การแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาบนพื้นฐานของพีระมิดข้อมูลแบบหลายระดับความละเอียดและหลายระดับเวลา

นายจตุรงค์ ลีลาวัฒนพาณิชย์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2555

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย
บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ที่ส่งผ่านทางบัณฑิตวิทยาลัย
The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository(CUIR)
are the thesis authors' files submitted through the Graduate School.

VISUALIZATION OF TEMPORAL GEOGRAPHIC DATA BASED ON MULTI-RESOLUTION DATA PYRAMIDS AND A MULTI-LEVEL TIME-SCALE

Mr. Jaturong Leelawattanapanit

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Engineering Program in Computer Engineering

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2012

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาบนพื้นฐานของพีระมิดข้อมูลแบบหลายระดับความละเอียดและหลายระดับเวลา
โดย	นายจตุรงค์ ลีลาวัฒนพาณิชย์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร.วีระ เหมืองสิน

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ
(รองศาสตราจารย์ ดร.ทวีตีย์ เสนีวงศ์ ณ อยุธยา)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ ดร.วีระ เหมืองสิน)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.เกริก ภิรมย์โสภา)

.....กรรมการภายนอกมหาวิทยาลัย
(ดร.ภัทรชนก ศรีวิหค)

จตุรงค์ ลีลาวัฒนพาณิชย์ : การแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาบนพื้นฐานของพีระมิดข้อมูลแบบหลายระดับความละเอียดและหลายระดับเวลา. (VISUALIZATION OF TEMPORAL GEOGRAPHIC DATA BASED ON MULTI-RESOLUTION DATA PYRAMIDS AND A MULTI-LEVEL TIME-SCALE) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร.วีระ เหมืองสิน, 118 หน้า.

งานวิจัยนี้เน้นที่ปัญหาของการขาดเครื่องมือที่สนับสนุนการแสดงข้อมูลทางภูมิศาสตร์ในเชิงเวลาบนแผนที่บนเว็บแอพพลิเคชัน ในขณะที่ข้อมูลเชิงพื้นที่สามารถสร้างได้เป็นพีระมิดข้อมูลและสามารถเลือกแสดงข้อมูลได้โดยใช้ตัวควบคุมการซูมบนแผนที่บนเว็บแอพพลิเคชัน ซึ่งคุณสมบัติด้านเวลาของข้อมูลอาจจะได้รับการสนับสนุนด้วยส่วนควบคุมของเส้นเวลาพื้นฐานโดยไม่มีความสามารถในการคัดเลือกข้อมูลมาแสดงผล ดังนั้นถ้าข้อมูลมีปริมาณมาก ๆ บนช่วงเวลาที่ยาว จะทำให้ไม่สามารถแสดงข้อมูลให้เป็นที่เข้าใจของผู้ใช้งานได้ ซึ่งงานวิจัยนี้ได้เสนอวิธีการแก้ปัญหาดังกล่าวโดยการประยุกต์ใช้เทคนิคหลายระดับความละเอียดของการแสดงข้อมูลเชิงพื้นที่มาใช้กับการแสดงข้อมูลเชิงเวลา กล่าวคือจะสร้างจะสร้างพีระมิดข้อมูลตามระดับความละเอียดของเวลาแทนระดับความละเอียดและขนาดขยายของพื้นที่และซอฟต์แวร์สำหรับดูแผนที่ที่มีส่วนควบคุมเส้นเวลาที่สามารถเปลี่ยนระดับความละเอียดของเวลาได้ โดยระบบต้นแบบจะพัฒนาเป็นแผนที่บนเว็บแอพพลิเคชันที่มีส่วนควบคุมเส้นเวลาพิเศษที่ใช้ปฏิสัมพันธ์กับฐานข้อมูลที่ประกอบด้วยพีระมิดข้อมูลของข้อมูลเชิงพื้นที่และเวลา

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....

สาขาวิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก .....

ปีการศึกษา 2555

5470137621 : MAJOR COMPUTER ENGINEERING

KEYWORDS : VISUALIZATION / TIMELINE / TEMPORAL GEOGRAPHIC DATA / MULTI-LEVEL TIME-SCALE

JATURONG LEELAWATTANAPANIT : VISUALIZATION OF TEMPORAL GEOGRAPHIC DATA BASED ON MULTI-RESOLUTION DATA PYRAMIDS AND A MULTI-LEVEL TIME-SCALE. ADVISOR : ASST. PROF. VEERA MUANGSIN, Ph.D., 118 pp.

This work addresses the problem on lack of supported tool for visualizing temporal geographic data on web mapping applications. While spatial data can be structured as a data pyramid and selectively visualized using the zoom control on a web map viewer, temporal property of the data may be supported with a basic timeline control without data selection capability. Therefore, a large amount of dynamic data over a long period of time can clutter the screen. The proposed solution is to apply the multi-resolution technique for visualizing spatial data onto temporal data. That is to create a data pyramid based on the time-scale instead of the altitude-level and map-viewer software with a timeline control interface that is zoom-able to different time-scales. A prototype system is implemented as a web mapping application with a specialized timeline control that interacts with the database containing the spatial-temporal data pyramid.

Department : Computer Engineering Student's Signature .....

Field of Study : Computer Engineering Advisor's Signature .....

Academic Year : 2012

กิตติกรรมประกาศ

งานวิจัยในวิทยานิพนธ์ฉบับนี้จะไม่มีทางสำเร็จลุล่วงไปได้ด้วยความสามารถของข้าพเจ้าเพียงคนเดียว หากไม่ได้รับการสนับสนุนจากผู้มีพระคุณหลายท่านที่กรุณาช่วยเหลือข้าพเจ้ามาโดยตลอดระยะเวลาการศึกษานี้ แม้เต็มไปด้วยความยากลำบากและอุปสรรคนานัปการ แต่ก็เป็นประสบการณ์อันมีค่ายิ่งที่สั่งสอนบ่มเพาะฝึกฝนข้าพเจ้า ให้รู้จักการเรียนรู้ด้วยตนเอง การคิดอย่างเป็นเหตุเป็นผล ให้มีความอดทน มีความรับผิดชอบและขยันหมั่นเพียร

ขอขอบพระคุณอาจารย์ที่ปรึกษาวิทยานิพนธ์ฉบับนี้ ผู้ช่วยศาสตราจารย์ ดร.วีระ เหมืองสิน ที่กรุณาสละเวลาอันมีค่าช่วยชี้แนะแนวทางในการทำวิจัยและอบรมแก้ไขศิษย์คนนี้ด้วยดีเสมอมา

ขอขอบพระคุณคณะกรรมการสอบวิทยานิพนธ์ฉบับนี้ ซึ่งประกอบไปด้วย รองศาสตราจารย์ ดร.ทวิตีย์ เสนีวงศ์ ณ อยุธยา, ผู้ช่วยศาสตราจารย์ ดร.เกริก ภิรมย์โสภา และ ดร.ภัทรชนก ศรีวิหค ที่ให้แนวทางแก้ไขและข้อคิดในงานวิจัยที่ยังมีจุดบกพร่อง เพื่อพัฒนาแก้ไขให้ดียิ่ง ๆ ขึ้นไป

ขอขอบพระคุณพี่วนพงศ์ แก้วสิงห์ ซึ่งเป็นรุ่นพี่ที่ห้องปฏิบัติการที่ให้ความรู้และคำปรึกษาในการทำวิจัยตลอดระยะเวลาการศึกษา

ขอขอบคุณเพื่อน ๆ ทั้งในและนอกห้องปฏิบัติการทุกคนที่คอยช่วยเหลือซึ่งกันและกันในทุก ๆ ด้าน และร่วมทุกข์ร่วมสุขด้วยกันมาตลอดระยะเวลาการศึกษา

สุดท้ายนี้ ขอขอบพระคุณบิดามารดา น้องสาว และน้าของข้าพเจ้าที่เป็นกำลังใจให้ในช่วงเวลาที่ยากลำบากเสมอมา

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง
บทคัดย่อภาษาอังกฤษ ..... จ
กิตติกรรมประกาศ ..... ฉ
สารบัญ ..... ช
สารบัญตาราง ..... ญ
สารบัญภาพ ..... ฏ
บทที่ 1 บทนำ 1
1.1 ความเป็นมาและความสำคัญของปัญหา 1
1.2 วัตถุประสงค์ 3
1.3 ขอบเขตของการวิจัย 3
1.4 ประโยชน์ที่จะได้รับ 3
1.5 วิธีการดำเนินการวิจัย 4
1.6 การเผยแพร่ผลงานวิจัย 4
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 5
2.1 ทฤษฎีที่เกี่ยวข้อง 5
2.1.1 Geographic Information System 5
2.1.2 ข้อมูลเชิงพื้นที่ 6
2.1.2.1 โครงสร้างข้อมูลแบบราสเตอร์ 6
2.1.2.2 โครงสร้างข้อมูลแบบเวคเตอร์ 7
2.1.3 ข้อมูลเชิงเวลา 9
2.1.4 เส้นเวลา 9
2.1.5 Web Mapping 10
2.1.6 GeoServer [17] 11
2.1.6.1 การทำงานของ GeoServer 12
2.1.6.2 การนำชั้นข้อมูลจาก GeoServer ไปใช้งาน 12
2.1.6.3 การติดตั้ง GeoServer 13
2.2 งานวิจัยที่เกี่ยวข้อง 13

หน้า

2.2.1 Google Earth [3, 4] 13
2.2.2 Challenges in Interactive Time-Based Information Visualization [7] 15
2.2.3 The TimeMap Project: Developing Time-Based GIS Display for Cultural Data [6] 15
2.2.4 Continuum: designing timelines for hierarchies, relationships and scale [1] 16
2.2.5 myHistro [5] 18
2.2.6 เครื่องมือสำหรับแสดงข้อมูลในเชิงเวลาที่มีอยู่ในปัจจุบัน 19
2.2.6.1 SIMILE Timeline 19
2.2.6.2 ProPublica's Timeline Setter 21
2.2.6.3 Timeglider 21
บทที่ 3 แนวคิดและการออกแบบระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา 23
3.1 ประเภทของข้อมูล 23
3.2 แนวคิดในการแสดงข้อมูลหลายระดับความละเอียดในเชิงเวลา 24
3.3 แนวคิดในการแสดงข้อมูลหลายระดับความละเอียดในเชิงพื้นที่และเวลา 28
3.4 แนวคิดในการออกแบบ Timeline 29
3.5 การออกแบบระบบต้นแบบ 30
3.5.1 ส่วนของ Server 30
3.5.1.1 การเก็บข้อมูลลงฐานข้อมูลเชิงพื้นที่ 31
3.5.1.2 การสร้างพีระมิดข้อมูลเชิงเวลา 31
3.5.1.3 การสร้างชั้นข้อมูลใน GeoServer 39
3.5.2 ส่วนของ Client 40
บทที่ 4 การพัฒนาเครื่องมือสำหรับการเตรียมข้อมูล 41
4.1 การพัฒนาเครื่องมือสำหรับการเตรียมข้อมูล 41
4.1.1 เครื่องมือสำหรับนำข้อมูลเข้าฐานข้อมูลเชิงพื้นที่ 43
4.1.2 เครื่องมือสำหรับสร้างพีระมิดข้อมูล 47
4.2 การใช้งานเครื่องมือสำหรับการเตรียมข้อมูล 50
บทที่ 5 การพัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลา และการพัฒนาระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา 54

หน้า

5.1 การพัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลา 54
5.1.1 ฟังก์ชันในการทำงานของ Timeline 55
5.1.2 การนำ Timeline ไปใช้งาน 59
5.2 การพัฒนาระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา 61
5.2.1 การสร้างแผนที่ 64
5.2.2 การสร้าง Timeline 66
5.2.3 การแสดงข้อมูล 67
บทที่ 6 การทดลองและวิเคราะห์ผลการทดลอง 70
6.1 การเตรียมข้อมูลที่ใช้ในการทดสอบ 70
6.1.1 แหล่งที่มาของข้อมูล 70
6.1.1.1 ข้อมูลแผ่นดินไหว 70
6.1.1.2 ข้อมูลอุณหภูมิ 72
6.1.2 การสร้างพีระมิดข้อมูล 73
6.2 การทดสอบเครื่องมือ Timeline 74
6.3 วิธีการวัดและประเมินผลการทดลอง 77
6.4 ผลการทดลองและวิเคราะห์ผลการทดลอง 78
6.4.1 ข้อมูลแผ่นดินไหว 78
6.4.2 ข้อมูลอุณหภูมิ 84
6.5 สรุปผลการทดลอง 94
บทที่ 7 สรุปผลการวิจัย อภิปรายผลและข้อเสนอแนะ 96
7.1 สรุปและอภิปรายผลการวิจัย 96
7.2 ข้อจำกัดและข้อเสนอแนะ 97
รายการอ้างอิง 99
ภาคผนวก 101
ภาคผนวก ก การติดตั้ง GeoServer บน Windows 102
ภาคผนวก ข การเพิ่มชั้นข้อมูลใน GeoServer 111
ภาคผนวก ค ระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา 115
ภาคผนวก ง ไฟล์ JavaScript ของ Timeline 117
ประวัติผู้เขียนวิทยานิพนธ์ 118

สารบัญตาราง

หน้า

ตารางที่ 2.1 ตารางแสดงตัวอย่างข้อมูลเชิงพื้นที่ในแต่ละมาตราส่วนเวลา....9
ตารางที่ 3.1 ตารางแสดงตัวอย่างเกณฑ์ในการคัดเลือกหรือคำนวณข้อมูลมาแสดงบน แผนที่ของข้อมูลประเภทต่างๆ....25
ตารางที่ 3.2 ตารางแสดงตัวอย่างการเลือกหรือคำนวณข้อมูลของข้อมูลแผ่นดินไหว และข้อมูลอุณหภูมิ....26
ตารางที่ 3.3 ตารางแสดงตัวอย่างการสร้างพีระมิดข้อมูลของข้อมูลแผ่นดินไหวและ ข้อมูลอุณหภูมิ....32
ตารางที่ 4.1 โค้ดตัวอย่างสำหรับการตรวจสอบชนิดข้อมูลที่ต้องการนำเข้าฐานข้อมูล....45
ตารางที่ 4.2 โค้ดตัวอย่างสำหรับการนำข้อมูลแผ่นดินไหวเข้าฐานข้อมูล....45
ตารางที่ 4.3 โค้ดตัวอย่างสำหรับการสร้าง view ใน PostgreSQL ของข้อมูลแผ่นดินไหว....49
ตารางที่ 4.4 โค้ดตัวอย่างสำหรับการสร้าง view ใน PostgreSQL ของข้อมูลอุณหภูมิ....49
ตารางที่ 5.1 โค้ดตัวอย่างของฟังก์ชันในการดึงชั้นข้อมูลจาก GeoServer....56
ตารางที่ 5.2 โค้ดตัวอย่างสำหรับการตรวจสอบการกดเลื่อนเส้นกั้นเวลา....57
ตารางที่ 5.3 โค้ดตัวอย่างสำหรับการตรวจสอบการกดปุ่มเลื่อนดูเวลาบน Timeline....58
ตารางที่ 5.4 โค้ดตัวอย่างสำหรับการสร้างส่วนสำหรับแสดง Timeline....59
ตารางที่ 5.5 โค้ดตัวอย่างสำหรับการเรียกใช้ฟังก์ชันสำหรับสร้าง Timeline....60
ตารางที่ 5.6 โค้ดตัวอย่างสำหรับสร้างส่วนแสดงแผนที่และส่วนแสดง Timeline....63
ตารางที่ 5.7 โค้ดตัวอย่างสำหรับการสร้าง base map บนชั้นข้อมูลของ openLayers....64
ตารางที่ 5.8 โค้ดตัวอย่างสำหรับการนำไฟล์ JavaScript ของ Timeline ทั้งหมดไปใส่ใน หน้าเว็บแอพพลิเคชัน....66
ตารางที่ 5.9 โค้ดตัวอย่างของการเรียกใช้ฟังก์ชันในการสร้าง Timeline ผ่าน JavaScript....67
ตารางที่ 5.10 โค้ดตัวอย่างสำหรับการสร้างชั้นข้อมูลบน openLayers....68
ตารางที่ 6.1 ตารางแสดงตัวอย่างข้อมูลแผ่นดินไหวในฐานข้อมูลเชิงพื้นที่....71
ตารางที่ 6.2 ตารางแสดงตัวอย่างข้อมูลอุณหภูมิในฐานข้อมูลเชิงพื้นที่....72
ตารางที่ 6.3 ตารางแสดงการสร้างพีระมิดข้อมูลเชิงเวลาของข้อมูลแผ่นดินไหวและ ข้อมูลอุณหภูมิ....74
ตารางที่ 6.4 โค้ดตัวอย่างสำหรับหน้าเว็บแอพพลิเคชันในการแสดงข้อมูลเชิงพื้นที่....75

