



DO
THINGS
THAT
~~SCALE~~
DON'T
SCALE

Y Combinator에서 제공하는 가장 일반적인 유형의 조언 중 하나는 확장되지 않는 작업을 수행하라는 것입니다. 많은 예비 창업자들은 스타트업이 성공할 수도 있고 그렇지 않을 수도 있다고 믿습니다. 무언가를 만들어서 사용할 수 있게 만들고, 더 나은 쥐뿔을 만들었다면 사람들은 약속한 대로 길을 따라 여러분의 집 문으로 다가갈 것입니다. 그렇지 않으면 시장이 존재하지 않아야 합니다. [1]

실제로 스타트업이 성공하는 이유는 창업자가 성공하도록 만들기 때문입니다. 스스로 성장한 소수가 있을 수 있지만 일반적으로 성장하려면 일종의 추진력이 필요합니다. 좋은 비유는 자동차 엔진이 전기 시동 장치를 갖기 전에 가졌던 크랭크입니다. 일단 엔진이 작동하면 계속 작동하지만 이를 작동시키기 위해서는 별도의 힘든 프로세스가 필요했습니다.

모집

창업자가 처음에 해야 하는 가장 일반적이고 확장 불가능한 일은 사용자를 수동으로 모집하는 것입니다. 거의 모든 스타트업이 그래야 합니다. 사용자가 당신에게 오기를 기다릴 수 없습니다. 나가서 가져와야 합니다.

Stripe는 우리가 자금을 지원한 가장 성공적인 스타트업 중 하나이며 그들이 해결한 문제는 시급한 문제였습니다. 누구든지 앉아서 사용자를 기다릴 수 있었다면 바로 Stripe이었습니다. 하지만 실제로 YC 내에서는 공격적인 초기 사용자 확보로 유명합니다.

다른 스타트업을 위한 제품을 만드는 스타트업은 우리가 자금을 지원한 다른 회사에 잠재 사용자가 많으며 Stripe보다 이를 더 잘 활용한 회사는 없습니다. YC에서는 그들이 발명한 기술에 대해 "콜리슨 설치"라는 용어를 사용합니다. 좀 더 확신이 없는 창업자들은 "저희 베타 버전을 사용해 보시겠어요?"라고 묻습니다. 대답이 '예'이면 "좋습니다. 링크를 보내드리겠습니다."라고 말합니다. 그러나 콜리슨 형제는 기다리지 않았습니다. 누구든지 Stripe를 사용해 보기로 동의하면 "그럼 노트북을 주세요"라고 말하고 그 자리에서 설치해 주곤 했습니다.

창업자들이 밖으로 나가서 사용자를 개별적으로 모집하는 것을 거부하는 데는 두 가지 이유가 있습니다. 하나는 수줍음과 게으름의 조합입니다. 그들은 나가서 낯선 사람들과 이야기를 나누고 아마도 그들 대부분에게 거절당하는 것보다 집에 앉아 코드를 작성하는 것을 더 좋아합니다. 하지만 스타트업이 성공하려면 적어도 한 명의 창업자(보통 CEO)가 영업과 마케팅에 많은 시간을 투자해야 합니다. [2]

창업자들이 이 경로를 무시하는 또 다른 이유는 처음에는 절대 숫자가 너무 작아 보이기 때문입니다. 크고 유명한 스타트업이 이렇게 시작할 수는 없다고 그들은 생각합니다. 그들이 저지르는 실수는 복합 성장의 힘을 과소평가하는 것입니다. 우리는 모든 스타트업이 주간 성장률을 기준으로 진행 상황을 측정하도록 권장합니다. 사용자가 100명이면 일주일에 10% 성장하려면 다음 주에 10명을 더 확보해야 합니다. 110이 100보다 별로 나아 보이지 않을 수도 있지만, 일주일에 10%씩 계속 성장한다면 그 숫자가 얼마나 커지는지 놀라게 될 것입니다. 1년 후에는 사용자가 14,000명이 될 것입니다. 그리고 2년 뒤에는 200만 달러를 갖게 될 것입니다.

