programmers

2. 데이터 기반 의사 결정

데이터 문해력의 기본이 되는 데이터 기반 의사 결정이란?

한기용

keeyonghan@hotmail.com

programmers

Contents

programmers

퀴즈 리뷰

지난 시간 퀴즈를 같이 풀어보자

❖ 데이터 팀의 역할퀴즈

- https://forms.gle/nCpS5VCyZ1D2yGT5A

programmers

데이터 기반 의사 결정이란?

데이터를 기반으로한 의사결정이 무엇인지 알아보자

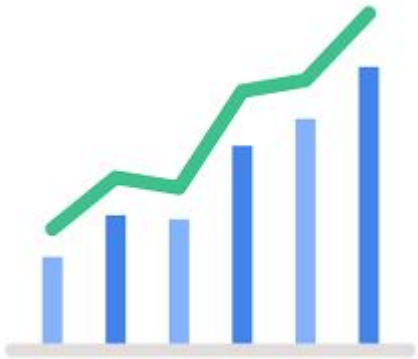
◆ 두 가지 형태의 데이터 기반 의사 결정

❖ 데이터란 기본적으로 과거의 기록

- 이를 바탕으로한 결정은 지금 하는 일의 최적화에 가까움 vs. 혁신

❖ Data Driven Decision

❖ Data Informed Decision



◆ 데이터에서 인사이트 찾기

❖ 중요 지표를 데이터를 기반으로 정의하고 시각화하기

- 뒤에서 더 설명

❖ 가설을 바탕으로 실제 데이터를 보고 확인하기

- “A/B 테스트”

◆ 데이터 분석 케이스들 살펴보기

- ❖ 중요 지표 대시보드 만들기
- ❖ 고객 이탈률 분석: 보통 코호트 분석으로 진행
- ❖ 고객 잔존률 분석: 보통 코호트 분석으로 진행
- ❖ 마케팅 기여도 분석
- ❖ …