หน้า

ตารางที่ 6.5 โค้ดตัวอย่างสำหรับการสร้าง base map บน openLayers.....76

สารบัญภาพ

หน้า

ภาพที่ 2.1 เว็บไซต์แสดงน้ำท่วมในประเทศไทย 5
ภาพที่ 2.2 เว็บไซต์แสดงอุณหภูมิในแต่ละพื้นที่ 6
ภาพที่ 2.3 โครงสร้างข้อมูลแบบราสเตอร์ 7
ภาพที่ 2.4 โครงสร้างข้อมูลแบบเวกเตอร์ 8
ภาพที่ 2.5 ตัวอย่าง Web Mapping 10
ภาพที่ 2.6 การทำงานของ GeoServer 12
ภาพที่ 2.7 Time Slider ของ Google Earth 14
ภาพที่ 2.8 การแสดงข้อมูลความขัดแย้งระหว่างประเทศต่างๆ ในงานวิจัย [7] 15
ภาพที่ 2.9 การแสดงข้อมูลเชิงพื้นที่บน TimeMap Data Viewer ในงานวิจัย [6] 16
ภาพที่ 2.10 ส่วนแสดงผลของ Continuum ในงานวิจัย [1] 17
ภาพที่ 2.11 เส้นเวลาของ myHistro [5] 18
ภาพที่ 2.12 SIMILE Timeline 20
ภาพที่ 2.13 TimeMap 20
ภาพที่ 2.14 TimelineSetter [16] 21
ภาพที่ 2.15 Timeglider 22
ภาพที่ 3.1 การแสดงข้อมูลแบบพีระมิดข้อมูล 24
ภาพที่ 3.2 ตัวอย่างในการสร้างพีระมิดข้อมูลที่มีหลายระดับความละเอียดในเชิงพื้นที่และเวลา 28
ภาพที่ 3.3 การออกแบบ Timeline 29
ภาพที่ 3.4 ระบบต้นแบบ 30
ภาพที่ 3.5 การสร้างพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล 34
ภาพที่ 3.6 การเรียกใช้ชั้นข้อมูลของพีระมิดข้อมูลตามความละเอียดของเวลา 35
ภาพที่ 3.7 การเรียกใช้ชั้นข้อมูลของพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล 36
ภาพที่ 3.8 การสร้างชั้นพีระมิดข้อมูลของข้อมูลจาก Twitter 37
ภาพที่ 3.9 การสร้างชั้นพีระมิดข้อมูลของข้อมูลการรายงานการเกิดอุบัติเหตุทางจราจร 37
ภาพที่ 3.10 การสร้างชั้นพีระมิดข้อมูลการ check in ของโรงแรมทั่วโลก 38
ภาพที่ 3.11 การสร้างชั้นพีระมิดข้อมูลการรายงานโรคระบาด 39

หน้า

ภาพที่ 3.12 การสร้างชั้นพีระมิดข้อมูลการรายงานโรค 39
ภาพที่ 4.1 โครงสร้างของเครื่องมือสำหรับการเตรียมข้อมูล 42
ภาพที่ 4.2 การทำงานของเครื่องมือสำหรับการเตรียมข้อมูล 42
ภาพที่ 4.3 หน้าเว็บสำหรับนำข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิเข้าฐานข้อมูล PostgreSQL 43
ภาพที่ 4.4 การเลือกชนิดข้อมูลที่จะนำเข้าฐานข้อมูล 43
ภาพที่ 4.5 หน้าเว็บสำหรับการเตรียมข้อมูล 47
ภาพที่ 4.6 ส่วนแสดง Data source ของหน้าเว็บสำหรับการเตรียมข้อมูล 48
ภาพที่ 4.7 ส่วนแสดง Level ของหน้าเว็บสำหรับการเตรียมข้อมูล 48
ภาพที่ 4.8 ส่วนแสดง Criteria configuration ของหน้าเว็บสำหรับการเตรียมข้อมูล 48
ภาพที่ 4.9 ส่วนแสดง Level name ของหน้าเว็บสำหรับการเตรียมข้อมูล 49
ภาพที่ 4.10 การสร้างชั้นล่างสุดของพีระมิดข้อมูล 51
ภาพที่ 4.11 การสร้างชั้นที่ 2 ของพีระมิดข้อมูล 51
ภาพที่ 4.12 การสร้างชั้นที่ 3 ของพีระมิดข้อมูล 52
ภาพที่ 4.13 การสร้างชั้นที่ 4 ของพีระมิดข้อมูล 53
ภาพที่ 5.1 ตัวอย่าง Timeline ที่พัฒนาขึ้นมา 54
ภาพที่ 5.2 Timeline ในระดับความละเอียดของเวลาเป็นปี 55
ภาพที่ 5.3 Timeline ในระดับความละเอียดของเวลาเป็นเดือน 55
ภาพที่ 5.4 Timeline ในระดับความละเอียดของเวลาเป็นวัน 55
ภาพที่ 5.5 Timeline ในระดับความละเอียดของเวลาเป็นชั่วโมง 56
ภาพที่ 5.6 การสร้าง Timeline บนหน้าเว็บแอพพลิเคชัน 61
ภาพที่ 5.7 โครงร่างของระบบต้นแบบ 62
ภาพที่ 5.8 ขั้นตอนการพัฒนาระบบ 63
ภาพที่ 5.9 โครงร่างของหน้าเว็บแอพพลิเคชันหลัก 64
ภาพที่ 5.10 หน้าเว็บแอพพลิเคชันหลักในส่วนของแผนที่ 66
ภาพที่ 5.11 หน้าเว็บแอพพลิเคชันหลักในส่วนของแผนที่และ Timeline 67
ภาพที่ 5.12 การแสดงชั้นข้อมูล 68
ภาพที่ 5.13 หน้าเว็บแอพพลิเคชันหลัก 69
ภาพที่ 6.1 หน้าเว็บแอพพลิเคชันสำหรับแสดงข้อมูลแผ่นดินไหว 77

หน้า

ภาพที่ 6.2 หน้าเว็บแอพพลิเคชันสำหรับแสดงข้อมูลอุณหภูมิ 77
ภาพที่ 6.3 การแสดงข้อมูลแผ่นดินไหวโดยไม่มีการคัดเลือกข้อมูล 78
ภาพที่ 6.4 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามระดับความละเอียดของ
เวลาในระดับชั่วโมงโดยใช้ฟังก์ชัน Greater than 79
ภาพที่ 6.5 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามระดับความละเอียดของ
เวลาในระดับวันโดยใช้ฟังก์ชัน Greater than 79
ภาพที่ 6.6 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามระดับความละเอียดของ
เวลาในระดับเดือนโดยใช้ฟังก์ชัน Greater than 80
ภาพที่ 6.7 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามระดับความละเอียดของ
เวลาในระดับปีโดยใช้ฟังก์ชัน Greater than 80
ภาพที่ 6.8 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล
ในระดับชั่วโมงโดยเลือกความรุนแรงตั้งแต่ 0-3 ริกเตอร์ 81
ภาพที่ 6.9 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล
ในระดับวันโดยเลือกความรุนแรงตั้งแต่ 3 ริกเตอร์ขึ้นไป 81
ภาพที่ 6.10 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล
ในระดับเดือนโดยเลือกความรุนแรงตั้งแต่ 4 ริกเตอร์ขึ้นไป 82
ภาพที่ 6.11 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล
ในระดับปีโดยเลือกความรุนแรงตั้งแต่ 3 ริกเตอร์ขึ้นไป 82
ภาพที่ 6.12 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล
ในระดับปีโดยเลือกความรุนแรงตั้งแต่ 6 ริกเตอร์ขึ้นไป 83
ภาพที่ 6.13 การแสดงข้อมูลอุณหภูมิโดยไม่มีการคำนวณข้อมูล 84
ภาพที่ 6.14 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา
ในระดับชั่วโมงโดยใช้ฟังก์ชัน Average 85
ภาพที่ 6.15 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา
ในระดับวันโดยใช้ฟังก์ชัน Average 85
ภาพที่ 6.16 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา
ในระดับเดือนโดยใช้ฟังก์ชัน Average 86
ภาพที่ 6.17 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา
ในระดับปีโดยใช้ฟังก์ชัน Average 86

หน้า

ภาพที่ 6.18 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา ในระดับชั่วโมงโดยใช้ฟังก์ชัน Maximum 87
ภาพที่ 6.19 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา ในระดับวันโดยใช้ฟังก์ชัน Maximum 87
ภาพที่ 6.20 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา ในระดับเดือนโดยใช้ฟังก์ชัน Maximum 88
ภาพที่ 6.21 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา ในระดับปีโดยใช้ฟังก์ชัน Maximum 88
ภาพที่ 6.22 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา ในระดับชั่วโมงโดยใช้ฟังก์ชัน Minimum 89
ภาพที่ 6.23 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา ในระดับวันโดยใช้ฟังก์ชัน Minimum 89
ภาพที่ 6.24 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา ในระดับเดือนโดยใช้ฟังก์ชัน Minimum 90
ภาพที่ 6.25 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลา ในระดับปีโดยใช้ฟังก์ชัน Minimum 90
ภาพที่ 6.26 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล ในระดับชั่วโมงโดยเลือกอุณหภูมิเฉลี่ยตั้งแต่ 30 องศาเซลเซียสขึ้นไป 91
ภาพที่ 6.27 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล ในระดับวันโดยเลือกอุณหภูมิเฉลี่ยตั้งแต่ 30 - 40 องศาเซลเซียส 91
ภาพที่ 6.28 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล ในระดับเดือนโดยเลือกอุณหภูมิเฉลี่ยตั้งแต่ 30 - 40 องศาเซลเซียส 92
ภาพที่ 6.29 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล ในระดับปีโดยเลือกอุณหภูมิเฉลี่ยตั้งแต่ 25 - 27 องศาเซลเซียส 92
ภาพที่ 6.30 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล ในระดับปีโดยเลือกอุณหภูมิสูงที่สุดตั้งแต่ 40 องศาเซลเซียสขึ้นไป 93
ภาพที่ 6.31 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล ในระดับปีโดยเลือกอุณหภูมิต่ำที่สุดที่น้อยกว่า 20 องศาเซลเซียส 93
ภาพที่ ก.1 การดาวน์โหลดซอฟต์แวร์ 102

หน้า

ภาพที่ ก.2 ขั้นตอนการติดตั้งซอฟต์แวร์ GeoServer 103
ภาพที่ ก.3 แสดงรายละเอียดเงื่อนไขของซอฟต์แวร์ 103
ภาพที่ ก.4 เลือกปลายทางการติดตั้งซอฟต์แวร์ GeoServer 104
ภาพที่ ก.5 แสดงชื่อโฟลเดอร์ของซอฟต์แวร์ที่ติดตั้ง 104
ภาพที่ ก.6 เลือก path ที่ติดตั้ง Java Runtime Environment 105
ภาพที่ ก.7 เลือก directory ของข้อมูล 105
ภาพที่ ก.8 กำหนดชื่อผู้ใช้และรหัสผ่าน 106
ภาพที่ ก.9 ระบุ port ของ GeoServer 106
ภาพที่ ก.10 เลือกชนิดของการติดตั้ง 107
ภาพที่ ก.11 แสดงรายละเอียดการติดตั้งซอฟต์แวร์ 108
ภาพที่ ก.12 แสดงหน้าจอขั้นตอนสิ้นสุดการติดตั้งซอฟต์แวร์ GeoServer 108
ภาพที่ ก.13 แสดงการ Start service GeoServer 109
ภาพที่ ก.14 แสดงหน้าต่างการ Start GeoServer 109
ภาพที่ ก.15 Web Administration Interface ของ GeoServer 110
ภาพที่ ข.1 การสร้าง Layer ใหม่ใน GeoServer 111
ภาพที่ ข.2 การเลือกฐานข้อมูลที่จะนำมาสร้าง Layer ใน GeoServer 111
ภาพที่ ข.3 การเข้าสู่ Configure new SQL view ใน GeoServer 112
ภาพที่ ข.4 การสร้าง SQL view ใน GeoServer 112
ภาพที่ ข.5 การเพิ่มรายละเอียดของ Layer ใน GeoServer 113
ภาพที่ ข.6 การกำหนดลักษณะรูปร่างของข้อมูลที่แสดงบนแผนที่ใน GeoServer 114
ภาพที่ ค.1 หน้าหลักของระบบต้นแบบ 115
ภาพที่ ค.2 หน้าสำหรับนำข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิเข้าฐานข้อมูล PostgreSQL 115
ภาพที่ ค.3 หน้าสำหรับการเตรียมข้อมูล 116
ภาพที่ ค.4 หน้าสำหรับแสดงเตรียมข้อมูลทางภูมิศาสตร์ 116

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันมีการใช้ระบบสารสนเทศเชิงภูมิศาสตร์ (Geographic Information System: GIS) และโปรแกรมประยุกต์เกี่ยวกับแผนที่ [11] กันอย่างแพร่หลาย และมีการนำแผนที่แบบโต้ตอบ (Interactive map) มาแสดงบนหน้าเว็บจำนวนมาก เรียกว่า Web Mapping แผนที่แบบนี้มักจะมีส่วนควบคุมการซูมและเลื่อน (zoom and pan control) ทำให้สามารถมองเห็นภาพเมืองหรือแม้กระทั่งแผนที่ทั้งโลกที่ระดับความละเอียดต่าง ๆ บนหน้าจอคอมพิวเตอร์เล็ก ๆ ได้ และยังมีข้อมูลทางภูมิศาสตร์ประเภทต่าง ๆ เช่น ภาพถ่ายจากดาวเทียม ถนน สถานที่สำคัญ ธุรกิจและบริการ โมเด็ลสามมิติ ฯลฯ ที่สามารถนำมาวางซ้อนกันบนแผนที่ได้ โดยผู้ใช้สามารถเลือกที่จะแสดงหรือไม่แสดงชั้นข้อมูลบางชั้นข้อมูลได้

โดยทั่วไปแผนที่ซึ่งสามารถซูมเข้า-ออกได้จะแสดงข้อมูลที่มีความละเอียดแตกต่างกันขึ้นอยู่กับระดับการซูม (zoom level) โดยในแต่ละระดับจะมีการเลือกข้อมูลมาแสดงบนแผนที่ให้เหมาะสมกับขนาดของหน้าจอเพื่อให้เห็นข้อมูลได้ชัดเจนและลดปริมาณข้อมูลที่ต้องนำมาแสดง ตัวอย่างเช่น ที่ระดับการซูมน้อย ๆ (มองจากระดับความสูงมาก ๆ) แผนที่อาจแสดงภาพถ่ายจากดาวเทียมที่มีมาตราส่วนเล็กที่สุด พร้อมกับชื่อสถานที่ระดับอำเภอ และแสดงถนนระดับทางหลวง แต่เมื่อซูมเข้าไปมาก ๆ (มองจากระดับความสูงน้อย ๆ) ก็จะแสดงภาพถ่ายจากดาวเทียมที่มีมาตราส่วนใหญ่ที่สุด พร้อมกับชื่อสถานที่ในระดับท้องถิ่น และถนนย่อย เป็นต้น ดังนั้น เพื่อความสะดวกในการนำข้อมูลในแต่ละระดับความละเอียดมาแสดง ข้อมูลเชิงพื้นที่ที่มีความละเอียดสูง ๆ มักจะถูกเตรียมเป็นชั้นข้อมูลหลาย ๆ ชั้นที่มีความละเอียดหรือมาตราส่วนลดหลั่นกันไป จนมีลักษณะเหมือนพีระมิดของข้อมูล ที่ชั้นล่างสุดคือข้อมูลที่มีความละเอียดสูงสุดและชั้นบนสุดคือชั้นที่มีความละเอียดต่ำสุด ในการแสดงผลก็จะเลือกเอาชั้นข้อมูลมาแสดงทีละชั้นโดยพิจารณาจากระดับการซูม

นอกจากคุณสมบัติในเชิงพื้นที่แล้ว ข้อมูลทางภูมิศาสตร์ยังมีคุณสมบัติในเชิงเวลาอีกด้วย อันได้แก่ เวลาที่เกิดเหตุการณ์ หรือเวลาที่บันทึกข้อมูล ตัวอย่างของข้อมูลเชิงพื้นที่ที่เกี่ยวเนื่องกับเวลาได้แก่ ข้อมูลภาพถ่ายดาวเทียมที่ถ่ายในช่วงเวลาต่าง ๆ, ข้อมูลความหนาแน่นของการจราจรบนถนน, ข้อมูลสภาพภูมิอากาศ และข้อมูลการเกิดแผ่นดินไหว เป็นต้น นอกจากนี้ ปริมาณข้อมูลเชิงพื้นที่ที่ถูกสร้างขึ้นมายังมีอัตราเพิ่มสูงขึ้นเรื่อย ๆ ส่วนหนึ่งก็เนื่องจากมีการใช้งานอุปกรณ์ที่มี

ระบบกำหนดตำแหน่งบนโลกหรือจีพีเอส (Global Positioning system: GPS) อยู่เป็นจำนวนมาก ทำให้คุณสมบัติด้านเวลาของข้อมูลมีความสำคัญมากขึ้น

อย่างไรก็ตาม การแสดงข้อมูลทางภูมิศาสตร์ในเชิงเวลายังไม่ได้รับการสนับสนุนที่ดีเท่ากับการแสดงข้อมูลในเชิงพื้นที่ หากนำข้อมูลในทุกช่วงเวลามาแสดงลงบนแผนที่พร้อมกัน ก็จะทำให้ไม่เห็นลำดับเวลาของการเกิดข้อมูล ดังนั้น จึงจำเป็นต้องมีเครื่องมือสำหรับช่วยแสดงข้อมูลเชิงเวลา เพื่อให้เห็นลำดับเหตุการณ์และความเปลี่ยนแปลงของข้อมูลในช่วงเวลาต่าง ๆ เครื่องมือที่นิยมใช้กันคือ เส้นเวลา (Timeline) แม้ว่าซอฟต์แวร์สำหรับดูแผนที่ (Map Viewer Software) บางตัวอาจมีเครื่องมือดังกล่าว แต่ถ้าแสดงข้อมูลปริมาณมาก ๆ ในช่วงเวลาที่ยาวนาน ก็จะทำให้ข้อมูลที่แสดงบนแผนที่มีปริมาณมากเกินไป จนอาจทำให้ดูข้อมูลบนแผนที่ไม่รู้เรื่อง