한 번에 수천 명의 사용자를 확보하면 다른 작업을 수행하게 되며 결국 성장은 둔화될 것입니다. 그러나 시장이 존재한다면 일반적으로 수동으로 사용자를 모집하는 것으로 시작한 다음 점차 덜 수동적인 방법으로 전환할 수 있습니다. [3]

Airbnb는 이 기술의 전형적인 예입니다. 시장은 시작하기가 너무 어렵기 때문에 처음에는 영웅적인 조치를 취해야 합니다. Airbnb의 경우, 이는 뉴욕에 집집마다 방문하여 새로운 사용자를 모집하고 기존 사용자가 목록을 개선하도록 돕는 것으로 구성되었습니다. YC 기간 동안 에어비앤비를 기억할 때면 그들이 롤리백을 들고 있는 모습이 떠오릅니다. 왜냐하면 그들이 화요일 저녁 식사에 참석할 때마다 항상 어딘가에서 비행기를 타고 돌아왔기 때문입니다.

취약한

에어비앤비는 지금은 막을 수 없는 거대 기업처럼 보이지만, 초기에는 약 30일 동안 나가서 사용자와 직접 소통하는 것이 성공과 실패의 차이를 만들 정도로 취약했습니다.

이러한 초기 취약성은 Airbnb의 고유한 특징이 아니었습니다. 거의 모든 스타트업은 처음에는 취약합니다. 그리고 이는 경험이 부족한 창업자와 투자자(그리고 기자와 포럼의 모든 것을 아는 사람들)가 그들에 대해 오해하는 가장 큰 것 중 하나입니다. 그들은 무의식적으로 기존 스타트업의 기준으로 애벌레 스타트업을 판단합니다. 그들은 갓 태어난 아기를 보고 "이 작은 생물이 어떤 일도 성취할 수 없다"고 결론짓는 것과 같습니다.

기자나 박식한 사람들이 귀하의 스타트업을 무시하는 것은 무해합니다. 그들은 항상 일을 잘못합니다. 투자자들이 귀하의 스타트업을 해고하더라도 괜찮습니다. 그들은 성장을 보면 마음을 바꿀 것입니다. 가장 큰 위험은 스타트업을 스스로 해산한다는 것입니다. 나는 그런 일이 일어나는 것을 보았습니다. 나는 종종 자신이 만들고 있는 것의 잠재력을 충분히 보지 못하는 창업자들을 격려해야 합니다. 빌 게이츠도 그런 실수를 했습니다. 그는 마이크로소프트를 창업한 뒤 가을학기에 하버드로 돌아왔다. 그는 오래 머물지는 않았지만 마이크로소프트가 지금보다 훨씬 작은 규모라도 될 것이라는 것을 깨달았다면 전혀 돌아오지 않았을 것입니다. [4]

초기 스타트업에 대해 물어야 할 질문은 “이 회사가 세계를 장악하고 있는가?”가 아닙니다. 하지만 “창립자들이 옳은 일을 한다면 이 회사는 얼마나 커질 수 있을까?” 그리고 올바른 일이 당시에는 힘들고 하찮아 보이는 경우가 많습니다. Microsoft는 앨버커키에서 수천 명의 애호가 시장을 위한 Basic 인터프리터를 작성하는 두 사람에게 불과했을 때 그다지 인상적으로 보일 수 없었지만, 돌이켜보면 그것이 마이크로컴퓨터 소프트웨어를 지배하는 최적의 경로였습니다. 그리고 브라이언 체스키(Brian Chesky)와 조 게비아(Joe Gebbia)는 첫 번째 호스트 아파트의 “전문적인” 사진을 찍으면서 큰 성공을 거두는 중이라는 느낌을 받지 못했다는 것을 알고 있습니다. 그들은 단지 살아남으려고 노력하고 있었을 뿐입니다. 그러나 돌이켜보면 그것 역시 큰 시장을 장악하는 최적의 길이었다.

수동으로 모집할 사용자를 어떻게 찾나요? [자신의 문제를](#) 해결하기 위해 무언가를 만든다면 동료를 찾기만 하면 되는데, 이는 일반적으로 간단합니다. 그렇지 않으면 가장 유망한 사용자 집단을 찾기 위해 더욱 신중한 노력을 기울여야 합니다. 이를 수행하는 일반적인 방법은 상대적으로 대상을 지정하지 않은 출시를 통해 일부 초기 사용자 세트를 확보한 다음 어떤 유형이 가장 열정적인지 관찰하고 그들과 유사한 사용자를 더 찾는 것입니다. 예를 들어, Ben Silbermann은 초기 Pinterest 사용자 중 다수가 디자인에 관심이 있다는 사실을 알아차리고 사용자를 모집하기 위해 디자인 블로거 컨퍼런스에 갔으며 이는 효과가 있었습니다. [5]

