

## 관리자가 관리하지 않는 이유: 기업 시스템의 해석적 사례 연구 tems 입양 관리

세실리 캐롤라인 팔치  
코펜하겐 IT 대학교  
[세파@이투.dk](mailto:세파@이투.dk)

올리버 크랜처  
코펜하겐 IT 대학교  
[올리크@이투.dk](mailto:올리크@이투.dk)

### 추상적인

채택 및 혜택 관리에 대한 연구는 많지만, 관리자가 엔터프라이즈 시스템 채택을 어떻게, 왜 관리하는지에 대해서는 놀랍게도 거의 알지 못합니다. 해석적 사례 연구를 수행하여 채택 관리에 대한 관리자의 인식을 살펴봅니다. 반구조화된 인터뷰를 통해 두 단계의 관리자가 애플리케이션이 라이브로 전환되면 원하는 혜택에 대한 사용자의 채택에 영향을 미치는 조치를 놀랍게도 거의 표현하지 않았다는 사실을 발견했습니다. 데이터 분석 결과, 관리자의 (부족한) 채택 조치를 정당화하는 정신적 합리화 패턴이 발견되었습니다. 이러한 명시적이고 수행적인 구조를 프로젝트 중심적 사고 방식으로 식별하여 프로젝트 완료의 엄격한 경계가 어떻게 우선순위가 매겨지고, 따라서 전체적인 채택 관리에 도전하는지 강조합니다.

**키워드:** 채택 관리, 사례 연구, 프로젝트 사고, 관리 활동, 관리 정당성.

### 소개

사용자 채택에 대한 연구는 의심할 여지 없이 정보 시스템(IS) 연구에서 가장 많이 언급되는 문헌 주제 중 하나입니다. 그러나 조직은 여전히 원하는 혜택을 위해 채택을 관리하는 데 어려움을 겪고 있습니다. IT 프로젝트에 할당된 리소스는 계속 증가하고 있지만 보고서에 따르면 평균적으로 예상보다 56% 적은 가치를 반환합니다(Bloch et al., 2012). 조직이 사용자 채택을 관리할 때 직면하는 과제를 이해하기 위해서는 자발적 사용 환경과 의무적 사용 환경을 구별해야 합니다. 저희 연구는 엔터프라이즈 소프트웨어의 의무적 사용 맥락에 초점을 맞춥니다. 의무적 환경은 자발적 사용 맥락과 상당히 다른 것으로 나타났습니다. 소프트웨어가 자발적일 경우 사용자 태도와 사용 의도는 실제 사용과 상관관계가 있는 것으로 입증되었지만 기업용 소프트웨어는 조직에서 의무화되어 있으므로 의도는 설명력이 거의 없습니다(McNally & Griffin, 2010; Bhattacharjee et al., 2018; Beaudry et al., 2020; Nah et al., 2004; Brown

et al., 2002; Koh et al., 2010). 따라서 엔터프라이즈 소프트웨어의 경우 사용자 채택은 다른 관리 접근 방식을 요구하는 보다 모호하고 역동적인 행동을 구현합니다(Seo et al., 2011; Van Offenbeek et al., 2013). Mintzberg(1971)의 관리 작업에 대한 기초 연구는 관리자의 작업 프로세스를 이해하여 개선하는 것이 필수적임을 보여주었습니다. 사용자 채택(Venkatesh et al., 2016; Beaudry & Pinsonneault, 2010)과 혜택 관리(Ward et al., 1996; Ashurst et al., 2008; Holgeid et al., 2022)에 대한 방대한 양의 문헌에도 불구하고 관리자가 의무적 환경에서 채택을 관리하는 방법을 이해하기 위한 연구는 비교적 적다는 것은 놀라운 일입니다. 궁극적으로 이는 관리자가 엔터프라이즈 소프트웨어 채택을 관리할 때 무엇을 하는지에 대해 더 많이 알아야 한다는 것을 의미합니다.

이 사례 연구는 입양 관리에 대한 인식을 이해하고 실무에서 관리가 직면한 과제를 밝혀야 할 필요성에서 비롯됩니다. 이는 또한 입양 관리의 경계를 더 탐구하고 현재 장애물을 설명하는 것입니다. 이 사례 연구는 다음과 같은 연구 질문을 탐구합니다. (1) *관리자들은 입양 관리와 입양 관리에 대한 자신의 조치를 어떻게 생각합니까?* 그리고 (2) *관리자는 채택 관리 활동을 어떻게 합리화하고 정당화할까요?* 이 논문의 나머지 부분에서는 채택, 혜택 관리, 조직적 일상에 대한 문헌을 검토하고, 세 조직에서 수행한 해석적 사례 연구의 방법과 결과를 설명하고, 함의를 논의합니다. 우리의 주요 기여는 관리자가 채택 관리에 제한적인 노력을 기울일 수 있는 이유를 설명하는 관리적 해석과 합리화를 식별하는 데 있습니다.

### 배경 문헌

문헌에서 입양 관리에 대한 명확한 초점은 거의 없으며 IS 용어로 현상을 한정할 필요가 있습니다. 최근 문헌(Falch & Krancher, 2022)에 따라 우리는 입양을 시스템이 주어진 문제를 가치 있는 결과로 해결하는 사용을 가능하게 할 때로 정의합니다. 우리는 이 정의의 측면을 교차하는 두 가지 광범위한 문헌 스트림을 구별합니다.

수용 및 저항에 관한 문헌과 프로젝트 및 혜택 관리에 관한 문헌.

## 프로젝트 및 혜택 관리

학문으로서의 혜택 관리(Benefits management)는 Ward et al.(1996)의 초기 연구에 크게 기반을 두고 있으며, 그들은 이후 연구의 많은 부분을 형성한 혜택 관리 프로세스 모델을 제시했습니다. 이 논문은 IS/IT가 그 자체로 혜택을 제공하지 않지만, 조직이 포괄적인 관리 프로세스를 통해 IT 투자에서 원하는 혜택을 실현할 수 있다고 주장합니다. 반복적 모델의 단계에는 혜택 식별, 실현 계획, 계획 실행, 추가 혜택의 잠재력 평가 및 분석이 포함됩니다(Ward et al., 1996). Project Management Institute와 같은 학자와 전문 조직은 이 모델을 기반으로 구조화되고 종종 반복적인 접근 방식의 중요성을 뒷받침하는 유사한 프레임워크를 개발했습니다(PMI, 2016; Peppard et al., 2000; Holgeid et al., 2022; Ashurst et al., 2008). 혜택 관리 문헌에서 공통적으로 나타나는 패턴은 논문이 대체로 처방적이고 관리자가 시스템 사용과 비즈니스 혜택을 일치시키기 위해 사용자 응답을 어떻게, 왜 관리해야 하는지에 초점을 맞추지 않는다는 것입니다. 따라서 도입에 초점을 맞추지 않습니다. 게다가 프로젝트를 시간 제한이 있는 일회성 활동의 집합으로 보는 개념(Cadle & Yeates, 2008; Ashurst et al., 2008)에 따라 프로젝트 및 혜택 관리 문헌은 "프로젝트 구현이 진행되고 완료됨에 따라 혜택이 실현"되도록 하는 프로세스와 관행을 강조하지만, 일부 프레임워크는 프로젝트가 종료된 후에도 혜택을 유지하는 것의 중요성을 인정합니다(PMI, 2016).

## 수용과 저항

수용 및 저항 문헌으로 특징지을 수 있는 것에 대해, 이 분야의 연구는 수용을 설명하거나(Davis, 1987; Ginzberg 1981; Beaudry & Pinsonneault, 2010) 사용자의 저항을 줄이는 것을 목표로 한다는 것이 일반적으로 알려져 있습니다(Hirschheim & Newman, 1988; Kim & Kankanhalli, 2008). 최근 연구에서는 의무적 시스템 사용의 복잡하고 역설적인 사용자 행동을 수용하기 위해 수용과 저항을 동시에 연구했습니다. 이 연구에서는 조직이 수용을 얻음으로써 사용자 요구 사항을 충족시키는 것이 보장된다는 개념에 도전하고 있습니다(Beaudry et al., 2020; Bhattacharjee et al., 2018). 대부분의 문헌은 다양한 매개체에 대한 설명적 및/또는 예측적 성격을 가지고 있습니다(예: 인식된 전환 비용(Polites & Karahanna, 2012)은 시스템 사용과 사용자 행동을 일치시키기 위해 주장됩니다. 그러나 이 연구는 일부 관리 개입(예: 교육, 의도적인 변화 관리 이니셔티브)을 지적하지만

입양을 촉진하기 위한 노력에도 불구하고, 이 연구에서는 관리자가 입양을 관리하기 위해 취하는 모든 조치를 조사하지 않았으며, 관리자가 이러한 조치를 취하는 이유도 조사하지 않았습니다.

