

# Docking Requirement Specification

Team 4

2014311773 김용태

2014310444 김은하

위함흔

2016314216 이상아

2009310261 임재현

## ***Table of Contents***

<b>Preface</b>	<b>6</b>
1.1 Objective	6
1.2 Readership	6
User Requirement Readership	6
System Requirement Readership	6
1.3 Document Structure	7
Preface	7
Introduction	7
Glossary	7
User Requirements Specification	7
System Architecture	8
System requirements Specification	8
System models	8
System Evolution	8
Appendices	8
1.4 Version of the Document	8
<b>Introduction</b>	<b>10</b>
2.1 Objective	10
2.2 Needs	10
2.3 DOCKING!	13
2.4 Expected Effect	13
<b>Glossary</b>	<b>15</b>
3.1 Objective	15
3.2 Term definitions	15
사용자 관련 용어	15
서비스 관련 용어	15
개발 관련 용어	16
<b>User Requirement Definition</b>	<b>18</b>
4.1 Objective	18

4.2 Functional Requirements	18
Sign In/Out	18
Log In/Out	18
My Page	18
Upload Pictures & Videos	19
Tag	19
Follow	19
Searching	19
Showing Main Page	19
Walking-Mate Matching	20
Walking-Information Page	20
Dogwatch Peristalsis	20
4.3 Non-Functional Requirements	20
Product Requirements	20
Organization Requirement	21
External Requirement	21
<b>System Architecture</b>	22
5.1 Objective	22
5.2 Overall Architecture	23
5.3 User Interface Structure	24
5.4 Sub-system Architecture	25
산책 System	25
커뮤니티 System	26
User System	27
<b>System Requirement Specification</b>	28
6.1 Objective	28
6.2 Functional requirements	28
Sign In/Out	28
Log In/Out	29
My Page	30
Upload Pictures & Videos	31

Tag	32
Follow	33
Searching	33
Showing Main Page	34
Walking-Mate Matching	35
Walking-Information Page	36
Dogwatch Peristalsis	36
6.3 Non-Functional Requirements	37
Product Requirements	37
Organization Requirements	38
External Requirements	38
6.4 Scenario	39
Walking Mate Matching Scenario	39
Walking Information Page Scenario	39
Community Main Page Scenario	40
Uploading Scenario	40
Sign in/out Scenario	41
Sign in Scenario	41
Initial Assumption	41
Normal Flow of Events	41
What can go wrong & concurrent Activities	42
System State on Completion	42
Sign out Scenario	42
Initial Assumption	42
Normal Flow of Events	42
What can go wrong & concurrent Activities	42
System State on Completion	42
Login /out Scenario	43
Login Scenario	43
Initial Assumption	43
Normal Flow of Events	43

What can go wrong & concurrent Activities	43
System State on Completion	43
Logout Scenario	43
Initial Assumption	43
Normal Flow of events	43
What can go Wrong & concurrent Activities	43
System State on Completion	43
Mypage Scenario	44
Initial Assumption	44
Normal Flow of Events	44
What can go wrong & concurrent Activities	44
System State on Completion	44
<b>System models</b>	45
7.1 Objective	45
7.2 Context Models	45
Context Model	45
Process Diagram	46
7.3 Interaction Models	49
Use case Diagram	49
Sequence Diagram	50
7.4 Structural Model	54
7.5 Behavioral Models	55
Data-Driven Modeling	55
Event-Driven Modeling	57
State model of Login system	57
<그림 3.15 State Model of Login system>	57
<b>System evolutions</b>	58
8.1 Objective	58
8.2 Limitation and Assumption	58
8.3 Evolutions of Hardware	59
스마트밴드	59

8.4 Evolutions of User Requirement	59
반려인의 건강까지 고려하는 시스템	59
커뮤니티 기능의 확장, 주기적인 산책 모임의 결성	60
매칭 시스템 상에서의 필터링 기능의 증진 및 악용 방지	60
8.4 Evolutions of Environment	61
반려견 외의 다른 반려동물들에 대한 지원	61
<b>Appendices</b>	61
9.1 Objectives	61
9.2 Database Requirements	62
User DB	62
Contents DB	62
Posts	62
Walking	62
Dog DB	62
<b>10 Index</b>	63
10.1 Table index	63
10.2 Figure, Diagram index	63

# 1. Preface

## 1.1 Objective

Preface에서는 본 요구사항 명세서의 전체적인 개요에 대해 설명한다. 기반으로 하는 문서 형식, 예상되는 독자층, 명세서의 구조로 이루어져 있다. 독자층에서는 각 요구사항이 대상으로 하는 독자층과, 그에 따라 어떤 내용을 담고 있는지에 대한 설명을 한다. 그 다음 본 문서의 전체적인 구조에 대해 간략한 소개를 한다.

## 1.2 Readership

본 요구사항 명세서에서의 명세는 두 가지로 구분된다. 사용자 요구사항과 시스템 요구사항으로 구분되며, 내용과 예상 독자는 다음과 같다.

### A. User Requirement Readership

사용자 요구사항에서는 제공해야 할 시스템과 준수해야 하는 시스템 제약 사항에 관해 개괄적으로 기록한다. 예상되는 주요 독자층으로는 'Docking' 이용자들, 제휴업체, 마케팅 담당자 등 비전문가들이다. 즉, 요구사항을 추상적으로 간략히 명세한 것이다.

- 이용자 : 'Docking'의 커뮤니티에서 반려인들, 반려동물 전문가들과 함께 반려동물에 대한 정보를 공유하고, 산책 메이트 매칭 시스템을 이용하고자 하는 사람들
- 제휴업체 : 'Docking'과 제휴하여 반려동물 용품을 홍보하고 판매하고자 하는 업체들
- 마케팅 담당자 : 'Docking'의 특징을 파악하여 서비스를 홍보하고자 하는 사람
- 

### B. System Requirement Readership

System requirements에서는 본 프로젝트에서 개발해야 하는 시스템의 특정 기능이 어떻게 구현되어야 하는지에 대해 자세하게 설명되어 있다. System requirements의 주된 독자는 해당 Software developer들이다. Software developer 외에도 Client engineers, System architects, 경우에 따라서는 System end-users까지 주요 독자가 될 수 있다. 따라서 개발의 구체적 방향을 잡아 개발의 완성도를 높이고 비용을 줄이고자 하는 독자들은 본 문서를 순서대로 읽어야 한다.

## 1.3 Document Structure

이 문서는 9개의 장으로 구성되어 있다. 9개의 장은 Preface, Introduction, Glossary, User Requirements Specification, System Architecture, System requirements Specification, System models, System Evolution, Appendices이다. 각 장의 역할과 내용은 아래와 같다

### A. Preface

Preface에서는 본문서의 예상되는 독자들과 문서의 전반적인 구조, 그리고 각 부분의 역할과 내용에 대하여 제시한다. 버전 관리 정책을 제시하고 이에 따른 버전 변경을 표로 기록하며 추가적인 문서 변경사항과 근거들을 서술한다.

### B. Introduction

Introduction에서는 목표 시스템의 필요성과 해당 시스템이 어떠한 니즈를 반영하고 있는가에 대해 서술한다. 목표 시스템의 주요 기능과 목표 시스템이 기존에 존재하는 다른 외부 시스템과 어떻게 상호작용 하는지에 대하여 간략하게 설명한다. 또한, 목표 시스템이 어떠한 과정을 통하여 시스템의 목적을 성취하는지에 대하여 설명한다.

### C. Glossary

Glossary에서는 문서에 등장하는 기술적인 용어들에 대해 정의한다. 전문 배경지식이 없는 독자가 읽더라도 본 문서를 이해할 수 있도록, 독자의 경험이나 전문지식 등이 어떤지를 추정해서는 안 되고, 가능한 한 모든 용어에 대해 설명해야 한다.

### D. User Requirements Specification

User Requirements Definition에서는 실제로 사용자에게 제공되는 서비스에 대해서 설명한다. 목표 스템이 가지고 있는 비 기능적인 요구사항도 서술한다. 자연어와 함께, 이해를 돕기 위한 다이어그램과 같은 시각적인 방법들을 사용하여 설명한다. 목표 시스템은 이 장에 서술되어 있는 모든 비 기능적인 요구사항들을 충족할 것이다.



## E. System Architecture

System Architecture에서는 목표 시스템에 대한 높은 수준의 개요를 보여준다. 시스템 기능의 전체적인 분포도 함께 보여준다. 이 과정에서 재사용되는 컴포넌트(Component)는 강조하여 표현하였다.

## F. System requirements Specification

System Requirements Specification에서는 기능적 요구 사항과 비기능적 요구 사항을 더 자세하게 설명한다. 요구되는 대략적인 비기능적 요구 사항은 추가적으로 세분화된 다수의 기능적 요구 사항의 형태로 더 자세하게 설명한다. 다른 외부 시스템과의 상호작용에 대한 인터페이스(Interface)가 정의된다.

## G. System models

System Models에서는 목표 시스템의 컴포넌트와 목표 시스템의 내외적 환경과의 관계를 다이어그램 형태로 표현한다.

## H. System Evolution

System Evolution에서는 목표 시스템이 취하고 있는 기본적인 가정들에 대하여 설명한다. 향후 목표 시스템에 대하여 발생할 수 있는 예상 가능한 모든 변경 사항에 대해서도 설명한다. 하드웨어(Hardware)의 발전, 사용자 요구 사항의 변화 등을 설명한다. 이장의 내용은 시스템 설계자에게 유용할 것이다. 이장을 통하여 향후 예측되는 변화에 잘 대처하는 시스템으로의 개발을 기대할 수 있다.

## I. Appendices

Appendices에서는 개발할 목표 시스템에 대한 더 구체적인 정보들을 제공한다. 목표 시스템에 사용될 하드웨어, 데이터베이스(Database;DB) 등을 설명한다. 목표 시스템 사용에 필요한 최소의 설정과 사용에 적합한 설정, 데이터베이스 상에서 결정되는 데이터(Data)의 관계 및 구조에 대하여 설명한다.

## 1.4 Version of the Document

Version	Modified Date	Explanation
0.1	2019.05.02	<ul style="list-style-type: none"> <li>- 요구사항 명세서 항목에 대한 목차 작성.</li> <li>- Preface 작성.</li> </ul>

		- Introduction 작성.
0.2	2019.05.03	- User Requirements Specification 작성. - System Architecture 작성
0.5	2019.05.04	- System Requirements Specification 작성. - System Models 작성. - System Evolution 작성.
1.0	2019.05.05	- Appendices 작성. - System Requirements Specification 수정 - System Models 수정 - 전체 양식 통일
2.0	2019.??.??	- 문서 보완.

〈Table 1.1 Version of the Document〉

## 2.Introduction

### 2.1 Objective

이 part에서는 Docking의 제안 배경과 어떠한 요구사항을 반영하고 있는지를 서술한다. 또한, Docking 시스템의 기능이 다른 기능과 어떻게 작용하는지 개략적으로 설명한다. 또한 Docking의 시스템이 어떻게 목적을 달성하는지 설명한다.

### 2.2 Needs

농림축산식품부가 2016년 실시한 조사에 따르면, 국내 등록 반려동물은 97만 마리에 이르렀다. 미등록 반려동물을 함께 고려한다면 150만 마리를 훌쩍 웃돌 것으로 예상된다. 또한 KB금융지주경영연구소가 진행한 '2017 반려동물 양육 실태 조사'에 따르면, 국내에서 반려동물을 기르는 가구는 전체 가구의 30.9%로 약 590만 가구로 추정된다. 또, 한국에서 반려동물을 길러본 가구는 3가구 중 2가구나 되며, 반려동물 중 개가 82%로 압도적인 수치를 차지한다.<sup>1</sup> 최근 이어지는 저출산, 고령화, 1인 가구 확대로 아이 대신 반려동물을 키우며 반려견·반려묘에 시간을 들이고 비용을 투자하는 소비자는 점차 늘어날 전망이다. 펫 시장 성장세도 이어질 것으로 예상된다. 국내 반려동물 산업은 2015년 1조9,000억원 규모에서 2017년 2조3,300억원으로 22.6% 확대될 것으로 예상되며, 2027년에는 6조원 규모까지 확대될 것으로 전망되는 성장세 높은 산업으로 꼽힌다.

---

<sup>1</sup>명순영 외 1명, '1인가구 시대 '펫코노미' 대세 '상팔자' 반려동물 시장 6조원', 매일경제, 2018.03.16.

<https://www.mk.co.kr/news/economy/view/2018/03/170900/>

펫코노미와 함께 부상한 반려동물 관련 신조어			
펫팸족	펫밀리	딩펫족	펫세프
반려동물을 가족처럼 여기고, 반려동물에 대한 투자를 아끼지 않는 사람들	반려동물과 가족의 영단어를 조합한 신조어 펫 밀 리 는 반려동물과 함께 살아가는 1인 가구 동을 지칭하는 신조어	'펫'과 'Double Income No Kids'의 줄임말 '딩크'의 합성어 아 이 없 이 반려동물을 키우며 사는 맞벌이 부부를 일컫는 단어로 재탄생	반려동물의 입맛과 건강을 고려하여 고급 수제 사료와 간식을 챙겨주는 이들을 뜻함

Source: 언론 기사 종합, 삼정KPMG 경제연구원 재정리

<그림1.1 펫코노미와 함께 부상한 반려동물 관련 신조어>

한국 정부는 지난 2016년 '투자활성화 대책' 중 하나로 '반려동물 산업'을 신산업으로 육성하고 활성화 하겠다고 밝혔다. 반려동물 관련 산업을 지속가능한 산업으로 성장을 유도하고자 한국 정부는 반려동물 산업 육성의 반열에 합류한 것이다. 농림축산식품부에서 2016년 12월, 개·고양이 등 반려동물의 보호수준을 높이는 동시에 관련 산업의 건강한 육성을 목적으로 '반려동물 보호 및 관련산업 육성 세부대책'을 발표하며 관련 내용을 후속적으로 추진 중이다. 농림축산식품부는 '사람과 반려동물의 조화로운 공존'이라는 비전을 설정하고 2020년까지 등록 동물 150만 마리, 반려동물 관련 산업의 시장 규모를 3조5,000억원으로 목표했다. 정부의 반려동물 산업 육성 정책의 중점적 세부 추진 과제는 크게 네 가지 분야로 구분된다. 첫째 생산 및 판매업 관리·감독 강화, 둘째 반려동물 관련 산업의 건강한 육성, 셋째 성숙한 반려동물 문화 정착, 넷째 산업육성 인프라 구축 및 일자리 창출이다.

이의 전체적인 비전은 **사람과 반려동물의 조화로운 공존**에 있다.<sup>2</sup>

반려견 훈련사로 잘 알려져 있는 강형욱은 반려견을 키울 때, 산책의 중요성을 역설한다. '반려견들은 걷고 싶어합니다. 그것도 자신이 제일 믿고 좋아하는 누군가와 같이 같은 방향을 바라보며 걷고 싶어합니다.'<sup>3</sup> 또 "독일은 1일 1회, 스위스는 1일 2회 산책이

<sup>2</sup> 삼정KPMG 경제연구원, 「펫코노미 시대, 펫 비즈니스 트렌드」, 삼정 KPMG, 제 93호, 2018년 10월.  
<https://assets.kpmg/content/dam/kpmg/kr/pdf/kr-im93-pet-business-trends.pdf>

<sup>3</sup> 강형욱, '강아지 산책하기 같이 강아지와 산책하며 땀흘리기..', 보듬, 2015.03.01.  
[http://www.bodeum.co.kr/html/edu\\_movie/community/community\\_list.php?mode=v&bbs\\_data=aWR4PTE1NjEmUGFnZU51bWJlcj0xJmxc3RObz0xOSZ0YWJsZT1pZF9iYnNfZGF0YSZjb2RlPWNvbW11bml0eV9jb2x1bW4mc2VhcmNoX2l0ZW09MSZzZWZyY2hfb3JkZXI97IKw7LGF%7C%7C](http://www.bodeum.co.kr/html/edu_movie/community/community_list.php?mode=v&bbs_data=aWR4PTE1NjEmUGFnZU51bWJlcj0xJmxc3RObz0xOSZ0YWJsZT1pZF9iYnNfZGF0YSZjb2RlPWNvbW11bml0eV9jb2x1bW4mc2VhcmNoX2l0ZW09MSZzZWZyY2hfb3JkZXI97IKw7LGF%7C%7C)

의무화”라며 “책임질 수 있는 사람만 반려견을 키우라”고 강조하기도 하였다.<sup>4</sup> 이를 위해 반려견을 대신 산책해주는 서비스, 즉 도그워킹 업체도 최근 한국에선 늘어나는 추세이지만<sup>5</sup>, 자신의 반려견을 위해 가장 좋은 것은 역시 반려인인 스스로가 반려인과 함께 시간을 내어 산책하는 것이다. 산책은 또, 반려견의 사회화에도 도움이 된다. 반려인뿐만 아닌, 야외에서 만나는 같은 강아지 친구들과 반려견들끼리의 커뮤니티를 형성하며, 마치 사람처럼, 사회화가 될 수 있다.

그런데 사람으로서는 혼자서 매일 몇 시간씩 산책을 하는 것이 쉬운 일은 아니다. 매일매일 산책을 하기 위한 체력이나 동기가 늘 높은 상태로 유지되는 것도 아니다. 늘 새로운 것에 호기심을 가지는 강아지와 달리, 사람은 회사 혹은 학교에서 지친 상태로 돌아온 후, 별 것 없는 길을 지루하게 혼자 다닌다고 생각할 수도 있다. 또, 그저 강아지를 공원 등지에 데려가 풀어놓을 수만도 없다. 반려견과 산책하기 위해서는 여러 가지, 반려견을 위해 지켜야 할 규칙들이 필요하다.

그리하여 반려인과 반려동물, 더 나아가 비-반려인과 반려인 및 반려동물이 공존하며 즐거운 산책 문화를 만들기 위한 방법으로 반려견 행동전문가 강형욱은 산책모임을 적극적으로 권장한다.