즐거움

사용자를 확보하는 것뿐만 아니라 사용자를 행복하게 만들기 위해 특별한 조치를 취해야 합니다. 가능한 한 오랫동안(놀랍도록 길었음) Wufoo는 각 신규 사용자에게 손으로 쓴 감사 메시지를 보냈습니다. 첫 번째 사용자는 귀하에게 가입한 것이 자신이 내린 최고의 선택 중 하나라고 느껴야 합니다. 그리고 당신은 그들을 기쁘게 할 새로운 방법을 생각하기 위해 머리를 써야 합니다.

왜 스타트업에게 이것을 가르쳐야 할까요? 창업자에게는 왜 직관에 반하는가? 내 생각에는 세 가지 이유가 있다.

하나는 많은 스타트업 창업자들이 엔지니어 교육을 받았고, 고객 서비스는 엔지니어 교육의 일부가 아니라는 것입니다. 견고하고 우아한 제품을 만들어야 하며 영업사원처럼 개인 사용자에게 맹목적으로 주의를 기울이지 않아야 합니다. 아이러니하게도 엔지니어링이 전통적으로 손으로 잡는 것을 싫어하는 이유 중 하나는 엔지니어가 전체 쇼를 운영하는 것이 아니라 건물을 만드는 좁은 영역만 담당하던 시절부터 그 전통이 이어졌기 때문입니다. Scotty라면 짜증을 낼 수 있지만 Kirk라면 그렇지 않습니다.

창업자들이 개인 고객에게 충분히 집중하지 않는 또 다른 이유는 고객이 확장되지 않을 것을 걱정하기 때문입니다. 그러나 애벌레 스타트업의 창업자들이 이에 대해 걱정할 때 나는 현재 상태에서는 잃을 것이 없다는 점을 지적한다. 아마도 기존 사용자를 매우 행복하게 만들기 위해 노력한다면 언젠가는 할 일이 너무 많아질 것입니다. 그것은 큰 문제가 될 것입니다. 당신이 그것을 실현할 수 있는지 확인하십시오. 그리고 그렇게 되면 고

객 만족이 예상보다 더 잘 확장된다는 것을 알게 될 것입니다. 부분적으로는 일반적으로 예상했던 것보다 더 큰 규모로 확장할 수 있는 방법을 찾을 수 있고, 부분적으로는 그때쯤이면 고객을 만족시키는 것이 문화에 스며들 것이기 때문입니다.

나는 스타트업이 초기 사용자를 행복하게 만들기 위해 너무 열심히 노력하여 막다른 골목에 빠지는 것을 한 번도 본 적이 없습니다.

그러나 창업자가 사용자에게 얼마나 세심한 배려를 할 수 있는지 깨닫지 못하게 하는 가장 큰 이유는 아마도 자신이 그런 관심을 경험한 적이 없다는 점일 것입니다. 고객 서비스에 대한 그들의 표준은 그들이 고객이었던 회사들에 의해 설정되었습니다. 그 회사들은 대부분 큰 회사들입니다. 팀 쿡은 노트북을 구입한 후에 손으로 쓴 메모를 보내지 않습니다. 그는 할 수 없습니다. 하지만 넌 할수있어. 이것이 바로 소규모 회사의 장점 중 하나입니다. 대기업이 제공할 수 없는 수준의 서비스를 제공할 수 있다는 것입니다. [6]

기존 규칙이 사용자 경험의 상한선이 아니라는 점을 깨닫고 나면 사용자를 만족시키기 위해 얼마나 멀리 갈 수 있는지 생각해 보는 것이 매우 흥미로울 것입니다.

경험

사용자에 대한 관심이 얼마나 극도로 높아야 하는지를 전달하는 문구를 생각하려고 하다가 스티브 잡스가 이미 그 일을 했다는 것을 깨달았습니다. 엄청나게 훌륭했습니다. 스티브는 단지 "매우"의 동의어로 "미친"을 사용한 것이 아니었습니다. 그는 말 그대로 일상생활에서 병리적인 것으로 간주될 정도로 실행의 질에 초점을 맞춰야 한다는 뜻이었습니다.

