

To.

Team #1 To Dot Requirement Specification

Contents

1. Preface	6
1.1. Objective	6
1.2. Readership	6
A. User Requirements	6
B. System Requirements	6
1.3. Document Contents	6
A. Preface	6
B. Introduction	6
C. Glossary	7
D. User Requirements Definition	7
E. System Architecture	7
F. System Requirements Specification	7
G. System Models	7
H. System Evolutions	7
I. Appendices	8
J. Index	8
K. Reference	8
2. Introduction	9
2.1 Objective	9
2.2. Needs	9
2.3 To Dot(투닷)	10
2.4 Signature	11
2.5 Expected Effect of the Service	12
A. Secret Space	12
B. Quantified Self	12
C. Sustainable Services	12
3. Glossary	13

3.1. Objective	13
3.2. Term Definitions	13
A. User Related Term	13
B. Service Related Term	13
C. Development Related Term	14
D. etc.	14
4. User Requirement Definition	15
4.1. Objective	15
4.2. Functional Requirements	15
A. Sign up	15
B. Login	15
C. Post	15
D. Analysis of Features	16
E. Community	16
F. Send Message	16
G. Evaluation	17
H. Skin	17
4.3. Non-Functional Requirements	17
A. Product requirement	17
A.1. Performance	17
B. Organizational requirement	18
C. External requirement	19
5. System Architecture	20
5.1. Objective	20
5.2. Posting System	20
5.3. Analyzing System	21
5.4. Community System	22
6. System Requirement Specification	23
6.1. Objective	23
6.2. Functional Requirements	23

A. Sign up	23
B. Login	23
C. Posting	24
D. Analysis of Features	24
E. Community	25
F. Send	25
G. Evaluation	26
H. Skin	26
6.3. Non-Functional Requirements	27
A. Product requirement	27
B. Organizational requirement	28
C. External requirement	28
6.4. Scenario	28
A. Join and Login Scenario	28
B. Add a Posting Scenario	29
C. Use Analyzing Service Scenario	30
D. Join a Community Scenario	30
E. Send Message Scenario	31
F. Make an Evaluation Scenario	31
G. Payment Scenario	32
7. System Models	33
7.1. Objective	33
7.2. Context Models	33
A. Context Model	33
B. Process Diagram	34
7.3. Interaction Models	36
A. Use case models	36
B. Tubular description for each use case	37
C. Sequence Diagram	39
7.4. Structural Models	41

A. User System Class Diagram	41
7.5. Behavioral Models	41
A. Data-Driven Modeling	41
B. Event-Driven Modeling	47
8. System Evolutions	49
8.1. Objective	49
8.2. Limitation and Assumption	49
8.3. Evolutions of User Requirement	50
A. Detailed Sentiment Analysis	50
B. Enhanced Filtering Function	51
C. Provision of Various Covers	51
9. Appendices	52
9.1. Objective	52
9.2. Database Requirements	52
9.3. User-System Requirements	52
9.4. Development Process	53
10. Index	54
10.1. Table Index	54
10.2. Diagram Index	55
10.3. Figure Index	56
11. Reference	57

1. Preface

1.1. Objective

서문에서는 본 문서에서 다룰 내용에 대해 간략히 소개한다. User requirements와 System requirements를 병렬 제시하는 것으로 사용자와 개발자 사이에 발생할 수 있는 miscommunication을 최소화 하며, 본 문서의 갱신 일자 및 변경 내용들을 기입하는 것으로 최종 명세서에 대한 근거를 제시한다.

1.2. Readership

A. User Requirements

User requirements는 사용자 중심의 자연어로 기술된 high-level requirements이며 여러 요구 사항들이 복합적으로 결합되어 있는 문서이다. 최종 시스템에 대한 대략적인 개요를 제공하는 데에 사용되며, 실제 개발 단계에 착수하기 위해선 이 User requirements를 상세하게 구분하여 재 정의하는 System requirements가 필요하다.

B. System Requirements

System requirements는 자연어로 기술된 User requirements를 실제 개발 단계에 적용시키기 위해 독립적인 하위 task로 분리하고 각 기능의 경계를 보다 명확히 구분 지어 서술한 문서이다. 이 단계에서, User requirements에서 서술된 요구사항이 누락되지는 않았는지, 혹은 하위 task로 분리하는 과정에서 중복이 발생하지는 않았는지에 대한 검토가 필요하다.

1.3. Document Contents

A. Preface

Preface에서는 본 문서에서 다룰 내용에 대해 간략히 소개한다. User requirements와 System requirements를 병렬 제시하는 것으로 사용자와 개발자 사이에 발생할 수 있는 miscommunication을 최소화 하며, 본 문서의 갱신 일자 및 변경 내용들을 기입하는 것으로 최종 명세서에 대한 근거를 제시한다.

B. Introduction

본 프로젝트의 필요성과 효용성을 제시하고, 타 유사 시스템과 비교하여 어떠한 차별적인 서비스를 제공하는지에 대해 소개한다.

C. Glossary

본 문서에서 사용하는 기술적인 용어에 대하여 정의한다. 가능한 한 모든 용어를 서술해야 하며, 독자의 배경 지식에 관계 없이 쉽게 이해할 수 있도록 작성한다. 용어의 혼동 여지가 있다고 판단될 경우 부가적인 설명을 덧붙여 요구사항에 대한 독자의 이해를 도울 수 있어야 한다.

D. User Requirements Definition

사용자의 요구사항에 대한 분석 및 서술이다. 자연어로 서술된 요구사항을 구체화 하는 단계로, System requirements와 같은 수준의 정교한 명세화는 아니지만 사용자와 개발자가 납득할 수 있는 수준으로 요구사항을 정제하여 서술한다. 이해를 돕기 위해 diagram 등을 사용할 수 있으며, Functional requirements와 Non-functional requirements를 구분하여 서술한다.

E. System Architecture

본 프로젝트의 전반적인 시스템 구조에 대하여 설명한다. 시스템의 굵직한 기능들과 해당 기능의 구현 방법 등에 대해 서술한다.

F. System Requirements Specification

User Requirements Definition에서 분석한 Functional requirements와 Non-functional requirements에 대하여 보다 구체적인 서술을 제시한다. 각 기능 별로 예상되는 입력과 출력 및 처리 과정에 대하여 기술한다. 중심 기능들에 대한 Scenario를 작성하여 사용자와 개발자의 기능 이해를 돕는다.

G. System Models

System Models에서는 System Component, System Environment 등에 대한 도식적인 이해를 제공한다. 각 기능의 입력/출력 결과 및 처리 절차 혹은 DB와의 상호작용 과정 등에 대한 시각적 자료를 제시하며, 차트나 표를 이용하여 이에 대한 설명을 보완하여 제공한다.

H. System Evolutions

System Evolutions에서는 본 프로젝트의 필요성을 뒷받침하는 근거들에 대하여 다시 한 번 설명하고 이를 기반으로 예상 가능한 변화에 대하여 서술한다. 또, 이러한 논의에 근거하여 설계 단계에서 Change avoidance와 Change Tolerance를 높일 수 있는 방법에 대한 비전을 제공한다.

I. Appendices

Appendices에서는 본 프로젝트와 관련한 보다 자세하고 구체적인 정보를 제공한다. Database requirements, User-system requirements, Development process 등에 관련한 추가적인 정보를 설명한다.

J. Index

Index에서는 문서에 포함된 테이블과 그림, 다이어그램의 인덱스가 포함된다.

K. Reference

Reference에는 본 요구사항 명세서에서 참고한 것들을 명세한다.

2. Introduction

2.1 Objective

Introduction에서는 시스템의 필요성과 해당 시스템이 어떤 Needs를 반영하고 있는지 설명한다. 또한 시스템의 기능과 다른 시스템과의 상호작용을 설명하고 어떻게 시스템의 목적을 성취하는지 설명한다.

2.2. Needs

사람들은 자신의 은밀한 문제나 고민을 공유하고 싶어하지만, 비밀스럽게 공유하고 싶어한다. 가령, 심리상담이나 정신과 상담의 철저한 비밀유지가 원칙인 이유이다. 이러한 상담을 부담스러워하거나, 문제가 그다지 무거운 문제가 아니라면, 익명 게시판을 이용하곤 한다. 최근 이런 익명 게시판으로 대나무숲이라는 페이스북 페이지가 있다.

대나무숲 페이지는 고등학교, 대학교, 직장, 지역 등의 단위로 구분되어 개설되어 있다. 자발적으로, 한 단위의 구성원이 페이지를 만드는 경우가 대부분이며, 그 구성원이 페이지를 운영한다. 대나무 숲의 이용률은 높은 편인데, 각 대학의 대나무 숲 이용자를 살펴보면, 성균관대학교의 경우 6만명, 고려대의 경우에는 33.5만명에 달한다.

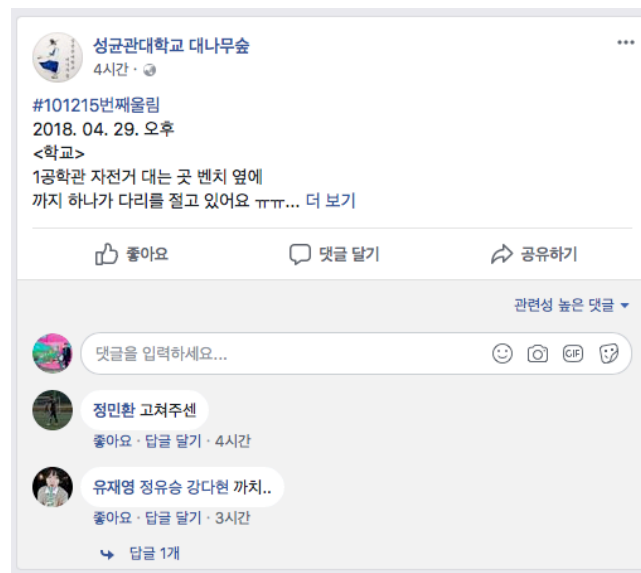


Figure 1 성균관대학교 대나무숲

익명의 글쓰기가 대나무숲에 글을 올리면, 익명이 아닌 댓글들이 달리게 되고, 어떠한 공감대가 형성된다. 그 공감대는 글의 성격에 따라 글에 대해 공감하는 공감대일 수도 있고, 글에 대해 반대하는 공감대가 형성되기도 한다. 이렇게 문제를 공론화 하면서 문제에 대한 많은 피드백, 위로 등을 받을 수 있다는 순기능이 있다.

하지만 여러 문제점들이 존재한다. 페이스북에 특성상, 글에 대한 댓글이 좋아요가 많은 순으로 정렬된다. 따라서, 제일 좋아요를 많이 받은 댓글, 이른바 베스트 댓글이 글에 대한 여론을 형성하게 된다는 점이다. 그리고 그 여론에 대해 반하는 댓글을 남기는 사람은 비난을 받기 쉽다. 여론에 지나치게 휩쓸리는 경향, 침묵의 나선이론이 작용하는 지점이다. 이러한 익명성이 불러올 수 있는 부작용을 해결하기 위해 관리자의 필터링 기능이 존재한다. 하지만 필터링 기준이 불명확하고, 익명이라는 대나무숲의 취지에 필터링을 적용하기 어렵다고 주장하는 의견도 있다. 익명성이 가져오는 문제와, 그러한 문제들을 해결하기 어렵다는 문제로 대나무숲을 이용하지 않는 사람들이 더 많을 수 밖에 없다. 더욱 관점을 확장시켜서, SNS 자체를 이용하지 않아 대나무숲을 이용하지 않는 사람들도 많다.

SNS를 이용하지 않는 사람들은 타인에게 자신의 정보를 노출하고 싶지 않고, SNS를 통해 얻게 되는 무분별하고, 편파적이고, 과격한 많은 정보에 지쳤다. 또한 우리 팀이 관심있게 지켜보는 고민이 있는 사람들은, 그 고민을 SNS에 공개했을 때 댓글로 인해 상처를 받기 쉽기 때문에 SNS를 이용하지 않는다.

따라서, 이러한 기존 익명 커뮤니티나 SNS의 문제를 해결하기 위해 선택적 공유 기능의 SNS가 필요하다. 자신만의 독립된 공간을 웹에서 확보하고, 고민을 정리해보고, 원한다면 다른 사람에게 공유하는 서비스가 이 프로젝트의 목표이다.

2.3 To Dot(투닷)

선택적 공유 기반의 To Dot(투닷) 서비스는 세가지의 기능을 제공한다.

1. 심플한 UI로 쉬운 기록
2. 기록한 내용에 대한 정량적인 분석 결과 제공
3. 커뮤니티 기능을 제공하여 작성한 글을 익명으로 공유하고 피드백 받기

심플한 UI는 사용자가 글을 쓰는 과정에서 피로를 느끼지 않도록 하기 위함이다. 자칫 자신의 생각이나 고민에 대해서 글로 풀어내는 과정이 귀찮다고 느끼기 쉬운데, 본 서비스에서는 간편하고 손쉽게 글을 쓸 수 있도록 할 것이다.

분석 결과 시스템은 사용자가 작성한 글에 대해서 감정, 빈도수 등의 분석 결과를 제공한다. 사용자가 작성한 글에 대해서 화남, 기쁨, 슬픔 등의 대략적인 감정 분석을 제공한다. 과거의 남긴 글을 분석해, 지정한 기간 동안 감정이 어땠는지, 글을 쓰는 행위로 감정이 해소되었는지 등의 효과를 가져올 수 있다.

커뮤니티 시스템은 원하는 사용자에게 한해, 글에 대해 선택적 공유 기능을 제공한다. 글을 쓰고 혼자만 읽는 것에 만족하지 못하는 사용자가 글을 다른 사람에게 공유하면서 피드백을 받길 원한다면 커뮤니티 기능을 활용할 수 있다.

2.4 Signature

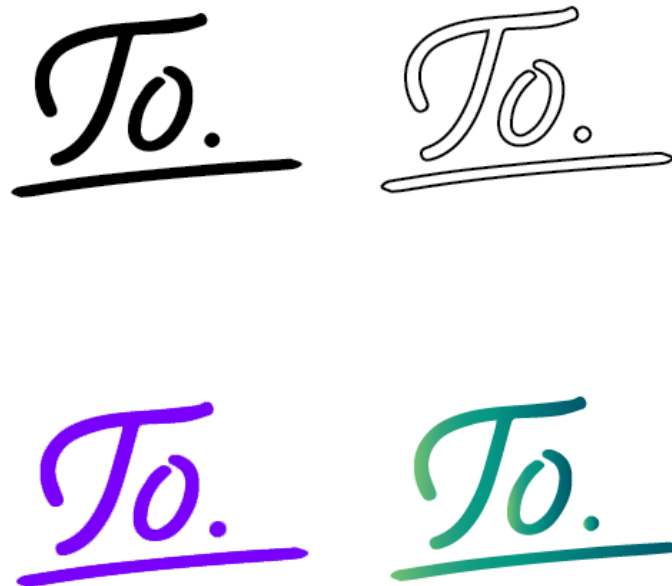


Figure 2 To Dot Signature

To Dot(투닷) 시스템의 시그니처이다. 익명의 상대방에게 편지를 보내듯 적어내려가는 일기장처럼, 받는 이가 비어있는 편지를 쓰는 행위를 글자체로 담았다. 시그니처의 색상은 스트로크, 컬러, 그라디언트 등 자유롭게 변형될 수 있으며, 다양한 유저가 모두 자신만의 글을 기록할 수 있는 공간을 가질 수 있도록 하는 To Dot(투닷)의 이념을 담고 있다. 이러한 시그니처를 따름으로써 Brand Identity를 지킬 수 있다.

