

Project. ON:DO

Team. 7 James

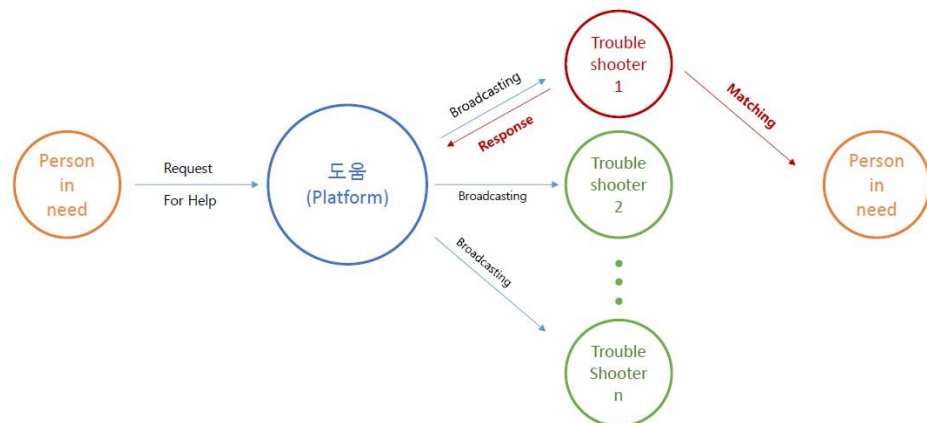
1. 프로젝트 개요

1) 프로젝트 소개

- i. P2P 음성/영상 통화를 Matching 해 주는 Platform Service를 구현합니다.

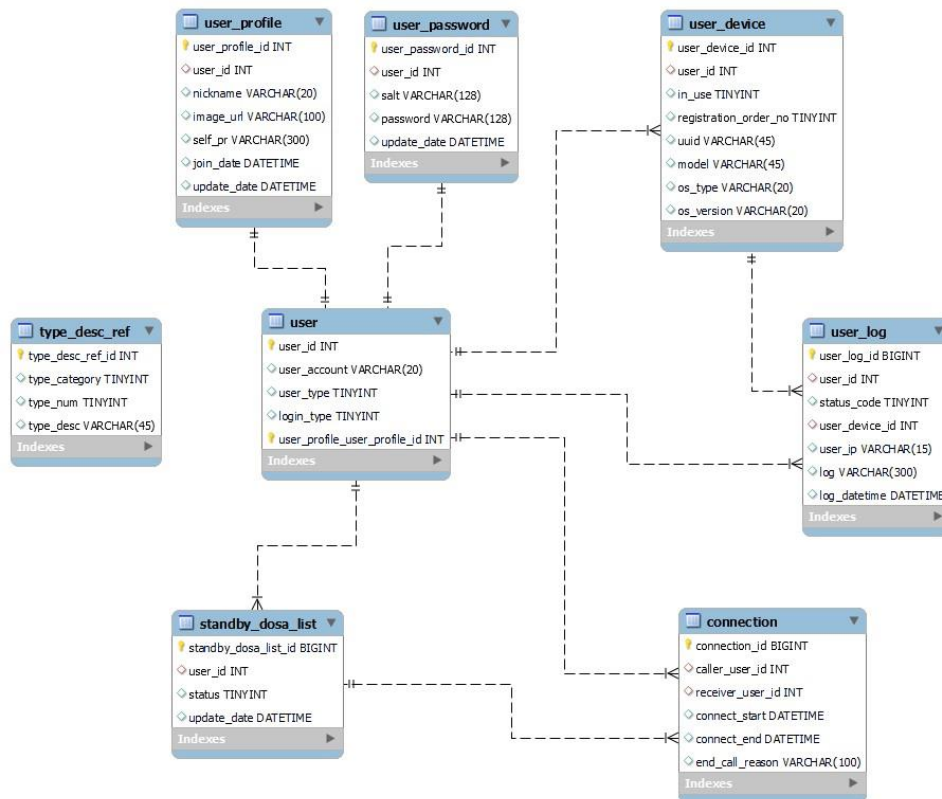
2) 프로젝트 시나리오

- i. Caller Group과 Callee Group이 회원타입으로 구분되어 있으며, Call을 대기 (Standby)하고 있는 Callee Group 중에서 제일 먼저 수락한 Client와 Connection을 연결합니다.