〈그림 1.2 산책모임〉

그가 ‘세상에 나쁜 개는 없다’, ‘보듬’ 등에서 산책모임을 열성적으로 홍보함에 따라 산책모임에 대한 수요는 늘어나고 있는 상황이다. 그러나 기존의 지역 애견모임은

<sup>4</sup> 한경석, “어쩌다 어른’ 강형욱 “독일은 1일 1회, 스위스는 1일 2회 개 산책 의무화”, 비즈엔터, 2018.02.15.

[http://enter.etoday.co.kr/view/news\\_view.php?varAtclId=135918](http://enter.etoday.co.kr/view/news_view.php?varAtclId=135918)

<sup>5</sup> 김성일, ‘내가 할 수 없다면? 반려견 산책서비스 이용해보세요’, 올치올치, 2016.07.21.

<https://www.olchiolchi.com/%EB%82%B4%EA%B0%80-%ED%95%A0-%EC%88%98-%EC%97%86%EB%8B%A4%EB%A9%B4-%EB%B0%98%EB%A0%A4%EA%B2%AC-%EC%82%B0%EC%B1%85%EC%84%9C%EB%B9%84%EC%8A%A4-%EC%9D%B4%EC%9A%A9-%ED%95%B4%EB%B3%B4%EC%84%B8%EC%9A%94/>

폐쇄적으로 운영되거나, 지역 애견모임이 아예 존재하지 않는 경우도 있으며, 애견모임에서의 산책모임 matching은 사용자간 구두에 의한 약속으로 이루어진다는 한계점을 지니고 있다. 따라서 애견모임 그 자체와는 독립적이며, online 상으로, 또한 선택에 따라서는 자동으로 산책 모임 matching을 하는 서비스에 대한 수요가 예상된다.

## 2.3 DOCKING!

Docking이 제공하는 서비스는 크게 세가지로 나뉜다.

- A. 반려견, 반려견에 관심이 있는 사람 및 반려견 전문가가 참여할 수 있는 커뮤니티.
  - 커뮤니티 서비스는 일반 반려인들끼리의 정보, 게시물을 공유할 수 있는 게시판과 반려견 전문가의 신빙성 있는 정보를 올릴 수 있는 게시판으로 나뉘어져 있다. 모든 사용자는 게시글을 열람할 수 있지만 전문가 게시판에는 사전에 허가받은 반려견 전문가만이 글을 게시, 수정 할수 있다.
- B. 반려견 산책모임 matching 서비스.
  - 산책메이트 matching 서비스는 반려인들의 산책모임 구성을 도우며, 반려인들에게 산책의 동기부여를 줄 수 있다. 단순 구두로 약속하는 모임이 아닌 가까운 지역의 산책메이트 수요를 분석, 제공하여 효율적이고 확실한 matching 이 가능하도록 한다.
- C. 반려견 촬영, 반려견 운동기록등 반려견 산책 관련 서비스.
  - 반려동물의 동영상 또는 사진에 인식 효과를 적용하여 아기자기하고 재미있는 영상 공유
  - 산책 시 주의해야 할 점, 산책하기 좋은 날 등을 알려주고 반려동물의 카밍 시그널에 대한 정보 제공.
  - 목줄에 스마트밴드 등을 적용시켜 반려동물의 상태를 알려주는 하드웨어와 어플리케이션 연동

## 2.4 Expected Effect

- A. 반려견 보호자
  - 반려견 산책모임을 matching 받을 수 있으며, 반려견의 활동을 기록, 열람 할 수 있다. 반려견 양육을 위해 필요한 지식을 얻을 수 있다.
- B. 반려견에 관심이 있는 사람들.
  - 반려견입양 여부 결정에 필요한 지식을 얻을 수 있으며, 반려견 입양 전 알아야 할 지식을 얻을 수 있다.

C.반려견 전문가

- 전문가는 커뮤니티 활동을 통하여 간접적으로 인지도를 올릴 수 있다. 또한 직접적인 광고를 통하여 홍보효과를 얻을 수 있다.

## 3. Glossary

### 3.1 Objective

Glossary에서는 문서에 등장하는 기술적인 용어들에 대해 정의한다. 배경지식이 없는 독자가 읽더라도 문서를 이해할 수 있도록 독자의 경험이나 전문지식 등이 어떤지를 추정해서는 안되고, 가능한 한 모든 용어에 대해 서술한다.

### 3.2 Term definitions

#### A. 사용자 관련 용어

Terms	Definition
반려견	한 명 이상의 사람과 함께 살아가는 중인 개를 말한다.
사용자(반려인)	반려견을 키우고 있으며, 서비스를 이용하는 사용자를 말한다
사용자(비반려인)	반려견을 키우고 있지 않으며, 서비스를 이용하는 사용자를 말한다.
사용자(전문가)	서비스를 이용하는 사용자들 중, 반려견에 대한 전문 지식을 가지고 있으며 이를 증명할 자격증 등을 보유한 사람들을 말한다.
관리자	본 서비스를 운영하는 관리자로, 서버 운영 및 유지보수를 담당

<Table 2.1 사용자 관련 용어>

#### B. 서비스 관련 용어

Terms	Definition
Sign in	사용자가 처음으로 도킹 서비스를 사용하기 위해 자신의 정보를 서비스에 기입하고 등록하는 절차를 말한다.
Login	사용자가 서비스를 사용하기 위해 기존에 sign in 한 정보와 본인 정보를 비교하는 절차를 말한다,



my page	사용자 본인이 가입 시에 입력한 스스로의 정보를 보여 주며, 원하는 정보를 추가적으로 기입할 수 있다. 비반려인 사용자의 경우 본인의 정보만을 등록할 수 있다. 반려인 사용자의 경우 반려견에 대한 정보를 등록할 수 있다. 전문가 사용자의 경우 자신이 전문가임을 증명하는 자료 등을 등록할 수 있다.
upload	사용자가 커뮤니티에 사진, 영상, 글 등을 올릴 수 있도록 하는 것을 말한다
follow	사용자가 자신 이외의 다른 사용자 계정을 구독 목록에 등록하는 것을 말한다
searching	사용자가 커뮤니티에서 반려견 정보 등을 검색할 수 있도록 하는 것을 말한다
community	사용자들이 자신의 산책과 관련한 게시물을 업로드 할 수 있도록 한 서비스를 말한다.
matching	둘 이상의 반려인들이 함께 산책할 수 있도록, 알고리즘에 따라 반려인 사용자들을 골라 붙여주는 서비스를 말한다.
peristalsis	반려견이 착용한 스마트밴드와 반려인의 스마트폰을 연동하는 것을 말한다.

&lt;Table 2.2 서비스 관련 용어&gt;

### C. 개발 관련 용어

Terms	Definition
서버(Server)	서비스에 응답하는 프로세스와 하드웨어를 통괄하는 용어.
데이터베이스(DB)	서버에서 서비스를 제공하기 위해 정보들을 저장하고 있는 저장소.
프레임워크(Framework)	프로그램을 개발, 사용하기 위해 제공되는 기본적인 틀.
깃허브(Gitgub)	팀 개발을 하는 데 있어 코드 공유 등 편리성을 제공하는 서비스.
응용 프로그램 인터페이스 (API)	시스템 자원 혹은 다른 서비스에 접속하기 위해 제공되는 인터페이스를 의미한다.

모바일 앱 (Mobile App)	스마트폰, 태블릿 PC 등 모바일 장치에서 실행되는 응용 소프트웨어.
-----------------------	--

<Table 2.3 개발 관련 용어>

## 4. User Requirement Definition

### 4.1 Objective

해당 장에서는 본 시스템이 필수적으로 제공해야 하는 서비스에 대하여 기술한다. 또한 제품이나 시스템 프로세스가 따라야 할 제약 사항에 대해서도 서술한다. Functional Requirements와, Non-Functional Requirements에 대하여 상세히 기술한다.

### 4.2 Functional Requirements

‘Docking’은 다음과 같은 Functional Requirements를 충족해야 한다.

#### A. Sign In/Out

사용자가 서비스를 이용하는 데에 필요한 개인 정보를 제공받아 회원으로 등록하는 기능이다. 회원 가입을 통해 로그인 절차를 거쳐야만 사용자는 ‘Docking’에서 제공하는 서비스를 이용할 수 있다. ‘Docking’을 사용하는 사용자는 일반 반려인, 비반려인과 반려인 전문가가 있을 수 있고, 가입 시 구분 절차를 거친다. 반려인의 경우 반려동물에 대한 정보 역시 입력한다. 또한 마이 페이지에서 탈퇴를 선택하면 회원 등록 취소와 개인 정보 삭제가 이루어진다.

#### B. Log In/Out

사용자가 회원 가입을 통해 생성된 아이디를 이용하여 시스템에 로그인하는 기능이다. 로그인 후 사용자는 자신의 데이터베이스에 접근할 수 있으며, 시스템에서 제공되는 서비스를 이용할 수 있다. 로그인 후 로그인 세션은 어플리케이션을 종료하더라도 자동으로 유지된다. 로그인을 위한 비밀번호를 잊었을 경우 관리자에게 문의하여 등록한 아이디를 통해 자기 확인을 거친 후 비밀번호가 초기화된다. 또한 마이페이지에서 간단하게 로그아웃할 수 있다.

#### C. My Page

마이페이지에는 프로필 사진과 업로드한 반려동물 관련 게시물 목록, 코멘트 내역, 팔로잉(Following)한 계정 목록을 확인할 수 있다. 프로필 사진은 제공하는 기본 사진 외에 사용자가 임의로 업로드 가능하다. 마이페이지에서는 회원 탈퇴, 로그아웃 등의 자기 정보 관리도 가능하다. 팔로잉 아이디 목록을 통하여 그 아이디에서

등록한 게시글 목록을 볼 수 있다. 또한 반려견과 산책 중일 경우, 사용자는 마이페이지에서 산책 경과 상황 페이지로 이동할 수 있다.

#### D. Upload Pictures & Videos

사용자가 촬영한 반려견의 영상이나 사진을 업로드 할 수 있는 기능이다. 원한다면 글도 함께 게시할 수 있다. 사용자는 본인이 게시한 게시물을 수정 및 삭제할 수 있다. 다른 사용자의 열람을 허용하거나 금지 할 수 있다.

#### E. Tag

사용자가 사진 및 영상을 게시할 때, '#'를 쓴 후 그 뒤에 게시물과 관련된 단어를 적어 게시물에 대해 간단히 설명할 수 있는 기능이다. 사용자들은 특정 태그를 검색하여 관련된 게시글들을 찾아볼 수 있다.

#### F. Follow

사용자는 원하는 계정들을 팔로우 (구독) 할 수 있다. 구독한 사용자의 사진이나 영상은 피드에 추천되어 올라오게 되고, 이를 통해 새로운 정보를 빠르게 확인할 수 있다. 팔로잉 / 팔로워 탭에서 사용자가 구독하거나 사용자를 구독하는 일련의 계정들을 확인하고, 그곳에서 구독을 취소하거나 계정을 차단할 수 있다.

#### G. Searching

사용자가 태그 또는 텍스트를 이용하여 반려견에 대한 정보를 검색할 수 있는 기능이다. 사용자는 검색을 하는 데에 있어서 다양한 검색 필터를 이용하여 보다 쉽게 검색을 할 수 있다.

#### H. Showing Main Page

어플리케이션의 첫 화면에 여러 게시글들을 추천순, 시간순에 따라 나열해주는 기능이다. 이때 사용자 추천 contents는 사용자가 팔로잉하는 계정들을 바탕으로 하기 때문에 로그인이 필요하다. 로그인을 하게 되면 사용자가 팔로잉한 계정에 최근 업로드된 게시글을 띄워주는 사용자 추천 contents를 보여준다. 사용자는 다른 사용자가 게시한 게시글에 대하여 '좋아요' 표시와 간단한 댓글을 남길 수 있다.

## I. Walking-Mate Matching

반려견 보호자들이 자신의 스케줄, 지역등의 정보와 반려견의 성향, 나이, 체급 등의 정보를 올려 사용자가 직접 원하는 산책mate를 고르거나, 자동으로 산책 메이트를 추천, 매칭해준다.

## J. Walking-Information Page

Device 를 통해 기록된 걸음수, 산책 경로등을 저장, 열람할 수 있다. 지금까지의 산책 히스토리를 열람할 수 있으며, 현재 산책중일 경우 반려견의 생체정보를 바탕으로 산책에 조언이 되는 다양한 추천 문구와 경고 등이 뜬다.

## K. Dogwatch Peristalsis

사용자는 도그워치 (가제) 라는 반려견에게 부착할 수 있는 디바이스를 이 곳에서 연동할 수 있다. 최초 등록시 1회만 연동이 가능하며, 그 이후에는 저장되어 재등록할 필요는 없다. 연동되어 있는 디바이스를 연동 해제할 수 있다.

# 4.3 Non-Functional Requirements

Non-Functional Requirement는 Product Requirement, Organizational Requirement, External Requirement로 나눌 수 있다. 이 부분에서는 DOCKING 사용자들에게 필요한 비기능적 요구사항들을 서술한다.

## A. Product Requirements

### a. Performance Requirement

사용자가 매칭 서비스를 사용할 때에 문제되는 사항이 없도록 하여야 한다. 사용자가 스스로의 개인 정보와 반려견의 정보를 등록하고, 등록된 정보를 기준으로 같이 산책하기 좋은 다른 반려인 유저와 매칭해주고, 서로 연락할 수 있는 서비스를 제공하는 일련의 프로세스를 거치는 동안 다양한 performance적 측면에서 사용자가 어려움을 느끼지 않도록 하여야 한다. 또한 매칭 서비스를 통해 또는 혼자 산책을 할 때 오늘 산책 한 정보가 누락되지 않고 기입 되어 해당 사용자의 히스토리에 잘 저장되도록 하여야 한다. 매칭하는 과정에서 불필요한 과정으로 인해 사용상의 불편함을 느끼게 하여서는 안 되고, 반대로 과정이 너무 생략되어서도 안 된다. 또한 매칭해주는 과정에 있어서, 조건에 맞는 다른 반려인을 찾아 알려주는 것에 너무 많은 시간을 허비하게 해서는 안 된다.

### b. Security Requirement

이 시스템은 사용자의 critical한 개인정보들을 많이 포함하고 있으므로 그 어떤 정보도 유출되지 않게 하여야 하며 보안에 신경 써야 한다. 사용자의 개인적인 신상 정보부터, 반려견에 대한 정보, 커뮤니티에 업로드 해왔던 글과, 지금까지 산책해온 히스토리 등의 정보를 보호하며 외부의 악의적인 접근으로부터 정보들을 보호해야 한다. 특히 히스토리에 포함된 지금까지의 산책 경로, 빈도수, 산책 장소 등의 통계 자료는 사용자의 실제 동선을 반영하고 있으므로 이 정보에는 사용자 본인만 접근할 수 있도록 하여야 한다.

#### c. Efficiency Requirement

DOCKING은 매칭 시스템, 산책 후 히스토리 저장 등을 위해 서버와의 지속적인 통신을 전제로 한다. 매칭 시스템을 사용하는 데에 있어서 적합한 사용자 목록을 load하는 데에 있어 너무 많은 시간을 쓰지 않게 해야 할 것이고, 산책이 끝난 후에 오늘의 산책 정보를 히스토리에 추가하여야 하는데, 이 과정에서 누락되는 정보가 생겨서는 안 된다.

#### d. Usability Requirement

DOCKING은 사용자가 가능한 한 매일 산책을 하며 실외에서도 어렵지 않게 쓸 수 있도록 하기 위해 사용하기 쉽고, 친숙한 user interface를 제공하여야 한다. 정보 파트, 커뮤니티 파트, 산책 파트를 서로 다른 depth로 분리하여 각 기능을 사용하는 데에 어려움이 없도록 해야 한다. 또한 실외 사용이 주가 되는 어플리케이션이므로 배터리 소모량이 많아서는 안 된다.

### B. Organization Requirement

#### a. Environmental Requirement

DOCKING은 사용자가 어느 공간에서도 편하게 사용할 수 있도록 하기 위해, 안드로이드 모바일의 다양한 기기에서 쉽게 사용할 수 있도록 구현되어야 한다. 또한 이에 추가하여, 반려견이 착용하는 웨어러블 기기와의 연동되어야 한다.

#### b. Operational Requirement

DOCKING은 산책 또는 반려견과 관련된 게시글을 올릴 수 있는 커뮤니티이므로 사용자 사이의 게시글을 통한 통신, 소통이 가능하도록 한다. 사용자는 커뮤니티를 통해 산책 또는 반려견에 대한 정보를 얻어갈 수 있다. 산책 매칭이 성공적으로 이루어졌을 경우 반려인들이 서로 연락을 할 수 있도록 서비스를 제공한다.

### C. External Requirement

#### a. Regularity Requirement

DOCKING의 서비스들은 사용자의 개인정보를 입력 받아 회원가입을 한 후에 사용하는 것을 기본 전제로 한다. 그러므로 사용자의 개인정보를 외부에 제공하게 될 경우, 사전에 이용약관을 통해서, 또는 차후의 사용자 동의를 받는 것을 우선으로

해야 한다. 사용자 동의 사항에 명시되지 않은 대상에게는 개인정보에 대한 접근 권한을 주지 않는다.

