



# 1 위험의 정의와 종류

# 01 위험의 정의와 종류


## 1 위험에 대한 정의 (일반적인 정의)


-  예상되는 위협에 의하여 자산에 발생할 가능성이 있는 손실의 기대치
-  자산의 가치 및 취약성과 위협 요소의 능력, 보호 대책의 효과 등에 의해 영향을 받음

※ 출처 : 네이버백과사전

# 01 위험의 정의와 종류

## 2 위험에 대한 정의 (프로젝트 관리의 입장에서)

 일정, 비용, 범위, 품질과 같은 프로젝트 목적들 중 적어도 하나의 목적에 나쁜 영향을 미칠지도 모르는 이벤트 혹은 상태

 “Project risk is an uncertain event or condition that, if it occurs, has a negative effect on at least one project objective, such as time, cost, scope or quality”

※ 출처 : PMBOK

# 01 위험의 정의와 종류

## 3 위험관리는?







프로젝트 성공을 위해 위험에 대비하고 관리하는 활동

- 위험관리 계획
- 위험 식별
- 정서적 위험분석과 정량적 위험분석
- 위험대응계획
- 위험 감시와 통제

# 01 위험의 정의와 종류

## 4 위험요소

-  인력 관리
  - 예기치 못한 인력의 추가 또는 변경
-  예산 관리
  - 예산의 증대
-  일정 관리
  - 전체적인 일정 지연 가능성
-  사용자 요구사항 변경
  - 프로젝트 시작후에 사용자의 요구사항 변경

# 01 위험의 정의와 종류

## 5 위험의 종류

- 🔍 알려진 위험(Known Risk)
  - 프로젝트 계획서, 기술적 환경, 정보 등에 의해 발견될 수 있는 위험
- 🔍 예측 가능한 위험(Predictable Risk)
  - 과거 경험으로부터 예측할 수 있는 위험
- 🔍 예측 불가능한 위험(Unpredictable Risk)
  - 사전에 예측이 매우 어려운 위험

※ 출처 : <http://yimma.tistory.com/128>

# 01 위험의 정의와 종류

## 5 위험의 종류



예측 가능한 위험의 종류

- 제품 크기 : 제작 또는 수정될  
소프트웨어의 크기에 대한 위험
- 비즈니스 영향 : 관리나 영업에 대한 위험
- 고객 특성 : 고객의 부당한 요구,  
의사소통에 관련된 위험
- 프로세스 정의 : 소프트웨어 개발 과정상의 위험
- 개발 환경 : 개발에 사용되는 도구 및 지원상의 위험
- 기술진의 규모와 경험  
: 기술진의 규모 및 프로젝트 경험과 관련된 위험

※ 출처 : <http://yimma.tistory.com/128>

# 01 위험의 정의와 종류

## 5 위험의 종류



### 비즈니스 위험(Business Risk)

- 소프트웨어의 생존 가능성을 위협하는 것으로, 원치 않는 제품이나 전략에 맞지 않는 제품 등을 개발하게 할 수 있음
  - 시장위험 : 아무도 원하지 않는 SW를 개발
  - 전략위험 : 조직의 전략과 무관한 SW를 개발
  - 사용위험 : 사용 하지도 않고 팔리지도 않을 SW를 개발
  - 경영위험 : 경영층의 지원이 없는 SW의 개발
  - 지원위험 : 예산의 지원이 중단되는 SW의 개발



# 01 위험의 정의와 종류

## 5 위험의 종류



프로젝트 위험(Project Risk)

- 프로젝트 계획을 위협하는 것으로, 일정이 지연되고 비용이 증가하게 됨
  - 예산, 일정, 인력, 자원 등에 문제가 발생
  - 고객 및 요구사항 문제가 프로젝트에 지장을 초래
  - 프로젝트의 복잡도 증가가 위험을 초래

# 01 위험의 정의와 종류

## 5 위험의 종류



### 기술 위험(Technical Risk)

- 소프트웨어의 품질이나 시기를 위협하는 것으로, 구현이 어려워지거나 불가능하게 됨
  - 기술적으로 구현이 불가능
  - 이미 퇴보된 기술로 SW를 개발하고 있음

## 2 위험의 이슈 및 문제

## 02 위험의 이슈 및 문제

### 1 위험의 본질



#### 불확실성

- 위험은 발생할 수도 있지만 발생하지 않을 수도 있음



#### 손실

- 위험이 실제로 발생되게 되면 원하지 않는 결과나 손실이 발생함

## 02 위험의 이슈 및 문제

### 2 위험의 이슈



#### 위험의 이슈화

- 위험의 발생으로 임박해진 상태  
객관적인 지표에서 이상이 나타나기 시작함
- 여러 가지 징후가 나타남  
한가지 또는 여러 가지 징후가  
동시에 나타나기 시작함
- 이미 이슈화가 되어 소문이 돌기도 함  
개발자보다는 외부에서 먼저 감지할 수도 있음

