

# PSU Knowledge Bank

June 2022

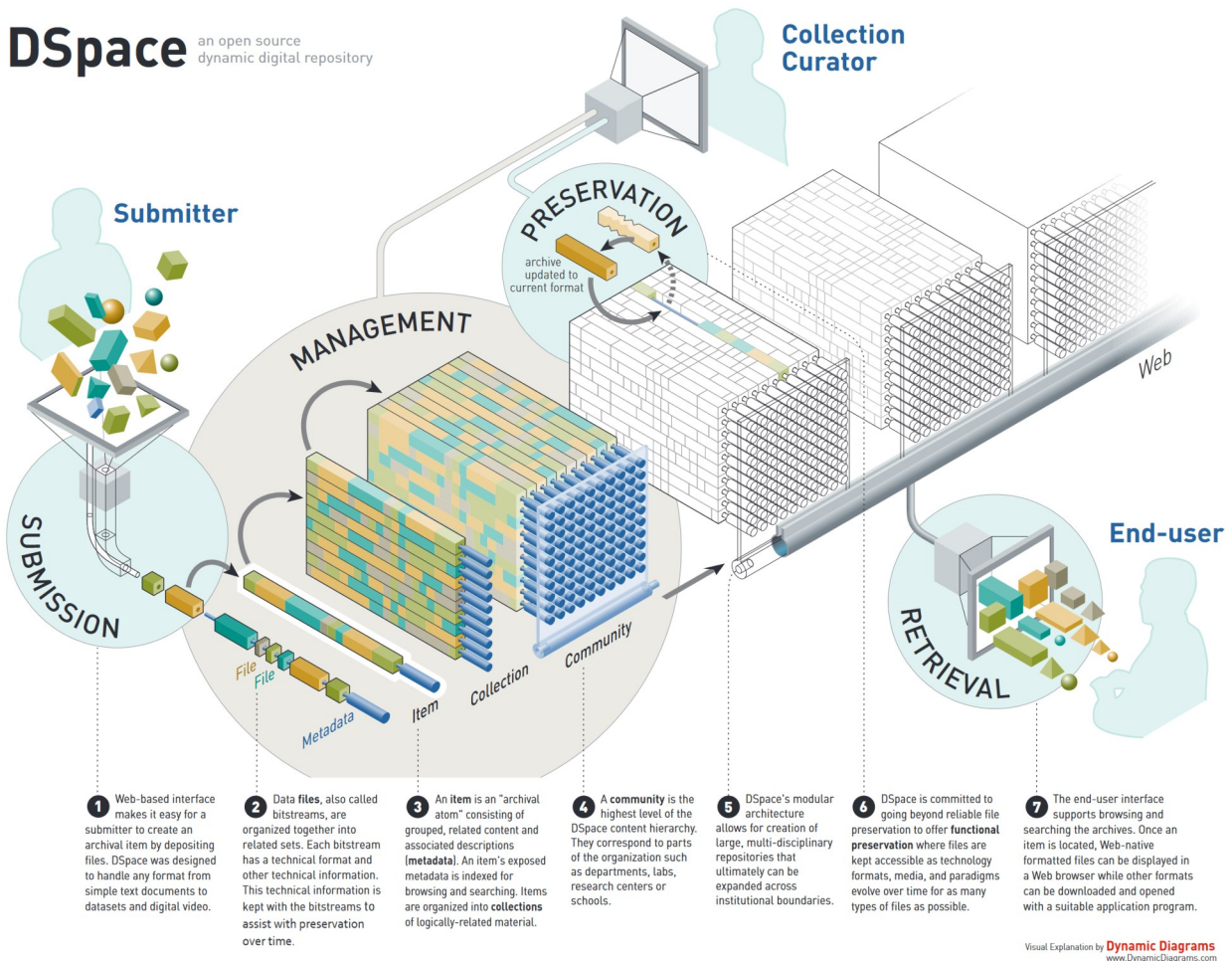
## Table of Contents

1. Introduction.....	3
1.1 บทนำ.....	3
1.2 องค์ประกอบที่สำคัญของคลังปัญญามหาวิทยาลัยสงขลานครินทร์.....	4
1.3 Browse and Search.....	5
2. การติดตั้งระบบ.....	6
2.1 ระบบปฏิบัติการ.....	6
2.2 การติดตั้งส่วนประกอบพื้นฐาน.....	6
2.3 การติดตั้ง DSpace.....	11
2.4 การปรับแต่ง Dspace.....	15
3. การดูแลระบบ.....	27
3.1 การจัดการฮาร์ดแวร์ของระบบ.....	27
3.2 การสำรองข้อมูล.....	27
3.3 การนำข้อมูลสำรองกลับมาใช้งาน.....	30
3.4 การจัดการระบบด้วยวิธี Manual.....	33
3.4.1 การจัดการชื่อและนามสกุล.....	33
3.4.2 การดึงรายการข้อมูลในระบบ.....	33
4. ภาคผนวก.....	36
4.1 DSpace Source Directories: /build/dspace-6.3-src-release.....	36
4.2 DSpace Web Applications.....	37
4.3 Dspace Database Structure.....	38
4.4 ไฟล์ /etc/cron.d/restarted.....	39
4.5 ไฟล์ /root/scripts/system_restarted.sh.....	39
4.6 ไฟล์ /etc/cron.d/add2deny.....	39
4.7 ไฟล์ /root/scripts/denyhosts.sh.....	39

# 1. Introduction

## 1.1 บทนำ

PSU Knowledge Bank หรือ คลังปัญญามหาวิทยาลัยสงขลานครินทร์ เป็น Institutional Repository หรือ IR ใช้เป็นแหล่งรวบรวมผลงานต่าง ๆ ของมหาวิทยาลัยสงขลานครินทร์ หรือผลงานที่เกี่ยวข้อง เป็นระบบที่พัฒนาโดยใช้ Dspace version 6.3 ซึ่งเป็นชุดโปรแกรมที่พัฒนาโดยห้องสมุดสถาบันเทคโนโลยีแห่งแมสซาชูเซตส์ (Massachusetts Institute of Technology Library) หรือ MIT Library โดยการสนับสนุนของบริษัทอิวเลตต์-แพคการ์ด และเผยแพร่สู่สาธารณะในลักษณะ Open Source Software เป็นโปรแกรมที่มีความเหมาะสมในการจัดเก็บ และเผยแพร่ผลงานดิจิทัลในระยะยาวที่มีความเสถียร และมีชุมชน DSpace Open Source Community ทำหน้าที่พัฒนาโปรแกรมโปรแกรมอย่างต่อเนื่อง ผังการทำงานของ DSpace มีลักษณะดังรูปที่ 1.1

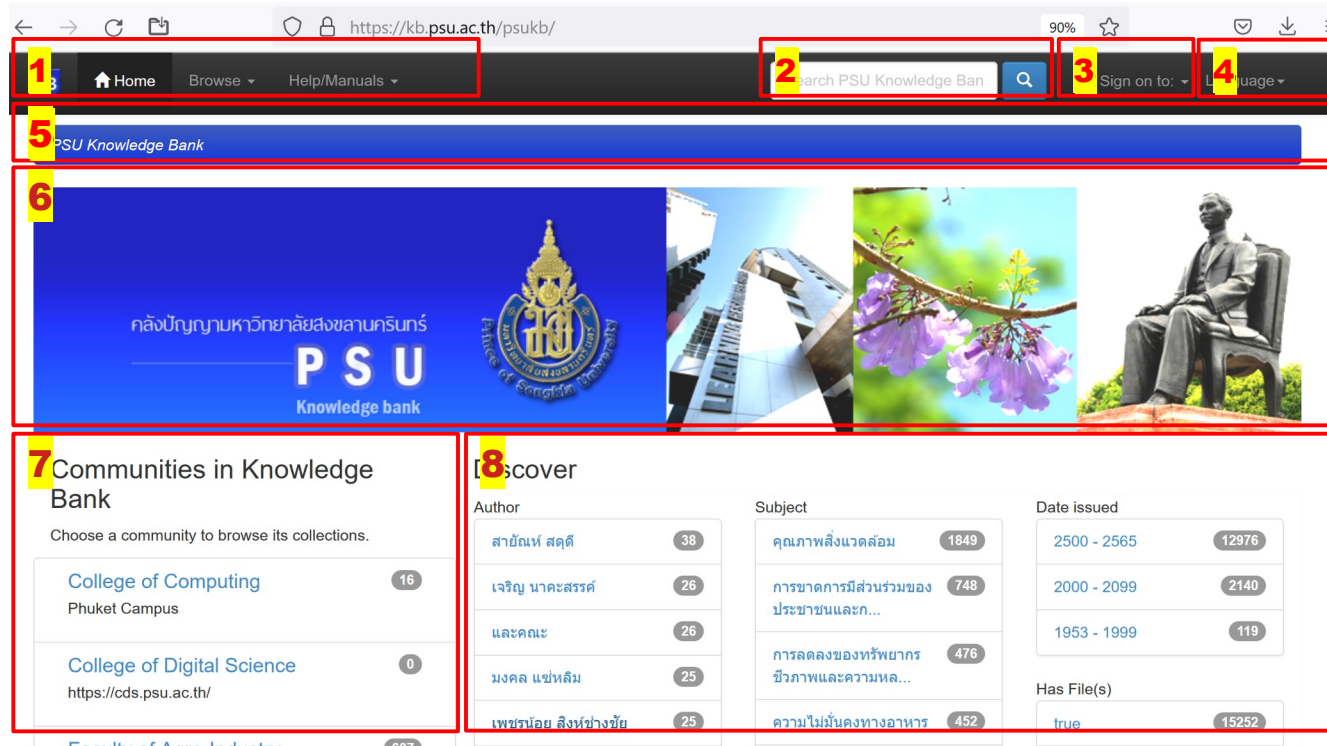


รูปที่ 1.1 แผนผังการทำงานของ DSpace

นอกจากนั้น โปรแกรม DSpace ยังมีระบบจัดการ Workflow ที่มีความยืดหยุ่น สามารถกำหนดสิทธิ์ในแต่ละขั้นตอนของ Workflow ได้ โดยขั้นตอนที่สำคัญใน Workflow ประกอบด้วย การนำเข้า (Submit) การยอมรับ/ปฏิเสธ(Approve/Reject) ปรับปรุง/แก้ไข(Edit) เป็นต้น

## 1.2 องค์ประกอบที่สำคัญของคลังปัญญามหาวิทยาลัยสงขลานครินทร์

ในหน้าแรกของคลังปัญญามหาวิทยาลัยสงขลานครินทร์ ประกอบด้วยองค์ประกอบหลัก ๆ ดังรูปที่ 1.3 และรูปที่ 1.4 ดังนี้



รูปที่ 1.2 องค์ประกอบในหน้าแรก



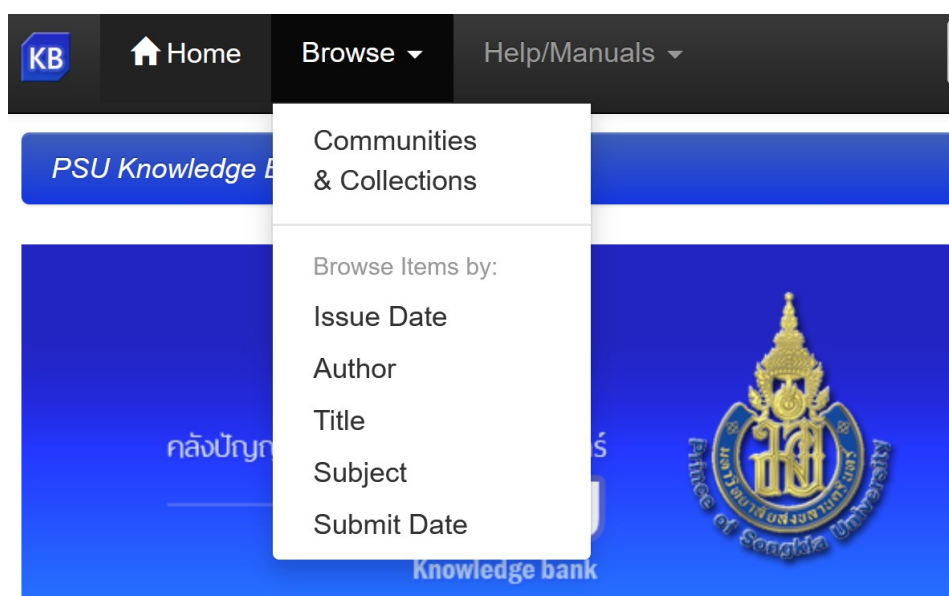
รูปที่ 1.3 องค์ประกอบในหน้าแรก (ต่อ)

