

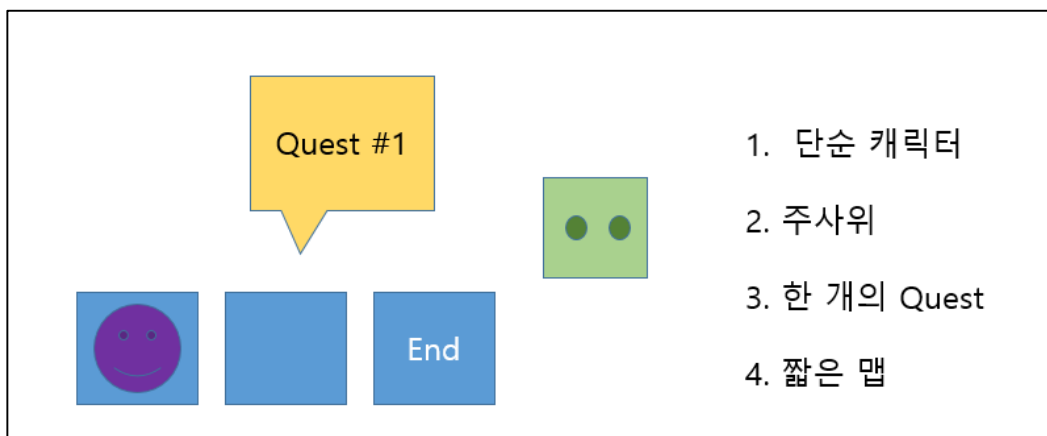
## 가상 프로젝트

화면상에 맵이 있고, 주사위를 굴리면 이동을 하며 각 칸에는 퀘스트가 있다.