**b. Safety, Security Requirement**

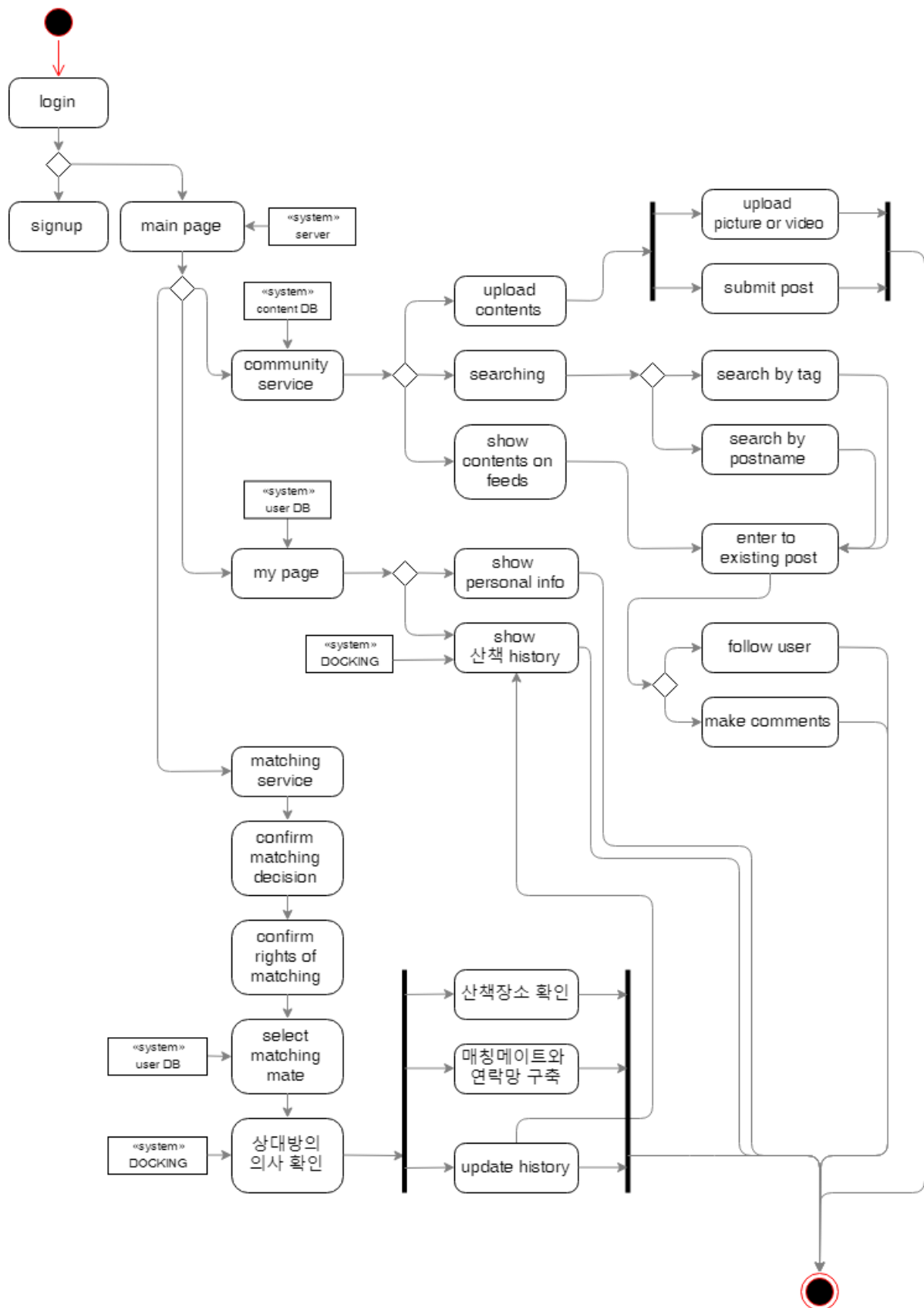
사용자들의 개인정보는 기본적으로 중앙 서버를 통해 관리된다. 관리자는 개인정보 보호를 위해 안전이 검증된 여러 방법을 사용해야 하며, 개인정보에 대한 비정상적인 접근을 예방하고, 막을 수 있어야 한다. 사용자가 제공한 개인정보의 보안성에 대한 대응 방식, 책임 사항 등은 회원 가입 시에 제공되는 약관을 통해 명시하고 법적인 보호를 보장하도록 한다.

## 5. System Architecture

### 5.1 Objective

System Architecture에서는 목표 시스템의 Architecture에 대한 고수준에서의 개요를 보여준다. 또한 시스템 기능의 전체적 분포를 보여준다. 재사용되는 컴포넌트들은 강조되어 있다.

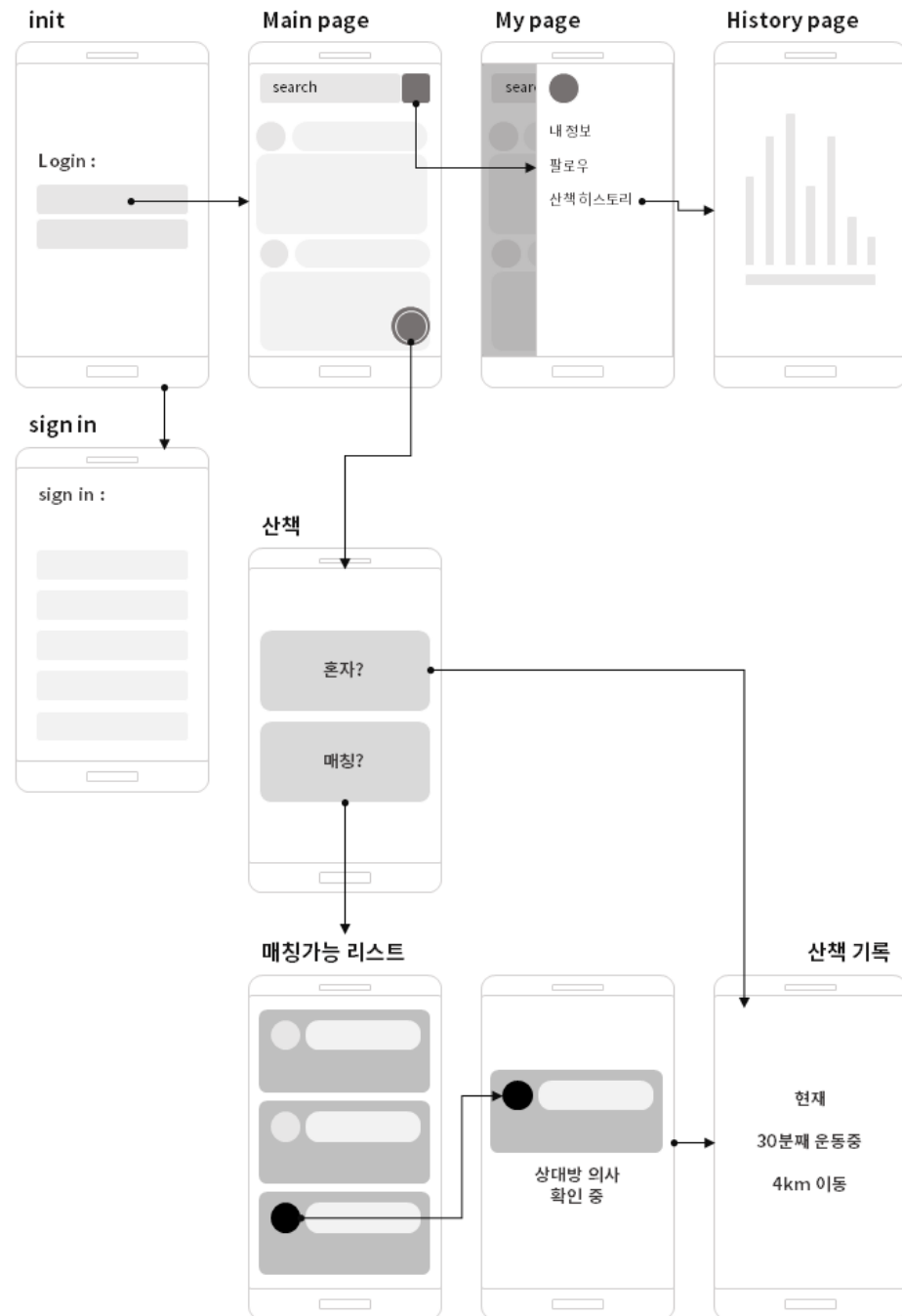
## 5.2 Overall Architecture



〈그림 2.1 Overall Architecture〉



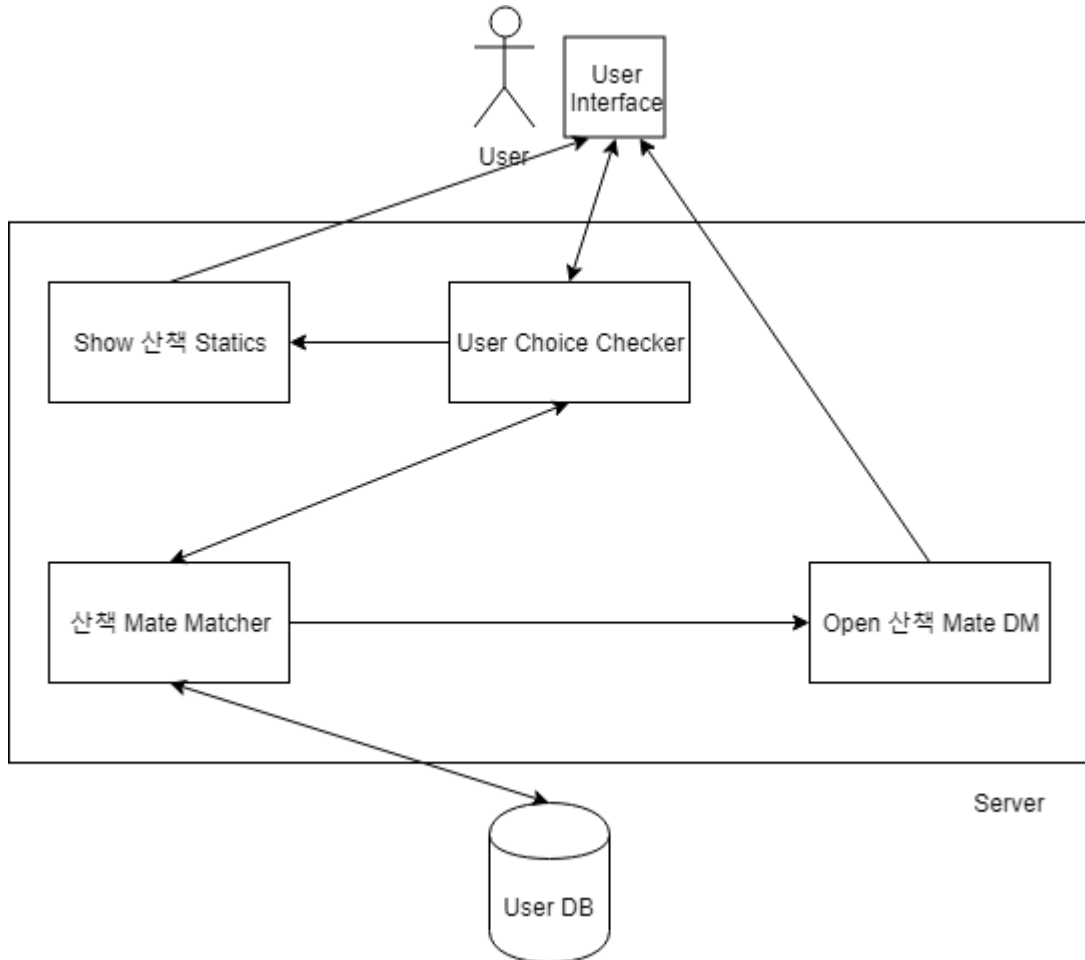
## 5.3 User Interface Structure



<그림 2.2 User Interface Structure>

## 5.4 Sub-system Architecture

### A. 산책 System



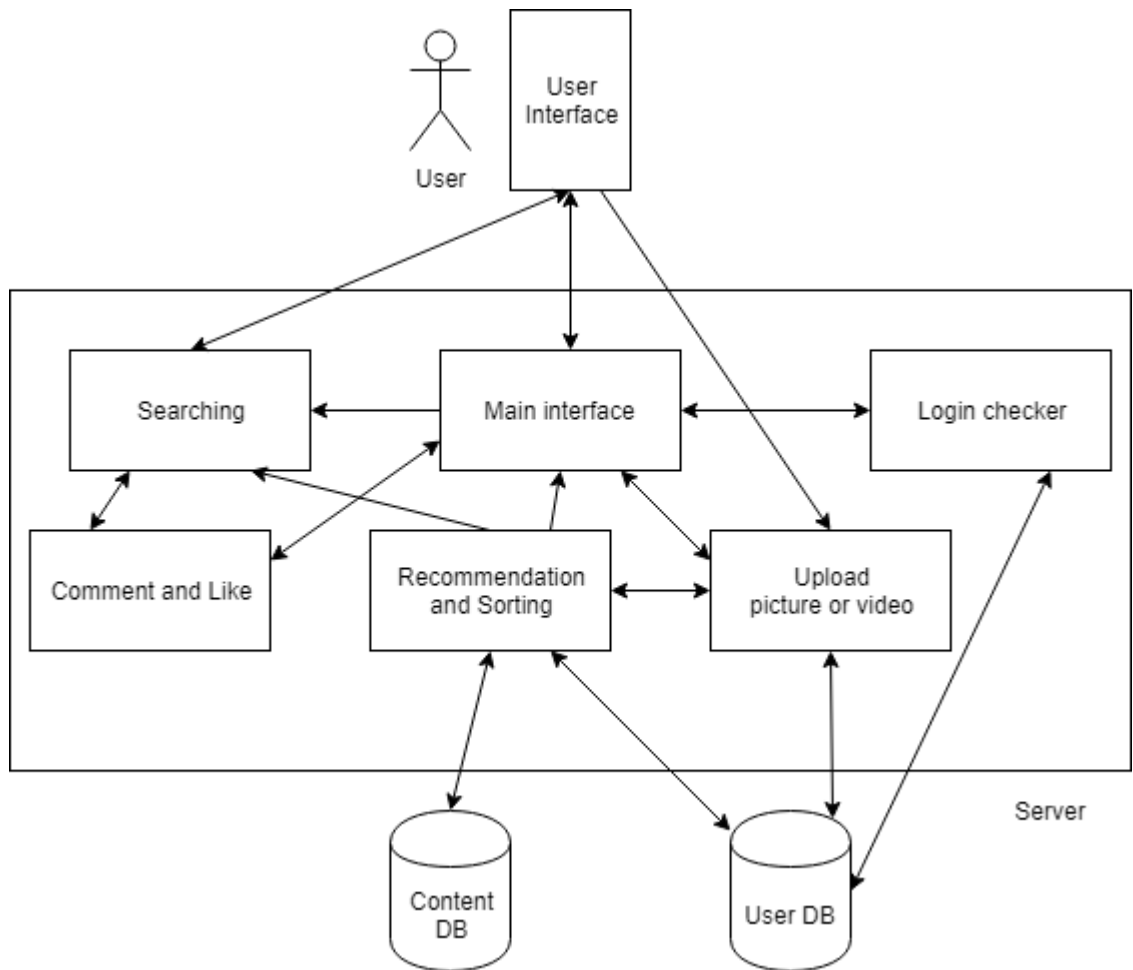
〈그림 2.3 산책 sub-system Architecture〉

산책 System은 사용자의 정보와 맞는 적절한 산책 Mate 를 매칭시켜주고 상대방의 동의가 있을 경우 Mate와 연락망을 개통해준다. 뿐만 아니라 디바이스의 GPS와 지도 API 를 이용하여 산책정보에 대한 표시를 해준다. 그 기록은 유저의 히스토리에 남아 마이페이지에서 추후에 확인 가능하다.

먼저 메인화면에서 ‘산책하자!’ 버튼을 클릭하면, ‘혼자’ 산책을 진행할지 ‘매칭’을 통해 Mate와 산책을 할지 system이 요청한다. ‘혼자’ 산책을 진행할 경우 바로 산책정보 표시 페이지로 넘어가며 그 시점부터의 활동을 기록하고 표시한다.

다른 산책 mate와의 ‘매칭’을 진행할 경우 사전에 system에 입력한 반려자의 정보, 반려견의 정보에 따른 적절한 산책 mate를 매칭해준다. 이후 매칭가능한 mate들의 리스트를 보여주고 그중 원하는 mate에게 의사 확인 요청을 한다. 상대방도 동의를 할 경우 mate간의 연락망을 개통해주고 산책기록 페이지로 넘어가게 된다.

