

# 데이터베이스 설계 프로젝트

## 보고서

- Creature Island -



|       |     |               |          |                       |
|-------|-----|---------------|----------|-----------------------|
| 과목명   |     | 데이터베이스 (002반) |          |                       |
| 담당교수님 |     | 노승민 교수님       |          |                       |
| 제출일   |     | 2019년 06월 10일 |          |                       |
|       | 이름  | 학번            | 학과       | 연락처                   |
| 팀장    | 김규리 | 16011212      | 디지털콘텐츠학과 | alley5653@naver.com   |
|       |     |               |          | 010-9257-5643         |
| 팀원    | 최승환 | 17013241      | 소프트웨어학과  | tmdghks1227@naver.com |
|       |     |               |          | 010-4999-0596         |

# 목 차

|                         |    |
|-------------------------|----|
| 1. 프로젝트 개요 .....        | 2  |
| 2. 시스템 구성도 .....        | 4  |
| 3. 요구사항 분석 명세서 .....    | 5  |
| 4. 논리적 데이터 모델 .....     | 18 |
| 5. 물리적 데이터 모델 .....     | 19 |
| 6. 용어 사전 .....          | 20 |
| 7. 도메인 기술서 .....        | 21 |
| 8. 테이블 기술서 .....        | 23 |
| 9. 정규화 과정 .....         | 28 |
| 10. 현존 오류 .....         | 32 |
| 11. 기여도 .....           | 33 |
| 12. 데이터베이스 Dump 자료..... | 별첨 |
| 13. 프로젝트 소스 파일.....     | 별첨 |

## 1. 프로젝트 개요

### 프로젝트 개요

#### 1. 프로젝트 주제

웹을 통해 간편히 즐길 수 있는 가상 펫 수집 게임 서비스

#### 2. 프로젝트 설명

기술의 발전에 따라 무선인터넷과 핸드폰을 필두로 하는 모바일 환경이 발전하고, 이에 따른 웹 접속 플랫폼이 다양해져 언제 어디서든 편리하게 웹 게임을 즐길 수 있을 것으로 보인다. 이러한 웹 게임의 경우 모바일 등의 환경으로 틈틈이 즐길 수 있다는 장점이 있고, 난이도가 높지 않으며 가벼운 편이라 연령대에 관계없이 쉽고 자유롭게 즐길 수 있을 것으로 추정된다. 또, 별도의 클라이언트 설치 없이 웹 브라우저 만으로도 플레이가 가능하며 컴퓨터 사양의 제약으로부터 자유롭다. 최근 현대인의 웹 이용 빈도가 높은 점과 연결한다면 접근성이 매우 우수한 게임을 개발할 수 있을 것이라 사료된다. 해당 프로젝트에서는 수집(hoarding), 힐링에 목표를 둔 반 방치형 펫 게임이 바쁜 현대인에게 매력적일 것이라 판단, 러프한 단계의 웹 펫 게임 개발을 목표로 한다.

