**Scrum(스크럼) 개발방법론**

**1. 개요**

Scrum은 프로젝트 관리를 위한 상호, 점진적 개발방법론으로 애자일 소프트웨어 공학 중 하나이다. 소프트웨어 개발을 위하여 고안된 것이지만 일반적인 제품개발이나 유지보수 등에 활용이 가능하다.

***Product Backlog -> Sprint Backlog -> Sprint -> Working increment of the SW***

**2. Scrum 구성의 원칙**

1) 솔루션에 포함할 기능/개선점에 대한 우선 순위를 부여

2) 개발 주기는 30일 정도로 조절하고 개발 주기마다 실제 동작할 수 있는 결과를 제공

3) 개발 주기마다 적용할 기능이나 개선에 대한 목록을 제공

4) 일일 15분 정도의 회의

5) 항상 팀 단위로 사고

6) 원활한 의사소통

**3. 스크럼 팀 구성원이 추구하는 가치(Ground Rule)**

1) 확약 : 약속한 것을 확실히 실현하는 것

2) 전념 : 확얀한 것의 실현에 전념하는 것

3) 정직 : 어떤 것이 자신에게 불리해도 숨기지 않는 것

4) 존중 : 자신과 다른 사람에게 경의를 표하는 것

5) 용기 : 팀 구성원은 자신이 옳은 일을 할 수 있도록 팀원간 갈등과 도전을 통해 작업할 수 있는 용기

**4. Scrum 프로세스의 요소**

1) 제품 백로그(Product Backlog)

* 요구사항목록

2) 스프린트(Sprint)

* 반복주기

3) 스프린트 계획 회의

* 반복안에서 제품 백로그로부터 대상을 선정하고 개발, 사용자 테스트 시점을 협의하는 과정 (각자의 역할 및 수행범위를 합의하는 과정으로 자율적인 의사 결정을 통하여 진행)
* 결과물에 대한 반복 완료시의 모습을 결정하고 수행에 필요한 각종 요구사항은 Scrum마스터에게 보고하여 이해 관계들로부터 지원을 받을 수 있도록 함.

4) 스프린트 백로그(Sprint Backlog)

* 스프린터 계획회의를 통하여 정리된 작업 목록

5) 일일 스크럼 회의(Daily Scrum Meeting)

* 스프린터 백로그에서 오늘 완료한 목록을 팀원들과 공유하고 목록에서 삭제하는 절차 (팀원의 구성은 제품 책임자가 필요한 기능 및 역할을 규정하면 Scrum 마스터가 적임자를 선발하여 팀원으로 구성하는 방법과 제품 책임자가 직접 적임자를 선정하는 방법을 사용하기도 함.)

6) 실행 가능한 제품(Shippable product) 개발

* 일일 스크럼 회의에서 삭제되는 작업절차는 해당절차가 완료되었다는 것을 검증 함. (기존 개발 방법론들과의 가장 큰 차이점은 하나의 작업단위는 완료를 검증할 수 있는 수준으로 세분화 되어야 한다.)

7) 제품 책임자(Product Owner)

Product Backlog를 정의하며, 우선순위를 결정하는 사람, 주로 시스템 설계자.

8) 스크럼 마스터(Scrum Master)

* 프로젝트 관리자(Project Manager)를 스크럼 마스터 역할과 유사하나, 차이점은 이해관계자들로부터 프로젝트 지원을 받아내는 역할이 상대적으로 큼. 스크럼 팀원들에게 대한 코칭의 역할이 강조되고 있음.

**5. Scrum 사용시 주의사항**

- 일의 내용을 1~2시간 단위의 완성여부 판단이 가능하도록 불할하지 못하면 팀원들의 신뢰가 무너지게 되는 요인으로 작용함.

- 일일 단위로 결과를 정리하지 않으면, 스크럼 효과는 기대하기 힘듦.

- 지금 당장 할 수 있는 일을 즉시 실행하고 가장 빠르게 끝나는 작업을 팀원들이 할 수 있도록 하는 팀 문화를 만들어야함.

- 국내 프로젝트의 애자일 도입은 많이 시도를 하지만 실패를 하게 되는 가장 큰 이유는 제품 책임자가 만들어내는 Product Backlog에 이해 부족으로 인한 경우와 역할에 대한 적임자 선정 시 오류를 범하는 경우가 많다.

<참고>

Scrum VS Waterfall 개발 방법론의 차이점

-Waterfall 방법론은 우리가 흔히 알고 있는 기획을 하고 개발을 들어가 계단형식의 개발방법이다. 따라서, 중간중간 이전단계로 돌아가서 다시 확인하고 수정해야 한다는 단점이 있어서 불필요한 시간이 생성된다.

-Scrum은 하나의 기능 단위. 즉, 모듈단위로 개발을 진행하는 것으로 모듈단위로 개발을 하기 때문에 기능하나를 구현하는 개발기간이 짧다. Waterfall 방법론에서 생성되는 불필요한 시간이 생길 가능성이 적어, 같은 시간 동안 좀 더 효율적으로 개발을 할 수 있다.