



LIFT ON

BEST ASSISTANT FOR YOUR LIFT LIFE

21912067

김동현

tmdrl583205@naver.com

[Revision history]

Revision date	Version #	Description	Author
2023/03/15	0.00	First Document	김동현

= Contents =

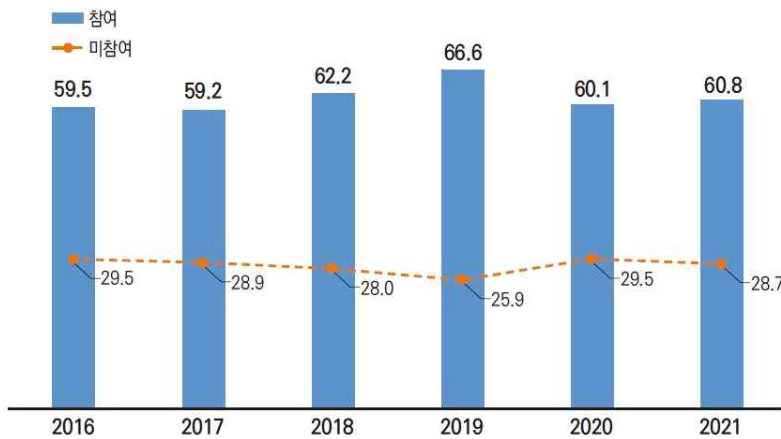
1. Business purpose	4
2. System context diagram	5
3. Use case list	6
4. Concept of operation	8
5. Problem statement	11
6. Glossary	12
7. References	12

1. Business purpose

2023년 많은 사람들은 운동을 즐긴다. 건강, 취미, 직업, 또는 심미적인 목적 등 다양한 목적을 위해 운동을 한다. 과거에 비해 운동을 즐기는 사람들이 더 늘어나고 그에 따라 피트니스 업계는 더욱 커지고 있다. 문화 체육 관광부의 조사인 그림 1에 따르면, 운동을 즐기는 사람은 최근 들어 크게 증가한 것으로 보인다.

〈그림 1〉 최근 5년간 생활체육 참여율

(단위: %)



출처: 문화체육관광부(2022). 2021 국민생활체육조사

그림 1 생활 체육 참여 실태

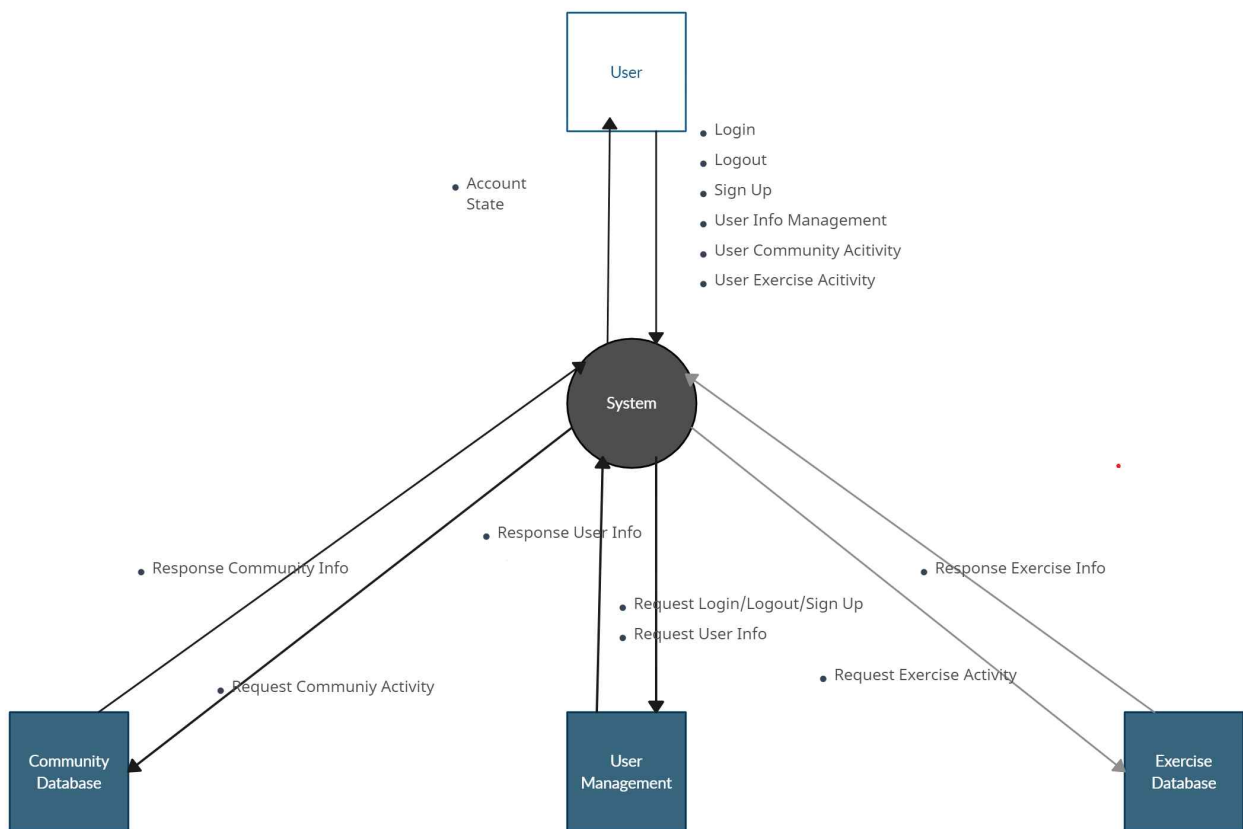
또한 많은 사람들이 헬스를 하므로 이 앱은 헬스를 중심으로 하는 애플리케이션이 될 것이며 차후에 다른 종목에 관한 정보를 추가할 것이다.