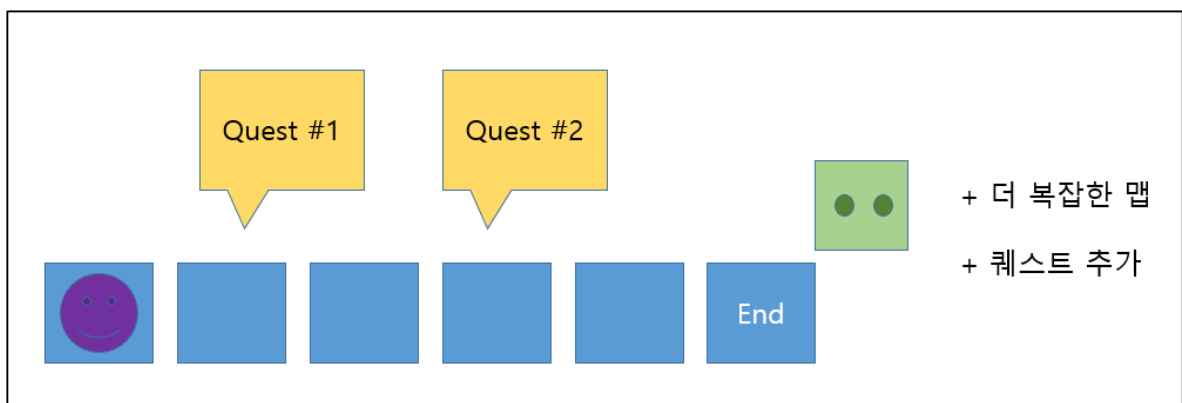


## MVP(MIN VIABLE PRODUCT)

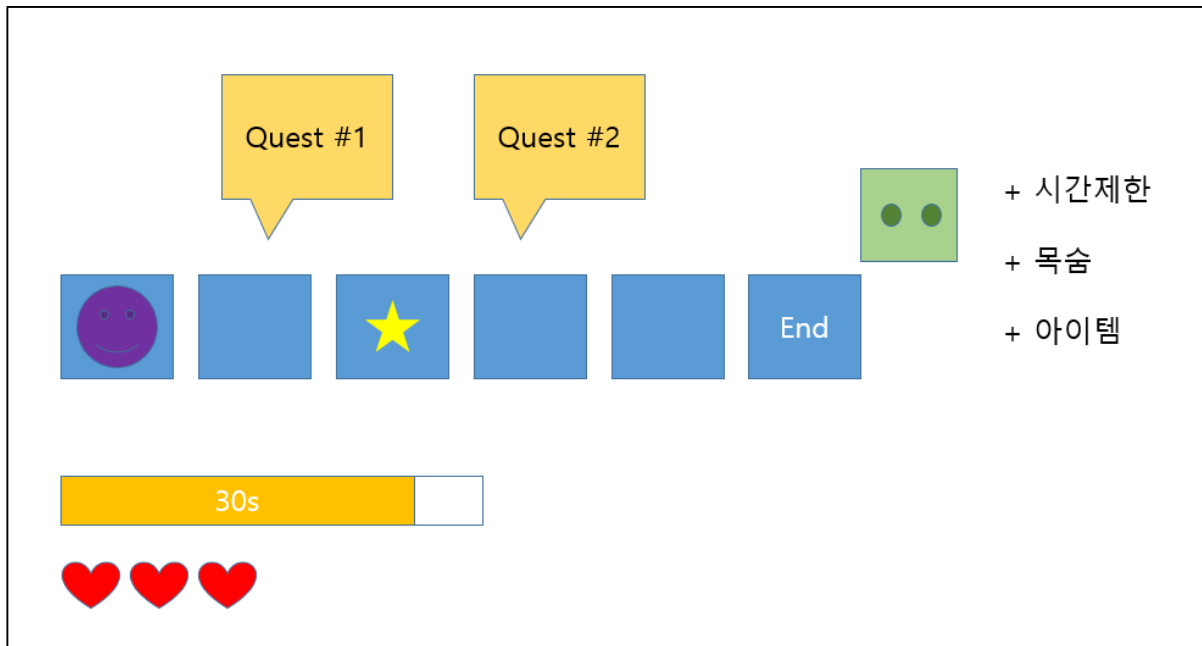
### 1. MVP-1



### 2. MVP-2



### 3. MVP-3



### 4. MVP-4

+ Design(생각보다 중요!!)

+ More Stage(1 단계, 2 단계, 보스 맵)

## GIT 을 통한 통합관리(MIN. GOOGLE DRIVE)

**핵심 : 버전관리를 해야한다!!**

#### 1. 버전 관리 유형

- A. 프로젝트 작업 날짜 - 매주 토요일(1,2,3,4,파이널)
- B. MVP 버전 - MVP1, MVP2, MVP3
- C. 특정 기능을 기준으로 - 캐릭터, 맵, 아이템 완성 등등

#### 2. Git 을 통한 버전 관리

- A. 매 소단위 기능 마다 로그를 남김
- B. MVP 단위로 버전 관리

#### 3. 구글 드라이브

- A. 프로젝트 작업 날짜 or 특정 기능 기준
- B. 폴더 복사본이 여러 개 있어야 함



2 명이 한 조가 되어 **컴퓨터 1 대로** 작업을 하는 것. 30 분 정도마다 번갈아 가면서 작업을 수행한다.

### 1. 장점

- A. 두 개의 컴퓨터로 작업을 할 때 보다 실수 감소
- B. 시간 절약
- C. 두 명 모두 한 파일의 진행 상황을 알고 있기 때문에 공백을 매울 수 있음(소통 시간 감소)
- D. 다른 한 명이 가이드라인 제시(다음 작업 지시)
- E. 보고 있는 사람이 객관적으로 상황을 볼 수 있음

### 2. 단점

- A. 참여도가 다를 수 있음
- B. 코드 스타일로 싸울 수 있음
- C. 강 짜증남...

## 코드 규약

여러 명이 프로그래밍을 할 때에는 코드 규칙이 필요함 – 익숙한 프로그래머들에게는 규약이 아닐 수도 있음(자연스럽게 녹아들어서)

- 1. 변수 이름은 명사, 함수 이름은 동사 (ex- var height, function makeRanNum)
- 2. 변수 이름이 길더라도 정확한 의미를 담자 (ex – var a) NONONO
- 3. 변수 이름은 소문자로 작성. 단어가 2 개 이상이면 다음 단어를 대문자로 구분
- 4. 상수 이름은 전부 대문자로

## 5. Js 파일 이름은 대문자로 시작

### 파일 분리

	Function	Style
Map	Map.js	Map.style.js
Player	Player.js	Player.style.js
Dice	Dice.js	Dice.style.js
Item	Item.js	Item.style.js
Timer	Timer.js	Timer.style.js
Life	Life.js	Life.style.js

GameManager.js (Main.js)

### 애자일 개발 방식

1. MVP
2. Pair-programming
3. 매 프로젝트 시작 마다 회의(20~30 분). 현재 상황 진행