

# 웹 및 모바일 최신기술 Term Project 설계서

팀 명	NeoPul				
작 품 명	GAIA( GArdening In App )				
개발기간	년 4 월 21 일 ~ 년 6 월 20 일				
담당교수	컴퓨터공학과			오병우 교수	
구 분	학년	학 번	성 명	휴대전화	E-mail
책임자(팀장)	4	20141280	진연주	6650-7951	aurora32s@naver.com
팀원1	4	20141238	조수정	9966-8334	<a href="mailto:qqq3444@gmail.com">qqq3444@gmail.com</a>
팀원2	2	20140342	김태우	4904-7895	ealiza@naver.com
팀원3	2	20141084	임준혁	2678-0724	dbghkd0724@naver.com

본인은 웹 및 모바일 최신기술 Term Project 설계서를 첨부과 같이 제출합니다.

2017년 5월 29일

팀 장 진 연 주 (서명)

컴퓨터공학심화프로그램

---

# GAIA( GArdening In App )

## 1. 개요

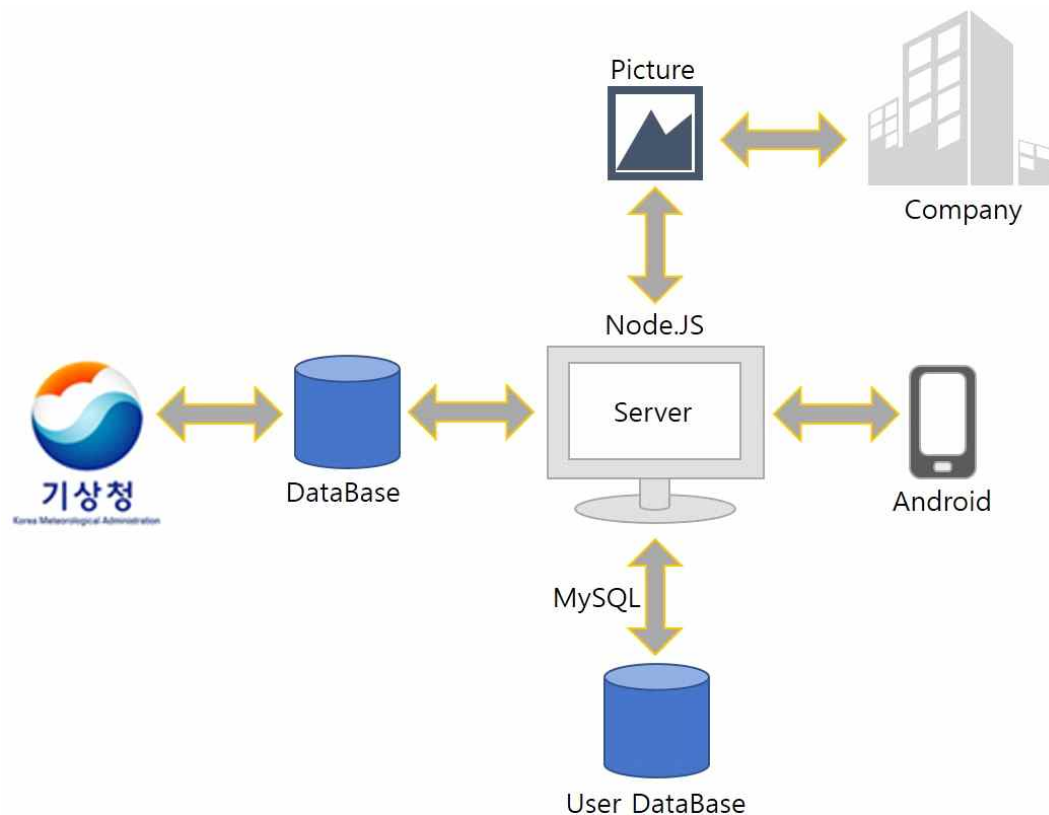
□ 스마트폰 사용자가 기하급수적으로 늘어나는 요즘, 그에 따라 모바일 게임 시장 또한, 빠른 성장률을 보이는 것을 알 수 있습니다. 모바일 게임 시장이 확대됨에 따라 게임 시장에는 여러 다양한 게임들이 하루에도 수십 개씩 쏟아져 나오고 있으며, 특히 여러 게임들 중에서 ○○○키우기 게임 등이 인기에 힘입어 다양한 콘텐츠로 제작되고 있습니다.

□ 이러한 키우기 게임에 초점을 맞추어, 요즘 반려동물 시장과 비례하여 점차 증가하는 반려식물 시장을 목표로 하여, 아래 그림과 같이, 기상청에서 제공하는 날씨 데이터를 가지고 현실 환경을 기반으로 한 모바일에서의 식물 재배와 일정관리 및 기상알림 등의 기능을 제공하는 융합 Application을 제작하고자 합니다.



□ 또한, App 제작을 통해, 현실 기반 식물 키우기라는 콘텐츠로 특허를 얻는 것을 시작으로, 직접 게임 시장에 제작된 App을 배포할 계획입니다. 최종적으로는 국내뿐만 아니라 전 세계적으로 여러 사람이 이용할 수 있는 게임 및 일정 관리 App 개발을 목표로 하고 있으며, 이미 대규모의 시장이 형성되어 있는 국제 게임 시장을 Target으로 하여, 높은 수익을 올릴 것으로 예상하고 있습니다.

