

IT 산업의 스쿼드 조직 모델 심층 연구

서론

스쿼드(Squad) 조직 모델은 소프트웨어 업계에서 “작은 스타트업 같은 자율 팀”으로 불리는 혁신적 조직 구조로, 2010년대 초 스포티파이(Spotify)가 소개한 이후 많은 IT 기업에 영향을 주었습니다 ①. 스쿼드 조직은 개발자, 디자이너, 기획자 등 다양한 직군이 하나의 목표를 위해 모인 6~12명 내외의 작은 팀으로, 팀 내에서 독립적으로 의사결정하고 제품의 기획부터 개발·운영까지 엔드투엔드로 책임지는 것이 특징입니다 ②. 이러한 크로스기능 자율팀 구조는 부서 간 사일로(silo)를 허물어 빠른 실행과 혁신을 도모하려는 취지에서 도입되었습니다.

본 보고서에서는 스쿼드 조직 모델에 대해 심층적으로 조사합니다. 먼저 여러 자료(백서, 블로그, 발표 등)를 통해 스쿼드의 정의와 주요 특징을 살펴보고, 각 자료가 스쿼드 모델에 대해 취하는 입장을 정리합니다. 이어서 해외 기업들의 스쿼드 적용 사례와 국내 기업들의 스쿼드 적용 사례를 조사하여, 실제 팀 규모나 운영 방식, 성과 등에 관한 정보를 제공합니다. 마지막으로 중소 IT기업이 스쿼드 조직을 도입·운영할 때 자주 묻는 의문점들을 FAQ 형식으로 정리하고 해설함으로써, 현실적인 시사점을 도출하고자 합니다. 결론에서는 주요 발견 사항을 요약하고, 스쿼드 조직 모델을 도입하려는 중소기업에 대한 조언을 제시합니다.

1. 스쿼드 조직의 정의와 주요 특징 (출처별 관점 비교)

스쿼드 조직 모델은 2012년 스포티파이 엔지니어링 문화 백서에서 처음 공개된 이후 업계에 알려졌으며, 이후 여러 전문가와 기업들이 이를 언급하거나 자기 경험을 공유해왔습니다. 아래에서는 대표적인 출처별로 스쿼드에 대한 정의와 특징, 그리고 각 출처의 입장을 정리합니다:

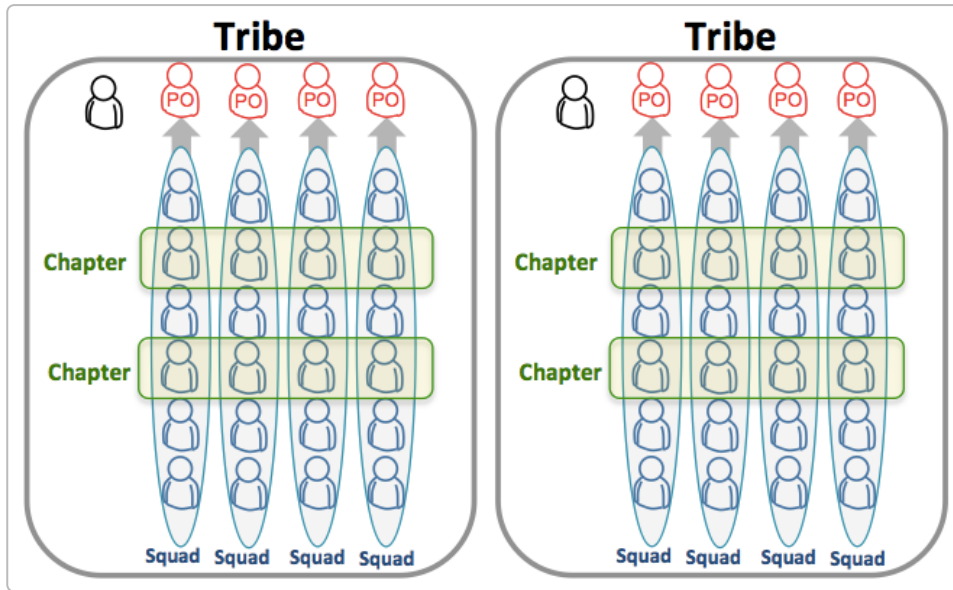
- 스포티파이 “Scaling Agile” 백서 (Henrik Kniberg & Anders Ivarsson, 2012) – 스쿼드 모델의 출발점:
스포티파이는 2012년 백서에서 전통적 조직 구조 대신 스크럼 팀과 유사하지만 더 자율적인 “스쿼드” 팀들로 개발 조직을 운영하는 방법을 소개했습니다 ③. 백서에 따르면 “A Squad is designed to feel like a mini-startup”로, 필요한 모든 기술을 가진 사람들이 모여 자율적으로 일하며 각자 방식으로 스프린트나 칸반 등을 운용한다고 합니다 ④ ⑤. 이 모델은 스포티파이가 30개 이상의 팀을 3개 도시에 분산시키면서도 애자일 문화를 유지한 경험에 기반하고 있습니다 ④. 백서는 챕터(Chapter)와 길드(Guild) 등의 개념으로 매트릭스형 조직을 구성하여 자율성과 규모의 경제를 모두 추구했으며, 스쿼드가 자기완결적 미션을 수행하고 챕터를 통해 전문성을 공유하도록 설계되었다고 합니다 ⑥ ⑦. 백서는 스쿼드 모델을 매우 긍정적으로 묘사하면서도 “이 글을 읽을 때쯤 우리는 또 변하고 있을 만큼, 이 모델은 진행 중인 여정”이라고 언급하여 특정 방식에 고정되지 않는 유연함을 강조했습니다 ③.
- 제레미아 리 (Jeremiah Lee, 前 Spotify PM) 2020년 블로그 – “Spotify도 실제로는 Spotify 모델을 쓰지 않았다”는 폭로: 2020년 전 스포티파이 프로젝트 매니저였던 제레미아 리는 「Failed #SquadGoals」라는 블로그 포스트에서 “Spotify doesn’t use ‘the Spotify model’ and neither should you.”라고 선언하며 큰 반향을 일으켰습니다 ⑧. 그는 2017년 스포티파이에 입사하여 본 스쿼드 모델이 “사실상 이상적인 청사진에 불과했고 완전히 구현된 적이 없었다”고 밝혔는데, 그의 동료들조차도 백서 내용이 현실과 다름에도 업데이트되지 않는 이유에 대해 “채용에 도움이 되니까”라는 반응을 보였다고 합니다 ⑨ ⑩. 제레미아는 “the famed squad model was only ever aspirational and never fully implemented” (유명한 스쿼드 모델은 애초에 추구만 되었을 뿐 결코 완전히 구현되지 않았다) ⑨ 고 직언하면서, 많은 기업들이 존재하지도 않은 모델을 맹목적으로 따라 하려는 현실을 비판했습니다. 또한 그는 스포티파이 모델의 문제점으로 △매트릭스형 엔지니어 관리 구조가 오히려 문제를 키운 점, △자율성만 강조되고 엔지니어링 리더십이 부재했던 점 등을 상세히 지적했습니다 ⑪ ⑫. 예를 들어, 스포티파이에서는 스쿼드 내 개발자들이 각자 전문 분야별로 챕터 리

더(엔지니어링 매니저)에 보고하고 있었는데, 단일 스쿼드에 전담 엔지니어 리더가 없다 보니 제품 책임자(PO)와 동등하게 의사결정을 조율할 기술 책임자가 없었고, 기술적 충돌 발생 시 여러 명의 챗터 리더와 일일이 조율하거나 최고 엔지니어 책임자까지 이슈가 escalate되는 비효율이 있었다고 합니다 12 13. 그는 이처럼 스쿼드 모델이 스포티파이 내부에서도 실패했다는 사실을 밝히며, 다른 조직들이 이를 그대로 모방하는 것을 강하게 경고했습니다 14 15. (참고로, 스포티파이 모델 공동저자인 Anders Ivarsson 역시 “우리도 이걸 완벽히 하고 있던 건 아니었다... 실제 존재하지 않는 것을 다른 사람들이 복제하려 애쓰고 있다”고 말한 바 있습니다 15.)

- **국내 조직문화 블로그 “일잘냥” (2025년)** – 성공과 실패 사례를 모두 본 현실적 시각: 국내 조직문화 컨설턴트로 보이는 블로거 “일잘냥”은 2025년에 스쿼드 조직에 관한 연속된 두 편의 글을 게시했습니다. 첫 번째 글 「스쿼드 조직 구조 완전정복」에서는 “스포티파이가 만든 혁신 조직 구조의 모든 것”이라며 스쿼드의 개념과 장점을 소개하고, 한국 스타트업의 적용 사례를 다뤘습니다 16. 이 글에서 저자는 스쿼드 도입으로 부서 간 조율 지연이 줄고 의사결정이 빨라져 “3배 업무 속도” 향상을 경험할 수 있다고 긍정적으로 설명했습니다 17 18. 실제 성공 사례로 국내 SaaS 스타트업 아이웹(Imweb)이 6개의 핵심 기능별 스쿼드로 조직을 재편해 팀 간 소통 비용을 크게 줄이고 기능별 전문성을 높인 성과를 거두었다고 소개했습니다 19 20. 그러나 동시에 저자는 “10명 남짓한 초기 스타트업이 5개 스쿼드로 나누었다가 오히려 효율이 떨어진” 실패 사례를 들며 조직 규모와 성숙도를 무시한 무작정 도입은 독이 될 수 있다고 경고했습니다 21. 이어진 두 번째 글 「스쿼드 조직이 실패하는 진짜 이유」에서는 한층 비판적인 톤으로, “2020년 스포티파이도 이미 이 모델을 포기했다”는 소식을 전하며 스쿼드 도입 실패율이 80%에 달한다고 주장했습니다 22 23. 이 글에서 그는 스쿼드 모델의 현실적인 문제점으로 ① 조직 규모 미고려, ② 리더십 공백, ③ 문화적 미성숙, ④ 성과측정 부재, ⑤ 형식만 추종 등을 꼽았고 24 25, 스타트업 규모별로 대안적인 조직 운영 모델(예: 20명 이하에겐 소규모 기능팀+강한 리더십 구조, 여러 프로젝트 동시 진행 시 태스크포스 조직, 성숙 조직에는 아마존식 2-피자팀 등)을 제안했습니다 26 27. 요컨대, 일잘냥의 입장은 “스쿼드는 잘 적용하면 혁신적이지만, 준비 안 된 조직에 무턱대고 적용하면 재앙이 될 수 있다”는 현실론으로 정리할 수 있습니다.