2. 설계

1) Relation Schema



3. 구현

1) Relation

i. 회원 관련 Table

```

-----
-- Table `james`.`user`
-----

CREATE TABLE IF NOT EXISTS `james`.`user` (
  `user_id` INT NOT NULL,
  `user_account` VARCHAR(20) NULL,
  `user_type` TINYINT NULL,
  `login_type` TINYINT NULL,
  `user_profile_user_profile_id` INT NOT NULL,
  PRIMARY KEY (`user_id`, `user_profile_user_profile_id`)
ENGINE = InnoDB;

-----

-- Table `james`.`user_profile`
-----

CREATE TABLE IF NOT EXISTS `james`.`user_profile` (
  `user_profile_id` INT NOT NULL,
  `user_id` INT NULL,

```

```

`nickname` VARCHAR(20) NULL,
`image_url` VARCHAR(100) NULL,
`self_pr` VARCHAR(300) NULL,
`join_date` DATETIME NULL,
`update_date` DATETIME NULL,
PRIMARY KEY (`user_profile_id`),
INDEX `fk_user_id_idx` (`user_id` ASC) VISIBLE,
CONSTRAINT `fk_user_profile_user_user_id`
    FOREIGN KEY (`user_id`)
    REFERENCES `james`.`user` (`user_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----

-- Table `james`.`user_device`
-----

CREATE TABLE IF NOT EXISTS `james`.`user_device` (
  `user_device_id` INT NOT NULL,
  `user_id` INT NULL,
  `in_use` TINYINT UNSIGNED NULL,
  `registration_order_no` TINYINT UNSIGNED NULL,
  `uuid` VARCHAR(45) NULL,
  `model` VARCHAR(45) NULL,
  `os_type` VARCHAR(20) NULL,
  `os_version` VARCHAR(20) NULL,
  PRIMARY KEY (`user_device_id`),
  INDEX `fk_user_device_user_user_id_idx` (`user_id` ASC) VISIBLE,
  INDEX `fk_user_device_user_log_device_id_idx` (`user_id` ASC, `user_device_id` ASC) VISIBLE,
  CONSTRAINT `fk_user_device_user_user_id`
    FOREIGN KEY (`user_id`)
    REFERENCES `james`.`user` (`user_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----

-- Table `james`.`user_password`
-----

CREATE TABLE IF NOT EXISTS `james`.`user_password` (
  `user_password_id` INT NOT NULL,
  `user_id` INT NULL,
  `salt` VARCHAR(128) NULL,
  `password` VARCHAR(128) NULL,
  `update_date` DATETIME NULL,
  PRIMARY KEY (`user_password_id`),
  INDEX `fk_user_password_user_user_id_idx` (`user_id` ASC) VISIBLE,
  CONSTRAINT `fk_user_password_user_user_id`

```

```

FOREIGN KEY (`user_id`)
REFERENCES `james`.`user` (`user_id`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----

-- Table `james`.`user_log`
-----

CREATE TABLE IF NOT EXISTS `james`.`user_log` (
  `user_log_id` BIGINT NOT NULL,
  `user_id` INT NULL,
  `status_code` TINYINT NULL,
  `user_device_id` INT NULL,
  `user_ip` VARCHAR(15) NULL,
  `log` VARCHAR(300) NULL,
  `log_datetime` DATETIME NULL,
  PRIMARY KEY (`user_log_id`),
  INDEX `fk_user_log_user_user_id_idx` (`user_id` ASC) VISIBLE,
  INDEX `fk_user_log_user_device_user_device_id_idx` (`user_id` ASC, `user_device_id` ASC) VISIBLE,
  CONSTRAINT `fk_user_log_user_user_id`
    FOREIGN KEY (`user_id`)
      REFERENCES `james`.`user` (`user_id`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
  CONSTRAINT `fk_user_log_user_device_user_device_id`
    FOREIGN KEY (`user_id`, `user_device_id`)
      REFERENCES `james`.`user_device` (`user_id`, `user_device_id`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

ii. Matching 관련 테이블

```

-----

-- Table `james`.`standby_dosa_list`
-----

CREATE TABLE IF NOT EXISTS `james`.`standby_dosa_list` (
  `standby_dosa_list_id` BIGINT NOT NULL,
  `user_id` INT NULL,
  `status` TINYINT NULL,
  `update_date` DATETIME NULL,
  PRIMARY KEY (`standby_dosa_list_id`),
  INDEX `fk_standby_dosa_list_user_user_id_idx` (`user_id` ASC) VISIBLE,
  CONSTRAINT `fk_standby_dosa_list_user_user_id`
    FOREIGN KEY (`user_id`)

```