## 2. 전체 구성도



[그림 2-1. Application 전체 구성도]

□ 위에 그림과 같이, 저희 App은 크게 사용자 관리, 식물 DB 관리 등의 기능을 담당하는 자체 Server와 현실 기반의 기상 데이터를 제공하는 기상청 날씨 정보 DB, 직접적으로 사용자에게 제공되는 android 기반의 Application이 있습니다.

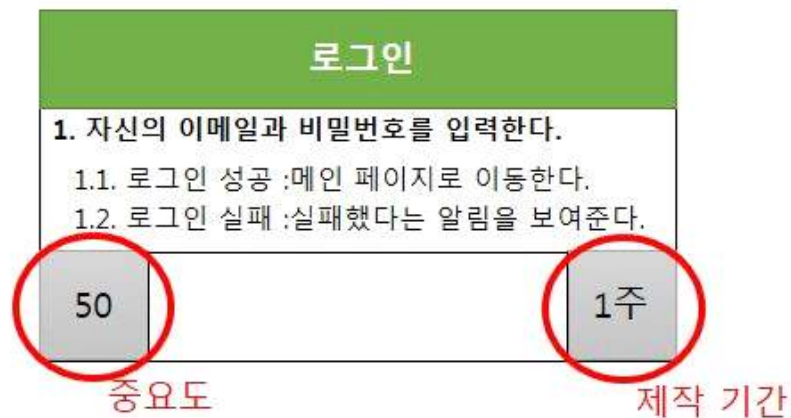
□ 게임 특성상 가장 중요한 UI는 자체적인 기술이 부족한 관계로, 직접 제작이 아닌, UI 전문 업체에게 제작 의뢰를 맡기는 형식으로 진행될 것입니다.

□ 사용자에게 직접적으로 제공되는 Application은 Android 기반으로 제작된 app으로, 로그인/회원가입, 식물 추가, 식물 내부관리, 식물 외부관리 그리고 식물 구입 등의 기능이 제공될 것입니다. Server와의 통신에서는 Volley Library를 사용해 데이터를 주고받을 계획입니다. 또한, 저희 app의 가장 핵심 기능인 app 외부에서의 식물 관리에서는 Overlay Screen 기술을 사용하여 제작할 것으로 정해졌습니다.

□ Server는 Node.js 기반으로 구축될 것이며, DB는 MySQL로 User Table(사용자 관리), Plant Table(식물 Data 관리), Pot Table(화분 Data 관리)로 설계하였습니다.(5. 자료 구조 part에서 자세히 설명)

□ Server는 Window가 아닌 Linux 환경에서 구축될 것입니다.

### 3. User Story 및 Task (요구 사항 분석)



[ 그림 3-1. User Story 예시 ]

#### 3.1 로그인/회원가입

로그인			회원가입		
1. 자신의 이메일과 비밀번호를 입력한다. 1.1. 로그인 성공 :메인 페이지로 이동한다. 1.2. 로그인 실패 :실패했다는 알림을 보여준다.			1. 사용할 이메일과 비밀번호, 닉네임을 입력한다. 1.1. 회원가입 성공 :로그인 페이지로 이동한다. 1.2. 회원가입 실패 :실패했다는 알림을 보여준다.		
50		1주	50		1주

##### □ 로그인

(1) 자신의 id와 password를 입력하고 로그인 button을 클릭한다.

- id 또는 password 둘 중 어느 것 하나라도 입력을 하지 않았거나, 알맞지 않으면 로그인에 실패했다는 알림을 보여준다.
- 로그인에 성공하였다면, App의 메인페이지로 이동한다.

##### □ 회원가입

(1) 로그인 화면에서 회원가입을 하지 않은 상태라면 회원 가입 버튼을 클릭한다.

(2) 회원 가입 페이지로 이동하면 원하는 id, password, nickname을 입력하고, 회원가입 button을 클릭한다.

- id, password 또는 nickname 중 어느 것 하나라도 입력을 하지 않았거나, 이미 존재하는 id / nickname이라면 회원가입에 실패했다는 알림을 보여준다.
- 회원가입에 성공하였다면, 다시 로그인 페이지로 이동한다.

-> 사용자 관리는 내부 기능 및 외부 기능이 완료된 후에 추가해도 되는 부가적인 기능이므로, 중요도는 50이며, 이미 여러번 만들어 보았던 기능이므로 제작기간은 1주일로 두었습니다.

---

□ 로그인 / 회원가입 기능 구현을 위한 Task

(1) Task 1 : User DB 구현

- User에 대한 정보를 Server에서 영구적으로 보관할 수 있도록, Server 측에 User 정보와 관련된 DB를 구축할 필요가 있다.

(User DB에 관련해서는 5. 자료구조 참조)

(2) Task 2 : Server 구현

- 사용자가 입력한 id, password(회원가입 시에는 nickname 정보를 포함)를 Server로 전송하여, User DB에 정보와 비교해야 한다. 만약 사용자가 입력한 id 또는 password가 맞는지 않으면, Server는 Client에게 로그인에 실패했다는 정보를 전송하고, 맞다면 Client에게 로그인에 성공했다는 정보를 전송한다.