- 1 Navigation bar ที่ประกอบด้วย Drop down main menu (มีความแตกต่างตามสิทธิ์ที่กำหนด)
- 2 Default search box ใน Navigation bar (สำหรับการค้นหาเฉพาะเนื้อหาภายในคลังปัญญาฯ)
- 3 Drop down menu สำหรับผู้ใช้งานที่มีการลงทะเบียน
- 4 Drop down menu สำหรับเลือกภาษา

- 5 Location bar
- 6 Banner หลักของระบบคลังปัญญาฯ
- 7 แสดงเนื้อหาตามโครงสร้างของคลังปัญญาฯ เมื่อคลิกที่แต่ละ Community ก็จะแสดงโครงสร้างย่อยภายใน Community นั้น
- 8 Discovery facet แสดงจำนวนเนื้อหาตาม Author, Subject, และ Year (Discovery facet sidebar)
- 9 Footer

### 1.3 Browse and Search

การเรียกดูข้อมูล(Browse) สามารถเรียกผ่าน Browse ใน Drop-down menu ดังรูปที่ 1.4 โดยสามารถเลือกเรียกดูตาม Communities/Collections, Date, Author, Title, Subject เป็นต้น



ปที่ 1.4 การค้นหาโดยเลือกรายการจากเมนู Browse

## 2. การติดตั้งระบบ

### 2.1 ระบบปฏิบัติการ

ในที่นี้ได้เลือกระบบปฏิบัติการลินุกซ์ของ Ubuntu โดยเป็นรุ่นล่าสุด คือ Ubuntu 20.04 LTS โดยแม่ข่ายจะเป็น Cloud Server หรือเป็นเครื่องคอมพิวเตอร์ที่จัดหามาดูแลเองก็ได้ สำหรับคลังปัญญาฯ ม.อ. ได้ใช้ VM ของสำนักนวัตกรรมดิจิทัลและระบบอัจฉริยะ โดยได้รับการจัดสรรทรัพยากร ดังนี้

- 1) จำนวน CPU: 8
- 2) หน่วยความจำ (RAM): 8GB
- 3) Swap disk: 8GB
- 4) Disk storage: 1TB

นอกจากนั้น ได้ประสานกับสำนักนวัตกรรมดิจิทัลและระบบอัจฉริยะ เพื่อลงทะเบียนเครื่องแม่ข่ายนี้ในชื่อ kb.psu.ac.th และควรมีแม่ข่ายสำรองโดยอาจจะตั้งชื่อเป็น kb2tmp.psu.ac.th เป็นต้น

### 2.2 การติดตั้งส่วนประกอบพื้นฐาน

หัวข้อนี้กล่าวถึงการติดตั้งส่วนประกอบพื้นฐานที่มีความจำเป็นสำหรับการทำงานของโปรแกรม DSpace ซึ่งประกอบด้วย

- (1) Java JDK 7 or 8 (OpenJDK or Oracle JDK)  
เนื่องจาก DSpace เป็นโปรแกรมที่พัฒนาด้วย Java Server Pages(JSP) จึงจำเป็นต้องมี Java Platform สำหรับการทำงานของโปรแกรมต่าง ๆ ในที่นี้ ได้เลือกใช้ OpenJDK 8 ที่อยู่ใน package ของ Ubuntu อยู่แล้ว
- (2) Apache Maven 3.0.5 or above (Java build tool)  
Apache Maven เป็นเครื่องมือสำหรับ compile โปรแกรมที่พัฒนาด้วย JSP
- (3) Apache Ant 1.8 or later (Java build tool)  
Apache Ant เป็นเครื่องมือสำหรับการติดตั้งโปรแกรมที่ผ่านการ compile เรียบร้อยแล้ว
- (4) Relational Database: (PostgreSQL or Oracle) เวอร์ชัน 9.4 ขึ้นไป  
ฐานข้อมูลสำหรับ DSpace ในที่นี้เลือกใช้ PostgreSQL 10
- (5) Servlet Engine (Apache Tomcat 7 or later, Jetty, Caucho Resin or equivalent)

Servlet Engine ทำหน้าที่จัดการโปรแกรมที่พัฒนาด้วย JSP เพื่อการโต้ตอบระหว่างแม่ข่ายกับลูกข่าย(ทำหน้าที่เหมือนกับ web server) ในที่นี้เลือกใช้ Apache Tomcat 8.5.51

(6) Web Server (Apache2 or Nginx)

ในที่นี้เลือกใช้ Nginx ที่อยู่ใน package ของ Ubuntu ทำหน้าที่เป็น web server

(7) certbot และ python3-certbot-nginx (กรณีที่ต้องการใช้ Certificate ฟรี)

เป็นเครื่องมือสำหรับสร้างใบรับรองด้วย Let's Encrypt เพื่อรองรับการใช้งานด้วย secure http หรือ https อย่างไรก็ตาม สำหรับคลังปัญญา ม.อ. ได้เลือกใช้ ssl certificate ที่จัดหาโดยสำนักนวัตกรรมดิจิทัลและระบบอัจฉริยะ จึงไม่ต้องติดตั้งเครื่องมือนี้

ในที่นี้ ขอสรุปรายละเอียดของการติดตั้ง และการปรับแต่งเบื้องต้น ดังนี้

1) ติดตั้งเครื่องมือข้างต้น(ยกเว้น Apache Tomcat) ด้วยคำสั่ง

```
sudo apt install openjdk-8-jdk ant maven postgresql postgresql-contrib nginx certbot python3-certbot-nginx
```

กรณีไม่ใช่ Let's Encrypt ก็ไม่ต้องติดตั้ง certbot python3-certbot-nginx

2) สร้าง user "tomcat" เพื่อเป็น user ที่ทำหน้าที่ให้บริการ Tomcat servlet engine และ DSpace

```
sudo useradd -m tomcat
```

กำหนดรหัสผ่านให้ tomcat

```
sudo passwd tomcat
```

3) ติดตั้ง Apache Tomcat 8.5.65 ด้วยขั้นตอนดังนี้

```
cd /tmp
wget https://archive.apache.org/dist/tomcat/tomcat-8/v8.5.65/bin/apache-tomcat-8.5.65.tar.gz
sudo tar xvzf apache-tomcat-8.5.65.tar.gz
sudo mv apache-tomcat-8.5.65 tomcat
sudo mv tomcat /opt
```

และให้ tomcat เป็นเจ้าของไฟล์ทั้งหมดใน /opt/tomcat

```
sudo chown -R tomcat.tomcat /opt/tomcat
```

4) สร้าง systemd service file เพื่อใช้เปิดและปิด Tomcat service

ใช้คำสั่ง `sudo pico /etc/systemd/system/tomcat.service` เพื่อสร้างไฟล์ที่มีเนื้อหาดังนี้

```
[Unit]
Description=Tomcat 8.5 servlet container
After=network.target
```

```
[Service]
```

```
Type=forking

User=tomcat
Group=tomcat

Environment="JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64"
Environment="CATALINA_BASE=/opt/tomcat"
Environment="CATALINA_HOME=/opt/tomcat"
Environment="CATALINA_PID=/opt/tomcat/temp/tomcat.pid"

Environment="CATALINA_OPTS=-Xms256m -Xmx2048m -server -XX:+UseParallelGC"
Environment="JAVA_OPTS=-Dfile.encoding=UTF-8 -Xms256m -Xmx2048m -XX:MaxPermSize=256m -server -XX:+UseParallelGC"

ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh

[Install]
WantedBy=multi-user.target
```

หลังจากนั้น ป้อนคำสั่งนี้เพื่อให้ไฟล์นี้พร้อมสำหรับการใช้งาน

```
sudo systemctl daemon-reload
```

ต่อไปก็จะสามารถสั่งให้เปิด ปิด และตรวจสอบการทำงานของ Tomcat ด้วยคำสั่ง

```
sudo systemctl start tomcat
sudo systemctl stop tomcat
sudo systemctl status tomcat
```

และคำสั่งต่อไปนี้ จะทำให้ Tomcat เริ่มทำงานทุกครั้งเมื่อเปิด(boot or reboot)ระบบปฏิบัติการ

```
sudo systemctl enable tomcat
```

5) ปรับแต่งการตั้งค่าของ Tomcat ที่ไฟล์ /opt/tomcat/conf/server.xml ในส่วนของ <Connector> โดยปรับให้มีการตั้งค่า ดังนี้

```
<!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->
<Connector port="8080"
  minSpareThreads="25"
  enableLookups="false"
  redirectPort="8443"
  connectionTimeout="20000"
  disableUploadTimeout="true"
  URIEncoding="UTF-8"/>
```

6) ทดสอบการทำงานของ Tomcat เริ่มต้นด้วยการ Start Tomcat ด้วยคำสั่ง

```
sudo systemctl start tomcat
```



จากนั้น ป้อน `http://kb.psu.ac.th:8080` ใน Browser

7) ปรับแต่ง Java ด้วยการเพิ่ม 2 บรรทัดนี้ในไฟล์ `/etc/profile`

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export CATALINA_HOME=/opt/tomcat
```

8) เลือกคำสั่ง java ที่เป็น default โดยพิมพ์คำสั่งนี้

```
sudo update-alternatives --config java
```

หากมีหลายตัวเลือก ให้เลือก `/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java 1081 manual mode`  
หากมีเพียง 1 ตัวเลือก ระบบจะแจ้งรายละเอียด ให้แน่ใจว่า คือ `/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java`

9) ปรับแต่ง nginx

ปรับการตั้งค่าในไฟล์ `/etc/nginx/nginx.conf` ดังนี้

```
keepalive_timeout 240;
types_hash_max_size 2048;
client_body_timeout 180;
client_max_body_size 512M;
ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref: POODLE
```

ปรับการตั้งค่าของไฟล์ใน `/etc/nginx/sites-enabled` โดยให้ตั้งค่า `server_name` ดังนี้

```
...
server_name kb.psu.ac.th;
...
```

ทดสอบการทำงานของ nginx ว่า ทำงานได้ในโหมด http โดยป้อน `http://kb.psu.ac.th` ใน Browser

10) กรณีที่ใช้ Let's Encrypt สร้าง และอัปเดตใบรับรองด้วย Let's Encrypt โดยการใช้คำสั่ง ดังนี้

```
sudo certbot --nginx -d kb.psu.ac.th
```

และตอบแต่ละคำถามตามข้อความสีฟ้า ดังนี้

```
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Plugins selected: Authenticator nginx, Installer nginx
Enter email address (used for urgent renewal and security notices) (Enter 'c' to cancel):
panyarak@yahoo.com ← หรือ email ของผู้ดูแลระบบ

-----
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf. You must
agree in order to register with the ACME server at
https://acme-v02.api.letsencrypt.org/directory

