

# 공개소프트웨어 PROGRESS REPORT

SUPREMETIM 8조

2014112103 조인택, 2016112105 박선희, 2016112141 양시연

## 1. Introduction – 소개

이 보고서는 공개소프트웨어 프로젝트 진행 상황 보고를 하기 위해 작성된 중간 리포트이다. 2018년 5월 2일부터 2018년 5월 29일까지 진행된 상황을 보고 할 것이다.

이 보고서에서 우리는 우리 프로젝트에 대한 설명을 하고 진행사항에 대해 요약할 것이다. 또한 우리가 마주친 문제는 무엇인지 바뀐 요구사항은 무엇이 있는지에 대해 기술 한 후 이에 따른 전반적인 평가를 할 것이다.

## 2. Project Description – 프로젝트 설명

SUPREMETIM의 **프로젝트 목표**는 오픈 소스 기반의 테트리스 게임에 더 많은 기능을 추가하여 우리 SUPREMETIM만의 차별화된 테트리스를 설계하는 것이다.

**주요 이슈 및 범위**는 다음과 같다.

- 아이템 사용 기능 추가 및 속도 시스템 수정
  - 멀티 플레이어의 즐거움을 더하기 위해 아이템 사용 기능 추가 및 단순로운 플레이를 지양하기 위해 속도 시스템 수정
- 점수 시스템 추가와 Database를 이용한 랭킹 시스템 구현
  - 기존에 없었던 점수 시스템을 추가하고 Mysql를 사용하여 랭킹 저장 시스템을 구현. 게임을 종료하더라도 점수가 데이터베이스에 그대로 유지.
- 리듬에 맞춰 블록을 떨어뜨릴 때 추가 점수를 주는 메소드 구현
  - 리듬게임 오픈소스를 사용하여 테트리스에 접목. 싱글 플레이어에서만 적용이 되며, 리듬에 맞춰 블록을 떨어뜨리면 추가 점수를 부여