(3) Task 3 : 로그인 / 회원가입 UI 구현

User DB 구현			Server 구현		
User의 정보가 들어있는 User Table 구축			사용자가 입력한 id, password, nickname에 따라 DB를 검사하고, 로그인/ 회원가입 성공, 실패 정보를 사용자에게 전송		
50		2틀	50		3일

### 3.2 아이템/식물 구매

아이템/식물 구매		
1. 상점 창으로 이동하는 버튼 클릭 2. 자신이 원하는 아이템/식물 선택 3. 구입 버튼 클릭 3.1. 구입 성공 : 인벤토리에 추가 3.2. 구입 실패 : 실패했다는 알림을 보여준다		
70		1주

- (1) 사용자는 아이템 또는 식물 구매를 원한다면, Application의 메인 화면에서 상점 버튼을 클릭한다.
- (2) 아이템 구입을 원한다면 아이템 페이지로, 식물 구입을 원한다면 식물 페이지로 이동한다.
- (3) 자신이 원하는 아이템 / 식물을 선택한다.
- (4) 최종적으로 선택이 완료되었으면, 구입 버튼을 클릭한다.
  - 구입에 성공하였다면, 사용자의 인벤토리에 아이템 / 식물이 추가된다.
  - 구입에 실패하였다면, 사용자에게 실패했다는 알림을 보여준다.

---

□ 아이템 / 식물 기능 구현을 위한 Task

(1) Task 1 : 식물 / 아이템 DB 구현

- 사용자에게 아이템 / 식물 정보를 보여주기 위해서는 Server에 아이템 / 식물 DB를 구축한 후, 사용자가 상점에 들어올 때, DB의 데이터를 전송하여 사용자에게 어떠한 아이템 / 식물이 있는지 보여준다.

(2) Task 2 : User DB와 Task 1의 DB를 join하여 새로운 Table 생성

- 사용자가 아이템을 구입하였을 시, 사용자가 어떠한 아이템 / 식물을 가지고 있는지에 대한 정보를 저장할 필요가 있으므로, 이를 위해, User Table의 id와 아이템 / 식물 Table의 id를 외래키로 한 새로운 테이블을 생성
- 사용자의 인벤토리를 보여줄 때 이 Table을 사용하여, 사용자가 가지고 있는 아이템 / 식물을 보여준다.

(3) Task 3 : Server 구현

- 사용자가 아이템 / 식물 구입 버튼을 클릭하였을시, user 정보와 아이템 / 식물 정보를 받아와 이를 Task 2에서 구축한 Table에 저장 또는 개수 등의 상태를 Update 한다.

(4) Task 4 : 상점 화면 구현

(5) Task 5 : 아이템 / 식물 이미지 제작

- Task 4와 Task 5는 자체적으로 제작하기에는 기술이 부족하므로, UI 전문 제작 업체에게 의뢰

### 3.3 식물 관리하기

식물 관리하기(app)		
1. 관리하고 싶은 식물 선택		
2. 관리창에서 원하는 버튼을 클릭하여 식물 관리		
90		2주

- 세로 버튼 : 식물의 EXP 증가 시켜준다.
- 가로 버튼 : 식물의 HP를 관리 해준다.
- On/Off 버튼 : 식물을 홈 화면에 등록 할 것인지를 나타내는 버튼.

- (1) 사용자는 내부에서 관리하고 싶은 식물을 클릭하여, 내부 관리 창으로 이동한다.
- (2) 사용자는 아이템 또는 관리 기능 버튼을 클릭하여 식물의 HP 및 EXP를 관리한다.

□ 식물 내부 관리 기능 구현을 위한 Task

(1) Task 1 : User가 가진 식물 / 아이템 DB 구현

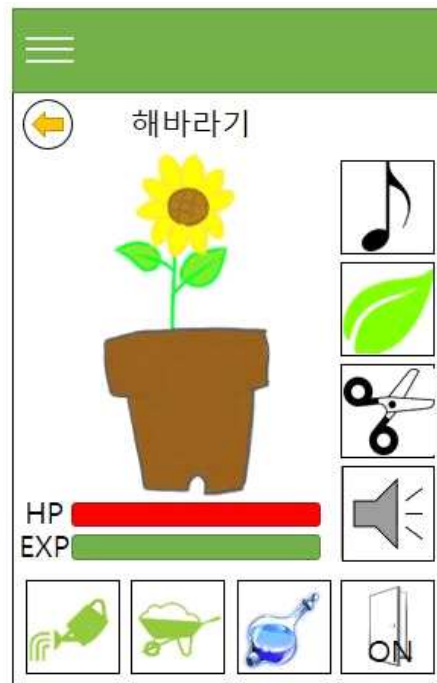
- 메인 화면에서 User가 가지고 있는 식물 DB를 사용하여, 사용자에게 UI로 보여준다.
- 사용자는 메인화면에 나타난 식물들 중에서 원하는 식물을 선택할 수 있다.