◆ 데이터 분석의 예 - 고객 이탈률

❖ 샌프란시스코 기반 전동 스쿠터 회사에서 돈을 많이 쓰는 고객들이 두세달 후에는 서비스를 그만 사용하는 현상이 발견됨

- 어느 서비스이건 돈을 많이 쓰는 사람(VIP)들의 이탈률을 트래킹하는 것이 중요함

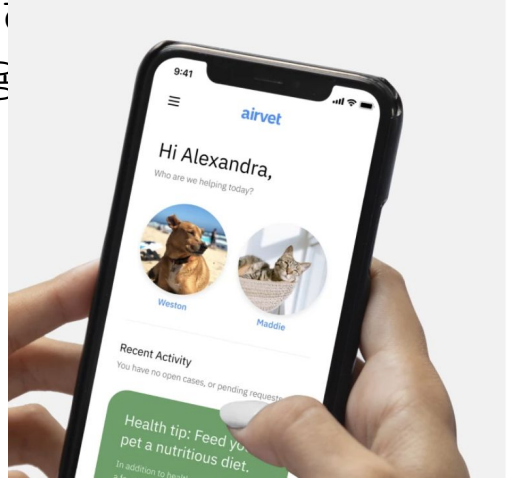
◆ 데이터 기반 부가가치 예 - 고객용 대시보드

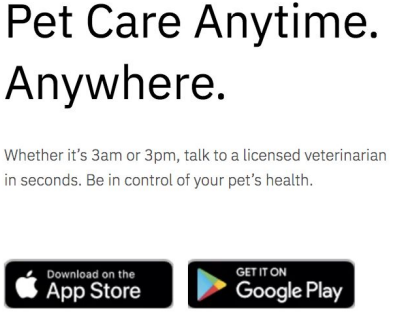
❖ Airvet이란 회사와 수행했던 프로젝트로 고객용 대시보드 개발

- Airvet은 원격 애완동물 진료 서비스를 제공해주는 마켓플레이스
- 애완동물 주인과 수의사/동물병원을 연결

❖ 세일즈포스와 내부 매출 정보를 연동하여 수의사/동물병원용 대시보드 개발

- 세일즈포스 정보를 데이터 웨어
- 룩커(Looker)를 대시보드로 사용





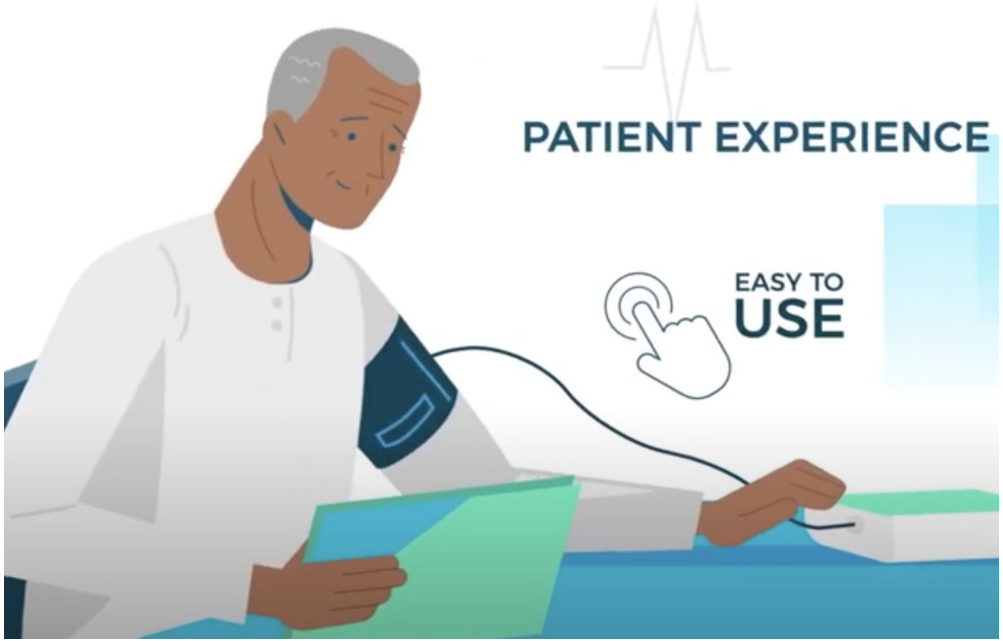
◆ 데이터 분석의 예 - 마케팅 기여도 분석

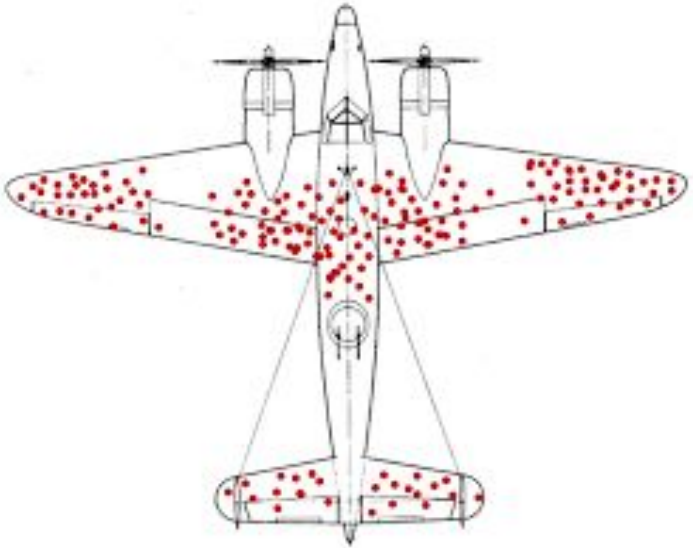
- ❖ 샌프란시스코 기반 화장 (火葬) 스타트업인 튤립의 예
- ❖ 다양한 광고 마케팅을 디지털 미디어 기반으로 수행
- ❖ 이 결과를 빠르게 분석하여 어느 채널에 어떤 형태의 마케팅이 효과적인지 파악
- ❖ 디지털 마케팅은 기본적으로 데이터 중심으로 돌아감

◆ 데이터 분석의 예 - 고객 불만과 이탈률간의 관계

❖ 서비스 관련해서 트러블슈팅 전화를 하는 고객들의 이탈률은 어떨까?

- Survivorship Bias & Confirmation Bias





◆ 데이터 분석가의 역할

❖ 비지니스 인텔리전스를 책임짐

- 중요 지표를 정의하고 이를 대시보드 형태로 시각화
  - 대시보드로는 태블로(Tableau)와 룩커(Looker)등의 툴이 가장 흔히 사용됨
  - 오픈소스로는 수퍼셋(Superset)이 많이 사용됨
- 이런 일을 수행하려면 비지니스 도메인에 대한 깊은 지식이 필요

❖ 회사내 다른 팀들의 데이터 관련 질문 대답

- 임원들이나 팀 리드들이 데이터 기반 결정을 내릴 수 있도록 도와줌
- 질문들이 굉장히 많고 반복적이기에 어떻게 셀프서비스로 만들 수 있느냐가 관건

◆ 데이터 분석가의 스킬셋

- ❖ SQL
  - 보통 코딩을 하지는 않지만 파이썬 코딩 능력이 있다면 금상첨화
- ❖ 데이터 모델링과 ELT (“T”)
  - dbt라는 툴을 많이 사용함
- ❖ 통계적 지식
- ❖ A/B 테스트 지식과 경험
- ❖ 지표 정의와 대시보드 (Tableau, Looker, Power BI, …)
- ❖ 비지니스 도메인에 관한 깊은 지식

◆ 데이터 분석가의 딜레마

- ❖ 보통 많은 수의 긴급한 데이터 관련 질문들에 시달림
- ❖ 많은 경우 현업팀에 소속되기도 함
  - 내 커리어에서 다음은 무엇인가?
  - 소속감이 불분명하고 내 고과 기준이 불명확해짐
- ❖ 데이터 분석가의 경우 조직구조가 더 중요함
  - 다음 섹션에서 보충 설명