## 02 위험의 이슈 및 문제

### 2 위험의 이슈

#### 위험과 이슈의 구분

| 구분   | 위험  | 이슈                                  |
|------|---|-------------------------------------|
| 정의   | 원치 않는 결과를 초래하게 될 발생 가능한 미래의 사건            | 현재 프로젝트에 장애 혹은 도움이 되는 영향을 미치게 되는 문제 |
| 발견시점 | 프로젝트 진행 중에 발견될 수도 있으나 진행 전에 모두 발견하는 것을 추구 | 모두 프로젝트 진행 중에 발견                    |

※ 출처 : [http://korea-sw-eng.blogspot.com/2016/04/blog-post\\_41.html](http://korea-sw-eng.blogspot.com/2016/04/blog-post_41.html)

## 02 위험의 이슈 및 문제

### 2 위험의 이슈

#### 위험과 이슈의 구분

| 구분   | 위험   | 이슈   |
|------|--|--|
| 처리시점 | <ul style="list-style-type: none"><li>- 발견 후 당장 해결하지 않아도 프로젝트는 진행 가능</li><li>- 일부는 완화전략이행으로 발생가능성 통제</li></ul> | <ul style="list-style-type: none"><li>- 발견되면 즉시 처리</li><li>- 발생이슈를 해결하지 못하면 프로젝트 진행에 차질 발생</li></ul> |

※ 출처 : [http://korea-sw-eng.blogspot.com/2016/04/blog-post\\_41.html](http://korea-sw-eng.blogspot.com/2016/04/blog-post_41.html)

## 02 위험의 이슈 및 문제

### 3 위험의 문제



#### 문제 해결

- 이미 발생하여 돌이킬 수 없음
  - 가역적인 문제 : 사용자 요구사항 수정 및 반영
  - 비가역적인 문제 : 지출한 비용 및 투입된 인력
- 해결을 위해서는 시간과 노력이 필요
  - 비가시적인 손실(예, 신용도 추락, 상실감)은 회복하기가 어려움



### 3 위험관리 원칙

## 03 위험관리 원칙

### 1 위험관리에 대한 7가지 원칙




- 🔍 전체적인 관점을 유지
- 🔍 미래지향적 관점을 유지
- 🔍 열린 대화로 격려
- 🔍 위험을 고려한 SW개발 프로세스
- 🔍 지속적인 위험 관리 프로세스의 강조
- 🔍 SW에 대한 비전을 공유
- 🔍 팀웍을 통한 단합

※ 출처 : SEI

## 03 위험관리 원칙

### 2 위험관리에 대한 7가지 원칙 설명

#### 전체적인 관점을 유지

-  다양한 요소를 고려해야 함
-  한 위험의 제거가 다른 위험의 발생을 초래할 수 있음
-  다양한 관점과 프로젝트 수행의 우선순위를 고려해야 함

## 03 위험관리 원칙

### 2 위험관리에 대한 7가지 원칙 설명




#### 미래지향적 관점을 유지

- 🔍 이미 지난 문제의 해결비용과 시간에 매달리는 경우 더 큰 위험을 초래할 수 있음
- 🔍 미래의 관점에서 발생할 위험을 미리 예견하고 영향도를 분석해야 함
- 🔍 이슈화가 되기 전에 미리 위험을 예측해야 함

## 03 위험관리 원칙

### 2 위험관리에 대한 7가지 원칙 설명




#### 열린 대화로 격려

-  대화의 경직은 문제를 키울 수 있음
-  조직의 상명하복의 관계는  
위험관리에 있어서 매우 부정적임
-  조직 내에서의 커뮤니케이션 채널을 다양화하여  
언제 어디서나 위험요소를 공유하는 자세가 필요

## 03 위험관리 원칙

### 2 위험관리에 대한 7가지 원칙 설명




#### 위험을 고려한 SW개발 프로세스

-  소프트웨어 개발과정 중에 언제든지 위험이 발생할 수 있다는 가정을 가지고 출발
-  복잡한 소프트웨어 개발에 있어서는 위험관리 단계를 추가할 필요가 있음
-  주기적인 위험요소 모니터링이 필요함

## 03 위험관리 원칙

### 2 위험관리에 대한 7가지 원칙 설명




#### 지속적인 위험 관리 프로세스의 강조

-  상시적인 위험 관리가 필요함
-  위험관리 전체에 대한 문서화 작업이 필요함
-  위험 식별단계부터 위험조치단계까지 전반적인 관리방법과 체계가 필요함

## 03 위험관리 원칙

### 2 위험관리에 대한 7가지 원칙 설명

#### SW에 대한 비전을 공유




-  사용자와 개발자, 테스터 등 다양한 구성원 들 사이에 개발할 소프트웨어에 대한 비전을 서로 공유할 필요
-  비전에 따라 위험을 보는 시각이 달라지고 우선순위가 달라질 수 있음
-  비전공유는 위험요소 식별 및 조치에 대한 장기적인 위험관리 계획을 유도가능



## 03 위험관리 원칙

### 2 위험관리에 대한 7가지 원칙 설명

#### 팀웍을 통한 단합

-  리더십과 더불어 팀웍은 성공적인 소프트웨어 완성의 필수적 조건
-  팀웍이 깨질 경우 전체적인 비용증가와 일정지연 및 품질저하를 초래할 수 있음
-  주기적인 온오프상의 의견공유 또는 친목이 필요