(2) Task 2 : Server 구현

- 사용자가 아이템 또는 관리 기능 버튼을 클릭하여 식물의 HP 및 EXP를 관리한다.
  - 이 때, 사용자가 가지고 있는 식물에 대한 DB에서 식물의 정보를 Update 해줘야 한다.
- ex) 사용자가 식물에게 물을 주었을 때, 해당 식물의 DB에 식물에게 최근의 물을 준 시간 등을 Update 시켜주어야 한다.

(3) Task 3 : 식물 관리 화면 구현

- 이 화면 또한, 자체적인 제작 능력 부족으로, 전문 업체에게 의뢰



### 3.4 스마트폰 홈 화면에 식물 등록

홈 화면에 등록		
<b>1. 홈 화면에 등록</b> 1.1. app 내에서 on/off 버튼 클릭 1.2. 식물을 톱 클릭하여 해당 식물을 홈 화면에 등록		
100		1일

- (1) 사용자는 메인화면에서 외부 관리 기능을 수행하기 원하는 식물을 길게 클릭하거나, 식물 내부 관리 창에서 외부 관리 on / off 기능에서 on 상태로 버튼을 클릭한다.

(2) 사용자는 외부로 나온 view를 움직여 자신이 원하는 위치에 setting 시킨다.

□ 스마트 폰 홈 화면에서의 식물 등록 및 외부 관리 기능 구현을 위한 Task

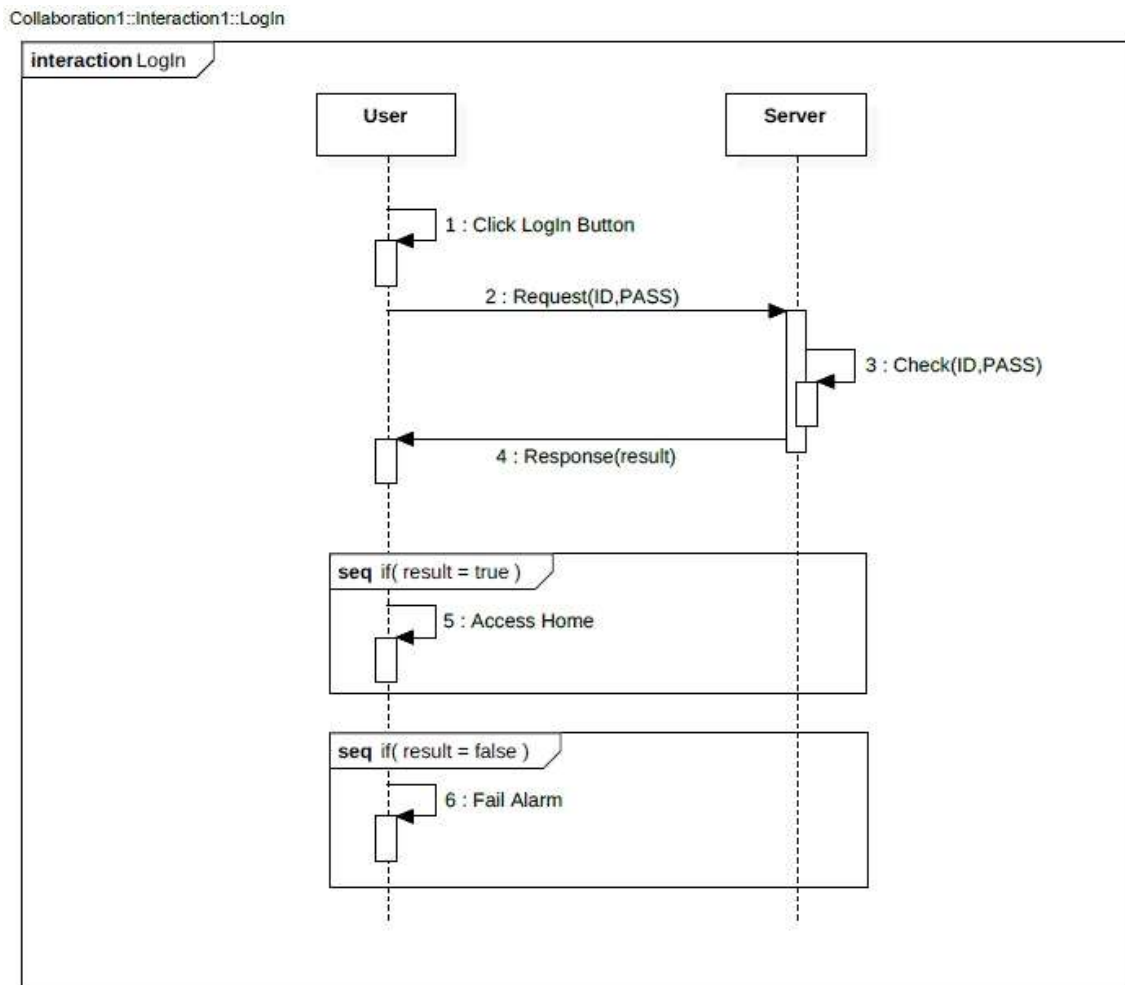
(1) Task 1 : Overlay Screen App 코드 구현

- 사용자가 식물을 클릭하였을 시, 식물이 사용자의 휴대폰 홈 화면에 옮기고 사용할 수 있도록 overlay Screen 기술을 사용하여, 식물 이미지를 overlay window에 Setting 하는 기능 구현
- 외부에서도 식물을 관리할 수 있도록, 외부로 나와 있는 식물을 클릭할 시, 휴대폰 화면상에서의 식물의 위치 정보를 받아와, 식물의 바로 오른쪽 윗부분에 식물을 관리할 수 있는 버튼들이 뜰 수 있도록 한다.

