

O2O Editor

(Offline to Online Editor)

상세설계서

Ver. 1.2

2016. 10. 13.

한국외국어대학교

정보통신공학과

Intermind팀 (6팀)

문서명: 상세설계서

문서 정보

구 분	소 속	성 명	날 짜	서 명
작성자	한국외국어대학교	정재원	2016.10.10	
	한국외국어대학교	김진우		
	한국외국어대학교	김형욱		
	한국외국어대학교	이경민		
	한국외국어대학교	권영은		
검토자	한국외국어대학교	정재원	2016.10.12	
	한국외국어대학교	이경민		
	한국외국어대학교	권영은		
사용자				
승인자	한국외국어대학교	홍 진 표	2016.10.13	

개정 이력

버전	작성자	개정일자	개정 내역	승인자
1.0	김진우	2016.10.10	초안 작성	
	김형욱			
	이경민			
	정재원			
	권영은			
	검토자	이경민, 정재원, 권영은		
1.1	김진우	2016.10.11	수정 및 보완	
	김형욱			
	이경민			
	정재원			
	권영은			
	검토자	정재원		
1.2	권영은	2016.10.12	전체 검토 및 수정	
	김형욱			
	김진우			
	이경민			
	검토자	정재원		
2.0				
	검토자			

목차

1. Introduction.....	6
1.1 제품 설명 및 구성도	6
1.1.1 O2O Editor 시스템 개요.....	6
1.1.2 O2O Editor 시스템 구성도	6
1.2 Detailed Design Document 개요 및 목적	7
1.3 Detailed Design Document 가이드 체계.....	7
2. Design Overview	8
2.1 Introduction.....	8
2.2 System Interfaces MVC with Django Model	8
2.3 class UML	9
2.4 Sequence UML.....	9
3. Object Description & 기능 명세서.....	10
3.1 개요	10
3.2 Object Description.....	10
3.3 use case.....	24

[그림 목차]

[Figure 1] O2O Editor 시스템 구성도	6
[Figure 2] Django and System Interface	8
[Figure 3] class diagram	9
[Figure 4] sequence diagram	9
[Figure 5] Use case.....	24

[표 목차]

[Table 1] O2O System 구성요소	7
---------------------------------	---

1. Introduction

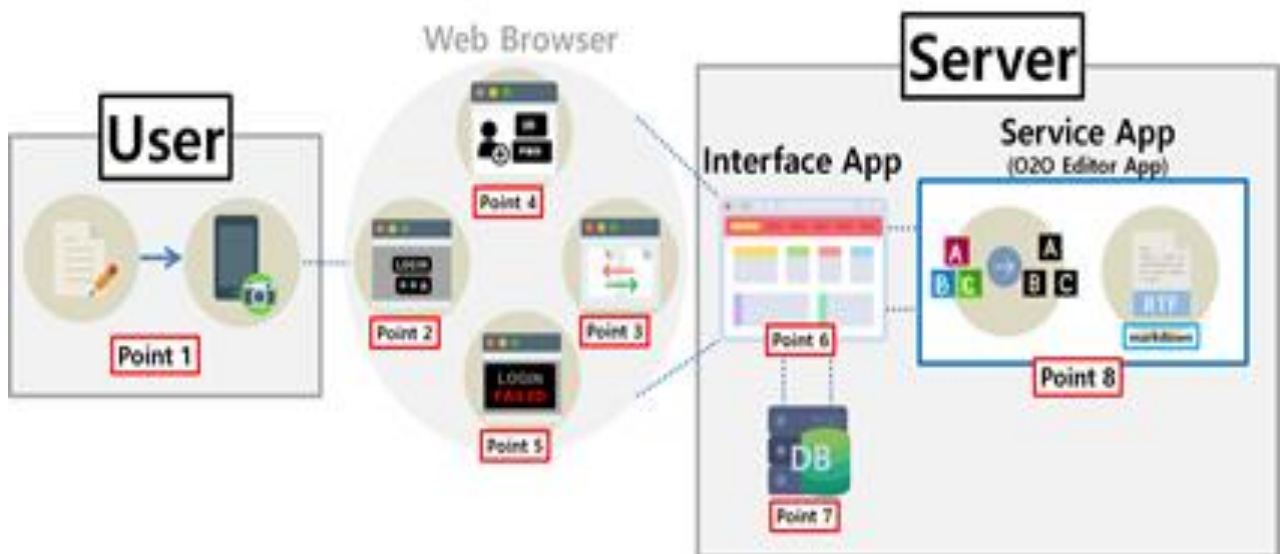
1.1 제품 설명 및 구성도

본 절에서는 O2O Editor 시스템에 대한 개요와 이의 구성도를 제시 한다.

1.1.1 O2O Editor 시스템 개요

O2O Editor(Off-line to On-line Editor) System은 종이에 쓴 hand-writing을 인식하여 저장 후, 편집가능 한 PC File 형태로 바꿔주는 동시, Off-line(hand-writing) 및 On-line(PC Editor) 두 곳 모두에서 편집 서비스를 제공하는 제품모형을 지향하고 있다.