그러나, 많은 사람들이 운동을 하는데 불편함을 겪는다. 예를 들어, 어느 분야든 운동을 처음 하는 사람은 그 운동에 대한 기초적인 자세를 숙지하지 못할 수 있다. 그러한 부분은 운동을 잘 아는 타인, 트레이너에게 배울 수 있다. 하지만, 개인이 그 비용을 감당하기란 만만치 않으며 이러한 이유로 많은 사람이 잘못된 자세로, 더 나아가 잘못된 운동 방향을 잡고 오히려 몸을 망치는 경우가 있다.

이러한 배경에서 운동을 하기 위한 올바른 자세와 올바른 방향성을 사람들에게 제시해 주기 위해 피트니스 애플리케이션인 "Lift On"를 만들게 되었다. 이 애플리케이션의 목표는 앞서 말했다시피 사용자에게 올바른 운동 방법을 알려주는데, 그 방법으로 다양한 이미지를 활용해 올바른 자세를 사용자에게 알려주는 기능을 가진다. 그리고 유저들이 운동에 관해 정보를 공유할 수 있는 커뮤니티를 만들고 서로 원활한 정보 공유가 가능토록 만든다. 또한 트레이너가 자신의 운동 관련 자격증을 등록해 유저들에게 전문적으로 운동을 가르쳐 주고, 조언해 줄 수 있다. 그 보상으로 자신의 체육관, 헬스장 또는 자신이 광고하는 운동 제품 등을 광고할 수 있게 한다. 마지막으로 유저 스스로가 운동에 대한 목표와 계획을 짜고, 수행한 운동을 기록할 수 있게 한다.

이러한 기능을 가진 애플리케이션은 운동을 즐기는 많은 유저, 운동을 가르치는 트레이너, 운동을 입문하는 초심자에게 많은 도움이 될 것으로 예상된다.

2. System context diagram



-
- Login
 - Logout
 - Sign Up
 - User Info Management
 - User Community Activity
 - User Exercise Activity
 - Account State
 - Request Login / Logout / Sign Up
 - Request User Info
 - Response User Info
 - Request Community Activity
 - Response Community Info
 - Request Exercise Activity
 - Response Exercise Info

3. Use case list

1) Login

Actor	User
Description	유저가 자신의 아이디와 패스워드로 로그인한다.

2) Logout

Actor	User
Description	유저가 사용을 현재 아이디의 사용을 끝내고 로그아웃한다.

3) Sign Up

Actor	User
Description	유저가 어플리케이션의 사용을 위해 회원가입한다.

4) User Info Management

Actor	User
Description	유저가 자신의 개인정보를 열람, 변경, 삭제하는 등의 관리를 한다.

5) User Community Activity

Actor	User
Description	유저가 자신이 올린 게시물을 관리한다. (게시물 열람, 삭제, 수정 등등)

6) User Exercise Activity

Actor	User
Description	유저가 일반 사용자인지 트레이너인지의 유형을 확인한다.

7) Account State

Actor	Server
Description	서버가 유저에게 계정 상태를 보낸다.

8) Request Login / Logout / Sign Up

Actor	User
Description	유저가 로그인 / 로그아웃 / 회원가입을 서버에 요청한다.

9) Request User Info

Actor	User
Description	유저가 서버에 자신의 개인 정보 관련 활동을 요청한다 (개인 정보 열람, 개인 정보 갱신 등등)

10) Response User Info

Actor	Server
Description	서버가 유저에게 요청받은 유저 정보를 보내준다.