#### 3. 주요 기능

- 회원 가입, 계정 소유
- 탐험 기능을 통해 랜덤 크리처 수집
- 게임 내 재화로 아이템을 구매
- 수집 기능을 통해 랜덤 아이템 수집

- 아이템을 재화로 전환

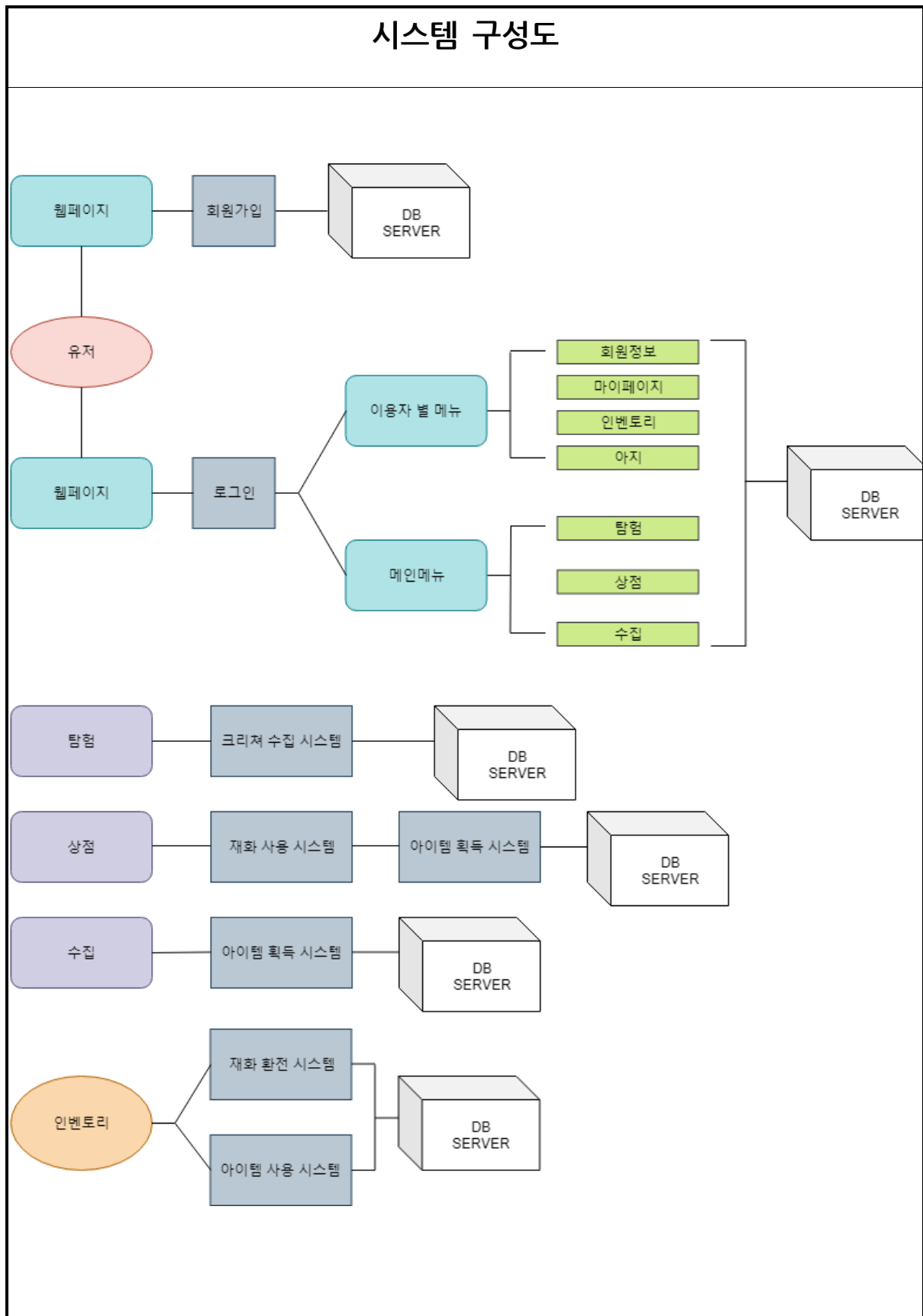
#### 4. 개발 언어 및 환경

서버: Apache2.4

DB: MySQL

언어: PHP, HTML, JSP

## 2. 시스템 구성도



### 3. 요구사항 분석 명세서

| 요구사항 분석   |      |
|---|------|
| 요구사항 명  | 회원가입 |
| <ol style="list-style-type: none"><li>1. 회원 아이디, 이름, 생년월일, 이메일, 비밀번호를 입력 받아 정보를 데이터베이스에 저장한다.</li><li>2. 빠진 정보가 있다면 가입하기 버튼을 눌러도 정보가 저장되지 않고, 경고 메시지를 띄운다.</li><li>3. 비밀번호는 암호화하여 저장한다.</li></ol> |      |

## 요구사항 분석

| 요구사항 명  | 로그인 |
|---|-----|
| <ol style="list-style-type: none"> <li>아이디와 비밀번호를 받아 가지고 있는 데이터와 대조하여 보유한 아이디일 경우 로그인을 수행한다.</li> <li>아이디가 입력되지 않은 채로 로그인 버튼이 클릭된다면 아이디 입력을 수행해달라는 경고 메시지를 띄운다.</li> <li>입력된 아이디가 데이터베이스에 등록되어 있지 않다면 존재하지 않는 아이디임을 경고 메시지를 띄워 알린다.</li> <li>비밀번호가 데이터베이스에 등록되어 있지 않다면 비밀번호가 잘못되었음을 경고 메시지를 띄워 알린다.</li> <li>로그인 정보는 24시간 동안 유지된다.</li> </ol> |     |

## 요구사항 분석

| 요구사항 명   | 회원정보 열람 |
|--|---------|
| <p>1. 회원정보 데이터 중에서 로그인 된 회원의 회원 아이디, 이름, 생년월일, 이메일 정보를 출력한다.</p> <p>2. 비밀번호는 공개하지 않는다.</p> <p>3. 로그인 상태가 아닐 경우 회원정보는 열람할 수 없다.</p> |         |



## 요구사항 분석

요구사항 명

회원 탈퇴

1. 회원 탈퇴 버튼을 누를 시, 데이터베이스에서 회원 정보를 삭제함과 동시에 로그아웃 되도록 한다.
2. 회원 탈퇴 시 가지고 있는 모든 크리쳐의 정보, 아이템의 정보, 크리쳐 획득 기록의 정보를 함께 삭제하도록 한다.

## 요구사항 분석

| 요구사항 명   | 로그아웃 |
|--|------|
| <ul style="list-style-type: none"><li>1. 로그아웃 한다.</li><li>2. 회원정보는 삭제되지 않는다.</li><li>3. 회원이 갖고 있는 크리쳐, 히스토리, 아이템 정보는 삭제되지 않는다.</li></ul> |      |

## 요구사항 분석

요구사항 명

획득 크리쳐 열람 (아지트)

1. 메인 메뉴에서 아지트 버튼을 누르면 획득한 크리쳐를 열람할 수 있다.
2. 데이터베이스에서 해당 계정이 소유한 크리쳐 이미지를 불러와 순차적으로 출력한다.
3. 크리쳐 이미지를 클릭하면 크리쳐의 자세한 정보를 열람 가능하다. 자세한 정보에는 크리쳐의 이름, 속성, 서식지, 몸무게, 신장, 설명이 포함된다.
4. 로그인 상태에서만 해당 메뉴에 접근할 수 있다.
5. 아지트에서 크리쳐 획득 기록으로 접근하는 링크를 접근 가능하여야 하며, 상성이 좋은 크리쳐를 검색할 수 있어야 한다.

## 요구사항 분석

요구사항 명

크리쳐 획득 기록 열람 (히스토리)

1. 크리쳐의 순번과 함께, 크리쳐 이미지, 이름, 무게, 신장, 상성이 좋은 크리쳐, 획득 시각을 함께 표시하여 나열한다.
2. 획득 시각은 2019-06-10 00:18:11 의 형식으로 표시한다.
3. 로그인 상태에서만 접근할 수 있다.

## 요구사항 분석

요구사항 명

상성이 좋은 크리쳐 목록 검색

1. 아지트에서 상성이 좋은 크리쳐를 검색할 수 있어야한다.
2. 데이터베이스에서 검색된 검색 결과는 순번으로 나열된다.
3. 크리쳐의 이미지, 이름, 상성이 좋은 크리쳐가 표시된다.