- **김준형 (스타트업 CTO) 2024년 Medium 기고** – “스쿼드는 환상이다” 극단적 비판: 2024년 7월에 발표된 김준형님의 Medium 글 「스쿼드(Squad)는 그만!!」은 스쿼드 조직에 대한 강도 높은 비판을 담고 있습니다. 필자는 스타트업에서 직접 스쿼드 도입을 시도했다 “경험은 고통이었다”고 회고하면서, 한국 스타트업계에서 스포티파이 모델의 스쿼드가 일종의 종교처럼 여겨진다고 꼬집었습니다 28 29. 그는 많은 스타트업들이 “스쿼드를 부르짖지만 정작 백서도 읽어보지 않아 진짜 스포티파이 모델이 뭔지도 모른다”며, 겉핥기식 도입이 만연함을 지적했습니다 30. 특히 “스쿼드는 현실 세계에서 운영을 시작하는 순간 인간의 정신 모델과 심각하게 충돌한다”고 표현하며 31, 스포티파이 조직 모델 자체에 근본적인 문제가 있다고 보았습니다. 김준형님은 스쿼드 모델이 전통 대기업의 사업부제보다도 괴이한 구조라며, 성공했다고 자평하는 사례들도 뜯어보면 “용어만 빌렸을 뿐 실제로는 사업부제에 더 가깝다”고 일축했습니다 30 32. 그의 전반적 입장은 스쿼드 조직에 대한 회의론으로, “환상적인 모델처럼 보이지만 현실에서는 동작하지 않는다”는 것입니다. 다만 글 말미에는 “그럼에도 불구하고 정말 스포티파이 모델을 성공시키고 싶다면?”이라는 부분을 두어, 충분한 준비와 철저한 이해 없이 도입하지 말 것을 강조하고 있습니다.

以上的 출처들을 종합하면, 스쿼드 조직 모델은 초기 백서나 성공담에서 강조되는 장점(자율성, 속도, 작은 팀의 민첩함)이 있는 반면, 실제 현실에 적용할 때 조직 문화, 리더십 공백, 성과 관리, 조직 규모 적합성 등의 문제로 인해 비판적 시각도 많이 존재함을 알 수 있습니다. 특히 원조 격인 스포티파이마저 내부 시행착오 끝에 기존 모델을 수정했다는 증언 9 15은, 어떤 조직도 맹목적으로 남의 구조를 복제해서는 안 되며 자기 맥락에 맞게 재해석해야 함을 시사합니다. 결국 스쿼드 모델에 대한 평가는 출처마다 긍정과 부정이 공존하며, 성공적으로 정착시키려면 장점은 살리고 단점은 보완하는 현실적인 접근이 필요하다는 공통된 교훈을 주고 있습니다.



Spotify 모델에서 제시된 스쿼드 매트릭스 조직 개념도. 여러 스쿼드(가로 열)들이 모여 하나의 트라이브(Tribe, 회색 박스)를 이루고, 각 스쿼드는 PO(Product Owner, 빨간색)와 개발자 등으로 구성된다. 개발자들은 동시에 자기 분야별 챗터(Chapter, 녹색 박스)의 일원으로 속하여, 목적 조직(스쿼드)과 기능 조직(챗터)의 이중 소속 구조를 갖는다^{33 34}. 이 매트릭스 구조를 통해 스쿼드 간 자율성과 전문성 공유를 모두 도모한다.

2. 해외 기업의 스쿼드 조직 적용 사례

스쿼드 모델은 스포티파이를 필두로 해외 여러 기업들에 직간접적으로 영향을 주었으며, 특히 기술 업계와 금융권 등에서 애자일한 조직 혁신 사례로 종종 언급됩니다. 여기서는 대표적인 해외 기업들의 스쿼드 도입/운영 사례와 그 특징을 살펴봅니다.

2.1 스포티파이 (Spotify) – 원조 스쿼드 모델의 실험과 교훈

스포티파이는 스쿼드 모델의 탄생지로서, 자사 엔지니어링 조직을 작은 자율팀들인 스쿼드로 나누고 그것들을 트라이브(Tribe)로 묶는 매트릭스 조직을 2012년 경 도입했습니다^{4 35}. 팀당 인원은 6~12명 내외로 제한하고, 각 스쿼드마다 PO(제품 책임자)가 제품 방향을 이끌며 디자이너, 개발자, 테스트 엔지니어 등이 한 팀에 속하는 완결형 구조였습니다^{36 5}. 엔지니어들은 기능별 챗터(예: 백엔드 챗터, 안드로이드 챗터 등)에 속하여 동일 전문분야 동료들과 같은 리더(챗터 리드)를 공유했고, 챗터 리드가 라인매니저로서 인사관리를 맡았습니다^{37 38}. 이러한 구조 덕분에 스포티파이는 한때 개발자 수백 명을 30여 개 팀으로 나누고도 유연성을 유지하며 빠른 제품 출시를 이어갔다고 전해집니다⁴. 실제로 2012년 발표 당시 이미 15백만 이상의 사용자를 가진 서비스로 급성장 중이었는데, 백서는 이 모델이 성과에 기여했다고 강조했습니다³⁹.

그러나 스포티파이 내부에서의 실제 경험은 앞서 언급한 백서의 이상과 달랐던 것으로 보입니다. 2015년경부터 스포티파이는 조직이 더 커지면서 점진적으로 구조를 조정했고, 2020년 경에는 초기 스쿼드/챗터 모델을 상당 부분 폐기한 것으로 알려졌습니다^{22 9}. 그 내부 경험을 전한 Jeremiah Lee에 따르면, 스쿼드 모델 자체는 “풀스택 작은 팀”이라는 개념으로서 효과적이었으나, 엔지니어링 매니저(챗터 리드)가 스쿼드 외부에 있는 매트릭스 구조는 여러 문제를 낳았습니다¹¹. 예를 들어 제품 PO와 짝을 이루는 기술 리더가 부재하여 의사결정의 균형이 어려웠고, 엔지니어들 간 갈등 발생 시 이를 조정할 명확한 리더십 라인이 없었다는 점 등이 드러났습니다^{12 40}. 결국 스포티파이는 더 전통적인 엔지니어링 관리 체계로 회귀하여 각 팀에 엔지니어링 매니저를 두는 형태로 변화해 갔다고 합니다^{11 41}. 스포티파이 사례의 교훈은, 스쿼드 모델이 혁신적이지만 만능은 아니며, 조직의 성장 단계에 따라 구조를 유기적으로 조정해야 한다는 것입니다. 스포티파이는 이 실험을 통해 얻은 통찰을 바탕으로 현재도 팀 자율성과 명확한 리더십의 균형을 찾기 위해 지속적으로 조직 문화를 개선해 나가고 있습니다.

2.2 ING은행 (네덜란드) - 대기업의 스쿼드 도입으로 민첩성 향상

유럽의 대형 금융기업 ING그룹은 2015년 경 스포티파이 모델에서 영감을 얻어 전사 애자일 전환을 단행한 대표 사례입니다. ING 네덜란드의 CIO와 COO는 맥킨지와 인터뷰에서, 2015년 6월 전통 조직을 스쿼드 기반 애자일 조직으로 “빅뱅” 전환했다고 밝혔습니다 ⁴². 그 결과 약 3,500명의 직원이 새로운 구조로 재편되었는데, 구체적으로 “약 350개의 9명짜리 스쿼드”를 만들고 이를 13개의 트라이브(tribes)로 묶었다고 합니다 ⁴³. 각 스쿼드는 마케팅, 상품, UX, 데이터 분석, IT엔지니어 등이 섞인 다기능팀으로 편성되어 고객 여정별 문제 해결에 집중하도록 구성되었습니다 ⁴⁴. 이때 기존의 부서, 프로젝트 관리자 계층 등을 제거하고 엔드투엔드(self-contained) 스쿼드에 권한을 실어준 것이 핵심이었습니다 ⁴⁵ ⁴⁶.

ING의 이 애자일 전환은 가시적인 성과를 가져왔습니다. 도입 1~2년 만에 시간 단축과 생산성 향상, 직원 몰입도 상승 등의 효과가 나타났는데, 임원들은 “이미 제품 출시 속도가 빨라지고 직원 참여도가 높아졌으며 생산성이 증가했다”고 평가했습니다 ⁴³. 예컨대 IT개발 방식을 개선하여 배포 주기를 연 5~6회에서 2주마다로 단축하고, 모바일 뱅킹 혁신을 선도할 수 있게 되었다고 합니다 ⁴⁷ ⁴⁸. 또한 스쿼드 내에 IT와 비즈니스 인력이 함께 앉아 협업함으로써 부서 간 핸드오프가 줄어들고 의사소통이 빨라졌습니다 ⁴⁵. 물론 전사 차원의 변화였던 만큼 성과측정 방식이나 기존 관리자들의 역할 변화 등 도전도 있었지만, ING는 명확한 역할 정의와 거버넌스를 마련하여 이를 극복했습니다 ⁴⁹ ⁵⁰. 무엇보다 경영진의 강력한 의지와 “끝까지 해 본다”는 일관된 추진력이 성공 요인으로 꼽힙니다. ING 사례는 보수적인 금융권 대기업도 애자일 스쿼드 모델을 통해 스타트업처럼 빠르게 움직일 수 있다는 가능성을 보여준 동시에, 규모에 맞게 모델을 현지화하는 중요성을 일깨워줍니다 (ING는 스포티파이 모델을 그대로 복제하기보다 자사 문화와 필요에 맞게 변형했다고 밝히고 있습니다 ⁵¹).

