

# 종합설계 프로젝트 수행 보고서

프로젝트명	3D프린팅 오브젝트 사이트
팀번호	S4-4
문서제목	수행계획서( O ) 2차발표 중간보고서( O ) 3차발표 중간보고서( O ) 최종결과보고서( O )

2020.12.04

팀원 : 손두영  
김명준  
성호중

지도교수 : 한익주 교수 (인)

교수님 메일 확인으로 통과했음을 알립니다.

## 문서 수정 내역

작성일	대표작성자	버전(Revision)	수정내용	
2020.01.20	손두영	1.0	수행계획서	최초작성
2020.03.02	성호중	2.0	2차발표자료	설계서추가
2020.05.02	김명준	3.0	3차발표자료	시험결과추가
2020.06.25	손두영	4.0	4차발표자료	시험결과추가
2020.12.04	손두영	5.0	최종자료	결과 추가

## 문서 구성

진행단계	프로젝트 계획서 발표	중간발표1 (2월)	중간발표2 (5월)	학기말발표 (6월)	최종발표 (10월)
기본양식	계획서 양식	계획서 양식	계획서 양식	계획서 양식	계획서 양식
포함되는 내용	I. 서론 (1~6) II. 본론 (1~3) 참고자료	I. 서론 (1~6) II. 본론 (1~4) 참고자료	I. 서론 (1~6) II. 본론 (1~5) 참고자료	I. 서론 (1~6) II. 본론 (1~7) 참고자료	I. 서론 (1~6) II. 본론 (1~7) 참고자료 III. 결론

이 문서는 한국산업기술대학교 컴퓨터공학부의 “종합설계”교과목에서  
프로젝트“3D프린팅 오브젝트 사이트”를 수행하는  
(S1-4, 손두영, 김명준, 성호중)들이 작성한 것으로 사용하기 위해서는  
팀원들의 허락이 필요합니다.

# 목 차

## I. 서론

1. 작품선정 배경 및 필요성 .....	4
2. 기존 연구/기술동향 분석 .....	5
3. 개발 목표 .....	6
4. 팀 역할 분담 .....	6
5. 개발 일정 .....	6
6. 개발 환경 .....	7

## II. 본론

1. 개발 내용 .....	8
2. 문제 및 해결방안 .....	8
3. 시험시나리오 .....	9
4. 상세 설계 .....	9
5. Prototype 구현 .....	11
6. 시험/ 테스트 결과 .....	12
7. Coding & DEMO .....	13

## III. 결론

1. 연구 결과 .....	17
2. 작품제작 소요재료 목록 .....	21

참고자료 .....	22
------------	----

## I. 서론

### 1. 작품 선정 배경 및 필요성

3D프린터가 언론이나 매체 속에서 이슈가 된 것은 오래 전 일이 아니다. 제 3의 산업혁명, 제조업의 혁신 등 3D프린터의 기술 앞에는 이러한 수식어들이 붙는다. 3D프린터가 사회적 이슈가 된 것은 가상에서 상상할 수 있었던 기술이 눈앞에서 점점 우리의 생활에 파고드는 혁신적인 기술이기 때문이다. 지금껏 제조의 방식은 자재를 깎아 만들거나 금형 틀에서 찍어내는 방식이었다. 이러한 방식을 탈피한 3D프린팅의 제조 기술은 혁신적일 수밖에 없다. 하지만 그렇다고 해서 3D프린팅의 기술은 최근에 개발된 새로운 기술은 아니다. 이미 3D프린터는 30여 년 전부터 개발되기 시작하였으며 성장을 이어오고 있다. 오늘날 3D프린터는 크게 3가지로 구분할 수 있다. 조형되는 방식과 그 재료의 형태에 따라 SLA, FDM, SLS 방식으로 나누어지고 있으며 이중 가장 많이 보급되어 사용되는 것이 FDM 방식의 3D프린터이다. FDM방식의 프린터는 필라멘트 형식의 수지를 사용하는 방식으로 다른 방식의 3D프린터보다 선호도가 높는데 제품과 원료의 가격이 저렴하고 출력물의 강한 내구성이 높다는 점 등의 장점이 많아 주로 개인이나 소규모 사업장에서 많이 사용되고 있다. 이러한 장점들은 보다 더 좋은 퀄리티의 출력물을 만들어 내기 위한 개발이 다양하게 진행되고 있다. 또한 FDM방식의 3D프린터는 소규모 중소기업이나 개개인의 다품종 소량생산들의 목적으로 한 제품의 양산에도 많이 사용되고 있다. 하지만 지금의 3D프린터가 시제품 제작이나 파일형 제품의 양산에도 활용되고 있는 등 좋은 반응에 반하여 여러 가지 문제점들이 소량생산을 목적으로 하는 사용자들에게 불편함이 느끼게 하는 단점으로 대두 되고 있다. 이러한 문제점들은 다음과 같다

첫째, 제품의 개당 출력 시간에 따른 수량이다. 1인 제조업의 선호도가 가장 좋은 FDM방식의 3D프린터는 한 개의 노즐이 x축, y축, z축을 디오하면서 익스트루더에 의해 적층하는 방식으로 다른 방식의 3D프린터보다 제품의 조형시간이 가장 느리다.

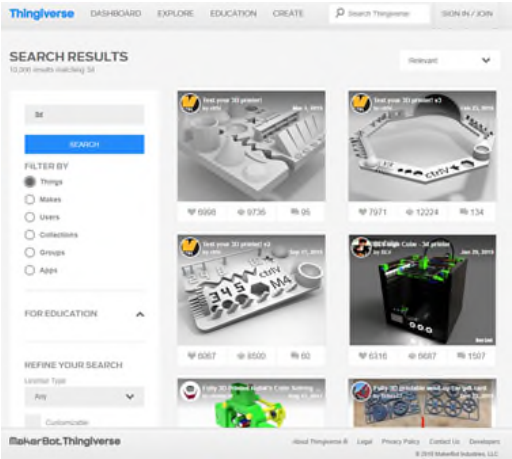
둘째, 동시에 여러 조형물을 출력하기 위해서는 다수의 3D프린터를 설치해야 한다. 복수 출력이 어려운 현재 제품의 특성으로는 여러 대의 3D프린터를 설치해야 하는데, 이럴 경우 프린터기 세팅을 위한 공간적 문제와 함께 각각 제어할 수 있는 프로그램을 구동시키기 위해서 각각의 3D프린터기에 PC와 프로그램을 별도 세팅해야하는 불편함이 있다.

