

Mobility Mate

교통약자를 위한 네비게이션 앱

Test Plan Document



2021 학기 1 학기 소프트웨어공학개론 3 팀

이석현 2014311748

안리아 2016312126

우준하 2015318620

이주민 2017310695

Table of Contents

1. Introduction - - - - -	3
1.1 Purpose - - - - -	3
1.2 Scope - - - - -	3
1.3 Definitions, Acronyms, and Abbreviation - - - - -	4
1.4 References - - - - -	5
1.5 Overview - - - - -	5
2. Approach - - - - -	6
2.1 Test method - - - - -	6
2.1.1 Software unit test methods - - - - -	6
2.1.2. Software interface test methods - - - - -	9
3. Software Unit Test - - - - -	13
3.1 Register - - - - -	13
3.1.1 Google account register test case - - - - -	13
3.1.2 Google account register test case - Negative - - - - -	13
3.2 검색 - - - - -	13
3.2.1 검색 - - - - -	13
3.2.2 검색 - Negative - - - - -	13

3.3 경로찾기 - - - - -	14
3.3.1 경로찾기 - - - - -	14
3.3.2 경로찾기 - Negative - - - - -	14
3.4 Quick Access - - - - -	15
3.3.1 Quick Access - - - - -	15
3.3.2 Quick Access - Negative - - - - -	15
4. Software Interface Test - - - - -	16
4.1 Login Server - - - - -	16
4.1.1 Connection test - - - - -	16
4.2 Service Server - - - - -	17
4.2.1 Connection test - - - - -	17
4.3 공사 현장 API - - - - -	18
4.3.1 Connection test - - - - -	18
4.4 MongoDB - - - - -	19
4.4.1 Connection test - - - - -	19
5. Supporting Information - - - - -	19
5.1 Document History - - - - -	19

1. Introduction

‘Mobility Mate’의 어플리케이션에 대해 간략히 소개한다.

1.1 Purpose

이 어플리케이션은 교통 약자 개인에게 맞는 경로를 추천하는 시스템을 제공한다. 사용자 정보에 기반한 알고리즘을 사용해 사용자의 이동 수행 능력과 보도 상황에 맞는 경로를 찾아 사용자에게 가장 적합한 추천 경로를 결정하여 제공한다.

이 문서의 목표는 요구 사항 및 테스트 레벨에 대한 개요를 제공하는 것이다. 각 요구사항은 일련의 테스트 절차로 분류된다. 이를 수행하는 프로세스는 표준 IEEE 829-2008 에 따라 수행된다.

1.2 Scope

이 테스트 계획서는 테스트 사양이 구현하는 기본 지침이다. ‘교통 약자 맞춤 경로 안내 어플’은 공사 중인 보도, 경사로 등의 보행 환경과 교통약자가 본인의 이동 수행 능력을 고려하여 옵션을 통해 고도가 낮은 길 또는 빠른 길을 안내한다. 이때, 주변에 가까운 병원, 휠체어 에스컬레이터, 수유실, 쉼터 등과 같은 시설 정보를 얻을 수 있다. 또한, 로그인을 통한 개인 맞춤 정보를 제공하므로, 교통약자가 직접 작성한 교통약자의 이동 수행 능력 정보를 수집할 수 있다. 관련된 정보들로는 휠체어와 같은 이동 보조기구 유무, 계단이나 오르막길을 다닐 수 있는 능력 등이 있다. GPS 트래킹 기능을 키면 교통약자의 이동 경로 또한 수집할 수 있다. 이는 다른 사용자에게 경로를 추천할 때 사용될 수 있다. 사용자가 기입한 정보와 이동 경로는 데이터베이스에 저장된다. 그 외에도 음성 검색, 실시간 대중교통 승강기 정비 현황 을 제공한다.

1.3 Definitions, Acronyms, and Abbreviation

<Table 1>에서는 이 문서에 사용할 용어를 정의한다.

교통약자	생활 차원에서 이동에 어려움을 겪는 사람들
User	서비스를 제공받는 사람으로 교통약자가 해당됨
User Interface	User 와 시스템 사이의 소통을 위해 제공되는 가상 매개체
트래킹	User 의 이동을 따라가면서 기록하는 것
Back-End	Server 나 DB 관련된 코드로 User 와 직접적으로 관련이 되어있지 않은 부분
Front-End	User 에게 직접 보이는 User Interface 와 관련된 부분

<Table 1> 용어와 정의

<Table 2>에서는 이 문서에 사용할 약어를 정의한다.

