

업무, 조직 세분화로 인해 발생하는 문제점은?

Spotify의 조직 문화 글을 읽다가 2010년에 조직 구조와 관련해 쓴 글이 있어 가져와 본다. Spotify의 조직 문화 글에서 다른 Spotify가 만든 구조와 비슷한 부분이 있고, 내가 해결책을 찾지 못했던 부분도 있기 때문이다. 국내에서 Spotify와 같은 조직 구조와 문화를 같이 만들어 가고 싶지만 생각보다 쉽지 않다. 하지만 소프트웨어 개발에 있어 이 같은 요구와 도전은 계속해서 발생할 것이라 생각한다.

소프트웨어 개발자는 늘어나는데 생산성이 높아지지 않는 이유는?에서는 개발자(프로젝트에 참여하는 모든 구성원을 통칭한다.)는 증가하지만 개발 생산성이 향상되지 않는 가장 큰 원인은 "업무가 세분화되고, 세분화된 업무 단위를 기반으로 조직 구조가 만들어지는 것이다."라는 이야기를 다루었다. 이번 글에서는 업무 단위를 기반으로 조직이 만들어질 경우 생산성 저하와 발생 가능성이 있는 문제점에 대하여 살펴보도록 하겠다. 문제점의 우선 순위는 파급효과가 큰 부분을 먼저 다루도록 하겠다.

급격하게 증가한 협업 비용

소프트웨어 개발 과정에서 업무를 세분화할 경우 가장 큰 문제점은 협업 비용이 급격하게 증가한다는 것이다. 이에 대한 가장 큰 원인은 앞 글에서 인용했던 Pete McBreen의 글을 통하여 확인할 수 있다. 다시 한번 인용해본다.

소프트웨어 개발에 있어서 작업을 세분화하고 분업화할 때 작업이 더 작은 단계로 세분화될수록, 한 사람에서 또 다른 사람으로 정보를 전달하는 데에 더 많은 시간이 걸린다. 생산라인 접근 방식은 수작업 노동에는 잘 맞을 수 있다. 그러나 지적인 작업에는 형편없이 실패한다.

소프트웨어 개발은 팀원들의 머리에서 일어난다. 사람들을 특정 활동에 전문화시킴으로써 간단한 프로젝트의 딜리버리도 여러 경로를 통해야 한다. 각 경로는 실수와 결함의 잠재성을 갖고 있는 값비싼 과정이다.
- Pete McBreen, 소프트웨어 장인 정신

Pete McBreen의 말처럼 업무를 세분화할 수록 한 사람에서 다른 사람에게 정보를 전달하는데 많은 비용이 발생한다. 특히 각 업무를 맡고 있는 담당자의 Context가 다른 경우 그 비용은 더 커진다. 비용을 좀 더 구체적으로 살펴보면 다음과 같이 구분할 수 있을 듯 하다.

- * 다른 파트에 정보를 전달하기 위한 문서화 및 문서에 대한 유지보수 비용
- * 각 파트간에 정보를 공유하기 위하여 발생하는 회의비용
- * 다른 파트에서 정보가 넘어오기를 기다리면서 소요되는 비용
- * 정보를 잘못 이해함으로써 원하는 형태의 결과물이 나오지 않음으로써 발생하는 비용

위에서 살펴본 네 가지 이외에도 많은 비용이 발생하게 된다. 위 네 가지 비용 중에서 가장 큰 문제가 되는 경우는 네 번째일 것이다. 잘못된 요구사항 파악 및 구현으로 인하여 추가적으로 발생하는 유지보수 비용은 상당히 클 것으로 생각한다.

한 서비스(또는 프로젝트)를 여러 단위로 나뉘어 프로젝트를 진행할 경우 이와 같이 비용이 발생한다. 이 상태에서만 보아도 많은 비용이 발생하고 있는데, 각각의 업무 단위로 조직이 만들어질 경우 추가적인 비용이 또 발생하게 된다.