2.3 아마존 (Amazon) - “두 피자 팀” 원칙으로 자율 스쿼드 운영

비록 “스쿼드”라는 용어를 쓰지는 않지만, 아마존은 작은 단위의 자율팀 철학을 일찍부터 도입한 대표적인 기업입니다. 제프 베조스(Jeff Bezos)는 “Two-Pizza Team”(두 판의 피자만 배분될 수 있는 팀) 원칙을 제시하여, 어느 팀이든 두 판의 피자만 배분되지 못할 정도로 크다면 팀을 쪼개야 한다고 강조했습니다 ⁵². 이는 보통 팀 인원을 5~10명 내외로 제한하는 것으로, 스포티파이의 스쿼드 규모 권장(6~12명 ⁵³)과도 일맥상통합니다. 베조스의 목표는 효율성과 확장성 두 가지였는데, “소규모 팀은 일정 조율이나 내부 보고에 쓰는 시간이 줄어들고 실제 일에 더 많은 시간을 쏟게 된다”고 그는 말했습니다 ⁵². 또한 작은 팀을 많이 운영하려면 이들이 공용 인프라와 리소스에 쉽게 접근하고 상호 조정할 수 있어야 하는데, 아마존은 이를 위해 모든 팀이 표준화된 방식(API 등)으로 소통하도록 강제했습니다 ⁵⁴ ⁵⁵. 덕분에 새로운 비즈니스 아이디어가 생기면 별도의 승인 절차 없이 작은 팀을 꾸려 실험하고, 공통 플랫폼(예: AWS 인프라)을 활용하여 대규모로 확장할 수 있었습니다 ⁵⁶ ⁵⁷. 업계에서는 이러한 아마존의 구조를 일종의 스쿼드 개념의 선구적 적용으로 평가하기도 합니다. “팀을 기계의 부품처럼 만들어 내는 기계”라는 비유처럼 ⁵⁸, 아마존은 작은 팀들이 자율적으로 움직이면서도 전체 조직이 일관된 플랫폼으로 연결되어 혁신의 속도와 규모를 동시에 달성했습니다. 그 결과 오늘날 AWS와 같은 신규 사업이 폭발적으로 성장하는 기반이 마련되었고 ⁵⁹ ⁶⁰, 아마존은 규모의 역설을 극복한 조직문화의 모범으로 자주 언급됩니다.

2.4 그 외 해외 사례: 마이크로소프트, 넷플릭스, 은행권의 도입 움직임 등

스쿼드 모델의 영향은 실리콘밸리의 테크기업뿐 아니라 전통 산업의 디지털 전환 사례에서도 찾아볼 수 있습니다. 예를 들어 마이크로소프트(Microsoft)는 Satya Nadella CEO 부임 후 많은 팀을 기능 중심에서 제품 중심의 소규모 크로스 기능 팀으로 재편하며, 클라우드 혁신을 가속화했습니다 (비공식적으로 “애자일 팟(pod)” 혹은 스쿼드 유사 구조를 도입). 넷플릭스(Netflix) 역시 “Context, not Control”이라는 문화 원칙 하에, 각 제품/서비스 기능별로 작은 풀스택 팀들이 자율적으로 서비스 구축을 책임지는 구조를 운용해 왔습니다 ⁶¹. 넷플릭스의 경우 스쿼드라는 용어는 쓰지 않지만, 유사한 자율팀 철학이 조직 전반에 구현되어 있다고 볼 수 있습니다.

한편 금융업계에서는 앞서 ING처럼 기존 계층조직을 헐고 핀테크 기업처럼 움직이려는 시도로 스쿼드 모델을 채택하는 사례가 있었습니다. 북미의 Capital One이나 호주의 ANZ Bank 등이 애자일 트랜스포메이션의 하나로 스쿼드 기반 조직을 도입한 바 있고, 유럽의 Nordea Bank 등도 참고 사례로 거론됩니다. 이러한 기업들은 공통적으로 기존의 부서 사이 장벽을 허물어 IT개발자와 기획자, 마케팅 인력이 한 팀에 소속되어 신규 디지털 제품을 빠르게 출시하는 것을 목

표로 삼았습니다. ING 사례에서 보았듯, 스쿼드로의 재편은 금융권에서도 제품 출시 기간 단축과 고객지향 문화 정착 등의 성과를 가져왔다는 평가입니다 ⁴³ ⁴⁵ . 다만 대규모 조직에서는 스쿼드 도입이 **부분적(특정 사업부에 한정)**이거나 **혼합형(일부는 기능조직 유지)**으로 이루어지는 경우도 많습니다. 예컨대 이후 소개할 신한카드의 사례처럼, **코어 영역에만 스쿼드를 적용하고 지원부서는 종전 형태를 유지하는 방식** 등입니다.

종합하면, 해외 기업들은 각자의 맥락에서 스쿼드(혹은 유사한 소규모 자율팀) 구조를 활용하여 혁신 속도를 높이거나 조직 문화를 변화시키려 했습니다. **스타트업 출신 기업**(스포티파이, 넷플릭스 등)은 스쿼드 모델을 성장기에 조직 혼란을 줄이며 확장하는 수단으로 사용했고, **전통 대기업**(ING, 은행들)은 이를 디지털 시대에 민첩성을 얻는 변혁의 도구로 삼았습니다. 이 과정에서 일부는 큰 성과를 거두었고, 일부는 어려움을 겪으며 수정해나갔습니다. 중요한 것은 “스쿼드”라는 이름보다, 소규모 자율팀이 효과를 내기 위한 조건들 - 적절한 팀 규모, 명확한 미션, 지원해주는 플랫폼과 리더십 - 이 제대로 갖춰졌는가 하는 점임을 해외 사례들은 보여주고 있습니다.

3. 국내 기업의 스쿼드 조직 적용 사례

국내에서도 2010년대 후반부터 스타트업을 중심으로 스쿼드형 조직 운영이 유행하기 시작했고, 최근에는 **대기업/금융권에도 영향을 미치는 추세**입니다. 여기서는 국내에서 스쿼드 모델을 도입한 기업들의 사례를 살펴보겠습니다. 특히 스타트업이나 중소 IT기업의 실험과 경험을 중심으로, 어떻게 스쿼드를 운영했고 어떤 효과와 교훈이 있었는지 알아봅니다.

3.1 아임웹(Imweb) - 스타트업의 성공적인 스쿼드 전환

아임웹은 누구나 손쉽게 온라인 쇼핑몰을 만들 수 있는 SaaS 웹사이트 빌더 서비스를 제공하는 **국내 스타트업**으로, 스쿼드 조직 전환의 성공 사례로 자주 언급됩니다 ¹⁹ . 2022년경 직원 수 100명 안팎이던 아임웹은, 제품 기능이 다양해지고 한 명의 직원이 너무 많은 부분을 담당하게 되면서 **협업 부담과 커뮤니케이션 비용이 증가**했습니다 ⁶² . 이를 해결하고자 **박재완 CPO** 주도로 **2022년 11월 스쿼드 제도**를 도입하였습니다 ⁶³ .

현재 아임웹에는 **6개의 스쿼드**가 존재하며, 각각 **온라인 쇼핑물 서비스의 주요 기능 영역**을 맡고 있습니다 ²⁰ . 예를 들어 **검색/회원관리, 쇼핑, 결제, 마케팅, 사이트 제작, 고객관리** 등 6가지 핵심 기능별로 스쿼드가 편성되어, 각 스쿼드가 해당 도메인에 집중하도록 했습니다 ²⁰ . 스쿼드당 인원은 **4~8명** 정도로, 기획자, 개발자, 디자이너가 골고루 섞인 크로스기능 팀입니다 ⁶⁴ . 각 스쿼드에는 **PO/기획자(Product Owner)**가 배치되어 백로그 우선순위를 정하고 팀을 이끌며, 개발 측면에서는 **Tech Lead(시니어 개발자)**가 존재해 기술 결정을 주도합니다. 아임웹은 또한 **챕터(Chapter)** 제도를 함께 운용하는데, 예를 들어 **프론트엔드 개발자 챕터, 백엔드 개발자 챕터, 디자인 챕터** 등 같은 직군끼리 모인 **4개의 챕터**가 있어 스쿼드 간에 전문 지식과 표준을 공유합니다 ⁶⁵ . 아임웹 메이커 조직은 **9개의 스쿼드와 4개의 챕터**로 구성되어 있으며 (2023년 기준 확대) ⁶⁵ , 스쿼드에서는 **제품 개발에 집중**하고 챕터에서는 **기술 전문성 향상과 개인 성장에 중점**을 두는 건강한 매트릭스 구조를 정착시켰습니다 ⁶⁶ .

아임웹의 스쿼드 도입 효과는 매우 긍정적인 것으로 보도되었습니다. 팀원들은 “**디자인팀과 개발팀을 따로 조율할 필요 없이 한 자리에서 논의하고 결정하니 속도가 빨라졌다**”고 말합니다 ⁶⁷ . 실제로 BF (사이트 제작 기능) 스쿼드의 회의에선 디자인, 개발 의견이 실시간으로 교환되며 **기존에 2~3단계 거쳤을 의사결정이 한 번에 이루어지는** 모습을 볼 수 있었다고 합니다 ⁶⁷ . 또한 개발자 입장에서, “**과거엔 서비스 전반을 모두 개발하느라 전문성 쌓기가 어려웠는데, 이제는 특정 도메인(예: 디자인 모드)만 집중 개발하니 더 깊은 전문성을 갖게 되고 그만큼 책임감도 느낀다**”고 전합니다 ⁶⁸ . 이는 스쿼드 구조가 **개발자들의 전문역량 강화와 주인의식 고취**에도 도움이 되었음을 시사합니다. 조직 차원에서는 **팀 간 의존도가 낮아져 병목이 줄고** ⁶⁹ , 각 스쿼드가 자신의 **명확한 KPI에 따라 움직이므로 성과 책임소재도 분명**해졌다고 합니다. 실제로 아임웹은 스쿼드 도입 후 **조직 규모와 MRR(월 반복매출)**이 2년 만에 **2.5배** 성장할 정도로 빠르게 확장하면서도, “**팀 단위, 스쿼드 단위로 일하는 체계가 잘 잡혀있고 데이터 기반으로 전사 협업이 원활**”하다는 평가를 받았습니다 ⁷⁰ .