이 두 문헌 흐름의 본질을 관찰하는 동안 세 가지 지식 격차가 존재한다는 것이 분명해졌습니다. 첫째, 혜택 관리 문헌이 시스템 구현과 비즈니스 혜택을 정렬하는 데 실질적인 관리 측면을 제공하는 반면, 채택 문헌은 시스템 사용과 사용자 행동을 정렬하는 방법에 대한 설명적이고 예측적인 모델을 제공합니다. 따라서 각 문헌은 채택 관리 개념에 대한 통찰력을 제공하지만 시스템 구현, 사용자 행동 및 비즈니스 혜택을 전체적으로 연결하는 누락된 연결 고리가 있는 것 같습니다. 둘째, 관리자가 채택을 관리하기 위해 취하는 구체적인 조치와 이러한 조치를 취하는 이유에 대해 거의 알지 못합니다. 규범적 프레임워크의 가치에도 불구하고 이러한 프레임워크를 구현하는 조직은 관리자가 이러한 프레임워크를 사용하거나 실제로 채택을 관리하기 위해 다른 조치를 취할 수 있는 이유를 이해하지 못하면 예상 결과를 얻지 못할 수 있습니다. 셋째, 두 문헌 흐름 모두 프로젝트 구현 단계 동안의 구조와 활동에 초점을 맞추어 프로젝트 관점을 취합니다. 그러나 조직이 프로젝트 조직에서 제품 조직으로 전환함에 따라(Wiedemann et al., 2020) 이러한 프로젝트 관점은 점점 더 어려움을 겪고 있습니다. 또한 환경은 변할 수 있으며 시스템의 수명 동안 시스템에서 가치를 창출할 수 있는 새로운 기회가 발생할 수 있다는 것을 깨닫게 됩니다(Strong et al., 2014; Orlikowski, 1996).

## 조직 루틴

데이터를 수집하고 분석하면서 루틴 이론(Feldman and Pentland, 2003)이 새로운 개념을 이해하고 구성하는 데 유용한 렌즈로 등장했습니다. 루틴은 조직에서 반복적이고 상호 의존적인 행동의 패턴입니다. 따라서 관리자의 채택 모니터링 및 후속 조치 행동은 이러한 행동에 반복적인 패턴이 있는 한 루틴으로 볼 수 있습니다. Feldman과 Pentland는 이론에서 Latour(1986)의 실천의 명시적 측면과 수행적 측면을 구분한 것에 기반을 둡니다. 명시적 측면이 루틴에 대한 아이디어를 나타내는 반면, 수행적 측면은 행위자가 루틴의 구체적인 사례에서 취하는 구체적인 행동을 나타냅니다(Feldman and Pentland, 2003). Latour(1986)와 인류지학적 연구(Feldman, 2000)를 바탕으로 Feldman과 Pentland는 루틴의 안정성과 변화 이유를 이론화합니다. 그들은 명시적 측면이 수행적 측면을 정당화하고 인도하는 반면, 수행적 측면은 명시적 측면을 창조하고, 강화하고, 수정한다고 주장한다.

## 방법

우리는 관리자가 입양 관리와 관행과 관련된 행동을 어떻게 인식하는지 이해하기 위해 해석적 사례 연구 (Walsham, 1995) 접근 방식을 사용했습니다. 우리는 해석적 접근 방식을 선택했는데, 인터뷰 대상자의 작업에 대한 주관적 이해에 깊이 내재된 표현과 인지 패턴의 출현을 탐구할 수 있었기 때문입니다. 게다가 우리는 연구 방법으로 사례 연구를 선택했는데, 이는 잘 이해되지 않는 문제를 탐구하는 데 매우 적합했기 때문입니다(Walsham, 1995).

우리는 다음 기준을 충족하는 사례를 선택했습니다. (1) 회사는 1,000명 이상의 직원을 고용해야 합니다. (2) 회사는 100명 이상의 사용자가 있는 엔터프라이즈 소프트웨어 애플리케이션에 대한 통찰력을 제공할 준비가 되어 있어야 합니다. (3) 애플리케이션은 1년 이상 라이브 상태여야 합니다. 이를 통해 대규모 기업으로 간주될 수 있는 사례 회사를 분석하고 소프트웨어가 많은 수의 사용자에게 작동하고 관리자가 구현 후 채택에 영향을 미칠 수 있는 충분한 시간이 있는 구현 프로젝트를 논의할 수 있었습니다. 이 연구는 이러한 기준을 준수하는 3개의 덴마크 회사와 다릅니다. 3개 회사의 각 직원 수는 약 1,500명에서 약 60,000명입니다. 3개 애플리케이션 각각의 사용자 수는 약 300명에서 약 16,000명입니다. 첫 번째 회사는 덴마크의 대형 소매 회사입니다. 우리는 최근 완료된 제품 반품 시스템 구현을 사례 프로젝트로 선택했습니다. 두 번째 회사도 덴마크에서 설립되었지만 대규모 국제 판매 및 생산 체인을 보유하고 있습니다. 제조 회사로서, 그들은 새로운 기계 유지 관리 시스템의 대규모 구현 사례를 제안했습니다. 마지막으로, 해운 산업에서 80척 이상의 선박이 운영되고 있는 세 번째 회사의 사례 프로젝트는 도입에 장애가 계속되는 장기적으로 운영되어 온 기업 조달 시스템에서 벗어났습니다. 세 사례 모두에서, 우리는 구현에 관여했거나 현재 사례 프로젝트의 소프트웨어 애플리케이션을 담당하고 있는 두 단계의 경영진을 인터뷰했습니다. 이러한 역할에는 최고 경영진과 제품 소유자가 포함되었습니다. 나아가, 우리는 경영진의 인식과 사용자의 현실 간의 일치를 비교하기 위해 각 애플리케이션에 대해 한 명의 사용자를 인터뷰했습니다. 각 특정 직함은 회사마다 다르지만, 우리는 이러한 보다 개별적인 범주를 활용하여 계층적 순서를 설명하고 여전히 익명성을 유지합니다.

인터뷰에 앞서 참가자들에게 연구의 초점은 사용자 반응, 채택 결과, 채택 및 원하는 혜택의 실행 가능한 관리를 이해하는 것이라고 말했습니다. 참가자들은 채택과 혜택에 대한 관리라는 주제에 대한 명확한 관심이 있었기 때문에 기여하도록 동기를 부여받았습니다.

그들은 그 주제가 매우 관련성이 있다고 느꼈고 자신의 경험을 되돌아보고 싶어했습니다.

## 데이터 수집

저희의 데이터 소스는 반구조화된 초점 질문 (Charmaz, 2006)을 사용한 인터뷰였습니다. 저희는 프로젝트의 상태와 인터뷰 대상자가 프로젝트를 어떻게 인식하는지에 대한 질문으로 시작했습니다. 반구조화된 인터뷰 방식을 따르면서 일부 질문은 다양했지만 핵심 목표는 동일하게 유지되었습니다. 즉, 관리 조치와 그러한 조치를 주도하는 주관적 구조를 이해하는 것입니다. 주요 인터뷰 질문은 표 1에 나와 있습니다. 저희 연구의 초점이 관리자의 해석과 합리화에 맞춰졌기 때문에 보관 데이터와 같은 대체 데이터 수집 전략보다 인터뷰를 주요 데이터 소스로 선호했습니다.

사용자가 애플리케이션을 수락했다고 말할 수 있습니까? - 왜?
사용자가 애플리케이션 사용 방법을 준수한다고 말할 수 있습니까? - 왜?
신청이 채택되었다고 말할 수 있나요? - 어떻게 알 수 있나요?
구현 후 채택을 관리하기 위해 어떤 조치를 취했습니까?

