

# 클라우드 시스템 및 인증체제 구현

2019. 10. 29

지도 교수 : 양환석 교수님

**클라우드 캐슬**  
(강지호, 최진석, 양유리, 오수빈)

# 목 차

---

- 01      **조원 편성**
- 02      **주제 선정**
- 03      **구 상 도**
- 04      **추진 경과**
- 05      **개발 환경 및 개발내용**
- 06      **개발 시스템 운영**
- 07      **결론 및 기대효과**

# 조원 편성

성명	담당
강지호(조장)	하둡 서버 구축 및 웹 연동, 백엔드 개발, DB설계
최진석	인증서버 구축 및 웹 연동, 보고서 작성
양유리	인증서버 구축 및 웹 연동, 보고서 작성
오수빈	인증서버 구축 및 웹 연동, 보고서 작성
공통	프론트엔드 개발, 웹 서버 구축, ppt작성

# 주제 선정

## □ 클라우드는 필수 인프라, ICT 업계 경쟁 치열

"클라우드"는 기업혁신 필수 인프라, 데이터 활용 방식이 세계 바꿀 것"

마이란 톰슨 부코백 AWS 부사장

"클라우드"로 민첩성 안키우면  
기업규모 상관없이 모두 도태돼"

오대석 기자 | 입력 : 2019.04.01 04:01:03



"데이터를 활용하는 새로운  
니다. 데이터가 더 많이 클  
은 물론이고 전혀 예상치 못  
로 혁신이 일어날 수 있습니  
택이 아니라 기업 생존을 위  
가 됐습니다."

지난달 13일 마이란 톰슨 부  
(AWS) S3 총괄 부사장은 매  
클라우드가 향후 5~10년 동  
심 근간이 될 것이라고 강조  
그가 총괄하고 있는 S3 서

### 클라우드 시장 잡아라 · ICT업계 대격돌

블루오션 '클라우드' 시장 향한 기업 진출 ↑  
보안·높은 서비스 품질 강조한 플랫폼 론칭  
클라우드 리딩 기업 두고 ICT사 경쟁 치열



장가람 기자 jay@

최근 기업들의 클라우드 전환이 증가하며, 리딩 기업을 향한 업계 경쟁도 치열해지고 있다. 각 사별  
로 보안과 높은 서비스 품질 등의 장점을 내세우며 시장 선점에 한창이다.