서비스(또는 프로젝트)가 개인 단위에서 조직 단위로 진행할 경우 각 조직의 성과와 정치적인 목적까지 개입하게 된다. 그러면서 추가적으로 발생하는 비용은 다음과 같은 부분이 있을 수 있다.

- * 각 조직의 프로세스에 따라서 업무를 요청하고 업무를 진행한다. 새로운 프로세스를 익히고, 요청 문서를 많드는데 비용이 발생한다.
- * 원활한 리소스 관리를 위하여 각 조직에서는 전체적인 리소스를 관리하기 위한 관리자가 필요한 상태가 된다.
- * 조직이 물리적으로 분리되어 있는 경우가 일반적이기 때문에 협업을 위하여 이동, 전화, 이메일 작성에 비용이 소요된다. 이 같은 비용은 결과적으로 협업 횟수를 줄임으로써 요구사항을 잘못 이해하고 개발함으로써 추가적인 비용을 발생하게 한다.
- * 서비스(또는 프로젝트)의 목표와 각 조직의 목표가 충돌하면서 추가적인 비용이 발생한다. 이 부분에 대해서는 잠시 후에 추가적으로 살펴보도록 하겠다.

이와 같이 업무 단위를 세부화하면서 업무 단위로 만들어진 조직까지 개입하게 되면 하나의 서비스(또는 프로젝트)를 진행하기 위하여 발생하는 협업 및 추가적인 비용은 급격하게 증가하게 된다. 이는 결과적으로 많은 사람을 투입했지만 생산성을 떨어트리는 결과를 가져온다.

원활한 리소스 관리를 위하여 각 조직에서는 전체적인 리소스를 관리하기 위한 관리자가 필요한 상태가 되면 여러 곳에 많은 관리자가 생겨나게 된다. 이와 같은 상황이 지속된다보면 가슴 아픈 이야기이지만 나 혼자 삽질하고 있는데 수 많은 관리자가 구경하면서 간섭하는 현상까지 발생한다. 즉, 고객의 가치를 위하여 일하는 시간과 사람보다 가치를 만들어내지 못하는 시간과 사람이 많아지게 된다.



이 같은 문제점을 해결한다는 명목하에 새로운 프로세스(방법론)가 도입되고, 값 비싼 도구를 구입하지만 결과적으로 실패하게 된다. 가장 큰 원인은 원론적인 부분을 해결하지 않은 상태에서 임시 방편의 해결책이기 때문이다. 이와 같은 상태의 가장 좋은 해결책은 업무 관련자들이 최대한 가까운 공간에서 일할 수 있도록 해야 하며, 각 조직의 정치적인 목적을 최대한 배제한 상태가 되어야 한다. 하지만 각 조직의 성과와 조직 관리를 해야 하는 입장에서는 쉽지 않은 선택이다.

조직의 목표와 서비스(또는 프로젝트) 목표와의 충돌

서비스의 목표는 고객에게 가치를 부여하면서 고객과 서비스가 같이 성장해 나가는 것이리라. 무슨 기술을 사용하고 방법론을 사용했느냐가 중요한 것이 아니라 서비스가 생존하는 것이 우선적인 목적이기 때문이다. 이를 실현하기 위해서는 서비스의 목표를 세우고 그 목표에 따른 우선 순위를 정하고 작업을 해나가는 것이 효율적인 방법이라.



그런데 업무 단위가 세분화되고, 더 나아가 조직까지 세분화되어 있는 상태에서는 가장 중심에 있는 서비스의 목표와 각 조직의 목표가 충돌하는 상황이 발생한다.



목표가 충돌할 경우 각 조직의 목표보다 서비스 목표가 우선시 되어야 하지만 많은 경우 각 조직의 목표를 우선시 하는 상황이 발생할 가능성이 있다. 이 상태에서 평가 권한까지 서비스(또는 프로젝트)에서 가지고 있는 것이 아니라 각 조직에 부여되어 있을 경우 더 이상 서비스 성공이나 목표가 중요한 것이 아니다. 각 조직의 성과를 내기 위한 조직의 목표를 맞추는 것이 더 중요한 상태가 되어 버린다.