## 요구사항 분석

요구사항 명

획득 아이템 열람 (인벤토리)

1. 메인 메뉴에서 인벤토리 페이지로 접속할 수 있다.
2. 로그인 된 상태에서만 접근이 가능하다.
3. 상단에 현재 소지한 금액을 회원정보로부터 불러와 출력한다.
4. 소지 금액 밑으로 소유한 아이템 정보들이 나열된다.
5. 아이템 정보에는 아이템 이미지, 아이템 이름, 소유 개수, 아이템 설명, 판매가격이 있다.
6. 아이템 정보 밑에는 아이템 판매 버튼과 함께, 판매할 아이템 개수를 입력할 수 있는 란이 있다.

## 요구사항 분석

### 요구사항 명

### 아이템 판매 (인벤토리)

1. 인벤토리 페이지에서 아이템 정보 란 밑에 있는 란에 아이템 개수를 입력 후 판매 버튼을 눌러 아이템을 판매할 수 있다.
2. 아이템 판매 시, 소지 금액은 아이템 가격 x 판매한 개수만큼 늘어난다.
3. 아이템 판매 시, 소지한 해당 아이템의 개수는 판매한 개수만큼 줄어든다.
4. 판매 한 종류의 아이템의 소지 개수가 0개가 된다면 아이템은 인벤토리에서 삭제된다.
5. 소유한 개수보다 많은 수가 판매 개수로 입력되었을 시, 유효하지 않은 값이라는 경고 메시지를 띄운다.

## 요구사항 분석

### 요구사항 명

### 랜덤 크리쳐 획득 (탐험)

1. 메인 메뉴의 '탐험' 을 눌러 탐험 페이지로 접속이 가능하다.
2. 로그인 상태에서만 해당 페이지에 접근이 가능하다.
3. 세가지의 이미지가 탐험 페이지에서 주어진다. 각각 숲, 물가, 동굴의 이미지이다.
4. 각 이미지를 클릭하였을 경우, 숲, 물가, 동굴의 세부 페이지로 연결되며, 해당 페이지에서, 해당 장소에서 얻을 수 있는 크리쳐를 랜덤으로 출력한다.
5. 탐험 세부 페이지에서는 크리쳐의 이미지, 이름, 타입, 무게, 신장을 출력한다.
6. 크리쳐의 무게와 신장은 데이터베이스에 등록된 최소값과 최대값 사이에서 랜덤하게 정해진다.
7. 출력된 크리쳐는 소유한 크리쳐 데이터에 등록되며 아지트에서 확인할 수 있다.



## 요구사항 분석

요구사항 명

랜덤 아이템 획득 (수집)

1. 메인 메뉴의 '수집'을 눌러 수집 페이지로 접속이 가능하다.
2. 로그인 상태에서만 해당 페이지에 접근이 가능하다.
3. 세가지의 이미지가 탐험 페이지에서 주어진다. 각각 흙, 강, 하늘의 이미지이다.
4. 각 이미지를 클릭하였을 경우, 흙, 강, 하늘의 세부 페이지로 연결되며, 해당 페이지에서 해당 장소에서 얻을 수 있는 아이템을 랜덤으로 출력한다.
5. 수집 세부 페이지에서는 아이템의 이름을 출력한다.
6. 아이템은 최소 한 개에서 최대 다섯 개까지 출력되며, 각각 다른 것일 수도 있고, 중복된 것일 수도 있다.
7. 출력된 아이템은 개수만큼 소유한 아이템 데이터에 등록되며 인벤토리에서 확인할 수 있다.