1.1.2 O2O Editor 시스템 구성도



[Figure 1] O2O Editor 시스템 구성도

문서명: 상세설계서

O2O System 구성요소			
User	Interface App	Service App	Database
User는 O2O System Service를 받는 대상이다.	Server에서 운용되는 Application으로 각종 구성요소와 Interface를 담당하고 있다.	Server에서 운용되는 Application으로 실제 O2O Editor Service를 담당하고 있다.	Server에서 운용되는 Database로 Interface App과 Data Language를 통해 정보를 주고 받아야 한다.
User는 본 System을 이용하기 앞서, 제약사항을 감안 해야 하고, 서비스를 받기 위한 Interface 과정을 숙지 해야 한다.	위 그림에서 볼 수 있듯이 Point 2, 3, 4에 해당하는 HTML파일을 Client에게 전송해줄 뿐 아니라, DB, Service App 과의 Interface도 담당하고 있다.	문자인식, RTF 파일 변환 등 실질적으로 주요 Service를 담당하고 있는 Application이다.	Client 정보, Train Data 등이 필요하다.

[Table 1] O2O System 구성요소

1.2 Detailed Design Document 개요 및 목적

이 O2O Editor Design Document는 software의 세부 설계 문서는 개발자 및 참여자들로 하여금 보다 구체적인 System Model을 제시 해주고, 개발에 있어 구체적 목표를 제시함으로써 개인 및 공동 개발에 용이성 제 공할 뿐 아니라 요구사항에 대한 논리성을 보강해줄 수 있다.

1.3 Detailed Design Document 가이드 체계

2.1절에서는 Design Overview 에서는 위에서 제시한 System 구성도 Figure[x]를 구현하기 위한 System Interface를 제시할 것이다.

2.2절에서는 class UML 부분에서는 설계한 class에 대한 대략적인 description (Attributes and methods)과 와 class간 관계를 Graphical하게 제시할 것이다.

3.1절에서는 3.2(Object Descriptions)절에 대한 개요이다.

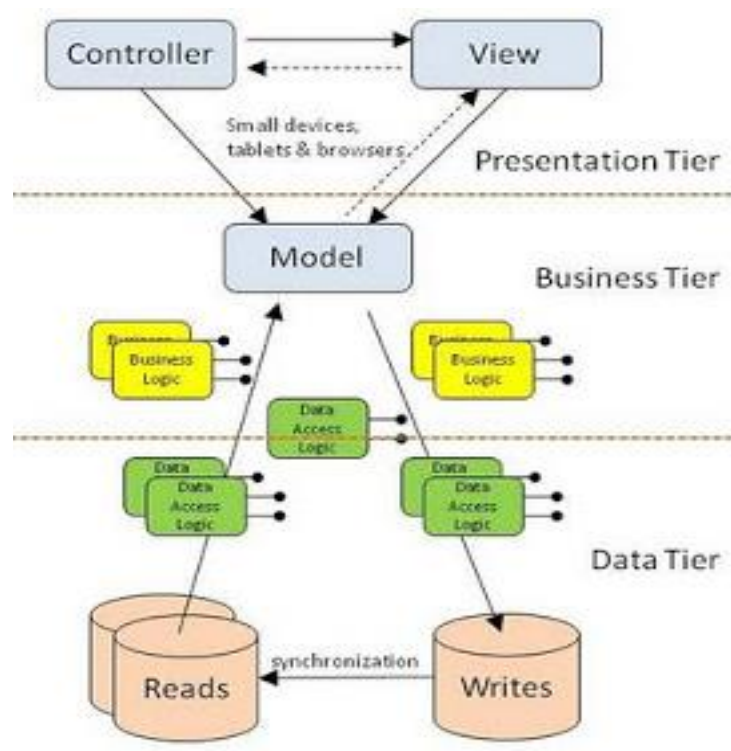
3.2절에서는 실제 Object의 Attributes와 methods를 정의하고, 이에 대한 구체적 설명과 논리적 Programming Sudo Code를 제시하고 있다.

2. Design Overview

2.1 Introduction

Design Overview 장에서는 제품 설계에 대한 간단한 overview를 제시한다.