물론 아임웹도 시행착오가 없진 않았습니다. 초기에는 **10명 남짓한 스타트업이 성급하게 5개 스쿼드로 나누었다가 실패한 사례**도 있었다고 하며 ²¹ , 아임웹의 경우 적절한 시기(직원수 100명 근처)에 도입했기에 효과를 본 측면도 있습니다. 박재완 CPO는 “**스쿼드에 필요한 인재를 계속 채용하여 각 기능을 전문 담당할 스쿼드를 갖춰나갈 계획**”이라고

밝히는 등 71, 인력 보강과 조직 성숙도를 함께 높여가는 전략을 병행하고 있습니다. 아임웹 사례는 국내 중소 IT기업이 스쿼드 조직을 성공적으로 안착시킨 좋은 예로서, 명확한 기능 단위로 팀을 쪼개고, 각 팀에 제품 책임자와 필요한 직능을 모두 넣어 빠른 실행력을 확보하면 큰 성과를 거둘 수 있음을 보여줍니다. 또한 작은 조직일수록 처음에는 일부 핵심 분야부터 스쿼드를 도입하고 점진적으로 확장하는 것이 효과적임을 시사합니다 72.

3.2 우아한형제들 (배달의민족) - 제품별 스쿼드 운영

국내 배달앱 1위 배달의민족을 운영하는 우아한형제들 역시 부분적으로 스쿼드 개념을 활용한 제품 개발팀을 운영하고 있습니다. 공식적으로 발표된 조직 구조 자료는 많지 않지만, 채용 공고나 사내 블로그 등을 통해 엿볼 수 있습니다. 예를 들어 우아한형제들의 개발자 채용공고에는 “라이브선물스쿼드” 등의 팀 이름이 등장하는데, “배민쇼핑라이브와 배민선물하기를 만들어가는 작고 거대한 애자일 조직”이라고 소개되어 있습니다 73. 이는 특정 제품/서비스(예: 라이브 커머스 선물 기능)를 책임지는 크로스펄링 팀을 스쿼드로 부르는 예시입니다. 이 팀은 프론트엔드, 백엔드 개발자와 기획자 등으로 구성된 것으로 알려져 있으며, 팀장 선에서 대부분 의사결정이 이루어지는 자율팀으로 운영됩니다. 우아한형제들은 여러 신사업이나 기능 개발에 이런 “미니 스타트업” 팀 문화를 적용해왔고, 그 덕에 업계 최초로 다양한 혁신 기능들을 빠르게 출시할 수 있었다는 평가를 받습니다. 우아한형제들 조직문화는 “실패를 두려워하지 않고 빠르게 실행”하는 것으로 유명하며, 이러한 문화에 스쿼드형 태스크포스가 잘 맞아떨어진 것으로 보입니다. 다만 우아한형제들의 전사 조직이 완전히 스쿼드로 재편된 것은 아니고, 핵심 제품 개발 부서 내에서 스쿼드 개념을 적용하는 정도로 파악됩니다.

3.3 카카오뱅크 - 매트릭스 조직 (캠프 & 클랜)으로 애자일 운영

카카오뱅크는 카카오 계열 인터넷전문은행으로, 스쿼드와 유사한 “목적 조직” 개념을 도입하여 운영하고 있습니다. 카카오뱅크 기술블로그에 따르면 이 회사는 “목적 조직과 기능 조직이 결합된 매트릭스 조직구조”로 일하고 있습니다 74. 여기서 목적 조직이란 다양한 직무의 구성원이 하나의 서비스/제품이라는 공통 목적을 갖고 모인 팀으로, 스포티파이의 스쿼드 또는 트라이브에 해당합니다 75. 카카오뱅크에서는 이러한 목적 조직 단위를 “캠프(Camp)”라고 부르며, 예컨대 투자, 결제, 사업자금융 등 상품 라인별로 캠프 조직을 꾸려 해당 영역 개발자, 기획자, 디자이너들이 한 팀에 속해있습니다 76. 반면 기능 조직은 동일한 직무별로 모인 조직으로, 여기서는 개발자들의 전문영역별 클랜(Clan)이 그 역할을 합니다 77. 현재 카카오뱅크에는 서버클랜, 코어뱅킹클랜, 모바일클랜 세 가지 큰 개발자 클랜이 있어서, 각 클랜이 소속 캠프를 초월해 전체 개발자들의 기술 방향과 표준을 논의합니다 78. 쉽게 말해, 캠프 = 제품별 크로스기능 팀, 클랜 = 역할별 챗터 모임인 셈입니다. 이러한 구조는 스포티파이의 스쿼드/챗터 모델을 카카오뱅크 현실에 맞게 적용한 사례로 볼 수 있습니다.

카카오뱅크는 클랜 제도 운영을 통해 스쿼드 구조의 장점과 단점을 균형잡고 있습니다. 각 캠프별로 보면 다양한 직무가 모여 있으니 속도감 있는 제품 개발이 가능하지만, 자칫 캠프 간 단절이나 기술 부채 누적 문제가 생길 수 있습니다 79. 이를 보완하기 위해 정기적으로 클랜 데이(Clan Day)라는 행사를 열어 같은 직무 개발자들이 한데 모여 기술 이슈를 공유하고 표준을 정립합니다 81 82. 예를 들어 서버개발자 클랜데이에서는 여러 캠프에서 공통으로 사용하는 API Gateway의 개선점이나 기술 부채 항목을 다같이 논의하고 우선순위를 매기는 식입니다 83. 이러한 노력으로 카카오뱅크는 새로운 상품 출시의 속도와 안정적인 기술 품질 관리를 모두 추구하고 있습니다. 실제로 조직구조 혁신을 통해 2017년 출범 이후 빠른 주기로 예·적금, 대출, 투자상품 등을 시장에 내놓으며 업계에 큰 파장을 일으켰습니다. 카카오뱅크 사례는 금융권 스타트업이 스쿼드형 매트릭스 구조를 성공적으로 운용한 예로서, “한국형 스포티파이 모델”의 한 형태라 할 수 있습니다.

3.4 신한카드 등 핀테크 전환 - 전통기업의 스쿼드 실험

흥미롭게도, 최근에는 전통 대기업이나 금융회사들도 “카카오처럼” 일하기 위해 스쿼드 조직 실험을 진행하고 있습니다. 2021년 한국경제 보도에 따르면, 신한카드는 자사의 플랫폼 비즈니스 부문을 개편하며 개발자+기획자+디자이너가 한 팀으로 일하고 팀장 선에서 의사결정하는 “스쿼드” 조직을 도입했습니다 84. 그동안 일부 TF로만 시범 운영하던 스쿼드를 본부 단위로 상설화한 것으로, 이는 카드업계 최초의 시도였습니다 84 85. 예컨대 신한카드의 DNA사업추진단 산하 3개 팀(마케팅데이터 플랫폼팀, 가맹점주 플랫폼팀, 디지털월렛팀)을 모두 스쿼드화하여, 각 팀에 기획자·개발자 약 13명씩을 배치하고 팀장이 (차장급) 권한을 갖고 신속히 결정하게 했습니다 86. 흥미로운 점은, 신한카드는 디자인과 서버인프라 인력을 해당 팀에 완전히 소속시키진 않고 별도의 중앙 조직(DX팀, IT본부)에서 지원하도록 한 “절충

형 스쿼드”를 채택했다는 것입니다 ⁸⁶. 이는 **핵심 인력은 스쿼드로 묶되 전문인력은 공유자원화하여 운용**하는 형태로, 완전한 크로스기능 팀이 아직 생소한 전통 조직의 현실을 반영한 것으로 보입니다.

신한카드의 스쿼드 도입 배경에는 **디지털 플랫폼 사업에서는 5단계 결재 라인으로는 속도전을 할 수 없다는 위기의식**이 있었습니다 ⁸⁵. 여러 직급을 거치는 의사결정 구조 대신, “**10명 안팎 스타트업처럼 운영**”하여 빠르게 제품을 개선하려는 것입니다 ⁸⁷. 업계에서는 이를 “**핀테크 회사로 거듭나기 위한 시도**”로 평가하며, **네이버·카카오와의 경쟁에서 뒤처지지 않으려는 준비**라고 분석했습니다 ⁸⁷. 신한카드 외에도 KB국민카드는 2018년부터 본부장들에게 조직 신설 권한을 부여해 **여러 팀을 임시 스쿼드로 운영**해왔고, **하나카드는 2021년 디지털플랫폼Tribe를 신설**하여 스쿼드제로 운영, **롯데카드도 앱 리뉴얼 프로젝트팀을 스쿼드 형태로 구성**하는 등 카드업계 전반으로 확산되고 있습니다 ⁸⁸. 이는 국내 전통 기업들도 **애자일한 작은 팀 구성을 통해 혁신 역량을 키우려는 움직임**이라 할 수 있습니다.

물론 이러한 대기업식 스쿼드는 **조직 문화나 인사제도 등이 완전히 바뀐 것은 아니**라서, 진정한 자율팀으로 자리잡기까지 과제가 남아있습니다. 예컨대 **직급 호칭이나 평가 체계는 종전 틀을 유지한 채 팀 편성만 바꾼** 경우도 있어, 구성원들이 초반에는 혼란을 겪을 수 있습니다. 그럼에도 불구하고 점차 **성과를 내는 사례가 나오면** 다른 부문으로 확대될 가능성이 있습니다. 실제 신한카드 사례 이후로 **다른 전통기업(예: 제조 대기업의 디지털 전환 조직)**에서도 유사한 스쿼드형 태스크포스 도입이 이어지는 추세입니다.