GPS	Global Positioning System
API	Application Programming Interface
DB	Database
ID	Industrial Design
PW	Password
UI	User Interface
HTTP	Hypertext Transfer Protocol
RAM	Random Access Memory
CPU	Central Processing Unit
JWT	JSON Web Tokens

<Table 2> 축약어 목록

1.4 References

“Study on requirements and architecture for enhancing pedestrian mobility”. 2017 International Conference on Information and Communication Technology Convergence (ICTC)

1.5 Overview

이 문서는 **Mobility Mate** 출시 전 이루어지는 테스트 단계에서 사용된다. 모듈 단위의 유닛테스트와 인터페이스 테스트 중심으로 어떻게 테스트 계획이 이루어지는지 의사코드 및 사용 툴에 대한 설명으로 이루어져있다.

2. Approach

2.1 Test method

시스템 표준 시험 절차에는 Unit test, API testing, Integration test, System test, Install/Uninstall testing, Agile testing 등이 있다. 본 테스트에서는 Unit test 와 Interface test 을 높은 우선순위로 두어 시행하고자 한다.

2.1.1 Software unit test methods

각 유닛 및 인터페이스에 테스트용 의사 코드를 입력하여 S / W 가 올바르게 수행되었는지 확인한다.

A. Register - Unit test

- **SetProfile(), ChangeProfile(), ShowProfile()**

Test case set as T < (ID, 휴대폰 번호, 출생년도, 이동 보조기구, 임신여부) >

SetProfile(), ChangeProfile(), ShowProfile() as U

Success message as S

Error message as E

U(T)

If result is true then

S

else

E

B. 검색 - Unit test

- **UserInput()**

UserInput() as U

Test case as T (building name or location name)

Success message as S

Error message as E

U(T)

If result is true then

S

else

E

- **GetInfo(), GetCurLoc(), GetUserTouch()**

GetInfo() as U

Success message as S

Error message as E

U()

If result is true then

S

else

E

C. 경로찾기 - Unit test

- **GetUserPerformance(), GetTransportation(), GetSlope(), GetTime()**

Test case set as T < (출발지, 목적지, Route) >

GetUserPerformance() as U

Error message as E

RequestResult(T)

if result is true then

GetTransportation()

GetSlope()

GetTime()

return Route

else

E

D. Quick access - Unit test

- **GetFacilityInfo(), GetPTEI()**

GetCurLoc()

Test case as T (random location)

Error case E

GetFacilityInfo(T)

if result is true then

 GetPTEI()

else

 E

2.1.2 Software interface test methods

A. Login Server - Connection Test

Test case set $T < (ID, Password), JWT >$

Success case S

Case requiring verification V

Error case E

For each case C in T do

Result1 = Check JWT

If Result1 then

Already have a JWT

S U C

```
Else if !Result1 then
```

Result2 = Verify User Info by ID, Password

If Result2 then

Issue a JWT = { ID : ID ,

loggedTime : String,

이동수행능력 : Number }

S U C

```
Else if !Result2 then
```

VUC

Else

EUC

Else

EUC

B. Service Server - Connection Test

Test case set $T < \text{JWT, verified user}(T/F) >$

Success case S

Case requiring verification V

Error case E

For each case C in T do

 status_code = HTTP 통신 결과

 If status_code == 200 OK then

 Result1 = Decode the JWT

 If Result1 then

 User id, issue time = Check the JWT

 Result2 = IsValidJWT(User id, issue time)

 If Result2 then

 Return True

 S U C

 Else if !Result2 then

 Return False

 S U C

 Else then

 V U C

 Else if !Result1 then

 V U C

 Else then

 E U C

 Else then

 E U C

C. 공사현장 API - Connection Test

Test case T < Road name, Construction progress >

Success case S

Error case E

For each case C in T do

status_code = HTTP 통신 결과

If status_code == 200 OK then

Result = 건설 알람이 사이트 조회(Road name)

If Result then

Return Result = { hasConstruction: True,

sidewalkAddress: String,

constructionAddress : String,

constructionLocation : String }

S U C

Else if ! Result then

Return Result = { hasConstruction: False,

sidewalkAddress: String,

constructionAddress : String,

constructionLocation : String }

S U C

Else

E U C

Else then

E U C

D. MongoDB - Connection Test

Test case T < (Telnet, Port), Port status(T/F) >

Success case S

Error case E

For each case C in T do

Result = telnet host port

If Result then

Return T

S U C

Else if ! Result then

Return F

S U C

Else

E U C

3. Software Unit Test

Android studio 라이브러리인 junit 을 활용해 unit test 를 진행한다.