2.2 System Interfaces MVC with Django Model

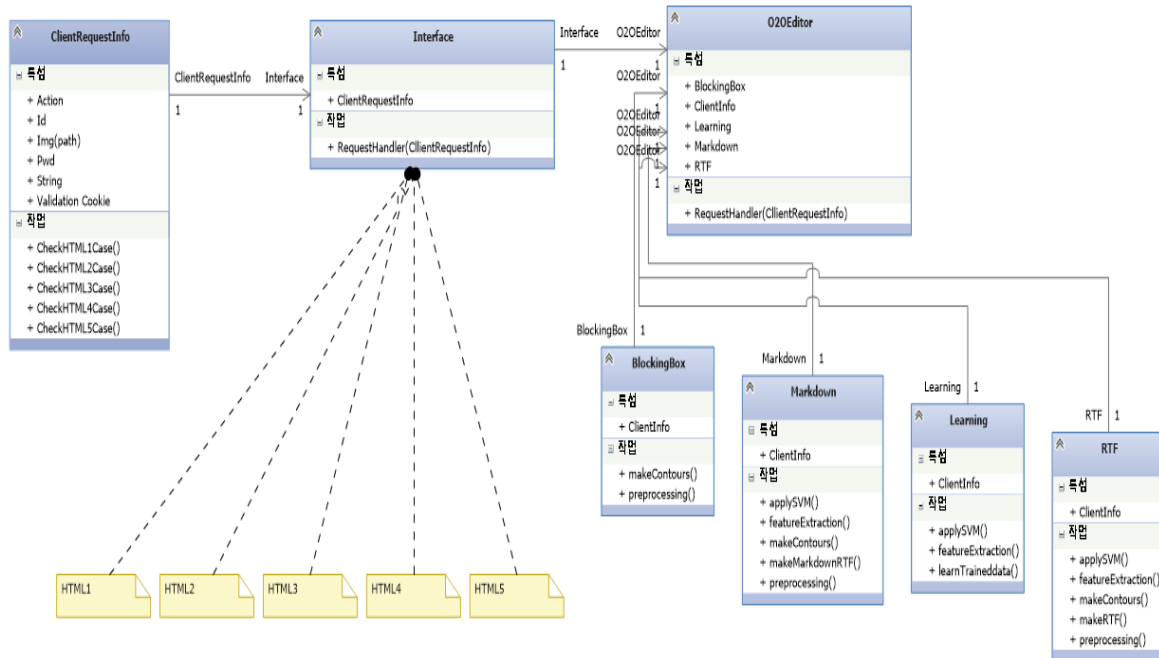


[Figure 2] Django and System Interface

O2O Editor System은 기본 적으로 Django를 기반으로 개발될 예정이고, 따라서 Model-View- Controller 라는 아키텍처 패턴으로 구성될 예정이다. 또한 DB 로는 MYSQL engine을 사용할 예정이고, 서버의 경우 상업 화 이전 까지는 Django에서 제공하는 Test Server를 이용해 local host 로 test를 할 예정이다.

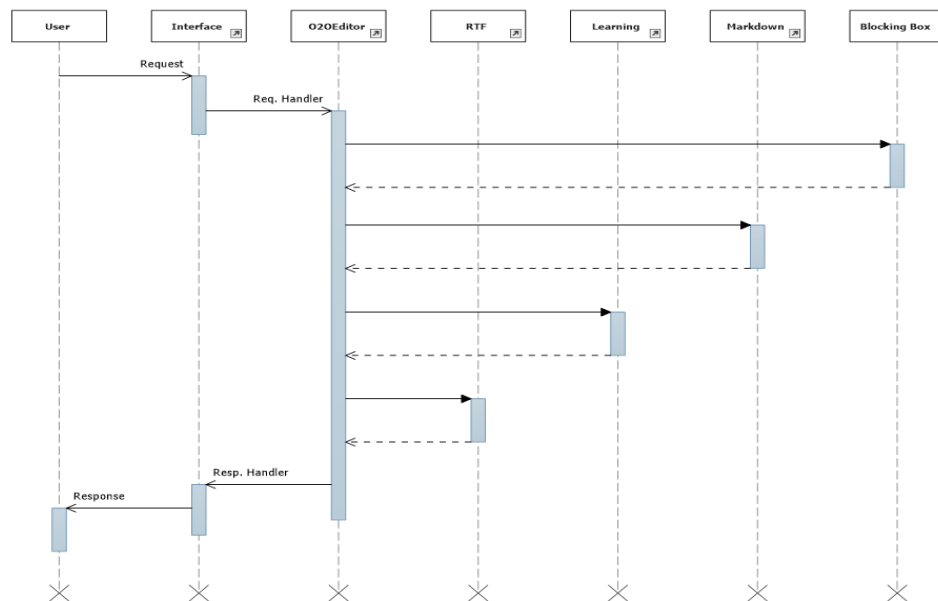
문서명: 상세설계서

2.3 class UML



[Figure 3] class diagram

2.4 Sequence UML



[Figure 4] sequence diagram

3. Object Description & 기능 명세서

3.1 개요

이 장에서는 '시스템 기능 명세서'와 설계한 'Object'를 Mapping하여, 요구 사항과 이에 대한 상세 설계를 병렬적으로 제공하고 있다.

또한 각 상세 설계를 하는 과정에서 모든 것을 Object으로 설계하고 표현하였고 뿐만 아니라 이에 대한 sudo code를 제공함으로써, 설계의 논리성과 객체 간 관계를 명확히 하였다.

이 장에서는 기능 요구서를 얼마나 충실히 반영하여 설계를 했는지와 구체적 설계 내용을 잘 알 수 있는 부분이다.