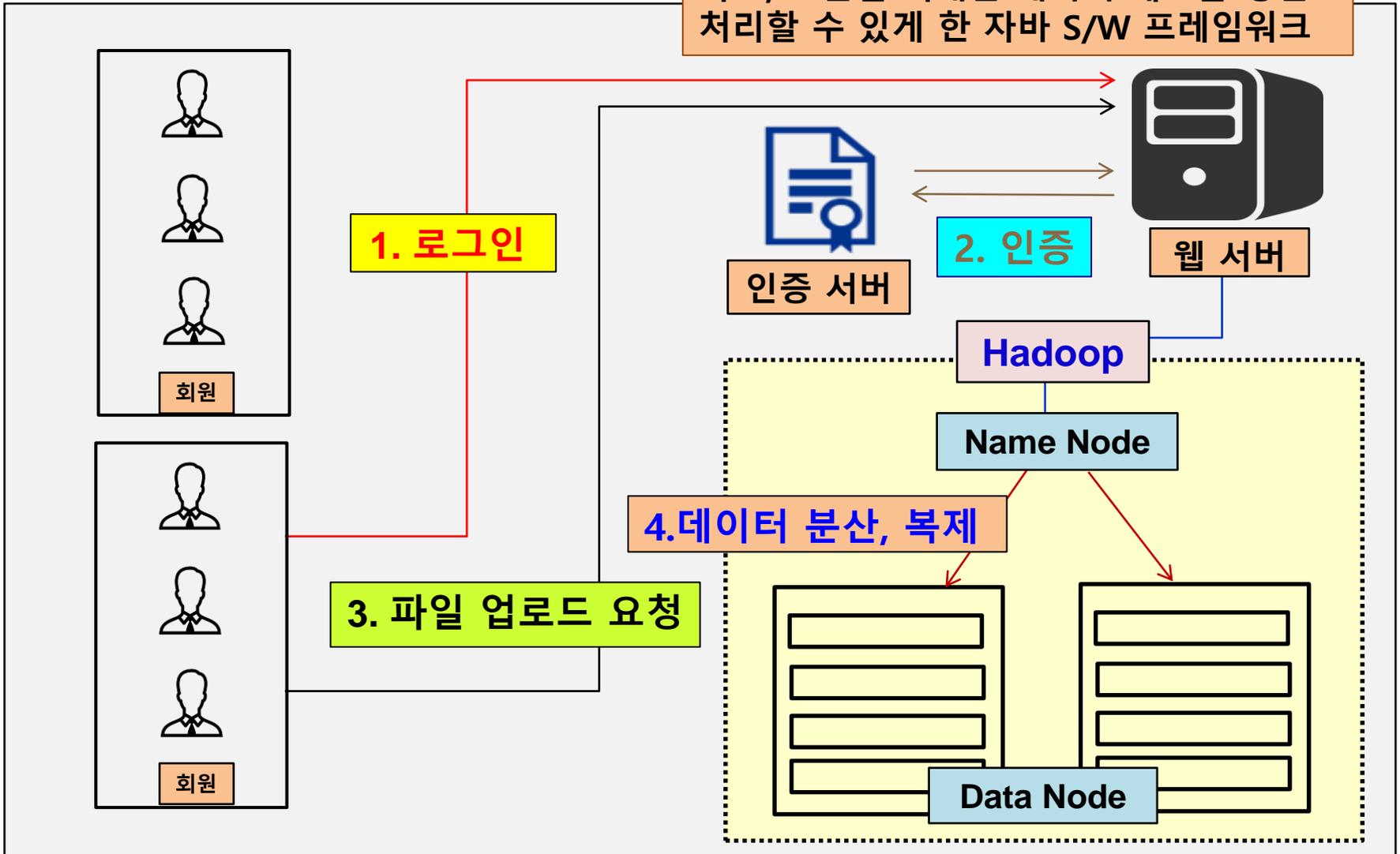
클라우드란 데이터를 인터넷과 연결된 중앙컴퓨터에 저장, 언제 어디서든 인터넷에 접속해 관련 데  
이터를 이용할 수 있는 서비스를 말한다. 기업이 클라우드로 전환할 경우 IT 인프라 구축 비용 및 운  
영비용 절감 등과 IT신기술을 R&D(연구개발), 생산, 마케팅 등 경영 전반에 빠르게 적용할 수 있는  
장점이 있다.