3.1 Register

3.1.1. Account register test case

- A. Test case object: 유효한 데이터를 입력했을 때 Register 기능이 작동하는지 확인한다.
- B. Test Inputs
 - a. ID: 한글 최소 4 글자 이상
 - b. 휴대폰 번호: 숫자 11 자
 - c. 출생년도: 숫자 6 자
 - d. 이동 보조기구: 수동 휠체어, 전동 휠체어, 흰지팡이, 목발 및 지팡이 중 택 1
 - e. 임신여부: 예, 아니오 중 택 1
- C. Expected Results: '등록 완료' 메시지를 보여준다.

3.1.2. account register test case – Negative

- A. Test case object: 유효하지 않은 데이터를 입력했을 때 Register 기능이 잘 작동하지 않는지 확인한다.
- B. Test Inputs
 - a. ID: 한글 3 글자 이하
 - b. 휴대폰 번호: 한글, 숫자 11 자가 아닌 입력
 - c. 출생년도: 한글, 숫자 6 자가 아닌 입력
 - d. 이동 보조기구: 수동 휠체어, 전동 휠체어, 흰지팡이, 목발 및 지팡이 중 택하지 않았을 경우
 - e. 임신여부: 예, 아니오 중 택하지 않았을 경우
- C. Expected Results
 - a. ID: 최소 4 글자 이상 입력하시오
 - b. 휴대폰 번호: 숫자 11 자를 입력하시오
 - c. 출생년도: 숫자 6 자를 입력하시오
 - d. 이동 보조기구: 이동 보조기구를 선택하시오

- e. 임신여부: 임신여부를 선택하시오

3.2 검색

3.2.1 검색

- A. Test case object: 알맞은 데이터를 입력했을 때 검색 기능이 장소에 대한 정보와 주소를 바르게 제공하는지 확인한다.
- B. Test Inputs: 장소, 주소
- C. Expected Results: 알맞은 장소를 지도에 표시하고 장소에 대한 정보 제공

3.2.2 검색 - negative

- A. Test case object: 맞지 않은 데이터를 입력했을 때 검색 기능이 제대로 된 검색 입력을 요구하는지 확인한다.
- B. Test Inputs: 장소, 주소
- C. Expected Results: 잘못된 장소 정보 제공 및 표시하거나 빈 화면을 표시한다.

3.3 경로 찾기

3.3.1 경로 찾기

- A. Test case object : 알맞은 데이터를 입력했을 때 경로 찾기 기능이 바르게 작동하는지 확인한다.
- B. Test Inputs
 - a. 출발지
 - b. 목적지
 - c. 옵션: 프로필에 있는 이동 수행능력
- C. Expected Results : 사용자의 이동 수행능력에 맞는 경로
- D. 절차
 - a. 출발지, 목적지에 맞는 경로 확인
 - b. 이동 수행 능력 있는지 확인 후 전달

- c. 이동 수행 능력에 맞는 교통수단, 경사도, 소요시간 확인
- d. 추천 경로 출력

3.3.2 경로 찾기 - negative

- A. Test case object : 맞지 않은 데이터를 입력했을 때 경로 찾기 기능이 작동하지 않는지 확인한다.
- B. Test Inputs : 출발지와 같은 목적지
- C. Expected Results : 잘못된 경로, 입력 데이터 요구
- D. 절차
 - a. 출발지와 목적지 비교
 - b. 같다면 입력 정보를 다시 요구한다.