## 4. Sequence Diagram

### 4.1 로그인/회원가입

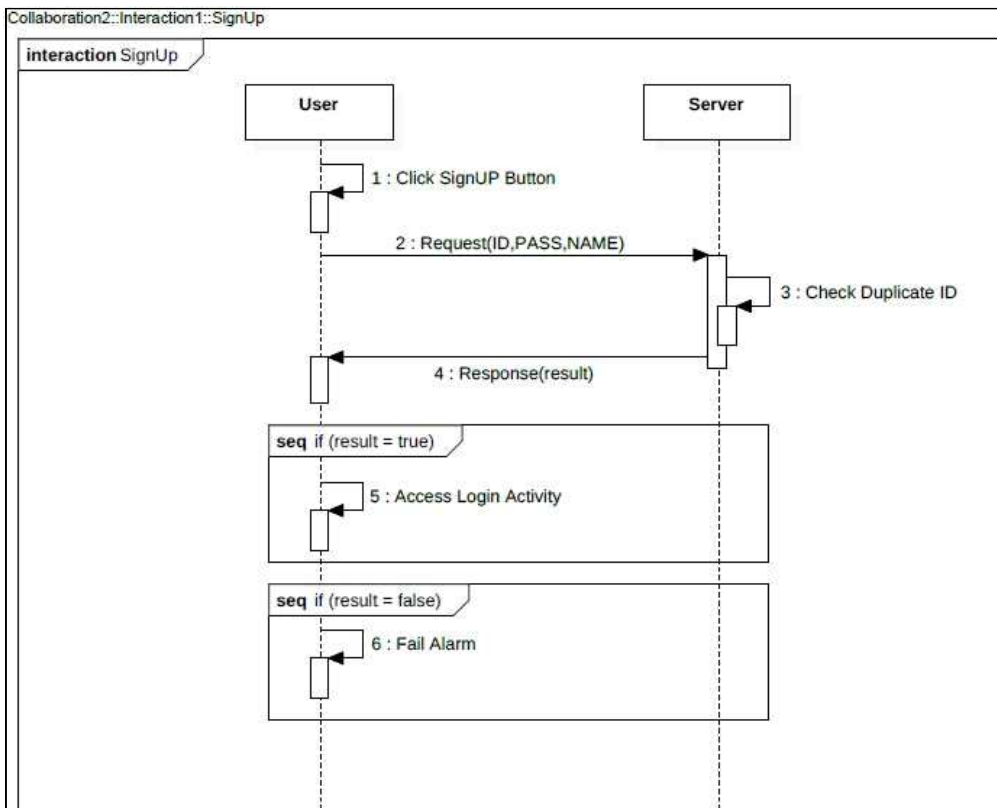
□ 로그인





- (1) 사용자는 자신의 ID와 Password를 입력하고 로그인 버튼을 누른다
- (2) ID, pass를 서버에 전송한다
  - > { id : id, pass : password }
- (3) Server는 사용자로부터 전송된 id와 pass가 존재하는지 또는 id와 password가 일치하는지 검사한다
  - > User Table을 사용하여 검사
- (4) Server는 검사 결과에 따라 result값을 Client에게 전송한다.
  - > result : String [“true” or “false”]
  - > client 부분에서도 String으로 받음.(StringRequest)
- (5) result값이 true이면 Application의 홈 화면으로 이동한다.
- (6) result값이 false이면 로그인에 실패했다는 알림을 화면에 띄어 준다.