셋째, 일반인들 즉 초보자들이 3D프린터에 접근하는데 어려움이 있다. 위에서 설명했듯이 여러 방식의 3D프린터들이 있고 중소기업 등 회사에서 3D프린터를 주로 사용하고 있지만 일반인들이 3D프린터를 구입하여 써보기에는 3D 도면을 얻기 힘들고 회사처럼 캐드를 직접 사용하여 3D모델링을 하는데에 어려움을 느낀다.

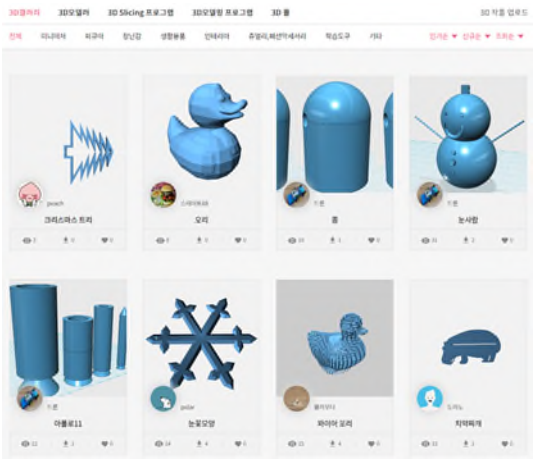
우리는 3D프린터가 현재 많은 이슈가 되고 있고 여러 기업들에서 사용하고 있지만 초보자들도 사용할 수 있도록 하기 위해 세 번째 문제를 다룬다. 웹사이트를 통해 자신들이 원하는 3D도면을 빠르고 쉽게 제공해주고 3D모델링 프로그램을 직접 다운로드 하지 않고 웹사이트를 통해 모델링을 직접 설계하고 만들어 줌으로써 일반인들도 3D프린터에 쉽게 접근할 수 있도록 하는 것에 목적을 두고 있다.

2. 기존 연구/기술 동향 분석

싱기버스(Thingivers<sup>3)</sup>) 사례로 현재 가장 많은 양의 도면 데이터를 가지고 있으나  
보유카테고리 분류 부족이며 영문버전이라 한국인이 이용하기 불편 초보자들은 특히 원하는 데이  
터를 얻는데 힘들고 사이트 자체에서 3D 도면 제작 불가능으로 판단.



(원)싱기버스 (오)3DCH



한국싱기버스(3dch<sup>4</sup>)외국버전에 비해 데이터 양이 부족 카테고리는 다양하게 분류 되어있음  
한국인들이 편리하게 사용할수 있게 되어있음 사이트 자체에서 3D도면 제작 불가능  
초보자들이 원하는 데이터를 쉽게 얻지 못한다는 사례로 판단.

### 3. 개발 목표

- ▶ 초심자들에게 도면을 쉽게 제공해주고 쉽게 3D프린팅에 접할 수 있는 사이트를 제작
- ▶ 3D프린터와 사이트 간에 백엔드로 연동
- ▶ 사이트 내에서 쉽게 일반인들도 3D도면을 제작할 수 있는 툴을 제공. 도면을 받으면 바로 사이트에서 수정이 가능하다는 장점 효과 기대치
- ▶ 카테고리를 여러 가지로 쉽게 나누고 여러 가지 도면 데이터를 확보

### 4. 팀 역할 분담

이름	역할
손두영	반응형 웹사이트 구현 및 CSS, JQuery 응용계층 접목 및 프로젝트 업무일지 보고서 담당
김명준	Big Data 수집 및 STL <sup>1)</sup> to GCODE <sup>2)</sup> 의 웹프로그래밍 응용 연구 및 팀 프로젝트 소스 작성
성호중	STL to GCODE Converting 구현(Cura <sup>5)</sup> Open Source 이용하여 구현), 팀 전체 프로젝트 소스 피드백 및 관리

### 5. 개발 일정

항목	1월	2월	3월	4월	5월	6월	7월	8월	9월	10월
요구사항 정리 및 분석										
시스템 설계 및 상세 설계										
시스템 구현 및 관리										
Try and Error										
문서화 및 발표 준비										
졸업작품 최종점검 및 작성										

## 6. 개발 환경

개발 환경	
OS	Windows 10
웹 서버	Apache, Tomcat
검색 엔진	루씬(Elasticsearch)
데이터베이스	MySQL, MariaDB
개발 언어	Java, HTML, CSS
3D프린터 연동	Cura
하드웨어	3D 프린터
3D모델링	CAD, 123D Design

### 웹사이트

- ▶ Windows 환경에서 Java와 HTML을 이용한 사이트 구현
- ▶ 3D프린터와 Converting 구현 -> STL to GCODE)
- ▶ 빅데이터 수집
- ▶ 3D모델링 도면 -> CAD

### Web

- ▶ Apache Tomcat을 이용한 서버 구축
- ▶ MySQL, MariaDB를 이용한 DB구축
- ▶ 루씬<sup>6)</sup>(Elasticsearch)를 이용한 검색 엔진 구축