以上的 국내 사례들을 통해 알 수 있듯이, 스타트업부터 금융 대기업까지 폭넓은 스펙트럼에서 스쿼드 조직 모델을 실험하고 있습니다. **스타트업의 경우 비교적 유연한 문화 덕에 스쿼드 도입이 빠르게 정착되고 성과로 이어지는** 경우가 많았고, **기존 기업들은** 부분 도입이나 혼합 모델로 시작하여 **徐徐**문화를 변화시켜가는 양상을 보입니다. 중요한 공통점은, **성공한 곳들은 스쿼드 모델을 자기 현실에 맞게 변용하고 보완**했다는 것입니다. 반대로 실패한 경우는 **유행에 편승해 피상적으로 조직도만 바꾼 채, 권한위임이나 문화 변화는 따라가지 못한** 사례들이었습니다 ³⁰. 이를 통해 **중요한 교훈은, 조직 구조 혁신은 수단일 뿐 목적이 아니며, 제대로 효과를 내려면 리더십의 의지, 구성원의 마인드셋 변화, 지원체계가 마련**이 함께 가야 한다는 점입니다.

4. 중소 IT기업의 스쿼드 도입 FAQ (자주 묻는 질문과 해설)

마지막으로, **중소 규모의 IT 기업**이 스쿼드 조직 방식을 도입할 때 현장에서 부딪치기 쉬운 **실제적인 질문들과 그에 대한 해설**을 정리했습니다. **기존 기능조직에서 스쿼드로 전환할 때의 고민, 스쿼드 운영 중 나타나는 문제 상황** 등을 Q&A 형식으로 다루어 보니다.

Q1. 스쿼드 수가 많아지면 조직이 너무 비대해지고 관리가 힘들어지지 않나요?

A: 스쿼드는 본래 **조직 비대화를 막고자 고안된 작은 팀 단위**입니다. 한 팀의 인원을 6~10명 선으로 제한함으로써 팀 자체의 효율을 높일 수 있지만, 회사가 성장하여 이런 팀이 수십 개로 늘어나면 **팀 간 조율과 정렬이 새로운 과제가 될 수 있습니다**. 이를 해결하기 위해 스쿼드 모델에서는 “**트라이브(Tribe)**” 개념으로 연관된 스쿼드 몇 개를 상위 그룹으로 묶어 관리합니다 ⁸⁹. 예를 들어 스포티파이는 **스쿼드들이 10~12개 모이면 하나의 트라이브로 편성**하고, **트라이브 간에는 느슨한 연계만 유지**하도록 했습니다 (트라이브 규모는 보통 100명 이하로 제한 ⁸⁹ 하여, 전체가 한 커뮤니티처럼 느낄 수 있도록 한 것입니다). 그러므로 **스쿼드 수가 늘어나는 것은 곧 트라이브가 늘어나는 것**이고, 각 트라이브 별로 또렷한 미션과 경계를 설정한다면 **전체 조직이 복잡해지는 것을 완화**할 수 있습니다.

실제로 조직이 작는데 **너무 많은 스쿼드로 쪼개면 오히려 비효율**입니다. 앞서 소개한 실패 사례처럼 **직원 10명 남짓한 스타트업이 5개 스쿼드로 나누는 식은 “탁상 위에서 축구팀을 나누는 격”**이라는 지적이 있습니다 ⁹⁰. 이 경우 각 팀이 너무 작아 **리소스 중복과 의사소통 부담만 늘어나**죠. 일잘랑 블로그도 “**처음에는 한두 개 스쿼드로 시작해 경험을 쌓은 후 점진적으로 확대하는 게 안전**”이라고 조언합니다 ⁷². 요약하면, **스쿼드가 많아지는 건 회사 성장의 자연스런 결과**지만 이를 다룰 수 있는 **상위 구조(트라이브 등)와 협업 체계(챗터, 길드 등)를 함께 도입**하면 관리가 가능합니다 ⁶ ⁹¹. 핵심은 **모든 스쿼드가 회사의 공통 비전 아래 정렬(Alignment)되도록 주기적으로 전체 방향을 공유**하고(예: PO들 로드맵 협의 ⁹²), **정보 공유 메커니즘(내부 위키, Demo Day 등)을 구축**하는 것입니다 ⁹³. 그렇게 하면 **작은 팀이 많아도 큰 조직처럼 영키지 않고, 오히려 작은 팀의 민첩함과 큰 조직의 자원 활용을 동시에 얻을 수** 있습니다.

Q2. 모든 스쿼드가 동등한 지위를 갖는 “평등한 팀”인가요? 핵심 기능을 맡은 스쿼드와 지원 성격의 스쿼드가 나뉘기도 하나요?

A: 이상적인 스쿼드 조직에서는 모든 스쿼드가 각각 독립된 미션과 가치를 지닌 작은 스타트업처럼 취급됩니다 ¹. 따라서 우열을 가리기보다는 각 스쿼드의 성격이 다를 뿐이라고 보는 편이 맞습니다. 예를 들어 어떤 스쿼드는 고객 facing 핵심 제품 기능을 개발할 수도 있고, 다른 스쿼드는 내부 플랫폼이나 공용 서비스(예: 결제 시스템, 추천 엔진)를 담당할 수 있습니다. 전자를 흔히 “스트림-얼라인드 팀”(고객 가치 흐름에 직접 align된 팀), 후자를 “플랫폼/엔에이블링 팀”(다른 팀을 지원하는 플랫폼 제공팀)이라고 부르기도 합니다. 결국 모든 팀이 목표와 KPI를 가지고 움직이지만, 그 KPI의 성격이 다를 수 있습니다. 핵심 기능 스쿼드는 제품 매출이나 사용자 지표가 성과일 수 있고, 지원 스쿼드는 시스템 안정성이나 내부 고객(다른 팀) 만족도가 성과일 수 있습니다.

현실적으로 기업에서는 모든 영역에 스쿼드를 꾸릴 만큼 인력이 충분치 않을 때, 중요도에 따라 어떤 분야는 스쿼드화하고 어떤 것은 공용 지원부서로 남겨두기도 합니다. 예컨대 앞서 언급한 신한카드 사례에서, 디자인이나 서버 인프라 인력은 각 스쿼드에 붙이지 않고 중앙 IT부서가 여러 스쿼드를 지원하도록 했습니다 ⁸⁶. 이렇듯 성격상 공통 서비스는 굳이 다수 스쿼드에 중복 배치하기보다 플랫폼팀으로 두는 편이 효율적일 수 있습니다. 아마존은 이런 전략을 극단까지 밀어올려, 공통으로 쓰는 기능은 각 팀이 API를 통해 호출하는 내부 플랫폼화를 이루었고 ⁵⁵, 심지어 이를 외부에 AWS로 상품화하기도 했습니다 ⁶⁰. 스포티파이의 경우도 대부분 스쿼드가 사용자 기능을 개발했지만 일부 인프라팀은 Chapter로 묶인 형태로 여러 스쿼드를 도와주는 식이었습니다 (예: 공동 테스트 인프라 등).

따라서 핵심 스쿼드 vs 지원 스쿼드 구분이 뚜렷이 존재한다기보다는, 각 스쿼드의 미션이 다양하며 어떤 것은 직접적 제품 기능, 어떤 것은 이를 가능케 하는 기반을 다룰 수 있습니다. 중요한 것은 지원 역할의 스쿼드라도 자신들의 서비스를 “내부 고객”인 다른 스쿼드에게 제공한다는 점에서 하나의 제품팀처럼 운영된다는 것입니다. 예를 들어 DevOps 플랫폼 스쿼드라면, 다른 개발 스쿼드들이 자사 플랫폼을 통해 배포/모니터링을 쉽게 할 수 있도록 제품처럼 개발/운영하고, 그 성과를 측정받습니다.

또한 회사 전략상 우선순위에 따라 “코어” 스쿼드와 “비코어” 스쿼드를 분류하기도 합니다. 일부 조직에서는 핵심사업 관련 스쿼드에는 더 많은 자원과 권한을 주고, 주변 기능 스쿼드는 규모를 작게 유지하는 경우도 있습니다. 하지만 이런 차별이 지나치면 스쿼드 간 위화감이 생길 수 있으므로 유의해야 합니다. 원칙적으로는 모든 스쿼드가 동등하게 자율성과 책임을 부여받되, 사업 임팩트의 크고 작음에 따라 투자 우선순위만 다를 뿐임을 분명히 하는 것이 바람직합니다. 즉, “균등한 팀”이라기보다 “독립된 스타트업들의 포트폴리오”로 보는 관점이 필요합니다.

Q3. 스쿼드를 도입하면 기존의 부서별 조직 (예: 기획팀, 개발팀, 운영팀)은 다 사라지나요? 기존 직능 조직과는 어떤 관계를 맺게 되나요?

A: 기존 기능별 부서는 “일하는 단위”로서는 사라지지만, “인적 운영 단위”로서는 형태를 바꿔 계속 존재합니다. 스쿼드 조직은 전통적 의미의 부서를 없애고 목적 조직으로 재편하는 것이기 때문에, 더 이상 “개발 1팀 vs 개발 2팀”처럼 일감을 따로 진행하지는 않습니다 ⁹⁴. 대신 각 스쿼드에 기획·디자인·개발 인력이 모두 속해 일하는 거죠. 그러나 그렇다고 직능별 관리와 교류가 완전히 사라지는 것은 아닙니다. 대부분 스쿼드 모델을 도입한 회사는 기존 부서를 챗터(Chapter)라는 형태로 유지합니다 ³⁵ ⁶⁵. 챗터는 앞서 설명했듯 동일한 직군/전문분야의 직원들 그룹이며, 챗터를 이끄는 챗터 리드(또는 클랜 리드)가 해당 직원들의 라인 매니저 역할을 수행합니다 ³⁷. 예를 들어 개발팀장이 챗터 리드가 되어, 여전히 그 개발자들의 멘토링, 역량 개발, 평가 등을 책임지는 식입니다 ³⁷. 다만 챗터 리드 자신도 어떤 스쿼드에 소속되어 실무를 병행하는 경우가 많아서 (예: 개발팀장이면서 A스쿼드 백엔드 개발자로 일함) 완전히 옛날 부서장과 같지는 않습니다 ⁹⁵.