ขอยกตัวอย่างการแสดงข้อมูลการเกิดแผ่นดินไหวดังนี้ ตามสถิติการเกิดแผ่นดินไหวทั่วโลก ในแต่ละปีมีแผ่นดินไหวขนาด 3-3.9 ริกเตอร์ ประมาณ 130,000 ครั้ง ขนาด 4-4.9 ริกเตอร์ ประมาณ 13,000 ครั้ง ขนาด 5-5.9 ริกเตอร์ ประมาณ 1,300 ครั้ง และขนาดตั้งแต่ 6 ริกเตอร์ขึ้นไป ประมาณหนึ่งร้อยกว่าครั้ง [20] ถ้าต้องการแสดงข้อมูลแผ่นดินไหวในช่วงเวลาหนึ่งเดือน หากแสดงตั้งแต่ขนาด 3 ริกเตอร์ก็จะมีจำนวนจุดมากกว่าหนึ่งหมื่นจุดทั่วโลก ซึ่งอาจจะมากเกินไป แต่หากแสดงตั้งแต่ขนาด 5 ริกเตอร์ขึ้นไปก็จะมีจำนวนจุดประมาณหนึ่งร้อยกว่าจุด ซึ่งจะเป็นการเน้นให้เห็นเหตุการณ์ที่สำคัญเท่านั้น ดังนั้น ในกรณีของการแสดงข้อมูลเหตุการณ์แผ่นดินไหวในช่วงเวลาระดับปี, เดือน และวัน ควรจะแสดงชุดข้อมูลที่แตกต่างกัน โดยอาจจะใช้ขนาดของแผ่นดินไหวเป็นเกณฑ์ในการคัดเลือกข้อมูลมาแสดงในแต่ละระดับความละเอียดของเวลา

อีกตัวอย่างหนึ่งคือกรณีที่ข้อมูล ณ ตำแหน่งเดียวกันมีการเปลี่ยนแปลงบ่อยมาก เช่น ข้อมูลอุณหภูมิที่ได้จากเซ็นเซอร์ที่มีตำแหน่งคงที่ และมีการวัดค่าทุกหนึ่งชั่วโมง เป็นต้น เมื่อแสดงบนแผนที่ก็จะเห็นความเปลี่ยนแปลงเป็นรายชั่วโมงได้ แต่ถ้าต้องการดูข้อมูลที่ระดับความละเอียดในเชิงเวลาอื่น ๆ ก็จะต้องนำข้อมูลรายชั่วโมงมาผ่านการประมวลผลให้ได้เป็นข้อมูลรายวัน เดือน หรือปี เช่น อุณหภูมิต่ำสุดและสูงสุดในแต่ละวัน เดือน หรือปี ซึ่งหากทำได้ก็จะเป็นประโยชน์อย่างยิ่ง

อย่างไรก็ตาม ด้วยเทคโนโลยีของ web mapping ในปัจจุบัน ทั้งด้านข้อมูล (เช่น KML [21]) และซอฟต์แวร์สำหรับดูแผนที่ (เช่น Google Earth [2]) ยังไม่สนับสนุนการลดปริมาณหรือการคัดเลือกข้อมูลตามระดับความละเอียดของเวลาดังตัวอย่างที่กล่าวข้างต้น แม้ว่าระบบฐานข้อมูลเชิงสัมพันธ์ทั่วไปสนับสนุนการคัดเลือกข้อมูลที่ต้องการโดยใช้โปรแกรมประเภท SQL (Structured Query Language) แต่การเรียกข้อมูลมาแสดงผลบนแผนที่ก็อาจใช้เวลานาน ไม่เหมาะกับการทำงานแบบโต้ตอบของ web map

ผู้วิจัยจึงมีความต้องการที่จะแก้ปัญหาในการแสดงข้อมูลทางภูมิศาสตร์ที่เกี่ยวเนื่องกับเวลาในปริมาณมาก ๆ อย่างมีประสิทธิภาพ โดยมีแนวคิดคือการสร้างพีระมิดข้อมูลที่แบ่งชั้นข้อมูลตามความละเอียดในเชิงเวลาแทนที่จะแบ่งตามความละเอียดในเชิงพื้นที่ และใช้ timeline control สำหรับเลือกดูข้อมูลในช่วงเวลาที่ต้องการ

1.2 วัตถุประสงค์

1. เพื่อออกแบบและพัฒนาเทคนิคในการแสดงข้อมูลที่มีหลายระดับความละเอียดของเวลา
2. เพื่อออกแบบและพัฒนาเทคนิคในการควบคุมปริมาณข้อมูลที่แสดงบนแผนที่ในแต่ละระดับความละเอียดของเวลา

1.3 ขอบเขตของการวิจัย

1. ระบบที่จะพัฒนาขึ้นเพื่อทดสอบจะเป็น Web Mapping Application ประกอบด้วยส่วน server และส่วน client
    - ส่วน server จะใช้ Web Map Server เช่น GeoServer
    - ส่วน client จะทำงานบน browser ที่เป็นที่นิยมใช้ เช่น Firefox และพัฒนาด้วยเครื่องมือที่เป็นที่นิยมเช่น JavaScript และ Google Map API เป็นต้น
2. ข้อมูลตัวอย่างที่จะนำมาทดสอบจะใช้ข้อมูลเชิงพื้นที่และเวลา ที่หาได้จากแหล่งข้อมูลสาธารณะอย่างน้อย 2 ตัวอย่าง เช่น ข้อมูลแผ่นดินไหว, ข้อมูลอุณหภูมิ เป็นต้น
3. งานวิจัยนี้มุ่งเน้นที่ข้อมูลหลายระดับความละเอียด (multi-resolution) ในเชิงเวลา ไม่รวมถึงระดับความละเอียดในเชิงพื้นที่

1.4 ประโยชน์ที่จะได้รับ

1. ได้เทคนิควิธีการในการแสดงข้อมูลทางภูมิศาสตร์ที่มีหลายระดับความละเอียดของเวลา
2. ได้ Web Mapping Application สำหรับแสดงข้อมูลบนแผนที่ที่รองรับหลายระดับความละเอียดของเวลา

3. ได้เทคนิคในการควบคุมปริมาณข้อมูลที่แสดงบนแผนที่ในแต่ละระดับความละเอียดของเวลา

1.5 วิธีการดำเนินการวิจัย

1. ศึกษาเกี่ยวกับแผนที่แบบหลายระดับความละเอียด และ Timeline
2. ออกแบบวิธีการแสดงผล Timeline แบบหลายระดับความละเอียด
3. ออกแบบวิธีการเตรียมข้อมูลที่มีความละเอียดเชิงเวลาหลายระดับ
4. พัฒนาระบบตามที่ได้ออกแบบ
5. ทดสอบการใช้งานของระบบ
6. วิเคราะห์ผลการทดสอบและสรุปผลการทดสอบ
7. เรียบเรียงวิทยานิพนธ์

1.6 การเผยแพร่ผลงานวิจัย

ส่วนหนึ่งของงานวิทยานิพนธ์นี้ ได้รับการตีพิมพ์เป็นบทความทางวิชาการจำนวน 1 เรื่อง ดังนี้

“Visualization of Temporal Geographic Data based on Multi-resolution Data Pyramids and a Multi-level Time-scale” โดย จตุรงค์ ลีลาวัฒนพาณิชย์ และ ผศ.ดร.วีระ เหมืองสิน ในงานประชุมวิชาการ “The 17th International Annual Symposium on Computational Science and Engineering (ANSCSE17)” ซึ่งจัดขึ้น ณ มหาวิทยาลัยขอนแก่น จังหวัดขอนแก่น ประเทศไทย ระหว่างวันที่ 27 มีนาคม ถึง 29 มีนาคม 2556

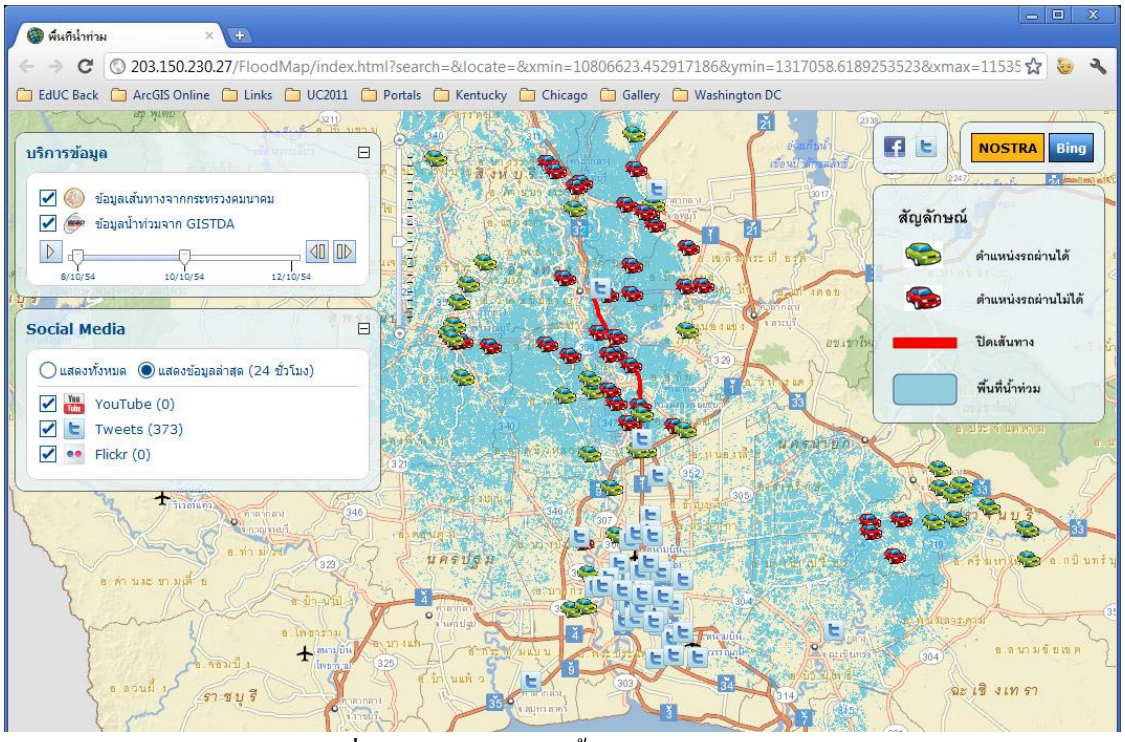
บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในหัวข้อทฤษฎีและงานวิจัยที่เกี่ยวข้องนี้จะกล่าวถึงทฤษฎีเกี่ยวกับระบบสารสนเทศเชิงภูมิศาสตร์, ข้อมูลเชิงพื้นที่, ข้อมูลเชิงเวลา, เส้นเวลา, Web Mapping และ GeoServer จากนั้นในส่วนของงานวิจัยที่เกี่ยวข้องจะแสดงถึงการแสดงข้อมูลทางภูมิศาสตร์ในเชิงพื้นที่ และเชิงเวลา พร้อมทั้งการวิเคราะห์และวิจารณ์งานที่ผ่านมาเหล่านั้น

2.1 ทฤษฎีที่เกี่ยวข้อง

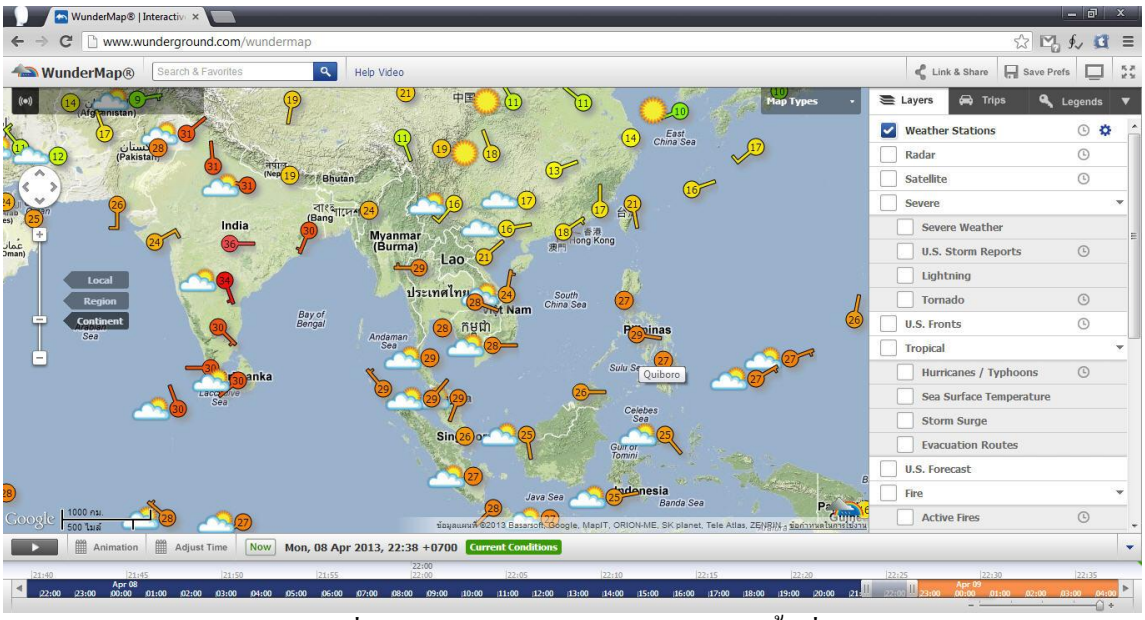
2.1.1 Geographic Information System

Geographic Information System (GIS) หรือระบบสารสนเทศเชิงภูมิศาสตร์ คือ ระบบสารสนเทศที่ถูกออกแบบมาเพื่อจัดเก็บ จัดการ วิเคราะห์ และแสดงข้อมูลเชิงพื้นที่ที่มีการอ้างอิงกับตำแหน่งทางภูมิศาสตร์ โดยในงานวิจัยนี้เน้นไปที่การแสดงข้อมูลเชิงพื้นที่บนแผนที่ในรูปแบบเว็บแอพพลิเคชัน ที่เรียกว่า Web GIS ซึ่งมี Web GIS ที่ใช้งานและเผยแพร่อยู่เป็นจำนวนมาก เช่น เว็บไซต์แสดงน้ำท่วม (ภาพที่ 2.1), เว็บไซต์แสดงอุณหภูมิ (ภาพที่ 2.2) เป็นต้น



ภาพที่ 2.1 เว็บไซต์แสดงน้ำท่วมในประเทศไทย

(ที่มา : http://203.150.230.27/FloodMap ของ ESRI)



ภาพที่ 2.2 เว็บไซต์แสดงอุณหภูมิในแต่ละพื้นที่

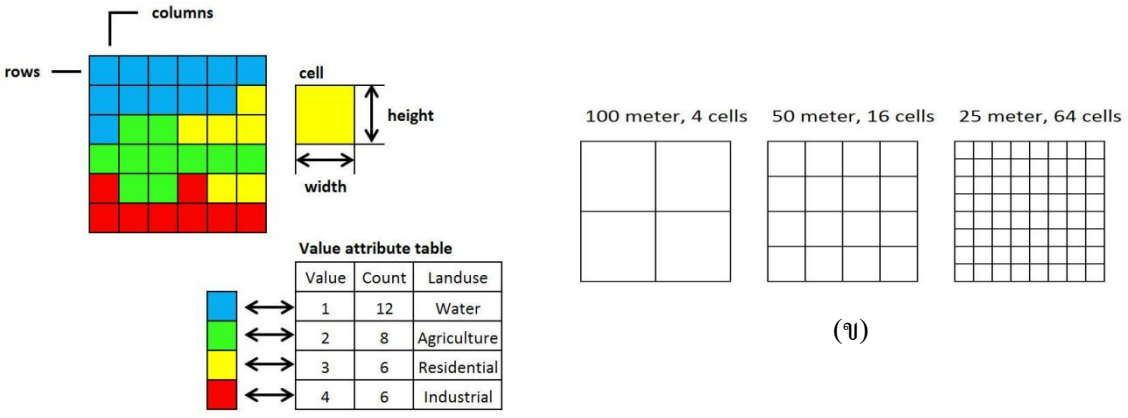
(ที่มา : http://www.wunderground.com/wundermap)

2.1.2 ข้อมูลเชิงพื้นที่

แผนที่บนเว็บแอพพลิเคชันส่วนใหญ่จะมีข้อมูลต่าง ๆ แสดงอยู่บนแผนที่ บ้างเป็นจุด บ้างเป็นเส้น บ้างเป็นรูปหลายเหลี่ยม ซึ่งข้อมูลเหล่านี้ล้วนแต่มีพิกัดตำแหน่งระบุอยู่บนแผนที่ที่สามารถนำไปใช้ในการอ้างอิงกับตำแหน่งทางภูมิศาสตร์หรือตำแหน่งจริงบนพื้นโลกได้ (Geo-referenced) ซึ่งเรียกข้อมูลเหล่านี้ว่าข้อมูลเชิงพื้นที่ (Spatial data) โดยการอ้างอิงสามารถทำได้หลายวิธี เช่น การอ้างอิงโดยใช้พิกัดทางภูมิศาสตร์ ละติจูด ลองจิจูด หรือการอ้างอิงโดยใช้พิกัดของกริด x, y เป็นต้น ซึ่งข้อมูลเชิงพื้นที่สามารถแบ่งโครงสร้างการจัดข้อมูลเก็บได้เป็น 2 ประเภทคือ

2.1.2.1 โครงสร้างข้อมูลแบบราสเตอร์

ข้อมูลแบบราสเตอร์ (Raster or grid representation) นี้มีโครงสร้างเป็นช่องสี่เหลี่ยม เรียกว่า จุดภาพ (Picture element cell) หรือกริดเซลล์ (Grid cell) เรียงต่อเนื่องกันในแนวราบและแนวดิ่ง แต่ละช่องเรียกว่า Cell หรือ Pixel ที่มีขนาดเท่า ๆ กัน ซึ่งจะจัดเก็บข้อมูลโดยการแทนค่าของข้อมูลจากพื้นที่จริงลงในจุดภาพเลย ซึ่งในแต่ละ Cell จะเก็บค่าได้เพียง 1 ค่าเท่านั้น ดังภาพที่ 2.3 (ก) และการอ้างอิงพิกัดนั้น Raster จะรู้ค่าพิกัด X, Y เพียง Pixel เดียว (ส่วนมากใช้จุดล่างซ้าย) จากนั้นเมื่อรู้ขนาดของแต่ละ Pixel ก็สามารถอ้างอิงพิกัดได้ทั้งหมดทั้งภาพ โดยความละเอียดของข้อมูล Raster ขึ้นอยู่กับขนาดของ Grid หรือ Pixel ดังภาพที่ 2.3 (ข)