표 1: 주요 인터뷰 질문 개요

## 데이터 분석

우리는 참가자의 경험과 그 경험의 의미와 행동에 대한 분석적 의미를 파악하기 시작하면서 근거 이론 (Charmaz, 2006)에서 알려진 데이터 분석 기술과 코딩 전략을 분석에 적용했습니다. 첫째, 모든 인터뷰는 필사되었고, 덴마크어로 진행된 인터뷰는 저자가 영어로 필사했습니다. 둘째, 결과는 초기 코딩과 집중 코딩 단계를 통해 밝혀졌습니다. 첫째, 채택이 어떻게 인식되었는지, 라이브로 전환한 후 관리가 애플리케이션 사용에 영향을 미치기 위해 적용한 조치는 무엇이었던지, 성공했는지 여부에 대한 주장을 어떻게 측정하거나 확인했는지 이해하는 데 도움이 되는 인용문 섹션을 코딩했습니다. 이러한 인용문 섹션 내에서 주제 패턴이 나타나기 시작했고, 집중 코딩으로 전환했습니다. 여기서 가장 유용한 초기 코드를 적용하고 모든 인터뷰 필사본에 일관되게 적용하기 시작했습니다. 코드에는 다음이 포함되었습니다. *사용은 채택이고, 사용 없이는 일이 없고, 측정 실수가 없고, 구조화된 접근 방식이 없고, 소식이 없으면 좋은 소식이 없고, 관리 정당성, 프로젝트 사고, 비교할 기준이 없고, 그냥 놓아두고 살아가고, 자신이 옳다는 것을 어떻게 알 수 있습니까?* 그리고 *상태 체크인*. 우리가 이러한 코드를 통합하기 시작하면서 일부는 더욱 얹히고 결합될 수 있었고, 두 가지 자연스러운 범주가 나타났습니다. 이러한 범주를 분석하면서 우리는

경영진이 구현 후 적용한 구체적인 조치가 다음과 같다는 것을 발견했습니다. *관리 조치* 코드에 표현된 내용은 다음과 같습니다. *측정 실수, 구조화된 접근 방식이 없음* 그리고 *살아 있어, 놓아줘*. 반면에 우리는 이러한 행동의 합리화가 포괄적인 용어에 속할 수 있다는 것을 발견했습니다. *경영 정당성* 그리고 다음 코드를 포함합니다: *사용법은 채택이다, 사용하지 않으면 일이 없습니다*. 그리고 *소식이 없다는 건 좋은 소식이다*. 집중적인 코딩을 한 후, 우리는 이론적 코딩으로 옮겼습니다(Charmaz, 2006). 이 단계에서, 라투르의 명시적과 수행적 구분은 우리의 발견에 대한 이론적 설명(Bygstad et al., 2016)으로 나타났습니다. 다음 섹션에서 우리는 이러한 발견을 자세히 제시합니다.

## 결과

세 가지 사례에서 관리자들은 구현 후 채택을 관리하기 위해 놀랍게도 거의 조치를 취하지 않았다고 표현했습니다. 데이터에 대한 귀납적 검토 결과, 관리자들은 채택에 대한 인지적 인식 패턴을 기반으로 이니셔티브 부족을 정당화했습니다. 조사 결과에 따르면, 최고 경영진과 제품 소유자 모두 엔터프라이즈 소프트웨어 채택에 대한 선행적이고 엄격한 인식으로 인해 정의된 문제를 해결하는 시스템 사용을 보장하기 위해 사용자 채택에 영향을 미치려는 집중적인 시도를 정확히 파악하는 데 어려움을 겪고 있습니다. 데이터 분석 결과, 채택에 대한 정신적 인식과 채택 중심 조치는 합리화 프로세스를 통해 상호 의존적인 관계가 존재하는 것처럼 보이기 때문에 밀접하게 얽혀 있음을 보여주었습니다. 코드 범주, 해당 하위 코드 및 이들 간의 상호 관계는 아래 그림 1에 나와 있습니다.

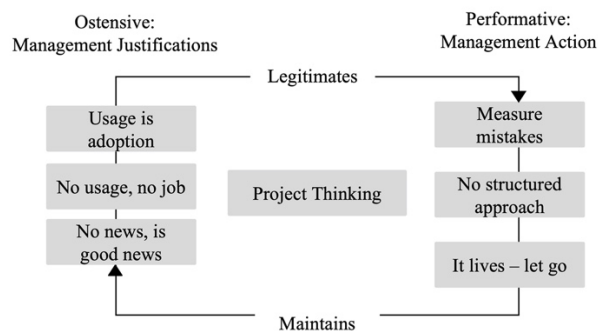


그림 1: 관리 정당화 및 관리 조치로 설명된 프로젝트 사고의 개념적 모델

이 두 범주의 본질을 살펴보면, 우리는 Feldman과 Pentland(2003)의 루틴 이론을 활용하는 것이 적절하다고 생각했습니다. 우리는 관리자의 구체적인 행동(즉, 수행적 측면)과 이러한 행동을 정당화하는 인지(즉, 명시적 측면) 사이에 자연스러운 상호 의존성이 있다고 주장합니다. 우리는 경영진이 채택을 인식하는 방식과

그에 따라 행동하는 것은 프로젝트 사고 사고방식으로 특징지어질 수 있습니다. 이 라벨을 활용함으로써 우리는 인터뷰에서 프로젝트 완료가 최종 제품보다 우선시되고 그것이 창출한 가치를 평가하는 것처럼 보이는 방식에 대한 의미를 부여합니다. 우리는 모델을 읽을 때 코드 범주 전체에서 정당화 및 유지 구조에 지속적인 변화가 있다고 주장합니다. 예를 들어, 선행 관계를 설명하지 않습니다. *사용법은 채택이다* 그리고 *실수를 측정합니다*. 대신, 우리는 예를 들어 코드를 강조하고 싶습니다. *사용법은 채택이다* 또한 코드를 교차 합법화합니다. *구조화된 접근 방식이 없습니다*. 이러한 횡단적 개념을 강조하여 채택 관리가 인식되는 방식의 사회 기술적 복잡성을 강조하는 것이 중요합니다. 다음 섹션에서는 각 코드의 기초와 모델의 두 측면 간의 역학이 인터뷰 중에 어떻게 입증되었는지 설명하고 프로젝트 사고라고 부르는 것으로 특징지을 수 있습니다.

## 경영진의 정당성

우리가 모델의 첫 번째 부분에 초점을 맞추면서, 프로젝트 사고의 명백한 측면은 경영 정당화가 무엇인지에 대한 인식을 형성합니다. 프로젝트 사고는 경영이 그들의 (부족한) 행동을 정당화하는 다양한 방식으로 묘사됩니다.

첫째, 경영진 사이에서는 애플리케이션이 라이브로 전환되고 프로젝트가 종료되고 사용자가 시스템을 활용하면 애플리케이션이 구현되고 채택된다는 인식이 두드러졌습니다. 구현과 채택의 구분은 의미론 측면에서 명확하지 않았지만 실제 사용은 채택을 의미한다는 것은 분명했습니다. 이러한 인용문은 다음과 같이 코딩되었습니다. *사용법은 채택이다*.

*저희는 이 시스템을 사용하고 있으며, 37,000건의 사례가 시스템을 통해 처리되고 있다는 사실은 이 시스템이 제대로 활용되고 있다는 분명한 증거라고 생각합니다.*(소매, 최고 경영진).

*기본적으로, 우리가 그것을 살펴본 유일한 방법은 전체 사용자 중 실제로 시스템을 사용하는 사용자 수였습니다. 새 시스템에서 실제로 수행되는 거래 수, 활동 수는 이전 시스템과 비교했을 때 얼마입니까? 다른 측정 기준은 없습니다.*(제조업, 제품 소유자).

이러한 인용문은 채택에 대한 매우 엄격한 인식을 제공하며, 잠재적으로 원하는 채택의 실현을 방해하는 시스템의 단순한 운영을 넘어서지 않습니다. 채택을 위한 적절한 측정 기술의 부족은 또한 관리자가 채택 실현을 향한 채택에 의미 있게 영향을 미치는 데 어려움을 겪는 것처럼 보였습니다.

