

NFR of the Game project

분류	고려사항 혹은 제한사항	결론
사용 언어	<ul style="list-style-type: none"> -프로그래머의 실력 -진행할 프로젝트에 대한 적합성 -학교에서 배포된 프로젝트 명세서 	-조사 결과, 자바의 GUI 라이브러리 등이 성능이 그리 좋지 않다고 판명됨. 또한 프로그래머가 자바보다 C++을 훨씬 잘 다룸. QT를 사용하려고 했으나, 직접 다뤄본 결과, QT를 잘 다루기까지 비용이 오래 걸릴 거라 생각하여, 포기. C++기반으로 콘솔로 제작하기로 방침변경.
프로젝트 진행	<ul style="list-style-type: none"> -프로젝트의 목적 -프로젝트의 스케일 	<ul style="list-style-type: none"> -‘소프트웨어 공학’을 고려하여, agile한 방법으로 코드를 개발하기로 결정. Iteration을 반복하고, 이후 ‘요구정의 및 분석 산출물’들은 일정 시간마다 꾸준히 업데이트 될 것. -그러나, 프로젝트의 스케일이 작고, 투입인원이 한명 뿐이므로, iteration이 매우 적게 반복될 확률이 높음 waterfall적인 경향이 강하게 들어날 것.
성능, 품질, 디자인	<ul style="list-style-type: none"> -프로젝트의 목적 	<ul style="list-style-type: none"> -코드 최적화는 평가 대상이 되므로, 특히 신경써서 디자인하고 프로그래밍 할 것. -OOAD 방식으로 진행해야 하므로, 이를 적극 고려하여 디자인 할 것.
확장성, 재사용성	<ul style="list-style-type: none"> -‘게임’이라는 프로젝트의 분류 특징 	<ul style="list-style-type: none"> -맵에서 왼쪽에서 오른쪽으로만 진행되는 경향이 것이 나중에 업데이트로 인해 여러 방향으로 가는 경향으로 업데이트 될 수 있으므로 이를 고려할 것. -그 외 여러가지 게임적인 기능들이 확장될 수 있으니 이를 디자인, 코딩할 때 고려할 것.
사용하는 다이어그램 및 도구	<ul style="list-style-type: none"> -해당 프로젝트의 특성, 볼륨 	-비즈니스 적인 분석이 필요없고, 구현하려는 프로그램의 볼륨이 작으므로, 사용되는 분석 산출물 등이 한정적임.