## B. 커뮤니티 System

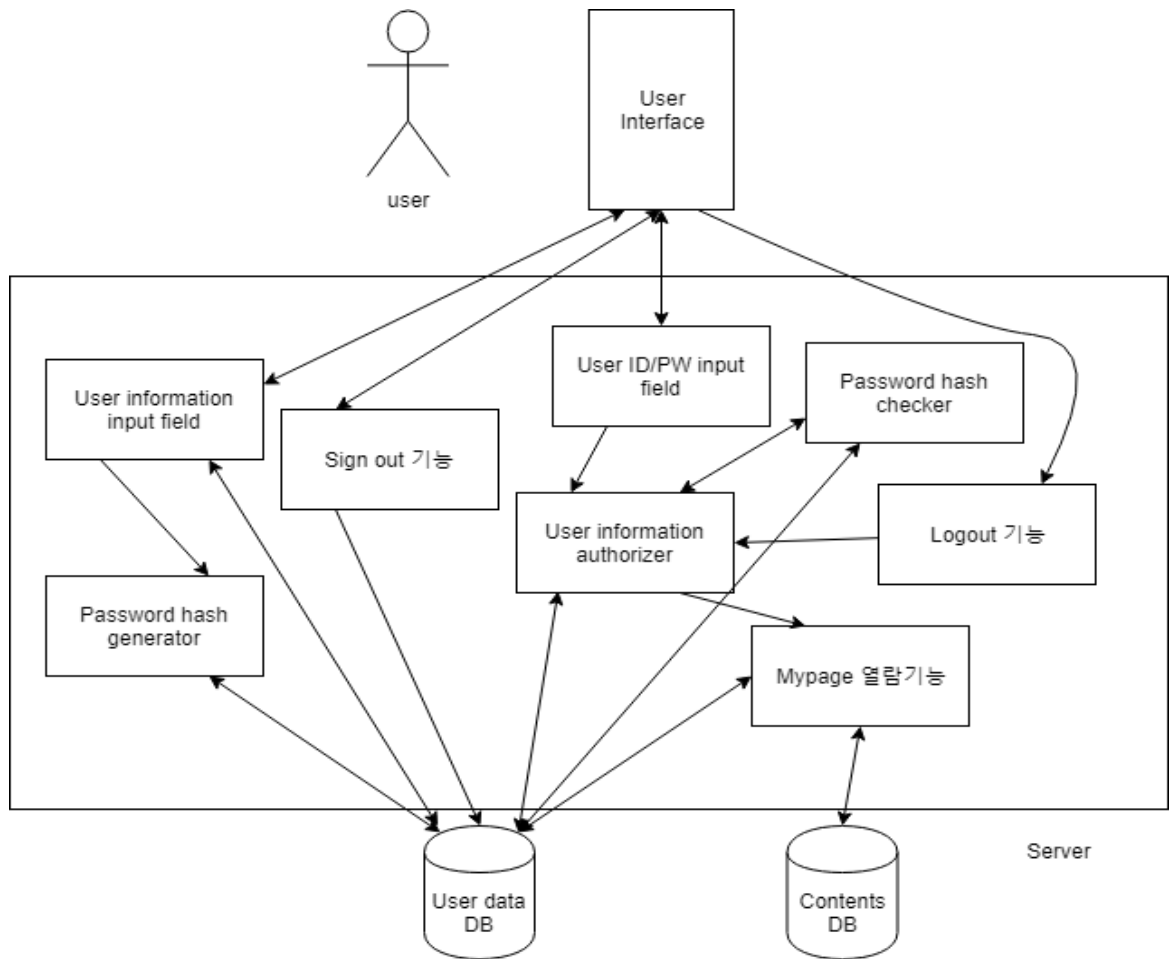


〈그림 2.4 커뮤니티 sub-system Architecture〉

커뮤니티 시스템은 사용자가 사진, 영상을 담은 게시물을 게시할 수 있게 해 주고, 사용자가 팔로우한 계정들이 올린 게시글 정보를 바탕으로 사용자의 community main page를 보여준다. community main page에서는 게시글의 추천수, 좋아요 수를 바탕으로 사용자에게 추천되는 순서, 그리고 시간순으로 sorting되어 나타난다.

사용자는 찾고자 하는 정보에 대해 태그 또는 텍스트를 이용하여 검색할 수 있다. 사용자가 입력한 태그 또는 텍스트를 입력받으면 Content DB에서 일치하는 게시글을 불러와 연관도 순으로 정렬하여 사용자의 화면에 띄워준다. 사용자는 search 결과 화면에서도 좋아요와 댓글을 달 수 있다.

## C. User System



&lt;그림 2.5 User sub-system Architecture&gt;

## a. Sign in/out

Sign in/out 시스템은 사용자의 정보를 생성하거나 삭제한다. Login 화면의 Sign in 버튼을 누르면 약관을 보여주는 페이지를 표시하고, 약관에 동의한 사용자로부터 개인정보를 입력받아 DB에 저장하고, Password는 Hash로 처리하여 DB에 저장한다. 중복된 사용자/ 중복된 ID의 생성을 방지하며, Password는 길이검사를 하며, 허용된 문자만이 입력 가능하다. Sign out 기능은 Mypage에서 접근 가능하며, 사용자의 개인정보를 삭제하거나 사용 불가처리후 약관에 명시된 기일 이후에 삭제한다.

## b. Login/out

Login/out은 이미 생성된 계정에 접속하여 게시물/개인정보에 접근하는 권한을 주는 시스템이다. 사용자로부터 ID와 password를 받아 password의 hash값이 DB에 저장된

해당 ID의 password hash값과 일치하는지 비교한다. login 된 이후에는 사용자가 직접 logout 할 때까지는 login 상태가 유지되도록 한다. 사용자가 logout 을 한 후에는 login 화면으로 되돌아간다.

### c. Mypage

Mypage에서는 비밀번호를 제외한 사용자 자신의 정보를 열람하거나 비밀번호를 포함한 자신의 정보를 수정할 수 있다. 또한 자신이 작성한 게시물의 일람을 확인하고, 접근, 삭제 또한 가능하다.

## 6. System Requirement Specification

### 6.1 Objective

System Requirement Specification 에서는 Functional requirements 또는 Non-functional requirements에 대해 더욱 자세히 설명한다. 다른 시스템에 대한 인터페이스 역시 이 장에서 정의된다.

### 6.2 Functional requirements

#### A. Sign In/Out

Subsystem	Sign In/Out
Function	사용자 가입/탈퇴
Use case	계정생성, 계정삭제.
Actor	User, user Database
Description	사용자가 서비스를 이용하는 데에 필요한 개인 정보를 제공받아 회원으로 등록하는 기능이다. 회원 가입을 통해 로그인 절차를 거쳐야만 사용자는 'Docking'에서 제공하는 서비스를 이용할 수 있다. 'Docking'을 사용하는 사용자는 일반 반려인, 비반려인과 반려인 전문가가 있을 수 있고, 가입 시 구분 절차를 거친다. 반려인의 경우 반려동물에 대한 정보 역시 입력한다. 또한 마이 페이지에서 탈퇴를 선택하면 회원 등록 취소와 개인 정보 삭제가 이루어진다.

Stimuli	<ul style="list-style-type: none"> <li>- 회원가입 진행 시 어플리케이션의 이용약관과 개인정보 수집 및 이용에 관한 약관에 모두 동의를 요청한다. 사용자가 모두 동의할 경우 정보 입력을 진행한다.</li> <li>- 반려인의 경우 반려동물에 대한 정보입력을 요청하고, 전문가의 경우 경력사항과 자격에 대한 정보를 요청한다.</li> </ul>
Response	<ul style="list-style-type: none"> <li>- 회원가입을 통해 입력받은 정보를 DB에 저장한다.</li> <li>- 중복가입 시도시엔 가입 불가능하다는 메시지를 출력한다.</li> <li>- 회원 탈퇴시엔 DB에서 사용자 정보를 사용 불가능하게 한다.</li> </ul>
Process	<ul style="list-style-type: none"> <li>- User DB에 사용자 정보를 저장한다.</li> <li>- Password는 Hash로 처리하여 저장한다.</li> <li>- 탈퇴신청을 한 사용자의 정보는 사용자 약관에 따라서 삭제 혹은 약관에 명시된 기간 이후에 삭제한다.</li> </ul>
Constraints	<ul style="list-style-type: none"> <li>- 한 사람은 하나의 계정만 가질 수 있다.</li> <li>- 같은 ID가 두개 이상 존재할 수 없다.</li> <li>- 패스워드는 알파벳 대소문자, 특수문자 아라비아숫자만을 사용하여 6자 이상 20자 미만으로 한다.</li> </ul>

〈Table 3.1 Sign In/Out Functional Requirements〉

## B. Log In/Out

Subsystem	Log In/Out
Function	User Login/out
Use case	사용자 로그인, 사용자 로그아웃.
Actor	User, User DB
Description	사용자가 회원 가입을 통해 생성된 아이디를 이용하여 시스템에 로그인하는 기능이다. 로그인 후 사용자는 자신의 데이터베이스에 접근할 수 있으며, 시스템에서 제공되는 서비스를 이용할 수 있다. 로그인 후 로그인 세션은 어플리케이션을 종료하더라도 자동으로 유지된다. 로그인을

	위한 비밀번호를 잊었을 경우 관리자에게 문의하여 등록한 아이디를 통해 자기 확인을 거친 후 비밀번호가 초기화된다. 또한 마이페이지에서 간단하게 로그아웃할 수 있다.
Stimuli	<ul style="list-style-type: none"> <li>- Docking 회원가입시 입력한 아이디, 비밀번호를 입력함으로 로그인한다.</li> <li>- 사용자가 직접 로그아웃하기 전까지는 로그인 상태 유지한다.</li> </ul>
Response	<ul style="list-style-type: none"> <li>- 로그인 성공시 로그인 성공 메시지를 출력한다.</li> <li>- 로그인 이후 Main page를 출력한다.</li> <li>- 로그아웃 시 로그인 화면을 출력한다.</li> </ul>
Process	<ul style="list-style-type: none"> <li>- 사용자에게 입력받은 ID 와 password를 검사하여 user DB와 일치하는지 확인한다.</li> <li>- Password 검사 성공시 client를 로그인 상태로 만든다.</li> <li>- 로그아웃시 로그인 상태를 해제한다.</li> </ul>
Constraints	<ul style="list-style-type: none"> <li>- 로그인 5회 이상 실패시 일시적으로 잠금상태로 들어간다.</li> <li>- 로그인 된 상태에서 또다시 로그인 할 수 없다</li> <li>- 한번에 하나의 계정만 로그인이 가능하다.</li> </ul>

〈Table 3.2 Log In/Out Functional Requirements〉

## C. My Page

Subsystem	My Page
Function	My Page
Use case	Checking & correcting User Info, Show Walking history
Actor	User, User DB
Description	마이페이지에는 프로필 사진과 업로드한 반려동물 관련 게시글 목록, 코멘트 내역, 팔로잉(Following)한 계정 목록을 확인할 수 있다. 프로필 사진은 제공하는 기본 사진 외에 사용자가 임의로 업로드 가능하다. 마이페이지에서는 회원 탈퇴, 로그아웃 등의 자기 정보 관리도 가능하다. 팔로잉 아이디 목록을 통하여 그 아이디에서 등록한 게시글 목록을 볼 수 있다.

	또한 반려견과 산책 중일 경우, 사용자는 마이페이지에서 산책 경과 상황 페이지로 이동할 수 있다.
Stimuli	<ul style="list-style-type: none"> <li>- 사용자가 원하는 사진으로 프로필 사진을 수정한다.</li> <li>- 산책 정보 확인을 위해 산책페이지로 넘어간다.</li> <li>- 회원 탈퇴를 원하는 경우 절차를 거쳐, 회원정보를 삭제한다.</li> </ul>
Response	<ul style="list-style-type: none"> <li>- 새로운 프로필 사진으로 업데이트 한다.</li> <li>- 산책 History를 보여준다.</li> <li>- 회원정보를 삭제하고, 로그아웃 한다.</li> </ul>
Process	<ul style="list-style-type: none"> <li>- 사용자 DB에 사진을 기존사진에서 변경한다.</li> <li>- 사용자 DB에 있는 산책기록 data를 불러온다.</li> <li>- 전체 user DB 리스트에서 사용자 데이터를 삭제한다.</li> </ul>
Constraints	<ul style="list-style-type: none"> <li>- 회원가입/로그인이 되어있어야 한다.</li> <li>- 이전 산책 기록이 있어야만 통계를 표시할 수 있다.</li> </ul>

〈Table 3.3 My Page Functional Requirements〉

#### D. Upload Pictures & Videos

Subsystem	Upload Pictures & Videos
Function	Upload Pictures & Videos
Use case	Posting, Deleting, Modifying
Actor	User, Contents Database
Description	사용자가 촬영한 반려견의 영상이나 사진을 업로드 할 수 있는 기능이다. 원한다면 글도 함께 게시할 수 있다. 사용자는 본인이 게시한 게시물을 수정 및 삭제할 수 있다. 다른 사용자의 열람을 허용하거나 금지 할 수 있다.
Stimuli	<ul style="list-style-type: none"> <li>- 사용자가 사진이나 비디오를 업로드 한다.</li> <li>- 사진, 동영상에 대한 설명글도 함께 게시할 수 있다.</li> </ul>
Response	<ul style="list-style-type: none"> <li>- 시스템은 게시된 글의 제목과 내용물을 DB에 저장한다.</li> <li>- 사진이나 영상이 허용된 용량을 초과할 경우, 게시가 불가능함을 출력한다.</li> </ul>
Process	<ul style="list-style-type: none"> <li>- Contents DB에 게시물의 정보를 저장하고, DB에 그</li> </ul>



	경로를 저장한다. - 게시물을 수정 및 삭제할 때는 DB에서 게시물의 정보를 수정 및 삭제한다.
Constraints	- 사진은 한 게시물에 10장까지만 게시 가능하다. - 동영상은 한 게시물에 하나씩만 게시 가능하다. - 사진이나 동영상 없이 글만 게시하는 것은 불가능하다. - 사진과 동영상을 동시에 게시하는 것은 불가능하다.

〈Table 3.4 Upload Pictures &amp; Videos Functional Requirements〉

## E. Tag

Subsystem	Tag
Function	Tag
Use case	Tagging, Searching
Actor	User, Contents Database
Description	사용자가 사진 및 영상을 게시할 때, '#'를 쓴 후 그 뒤에 게시물과 관련된 단어를 적어 게시물에 대해 간단히 설명할 수 있는 기능이다. 사용자들은 특정 태그를 검색하여 관련된 게시글들을 찾아볼 수 있다.
Stimuli	- 사용자가 '#' + 텍스트 를 사진이나 동영상 설명과함께 게시한다. - 사용자는 '#' + 텍스트를 통해 태그를 검색한다.
Response	- 게시물의 하단에 게시물에 대한 설명과 함께 태그를 보여준다. - 태그를 검색할 경우 해당 태그가 작성되어 있는 게시물들을 보여준다. - 태그가 수정될 경우 Contents DB를 업데이트한다.
Process	태그를 검색하면 Database에서 해당 태그를 가지고 있는 게시물을 불러온다.
Constraints	Space(띄어쓰기)는 Tag 할 수 없다.

〈Table 3.5 Tag Functional Requirements〉

## F. Follow

Subsystem	Follow
Function	Follow
Use case	Follow, UnFollow, Block
Actor	User, User Database, User's channel
Description	사용자는 원하는 계정들을 팔로우 (구독) 할 수 있다. 구독한 사용자의 사진이나 영상은 피드에 추천되어 올라오게 되고, 이를 통해 새로운 정보를 빠르게 확인할 수 있다. 팔로잉 / 팔로워 탭에서 사용자가 구독하거나 사용자를 구독하는 일련의 계정들을 확인하고, 그곳에서 구독을 취소하거나 계정을 차단할 수 있다.
Stimuli	<ul style="list-style-type: none"> <li>- 구독하기를 원하는 계정의 피드에서 '팔로우' 버튼을 누른다.</li> <li>- 구독을 취소하거나 차단하고 싶은 계정에 대해 '팔로우 취소', '차단' 버튼을 누른다.</li> </ul>
Response	<ul style="list-style-type: none"> <li>- 팔로우 리스트에 원하는 계정이 추가된다.</li> <li>- 팔로우 리스트에서 해당 계정이 삭제된다.</li> </ul>
Process	팔로우 기능의 주체인 사용자와 객체인 사용자의 데이터베이스를 갱신한다.
Constraints	본인의 계정은 팔로우 (구독) 할 수 없다.

〈Table 3.6 Follow Functional Requirements〉

## G. Searching

Subsystem	Searching
Function	Search Contents
Use case	Search Contents by Tag, Search Contents by Text
Actor	User, Contents Database

Description	사용자가 태그 또는 텍스트를 이용하여 반려견에 대한 정보를 검색할 수 있는 기능이다. 사용자는 검색을 하는 데에 있어서 다양한 검색 필터를 이용하여 보다 쉽게 검색을 할 수 있다.
Stimuli	<ul style="list-style-type: none"> <li>- 사용자가 원하는 태그 또는 텍스트에 대해 검색한다.</li> <li>- 필터를 추가하여 선택적인 정보를 얻는다.</li> </ul>
Response	<ul style="list-style-type: none"> <li>- 태그 검색의 경우 Contents DB에서 Contents들의 태그 정보를 검색하여 일치하는 태그가 있을 경우 일치하는 태그가 있는 게시글들의 대표 이미지의 바둑판 형식으로 화면에 띄워준다.</li> <li>- Text 검색의 경우 Contents DB에 사용자가 입력한 Text와 동일한 Text를 포함하는 Contents가 있을 경우 일치하는 태그가 있는 게시글들의 대표 이미지의 바둑판 형식으로 화면에 띄워준다.</li> <li>- 만약 검색결과가 없을 경우 검색 결과가 없음을 알려주고 다른 이름으로 검색할 것을 추천한다.</li> <li>- 검색어를 입력하지 않았을 경우 검색어를 입력하라고 알려준다.</li> <li>- 검색어 수가 제한 범위를 넘을 경우 입력이 되지 않도록 한다.</li> </ul>
Process	사용자가 입력한 태그 또는 Text와 일치하는 Contents를 종합하여 나타낸다.
Constraints	<ul style="list-style-type: none"> <li>- 필터 수를 제한한다.</li> <li>- 검색어가 없을 수 없다.</li> <li>- 검색어는 한 글자일 수 없다.</li> <li>- 태그는 한 번에 하나만 검색 가능하다.</li> </ul>

〈Table 3.7 Searching Functional Requirements〉

## H. Showing Main Page

Subsystem	Showing Main Page
Function	Showing Main Page
Use case	Recommended, Recent Update, Like, Comments
Actor	User, Contents Database, User Database

Description	어플리케이션의 첫 화면에 여러 게시글들을 추천순, 시간순에 따라 나열해주는 기능이다. 이때 사용자 추천 contents는 사용자가 팔로잉하는 계정들을 바탕으로 하기 때문에 로그인이 필요하다. 로그인을 하게 되면 사용자가 팔로잉한 계정에 최근 업로드된 게시글을 띄워주는 사용자 추천 contents를 보여준다. 사용자는 다른 사용자가 게시한 게시글에 대하여 '좋아요' 표시와 간단한 댓글을 남길 수 있다.
Stimuli	- 사용자의 로그인 정보를 바탕으로 구독 정보를 전송한다.
Response	<ul style="list-style-type: none"> <li>- 사용자의 접속이 확인되면, 사용자가 팔로우하고 있는 구독자의 게시글 중 해당 시간으로부터 일정 시간 간격 동안 contents의 조회수를 Contents DB를 통해 추적하여 조회수가 높은 contents 순으로 Recommend한다.</li> <li>- 사용자의 접속이 확인되면, Contents DB를 통해 사용자가 팔로우하고 있는 구독자의 게시글 중 가장 최근에 업로드된 Contents를 기준으로 최근 업데이트 contents를 선정한다.</li> </ul>
Process	사용자가 접속하면 사용자가 구독하고 있는 유저의 contents들을 추천순, 시간순으로 sorting한다.
Constraints	

<Table 3.8 Showing Main Page Functional Requirements>

## I. Walking-Mate Matching

Subsystem	Walking-Mate Matching
Function	Walking-Mate Matching
Use case	Mate Matching with provided Information
Actor	반려인
Description	반려견 보호자들이 자신의 스케줄, 지역등의 정보와 반려견의 성향, 나이, 체급 등의 정보를 올려 사용자가 직접 원하는 산책mate를 고르거나, 자동으로 산책 메이트를 추천, 매칭해준다.