(ก) (ข)

ภาพที่ 2.3 โครงสร้างข้อมูลแบบราสเตอร์

(ก) การอ้างอิงพิกัด (ข) ความละเอียดของข้อมูล

ลักษณะข้อมูลแบบราสเตอร์ได้แก่ ข้อมูลภาพถ่ายดาวเทียม เป็นต้น ซึ่งข้อมูลแบบราสเตอร์มีจุดเด่นที่มีโครงสร้างไม่ซับซ้อน ทำให้การประมวลผลในระดับจุดภาพ การเปรียบเทียบระหว่างจุดภาพหรือการซ้อนทับข้อมูลเชิงพื้นที่ การนำไปใช้งานร่วมกับภาพถ่ายดาวเทียม การนำไปแทนลักษณะของข้อมูลพื้นผิว (Surface) ที่มีความต่อเนื่องของข้อมูล แต่ข้อด้อยที่สำคัญของข้อมูลราสเตอร์ ก็คือ ขนาดของไฟล์มีขนาดใหญ่ทำให้ใช้พื้นที่ในการจัดเก็บค่อนข้างมาก และในกรณีของแผนที่บนเว็บ ข้อมูลราสเตอร์ก็จะใช้เวลาในการส่งข้อมูลมากด้วย

2.1.2.2 โครงสร้างข้อมูลแบบเวคเตอร์

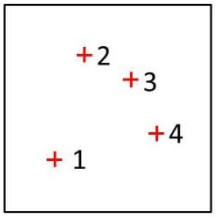
ข้อมูลแบบเวคเตอร์ (Vector representation) จะเป็นการใช้ค่าพิกัด X, Y หรือ X, Y, Z เป็นตัวอ้างอิงกับตำแหน่งบนพื้นโลกจริงในการอธิบายรูปร่างของจุด, เส้น, พื้นที่ ข้อมูลที่จัดเก็บในลักษณะเวคเตอร์ตำแหน่งของวัตถุจะถูกบันทึก และถ่ายทอดได้ละเอียดถูกต้องกว่าโครงสร้างแบบราสเตอร์ และใช้พื้นที่ในการจัดเก็บข้อมูลน้อยกว่า ตัวอย่างของข้อมูลที่จัดเก็บในประเภทนี้ได้ แก่ ข้อมูลแนวถนน, แม่น้ำลำคลอง, ขอบเขตการปกครอง เป็นต้น

รูปแบบของข้อมูลเชิงพื้นที่ในรูปแบบของเวคเตอร์จะมีลักษณะและรูปแบบ (Spatial Features) ต่าง ๆ ดังนี้

- จุด (Point) ดังภาพที่ 2.4 (ก) เป็นหน่วยย่อยที่สุดของเวคเตอร์ จะไม่มีขนาดของพื้นที่และระยะทาง ไม่มีมิติ แต่มีตำแหน่งเป็นค่าพิกัดของ x, y คู่หนึ่ง โดยจะอธิบายถึงตำแหน่งที่ตั้งของข้อมูล ตัวอย่างของข้อมูลภูมิศาสตร์ที่ใช้จุดเป็นสัญลักษณ์แทนใน

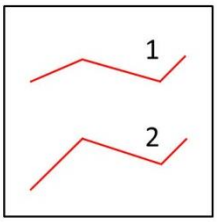
แผนที่ เช่น ตำแหน่งของเสาไฟฟ้า, ตำแหน่งของหม้อแปลงไฟฟ้า และที่ตั้งของสถานีย่อย เป็นต้น

- เส้น (Line) ดังภาพที่ 2.4 (ข) ประกอบไปด้วยลักษณะของเส้นตรง, เส้นหักมุม และเส้นโค้ง โดยที่เส้นประกอบด้วยจุดเริ่มต้น (From Node) และจุดสิ้นสุด (To Node) และจุดเปลี่ยนทิศทาง (Vertex) ที่ทำให้เส้นเกิดการเปลี่ยนทิศทางในการวางตัว ซึ่งทำให้เกิดเป็นรูปร่างของเส้น ซึ่งจะอธิบายถึงลักษณะต่าง ๆ โดยอาศัยขนาดทั้งความกว้างและความยาว เช่น ถนน, แม่น้ำ และเส้นทางรถไฟ เป็นต้น
- พื้นที่ (Polygon) ดังภาพที่ 2.4 (ค) เป็นลักษณะขอบเขตพื้นที่ที่เรียกว่า รูปปิดหลายเหลี่ยม (Polygon) ซึ่งจะต้องประกอบด้วยจุดมากกว่า 3 จุดขึ้นไป สามารถคำนวณขอบเขตเนื้อที่และเส้นรอบวงของรูปปิดหลายเหลี่ยมนั้นได้ เป็นลักษณะที่ใช้แสดงพื้นที่หรือขอบเขตตัวอย่างเช่น ขอบเขตของพื้นที่ป่าไม้, อาณาเขตการปกครอง, ขอบเขตการถือครองที่ดิน และพื้นที่จังหวัด เป็นต้น



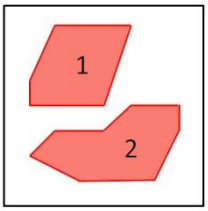
Point Number	x,y Coordinates
1	2,2
2	3,6
3	5,5
4	6,3

(ก)



Line Number	x,y Coordinates
1	1,5 3,6 6,5 7,6
2	1,1 3,3 6,2 7,3

(ข)



Polygon Number	x,y Coordinates
1	1,4 1,5 2,7 5,7 4,4 1,4
2	1,2 2,3 4,3 5,4 7,5 7,3 6,1 3,1 1,2

(ค)

ภาพที่ 2.4 โครงสร้างข้อมูลแบบเวกเตอร์

(ก) จุด (ข) เส้น (ค) พื้นที่

2.1.3 ข้อมูลเชิงเวลา

ข้อมูลเชิงเวลา (Temporal data) คือข้อมูลเชิงพื้นที่ที่มีเวลามาเกี่ยวข้อง ซึ่งข้อมูลเชิงพื้นที่ ณ พื้นที่หนึ่ง ๆ ในช่วงเวลาที่แตกต่างกันจะมีข้อมูลที่แตกต่างกัน ซึ่งเรียกข้อมูลประเภทนี้ว่า ข้อมูลเชิงเวลา ตัวอย่างของข้อมูลเชิงเวลาได้แก่ ข้อมูลภาพถ่ายดาวเทียมของพื้นที่เดียวกันที่ถ่ายในช่วงเวลาต่าง ๆ, ข้อมูล GPS นำทาง, ข้อมูลการจราจร, ข้อมูลสภาพภูมิอากาศ และข้อมูลแผ่นดินไหว เป็นต้น ซึ่งหากนำข้อมูลเหล่านี้มาแสดงผลบนแผนที่โดยนำข้อมูลในทุกช่วงเวลามาแสดงลงบนแผนที่พร้อมกัน จะทำให้ไม่เห็นลำดับเวลาของการเกิดข้อมูล ดังนั้นจึงจำเป็นต้องมีเครื่องมือสำหรับช่วยแสดงข้อมูลเชิงเวลา เพื่อให้เห็นลำดับเหตุการณ์ของข้อมูลในช่วงเวลาต่าง ๆ เครื่องมือดังกล่าวคือ เส้นเวลา (Timeline) ซึ่งจะกล่าวในหัวข้อถัดไป

2.1.4 เส้นเวลา

เส้นเวลา (Timeline) คือ เส้นสมมติที่กำหนดให้ตำแหน่งบนเส้นแทนเวลาจากอดีตไปยังอนาคต ใช้บ่งบอกว่า ณ เวลาใดเกิดเหตุการณ์ใดขึ้นบ้าง โดยจะเรียงเหตุการณ์ตามลำดับเวลาของการเกิดเหตุการณ์ก่อนหลัง ซึ่งเส้นเวลาเป็นเครื่องมือที่ใช้อธิบายเหตุการณ์ที่ผ่านมาแล้วได้อย่างดี ซึ่งจะทำให้เรามองเห็นภาพรวมของเหตุการณ์ตั้งแต่อดีตถึงปัจจุบันได้อย่างชัดเจน และเส้นเวลายังนำมาใช้ในการแสดงข้อมูลต่าง ๆ ในหลายด้าน เช่น ประวัติศาสตร์ [8], วิวัฒนาการ [10] และดาราศาสตร์ [15] เป็นต้น

นอกจากนี้เส้นเวลายังนำมาใช้ในการแสดงข้อมูลเชิงพื้นที่ที่มีเวลามาเกี่ยวข้องด้วย ซึ่งจะแสดงบนเว็บแอพพลิเคชันที่มีแผนที่สำหรับแสดงข้อมูลเชิงพื้นที่ ณ ตำแหน่ง (ละติจูด, ลองจิจูด) ต่าง ๆ และมีเส้นเวลาสำหรับจัดการลำดับเวลาในการแสดงข้อมูลเชิงพื้นที่ โดยเส้นเวลาจะใช้มาตราส่วนเวลา (Time-scale) ใดก็ได้ขึ้นอยู่กับข้อมูลว่ามีความถี่ในการเกิดเหตุการณ์หรือความเปลี่ยนแปลงมากแค่ไหน ซึ่งโดยทั่วไปนิยมใช้มาตราส่วนเวลาเป็นเชิงเส้นดังตารางที่ 2.1

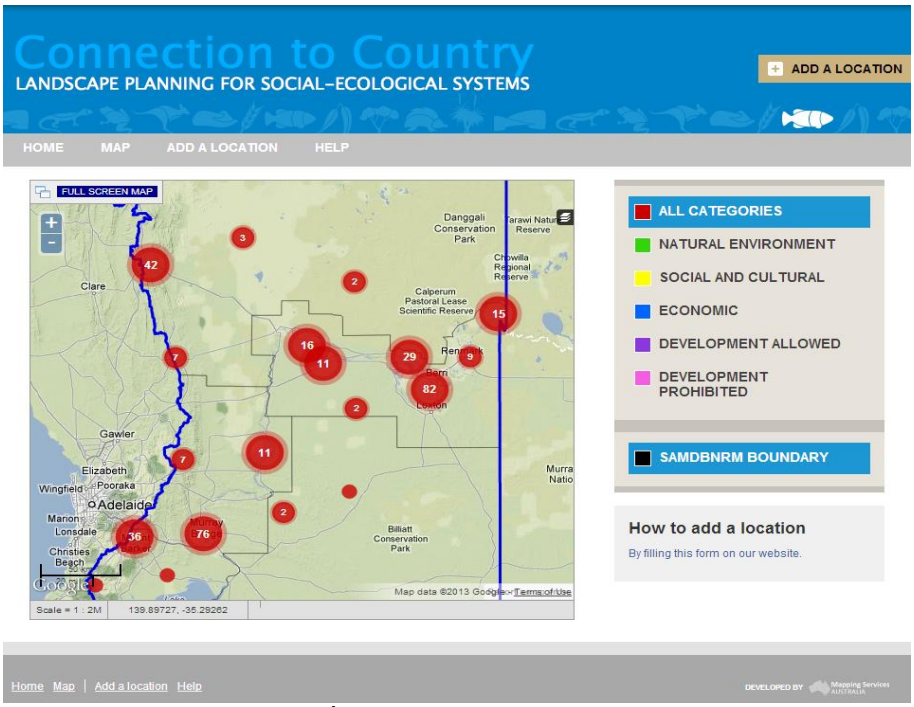
ตารางที่ 2.1 ตารางแสดงตัวอย่างข้อมูลเชิงพื้นที่ในแต่ละมาตราส่วนเวลา

มาตราส่วนเวลา	**ตัวอย่างข้อมูล**
นาที	เส้นทางการเดินทางของรถ
ชั่วโมง	ระดับน้ำในแม่น้ำ, ปริมาณรถบนถนน
วัน	ปริมาณน้ำฝน, อุณหภูมิ, เหตุการณ์ในประวัติศาสตร์
เดือน	การเกิดไฟป่า

มาตราส่วนเวลา	ตัวอย่างข้อมูล
ปี	จำนวนประชากร
ศตวรรษ	การสำรวจทางทะเล
ล้านปี	วิวัฒนาการของสิ่งมีชีวิต, การเปลี่ยนแปลงภูมิอากาศของโลก
ไม่คงที่	ยุคสมัยของการปกครอง

2.1.5 Web Mapping

Web Mapping เป็นกระบวนการของการออกแบบ, การดำเนินการ, การผลิตและการส่งแผนที่ผ่านเครือข่ายอินเทอร์เน็ต ซึ่ง Web GIS จะเหมือนกับ Web Mapping แต่ Web GIS นั้นจะเน้นไปที่การวิเคราะห์ข้อมูลทางภูมิศาสตร์ที่หลากหลายในระบบคอมพิวเตอร์ ข้อมูลทางภูมิศาสตร์ได้มาจากแผนที่ รีโมตเซนซิ่ง นอกจากนี้ยังใช้วิธีการสำรวจด้วย ซึ่งWeb GIS และ Web Mapping มักจะถูกนำมาใช้เสมือนมีความหมายเดียวกัน ถึงแม้ว่าทั้ง 2 คำจะมีความหมายที่ไม่เหมือนกัน แต่ในความเป็นจริงแล้วขอบเขตของ Web GIS และ Web Mapping นั้นยังไม่ชัดเจน จึงทำให้ทั้ง 2 คำใช้แทนความหมายเดียวกัน คือเป็นเว็บแอพพลิเคชันสำหรับแสดงข้อมูลเชิงพื้นที่บนแผนที่ ซึ่งมีตัวอย่างของ Web Mapping ดังภาพที่ 2.5 โดยมีเครื่องมือสำหรับจัดการข้อมูลเชิงพื้นที่ เช่น GeoServer ซึ่งจะกล่าวถึงในหัวข้อถัดไป



ภาพที่ 2.5 ตัวอย่าง Web Mapping

(ที่มา : http://www.mappingsa.com.au/new-on-line-tool-maps-communitys-most-treasured-spots/)

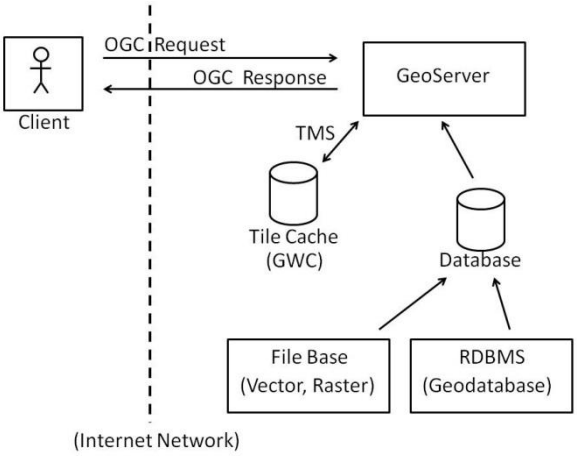
2.1.6 GeoServer [17]

GeoServer เป็น Open source software server สำหรับพัฒนาระบบแผนที่บนเว็บด้วยภาษา Java ที่มีมาตรฐานในการแบ่งปันและแก้ไขข้อมูลเชิงพื้นที่แบบออนไลน์ โดย GeoServer ได้รับการพัฒนา ทดสอบ และสนับสนุนโดยหลากหลายกลุ่มทั้งบุคคลทั่วไปและองค์กรต่าง ๆ จากทั่วโลก กล่าวคือ GeoServer เป็นเว็บเซอร์วิสสำหรับให้บริการชั้นข้อมูลทางภูมิศาสตร์

โดย Service ที่ GeoServer ให้บริการ ตามมาตรฐานของ Open Geospatial Consortium (OGC Standard) [13] มีดังนี้

- WMS (Web Map Service) คือส่วนที่ให้บริการข้อมูลในส่วนของข้อมูลภาพ ได้แก่ ภาพถ่ายดาวเทียมภาพถ่ายทางอากาศ
- WFS (Web Feature Service) คือส่วนที่ให้บริการข้อมูลในส่วนของข้อมูลที่เป็น Vector สามารถเพิ่ม, ลบและแก้ไข ข้อมูล Vector ได้
- WCS (Web Coverage Service) คือส่วนที่ให้บริการข้อมูลในส่วนของข้อมูลที่เป็น Raster
- TMS (Tile Map Service) คือส่วนที่ให้บริการข้อมูลในส่วนของข้อมูลภาพ โดยจะมีการทำ Tile Cache เพื่อเพิ่มความเร็วในการแสดงภาพ
- WMS-C (Web Mapping Service – Cached) คือส่วนที่ให้บริการข้อมูลในส่วนของข้อมูลภาพ
- WMTS (Web Map Tile Service) คือส่วนที่ให้บริการข้อมูลในส่วนของข้อมูลภาพ โดยพัฒนามาจาก TMS

2.1.6.1 การทำงานของ GeoServer



ภาพที่ 2.6 การทำงานของ GeoServer

ขั้นตอนการทำงานของ GeoServer จะเป็นไปตามภาพที่ 2.6 โดยเริ่มจาก Client เรียก Service OGC ไปที่ GeoServer เช่น เรียก Service WMS (Server ตอบกลับเป็นภาพ) จากนั้น GeoServer จะไปดึงข้อมูลใน Database มา Process เป็นภาพ แล้วส่งกลับผ่าน OGC Response ไปแสดงบน Browser ของ Client