## II. 본론

### 1. 개발 내용

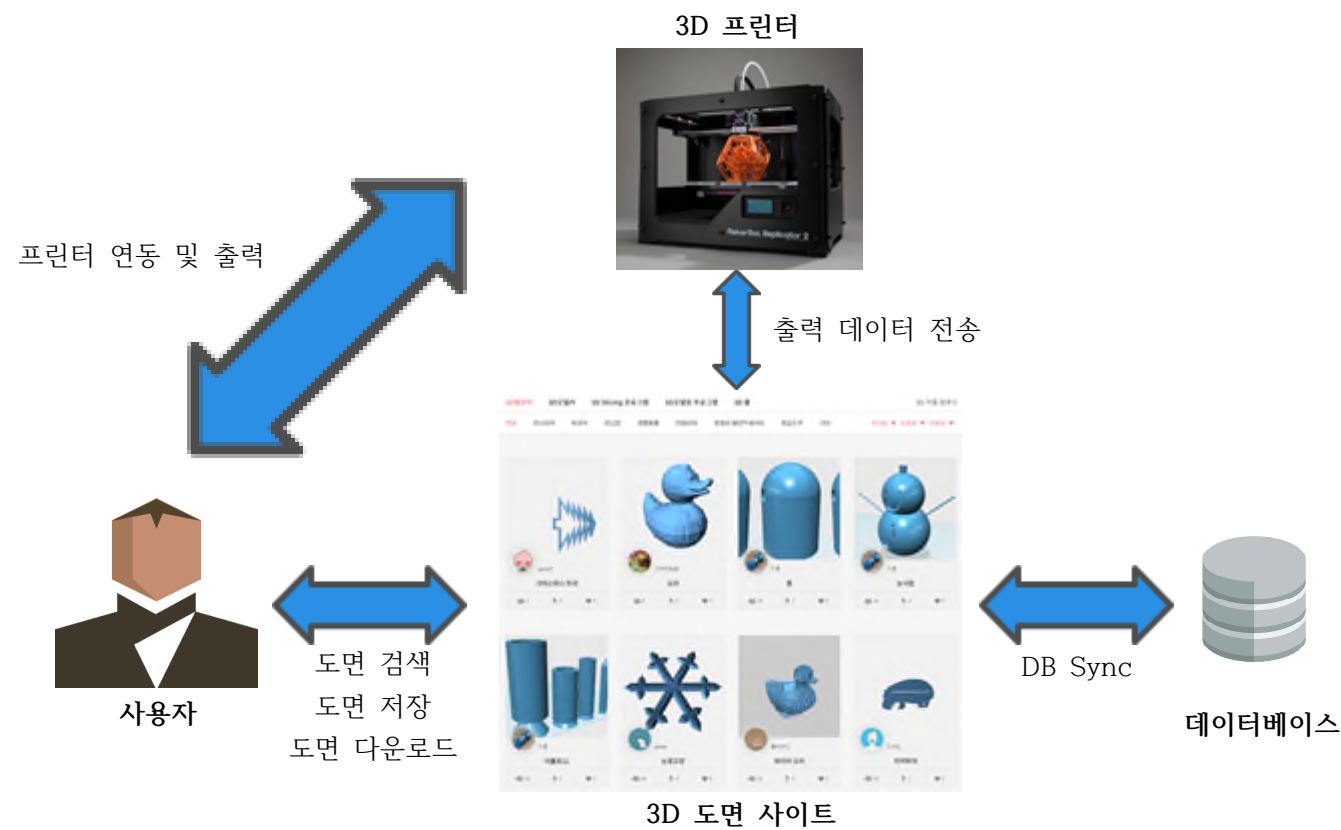
- 1) 3D프린트 도면 제작 프로그램에 대한 이해도 개발  
도면 제작 프로그램에 대한 웹의 접목시키는 것을 개발
- 2) 3D프린트 도면을 홈페이지에서 심플하게 수정 할 수 있는지?  
도면 제작 프로그램의 간단하게 웹 상에서도 수정할수 있는지 테스트 개발
- 3) 3D프린트 stl/obj code의 변환에 대한 이해도 개발  
STL/OBJ 코드의 변환으로 자동화 시키는 것에 대해 개발
- 4) 3D프린트 웹상의 접
- 5) 카테고리 개발  
검색 엔진을 이용하여 더 쉽고 간편하게 사용 할 수 있게 개발

### 2. 문제 및 해결 방안

- 1) 3D프린터에 대한 배경지식 부족.
  - 3D프린터에 대하여서 배울 기회가 존재하지 않았음. 그렇기 때문에 3D프린터에 관한 배경지식이 충분하지 않고, 종합설계를 진행하는데에 있어서 많은 어려움이 있을 것으로 예상됨. 이에 대하여 지인이나 참고문헌등을 통하여 관련 지식을 습득하고 종합설계에 적용할 것
- 2) 웹 개발에 대한 미숙함.
  - 웹 개발 환경 자체(JAVA)에 대해서는 익숙하나 HTML이나 JSP같은 웹 프로그래밍 언어에 미숙함. 웹 프로그래밍 강의를 들은 팀원이 있지만 대규모 프로젝트로는 처음이기에 시작부터 어떻게 해야할지 많은 고민이 되었음. 특히나 검색부분에 대하여 많은 고민이 있었음. 이에 대하여 팀원끼리 이야기를 하고, 스터디를 하면서 지식을 넓히면서 종합설계를 진행해 나갈 것



3. 시험시나리오



4. 상세 설계

1) DataBase

데이터베이스는 MySQL을 사용하고 ENGINE은 MariaDB를 사용하였다. 테이블은 사용자의 계정을 나타내는 user와 도면게시글을 관리하기 위한 user가 있다. 각각의 테이블은 다음과 같이 구성되어 있다.

(1) user

- userID : 사용자의 ID (PK)
- userName : 사용자 이름
- userPassword : 사용자 계정의 비밀번호
- userNickname : 사용자가 사이트에서 활동할 때 사용하는 이름
- userEmail : 사용자의 e-mail 주소

## (2) content

- seq : 게시번호
- userNickname : 사용자가 사이트에서 활동할 때 사용하는 이름
- view : 조회수
- download : 다운로드 횟수
- favorite : 좋아요 개수
- thumbnail\_img : 썸네일 이미지
- comment : 댓글

## 2) Page

제작하고자 웹사이트의 구성으로는 크게 로그인페이지, 도면 목록, 도면 상세페이지, 사용자 정보페이지 등으로 구성되어있다.

(1) loginPage.jsp : 로그인페이지로 ID와 비밀번호를 입력하여 로그인하는 페이지이다.

(2) modelListPage.jsp : 사이트에 올라와있는 도면의 목록을 나타내는 페이지이다.

(3) modelPage.jsp : 하나의 도면의 정보를 나타내는 페이지이다.

(4) userPage.jsp : 사용자 정보(사용자가 올린 도면의 목록, 사용자가 사용하는 3D프린터의 정보 등)를 나타내는 페이지이다.

(5) 3d\_Image\_download : 업로드 되어있는 stl파일을 다운로드 할 수 있게 해준다. 파일을 다운받고 캐드와 연결하여 출력, 도면 수정이 가능하다.

이 페이지들을 기반으로 전체적인 웹사이트를 제작하였다.