Stimuli	<ul style="list-style-type: none"> <li>- 사용자는 본인의 지역, 스케줄 정보와 반려견의 정보를 입력한다.</li> <li>- 자동 매칭을 기다리거나 직접 메이트를 고른다.</li> </ul>
Response	<ul style="list-style-type: none"> <li>- 매칭이 완료되면 알림을 준다.</li> <li>- 산책메이트 연락망 개통</li> </ul>
Process	사용자DB의 정보와 맞는 mate 추천
Constraints	한번에 1사람끼리만 매칭 가능, 반려인 승인을 받아야함.

〈Table 3.9 Walking-Mate Matching Functional Requirements〉

## J. Walking-Information Page

Subsystem	Walking-Information Page
Function	Walking-Information Page
Use case	Show distance, time, calories and routes.
Actor	산책히스토리가 있는 사용자, 산책중인 사용자
Description	Device 를 통해 기록된 걸음수, 산책 경로등을 저장, 열람할 수 있다. 지금까지의 산책 히스토리를 열람할 수 있으며, 현재 산책중일 경우 반려견의 생체정보를 바탕으로 산책에 조언이 되는 다양한 추천 문구와 경고 등이 뜬다.
Stimuli	<ul style="list-style-type: none"> <li>- Device를 통해 측정된 산책데이터를 열람한다.</li> <li>- 산책을 위한 조언, 정보를 얻는다.</li> </ul>
Response	<ul style="list-style-type: none"> <li>- 현재 걷고있는 Km수, 시간, 칼로리 등을 표시한다.</li> <li>- 이전 산책 history에 대한 통계자료 확인가능.</li> </ul>
Process	지도 API 및 gps 연동으로 운동 기록 저장, 표시
Constraints	산책 history가 있어야만 통계 확인가능, 디바이스의 gps 필요

〈Table 3.10 Walking-Information Page Functional Requirements〉

## K. Dogwatch Peristalsis

Subsystem	Dogwatch Peristalsis
Function	Dogwatch Peristalsis

Use case	Sync band with application
Actor	Dogwatch, user, Smartphone
Description	사용자는 도그워치 (가제) 라는 반려견에게 부착할 수 있는 디바이스를 이 곳에서 연동할 수 있다. 최초 등록시 1회만 연동이 가능하며, 그 이후에는 저장되어 재등록할 필요는 없다. 연동되어 있는 디바이스를 연동 해제할 수 있다.
Stimuli	<ul style="list-style-type: none"> <li>- 도그워치를 어플리케이션과 연동한다.</li> <li>- 최초 등록인 경우에만 등록 절차를 거친다.</li> </ul>
Response	<ul style="list-style-type: none"> <li>- 새로 연동한 Dogwatch의 정보가 저장된다.</li> <li>- 삭제시 지정한 Dogwatch의 정보가 삭제된다.</li> </ul>
Process	Bluetooth 통신을 통해 Dogwatch 의 고유정보를 전송받아 저장한다.
Constraints	한번 등록한 Dogwatch를 또다시 새로운 장비로서 등록할 수 없다. 시스템에 등록한 반려견의 수 만큼 Dogwatch를 등록 할 수 있다.

〈Table 3.11 Dogwatch Functional Requirements〉

## 6.3 Non-Functional Requirements

Non-Functional Requirement는 Product Requirement, Organizational Requirement, External Requirement로 나눌 수 있다. 이 부분에서는 DOCKING 시스템에 필요한 비기능적 요구사항들을 서술한다.

### A. Product Requirements

#### a. Performance Requirement

DOCKING 시스템은 매칭 및 산책 시스템을 사용할 때 performance 측면에서 문제가 발생하지 않도록 하여야 한다. 매칭 서비스에서, 알고리즘을 통해 같이 산책하기에 적절한 다른 반려인을 찾는 로딩 시간이 너무 길어져서는 안된다. 또한 매칭이 성공하였을 경우 산책 기록 verse로 넘어가서, 휴대 전화의 일부 기능을 사용하여 이를 적절히 기록하는 과정으로 이어질 때 진행상의 문제가 없어야 한다.

#### b. Security Requirement

DOCKING은 사용자의 개인적인 신상 정보부터, 반려견에 대한 정보, 커뮤니티에 업로드 해왔던 게시글과, 지금까지 산책해온 히스토리 등의 정보를 보호하며 외부의 악의적인 접근으로부터 정보들을 보호해야 한다. 히스토리에 포함된

지금까지의 산책 경로, 빈도수, 산책 장소 등의 통계 자료 정보에는 사용자 본인만 접근할 수 있도록 권한 설정을 하여야 한다.

#### c. Efficiency Requirement

DOCKING은 매칭 시스템, 산책 후 히스토리 저장 등을 위해 서버와의 지속적인 통신을 전제로 한다. 따라서, 서버와 효율적으로 통신할 수 있는 시스템으로 개발하여야 하며, 매칭 시스템을 사용하는 데에 있어서 적합한 사용자 목록을 load하는 데에 있어 너무 많은 시간을 쓰지 않게 해야 할 것이다.

#### d. Usability Requirement

DOCKING은 사용자가 가능한 한 매일 산책을 하며 실외에서도 어렵지 않게 쓸 수 있도록 하기 위해 사용하기 쉽고, 친숙한 user interface를 제공하여야 한다. 정보 파트, 커뮤니티 파트, 산책 파트를 서로 다른 depth로 분리하여 각 기능을 사용하는 데에 어려움이 없도록 해야 한다. 또한 실외 사용이 주가 되는 어플리케이션이므로 배터리 소모량이 많아서는 안 된다.

### B. Organization Requirements

#### a. Environmental Requirement

DOCKING은 사용자가 어느 공간에서도 편하게 사용할 수 있도록 하기 위해, 안드로이드 모바일의 다양한 기기에서 쉽게 사용할 수 있도록 구현되어야 한다. 또한 이에 추가하여, 반려견이 착용하는 웨어러블 기기와의 연동되어야 한다. 연동을 통해 반려견의 상태와 관련한 정보를 받아올 수 있어야 한다.

#### b. Operational Requirement

DOCKING은 산책 또는 반려견과 관련된 게시글을 올릴 수 있는 커뮤니티이므로 사용자 사이의 게시글을 통한 통신, 소통이 가능하도록 한다. 사용자는 커뮤니티를 통해 산책 또는 반려견에 대한 정보를 얻어갈 수 있고, 커뮤니티는 사용자들이 매일 산책을 하도록 유도하는 효과를 갖는다. 산책 매칭이 성공적으로 이루어졌을 경우 반려인들이 서로 연락을 할 수 있는 서비스를 제공한다.

### C. External Requirements

#### a. Regulatory Requirement

DOCKING의 서비스들은 사용자의 개인정보를 입력 받아 회원가입을 한 후에 사용하는 것을 기본 전제로 한다. 그러므로 사용자의 개인정보를 외부에 제공하게 될 경우, 사전에 이용약관을 통해서, 또는 차후의 사용자 동의를 받는 것을 우선으로 해야 한다. 사용자 동의 사항에 명시되지 않은 대상에게는 개인정보에 대한 접근 권한을 주지 않는다.

#### b. Safety, Security Requirement

사용자들의 개인정보는 기본적으로 중앙 서버를 통해 관리되며, 개인정보 보호를 위해 안전이 검증된 여러 방법을 사용해야 한다. 이를 기반으로 시스템을 구축하여 관리자는 개인정보에 대한 비정상적인 접근을 예방하고, 막을 수 있어야 한다. 사용자가 제공한 개인정보의 보안성에 대한 대응 방식, 책임 사항 등은 회원 가입 시에 제공되는 약관을 통해 명시하고 법적인 보호를 보장한다.

## 6.4 Scenario

### A. Walking Mate Matching Scenario

#### a. Initial Assumption

사전에 유저는 반려동물이 있는 반려인임이 인증 되어야 하고, 회원가입시 유저의 정보와 반려동물에 대한 정보를 system에 입력한다.

#### b. Normal Flow of Events

매칭을 원하는 유저는 본인이 입력한 정보(위치, 시간, 반려동물 종류 등)를 기반으로 가장 적절한 Mate의 list를 추천받는다. 그중 원하는 mate에게 의사 확인 요청을 하고, 만약 상대 Mate 의 동의를 얻는다면 두 Mate 사이의 연락망(Direct Message)를 개통하고 둘사이의 연락이 가능하게 한다.

#### c. What can go wrong & Concurrent Activities

2명 이상의 mate 와 약속을 잡는것을 방지하기 위해, list 에서 mate선택 후에는 다른 mate를 선택할 수 없게 하고, 상대방의 동의가 있기 전에는 연락망을 개통해주지 않는다.

#### d. System state on Completion

매칭 이후 연락망 개통을 통해 산책 Mate 들의 산책모임을 돕는다.

### B. Walking Information Page Scenario

#### a. Initial Assumption

반려인들은 산책을 하고 device를 통해 수집된 정보는 system에 기록된다.

#### b. Normal Flow of Events

산책을 시작할 시점부터 끝날때까지의 산책기록(거리, 시간, 소모 칼로리 등)을 device를 통해 수집하고 해당 user의 Database에 저장한다. 이후 유저는 이전의 Walking history에 대한 통계를 열람할 수 있다. 지도 Api를 이용, 지도에 산책 course를 표시한다.

또한 산책 시간, 거리 정보를 통해 반려동물의 상태에 대한 조언이나 알림을 받을 수 있다.

#### c. What can go wrong & Concurrent Activities



Device의 gps 기능이 꺼져있는 경우 기록측정 전 유저에게 알림을 준다. 개인정보 등의 이유로 산책 history를 저장하고싶지 않는 유저의 경우 데이터베이스에 기록을 남기지 않는다.

#### d. System State on Completion

산책중인 반려인과 반려동물의 운동 정보를 실시간으로 보여주며, 추후 그에대한 통계자료를 열람할 수 있다.

### C. Community Main Page Scenario

#### a. Initial Assumption

커뮤니티의 메인 페이지를 볼 수 있는 사람들은 로그인 상태이다. User는 1명 이상을 팔로우하고 있다. 즉, 1명 이상의 사람을 친구추가 하여 친구인 상태여야 한다. 그래야 팔로우한 계정의 게시물을 볼 수 있고, 그 게시물에 좋아요를 남기거나 코멘트를 달 수 있다.

#### b. Normal Flow of Events

사용자가 팔로우하는 계정들의 좋아요 개수, 댓글 수를 기준으로 추천도를 결정하고, 최신순에 따라 sorting하여 사용자의 커뮤니티 메인 화면에 노출시킨다. 사용자는 여기에서 다른 사용자 혹은 자신의 게시글에 좋아요를 누르거나 코멘트를 달 수 있다.

사용자는 상단의 검색바에서 태그, 텍스트 등을 이용하여 찾고자 하는 정보를 검색할 수 있다.

#### c. What can go wrong & Concurrent Activities

로그인 상태가 아닌 경우 사용자에게 로그인 화면을 띄워 준다.

팔로우하고 있는 계정이 없는 경우 사용자에게 팔로우 추천 계정을 띄워 준다.

사용자가 검색창에 입력한 텍스트가 한 글자일 경우 경고 팝업을 띄워 수정하도록 한다. 검색 결과가 존재하지 않는 경우 검색 결과가 존재하지 않습니다, 라는 문구와 함께 추천 검색어를 띄워준다.

#### d. System State on Completion

다른 유저들이 남긴 게시글을 사용자가 보고, 코멘트란에서 정보를 교환할 수 있다.

사용자가 원하는 결과를 검색함으로써 찾을 수 있다.

### D. Uploading Scenario

#### a. Initial Assumption

자신의 페이지에 포스팅을 할 수 있는 사람들은 정상적인 회원가입 과정을 통해 계정이 생산된 사람들이다. 그리고 회원 가입 시, 반려인, 비반려인,

전문가 중 한 가지의 카테고리를 고른 사람들이다. 사용자 카메라 또는 앨범에 대한 접근 허가를 받은 상태이다.

**b. Normal Flow of Events**

자신의 페이지에 게시글을 올리하고자 하는 사람들은 자신의 앨범에서 사진 또는 동영상을 선택하거나, 새로운 사진을 찍는다. 게시 당시 약간의 보정 과정을 거칠 수 있다. 게시 후 게시글 수정은 불가능하며, 삭제만 가능하다. 게시글에 대한 좋아요는 1회 가능하다. 글을 함께 게시할 수도 있고, 원하지 않는다면 사진 혹은 영상만을 업로드할 수 있다. 태그 역시 사용할 수 있다. 그 후 그 게시글을 자신의 페이지에 포스팅할 것인지 여부를 묻는 창이 뜬다. 그 후 User가 Yes를 누르면 사진 혹은 영상과 글이 게시된다.

**c. What can go wrong & Concurrent Activities**

영상을 게시할 때는 한 번에 하나씩만 가능하다. 태그는 한 게시글당 10개 이상 남길 수 없다. 사진은 한 게시글에 최대 10개 게시 가능하다. 동영상과 사진을 한 게시글에 함께 게시할 수 없다. 사진이나 영상 없이 글만 게시하는 것은 불가능하다. 사진이나 영상이 허용된 용량을 초과하는 경우 게시가 불가능하다. 이 경우, 사용자에게 경고 팝업 알림을 띄우고, 게시글을 수정하도록 한다.

사용자의 앨범 혹은 카메라에 대한 접근이 비인가 상태일 경우 게시글을 올릴 수 없다. 이 경우, 사용자에게 경고 팝업 알림을 띄우고, 인가를 받도록 한다.

**d. System State on Completion**

내가 올린 게시글의 목록이 스크롤 형식으로 보여지며, 바둑판 형식과 스크롤 형식 중 보는 방식을 선택할 수 있다.

## E. Sign in/out Scenario

**a. Sign in Scenario**

**i. Initial Assumption**

처음 서비스에 접속한 사용자는 회원 가입을 시도한다. 사용자는 약관에 따라 정해진 개인정보를 제공할 의향이 있으며, 아직 가입되지 않은 상태이다.

**ii. Normal Flow of Events**

서비스를 처음 사용하고자 하는 사람들은 어플리케이션을 실행 한 후 Login 화면에 접속하여 Sign in 버튼을 누른다. 약관과 개인정보 이용동의서에 동의한 후 정해진 개인정보를 입력한 뒤 계정을 생성한다. 이때 ID는 Email

address에 해당하며, 새로 계정을 생성한 사용자는 최초 1회 이메일주소 인증을 받아야 한다.

### iii. What can go wrong & concurrent Activities

이미 가입된 이메일 주소로는 새로운 계정을 또 생성할 수 없다. Password는 충분한 보안을 제공하기 위하여 최소 6글자로 하며 알파벳 대소문자, 아라비아 숫자, 특수문자만이 이용가능하다.

### iv. System State on Completion

이메일 인증을 완료한 후, 회원가입이 완료되었다는 메시지와 함께 main page를 출력한다.

## b. Sign out Scenario

### i. Initial Assumption

계정을 가졌으며 탈퇴를 원하는 사용자는 Sign out 기능을 이용한다.

### ii. Normal Flow of Events

Mypage 에 접속한 사용자는 Sign out 기능을 선택하여 Sign out을 시도한다. 이때 탈퇴 여부를 한번 더 확인하며, Password를 한번 더 확인한다.

### iii. What can go wrong & concurrent Activities

Password를 일정 횟수 이상 틀리면 Sign out 이 불가능하며, Login 연속 실패와 동일하게 계정을 잠그고 확인 Email을 발송한다. 약관 혹은 개인정보 이용동의에 명시된 경우, 개인정보가 즉시 삭제되지 않고 명시된 기간동안 보관된다.

### iv. System State on Completion

계정이 사용 불가능하게 되며, 개인정보가 삭제, 혹은 삭제시까지 접근불가처리된다.

## F. Login /out Scenario

### a. Login Scenario

#### i. Initial Assumption

사용자는 서비스를 이용하기 위해 Login을 시도한다. 이때 사용자는 이미 계정을 생성한 상태이다.

#### ii. Normal Flow of Events

Login화면을 확인한 사용자는 ID란에 ID (Email Address)를, Password란에 자신이 기억하는 Password를 입력한다. 입력 후 Login 버튼을 눌러 Login을 시도한다. ID에 맞는 Password임이 확인되면 login에 성공한다.

#### iii. What can go wrong & concurrent Activities

Login 시에는 반드시 ID에 맞는 Password가 입력되어야 하며, Password가 일치하지 않을 시, Login에 실패한다. Login을 일정횟수 이상 실패시, 로그인기능이 일시적으로 잠금상태가 되며, Email 주소로 로그인 시도 실패 안내 메일을 발송한다.

#### iv. System State on Completion

Login에 성공 후 main page를 출력한다.

### b. Logout Scenario

#### i. Initial Assumption

사용자는 현재 계정으로부터 Logout 하고자 한다.

#### ii. Normal Flow of events

사용자는 Mypage에 접속하여 Logout 버튼을 누른다. 기기에서 Login 정보가 삭제되고 사용자는 Mainpage로 돌아간다.

#### iii. What can go Wrong & concurrent Activities

#### iv. System State on Completion

사용자의 기기로부터 Login 정보가 삭제되고 앱을 처음 실행한 상태로 돌아간다.