*그리고 구체적으로, 우리는 여전히 제품을 제공하고 있으며 공급망은 운영되고 있습니다. 그것이 어떻게 작동하는지에 대한 매우 구체적인 예입니다.*(제조, 상위 경영진)

사용자들이 생각하지 않는다고 생각합니다. 저는 시스템을 사용하면서 정말 많은 것을 얻습니다. 그들은 시스템에서 무엇을 해야 하는지 생각합니다. 그래서 그들은 그것이 자신의 업무 기능에 어떻게 필요한지 알 수 있고 그런 식으로 가치를 얻는다고 생각합니다. 하지만 그들은 그것으로 인해 풍요로워진다고 느끼지 않는다고 생각합니다.(운송, 최고 경영진)

인용문에서 알 수 있듯이, 경영진의 초점은 운영 및 기능적 사용에 있었고 더 큰 비즈니스 혜택이나 사용자 가치에 전체적으로 연결되지 못했습니다. 이러한 측면을 고려하지 않으면 회사는 표면화하기 어려운 모호한 행동을 간과할 위험이 더 큼니다.

우리 데이터에서 나타난 또 다른 관리 정당성은 다음과 같습니다. *사용하지 않으면 일이 없다.* 이는 경영진이 주어진 애플리케이션을 채택하려는 사용자의 자발적인 선택이 부족하여 채택 조치에 어떤 영향을 미쳤는지 표현한 경우를 분류하는데 사용되었습니다. 이러한 인용문에는 다음이 포함되었습니다.

솔직히 말해서, 그들이 신청을 수락하지 않으면 그들은 자신의 업무를 수행할 수 없습니다. 그리고 그들이 업무를 수행하지 않으면, 저는 그것에 대해 듣게 될 것입니다. 왜냐하면 그러면 무언가 잘못되었을 것이기 때문입니다. 그러면 우리는 도움을 받지 못하는 고객들을 갖게 될 것입니다.(소매, 제품 소유자).

제 생각엔 그들이 이 애플리케이션을 채택한 건 자발적인 선택이 사실상 없기 때문입니다. 결국 회사가 결정하고 당신은 여기 있는 대가로 돈을 받고 있고, 그것은 당신이 따라야 할 몇 가지 프로세스가 있다는 것을 의미합니다. 그리고 이것이 그 중 하나입니다.(제조, 최고 경영진).

인용문은 직원이 회사의 운영 절차를 준수해야 한다는 논리를 따릅니다. 이러한 선행적 사용 아이디어를 따르면 사용자의 관점을 얻는 것이 흥미로웠습니다. 여기서 우리는 그들이 자신의 사용이 회사의 원하는 혜택과 일치한다고 생각하는지 물었습니다.

그들이 우리에게 시스템을 어떻게 사용하기를 원하는지에 대한 공통된 이해가 부족해서, 우리가 실제로 그들이 원하는 방식으로 시스템을 사용할 수 없다고 생각합니다. (소매, 사용자).

네, 우리는 선택의 여지가 없기 때문에 신청을 수락했습니다. 이 신청은 우리가 실행하는 사업적 결정입니다. 그리고 여러분은 우리가 사용하는 모든 시스템에 대한 의견을 가질 수 있는 것과 같은 방식으로 그것에 대한 의견을 가질 수 있습니다. 그리고 그것이 좋은 나쁜 의견을 가질 수 있습니다. 하지만 사실은, 여러분이 여기에서 일하고 싶다면, 그것은 우리 모두가 일하는 이 시스템입니다. 그리고 여러분이 그것을 좋아하지 않는다면, 다른 일할 방법은 없습니다.(배송, 사용자).

여기서 배송 사용자의 경험이 분명해졌습니다. *사용하지 않으면 일이 없다* 문제가 있는 사고방식으로 간주되었지만, 경영진의 정당화로서 그것은 사용자에게 스며들어 기업 소프트웨어가 회사 운영의 광범위한 부분이므로 그들의 의견이 중요하지 않다는 이해로 나타났습니다. 소매 사용자는 계속해서

사용자 행동에 대한 공통적이고 통합된 이해가 부족하여 가치 있는 사용이 방해받았다는 점을 강조하면서 동일한 종류의 문제화를 했습니다. 인터뷰 동안 최고 경영진과 제품 소유자는 채택 조치에 대한 자세한 설명과 해당 조치의 정당성을 번갈아가며 설명했습니다. 이 과정에서 우리는 가장 두드러진 인지 패턴 중 하나가 우리가 *소식이 없다는 것은 좋은 소식이다*. 여기서 경영진은 사업이 여전히 운영 중이며 이를 방해할 수 있는 주요 문제는 주의를 환기할 것이라고 주장함으로써 희소한 채택 관리 조치를 합리화했습니다. 따라서 생산에 대한 단일 수준 및 선행적 아이디어로서 상황화된 사용.

우리는 반드시 나가서 많은 질문을 하는 것은 아닙니다. 우리는 아마도 더 고전적인 접근 방식을 가지고 있을 것입니다. 만약 아무도 충분히 좋지 않은 것에 대해 큰 소리로 외치지 않는다면, 우리는 아마도 괜찮았을 것입니다. 그것이 우리가 택한 접근 방식입니다.(소매, 제품 소유자).

사실, 반응은 매우 조용했습니다. 솔직히 말해서, 우리는 별로 듣지 못했습니다. 적어도 이전 시스템이 종료된 후 첫 몇 주 동안은요. 그리고 그 시점에서는 어떤 정보도 좋은 정보가 아닙니다. 우리는 그렇게 생각했습니다. 아무도 중요한 기능을 놓치고 있지 않았습니 다. 아무도 불평하지 않았습니 다. 모두가 자신의 일을 할 수 있었습니 다. (제조, 제품 소유자).

이 수준의 명시적 합리화는 놀라운 발견이었습니다. 이 인지 패턴의 경직성은 경영진과 사용자 사이에 보이지 않는 장벽을 만들었습니다. 경영진은 체계적인 피드백을 적극적으로 구하지 않았고, 사용자에게는 피드백을 제공하는 구조화된 방법이 제공되지 않아 지속적인 무지의 순환이 형성되었습니다.

다른 소유자에게 업데이트를 보내면, 그저 아무 말 없이 침묵만 흐릅니다. 아마도 슈퍼 유저 중 한 명이 반응할 겁니다. 하지만 애플리케이션 주변에는 커뮤니티 감정이 없고, 우리가 노력한다면 우리에게 도움이 되는 시스템을 가질 수 있고, 우리에게 불리하지 않을 수 있습니다.(배송, 최고 경영진).

저는 그들이 노력하고 있다고 생각합니다. 하지만 그것은 우리가 주도하고 있으며, 우리가 손을 내밀어 좌절감을 표현하고 있다는 것입니다. 그것은 그들이 우리가 목표에 도달했는지 묻는 사전 예방적 접근 방식이 아닙니다. 공을 집어 들고 말하는 것은 우리입니다. 이것은 충분하지 않습니다.(제조, 사용자).

여기서, 최고 경영진은 도전을 애플리케이션에 대한 커뮤니티 감각의 부족으로 인식했지만, 사용자는 도전이 경영진의 사전 예방적 조치 부족으로 인해 발생했다고 표현했습니다. 우리가 관리 정당화에 대한 코드를 전체적으로 볼 때, 이러한 인지적 패턴을 통해 몇 가지 명확한 구조적 기동이 형성되고 있습니다. 우리는 프로젝트 사고라고 불리는 관리 합리화의 명백한 측면이 사용과 혜택이 프로젝트 활동(예: 개발, 교육)의 완료로부터 자동으로 따를 것이라는 사고방식에서 인식된다고 주장합니다. 따라서 우리는 다음과 같이 주장합니다.

그림 1은 프로젝트 사고와 제품 사고의 의미적 차별화가 커지고 있음을 보여줍니다. 프로젝트 중심적 사고방식은 여기서 분명하게 드러납니다. 프로젝트가 릴리스되면 다음 프로젝트로 넘어가고 회사는 후속 조치를 취하지 못하고 첫 번째 프로젝트가 목표를 달성했는지 이해하지 못합니다. 본질적으로 채택 관리(adoption management)는 제품 중심적 사고방식이 사용자의 요구에 따라 행동함으로써 비즈니스의 요구를 충족하는 것을 목표로 한다는 Cagan(2017)의 주장을 따릅니다. 그러나 이러한 연관성은 관리의 인지적 패턴에서는 분명하지 않습니다.

## 관리 조치

모델의 수행적 부분에 초점을 맞춰서, 구현이 끝나고 애플리케이션이 라이브로 전환된 후, 경영진이 채택을 관리하거나 사용자에게 영향을 미치기 위해 수행하겠다고 표명한 구체적인 조치에 초점을 맞춥니다.