## 요구사항 분석

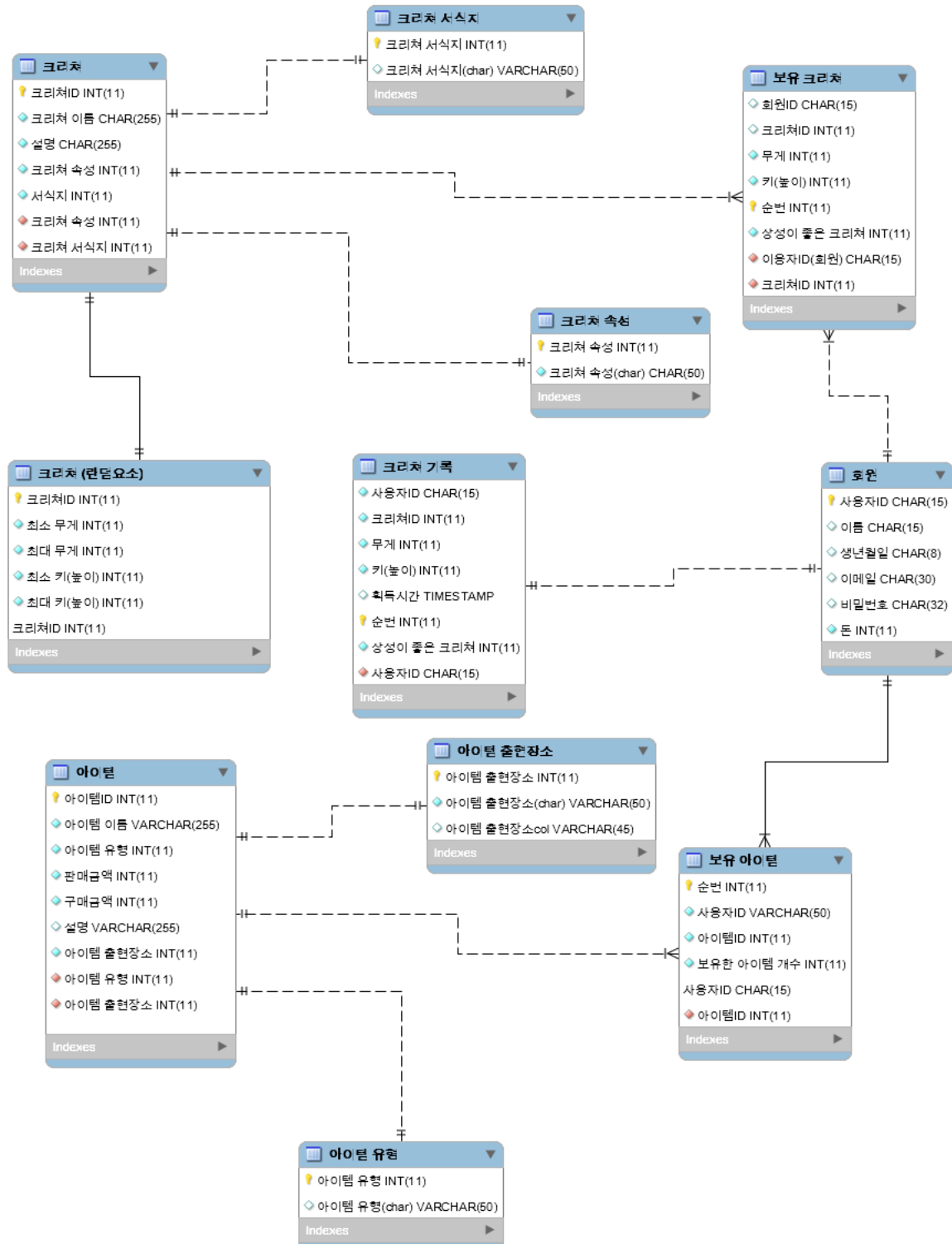
### 요구사항 명

### 아이템 구매 (상점)

1. 상점 페이지에는 메인 메뉴의 링크에서 접속할 수 있다.
2. 로그인 상태에서만 해당 기능에 접근이 가능하다.
3. 상점 페이지 상단에 현재 소지 금액이 표기된다.
4. 소지 금액 하단에는 상점에서만 판매하는 아이템의 정보가 나열되어 출력된다.
5. 아이템의 정보에는 아이템 이미지와 아이템 이름, 아이템 설명이 있다.
6. 아이템 정보 하단에는 구매 버튼과 구매 개수를 입력할 수 있는 란이 존재한다.
7. 값을 입력하지 않고 구매 버튼을 누르면 값을 입력해달라는 경고 메시지가 뜨게 된다.
8. 소지 금액 보다 아이템 구매가의 합이 높을 경우 잔액이 부족하다는 경고 메시지가 뜨게 된다.
9. 아이템 구매 버튼을 누를 경우, 소유한 아이템 데이터베이스에 해당 아이템이 추가된다. 이는 인벤토리에서 확인할 수 있다.

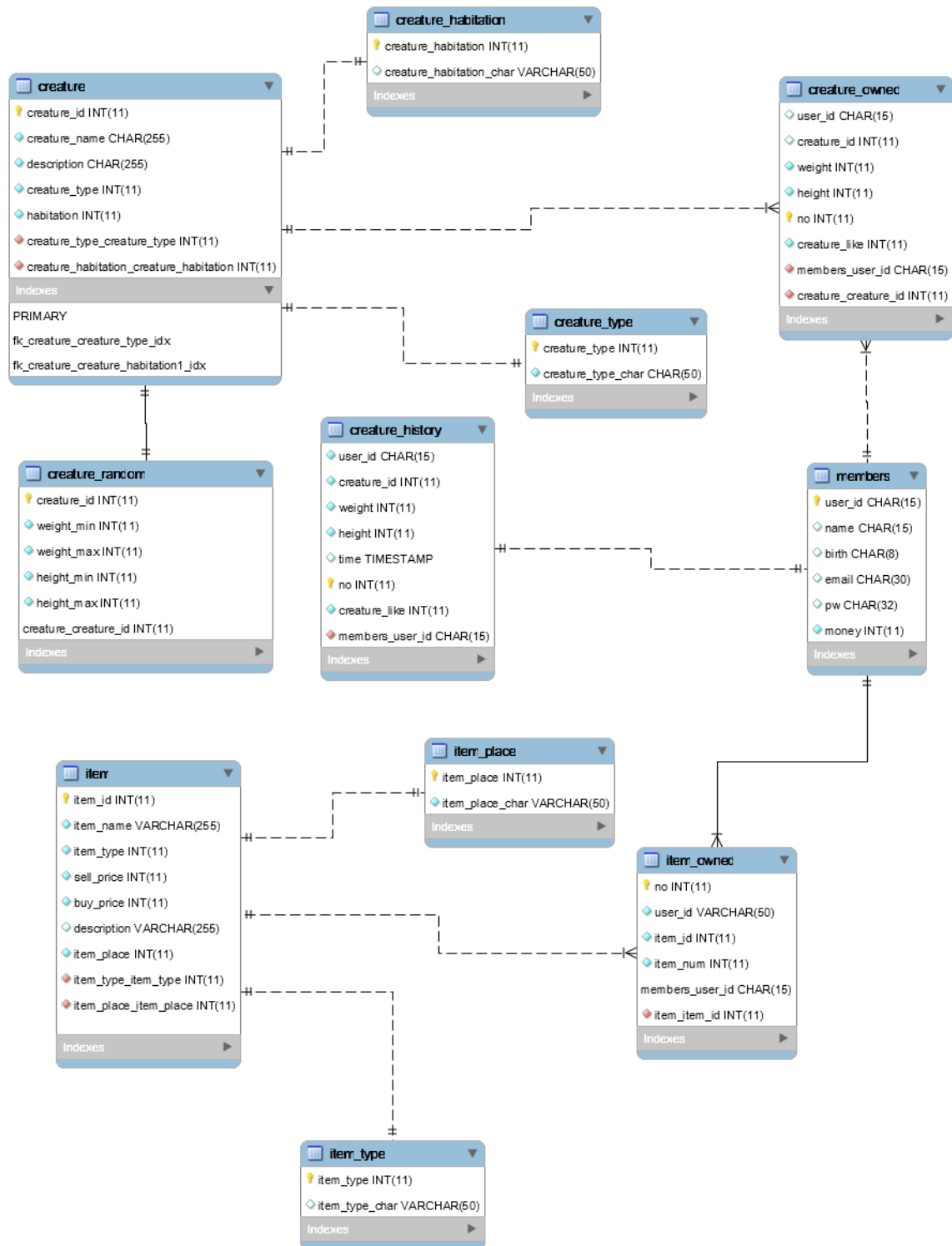
## 4. 논리적 데이터 모델(ERD)