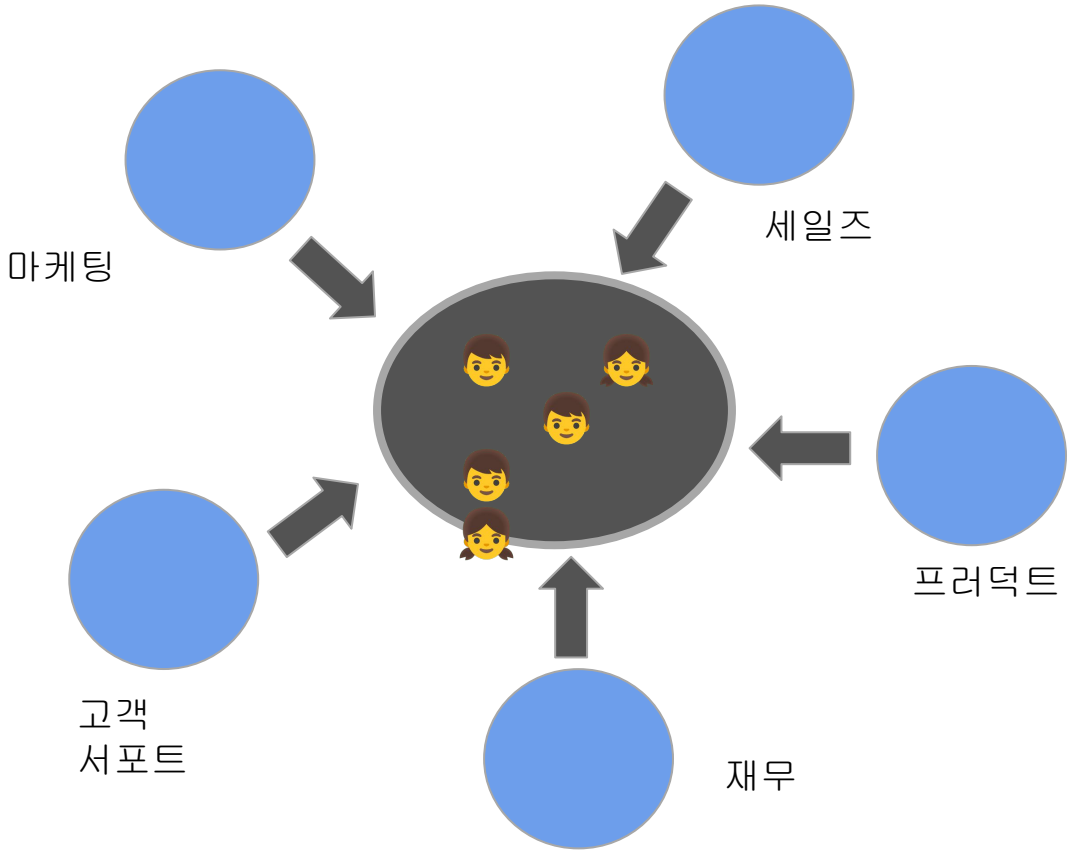
programmers

조직 구조의 중요성과 트렌드

데이터 조직이 회사 전체로 볼때 갖는 조직 형태를 살펴보자

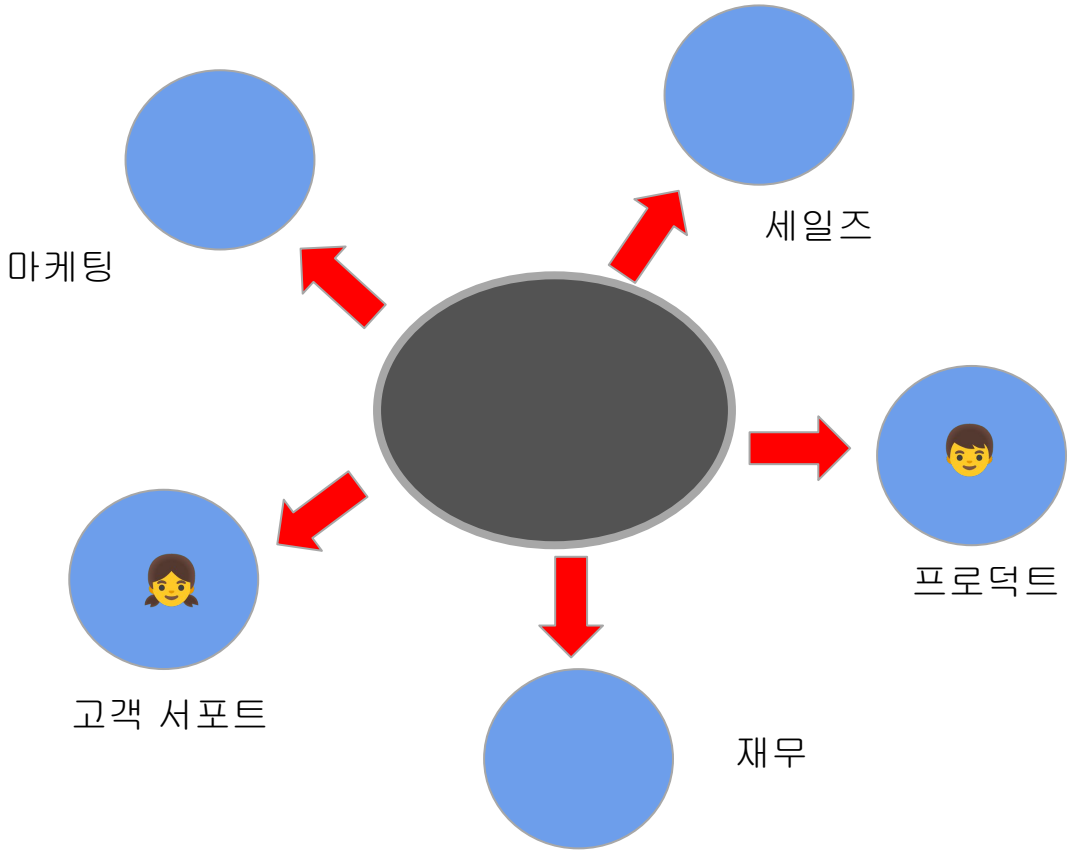
- 3가지 데이터 팀 조직 구조
  - 중앙 집중 구조
  - 분산 구조
  - 하이브리드 구조

◆중앙집중 구조: 모든 데이터 팀원들이 하나의 팀으로 존재



- 일의 우선 순위는 중앙 데이터팀이 최종 결정
- 데이터 팀원들간의 지식과 경험의 공유가 쉬워지고 커리어 경로가 더 잘 보임
- 하지만 현업부서들의 만족도는 상대적으로 떨어짐

◆분산 구조: 데이터 팀이 현업 부서별로 존재



- 일의 우선 순위는 각 팀별로 결정
- 데이터 일을 하는 사람들간의 지식/경험의 공유가 힘들고 데이터 인프라나 데이터의 공유가 힘들어짐
- 현업부서들의 만족도는 처음에는 좋지만 많은 수의 데이터 팀원들이 회사를 그만두게 됨

◆분산 구조: 데이터 팀이 현업 부서별로 존재

❖ 2가지 경우 존재

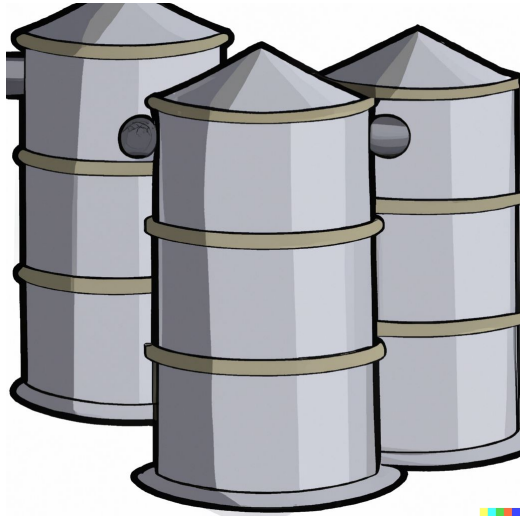
- 기존 중앙 집중 구조에서 조직 변경을 통해 분산 구조화
- 자생적으로 혹은 인수합병 등을 통해 조직별 데이터팀 존재

❖ 어떤 문제들이 존재하는가?