첫째, 경영진이 시스템에서 얻기를 바라는 바람직한 이점을 이해하는 과정에서 우리는 그들이 그 목표에 도달했는지 여부를 어떻게 검증했는지 물었습니다. 이에 대한 답은 애플리케이션 사용과 얼마나 많은 사용자 오류가 수행되었는지 살펴보는 것이었습니다. 이는 다음과 같이 코딩되었습니다. 측정 실수다음 인용문과 같습니다.

*같은 매장에서 같은 실수를 한다고 측정하면 뭔가 잘못된 것입니다. 하지만 그게 우리가 가진 유일한 데이터드 라이브입니다. 프로세스에 실수가 있다면 그 사람에게 연락해서 올바른 방법으로 해야 한다고 말해야 할 수도 있습니다.(소매, 최고 경영진)*

기본적으로, 우리가 그것을 보는 유일한 방법은 전체 사용자 중 얼마나 많은 사용자가 새 시스템을 사용하고 있는지, 그리고 그들이 어떻게 사용하고 있는지였습니다. 얼마나 많은 거래가 이루어지고 있고, 이전 시스템과 비교했을 때 실제로 얼마나 많은 활동이 이루어지고 있습니까? 우리는 다른 측정치를 가지고 있지 않았었습니다.(제조, 제품 소유자)

혜택 관리 문헌은 일반적으로 평가가 원하는 결과와 혜택을 실현하는 데 중요한 단계라고 주장합니다(Ward et al., 1996). 연구에 따르면 이러한 유형의 관행에 참여하는 회사는 경쟁사보다 성과가 더 좋습니다(Holgeid et al., 2022). 그러나 사용자 오류 모니터링, 수동 실수 수정 및 절차 준수 보장은 혜택 관리 관행의 일부로 간주되는 경우가 드뭅니다.

*우리는 들어오는 데이터의 품질을 살펴봅니다. 다른 부서와 이야기할 때, 완전히 미친 일이 들어오더라도 그들은 꽤 좋은 개요를 가지고 있습니다.(운송, 최고 경영진)*

그래서 저는 프로세스의 여러 부분에서 우리가 어떻게 작업하는지 실제로 모니터링하는 사람들이 있습니다. 그래서 저는 그 일을 위한 팀을 가지고 있는데, 그들은 사용자가 어떻게 작업하는지 지속적으로 확인하고 모니터링하고 사용자에게 보여줍니다.

*어떻게 해야 하는지, 그리고 잘했을 때 칭찬해주는 방법을 알려줍니다.(배송, 제품 소유자)*

경영진이 절차 및 프로세스 정렬을 보장하기 위한 몇 가지 프로세스를 자세히 설명했지만, 원하는 혜택의 실현을 위한 후속 조치나 그러한 실현을 향한 채택 관리를 목표로 하는 조치의 어떤 유형도 식별할 수 없었습니다. 어느 정도 경영진은 애플리케이션의 성능과 구현 상태에 부분적으로 전념한 IT 설문 조사와 상태 회의를 설명할 수 있었습니다. 하지만 다시 말하지만, 체계적인 조치의 부족이 분명했고, 코드 구조화된 접근 방식이 없음관련성이 있는 것으로 나타났습니다.

*우리는 때때로 사용자 만족도 설문 조사를 실시하고 있습니다. 하지만 매달 실시한 것은 아니고, 1년에 3번 정도 실시한 것 같습니다. 때로는 특정 기능에 더 집중하기도 하고, 때로는 우리가 질문한 전체 시스템에 집중하기도 합니다.(제조, 제품 소유자)*

*IT 부서에서는 일년에 두 번 정도 사용자 설문 조사를 실시하여 IT 서비스 데스크에 대한 경험에 대해 모든 것을 묻습니다. 휴대전화와 컴퓨터에 만족하십니까? 코로나 기간 동안 재택근무를 어떻게 생각하십니까? 그리고 다양한 애플리케이션에 대해 어떻게 생각하십니까? 그리고(사례 적용) 언제나 엄청난 비판을 받는다.(운송, 최고 경영진)*

이러한 조치는 효과 중심 IT 개선에 필요한 체계적이고 반복적인 접근 방식을 따르지 않습니다(Simonsen & Hertzum, 2022). 구조화된 접근 방식이 없음또한, 구현된 애플리케이션 솔루션에 대한 후속조치 시도였던 이러한 관리 조치의 특성을 자세히 설명한 다음 인용문도 포함되었습니다.

*팬츠는 솔루션이고 제가 원하는 매개변수에서 작동합니다. 하지만 방법론적으로는 우리가 따라잡고 있는 것이 아닙니다. 시간이 없지만, 고려해야 할 일 중 하나입니다.(소매, 최고 경영진)*

*우리는 전통적인 접근 방식을 가지고 있고 뉴스레터로 시도해 보았습니다. 그리고 아시다시피, 팀 리더를 참여시키는 것이 그들이 의견을 제시할 수 있는 방법입니다.(제조, 상위 경영진)*

이러한 겉보기에 비구조적인 접근 방식은 우리가 호출하는 최종 코드에서도 분명하게 나타납니다. *살아 있으니, 놓아줘*. 여기서 우리는 경영진이 프로젝트의 원하는 혜택과 구현 후 전반적인 채택과 어떻게 관련되는지를 상징하는 인용문을 분류했습니다. 첫째, 개발 중에 사용자와 긴밀히 협력하면 경영진이 애플리케이션이 실행되면 한 걸음 물러설 수 있다는 것을 어떻게 보장하는지 표현했습니다.

*우리는 많은 사용자를 설계 단계에 포함시킵니다. 5~6명의 부서장과 다른 정보 제공자가 있습니다. 그래서 그들은 참여합니다. 하지만 우리가 놓아주면, 그것이 구현되고, 그 다음에는 스스로 하세요.(소매, 제품 소유자).*

경영진이 구현 후에 변경을 했다고는 말할 수 없습니다. 시스템 공급업체에서 몇 가지 변경이 있었습니다. 하지만 시스템을 사용하거나 다르게 사용할 것인지에 대한 내용은 없었습니다. 그 후에 아무런 소식도 듣지 못했습니다.(소매, 사용자)

이 시스템은 직관적이고, 사용자는 그 여정의 일부였습니다. 그 모든 상호작용은 도움이 되었다고 확신합니다. 숫자가 있나요? 아니요, 숫자가 없습니다. 측정하는 것이 아니고, 직원 만족도 설문 조사에서 볼 수 있는지 모르겠습니다. 볼 수 있다면 정말 대단할 겁니다. 하지만 측정한다면, 실제로 더 만족할 것이라고 믿습니다. (제조, 상위 경영진)

우리가 경영진에게 이러한 광범위한 노력에도 불구하고 프로젝트와 애플리케이션 구현이 성공적이었는지 어떻게 알았는지 물었을 때 그들은 물려셨습니다. 살아있어, 놓아줘더욱 지속되었습니다. 한 가지 질문이 직접 제기되었습니다. 신장이 성공했는지 어떻게 알 수 있나요? 또는 측정할 수 있나요? 이에 대한 답은 다음과 같습니다. 순수한 피드백으로. 그리고 느낌으로. 데이터 없음.(소매, 최고 경영진). 데이터 기반 성공 추정의 부족은 경영진이 애플리케이션이 실행되면 한 걸음 물러선다는 개념을 유지합니다. 한 가지 성공 기준은 이전 시스템을 단순히 교체하는 것과 직접 연결되었습니다.

이전 시스템이 폐쇄되었기 때문에 성공했습니다. 그것이 성공 기준이었습니다. 그것이 우리가 원했던 것입니다. 1,000만 달러의 비용을 절감해야 했던 것이 아니었습니다. 모든 것은 이전 시스템을 폐쇄하는 것이었습니다. 우리는 지금 거기에 있으며, 약간의 이탈을 보기 시작했고 원래는 논거로 사용하지 않았지만 놀랍게도 거기에 있는 몇 가지 이점을 실제로 보고 있습니다.(제조, 상위 경영진)

이 인터뷰에서 우리는 경영진이 어떻게 사용이나 사용자 행동에 영향을 미쳐 혜택을 얻으려 하는지에 대한 주제를 계속해서 논의했지만, 경영진이 뒤로 물러서는 느낌은 여전히 남았습니다.