11) Request Community Activity

Actor	User
Description	유저가 자신의 커뮤니티 관련 활동을 서버에 요청한다. (게시물 게시, 게시물 삭제, 댓글 작성 등등)

12) Response Community Info

Actor	Server
Description	서버가 유저에게 요청받은 커뮤니티 관련 활동 정보를 보낸다.

13) Request Exercise Activity

Actor	User
Description	유저가 자신의 운동 관련 활동을 서버에 요청한다. (기록 열람, 기록 갱신, 목표 설정 등등)

14) Response Exercise Info

Actor	Server
Description	서버가 유저에게 요청받은 유저 정보를 보내준다

4. Concept of operation

1) Login

Purpose	앱을 사용하기 위해 등록된 유저인지 확인
Approach	유저가 앱을 실행 후 로그인 시, ID, PW를 입력 후 로그인을 요청하면 서버에서 회원 정보를 조회 후 로그인 성공/실패 여부, 즉 계정 상태를 반환
Dynamics	앱 실행 시 로그인할 경우
Goals	로그인 기능을 구현한다.

2) Logout

Purpose	다른 아이디로 로그인 또는 다른 이유로 로그아웃
Approach	유저가 앱을 실행하는 중 로그아웃 할 수 있도록 한다.
Dynamics	로그아웃을 실행하는 경우
Goals	로그아웃 기능 구현

3) Sign Up

Purpose	앱을 사용하기 위해 계정을 등록
Approach	유저가 등록한 개인 정보를 받은 후 중복되는 ID가 없는지 확인 후 서버에 계정을 등록 (저장)
Dynamics	유저가 로그인하기 위하는 경우
Goals	회원가입 기능을 구현

4) User Info Management

Purpose	유저가 자신의 개인 정보를 열람, 수정, 삭제 등등의 개인 정보 관리 활동 수행
Approach	유저가 자신의 개인 정보를 서버에 요청하고 그 정보를 받아 다양한 활동을 수행할 수 있다.
Dynamics	유저가 자신의 개인 정보를 수정하기를 원하는 등의 경우
Goals	개인 정보 관리 기능, 개인 정보 요청, 반환 등의 기능 구현하여 원활한 정보 관리를 할 수 있도록 한다

5) User Community Activity

Purpose	유저가 커뮤니티에서 게시물 등록 / 삭제 / 수정 또는 댓글 등록 / 삭제 / 수정 등의 활동 수행
Approach	유저는 커뮤니티에 자유롭게 다양한 게시물 또는 댓글을 등록하는 등의 활동 수행할 수 있다.
Dynamics	유저가 커뮤니티에서 활동하기를 원하는 경우
Goals	유저의 게시물 등록 / 삭제 / 수정 or 댓글 등록 / 삭제 / 수정 등의 활동 수행을 하는 기능을 구현하여 커뮤니티에서 다양한 활동할 수 있도록 한다.

6) User Exercise Activity

Purpose	유저가 자신의 운동 계획, 운동 기록, 운동 목표를 설정
Approach	유저는 자신의 운동에 관한 것을 계획하고 기록하며 목표를 설정할 수 있다.
Dynamics	유저가 자신의 운동 계획, 운동 기록, 운동 목표를 설정하기를 원하는 경우
Goals	운동 계획, 운동 기록, 운동 목표를 설정하는 기능 구현하여 유저의 운동에 충분한 보조 역할이 될 수 있도록 한다.

7) Account State

Purpose	유저가 앱을 실행했을 때 서버가 유저의 계정 상태 (로그인, 로그아웃)를 보냄
Approach	유저는 자신이 로그인, 로그아웃되었는지 확인한다.
Dynamics	유저가 자신이 로그인되었는지 로그아웃 되었는지 확인하고 싶은 경우
Goals	유저가 자신의 상태를 확인할 수 있도록 한다.

8) Request Login / Logout / Sign Up

Purpose	유저가 로그인 / 로그아웃 / 회원가입 요청
Approach	유저는 로그인 / 로그아웃 / 회원가입 요청한다.
Dynamics	유저가 로그인 / 로그아웃 / 회원가입 하고 싶은 경우
Goals	유저가 로그인 / 로그아웃 / 회원가입을 원활히 할 수 있도록 한다.

9) Request User Info

Purpose	유저가 자신의 개인 정보 요청
Approach	유저는 자신의 개인 정보를 다양한 목적을 위해 요청할 수 있다.
Dynamics	유저가 자신의 개인 정보를 열람, 수정, 삭제하고 싶은 경우
Goals	유저가 자신의 개인 정보를 원활히 열람, 수정, 삭제할 수 있도록 기능을 구현한다.