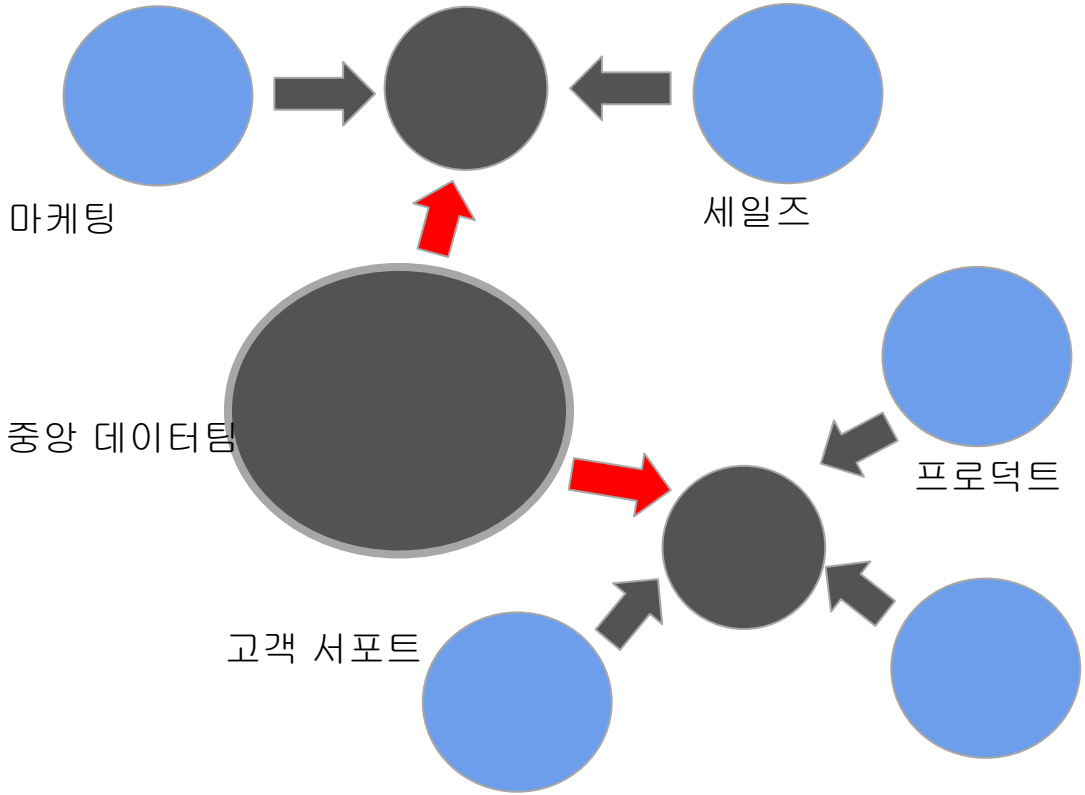
- 서로 다른 데이터 전략
- 회사 전체로 볼때 불완전한 데이터 셋
- 중복 투자
- 보안/규제 관련 이슈 발생 가능성 증가

❖ 하지만 이는 어쩔 수 없는 트렌드로 보임

- 이 관점에서는 클라우드 이전이 도움이 됨
- 클라우드도 대세임



◆중앙집중과 분산의 하이브리드 모델



- 가장 이상적인 조직 구조
- 데이터 팀원들은 일부는 중앙에서 인프라적인 일을 수행하고 나머지는 현업팀에서 작업
- 중소 규모 회사에서는 기능/목적 조직구조의 형태로 데이터팀 안에서 커리어 경로를 만들 수 있음

◆회사의 크기에 따라 데이터 조직의 형태가 아주 다름

- ❖ 회사가 아주 커지면 회사 전체 데이터 웨어하우스의 구성은 불가능해짐
  - 조직 별로 데이터 시스템을 별도로 갖추게 되고 필요로 따라 통합 시스템을 구성
  - 데이터 메쉬의 필요성이 점점 대두됨
- ❖ 마이크로소프트의 예
  - 2018년 기준 법무팀과 재무팀의 경우 별도 데이터 시스템 구축
    - 그 전에는 조직별로 별도 법무팀과 재무팀 존재
    - 별도 시스템을 구성하면서 시스템을 Azure로 이전

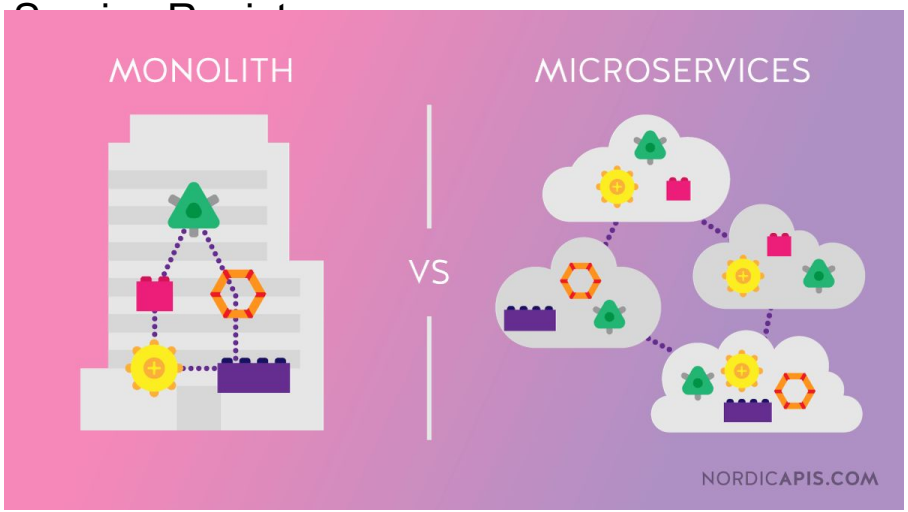
❖ 용어 설명: 데이터 메쉬 (Data Mesh)

- Zhamak Dehghani가 2019년 처음 제안
- 데이터 메시는 중앙 관리와 표준을 염두에 둔 데이터 분산 데이터 아키텍처
  - 현재로는 기술이라기 보다는 개념에 가까움
  - 조직/문화적인 준비가 필요한 개념이기도 함
  - 마이크로서비스와 아주 흡사한 원칙을 갖고 있음

1980년대 후반	2000년대 후반	2010년 중반	2021년
Data Warehouse (Top-down)	Data Lake (Bottom-up)	Cloud Data Platform	**Data Mesh**
중앙 시스템			**분산 시스템**

❖ 잠깐 마이크로서비스란?