-----
(A)gree/(C)ancel: A
```

-----  
Would you be willing to share your email address with the Electronic Frontier Foundation, a founding partner of the Let's Encrypt project and the non-profit organization that develops Certbot? We'd like to send you email about our work encrypting the web, EFF news, campaigns, and ways to support digital freedom.  
-----

(Y)es/(N)o: **Y**

Obtaining a new certificate

Performing the following challenges:

http-01 challenge for [kb.psu.ac.th](http://kb.psu.ac.th)

Waiting for verification...

Cleaning up challenges

Deploying Certificate to VirtualHost /etc/nginx/sites-enabled/test.kb

Please choose whether or not to redirect HTTP traffic to HTTPS, removing HTTP access.

-----  
1: No redirect - Make no further changes to the webserver configuration.

2: Redirect - Make all requests redirect to secure HTTPS access. Choose this for new sites, or if you're confident your site works on HTTPS. You can undo this change by editing your web server's configuration.

-----  
Select the appropriate number [1-2] then [enter] (press 'c' to cancel): **2**

Redirecting all traffic on port 80 to ssl in /etc/nginx/sites-enabled/test.kb

-----  
Congratulations! You have successfully enabled <https://kb.psu.ac.th>

You should test your configuration at:

<https://www.ssllabs.com/ssltest/analyze.html?d=kb.psu.ac.th>

-----  
**IMPORTANT NOTES:**

- Congratulations! Your certificate and chain have been saved at:  
</etc/letsencrypt/live/kb.psu.ac.th/fullchain.pem>  
Your key file has been saved at:  
</etc/letsencrypt/live/kb.psu.ac.th/privkey.pem>  
Your cert will expire on **2021-11-28**. To obtain a new or tweaked version of this certificate in the future, simply run certbot again with the "certonly" option. To non-interactively renew *\*all\** of your certificates, run "certbot renew"
- Your account credentials have been saved in your Certbot configuration directory at /etc/letsencrypt. You should make a secure backup of this folder now. This configuration directory will also contain certificates and private keys obtained by Certbot so

```
making regular backups of this folder is ideal.
- If you like Certbot, please consider supporting our work by:
Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate
Donating to EFF: https://eff.org/donate-le
```

Certbot จะตรวจสอบอายุของใบรับรองตามที่ตั้งค่าไว้ใน /etc/cron.d/certbot และจะทำการ renew ให้ใหม่ อ่านรายละเอียดเกี่ยวกับ Let's Encrypt เพิ่มเติมได้ที่

```
https://www.digitalocean.com/community/tutorials/how-to-secure-nginx-with-let-s-encrypt-on-ubuntu-20-04
```

- 11) หากไม่ใช่ Let's Encrypt ให้กำหนด SSL certificate ของ ม.อ. ใน nginx ให้ถูกต้องด้วย
- 12) สร้างฐานข้อมูลสำหรับ DSpace ด้วยขั้นตอน ดังนี้

หาก postgresql@12-main ยังไม่เริ่มต้นทำงาน ให้ใช้คำสั่ง ดังนี้

```
sudo systemctl start postgresql@12-main
```

เปลี่ยนบทบาทเป็น postgres (super user ของ PostgreSQL database server)

```
sudo su postgres
```

สร้าง PostgreSQL user หรือ role "tomcat" เพื่อทำหน้าที่เป็นเจ้าของฐานข้อมูล

```
createuser -U postgres -d -A -P tomcat (ในที่นี้ใช้ pw: [ระบุรหัสผ่าน] ให้ป้อนรหัสผ่านนี้ 2 ครั้ง)
Enter password for new role:
Enter it again:
```

ออกจากบทบาท postgres ด้วยคำสั่ง exit

สร้างฐานข้อมูลชื่อ trc โดยใช้บทบาทของ tomcat

```
sudo -u tomcat createdb -U tomcat -E UNICODE trc
```

เปลี่ยนบทบาทเป็น postgres แล้วเพิ่ม pgcrypto extension ให้ฐานข้อมูล trc

```
sudo su postgres
psql --username=postgres trc -c "CREATE EXTENSION pgcrypto;"
```

เมื่อทำสำเร็จแล้ว ให้ออกจากบทบาท postgres ด้วยคำสั่ง exit

## 2.3 การติดตั้ง DSpace

ในส่วนของ DSpace แบ่งพื้นที่ออกเป็น 2 ส่วนใหญ่ ๆ คือ

- (1) DSpace Source เป็นพื้นที่สำหรับ source codes ของโปรแกรม ใช้สำหรับการ compile เพื่อให้ได้ executable files เพื่อนำไปใช้งานจริงต่อไป ก่อนการ compile ต้องตั้งค่าต่าง ๆ ที่จำเป็นให้เรียบร้อยก่อนเสมอ ในที่นี้ ให้พื้นที่นี้อยู่ที่ /build
- (2) DSpace Applications เป็นพื้นที่ติดตั้งโปรแกรมที่ผ่านการ compile เรียบร้อยแล้ว รวมทั้งเป็นพื้นที่จัดเก็บไฟล์ข้อมูล และไฟล์ log ต่าง ๆ ด้วย ในที่นี้ ให้พื้นที่นี้อยู่ที่ /dspace

ขั้นตอนการติดตั้งโปรแกรม DSpace มีดังนี้

- 1) สร้าง directory /build และ /dspace โดยให้ user **panya** (หรือ user ที่ทำหน้าที่ดูแลระบบ) และ tomcat เป็นเจ้าของตามลำดับ

```
sudo mkdir /build /dspace
sudo chown panya.panya /build (ชื่อและกรุปที่เป็นสีแดง ให้ใช้ชื่อของผู้ดูแลระบบ)
sudo chown tomcat.tomcat /dspace
```

- 2) ดาวน์โหลดโปรแกรม DSpace รุ่น 6.3 (หรือรุ่นล่าสุดที่ต้องการ) และแตกไฟล์ไว้ใน /build ดังนี้

```
cd /build
wget https://github.com/Dspace/Dspace/releases/download/dspace-6.3/dspace-6.3-src-release.tar.gz
tar -zxf dspace-6.3-src-release.tar.gz
```

- 3) ปรับปรุงการตั้งค่าใน /build/dspace-6.3-src-release/dspace/config/local.cfg (ดูรายละเอียดในไฟล์ [ดังกล่าว](#)) ทั้งนี้ ตั้งแต่ version 6 เป็นต้นมา ผู้พัฒนาได้ออกแบบให้ไม่ต้องตั้งค่าในไฟล์ต่าง ๆ เช่น dspace.cfg และบรรดาไฟล์ที่นามสกุล cfg อีก แต่สามารถนำ parameter ที่มีการตั้งค่าสำหรับระบบที่ต้องการให้บริการมาไว้ใน local.cfg เพียงไฟล์เดียว

- 4) สำเนาไฟล์บางไฟล์ใน /build/dspace-6.3-src-release/dspace-jspui/src/main/webapp/ ที่ต้องการปรับเปลี่ยนไปที่ /build/dspace-6.3-src-release/dspace/module/jspui/src/main/webapp/ และปรับเปลี่ยนไฟล์เหล่านี้ให้ตรงกับความต้องการ ตัวอย่างไฟล์เหล่านี้ได้แก่ home.jsp favicon.ico collection-home.jsp รวมทั้งไฟล์ใน sub-directories ด้วย ฯลฯ

- 5) ไฟล์ Messages.properties

ให้สำเนาไฟล์ดังกล่าวจาก /build/dspace-6.3-src-release/dspace-api/src/main/resources ไปไว้ที่ /build/dspace-6.3-src-release/dspace/module/jspui/src/main/resources รวมทั้งไฟล์ที่เป็นภาษาอื่น ๆ (ถ้ามี) ด้วย เช่น Messages\_th.properties เป็นต้น

- 6) ตัวอย่างไฟล์ที่มีการปรับปรุง เช่น /build/dspace-6.3-src-release/dspace/config/input-form.xml เพิ่ม “Advisors” field ใน submit form หลัง “Authors” ดังนี้ (ตัวอย่างการปรับไฟล์อื่น ๆ ดูเพิ่มเติมที่ข้อ 2.4)

```
<field>
  <dc-schema>dc</dc-schema>
  <dc-element>contributor</dc-element>
  <dc-qualifier>advisor</dc-qualifier>
  <repeatable>true</repeatable>
  <label>Advisors</label>
  <input-type>name</input-type>
  <hint>Enter the names of the advisors of this item below.</hint>
  <required></required>
</field>
```

เพิ่ม “Abstract-th” field ใน submit form ได้ “Abstract” ดังนี้

```
<field>
  <dc-schema>dc</dc-schema>
  <dc-element>description</dc-element>
  <dc-qualifier>abstract-th</dc-qualifier>
  <repeatable>>false</repeatable>
  <label>Abstract in Thai</label>
  <input-type>textarea</input-type>
  <hint>Enter the abstract of the item in Thai. </hint>
  <required></required>
</field>
```

- 7) ดาวน์โหลดฐานข้อมูล GeoLite2 City จาก <https://www.maxmind.com/> และนำไฟล์ GeoLite2-City.mmdb ที่ได้มา ไปวางไว้ใน /build/dspace-6.3-src-release/dspace/config มิฉะนั้น จะทำให้เกิดข้อผิดพลาดในระหว่างการ compile ได้
- 8) เมื่อตั้งค่าในไฟล์ต่าง ๆ เรียบร้อยแล้ว ให้เริ่ม compile โปรแกรมด้วยขั้นตอนดังนี้ (ในขั้นตอนนี้ ไม่จำเป็นต้องมีบทบาทเป็น Super user)

```
cd /build/dspace-6.3-src-release/dspace
mvn -U clean package
```

- 9) หาก compile สำเร็จ จะได้ข้อความ **BUILD SUCCESS** ดังรูปที่ 2.1

```
[INFO] -----
[INFO] Reactor Summary for DSpace Parent Project 6.3:
[INFO]
[INFO] DSpace Parent Project ..... SUCCESS [ 0.694 s]
[INFO] DSpace Services Framework :: API and Implementation SUCCESS [ 4.071 s]
[INFO] DSpace Kernel :: API and Implementation ..... SUCCESS [ 13.294 s]
[INFO] DSpace Addon Modules ..... SUCCESS [ 0.010 s]
[INFO] DSpace Kernel :: Additions and Local Customizations SUCCESS [ 1.628 s]
[INFO] DSpace XML-UI (Manakin) ..... SUCCESS [ 10.845 s]
[INFO] DSpace XML-UI (Manakin) :: Local Customizations .... SUCCESS [ 9.154 s]
[INFO] DSpace JSP-UI ..... SUCCESS [ 2.742 s]
[INFO] DSpace JSP-UI :: Local Customizations ..... SUCCESS [ 5.939 s]
[INFO] DSpace RDF ..... SUCCESS [ 5.605 s]
[INFO] DSpace RDF :: Local Customizations ..... SUCCESS [ 6.203 s]
[INFO] DSpace REST :: API and Implementation ..... SUCCESS [ 9.121 s]
[INFO] DSpace REST :: Local Customizations ..... SUCCESS [ 6.960 s]
[INFO] DSpace SWORD ..... SUCCESS [ 2.719 s]
[INFO] DSpace SWORD :: Local Customizations ..... SUCCESS [ 5.106 s]
[INFO] DSpace SWORD v2 ..... SUCCESS [ 1.285 s]
[INFO] DSpace SWORD v2 :: Local Customizations ..... SUCCESS [ 5.164 s]
[INFO] Apache Solr Webapp ..... SUCCESS [ 16.504 s]
[INFO] DSpace SOLR :: Local Customizations ..... SUCCESS [ 5.971 s]
[INFO] DSpace OAI-PMH ..... SUCCESS [ 2.361 s]
[INFO] DSpace OAI-PMH :: Local Customizations ..... SUCCESS [ 7.387 s]
[INFO] DSpace Assembly and Configuration ..... SUCCESS [ 17.928 s]
[INFO] DSpace XML-UI Mirage2 Theme ..... SUCCESS [ 0.144 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 02:21 min
[INFO] Finished at: 2019-09-11T21:46:00Z
[INFO] -----
```

รูปที่ 2.1

10) เมื่อ compile สำเร็จแล้ว ให้ติดตั้งโปรแกรม DSpace ด้วยคำสั่ง ดังนี้