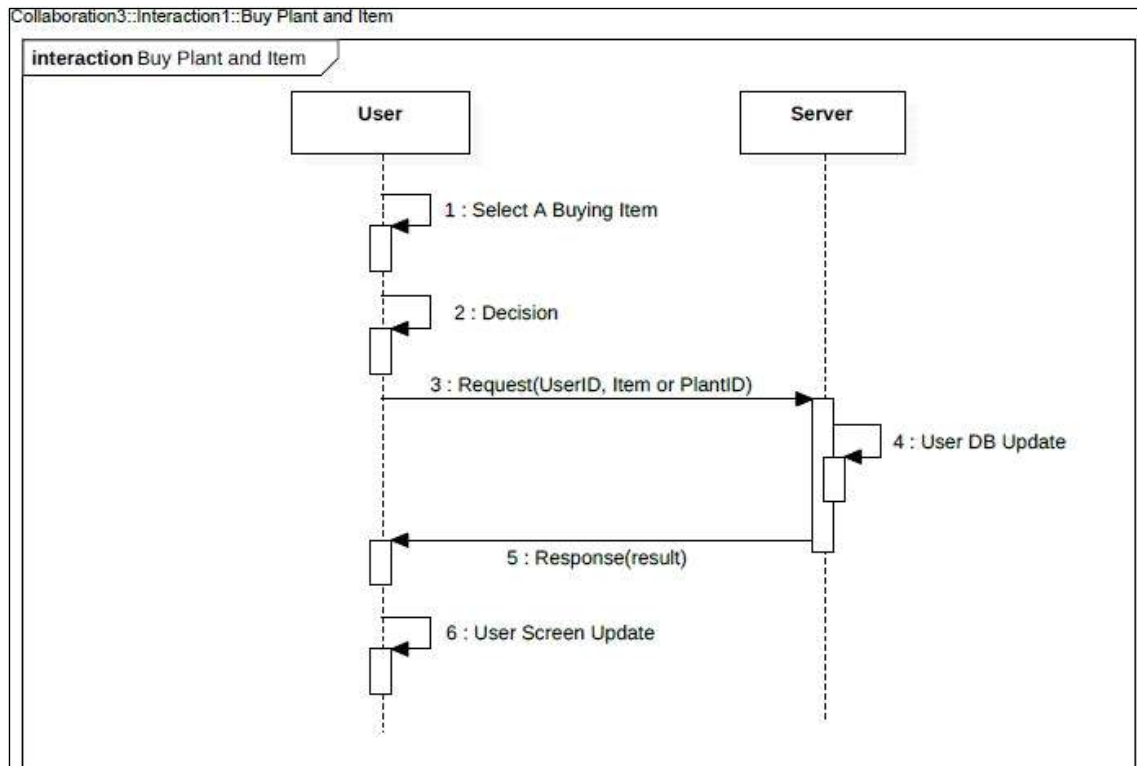
## (2) 회원가입



- (1) 사용자는 자신이 원하는 id, password, nickname을 작성하여 회원가입 버튼을 누른다.
- (2) Server로 id, password, nickname을 서버에 전송한다.
  - > {id : id, pass : password, nickname : nickname}
- (3) Server는 전송된 id, name이 이미 존재하는지 중복검사를 한다.
- (4) Server는 검사 결과에 따라 result값을 Client에게 전송한다.
  - > result : String [“true” or “false”]
  - > client 부분에서도 String으로 받음(StringRequest)
- (5) result 값이 true면 회원가입 성공 -> 로그인 화면으로 이동한다.

(6) result 값이 false면 회원가입에 실패했다는 알림을 화면에 띄어 준다.

## 4.2 아이템/식물 구매



(1) 사용자는 구매할 아이템 혹은 식물을 선택한다.

(2) 사용자는 선택이 완료되었다면 구매 결정하고 구매 버튼을 클릭한다.

(3) 구매를 하는 userID와 구매하는 식물 혹은 아이템의 ID를 Server로 전송한다

-> {userID : userID, item/plantID : item/plant}.

(4) Server는 UserDB에

- 구입한 물품이 식물이면 User가 가진 식물 정보에 해당 식물을 추가한다.

- 구입한 물품이 아이템이면 user가 가진 아이템 정보에 해당 아이템을 추가한다.

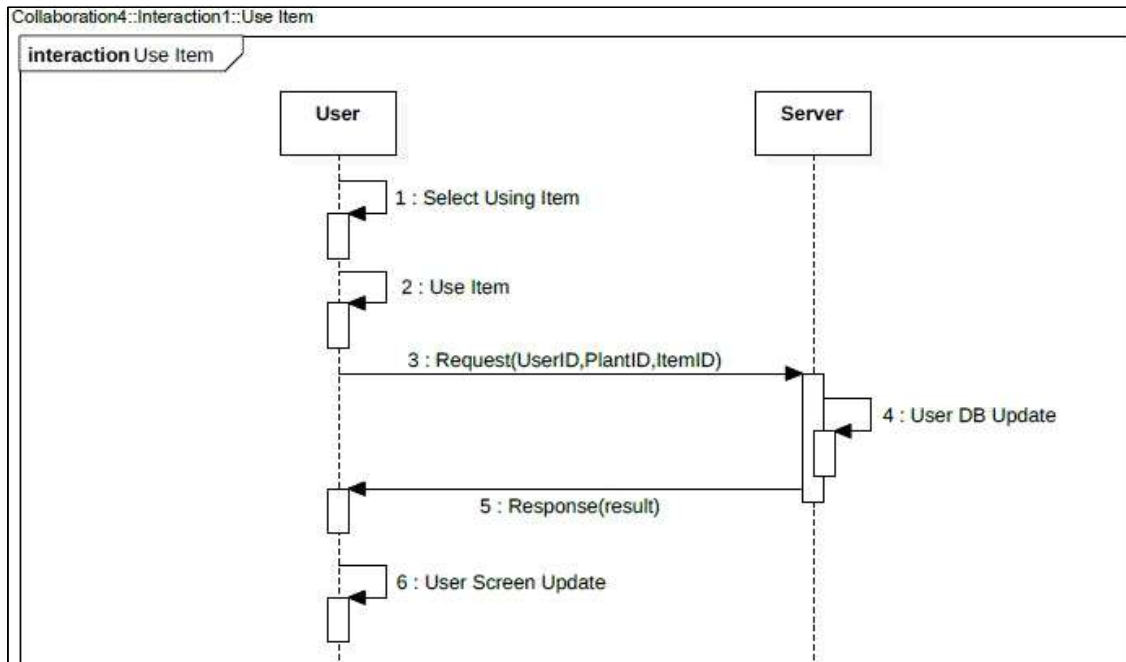
(5) Server는 DB에 정보를 Update하는 작업이 완료되면 Client에게 결과값(result)를 전송한다.