**보안관리체제가 완비된 클라우드 시스템 구축방안 연구**

# 구상도(1/3)

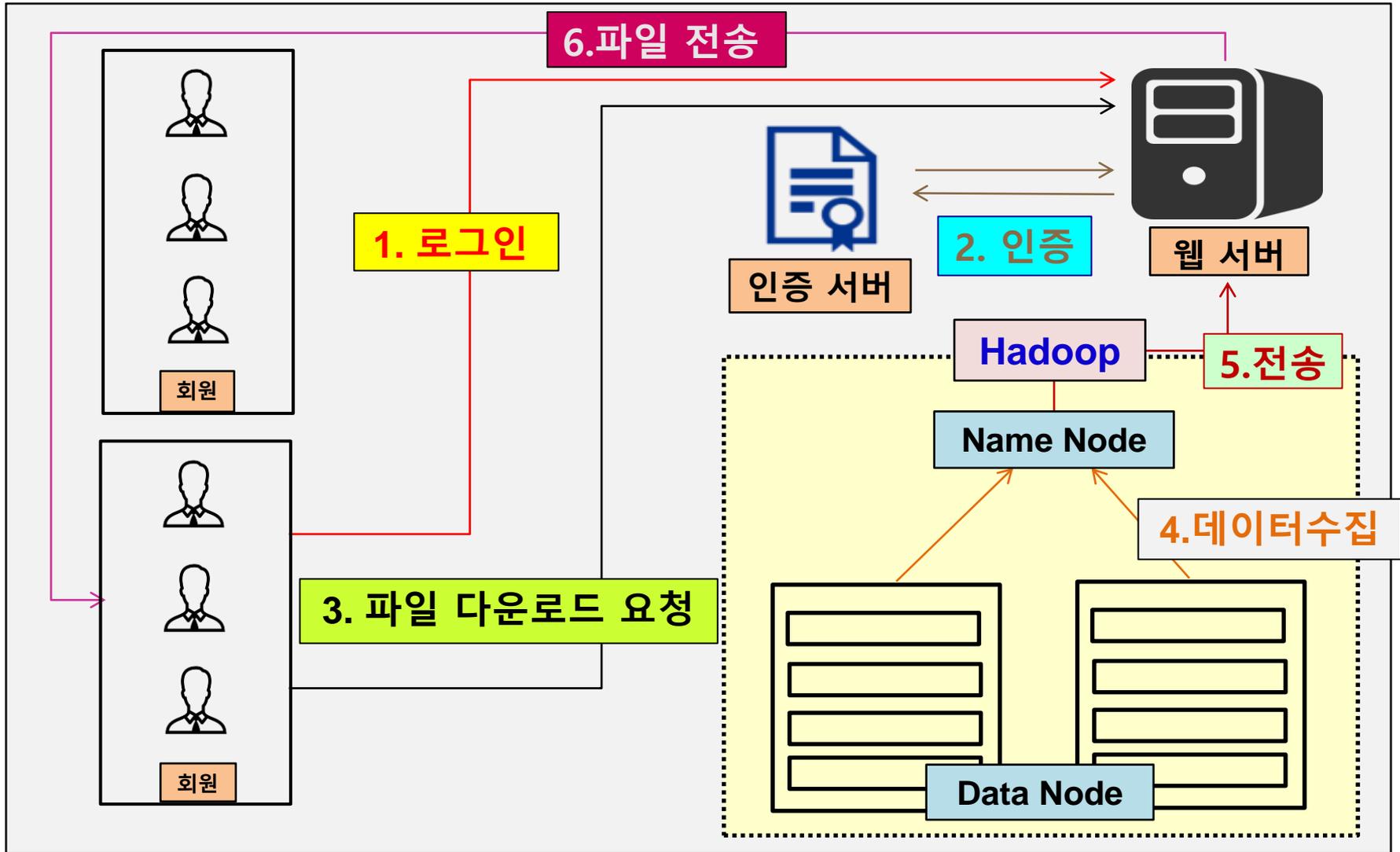
## 로그인과 업로드 시 구상도

**Hadoop** : 가상화된 대형 스토리지를 형성하고, 보관된 거대한 데이터 세트를 병렬 처리할 수 있게 한 자바 S/W 프레임워크



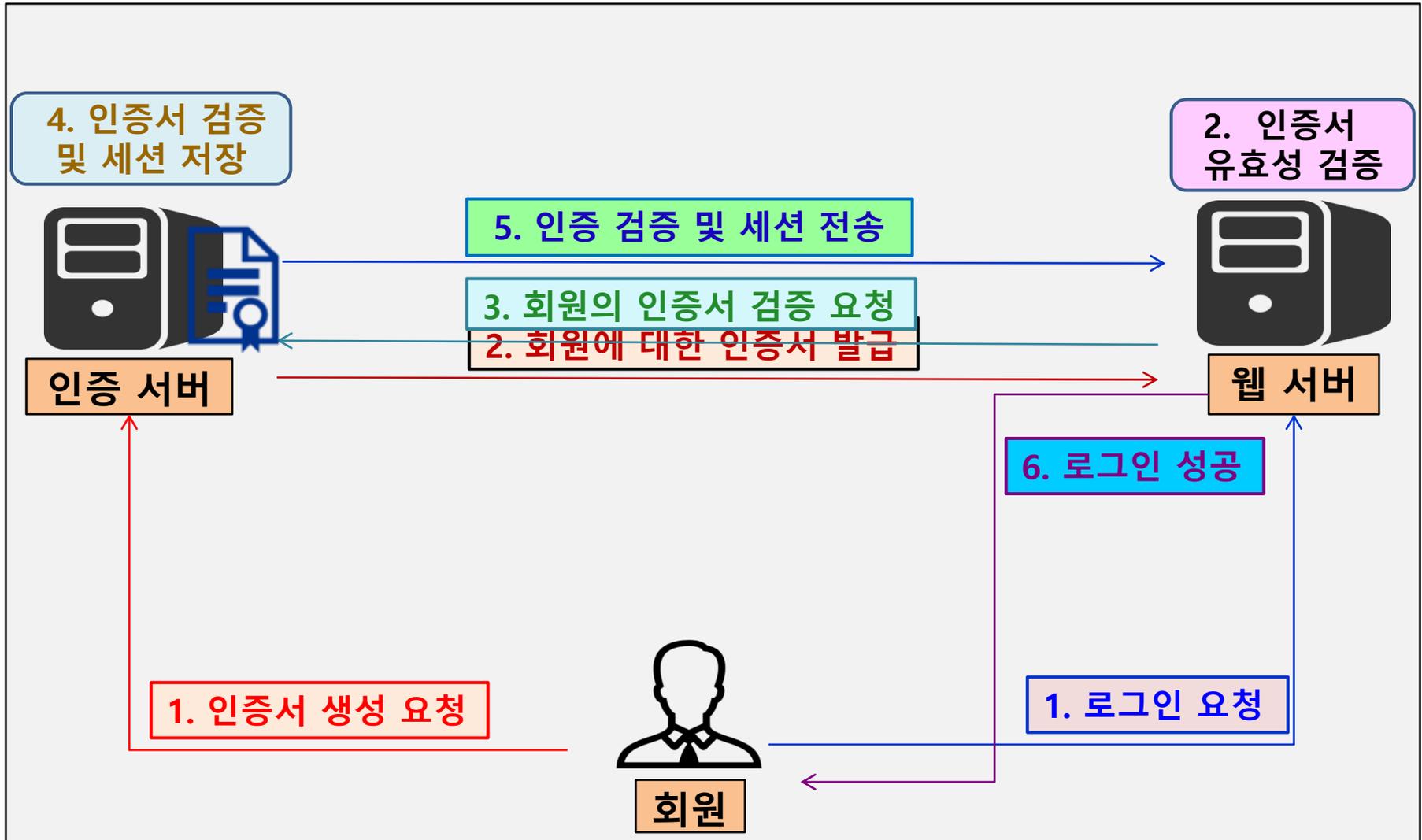
# 구상도(2/3)

## 로그인과 다운로드 시 구상도



# 구상도(3/3)

## □ 인증절차 구상도



# 추진 경과

추진 기간 (2019년)	추진 기간								
	3월	4월	5월	6월	7월	8월	9월	10월	
추진 업무									
자료조사/분석	■								
구상도 설계		■							
웹 서버 구축 및 웹 백엔드 개발		■							
하둡 서버 구축 및 웹 연동				■					
인증 서버 구축 및 웹 연동						■			
웹 프론트엔드 개발 및 최종 확인								■	

# 개발 환경 및 개발 내용(1/8)

## 개발 환경



서버운영체제

CentOS 7(Linux)  
CentOS 6(Linux)



개발 언어

JSP  
HTML/CSS  
JAVA



D B

Oracle Database



인증 시스템

Kerberos

# 개발 환경 및 개발 내용(2/8)

## 회원 관리

```
public void addMember(MemberVO m) {
    try {
        conn = dataFactory.getConnection();
        String sql = "insert into t_member2 (id, pwd, name, email) values (?, ?, ?, ?)";
        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setString(1, m.getId());
        pstmt.setString(2, m.getPwd());
        pstmt.setString(3, m.getName());
        pstmt.setString(4, m.getEmail());
        pstmt.executeUpdate();
        conn.close();
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
}
```

회원 추가

```
public void delMember(String id) {
    try {
        conn = dataFactory.getConnection();
        String sql = "delete from t_member2 where id=?";
        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setString(1, id);
        pstmt.executeUpdate();
        conn.close();
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
}
```

회원 삭제

```
public void modMember(MemberVO memberVO) {
    String id = memberVO.getId();
    String pwd = memberVO.getPwd();
    String name = memberVO.getName();
    String email = memberVO.getEmail();
    try {
        conn = dataFactory.getConnection();
        String query = "update t_member2 set pwd=?,name=?,email=? where id=?";
        System.out.println(query);
        PreparedStatement pstmt = conn.prepareStatement(query);
        pstmt.setString(1, pwd);
        pstmt.setString(2, name);
        pstmt.setString(3, email);
        pstmt.setString(4, id);
        pstmt.executeUpdate();
        pstmt.close();
        conn.close();
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
}
```