### 논리적 데이터 모델(ERD)



## 5. 물리적 데이터 모델(ERD)

### 물리적 데이터 모델(ERD)



## 6. 용어사전

| 용어사전       |                     |                                |
|------------|---------------------|--------------------------------|
| 논리명        | 물리명                 | 설명                             |
| 크리쳐ID      | creature_id         | 크리쳐가 가지는 고유한 숫자                |
| 크리쳐 이름     | creature_name       | 크리쳐의 이름                        |
| 설명         | description         | 아이템 / 크리쳐에 대한 설명               |
| 크리쳐 서식지    | creature_habitation | 크리쳐가 위치하는 곳                    |
| 유저ID       | user_id             | 게임상 회원이 쓰는 별명                  |
| 무게         | weight              | 크리쳐의 무게                        |
| 키(높이)      | height              | 크리쳐의 키(높이)                     |
| 순번         | no                  | 데이터가 저장된 순번, primary key용으로 사용 |
| 상성이 좋은 크리쳐 | creature_like       | 개별 크리쳐와 상성이 좋은 크리쳐, 랜덤으로 지정    |
| 크리쳐 속성     | creature_type       | 크리쳐가 가지는 속성 (열...etc)          |
| 획득시간       | time                | 크리쳐를 획득한 시간                    |
| 이름         | name                | 회원가입시 입력하는 실명                  |
| 생년월일       | birth               | 회원가입시 입력하는 생년월일                |
| 이메일        | email               | 회원가입시 입력하는 이메일                 |
| 비밀번호       | pw                  | 회원가입시 입력하는 비밀번호                |
| 아이템ID      | item_id             | 아이템이 가지는 번호                    |
| 아이템 이름     | item_name           | 아이템의 이름                        |
| 아이템 유형     | item_type           | 아이템이 가지는 유형                    |
| 판매가격       | sell_price          | 아이템을 판매시 획득하는 가격               |
| 구매가격       | buy_price           | 아이템을 구매시 소모하는 가격               |
| 아이템 출현장소   | item_place          | 아이템이 출현하는 위치                   |

## 7. 도메인 기술서

| 도메인 기술서       |                          |             |                 |
|---------------|--------------------------|-------------|-----------------|
| 논리명           | 물리명                      | Data type   | 설명              |
| 크리쳐ID         | creature_id              | INT(11)     | 크리쳐의 아이디        |
| 크리쳐 이름        | creature_name            | CHAR(255)   | 크리쳐의 이름         |
| 설명            | description              | CHAR(255)   | 크리쳐에 대한 설명      |
| 크리쳐 서식지       | creature_habitation      | INT(11)     | 크리쳐의 서식지 아이디    |
| 크리쳐 서식지(char) | creature_habitation_char | VARCHAR(50) | 크리쳐 서식지 이름 문자열  |
| 유저ID          | user_id                  | CHAR(15)    | 유저의 아이디         |
| 무게            | weight                   | INT(11)     | 크리쳐의 무게         |
| 키(높이)         | height                   | INT(11)     | 크리쳐의 신장         |
| 순번            | no                       | INT(11)     | 데이터가 저장된 순번     |
| 상성이 좋은 크리쳐    | creature_like            | INT(11)     | 상성이 좋은 크리쳐의 아이디 |
| 크리쳐 속성        | creature_type            | INT(11)     | 크리쳐의 속성 아이디     |
| 크리쳐 속성(char)  | creature_type_char       | CHAR(50)    | 크리쳐의 속성 문자열     |
| 최소 무게         | weight_min               | INT(11)     | 종류별 크리쳐의 최소 무게  |
| 최대 무게         | weight_max               | INT(11)     | 크리쳐의 최대 무게      |
| 최소 키(높이)      | height_min               | INT(11)     | 크리쳐의 최소 신장      |
| 최대 키(높이)      | height_max               | INT(11)     | 크리쳐의 최대 신장      |
| 획득시간          | time                     | TIMESTAMP   | 크리쳐를 획득한 시간     |
| 이름            | name                     | CHAR(15)    | 유저 이름           |
| 생년월일          | birth                    | CHAR(8)     | 유저 생년 월일        |
| 이메일           | email                    | CHAR(30)    | 유저 이메일          |
| 비밀번호          | pw                       | CHAR(32)    | 유저 비밀번호         |
| 아이템ID         | item_id                  | INT(11)     | 아이템의 아이디        |

|          |            |              |                 |
|----------|------------|--------------|-----------------|
| 아이템 이름   | item_name  | VARCHAR(255) | 아이템의 이름 문자열     |
| 아이템 유형   | item_type  | INT(11)      | 아이템의 유형 아이디     |
| 판매가격     | sell_price | INT(11)      | 아이템 판매가격        |
| 구매가격     | buy_price  | INT(11)      | 아이템 구매가격        |
| 아이템 출현장소 | item_place | INT(11)      | 아이템을 얻을 수 있는 장소 |