사용자는 어깨를 으쓱하며 그게 바로 시스템이라고 말하는 데 익숙할 수 있습니다. 그리고 그것은 불행한 일인데, 우리가 그것을 고칠 수 있을지도 모르기 때문입니다. 하지만 그러면 우리는 18,000명의 사용자에게 따라 수정해야 합니다. 그것은 많은 수정입니다. 각 사용자의 말을 경청하기란 어렵습니다.(소매, 최고 경영진)

우리는 들어오는 모든 신입 직원을 교육하지 않습니다. 그것은 각 부서에 달려 있습니다. 따라서 누가 당신 옆에 앉느냐에 따라 달라집니다. 어디서 시작해야 하는지, 절차나 프로세스를 어떻게 따라야 하는지 알려주는 e러닝 세계가 없습니다.(배송, 사용자).

에 대한 코드 살아있어, 놓아줘이는 회사들이 비즈니스 결과(outcomes)를 이끌어내기 위해 사용자 행동을 바꾸는 데 집중하는 대신 기능을 만들고 개발하고 프로젝트를 완료하는 데 매우 집중하고 있다는 것을 시사합니다 (Seiden, 2019). 따라서 우리는 프로젝트 사고와 제품 사고를 구별하는 것의 관련성을 계속 보고 있으며 이를 뒷받침하는 결과가

그림 1은 프로젝트 사고와 유사합니다. 관리자는 프로젝트 완료에 집중하고 제품이 창출한 가치에는 집중하지 않으므로 프로젝트 중심적 사고방식을 보여주었습니다. 그림 1은 인지적 패턴이 관리 행동을 정당화(부족)하는 프로젝트 사고방식을 생략하고, 그 대가로 그러한 행동의 결과가 인식에 도전하지 않고 정신적 구조를 유지하는 복잡한 사회 기술적 이념을 나타낸다고 주장합니다. 위에서 개발한 코드는 인지적 패턴이 회사가 운영되고 직원이 기능 부족을 표현하지 않으면 엔터프라이즈 소프트웨어 구현이 성공적이라는 생각을 강화함에 따라 프로젝트 사고방식의 사례가 됩니다. 프로젝트 사고방식은 사용자가 직무를 완수하기 위해 이러한 운영을 준수해야 하므로 운영적 사용을 채택으로 설정합니다. 프로젝트 사고 사고방식은 오류 측정에 의존하는 행동 표현에서 관찰됩니다. 추가로 적용되는 관리 행동은 구조화되지 않았으며 프로젝트가 구현되면 관리가 물러서는 느낌이 듭니다.

## 논의

요약하자면, 우리는 관리자가 채택을 관리하기 위해 거의 노력하지 않은 세 가지 조직 사례를 분석했습니다. 이러한 경우 관리자의 행동은 경직되고 구조화되지 않았으며 여러 가지 인지적 패턴에 의해 합리화되었습니다. 우리는 행동을 정당화하고 인식을 유지하는 과정을 프로젝트 사고로 요약합니다. 이는 제품 자체와 그것이 창출하는 가치에 대한 집중보다 프로젝트 완료의 엄격한 경계를 우선시하는 사고방식에 대한 의미를 부여하기 위한 것입니다. 이 결과는 혜택 관리와 채택 관리 사이의 공간을 탐색하는 이전 연구와 결합됩니다. 여기에서 연구는 또한 사양을 충족하고 시간과 예산 내에서 솔루션을 제공하는 데 중점을 두고 있기 때문에 구현 후에 적용되는 혜택 중심 관행이 얼마나 적은지 강조합니다(Ashurst et al., 2008; Peppard et al., 2007). 이는 비즈니스 사례를 공식화하고 이를 후속 조치하는 회사가 IS 프로젝트에서 더 큰 혜택을 얻는다는 것을 보여주는 연구에도 불구하고 그렇습니다(Thomas & Fernandez, 2008; Holgeid et al., 2022). 제시된 진술의 경직성은 행복과 불안과 같은 개인의 다양한 감정과 IT 사용 간의 관계를 인정하는 것의 중요성을 보여준 이전의 다른 연구들과 모순됩니다(Beaudry & Pinsonneault, 2010). 인터뷰에서 묘사된 사고방식은 사용자 행동과 애플리케이션 사용의 복잡한 측면을 무시하는 것처럼 보이며, 따라서 기술 도입 결정, 그룹 상호 작용 및 결정 프로세스에 대한 긍정적 및 부정적 감정의 정도 간의 상호 연관성을 증명한 연구에 도전합니다(Sarker et al., 2005). 또한 사용의 다단계적 특성과 효과적인 IT 사용을 개념화하는 잘 확립된 이론을 무시합니다.



원하는 혜택을 얻을 수 있는 메커니즘(Burton-Jones & Volkoff, 2017). 이 연구는 성공적인 채택 관리가 규범적 프레임워크와 방법만큼 관리자의 인지에 초점을 맞추는 것을 수반한다고 제안합니다. 저희 정보 제공자의 프로젝트 사고는 소프트웨어의 수명 주기 동안 책임을 지는 제품 팀으로 프로젝트 팀을 대체해야 한다는 DevOps와 같은 현재의 움직임과 대조됩니다(Wiedemann 2020). 저희 연구는 여기에 프로젝트 중심 조직에서 제품 중심 조직으로 이동하는 것은 개발자와 운영 부서 간의 협업뿐만 아니라 관리자가 소프트웨어 구현에서 스스로에게 보는 역할의 변화도 수반할 수 있다는 점을 추가합니다. 관리 사고방식을 프로젝트 사고방식으로 요약함으로써 연구 사례에서 IT 구현 프로젝트가 가치를 반환하는 데 어려움을 겪는 이유에 대한 몇 가지 설명을 제공합니다. 프로젝트 사고 사고방식은 관리자가 비즈니스 혜택 실현 및 사용자 가치를 추적할 필요성이나 가치가 없다고 생각할 수 있는 세상에 대한 정신적 구성인 듯합니다. *어떻게/for*. 이 모델을 통해 이 논문은 프로젝트 사고의 장점에 도전하는 것을 목표로 하지 않습니다. 대신 개념적 모델은 프로젝트 사고가 일부 관리자에게 어떻게 나타나는지, 그리고 기업 소프트웨어 도입을 관리하는 과정에서 사고방식이 어떻게 도전이 될 수 있는지를 요약합니다. Project Management Institute에 따르면 프로젝트는 "고유한 제품, 서비스 또는 결과를 만들기 위해 수행되는 일시적인 노력"입니다(Ding, 2016). 그리고 프로젝트 사고의 일시적인 특성이 도입 관리의 반복적 평가 필요성에 도전합니다.

관리자와 다른 실무자들이 채택 관리와 관련된 자신의 행동과 인식을 보다 주의 깊게 살펴보는 방법에 대한 연구 결과에서 영감을 얻기를 바랍니다. 채택 중심의 행동 관리가 구현 후에 얼마나 적게 적용되었는지 놀랍지만, 이러한 (부족한) 행동을 주도하는 이러한 합리화 패턴을 표면화함으로써 관리자들이 프로젝트 중심 사고방식에서 제품 중심 사고방식으로 전환하는 첫 단계를 밟을 수 있도록 하여 엔터프라이즈 소프트웨어의 전체적인 채택 관리에 집중할 수 있습니다. 채택 관리 인식을 보다 주의 깊게 살펴보는 과정에서 관리자와 학자들이 필수 소프트웨어의 결과에 주의를 기울여야 합니다. 엔터프라이즈 소프트웨어를 구현하면 사용자는 작업을 준수하고 애플리케이션을 활용하여 작업을 수행해야 합니다. 그러나 조직과 사용자 모두에게 이점과 가치를 실현하기 위해 사용자가 고의로 오류를 저지르거나(Ferneley & Sobreperéz, 2006) 시스템에 대해 불평하는 경우(Markus, 1983) 단순히 사용하는 것이 바람직하지 고려해야 합니다. 기업용 소프트웨어의 경우, 사용이 사용자의 수용이나 만족을 자동으로 나타내는 것으로 간주할 수 없으므로 사용의 맥락화가 필요합니다(Beaudry et al., 2020).