또한 챗터는 공식 조직이라기보다 느슨한 커뮤니티 같은 성격이라, 정기적으로 모여 각자의 스쿼드에서 배운 교훈을 공유하거나 기술 표준, 업무 룰 등을 논의합니다 ⁹⁶. 결국 기능 조직은 챗터/클랜 형태로 남아 조직 전체의 전문성 일관성을 유지하는 역할을 하고, 일상적인 프로젝트 추진은 스쿼드에서 이루어지는 이원 구조(매트릭스 조직)가 됩니다 ⁶⁵ ⁹⁷. 흔히 “스쿼드는 목적 조직, 챗터는 기능 조직”이라고 부르며, 두 축이 동시에 존재합니다 ⁷⁷.

따라서 스쿼드 도입 후에도 기존 직무별 리더(개발팀장, 디자인팀장 등)는 챗터 리드 또는 트라이브 리드 등으로 역할이 변화할 뿐 완전히 없어지진 않습니다. 다만 과거처럼 모든 결정을 부서장이 내리는 구조가 아니므로, 이들 리더는 조율

자 및 코치 역할에 집중하게 됩니다. 예를 들어 **챗터 리더**는 “**교수형 리더**”로서 기술적 품질과 인재 성장을 신경 쓰고, **PO(프로덕트 오너)**는 “**미니 CEO**”로서 제품 가치 극대화에 집중하는 식으로 **역할 분담**이 이루어집니다 33 98 .

조직도 상으로는 기존 부서 구분이 희미해지고, 직원들은 “**OO스쿼드 소속 X직군**”으로 표기되는 식으로 바뀝니다. 그러나 **HR평가, 채용, 교육 체계** 등은 여전히 직군별 라인이 담당하는 경우가 일반적입니다. 예를 들어 **신입 채용은 개발 챗터에서 일괄 진행해 각 스쿼드로 배정**하거나, **디자이너들의 커리어 패스**는 **디자인 챗터장**이 관리하는 식입니다. 정리하면, 스쿼드 모델에서는 기존 기능 조직이 형태만 바뀔 뿐 “**사라지는 것**”이 아니라 “**백그라운드로 물러난다**”고 볼 수 있습니다. **현업 프로젝트의 전면**에는 스쿼드가 있고, 그 뒤에서 **챗터/기능조직**이 **인사/전략**을 뒷받침하는 구조로 이해하시면 됩니다.

추가로, 스쿼드 도입 초기 과도기에는 **옛 부서와 새로운 스쿼드 구조가 혼재**하여 혼란이 올 수 있습니다. 이때는 **TF를 활용한 시범 운영**이나, **일부 프로젝트에만 스쿼드 적용** 등으로 단계적 전환을 하는 것이 좋습니다. 점진적으로 **사람들의 인식**이 “**우리 팀(스쿼드)**” 중심으로 바뀌도록 유도하고, 일정 시점 이후 기존 부서 역할을 공식적으로 없애는 식이죠. 스포티파이 사례에서도 “**기존 매니저들의 자리를 없애고 새로운 Chapter Lead/Tribe Lead 역할로 재편**”하는 변화 과정이 있었다고 합니다 49 50 . 내부 커뮤니케이션을 통해 **왜 기능조직 해체가 필요한지 공감대**를 형성하는 것도 중요합니다 (“부서 간 벽을 없애야 빠르게 움직일 수 있다” 등의 공감). 이렇게 **기존 조직을 없애는 것이 아니라 더 효과적으로 일하기 위해 재구성**하는 것임을 인지시키면, 전환 충격을 줄일 수 있습니다.

Q4. 우리 제품은 신규 개발보다 유지보수 업무가 많은데, 이런 환경에서 스쿼드의 기민함을 유지할 수 있을까요?

A: 가능합니다. 스쿼드 조직은 반드시 혁신적인 신기능 개발에만 적합한 것은 아니며, 오히려 지속적인 제품 개선과 유지보수에도 유리한 구조라는 시각이 있습니다. **왜냐하면** 특정 제품/모듈을 책임지는 스쿼드는 **그 영역의 모든 일을 end-to-end로 다루기 때문**입니다 65 97 . 새로운 기능 개발이든, 버그 수정이나 성능 개선이든 **모두 같은 팀에서 우선순위를 정해 처리**하면 되므로 **업무 전환이 유연**합니다. 전통적인 조직에서는 신규 개발 부서와 운영 부서가 나뉘어 있어 서로 미룰 수도 있지만, 스쿼드에선 **개발과 운영 인력이 한 팀**이라 “**운영도 우리 일**”이 됩니다. 예컨대 **아임웹의 한 스쿼드**는 “**서비스 기획부터 참여해 더 나은 제품을 직접 만들어가고... 배포 후 모니터링까지 전 과정 협력**”한다고 하는데 99 100 , 이처럼 런칭 이후 지속적 관리까지 자기 팀의 임무로 여기면 유지보수 단계에서도 책임감 있고 빠른 대응이 가능합니다.

물론 유지보수 위주 상황에서는 **팀이 반복적인 작업에 안주**하거나 **기술 부채가 쌓일 위험**이 있는데, 이를 방지하려면 정기적인 리팩터링/개선 목표를 스쿼드 OKR에 포함시키는 것이 좋습니다. 또한 스쿼드 간에 **공통 기술 이슈**를 논의하는 장(앞서 언급한 클랜 데이나 챗터 모임)을 통해 더 나은 아키텍처나 프로세스를 함께 모색할 수 있습니다 79 78 . 예를 들어 어떤 스쿼드에서 효과 본 유지보수 자동화 스크립트를 다른 스쿼드도 공유받아 쓰는 식입니다. **카카오뱅크처럼 여러 캠프(스쿼드)의 개발자들이 클랜에서 신기술 도입이나 공통 시스템 개편**을 논의하는 것도 같은 맥락입니다 79 78 .

만약 제품이 성숙기에 접어들어 **개선 작업이 소규모로 산발적**이라면, 스쿼드의 형태를 약간 유연하게 운영할 수도 있습니다. 예를 들어 **필요한 기간만 임시로 모이는 태스크포스(TF)**로서의 스쿼드입니다. **일잘냥 블로그**에서는 “**프로젝트 기반 태스크포스**”를 스쿼드의 대안 중 하나로 들었는데, **다양한 프로젝트가 동시에 진행되는 조직**에서는 TF형 팀이 현실적일 수 있다고 합니다 101 . 이때도 TF를 스쿼드처럼 자율적으로 운영하면 유사한 효과를 낼 수 있습니다.

지속적인 유지보수의 경우 **팀원들의 동기부여**도 고려해야 합니다. 새로운 도전 없이 버그FIX만 하는 데 지칠 수 있으므로, **순환근무나 스쿼드 개편**으로 새로운 목표를 부여하는 방법도 있습니다. 예컨대 1년 정도 지나면 스쿼드 간 인원을 일부 교체하여 신선한 시각을 불어넣거나, 아예 유사 영역 두 스쿼드를 합쳐 리소스 재배치할 수도 있습니다. 스포티파이도 초기엔 스쿼드 이동 시에도 매니저 변경 없게 하려 했지만, 실제로는 **팀이 장기 존속**하면서 이동이 드물었고 오히려 한 팀 내에 엔지니어 리더를 두는 방향으로 선회했었습니다 102 12 . 이처럼 **상황 변화에 따라 스쿼드 운영방식**을 탄력 조정하는 것이 좋습니다. 그러나 원칙은 유지보수 역시 해당 도메인의 스쿼드 책임으로 두는 것이며, 그렇게 해야 사용자 피드백 반영이나 장애 대응이 빠르고 개선사항이 축적됩니다. DevOps 관점에서 “**빌드하고 운영까지**” 하는 팀이 궁극적으로 서비스 품질을 높인다는 믿음과 일치합니다.

Q5. 스쿼드 체제에서는 인사평가를 누가, 어떻게 하나요?

A: 스쿼드 체제로 가면 전통적인 상사-부하 관계가 흐려지기 때문에 **인사평가(성과평가, 피드백 등) 방식**도 변화합니다.

일반적으로는 앞서 설명한 **챗터/기능조직의 리더(=라인 매니저)**가 부하 직원의 평가를 책임지는 체계를 유지합니다 37. 예컨대 개발자 A씨는 “백엔드 챗터” 소속으로 챗터장에게 보고하므로, 챗터장이 A씨의 연간 평가를 주도하되 A씨가 속한 스쿼드의 PO나 스쿼드 동료들로부터 360도 피드백을 받아 종합하는 식입니다. 실제 스포티파이도 “**Chapter Lead is line manager for his chapter members, with responsibilities like developing people, setting salaries**”라고 명시했는데 37 103, 이런 챗터 리드(=기능 매니저)가 **성과평가 주관자**가 됩니다. 다만 Jeremia Lee의 회고에 따르면, 스포티파이의 챗터 기반 평가에는 한계가 있었는데, **챗터 리드가 팀의 일상 업무에는 관여하지 않다 보니** 부하의 업무 기여도나 팀 내 협업 능력을 제대로 파악하기 어려웠다는 지적이 있습니다 104 40. 즉, “**Chapter leads... don’t have accountability for delivery. It’s easy to see the product owner as the manager for the team.**”라는 코멘트처럼 40, 실질적으로는 PO가 팀 리더처럼 보이는데 공식 평가는 챗터장이 하니, 괴리가 있었다는 것이죠.