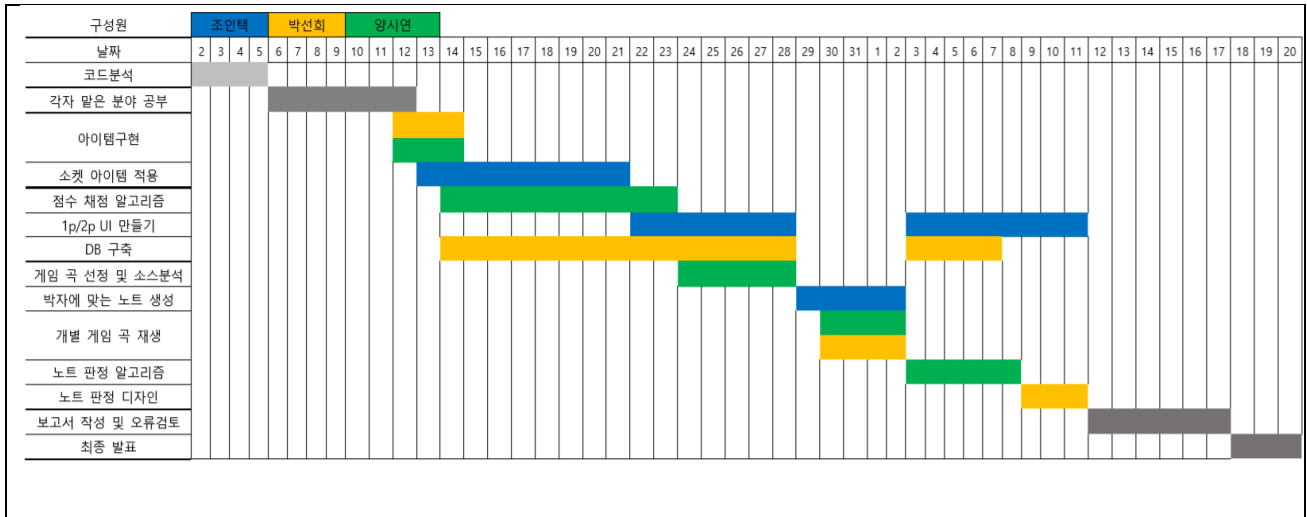
**팀원의 역할**에 대해서 소개를 하자면,

조인택(조장) – 아이템 구현 및 속도 시스템 수정

박선희 – DataBase 구축 및 관리

양시연 – 레벨 시스템 구현 및 리듬게임 메소드 구현

프로젝트는 2018년 5월 2일부터 6월 20일까지 진행 될 예정이며, 초기 간트차트는 아래와 같다.



### 3. Progress Summary – 진행요약

우리 팀이 **완료한 일, 진행중인 일, 앞으로 해야 할 일**은 다음과 같다.

#### Work Done

조인택 - 속도 버튼 제거. 랜덤으로 상대방에게 아이템 전달하기. 상대방의 속도를 5초간 늘려주는 아이템, 자신의 블록 라인을 지우는 아이템 구현.

박선희 - 점수 채점 알고리즘 구현. DB연결 및 DB의 USER TABLE에 점수 저장.

양시연 - 점수 채점 알고리즘 구현. 레벨 시스템 구현. 점수와 레벨을 표시하여 UI를 수정. 그리고 리듬게임에 사용할 게임 곡 선정 및 소스 분석.

#### Work in progress

조인택 - 아이템 생성 기준 밸런스 조정.

박선희 - MySQL connector JDBC Driver maven 디펜던시에 추가

양시연 - 리듬게임 구체화

#### Work to be done

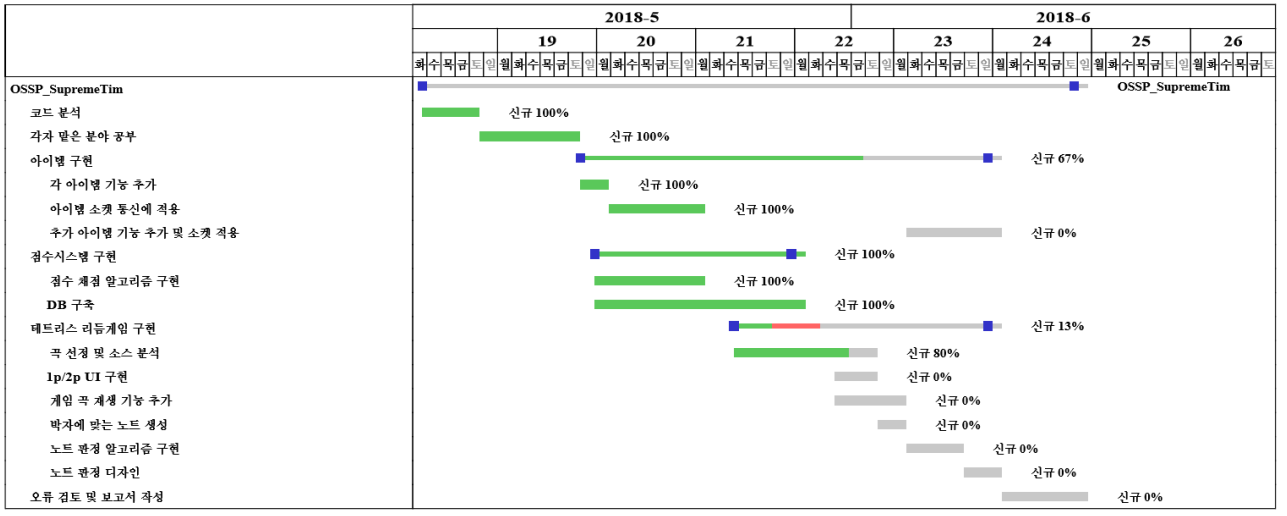
조인택 - 나머지 아이템 구현. 박자에 맞는 노트 생성.

박선희 - 개별 게임 곡 재생, 노트 판정 디자인.

양시연 - 노트 판정 알고리즘 구현.