우리가 자금을 지원한 가장 성공적인 스타트업은 모두 가지고 있으며, 이는 아마도 예비 창업자들에게 놀라운 일이 아닐 것입니다. 초보 창업자들이 얻지 못하는 것은 애벌레 스타트업에서는 미친 듯이 훌륭하다는 뜻입니다. 스티브 잡스가 이 문구를 사용하기 시작했을 때 애플은 이미 확고한 기업이었습니다. 그는 Mac(및 그에 대한 문서, 심지어 포장까지도 집착의 본질)이 미친 듯이 잘 설계되고 제조되어야 한다는 것을 의미했습니다. 엔지니어가 이해하는 것은 어렵지 않습니다. 이는 견고하고 우아한 제품을 디자인하는 것의 좀 더 극단적인 버전일 뿐입니다.

창업자들이 파악하는 데 어려움을 겪는 것(그리고 스티브 자신도 파악하는 데 어려움을 겪었을 수도 있음)은 시간 슬라이더를 스타트업 생활의 처음 몇 달로 되돌릴 때 미친 듯이 훌륭하게 변하는 것입니다. 엄청나게 훌륭해야 하는 것은 제품이 아니라 사용자가 되는 경험입니다. 제품은 그 중 하나의 구성 요소일 뿐입니다. 대기업의 경우 반드시 지배적인 회사입니다. 그러나 주의를 기울여 차이를 보완한다면 초기의 불안전하고 버그가 많은 제품으로 사용자에게 미친 듯이 훌륭한 경험을 제공할 수 있고 또 제공해야 합니다.

아마도 가능하지만 그래야 할까요? 예. 초기 사용자와의 과도한 참여는 단지 성장을 위해 허용되는 기술이 아닙니다. 대부분의 성공적인 스타트업의 경우 이는 제품을 좋게 만드는 피드백 루프의 필수 부분입니다. 더 나은 쥐뿔을 만드는 것은 원자적인 작업이 아닙니다. 대부분의 성공적인 스타트업이 하는 방식으로 시작하더라도 자신에게 필요한 것을 구축함으로써 가장 먼저 구축하는 것은 결코 옳지 않습니다. 그리고 실수로 큰 불이익을 받는 영역을 제외하고는 처음에는 완벽을 목표로 하지 않는 것이 더 나은 경우가 많습니다. 특히 소프트웨어에서는 효용성이 커지자마자 사용자에게 무언가를 보여주고 사용자가 이를 사용하여 무엇을 하는지 확인

하는 것이 일반적으로 가장 효과적입니다. 완벽주의는 종종 미루기에 대한 변명이며, 어떤 경우에도 초기 사용자 모델은 항상 부정확합니다. 심지어 귀하가 그들 중 하나라고 해도 마찬가지입니다. [7]

초기 사용자와 직접 소통하면서 얻는 피드백은 역대 최고가 될 것입니다. 규모가 너무 커서 포커스 그룹에 의존해야 한다면 사용자의 집과 사무실에 가서 소수만이 있을 때처럼 사용자가 귀하의 제품을 사용하는 모습을 지켜보고 싶을 것입니다.

화재

때때로 확장 불가능한 올바른 트릭은 의도적으로 좁은 시장에 집중하는 것입니다. 이는 더 많은 통나무를 추가하기 전에 처음에 불을 억제하여 매우 뜨겁게 만드는 것과 같습니다.

페이스북이 그랬어요. 처음에는 하버드 학생들만을 위한 것이었어요. 그 형태에서는 단지 수천 명의 잠재 시장이 있었지만, 그것이 정말로 자신들을 위한 것이라고 느꼈기 때문에 그들 중 임계 질량이 등록했습니다. 페이스북은 하버드 학생들을 위한 서비스가 중단된 후에도 꽤 오랫동안 특정 대학의 학생들을 위한 서비스로 남아 있었습니다. Startup School에서 Mark Zuckerberg를 인터뷰했을 때 그는 각 학교의 강좌 목록을 작성하는데 많은 작업이 필요했지만 그렇게 하면 학생들이 이 사이트가 자연스러운 집이라고 느끼게 되었다고 말했습니다.