ถ้า Client เรียกใช้ Service TMS คือจะมีการเก็บภาพลงใน Tile Cache จะช่วยให้แสดงภาพได้เร็วขึ้น หลักการทำงานของ GeoServer คือเมื่อ GeoServer ได้รับ OGC Request แล้วจะไปค้นหาภาพใน Tile Cache ก่อน

- ถ้าไม่มีใน Tile Cache ก็จะไปดึงข้อมูลใน Database แล้วมา Process เป็นภาพเก็บไว้ใน Tile Cache แล้วส่งกลับผ่าน OGC Response ไปแสดงบน Browser ของ Client
- แต่ถ้ามีใน Tile Cache ก็จะนำภาพใน Tile Cache ส่งกลับผ่าน OGC Response ไปแสดงบน Browser ของ Client เลยโดยไม่ต้องทำการ Process ก่อน ซึ่งจะทำให้ประสิทธิภาพในการใช้งาน GeoServer เพิ่มมากขึ้น

2.1.6.2 การนำชั้นข้อมูลจาก GeoServer ไปใช้งาน

ใน GeoServer จะมี Client Software เป็น API ตัวหนึ่งชื่อ OpenLayers [14] ซึ่งเป็น JavaScript สำหรับแสดงแผนที่บนเว็บแอพพลิเคชัน และจะเรียกใช้ Service ของ GeoServer เพื่อนำชั้นข้อมูลต่าง ๆ ของ GeoServer มาแสดงบนแผนที่บนเว็บแอพพลิเคชัน

2.1.6.3 การติดตั้ง GeoServer

สำหรับขั้นตอนการติดตั้ง GeoServer สามารถศึกษาเพิ่มเติมได้ในภาคผนวกท้ายเล่มวิทยานิพนธ์

2.2 งานวิจัยที่เกี่ยวข้อง

ในหัวข้อนี้จะกล่าวถึงงานวิจัยต่าง ๆ ที่เกี่ยวข้องกับการแสดงข้อมูลเชิงพื้นที่ที่มีเวลามาเกี่ยวข้อง และในส่วนท้ายจะกล่าวถึงเครื่องมือสำหรับแสดงข้อมูลในเชิงเวลาที่มีผู้พัฒนาไว้แล้ว

2.2.1 Google Earth [3, 4]

โปรแกรม Google Earth เป็นซอฟต์แวร์ที่พัฒนาโดยบริษัท Google สำหรับอำนวยความสะดวกในการค้นหาข้อมูลและแสดงข้อมูลภาพถ่ายทางอากาศพร้อมทั้งแผนที่ เส้นทาง และผังเมืองซ้อนทับลงในแผนที่ โดย Google ได้นำเอาภาพถ่ายทางอากาศและภาพถ่ายจากดาวเทียมมาผสมผสานกับเทคโนโลยี Streaming และทำการเชื่อมโยงข้อมูลจากฐานข้อมูลของ Google เพื่อนำเราไปยังจุดต่าง ๆ ที่ต้องการบนแผนที่โลกดิจิตอล

ข้อมูลที่แสดงบนโปรแกรม Google Earth จะมีทั้งข้อมูลที่เป็นจุด, เส้น, รูปภาพ, รูปหลายเหลี่ยม รวมทั้งโมเดล ซึ่งข้อมูลที่แสดงบนโปรแกรม Google Earth จะถูกเก็บอยู่ในรูปแบบของไฟล์ KML (Keyhole Markup Language) ซึ่งเป็นโครงสร้างภาษา XML รูปแบบหนึ่ง โดยไฟล์ KML มีโครงสร้างพื้นฐานเป็น Tag ที่มีชื่อกำหนดและ Attribute สำหรับระบุรายละเอียดข้อมูลของ Tag ซึ่งจะใช้สำหรับแสดงคุณลักษณะเชิงพื้นที่ของข้อมูลทางภูมิศาสตร์ โดยมี Tag ที่ใช้งานบ่อยที่พบได้ในไฟล์ KML ทั่วไป ได้แก่

- <Plackmark> เป็น Tag ที่ถูกใช้บ่อยที่สุด ซึ่งใช้สำหรับการสร้างตำแหน่งของข้อมูลทางภูมิศาสตร์บนแผนที่ โดยการแทนด้วยรูปหมุดที่ปักบนแผนที่ในตำแหน่งที่ต้องการ
- <GroundOverlay> เป็น Tag ที่ใช้สำหรับการนำรูปภาพมาวางซ้อนบนแผนที่ในตำแหน่งที่ต้องการ
- <LineString> เป็น Tag ที่ใช้สำหรับสร้างเส้นทางที่ลากเชื่อมระหว่างพิกัด (ลองติจูด, ละติจูด, ระดับความสูงจากระดับน้ำทะเล) ต่าง ๆ บนแผนที่
- <Polygon> เป็น Tag ที่ใช้สำหรับสร้างรูปหลายเหลี่ยมซึ่งแทนอาคารแบบง่ายๆและสร้างรูปร่างอื่น ๆ เช่น รูปห้าเหลี่ยมที่มีลักษณะเป็นวงแหวนห้าเหลี่ยม เป็นต้น

KML นอกจากจะมี Tag ที่ใช้สำหรับแสดงคุณลักษณะเชิงพื้นที่ของข้อมูลทางภูมิศาสตร์แล้วยังมี Tag ที่รองรับคุณลักษณะเชิงเวลาของข้อมูลทางภูมิศาสตร์อีกด้วย ได้แก่

- <TimeStamp> เป็น Tag ที่ใช้สำหรับแสดงข้อมูล ณ เวลาชั่วขณะ คือ ณ เวลาหนึ่งจะแสดงข้อมูลหนึ่ง
- <TimeSpan> เป็น Tag ที่ใช้สำหรับแสดงข้อมูลในช่วงเวลาต่าง ๆ คือ ณ ช่วงเวลาหนึ่งจะแสดงข้อมูลหนึ่ง โดยมีเวลาเริ่มต้นของช่วงนั้น ๆ และ เวลาสิ้นสุดของช่วงนั้นๆ

ในการแสดงไฟล์ KML ที่มี Tag สำหรับแสดงคุณลักษณะเชิงเวลาบนโปรแกรม Google Earth จะมีเครื่องมือในการควบคุมเวลาเพิ่มเติมขึ้นมาเรียกว่า Time Slider ซึ่งจะปรากฏอยู่ที่ตำแหน่งด้านบนซ้ายของโปรแกรม Google Earth ดังภาพที่ 2.7 ซึ่งผู้ใช้งานสามารถเลื่อนดูข้อมูลในเวลาต่าง ๆ บนเส้นเวลาได้ และ Time Slider ของ Google Earth ยังรองรับการแสดงข้อมูลแบบหลายระดับความละเอียดของเวลา คือผู้ใช้สามารถดูข้อมูลในระดับความละเอียดของเวลาต่าง ๆ ได้โดยการซูมเข้า-ออกที่รูปแว่นขยายของ Time Slider แต่ข้อมูลที่แสดงในแต่ละระดับความละเอียดของเวลาจะไม่มีการกรองข้อมูลก่อนที่จะแสดงบนแผนที่ ดังนั้นถ้าข้อมูลมีปริมาณมาก ๆ แล้วเมื่อซูมเวลาออกมา ข้อมูลที่แสดงบนแผนที่จะมีปริมาณมากตามไปด้วย และจะทำให้ไม่สามารถแสดงข้อมูลให้เป็นที่เข้าใจของผู้ใช้งานได้ ซึ่งถือว่าเป็นข้อจำกัดหนึ่งของโปรแกรม Google Earth

ภาพที่ 2.7 Time Slider ของ Google Earth

2.2.2 Challenges in Interactive Time-Based Information Visualization [7]

งานวิจัยนี้พัฒนาระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์ในเชิงเวลา (Time-based) ในรูปแบบ Web-based คือ www.conflicthistory.com ดังภาพที่ 2.8 ซึ่งเป็นเว็บไซต์ที่แสดงข้อมูลความขัดแย้งระหว่างประเทศต่าง ๆ ในช่วงเวลาต่าง ๆ โดยที่ Timeline ของงานวิจัยนี้มีข้อจำกัดคือมีระดับความละเอียดของเวลาแค่ระดับเดียวคือปี ทำให้ไม่สามารถดูข้อมูลในระดับความละเอียดอื่น ๆ ได้ เช่น หากผู้ใช้ต้องการดูข้อมูลความขัดแย้งระหว่างประเทศในแต่ละเดือนหรือในแต่ละวัน Timeline ของเว็บไซต์นี้จะไม่สามารถเปลี่ยนระดับความละเอียดของเวลาไปเป็นเดือนหรือวันได้ และงานวิจัยนี้กล่าวถึงความท้าทายในการแสดงข้อมูลในเชิงเวลาคือ ข้อมูลที่นำมาแสดงนั้นมาจากหลายแหล่งข้อมูลทำให้ข้อมูลมีรูปแบบที่ไม่สอดคล้องกัน จึงจำเป็นต้องจัดข้อมูลให้อยู่ในรูปแบบเดียวกันก่อน และการแสดงข้อมูลเชิงพื้นที่ในหลายมิติเป็นอีกความท้าทายในการแสดงข้อมูล ซึ่งในงานวิจัยนี้ได้จัดหมวดหมู่ของข้อมูลเชิงพื้นที่ที่มีเวลามาเกี่ยวข้องไว้โดยเป็นการรวมกันของข้อมูลเชิงพื้นที่ในจำนวนมิติต่าง ๆ กับเวลาในช่วงเวลาต่าง ๆ

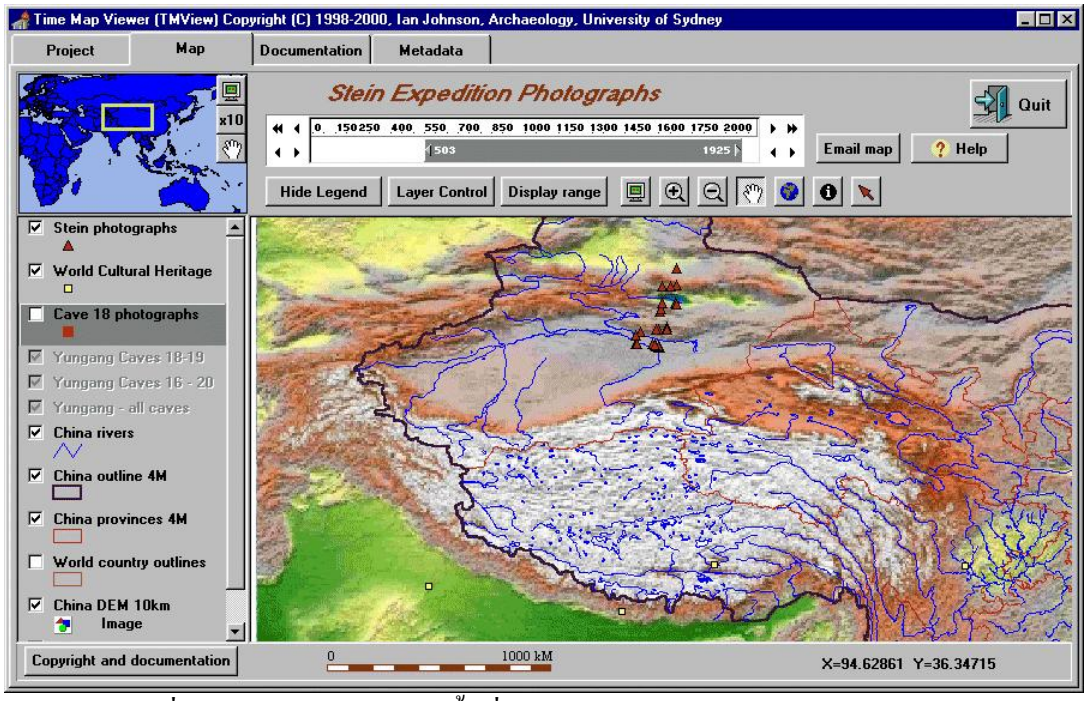
ภาพที่ 2.8 การแสดงข้อมูลความขัดแย้งระหว่างประเทศต่างๆ ในงานวิจัย [7]
(ที่มา : http://www.conflicthistory.com)

2.2.3 The TimeMap Project: Developing Time-Based GIS Display for Cultural Data [6]

งานวิจัยของ TimeMap Project นี้ได้พัฒนาวิธีการในการจัดการข้อมูลเชิงพื้นที่ที่มีความเกี่ยวข้องกับเวลา โดยใช้ข้อมูลทางวัฒนธรรม ซึ่งแบ่งงานเป็น 3 ขั้นตอน คือ

1. การเก็บข้อมูลทางวัฒนธรรมที่มีเวลามาเกี่ยวข้อง ซึ่งเป็นข้อมูลทางภูมิศาสตร์ประเภทเวกเตอร์ (จุด, เส้น, พื้นที่)
2. การสร้างส่วนปฏิสัมพันธ์กับผู้ใช้ (Interface) สำหรับแสดงแผนที่ที่มีเวลามาเกี่ยวข้อง (Time-based map)
3. การสร้างภาพเคลื่อนไหว (Animation) เพื่อแสดงข้อมูลในเวลาต่าง ๆ

ซึ่งในงานวิจัยของ TimeMap Project ได้ใช้ TimeMap Data Viewer (TMView) ในการแสดงข้อมูลเชิงพื้นที่ที่มีเวลามาเกี่ยวข้องดังภาพที่ 2.9 โดยข้อมูลต่าง ๆ จะจัดเก็บลงในตารางฐานข้อมูลที่อยู่ใน TimeMap Clearing Browser แล้วถูกดึงมาแสดงผลบนแผนที่ในช่วงเวลาต่าง ๆ ตามช่วงเวลาที่เลือกบน Timeline ซึ่ง Timeline ของ TimeMap Project นี้มีข้อจำกัดคือไม่รองรับหลายระดับความละเอียดของเวลา คือผู้ใช้ไม่สามารถดูข้อมูลในระดับความละเอียดของเวลาอื่น ๆ ได้



ภาพที่ 2.9 การแสดงข้อมูลเชิงพื้นที่บน TimeMap Data Viewer ในงานวิจัย [6]

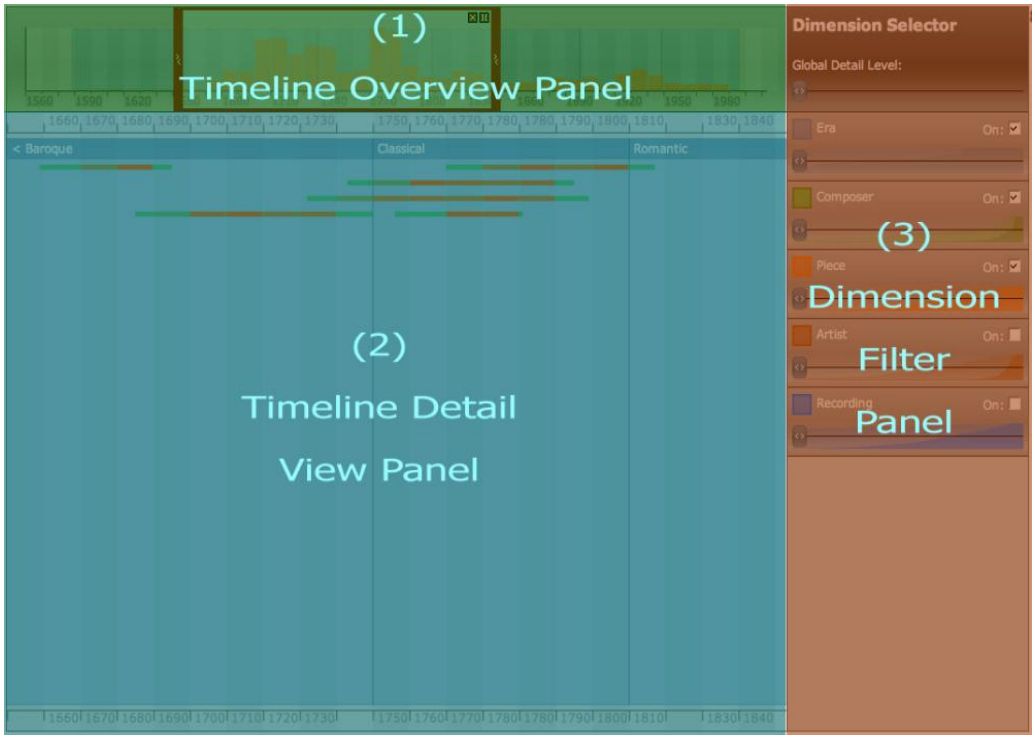
2.2.4 Continuum: designing timelines for hierarchies, relationships and scale [1]

งานวิจัยนี้พัฒนาเครื่องมือสำหรับแสดงข้อมูลที่เกี่ยวข้องกับเวลาในรูปแบบของ Timeline ชื่อว่า “Continuum ” ซึ่งเป็นการแสดงข้อมูลเกี่ยวกับประวัติการแต่งเพลงที่ถูกแต่งโดยศิลปินในช่วงเวลาต่าง ๆ โดยจะแบ่งข้อมูลเป็นลำดับชั้นคือ ช่วงเวลา, ผู้แต่งเพลง, เครื่องดนตรี และเพลง

โดยข้อมูลทั้งหมดจะแสดงเป็นลำดับชั้นแบบไดนามิก คือผู้ใช้สามารถปรับเปลี่ยนปริมาณข้อมูลของแต่ละชั้นได้ และยังสามารถเลือกแสดงหรือไม่แสดงข้อมูลบางชั้นข้อมูลได้ ดังนั้นจึงสามารถรองรับข้อมูลในปริมาณมาก ๆ ได้โดยไม่ต้องแสดงข้อมูลทั้งหมดลงในแผนที่พร้อมกัน และเครื่องมือนี้สามารถแสดงข้อมูลในเชิงเวลาที่มีความสัมพันธ์ระหว่างกันข้ามชั้นข้อมูลได้ อีกทั้งยังสามารถซูม (Zoom) ดูข้อมูลในแต่ละชั้นข้อมูลได้ แต่ในงานวิจัยนี้ยังมีข้อจำกัดในส่วนของ Timeline คือไม่รองรับหลายระดับความละเอียดของเวลา ทำให้ไม่สามารถดูข้อมูลในระดับความละเอียดของเวลาอื่น ๆ ได้