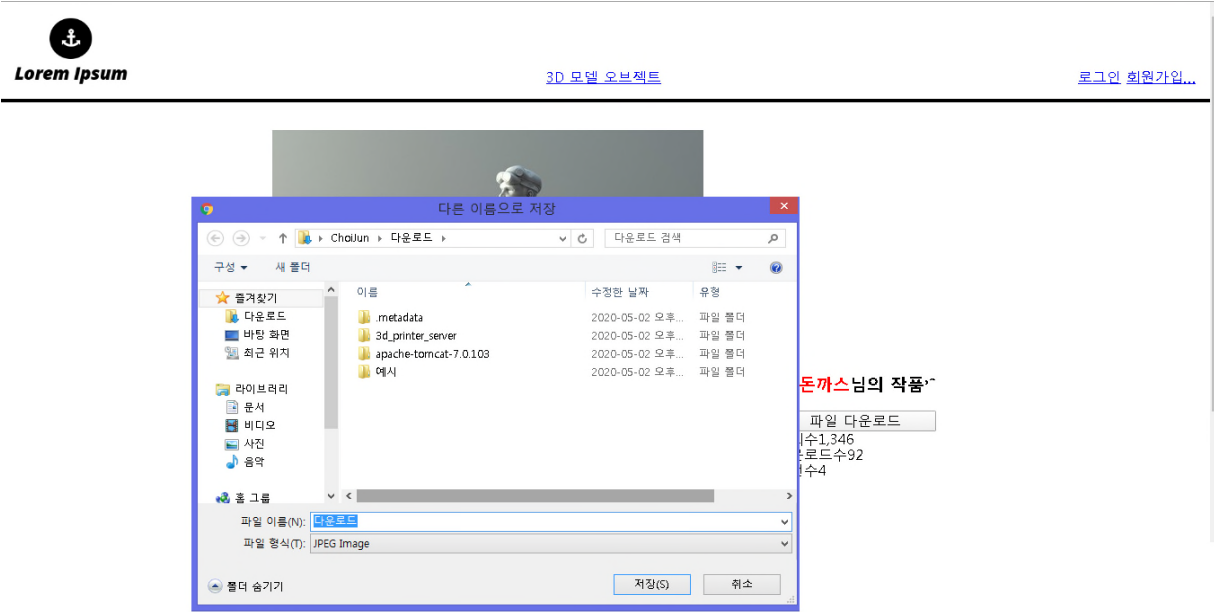
## 3) Server

이 웹페이지와 연동하는 서버로는 Apache Tomcat Web Server를 사용하였다.

cafe 24를 이용하여 Apache Tomcat 서버와 호스팅. 데이터베이스 구축은 MYSQL을 사용하여 호스팅 및 테이블 구성하였다.

5. Prototype 구현

HTML 홈페이지에서의 파일 업로드/다운로드 에 대한 결과 창.



내부 소스 스키마 모델 작성의 예시

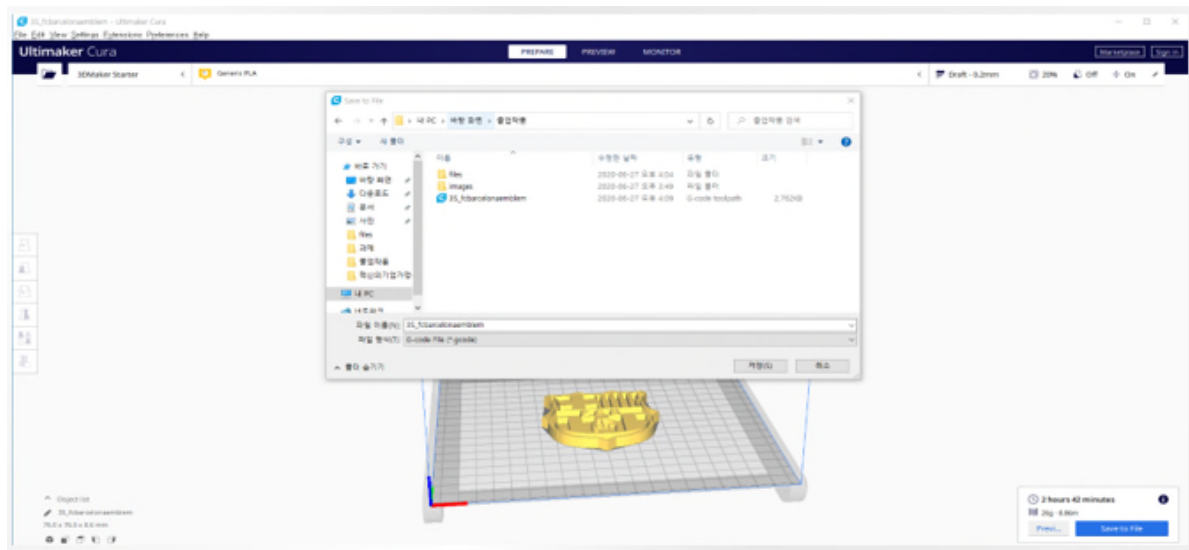
3d_files	2020-05-02 오후 7:41	파일 폴더	
3d_images	2020-05-02 오후 9:21	파일 폴더	
css	2020-05-02 오후 9:21	파일 폴더	
fonts	2020-04-27 오전 6:27	파일 폴더	
img	2020-05-02 오후 9:21	파일 폴더	
js	2020-04-27 오전 10:07	파일 폴더	
META-INF	2020-05-02 오후 9:21	파일 폴더	
WEB-INF	2020-05-02 오후 9:21	파일 폴더	
3d_image_download.jsp	2020-05-02 오후 8:03	JSP 파일	2KB
loginJSP.jsp	2020-05-02 오후 7:56	JSP 파일	2KB
loginPage.jsp	2020-05-02 오후 8:04	JSP 파일	2KB
modelListPage.jsp	2020-05-02 오후 6:16	JSP 파일	8KB
modelPage.jsp	2020-05-02 오후 8:24	JSP 파일	2KB
userPage.jsp	2020-05-02 오후 7:02	JSP 파일	2KB