회원정보 수정

회원 추가, 삭제, 정보수정 등 관리자의 회원 관리 지원

# 개발 환경 및 개발 내용(3/8)

## 프로젝트 관리

```
public void addtitleMem(int articleNO, List<MemberVO> mem
    try {
        conn = dataFactory.getConnection();
```

프로젝트 추가

```
public void deleteArticle(int articleNO) {
    try {
        conn = dataFactory.getConnection();
```

프로젝트 삭제

```
public void updateArticle(ArticleVO article) {
    int articleNO = article.getArticleNO();
    String title = article.getTitle();
    String content = article.getContent();
    String FileName = article.getFileName();
```

프로젝트 수정

```
    try {
        conn=dataFactory.getConnection();
        String query = "update t_board2 set title=?,content=?";
        if(FileName != null && FileName.length() != 0) {
            query += ",FileName=?";
        }
        query += " where articleNO=?";

        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, title);
        pstmt.setString(2, content);
        if(FileName != null && FileName.length() != 0) {
            pstmt.setString(3, FileName);
            pstmt.setInt(4, articleNO);
```

프로젝트 추가, 삭제, 정보수정 등 관리자의 프로젝트 관리 지원

# 개발 환경 및 개발 내용(4/8)

## 회원 권한 관리

일반회원 정보 획득

```
public List<ArticleVO> listArticles(String id){
    List<ArticleVO> articlesList = boardDAO.selectAllArticles(),
    List<ArticleVO> memList =
    List<ArticleVO> resultList;
    ArticleVO memVO = null;
    int articleNO,parentNO;
    for(int i=0;i<articlesList
        ArticleVO articleVO =
        articleVO=articlesList
        articleNO=articleVO.ge
        parentNO=articleVO.ge
        for(int j=0;j<memList
            memVO=memList.get
            if((articleNO==mem
                resultList.add
            }
        }
    }
    return resultList;
}
```

관리자 정보 획득

```
public List listMembers() {
    List<MemberVO> membersList = new ArrayList();

    try {
        conn = dataFactory.getConnection();
        String query = "select * from t_member2 order by joinDate desc";
        System.out.println(query);
        pstmt=conn.prepareStatement(query);
        ResultSet rs = pstmt.executeQuery();
        while(rs.next()) {
            MemberVO vo = new MemberVO();
            String id = rs.getString("id");
            String pwd = rs.getString("pwd");
            String name = rs.getString("name");
            String email = rs.getString("email");
            Date joinDate = rs.getDate("joinDate");
            MemberVO memberVO = new MemberVO(id,pwd,name,email,joinDate);
            membersList.add(memberVO);
        }
    }
}
```

회원 및 관리자 로그인 시 정보를 획득, 권한 관리에 활용

## 프로젝트 내의 글 관리

```
public int insertNewArticle(ArticleVO article) {
    int articleNO = getNewArticleNo();
    try {
        conn = dataFactory.getConnection();
        int parentNO = article.getParentNO();
        String title = article.getTitle();
        String content = article.getContent();
        String id = article.getId();
        System.out.println("filename =====");
        String fileName = article.getFileName();
        String query = "INSERT INTO t_board2 (parentNO, title, content, fileName, id)
            + " values (?, ?, ?, ?, ?)";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setInt(1, articleNO);
        pstmt.setInt(2, parentNO);
        pstmt.setString(3, title);
        pstmt.setString(4, content);
        pstmt.setString(5, fileName);
        pstmt.setString(6, id);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

파일/글 추가

```
public void deleteArticle(int articleNO)
    try {
        conn = dataFactory.getConnection();
        String query = "delete from t_board2 where articleNO=?";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setInt(1, articleNO);
        pstmt.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