필수 소프트웨어 공간을 관리할 때 소프트웨어 도입을 보다 전체적으로 고려하여 의무 환경에서 발생하는 다양한 수용 및 저항 반응을 포괄할 필요가 있습니다(Bhattacharjee et al., 2018). 이 연구에 참여한 관리자들은 소프트웨어 의무화의 효과를 인정했지만, 그에 따라 사용자 도입에 영향을 미칠 수는 없었습니다. 의무 환경에서 열악한 소프트웨어는 무딘 건축 도구 세트와 같습니다. 경영진이 건축업자에게 날카롭지 않은 도구를 제공하여 집을 짓게 하면 건설 과정이 늦어지고, 집의 품질이 떨어지며, 건축업자에게 부담이 됩니다. 반복적인 도입 관리 관행에 참여하지 않으면 조직은 사람들이 소프트웨어에 불만족스러워한다는 사실을 알아낼 기회를 놓칩니다. 따라서 경영진은 근로자에게 열악한 엔터프라이즈 소프트웨어를 제공하여 생산성을 저해하고, 소프트웨어의 잠재력을 최대한 발휘하지 못하게 하며, 직원을 좌절시킬 위험이 있습니다. 이 아이디어를 현대 조직에 적용하면, 현대적이고 디지털적인 조직이라는 주장은 더 이상 외부 비즈니스 운영과 고객 만족에만 국한되지 않습니다. 이러한 변화는 인사부(HR) 관리(Plaskoff, 2017; Tucker, 2020)뿐만 아니라 모든 기능 부서에서 직원 경험과 완료해야 할 업무에 대한 관심을 확대하고 있습니다. IT 부서의 경우 앞으로의 궤적은 직원들이 개인 생활에서 익숙한 사용자 친화적 경험과 일치하지 않는 솔루션을 도입하고 시행하지 않는 데 중점을 두는 것으로 보입니다. 젊은 디지털 네이티브가 미래에 기술과 상호 작용하는 방식에 대한 기대치를 낮출 것이라고 상상하기는 어렵습니다. 그렇다면 직원의 디지털 경험을 모니터링하고 이를 바탕으로 조치를 취하지 못하는 조직은 어떤 영향을 받을까요? 디지털 네이티브가 열악한 소프트웨어 솔루션을 사용하는 직장 생활로 이동할 것이라고 현실적으로 상상할 수 있을까요? 연구에 따르면 특전이 직원 참여에 대한 유일한 답은 아닙니다. 조직은 직원과 함께 배려와 직원의 요구 사항에 대한 더 깊은 이해를 보여주는 경험을 공동 설계해야 합니다(Plaskoff, 2017). 성공적인 기업의 미래는 직원의 전체 경험을 중심에 두는 데 있습니다. 그러나 이는 직장에서 직원을 참여시키는 전통적인 접근 방식에 잠재적으로 도전합니다. 이는 특전을 늘리거나 직장에서 "재미있게 만드는 것"을 넘어서기 때문입니다(Plaskoff, 2017). 이러한 결과를 HR 관리 분야에서 IS 구현 설정으로 변환하면, 의무적 환경에서는 연구자와 실무자가 조직의 원하는 혜택과 일치하면서도 직원의 요구를 효과적으로 충족하는 솔루션을 설계, 구현 및 활용하도록 지시하는 동일한 전체론적 접근 방식이 필요하다는 것을 알 수 있습니다.



의무적 환경은 프로젝트에서 제품 사고방식으로의 전환 과정에서 경영진에게 또 다른 어려운 결과를 제공합니다. 조직이 직원의 디지털 경험을 중앙 집중화하기 시작하면서 측정 가능성의 과제가 분명해질 것으로 예상합니다. 관리자가 IS 구현 프로세스에 영향을 미쳐 주어진 문제를 가치 있는 결과로 해결하는 시스템 사용을 보장할 수 있도록 하려면 지침 지표를 사용할 수 있어야 하고 정보 중심이어야 합니다. 사례 연구에서 두 수준의 경영진 모두 솔루션의 수행 방식과 가치를 반환했는지 측정하는 데 어려움을 표명했습니다. 자발적 및 의무적 소프트웨어의 관리를 구분하는 최근 연구(Nah et al., 2004; Brown et al. 2002; McNally, & Griffin, 2010; Bhattacharjee et al., 2018; Beaudry et al., 2020)의 주장에 따라 로그인 빈도, 활성 사용자 수, 애플리케이션에서 소비한 시간 등을 통해 IT 성과를 측정하는 기존 방식은 가치 실현을 확인하려고 할 때 무효화됩니다. 개인용 소프트웨어와 같이 여러 가지 대안이 존재하는 자발적인 애플리케이션의 맥락에서 이러한 지표를 추적하는 것은 실제로 의미가 있습니다. 그런 다음, 이러한 지표는 높은 일일 로그인, 참여 사용자 및 광범위한 사용 시간을 반영하여 애플리케이션의 바람직성에 대한 개요를 제공하는 데 도움이 됩니다. 그러나 복잡하고 시간이 많이 걸리는 작업을 수행하기 위해 애플리케이션에 액세스해야 하는 직원을 고려할 때 이러한 지표의 의미는 약화되고 희소한 통찰력을 제공합니다.

마지막으로, 우리는 우리 연구의 몇 가지 한계를 인정합니다. 첫째, 사례 연구는 세 개의 선택된 회사만 포함하기 때문에 제한된 범위를 제공하므로 결과를 더 광범위한 조직으로 일반화하는 데 제한이 있습니다. 따라서 우리는 이 관점을 더 광범위하고 종단적인 사례에 적용하여 프로젝트 사고 사고방식이 엔터프라이즈 소프트웨어의 도입 관리에 어떤 영향을 미치는지 이해하기 위한 향후 연구를 요구합니다. 둘째, 다른 데이터 소스를 포함하여 조치의 삼각 측량을 확장하고

## 참고문헌

Ashurst, C., Doherty, NF, & Peppard, J. (2008). IT 개발 프로젝트의 영향 개선: 혜택 실현 역량 모델. *European Journal of Information Systems*, 17(4), 352–370. <https://doi.org/10.1057/ejis.2008.33>

Beaudry, A., & Pinsonneault, A. (2010). 수용의 다른 측면: 정보 기술 사용에 대한 감정의 직접적 및 간접적 효과 연구. *MIS Quarterly*, 34(4), 689–710. <https://doi.org/10.2307/25750701>

Beaudry, A., Vaghefi, I., Bagayogo, F., & Lapointe, L. (2020). IT 사용자 행동의 영향: 관찰을 통한

입양 관리에 대한 개념. 이 연구는 9개의 인터뷰만 포함하고 있으며, 입양에 대한 다양한 인식이 실제 관리 조치에 어떤 영향을 미치는지 이해하기 위해 여러 관리 수준에서 더 많은 관리자를 인터뷰하는 것이 더 방법론적으로 엄격할 것입니다.

## 결론

이 논문에서 우리는 엔터프라이즈 소프트웨어 도입을 관리하기 위해 거의 노력하지 않은 관리자들과 합리화를 살펴보았습니다. 일반적으로 관리자가 어떻게 도입을 관리하는지 추론할 수는 없지만, 우리의 사례 연구는 적어도 일부 관리자가 도입 관리에 적극적인 입장을 취하지 않는 이유에 대한 통찰력을 제공합니다. 이러한 관리자의 경우, 그들의 관리 행동은 비구조화되고 사용자 실수나 수동 오류의 측정에 기반을 두었습니다. 이러한 실수가 나타나면, 경영진은 오류를 수정하고 절차를 조정하여 다시는 발생하지 않도록 하려고 합니다. 거의 설명되지 않은 행동은 모범 사례에 대한 도입 관리 또는 혜택 관리 정의를 준수하지 않습니다. 더욱이, 이러한 행동은 채택에 대한 관리자의 인지적 인식과 의무적 환경에서 이를 관리하는 방법에 의해 정당화됩니다. 여기서, 비즈니스가 운영되고 사용자가 작업을 완료하고 이행할 수 있는 경우 사용이 도입으로 인식됩니다. 경영진은 엔터프라이즈 소프트웨어와 관련하여 사용자가 자발적으로 선택할 수 없다는 것을 인정하지만, 여전히 소식이 없는 것은 좋은 소식이라는 생각을 유지합니다. 우리는 이러한 명시적이고 수행적인 구조를 프로젝트 사고 사고방식으로 분류하여 연구에서 경영진이 채택 프로세스에 영향을 미치는 데 어려움을 겪고 있다는 것을 강조합니다. 일단 구현이 완료되면 집중이 필요한 새로운 프로젝트가 시작되기 때문입니다. 우리는 제시된 모델이 미래의 관리자에게 채택을 바라보는 방식과 그것을 관리하는 방법을 더 주의 깊게 생각하도록 영감을 줄 수 있기를 바랍니다. 잠재적으로 원하는 혜택을 실현할 위험이 있기 때문입니다.