3.2 Object Description






유형	상세 유형		요구사항 명	ID	비고
시스템 기능 요 구사항 (FR)	서버	Interface Application (FR-I)	로그인 HTML 파일 전송	FR-I-001	
			회원가입 기능 제공	FR-I-002	
			DB에 일치 데이터 요청	FR-I-003	
			로그인 불일치 HTML 파일 전송	FR-I-004	
			Download 기능 제공	FR-I-005	
			Markdown 기능 제공	FR-I-006	
			Learning 기능 제공	FR-I-007	
		Service Application (FR-S)	Get Image file and log-in ID from I.A.	FR-S-001	
			이미지의 이진화 과정 제공	FR-S-002	
			Noise, Dust 티끌제거	FR-S-003	
			Morphological Opening 수행	FR-S-004	
			Blocking 과정	FR-S-005	
			Box 객체에 담긴 신호에 대한 feature extraction	FR-S-006	
			Customized SVM Classifier의 Character Recognition	FR-S-007	
			Box 객체 리스트 RTF에 문자 입력	FR-S-008	
			Markdown 형식으로 변환	FR-S-009	

문서명: 상세설계서

	데이터베이스 기능 요구사항 (FR-D)	Making train data	FR-S-010	
		사용자 회원가입 시 이름, ID, PW를 저장	FR-D-001	
		Log-in인 시도 시 정보의 일치 여부를 서버에 알림	FR-D-002	
		Default Data 관리	FR-D-003	
		사용자의 고유의 Traindata 저장 및 관리	FR-D-004	

상세유형	요구사항 명	ID
Interface Application	로그인 HTML 파일 전송	FR-I-001
	회원가입 기능 제공	FR-I-002
	DB에 일치 데이터 요청	FR-I-003
	로그인 불일치 HTML 파일 전송	FR-I-004
	Download 기능 제공	FR-I-005
	Markdown 기능 제공	FR-I-006
	Learning 기능 제공	FR-I-007
상세설명		
Class name	Interface	
Brief description	이 클래스는 Interface Application에서 요구하는 사항들의 전반적인 부분을 처리하는 클래스이다.	
Attribute (fields)	Attribute Description	
ClientRequestInfo	ClientRequestInfo class의 객체를 생성한다.	
	Program Description Language	
	ClientRequestInfo	
Methods (operations)	Method Description	
RequestHandler(ClientRequestInfo)	사용자의 요청을 분석하여 요청에 따른 html을 반환	

문서명: 상세설계서

	 <p><HTML 1></p>  <p><HTML 2></p>  <p><HTML 3></p>  <p><HTML 4></p>  <p><HTML 5></p>	
	<p style="text-align: center;">Program Description Language</p> <pre> RequestHandler(ClientRequestInfo): if(ClientRequestInfo.CheckHTML1Case()): return HTML1 elif(ClientRequestInfo.CheckHTML1_1Case()): return HTML1 + Error elif(ClientRequestInfo.CheckHTML2Case()): return HTML2 elif(ClientRequestInfo.CheckHTML2_1Case()): return HTML2 + Error elif(ClientRequestInfo.CheckHTML3Case()): return HTML3 elif(ClientRequestInfo.CheckHTML4Case()): return HTML4 elif(ClientRequestInfo.CheckHTML5Case()): return HTML5 </pre>	
Class name	ClienRequestInfo	
Brief description	이 클래스는 사용자의 요청을 분석하여 그에 따른 html을 반환하는 클래스이다.	
Attribute (fields)	Attribute Description	
id	사용자 고유의 id.	

문서명: 상세설계서

	Program Description Language
	String id
pwd	Attribute Description
	사용자의 정보를 보안하기 위한 password
	Program Description Language
	String pwd
img	Attribute Description
	사용자가 서비스 받길 원하는 사용자가 업로드한 이미지.
	Program Description Language
	String img
action	Attribute Description
	사용자가 하고자 하는 요청 내용. (ex. rtf반환, markdown rtf반환, learning 등등)
	Program Description Language
	String action
string	Attribute Description
	사용자가 직접 지정해 준 train할 이미지에 대한 내용.
	Program Description Language
	String string
validatedCookie	Attribute Description
	사용자의 로그인 유지 표시. 쿠키가 유지될 경우 로그인 유지.
	Program Description Language
	validatedCookie
Methods (operations)	Method Description
CheckHTML1Case()	Client가 어떠한 Attribute도 가지고 있지 않을 경우, True를 반환한다.
	Program Description Language
	Boolean CheckHTML1Case(): return All Attributes are false
CheckHTML1_1Case()	Method Description
	Client의 ID와 pwd가 존재하지 않을 경우, True를 반환한다.
	Program Description Language
	Boolean CheckHTML1_1Case(): return non validation of id, pwd
CheckHTML2Case()	Method Description
	Client가 회원가입에 필요한 정보(ID,pwd,이름)만을 가지고 있을 경우, true를 반환한다.
	Program Description Language
	Boolean CheckHTML2Case():