```
cd /build/dspace-6.3-src-release/dspace/target/dspace-installer
sudo ant fresh_install
```

หากไม่มีข้อผิดพลาดใด ๆ ให้เปลี่ยนเจ้าของไฟล์ต่าง ๆ ใน /dlib และ restart Tomcat ดังนี้

```
sudo chown tomcat.tomcat -R /dspace
sudo systemctl restart tomcat
```

11) สร้าง administrator ของระบบ ด้วยคำสั่ง

```
sudo /dspace/bin/dspace create-administrator
```

administrator นี้ ใช้สำหรับ login เข้าระบบผ่าน UI ของ TDC และมีสิทธิในการจัดการระบบทั้งหมด รวมทั้งสามารถมอบสิทธิในบางเรื่องให้กับผู้ใช้งานบางคน หรือบางกลุ่มได้

12) ปรับปรุงค่าตัวแปร JAVA\_OPTS ในไฟล์ /dspace/bin/dspace ให้ใช้หน่วยความจำได้อย่างเพียงพอ หรือเหมือนกับที่ระบุใน /etc/systemd/system/tomcat.service ตัวอย่างเช่น

```
JAVA_OPTS="-Xmx2048m -Xms1024m -Dfile.encoding=UTF-8"
```

13) สร้างไฟล์ xml ใน /opt/tomcat/conf/Catalina/localhost เพื่อชี้ไปยัง application ต่าง ๆ ตามที่ต้องการ เช่น ไฟล์ psukb.xml เพื่อชี้ไปยัง /dspace/webapps/jspui ทำให้สามารถเรียกใช้งานผ่าน url <https://kb.psu.ac.th/psukb> โดยเนื้อหาในไฟล์ psukb.xml มีดังนี้

```
<?xml version='1.0' encoding='utf-8'?>
<Context
  docBase="/dlib/webapps/jspui"
  debug="0"
  reloadable="true"
  cachingAllowed="false"
  allowLinking="true"
/>
```

ตัวอย่างอีกไฟล์ คือ oai.xml สำหรับเรียกดูข้อมูล oai ผ่าน <https://kb.psu.ac.th/oai> โดยมีเนื้อหา ดังนี้

```
<?xml version='1.0' encoding='utf-8'?>
<Context
  docBase="/dspace/webapps/oai"
  debug="0"
  reloadable="true"
  cachingAllowed="false"
  allowLinking="true"
/>
```

จะสังเกตได้ว่า บรรทัดที่แตกต่าง คือ docBase="" และสร้างไฟล์ที่เหลือให้ครบ คือ rdf.xml, rest.xml, solr.xml, sword.xml, swordv2.xml

14) Restart Tomcat แล้วรอสัก 1-2 นาที แล้วใช้ browser เปิดหน้าของ <https://kb.psu.ac.th>

## 2.4 การปรับแต่ง Dspace

1) การตั้งค่าใน `/dspace/config/local.cfg`

```
#####  
# SERVER CONFIGURATION #  
#####  
  
# DSpace installation directory.  
# This is the location where you want to install DSpace.  
# Windows note: Please remember to use forward slashes for all paths (e.g. C:/dspace)  
dspace.dir=/dspace  
  
# DSpace host name - should match base URL. Do not include port number  
dspace.hostname = kb.psu.ac.th  
  
# DSpace base host URL. Include port number etc.  
dspace.baseUrl = http://${dspace.hostname}  
  
# The user interface you will be using for DSpace. Common usage is either xmlui or jspui  
dspace.ui = jspui  
  
# Full link your end users will use to access DSpace. In most cases, this will be the baseUrl followed by  
# the context path to the UI you are using.  
#  
# Alternatively, you can use a url redirect or deploy the web application under the servlet container root.  
# In this case, make sure to remove the /${dspace.ui} from the dspace.url property.  
dspace.url = ${dspace.baseUrl}/psukb  
  
# Name of the site  
dspace.name = PSU Knowledge Bank  
  
# assetstore.dir, look at DSPACE/config/spring/api/bitstore.xml for more options  
# default is ${dspace.dir}/assetstore, uncomment and modify if you need to use a different path  
assetstore.dir = ${dspace.dir}/assetstore  
  
# Default language for metadata values  
default.language = en_US  
webui.supported.locales = en, th  
  
# Solr server/webapp.  
# DSpace uses Solr for all search/browse capability (and for usage statistics by default).  
# The included 'solr' webapp MUST be deployed to Tomcat for DSpace to function.  
# Usually it will be available via port 8080 and the 'solr' context path.  
# But, you may need to modify this if you are running DSpace on a custom port, etc.  
solr.server = http://localhost:8080/solr  
  
#####
```

```

# DATABASE CONFIGURATION #
#####
# DSpace only supports two database types: PostgreSQL or Oracle

# URL for connecting to database
# * Postgres template: jdbc:postgresql://localhost:5432/dspace
# * Oracle template: jdbc:oracle:thin:@//localhost:1521/xe
db.url = jdbc:postgresql://localhost:5432/kb

# JDBC Driver
# * For Postgres: org.postgresql.Driver
db.driver = org.postgresql.Driver

# Database Dialect (for Hibernate)
# * For Postgres: org.dspace.storage.rdbms.hibernate.postgres.DSpacePostgreSQL82Dialect
db.dialect = org.dspace.storage.rdbms.hibernate.postgres.DSpacePostgreSQL82Dialect

# Database username and password
db.username = tomcat
db.password = {password}

# Database Schema name
# * For Postgres, this is often "public" (default schema)
#   so this may be set to ${db.username} in most scenarios.
db.schema = public

### Connection pool parameters

# Maximum number of DB connections in pool (default = 30)
db.maxconnections = 90

# Maximum time to wait before giving up if all connections in pool are busy (milliseconds)
db.maxwait = 5000

# Maximum number of idle connections in pool (-1 = unlimited)
db.maxidle = 60

#####
# EMAIL CONFIGURATION #
#####

# SMTP mail server (allows DSpace to send email notifications)
mail.server = smtp.psu.ac.th

# SMTP mail server authentication username and password (if required)
#mail.server.username = myusername
#mail.server.password = mypassword

# SMTP mail server alternate port (defaults to 25)
mail.server.port = 25

# From address for mail
# All mail from the DSpace site will use this 'from' address
mail.from.address = psukb-admin@psu.ac.th

```



```

# When feedback is submitted via the Feedback form, it is sent to this address
# Currently limited to one recipient!
feedback.recipient = psukb-admin@psu.ac.th

# General site administration (Webmaster) e-mail
mail.admin = psukb-admin@psu.ac.th

# Recipient for server errors and alerts (defaults to mail.admin)
alert.recipient = ${mail.admin}

# Recipient for new user registration emails (defaults to unspecified)
registration.notify = psukb-admin@psu.ac.th

#####
# HANDLE CONFIGURATION #
#####
#
handle.canonical.prefix = ${dspace.url}/handle/

# CNRI Handle prefix
# (Defaults to a dummy/fake prefix of 123456789)
handle.prefix = 2016

# Directory for installing Handle server files
handle.dir = ${dspace.dir}/handle-server

#####
# AUTHENTICATION METHODS #
#####

# Shibboleth authentication/authorization. See authentication-shibboleth.cfg for default configuration.
#plugin.sequence.org.dspace.authenticate.AuthenticationMethod =
org.dspace.authenticate.ShibAuthentication

# X.509 certificate authentication. See authentication-x509.cfg for default configuration.
#plugin.sequence.org.dspace.authenticate.AuthenticationMethod =
org.dspace.authenticate.X509Authentication

# IP-based authentication/authorization. See authentication-ip.cfg for default configuration.
#plugin.sequence.org.dspace.authenticate.AuthenticationMethod =
org.dspace.authenticate.IPAAuthentication

# LDAP authentication/authorization. See authentication-ldap.cfg for default configuration.
plugin.sequence.org.dspace.authenticate.AuthenticationMethod =
org.dspace.authenticate.LDAPAuthentication

# Authentication by Password (encrypted in DSpace's database). See authentication-password.cfg for
default configuration.
# Enabled by default in authentication.cfg
plugin.sequence.org.dspace.authenticate.AuthenticationMethod =
org.dspace.authenticate.PasswordAuthentication

#-----#
#-----USAGE STATISTICS CONFIGURATIONS-----#
#-----#

```

```

# Anyone with READ permissions on the DSpaceObject will be able
# to view the statistics.
usage-statistics.authorization.admin.usage=false

##### Google Analytics #####
# If you would like to use Google Analytics to track general website statistics then
# use the following parameter to provide your Analytics key. First sign up for an
# account at http://analytics.google.com, then create an entry for your repository
# website. Analytics will give you a snippet of JavaScript code to place on your site,
# inside that snippet is your Google Analytics key usually found in this line:
# _uacct = "UA-XXXXXXX-X"
# Take this key (just the UA-XXXXXX-X part) and place it here in this parameter.
jspui.google.analytics.key=UA-129379064-1

# Configuration settings for ORCID based authority control, uncomment the lines below to enable
configuration
solr.authority.server=${solr.server}/authority

#-----#
#-----JSPUI & XMLUI CONFIGURATIONS-----#
#-----#
# These configs are used by both JSP and XML User Interfaces, #
# except where explicitly stated otherwise. #
#-----#

webui.preview.brand = Prince of Songkla University
webui.preview.brand.abbrev = PSU

##### Settings for item count (strength) information #####
webui.submit.blocktheses = false
webui.strengths.show = true
webui.strengths.cache = false

##### Settings for Submission Process #####
webui.submit.upload.required = false
webui.submission.restrictstep.enableAdvancedForm = false

##### Syndication Feed (RSS) Settings #####
webui.feed.localresolve = true

##### Creative Commons settings #####
webui.submit.enable-cc = true
cc.license.jurisdiction = th
cc.license.locale = en

##### Session invalidation #####
# Enable or disable session invalidation upon login or logout.
# This feature is enabled by default to help prevent session hijacking
# but may cause problems for shibboleth, etc
#
#webui.session.invalidate = false

# For compatibility with previous versions:
#

```

```

webui.browse.index.1 = dateissued:item:dateissued
webui.browse.index.2
    = author:metadata:dc.contributor.*\,dc.creator:text
webui.browse.index.3 = title:item:title
webui.browse.index.4 = subject:metadata:dc.subject.*:text
webui.browse.index.5 = dateaccessioned:item:dateaccessioned

#webui.browse.index.tagcloud.4 = true

# tell the community and collection pages that we are using the Recent
# Submissions code
plugin.sequence.org.dspace.plugin.SiteHomeProcessor = \
    org.dspace.app.webui.components.TopCommunitiesSiteProcessor,\
    org.dspace.app.webui.components.RecentSiteSubmissions

plugin.sequence.org.dspace.plugin.CommunityHomeProcessor = \
    org.dspace.app.webui.components.RecentCommunitySubmissions

plugin.sequence.org.dspace.plugin.CollectionHomeProcessor = \
    org.dspace.app.webui.components.RecentCollectionSubmissions,\
    org.dspace.app.webui.components.CollectionItemList

##### Sidebar Facets #####
# to show facets on the site home page, community, collection
# comment out the following lines if you disable Discovery or don't want
# to show facets on side bars
# TagCloudProcessor is responsible for displaying a tag-cloud facet on the
# site home page, community or collection home page
plugin.sequence.org.dspace.plugin.CommunityHomeProcessor = \
    org.dspace.app.webui.components.RecentCommunitySubmissions,\
    org.dspace.app.webui.discovery.SideBarFacetProcessor
#    org.dspace.app.webui.tagcloud.TagCloudProcessor

plugin.sequence.org.dspace.plugin.CollectionHomeProcessor = \
    org.dspace.app.webui.components.CollectionItemList,\
    org.dspace.app.webui.discovery.SideBarFacetProcessor,\
#    org.dspace.app.webui.tagcloud.TagCloudProcessor,\
    org.dspace.app.webui.components.RecentCollectionSubmissions

plugin.sequence.org.dspace.plugin.SiteHomeProcessor = \
    org.dspace.app.webui.components.TopCommunitiesSiteProcessor,\
    org.dspace.app.webui.components.RecentSiteSubmissions,\
    org.dspace.app.webui.discovery.SideBarFacetProcessor
#    org.dspace.app.webui.tagcloud.TagCloudProcessor