เครื่องมือนี้จะแบ่งพื้นที่แสดงผลเป็น 3 ส่วนดังภาพที่ 2.10 คือ

1. ส่วนสำหรับแสดง Timeline (Timeline Overview Panel)
2. ส่วนสำหรับแสดงรายละเอียดของข้อมูลใน Timeline (Timeline Detail View Panel)
3. ส่วนสำหรับคัดกรองข้อมูลที่จะใช้แสดงผล (Dimension Filter Panel)

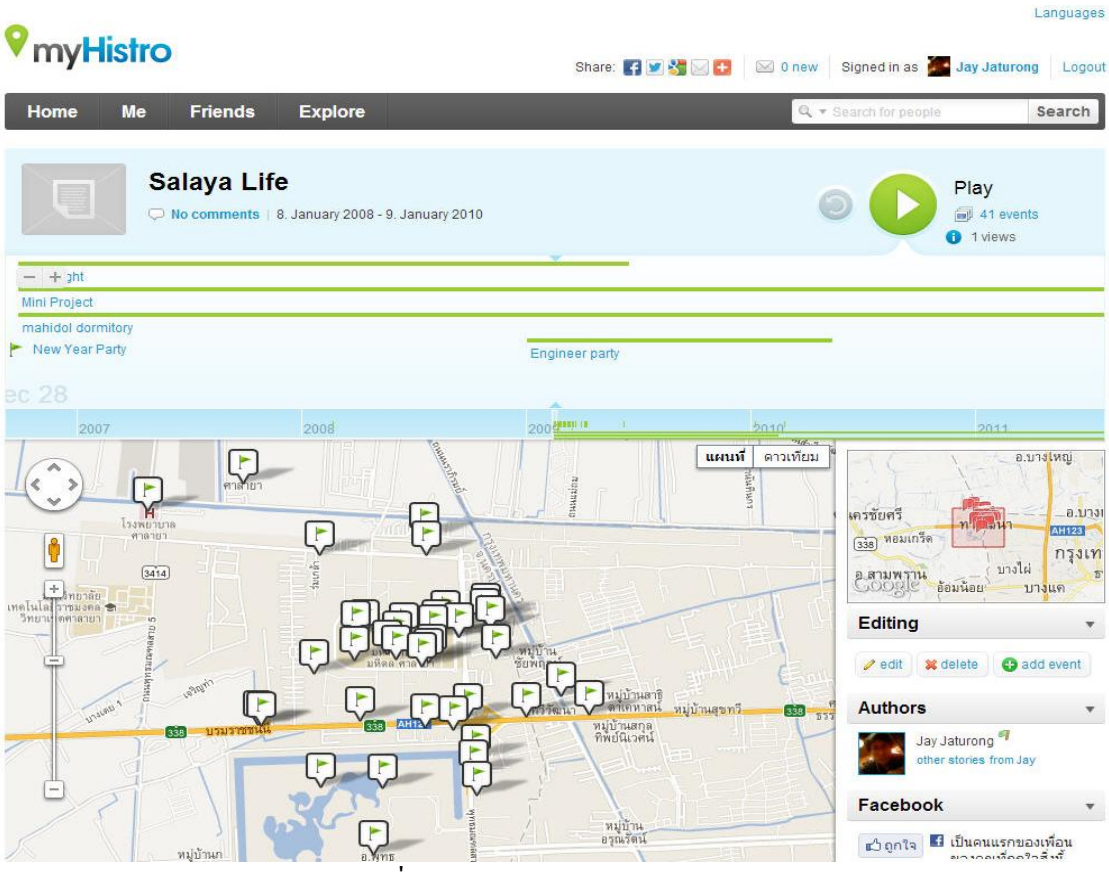


ภาพที่ 2.10 ส่วนแสดงผลของ Continuum ในงานวิจัย [1]

ในงานวิจัยนี้พัฒนาระบบสำหรับแสดงข้อมูลทางด้านดนตรีที่มี Timeline ซึ่งพัฒนาขึ้นมาเอง โดยไม่ได้แสดงข้อมูลบนแผนที่ เนื่องจากข้อมูลที่เก็บมาไม่มีตำแหน่งของที่ที่แต่งเพลงมาด้วย ดังนั้นระบบนี้จึงไม่มีการนำแผนที่มาใช้ร่วมกับ Timeline

2.2.5 myHistro [5]

myHistro เป็นเว็บไซต์ที่แสดงข้อมูลลำดับเหตุการณ์ของบุคคลว่าเคยไปสถานที่ใดเวลาใดบ้าง โดยจะแสดงพิกัดหมุดตำแหน่งสถานที่ที่เคยไปบนแผนที่และใช้ Timeline มาแสดงข้อมูลในด้านเวลาดังภาพที่ 2.11 ซึ่ง Timeline ของเว็บไซต์นี้สามารถเปลี่ยนระดับความละเอียดของเวลาได้ เช่น เปลี่ยนจากระดับเดือนเป็นระดับวัน หรือ เปลี่ยนจากระดับวันเป็นระดับเดือน เป็นต้น แต่ในด้านการแสดงข้อมูลบนแผนที่นั้นไม่มีการกรองข้อมูลสำหรับแสดงผลในแต่ละระดับความละเอียดของเวลา ดังนั้นหากข้อมูลมีเป็นปริมาณมากๆ จะทำให้ในแต่ละระดับความละเอียดของเวลาจะมีข้อมูลแสดงบนแผนที่เป็นปริมาณมากตามไปด้วยดังที่เห็นในแผนที่ ทำให้ไม่สามารถแสดงข้อมูลให้เป็นที่เข้าใจของผู้ใช้งานได้ ซึ่งถ้ามีการคัดกรองข้อมูลสำหรับแสดงบนแผนที่ในแต่ละระดับความละเอียดของเวลา จะสามารถแก้ไขปัญหาการแสดงข้อมูลในปริมาณมากบนแผนที่ดังกล่าวได้ ตัวอย่างเช่น ถ้าเปลี่ยนระดับความละเอียดของเวลาจากวันเป็นเดือน อาจจะเลือกข้อมูลในตำแหน่งสถานที่เคยมาซ้ำหลายครั้งมาแสดง และถ้าเปลี่ยนระดับความละเอียดของเวลาจากเดือนเป็นปี ปริมาณข้อมูลก็จะมีมากขึ้นกว่าระดับเดือน ดังนั้นอาจจะเลือกข้อมูลในตำแหน่งสถานที่เคยมาซ้ำบ่อยครั้งกว่าในระดับเดือนมาแสดงก็ได้



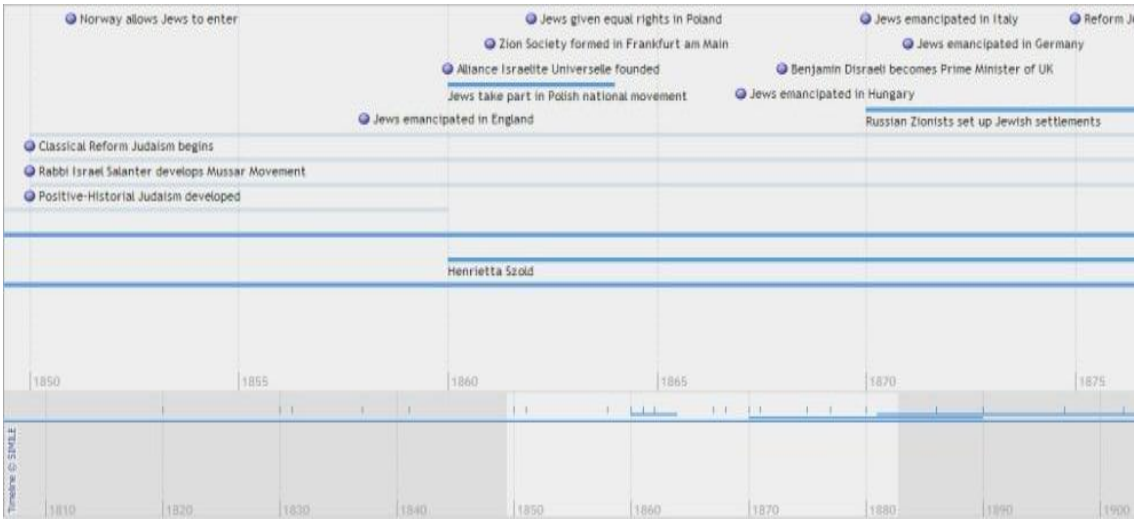
ภาพที่ 2.11 เส้นเวลาของ myHistro [5]

2.2.6 เครื่องมือสำหรับแสดงข้อมูลในเชิงเวลาที่มีอยู่ในปัจจุบัน

เครื่องมือที่ใช้แสดงข้อมูลในเชิงเวลามีอยู่หลากหลายเครื่องมือที่มีผู้พัฒนาไว้แล้ว เช่น SIMILE Timeline [9], ProPublica's Timeline Setter [16] และ Timeglider [12] เป็นต้น ซึ่งเครื่องมือเหล่านี้ได้ออกแบบมาเพื่อแสดงลำดับเหตุการณ์ของเรื่องราว โดยจะแสดงเหตุการณ์เป็นข้อมูลที่มีหัวข้อใจความสั้น ๆ ดังนั้นเครื่องมือเหล่านี้จึงไม่รองรับการทำงานร่วมกับ Web Mapping กล่าวคือ Timeline ดังกล่าวไม่สามารถแสดงลำดับเหตุการณ์ของข้อมูลบนแผนที่บนเว็บแอพพลิเคชันได้ ซึ่งเป็น Timeline ที่ใช้สำหรับแสดงข้อมูลเหตุการณ์ที่เป็นข้อความหรือรูปภาพ แต่เนื่องจากข้อมูลเหตุการณ์บางเหตุการณ์มีตำแหน่งพิกัดมาเกี่ยวข้อง ดังนั้นในการแสดงข้อมูลเป็นข้อความจะทำให้ไม่สามารถมองเห็นภาพของตำแหน่งของข้อมูลได้ จึงจำเป็นต้องแสดงข้อมูลดังกล่าวบนแผนที่ ซึ่งมีหน่วยงาน TimeMap Open Source Consortium ได้พัฒนาให้ SIMILE Timeline สามารถใช้งานร่วมกับ Web Mapping ได้ ซึ่งจะอธิบายในลำดับถัดไป

2.2.6.1 SIMILE Timeline

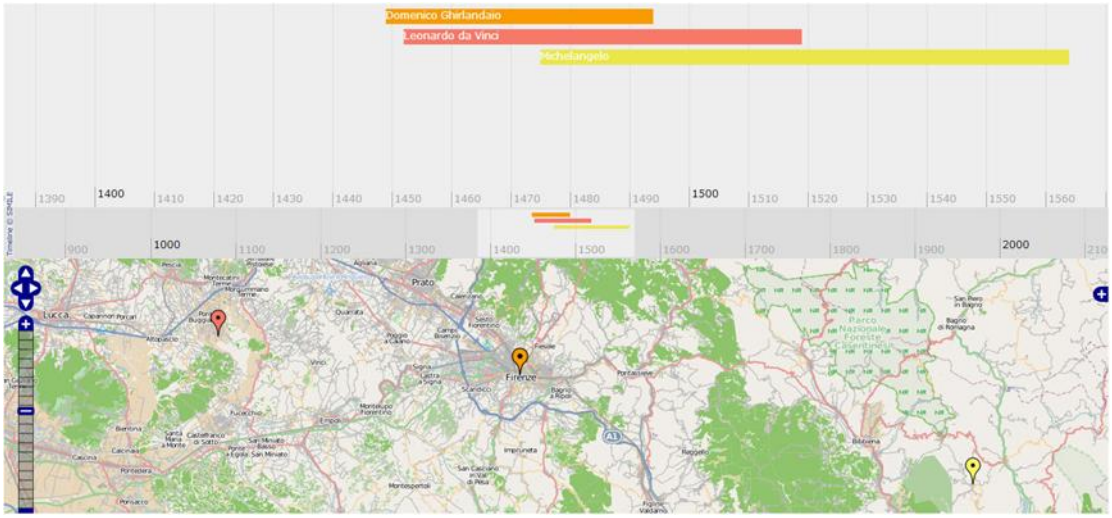
SIMILE Timeline เป็นเครื่องมือที่ใช้แสดงข้อมูลเชิงเวลาที่พัฒนาขึ้นโดย Massachusetts Institute of Technology (MIT) ซึ่งเป็นเครื่องมือที่เว็บไซต์ส่วนใหญ่นิยมนำมาพัฒนาและประยุกต์ใช้มากที่สุด เนื่องจากมีความสวยงาม และเป็นซอฟต์แวร์แบบเปิด (Open source software) แต่ SIMILE Timeline ยังมีข้อจำกัดบางประการ คือรองรับแค่ระดับความละเอียดของเวลาเดียว จากภาพที่ 2.12 จะเห็นแถบล่างของ Timeline มีระดับความละเอียดของเวลาเป็น 100ปี และในแถบบนของ Timeline มีระดับความละเอียดของเวลาเป็น 10 ปี ซึ่งจะเห็นได้ว่ามีเป็น 2 ระดับความละเอียดของเวลา แต่เนื่องจากทั้ง 2 ระดับความละเอียดของเวลาที่แสดงไม่สามารถเปลี่ยนระดับความละเอียดของเวลาไปมาได้ ซึ่งหมายความว่า SIMILE Timeline มีแค่ 1 ระดับความละเอียดของเวลาคือระดับความละเอียดของเวลาในแถบบน ส่วนระดับความละเอียดของเวลาในแถบล่างคือจะย่อระดับความละเอียดของเวลาของแถบบนเพื่อให้เห็นลำดับเหตุการณ์ข้อมูลได้กว้างขึ้น อีกทั้งในการแสดง Timeline จะมีระดับความละเอียดของเวลาที่ตายตัว ซึ่งไม่สามารถเปลี่ยนระดับความละเอียดของเวลาเป็นระดับอื่น ๆ ได้



ภาพที่ 2.12 SIMILE Timeline

(ที่มา : http://www.simile-widgets.org/timeline/examples/religions/jewish-history.html)

ในปัจจุบันมีผู้พัฒนาให้ SIMILE Timeline สามารถใช้งานร่วมกับแผนที่ได้เพื่อใช้สำหรับการแสดงข้อมูลเชิงพื้นที่ที่มีเวลามาเกี่ยวข้อง คือ TimeMap Open Source Consortium [19] ได้พัฒนา Timemap.js ซึ่งเป็น JavaScript library แบบ Open Source ขึ้นมาเพื่อช่วยให้สามารถใช้ SIMILE timeline แสดงลำดับเหตุการณ์ของข้อมูลบนแผนที่บนเว็บแอพพลิเคชันได้ดังภาพที่ 2.13 ด้านบนจะเป็นส่วนของ SIMILE Timeline และด้านล่างจะเป็นแผนที่สำหรับแสดงข้อมูลเหตุการณ์ในช่วงเวลาต่าง ๆ ดังนั้น SIMILE timeline จึงเป็นเครื่องมือที่รองรับการทำงานกับ Web Mapping โดยอาศัย Timemap.js



ภาพที่ 2.13 TimeMap

(ที่มา : http://timemap.googlecode.com/svn/tags/2.0.1/examples/basic_openlayers.html)

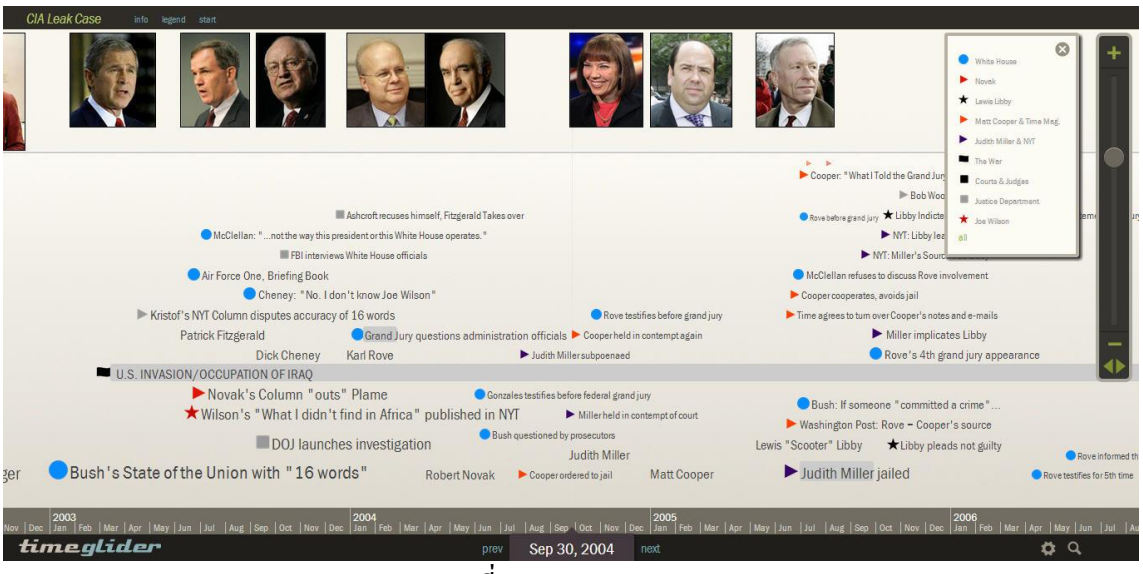
2.2.6.2 ProPublica's Timeline Setter

TimelineSetter เป็นซอฟต์แวร์แบบเปิด (Open source software) ซึ่งเป็นเครื่องมือสำหรับแสดงข้อมูลในเชิงเวลา โดยข้อมูลที่นำมาแสดงบน Timeline นี้จะอยู่ในรูปแบบของไฟล์ .CSV แต่ TimelineSetter ยังคงมีข้อจำกัดเช่นเดียวกับ SIMILE Timeline คือรองรับแค่ระดับความละเอียดของเวลาเดียว จากภาพที่ 2.14 เป็น TimelineSetter ที่รองรับระดับความละเอียดของเวลาในระดับวัน ซึ่งไม่สามารถเปลี่ยนเป็นระดับความละเอียดของเวลาอื่น ๆ ได้ อีกทั้ง TimelineSetter ยังไม่ถูกนำไปพัฒนาให้ใช้งานร่วมกับ Web mapping ได้ ในปัจจุบัน TimelineSetter จึงยังไม่ค่อยเป็นที่นิยมใช้งาน