파일/글 삭제

```
public void updateArticle(ArticleVO article)
    int articleNO = article.getArticleNO();
    String title = article.getTitle();
    String content = article.getContent();
    String fileName = article.getFileName();

    try {
        conn=dataFactory.getConnection();
        String query = "update t_board2 set title=?,content=?";
        if(fileName != null && fileName.length()!=0) {
            query += ",fileName=?";
        }
        query += " where articleNO=?";

        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, title);
        pstmt.setString(2, content);
        if(fileName != null && fileName.length()!=0) {
            pstmt.setString(3, fileName);
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

파일/글 수정

관리자와 사용자가 프로젝트 내의 파일/글 관리

# 개발 환경 및 개발 내용(6/8)

## KDC(인증서버) 설정(1/2)

※ DNS와 NTP는 기본적으로  
설정이 되어야 함

krb5.conf 파일 수정

kadm5.acl 파일 수정

KDC DB master key 등록

```
[root@kerberos ~]# cat /etc/krb5.conf
[logging]
default = FILE:/var/log/krb5libs.log
kdc = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log

[libdefaults]
dns_lookup_realm = false
ticket_lifetime = 8h
renew_lifetime = 7d
forwardable = true
rdns = false
default_realm = JBHARD.COM
default_ccache_name = CCACHE_NAM

[realms]
JBHARD.COM = {
  kdc = jbhard.kerberos.org
  admin_server = jbhard.kerberos.org
}

[domain_realm]
.kerberos.org = kerberos.org
.kerberos.com = kerberos.com
.kerberos.net = kerberos.net
.kerberos.edu = kerberos.edu
.kerberos.gov = kerberos.gov
.kerberos.mil = kerberos.mil
.kerberos.net.au = kerberos.net.au
.kerberos.net.br = kerberos.net.br
.kerberos.net.cn = kerberos.net.cn
.kerberos.net.nz = kerberos.net.nz
.kerberos.net.pe = kerberos.net.pe
.kerberos.net.pk = kerberos.net.pk
.kerberos.us = kerberos.us
.kerberos.uk = kerberos.uk
.kerberos.za = kerberos.za
.com = com
.com.au = com.au
.com.br = com.br
.com.cn = com.cn
.com.hk = com.hk
.com.mx = com.mx
.com.tw = com.tw
.gov = gov
.gov.au = gov.au
.gov.br = gov.br
.gov.cn = gov.cn
.gov.hk = gov.hk
.gov.in = gov.in
.gov.uk = gov.uk
.gov.za = gov.za
.mil = mil
.mil.au = mil.au
.mil.br = mil.br
.mil.cn = mil.cn
.mil.hk = mil.hk
.mil.in = mil.in
.mil.jp = mil.jp
.mil.kr = mil.kr
.mil.us = mil.us
.mil.uk = mil.uk
.mil.za = mil.za
.net = net
.net.au = net.au
.net.br = net.br
.net.cn = net.cn
.net.hk = net.hk
.net.in = net.in
.net.jp = net.jp
.net.kr = net.kr
.net.nz = net.nz
.net.pe = net.pe
.net.pk = net.pk
.net.us = net.us
.net.uk = net.uk
.net.za = net.za
.org = org
.org.au = org.au
.org.br = org.br
.org.cn = org.cn
.org.hk = org.hk
.org.in = org.in
.org.jp = org.jp
.org.uk = org.uk
.org.za = org.za
```

```
[root@kerberos ~]# kdb5_util create -s -r JBHARD.COM
Loading random data
Initializing database '/var/kerberos/krb5kdc/principal' for realm 'JBHARD.COM',
master key name 'K/M@JBHARD.COM'
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.
Enter KDC database master key:
Re-enter KDC database master key to verify:
```

도메인 구성 및 설정 파일 수정 후 KDC Database 설정

# 개발 환경 및 개발 내용(7/8)

## KDC(인증서버) 설정(2/2)

Root와 사용자 암호 등록

```
[root@kerberos ~]# kadmin.local
Authenticating as principal root/admin@JBHARD.COM with password.
kadmin.local: addprinc test
WARNING: no policy specified for test@JBHARD.COM; defaulting to no policy
```

사용자 keytab 파일 생성