##### OpenSearch Settings #####
# NB: for result data formatting, OpenSearch uses Syndication Feed Settings
# so even if Syndication Feeds are not enabled, they must be configured
# enable open search
websvc.opensearch.enable = false
# context for html request URLs - change only for non-standard servlet mapping
websvc.opensearch.uicontext = simple-search
# context for RSS/Atom request URLs - change only for non-standard servlet mapping
websvc.opensearch.svccontext = open-search/

```

```

# present autodiscovery link in every page head
websvc.opensearch.autolink = true
# number of hours to retain results before recalculating
websvc.opensearch.validity = 48
# short name used in browsers for search service
# should be 16 or fewer characters
websvc.opensearch.shortname = PSUKB
# longer (up to 48 characters) name
websvc.opensearch.longname = ${dspace.name}
# brief service description
websvc.opensearch.description = ${dspace.name} Repository
# location of favicon for service, if any must be 16X16 pixels
websvc.opensearch.faviconurl = ${dspace.url}/favicon.ico
# sample query - should return results
websvc.opensearch.samplequery = photosynthesis
# tags used to describe search service
websvc.opensearch.tags = IR PSUKB
# result formats offered - use 1 or more comma-separated from: html,atom,rss
# NB: html is required for autodiscovery in browsers to function,
# and must be the first in the list if present
websvc.opensearch.formats = html,atom,rss

##### Item Recommendation Settings #####

# show a link to the item recommendation page from item display page
webui.suggest.enable = true
#
# Enable only, if the user is logged in.
# If not set the default value is false
webui.suggest.loggedinusers.only = true

##### Web Interface Settings #####

# Customise the DC metadata fields to show in the default simple item view.
#
webui.itemdisplay.default = dc.title, dc.title.alternative, dc.contributor.*, \
    dc.subject(nobreakline), dc.date.issued(date), dc.publisher, \
    dc.identifier.citation, dc.relation.ispartofseries, \
    dc.description.abstract, dc.description.abstract-th, dc.description, \
    dc.identifier.govdoc, dc.identifier.uri(link), \
    dc.identifier.isbn, dc.identifier.issn, \
    dc.identifier.ismn, dc.identifier

```

2) /dspace/config/modules/authentication-ldap.cfg

```

#-----#
#-----LDAP AUTHENTICATION CONFIGURATIONS-----#
#-----#

authentication-ldap.enable = true

##### LDAP AutoRegister Settings #####
authentication-ldap.autoregister = true

```

```

authentication-ldap.provider_url = ldaps://dc1.psu.ac.th:636/
authentication-ldap.id_field = sAMAccountName

authentication-ldap.object_context = dc=psu\,dc=ac\,dc=th
authentication-ldap.search_context = dc=psu\,dc=ac\,dc=th
authentication-ldap.email_field = mail
authentication-ldap.surname_field = sn
authentication-ldap.givenname_field = givenName

##### Hierarchical LDAP Settings #####

authentication-ldap.search_scope = 2
authentication-ldap.search.anonymous = false
authentication-ldap.search.user =
CN={username}\,OU=Library\,OU=Hatyai\,OU=Applications\,DC=psu\,DC=ac\,DC=th

authentication-ldap.search.password = {password}
authentication-ldap.netid_email_domain = @psu.ac.th

authentication-ldap.login.groupmap.1 = ou=Staffs:staff_clone
authentication-ldap.login.groupmap.2 = ou=Students:Students

```

/dspace/webapps/jspui/layout/header-default.jsp

3) ใส่ branding banner ที่ส่วนหัวของหน้า

```

<header class="navbar navbar-inverse navbar-fixed-top">
<div class="img-responsive"></div>

```

ปรับความสูงของ header ตามความสูงของ branding banner ข้างต้น และคอมเมนต์ส่วนที่เป็น branding ของ DSpace ออก

```

<main id="content" role="main">
<div class="container banner" style="margin-top: 120px">
<!--div class="container banner">
  <div class="row">
    <div class="col-md-9 brand">
      <h1><fmt:message key="jsp.layout.header-default.brand.heading" /></h1>
      <fmt:message key="jsp.layout.header-default.brand.description" />
    </div>
    <div class="col-md-3">
    </div>
  </div>
</div>
<br-->

```

4) /dspace/webapps/jspui/layout/footer-default.jsp (ปรับการแสดงผล footer ให้เป็นของสถาบัน) เช่น

```
<!-- Page footer -->
```

```

<footer class="navbar navbar-inverse navbar-bottom">
<div id="designedby" class="container text-muted">
  Supported by:
  <a href="https://www.thaihealth.or.th/">
  <a href="https://www.trc.or.th/"></a>
<!--
  <fmt:message key="jsp.layout.footer-default.theme-by"/>
  <a href="http://www.cineca.it"><>
-->
  <div id="footer_feedback" class="pull-right">
    <p class="text-muted">
      <fmt:message key="jsp.layout.footer-default.text"/>&nbsp;-
      <a target="_blank" href="<%= request.getContextPath() %>/feedback"><fmt:message
key="jsp.layout.footer-default.feedback"/></a>
      <a href="<%= request.getContextPath() %>/htmlmap"></a>
    </p>
  </div>
</div>
</footer>

```

ปรับข้อความ jsp.layout.footer-default.text และ jsp.layout.footer-default.feedback ในไฟล์ Messages.properties ตามความเหมาะสม

5) /dspace/webapps/jspui/layout/navbar-default.jsp

ปรับโลโก้ และเมนู(ถ้ามี)

```

<a class="navbar-brand" href="<%= request.getContextPath() %>/"></a>

```

และเมนู(ถ้ามี)

```

<%-- Modified Help -- changed to drop down menu --%>
  <li class="dropdown">
    <a href="#" class="dropdown-toggle" data-toggle="dropdown"><fmt:message
key="jsp.layout.navbar-default.help"/> <b class="caret"></b></a>
    <ul class="dropdown-menu">
      <li class="<%= ( currentPage.endsWith( "/help" ) ? "active" : "" ) %>"><dspace:popup
page="<%= LocaleSupport.getLocalizedMessage(pageContext, "\help.index\") %>"><fmt:message
key="jsp.layout.navbar-admin.help"/></dspace:popup></li>

      <%-- Thesis submission manuals --%>
      <li class="">
<%--
  <script type="text/javascript">
  <!-- Javascript starts here
  document.write('<a href="#" onClick="var popupwin =
window.open('\psukb/help/index.html',\dspacepopup',\height=600,width=550,resizable,scrollbars\');po

```

```

pupwin.focus();return false;">Help</a>');
// --> </script>
--%>
    <a href="#" onclick="var popupwin = window.open('/psukb/help/thesis-submit-EN-
ver6.pdf','dspacepopup','height=600,width=550,resizable,scrollbars');popupwin.focus();return
false;">Thesis Submission Manual(English)</a>
    <noscript><a href="/psukb/help/thesis-submit-EN-ver6.pdf" target="dspacepopup">Submission
Manual (English)</a>
    </noscript></li>

    <li class="">
    <a href="#" onclick="var popupwin = window.open('/psukb/help/thesis-submit-TH-
ver6.pdf','dspacepopup','height=600,width=550,resizable,scrollbars');popupwin.focus();return
false;">Thesis Submission Manual(Thai)</a>
    <noscript><a href="/psukb/help/thesis-submit-TH-ver6.pdf" target="dspacepopup">Submission
Manual (Thai)</a>
    </noscript></li>

    <li class="">
    <a href="#" onclick="var popupwin =
window.open('https://youtu.be/YHMP_HCIbS0','dspacepopup','height=600,width=550,resizable,scrollbar
s');popupwin.focus();return false;">Thesis Submission How-to(Video)</a>
    <noscript><a href="https://youtu.be/YHMP_HCIbS0" target="dspacepopup">Thesis Submission
How-to (Video)</a>
    </noscript></li>

    <li class="">
    <a href="#" onclick="var popupwin =
window.open('/psukb/handle/2016/17435','dspacepopup','height=600,width=550,resizable,scrollbars');p
opupwin.focus();return false;">PSUKB Operation Manual (Thai)</a>
    <noscript><a href="/psukb/handle/2016/17435" target="dspacepopup">PSUKB Operation
Manual (Thai)</a>
    </noscript></li>
</ul>
</li>
<%-- Modified Help -- changed to drop down menu --%>
</ul>

```

6) /dspace/webapps/jspui/home.jsp

เพิ่มสคริปต์ที่จำเป็นกรณีต้องการเชื่อมกับ facebook page (ถ้ามี) ประมาณบรรทัดที่ 47 หลัง <%@ page  
import="org.dspace.services.factory.DSpaceServicesFactory" %>

```

// Java Script SDK for embedding facebook page
out.println("<div id='fb-root'></div>");
out.println("<script async defer crossorigin='anonymous'
src='https://connect.facebook.net/en_US/sdk.js#xfbml=1&version>

```

ปรับการแสดงผลของ Top-news (ไม่แสดง Top News)

```

<dspace:layout locbar="nolink" titlekey="jsp.home.title" feedData="<%= feedData %>">

<!-- <div class="jumbotron">
<%= topNews %>

```

```
</div>
-->
```

ปรับการแสดงผลส่วนของ Side-news ให้แสดงเต็มความกว้าง เพื่อใช้แสดง banner ของระบบ

```
<div class="container">
  <%= sideNews %>
</div>
```

อาจเพิ่มเพิ่มข้อความอื่น ๆ ที่ต้องการ แต่ในที่นี้ไม่ได้เพิ่มข้อความใด ๆ

7) /dSPACE/webapps/jspui/browse/full.jsp

เพิ่มอักษรไทย ณ บรรทัดที่ 280

```
<%
    for (char c = 'ก'; c <= 'ฮ'; c++) {
%>
        <a class="label label-default" href="<%= sharedLink %>&startsWith=<%= c %>"><%= c
%></a>
<%
    }
    for (char c = 'ั'; c <= '็'; c++) {
%>
        <a class="label label-default" href="<%= sharedLink %>&startsWith=<%= c %>"><%= c
%></a>
<%
    }
%><br/>
```

8) /dSPACE/webapps/jspui/browse/single.jsp

เพิ่มอักษรไทย ณ บรรทัดที่ 196

```
<%
    for (char c = 'ก'; c <= 'ฮ'; c++) {
%>
        <a href="<%= sharedLink %>&startsWith=<%= c %>"><%= c %></a>
<%
    }
    for (char c = 'ั'; c <= '็'; c++) {
%>
        <a href="<%= sharedLink %>&startsWith=<%= c %>"><%= c %></a>
<%
    }
%>
<br/>
```

9) /dSPACE/webapps/jspui/static/css/bootstrap/bootstrap-theme.min.css & bootstrap.min.css

ปรับปรุงการแสดงผลสีของส่วนต่าง ๆ เช่น

แก้ไขไฟล์ dSPACE-theme.css เพื่อเอาข้อความส่วนบนออก



เพิ่มตัวแปรนี้ที่บรรทัดสุดท้าย

```
.banner { display: none;}
```

แก้ไขไฟล์ bootstrap-theme.min.css

```
/* เปลี่ยนสีโดยการแทนที่ค่าสีดังนี้ */
```

```
#5cb85c => #3d5eb8
```

```
#249d44 => #1436e1
```

```
/* เปลี่ยนสีตอนคลิกด้วยการแทนที่ตัวแปรนี้ */
```

```
btn-success:active,.btn-success.active{background-color:#3d5bee;border-color:#012bfd}
```

