledge for R&D differs from regular software development.
- Outline the R&D effort in terms of cost and time to identify existing solutions that may be leveraged or new solutions that can be developed
- Describe the experiments that can be taken to justify the dollar invested vs solution identified.

ABOUT US

FOCUS

Software Engineering Services for Product Firms and E-commerce Organization

OPEN SOURCE COE

A Centre of Excellence team to constantly build capabilities and frameworks in PHP, JAVA, and Python.

STABILITY

A team backed by eminent industry advisors and venture capitalists.

EXPERIENCE

100+ man-years of expertise in working with opensource technologies.

CREDIBILITY

99% of our customers would vouch for our ability to go beyond the typical customer-vendor relationship

INNOVATIVE ENGAGEMENT MODELS

Innovative Milestone based engagement Model with Pay per Hour Model and Pay per Line of Code Model.

www.expedux.com
info@expedux.com

twitter.com/expedux
linkedin.com/expedux