## 6. 시험 / 테스트 결과

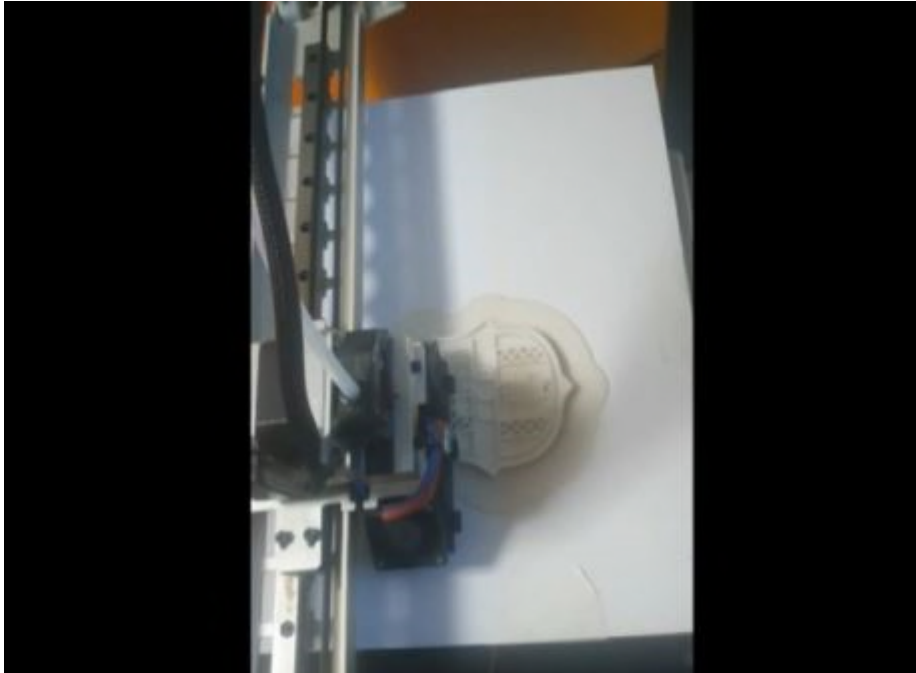
### 1) 파일 다운로드



### 2) 3d프로그램에서 stl to gcode 실행



### 3) 출력



## 7. Coding & Demo

### - 3D\_image\_download.jsp 에 대한 주석

```
<%@ page contentType="application/download;charset=KSC5601" %>
```

```
<%@ page import="java.io.*" %>
```

```
<%
```

```
//파일 이름을 전달 받는다.
```

```
String fileName = request.getParameter("fileName");
```

```
//다운로드창 설정
```

```
response.setContentType("application/download");
```

```
//아래처럼 attachment 를 사용하면 브라우저는 무조건 다운로드 창을 띄우고 파일명을 보여준다.
```

```
response.setHeader("Content-Disposition", "attachment;filename="+ fileName+ "");
```

```
//파일을 outputStream으로 받는다.
```

```
//outputstream은 파일을 바이트 단위로 쪼개서 보내는 방식이다.
```

```
ServletOutputStream sos = null;
```

```
try{
```

```
    sos = response.getOutputStream();
```

```
}catch(Exception e){e.printStackTrace();}
```

```
FileInputStream fio = null;
```

```
byte[] buf = new byte[1024];
```

```
//서버에 3d_files 폴더에서 전달 받은 파일을 지정한다.
```

```
//바이트 단위로 전송을 준비하는 과정이다.
```

```
fio = new FileInputStream("C:/Tomcat5.5/webapps/ROOT/webContent/3d_files"+ fileName);
```

```
int n = 0;
```

```
//반복문을 사용하여 서버에서 오는 바이트가 안올때까지 계속해서 바이트 데이터를 받는다.
```

```
while((n=fio.read(buf, 0, buf.length))!=-1) {
```

```
    sos.write(buf, 0, n);
```

```
    sos.flush();
```

```
}
```

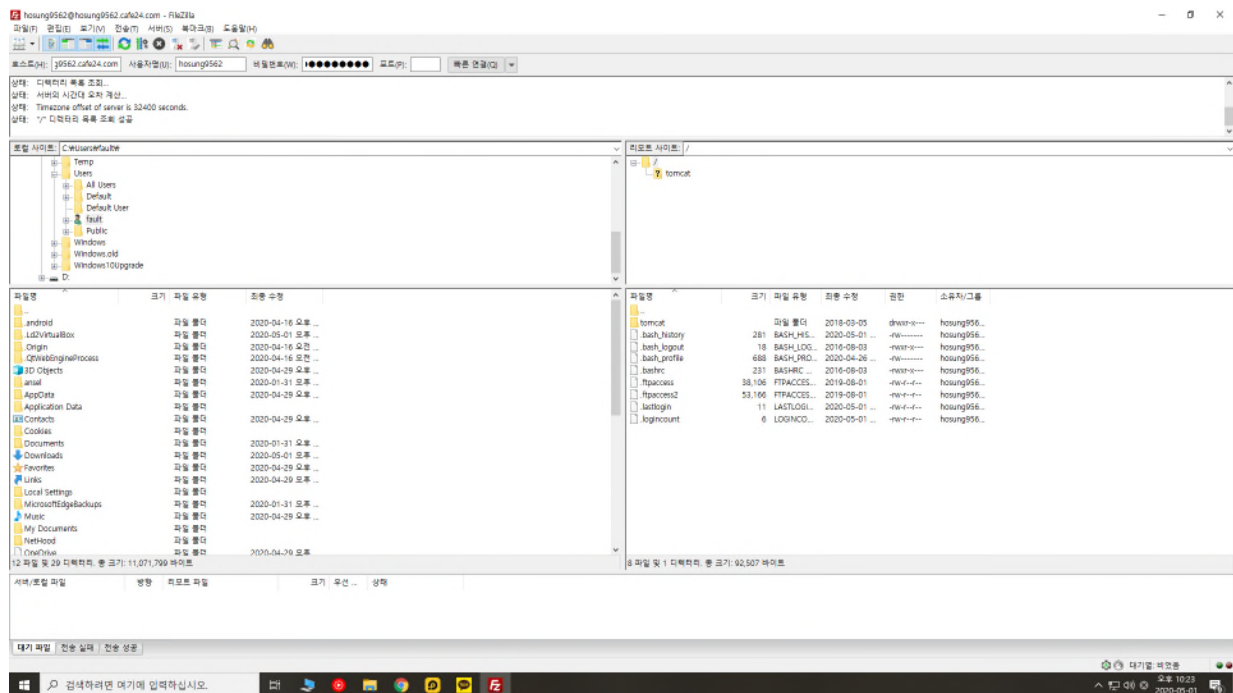
```
sos.close();
```

```
%>
```

## - 서버 호스팅에 대한 연동

FileZila의 상태 창

현재 Cafe24의 호스팅 서버에 연결 성공.