3.4. Quick Access

3.4.1. Quick Access

- A. Test case object : 다양한 지역에서 테스트 했을 때 설정한 반경 내에서 주변 시설 정보, 승강기 정비 현황 정보를 제공하는지 확인한다.
- B. Test Inputs : gps 에 받은 위치 정보
- C. Expected Results : 설정한 반경 내 장애인 시설 정보, 승강기 정비 현황

3.4.2. Quick Access - negative

- A. Test case object : 맞지 않은 데이터를 입력했을 때 설정한 반경 내에서 주변 시설 정보, 승강기 정비 현황 정보를 제공하는지 확인한다.
- B. Test Inputs : 현 위치와 다른 gps 정보
- C. Expected Results : 현 위치와 다른 gps 상의 위치에 맞는 주변 시설 정보와 승강기 정비 현황을 제공한다.

4. Software Interface test

외부 API 와 서버 연결 여부는 REST API 방식으로 Postman 툴을 활용하여 연결여부를 확인한다.

4.1. Login Server

4.1.1 Connection test

- A. Test case object : 어플에 접속할 때 유효한 유저인지 확인
- B. Test Input :
 - a. ID
 - b. Password
- C. Expected Results : JWT 발급
 - a. Result = {

ID : ID,

loggedTime : String,

이동수행능력 : Number

}
- D. 절차:
 - a. JWT 가지고 있는지 확인
 - b. 가지고 있으면 종료, 없으면 ID, PW 입력
 - c. ID,PW 기반 로그인 서버에서 유저 정보 확인
 - d. 기존에 등록된 유저로 확인 시 신규 JWT 발급(ID, 발행시간, 이동수행 능력 등 다른 서비스를 활용하기 위해 필요한 유저 정보가 포함된 토큰)

4.2 Service Server

4.2.1 Connection test

- A. Test case object : 로그인/회원가입 서버에서 받은 JWT 토큰을 활용한 서비스 서버로의 접근이 유효한지 확인
- B. Test Input :
 - a. JWT
- C. Expected Results :
 - a. HTTP 통신 status code 200 OK
 - b. Log 에 “IsValidJWT : True” 반환
- D. 절차 :
 - a. 유저가 가지고 있는 JWT 를 확인
 - b. JWT 디코딩
 - c. JWT 정보 확인(유저 아이디, 발행 시간)
 - d. 발행 시간 과 유저 정보를 토대로 유효한 토큰인지 확인(예를 들면 IsValidJWT(ID,발행시간))
 - e. 유효한 JWT 이면 다른 서비스들 이용 가능, 아니면 로그아웃

4.3. 공사 현장 API

4.3.1 Connection test

- A. Test case object : 건설알림이 사이트로부터 경로 상에 진행중인 공사 현장 정보를 가져오는 API 가
- B. Test Input : 도보 정보
- C. Expected Results : JSON 형식의 공사 진행 여부
 - a. HTTP 통신 Status Code : 200 OK
 - b. Result = {

 hasConstruction: True,

 sidewalkAddress: String,

 constructionAddress : String,

 constructionLocation : String

 }
- D. 절차 :
 - a. 이동 가능한 다음 도보의 정보 입력(hasConstructionThisSidewalk(도보 정보))
 - b. 입력한 도보 정보를 기준으로 건설 알림이 사이트에서 조회
 - c. 조회된 결과에 따라 진행 여부 및 공사 정보 반환

4.4 MongoDB

4.4.1 Connection test

- A. Test case object : 각 마이크로서비스마다 가지고 있는 MongoDB 와의 연결을 확인
- B. Test Input : CMD 에 telnet 명령어와 port 정보 입력
- C. Expected Results : 해당 port 정보
- D. 절차 :
 - a. CMD 에 telnet 활용
 - b. 해당 DB 의 포트가 열려 있는지 확인

5. Supporting Information

5.1 Document History

Date	Version	Description	Writer
May 24	0.1	Title and Content	안리아
May 26	0.2	addition of 1.1, 1.2, 1.3, 1.4, 2.1	안리아
May 27	0.3	addition of 3.1, 3.2, 3.3, 3.4	우준하
May 27	0.4	addition of 4.1, 4.2	이석현
May 28	0.5	addition of 4.3 4.4	이석현
May 29	0.6	addition of 2.1.2	이주민
May 29	0.7	addition of 2.1.1	우준하
May 29	0.8	addition of 2.1.1, 1.5	안리아
May 30	0.9	addition of 4.1 4.2 4.3 4.4	이석현
May 30	1.0	addition of 2.1.2	이주민
May 30	1.1	검수	이석현

<Table 3> Document History