## G. Mypage Scenario

### a. Initial Assumption

사용자는 자신의 이용내역 혹은 정보를 확인, 편집하고자 Mypage 버튼을 누른다.

### b. Normal Flow of Events

Mypage 화면을 확인한 사용자는 산책 정보를 확인하기 위해 Walking information 에 접근한다. 자신의 개인정보를 확인하거나 Password를 포함한 개인정보를 변경 가능하다. 자신의 게시글을 확인하기 위해서는 게시물 기능에 접근한다.

### c. What can go wrong & concurrent Activities

작성한 게시물이 없다면 게시물을 열람 할 수 없다. 반려견 정보를 등록하지 않았다면 반려견 정보를 열람할 수 없다.

### d. System State on Completion

사용자가 community로 돌아가거나, walking information 페이지에 접속한다.

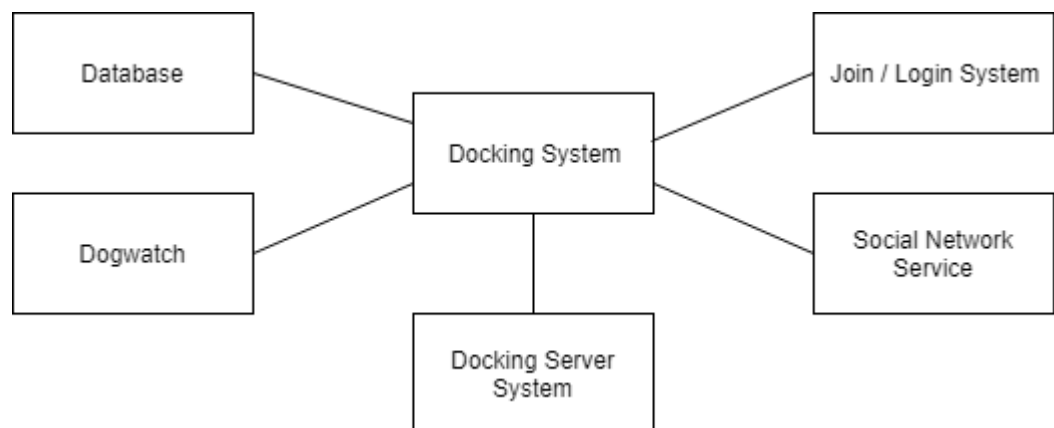
## 7. System models

### 7.1 Objective

System model에서는 시스템 컴포넌트, 시스템 그리고 시스템 환경 사이의 관계를 보여준다. 가능한 그래픽 모델로는 object models, data-flow models, semantic data models 등이 있다.