2.5 Expected Effect of the Service

A. Secret Space

나만의 글을 기록할 수 있는 공간이다. 선택적 공유 기능으로 공개 여부 설정이 가능하다. 글을 쓰는 행위로 사용자가 가지고 있던 고민이나 생각을 정리할 수 있다. 또한 단순히 글을 쓰는 것만으로 정리가 되지않는다면, 공유기능을 활용해 다른 사용자에게 피드백을 받고 정리할 수 있다.

B. Quantified Self

객관적인 Sentiment Analysis을 통해 정량적으로 자신의 감정을 파악할 수 있다. 화남, 기쁨, 슬픔등의 간단한 감정 분석을 제공한다. 또한, 과거의 기록해놓은 글 들의 분석함으로써, 한 주간, 한 달간 감정이 어떠했는지 분석해서 제공한다. 지정해 놓은 기간동안 글을 얼마나 썼는지의 빈도수 등을 제공한다.

C. Sustainable Services

감성적인 UI 및 악의적인 피드백을 차단하여 지속적인 서비스 사용을 유도한다. 감성적인 유저 인터페이스로 글을 쓰는 느낌을 극대화 하고, 심플한 유저 인터페이스로 글을 쓰는 행위가 귀찮게 느껴지도록 하지 않게 한다. 또한 글을 공유했을 경우 신고를 통해 악의적인 피드백을 하는 사용자를 차단하도록 한다.

3. Glossary

3.1. Objective

본 문서에서 사용하는 기술적인 용어에 대하여 정의한다. 가능한 한 모든 용어를 서술해야 하며, 독자의 배경 지식에 관계 없이 쉽게 이해할 수 있도록 작성한다. 용어의 혼동 여지가 있다고 판단 될 경우 부가적인 설명을 덧붙여 요구사항에 대한 독자의 이해를 도울 수 있어야 한다.

3.2. Term Definitions

A. User Related Term

Terms	Definition
사용자	To dot을 이용하는 모든 사용자.
답변자	공유된 글에 답변(쪽지)를 보내는 사용자.
평가	답변(쪽지)를 보낸 사람에 대한 수신자의 평가. '성실한 답변', '무관한 답변' 등의 선택지를 두어 송신자에 대한 피드백 및 악의적인 사용자를 색출할 수 있는 기능.

Table 1 사용자 관련 용어

B. Service Related Term

Terms	Definition
로그인	To Dot(투닷) 웹 어플리케이션에 접속하는 절차.
포스팅	작성한 글을 최종 저장하여 개인 사용자의 글 목록에 추가하는 작업. 작성한 글이 공유되는 것이 아니라는 점에서 일반적인 SNS에서의 '포스팅'과 명백한 의미상의 차이를 가짐.
커뮤니티	각 사용자의 글을 익명으로 공유할 수 있는 일종의 게시판.
공유	개인 사용자의 글을 To Dot 서비스 내 커뮤니티에 게시하는 과정.
쪽지	커뮤니티에 게시된 글에 대한 답변. 댓글이 아닌 메일에 가까운 형식.
차단	To Dot 서비스 내의 악의적인 사용자를 걸러낼 수 있는 기능.
감정 분석	글마다 사용된 단어를 분석하여 각 글에 담긴 감정 요소에 대한 통계치 제공.
스킨	사용자가 각 글에 담긴 느낌이나 기분을 표현할 수 있게 하는 표지.

Table 2 서비스 관련 용어

C. Development Related Term

Terms	Definition
서버(Server)	요청 받은 서비스를 응답하는 프로세스 및 하드웨어와 관련된 서브시스템
데이터베이스(Database)	요청 받은 서비스를 제공할 수 있도록 통합하여 관리되는 데이터의 집합
깃허브(Github)	분산 버전 관리 툴인 깃(Git)을 사용하는 프로젝트를 지원하는 웹호스팅 서비스
프레임워크(Framework)	소프트웨어의 개발에 필수적이고 표준적인 부분에 해당하는 설계와 구현을 재사용 가능하도록, 일련의 협업화 된 형태의 클래스들로 제공하는 반제품 소프트웨어 모듈
API	운영체제와 응용프로그램 사이의 통신에 사용되는 언어나 메시지 형식

Table 3 개발 및 연산 관련 용어

D. etc.

Terms	Definition
소셜네트워크서비스(SNS)	특정한 관심이나 활동을 공유하는 사람들 사이의 관계망을 구축해 주는 온라인 서비스
대나무숲	트위터, 페이스북 등 SNS상에서 익명으로 소통하는 게시판

Table 4 기타 관련 용어

4. User Requirement Definition

4.1. Objective

User Requirement는 사용자에게 제공되는 서비스를 기능적, 비기능적 부분으로 설명한다. 자연어와 다이어그램을 사용하여 설명하며, 시스템은 4에 서술된 요구사항들을 반드시 충족한다. 또한 프로세스가 반드시 따라야 하는 표준에 대해서도 서술한다.

4.2. Functional Requirements

To Dot(투닷)은 다음과 같은 기능을 제공하여야 한다.

A. Sign up

사용자에게 회원 가입을 위해 필요한 정보를 요청하는 기능이다. 사용자는 To Dot(투닷)에서 제공하는 서비스를 이용하기 위해 회원 가입을 해야한다. 단, 익명성을 보장하는 서비스이므로 중복 가입을 방지하기 위한 최소한의 정보를 수집한다.

B. Login

사용자에게 회원 가입을 통해 생성된 계정을 이용하여 시스템에 로그인하는 기능이다. 사용자는 To Dot(투닷)에 가입된 계정을 이용해 로그인을 통해 자신의 데이터베이스에 접근하고, 시스템에서 제공하는 서비스를 이용할 수 있다.

C. Post

사용자가 개인 페이지에 새로운 글을 작성하거나 기존의 글을 수정, 삭제할 수 있는 기능이다. 기존에 존재하는 SNS에 대해 사용자는 광고와 원치 않는 과도한 정보, 복잡한 기능 등으로 인해 피로감을 느꼈을 것이다. 따라서 간단한 UI와 최소한의 기능을 사용하여, 사용자가 간편하게 서비스를 이용할 수 있도록 한다.

새로운 글을 작성할 때 사용자는 제목과 내용을 입력한다. 기본 스킨과 공유하지 않음이 기본값으로 설정되어 있으며, 원할 경우 스킨을 선택하고 공유를 설정할 수 있다.

D. Analysis of Features

사용자가 시스템에서 제공되는 정량적 분석 결과를 확인할 수 있는 기능이다. 분석 시스템은 키워드 분석 결과, 감정 분석 결과, 키워드 기반 커뮤니티 글 추천 기능 등을 제공한다. 사용자가 분석 탭에 접근할때마다 분석 결과를 업데이트하고, 글의 수정이나 삭제와 같은 변경이 일어날 때도 결과를 반영하도록 한다.

E. Community

사용자들 간 커뮤니티를 통해 공감대를 형성하고 고민을 나눌 수 있는 기능이다. 커뮤니티에 글을 공유할지 선택할 수 있다는 점은 타 서비스와 차별되는 To Dot(투닷)의 장점이다. 커뮤니티의 모든 글은 익명으로 보여지고, 공유 된 날짜와 시간 순서에 따라 최신의 글이 상위에 노출된다. 커뮤니티는 댓글 기능을 제공하지 않고, 쪽지를 주고받을 수 있다.

F. Send Message

사용자가 커뮤니티를 통해 공유된 글의 작성자에게 쪽지를 보낼 수 있는 기능이다. 간편한 댓글 기능을 사용하지 않고 쪽지를 사용하게 함으로써 동조 현상과 가볍게 다는 악의적인 댓글을 줄일 수 있다는 장점이 있다. 사용자는 쪽지 기능을 통해 해당 글의 작성자에게 공감의 글이나 조언, 응원 등의 글을 보낼 수 있고, 수신자는 이에 대해 답장을 보낼 수 있다. 이 때 발신자와 수신자는 아이디가 아닌 무작위로 생성된 익명의 이름을 사용하게 된다.



Figure 3 Expected Sharing Process

G. Evaluation

악의적인 쪽지를 작성하는 이용자를 차단하기 위한 기능이다. 사용자는 수신된 쪽지에 대해 긍정이나 부정의 평가(차단)를 내릴 수 있으며, 그 평가는 발신자에게는 보이지 않는다. 발신자를 차단하면 더 이상 해당 사용자로 부터의 쪽지가 보이지 않고, 차단을 일정 횟수 이상 받은 이용자는 관리자의 확인 후 쪽지 기능의 사용을 제재 받게 된다.

H. Skin

사용자가 원하는 스킨을 구매하고 사용할 수 있는 기능이다. To Dot(투닷)은 감성적인 일러스트, 포맷(원고지 등), 사진 등의 스킨을 제공하여 사용자는 작성하려는 글이나 쪽지에 맞는 분위기의 스킨을 적용할 수 있다. 구매 시 외부의 전자결제시스템이 작동한다.

4.3. Non-Functional Requirements

Non-Functional Requirements는 Product requirement, Organizational requirement, External requirement로 분류된다.

A. Product requirement

A.1. Performance

해당 시스템은 사용자가 글을 작성한 후 개인 페이지에서 바로 확인할 수 있도록 해야 한다. 또한 내용의 누락이 없어야 하며, 원하는 경우에만 커뮤니티에 글이 공유되도록 한다. 시스템은 사용자가 기존에 작성한 글을 누락하지 않고 개인 페이지에서 확인할 수 있도록 제공해야 한다. 또한 기존의 글을 수정하거나 삭제한 결과가 개인 페이지는 물론 커뮤니티에 즉시 반영되어야 한다. 커뮤니티에서 쪽지를 주고받을 때, 수신자와 발신자의 정보에 오류가 없어야 한다. 쪽지를 주고받거나 글을 작성할 때 글의 길이, 포맷에 상관없이 스킨이 잘 적용되어야 하고, 구매한 결과와 사용한 결과가 즉시 반영되어야 한다.

A.2. Security

해당 시스템은 개인적인 글을 작성하고 보관하며, 익명의 커뮤니티를 통해 소통하는 서비스를 제공하기 때문에 사용자의 계정 정보가 유출되지 않도록 해야 한다. 사용자의 비밀번호는 암호화되어 관리자도 확인할 수 없도록 한다. 커뮤니티에서 글을 공유하거나 쪽지를 주고받을 때는 임의로 생성된 별명을 사용하도록 한다. 또한 전자 결제 시스템의 경우 보안성과 신뢰성이 높은 외부 시스템을 사용하여 고객의 결제 정보가 유출되지 않도록 한다.

A.3. Efficiency

해당 시스템은 웹 환경에서 동작한다. 기존에 작성한 글을 조회하거나 보유/구매 가능한 스킨을 조회할 때, 커뮤니티에서 글을 조회할 때 등의 상황에서 정보를 불러오는 대기 시간이 길어지지 않도록 한다. 이는 데이터베이스를 효율적으로 구축하고, 불필요한 폰트나 이미지의 사용을 자제하여 웹페이지를 전체적으로 가볍게 개발함으로써 구현할 수 있다.

A.4. Dependability

B. A.4. Dependability 해당 시스템은 서버와 지속적인 통신을 전제한다. 서버와의 통신이 끊어지면 작성한 글이나 수정 및 변경 사항, 스킨의 구매 및 사용 내역이 제대로 반영되지 않아 문제가 될 수 있다. 따라서 안정적인 서버를 구축하여 글의 작성이나 구매 내역에 오류가 없도록 한다. 또한 정확도가 높은 자연어 처리와 감성 분석 API를 사용하여, 사용자에게 신뢰도가 높은 분석 결과를 제공한다.

A.5. Usability

해당 시스템은 간단한 UI를 제공하여 사용자들이 서비스를 이용하는 것에 불편함이 없도록 한다. 직관적인 탭 이동과 버튼 배치 등을 통해 시스템의 기능을 제공한다.

B. Organizational requirement

B.1. Environment

해당 시스템은 부트스트랩을 사용한 반응형 웹으로 개발되어, 사용자의 환경에 최적화된 화면을 제공한다. 반응형 웹의 경우 지원하지 않는 브라우저의 버전도 있으므로, 이를 사용자에게 명시한다.

B.2. Operation

해당 시스템은 사용자 개인의 페이지와 커뮤니티 페이지를 구분하여, 원하지 않는 경우 글의 공유를 제한한다. 사용자 간의 소통은 커뮤니티 페이지를 통해 쪽지와 그에 대한 답장으로만 이루어진다. 쪽지의 수신 및 발신은 사용자의 아이디 대신 임의로 생성된 별명을 사용하여 서로를 인식할 수 있는 최소한의 기능을 제공한다.

C. External requirement

C.1. Safety / Security

해당 서비스는 nodeJS를 사용하여 구축된 서버로 시스템을 관리한다. 서버 관리자는 해당 서비스에 회원 가입하여 시스템을 이용하는 사용자들을 관리하고, 악의적인 접근을 막는다. 또한 전자 결제 시스템의 경우 보안성과 신뢰성이 높은 외부 시스템을 사용하여 고객의 결제 정보가 유출되지 않도록 한다. 전자 결제 시스템이 요구하는 정보를 구매자에게 요청하고, 이를 전자 결제 시스템의 형식에 맞게 보내어 오류가 발생하지 않도록 한다.

5. System Architecture

5.1. Objective

System Architecture에서는 목표 시스템의 Architecture에 대한 고수준에서의 개요와 시스템 기능의 전체적인 분포 및 컴포넌트들의 관계를 명세한다.