ภาพที่ 2.14 TimelineSetter [16]

2.2.6.3 Timeglider

Timeglider เป็นเครื่องมือสำหรับแสดงข้อมูลในเชิงเวลา ที่รองรับหลายระดับความละเอียดของเวลา แต่เป็นซอฟต์แวร์ที่เสียค่าใช้จ่าย (Commercial software) ซึ่ง Timeglider มีข้อจำกัดในด้านการดูข้อมูลเหตุการณ์ คือสามารถเลือกดูข้อมูลได้แค่เวลาเดียว ซึ่งไม่สามารถเลือกดูข้อมูลเป็นช่วงเวลาได้ ดังภาพที่ 2.15 จะแสดงข้อมูล ณ วันที่ 30 กันยายน 2547 ซึ่งผู้ใช้ไม่สามารถเลือกดูข้อมูลเป็นช่วงเวลา เช่น วันที่ 1 กันยายน 2547 – วันที่ 30 กันยายน 2547 ได้ อีกทั้ง Timeglider ยังไม่ถูกนำไปพัฒนาให้ใช้งานร่วมกับ Web mapping ได้ ในปัจจุบัน Timeglider จึงยังไม่ค่อยเป็นที่นิยมใช้งาน



ภาพที่ 2.15 Timeglider

(ที่มา : http://timeglider.com/timeline/line_cialeak)

จากงานวิจัยที่ผ่านมา จะพบว่าเว็บแอพพลิเคชันและเครื่องมือที่รองรับการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาที่มีอยู่ในปัจจุบัน มีบางเว็บแอพพลิเคชันที่รองรับหลายระดับความละเอียดของเวลา แต่เว็บแอพพลิเคชันทั้งหมดยังไม่รองรับการคัดเลือกหรือคำนวณข้อมูลมาแสดงในแต่ละระดับความละเอียดของเวลา ซึ่งหากข้อมูลมีปริมาณมากจะทำให้ไม่สามารถแสดงข้อมูลให้เป็นที่เข้าใจของผู้ใช้งานได้ ดังนั้นในงานวิจัยของวิทยานิพนธ์ฉบับนี้จึงพัฒนาระบบสำหรับการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาที่รองรับทั้งหลายระดับความละเอียดของเวลาและรองรับการคัดเลือกหรือคำนวณหาตัวแทนข้อมูลมาแสดงในแต่ละระดับความละเอียดของเวลาเพื่อควบคุมปริมาณข้อมูลที่แสดงบนแผนที่ไม่ให้ล้นหน้าจอแสดงผล

บทที่ 3
แนวคิดและการออกแบบระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา

เมื่อกล่าวถึงการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาบนแผนที่บนเว็บแอพพลิเคชัน จะนึกถึงเครื่องมือที่ช่วยในการแสดงข้อมูลเชิงเวลาคือ เส้นเวลา (Timeline) ดังนั้นในการแสดงข้อมูลเชิงเวลาจำเป็นต้องมี Timeline เพื่อช่วยในการจัดลำดับเหตุการณ์ของข้อมูลที่แสดง โดยในงานวิจัยนี้ได้พัฒนาระบบต้นแบบที่มี Timeline ที่รองรับหลายระดับความละเอียดของเวลา และระบบรองรับการคัดเลือกหรือคำนวณข้อมูลมาแสดงในแต่ละระดับความละเอียดของเวลาเพื่อควบคุมปริมาณข้อมูลที่แสดงบนแผนที่ในแต่ละระดับความละเอียดของเวลา โดยเริ่มต้นจากการแบ่งประเภทของข้อมูลที่จะใช้เป็นข้อมูลตัวอย่างในระบบต้นแบบของงานวิจัย จากนั้นจะอธิบายแนวคิดในการแสดงข้อมูลหลายระดับความละเอียดในเชิงเวลาของข้อมูลตัวอย่างโดยสร้างเป็นพีระมิดข้อมูลเชิงเวลา และแนวคิดในการแสดงข้อมูลหลายระดับความละเอียดในเชิงพื้นที่และเวลา รวมถึงแนวคิดในการออกแบบ Timeline โดยใช้แนวคิดของพีระมิดข้อมูล และในส่วนท้ายของบทจะกล่าวถึงการออกแบบระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา

3.1 ประเภทของข้อมูล

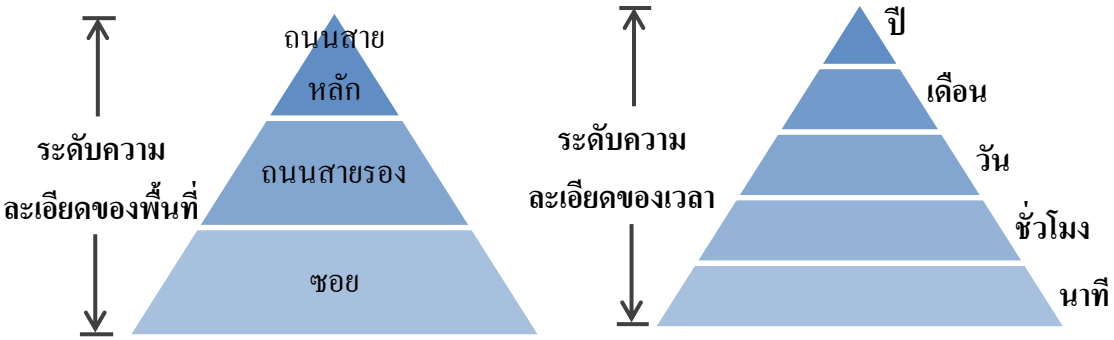
ข้อมูลตัวอย่างที่นำมาใช้ในงานวิจัยสามารถแบ่งออกได้เป็น 2 กลุ่ม คือ

1. กลุ่มที่มีตำแหน่งไม่คงที่ เช่น ข้อมูลแผ่นดินไหว, ข้อมูลจากจีพีเอส เป็นต้น ซึ่งข้อมูลประเภทนี้จะมีจำนวนตำแหน่งข้อมูลที่แสดงบนแผนที่เป็นจำนวนมากมายตามจำนวนการเกิดเหตุการณ์และข้อมูลจะไม่ค่อยเกิดซ้ำตำแหน่งเดิม ซึ่งเป็นผลให้เกิดข้อมูลล้นหน้าจอในการแสดงข้อมูลบนแผนที่
2. กลุ่มที่มีตำแหน่งคงที่ เช่น ข้อมูลอุณหภูมิ, ข้อมูลปริมาณน้ำฝน เป็นต้น ซึ่งข้อมูลประเภทนี้จะมีจำนวนตำแหน่งข้อมูลที่แสดงบนแผนที่ในแต่ละช่วงเวลาที่คงที่ แต่จำนวนข้อมูลในแต่ละตำแหน่งจะมีเป็นปริมาณมากตามช่วงเวลาที่เก็บข้อมูล ซึ่งเป็นปัญหาในการแสดงข้อมูลบนแผนที่ว่าจะแสดงข้อมูลหลายข้อมูล ณ ตำแหน่งเดียวกันอย่างไร

ในงานวิจัยนี้ได้เลือกใช้ข้อมูลแผ่นดินไหว และข้อมูลอุณหภูมิ ซึ่งเป็นตัวแทนของข้อมูลทั้ง 2 กลุ่มมาเป็นข้อมูลตัวอย่างในการพัฒนาระบบต้นแบบ ในการแสดงข้อมูลทั้ง 2 ข้อมูลมีแนวคิดในการแสดงข้อมูลเป็นพีระมิดข้อมูลซึ่งจะอธิบายในหัวข้อถัดไป

3.2 แนวคิดในการแสดงข้อมูลหลายระดับความละเอียดในเชิงเวลา

ในการแสดงข้อมูลเชิงเวลา ผู้วิจัยได้ประยุกต์ใช้เทคนิค Multi-resolution ของข้อมูลเชิงพื้นที่มาใช้กับข้อมูลเชิงเวลา เพื่อลดปริมาณการแสดงข้อมูลบนแผนที่ โดยจะสร้างเป็นพีระมิดข้อมูลตามความละเอียดของเวลา โดยในชั้นล่างของพีระมิดข้อมูลจะมีข้อมูลอยู่ปริมาณมาก และในชั้นบนของพีระมิดข้อมูลจะมีข้อมูลอยู่ปริมาณน้อยลงตามลำดับ ซึ่งการเก็บข้อมูลในลักษณะนี้เหมือนรูปพีระมิด จึงเรียกว่าพีระมิดข้อมูล ดังภาพที่ 3.1



(ก) (ข)

ภาพที่ 3.1 การแสดงข้อมูลแบบพีระมิดข้อมูล
(ก) ข้อมูลเชิงพื้นที่ (ข) ข้อมูลในเชิงเวลา

จากภาพที่ 3.1 (ก) เป็นการแสดงข้อมูลเชิงพื้นที่โดยใช้เทคนิค Multi-resolution ของข้อมูลถนน เสมือนกำลังดูอยู่บนแผนที่ เมื่อเปิดแผนที่ขึ้นมาจะแสดงเฉพาะถนนสายหลัก เมื่อซูมแผนที่เข้าไปจะพบถนนสายรองปรากฏขึ้นมา และเมื่อซูมแผนที่เข้าไปอีกจะพบซอยปรากฏขึ้นมา ซึ่งการแสดงข้อมูลในลักษณะนี้คือการแสดงข้อมูลแบบ Multi-resolution เพื่อหลีกเลี่ยงการแสดงข้อมูลทั้งหมดในปริมาณมาก ๆ บนแผนที่ซึ่งจะทำให้ไม่สามารถแสดงข้อมูลให้เป็นที่เข้าใจของผู้ใช้งานได้

และภาพที่ 3.1 (ข) เป็นการแสดงข้อมูลในเชิงเวลาโดยใช้เทคนิค Multi-resolution โดยที่ชั้นล่างสุดแทนระดับความละเอียดของเวลาที่ละเอียดสูงสุดคือนาที และในชั้นบนแทนระดับความละเอียดของเวลาที่ละเอียดต่ำสุดคือปี โดยข้อมูลในชั้นนาทีคือข้อมูลดิบที่เก็บมาในทุกนาที และข้อมูลในชั้นบนขึ้นไป เช่น ชั้นชั่วโมง, วัน, เดือน และปี จะได้มาจากการประมวลผลข้อมูลในชั้นนาทีให้ได้เป็นข้อมูลในชั้นชั่วโมง, วัน, เดือน และปี ซึ่งจะทำให้ข้อมูลในชั้นบนมีปริมาณข้อมูลน้อยกว่าชั้นล่าง

ระดับความละเอียดของเวลาอาจจะมีนอกเหนือจากในภาพที่ 3.1 (ข) เช่น วินาที, ทศวรรษ, ศตวรรษ, สหัสวรรษ รวมถึงระดับความละเอียดที่ช่วงเวลาไม่คงที่ เช่น ยุค, สมัย เป็นต้น โดยในชั้นล่างสุดของพีระมิดข้อมูลสามารถเปลี่ยนเป็นระดับความละเอียดของเวลาที่ละเอียดมากกว่านาทีได้ เช่น วินาที, มิลลิวินาที, ไมโครวินาที เป็นต้น และในชั้นบนสุดของพีระมิดข้อมูลสามารถเปลี่ยนเป็นระดับความละเอียดของเวลาที่ละเอียดต่ำกว่าปีได้ เช่น ทศวรรษ, ศตวรรษ, สหัสวรรษ เป็นต้น

ในการสร้างพีระมิดข้อมูลของข้อมูลทั้ง 2 ข้อมูลคือ ข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ มีขั้นตอนในการสร้างพีระมิดข้อมูลเชิงเวลาดังนี้

1. เริ่มต้นจากชั้นล่างสุดที่มีความละเอียดสูงสุด และมีปริมาณข้อมูลมากที่สุด อาจเป็นข้อมูลทั้งหมดที่ได้เก็บข้อมูลมา เนื่องจากข้อมูลตั้งต้นที่ได้มามักมีระดับความละเอียดของเวลาเดียว เช่น ข้อมูลทุกชั่วโมง เป็นต้น ดังนั้นจึงเก็บไว้ในชั้นล่างสุดของพีระมิดข้อมูล
2. ในชั้นข้อมูลที่สูงขึ้นจะได้มาจากการนำเอาข้อมูลมาประมวลผลตามเกณฑ์บางอย่าง ซึ่งนอกจากใช้วิธีการเลือกจากข้อมูลที่มีอยู่ เช่น เลือกจุดที่มีค่ามากถึงระดับหนึ่ง (แผ่นดินไหวที่มีความรุนแรงเกิน 7 ริกเตอร์) ยังอาจต้องทำการคำนวณแทน เช่น ค่าเฉลี่ย (อุณหภูมิเฉลี่ยตลอดเดือน)

โดยมีเกณฑ์ในการพิจารณาเลือกหรือคำนวณข้อมูลของข้อมูลประเภทต่าง ๆ มาแสดงบนแผนที่ในแต่ละระดับความละเอียดของเวลาดังตารางที่ 3.1 และมีตัวอย่างการเลือกหรือคำนวณข้อมูลของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิดังตารางที่ 3.2

ตารางที่ 3.1 ตารางแสดงตัวอย่างเกณฑ์ในการคัดเลือกหรือคำนวณข้อมูลมาแสดงบนแผนที่ของข้อมูลประเภทต่างๆ

ประเภทข้อมูล	**วิธีการคัดเลือกหรือคำนวณข้อมูล**
ข้อมูลแผ่นดินไหว	- ระดับความรุนแรงของแผ่นดินไหว ที่ระดับความละเอียดของเวลาสูง แสดงข้อมูลแผ่นดินไหวตั้งแต่ระดับความรุนแรงน้อย ๆ ที่ระดับความละเอียดของเวลาต่ำ แสดงข้อมูลตั้งแต่ระดับความรุนแรงมาก ๆ

ประเภทข้อมูล	วิธีการคัดเลือกหรือคำนวณข้อมูล
ข้อมูลการเดินทางที่บันทึกด้วย GPS	- เลือกตำแหน่งที่ช่วงเวลาคงที่ เช่น ถ้าอยู่ในระดับความละเอียดของเวลาเป็นชั่วโมงจะเลือกข้อมูล ณ เวลา 0.00 น., 1.00 น., 2.00 น. … ถ้าระดับความละเอียดของเวลาเป็นวัน อาจเลือกข้อมูลที่เวลา 12.00 น. ของแต่ละวัน เป็นต้น
ข้อมูลการจราจร	- ระดับความหนาแน่นของรถยนต์ในช่วงเวลา - ความเร็วเฉลี่ย
ข้อมูลน้ำท่วม	- ระดับความสูงของน้ำท่วม - ระยะเวลาที่น้ำท่วม - ความถี่ในการเกิดน้ำท่วม
ข้อมูลอุณหภูมิ	- อุณหภูมิเฉลี่ย ต่ำสุด หรือสูงสุด
ข้อมูลปริมาณน้ำฝน	- ปริมาณน้ำฝนเฉลี่ย

ตารางที่ 3.2 ตารางแสดงตัวอย่างการเลือกหรือคำนวณข้อมูลของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ

ประเภทข้อมูล	**ระดับความละเอียดของเวลา**	**วิธีในการคัดเลือกหรือคำนวณข้อมูล**
ข้อมูลแผ่นดินไหว	วัน	แสดงทุกข้อมูลที่เก็บมาเป็นรายวัน
	เดือน	ระดับความรุนแรงของแผ่นดินไหวมากกว่า 5 ริกเตอร์
	ปี	ระดับความรุนแรงของแผ่นดินไหวมากกว่า 6 ริกเตอร์
ข้อมูลอุณหภูมิ	วัน	อุณหภูมิรายวัน
	เดือน	อุณหภูมิเฉลี่ยต่อเดือน
	ปี	อุณหภูมิเฉลี่ยต่อปี

ตารางที่ 3.2 เป็นตัวอย่างของการคัดเลือกหรือคำนวณข้อมูลของข้อมูลแผ่นดินไหวและอุณหภูมิ ซึ่งข้อมูลแผ่นดินจะใช้วิธีการคัดเลือกข้อมูลโดยพิจารณาจากระดับความรุนแรงของแผ่นดินไหวในการสร้างชั้นพีระมิดข้อมูล โดยข้อมูลเก็บมาเป็นรายวันดังนั้นจึงเก็บอยู่ในระดับวัน และในระดับเดือนจะคัดเลือกข้อมูลจากระดับวันโดยเลือกข้อมูลที่มีระดับความรุนแรงของแผ่นดินไหวมากกว่า 5 ริกเตอร์ และในระดับปีซึ่งจะมีข้อมูลมากกว่าระดับเดือน ดังนั้นจึงต้องมี

เกณฑ์ในการเลือกข้อมูลที่สูงกว่าระดับเดือน โดยเลือกข้อมูลที่มีระดับความรุนแรงของแผ่นดินไหวมากกว่า 6 ริกเตอร์