### 7.2 Context Models

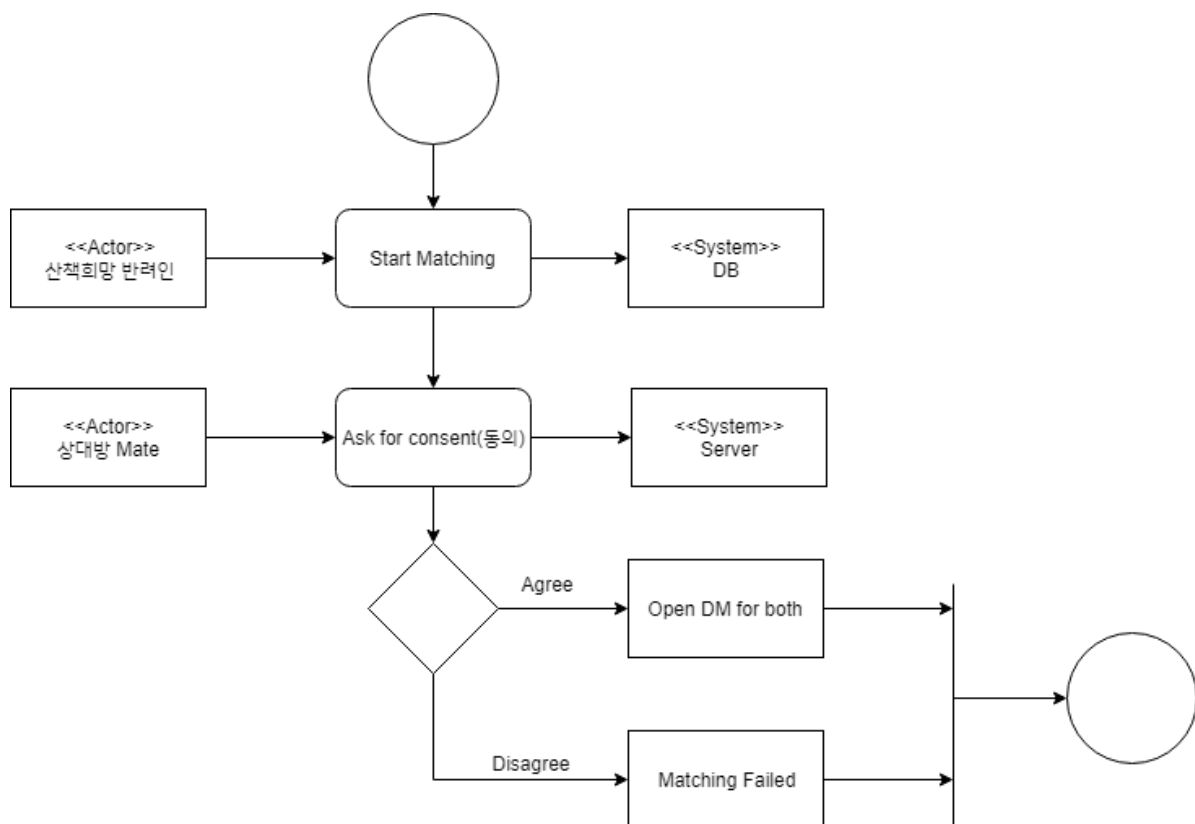
#### A. Context Model



<그림 3.1 Context Model>

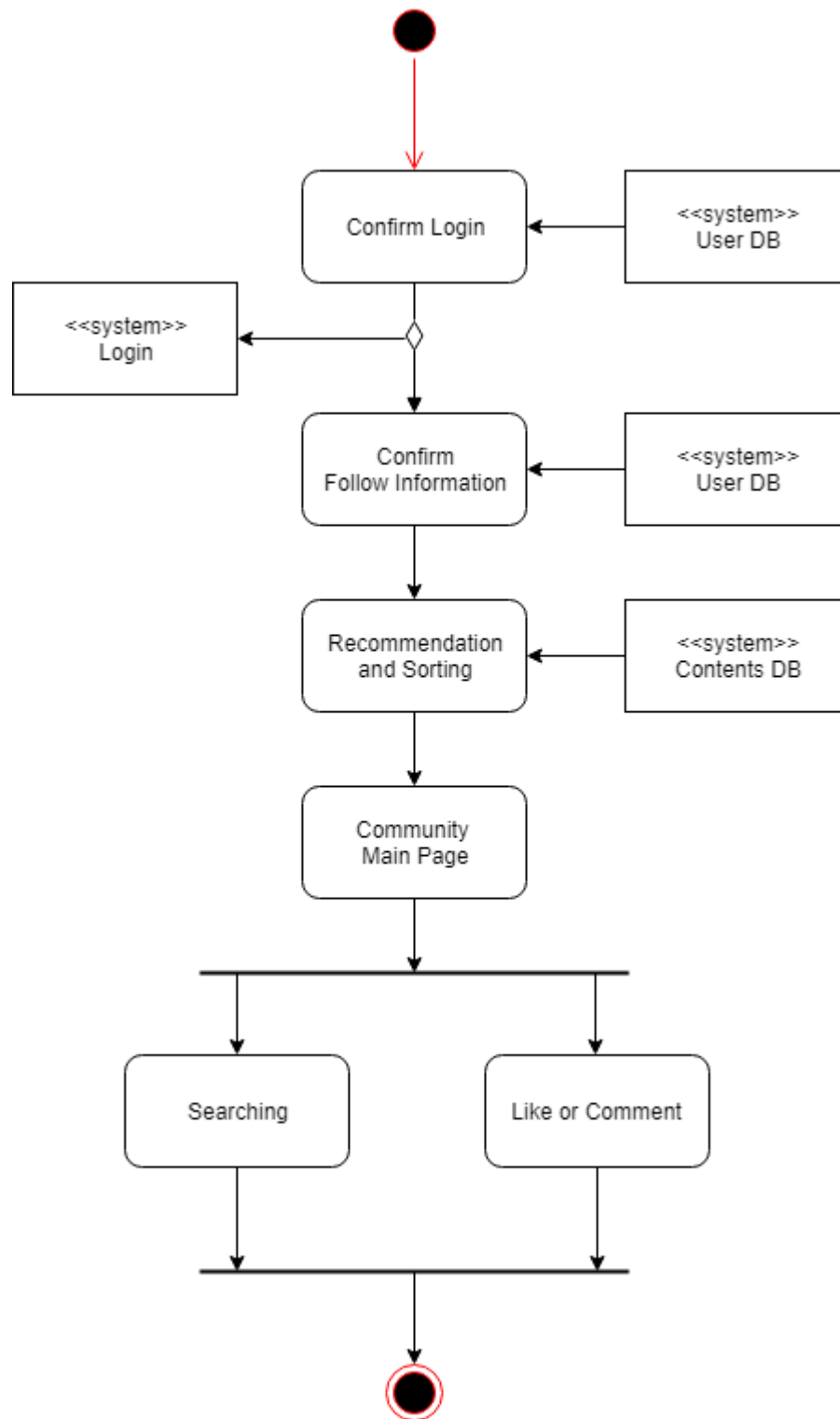
## B. Process Diagram

### a. Walking Mate Matching Process



<그림 3.2 Walking Mate Matching Process Diagram>

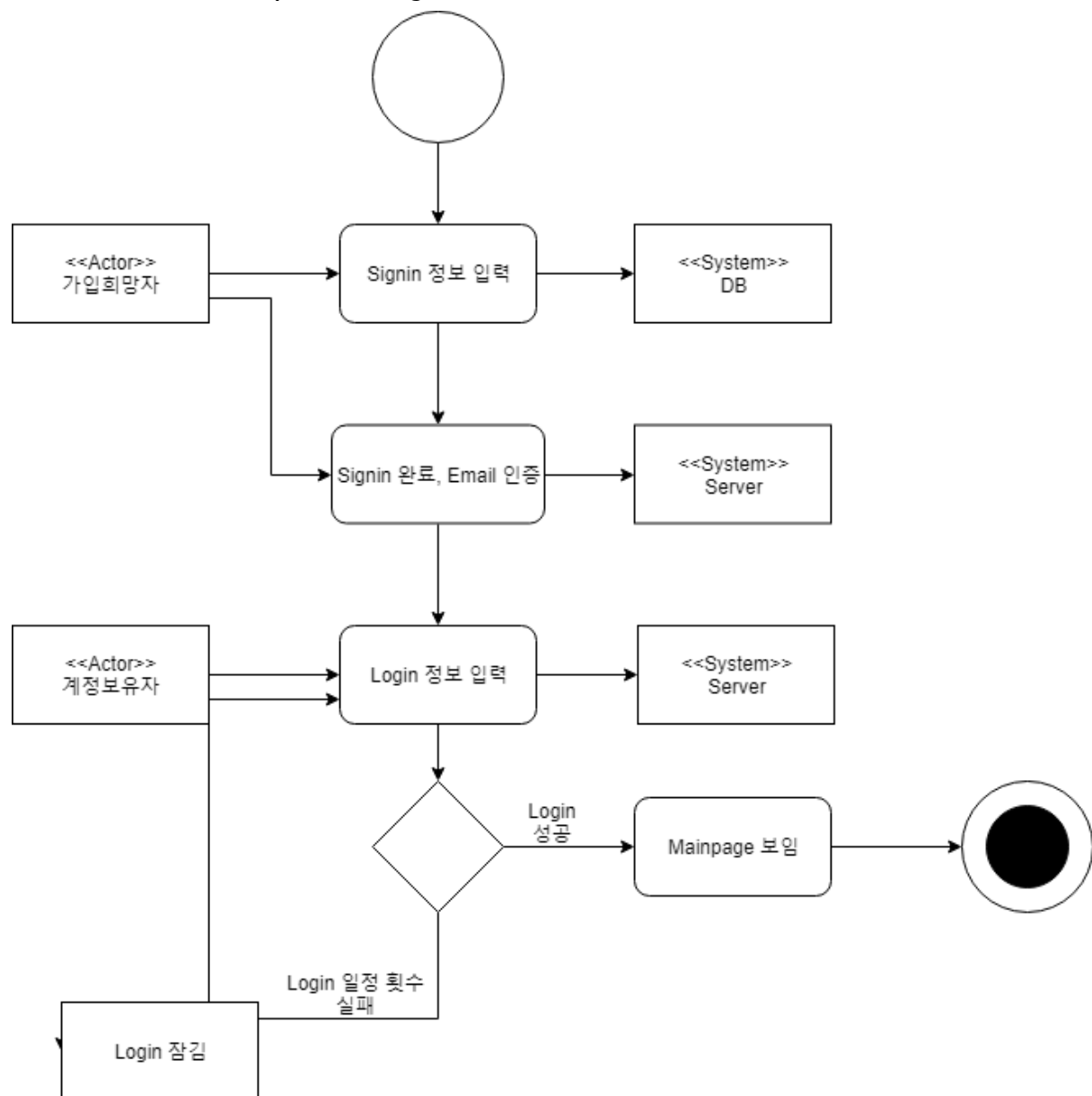
b. Community and Searching Process



<그림 3.3 Community and Searching Process Diagram>



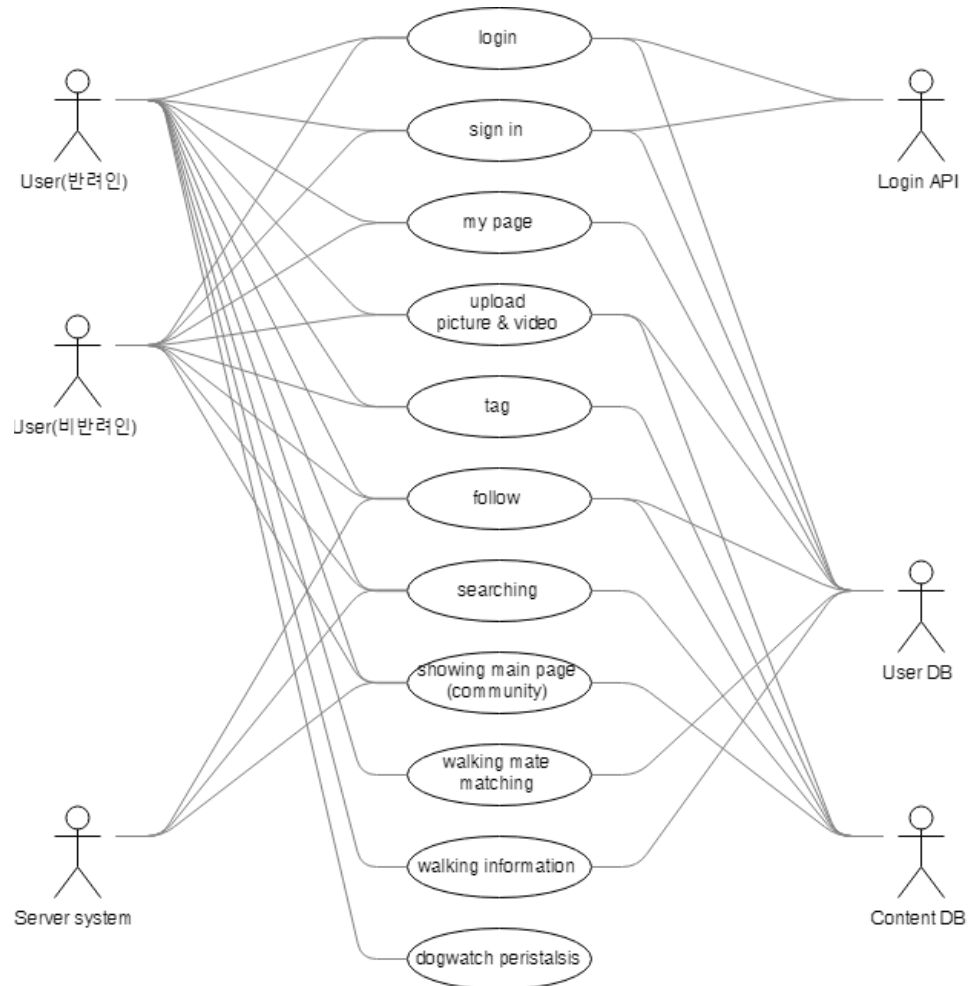
c. User process diagram



〈그림 3.4 User Process Diagram〉

## 7.3 Interaction Models

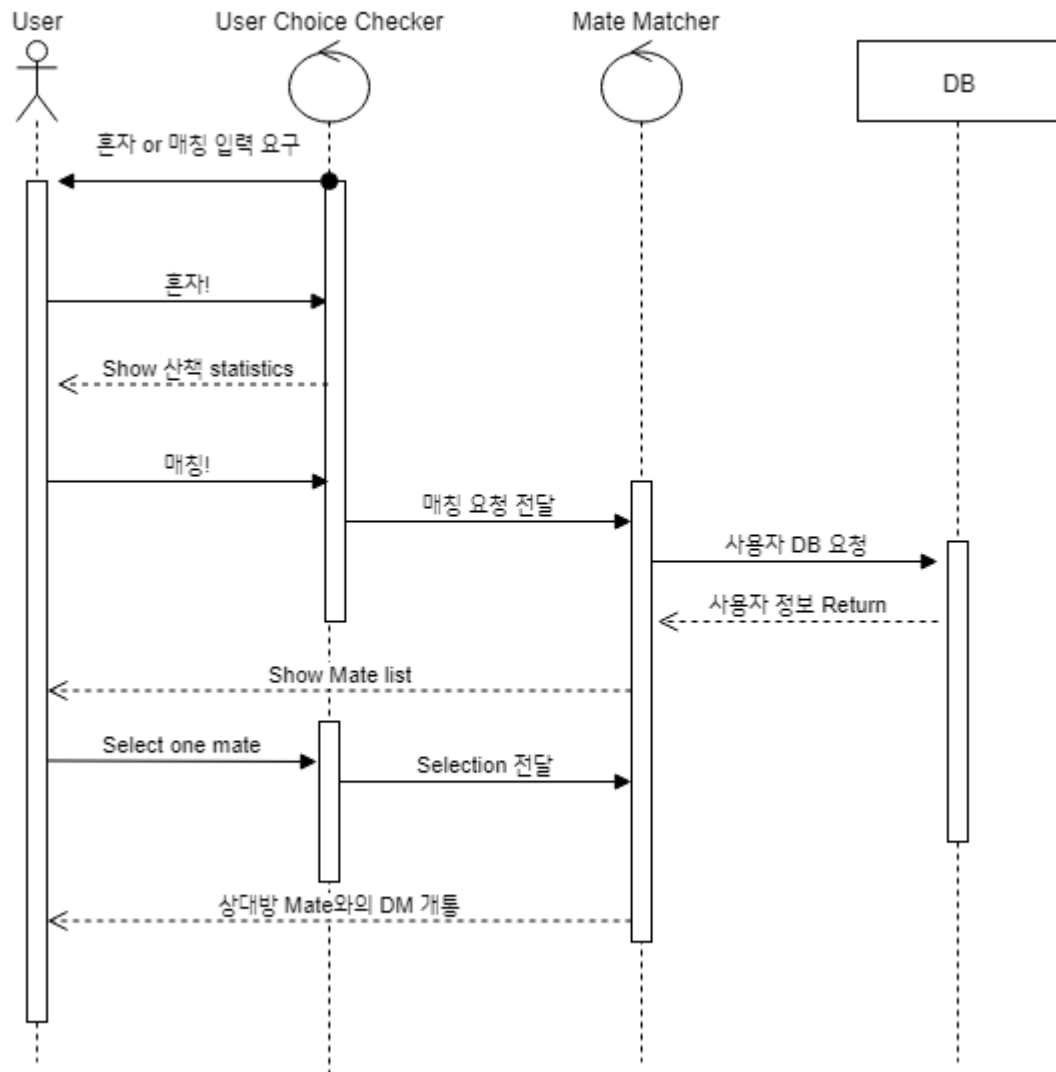
### A. Use case Diagram



<그림 3.5 Use case Diagram>

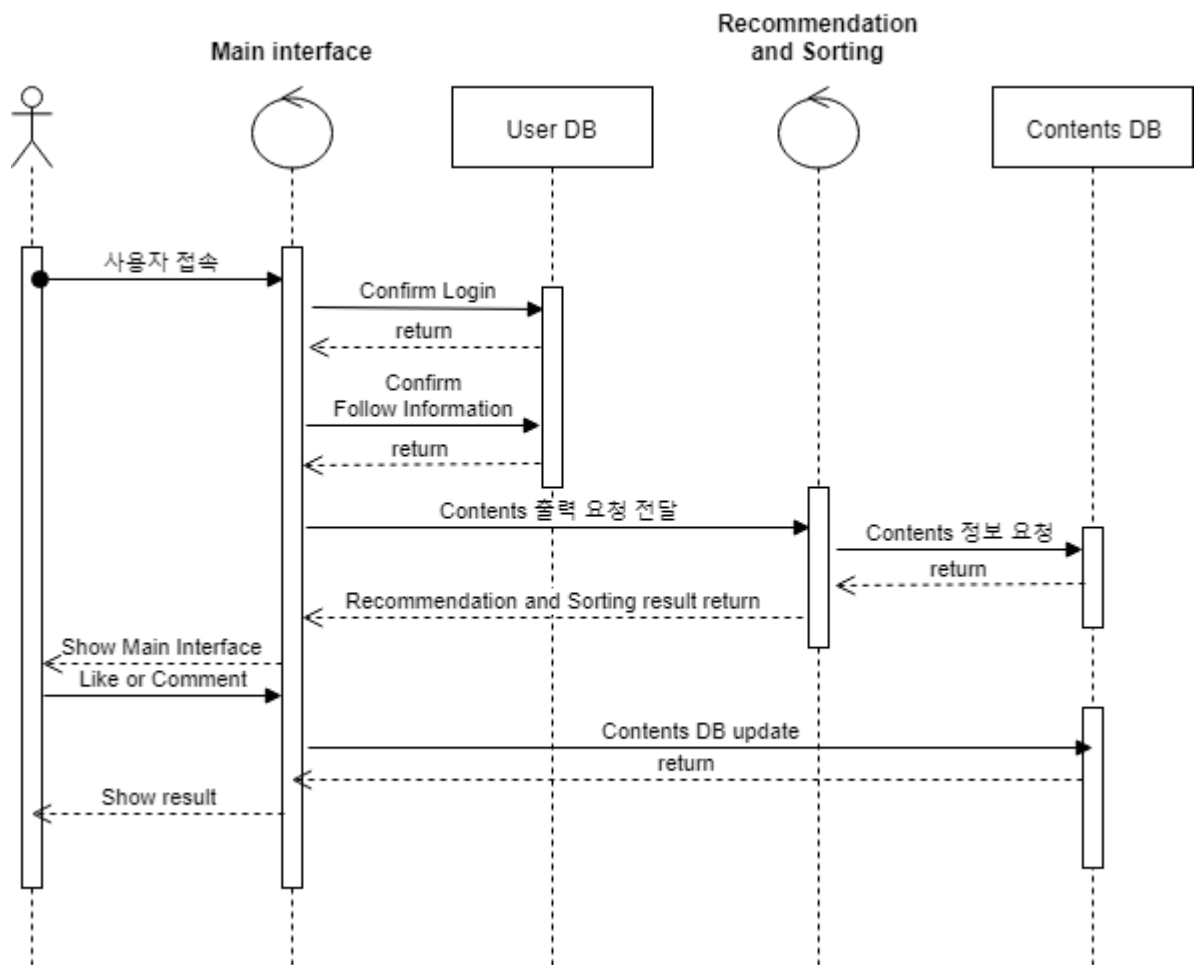
## B. Sequence Diagram

## a. Sequence Diagram of Walking System



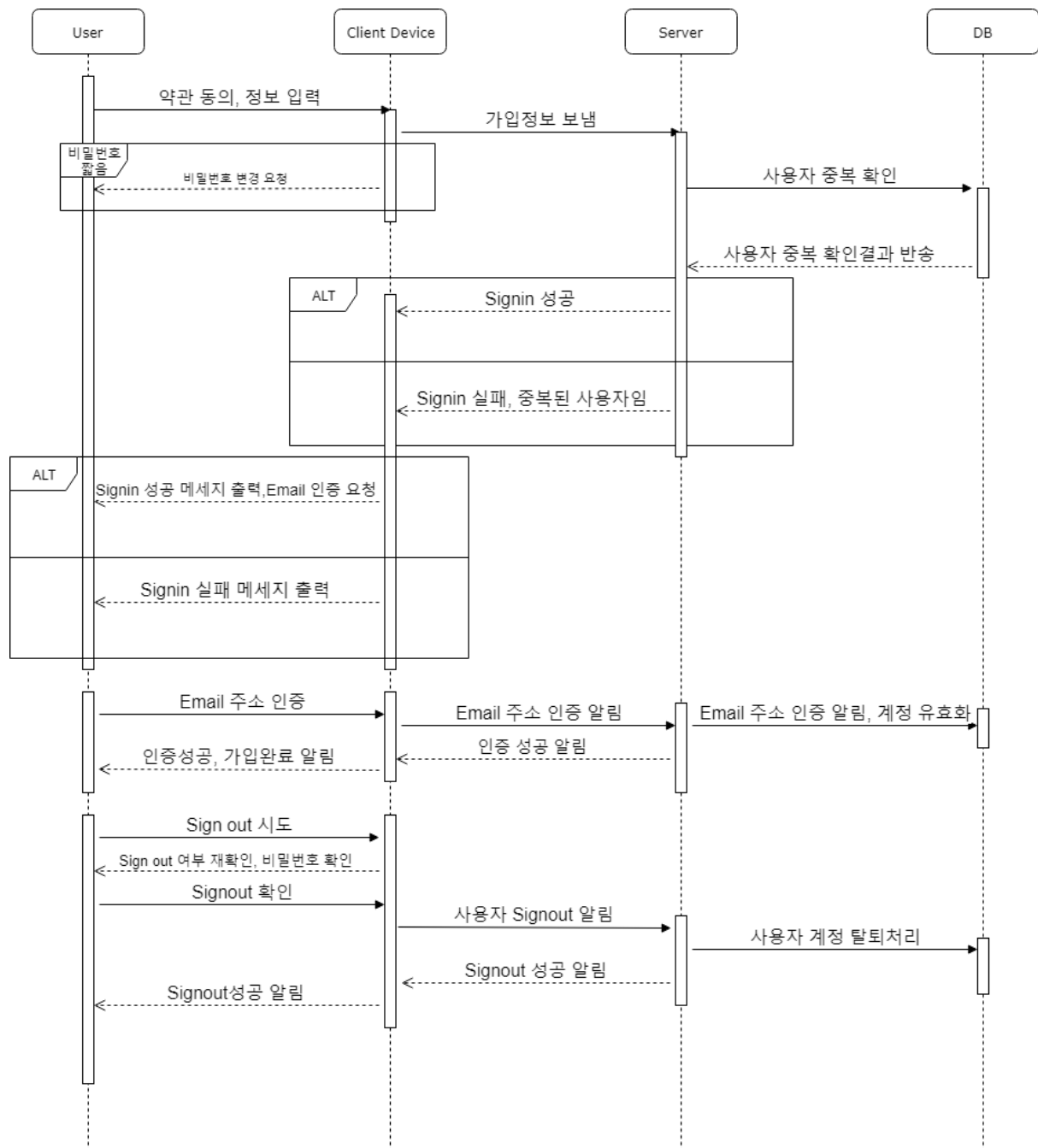
〈그림 3.6 Sequence Diagram of Walking System〉

## b. Sequence Diagram of Community System



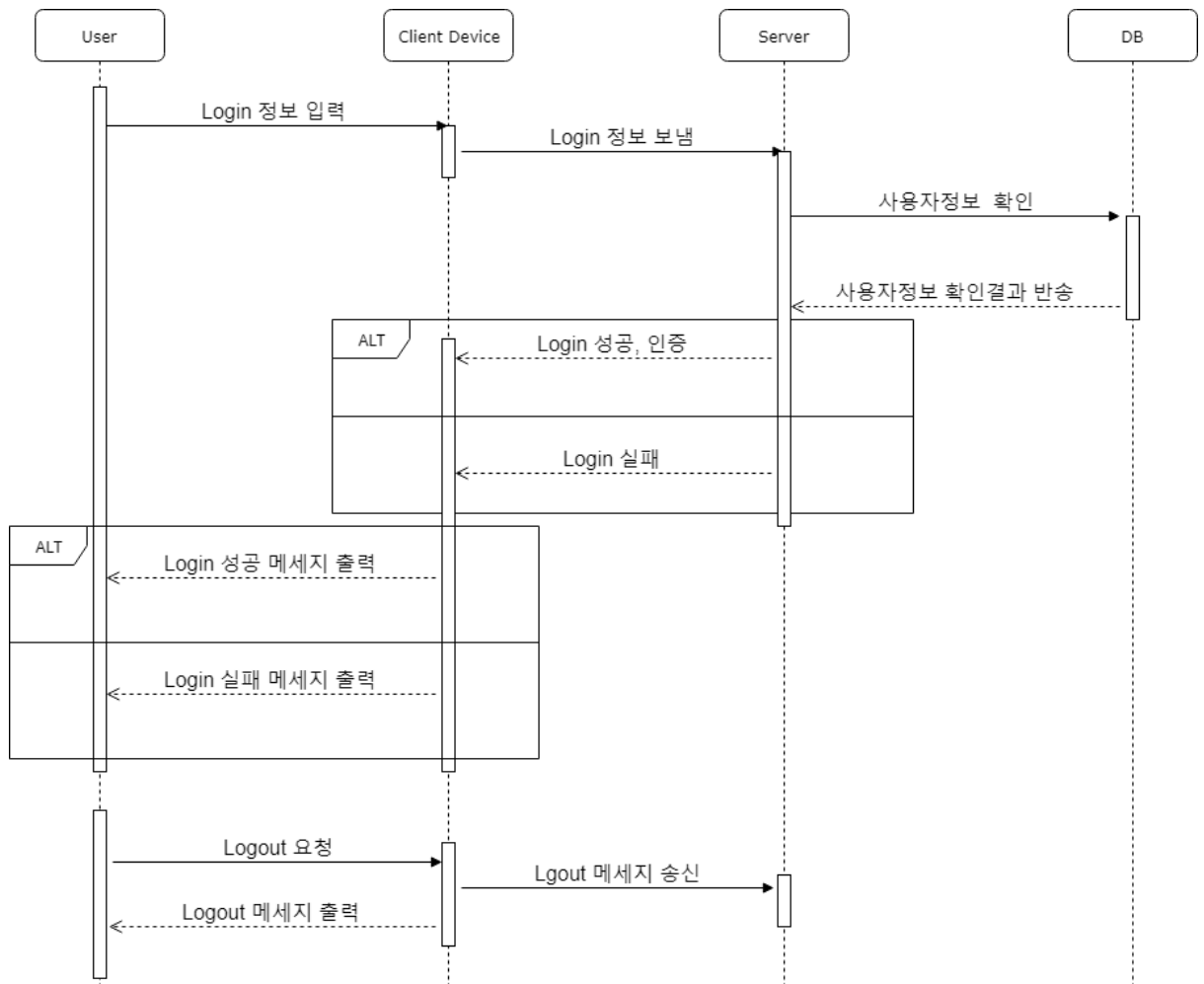
〈그림 3.7 Sequence Diagram of Community System〉

## c. Sequence Diagram of Signin/out



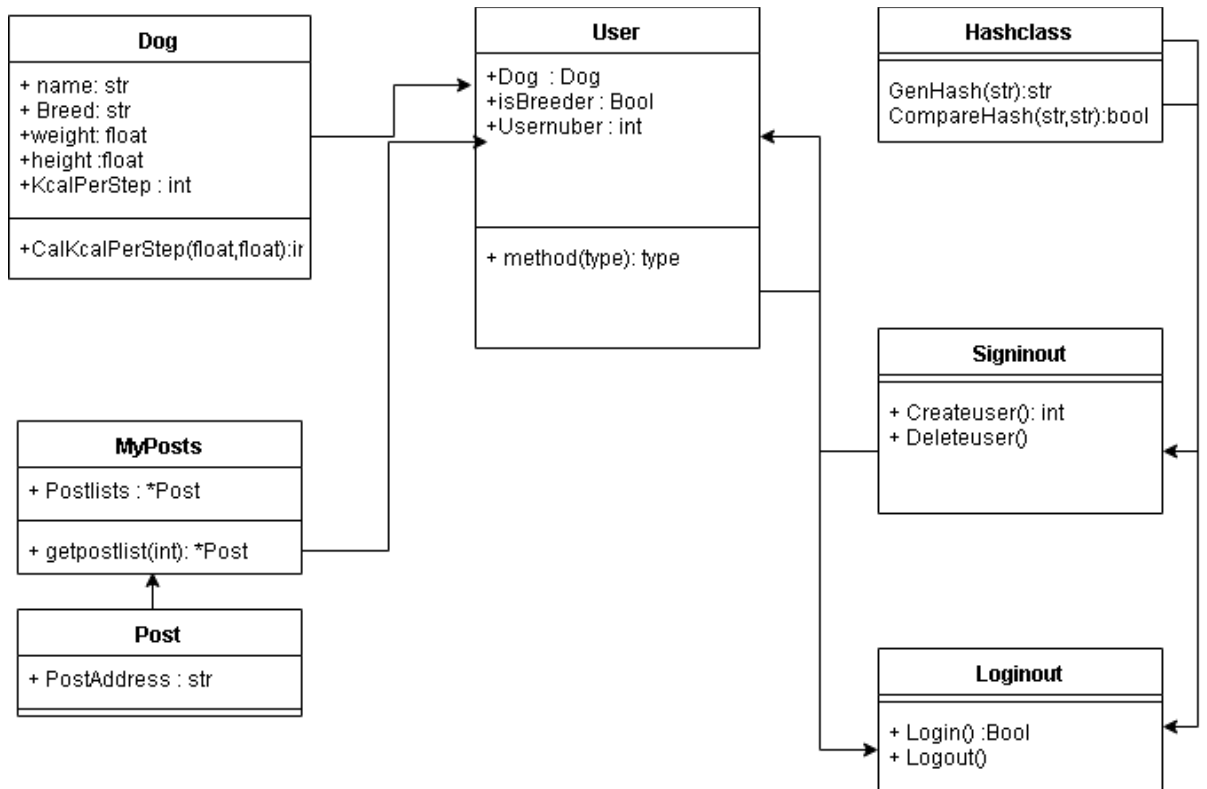
〈그림 3.8 Sequence Diagram of Signin/out〉

## d. Sequence Diagram of Login/out



〈그림 3.9 Sequence Diagram of Login/out〉

## 7.4 Structural Model

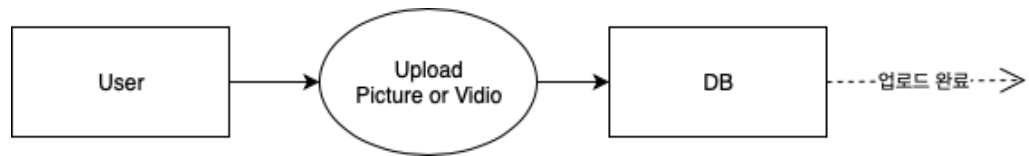


〈그림 3.10 Structural Model〉

## 7.5 Behavioral Models

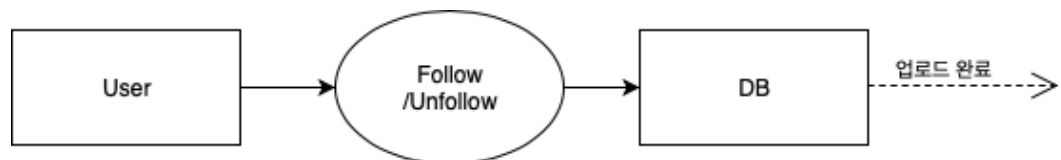
### A. Data-Driven Modeling

#### a. Upload Pictures & Video



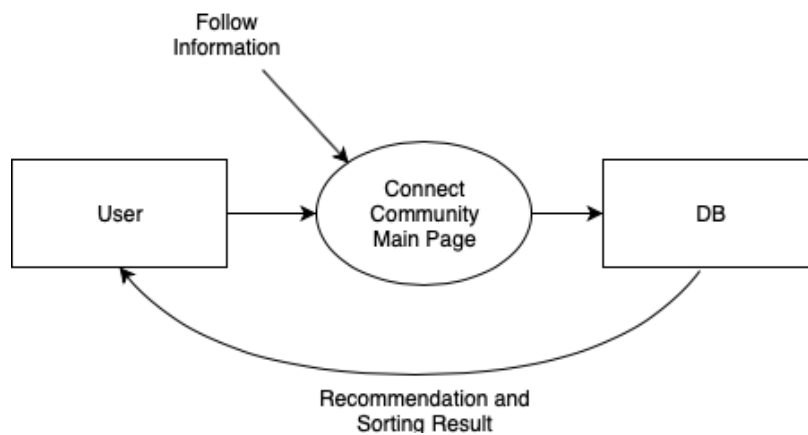
<그림 3.11 Data-Driven Model of Upload Pictures & Video>