문서명: 상세설계서

	Return Only (ID, pwd, 이름) exist
CheckHTML2_1Case()	Method Description
	Client가 HTML2의 회원가입 Form에서 작성한 id와 password에 해당하는 값이 이미 database에 있을 경우, method가 체크하는 HTML2_1에 해당하는 Case이므로 True를 반환한다.
	Program Description Language
	Boolean CheckHTML2_1Case(): Return already id, pwd in database
CheckHTML3Case()	Method Description
	Client가 작성한 Log-in Form의 id와 pwd가 데이터베이스 상에서 존재할 경우, method가 체크하는 HTML3에 해당하는 Case이므로 True를 반환한다.
	Program Description Language
	Boolean CheckHTML3Case(): return validation of id, pwd
CheckHTML4Case()	Method Description
	Client가 HTML3에서 취한 행동을 명시하는 action에 대한 값이 'convert' 이거나 'markdown' 일 경우, method에서 체크하는 HTML4에 해당하는 Case이므로 True를 반환한다.
	Program Description Language
	Boolean CheckHTML4Case(): if action == 'convert' action == 'markdown': return True
CheckHTML5Case()	Method Description
	Client가 HTML3에서 취한 행동을 명시하는 action에 대한 값이 'learning'일 경우, method에서 체크하는 HTML5에 해당하는 Case이므로 True를 반환한다.
	Program Description Language
	Boolean CheckHTML5Case(): if action == 'learning': return True
Version	<1.0>
Date	2016.10.12
Importance	상
Level of Difficulty	중
Administrator	김진우, 권영은

상세유형	요구사항 명	ID
Service Application	Get Image file and log-in ID from I.A	FR-S-001

문서명: 상세설계서

	이미지의 이진화 과정 제공	FR-S-002
	Noise, Dust 티끌제거	FR-S-003
	Morphological Opening 수행	FR-S-004
	Blocking 과정	FR-S-005
	Box class를 만들어 각 box을 객체화하고 정보 측정 및 저장	FR-S-006
	Box 객체에 담긴 신호에 대한 feature extraction	FR-S-007
	Customized SVM Classifier의 Character Recognition	FR-S-008
	Box 객체 리스트 RTF에 문자 입력	FR-S-009
상세설명		
Class name	RTF class(ClientRequestInfo.img)	
Brief description	이 클래스는 O2OEditor class의 객체로, 사용자의 이미지 파일을 사용자의 문자 정보를 이용해 인식한 후, rtf파일로 변환하는 클래스이다.	
Attribute (fields)		Attribute Description
ClientRequestInfo.img	Learning 및 preprocessing 된 이미지가 저장되는 변수이고, 이 변수를 통해 char를 만들어 준다.	
	Program Description Language	
	Image ClientRequestInfo.img	
coords	Attribute Description	
	좌표에 대한 정보가 저장되는 변수이다.	
	Program Description Language	
character	Attribute Description	
	ClientRequestInfo.img의 파일이 좌표정보와 함께 .docx에 저장되므로, character 형태로 문자가 저장되어야 한다.	
	Program Description Language	
	char character	
	Method Description	
preprocessing(ClientRequestInfo)	ClientRequestInfo의 image에 대해 전처리 과정을 실행한다. 과정을 3단계로 나뉘지며, 각 과정은 이진화, 티끌제거, 형태학적 열기를 실행한다.	
	Program Description Language	
	<pre> preprocessing(ClientRequestInfo): binarization(ClientRequestInfo.image) // ClientRequestInfo.image를 이진화. eraseDust(ClientRequestInfo.image) // ClientRequestInfo.image의 티끌을 제거. </pre>	

문서명: 상세설계서

	morphologicalOpenning(ClientRequestInfo.image) // ClientRequestInfo.image에 대해 형태학적 열기.
makeContours(ClientRequestInfo.img)	Method Description
	ClientRequestInfo의 image의 전처리 과정을 거친 이후에 각 이미지에 대해 boxing 과정을 거쳐서 이미지를 뽑아주어야 한다. Boxing 과정(즉 makeContours를 거쳐야만 각 이미지에 대한 정보를 추출할 수 있다.)
	Program Description Language
	makeContours(ClientRequestInfo.img): makeBoxing(ClientRequestInfo.img) // ClientRequestInfo.image를 blocking 작업 Return blockedImg
featureExtraction(blockedImg)	Method Description
	윤곽화된 문자 이미지로부터 문자인식을 위한 특징을 추출한다.
	Program Description Language
	featureExtraction(blockedImg) : charFeature = getFeature(blockedImg) // 이미지 신호로부터 문자인식을 위한 특징을 추출한다. return charFeature
applySVM(charFeature)	Method Description
	한 문자 신호에 대한 특징(charFeature)을 입력으로 받아서 해당 문자에 대한 문자인식 결과를 반환한다.
	Program Description Language
	applySVM(charFeature) : selectMyTrainedData(ClientRequestInfo.ID) // ID에 맞는 trained데이터를 TrainedData.db에서 가져온다. characterRecognition(trainedData, charFeature) // trainedData와 featureExtraction에서 얻은 charFeature로 문자인식을 수행한다. return character
makeRTF((character, coordinate))	Method Description
	RTF문서 상, 주어진 좌표에 해당하는 위치에 문자를 배치한다