5.2. Posting System

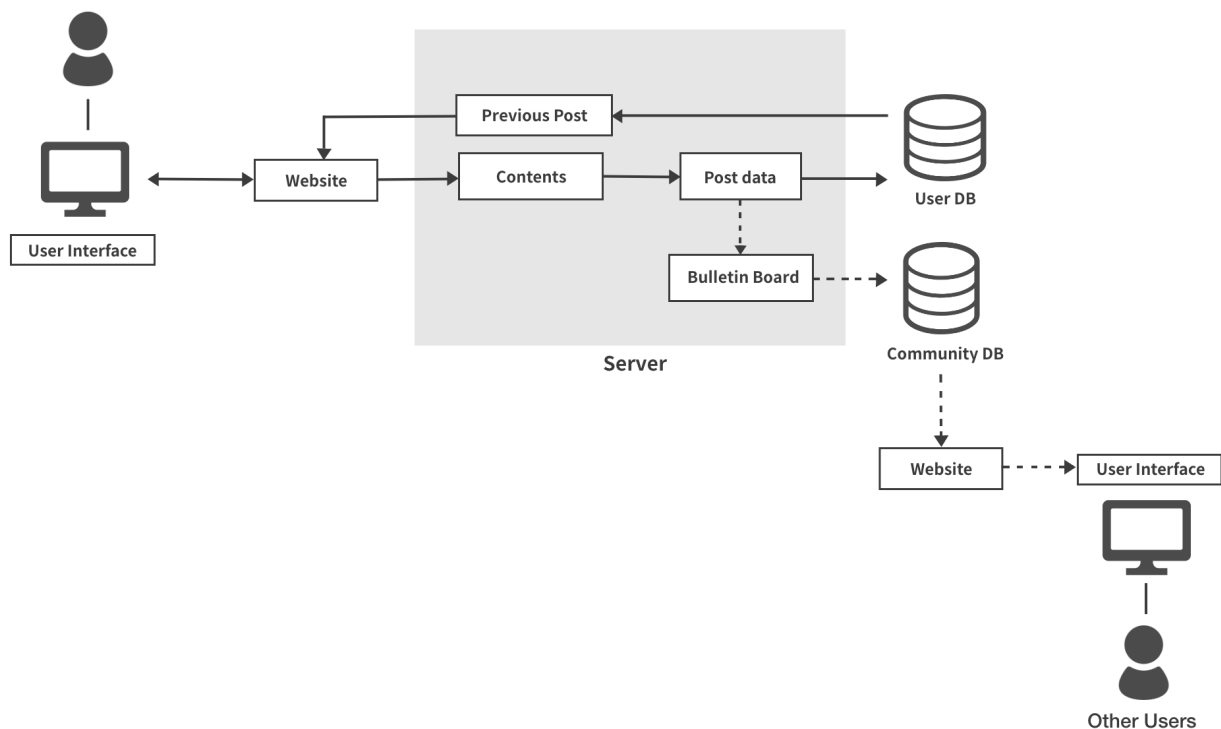


Diagram 1 Posting System

Post 시스템은 사용자의 개인적인 공간에 글을 작성할 수 있는 시스템이다. 데이터베이스에서 기존에 작성한 Post가 제공되어 사용자가 과거의 글을 열람할 수 있으며, 새로운 글을 작성하면 사용자의 데이터베이스에 저장된다. 이 때 스킴 사용 설정 및 커뮤니티 공유 여부를 선택할 수 있다. 공유가 가능하도록 설정하면 커뮤니티 데이터베이스에도 저장되어 다른 사용자들이 글을 볼 수 있게 된다.

5.3. Analyzing System

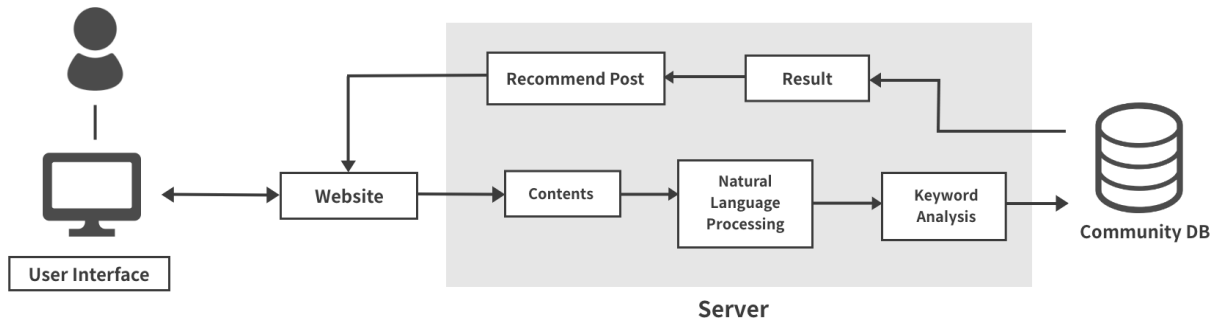


Diagram 2 Analyzing System

사용자가 작성한 글을 자연어처리를 통해 키워드를 분석하고, 커뮤니티에 있는 비슷한 키워드의 글을 사용자에게 추천해준다.

5.4. Community System

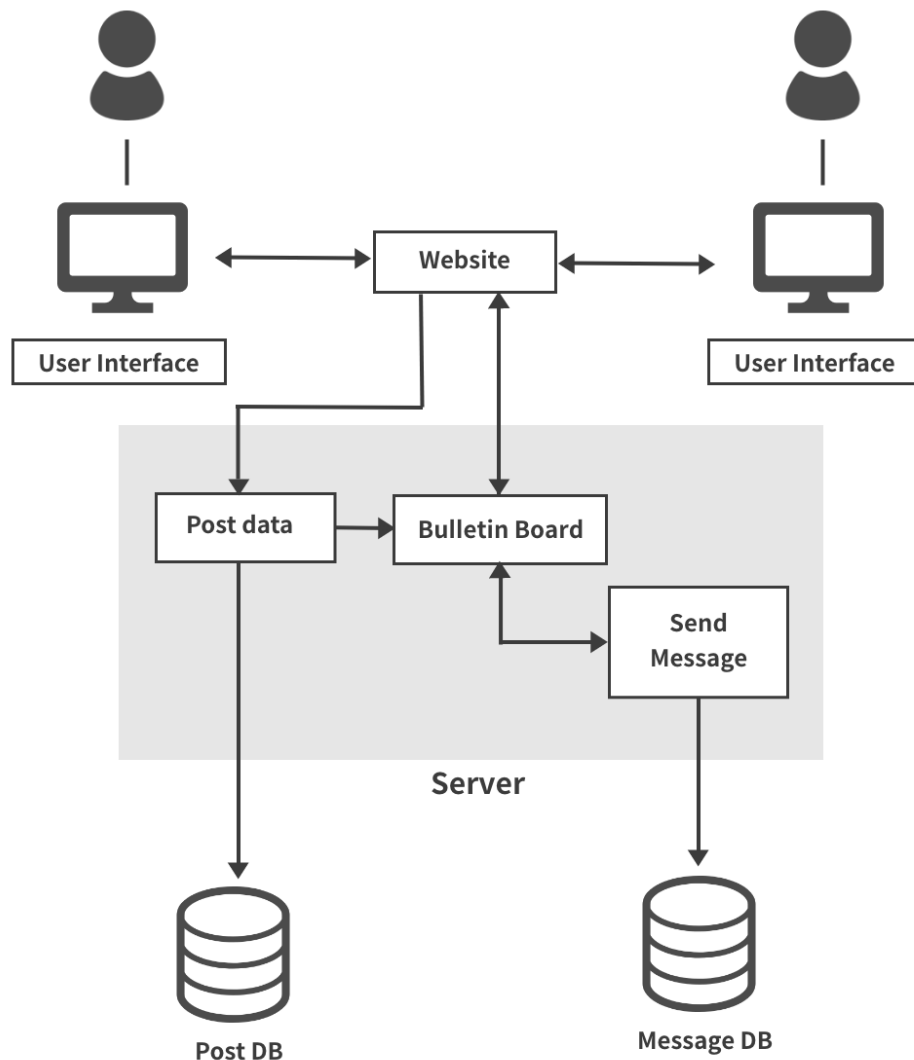


Diagram 3 Community System

커뮤니티 시스템은 공유된 글을 게시판 형식으로 정렬하여 모든 사용자가 읽을 수 있도록 제공하는 시스템이다. 각 게시글마다 쪽지 기능이 있어 사용자는 게시글 작성자에게 답장을 보낼 수 있다.

6. System Requirement Specification

6.1. Objective

시스템 요구사항에서는 기능적 요구사항 또는 비기능적 요구사항에 대해 설명하고, 시스템을 이용하는 것에 대한 시나리오를 다룬다.

6.2. Functional Requirements

A. Sign up

기능적 요구사항	
기능	회원가입
설명	사용자에게 회원 가입을 위해 필요한 정보를 요청하는 기능이다. 사용자는 To Dot(투닷)에서 제공하는 서비스를 이용하기 위해 회원 가입을 해야한다. 단, 익명성을 보장하는 서비스이므로 중복 가입 방지와 비밀번호 찾기 등을 위한 최소한의 정보만을 수집한다.
입력	사용자로부터 메일과 아이디, 비밀번호를 입력받는다. 메일과 아이디의 경우 이미 가입된 계정이 있는지 중복 확인을 하고, 비밀번호의 경우 두 번 입력하는 것을 통해 사용자의 실수를 방지한다.
출력	사용자가 가입을 완료한 경우, 회원가입이 완료되었음을 알리는 창을 출력한다.
처리	사용자가 가입을 완료한 경우, 시스템은 그 정보를 데이터베이스에 저장한다.

Table 5 Sign up Function

B. Login

기능적 요구사항	
기능	로그인
설명	사용자에게 회원 가입을 통해 생성된 계정을 이용하여 시스템에 로그인하는 기능이다. 사용자는 로그인을 통해 자신의 데이터베이스에 접근하고, 시스템에서 제공하는 서비스를 이용할 수 있다.
입력	사용자로부터 아이디와 비밀번호를 입력받는다.
출력	로그인 정보가 일치하는 경우, 사용자의 개인 페이지로 이동한다. 로그인 정보가 일치하지 않는 경우, 로그인 정보가 틀렸음을 알리는 팝업창을 출력한다.
처리	시스템은 사용자가 입력한 정보를 데이터베이스에서 조회하여, 일치하는 경우 해당 계정의 개인 페이지로 이동하고, 일치하지 않는 경우 로그인 정보가 틀렸음을 알리는 팝업창을 출력한다.

Table 6 Login Function

C. Posting

기능적 요구사항	
기능	글 작성
설명	사용자가 개인 페이지에 새로운 글을 작성하거나 기존의 글을 수정, 삭제할 수 있는 기능이다.
입력	새로운 글 작성을 원할 경우 사용자는 초기 화면(개인 공간)에서 글 작성 버튼을 클릭하고, 제목과 내용, 그리고 원할 경우 스킨을 선택하여 글을 작성한다. 기존의 글을 수정, 삭제하려면 사용자는 해당 글을 선택하여 작업을 수행한다. 새 글 작성 또는 수정 과정에서 해당 글을 커뮤니티에 공유할지를 선택할 수 있다.
출력	새 글 작성 또는 수정 과정에서 제목이나 내용 값이 없으면 팝업창을 띄워 알린다. 시스템은 삭제 기능을 실행하기 전에 팝업창을 띄워 다시 한 번 사용자의 확인을 받는다.
처리	시스템은 새로운 글이 게시된 경우, 그 내용과 제목, 작성 날짜 등을 데이터베이스에 저장한다. 시스템은 글이 수정된 경우 해당 게시글에 수정된 내용을 데이터베이스에 업데이트한다. 시스템은 글이 삭제된 경우, 해당 글을 데이터베이스에서 지운다. 만약 사용자가 글을 커뮤니티에 공유 할 경우, 익명으로 공유되고, 모든 사용자들이 커뮤니티에서 해당 글을 확인할 수 있다.

Table 7 Posting Function

D. Analysis of Features

기능적 요구사항	
기능	분석
설명	사용자가 시스템에서 제공되는 정량적 분석 결과를 확인할 수 있는 기능이다.
입력	시스템은 각 사용자별 분석 결과를 제공하고, 사용자는 자신이 작성했던 글에 대한 분석 결과를 확인할 수 있다. 사용자는 개인 페이지의 분석 결과 탭을 눌러 분석 결과를 열람할 수 있다.
출력	시스템은 키워드 분석 결과, 감정 분석 결과, 키워드 기반 커뮤니티의 글 추천 등을 출력한다.
처리	시스템은 자연어 처리와 감성 분석 오픈소스, 구글 차트 API 등을 활용하여 분석 결과를 도출한다. 사용자가 분석 서비스를 이용할때마다 분석 결과를 업데이트하고, 글의 수정/삭제와 같은 변경 또한 분석 결과에 반영한다.

Table 8 Analysis of Features Function

E. Community

기능적 요구사항	
기능	커뮤니티
설명	사용자가 커뮤니티를 통해 공유된 글을 확인할 수 있는 기능이다. 커뮤니티의 모든 글은 익명으로 보여지고, 공유된 날짜와 시간 순서에 따라 최신의 글이 상위에 노출된다.
입력	사용자는 커뮤니티를 사용하기 위해 '커뮤니티' 탭을 클릭하여 이동한다. 스크롤을 이동하며 글을 미리보기로 확인할 수 있고, '더 보기'를 클릭하면 전체 글을 확인할 수 있다.
출력	시스템은 커뮤니티에 공유된 순서대로 글을 출력한다. 커뮤니티의 글은 페이스북의 뉴스피드와 같은 방식으로 보여지고, 제목과 다섯 줄 가량의 글을 미리보기로 볼 수 있으며 '더 보기'를 누르면 전체 글을 확인할 수 있다.
처리	시스템은 데이터베이스에 커뮤니티의 글에 대한 정보를 저장한다. 만약 사용자가 커뮤니티에 글을 공유할 경우, 해당 글을 커뮤니티에 익명을 노출시키고, 수정이나 삭제가 일어난 경우 이를 즉시 반영한다.

Table 9 Community Function

F. Send

기능적 요구사항	
기능	쪽지 보내기
설명	사용자가 커뮤니티를 통해 공유된 글의 작성자에게 쪽지를 보낼 수 있는 기능이다. 간편한 댓글 기능을 사용하지 않고 쪽지를 사용하게 함으로써 동조 현상과 가볍게 다는 악의적인 댓글을 줄일 수 있다는 장점이 있다. 사용자는 쪽지 기능을 통해 해당 글의 작성자에게 공감의 글이나 조언, 응원의 글을 보낼 수 있고, 수신자는 이에 대해 답장을 보낼 수 있다. 이 때 발신자와 수신자는 아이디가 아닌 무작위로 생성된 익명의 이름을 사용하게 된다.
입력	사용자는 커뮤니티를 통해 공유된 글을 클릭하여 쪽지 보내기 버튼을 클릭하고, 팝업창에서 내용을 입력한다. 사용자가 쪽지 발신 시 사용하고 싶은 스킨이 있는 경우, 옵션으로 스킨 종류를 입력받는다.
출력	성공적으로 쪽지가 보내진 경우, 시스템은 쪽지가 발신되었다는 팝업창을 띄운다. 내용이 입력되어있지 않은 경우, 시스템은 내용을 입력하라는 팝업창을 띄운다.
처리	시스템은 발신자의 보낸 쪽지함에 쪽지를 발신한 날짜, 수신자의 랜덤 이름, 내용, 스킨을 저장한다. 수신자의 받은 쪽지함에 쪽지를 수신한 날짜, 발신자의 랜덤 이름, 내용, 스킨을 저장한다.