이 같은 상태가 되면 서비스에 대한 우선 순위를 조정하기 힘들 뿐만 아니라 고객에게 가치를 줄 수 있는 부분보다는 각 조직에 성과를 내는 데 집중하게 됨으로써 서비스의 품질은 점진적으로 떨어지는 결과를 초래하게 된다. 이는 장기적으로 서비스에 큰 위협요소로 작용한다.

서비스 목표와 각 조직의 목표가 충돌하고, 조직의 세분화가 지속될 경우 대부분의 조직이 "프로젝트가 서쪽으로 간 까닭은?"에서 이야기하는 아드레날린 중독증에 빠진 상태가 될 가능성이 높다.

다음은 아드레날린 중독증에 걸린 조직이 보이는 특징이다. 아마 익숙하리라.

- # 우선순위가 계속 변한다.
- # 어제까지 모든 결과물이 나왔어야 했다.
- # 시간이 언제나 부족하다.
- # 모든 프로젝트가 긴급하다.
- # 긴급한 프로젝트가 계속 쏟아진다.
- # 모두가 언제나 미친 듯이 바쁘다.

이런 조직에서 일하는 사람들은 전략적으로 사고하지 않는다. 무조건 긴급한 업무부터 처리한다. '긴급도'가 낮은 프로젝트는(장기적인 이익이 보장되어도) 일단 무시한다. 그러다가 갑자기 급해지면 그제야 돌아본다. 아드레날린 중독증에 걸린 조직은 계획보다 전력 질주라 최선의 방법이라 믿는다.

- 톰 드마르코, 팀 리스터의 지금, 프로젝트가 서쪽으로 간 까닭은

모든 개발자들이 바쁘게 일하고, 열심히 일하고 있지만 생산성은 떨어지고, 개발자는 지속적으로 더 필요한 상태가 된다. 이 상태에서 개발자를 더 충원한다고 개발자의 삶이 나아질 수 있을까? 오히려 더 바빠지면 바빠졌지 나은 상태가 되지 않을 것이다. 왜? 이 상태에서 개발자를 충원하는 것이 원론적인 해결책이 아니기 때문이다. 하지만 아드레날린 중독증에 대해서는 조직 구조 개편이나 새로운 관리자로 대체한다고 해결할 수 있는 쉬운 문제가 아니다.

조직은 응급 상태가 아닐 때 가장 효율적이라는 사실을 인정하는 관리자가 필요하다. 하지만 이런 인력 교체는 거의 불가능하다. 흔히 조직이 미친 듯이 바쁘기를 바라는 사람은 경영진이나 CEO이기 때문이다. 그들은 회사가 미친 듯이 바빠야 잘 돌아간다고 믿는다. 회사 경영진이 아드레날린 중독증에 걸렸다면 프로젝트 팀들도 금방 따라 걸린다.

- 톰 드마르코, 팀 리스터의 지금, 프로젝트가 서쪽으로 간 까닭은

경영진이나 CEO가 자진해서 변할 수 있을까? 이 같은 문제점을 이야기할 수 있는 사람은 몇 사람이나 될 것이며, 이야기한다고 진실되게 들을 수 있는 경영진은 얼마나 될까? 이에 대한 대답은 글썽울씨가 맞을 듯 하다.

공정한 평가를 하기 힘들다.

평가를 공정하게 한다는 것은 사실 불가능한 일이다. 평가와 인센티브는 오히려 하지 않는 것이 지식 근로자에게 있어서는 더 나은 선택이 아닐까 생각한다. 왜냐하면 지식 근로자의 업무를 정량적으로 판단하기 힘든 부분이 대부분이기 때문이다. 특히 소프트웨어 개발 업무는 정량화하기 더 힘든 영역에 속한다. 많은 사람들이 정량화를 시도하고 있지만 그 내면을 보면 대부분이 경성적인 사실이다. 따라서 평가를 아무리 잘 하더라도 불만이 나올 수 밖에 없다.