10) Response User Info

Purpose	서버가 유저로부터 받은 개인 정보를 반환
Approach	서버는 유저의 개인 정보를 반환하고 유저는 개인 정보를 관리할 수 있다.
Dynamics	서버가 요청받은 유저의 개인 정보를 보내는 경우
Goals	서버의 response 가 잘 작동하도록 기능을 구현하여 유저가 개인 정보를 잘 관리할 수 있도록 한다.

11) Request Community Activity

Purpose	유저가 커뮤니티 관련 활동을 할 때 게시물 / 댓글 작성, 삭제, 수정 등의 활동을 하기 위해 서버에 요청한다.
Approach	유저는 커뮤니티 관련 활동을 원활히 할 수 있다.
Dynamics	유저가 커뮤니티에 게시물 / 댓글을 작성, 삭제, 수정할 경우
Goals	유저가 커뮤니티 관련 활동을 하기 위해 서버에 원활히 요청할 수 있도록 한다.

12) Response Community Info

Purpose	유저가 앱을 실행했을 때, 커뮤니티의 글 목록을 열람할 수 있고, 자신이 작성했던 글, 댓글을 열람할 수 있도록 한다.
Approach	서버는 유저에게 커뮤니티 관련 정보를 보내준다.
Dynamics	유저가 앱을 실행했을 때 또는 열람 요청을 하여 서버가 유저에게 커뮤니티 관련 정보를 보내줘야할 때
Goals	서버는 항상 유저에게 커뮤니티 관련 정보를 원활히 보낼 수 있도록 한다.

13) Request Exercise Activity

Purpose	유저가 자신의 운동 정보를 열람, 수정, 삭제하고 싶을 때 서버에 요청한다.
Approach	유저는 자신의 운동 정보를 원활히 관리할 수 있다.
Dynamics	유저가 자신의 운동 정보를 관리하고 싶은 경우
Goals	유저는 운동 계획, 목표, 기록을 설정하고 이를 서버에 저장하기 위해 요청하는데, 이를 원활히 하도록 한다.

14) Response Exercise Info

Purpose	서버가 유저로부터 요청받은 운동 정보를 반환해준다.
Approach	서버가 유저에게 유저의 운동 관련 정보를 보낸다
Dynamics	서버가 유저에게 유저의 운동 관련 정보를 보내야 할 경우
Goals	유저가 요청한 정보를 서버가 원활히 보낼 수 있도록 한다.

5. Problem statement

Problem #1 – 애플리케이션 개발을 위해 Flutter 프레임워크를 사용해야 하는데, 애플리케이션 개발을 해본 적이 없다. Flutter 프레임워크의 이해도를 깊이 하기 위해 공부할 것이다.

Problem #2 – 서버 구축을 위해 TypeScript 와 Express.js 를 이용해야 하는데, TypeScript 로는 개발 경험이 없다. 이를 위해 TypeScript 의 경험과 이해도를 높이기 위해 공부할 것이다.

Problem #3 – 내가 TypeScript 와 Express.js를 이용해 만든 서버와 Flutter 프레임워크를 어떻게 적절히 연결해야 하는지 숙지를 못했다. 이를 위해 서버와 클라이언트를 연결하기 위한 공부를 할 것이다.

Problem #4 – 일반 사용자와 트레이너의 User Type 을 잘 분리하기 위해 적절한 자격 인증 절차를 밟아야 하는데, 그 절차를 만들기 위해 트레이너가 자신의 자격증 등을 올리고 유저가 이를 열람하여 트레이너가 게시하는 게시물의 정보를 믿을 수 있는지의 여부를 체크하고 좋아요 와 같은 버튼을 설정하여 그 트레이너의 신뢰도 여부를 측정할 것이다.

6. Glossary

- **유저** : 실질적인 앱의 사용자로, 운동을 즐기고 배우기 위한 일반 사용자와 운동을 가르치고 조언하는 트레이너로 나누어져 있다.
- **계정 상태** : 현재 유저가 로그인했는지, 로그아웃했는지 등의 상태
- **커뮤니티** : 유저들이 커뮤니티 공간에서 다양한 운동 정보를 주고받으며 서로 성장을 도모할 수 있도록 한다
- **운동 목표 설정** : 유저가 장기적인 목표를 설정할 수 있도록 한다.
- **운동 계획 설정** : 유저가 내일 또는 근 미래의 운동에 관한 계획을 세울 수 있도록 한다.
- **운동 기록 저장** : 유저가 최근에 했던 운동 기록을 서버에 저장한다.

7. References

- 그림 1 생활 체육 참여 실태 : 문화체육관광부 2021 국민생활체육참여조사