Table 10 Send Function

G. Evaluation

기능적 요구사항	
기능	평가
설명	악의적인 쪽지를 작성하는 이용자를 차단하기 위한 기능이다. 사용자는 수신된 쪽지에 대해 긍정이나 부정의 평가(차단)를 내릴 수 있으며, 그 평가는 발신자에게는 보이지 않는다. 발신자를 차단하면 더 이상 해당 사용자로부터의 쪽지가 보이지 않고, 차단을 일정 횟수 이상 받은 이용자는 관리자의 확인 후 쪽지 기능의 사용을 제재받게 된다.
입력	사용자는 수신된 쪽지에 대해 긍정이나 부정의 평가(차단)를 내린다.
출력	성공적으로 평가가 반영된 경우 시스템은 팝업창을 띄워 사용자에게 알린다.
처리	부정의 평가(차단)를 받은 경우, 해당 쪽지의 발신자가 보낸 쪽지는 더 이상 수신자에게 보이지 않는다. 또한 일정 횟수 이상 차단을 받은 사용자에게 대해서는 관리자의 확인 후 쪽지 기능이 작동하지 않도록 한다.

Table 11 Evaluation Function

H. Skin

기능적 요구사항	
기능	스킨 이용
설명	사용자가 원하는 스킨을 구매하고 사용할 수 있는 기능이다. To Dot(투닷)은 감성적인 일러스트, 포맷(원고지 등), 사진 등의 스킨을 제공하여 사용자는 작성하려는 글이나 쪽지에 맞는 분위기의 스킨을 적용할 수 있다.
입력	사용자는 개인 페이지에서 스킨 탭으로 이동하여 현재 보유한 스킨에 대한 정보, 구매 가능한 스킨에 대한 정보 등을 확인할 수 있다. 구매를 원하는 경우 전자결제시스템이 작동한다.
출력	시스템은 사용자가 현재 보유한 스킨과 구매 가능한 스킨을 출력한다.
처리	시스템은 사용자가 보유하거나 새롭게 구매한 스킨 정보를 데이터베이스에 저장하고, 사용할때마다 그 정보를 업데이트한다.

Table 12 Skin Function

6.3. Non-Functional Requirements

비기능적 요구사항으로는 Product requirement, Organizational requirement, External requirement이 있다.

A. Product requirement

A.1. Performance

해당 시스템은 사용자가 글을 작성한 후 개인 페이지에서 바로 확인할 수 있도록 해야 한다. 또한 내용의 누락이 없어야 하며, 원하는 경우에만 커뮤니티에 글이 공유되도록 한다. 시스템은 사용자가 기존에 작성한 글을 누락하지 않고 개인 페이지에서 확인할 수 있도록 제공해야 한다. 또한 기존의 글을 수정하거나 삭제한 결과가 개인 페이지는 물론 커뮤니티에 즉시 반영되어야 한다. 커뮤니티에서 쪽지를 주고받을 때, 수신자와 발신자의 정보에 오류가 없어야 한다. 쪽지를 주고받거나 글을 작성할 때 글의 길이, 포맷에 상관없이 스킨이 잘 적용되어야 하고, 구매한 결과와 사용한 결과가 즉시 반영되어야 한다.

A.2. Security

해당 시스템은 개인적인 글을 작성하고 보관하며, 익명의 커뮤니티를 통해 소통하는 서비스를 제공하기 때문에 사용자의 계정 정보가 유출되지 않도록 해야 한다. 사용자의 비밀번호는 암호화되어 관리자도 확인할 수 없도록 한다. 커뮤니티에서 글을 공유하거나 쪽지를 주고받을 때는 임의로 생성된 별명을 사용하도록 한다. 또한 전자 결제 시스템의 경우 보안성과 신뢰성이 높은 외부 시스템을 사용하여 고객의 결제 정보가 유출되지 않도록 한다.

A.3. Efficiency

해당 시스템은 웹 환경에서 동작한다. 기존에 작성한 글을 조회하거나 보유/구매 가능한 스킨을 조회할 때, 커뮤니티에서 글을 조회할 때 등의 상황에서 정보를 불러오는 대기 시간이 길어지지 않도록 한다. 이는 데이터베이스를 효율적으로 구축하고, 불필요한 폰트나 이미지의 사용을 자제하여 웹페이지를 전체적으로 가볍게 개발함으로써 구현할 수 있다.

A.4. Dependability

해당 시스템은 서버와 지속적인 통신을 전제한다. 서버와의 통신이 끊어지면 작성한 글이나 수정 및 변경 사항, 스킨의 구매 및 사용 내역이 제대로 반영되지 않아 문제가 될 수 있다. 따라서 안정적인 서버를 구축하여 글의 작성이나 구매 내역에 오류가 없도록 한다. 또한 정확도가 높은 자연어 처리와 감성 분석 API를 사용하여, 사용자에게 신뢰도가 높은 분석 결과를 제공한다.

A.5. Usability

해당 시스템은 간단한 UI를 제공하여 사용자들이 서비스를 이용하는 것에 불편함이 없도록 한다. 직관적인 탭 이동과 버튼 배치 등을 통해 시스템의 기능을 제공한다.

B. Organizational requirement

B.1. Environment

해당 시스템은 부트스트랩을 사용한 반응형 웹으로 개발되어, 사용자의 환경에 최적화된 화면을 제공한다. 반응형 웹의 경우 지원하지 않는 브라우저의 버전도 있으므로, 이를 사용자에게 명시한다.

B.2. Operation

해당 시스템은 사용자 개인의 페이지와 커뮤니티 페이지를 구분하여, 원하지 않는 경우 글의 공유를 제한한다. 사용자 간의 소통은 커뮤니티 페이지를 통해 쪽지와 그에 대한 답장으로만 이루어진다. 쪽지의 수신 및 발신은 사용자의 아이디 대신 임의로 생성된 별명을 사용하여 서로를 인식할 수 있는 최소한의 기능을 제공한다.

C. External requirement

C.1. Safety/Security

해당 서비스는 nodeJS를 사용하여 구축된 서버로 시스템을 관리한다. 서버 관리자는 해당 서비스에 회원 가입하여 시스템을 이용하는 사용자들을 관리하고, 악의적인 접근을 막는다. 또한 전자 결제 시스템의 경우 보안성과 신뢰성이 높은 외부 시스템을 사용하여 고객의 결제 정보가 유출되지 않도록 한다. 전자 결제 시스템이 요구하는 정보를 구매자에게 요청하고, 이를 전자 결제 시스템의 형식에 맞게 보내어 오류가 발생하지 않도록 한다.

6.4. Scenario

A. Join and Login Scenario

A.1. Initial assumption

To Dot(투닷) 시스템을 이용하기 위해서는 회원 가입 후 로그인을 해야한다. 사용자는 간단한 정보(메일, 아이디, 비밀번호) 제공 후 메일 및 아이디 중복확인과 비밀번호 확인 후 가입을 완료한다. 사용자는 처음 한 번 회원 가입을 마치면 그 다음부터는 아이디와 비밀번호를 입력하여 로그인함으로써 서비스를 이용할 수 있다.

A.2. Normal flow of events

사용자는 To Dot(투닷)에 정보 제공 동의를 한 후, 메일, 아이디, 비밀번호를 입력하여 회원 가입을 한다. 메일 및 아이디 중복 확인과 비밀번호 확인의 과정을 거쳐 회원 가입을 완료한 후에는 아이디와 비밀번호로 로그인하여 To Dot(투닷)의 서비스를 이용한다.

A.3. What can go wrong & concurrent activities

회원 가입 시, 메일과 아이디의 경우 중복 확인을 하고 이미 해당 메일, 아이디로 가입된 계정이 존재하는 경우 팝업창이 뜨며 새로운 메일, 아이디를 요구한다. 또한 필수로 요구하는 정보를 누락 했을 때, 회원 가입이 진행되지 않는다.

로그인 시, 데이터베이스에서 아이디와 비밀번호가 일치하는 계정이 없는 경우 팝업창이 뜨며 로그인에 실패한다.

A.4. System state on completion

로그인 후 To Dot(투닷)에서 제공되는 서비스를 이용할 수 있다.

B. Add a Posting Scenario

B.1. Initial assumption

로그인 후 사용자에게 가장 먼저 보여지는 것은 사용자의 개인 페이지이다. 사용자는 개인 페이지에 새로운 글을 작성하거나 기존의 글을 수정, 삭제할 수 있다. 또한 분석 결과를 확인하거나 커뮤니티 페이지로 이동할 수 있다.

B.2. Normal flow of events

새로운 글 작성을 원할 경우 사용자는 개인 페이지에서 글 작성 버튼을 클릭한다. 글을 작성할 때는 제목과 내용을 입력하고, 원할 경우 스킨 사용과 해당 글을 커뮤니티에 공개할지의 여부를 선택한다. 기존의 글을 수정, 삭제하려면 사용자는 해당 글을 선택하여 작업을 수행한다.

B.3. What can go wrong & concurrent activities

사용자가 글을 등록하거나 수정할 때, 제목이나 내용 값이 없으면 글이 정상적으로 등록되지 않고 팝업창을 띄워 정보 입력을 요구한다.

B.4. System state on completion

글이 정상적으로 등록되면 사용자는 해당 글을 개인 페이지에서 확인할 수 있다. 수정 결과 또한 곧바로 반영되는 것을 확인할 수 있다. 글을 커뮤니티에 공개한 경우, 커뮤니티 페이지에서도 확인 가능하다.

C. Use Analyzing Service Scenario

C.1. Initial assumption

To Dot(투닷)은 사용자가 작성한 글을 토대로 개인별로 정량적인 분석 결과를 제공한다. 분석 결과를 원하는 경우 사용자는 개인 페이지에서 분석 결과 탭을 클릭한다.

C.2. Normal flow of events

사용자는 To Dot(투닷) 시스템으로부터 키워드 분석 결과, 감정 분석 결과 등을 제공받고, 키워드를 기반으로 한 커뮤니티의 글 추천을 받을 수 있다. 새로운 게시글이 작성되거나, 수정/삭제가 발생할때마다 업데이트된 분석 결과를 제공받을 수 있다.

C.3. What can go wrong & concurrent activities

등록된 글이 없는 경우 분석 결과가 제공되지 않는다.

D. Join a Community Scenario

D.1. Initial assumption

To Dot(투닷)은 사용자에게 개인 페이지의 글 작성과 분석 결과 제공 외에도 커뮤니티 기능을 제공한다. 사용자는 개인 페이지에 작성한 글을 공유하고 싶은 경우 커뮤니티에 공유할 수 있으며, 이 때 해당 글은 익명으로 공유된다. 커뮤니티의 글은 페이스북과 같은 뉴스피드 형식으로 제공되고, 상위 뉴스피드에는 최근 공유된 순서대로 글이 나타난다.

D.2. Normal flow of events

사용자는 커뮤니티 페이지로 이동하여 다른 사용자들이 공유한 글을 읽을 수 있다. 스크롤을 이동하며 글을 미리보기로 확인할 수 있고, '더 보기'를 클릭하여 전체 글을 읽을 수 있다.

D.3. What can go wrong & concurrent activities

커뮤니티에 공유된 글 중 광고나 욕설 등의 부적절한 글은 작성자의 의사와 상관없이 관리자에 의해 블라인드 처리 될 수 있다. 커뮤니티에서 블라인드 처리되어도 작성자의 개인 페이지에서는 글을 확인할 수 있다.

D.4. System state on completion

새로운 글이 공유되면 To Dot(투닷)을 이용하는 모든 사용자가 그 글을 열람할 수 있다. 또한 공유된 글은 키워드가 분석되어, 비슷한 키워드의 글을 작성한 사용자들에게 추천 글로 보내질 수 있다.

E. Send Message Scenario

E.1. Initial assumption

사용자는 커뮤니티를 통해 공유된 글의 작성자에게 쪽지를 보낼 수 있고, 수신자는 이에 대해 답장을 보낼 수 있다. 간편한 댓글 기능을 사용하지 않고 쪽지를 사용하게 함으로써 동조 현상 및 악의적인 댓글을 줄일 수 있다. 이 때 발신자와 수신자는 아이디가 아닌 무작위로 생성된 익명의 이름을 사용하게 된다.

E.2. Normal flow of events

사용자는 커뮤니티를 통해 공유된 글을 클릭하여 쪽지 보내기 버튼을 클릭한다. 팝업창에서 보낼 쪽지의 내용을 입력하고 원하는 경우 스킨을 지정한다.

E.3. What can go wrong & concurrent activities

쪽지의 내용이 누락된 경우 쪽지가 보내지지 않고 내용을 입력하라는 팝업창이 뜬다.

E.4. System state on completion

쪽지가 정상적으로 보내진 경우 쪽지가 발신되었다는 팝업창이 뜬다.

F. Make an Evaluation Scenario

F.1. Initial assumption

지속적으로 악의적인 쪽지를 작성하는 이용자를 차단하는 기능이다.

F.2. Normal flow of events

사용자는 수신된 쪽지에 대해 긍정이나 부정의 평가(차단)를 내릴 수 있다. 긍정이나 부정을 선택하면 팝업창이 뜨고, 사용자로부터 다시 한번 확인의 절차를 거친다.

F.3. System state on completion

성공적으로 평가가 반영된 경우 팝업창이 뜬다. 차단한 사용자로부터 받은 쪽지는 쪽지함에서 삭제되며, 더 이상 해당 사용자로부터 쪽지를 받지 않게 된다. 차단을 일정 횟수 이상 받은 이용자는 관리자의 확인 후 쪽지 기능의 사용을 제재받게 된다.

G. Payment Scenario

G.1. Initial assumption

To Dot(투닷)은 스킨 결제시 외부의 전자 결제 시스템을 이용한다.

G.2. Normal flow of events

사용자는 구매를 원하는 스킨을 선택한 후 결제를 요청한다. 시스템은 사용자의 결제 요청 시 외부 시스템을 이용하여 결제를 진행하고, 그 정보를 데이터베이스에 저장한다.

G.3. What can go wrong & concurrent activities

사용자가 결제 시스템에서 필수적인 정보를 누락한 경우, 결제 요청이 거부된다.

G.4. System state on completion

결제가 완료되면 시스템은 그 결과를 사용자에게 출력하고, 사용자는 개인 페이지의 스킨 탭에서 보유한 스킨 목록이 업데이트 된 것을 확인할 수 있다.

7. System Models

7.1. Objective

System Models에서는 시스템 컴포넌트, 시스템 그리고 시스템 환경 사이의 관계를 보여준다. Context diagram과 Process diagram을 이용하여 시스템과 외부 환경의 상호작용을 나타내고, Use case diagram과 sequence diagram을 이용하여 사용자와 시스템, 또는 components 사이의 상호작용을 나타낸다. 또한 Class diagram을 이용하여 시스템의 구성과 데이터 구조를 나타내고, Data-driven modeling과 Event-driven modeling으로 Behavioral perspective를 표현한다.