문서명: 상세설계서

	다.
	Program Description Language
	makeRTF((character, coordinate)): locateCharacter(character, coordinate) // 좌표의 위치로 글자 배치.
Version	<1.0>
Date	2016.10.12
Importance	상
Level of Difficulty	상
Administrator	이경민

상세유형	요구사항 명	ID
Service Application	Markdown 기능 제공	FR-I-009
	이미지의 이진화 과정 제공	FR-S-002
	Noise, Dust 티끌제거	FR-S-003
	Morphological Opening 수행	FR-S-004
	Blocking 과정	FR-S-005
	Box class를 만들어 각 box을 객체화하 고 정보 측정 및 저장	FR-S-006
	Box 객체에 담긴 신호에 대한 feature extraction	FR-S-007
	Customized SVM Classifier의 Character Recognition	FR-S-008
	Box 객체 리스트 RTF에 문자 입력	FR-S-009
	Markdown 형식으로 변환	FR-S-010
	완성된 Rtf 파일을 I.A. 에 전달	FR-S-011
상세설명		
Class name	Markdown class(ClientRequestInfo)	
Brief description	이 클래스는 이미지 파일을 받아 전처리 과정 후 문자인식을 한 내용을 markdown 문법을 적용해 rtf파일로 저장하는 클래스이다.	
Attribute (fields)		Attribute Description
ClientRequestInfo	사용자가 요청한 image 및 사용자의 ID가 담겨있는 클래스. 사용자의 ID와 TrainedData DB를 사용해 customize된 문자인식 시스템을 제공한다.	
	Program Description Language	
	ClientRequestInfo ClientRequestInfo	
Cords	Attribute Description	
	좌표에 대한 정보가 저장되는 변수이다.	
	Program Description Language	
		Tuple <plot x, plot y>

문서명: 상세설계서

Character	Attribute Description
	ClientRequestInfo.img의 파일이 좌표정보와 함께 .docx에 저장이 되므로, character 형태로 문자가 저장되어야 한다.
	Program Description Language
	char character
Methods (operations)	Method Description
preProcessing(ClientRequestInfo.img)	ClientRequestInfo의 image에 대해 전처리 과정을 실행한다. 과정을 3단계로 나뉘지며, 각 과정은 이진화, 티끌제거, 형태학적 열기를 실행한다.
	Program Description Language
	<pre> preprocessing(ClientRequestInfo.img): binarization(ClientRequestInfo.img) // ClientRequestInfo.image를 이진화. eraseDust(ClientRequestInfo.img) // ClientRequestInfo.image의 티끌을 제거. morphologicalOpenning(ClientRequestInfo.img) // ClientRequestInfo.image에 대해 형태학적 열기. </pre>
makeContours(ClientRequestInfo.img)	Method Description
	ClientRequestInfo의 image의 전처리 과정을 거치기 전에 각 이미지에 대해 boxing 과정을 거쳐서 이미지를 뽑아주어야 한다. 전처리 과정을 하기 위해선 여러 character를 윤곽화(boxing)하는 작업을 해야 각 이미지를 전처리 하기에 수월하다.
	Program Description Language
	<pre> makeContours(ClientRequestInfo.img): makeBoxing(ClientRequestInfo.img) // ClientRequestInfo.image를 blocking 작업 Return blockedImg </pre>
featureExtraction(blockedImg)	Method Description
	윤곽화된 문자 이미지로부터 문자인식을 위한 특징을 추출한다.
	Program Description Language
	<pre> featureExtraction(blockedImg) : charFeature = getFeature(blockedImg) </pre>

문서명: 상세설계서

	// 이미지 신호로부터 문자인식을 위한 특징을 추출한다. return charFeature
applySVM(charFeature)	Method Description
	한 문자 신호에 대한 특징(charFeature)을 입력으로 받아서 해당 문자에 대한 문자인식 결과를 반환한다.
	Program Description Language
	applySVM(charFeature) : selectMyTrainedData(ClientRequestInfo.ID) // ID에 맞는 trained데이터를 TrainedData.db에서 가져온다. characterRecognition(trainedData, charFeature) // trainedData와 featureExtraction에서 얻은 charFeature로 문자인식을 수행한다. return character
makeMarkdownRTF((character, coords))	Method Description
	RTF문서 상, 주어진 좌표에 해당하는 위치에 markdown 문법이 적용된 문자를 배치한다.
	Program Description Language
	makeMarkdownRTF((character, coordinate)): locateCharacter(coordinate) // 좌표의 위치로 글자 배치. markdownString(characters) // 문자가 이룬 문장에 대해 markdown문법 적용
Version	<1.0>
Date	2016.10.12
Importance	상
Level of Difficulty	상
Administrator	이경민