```
kadmin.local: addprinc -randkey test.jbhard.com
WARNING: no policy specified for test.jbhard.com@JBHARD.COM; defaulting to no policy
Principal "test.jbhard.com@JBHARD.COM" created.
kadmin.local: ktadd test.jbhard.com
Entry for principal test.jbhard.com with kvno 2, encryption type aes256-cts-hmac-sha1-96 added to keytab FILE:/etc/krb5.keytab.
```

KDC 통신용 방화벽 구성

```
[root@kerberos ~]# firewall-cmd --permanent --add-service=krb5
Warning: ALREADY_ENABLED: krb5
success
[root@kerberos ~]# firewall-cmd --reload
success
[root@kerberos ~]# netstat -antup | grep krb
tcp        0      0 0.0.0.0:88          0.0.0.0:*          LISTEN    7547/krb5kdc
tcp6       0      0 :::88             :::*                LISTEN    7547/krb5kdc
udp        0      0 0.0.0.0:88          0.0.0.0:*          7547/krb5kdc
udp6       0      0 :::88             :::*                7547/krb5kdc
```

사용자 암호를 입력하고 복호화하는 keytab 파일 생성 및 저장

# 개발 환경 및 개발 내용(8/8)

## Client-Webserver 설정

※ DNS와 NTP는 기본적으로  
설정이 되어야 함

Gtk 설치 및 KDC 연결

인증 설정파일을 추가 작성

KDC에 접속해 ticket 받기

```
[root@www ~]# kinit test
Password for test@JBHARD.COM:
[root@www ~]# klist
Ticket cache: KEYRING:persistent:0:0
Default principal: test@JBHARD.COM

Valid starting    Expires          Service principal
2019-10-08T00:05:19  2019-10-08T00:06:19  krbtgt/JBHARD.COM@JBHARD.COM
```

gtk 패키지를 사용해 KDC와 연결한 후 설정 파일  
확인 또는 추가 후 KDC에서 ticket 받아 오기

# 개발 시스템 운영 (1/10)

## 메인 화면

The image displays a screenshot of the JBHard website with several callout boxes highlighting specific sections:

- 메인 화면** (Main Page): Points to the top navigation bar containing 'SERVICES' and 'TEAM'.
- 서비스 소개** (Service Introduction): Points to the 'SERVICES' section, which features a 'Data Store' service with a yellow elephant icon and the text '데이터 저장' (Data Storage).
- 팀원 소개** (Team Introduction): Points to the 'CLOUD CASTLE TEAM' section, which lists team members with their photos, names, roles, and social media links. The team members shown are:
  - Kang Jiho**: Hadoop Developer, Web server Developer
  - Choi Jinsuk**: Web Designer, Web server Developer
  - Yang Yuri**: Authentication Server Developer, Web server Developer
  - Oh Subin**: Authentication Server Developer, Web server Developer

A pink callout box at the bottom of the screenshot reads: **메인화면 및 서비스, 팀원소개** (Main page and services, team introduction).

# 개발 시스템 운영 (2/10)

## 인증 및 로그인 화면

로그인 전 인증서버를 통한 사용자 인증 화면

사용자 로그인화면

JBHard

Welcome

JBHARD는 커버로스 인증

아이디를 입력해주세요

비밀번호를 입력해주세요

Login

로그인 전 인증서버에 저장된 ID와 비밀번호를 검증 후 사용자 웹로그인

## 관리자 및 사용자 화면

The image displays two browser windows side-by-side, illustrating the system's user interfaces. The left window, titled '관리자 기본화면' (Admin Basic Screen), shows the admin interface at localhost:8080/Gproject22/member/listMembers.do. It features the 'JB-Hard' logo and a table with columns for '프로젝트명:1' and '작성자' (author), listing 'admin'. The right window, titled '사용자 기본화면' (User Basic Screen), shows the user interface at localhost:8080/Gproject22/board/listArticles.do. It features the 'JB-Hard' logo, a '로그아웃' (logout) link, and a large blue graphic with the text 'Welcome To JB-Hard! USER1'. Below this, it displays '프로젝트' (Project) and '현재사용할수있는 프로젝트입니다' (Current project you can use). A pink banner at the bottom of the image reads '로그인 후 관리자 및 사용자 화면 출현' (Appearance of admin and user screens after login).

## 회원추가

회원 추가

회원정보 입력

추가된 회원 목록

아이디	비밀번호	이름	이메일	가입일	수정	삭제
user2	user2	유민준	user2@joongbu.ac.kr	2019-10-13	수정	삭제
user1	user1	서지후	user1@joongbu.ac.kr	2019-06-17	수정	삭제
admin	admin	관리자	wkccstrl@sharklasers.com	2019-06-04	수정	삭제

회원 아이디, 비밀번호, 이름 및 이메일을 입력하고 회원 추가

## 회원삭제

회원 삭제

삭제 결과창

삭제 후 회원 목록

### 회원정보

회원정보를 수정할수있습니다

아이디	비밀번호	이름	이메일	가입일	수정	삭제
user1	user1	서지후	user1@joongbu.ac.kr	2019-06-17	수정	삭제
admin	admin	관리자	wkccstrl@sharklasers.com	2019-06-04	수정	삭제

회원 삭제 버튼 클릭 시 회원 삭제

# 개발 시스템 운영-관리자 (6/10)

## 회원수정

회원 수정

회원정보 수정 입력

수정 후 회원 목록

### 회원정보

회원정보를 수정할수있습니다

아이디	비밀번호	이름	이메일	가입일	수정	삭제
user1	user1	person	user1@joongbu.ac.kr	2019-06-17	수정	삭제
admin	admin	관리자	wkccstrl@sharklasers.com	2019-06-04	수정	삭제

회원 비밀번호, 이름 및 이메일 변경 가능

# 개발 시스템 운영-관리자 (7/10)

## 프로젝트 추가

프로젝트 추가

회원 지정 및 파일 저장

추가 후 프로젝트 목록

추가 후 하둡에 파일 저장

프로젝트 사용 회원 및 파일 저장 시 하둡에 저장