7.2. Context Models

A. Context Model

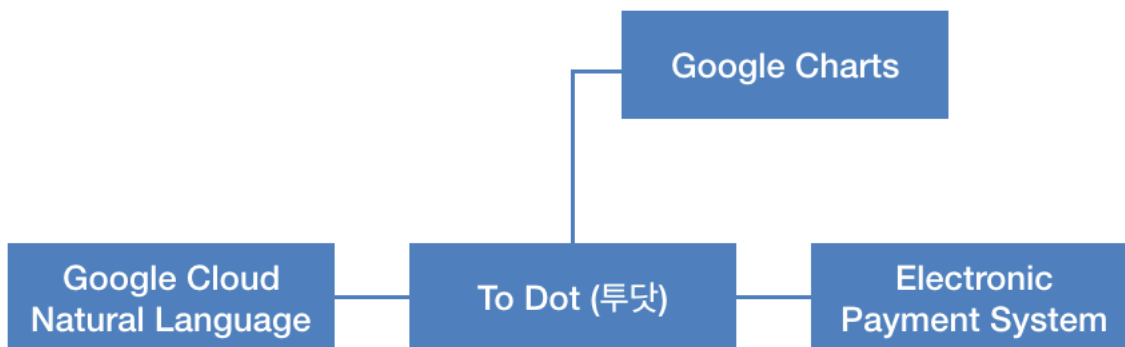


Diagram 4 Context Diagram

B. Process Diagram

B.1. Overall Process

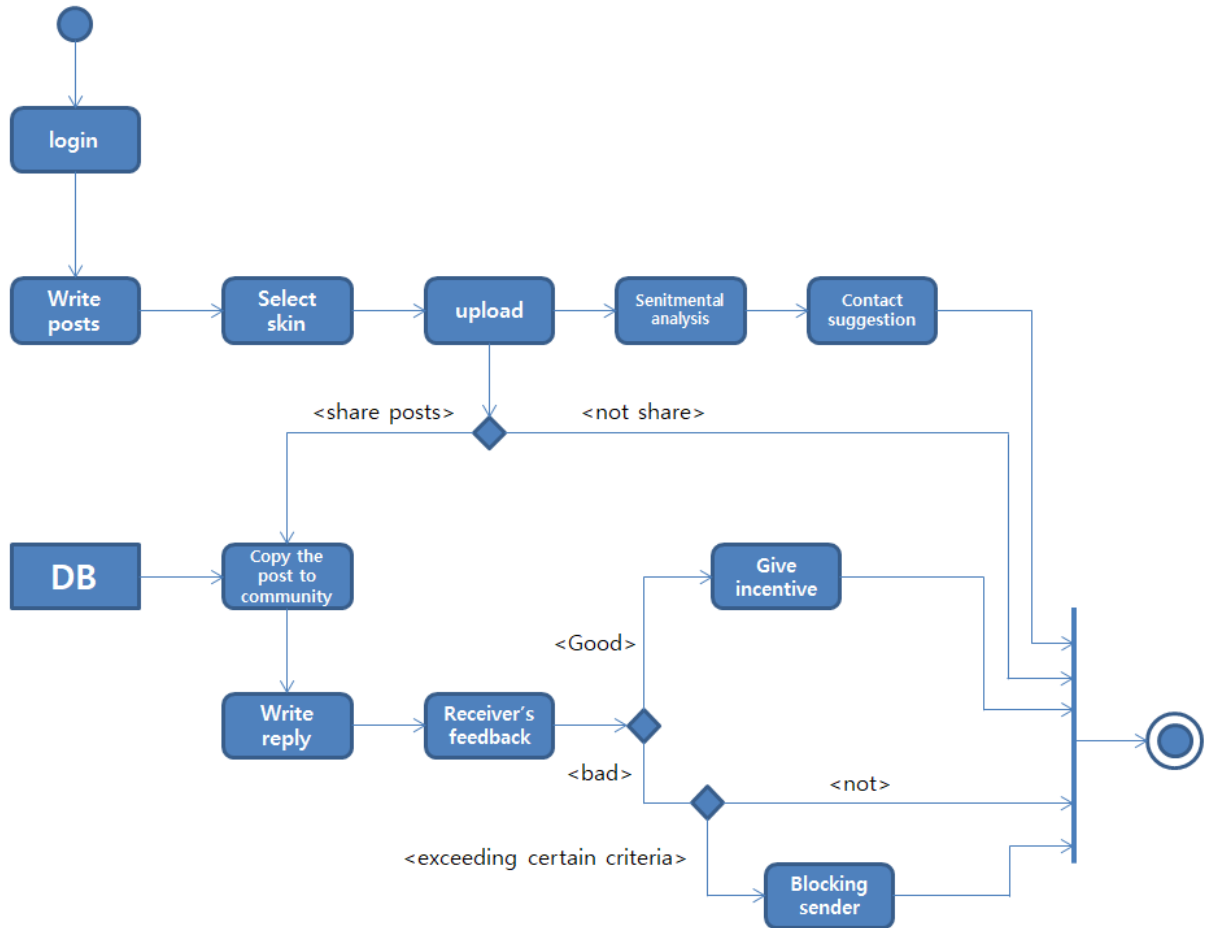


Diagram 5 Overall Process Diagram

B.2. Posting Process

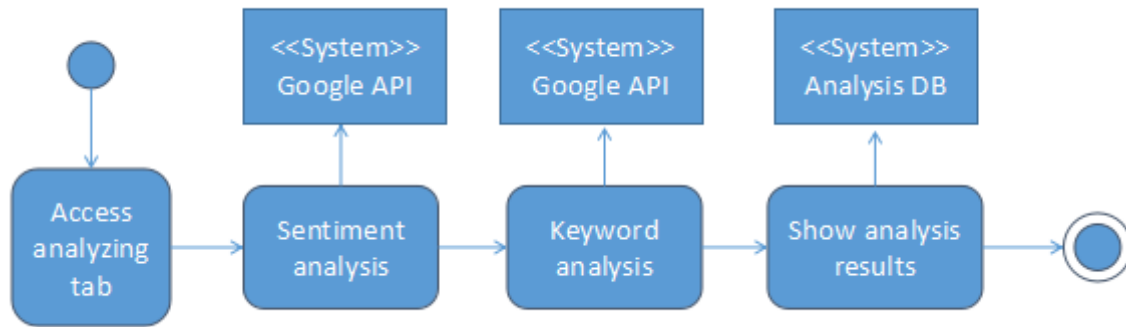


Diagram 6 Posting Process Diagram

B.3. Analyzing Process

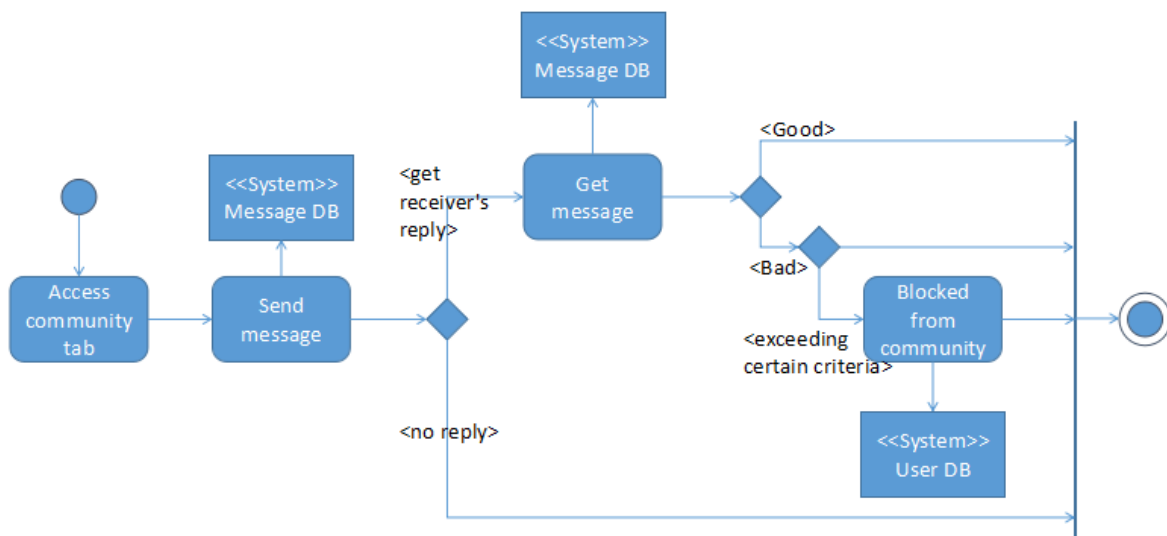


Diagram 7 Analyzing Process Diagram

B.4. Feedback Process

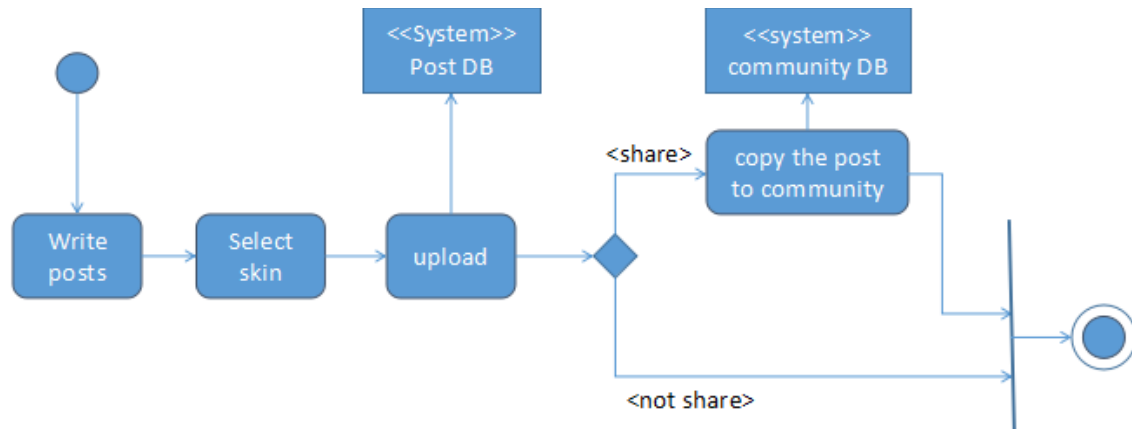


Diagram 8 Feedback Process Diagram

7.3. Interaction Models

A. Use case models

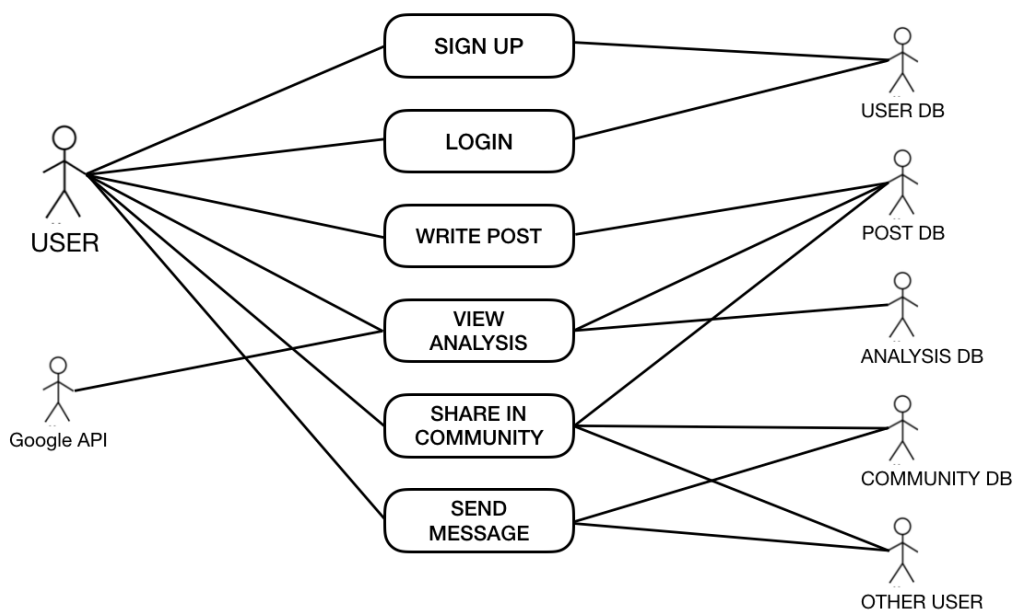


Diagram 9 Use Case Diagram

B. Tubular description for each use case

B.1. Sign up

SUB SYSTEM	SIGN UP SYSTEM
USE CASE	Sign up
ACTOR	User, user DB
DESCRIPTION	- Users join To Dot service through registering their information through sign-up system.
STIMULUS	- Users request sign-up service and wait for page to fill the form. - After completing the form, users send request for sign-up.
RESPONSE	- If completed form is received, it sends DB update query to user DB.

Table 13 Tubular Table : Sign up

B.2. Login

SUB SYSTEM	LOGIN SYSTEM
USE CASE	Login
ACTOR	User, User DB
DESCRIPTION	- Members can login with their account information.
STIMULUS	- When users insert their login information, request query if the account information is correct to the user DB.
RESPONSE	- If the account information is correct, users can complete login step and access their own DB.

Table 14 Tubular Table : Login

B.3. Write Post

SUB SYSTEM	WRITING POST SYSTEM
USE CASE	Write post
ACTOR	User, Post DB
DESCRIPTION	Users write posts and save them at post DB.
STIMULUS	A query of users writing, editing or deleting their posts.
RESPONSE	- If users write new post, save the related information about that post in the users' Post DB. - If users edit or delete existing post, update users' Post DB.

Table 15 Tubular Table : Writing Post

B.4. View Analysis

SUB SYSTEM	ANALYZING SYSTEM
USE CASE	View analysis
ACTOR	User, Post DB, Analysis DB, Google API
DESCRIPTION	<ul style="list-style-type: none"> - Users can see the result of analysis about their posts. - If users access analyzing system, the system analyze the users' posts using Google Natural Language API.
STIMULUS	A query of users requesting the result of analysis about their posts.
RESPONSE	<ul style="list-style-type: none"> - Analyze the users' posts and show the result of analysis to the user. - Update the result of analysis in the user's Analysis DB.

Table 16 Tubular Table : View Analysis**B.5. Share in Community**

SUB SYSTEM	COMMUNITY SYSTEM
USE CASE	Share in community
ACTOR	User, Post DB, Community DB, Other user
DESCRIPTION	Users can share their posts in case they want, so every users of To Dot can read their posts and interact through the message service.
STIMULUS	Users write new post and select the sharing option, or users edit existing post and change option to share the post.
RESPONSE	The data of user and post is copied or updated to the Community DB.

Table 17 Tubular Table : Share in Community**B.6. Send Message**

SUB SYSTEM	MESSAGE SYSTEM
USE CASE	Send message
ACTOR	User, Community DB, Other user
DESCRIPTION	Users can send message to the other user through the community. The information about sender and getter is never shown to each other.
STIMULUS	Users send message to the specific post's writer, through the community service.
RESPONSE	The data of message is saved to the sender's message DB and getter's message DB, except the personal information such as ID.