- result : String [ "true" or "false" ]

(6) result값이 true면 Application의 인벤토리 화면을 갱신한다.

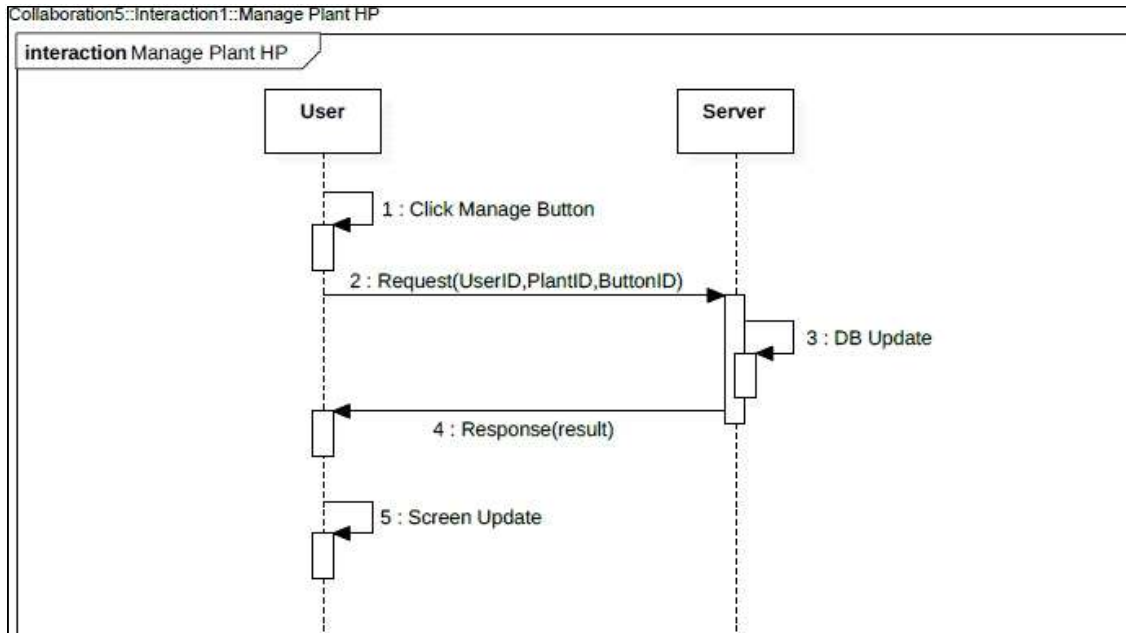
## 4.3 식물 관리 하기

(1) 아이템을 사용하여 식물 EXP 증가



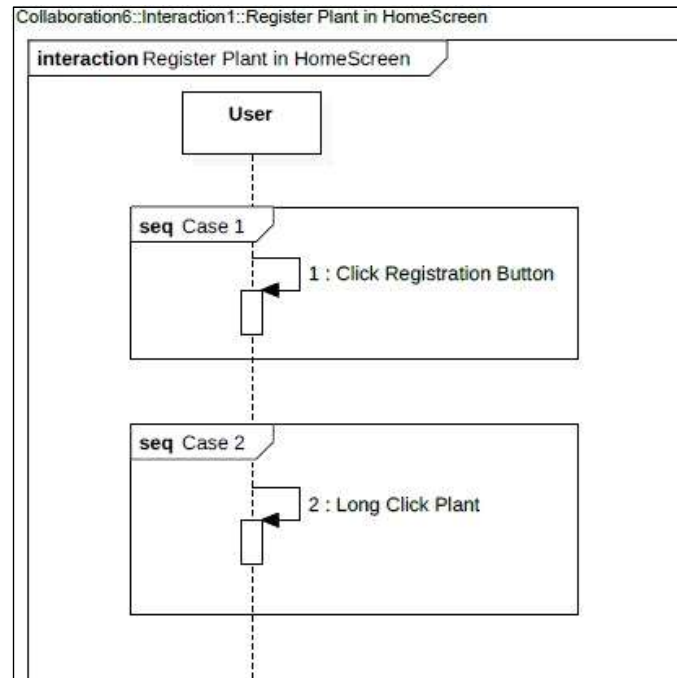
- (1) 사용자는 App의 식물 내부 관리 창에서 사용할 아이템을 선택한다.
- (2) App의 화면에 아이템을 사용하는 효과를 보여준다.
- (3) 사용된 아이템과 아이템이 사용된 식물, user의 ID를 Server로 전송한다
  - {userID : userID , itemID : item, plantID : plant }
- (4) Server는 User DB에서 해당 user가 사용한 아이템만큼 해당하는 아이템의 개수를 차감한다.
- (5) Server는 해당 아이템이 사용된 식물의 EXP를 아이템의 효과에 따라 증가시킨다.
  - item DB에 존재하는 정보를 이용
- (5) Server에서 아이템에 따른 식물 정보 Update 작업이 완료 되면 Client로 result 값을 전송한다.
  - result : String ["true" or "false"]
- (6) result 값이 true이면 Application 내부 관리 화면에서 해당 식물의 EXP Bar를 갱신해준다.