- 웹 서비스를 다수의 작은 서비스(microservice)들로 구현하는 방식
- 각 서비스들은 팀 단위로 원하는 언어/기술로 개발하는 자율성을 가짐
- 각 서비스들은 계약관계로 지켜야하는 책임이 있고 서비스 정보를 등록해야함
  - Service Registry



- Decentralization
- Modularity
- Domain driven design
- Focus on empowering teams

programmers

데이터 조직의 일주일 살펴보기

데이터 팀이 무슨 일을 하는지 한 주를 살펴보자

◆ 애자일 개발방법론이란?

❖ 세상이 빠르게 변화하면서 미리 소프트웨어의 요구 사항을 알 수 없음

- 알아도 시간이 지나면서 변하게 됨
- 소프트웨어 개발은 폭포수 모델(Waterfall Model)이 아닌 애자일 방법론이 대세가 됨

❖ 애자일 개발방법론의 특징

- 아는 만큼 보이는 만큼 만들어가자! 매 사이클마다 바로 쓸 수 있는 기능을 구현
- 짧은 사이클이 특징 (보통 1-3주). 보통 스프린트(Sprint)라고 부름
- 매 스프린트마다 다음 스텝들을 반복
  - 플래닝 미팅: 스프린트 동안 무엇을 할지 결정
  - 매일 스탠드업 미팅: 매일 짧게 모두 만나서 경과보고
  - 데모/회고 미팅: 스프린트의 마지막에 만나서 성과 공유 후 토론

❖ 데이터팀도 애자일 방법론을 사용하는 것이 일반적

◆ 월요일

❖ 지난 스프린트 리뷰

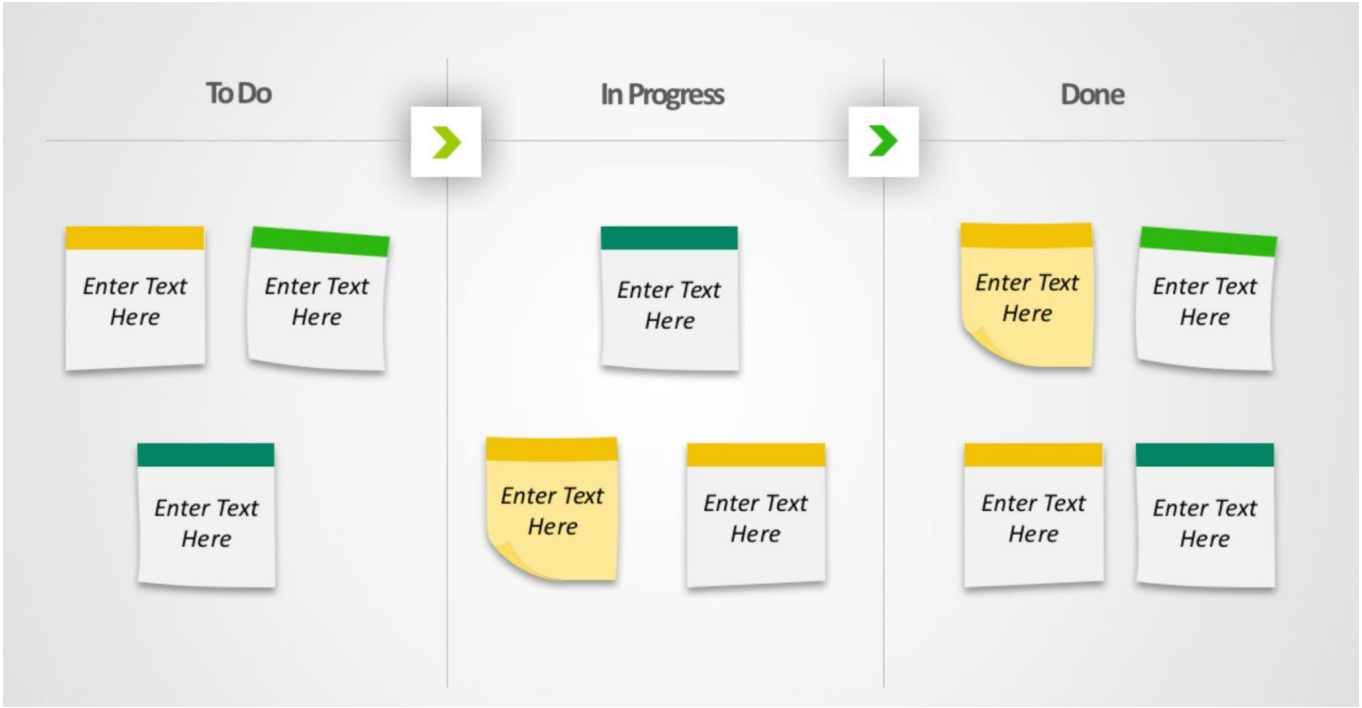
- 지난 스프린트에 한 일들을 리뷰: 각자 자기가 한 일을 데모
- 회고 미팅: 뭐가 잘 되었고, 뭐가 더 잘 될 수 있었고, 기타 논의할 점이 있는지?

❖ 새로운 한주 계획

- 이번 한주에 무슨 일들을 할지 결정
- 미팅 제외 하루 5시간쯤 일한다고 가정
- 30% 정도의 시간은 유지보수에 사용한다고 가정
- 온콜 엔지니어와 분석가 지정
  - 데이터 ETL 관련 이슈와 다양한 데이터 관련 질문을 맡을 사람들을 별도로 지정
  - 특히 ETL 관련 이슈 해결은 데이터 시스템의 안정성에 중요함

◆ 애자일/스크럼 보드

- ❖ JIRA가 가장 많이 사용됨
- ❖ Swit, ClickUp등의 후발주자도 많이 사용되는 추세



◆ 화요일

❖ 매일 스탠드업 미팅 (Daily Standup)

- 매일 5분 정도 모두가 모여서 다음에 대해 이야기
- 어제 무슨 일을 했는가?
- 오늘 무슨 일을 하는가?
- 어제나 오늘을 하면서 어떤 문제가 있는가?