Table 18 Tubular Table : Send Message

C. Sequence Diagram

C.1. Sequence Diagram of Managing User Posts

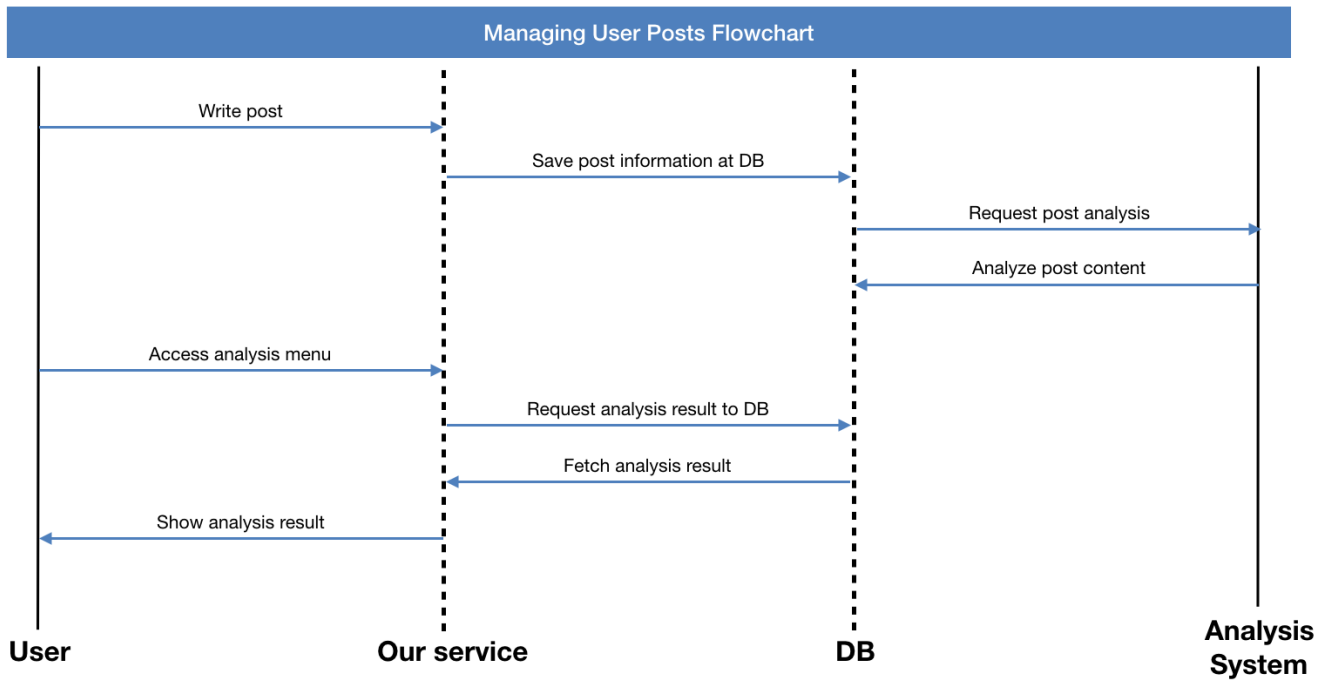


Diagram 10 Sequence Diagram of Managing User Posts

C.2. Sequence Diagram of Using Community Service

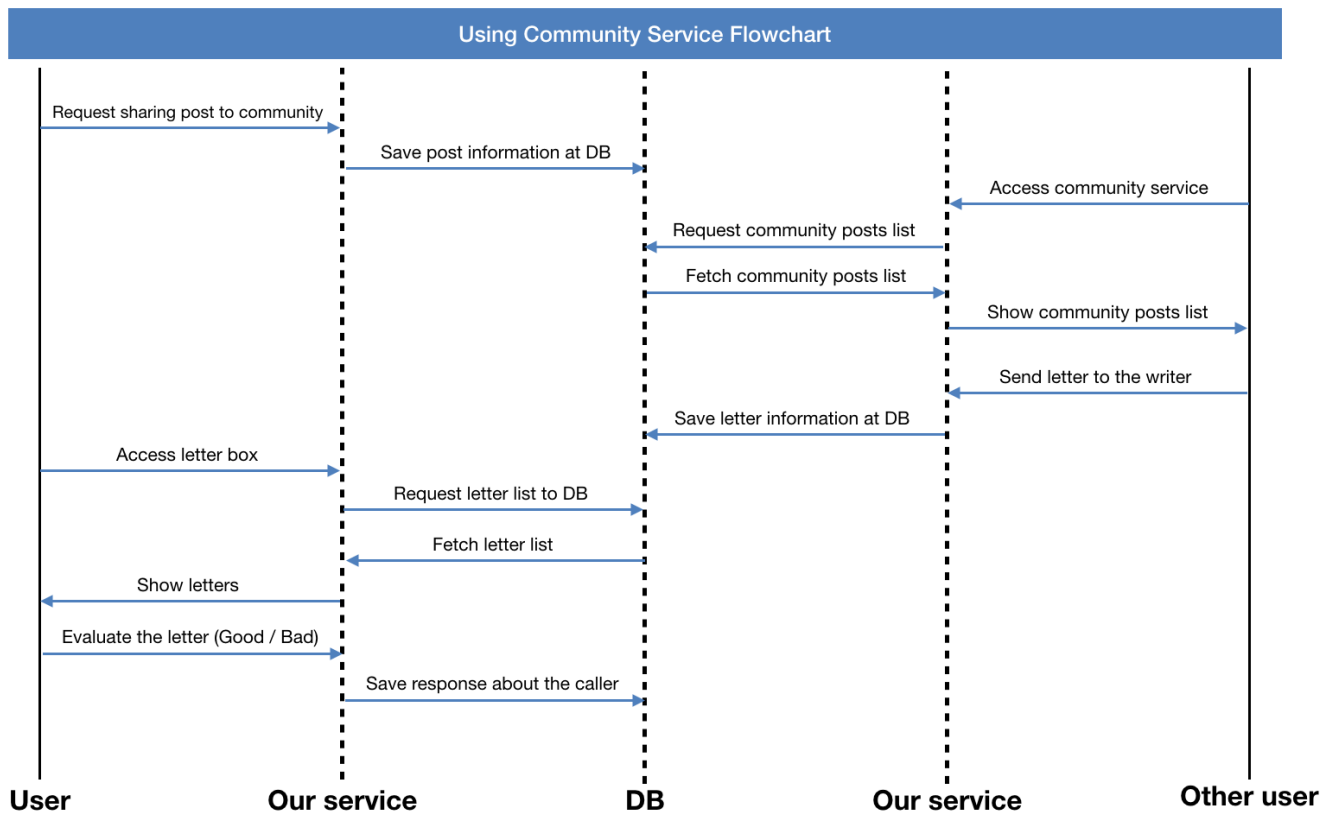


Diagram 11 Sequence Diagram of Using Community Service

7.4. Structural Models

A. User System Class Diagram

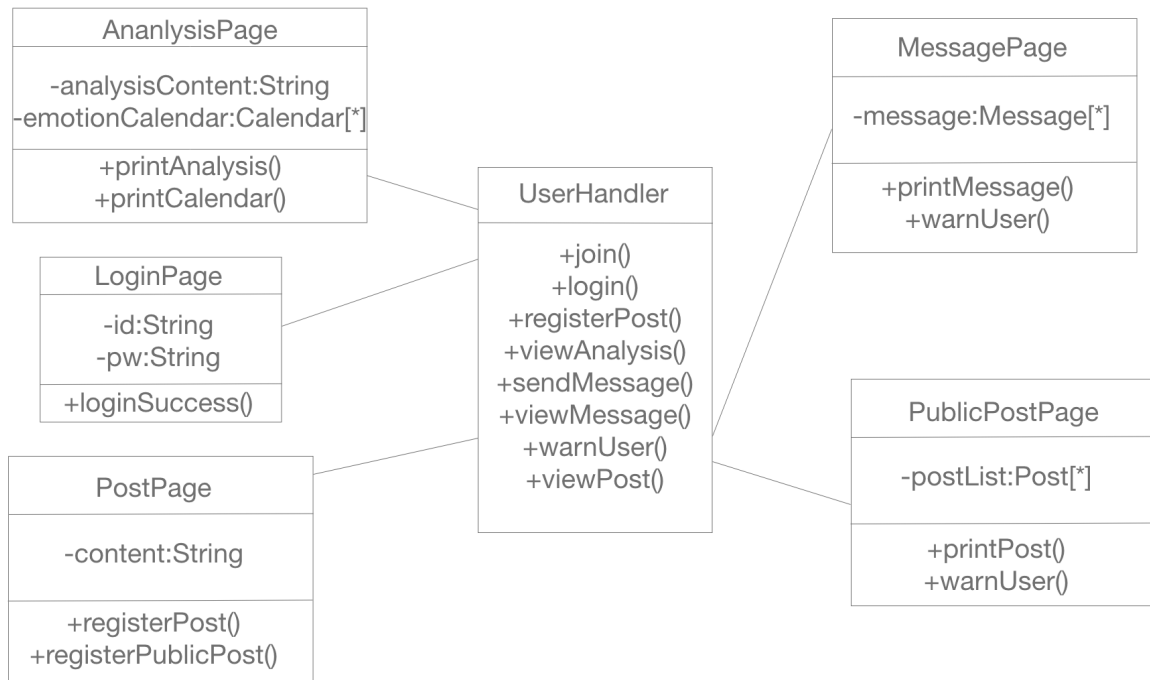


Diagram 12 User System Class Diagram

7.5. Behavioral Models

이 장에서는 User와 DB, Server간에 일어나는 주요한 기능들에 대해 다루었다.

A. Data-Driven Modeling

총 3개의 Object에 12개의 Action이 존재한다. 각 Object마다의 Action 수는 다음과 같다.

User: 4개 / 글쓰기, 사용자 신고, 감정 분석, 쪽지 보내기

DB: 6개 / 글저장, 글목록 요청, 공개글 목록 요청, 사용자 신고, 감정 분석, 쪽지 보내기

Server: 2개 / 감정 분석, 쪽지 전송

A.1. User

-글쓰기



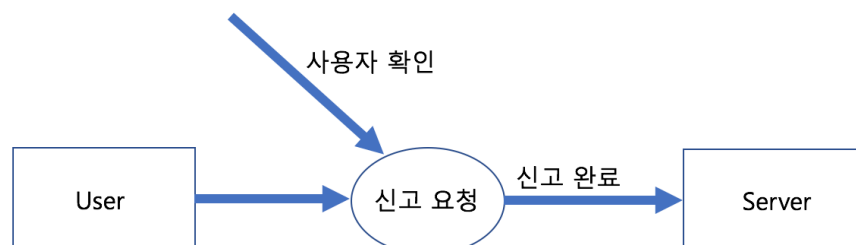
Diagram 13 Data Flow Diagram of User : Writing Posts

-감정 분석



Diagram 14 Data Flow Diagram of User : Sentiment Analysis

-사용자 신고



-쪽지 보내기

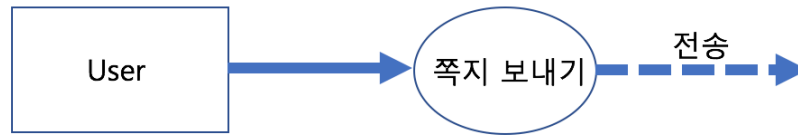
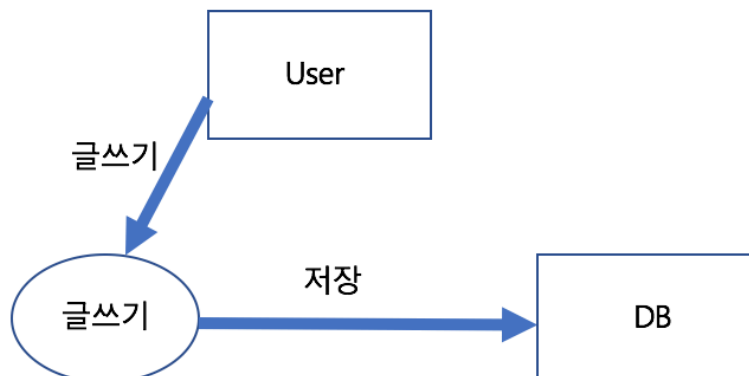


Diagram 15 Data Flow Diagram of User : Blocking Users
Diagram 16 Data Flow Diagram of User : Send Message