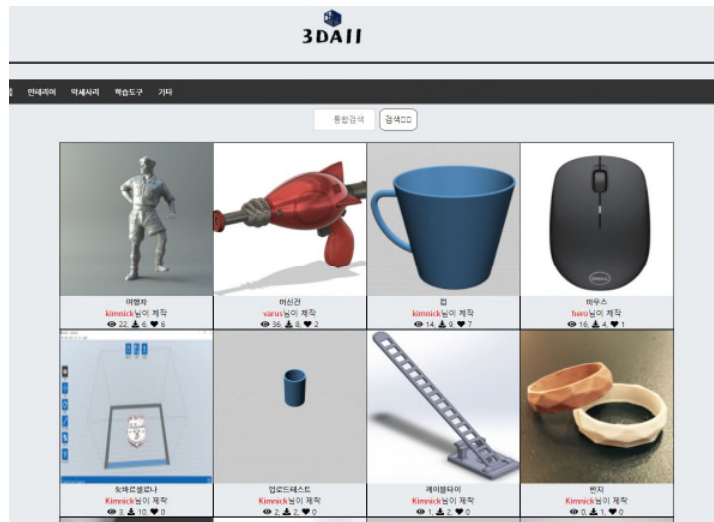


Tomcat 내부에 데이터베이스와 연동하여 작업을 진행하였다.

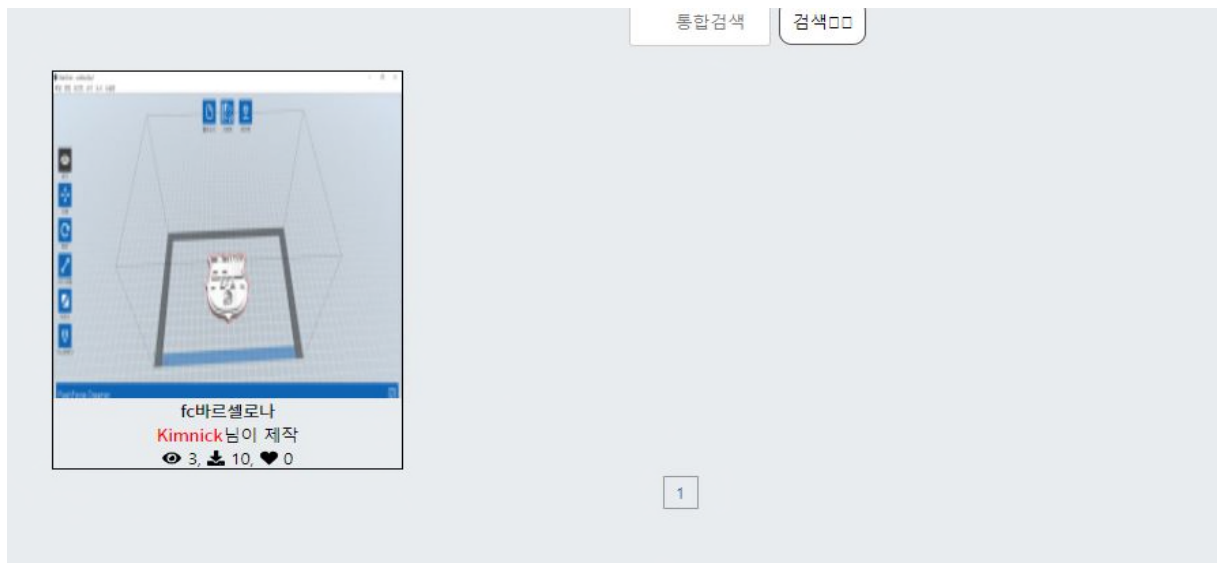
- 실행 화면



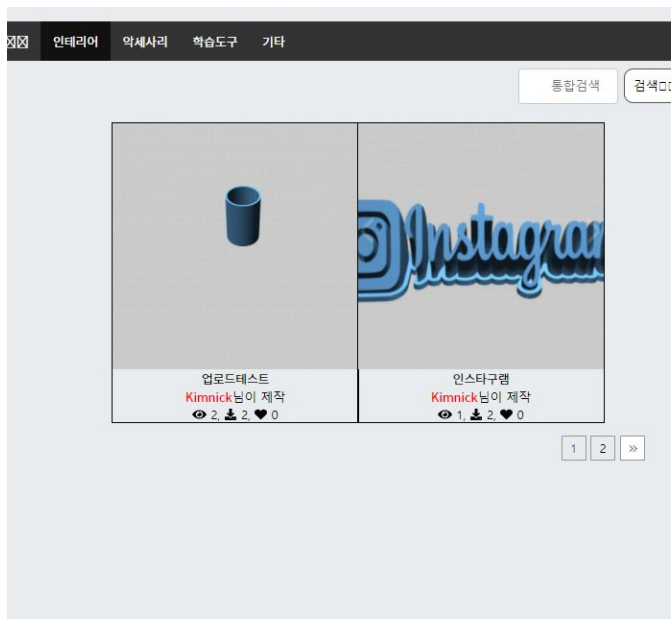
메인 화면의 모습이다.



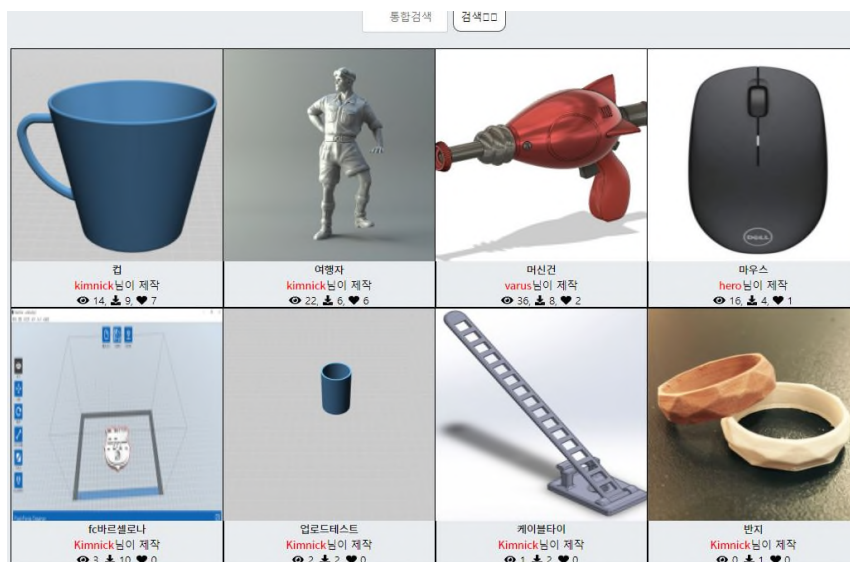
등록된 3D 모델들을 나열한 페이지이다.