```
/*สีพื้นที่หัวกล่อง Discover */
```

```
#dff0d8 => #b2ccf0
```

```
#d0e9c6 => #99bfff
```

```
/*ปุ่ม*/
```

```
#5bc0de
```

```
#31b0d5
```

เปลี่ยน Discover box ที่ไฟล์ bootstrap.min.css

```
/*เปลี่ยนสีข้อความหัวข้อหลัก */
```

```
#068847 => #062988
```

```
/* เส้นขอบกล่อง */
```

```
background-color:#dff0d8 => #e9e9ea
```

```
border-color:#d6e9c6 => #e9e9ea
```

10) ไฟล์รูปภาพทั้งหมดเก็บภายใต้ /dspace/webapps/jspui/image

11) สร้าง favicon.ico ของตัวเอง (เลือกใช้เครื่องมือที่ถนัด)

12) Messages.properties และ Messages\_th.properties

หลัง compile แล้ว ไฟล์ทั้งสองนี้ จะอยู่ที่ /dspace/webapps/jspui/WEB-INF/class

สำหรับ Messages.properties ซึ่งเป็นภาษาอังกฤษ สามารถเปิดและแก้ไขด้วยโปรแกรม Text editor แต่สำหรับไฟล์ภาษาอื่น ๆ เช่น ภาษาไทย กำหนดให้เก็บเป็น Unicode และตั้งชื่อไฟล์เป็น Messages\_th.properties จึงต้องแปลงเป็น ASCII ก่อน จากนั้นแก้ไขด้วย Text editor หลังจากนั้น ต้องแปลงกลับไปเป็น Unicode จึงจะนำไปใช้ได้

ใน Linux ใช้คำสั่ง native2ascii การแปลงจาก Unicode เป็น ASCII

```
native2ascii -reverse Message_th.properties Message_th.properties.utf
```

และใช้คำสั่งเดียวกันในการแปลงจาก ASCII เป็น Unicode

```
native2ascii -encoding UTF-8 Message_th.properties.utf Message_th.properties
```

13) ปรับปรุงไฟล์ทั้งหมดภายใต้ /dspace/config/email เพื่อให้เป็นของสถาบัน ตัวอย่างเช่น ไฟล์ subscription

```
#  
Subject: DSpace Subscription  
New items are available in the collections you have subscribed to:  
  
{0}  
  
DSpace
```

แก้ไขเป็น

```
#  
Subject: PSUKB Subscription  
New items are available in the collections you have subscribed to:  
  
{0}  
  
PSU Knowledge Bank
```

## 3. การดูแลระบบ

### 3.1 การจัดการอัตโนมัติของระบบ

เพื่อให้ระบบมีความปลอดภัย และการทำงานของระบบมีความสมบูรณ์ จึงต้องมีการตั้งให้ระบบทำงานบางอย่างโดยอัตโนมัติตามวันเวลาที่กำหนด สรุปได้ดังนี้

- (1) ตรวจสอบการบุกรุกโดยผู้ไม่ประสงค์ดีทุก ๆ 5 นาที (/etc/cron.d/add2deny)
- (2) ตั้งให้ระบบ DSpace ทำงานต่าง ๆ ตามคำแนะนำของทีมงาน DSpace (/etc/cron.d/dspace)
- (3) สำรองข้อมูลทุกวัน (/etc/cron.d/backup\_kb)
- (4) แจ้งผู้ดูแลระบบเมื่อมีการ Restart แม่ข่าย (/etc/cron.d/restarted)

### 3.2 การสำรองข้อมูล

เนื่องจากระบบคลังปัญญา มีข้อมูลจำนวนมาก สำนักนวัตกรรมการศึกษา จึงขอให้สำนักทรัพยากรการเรียนรู้คุณหญิงหลงฯ ทำหน้าที่สำรองข้อมูลเพื่อความปลอดภัยของข้อมูลด้วย จึงได้ออกแบบให้มีการสำรองข้อมูลไว้ในเครื่องคอมพิวเตอร์ของหน่วยงาน

ข้อมูลที่ต้องสำรอง แบ่งออกเป็น 3 ส่วน คือ

- (1) **ต้นฉบับโปรแกรม** อยู่ที่ /build/dspace-6.3-src-release

ข้อมูลส่วนนี้ คือ ต้นฉบับของโปรแกรม DSpace ซึ่งเป็นข้อมูลที่เปลี่ยนแปลงน้อย ควรสำรองทุกครั้งที่มีการเปลี่ยนแปลง และสามารถสำรองด้วยวิธีการถ่ายโอนข้อมูล หากคอมพิวเตอร์ที่ใช้สำรองใช้ระบบปฏิบัติการลินุกซ์เหมือนกัน ก็สามารถใช้คำสั่งนี้ สำรองข้อมูลมาได้ (ให้ระบุ username สำหรับ login เข้าแม่ข่าย และระบุ directory ที่ใช้เก็บข้อมูลสำรอง

```
rsync -av [username]@kb.psu.ac.th:/build/dspace-6.3-src-release [ระบุ directory ที่ใช้เก็บข้อมูลสำรอง]
```

- (2) **ไฟล์ข้อมูล ไฟล์ log และโปรแกรมที่ compile แล้ว** อยู่ที่ /dspace

ข้อมูลส่วนนี้ ประกอบด้วยไฟล์ที่ประกอบข้อมูลแต่ละรายการ(อยู่ใน /dspace/assetstore) ข้อมูลการใช้งานต่าง ๆ (/dspace/log) และส่วนของโปรแกรมต่าง ๆ และเป็นส่วนที่ใช้พื้นที่จัดเก็บมากที่สุด ข้อมูลส่วนนี้มักมีการเปลี่ยนแปลงตลอดเวลา แต่ก็ไม่จำเป็นต้องสำรองถี่มาก ปัจจุบันตั้งให้มีการสำรองวันละ 1 ครั้ง ตามที่กำหนดใน /etc/cron.d/backup\_kb (ซึ่งจะสำรองส่วนที่ 3 ไปพร้อมกันด้วย)

- (3) **ฐานข้อมูลใน PostgreSQL** โดยทั่วไปเรียกว่า back-end ข้อมูลดิบจัดเก็บในพื้นที่ที่กำหนดโดย PostgreSQL

มีการสำรองข้อมูลนี้วันละ 1 ครั้ง โดยต้องกำหนดให้สำรองข้อมูลในระยะเวลาใกล้เคียงกับการสำรองข้อมูล ส่วนที่ 2 เพราะข้อมูลทั้งส่วนที่ 2 และส่วนที่ 3 ควรเป็นข้อมูล ณ วันเวลาเดียวกัน

การสำรองข้อมูลของระบบคลังปัญญา มีขั้นตอนโดยสรุป ดังนี้

- 1) เครื่องแม่ข่าย kb.psu.ac.th สำรองข้อมูลส่วนที่ 3 (ฐานข้อมูล kb ใน PostgreSQL) ณ เวลา 02:00 น. ตามที่กำหนดใน /etc/cron.d/backup\_kbdb

```
SHELL=/bin/sh
MAILTO=panyarak.n@psu.ac.th

0 2 * * * tomcat /bin/sh /root/scripts/backup_kbdb.sh
```

โดยไฟล์ /root/scripts/backup\_kbdb.sh เป็น Shell script ที่มีเนื้อหา ดังนี้

```
#!/bin/bash

day=$(LC_ALL=us_US.utf8 date "+%a")
log_file=/build/log/backup.log

DATE=`date`
echo ${DATE} "Start dumping Database kb." >>$log_file
/usr/bin/pg_dump kb > /build/postgres/kb.sql
DATE=`date`
echo ${DATE} "Database kb dumped." >>$log_file
```

Script ข้างต้น ทำหน้าที่เพียง dump ฐานข้อมูล kb ไปเก็บไว้ที่ /build/postgres/kb.sql

- 2) เครื่องแม่ข่าย kb2tmp.psu.ac.th สำรองข้อมูลส่วนที่ 2 และส่วนที่ 3 ณ เวลา 02:01 น. ตามที่กำหนดใน /etc/cron.d/backup\_kb

```
SHELL=/bin/sh
#MAILTO=panyarak@gmail.com, sompong.h@psu.ac.th
MAILTO=panyarak@gmail.com

1 2 * * * tomcat /bin/sh /backup/cron/backup_kb.sh
```

โดยไฟล์ /backup/cron/backup\_kb.sh เป็น Shell script ที่มีเนื้อหา ดังนี้

```
#!/bin/bash

day=$(LC_ALL=us_US.utf8 date "+%w")
dom=$(date +%d)

## Determine day of week. 0 is Sunday.
case $day in
  [0,1,2]*)
```

```

    BACKUP='/backup/backup1'
    ;;
[3,4,5,6]*)
    BACKUP='/backup/backup2'
    ;;
esac

## Our log file
log_file=/backup/cron/backup.log

DATE=`date`
echo ${DATE} "Start transferring files from kb.psu.ac.th to ${BACKUP}." >>$log_file

rsync -e 'ssh -p 22' -avzp --delete kb@kb.psu.ac.th:/dspace ${BACKUP}
rsync -e 'ssh -p 22' -avzp --delete kb@kb.psu.ac.th:/build/postgres ${BACKUP}

DATE=`date`
echo ${DATE} "Finished transferring files from kb.psu.ac.th to ${BACKUP}" >>$log_file

## If Sunday then also transfer files to 192.168.27.191
case $dom in
    25)
        rsync -e 'ssh -p 22' -avzp --delete ${BACKUP}/dspace tomcat@192.168.27.191:/home/${BACKUP}/
        rsync -e 'ssh -p 22' -avzp --delete ${BACKUP}/postgres tomcat@192.168.27.191:/home/${BACKUP}/
        DATE=`date`
        echo ${DATE} "Finished transferring files from ${BACKUP} to Superblack" >>$log_file
    ;;
*) ;;
esac

```

Script /backup/cron/backup\_kb.sh ทำหน้าที่ ดังนี้

- วันอาทิตย์-วันอังคารของสัปดาห์ สำรองข้อมูลส่วนที่ 2 และ 3 เก็บไว้ที่ /backup/backup1
- วันพุธ-วันเสาร์ของสัปดาห์ สำรองข้อมูลส่วนที่ 2 และ 3 เก็บไว้ที่ /backup/backup2
- หากตรงกับวันที่ 25 ของทุกเดือน ก็จะสำรองข้อมูลส่วนที่ 2 และ 3 ไปเก็บไว้ที่เครื่องคอมพิวเตอร์ 192.168.27.191 ด้วย

เนื่องจากการสำรองข้อมูลดังกล่าว เป็นกระบวนการอัตโนมัติด้วย username: tomcat จึงต้องมีการดำเนินการเพื่อให้ tomcat ใน kb2tmp.psu.ac.th สามารถ login เข้า kb.psu.ac.th และ 192.168.27.191 ได้โดยไม่ต้องป้อนรหัสผ่าน และมีขั้นตอน ดังนี้

- (1) ล็อคอินเข้าเครื่องแม่ข่าย kb2tmp.psu.ac.th แล้วสร้าง key สำหรับนำไปใส่ในแม่ข่าย kb.psu.ac.th และ 192.168.27.191 ดังนี้

```
sudo -u tomcat ssh-keygen
```

โปรแกรมจะแสดงผล ดังนี้

```
[sudo] password for panya: <--- ชื่อ user จะปรากฏตามชื่อผู้ใช้ ตรงนี้ ให้ป้อนรหัสผ่านของตัวเอง
Generating public/private rsa key pair.
Enter file in which to save the key (/opt/tomcat/.ssh/id_rsa): <--- กด Enter
Enter passphrase (empty for no passphrase): <--- กด Enter (ไม่มีรหัสผ่าน) ครั้งที่ 1
Enter same passphrase again: <--- กด Enter (ไม่มีรหัสผ่าน) ครั้งที่ 2
```

จะได้ไฟล์ id\_rsa.pub ใน directory .ssh ณ Home ของ tomcat ซึ่งในที่นี้ คือ /opt/tomcat/.ssh

- (2) ถ่ายโอนไฟล์ id\_rsa.pub ไปที่เครื่อง kb.psu.ac.th และ 192.168.27.191 ด้วยคำสั่ง ดังนี้

