APPLICATION DEVELOPMENT DURING 2020 COVID19 PANDEMIC - A CASE STUDY OF REKA

Maclean Patrick Sibat, Mohd Razif Baital @ Latif, Mahani Huja Husin
Centre for IT Development & Services, Universiti Malaysia Sarawak

Abstract – *Extra ordinary situations require extraordinary solutions and approaches in order to maintain the same level of productivity and efficiency in any given application development process. This is clearly true in order to circumvent the impact of the greatest disruptor to application development in the IT industry for 2020 – the COVID19 pandemic. In this case study paper, we look at the impact of REKA – a rapid development tool; developed in-house in Universiti Malaysia Sarawak (UNIMAS), towards the existing application development structure, practices and norms; and the ways it has helped maintain, if not exceed, the productivity levels of UNIMAS application developers during an adverse situation brought about by the COVID19 pandemic compared to productivity levels during past years. We look at the reasons REKA has been successful in meeting the application development needs of the university and the disruption it has caused towards existing established and accepted development methodologies employed by the application development teams at the Centre for IT Development & Services (CITDS), UNIMAS.*

INTRODUCTION

Application development has remained fairly consistent in the IT industry. Often times, any given software or application development[1] methodology has been able to resist the changes that come about through the emergence of new programming languages, scripts or development tools. This has held true till the emergence of Covid19, which has challenged established norms in application development on more than one level, not as a disruptor of technology but as a disruptor of the process itself and has impacted the productivity levels of developers across the world. For Universiti Malaysia Sarawak (UNIMAS) developers in the Centre for IT Development & Services (CITDS), the government's directive to impose the Movement Control Order (MCO)[2] on March 18; cause a disruption not just to the steps that constitute the normal steps to its application development methodology but also presented logistical and emotional stresses challenges. It was during this time that the development team aggressively adopted and elevated an in-house developed application development tool – REKA, to the fore-front; seemingly overtaking other established development tools and languages in the process, and ultimately provided the means to maintain (and even exceed) CITDS productivity level in spite of the restrictions caused by the Covid19 mitigation efforts imposed by the government. REKA ran concurrently with the other development tools yet gain popularity among developers within CITDS and also those outside of CITDS.

---

[1] For the purpose of this paper, the term software and application are interchangeable in the context of development. What is true for application development is true for software development.
[2] MCO period was from March 18 – June 9.

PROBLEM STATEMENT

Time is of the essence in application software and the quicker a developer can take a customer's request, conceptualize it and present forward a working IT system is equated directly to the capability and skill of the developer. The higher the trust in the developer's skills, the more work can be offered and the higher the productivity.

Development Time = Productivity = Trust in Skills (perceived) = Workload

Higher workload is often equated to mean more time spent in working out the results of said workload but it does not necessary hold true. Turnaround time to produce the workload is also essentially in application development. The quicker turn-around time means more results can be produced.

Development Time ↓ = Productivity ↑ = Trust in Skills (perceived) ↑ = Workload ↑

What happens when this equation is disrupted by a pandemic such as Covid19?

The implementation of the MCO essentially extended the development time of application development through the challenges presented by a work-from-home order.

Development Time ↑ = Productivity ↓ = Trust in Skills (perceived) ↓ = Workload ↑

A work-from-home order caused a break in the established routine and daily practices for a development team that is accustomed to working in a controlled office environment where all their tools were within arms-reach. Access to the system owners themselves, inhibited developers from acting on comments or feedback in order to perform reworks to their prototypes or test systems

As shown above, as development time goes up, productivity and trust in skills go down but the workload count continues to increase and contributes to an increasingly long backlog list of tasks.

Having a short development period pays off in less time and effort (Blackburn, Scudder, & Van Wassenhove, 1997) exerted by the development team and directly contributes to more tasks being able to be completed.

The issues faced by CITDS developers take on another dimension when pre-established application development methodologies that are inherently rigid and follow set routines and work-flows are derailed when developers are unable to operate within a controlled environment, such as an office setting. The restrictions set about by application development methodologies that help maintain order and structure under normal circumstances becomes a handicap when confronted by a situation such as one presented by Covid19.

How then, can CITDS developers maintain their, pre-Covid19, level of productivity, whilst under stay at home restrictions (MCO) during the Covid19 pandemic?