마켓플레이스로 설명될 수 있는 스타트업은 일반적으로 시장의 하위 집합에서 시작해야 하지만 이는 다른 스타트업에서도 작동할 수 있습니다. 임계량의 사용자를 신속하게 확보할 수 있는 시장의 하위 집합이 있는지 항상 물어볼 가치가 있습니다. [8]

봉쇄 사격 전략을 사용하는 대부분의 스타트업은 무의식적으로 이를 수행합니다. 그들은 멀리 어답터인 자신과 친구들을 위해 무언가를 만들고 나중에서야 그것을 더 넓은 시장에 제공할 수 있다는 것을 깨닫습니다. 이 전략은 무의식적으로 수행해도 잘 작동합니다. 이 패턴을 의식적으로 인식하지 못하는 가장 큰 위험은 순진하게 그 일부를 버리는 사람들에게 있습니다. 예를 들어, 자신과 친구를 위해 무언가를 만들지 않거나 구축하더라도 기업 출신이고 친구가 멀리 어답터가 아닌 경우 더 이상 완벽한 초기 시장을 제공받을 수 없습니다. 플래터.

기업 중에서 최고의 멀리 어답터는 대개 다른 스타트업입니다. 그들은 천성적으로 새로운 것에 더 개방적이며, 막 시작했기 때문에 아직 모든 선택을 하지 않았기 때문입니다. 게다가 그들이 성공하면 그들은 빠르게 성장하고 당신도 그들과 함께합니다. B2B 스타트업이 이제 수백 개의 다른 스타트업을 즉시 이용할 수 있는 시장을 갖게 된 것은 YC 모델(특히 YC를 크게 만드는 것)의 예상치 못한 많은 이점 중 하나였습니다.

Meraki [하드웨어 스타트업](#)

의 경우 "Meraki 당기기"라고 부르는 확장되지 않는 작업을 수행하는 변형이 있습니다. 비록 우리가 Meraki에 자금을 지원하지는 않았지만 창립자는 Robert Morris의 대학원생이었기 때문에 그들의 역사를 알고 있습니다. 그들은 확장이 불가능한 일, 즉 라우터 자체를 조립하는 것부터 시작했습니다. 하드웨어 스타트업은 소프트웨어 스타트업이 겪지 않는 장애물에 직면합니다. 공장 생산을 위한 최소 주문 금액은 일반적으로 수십만 달러입니다. 제품이 없으면 제품 제조에 필요한 자금을 조달하는 데 필요한 성장을 이룰 수 없습니다. 하드웨어 스타트업이 돈을 투자자에게 의존해야 했을 때, 이를 극복하려면 꽤 설득력이 있어야 했습니다. 클라우드

펀딩(더 정확하게는 사전 주문)의 등장이 많은 도움이 되었습니다. 하지만 그럼에도 불구하고 저는 스타트업들에게 가능하다면 처음에는 Meraki를 선택하라고 조언하고 싶습니다. 페블이 그랬어요. Pebbles는 처음 수백 개의 시계를 직접 [조립했습니다](#). 만약 그들이 그 단계를 거치지 않았다면 아마도 Kickstarter에 갔을 때 천만 달러 상당의 시계를 팔지 못했을 것입니다. 초기 고객에게 과도한 관심을 기울이는 것처럼 직접 제작하는 것도 하드웨어 스타트업에게는 가치 있는 일입니다. 공장에 가면 디자인을 더 빠르게 수정할 수 있고, 다른 곳에서는 결코 알지 못했던 것들을 배울 수 있습니다. Pebble의 Eric Migicovsky는 그가 배운 것 중 하나가 "좋은 나사를 조달하는 것이 얼마나 가치 있는지"라고 말했습니다. 누가 알았 겠어? 상담 때때로 우리는 B2B 스타트업 창업자에게 극도로 과도한 참여를 유도하고 단일 사용자를 선택하고 마치 해당 사용자만을 위한 무언가를 구축하는 컨설턴트인 것처럼 행동하라고 조언합니다. 초기 사용자는 금형의 양식 역할을 합니다. 그들의 요구 사항에 완벽하게 맞을 때까지 계속 조정하면 일반적으로 다른 사용자도 원하는 것을 만들었다는 것을 알게 될 것입니다. 비록 그 수가 많지 않더라도 아마도 더 많은 것을 가지고 있는 인접 영토가 있을 것입니다. 정말로 무언가를 필요로 하고 그 필요에 따라 행동할 수 있는 사용자를 단 한 명만 찾을 수 있다면 사람들이 원하는 것을 만들 수 있는 발판을 마련하게 되며 이는 초기 스타트업이 필요로 하는 만큼의 것입니다. [9]