그런데 조직 구조가 세분화되어 있고, 여러 조직이 하나의 프로젝트에 투입되는 상황에서 평가를 어떻게 해야할까? 만약 서비스 또는 프로젝트의 PM에게 평가 권한을 준다면 PM은 각 업무에 대한 이해도가 있어야 하며, 각 업무에 대한 난이도와 품질에 대한 기준도 알고 있어야 할 것이다. 다른 대안은 각 업무 조직에게 평가 권한을 위임하는 경우이다. 이 경우에는 팀원들이 각 프로젝트에서 어느 정도의 기여를 했는지를 판단해야 할 것이다. 이 경우 잘못할 경우 프로젝트에 대한 기여보다는 각 조직에 어느 정도의 기여를 했는지에 따라서 평가가 진행될 수 있다. 이럴 경우 앞에서 이야기한 것처럼 프로젝트 목표보다 조직의 목표를 우선시 하는 부작용이 발생한다.

어떤 방식을 취하더라도 문제가 있을 수 있으며, 이 두가지 방법을 조합해서 평가를 한다고 하더라도 조직장의 기준에 따라서 평가 기준이 달라질 수 밖에 없다. 평가에 대한 정답은 없다. 하지만 위와 같이 많은 문제점을 가진 상태라면 오히려 평가와 인센티브 지급이 생산성을 높이는 것이 아니라 저하시킨다. 이 상태라면 인센티브는 동일하게 지급하고, 평가는 개인의 역량을 강화하기 위한 목적으로 진행하는 것이 훨씬 더 효율적인 것이다.

이 글에서는 업무와 조직을 세분화할 경우 발생할 수 있는 문제점에 대하여 살펴보았다. 이 같은 문제점은 사람을 꾸준히 늘리더라도 생산성이 높아지는 것이 아니라 떨어트린다. 소프트웨어 개발회사에서 이 같은 문제점은 일정 수준 규모에서 급격하게 사람을 늘릴 경우 발생할 가능성이 높은 현상으로 판단된다.

지금까지 두편의 글에서 원인과 문제점에 대하여 살펴보았다. 그렇다면 "내가 생각하는 해결책은 뭔데?"라는 질문을 던질 것이다. 다음 글에서는 내가 생각하는 해결책에 대하여 다루도록 하겠다.

- 소프트웨어 개발자는 늘어나는데 생산성이 높아지지 않는 이유는?
- 업무, 조직 세분화로 인해 발생하는 문제점은?
- 웹 서비스 소프트웨어 개발 회사의 효율적인 조직 구조 및 관리

업무, 조직 세분화로 인해 발생하는 문제점은?

이 글은 내가 포털 회사에 4년 간 몸담고 있으면서 느꼈던 내 나름대로의 생각을 정리한 것이다. 이 글의 내용은 포털과 같이 하나의 소프트웨어를 지속적으로 관리하고 성장시켜 나가야하는 소프트웨어 개발 회사에 도움이 될 듯하다. 나는 지금까지 웹 에이전시, SI 회사에 몸담은 경험이 더 많은데 웹 에이전시, SI와 같이 특정 회사에 업무 의뢰를 받아 단기간에 개발을 완료해야하는 조직에는 적합하지 않을 수 있다.

이 글의 내용은 내가 몸담고 있는 조직의 의견과는 무관하다. 지금까지 소프트웨어 개발 업무를 진행하면서 느꼈던 내 나름대로의 생각을 정리한 것이다. 내가 이 글을 쓰는 가장 큰 이유는 내가 잘못 판단하고 있는 부분도 많을 것이기 때문에 그에 대한 의견을 받아보고 더 바람직한 모습에 대한 논의를 해보고자 함이다.

논리적으로 타당하지 않거나 근거가 부족하다고 생각하는 부분에 대해서는 가차없이 의견 마구마구 날려주기 바란다.