새로운 렌즈. 정보 시스템 협회의 커뮤니케이션, 46, 331–364.

<https://doi.org/10.17705/1cais.04615>

Bhattacharjee, A., Davis, CJ, Connolly, AJ, & Hikmet, N. (2018). 의무적 IT 사용에 대한 사용자 반응: 대치 이론 관점. *European Journal of Information Systems*, 27(4), 395–414. <https://doi.org/10.1057/s41303-017-0047-0>

Bloch, M., Blumberg, S., & Laartz, J. (2012). 대규모 IT 프로젝트를 정해진 시간, 예산, 가치에 맞춰 전달하기(pp. 1-6). Business Technology Office: McKinsey & Company.

Brown, SA, Massey, AP, Montoya-weiss, MM, & Burkman, JR(2002). 정말 해야 하나요? 의무 기술에 대한 사용자 수용. *유럽 정보 시스템 저널*, 11(4), 283–295.

<https://doi.org/10.1057/palgrave.ejis.3000438>

Burton-Jones, A., & Volkoff, O. (2017). 효과적인 사용에 대한 맥락화된 이론을 어떻게 개발할 수 있을까? 커뮤니티 케어 전자 건강 기록의 맥락에서의 시범. *정보 시스템 연구*, 28(3), 468-489.

Bygstad, B.; Munkvold, BE; 및 Volkoff, O. (2016). 가능성을 통한 생성 메커니즘 식별: 중요한 현실주의 데이터 분석을 위한 프레임워크. *정보기술 저널*, 31(1), 83-96.

카간, M. (2017). *영감: 고객이 좋아하는 기술 제품을 만드는 방법*. 존 와일리 앤 선즈.

Cadle, J., & Yeates, D. (편집자). (2004). *정보 시스템을 위한 프로젝트 관리*. 피어슨 교육.

샤르마즈, K. (2006). *근거 이론 구성: 질적 분석을 통한 실용적 가이드*. 세이지.

데이비스, F. (1987). 정보 시스템의 사용자 수용: 기술 수용 모델. 연구 논문 시리즈, 워킹페이퍼(529), 1-33. 미시간 대학교. <https://hdl.handle.net/2027.42/35547>

딩, R. (2016). *효과적인 프로젝트 사고를 기반으로 한 핵심 프로젝트 관리*. 슈프링거 출판사 베를린 하이델베르크.

Falch, CC & Krancher, O., 입양 관리: 혜택 관리 및 입양 문헌 검토(2023). 31회 회의록, 유럽 정보 회의

시스템. 402. <https://aisel.aisnet.org/ecis2023-rp/402>

Feldman, MS (2000). 지속적인 변화의 원천으로서의 조직적 일상. *조직과학*, 11(6), 611-629.

Feldman, MS, & Pentland, BT(2003). 유연성과 변화의 원천으로서 조직 일상을 재개념화하기. *행정과학분기*, 48(1), 94-118.

Ferneley, EH, & Sobreperez, P. (2006). 저항, 준수 또는 해결 방법? 정보 시스템에 대한 사용자 참여의 다양한 측면에 대한 검토. *European Journal of Information Systems*, 15(4), 345-356.

Ginzberg, MJ(1981). MIS 구현 실패의 초기 진단: 유망한 결과와 답이 없는 질문. *Management Science*, 27(4), 459-478.

Hirschheim, R., & Newman, M. (1988). 정보 시스템과 사용자 저항: 이론과 실제. *The Computer Journal*, 31(5), 398-408. <https://doi.org/10.1093/comjnl/31.5.398>

Holgeid, KK, Jørgensen, M., Sjøberg, DI, & Krogstie, J. (2022). 소프트웨어 개발의 이점 관리: 경험적 연구에 대한 체계적인 검토. *IET 소프트웨어*, 15(1), 1-24. <https://doi.org/10.1049/sfw2.12007>

Kim, H.-W., & Kankanhalli, A. (2009). 정보 시스템 구현에 대한 사용자 저항 조사: 현상 유지 편향 관점. *MIS Quarterly*, 33(3), 567-582. <https://doi.org/10.2307/20650309>

Koh, CE, Prybutok, VR, & Ryan, SD(2010). 소프트웨어 기술의 의무적 사용을 위한 모델: 과학을 알리는 여러 수준의 추상화를 적용한 통합적 접근 방식. *과학을 알리다*, 13, 177.

라투르, B. (1986). *결사의 권한*. J. Law(편집), 264-280. 런던: Routledge and Kegan Paul.

Markus, LM(1983). 권력, 정치 및 MIS 구현. *ACM 커뮤니케이션*, 26(6), 430-444.

McNally, RC, & Griffin, A. (2010). 고객 관계 관리 정보 기술의 의무적 채택: 태도-의도에서 고객 지원의 역할

모델. *국제 비즈니스 혁신 및 연구 저널*, 4(1-2), 30-47.

민츠버그, H. (1971). 관리 작업: 관찰을 통한 분석. *관리 과학*, 18(2), B-97.

Orlikowski, WJ (1996). 시간 경과에 따른 조직 변형의 즉흥적 개선: 상황적 변화 관점. *정보 시스템 연구*, 7(1), 63-92.

Peppard, J., Ward, J., & Daniel, E. (2007). IT 투자로부터 비즈니스 혜택 실현 관리. *MIS 분기별 임원*, 6(1), 1-11.

PMI(2016). 혜택 실현 관리 프레임워크. Thought Leadership Series. 2016년 11월. <https://www.pmi.org/learning/thought-leadership/series/benefits-realization/benefits-realization-managementframework>.

Polites, GL, & Karahanna, E. (2012). 현상 유지에 얽매임: 현직자의 억제 효과

시스템 습관, 전환 비용 및 새로운 시스템 수용에 대한 관성. *MIS 분기별*, 36(1), 21-

42. <https://www.jstor.org/stable/i40068646>.

Sarker, S., Valacich, JS, & Sarker, S. (2005). 그룹별 기술 도입: 가치 관점. *Journal of the Association for Information Systems*, 6(2), 3.

세이트, J. (2019). *산출보다 결과가 중요하다*. 센스앤리스폰드 프레스.

Seo, D., Boonstra, A., & Offenbeek, M. (2011). 모호한 집단에서 IS 채택 관리. *ACM의 커뮤니케이션*, 54(11), 68-73.

<https://doi.org/10.1145/2018396.2018416>

Simonsen, J., & Hertzum, M. (2022). 효과 중심 IT 개선: 지역적 사후 구현 기회 추구. *Scandinavian Journal of Information Systems*, 34(1), 2.

Strong, DM, Volkoff, O., Johnson, SA, Pelletier, L. R., Tulu, B., Bar-On, I., ... & Garber, L. (2014). 조직-EHR 제공 가능성 실현 이론. *정보시스템학회 저널*, 15(2), 2.

Thomas, G., & Fernández, W. (2008). IT 프로젝트에서의 성공: 정의의 문제?. *국제 프로젝트 관리 저널*, 26(7), 733-742.

Van Offenbeek, M., Boonstra, A., & Seo, D. (2013). 수용과 저항 연구 통합을 향해: 텔레케어 사례 연구의 증거. *유럽 정보 시스템 저널*, 22(4), 434-454.

Venkatesh, V., Thong, JY, & Xu, X. (2016). 기술 수용 및 사용에 대한 통합 이론: 종합 및 앞으로의 길. *정보 시스템 학회 저널*, 17(5), 328-376.

Walsham, G. (1995). IS 연구의 해석적 사례 연구: 본질과 방법. *European Journal of Information Systems* (4:2), pp. 74-82

Ward, J., Taylor, P., & Bond, P. (1996). IS/IT 혜택의 평가 및 실현: 현재 관행에 대한 실증적 연구. *European Journal of Information Systems*, 4(4), 214-225.

Wiedemann, A., Wiesche, M., Gewald, H., & Krcmar, H. (2020). DevOps가 개발과 운영을 어떻게 조정하는지 이해하기: IT 내부 조정의 3부 모델. *유럽 정보 시스템 저널*, 29(5), 458-473.