컨설팅은 확장되지 않는 작업의 전형적인 예입니다. 그러나 (자신의 호의를 후하게 베푸는 다른 방법과 마찬가지로) 돈을 받지 않는 한 그렇게 하는 것이 안전합니다. 기업이 선을 넘는 곳이 바로 여기입니다. 당신이 고객에게 특별히 세심한 주의를 기울이는 제품 회사라면 모든 문제를 해결하지 못하더라도 고객은 매우 고마워할 것입니다. 그러나 그들이 당신에게 그 세심한 배려에 대해 특별히 돈을 지불하기 시작할 때, 즉 그들이 당신에게 시간당 지불을 시작할 때, 그들은 당신이 모든 것을 하기를 기대합니다.

초기에 미지근한 사용자를 모집하기 위한 컨설팅과 유사한 또 다른 기술은 사용자를 대신하여 소프트웨어를 직접 사용하는 것입니다. 우리는 Viaweb에서 그렇게 했습니다. 우리가 우리 소프트웨어를 사용하여 온라인 상점을 만들고 싶은지 묻는 판매자에게 접근했을 때 일부는 거절했지만 우리가 자신을 위해 하나를 만들도록 허락했습니다. 우리는 사용자를 확보하기 위해 무엇이든 할 것이기 때문에 그렇게 했습니다. 그 당시 우리는 꽤 절름발이라고 느꼈습니다. 대규모의 전략적 전자상거래 파트너십을 구성하는 대신 여행가방과 펜, 남성용 셔츠를 판매하려고 했습니다. 그러나 돌아해보면 그것은 정확히 옳은 일이었습니다. 왜냐하면 판매자가 우리 소프트웨어를 사용하는 것이 어떤 느낌인지 가르쳐 주었기 때문입니다. 때로는 피드백 루프가 거의 즉각적으로 이루어지기도 했습니다. 판매자 사이트를 구축하는 도중에 우리가 갖고 있지 않은 기능이 필요하다는 사실을 발견하고 이를 구현하는 데 몇 시간을 소비한 다음 사이트 구축을 재개했습니다.

수동

소프트웨어를 사용하는 것이 아니라 소프트웨어를 사용하는 더욱 극단적인 변형이 있습니다. 사용자 수가 적은 경우 나중에 자동화하려는 작업을 수동으로 수행하지 않아도 되는 경우가 있습니다. 이를 통해 더 빠르게 시작할 수 있으며, 마침내 루프에서 벗어나 자동화할 때 직접 수행함으로써 근육 기억을 갖게 되므로 무엇을 구축해야 할지 정확히 알 수 있습니다.

수동 구성 요소가 사용자에게 소프트웨어처럼 보일 때 이 기술은 장난스러운 측면을 갖기 시작합니다. 예를 들어, Stripe가 첫 번째 사용자에게 "인스턴트" 판매자 계정을 제공한 방식은 창립자가 배후에서 수동으로 기존 판매자 계정에 등록하는 것이었습니다.

일부 스타트업은 처음에는 완전히 수동으로 운영될 수 있습니다. 해결해야 할 문제가 있는 사람을 찾고 수동으로 해결할 수 있다면 가능한 한 오랫동안 그렇게 한 다음 점차적으로 병목 현상을 자동화하십시오. 아직 자동이 아닌 방식으로 사용자의 문제를 해결하는 것은 약간 두려울 수 있지만, 아직 누군가의 문제를 해결하지 못하는 자동 무언가를 갖는 훨씬 더 일반적인 경우보다 덜 두렵습니다.

큰

일반적으로 작동하지 않는 일종의 초기 전술인 대규모 출시를 언급해야 합니다. 나는 스타트업이 동력 항공기라기보다 발사체이며, 충분한 초기 속도로 발사된다면 성공할 수 있을 것이라고 믿는 창업자들을 가끔 만난다. 그들은 엠바고가 있는 8개의 다른 출판물을 동시에 출시하기를 원합니다. 물론 화요일에는 뭔가를 시작하기에 가장 좋은 날이라는 내용을 어딘가에서 읽었기 때문입니다.