## 8. 테이블 기술서

| Table 기술서 |               |           |                    |     |                   |         |             |
|-----------|---------------|-----------|--------------------|-----|-------------------|---------|-------------|
| 테이블 이름    |               |           | creature           |     |                   |         |             |
| 테이블 설명    |               |           | 게임에서 제공하는 크리처의 데이터 |     |                   |         |             |
| No        | Attribute     | Data Type | NN                 | Key | Option            | Default | Description |
| 1         | Creature_id   | Int(11)   | Y                  | PK  | Auto_increment    |         | 크리처의 고유 아이디 |
| 2         | Creature_name | Char(255) | Y                  |     |                   |         | 크리처의 이름     |
| 3         | Description   | Char(255) | Y                  |     |                   |         | 크리처 설명      |
| 4         | Creature_type | Int(11)   | Y                  | FK  | ON UPDATE CASCADE |         | 크리처 속성      |
| 5         | Habitation    | Int(11)   | Y                  | FK  | ON UPDATE CASCADE |         | 크리처 서식지     |

| Table 기술서 |             |           |                  |     |                                     |         |             |
|-----------|-------------|-----------|------------------|-----|-------------------------------------|---------|-------------|
| 테이블 이름    |             |           | Creature_owned   |     |                                     |         |             |
| 테이블 설명    |             |           | 유저가 소유한 크리처의 데이터 |     |                                     |         |             |
| No        | Attribute   | Data Type | NN               | Key | Option                              | Default | Description |
| 1         | Creature_id | Int(11)   | Y                | FK  | ON DELETE CASCADE ON UPDATE CASCADE |         | 크리처의 고유 아이디 |
| 2         | User_id     | Char(15)  |                  | FK  | ON DELETE CASCADE ON                | NULL    | 유저 아이디      |



|   |               |         |   |    |                   |  |                   |
|---|---------------|---------|---|----|-------------------|--|-------------------|
|   |               |         |   |    | UPDATE<br>CASCADE |  |                   |
| 3 | weight        | Int(11) | Y |    |                   |  | 크리쳐 몸무게           |
| 4 | height        | Int(11) | Y |    |                   |  | 크리쳐 신장            |
| 5 | no            | Int(11) | Y | PK | Auto_increment    |  | 순번                |
| 6 | Creature_like | Int(11) |   |    |                   |  | 상성이 좋은<br>크리쳐 아이디 |

|        | Table 기술서                |             |                     |     |        |         |                 |
|--------|--------------------------|-------------|---------------------|-----|--------|---------|-----------------|
| 테이블 이름 |                          |             | Creature_habitation |     |        |         |                 |
| 테이블 설명 |                          |             | 크리쳐 서식지 데이터         |     |        |         |                 |
| No     | Attribute                | Data Type   | NN                  | Key | Option | Default | Description     |
| 1      | Creature_habitation      | Int(11)     | Y                   | PK  |        |         | 크리쳐의 서식지<br>아이디 |
| 2      | Creature_habitation_char | Varchar(50) |                     |     |        | NULL    | 크리쳐의 서식지<br>이름  |

|        | Table 기술서   |           |                                   |       |        |         |             |
|--------|-------------|-----------|-----------------------------------|-------|--------|---------|-------------|
| 테이블 이름 |             |           | Creature_random                   |       |        |         |             |
| 테이블 설명 |             |           | 크리쳐에게 랜덤으로 부여되는 몸무게, 신장의 최대최소 데이터 |       |        |         |             |
| No     | Attribute   | Data Type | NN                                | Key   | Option | Default | Description |
| 1      | Creature_id | Int(11)   | Y                                 | PK/FK |        |         | 크리쳐의 아이디    |
| 2      | Weight_min  | Int(11)   | Y                                 |       |        |         | 크리쳐의 몸무게    |

|   |            |         |   |  |  |  |                 |
|---|------------|---------|---|--|--|--|-----------------|
|   |            |         |   |  |  |  | 최소값             |
| 3 | Weight_max | Int(11) | Y |  |  |  | 크리처의 몸무게<br>최대값 |
| 4 | Height_min | Int(11) | Y |  |  |  | 크리처 신장<br>최소값   |
| 5 | Height_max | Int(11) | Y |  |  |  | 크리처 신장<br>최대값   |

|        | Table 기술서              |             |               |     |        |         |               |
|--------|------------------------|-------------|---------------|-----|--------|---------|---------------|
| 테이블 이름 |                        |             | Creature_type |     |        |         |               |
| 테이블 설명 |                        |             | 크리처의 속성 데이터   |     |        |         |               |
| No     | Attribute              | Data Type   | NN            | Key | Option | Default | Description   |
| 1      | Creature_type          | Int(11)     | Y             | PK  |        |         | 크리처의 속성       |
| 2      | Creature_type_c<br>har | Varchar(50) | Y             |     |        |         | 크리처의 속성<br>이름 |

|        | Table 기술서 |              |                    |     |        |         |             |
|--------|-----------|--------------|--------------------|-----|--------|---------|-------------|
| 테이블 이름 |           |              | item               |     |        |         |             |
| 테이블 설명 |           |              | 게임에서 제공하는 아이템의 데이터 |     |        |         |             |
| No     | Attribute | Data Type    | NN                 | Key | Option | Default | Description |
| 1      | Item_id   | Int(11)      | Y                  | PK  |        |         | 아이템의 아이디    |
| 2      | Item_name | Varchar(255) | Y                  |     |        |         | 아이템의 이름     |
| 3      | Item_type | Int(11)      | Y                  | FK  |        |         | 아이템 유형      |

|   |             |              |   |    |  |      |           |
|---|-------------|--------------|---|----|--|------|-----------|
| 4 | Sell_price  | Int(11)      | Y |    |  |      | 판매가       |
| 5 | Buy_price   | Int(11)      | Y |    |  |      | 구매가       |
| 6 | Description | Varchar(255) |   |    |  | NULL | 아이템 설명    |
| 7 | Item_place  | Int(11)      | Y | FK |  |      | 아이템 출현 장소 |