이 문제를 인지한 여러 회사들은 평가에 **Squad 리더(PO) 의견**을 강하게 반영하거나, 아예 스쿼드 내에 실질적 리더를 두어 그가 평가하도록 개선했습니다 11 12. 예를 들어 아마존이나 다른 기업들은 한 팀에 엔지니어링 매니저(팀장)를 두는 형태라, 그 팀장이 팀원 평가를 담당합니다. 스쿼드 모델을 도입하더라도, **각 Squad별 테크 리드나 PO가 존재한다면 이들을 평가에 참여시켜야** 공정성이 높아집니다. 국내 스타트업들은 보통 스쿼드 도입 시 **기존 팀장들을 챗터장 겸 PO 등으로 배치**하기에, 평가 주체가 크게 달라지지 않을 수도 있습니다. 예컨대 5개 스쿼드로 구성된 회사라면, **기존 개발팀장→백엔드 챗터장, 기획팀장→여러 스쿼드의 PO 총괄** 등으로 역할 이동을 시키고, 평가시 **공동으로 피드백**을 주고받는 방식이 있습니다.

인사평가 방식 자체도 좀 더 **성과 목표 중심(OKR 등)**으로 바뀌는 경향이 있습니다. 개인의 활동보다는 **스쿼드 단위 성과와 그에 대한 기여도**를 평가하는 것이 합리적이므로, **공동 OKR 달성여부 + 개인 역할 기여** 조합으로 보는 식입니다. 또한 **동료 평가(peer review)**의 비중을 높여, 스쿼드 동료 간 피드백을 정량/정성적으로 수집하기도 합니다. 중요한 건, **평가 주체와 피평가자의 일하는 맥락이 동떨어지지 않게** 하는 것입니다. 따라서 **중소기업의 경우 CEO나 임원이** 일일이 전 직원 평가를 했던 구조라면, 스쿼드 도입 후에는 **챗터장/스쿼드리더들에게 권한을 위임**하고, **HR부서는 360피드백 운영과 공정성 검증** 역할에 집중할 수 있습니다.

마지막으로, **성과보상** 측면에서도 변화를 줄 수 있습니다. 팀 단위 인센티브를 도입하여 **스쿼드 전체의 성과 목표 달성 여부에 따라 보너스**를 지급하면, 자연스럽게 같은 스쿼드 내에서 협업과 동기부여가 촉진됩니다. 개인 평가와 팀 평가를 병행하여 균형을 맞출 수도 있겠습니다.

정리하면, 스쿼드 체제에서의 인사평가는 “**한 손에 두 줄 잡기**” 방식으로, 공식적 라인 매니저(챗터리드 등)가 책임을 지되 비공식 팀 리더(PO/Tech Lead)의 의견을 적극 반영하고, 동료 피드백을 함께 고려하는 구조가 바람직합니다. 조직의 성숙도에 따라 최적 해법은 다르지만, **평가자와 피평가자 간에 업무 이해의 간극이 최소화되도록** 설계하는 것이 핵심입니다 40.

Q6. 각 스쿼드마다 PO(Product Owner)나 기획자가 별도로 다 있어야 하나요? 작은 회사는 PO 한 명이 여러 팀을 맡으면 안 될까요?

A: 이론적으로는 이상적인 스쿼드마다 전담 PO를 두는 것이 권장됩니다. 스쿼드는 작은 제품 팀이고, 그 팀에 비즈니스 방향을 책임질 **Product Owner(또는 PM)**이 반드시 필요합니다 36. “**The squad has a dedicated product owner**” 36 라는 원칙이 있을 정도로, PO의 존재는 스쿼드 성공의 열쇠입니다. **한 명의 PO가 두 개 이상의 스쿼드를 겸임하게 되면, 시간과 관심 분산**으로 인해 각 팀의 **민첩한 의사결정**이 어려워집니다. 가능하면 **스쿼드당 1명의 PO**를 확보해야 하며, 이것이 부담된다면 초기에 **스쿼드 수를 PO 수에 맞춰 적정하게 설정**하는 것이 좋습니다. 예컨대 PO가 2명뿐인 회사라면 처음엔 두 개 스쿼드로 시작하고, 향후 PO를 충원하며 스쿼드를 늘리는 식입니다 72.

물론 **현실적으로 인력이 부족한 소규모 기업**은 한 사람이 여러 역할을 겸해야 할 때가 있습니다. 그런 경우 **임시 방편**으로 **한 PO가 두 스쿼드의 우선순위를 관리**하되, 각 스쿼드 내에 PO 대리를 맡을 수 있는 **시니어**를 두어 보완할 수 있습니다. 또는 **CEO나 최고기획자(CPO)**가 여러 팀의 PO 역할을 부분 담당하면서, 팀 내 리더들과 협업해 의사결정을 하기도 합니다. 그러나 이는 어디까지나 과도기 전략이고, 장기적으로는 “**한 팀 한 PO**” 원칙을 지향해야 합니다.

국내 아임웹 사례에서도 **CPO 산하에 제품 기획 조직이 존재**하며, CPO가 직접 6개 스쿼드 전체를 감독하지만 **각 스쿼드마다 책임 PO/PM을 지정**하여 운영한 것으로 보입니다 ⁶³. “**기획자, 디자이너, 개발자가 섞여 공동체처럼 일한다**”고 소개된 부분에서 기획자가 곧 PO 역할을 하는 셈입니다 ⁶⁴. 회사 여건에 따라 **PO와 기획자를 동일 인물로 볼 수도 있고, 분리된 역할로 볼 수도** 있습니다. 만약 기획자가 따로 있다면 **기획자는 요구사항 도출과 문서화 역할**, PO는 **우선순위 결정과 비즈니스 판단 역할**로 분담하기도 합니다. 하지만 스타트업에서는 흔히 **PO = 기획자(PM)**로 겸하며, 스쿼드마다 한 명씩 투입하는 것이 일반적입니다.

PO를 스쿼드마다 두지 못하면 어떤 일이 생길까요? **우선순위 충돌과 의사결정 지연**이 생길 가능성이 큼니다. 예를 들어 한 명의 PO가 A, B 두 스쿼드를 본다면, 두 팀의 요구사항이 경쟁할 때 적시에 결정을 못 내리거나 한 팀은 대기해야 하는 상황이 나오겠죠. 또한 팀원들도 “**우리 전담 PO가 없다**”고 느끼면 제품 방향에 대한 궁금증이나 혼선을 겪을 수 있습니다. **스쿼드 모델의 핵심은 자율성과 책임**이기에, **결정권을 줬 PO가 팀에 항상 상주**하여 비즈니스 의사결정을 바로 내려주는 것이 중요합니다 ³⁶. 그래서 규모가 작은 회사라면 **차라리 스쿼드 수를 줄여서라도 PO 집중도를 높이는 편이 낫습니다**.

만약 현실적으로 PO 인력 충원이 어렵다면, **차선책으로 제품 백로그를 투명하게 공개**하고 **경영진이 정한 우선순위 원칙**을 팀원들이 충분히 이해하도록 하여 **어느 정도 팀이 self-prioritize**하게 할 수는 있습니다. 하지만 이것도 **한계**가 있으므로, 결국 **제품 책임자를 키우는 것이 조직의 투자 과제**가 됩니다. 많은 국내 스타트업들이 **PO/PM 인력 양성에 힘쓰고 채용**하는 이유가 여기에 있습니다.

결론적으로, “**스쿼드마다 PO/기획자 배치**”는 **이상적인 표준**이며, **소규모 기업도 이를 염두에 두고 인력계획을 세우는 것이 바람직**합니다. 현실적으로 초기엔 일부 겸직이 불가피하더라도, **조직이 성장하면 가장 먼저 PO 인력을 확충**해주는 것이 스쿼드 운영의 효과를 극대화하는 길입니다.

以上の FAQ를 통해, **중소 IT기업이 스쿼드 조직을 도입할 때 부딪히는 구체적인 의문들과 대응 방안**을 살펴보았습니다. 작은 조직은 작은 조직 나름대로 **장점(유연한 문화, 빠른 변화 가능)**이 있으므로, **자신만의 맥락에 맞는 스쿼드 운영 원칙을 정립**하는 것이 중요합니다. 특히 리더십은 스쿼드 도입 초기의 혼란을 줄이고, **새로운 조직문화로 정착**시킬 수 있도록 명확한 방향을 제시해야 합니다. 그리고 **구성원들의 질문과 우려에 귀 기울여 같이 답을 찾아가는 과정** 자체가 애자일 문화 확산의 시작이라는 점을 기억해야 할 것입니다.

결론

스쿼드 조직 모델은 “**작은 팀의 민첩성과 큰 조직의 자원 활용을 동시에 노리는**” 현대적 조직 운영 방식으로, 지난 10여 년간 IT산업에서 많은 관심을 받아왔습니다. 본 보고서에서 살펴본 바에 따르면, 스쿼드 모델의 **핵심 특징**은 **6~12명 내외의 크로스기능 팀이 제품/서비스 단위로 자율적 책임**을 지는 구조이며, 이를 뒷받침하기 위해 **채터/길드 등의 매트릭스적 요소로 전문성 공유와 일관성을 유지**한다는 점입니다 ⁶⁵ ⁶.

스쿼드 모델에 대한 **자료들은 상반된 견해를 보여주었는데, 원조 격인 스포티파이 백서나 일부 성공 기업 사례는 스쿼드의 혁신성과 성과를 강조**한 반면 ⁴³ ¹⁸, **후일담이나 비판적 기고에서는 현실적인 시행착오와 실패 원인**을 부각시켰습니다 ⁹ ³¹. 이를 모두 종합해볼 때 얻을 수 있는 통찰은, **스쿼드 모델이 만병통치약은 아니지만 적절한 조건과 문화가 뒷받침되면 유용한 도구**가 될 수 있다는 것입니다. 특히 **한국의 스타트업 아임웹 사례는 명확한 목표별로 스쿼드를 구성하고 단계적으로 정착시켜 큰 효과**를 본 예로서, **중소 IT기업들이 참고할 만합니다** ⁶⁷ ⁶⁶. 한편 **Spotify 본사의 경험이나 국내 일부 실패담은, 조직의 성숙도/규모를 무시한 채 형식만 따라하면 오히려 혼란**이 초래됨을 경고합니다 ¹⁰⁵ ³⁰.