❖ 다양한 미팅들 (화요일은 미팅의 날)

- 내부 팀원들과의 미팅
  - 데이터 엔지니어 <-> 데이터 분석가
  - 데이터 엔지니어 <-> 데이터 과학자
  - 데이터 분석가 <-> 데이터 과학자
- 다른 팀과의 sync-up 미팅
  - 마케팅, 프로덕트 매니저, 세일즈, ...

◆ 수요일/목요일

❖ 매일 스탠드업 미팅 (Daily Standup)

- 보통 비디오 콜로 하거나 경우에 따라서는 슬랙등의 메세지 툴로 대신하기도 함

❖ 중요 지표 리뷰 미팅

- 대시보드를 보면서 중요 지표에 어떤 변화가 있는지 살펴봄

❖ 머신러닝 모델 개발 리뷰 미팅

- 개발 중인 머신러닝 모델에 관련해서 전체적으로 리뷰하는 미팅
- 만일 A/B 테스트중인 머신러닝 모델이 있다면 지금까지의 성능에 대해 리뷰하고 성공여부 결정

◆ 금요일

❖ 매일 스탠드업 미팅

❖ 데이터팀 주간 스태프 미팅

- 중요 지표와 회사/팀 목표 리뷰
  - ETL 성공/실패 비율 리뷰
  - 머신러닝 모델 관련 리뷰 (개발상황, A/B 테스트 진행상황)
- 구인 상황과 팀내 중요인력 상황 점검
- 주간 사고 리뷰 - 필요하다면 별도 리뷰 미팅
- 메인 프로젝트 리뷰
  - 외부 프로젝트 (다른 팀과 협업하는 프로젝트)
  - 내부 프로젝트
- 팀/개인 업데이트

programmers

좋은 지표(KPI)란?

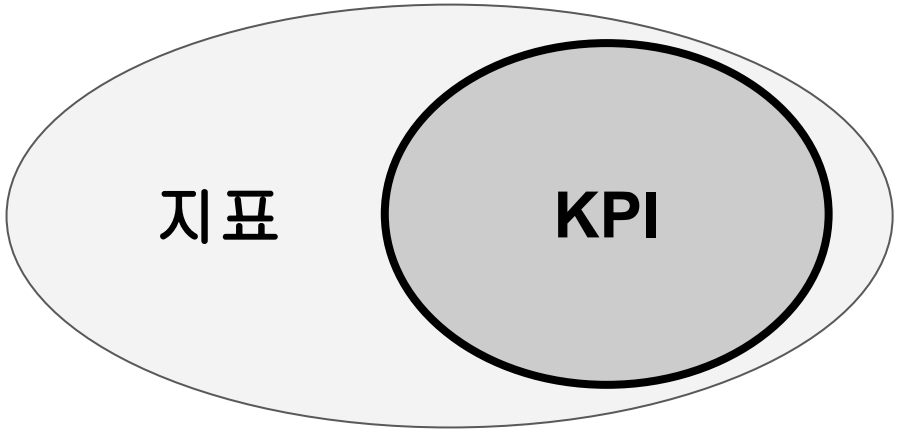
어떤 지표가 좋은 지표인지 알아보자!

KPI(Key Performance Indicator)란?

- 조직내에서 달성하고자 하는 중요한 목표
  - 보통 정량적인 숫자가 선호됨
  - 예를 들면 매출액 혹은 유료 회원의 수/비율
  - 명확한 정의가 *중요*함 -> 지표 사전이 필요
- KPI의 수는 적을수록 좋음
- 잘 정의된 KPI -> 현재 상황을 알고 더 나은 계획 가능
  - 정량적이기에 시간에 따른 성과를 추적하는 것이 가능
  - OKR(Objectives and Key Results)과 같은 목표 설정 프레임웍의 중요한 포인트

지표(Metrics)란?

- 지표와 KPI의 차이점은 중요도
  - KPI는 회사에서 중요한 지표. 즉 지표가 더 큰 개념
- 팀/개인별로 중요한 성과 목표를 정량적으로 갖는 것이 중요
- 데이터 문해력(Data Literacy)의 시작점



KPI 기준

- Represent delivery of real value
- Captures recurring value
  - MRR (Monthly Recurring Revenue) vs. Total revenue
- Lagging indicator (후행지표)
  - vs. Leading indicator (선행지표)
  - Registered users vs. Paid users
- Usable feedback mechanism
  - Used for decision making: WAU vs. MAU

좋은 지표의 특성

- 3A (Accessible, Actionable, Auditable)
- 쉽게 볼 수 있어야 함 (Accessible)
  - 지표를 보는 것이 쉬어야함 -> 시각화툴이 바로 여기서 도움이 됨
- 실행가능한 통찰력이 제공되어야 함 (Actionable)
  - 지표 등락의 의미가 분명해야함
- 감사가 가능해야 함 (Auditable)
  - 지표 계산이 제대로 되었는지 검증이 가능해야함
  - 데이터 기반이어야 가능

Next Dashboard Fallacy

- 기존 지표 기반 결정을 못하고 대시보드를 계속해서 만드는 현상
  - 의사결정 장애의 일종 :)
- 지표의 수는 적을수록 좋고 따라서 대시보드의 수도 적을수록 좋음
  - 대시보드의 수가 늘어나면 Dashboard Discovery 이슈가 발생함
- 비슷한 것으로 Next Feature Fallacy가 있음