상세유형	요구사항 명	ID
Service Application	Get Image file and log-in ID from I.A	FR-S-001
	이미지의 이진화 과정 제공	FR-S-002
	Noise, Dust 티끌제거	FR-S-003
	Morphological Opening 수행	FR-S-004

문서명: 상세설계서

	Blocking 과정	FR-S-005
	Box class를 만들어 각 box를 객체화 하고 정보 측정 및 저장	FR-S-006
상세설명		
Class name	BlockingBox class(Image ClientRequestInfo.img)	
Brief description	이 클래스는 이미지 파일을 받아 윤곽(blocking)작업을 한 뒤 전처리 과정을 거치는 클래스이다.	
Attribute (fields)		Attribute Description
ClientRequestInfo.img	처음에 윤곽이 잡힌 이미지 파일을 저장하고 전처리 과정이 끝난 후에는 이 변수에 저장을 한 뒤 반환을 한다.	
	Program Description Language	
	Image ClientRequestInfo.img	
Methods (operations)		Method Description
preProcessing(ClientRequestInfo.img)	ClientRequestInfo의 image에 대해 전처리 과정을 실행한다. 과정을 3단계로 나뉘지며, 각 과정은 이진화, 티끌제거, 형태학적 열기를 실행한다.	
	Program Description Language	
	<pre> preprocessing(ClientRequestInfo.img): binarization(ClientRequestInfo.img) // ClientRequestInfo.image를 이진화. eraseDust(ClientRequestInfo.img) // ClientRequestInfo.image의 티끌을 제거. morphologicalOpenning(ClientRequestInfo.img) // ClientRequestInfo.image에 대해 형태학적 열기. </pre>	
makeContours(ClientRequestInfo.img)	Method Description	
	ClientRequestInfo의 image의 전처리 과정을 거치기 전에 각 이미지에 대해 boxing 과정을 거쳐서 이미지를 뽑아주어야 한다. 전처리 과정을 하기 위해선 여러 character를 윤곽화(boxing)하는 작업을 해야 각 이미지를 전처리 하기에 수월하다.	
	Program Description Language	
	<pre> makeContours(ClientRequestInfo.img) makeBoxing(ClientRequestInfo.img) // ClientRequestInfo.image를 boxing 작업 </pre>	
Version		<1.0>

문서명: 상세설계서

Date	2016.10.12
Importance	상
Level of Difficulty	상
Administrator	이경민

상세유형	요구사항 명	ID
Service Application	Box 객체에 담긴 신호에 대한 feature extraction	FR-S-007
	Customized SVM Classifier의 Character Recognition	FR-S-008
	Box 객체 리스트 RTF에 문자 입력	FR-S-009
	사용자의 고유의 Traindata 저장 및 관 리	FR-D-004
상세설명		
Class name	Learn class(ClientRequestInfo)	
Brief description	이 클래스는 사용자에게 제공한 learnig html을 통해 얻은 데이터로부터 trained data를 얻을 후 데이터베이스에 저장하는 클래스이다.	
Attribute (fields)		Attribute Description
ClientRequestInfo		사용자가 요청한 image 및 사용자의 ID가 담겨있는 클래스. 사용자의 ID와 TrainedData DB를 사용해 customize된 문자인 식 시스템을 제공한다.
		Program Description Language
		ClientRequestInfo ClientRequestInfo
Cords		Attribute Description
		좌표에 대한 정보가 저장되는 변수이다.
		Program Description Language
Character		Tuple <plot x, plot y>
		Attribute Description
		ClientRequestInfo.img의 파일이 좌표정보와 함께 .docx에 저장 이 되므로, character 형태로 문자가 저장 이 되어야 한다.
		Program Description Language
String		char character
		Attribute Description
		Learning html로부터 얻은 사용자가 지정한 문장의 내용.
		Program Description Language
Methods (operations)		Method Description
		사용자의 문자정보가 담긴 TrainedData를 생성하고 이를, 데 이터 베이스에 저장한다.
learnTrainedData(blockedImg, string)		Program Description Language

문서명: 상세설계서

	learnTrainedData(blockedImg, string) : TrainedData = makeTrainedData(blockedImg, string) // 사용자가 지정한 문장의 정보(string)에 따라 문자정보가 담긴 TrainedData를 생성. addTrainedDataToDB(TrainedData) // 사용자의 문자정보가 담긴 TrainedData를 TrainedData.db에 저장 <1.0>
Version	2016.10.12
Date	상
Importance	상
Level of Difficulty	이경민
Administrator	