```
sudo su tomcat <--- เปลี่ยนเป็น user: tomcat
cd <--- เปลี่ยนไดเรกทอรีปัจจุบันไปที่ home ของ tomcat
ssh-copy-id -i .ssh/id_rsa.pub tomcat@kb.psu.ac.th
```

ใช้คำสั่ง ssh-copy-id -i .ssh/id\_rsa.pub tomcat@192.168.27.191 อีกครั้งเพื่อถ่ายโอนไฟล์ key ไปที่เครื่อง 192.168.27.191

- (3) จากนั้นให้ login เข้าเครื่อง kb.psu.ac.th และ 192.168.27.191 หากเป็นการ login ครั้งแรก อาจจะได้รับข้อความที่ต้องโต้ตอบ ดังตัวอย่างต่อไปนี้

```
ssh kb.psu.ac.th
The authenticity of host 'kb.psu.ac.th (192.168.105.101)' can't be established.
ECDSA key fingerprint is SHA256:mREEDbKmK2GD1WEIzjcLAVIQVenA65xcJvpHyJuw/4M.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'scholar,192.168.105.101' (ECDSA) to the list of known hosts.
```

### 3.3 การนำข้อมูลสำรองกลับมาใช้งาน

ในกรณีที่มีความจำเป็นต้องนำข้อมูลที่สำรองไว้มาใช้งาน ให้มั่นใจว่า ข้อมูลในส่วนที่ (2) และส่วนที่ (3) ในหัวข้อที่แล้ว เป็นชุดข้อมูลในระยะเวลาเดียวกัน เพราะหากเป็นข้อมูลคนละวันเวลา แม้ว่า อาจจะสามารถใช้ได้ แต่ข้อมูลบางส่วนอาจจะเกิดความผิดพลาดได้

เพื่อความมั่นใจ ควรจัดให้มีแม่ข่ายสำรองอีก 1 ตัวเพื่อทดลองนำข้อมูลสำรองไปทดสอบการใช้งานก่อน จะได้ตรวจสอบเบื้องต้นว่า ข้อมูลชุดดังกล่าว ใช้งานได้หรือไม่อย่างไร ในที่นี้ขอยกตัวอย่างโดยใช้เครื่องแม่ข่ายสำรอง kb2tmp.psu.ac.th โดยได้ตั้งให้แม่ข่ายสำรองนี้ นำข้อมูลสำรองกลับมาใช้งานโดยอัตโนมัติทุกวัน

ในขั้นแรก ต้องติดตั้งโปรแกรมให้เรียบร้อยตามที่กล่าวไว้ใน 2.1 -2.4 แต่สิ่งที่แตกต่าง คือ ชื่อของแม่ข่าย ในที่นี้ก็คือ kb2tmp.psu.ac.th

ในเครื่องแม่ข่ายสำรอง kb2tmp.psu.ac.th มีข้อมูลที่สำรองไว้ทุกวันเก็บไว้ที่ /backup/backup1 (ข้อมูลทุกวันอาทิตย์-วันอังคาร) และ /backup/backup2 (ข้อมูลทุกวันพุธ-วันเสาร์) ให้ดำเนินการดังนี้

- (1) เขียน shell script เพื่อทำหน้าที่สำเนาไฟล์ข้อมูลต่าง ๆ จากที่สำรองไว้ไปที่ /dspace ตั้งชื่อไฟล์เป็น backupfromkb2tmp.sh เก็บไว้ที่ /opt/tomcat/bin โดยให้มีเนื้อหา ดังนี้

```
#!/bin/bash

day=$(LC_ALL=us_US.utf8 date "+%w")

## Determine day of week. 0 is Sunday.
case $day in
[0,1,2]*)
BACKUP='/backup/backup1'
;;
[3,4,5,6]*)
BACKUP='/backup/backup2'
;;
esac

## Our log file
log_file=/build/backup/backup2.log
DATE=`date`

error_handling()
{
echo "$1" >>$log_file
}

echo ${DATE} "Updating files from ${BACKUP}." >>$log_file

rsync -av --delete ${BACKUP}/dspace/assetstore /dspace || error_handling "***** There are errors during tranferring files! *****"
rsync -av --delete ${BACKUP}/dspace/log /dspace || error_handling "***** There are errors during tranferring files! *****"
rsync -av --delete ${BACKUP}/dspace/reports /dspace || error_handling "***** There are errors during tranferring files! *****"
rsync -av --delete ${BACKUP}/dspace/solr /dspace || error_handling "***** There are errors during tranferring files! *****"
rsync -av --delete ${BACKUP}/dspace/sitemaps /dspace || error_handling "***** There are errors during tranferring files! *****"
rsync -av --delete ${BACKUP}/dspace/imports /dspace || error_handling "***** There are errors during tranferring files! *****"
rsync -av --delete ${BACKUP}/dspace/exports /dspace || error_handling "***** There are errors during tranferring files! *****"
rsync -av --delete ${BACKUP}/dspace/upload /dspace || error_handling "***** There are errors during tranferring files! *****"
rsync -av --delete ${BACKUP}/postgres /build || error_handling "***** There are errors during tranferring files! *****"
```

```
DATE=`date`  
echo ${DATE} "Finished updating files from ${BACKUP}." >>$log_file
```

ขั้นตอนโดยสรุปก็คือ โปรแกรมจะตรวจสอบว่า เป็นวันอะไรของสัปดาห์ กำหนดไฟล์สำหรับเก็บ log หลังจากนั้นก็นำไฟล์จากต้นทาง(/backup/backup1 หรือ /backup/backup2) ไปยังปลายทาง /dspace และนำไฟล์ฐานข้อมูลไปไว้ที่ /build/postgres

อย่าลืมสร้าง sub-directories /build/backup และ /build/postgres โดยกำหนดให้ tomcat เป็นเจ้าของพื้นที่ดังกล่าวด้วย

(2) เขียน shell script ชื่อ restore\_kb.sh และเก็บไว้ที่ /opt/tomcat/bin ให้มีเนื้อหาดังนี้

```
#!/bin/bash  
  
## Our log file  
log_file=/build/backup/backup.log  
  
error_handling()  
{  
  echo "$1" >>$log_file  
}  
  
DATE=`date`  
echo ${DATE} "Stop TOMCAT." >>$log_file  
sudo service tomcat stop || error_handling "***** There are errors! *****"  
  
## Updating data from kb2tmp  
sudo -iu tomcat sh /opt/tomcat/bin/backupfromkb2tmp.sh  
  
DATE=`date`  
echo ${DATE} "Delete database kb." >>$log_file  
sudo -iu tomcat dropdb kb || error_handling "***** There are errors! *****"  
  
DATE=`date`  
echo ${DATE} "Create database kb." >>$log_file  
sudo -iu tomcat createdb -U tomcat -E UNICODE kb || error_handling "***** There are errors!  
*****"  
  
DATE=`date`  
echo ${DATE} "Create extension pgcrypto for database kb." >>$log_file  
sudo -iu postgres psql kb -c "CREATE EXTENSION pgcrypto;" || error_handling "***** There are  
errors! *****"  
DATE=`date`  
echo ${DATE} "Restore data to database kb." >>$log_file  
sudo -iu postgres psql kb < /build/postgres/kb.sql || error_handling "***** There are errors! *****"  
DATE=`date`  
echo ${DATE} "Change owner of database kb." >>$log_file  
sudo -iu postgres psql kb -c "ALTER DATABASE kb OWNER TO tomcat;" || error_handling "*****  
There are errors! *****"  
  
DATE=`date`
```



```

echo ${DATE} "Start TOMCAT." >>$log_file
sudo service tomcat start || error_handling "***** There are errors! *****"

DATE=`date`
echo ${DATE} "Reindex database." >>$log_file
sudo /dSPACE/bin/dSPACE index-discovery -b || error_handling "***** There are errors! *****"

DATE=`date`
echo ${DATE} "Optimize database index." >>$log_file
sudo /dSPACE/bin/dSPACE index-discovery -o || error_handling "***** There are errors! *****"

DATE=`date`
echo ${DATE} "Done optimizing database index." >>$log_file

```

Shell script ข้างต้นทำงานโดยการหยุดการทำงานของ Tomcat เรียกโปรแกรม backupfromkb2tmp.sh หลังจากก็ลบฐานข้อมูล kb (เพื่อเคลียร์ข้อมูลทั้งหมดในฐานข้อมูล) สร้างฐานข้อมูล kb ใหม่ นำเข้าข้อมูลจาก /build/postgres/kb.sql กำหนดให้ tomcat เป็นเจ้าของฐานข้อมูล kb และ re-index ฐานข้อมูลใหม่ จบด้วยการ start service Tomcat

- (3) ตั้งให้โปรแกรมทำงานอัตโนมัติทุกวัน เวลา 05:35 น. โดยการสร้างไฟล์ /etc/cron.d/update\_kb โดยมีเนื้อหา ดังนี้ (เวลาที่กำหนด ให้เป็นเวลาหลังการสำรองข้อมูลจากแม่ข่าย kb.psu.ac.th เสร็จสิ้นแล้ว)

```

SHELL=/bin/sh
MAILTO=panyarak.n@psu.ac.th

35 5 * * * root /bin/sh /opt/tomcat/bin/restore_kb.sh

```

### 3.4 การจัดการระบบด้วยวิธี Manual

#### 3.4.1 การจัดการชื่อและนามสกุล

#### 3.4.2 การดึงรายการข้อมูลในระบบ

การดึงข้อมูลรายการต่าง ๆ ในแต่ละ Community หรือในแต่ละ Collection สามารถทำได้โดยใช้ REST ใน DSpace สรุปเป็นขั้นตอน ดังนี้

- (1) ติดตั้งเครื่องมือที่จำเป็น เช่น json2csv, xmllint ด้วยคำสั่งดังนี้

```

sudo apt install libxml2-utils npm
sudo npm install -g json2csv

```

- (2) ดึง UUID ของ Communities

ดึงข้อมูลของ Communities แสดงผลในรูปแบบ json

```

curl -s -H "Accept: application/json" http://localhost:8080/rest/communities

```

ดึงข้อมูลของ Communities แสดงผลในรูปแบบ csv

```
curl -s -H "Accept: application/json" http://localhost:8080/rest/communities | json2csv
```

ตัดเอาเฉพาะข้อมูล uuid และ name

```
curl -s -H "Accept: application/json" http://localhost:8080/rest/communities | json2csv | cut -f1-2 -d","
```

บันทึกข้อมูลที่ได้ในไฟล์ communities.uuid

```
curl -s -H "Accept: application/json" http://localhost:8080/rest/communities | json2csv | cut -f1-2 -d"," > communities.uuid
```

(3) คำสั่งตามข้อ (2) สามารถดึงได้เพียงครั้งละ 100 รายการเท่านั้น หากต้องการดึงข้อมูลมากกว่านั้น ให้เพิ่มตัวแปร offset เข้าไปในคำสั่ง เช่น หากต้องการดึงรายการที่ 101-200 ให้ใช้คำสั่ง ดังนี้

```
curl -s -H "Accept: application/json" http://localhost:8080/rest/communities?offset=1
```

(4) ดึง uuid และ name ของ Collections ในแต่ละ Community และรายการในแต่ละ Collection

ใช้ shell script ดังนี้