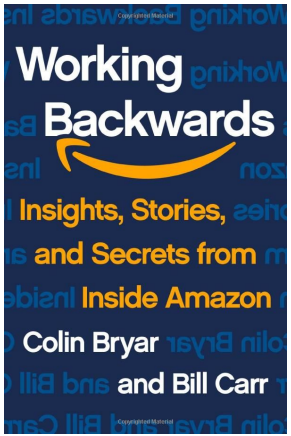
programmers

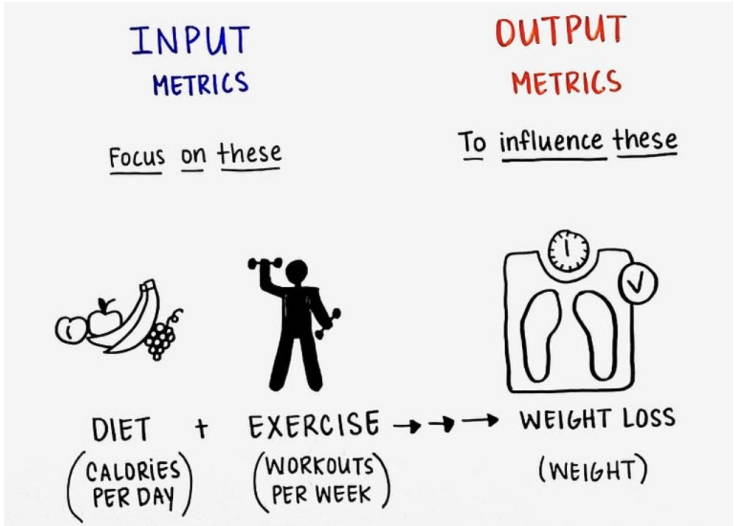
KPI와 선행/후행 지표 예

KPI의 몇 가지를 예를 들고 선행 지표와 후행 지표가 무엇인지 알아보자

Controllable Input Metrics vs Output Metrics

- Working Backwards라는 책의 6장에 있는 내용
- 입력에 초점을 맞춰서 출력에 긍정적인 영향을 끼쳐라
  - 인풋(input): 입력, 투입물
  - 아웃풋(output): 출력, 결과물





https://www.factoftheday1.com/

Controllable Input Metrics vs Output Metrics

- 인풋 지표: '아웃풋 지표를 움직이는 지표'이며, 직접 통제 가능한 지표
  - 예: 제품 다양성, 가격, 편의성, 새로운 강의들
  - 선행 지표 (Leading Indicator)
- 아웃풋 지표: 인풋 지표의 결과로, 직접 통제 불가능한 것
  - 예: MAU, 판매량, 계약건수, 매출, 이익
  - 후행 지표 (Lagging Indicator)

KPI와 선행 지표 예

- 매출액 (vs. Active Users)
  - 기존 고객 매출 (recurring) vs. 새로운 고객 매출 (new)
- 매출 = 가격(P) * 판매량 (Q)
  - P가 고정되었다는 전제하에서는 Q를 늘릴 방법을 찾아야함

- Q에 영향을 주는 인풋 지표 (선행 지표)는?
  - 예를 들어 일반 영업팀이라면 아웃바운드 영업 건수, ...?
  - 예를 들어 온라인 교육 사이트라면 온라인 강의 수, 사이트 방문자 수?
  - 시간을 두고 선행 지표를 발전시켜야함

두 가지 중요한 KPI

- **매출** vs. **서비스 사용 고객수** (DAU, WAU, MAU)
- 보통 매출이 훨씬 더 중요한 지표
  - 단 새 고객에서 발생하는 매출과 기존 고객에서 발생하는 매출을 따로 볼 것
- 네트워크 현상이 중요한 도메인에서는 “서비스 사용 고객수"도 중요한 지표
  - 이 때 “서비스 사용 (Active)”의 정의가 중요
    - 유료 고객 vs. 무료 고객
    - 마켓플레이스라면 콘텐츠 공급자 vs. 소비자

programmers

다양한 시각화 툴 소개

어떤 대시보드들이 있는지 알아보자

시각화 툴이란?

- 대시보드 혹은 BI(Business Intelligence)툴이라고 부르기도 함
- **KPI** (Key Performance Indicator), **지표**, 중요한 데이터 포인트들을 **데이터를 기반**으로 계산/분석/표시해주는 툴
- 결국은 결정권자들로 하여금 흔히 이야기하는 데이터 기반 의사결정을 가능하게 함
  - 데이터 기반 결정 (Data-Driven Decision)
  - 데이터 참고 결정 (Data-Informed Decision)
- 현업 종사자들이 데이터 분석을 쉽게 할 수 있도록 해줌

어떤 툴들이 존재하나?

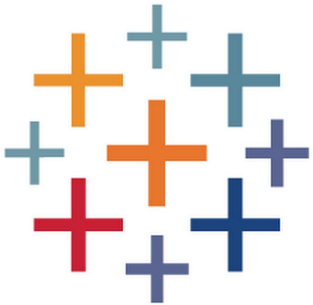
- Excel, Google Spreadsheet: 사실상 가장 많이 쓰이는 시각화 툴
- Python: 데이터 특성 분석(EDA: Exploratory Data Analysis)에 더 적합
- Looker (구글)
- Tableau (세일즈포스)
- Power BI (마이크로소프트)
- Apache Superset (오픈소스)
- Mode Analytics, ReDash
- Google Studio
- AWS Quicksight

Looker

- 2012년 미국 캘리포니아 산타크루즈에서 시작
- 구글이 2019년 6월에 $2.6B에 인수
  - 지금은 구글 클라우드의 일부
- 특징
  - LookML이 자체언어로 데이터 모델을 만드는 것으로 시작
  - 내부 고객뿐만 아니라 외부 고객을 위한 대시보드 작성가능
  - 고가의 라이센스 정책을 갖고 있으나 굉장히 다양한 기능 제공

Tableau

- 2002년 미국 캘리포니아 마운틴뷰에서 시작하여 2013년 상장
- 세일즈포스가 2019년 6월에 $15.7B에 인수함
- 특징
  - 다양한 제품군 보유. 일부는 사용이 무료
  - 제대로 배우려면 시간이 꽤 필요하지만 강력한 대시보드 작성가능
  - Looker가 뜨기 전까지 오랫동안 마켓 리더로 군림



어떤 시각화 툴을 선택할 것인가?