```
[hadoop@wikibooks01 바탕화면]$ hadoop dfs -ls /user/hive/warehouse/11
Warning: $HADOOP_HOME is deprecated.

Found 1 items
-rw-r--r--  3 hive supergroup      19621 2019-06-27 23:16 /user/hive/warehouse/11/joongbu.jpg
[hadoop@wikib
```

## 회원 로그인

The image shows a sequence of three browser windows illustrating user navigation. The first window displays a project list with a highlighted item. The second window shows the content of the selected project, featuring a large image placeholder and the Joongbu University logo. The third window shows another project content page with a similar layout. A red box highlights a project item in the list, and a pink box at the bottom summarizes the navigation capabilities.

회원 로그인 시 프로젝트 목록

프로젝트 내용1

프로젝트 내용 2

회원 로그인 시 참여 프로젝트 목록 및 프로젝트 내용 열람 가능

## 프로젝트 삭제

프로젝트 삭제

삭제 버튼 클릭

삭제 후 프로젝트 목록

프로젝트 삭제 시 하위 파일 및 글도 함께 삭제

## 프로젝트안 파일 및 글추가

파일/글 추가할 프로젝트

추가 버튼 클릭

추가할 파일/글 입력

프로젝트안 글 목록

작성자	제목	작성일
admin	2_1	2019-09-10

프로젝트명 : test

파일/글 추가후 하둡에 파일 저장

```
[hadoop@wikibooks01 바탕화면]$ hadoop dfs -ls /user/h
Warning: $HADOOP_HOME is deprecated.

Found 1 items
-rw-r--r-- 3
```

파일 및 글 추가시 하둡에 파일을 저장하여 관리

# 결론 및 기대효과



## ❖ 결 론

- 하둡 서버 및 웹 서버를 연동하여 사용자 그룹 파일 드라이브 웹 사이트를 개발/구축하는데 성공
- 로그인 과정에서 Kerberos 인증 방식을 사용하여 KDC에서 발급한 ticket으로 사용자를 검증하는 인증 체제로 시스템의 보안성을 확보

## ❖ 기대효과

- 모든 조원들이 업무를 적절히 분담하여 맡은 부분을 완성함에 따라 프로그래밍 능력 및 시스템 구축 능력을 향상시키는 계기
- 아울러 작품 완성을 통해 클라우드 시스템과 보안 관리의 중요성을 재인식

-끝-

**Q & A**

---

**Thank you**

---