작은 발사가 얼마나 중요한지 쉽게 알 수 있습니다. 성공적인 스타트업을 생각해 보세요. 얼마나 많은 출시를 기억하시나요? 출시부터 필요한 것은 초기 핵심 사용자뿐입니다. 몇 달 후의 성과는 사용자 수보다 해당 사용자를 얼마나 행복하게 만들었는지에 따라 달라집니다. [10]

그렇다면 창업자들은 왜 출시가 중요하다고 생각할까요? 유아론과 게으름의 결합. 그들은 자신들이 만들고 있는 것이 너무 훌륭해서 그것에 대해 듣는 모든 사람이 즉시 가입할 것이라고 생각합니다. 게다가 한 번에 한 명씩 모집하는 것이 아니라 단지 자신의 존재를 알리는 것만으로 사용자를 확보할 수 있다면 작업이 훨씬 줄어들 것입니다. 그러나 당신이 만들고 있는 것이 정말 훌륭하더라도 사용자를 확보하는 것은 항상 점진적인 과정이 될 것입니다. 부분적으로는 훌륭한 것이 일반적으로 참신하기도 하지만, 주로 사용자가 생각할 다른 사항이 있기 때문입니다.

파트너십도 일반적으로 작동하지 않습니다. 일반적으로 스타트업에는 효과가 없지만 특히 성장을 시작하는 방법으로는 효과가 없습니다. 경험이 없는 창업자들이 큰 회사와의 파트너십이 큰 기회가 될 것이라고 믿는 것은 흔히 저지르는 실수입니다. 6개월 후 그들은 모두 같은 말을 했습니다. 그것은 우리가 예상했던 것보다 훨씬 더 많은 일이었고 결국 우리는 그 일에서 거의 아무것도 얻지 못했습니다. [11]

처음에 뭔가 특별한 일을 하는 것만으로는 충분하지 않습니다. 처음에는 엄청난 노력을 기울여야 합니다. 사용자를 확보하기 위한 대규모 출시를 기대하든 큰 파트너를 기대하든 노력을 생략하는 모든 전략은 사실상 의심스럽습니다.

Vector

시작하기 위해 확장할 수 없을 정도로 힘든 작업을 수행해야 하는 필요성은 거의 보편적이므로 스타트업 아이디어를 스칼라로 생각하지 않는 것이 좋습니다. 대신 우리는 이를 구축할 항목과 회사를 운영하기 위해 처음에 수행할 확장 불가능한 항목의 쌍으로 생각해야 합니다.

스타트업 아이디어를 이런 식으로 보는 것은 흥미로울 수 있습니다. 이제 두 가지 구성 요소가 있으므로 첫 번째와 두 번째에 대해 상상력을 발휘할 수 있기 때문입니다. 그러나 대부분의 경우 두 번째 구성 요소는 일반적으로 사용자를 수동으로 모집하고 그들에게 압도적으로 좋은 경험을 제공하는 것입니다. 스타트업을 벡터로 취급하는 주요 이점은 창업자에게 두 가지 차원에서 열심히 일해야 함을 상기시키는 것입니다. [12]

가장 좋은 경우에는 벡터의 두 구성 요소가 모두 회사의 DNA에 기여합니다. 시작하기 위해 수행해야 하는 확실할 수 없는 일은 단순히 필요악이 아니라 회사를 더 나은 방향으로 영구적으로 변화시키는 것입니다. 규모가 작을 때 사용자 확보에 대해 공격적이어야 한다면 규모가 커져도 여전히 공격적일 것입니다. 자신의 하드웨어를 제조해야 하거나 사용자를 대신하여 소프트웨어를 사용해야 한다면 다른 방법으로는 배울 수 없었던 것들을 배우게 될 것입니다. 그리고 가장 중요한 것은, 사용자가 소수일 때 사용자를 기쁘게 하기 위해 열심히 일해야 한다면, 사용자가 많을 때에도 계속 그렇게 할 것이라는 점입니다.

참고

[1] 실제로 에머슨은 쥐뿔에 대해 구체적으로 언급한 적이 없습니다. 그는 "만약 어떤 사람이 팔 만한 좋은 옥수수나 나무, 판자, 돼지를 가지고 있고, 다른 누구보다 더 좋은 의자나 칼, 도가니, 교회 오르간을 만들 수 있다면, 당신은 그의 집으로 가는 넓고 험난한 길을 찾을 것입니다. , 비록 숲 속에 있을지라도."