키워드(fc) 검색 후 나타난 페이지이다.



카테고리가 인테리어인 게시글을 따로 추출한 모습이다.



하트 개수가 많은 순서대로 게시글을 나열한 모습이다.



### III. 결론

#### 1. 연구 결과

##### 1) 프린터 연동에 대한 실패 사안

(1) 3D 모델마다 출력하는 드라이버가 다르다.

여러 가지 브랜드의 3D 출력 프린터의 경우 3. 3D 프린터의 기본 원리 및 방식에 다른 종류들이 아래와 같다.

Binder Jetting (접착제 분사식)

3DP(3 Dimension Printing)

Directed Energy Deposition (에너지 집중식 퇴적 방식)

Material Extrusion (재료 압출식)

FDM(Fused Deposition Modeling)

Material Jetting (재료 분사식)

Polyjet

MJM(Multi Jet Modeling)

Powder Bed Fusion

SLS(Selective Laser Sintering)

Sheet Lamination (표면 접착식)

LOM(Laminated Object Manufacturing)

Vat Photopolymerization (광수지화 방식)

DLP(Digital Light Processing)

SLA(Stereo Lithography Apparatus)

AOM(Anti-gravity Object Modeling)

위의 경우처럼 다양한 종류들의 프린터 모델들이 존재하는데, 웹 사이트 백-엔드 구현화 하려면, 모든 종류의 API 규격을 가져와야 한다, 또한 브랜드 별로 원리 방식의 대한 드라이버도 각기 다르게 존재하기 때문에, 단 기간 6개월의 프로젝트로 구현하기 힘들뿐더러, 각 브랜드 회사에 의뢰하여 소스 코딩 및 드라이버 내부에 있는 백-엔드 구현화 된 소스들을 가져와야 하기 때문이다. 각 기업 별로 간추려서 해외, 국내에 여러 번 질문 했으나, 기업비밀이기 때문에 힘들다, 또는 회신이 없기 때문

(2) 3d 프린터의 경우 소프트웨어가 따로 존재하기 때문에 도면을 다운받아서 바로 출력하는게 더 간편화 및 유동성 때문에 연동 시킬 필요는 없다. 라는게 현 기업들의 생각, 그 이유 때문에 자회사들도 굳이 웹페이지를 통한 백-엔드 구현화를 미룸, 단 국내에 도면 사이트가 없기 때문에, 또는 간략화 된 것 밖에 없기 때문에, 본인들의 프로젝트는 굉장히 사업적으로 유망하다고 전달

(3) 바로 출력할 필요가 없는데 이것은 편의성에서도 관련이 없다, 그냥 웹에서 hwp 파일 실행해서 인쇄하는 것과 같은 이론, 단 모델한정 소프트웨어에서 이지만,

(4) API독자적 개발이 불가능함, 앞서 이야기 했던 것 처럼 3d 프린터 모델의 드라이버를 뜯어내야 함, 그것은 졸작이라 하더라도 해선 안되는 경우, 독자적인 드라이버 및 3d 프린터기를 개발하지 않는 이상 보안법에 위반, 아래에 참조 자료

미국 연방순회항소법원("CAFC")은 지난 2018. 3. 27. Oracle America, Inc.("Oracle")가 Google LLC("Google")를 상대로 제기한 Java API 패키지 저작권침해 소송에서 Google의 공정이용(fair use) 항변을 배척하고 Google에게 저작권침해에 따른 손해배상책임을 인정했습니다. 위 사건에서는 소프트웨어 산업에 지대한 영향을 미칠 수 있는 독특한 법적 쟁점이 다루어졌고, 2010년에 첫 소송이 제기된 뒤로 거의 매 십급마다 판단이 엇갈렸을 정도로 치열한 법리적 다툼이 이어졌기 때문에, 과연CAFC가 어떠한 판단을 내릴 것인지가 소프트웨어 업계와 학계의 큰 관심사였습니다. 이하에서는 이 사건의 사실관계와 주요 쟁점에 관한 CAFC의 판단 및 그 근거에 대하여 살펴보겠습니다.

Java는 Sun Microsystems, Inc.("Sun")가 1990년대에 개발한 컴퓨터 프로그래밍 언어입니다. Sun은 프로그래머들이 Java 프로그래밍 언어를 사용하여 손쉽게 애플리케이션 프로그램을 개발할 수 있도록 입출력, 수학연산, 보안, 텍스트 처리 등의 기능을 미리 구현하여 이를 Java 2 Standard Edition("Java SE")이라는 API(Application Programming Interface)의 형태로 제공했습니다. Java SE는 2008년을 기준으로 약 3,000개의 클래스(class)와 30,000개의 메소드(method)를 포함하는 166개의 API 패키지로 구성되었습니다.

그 무렵 Google은 모바일 기기용 소프트웨어 플랫폼인 Android를 출시하였습니다. Google은 Java 프로그래밍 언어와 Sun이 제공해온 개발환경에 익숙한 개발자들이 Android용 애플리케이션 프로그램을 쉽게 개발할 수 있도록 Sun으로부터 Java API에 관한 라이선스를 획득하고자 하였습니다. 그러나 Sun과의 협상에 실패했고, Google은 Android API를 직접 개발하여 배포하였습니다. Android API는 각 클래스와 메소드를 Sun의 Java API를 그대로 사용하지 않고 독자적으로 개발하였으므로 소스코드상의 공통점은 거의 찾을 수 없었습니다. 그러나 Google은 Android API에 37개의 Java API 패키지의 각 클래스와 메소드의 형식을 차용했고, 그 결과 Android API에는 Sun의 Java API의 해당 37개 패키지와 그 안에 포함된 클래스 및 메소드들의 선언코드<sup>1)</sup> 약 11,500줄이 동일하게 포함되었습니다.

[각주1] '선언코드(declaring code)'란 객체를 정의하는 단위인 클래스의 이름과 타입, 기능 수행 단위인 메소드의 이름과 그 입력 변수의 이름 및 타입을 정의한 코드를 의미합니다.