#### b. Follow



<그림 3.12 Data-Driven Model of Follow>

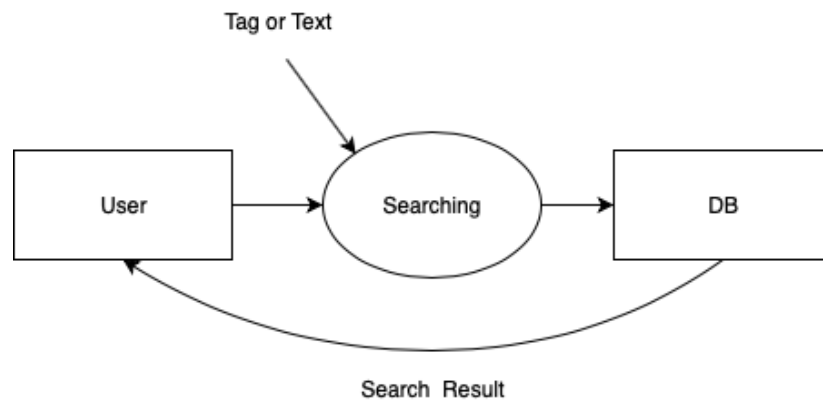
#### c. Showing Main Page



<그림 3.13 Data-Driven Model of Showing Main Page>

#### d. Searching

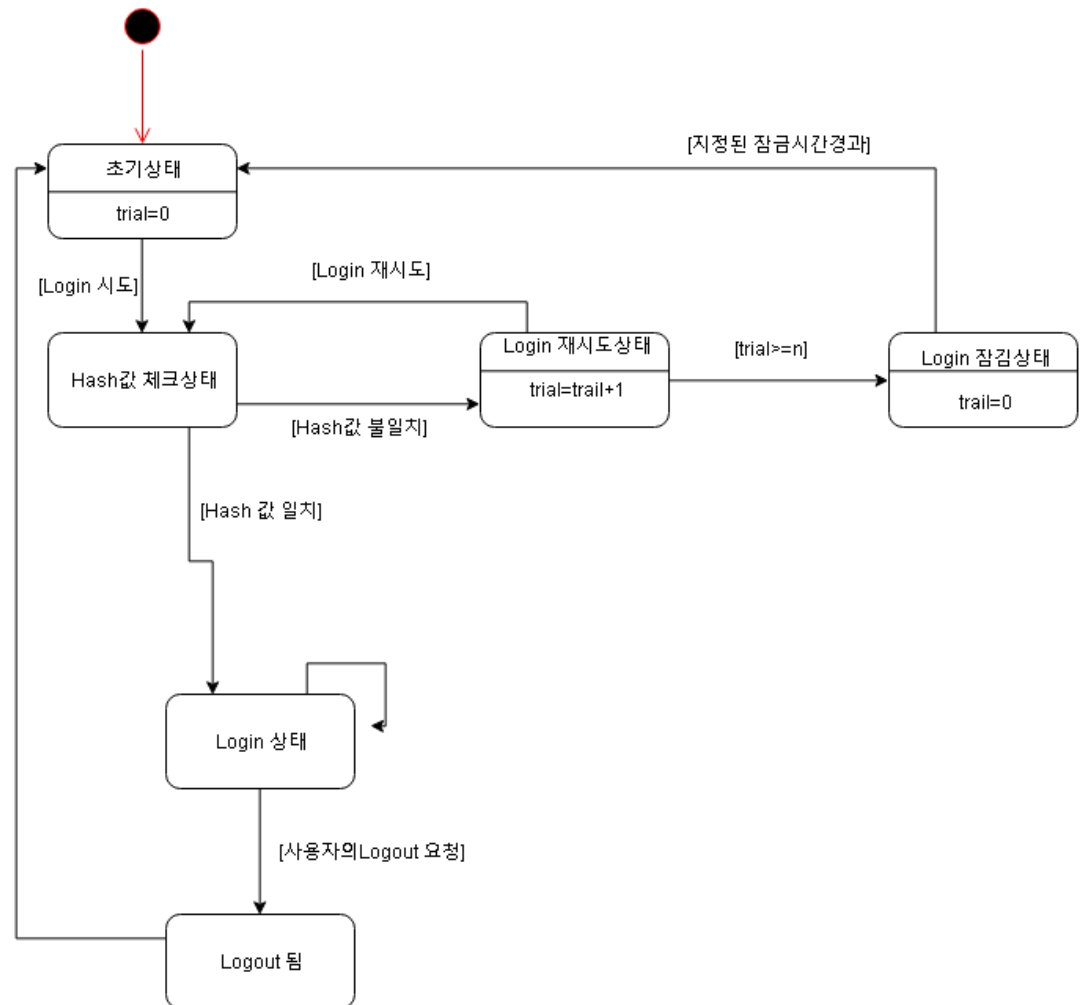




<그림 3.14 Data-Driven Model of Searching>

## B. Event-Driven Modeling

## a. State model of Login system



〈그림 3.15 State Model of Login system〉

## 8. System evolutions

### 8.1 Objective

System Evolution에서는 시스템이 기반하고 있는 주요 가정들과 제약사항들에 대해서 서술한다. 또한 이후에 일어날 것으로 예상되는 하드웨어 및 소프트웨어의 발전, 사용자의 니즈 변화, 사회 환경적 변화들에 대해 설명하며 이후 DOCKING 시스템이 어떻게 발전하고 진화하게 될 지 예측한다. 이러한 예측 과정을 통해, 시스템 발전과 진화의 가이드라인을 구성하며 다양한 사용자 요구사항 변형에 대한 검증과 시스템의 발 빠른 대응 및 적용이 가능한 시스템을 개발할 수 있도록 하는 것에 목표를 둔다.

### 8.2 Limitation and Assumption

앞서 Introduction에서 설명한 것처럼 반려동물 수는 매년 증가하고 있으며, 이에 따라 반려동물 관련 시장이 빠르게 성장하고 있는 추세이다. 또한 저출산, 고령화, 1인가구 증가 등의 사회적 변화로 인해 반려동물 산업은 성장세가 높은 산업 분야로 꼽혔다.

이렇게 반려동물 사업이 확장하는 과정에서, 반려동물의 라이프스타일에 관련한 시장이 전반적으로 성장하게 되었다. 즉 반려동물이 잘 먹고, 잘 자며 살아가는 것을 넘어서 더 퀄리티 있는 삶을 영위할 수 있는 것을 추구하는 방향으로 사회 분위기가 전환하고 있다. 최근 수립된 정부의 반려동물 산업 육성 정책에서는 (1) 생산 및 판매에서의 관리 감독 강화, (2) 반려동물 관련 산업의 건강한 육성, (3) 성숙한 반려동물 문화 정착, (4) 산업육성 인프라 구축 및 일자리 창출 등 4가지 세부 추진 과제를 제시하였다. 동시에, ‘사람과 반려동물의 조화로운 공존’을 주요 비전으로 내걸었다. 이를 통해, 반려동물의 라이프스타일 전반에 대한 관심이 높아지고 있다는 사실을 알 수 있다.

이렇게 반려동물의 삶의 질을 향상시키는 것에 대해 이야기할 때 가장 많이 등장하는 것 중 하나는 ‘반려견과의 산책’이다. 산책은 기본적으로 반려견의 건강과 체력을 유지하도록 해 주고, 좋은 기분을 유지시켜주며, 추가적으로 반려견의 사회화에 도움을 줄 것으로 예상된다. 여기에 이전 문단에서 언급하였던 사회-문화의 변화에 따라 반려견을 키우는 반려인들에게 반려견을 산책시키는 행위는 반려인의 선택이 아닌 필수사항이 되었다. 하지만 피로로 가득 차 본인의 건강도 잘 챙기지 못하기 일쑤인 현대인들이, 매일 일정 시간을 할애하여 산책과 같은 아웃도어 액티비티를 진행하는 것은 사실 어렵다. 항상 산책에 대한 체력과 동기가 높은 상태로 유지되는 반려견과 달리 반려인은 여타 사회생활 등을 병행하고 있을 것이므로 반려견과의 산책 활동이 뒤로 밀릴 수 밖에 없기 때문이다.

이러한 문제에 대해 주 해결책으로 제시되는 것으로는 ‘산책 모임’이 있다. 산책 모임을 결성하고 함께 산책하는 반려인과 함께 반려견과 함께하는 산책의 필요성을

remind하는 과정을 통해 개념을 환기하고 동기를 부여하며, 결과적으로는 반려인 스스로가 산책에 더욱 더 능동적인 자세로 참여하도록 만든다.

비슷한 맥락에서, 같은 주제를 가지는 커뮤니티는 산책에 대한 동기부여에 힘을 실어줄 수 있다. 그저 산책하는 것 이외에, 커뮤니티를 활용하여 다른 반려인 유저들과 소통하고 정보를 알아가는 과정에서 동기를 굳건히 할 수 있게 될 것으로 기대된다.

그래서 DOCKING에서는 반려인들에게 다시 동기부여를 위해 ‘산책 매칭’과 ‘커뮤니티’ 두가지의 해결 방안을 제안한다.

## 8.3 Evolutions of Hardware

### A. 스마트밴드

위에 서술한 것처럼 반려견 시장이 성장하고 있어 DOCKING에서 제공하는 스마트밴드 외에 실제 반려견을 대상으로 한 스마트밴드가 출시될 수 있다. 만약 충분히 사용성이 좋은 스마트밴드가 등장할 경우 DOCKING 시스템과 연동할 수 있도록 해 두어야 한다. 이러한 하드웨어적 진화에 대비하기 위해 DOCKING은 적용성이 뛰어난 시스템을 만들어 두어야 한다.

## 8.4 Evolutions of User Requirement

### A. 반려인의 건강까지 고려하는 시스템



<그림 4.1 스마트밴드>

스마트밴드를 반려견에게 착용하도록 하여 산책에 대한 기록을 할 수 있는 것을 넘어서, 반려동물의 건강 상태 등을 체크하여 사용자의 스마트폰으로 전송하여 열람해 볼 수 있다.

## B. 커뮤니티 기능의 확장, 주기적인 산책 모임의 결성



〈그림 4.2 주기적 산책 모임〉

단순히 서로의 산책을 공유하는 피드를 올리고 댓글로 소통하는 기능 이외에 사용자들 간에 주기적인 산책 모임을 결성하게 될 수도 있다. DOCKING 시스템에서는 이후 사용자들을 매칭해주는 것을 넘어서 그룹 단위의 산책 모임들을 위한 다양한 기능을 제공할 예정이다.

## C. 매칭 시스템 상에서의 필터링 기능의 증진 및 악용 방지



〈그림 4.3 매칭 시스템 보안〉

시스템에서 제시하는 조건을 만족한다면 산책 상대를 매칭해준다는 점을 악용한 사례를 방지하기 위해, 다양한 기준을 도입하여 악의적으로 매칭 서비스를

사용하려고 하는 사용자들을 필터링하여, 시스템의 본 목적에 충실하도록 시스템을 evolution 할 예정이다.

## 8.4 Evolutions of Environment

### A. 반려견 외의 다른 반려동물들에 대한 지원



〈그림 4.4 반려견 외 여타 반려동물들〉

산책을 필요로 하는 반려견을 위주로 매칭 시스템을 제공하지만, 반려견을 넘어 다른 많은 종의 반려동물들을 위한 커뮤니티를 구축한다. 소통 및 정보공유를 주 목적으로 하는 커뮤니티는 정보가 적은 다른 반려동물들에게 더욱 필요하기 때문이다.

## 9. Appendices

### 9.1 Objectives

Appendices에서는 개발되는 시스템에 대한 자세하고 구체적인 정보들을 제공한다. 시스템 사용에 적합한 최소의 설정, 데이터베이스 상에서 결정되는 데이터의 관계, 데이터의 구조에 대해서 설명한다

## 9.2 Database Requirements

Database Requirements에서는 구축하고자 하는 ‘DOCKING’ 데이터베이스의 요구사항을 서술한다. 이하 서술하는 내용들은 테이블 안에 포함되는 Relation의 형태로 저장되어야 한다. 각 시스템 별로, *Attribute*는 이탤릭체, **Entity type**은 **볼드체**, *Relationship*은 **볼드 이탤릭체**, Constraints는 밑줄로 표시한다.

### A. User DB

DOCKING SYSTEM은 **사용자**에 관한 데이터베이스인 User DB를 가진다. 각 사용자는 고유번호(SSN)으로 식별된다. 사용자는 고유번호 외에 ID, PW, 닉네임, 이름, 나이, 성별을 필수로 가지며, 선택적으로 주소, 전화번호를 가질 수 있다.

### B. Contents DB

DOCKING SYSTEM은 **커뮤니티 게시글**에 관한 데이터베이스인 Contents DB를 가진다. 각 게시글은 고유번호(SSN)으로 식별되며, 작성자, 제목, 조회수를 필수로 가진다.

### C. Posts

각 사용자는 **자신의 게시글**에 대한 Posts 데이터베이스를 가질 수 있다. 각 게시글은 고유번호(SSN)으로 식별되며, 제목, 조회수를 필수로 가진다.

### D. Walking

각 사용자는 **자신의 산책 히스토리**에 대한 Walking 데이터베이스를 가질 수 있다. 각 산책 기록은 고유번호(SSN)으로 식별되며, 날짜, 소요시간, 거리를 필수로 가진다. 선택적으로 장소, 날씨정보를 가질 수 있다.

### E. Dog DB

사용자는 **자신의 반려견**에 대한 Dog DB를 가질 수 있다. 각 반려견은 사용자(반려인) 고유번호(SSN)를 가지고 있으며 이것을 식별자로 사용한다. 필수로 반려견 이름, 반려견 추정나이, 반려견 무게, 반려견 성별을 가지며, 추가적으로 품종 정보를 가질 수 있다.

## 10 Index

### 10.1 Table index

<Table 1.1 Version of the Document>	7
<Table 2.1 사용자 관련 용어>	
12	
<Table 2.2 서비스 관련 용어>	
13	
<Table 2.3 개발 관련 용어>	13
<Table 3.1 Sign In/Out Functional Requirements>	24
<Table 3.2 Log In/Out Functional Requirements>	25
<Table 3.3 My Page Functional Requirements>	25
<Table 3.4 Upload Pictures & Videos Functional Requirements>	26
<Table 3.5 Tag Functional Requirements>	26
<Table 3.6 Follow Functional Requirements>	27
<Table 3.7 Searching Functional Requirements>	28
<Table 3.8 Showing Main Page Functional Requirements>	
29	
<Table 3.9 Walking-Mate Matching Functional Requirements>	29
<Table 3.10 Walking-Information Page Functional Requirements>	
30	
<Table 3.11 Dogwatch Functional Requirements>	30

### 10.2 Figure, Diagram index

<그림 1.1 펫코노미와 함께 부상한 반려동물 관련 신조어>	8
<그림 2.1 Overall Architecture>	18
<그림 2.2 User Interface Structure>	19



<그림 2.3 산책 sub-system Architecture>	20
<그림 2.4 커뮤니티 sub-system Architecture>	21
<그림 2.5 User sub-system Architecture>	22
<그림 3.1 Context Model>	37
<그림 3.2 Walking Mate Matching Process Diagram>	38
<그림 3.3 Community and Searching Process Diagram>	39
<그림 3.4 User Process Diagram>	40
<그림 3.5 Use case Diagram>	41
<그림 3.6 Sequence Diagram of Walking System>	42
<그림 3.7 Sequence Diagram of Community System>	43
<그림 3.8 Sequence Diagram of Signin/out>	44
<그림 3.9 Sequence Diagram of Login/out>	45
<그림 3.10 Structural Model>	46
<그림 3.11 Data-Driven Model of Upload Pictures & Video>	47
<그림 3.12 Data-Driven Model of Follow>	47
<그림 3.13 Data-Driven Model of Showing Main Page>	47
<그림 3.14 Data-Driven Model of Searching>	47
<그림 3.15 State Model of Login system>	48
<그림 4.1 스마트밴드>	50
<그림 4.2 주기적 산책 모임>	50
<그림 4.3 매칭 시스템 보안>	51
<그림 4.4 반려견 외 여타 반려동물들>	51