## <RedMine 에서 이루어진 일감 Gantt차트>

### OSSP\_SupremeTim



## 4. Problems Encountered – 마주친 문제

- <랜덤한 client> : 처음에는 소켓을 통해 상대방에게 아이템을 보내는 것만 생각했는데 최대 3명의 클라이언트까지 이용이 가능하였다. 그래서 어떤 client에게 아이템을 보내게 될지 또한 정하기로 하였다. 그리고 아이템이 보드판에 띄워지고 그 아이템을 획득하는 방식으로 생각하였다. 하지만 지워진 줄 수의 콤보가 채워 졌을 때, 구현된 아이템 중 랜덤하게 아이템이 상대방에게 보내는 형식으로 구현하였다.

- <RDS>: DB를 연결함에 있어서 local로 연결하게 되면 본인 컴퓨터만 DB에 연결이 가능하다. 따라서 어떤 IP 이든 DB에 접근하기 위해서 AWS 서비스를 사용하였다. AWS의 RDS를 사용하여 도메인(mytestdb.c0l7ddni7jh4.ap-northeast-2.rds.amazonaws.com)의 DB를 연결하였다. 그리고 mysql table, Import와 export를 편리하게 개발하기 위해서 MYSQL WORKBENCH를 사용하였다. Mysql connector java를 build path에 외부 .jar 파일 추가하여 mysql과 java를 연결하였다.

- <Maven>: git을 이용하여 팀 프로젝트를 할 때, 다른 팀원들의 개발환경을 전부 맞춰주기에 너무 번거로웠다. 따라서 메이븐의 라이브러리 관리 기능을 사용하기로 하였다. 그리고 메이븐은 팀프로젝트가 Java 개발툴로 사용하는 이클립스에 기본으로 탑재 되어 있으므로 별다른 설정 없이 그냥 사용하면 된다. pom.xml이라는 파일에 우리가 만들 App에 어떠한 라이브러리를 사용할 것인지 적어 놓기만 하면 Maven이 알아서 다운 받고 알아서 설치해 주고 경로도 지정해준다. pom.xml파일만 있으면 Maven이 설치되어 있는 IDE라던지 혹은 시스템에 설치가 되어 있으면 Maven빌드를 통해 해당 파일을 다른 곳에서도 쉽게 설치해준다. 따라서 메이븐을 공부하여 팀원들에게 더 편리하게 개발 환경을 제공하기로 하였다.

## 5. Changes in Requirements – 요구사항 변화

- <랜덤한 client>: 2p라고 하였지만, 실제로는 수많은 client가 접속이 가능하다. 따라서 아이템을 상대방에게 전송할 때는 접속되어 있는 client 중 랜덤한 한 client에게 전송하기로 하였다.
- <아이템사용>: 원래 기능이었던 HOLD기능을 제거하고 hold기능을 수행하였던 tab 키를 이용하여 보유한 아이템을 사용하게 하려고 하였으나, 실제로 게임을 진행해보면서 몇 가지의 궁금증이 들었다. 우선적으로 상대방의 진전속도를 알지 못하기 때문에 굳이 아이템을 가지고 있으면서 전송하고 싶을 때 전송할 필요가 없다고 생각하였다. 그리고 더 경쟁적인 요소를 위해 hold 기능이 필요 없다고 생각하였으나, 아이템을 상대방에게 전송하기 위해서는 지워진 줄 수의 콤보를 채워야하기 때문에 HOLD 기능이 더 필요하다는 생각이 들었다. 따라서 hold기능을 유지하고 아이템 사용을 가능하게 하였다. 이러한 변화로 인해서 한 블록으로 인해 여러줄이 지워지면서 상대방에게 아이템을 보내게 될 때, 짜릿함과 즐거움을 느낄 수 있게 되어 즐거움을 극대화하는 본연의 목표를 달성하였다.
- <1p/2p UI 구현 삭제>: 리듬게임은 1p용으로 멀티플레이는 2p용으로 생각하여서 처음 접속 시에 1p와 2p를 구분하기로 하였다. 하지만 서버로 접속한다면 자연스럽게 1P로서 진행이 되고, 같은 IP로 클라이언트로 접속하게 된다면 멀티플레이가 진행된다. 따라서 1P/2P UI가 굳이 필요 없다고 생각하여 구현을 삭제하였다.
- <DB>: 원래의 목적은 실제 오락실 게임게임이 끝나고도 점수를 확인할 수 있게 하는 것이었다. 따라서 local DB 가 아닌 aws 서비스의 MySQL 5.6.39 인스턴스를 사용하였다. 이로 인해 다른 컴퓨터에서 build 된 파일을 실행할 때에도 게임이 끝나고 점수를 확인할 수 있게 하였다.
- <Maven>: 팀 프로젝트인 만큼 팀원간의 협력이 가장 중요하다는 목표를 세웠었다. DB작업을 하면서 여러 환경요소들이 추가가 되었다. 팀원들이 GIT에서 pull하여 개발을 진행할 때 개발환경을 맞추기 위해서 maven을 사용하기로 하였다.