- Looker 혹은 Tableau가 가장 많이 사용되는 추세
  - 두 툴 모두 처음 배우는데 시간이 필요함
  - Tableau의 가격이 더 싸고 투명하며 무료 버전도 존재해서 공부가 가능
- 중요한 포인트는 셀프서비스 대시보드를 만드는 것
  - 안 그러면 매번 사람의 노동이 필요해짐
    - 60-70%의 질문을 셀프서비스 대시보드로 할 수 있다면 대성공
  - 또한 사용하기가 쉬워야 더 많은 현업 인력들이 직접 대시보드를 만들 수 있음
    - 데이터 민주화 (Data Democratization), 데이터 탈중앙화 (Data Decentralization)
    - 데이터 품질이 점점 더 중요해지며 데이터 거버넌스가 필요한 이유가 됨!
  - 이런 측면에서는 Looker가 더 좋은 선택이지만 가격이 상당히 비쌈

programmers

실습: 지표 정의하고 차트 만들어보기

Tableau 제품군 소개 (1)

- Tableau Desktop
  - 코어 제품으로 대시보드를 만들 수 있는 저작환경으로 맥용과 윈도우용 제공
- Tableau Server
  - 엔터프라이즈 레벨 플랫폼으로 사용자들간에 대시보드, 웍북, 데이터 소스등의 공유와 웹/앱으로 접근 가능
  - 중앙 플랫폼이기에 데이터 거버넌스, 보안 등을 제공
  - 소프트웨어를 구매하여 직접 설치하고 운영 필요
- Tableau Online
  - 클라우드 버전의 Tableau Server. 클라우드이기에 직접 설치하고 운영이 필요하지 않다는 장점 존재

Tableau 제품군 소개 (2)

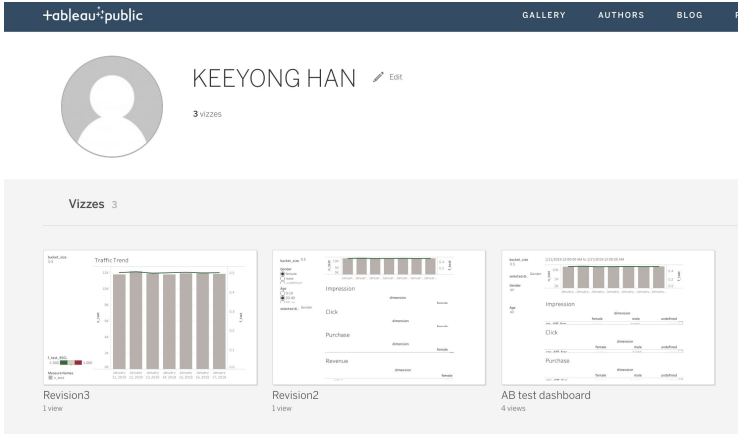
- Tableau Prep
  - 데이터를 대시보드에서 사용하기 전에 다양한 데이터 변환과 분석등을 코딩없이 하는 데이터 전처리 툴
  - Tableau Desktop, Tableau Server와 연동하여 사용되는 것이 일반적
- Tableau Public
  - 기능에 있어 제약이 있는 Tableau의 무료 버전으로 학습을 위한 용도로 많이 사용됨
  - 이걸 이번 강의에서 사용해볼 예정
- Tableau Mobile
  - iOS나 안드로이드 동작 앱으로 Tableau 대시보드 뷰어 용도로 사용됨

Tableau Public 소개

- 장점은 무료라는 것!
  - Tableau의 기능을 학습하는 용도로 사용 가능
  - 보통 Desktop 버전을 다운로드 받아 사용하는 것이 일반적
- 단점은 추출된 데이터 원본(CSV 파일)만 데이터 소스로 지원
  - 데이터에 대한 라이브 연결은 지원하지 않음
  - 최대 천5백만개의 레코드를 읽어올 수 있음
- 기타 특징
  - 내가 만든 대시보드는 기본으로 모두에게 공개가 되기 때문에 포트폴리오로 사용 가능

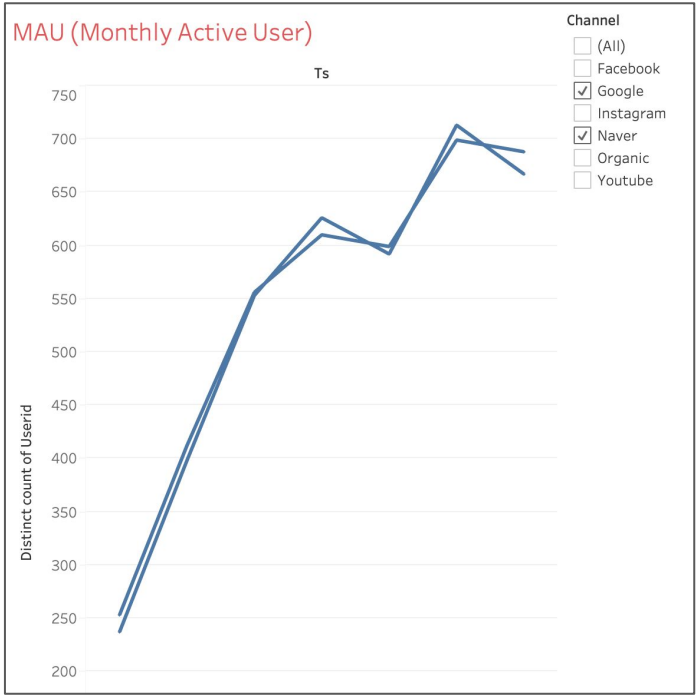
Tableau Public 설치

1. https://public.tableau.com/en-us/s/ 방문
2. 자신의 계정 생성
   a. “Sign-in”을 클릭
3. 데스크탑 버전 다운로드
   a. https://public.tableau.com/s/download
   b. 앞서 언급한 단점이 존재하지만 학습용도로는 충분



전체 과정 설명

1. user_session_summary.csv 파일을 스쿨 페이지에서 다운로드 받을 것
   a. user_id, ts, channel, session_id
2. 이를 Tableau Public으로 업로드
3. 다음으로 멀티라인 MAU 차트 생성
4. 이를 가지고 대시보드 생성
5. 최종적으로 대시보드 저장



programmers

Q & A

이번 강의에 질문이 있으면 알려주세요!