## (2) 식물 HP 관리



- (1) 사용자는 식물 내부 관리 창에서 적용하고 싶은 관리 버튼을 클릭한다
- (2) user와 관리를 받을 식물, 관리 버튼의 ID를 서버로 전송한다.
  - { userID : userID, manageID : mID , plantID : plant }
- (3) Server는 User DB에서 해당 user가 누른 관리 버튼의 효과에 따라 사용자가 관리하기 원하는 식물 (plantID)의 이 HP를 증가시킨다.
  - 관리기능과 관련된 DB 사용
- (4) Server에서 관리 버튼의 따른 식물의 HP 정보 Update 작업이 완료 되면 result 값을 Client에게 전송한다.
- (5) result 값이 true면 Application 내부관리 기능에서 HP Bar의 상태를 갱신한다.

## 4.4 스마트폰 홈 화면에 식물 등록



1. 사용자는 식물 내부 관리창에 있는 ON / OFF 버튼을 클릭한다.
  - OFF 상태로 변경할 경우 : 식물을 App 외부에서 내부로 가지고 온다.
  - ON 상태로 변경할 경우 : 식물을 App 내부에서 외부로 가지고 간다.
    - > Overlay Screen에 해당하는 view를 추가한다.
2. 사용자는 App의 메인 화면(식물이 나열되 있는 화면)에서 외부로 가지고 가길 원하는 식물을 길게 클릭한다.
  - 해당 식물을 외부 관리 창에서 원하는 위치에 이동시킨다.

---

## 5. 자료 구조 설계

### (1) DB 설계

#### ☐ User table

UserNo	User E-mail	User PW	Nickname	Gold	Crystal

- UserNo int NOT NULL AUTO\_INCREMENT PRIMARY\_KEY
  - 유저에게 부여 된 고유 번호(유저 관리 등에 사용)
- Email varchar(20) NOT NULL
  - 사용자가 지정한 비밀번호
- Pw varchar(50) NOT NULL
  - 사용자가 지정한 비밀번호(후에 암호화 수행)
- Nickname varchar(20) NOT NULL
  - 사용자가 지정한 닉네임
- Gold int
  - 사용자가 가지고 있는 게임 내부의 재화
- Crystal int
  - 사용자가 가지고 있는 현금 재화

#### ☐ Plant table

PlantNo	PlantName	explain	WaterTime	GrowTime	imagePath

- PlantNo int NOT NULL AUTO\_INCREMENT PRIMARY\_KEY
  - 식물이 가지는 unique 번호 ( primary key)
- PlantName varchar(20) NOT NULL
  - 식물의 이름
- Explain varchar(200) NOT NULL
  - 식물에 대한 간단한 설명
- WaterTime int NOT NULL
  - 식물에게 물을 줘야하는 최대 시간
- GrowTime int NOT NULL
  - 식물이 모두 자라는데 걸리는 시간
- ImagePath varchar(50) NOT NULL
  - 식물 이미지 경로 (성장 단계 별로 지정)

#### ☐ Pot table

PotNo	HPeffect	EXPeffect	imagePath

- PotNo int NOT NULL AUTO\_INCREMENT PRIMARY\_KEY  
- 화분이 가지는 unique ID(primary key)
- HPeffect int NOT NULL  
- 화분이 HP에 주는 효과
- EXPeffect int NOT NULL  
- 화분이 EXP에 주는 효과
- ImagePath varchar(20) NOT NULL  
- 화분 이미지 경로

#### ☐ Join table

User No	Plant No	PotNo	Plant Name	Last Time Limit	Growing Time	HP	EXP	LEVEL

- UserNo : User Table의 primary key (Foreign key)
- PlantNo : Plant Table의 primary key(Foreign key)
- PotNo : Pot Table의 primary key(Foreign key)  
- >위 3개 정보로 유저가 키우는 식물마다 DB에 따로 저장
- PlantName varchar(20) default "Plant Table의 Name"  
- 사용자가 지정한 식물의 이름(사용자가 지정하지 않을 시 default 이름으로 삽입)
- LastWaterTime : 마지막으로 물을 준 시간
- GrowingTime : 식물을 기른 시간
- HP : 해당 식물의 체력
- EXP : 해당 식물의 경험치
- LEVEL : 식물의 성장 단계

## 6. 결론 및 자체 평가

### - 설계 과정 요약

- App 내부 기능은 설계대로 구현 했으며, server&DB는 아직 구현하지 않았기 때문에 구현 전에 설계를 더 자세하게 하여 구현 할 예정이다.

(설계 과정과 결과에 대하여 어떻게 느끼고 생각하는지 적고 수행 수준을 스스로 평가하여 기록.)

- 
- 설계를 자세하게 작성하지 않아서 힘들게 구현 하다가 결국 다시 설계를 하는 과정이 발생했다. 팀원들과 많은 회의를 거쳐 다시 설계를 하여 해결하였고, 기초설계의 중요성을 느꼈다.

(과제를 설계하면서 어려웠던 점과 그 문제를 어떻게 해결했는지 적을 것)