[2] 이를 명시적으로 제안한 Sam Altman에게 감사드립니다. 그리고 아니요, 판매를 대신해 줄 사람을 고용하여 판매를 피할 수는 없습니다. 처음에는 판매를 직접해야 합니다. 나중에 당신을 대신할 실제 영업사원을 고용할 수 있습니다.

[3] 이것이 효과가 있는 이유는 몸이 커질수록 크기가 성장하는 데 도움이 되기 때문입니다. Patrick Collison은 "어느 시점에서 Stripe의 느낌에 매우 눈에 띄는 변화가 있었습니다. 그것은 우리가 밀어야 했던 이 바위에서 실제로 자체 추진력을 가진 기차 차량으로 바뀌었습니다."라고 썼습니다.

[4] YC가 창업자를 도울 수 있는 가장 미묘한 방법 중 하나는 그들의 야망을 조정하는 것입니다. 왜냐하면 우리는 많은 성공적인 스타트업이 막 시작했을 때의 모습을 정확히 알고 있기 때문입니다.

[5] 소규모 사용자 집단이 쉽게 관찰할 수 없는 것(예: 엔터프라이즈 소프트웨어)을 구축하고 있고 연결이 없는 도메인에서 구축하는 경우 콜드 콜 및 소개에 의존해야 합니다. . 하지만 그런 아이디어를 연구해야 할까요?

[6] Garry Tan은 창업자들이 처음에 빠지는 흥미로운 함정을 지적했습니다. 그들은 크게 보이고 싶어 개인 사용자에게 대한 무관심과 같은 대기업의 결점까지도 모방합니다. 이것은 그들에게 더 "전문적"인 것 같습니다. 실제로 자신이 작다는 사실을 받아들이고 그에 따른 이점을 활용하는 것이 더 좋습니다.

[7] 사용자 모델은 완벽하게 정확할 수 없습니다. 사용자의 요구 사항은 개발자가 구축한 항목에 따라 변경되는 경우가 많기 때문입니다. 마이크로컴퓨터를 구축하면 갑자기 스프레드시트를 실행해야 합니다. 새 마이크로컴퓨터가 출시되면 누군가가 스프레드시트를 발명하게 되기 때문입니다.

[8] 가장 빨리 가입할 하위 집합과 가장 많은 비용을 지불할 하위 집합 중에서 선택해야 한다면 일반적으로 전자를 선택하는 것이 가장 좋습니다. 왜냐하면 이들이 아마도 멀리 어답터일 것이기 때문입니다. 그들은 귀하의 제품에 더 나은 영향을 미칠 것이며 귀하가 판매에 많은 노력을 기울이게 만들지 않을 것입니다. 그리고 자금이 적더라도 초기에 목표 성장률을 유지하는 데 그렇게 많은 돈이 필요하지 않습니다.

[9] 네, 한 명의 사용자에게만 정말 유용한 것을 만들 수 있는 경우를 상상할 수 있습니다. 그러나 이는 경험이 없는 창업자에게도 대개 분명합니다. 그러므로 당신이 하나의 시장을 위해 무언가를 만들 것인지가 확실하지 않다면 그 위험에 대해 걱정하지 마십시오.

[10] 출시 규모와 성공 사이에는 역의 상관관계가 있을 수도 있습니다. 내가 기억하는 유일한 출시는 Segway 및 Google Wave와 같은 유명한 실패작입니다. Wave는 특히 놀라운 예입니다. 왜냐하면 내 생각에 Wave는 지나친 출시로 인해 부분적으로 사라졌기 때문에 사실 좋은 아이디어였기 때문입니다.

[11] 구글은 야후에 힘입어 크게 성장했지만, 그것은 파트너십이 아니었습니다. 야후는 그들의 고객이었습니다.

[12] 또한 두 번째 구성 요소가 비어 있는 아이디어(예: 수동으로 모집할 사용자를 찾을 방법이 없기 때문에 시작하기 위해 할 수 있는 것이 아무것도 없는 아이디어)는 아마도 나쁜 아이디어일 수 있다는 사실을 창업자들에게 상기시켜 줄 것입니다. 적어도 그 창업자들에게는 말이죠. 초안을 읽어주신 Sam Altman, Paul Buchheit, Patrick Collison, Kevin Hale, Steven Levy, Jessica Livingston, Geoff Ralston 및 Garry Tan에게

감사드립니다 .