ANALYSIS

CITDS is actively working on and supporting about 120 modules with level of complexity ranging from medium to large-scale with 23 actives developers and 12 maintenance and support personnel.
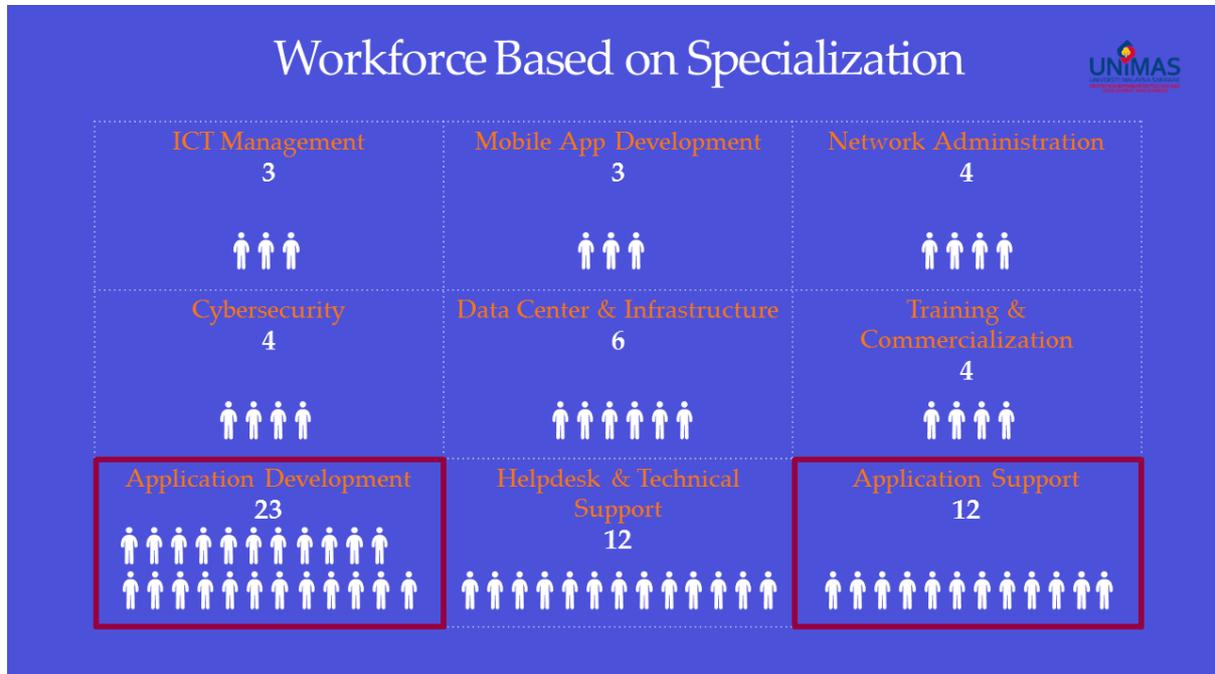
Ad-hoc requests[3] have always been a norm but during the pandemic, these requests have taken on an abnormal trend.

Ad-hoc requests fall into two main categories, 1) requests that are improvements or add-ons to existing systems and 2) requests that are totally new and not tied to any existing system.

The number of ad-hoc requests that fall under the 2nd category increased during the pandemic because the need for manual processes to be digitize and supported on an online platform was required in order to comply to new standard operation procedures related to Covid19. These ad-hoc requests were expected to be completed within a short time-frame.

---

[3] What is an **Ad Hoc Request**? An **ad hoc request** in the industry of project management typically indicates a task or job that was unexpected, and therefore, largely unscheduled. It's like a surprise popping up in the middle of your busy work week. (www.clarizen.com)

PROPOSED SOLUTION

1. REKA applies a 'building block' approach to application development.
2. REKA consolidates the application development process.
3. REKA consolidates the roles within an application development team.
4. Caveat – REKA does not discount the use of existing conventional application development methodology in used in CITDS. Instead, in an environment where rapid and ad-hoc development is needed REKA excels.

INNOVATIVE FEATURES

1. 'Building block' approach – provides blocks of features that developers can put together to form an application.
2. Low learning curve.
3. No source-code. – minimal coding needed for special features
4. Cloud-based solution.
5. Tolerant to chaotic circumstances or situations such as that caused by Covid19.

POTENTIAL (EFFECTIVENESS IMPACT)

Now everyone can be an application developer.

REKA allows the application developer to more creative in building their application much like the same way LEGO builders are released to build whatever design that comes to mind without having to be weigh down by the thought of technical matters.

References

Blackburn, J., Scudder, G., & Van Wassenhove, L. N. (January, 1997). Improving Speed and
Productivity of Software Development: A Survey of European Software Developers. *IEEE
Transactions on Software Engineering*, p. 16.