중소 IT기업이 스쿼드 조직을 도입하려면, **다음과 같은 실용적 시사점**을 염두에 두어야 합니다:

- **점진적 도입**: 처음부터 전사 모든 팀을 스쿼드로 재편하기보다, **파일럿 스쿼드 1~2개**를 운영하며 경험을 축적하고 확대하는 것이 좋습니다 ⁷². 조직 규모 30~100명, 기본 프로세스 안정화 시기가 도입 적기로 거론됩니다

¹⁰⁶.

- **자율과 책임의 균형:** 스쿼드에 권한을 줄수록 책임도 명확히 부여해야 하며, 성과를 팀 단위로 측정하고 피드백하는 체계를 만드십시오 107 25 . 경영진은 간접 대신 방향성 제시와 코칭에 집중하고, 실패에 대한 문책보다 학습을 장려해야 자율문화가 자리잡습니다.
- **리더 역할 재정립:** 기존 팀장/임원의 역할이 모호해지지 않도록 **캡터 리드, PO, Tribe 리더** 등 새로운 역할을 명확히 정의해야 합니다 37 33 . 라인매니저(캡터장)는 인재 양성과 전문성 향상에, PO는 제품 비전에 집중하는 등의 “**듀얼 리더십**” 모델을 운영합니다.
- **협업 도구와 정보 공유:** 스쿼드가 늘어나면 자칫 **사일로(silo)**가 생길 수 있으므로, 주간 스쿼드 공유 세션, 내부 위키/툴 활성화 등으로 팀 간 정보 흐름을 원활히 유지해야 합니다 93 . 아울러 공통 기준(코딩 표준, 디자인 가이드 등)을 캡터 차원에서 정하고 적용 일관성을 모니터링합니다.
- **문화적 준비:** 수직적 문화에서 수평적 자율팀 문화로 전환하려면 **구성원 교육과 의식 전환** 노력이 필요합니다. 애자일 방법론 교육, 초기 스쿼드에 애자일 코치 투입, 성과 공유 이벤트 등을 통해 스쿼드 방식에 대한 이해와 공감을 높입니다.
- **스쿼드 규모/구성 최적화:** “**피자 두 판 룰**”을 기억하며 스쿼드 인원이 너무 크거나 작지 않게 유지합니다 (7±2명이 적절) 108 . 또한 스쿼드 내 역할 폴셋(기획/디자인/FE/BE 등)을 가능하면 채워 외부 의존도를 줄이고 완결성을 갖추도록 합니다 2 . 현실상 부족한 역할은 공유 인력으로 두되, 점차 스쿼드 전담으로 전환합니다.

마지막으로, 스쿼드 조직은 도구일 뿐 목적이 아닙니다. 궁극적으로 달성하고자 하는 바는 더 빠른 고객 가치 전달, 직원들의 몰입도 향상, 그리고 시장 변화에 대한 민첩한 대응일 것입니다. 스쿼드 모델 도입이 이러한 목적에 부합하도록, 자사 비즈니스와 문화에 맞게 창의적으로 적용하고 필요 시 변형하는 태도가 중요합니다 109 . 남의 성공 사례를 맹목적으로 복제하기보다는, 그 밑바탕의 원리를 이해하고 자기 조직에 맞는 자기만의 모델을 만드는 것 - 이것이 많은 전문가들이 강조하는 점입니다 110 30 .

결론적으로, 스쿼드 조직 모델은 중소 IT기업이 성장통을 겪을 때 조직 혁신의 돌파구가 될 수 있습니다. 부서 간 장벽으로 인한 지연, 책임 소재 불명확, 조직 비대화로 인한 관료화 등의 증상이 보인다면, 작은 스타트업 정신을 조직에 심어주는 스쿼드 방식을 고려해볼 만합니다 111 . 다만 그 성공여부는 **구현 방식에 달려있음**을 기억하고, 본 보고서에서 논의한 사례들의 교훈을 토대로 현명하게 접근한다면, “여러 개의 작은 스타트업”으로서의 당신 회사도 한층 민첩하고 창의적인 조직으로 거듭날 수 있을 것입니다.

1 2 16 17 18 19 21 53 72 93 94 106 107 108 111 스쿼드(Squad) 조직 구조 완전정복 | 스포티파이 모델부터 한국 스타트업 성공 사례까지

<https://smart-work.tistory.com/entry/%EC%8A%A4%EC%BF%BC%EB%93%9CSquad-%EC%A1%B0%EC%A7%81-%EA%B5%AC%EC%A1%B0-%EC%99%84%EC%A0%84%EC%A0%95%EB%B3%B5-%EC%8A%A4%ED%8F%AC%ED%8B%B0%ED%8C%8C%EC%9D%B4-%EB%AA%A8%EB%8D%B8%EB%B6%80%ED%84%B0-%ED%95%9C%EA%B5%AD-%EC%8A%A4%ED%83%80%ED%8A%B8%EC%97%85-%EC%84%B1%EA%B3%B5-%EC%82%AC%EB%A1%80%EA%B9%8C%EC%A7%80>

3 4 6 7 34 35 36 37 39 91 92 95 96 98 103 blog.crisp.se
<https://blog.crisp.se/wp-content/uploads/2012/11/SpotifyScaling.pdf>

5 8 9 10 11 12 13 14 15 33 38 40 41 102 104 110 Spotify's Failed #SquadGoals
<https://www.jeremiahlee.com/posts/failed-squad-goals/>

20 62 63 64 67 68 69 71 [인턴 H가 간다]직군없이 뒤섞이는 스쿼드 조직...요즘 스타트업은 이렇게 일한다
<https://v.daum.net/v/20230506130115973?f=p>

22 23 24 25 26 27 89 90 101 105 스쿼드(Squad) 조직이 실패하는 진짜 이유 | 애자일의 환상과 현실, 스타트업
을 위한 실전 대안

<https://smart-work.tistory.com/entry/%EC%8A%A4%EC%BF%BC%EB%93%9CSquad-%EC%A1%B0%EC%A7%81%EC%9D%B4-%EC%8B%A4%ED%8C%A8%ED%95%98%EB%8A%94-%EC%A7%84%EC%A7%9C-%EC%9D%B4%EC%9C%A0-%EC%95%A0%EC%9E%90%EC%9D%BC%EC%9D%98-%ED%99%98%EC%83%81%EA%B3%BC-%ED%98%84%EC%8B%A4-%EC%8A%A4%ED%83%80%ED%8A%B8%EC%97%85%EC%9D%84-%EC%9C%84%ED%95%9C-%EC%8B%A4%EC%A0%84-%EB%8C%80%EC%95%88>

28 29 30 31 32 [조직문화] 스쿼드(Squad)는 그만!! 201x년 스쿼드의 경험은 고통이었다. 첫 제품을 만들어야하
는... | by 김준형 | Medium

<https://medium.com/@ghilbut/%EC%A1%B0%EC%A7%81%EB%AC%B8%ED%99%94-%EC%8A%A4%EC%BF%BC%EB%93%9C-squad-%EB%8A%94-%EA%B7%B8%EB%A7%8C-738b7ebd6998>

42 43 44 45 46 47 48 49 50 ING's agile transformation | McKinsey

<https://www.mckinsey.com/industries/financial-services/our-insights/ings-agile-transformation>

51 109 Spotify Squad Models feat. Kevin Goldsmith, Former VPE of ... - Interna

<https://www.interna.com/post/spotify-squad-models-feat-kevin-goldsmith-former-vpe-of-spotify>

52 54 55 56 57 58 59 60 The two-pizza rule and the secret of Amazon's success | Amazon | The
Guardian

<https://www.theguardian.com/technology/2018/apr/24/the-two-pizza-rule-and-the-secret-of-amazons-success>

61 기술 기업의 조직 구조와 문화가 중요한 이유 - LinkedIn

<https://kr.linkedin.com/pulse/%EA%B8%B0%EC%88%A0-%EA%B8%B0%EC%97%85%EC%9D%98-%EC%A1%B0%EC%A7%81-%EA%B5%AC%EC%A1%B0%EC%99%80-%EB%AC%B8%ED%99%94%EA%B0%80-%EC%A4%91%EC%9A%94%ED%95%9C-%EC%9D%B4%EC%9C%A0-yunseo-cho-%EC%A1%B0%EC%9C%A4%EC%84%9C--fhds>

65 66 97 99 100 백엔드 개발자가 말하는 아임웹의 매력 4가지 : imweb Team | 아임웹 채용

https://team.imweb.me/blog_people/?q=YToyOntzOjEyOjRZl3b3JkX3R5cGUlO3M6MzoiYWxsIjtzOjQ6InBhZ2UiO2k6MTt9&bmode=view&idx=154470695&t=board

70 아임웹의 미래를 그리는 전략적 설계자들, Corp Dev팀

https://team.imweb.me/blog_people/?bmode=view&idx=116914710

73 우아한형제들(배달의민족)-라이브선물스쿼드 서버 개발자 - 원티드

<https://www.wanted.co.kr/wd/87533>

74 75 76 77 78 79 80 81 82 83 같은 직무, 다른 경험을 공유하는 Clan Day 살펴보기

<https://tech.kakaobank.com/posts/2405-same-but-diff-exploring-clan-day/>

84 85 86 87 88 카카오톡 조직 바꾸는 카드업계...사장 대신 팀장이 결재 | 한국경제

<https://www.hankyung.com/article/202103104111i>