-글 저장



A.2 DB

Diagram 17 Data Flow Diagram of DB : Saving Posts

-글 목록 요청

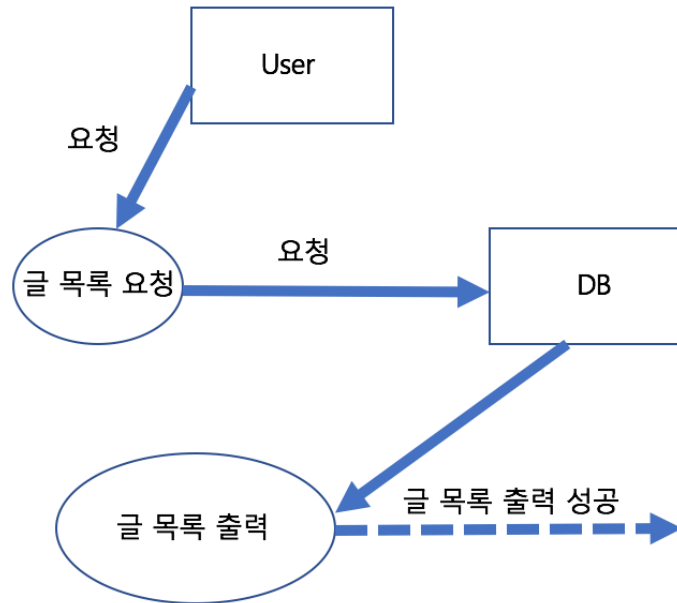


Diagram 18 Data Flow Diagram of DB : Request Posts list

-공개글 목록 요청

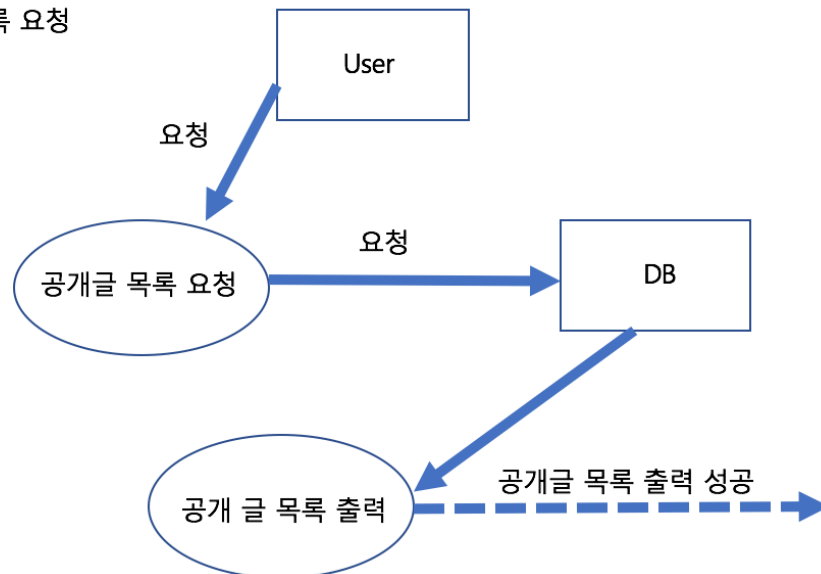


Diagram 19 Data Flow Diagram of DB : Request Shared Posts list

-사용자 신고

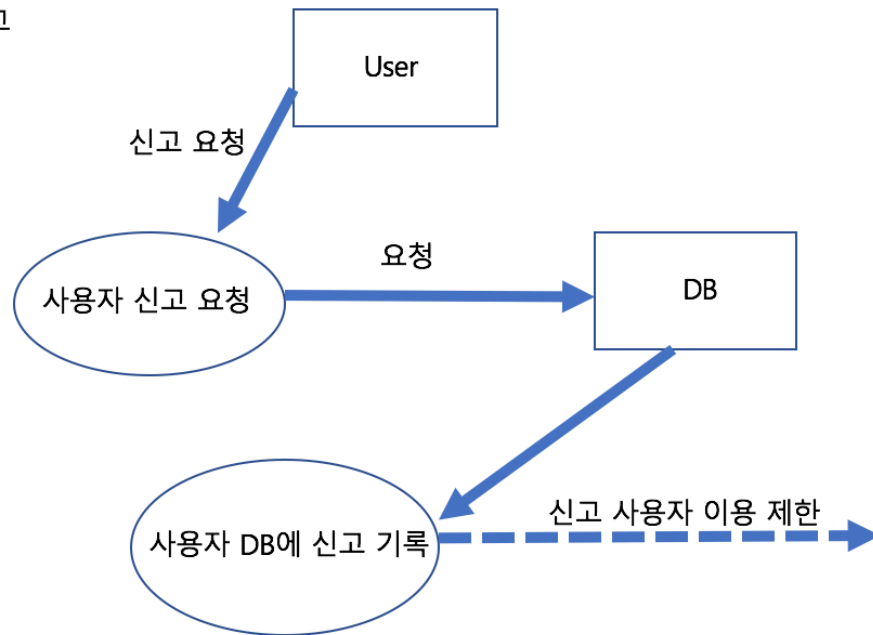


Diagram 20 Data Flow Diagram of DB : Blocking Users

-감정 분석

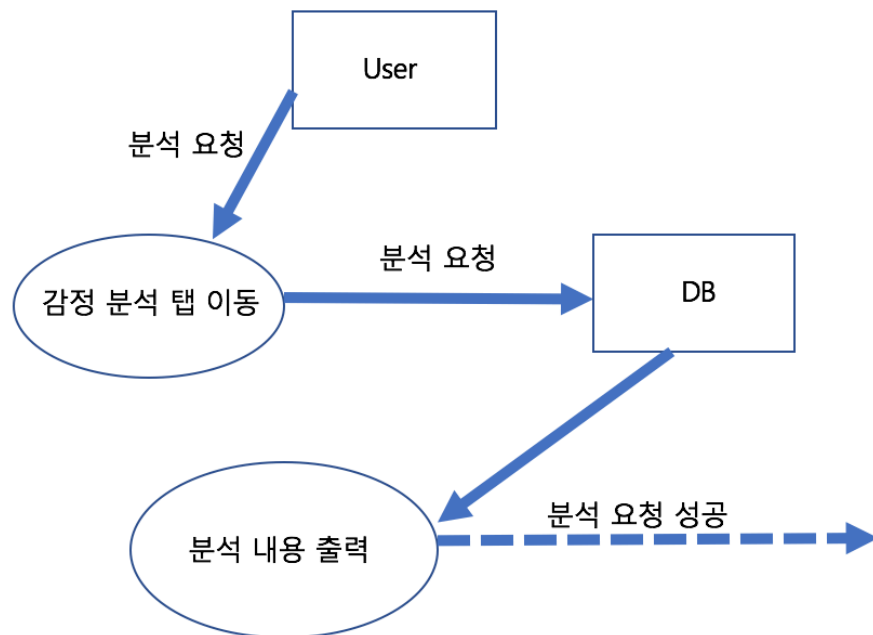


Diagram 21 Data Flow Diagram of DB : Sentiment Analysis

-쪽지 보내기

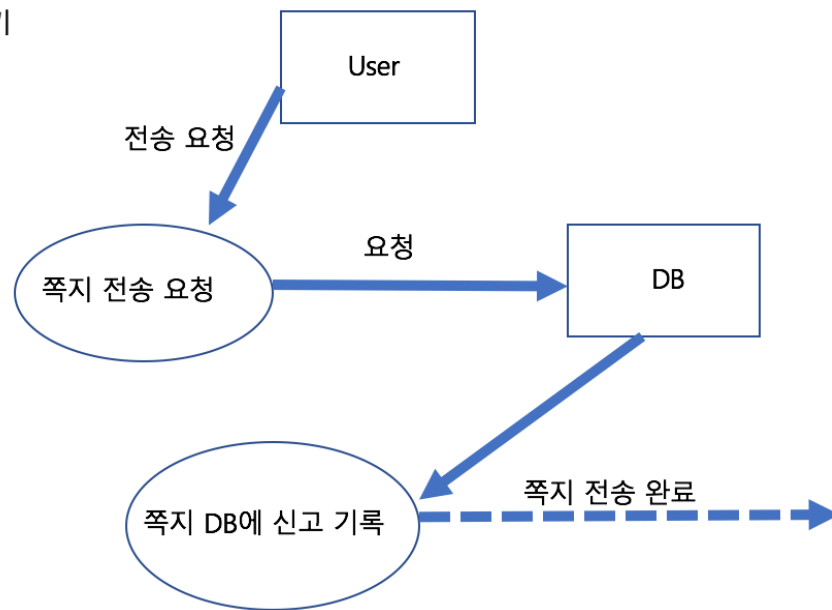


Diagram 22 Data Flow Diagram of DB : Send Message

A.3 Server

-감정 분석

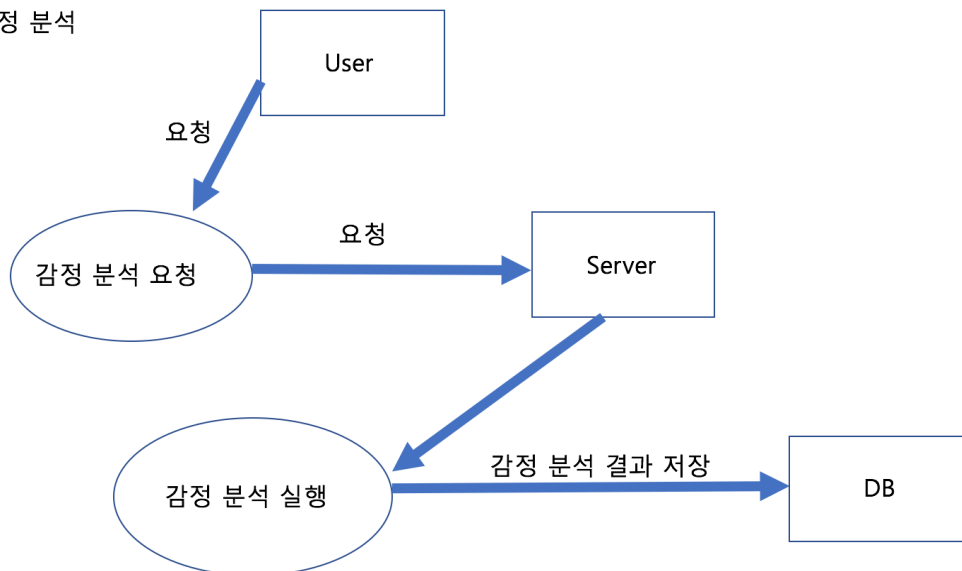


Diagram 23 Data Flow Diagram of Server : Sentiment Analysis

-쪽지 전송

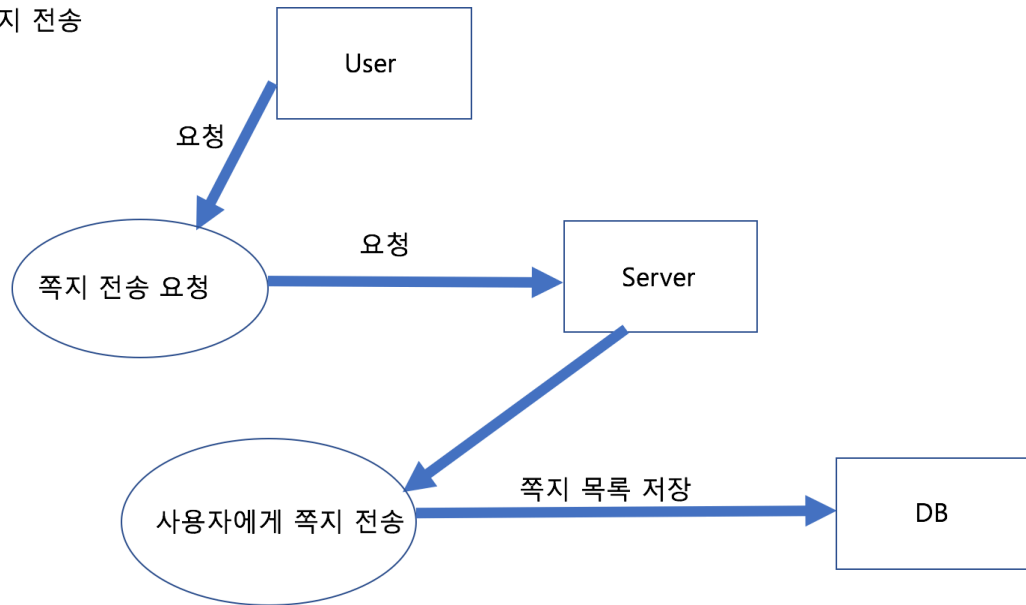


Diagram 24 Data Flow Diagram of Server : Send Message

B. Event-Driven Modeling

B.1. Sharing

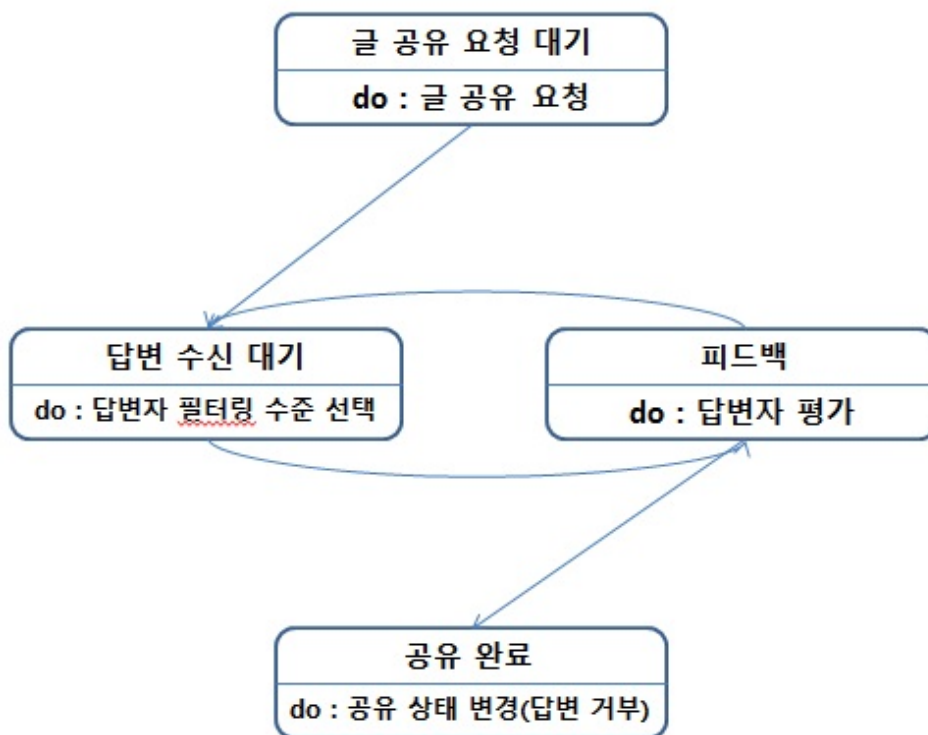


Diagram 25 Event Driven Diagram : Using Community Service

B.2. Posting

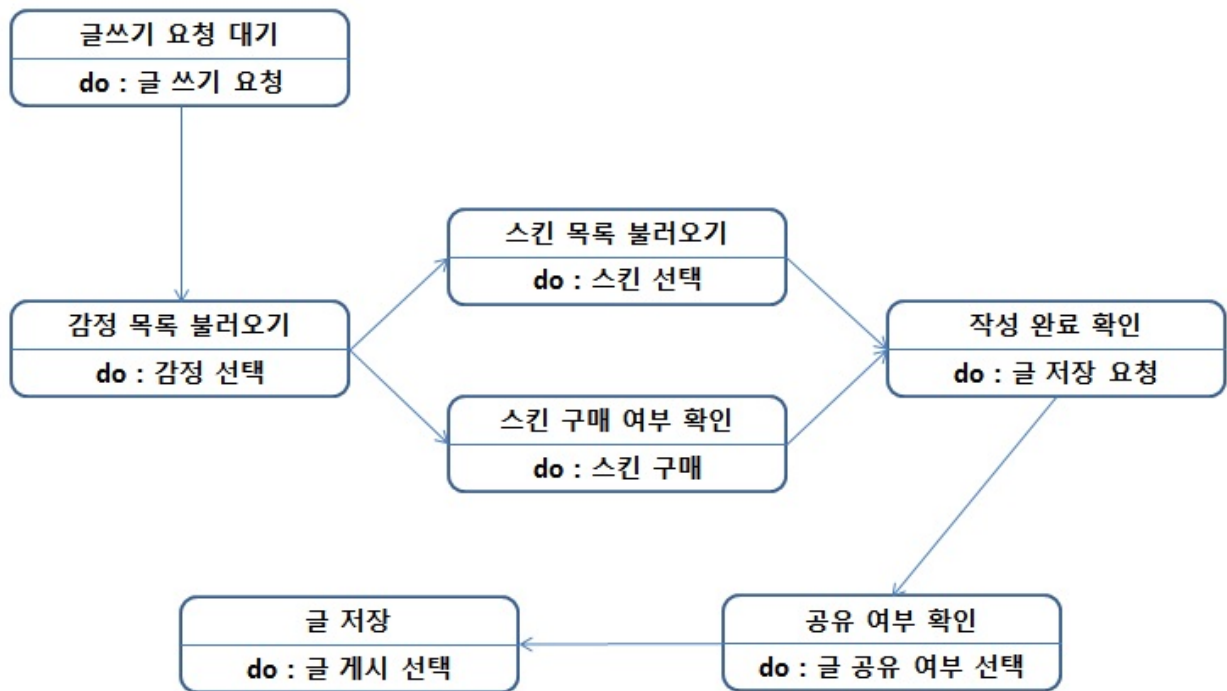


Diagram 26 Event Driven Diagram : Posting

8. System Evolutions

8.1. Objective

System Evolution에서는 시스템이 가정하는 사항들과 시스템의 변경을 발생시킬 수 있는 요인들에 대해 설명한다. 이 장은 차후에 발생할 시스템의 설계 변경의 비용을 낮추는 것에 도움이 되므로 시스템 설계자가 참고하면 좋을 것이다.