```
curl -s -H "Accept: application/json" http://localhost:8080/rest/communities | json2csv | cut -f1-2 -d"," > communities.uuid
sed -i '/uuid/d' communities.uuid

while read COMM
do
  uuid=$(echo $COMM | cut -f1 -d",")
  name=$(echo $COMM | cut -f2 -d",")
  # echo $uuid
  curl -s -H "Accept: application/xml" http://localhost:8080/rest/communities/${uuid}/collections | xmllint --format - >${name}.xml
  curl -s -H "Accept: application/json" http://localhost:8080/rest/communities/${uuid}/collections >${name}.json
  cat "${name}.json" | /usr/bin/json2csv >${name}.csv
  cut -f1-2 -d"," "${name}.csv" >${uuid}.uuid
  sed -i '/uuid/d' "${uuid}.uuid"
  mkdir "${name}"
  cd "${name}"
  while read COLL
  do
    coll_id=$(echo $COLL | cut -f1 -d",")
    coll_name=$(echo $COLL | cut -f2 -d",")
    curl -s -H "Accept: application/xml" http://localhost:8080/rest/collections/${coll_id}/items | xmllint --format - >${coll_name}.xml
    curl -s -H "Accept: application/json" http://localhost:8080/rest/collections/${coll_id}/items >${coll_name}.json
    cat "${coll_name}.json" | json2csv >${coll_name}.csv
    cut -f1-2 -d"," "${coll_name}.csv" >${coll_name}.uuid
    sed -i '/uuid/d' "${coll_name}.uuid"
  done
done
```

```
done < ../${uuid}.uuid  
cd ..  
done < communities.uuid
```

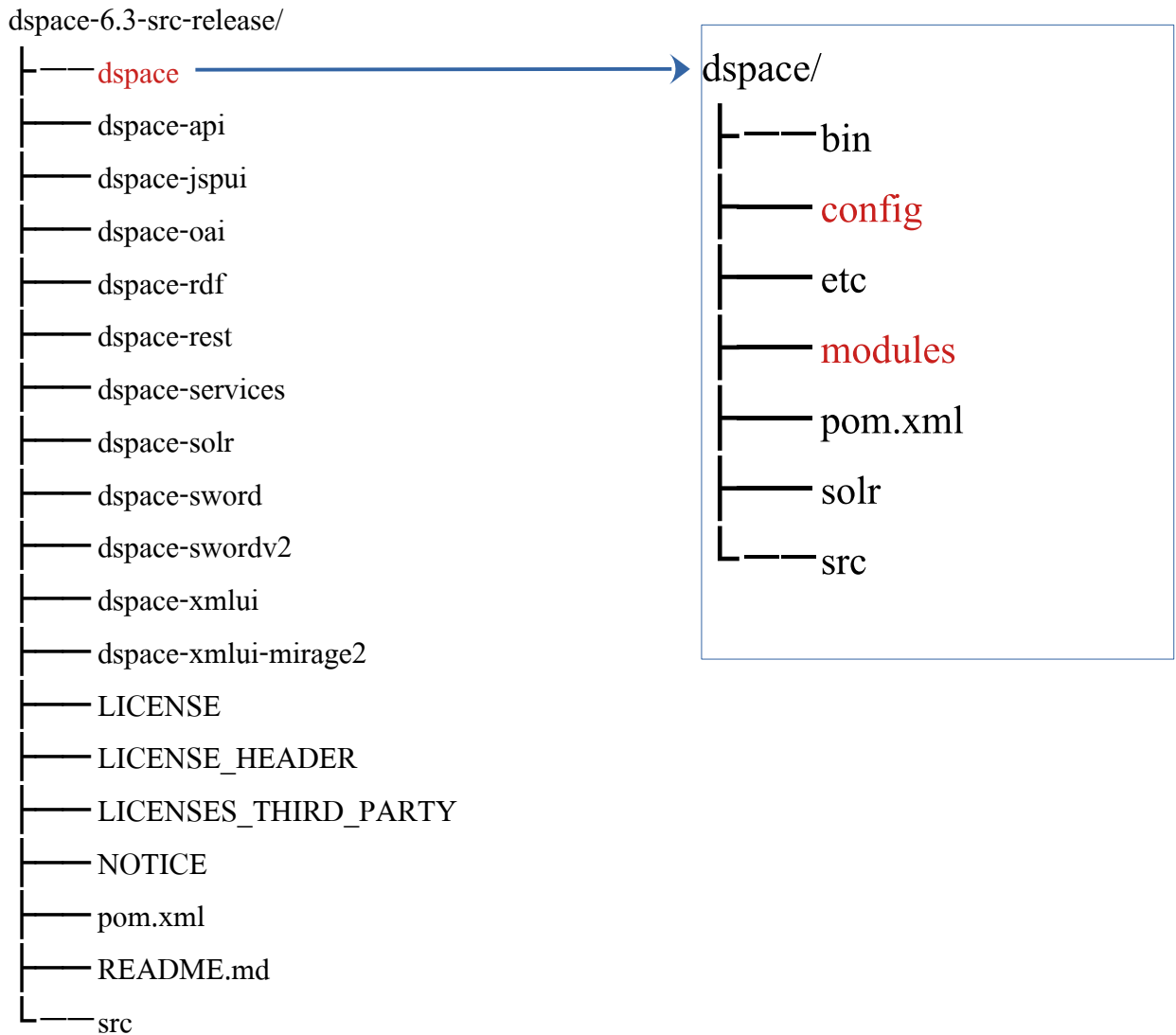
เมื่อสั่ง shell script ข้างต้นทำงาน จะได้ไฟล์ข้อมูลในรูปแบบ XML, JSON และ CSV โดยไฟล์ CSV สามารถเปิดได้ด้วยโปรแกรม Spreadsheet

นอกจากนั้น ยังสามารถดึงข้อมูลผ่าน Browser ได้เช่นกัน โดยการป้อน URL ใน Browser แต่ข้อมูลที่ได้จะอยู่รูปแบบ xml เท่านั้น ตัวอย่างเช่น

```
https://kb.psu.ac.th/rest/communities  
https://kb.psu.ac.th/rest/communities?offset=100
```

## 4. ภาคผนวก

### 4.1 DSpace Source Directories: /build/dspace-6.3-src-release



## 4.2 DSpace Web Applications

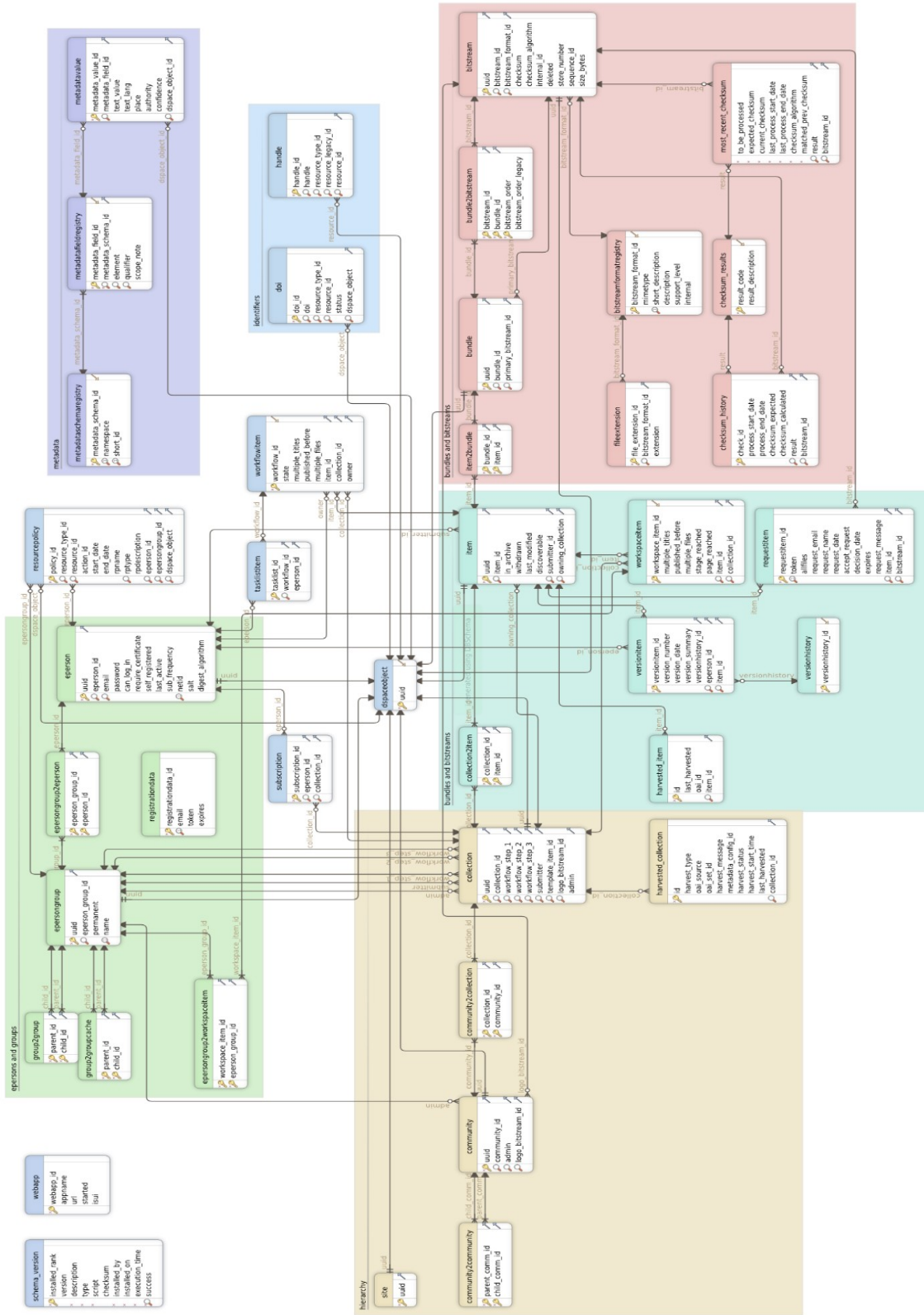
/dSPACE

- assetstore
- bin
- config
- ctqueues
- etc
- exports
- handle-server
- imports
- lib
- log
- reports
- search
- sitemaps
- solr
- triplestore
- upload
- var
- webapps

/dSPACE/webapps/

- .jspui
- oai
- rdf
- rest
- solr
- sword
- swordv2
- xmlui

### 4.3 Dspace Database Structure



#### 4.4 ไฟล์ /etc/cron.d/restarted

สำหรับให้ระบบแจ้งทางอีเมล กรณีที่มีการ reboot เครื่องแม่ข่าย

```
SHELL=/bin/sh
@reboot root /bin/sh /root/scripts/system_restarted.sh
```

#### 4.5 ไฟล์ /root/scripts/system\_restarted.sh

```
#!/bin/bash
echo `hostname`" has restarted at ""LC_ALL=us_US.utf8 date '+%H:%M:%S %a %d %b %Y'|
/usr/bin/mail -s "PSU Knowledge Bank Restarted" panyarak.n@psu.ac.th
```

#### 4.6 ไฟล์ /etc/cron.d/add2deny

```
SHELL=/bin/sh
#MAILTO=panya
*/5 * * * * root /bin/sh /root/scripts/denyhosts.sh
```

#### 4.7 ไฟล์ /root/scripts/denyhosts.sh

```
#!/bin/bash

log_file=/root/tmp/list.tmp
log_file2=/root/tmp/list.deny
/bin/rm $log_file $log_file2

grep sshd /var/log/auth.log /var/log/auth.log.1 |grep Failed|grep -v invalid|grep " " |cut -d" " -f12
>$log_file
grep sshd /var/log/auth.log /var/log/auth.log.1 |grep Failed|grep -v invalid|grep -v " " |cut -d" " -f11
>>$log_file
grep sshd /var/log/auth.log /var/log/auth.log.1 |grep Failed|grep invalid|grep -v " " |cut -d" " -f13
>>$log_file
grep sshd /var/log/auth.log /var/log/auth.log.1 |grep Failed|grep invalid|grep " " |cut -d" " -f14
>>$log_file
grep sshd /var/log/auth.log*|grep rhost|cut -d=' ' -f7|cut -f1 -d" "|sort|uniq|grep -v 192.168.26 >>$log_file

cat $log_file|sort|uniq|grep -v 192.168.26 |grep -v 192.168.27 |grep -v Failed|grep -v port|grep '[0-9]\.'|
awk '{print "ALL: ",$1}' >$log_file2

cat /root/scripts/hosts.deny $log_file2 >/etc/hosts.deny
```