| Table 기술서 |           |                  |    |     |  |         |             |
|-----------|-----------|------------------|----|-----|--|---------|-------------|
| 테이블 이름    |           | Item_owned       |    |     |  |         |             |
| 테이블 설명    |           | 유저가 소유한 아이템의 데이터 |    |     |  |         |             |
| No        | Attribute | Data Type        | NN | Key | Option                                       | Default | Description |
| 1         | no        | Int(11)          | Y  | PK  |  |         | 순번          |
| 2         | User_id   | Varchar(50)      | Y  | FK  | ON DELETE<br>CASCADE ON<br>UPDATE<br>CASCADE |         | 유저 아이디      |
| 3         | Item_id   | Int(11)          | Y  | FK  | ON DELETE<br>CASCADE ON<br>UPDATE<br>CASCADE |         | 아이템 아이디     |
| 4         | Item_num  | Int(11)          | Y  |     |  |         | 아이템 소유 개수   |

| Table 기술서 |           |             |    |     |        |         |             |
|-----------|-----------|-------------|----|-----|--------|---------|-------------|
| 테이블 이름    |           | Item_type   |    |     |        |         |             |
| 테이블 설명    |           | 아이템 유형의 데이터 |    |     |        |         |             |
| No        | Attribute | Data Type   | NN | Key | Option | Default | Description |
| 1         | Item_type | Int(11)     | Y  | PK  |        |         | 아이템 유형      |

|   |                |             |   |  |  |  |           |
|---|----------------|-------------|---|--|--|--|-----------|
|   |                |             |   |  |  |  | 아이디       |
| 2 | Item_type_char | Varchar(50) | Y |  |  |  | 아이템 유형 이름 |

|        | Table 기술서       |             |                |     |        |         |                 |
|--------|-----------------|-------------|----------------|-----|--------|---------|-----------------|
| 테이블 이름 |                 |             | Item_place     |     |        |         |                 |
| 테이블 설명 |                 |             | 아이템 출현 장소의 데이터 |     |        |         |                 |
| No     | Attribute       | Data Type   | NN             | Key | Option | Default | Description     |
| 1      | Item_place      | Int(11)     | Y              | PK  |        |         | 아이템 출현장소<br>아이디 |
| 2      | Item_place_char | Varchar(50) | Y              |     |        |         | 아이템 출현장소<br>이름  |

|        | Table 기술서 |           |           |     |        |         |             |
|--------|-----------|-----------|-----------|-----|--------|---------|-------------|
| 테이블 이름 |           |           | members   |     |        |         |             |
| 테이블 설명 |           |           | 회원 정보 데이터 |     |        |         |             |
| No     | Attribute | Data Type | NN        | Key | Option | Default | Description |
| 1      | User_id   | Char(15)  | Y         | PK  |        |         | 유저 아이디      |
| 2      | name      | Char(15)  |           |     |        | NULL    | 유저 이름       |
| 3      | birth     | Char(8)   |           |     |        | NULL    | 유저 생일       |
| 4      | email     | Char(30)  |           |     |        | NULL    | 유저 이메일      |
| 5      | pw        | Char(32)  |           |     |        | NULL    | 유저 비밀번호     |
| 6      | money     | Int(11)   | Y         |     |        | 0       | 유저별 소지한 돈   |

## 9. 정규화 과정

### 정규화 과정

#### 0. 개요

테이블 creature 를 이용해 정규화 과정을 설명한다. 이하 예시는 1 차 정규화 이전의 테이블이다. user\_id 를 primary key 로 가진다. 이 경우 아래와 같은 이상현상이 발생하게 된다.

| User_id   | Creature_id | Creature_type | User_name |
|-----------|-------------|---------------|-----------|
| Alley5653 | 1,2,4       | ...           | Kyuri Kim |
| test      | 3           | ...           | Choi      |
| Alley5653 | 9           | ...           | Kyuri Kim |
| Alley5653 | 5           | ...           | Kyuri Kim |

삽입 이상 : 삽입이 불가능하다. primary key 는 중복되지 않아야한다.

삭제 이상 : 테이블에서 test 라는 유저를 삭제한다면 크리처의 이름 정보도 함께 삭제되어버린다.

갱신 이상 : User\_name 을 수정하고자 할 경우 총 3 번의 update 가 필요하다. 불필요한 update 연산이 생긴다.

#### 1. 1 차 정규화

아래와 같이 테이블을 creature 와 members 로 분리하였다. 1 차 정규형은 각 열마다 컬럼의 값이 1 개씩만 존재하여야한다. 이를 컬럼이 원자값을 갖는다고 표현한다. creature 테이블의 기본키는 이제 no 이다.

<Table Creature>

| no | User_id   | Creature_id | Creature_type | Creature_type_char |
|----|-----------|-------------|---------------|--------------------|
| 1  | Alley5653 | 1           | 1             | 불                  |
| 2  | test      | 3           | 1             | 불                  |
| 3  | Alley5653 | 9           | 2             | 땅                  |
| 4  | Alley5653 | 5           | 3             | 바이러스               |

<Table members>

| User_id   | User_name |
|-----------|-----------|
| Alley5653 | Kyuri Kim |

삽입 이상 : test 와 같이 다른 회원이 추가될 경우, creature 정보가 중복된다.

삭제 이상 : creature\_type\_char = 땅을 삭제할 경우 creature\_id 정보도 함께 삭제되어 버린다.

갱신 이상 : creature\_type\_char 을 수정할 때, update 를 두번 반복해야한다.

## 2. 2 차 정규화

아래와 같이 테이블을 creature 와 creature\_type 과 members 로 분리하였다. 2 차 정규화는 1 차 정규화의 속성을 만족하면서, 테이블에 존재하는 모든 함수 종속 관계가 완전 함수 종속이어야 한다.