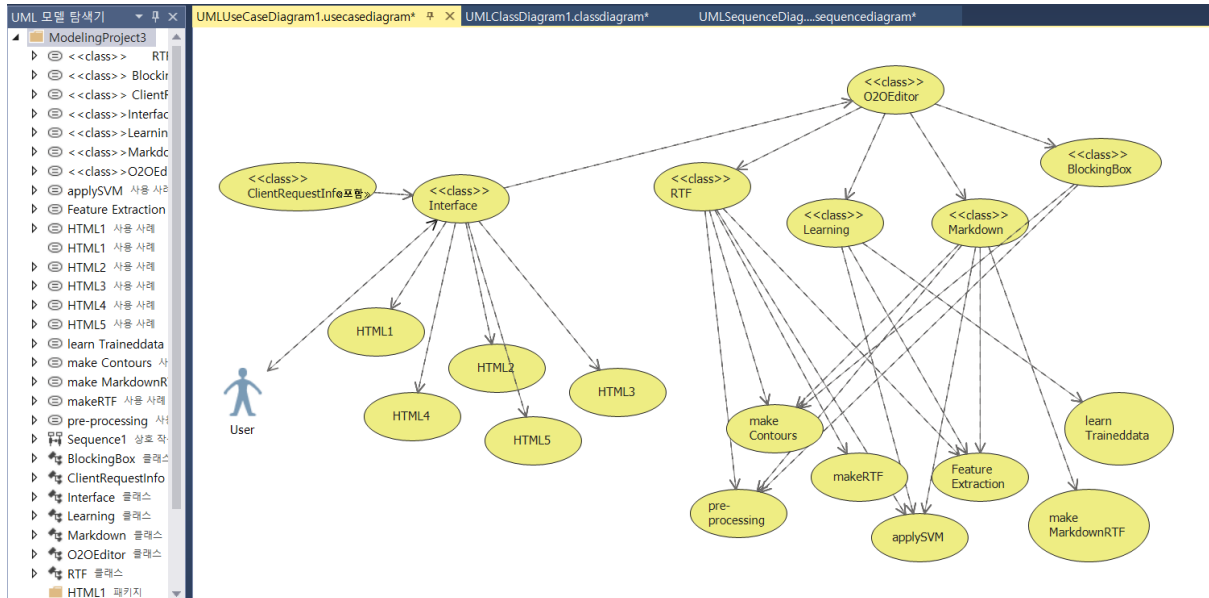
상세유형	요구사항 명	ID
데이터베이스 기능 요구사항	사용자 회원가입 시 이름, ID, PW를 저장	FR-D-001
	Log-in인 시도 시 정보의 일치 여부를 서버에 알림	FR-D-002
	Default Data 관리	FR-D-003
	사용자의 고유의 Traindata 저장 및 관리	FR-D-004
상세설명		
Class name	Database	
Brief description	이 클래스는 '데이터베이스'에서 요구하는 사항들의 전반적인 부분으로 Client의 id와 password를 저장하고, Client 개인의 TrainedData를 저장	
Attribute (fields)	Attribute Description	
id	Client.db의 Primary가 되는 사용자 고유의 ID	
	Program Description Language	
	String id	
pwd	Attribute Description	
	사용자의 정보를 보안하기 위한 pwd	
	Program Description Language	
	String pwd	
traindata	Attribute Description	
	사용자의 개인의 필체 정보를 담고 있는 data	
	Program Description Language	
	traindata	
Methods (operations)	Method Description	
InsertTuple()	Client.db에 저장하기 위한 데이터의 묶음	

문서명: 상세설계서

	Program Description Language
	InsertTuple(ClientRequestInfo.id,ClientRequestInfo.pwd): id = ClientRequestInfo.id pwd = ClientRequestInfo.pwd // 데이터베이스(Client.db)에 아이디, 비밀번호 삽입
CheckTuple()	Method Description
	로그인 정보 tuple과 데이터 베이스의 정보가 일치하는지 확인
	Program Description Language
	CheckTuple(ClientRequestInfo.id,ClientRequestInfo.pwd): return (id,pwd) in Client.db
AddTrainDataToDB	Method Description
	사용자의 개인의 traineddata를 client.db에 저장
	Program Description Language
	AddTrainDataToDB(TrainedData): //데이터베이스(Client.db)에 Client의 traineddata 저장
Version	<1.0>
Date	2016.10.12
Importance	상
Level of Difficulty	중
Administrator	김형욱

3.3 use case

<사용자가 문자인식 service를 받는 routine>



[Figure 5] Use case

1,

User 의 request message 에 따라 총 5 개의 Html 화면을 경험할 수 있다.

각각은 로그인 페이지, 회원가입, service 페이지 등등이 된다.

이는 Interface class 에서 판단할 일이다.

2.

사용자가 log in 에 성공했고, Image File 을 올리라는 Form 작성해 서버로 그 정보가 왔다고 해보자 이제 Interface class 는 o2o editor class 에게 request 에 대한 정보를 넘긴다. 그리고 Action 을 통해 어떤 요구였는 지도 전달한다.

3.

Upload 된 이미지 파일을 기반으로 o2o Editor 는 action 에 맞는 일을 해서 다시 결과를 interface app 에게 넘겨준다. 아마 편집가능한 pc 문서 파일 정도가 될 것이다.

4.

마지막으로 이 편집가능한 pc 문서를 다운받을 수 있는 html 을 전송함으로써 Case 의 routine 이 끝이 난다.