

# 오픈 소스 개발 협업 과정

charsyam@naver.com

# 일반 회사에서의 개발의 과정

- Trigger
- 업무 분장
- 개발
- 테스트
- 회고

# 일반 회사에서의 개발의 과정

- Trigger
- 업무 분장
- 개발
- 테스트
- 회고



가끔씩 슬프게도  
없는 조직들도 존재

# Trigger - 일반 회사

- Trigger
  - 연관자들로부터의 기능 요청
    - 기획자
    - 디자이너
    - 외부팀
    - 내부 개발자 or 자기 자신

# 업무분장 - 일반 회사

- 보통 연관성 있는 모듈을 개발했던 개발자
- 해당 기능에 관심을 보이는 개발자
- 일부러 "버스 넘버" 를 줄이기 위해서 업무 교체를 시키기도 함.

# 개발 - 일반 회사

- 사람이나 조직에 따라 차이가 많이 존재함.
  - 디자인 리뷰 -> 기능 구현 형태나...
  - 바로 기능 구현 형태도 존재...
- 실제 개발이 되어야만 제품이 존재
- 자동화된 테스트는 개발 단계에 존재

# 테스트 - 일반 회사

- 테스트 전문 조직이나 개발팀 또는 연관팀 형태의 유저 테스트
- 여기서 보통 이슈가 제기됨
  - 기획적인 의도와 다름...
  - 생각하지 않았던 부분의 버그들도...
- 테스트를 성공했다고 버그가 없는 것을 보장하지 않지만,
- 테스트에 문제가 있다면 버그가 있다는 것을 보장한다.
- 자동화된 개발자 테스트는 개발에 포함.

# 회고 - 일반 회사

- 해당 기능 구현이나 일정에 어떤 이슈가 있었는지
- 개선할 점이 무엇인지 확인하는 것



# 오픈소스 개발의 과정

- Trigger
- 개발

# Trigger - 오픈 소스

- Trigger
  - 누군가의 기능 요청이나 공헌으로 시작
- Contribution의 경우는 이미 개발 자체가 끝나있음.

# 개발 - 오픈 소스

- **오픈소스에 따라서 많이 다름**
  - 어떤 프로젝트들은 기능 제안시에 디자인 리뷰로 부터 시작(KAFKA)
    - 왜 필요하고 어떻게 구현할지 등에 대해서 논의를 하고 이를 통과해야만 PR을 받아줌
- **많은 프로젝트들이 PR과 함께 개발이 끝남**
  - 이미 기능 제안시에 코드도 함께 존재
- **PR 이후는 이제 커미터와 긴 리뷰의 시간을 거쳐야 함.**
  - 설계(디자인), 코드 구현 품질, 보안적인 이슈, 테스트 코드, 코딩 스타일등 지적거리가 많음.(라이센스 이슈도 큼)
  - 기능이 포함되면 그 때 부터는 커미터의 책임