<Table Creature>

| no | User_id   | Creature_id | Creature_type |
|----|-----------|-------------|---------------|
| 1  | Alley5653 | 1           | 1             |
| 2  | test      | 1           | 1             |
| 3  | Alley5653 | 9           | 2             |
| 4  | Alley5653 | 5           | 3             |

<Table creature\_type>

| Creature_type | Creature_type_char |
|---------------|--------------------|
| 1             | 불                  |
| 2             | 땅                  |
| 3             | 바이러스               |

<Table members>

| User_id   | User_name |
|-----------|-----------|
| Alley5653 | Kyuri Kim |

삽입 이상 : 여전히, test 와 같이 다른 회원이 추가될 경우 creature 정보가 중복된다.

삭제 이상 : 여전히, alley5653 이라는 회원이 탈퇴할 경우 creature 정보도 함께 삭제되어 버린다.

갱신 이상 : 여전히, creature\_type 을 갱신할 경우 update 문을 두번 반복해야한다.

## 3. 3 차 정규화

아래와 같이 테이블을 creature 와 creature\_owned 와 creature\_type 과 members 로 분리하였다. 2 차 정규화의 정의를 만족하면서, 각 테이블에 존재하는 키가 아닌 필드들이 서로 독립적이고, 이행적 함수 종속이 없어야 한다.

<Table Creature\_owned>

| no | User_id   | Creature_id |
|----|-----------|-------------|
| 1  | Alley5653 | 1           |

|   |           |   |
|---|-----------|---|
| 2 | test      | 1 |
| 3 | Alley5653 | 9 |
| 4 | Alley5653 | 5 |

<Table Creature>.

| Creature_id | Creature_type |
|-------------|---------------|
| 1           | 1             |
| 1           | 1             |
| 9           | 2             |
| 5           | 3             |

<Table creature\_type>

| Creature_type | Creature_type_char |
|---------------|--------------------|
| 1             | 불                  |
| 2             | 땅                  |
| 3             | 바이러스               |

<Table members>

| User_id   | User_name |
|-----------|-----------|
| Alley5653 | Kyuri Kim |

## 정규화 과정 이후, 테이블 간의 관계

### 1. Creature\_owned 테이블과 Creature 테이블 간의 관계

. Creature\_owned 에서 다루는 크리쳐 관련 정보는 모두 Creature 테이블에 속해 있고, Creature\_owned 테이블에서는 Creature\_id 만을 가진다. creature\_id 를 통해 Creature\_owned 테이블은 creature 테이블에서 크리쳐 관련 정보를 얻을 수 있다.

### 2. Item\_owned 테이블과 Item 테이블 간의 관계

item\_owned 에서 다루는 아이템 관련 정보는 모두 Item 테이블에 속해 있고, item\_owned 테이블에서는 item\_id 만을 가진다. item\_id 를 통해 item\_owned 테이블은 item 테이블에서 아이템 관련 정보를 얻을 수 있다.

### 3. Creature\_habitation 테이블과 Creature 테이블 간의 관계

Creature 테이블에서 다루는 크리쳐 서식지의 정보는 모두 Creature\_habitation 테이블에 속해 있고, Creature 테이블에서는 habitation 정수값만을 가진다. habitation 값을 통해 creature 테이블은 creature\_habitation 테이블에서 크리쳐 서식지의 이름, 즉 문자열을 얻을 수 있다.

### 4. Creature\_type 테이블과 Creature 테이블 간의 관계

Creature 테이블에서 다루는 크리쳐 속성의 정보는 모두 Creature\_type 테이블에 속해 있고, Creature 테이블에서는 creature\_type 정수값만을 가진다. creature\_type 값을 통해 creature 테이블은 creature\_type 테이블에서 크리쳐 속성의 이름, 즉 문자열을 얻을 수 있다.

### 5. member 테이블과 Creature\_owned 테이블, item\_owned 테이블, creature\_history 테이블 간의 관계

creature\_owned 테이블과 item\_owned 테이블, creature\_history 테이블은 모두 user\_id 를 저장하는 칼럼을 가지고 있다. member 테이블에 저장된 user\_id 값 만으로, 세가지 테이블에서 해당 유저가 가지고 있는 크리쳐, 아이템, 크리쳐 획득 기록 정보를 얻을 수 있다.



## 10. 현존 오류

### 현존 오류

1. 제작한 홈페이지에서, 직전에 로그인하였다가 탈퇴한 아이디 회원가입을 재진행했을 경우 회원가입 절차가 끝나면 해당 아이디로 자동로그인 되는 현상이 발견되었다. 본래라면 회원가입 후 별도의 로그인 절차를 거쳐야만 한다.

캐시 사용의 문제임으로 추정하고 있으나, 프로젝트 종료 시까지 이러한 문제를 해결하지 못하였다.

## 11. 프로젝트 기여도

| 기여도   |  |
|---|--|
| 김규리   | 최승환  |
| <ol style="list-style-type: none"> <li>1. 개발 환경 구축 : Apache 2.4 설치 등 서버 구축과 php, atom IDE 등의 개발환경을 구축하였다.</li> <li>2. 웹페이지 개발 : 모든 웹페이지 개발을 전담하였다. 개인 소유의 노트북 localhost 에서 프로젝트를 개발하였다.</li> <li>3. 데이터베이스 구축 : 프로젝트에 필요한 모든 데이터베이스를 구축하였다. 프로젝트에서 사용한 데이터베이스는 해당 인원의 mysql root 계정에 구축되어 있다.</li> <li>4. 보고서 작성 : 보고서 중 요구사항 분석, 테이블 기술서, 정규화 과정 등을 작성하였다.</li> <li>5. 발표자료 작성 : 발표자료 초안을 받아 마무리 작업을 진행하였다.</li> </ol> | <ol style="list-style-type: none"> <li>1. 보고서 작성 : 보고서 중 도메인 기술서, 용어사전, 프로젝트 개요 등을 작성하였다.</li> <li>2. 발표자료 작성 : 발표자료 초안을 작성하였다.</li> <li>3. 논리, 물리 E-R Diagram 을 작성하였다.</li> </ol> |