이에 2010년에 Sun을 인수한 Oracle은 같은 해 캘리포니아 남부지방법원(US District Court for Northern District of California; NDCA)에 Java API 패키지의 저작권을 침해하였다는 이유로 Google을 제소하였습니다.<sup>2)</sup>

[각주2] Oracle은 당초 특허권의 침해도 주장했으나, 이후 저작권 침해 여부만 쟁점이 되었으므로 특허권 침해에 관한 사항은 다루지 않습니다.

이렇듯, 임의 API를 가져와 백-엔드 구현화를 하게 되면 졸작이라고 하더라도 보안법 상 위촉되기

에  
사실상 독자적 3D프린터 개발 및 드라이버 구현을 하지 않는 이상 불가능

(5) 드라이버를 임의를 뜯어낸다 하더라도, 회사마다 소스 제공이 다르기 때문에 사실상 백엔드 연동이나 과정이 매우 힘들어지기에 학부생 수준에서는 불가능으로 인한 것 때문에, 단순, 홈페이지에서 검색 엔진에 특화를 주어, 직접 파일을 다운로드 후 프린팅 하는 것을 과정으로 보여주는 것

으로 결론.

#### - 고유 DB 내의 검색 엔진 시도에 있을 때

- 인덱스(Index) | 데이터베이스(Database)
- 샤드(Shard) | 파티션(Partition)
- 타입(Type) | 테이블(Table)
- 문서(Document) | 행(Row)
- 필드(Field) | 열(Column)
- 매핑(Mapping) | 스키마(Schema)
- Query DSL | SQL

참고 1) 6.0이하 버전의 엘라스틱서치에서는 하나의 인덱스 내부 기능에 따라 데이터 분류 후에 여러 개의 타입을 만들어 사용했지만 현재는 하나의 인덱스에 하나의 타입만을 구성해야 한다.

참고 2) 매핑은 필드의 구조와 제약조건에 대한 명세를 말하며 관계형 DB의 스키마와 같다

참고 3) 관계형 DB와 엘라스틱서치는 인덱스라는 개념을 다르게 사용하는데, 관계형 DB에서 인덱스는 그저 Where절의 쿼리와 Join을 빠르게 만들어주는 보조데이터의 도구로 사용된다.

인덱스 내부에 있는 데이터베이스 필드를 바탕으로 검색을 시도하였다, 번외로

STL G-CODE 내부에 있는 파일을 내부 해석 했을 때 고유 시리얼 넘버링을 찾아 검색 엔진에 도입하려 했으나 고유 시리얼 넘버링에 대한 복잡 관계도가 성립이 되질 않아 결국 각기 따로 노는 파일이 됨으로써, 편의성 제공을 위해 카테고리 별 타입, 테이블에 검색을 추가하여 검색 예정

조회수, 리뷰수, 좋아요, 싫어요에 따른 차별화를 두어 한눈에 직접 볼 수 있게 대중성을 두어 구현 예정이다.

홈페이지 구동 및 3D 프린터에서의 출력 결과물은 성공적이나, 애초에 계획했던 직접적인 연동 서비스 구현에 있어 백 엔드- 구현이 필요한 상황이었으나,

생각했던 것과 다르게 필요한

#### 1) 웹 생태계의 스펙

HTML, HTTP(1.1 , HTTP/2)

#### 2) 기본 SDK, 라이브러리/프레임워크 이해와 활용

#### 3) 클라이언트를 위한 API 설계

#### 4)서버/컴퍼넌트/객체 간의 역할 분담/의존성/통신 방법 설계

#### 저장소 활용

DBMS 설계

Cache 적용

*Global/Local cache 적용범위, 라이프 싸이클, 솔루션 선택*

파일 저장 정책/솔루션 선택 활용

#### 5) 검색엔진 연동 방식 결정

#### 6) 빌드 도구

Maven/Gradle

#### 7) 배포 전략

#### 8) 성능 테스트/프로파일링/튜닝

JVM 레벨의 튜닝 (GC 옵션 등)

*웹 서버(Nginx, Tomcat)등의 설정/튜닝*

OS 설정의 주요 값 확인

#### 9) 인접 기술에 대한 이해

DBMS, Front End 등

#### 10) 서버 개발자에만 해당하지는 않는 항목

테스트 코드 작성/리팩토링 기법

버전 관리 전략

*branch 정책 등*

이 필요로 하다.

직접적인 연동을 위해 다양하게 접근을 시도 했으나 시간과, 3D 프린터 보안법에 의해 자체 시스템을 고칠 수가 없었기에 시행하지 못했다.

그 외에 홈페이지 구동 방식이나, 3D 프린터에 대한 전반적인 지식과 프리웨어 사이트에 대한 구조 및 서칭 시스템에 대해 학습을 할 수 있었으며

3D 프린터의 시장의 경우 아직까지 많이 미흡하며, 보급화가 쉬워졌다고는 하나, 아직까지는 활발하지 않은 상태라고 판단하였다.

## 2. 작품제작 소요재료 목록

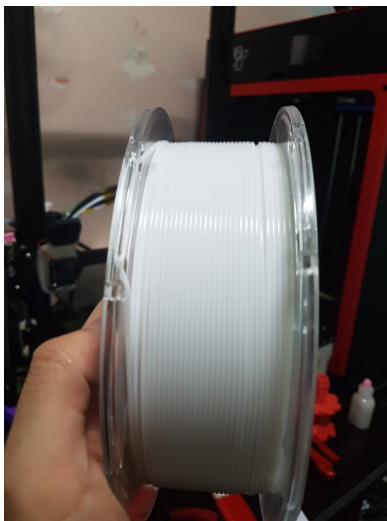
### 1) 3D프린터

- 파인봇 FB-9600



### 2) 3D물체 출력위한 재료

- 화이트 필라멘트(PLA 필라멘트)



- 참고자료 -

- 1) STL - <https://ko.3dsystems.com/quickparts/learning-center/what-is-stl-file>
- 2) G-CODE - <https://ko.wikipedia.org/wiki/G-code>
- 3) Thingiverse - <https://www.thingiverse.com/>
- 4) 3dch - <http://3dch.co.kr/>
- 5) Cura  
<https://m.blog.naver.com/PostView.nhn?blogId=ivycus&logNo=220629784338&proxyReferer=https://www.google.com/>
- 6) Lucene - <https://brownbears.tistory.com/6>