<변화된 타임라인 5.30~6.20>																						
구성원	조인택			박선희				양시연														
	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
코드분석																						
각자 맡은분야 공부																						
아이템구현																						
소켓 아이템 적용																						
아이템 밸런스 조정																						
점수 채점 알고리즘																						
1p/2p UI 만들기																						
DB 구축																						
게임 곡 선정 및 소스 분석																						
박자에 맞는 노트 생성																						
개별 게임 곡 재생																						
노트 판정 알고리즘																						
노트 판정 디자인																						
보고서 작성 및 오류검토																						
최종 발표																						

## 6. Overall Assessment of the Project – 전반적인 평가

Professional opinion: 서버와 Database를 관리하기에 가장 편한 리눅스를 이용하여 훨씬 수월했다. 뿐만 아니라 Java는 서버측 기술을 관리하기에 적합한 언어이기에 더 쉽게 접근할 수 있었다.

또한 원래 하려고 했던 c언어보다 객체지향적인 Java언어를 사용하면서 thread와 동기화를 통해 더 쉽게 구현할 수 있었다. 그리고 이러한 클래스를 잘 활용하여 이미 오픈소스에 구현된 함수들이 많았다. 아이템을 개발할 때 이미 구현된 함수들을 사용함으로써 메모리 낭비를 줄이고 효율적으로 구현할 수 있었다.

자바를 이번에 수강하는데 소켓프로그래밍까지 하려고 하니 초반에 너무 버거웠다. 그리고 아이템이 제대로 사용이 되는지에 대한 테스트가 많이 필요했다. 일일이 특수 상황을 만들어주면서 테스트를 하자니 좀 까다로웠다. 하지만 오히려 이 프로젝트 덕분에 자바의 새로운 기능들을 알게 되면서 자바언어만 수강하는 것보다 더 성장하게 되는 계기가 되었다.

공개SW 프로젝트 말고도 다른 일에도 치이다 보니 내가 계획했던 일들을 시간에 맞춰 하지 못하고 미루는 일이 많이 생겼다. 계획을 짤 때는 다른 요소들을 많이 고려해서 짜야겠다는 것을 느꼈고, 좋은 계획을 짰 후에는 어떻게든 그 날짜를 맞춰서 일을 수행해야겠다는 생각을 했다

지금까지 순탄하지만은 않았지만 팀원들과 협력하면서 처음 세웠던 목표들을 차근차근 이루어갔다. 또한 팀 프로젝트 진행 중 REDMINE, GITHUB을 통해서 협업하는 방법과 그 중요성을 알게 되었다. 특히 GIT을 중심으로 다루면서 실전에서 사용방법에 대해 익숙해졌다. 이제는 조금의 밸런스 조정과 리듬게임 기능 구현이 남아있다. 지금까지 해왔던 것처럼 협업하여 남들과 차별화된 테트리스를 만들기 위해 노력하기로 하였다.