8.2. Limitation and Assumption

After a hurried day, many people like to write a diary to summarize themselves. Diary is a tool for people who write it and the main functions of diary is pour out.

Diary is a silent friend, when you are angry, happy, sad, confused, you can write those in your diary, the diary as a friend and share with it. When you talk about your secret in your diary, curse the leaders who make you difficult, and sort out your entangled thoughts, your diary will silently accept. Similarly, at a great happiness time, if you write a diary, your happiness will be extended. Like a friend can double happiness. But it's not a good thing. In a diary, if you record your lover's abandonment and express your pain, it will increase your anger and even in the sad mood can not extricate themselves.

With the development of communication technology, information dissemination gradually changes from paper media to network media. People used to write their secrets in their diary. Now, some websites have opened up a platform for all kinds of people who want to pour out but do not want to reveal their names. Some places on the Internet have become a more convenient and fashionable choice for people to pour out their privacy.

Most of this website is to hide the IP address through the user, to numbered the message, and the user can find the message by numbering. The message issued is completely anonymous.

So this platform needs users' participation most. If there is no effective participation of users, the platform is basically a no content media. They give users great participation space, allowing users to generate content.

For users, websites need protect privacy. Managers will check the secrets before every secret release. In the secret the real name can not be found. It can not contain the information such as address, e-mail address, telephone number, and even the promulgator's IP is kept secret.

8.3. Evolutions of User Requirement

A. Detailed Sentiment Analysis

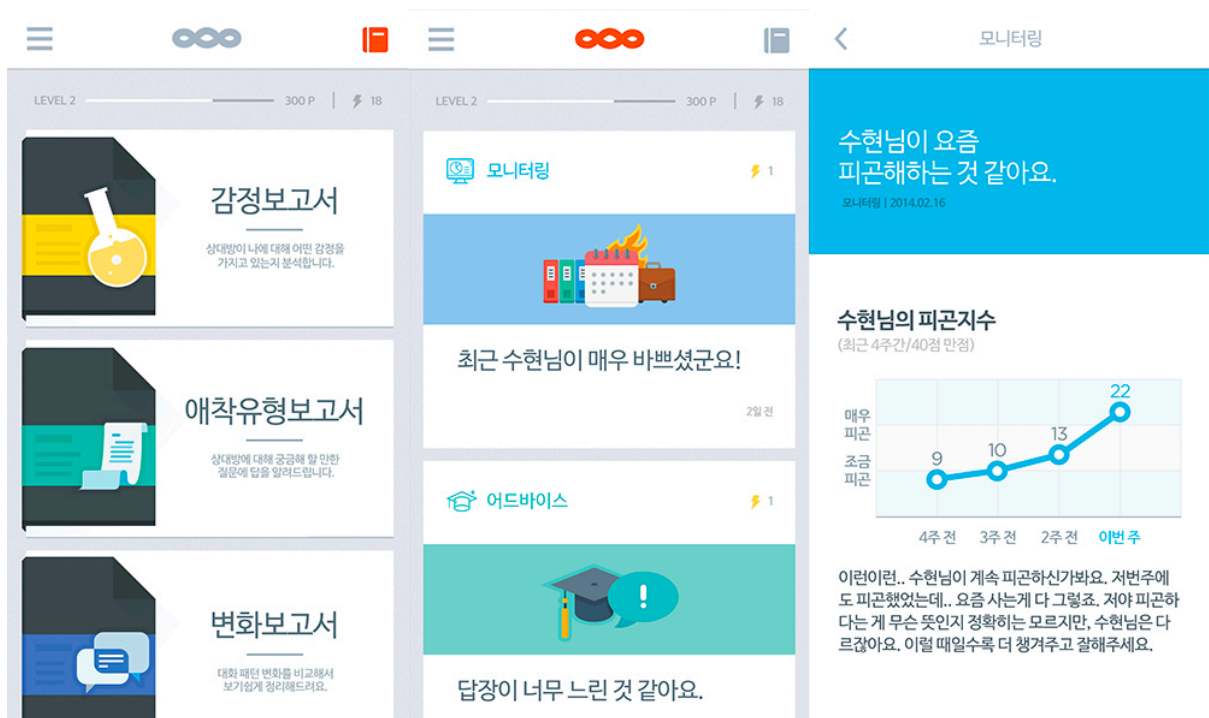


Figure 4 Ginger Service

현재 To Dot(투닷)의 분석 서비스에서는 키워드 분석과 간단한 감성 분석을 통한 일차원적인 분석 결과를 제공한다. 분석 결과를 통해 치유적 글쓰기의 효과를 증대시키고 본인의 감정에 대해 효과적으로 피드백하는 것이 목표이지만, 현재는 그만한 기능은 제공하기 어려운 상태이다.

Figure 4에 나타난 '진저'는 채팅분석 스타트업 스캐터랩이 제공하는 서비스로, 연인과의 채팅 내용을 분석하고 상황을 인지해 감정 정보를 생성, 제공하는 인공지능 앱이다. 키워드 분석 등을 통한 변화 보고서 제공 외에도 로맨틱한 메시지를 자동으로 인식해 저장하고, 상대방이 잠드는 시간을 분석해 피로도를 체크하는 등 다양하고 흥미로운 서비스를 제공하는 것이 '진저' 앱의 특징이다.

To Dot(투닷) 서비스의 사용자들이 많아지고, 작성되는 글 또한 많아진다면 이러한 감정 분석 서비스를 시스템에 적용할 수 있다. 상세하고 정확한 분석 결과를 제공하여 사용자들의 분석 결과에 대한 신뢰도를 높이고, 더욱 자세한 분석 결과에 대해서는 소정의 비용을 제공해야 열람할 수 있도록 한다면 수익을 창출할 수 있을 것이다.

B. Enhanced Filtering Function

To Dot(투닷) 서비스는 댓글 기능 대신 메세지 기능을 제공하여 사용자들간의 유대감을 높이고, 피드백 기능을 사용하여 악의적인 사용자를 차단한다. 또한 현재로써는 커뮤니티에 공유되는 글에 대한 필터링 기능은 제공하지 않는 상태이다.

앞서 언급한 대나무숲과 같은 서비스는 관리자가 존재하여 민감한 주제나 홍보와 관련된 글을 제재하고, 욕설이나 특정 단체/기관의 이름을 필터링한다. 이러한 기능은 순기능도 존재하지만 사용자 발언의 자유를 억압하는 역기능도 있으므로, To Dot(투닷)은 필터링 기능을 제공하지 않는다. 하지만 사용자가 많아지고, 커뮤니티에 서비스의 목적과 맞지 않는 홍보, 비방과 관련한 글이 지속적으로 공유된다면 이를 해결할 방안이 필요하다.

C. Provision of Various Covers

To Dot(투닷) 서비스는 글과 메세지에 다양한 표지를 적용시키는 기능을 제공한다. 소정의 비용을 내고 표지를 구매하는만큼, 사용자는 더욱 높은 품질과 다양한 선택지를 요구할것이다. 향후 To Dot(투닷) 서비스의 표지 판매 및 구매를 연결하는 기능을 구축하여, 사용자가 제작한 표지를 다른 사용자들이 구매할 수 있도록 한다면, 사용자들은 더욱 다양한 선택지를 제공받고, To Dot(투닷)은 중개에 대한 소정의 수수료를 지급받을 수 있다.

9. Appendices

9.1. Objective

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

9.2. Database Requirements

이 절에서는 구축하고자 하는 ToDot(투닷) Database의 요구사항을 설명한다. ToDot(투닷) 데이터베이스의 요구사항은 다음과 같다. 아래에 설명하는 내용들은 적절한 형태로 데이터베이스 테이블 형태로 저장되어야 한다. 각 요구사항에 Attribute는 이탤릭체, Entity type은 굵은 글씨체, Relation type은 굵은 이탤릭체, 제약조건은 밑줄로 표시하였다.

ToDot(투닷)은 사용자 그룹이 1개 있다. 사용자 그룹은 사용자들의 tuple로 구성되어있다. 각 사용자는 id값(id_value)으로 식별된다. 사용자는 ID, PW, 이름, 별명을 필수로 가진다.

ToDot(투닷)은 사용자가 작성한 글과, 쪽지, 감정 분석 결과에 대한 데이터베이스를 가진다. 글, 쪽지, 감정 분석 결과 모두 작성한 사용자의 id값으로 구별된다. 여기서 id값은 사용자 table에 대한 Foreign Key로 설정된다. 글은 작성일자 정보를 가지고 있고, 쪽지는 작성시간, 받는 사용자 id 값을 가지고 있고, 감정분석결과는 분석내용을 정보로 가진다.

9.3. User-System Requirements

To Dot(투닷) 시스템은 HTML5 기반의 nodeJS API를 이용하여 설계된다. 따라서 사용자는 HTML5를 지원하는 웹 브라우저가 필요하다. 데스크탑과 모바일 모두 동일한 조건을 필요로 한다.

Browser(Desktop, Mobile)	
Minimum	Any browser can support HTML5
Recommended	Google Chrome, Mozilla Firefox (Internet Explorer is not recommended)

Table 19 Desktop, Mobile Browser

9.4. Development Process

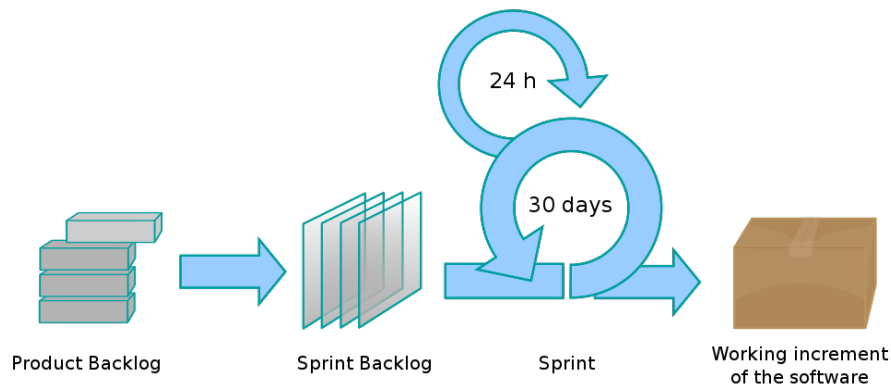


Figure 5 Scrum Process

본 시스템은 Post, 분석, 커뮤니티의 세 가지 시스템이 서로 데이터를 교환하는 형식을 취하고 있다. 이에 따라 세 가지의 하위 시스템이 교환하는 데이터의 구조 및 인터페이스를 시스템 개발 단계부터 완벽하게 계획되기는 힘들다. 따라서 본 프로젝트에서는 Agile 방법론의 하나인 Scrum을 사용하여 개발을 진행하려 한다. 전체적인 개발 대상을 정해놓고, 그 일부분을 우선 개발한다. 그리고 해당 부분의 개발이 종료되면, 다른 부분을 개발한다. 개발 과정에서 팀원 사이에서 코드를 공유하고, 빈번한 회의를 통해 전체적으로 개발하는 데 있어서 팀원 모두가 참여하는 방식을 취한다.

10. Index

10.1. Table Index

Table 1 사용자 관련 용어	13
Table 2 서비스 관련 용어	13
Table 3 개발 및 연산 관련 용어	14
Table 4 기타 관련 용어	14
Table 5 Sign up Function	23
Table 6 Login Function	23
Table 7 Posting Function	24
Table 8 Analysis of Features Function	24
Table 9 Community Function	25
Table 10 Send Function	25
Table 11 Evaluation Function	26
Table 12 Skin Function	26
Table 13 Tubular Table : Sign up	37
Table 14 Tubular Table : Login	37
Table 15 Tubular Table : Writing Post	37
Table 16 Tubular Table : View Analysis	38
Table 17 Tubular Table : Share in Community	38
Table 18 Tubular Table : Send Message	38
Table 19 Desktop, Mobile Browser	52

10.2. Diagram Index

Diagram 1 Posting System	20
Diagram 2 Analyzing System	21
Diagram 3 Community System	22
Diagram 4 Context Diagram	33
Diagram 5 Overall Process Diagram	34
Diagram 6 Posting Process Diagram	35
Diagram 7 Analyzing Process Diagram	35
Diagram 8 Feedback Process Diagram	36
Diagram 9 Use Case Diagram	36
Diagram 10 Sequence Diagram of Managing User Posts	39
Diagram 11 Sequence Diagram of Using Community Service	40
Diagram 12 User System Class Diagram	41
Diagram 13 Data Flow Diagram of User : Writing Posts	42
Diagram 14 Data Flow Diagram of User : Sentiment Analysis	42
Diagram 15 Data Flow Diagram of User : Blocking Users	43
Diagram 16 Data Flow Diagram of User : Send Message	43
Diagram 17 Data Flow Diagram of DB : Saving Posts	43
Diagram 18 Data Flow Diagram of DB : Request Posts list	44
Diagram 19 Data Flow Diagram of DB : Request Shared Posts list	44
Diagram 20 Data Flow Diagram of DB : Blocking Users	45
Diagram 21 Data Flow Diagram of DB : Sentiment Analysis	45
Diagram 22 Data Flow Diagram of DB : Send Message	46
Diagram 23 Data Flow Diagram of Server : Sentiment Analysis	46
Diagram 24 Data Flow Diagram of Server : Send Message	47
Diagram 25 Event Driven Diagram : Using Community Service	47
Diagram 26 Event Driven Diagram : Posting	48

10.3. Figure Index

Figure 1 성균관대학교 대나무숲	9
Figure 2 To Dot Signature	11
Figure 3 Expected Sharing Process	16
Figure 4 Ginger Service	50
Figure 5 Scrum Process	53

11. Reference

- [1] "'대나무숲'등 대학교 온라인 페이지 활성화의 그늘", 금강일보, 2017. [Online]. Available: <http://www.ggilbo.com/news/articleView.html?idxno=431082>. [Accessed: 06- May- 2018].
- [2] "'대나무숲 폐쇄하자'...무분별한 '익명성'에 뿔난 대학생들", 뉴스1, 2017. [Online]. Available: <http://news1.kr/articles/?3159403>. [Accessed: 06- May- 2018].
- [3] 신정철, "왜 적어야 하나? 2년간 노트를 쓰며 내게 일어난 변화", 마인드와칭, 2018. [Online]. Available: <http://mindwatching.kr/entry/journal-writing>. [Accessed: 16- April- 2018].
- [4] 박해랑, "대학생을 통한 글쓰기의 치료적 효과 연구", 문화와 융합, vol. 37, no. 2, pp. 213-254, 2015.
- [5] "Quantified Self - Self Knowledge Through Numbers", Quantified Self, 2018. [Online]. Available: <http://quantifiedself.com>. [Accessed: 09- Apr- 2018].
- [6] "What is Scrum?", Scrum.org. [Online]. Available: <https://www.scrum.org/resources/what-is-scrum>. [Accessed: 06- May- 2018].
- [7] "Ginger | 사랑을 이해하는 인공지능 앱", iamginger.net, 2018. [Online]. Available: <http://iamginger.net/>. [Accessed: 06- May- 2018].