```

REFERENCES `james`.`user` (`user_id`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----

-- Table `james`.`connection`
-----

CREATE TABLE IF NOT EXISTS `james`.`connection` (
  `connection_id` BIGINT NOT NULL,
  `caller_user_id` INT NULL,
  `receiver_user_id` INT NULL,
  `connect_start` DATETIME NULL,
  `connect_end` DATETIME NULL,
  `end_call_reason` VARCHAR(100) NULL,
  PRIMARY KEY (`connection_id`),
  INDEX `fk_connection_user_caller_user_id_idx` (`caller_user_id` ASC) VISIBLE,
  INDEX `fk_connection_user_receiver_user_id_idx` (`receiver_user_id` ASC) VISIBLE,
  CONSTRAINT `fk_connection_user_caller_user_id`
    FOREIGN KEY (`caller_user_id`)
      REFERENCES `james`.`user` (`user_id`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  CONSTRAINT `fk_connection_user_receiver_user_id`
    FOREIGN KEY (`receiver_user_id`)
      REFERENCES `james`.`standby_dosa_list` (`user_id`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

iii. 기타 테이블

```

-----

-- Table `james`.`type_desc_ref`
-----

CREATE TABLE IF NOT EXISTS `james`.`type_desc_ref` (
  `type_desc_ref_id` INT NOT NULL,
  `type_category` TINYINT NULL,
  `type_num` TINYINT NULL,
  `type_desc` VARCHAR(45) NULL,
  PRIMARY KEY (`type_desc_ref_id`))
ENGINE = InnoDB;

```

2) 기능 별 쿼리

i. 회원 관련 테이블

```
INSERT INTO `james`.`user`
(`user_id`,
`user_account`,
`user_type`,
`login_type`,
`user_profile_user_profile_id`)
VALUES
(<{user_id: }>,
<{user_account: }>,
<{user_type: }>,
<{login_type: }>,
<{user_profile_user_profile_id: }>);

INSERT INTO `james`.`user_device`
(`user_device_id`,
`user_id`,
`in_use`,
`registration_order_no`,
`uuid`,
`model`,
`os_type`,
`os_version`)
VALUES
(<{user_device_id: }>,
<{user_id: }>,
<{in_use: }>,
<{registration_order_no: }>,
<{uuid: }>,
<{model: }>,
<{os_type: }>,
<{os_version: }>);

INSERT INTO `james`.`user_log`
(`user_log_id`,
`user_id`,
`status_code`,
`user_device_id`,
`user_ip`,
`log`,
`log_datetime`)
VALUES
(<{user_log_id: }>,
<{user_id: }>,
```

```

<{status_code: }>,
<{user_device_id: }>,
<{user_ip: }>,
<{log: }>,
<{log_datetime: }>;

INSERT INTO `james`.`user_password`
(`user_password_id`,
`user_id`,
`salt`,
`password`,
`update_date`)
VALUES
(<{user_password_id: }>,
<{user_id: }>,
<{salt: }>,
<{password: }>,
<{update_date: }>);

INSERT INTO `james`.`user_profile`
(`user_profile_id`,
`user_id`,
`nickname`,
`image_url`,
`self_pr`,
`join_date`,
`update_date`)
VALUES
(<{user_profile_id: }>,
<{user_id: }>,
<{nickname: }>,
<{image_url: }>,
<{self_pr: }>,
<{join_date: }>,
<{update_date: }>);

```

ii. 매칭 관련 테이블

```

INSERT INTO `james`.`standby_dosa_list`
(`standby_dosa_list_id`,
`user_id`,
`status`,
`update_date`)
VALUES
(<{standby_dosa_list_id: }>,
<{user_id: }>,
<{status: }>,
<{update_date: }>);

```

```
INSERT INTO `james`.`standby_dosa_list`  
(`standby_dosa_list_id`,  
`user_id`,  
`status`,  
`update_date`)  
VALUES  
(<{standby_dosa_list_id: }>,  
<{user_id: }>,  
<{status: }>,  
<{update_date: }>);
```

iii. 기타 테이블

```
INSERT INTO `james`.`type_desc_ref`  
(`type_desc_ref_id`,  
`type_category`,  
`type_num`,  
`type_desc`)  
VALUES  
(<{type_desc_ref_id: }>,  
<{type_category: }>,  
<{type_num: }>,  
<{type_desc: }>);
```

3) System Architecture