สำหรับข้อมูลอุณหภูมิจะใช้วิธีการคำนวณอุณหภูมิในการสร้างชั้นพีระมิดข้อมูล โดยข้อมูลเก็บมาเป็นรายวันดังนั้นจึงเก็บอยู่ในระดับวัน และในระดับเดือนปีจะเป็นการคำนวณหาอุณหภูมิเฉลี่ยในแต่ละเดือนและอุณหภูมิเฉลี่ยในแต่ละปีตามลำดับ

จากตัวอย่างการคัดเลือกหรือคำนวณข้อมูลของข้อมูลแผ่นดินไหวและอุณหภูมิเป็นเพียงตัวอย่างในการคัดเลือกหรือคำนวณข้อมูล ซึ่งสามารถเปลี่ยนแปลงเกณฑ์ในการคัดเลือกหรือคำนวณข้อมูลได้ และสามารถประยุกต์ใช้แนวคิดในข้างต้นเพื่อสร้างชั้นข้อมูลอื่นๆ โดยสามารถตั้งค่าในการสร้างชั้นข้อมูลได้ดังนี้

1. เลือกคอลัมน์ของข้อมูลที่จะใช้ในเป็นเกณฑ์ในการเลือกหรือคำนวณ โดยข้อมูลในคอลัมน์นี้จะต้องเป็นตัวเลข

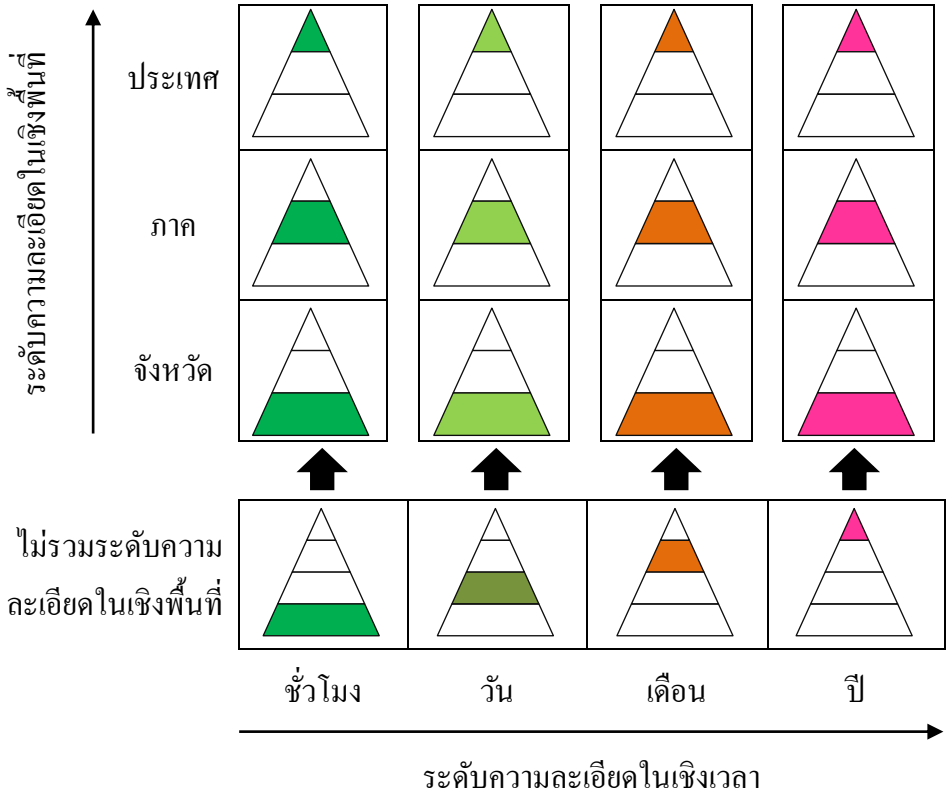
2. เลือกฟังก์ชันที่จะใช้ในการกรองข้อมูล เช่น ฟังก์ชันเฉลี่ย (Avg), ฟังก์ชันหาค่าต่ำสุด (Min), ฟังก์ชันหาค่ามากสุด (Max), ฟังก์ชันมากกว่า (Greater than) และฟังก์ชันน้อยกว่า (Less than) เป็นต้น

เมื่อเลือกคอลัมน์ของข้อมูลและฟังก์ชันที่ต้องการใช้ในการกรองข้อมูลแล้ว จะใช้ SQL Query ทำการ query ข้อมูลจากฐานข้อมูลโดยพิจารณาตามคอลัมน์ของข้อมูลที่เลือก และใช้ฟังก์ชันที่เลือกในการกรองข้อมูล เช่น หากต้องการสร้างชั้นข้อมูลของข้อมูลแผ่นดินไหวในระดับความละเอียดของเวลาเป็นเดือน โดยจะเลือกเฉพาะข้อมูลที่มีระดับความรุนแรงตั้งแต่ 5 ริกเตอร์ขึ้นไป ดังนั้นจึงต้องเลือกคอลัมน์ของข้อมูลเป็นระดับความรุนแรงของแผ่นดินไหว (Magnitude) และเลือกฟังก์ชันในการกรองข้อมูลเป็นฟังก์ชันมากกว่า (Greater than) ซึ่งจะมี SQL Query ในรูปแบบดังนี้ SELECT Magnitude FROM table WHERE Magnitude >= 5

อีกหนึ่งตัวอย่างคือ หากต้องการสร้างชั้นข้อมูลของข้อมูลอุณหภูมิในระดับความละเอียดของเวลาเป็นเดือน โดยจะแสดงเป็นอุณหภูมิเฉลี่ยต่อเดือนของแต่ละสถานีวัดอุณหภูมิ (Station) ดังนั้นจึงต้องเลือกคอลัมน์ของข้อมูลเป็นอุณหภูมิ (Temperature) และเลือกฟังก์ชันในการกรองข้อมูลเป็นฟังก์ชันเฉลี่ย (Avg) ซึ่งจะมี SQL Query ในรูปแบบดังนี้ SELECT AVG(Temperature) FROM table GROUP BY Station

3.3 แนวคิดในการแสดงข้อมูลหลายระดับความละเอียดในเชิงพื้นที่และเวลา

ในหัวข้อที่ผ่านมาเป็นแนวคิดในการแสดงข้อมูลหลายระดับความละเอียดในเชิงเวลา ซึ่งไม่รวมถึงระดับความละเอียดในเชิงพื้นที่ ดังนั้นในการสร้างพีระมิดข้อมูลจะไม่คำนึงถึงความละเอียดเชิงพื้นที่ แต่ถ้าหากคำนึงถึงระดับความละเอียดเชิงพื้นที่ด้วย จะทำให้มีความซับซ้อนในการสร้างพีระมิดข้อมูลมากขึ้น เนื่องจากเมื่อสร้างพีระมิดข้อมูลเชิงเวลาเสร็จแล้ว จะต้องนำข้อมูลในแต่ละชั้นของพีระมิดข้อมูลมาสร้างพีระมิดข้อมูลเชิงพื้นที่อีก โดยมีตัวอย่างในการสร้างพีระมิดข้อมูลที่มีหลายระดับความละเอียดในเชิงพื้นที่และเวลา ดังภาพที่ 3.2



ภาพที่ 3.2 ตัวอย่างในการสร้างพีระมิดข้อมูลที่มีหลายระดับความละเอียดในเชิงพื้นที่และเวลา

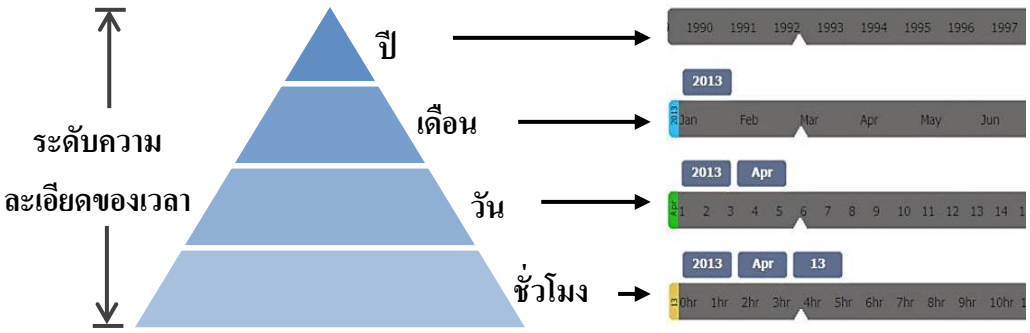
จากตัวอย่างข้างต้น พีระมิดข้อมูลในแถวล่างคือพีระมิดข้อมูลที่มีหลายระดับความละเอียดในเชิงเวลาซึ่งไม่รวมระดับความละเอียดในเชิงพื้นที่ ซึ่งประกอบด้วยชั้นของพีระมิดข้อมูลทั้งหมด 4 ชั้นแทนระดับความละเอียดของเวลาในระดับชั่วโมง, วัน, เดือน และปีตามลำดับ ซึ่งในการสร้างพีระมิดข้อมูลที่มีหลายระดับความละเอียดในเชิงพื้นที่และเวลา จะต้องนำชั้นข้อมูลในแต่ละระดับความละเอียดของเวลามาสร้างพีระมิดข้อมูลตามระดับความละเอียดเชิงพื้นที่ ซึ่งในตัวอย่างมีความละเอียดเชิงพื้นที่ 3 ระดับความละเอียดคือ ระดับจังหวัด, ระดับภาค และระดับประเทศ ข้อมูลใน

ระดับชั่วโมงที่ไม่รวมระดับความละเอียดเชิงพื้นที่ (แถวล่างซ้าย) เมื่อนำมาสร้างพีระมิดข้อมูลเชิงพื้นที่จะได้ชั้นของพีระมิดข้อมูล 3 ชั้นในระดับจังหวัด, ภาค และประเทศ ที่มีความละเอียดของเวลาเป็นชั่วโมง (คอลัมน์ซ้ายสุด) และข้อมูลในระดับวัน, เดือน และปีที่ไม่รวมระดับความละเอียดเชิงพื้นที่ (แถวล่าง) เมื่อนำมาสร้างพีระมิดข้อมูลเชิงพื้นที่จะได้ชั้นของพีระมิดข้อมูลอย่างละ 3 ชั้นในระดับจังหวัด, ภาค และประเทศ ซึ่งรวมชั้นของพีระมิดข้อมูลที่มีหลายระดับความละเอียดในเชิงพื้นที่และเวลาได้ทั้งหมด 12 ชั้น ถ้าหากระดับความละเอียดในเชิงเวลาและระดับความละเอียดในเชิงพื้นที่มีหลายระดับ เมื่อสร้างพีระมิดข้อมูลในเชิงพื้นที่และเวลาจะมีจำนวนชั้นของพีระมิดมากตามไปด้วย

จากตัวอย่างเป็นการสร้างพีระมิดข้อมูลที่มีหลายระดับความละเอียดในเชิงพื้นที่และเวลาสำหรับข้อมูล 1 ข้อมูลและฟังก์ชันในการคัดกรองข้อมูล 1 ฟังก์ชัน ซึ่งข้อมูล 1 ข้อมูลอาจจะมีฟังก์ชันในการคัดกรองข้อมูลมากกว่า 1 ฟังก์ชัน เช่น ข้อมูลอุณหภูมิสามารถคัดกรองได้โดยการคำนวณข้อมูลโดยหาค่าเฉลี่ยของอุณหภูมิ หรือคำนวณข้อมูลโดยการหาค่าต่ำสุดของอุณหภูมิ เป็นต้น ซึ่งในการสร้างพีระมิดข้อมูล จะต้องสร้างพีระมิดข้อมูลตามภาพที่ 3.2 มีจำนวนชุดเท่ากับจำนวนฟังก์ชันที่ต้องการใช้ในการคัดกรองข้อมูล

3.4 แนวคิดในการออกแบบ Timeline

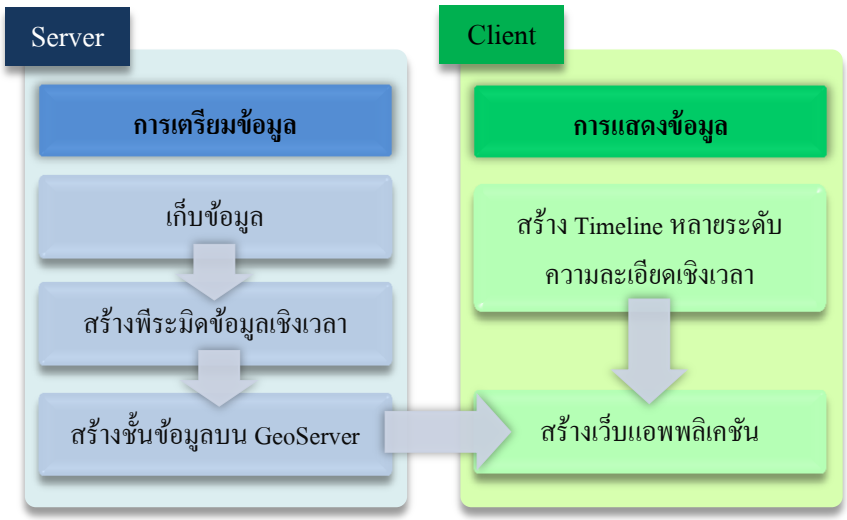
ในหัวข้อนี้จะเป็นการออกแบบ Timeline ที่รองรับหลายระดับความละเอียดของเวลา โดยใช้แนวคิดในการสร้างพีระมิดข้อมูลเชิงเวลามาช่วยในการออกแบบ เนื่องจากมีการแบ่งพีระมิดข้อมูลตามระดับความละเอียดของเวลา ดังนั้นจึงให้แต่ละชั้นของพีระมิดข้อมูลแทนด้วย Timeline ในระดับความละเอียดของเวลาต่าง ๆ และ Timeline สามารถเปลี่ยนระดับความละเอียดไปมาในแต่ละระดับความละเอียดของเวลาได้ ดังภาพที่ 3.3 ด้านซ้ายมือแสดงพีระมิดข้อมูลในระดับความละเอียดของเวลาต่าง ๆ และด้านขวามือแสดง Timeline ในระดับความละเอียดของเวลาต่าง ๆ



ภาพที่ 3.3 การออกแบบ Timeline

3.5 การออกแบบระบบต้นแบบ

จากแนวคิดที่กล่าวมาทั้งหมดในข้างต้นได้นำมาสู่ระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา ซึ่งเป็นระบบที่พัฒนาขึ้นมาสำหรับแสดงข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ โดยระบบจะแบ่งออกเป็น 2 ส่วนหลัก ๆ คือ ส่วนของ Server และส่วนของ Client ดังภาพที่ 3.4



ภาพที่ 3.4 ระบบต้นแบบ

3.5.1 ส่วนของ Server

ในส่วนนี้จะเป็นการเตรียมข้อมูลทั้งหมดที่จะใช้ในการแสดงผลของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ โดยในการเตรียมข้อมูลจะใช้แนวคิดของการสร้างพีระมิดข้อมูลมาเตรียมข้อมูลทั้ง 2 ข้อมูลเพื่อใช้ในการแสดงข้อมูลในระบบต้นแบบ โดยข้อมูลจะถูกเตรียมโดยผู้ดูแลระบบที่ให้บริการข้อมูล ซึ่งขั้นตอนในการเตรียมข้อมูลเริ่มจากเก็บข้อมูล จากนั้นสร้างชั้นพีระมิดข้อมูลเชิงเวลา และนำพีระมิดข้อมูลมาสร้างเป็นชั้นข้อมูลบน GeoServer โดยใน 2 ขั้นตอนแรกสามารถเตรียมข้อมูลผ่านเครื่องมือสำหรับการเก็บข้อมูลและสร้างพีระมิดข้อมูลได้โดยจะอธิบายการพัฒนาและใช้งานเครื่องมือในบทถัดไป ส่วนขั้นตอนการนำพีระมิดข้อมูลมาสร้างเป็นชั้นข้อมูลบน GeoServer ผู้เตรียมข้อมูลต้องสร้างชั้นข้อมูลผ่านอินเตอร์เฟสของ GeoServer

3.5.1.1 การเก็บข้อมูลลงฐานข้อมูลเชิงพื้นที่

ในขั้นตอนนี้จะเป็นการเก็บข้อมูล 2 ข้อมูลคือข้อมูลแผ่นดินไหว และข้อมูลอุณหภูมิ ลงในฐานข้อมูลเชิงพื้นที่คือ PostgreSQL [18] ซึ่งเป็นฐานข้อมูลเชิงพื้นที่ที่เป็นที่นิยมและง่ายต่อการใช้งาน

3.5.1.2 การสร้างพีระมิดข้อมูลเชิงเวลา

เมื่อเก็บข้อมูลลงฐานข้อมูลแล้ว ขั้นตอนต่อไปคือการนำข้อมูลมาสร้างพีระมิดข้อมูลเชิงเวลา โดยข้อมูลทั้ง 2 กลุ่ม มีการสร้างพีระมิดข้อมูลที่แตกต่างกันดังนี้

ข้อมูลกลุ่มที่มีตำแหน่งไม่คงที่ จะสามารถสร้างพีระมิดข้อมูลได้ 2 รูปแบบคือ

- สร้างพีระมิดข้อมูลตามระดับความละเอียดของเวลา โดยในแต่ละชั้นของพีระมิดจะเก็บข้อมูลในแต่ละระดับความละเอียดของเวลา
- สร้างพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล (criteria) เช่น ระดับความรุนแรง โดยในแต่ละชั้นของพีระมิดข้อมูลจะเก็บข้อมูลในแต่ละระดับความรุนแรง

ข้อมูลกลุ่มที่มีตำแหน่งคงที่ จะสามารถสร้างพีระมิดข้อมูลได้ 1 รูปแบบคือ

- สร้างพีระมิดข้อมูลตามระดับความละเอียดของเวลา โดยในแต่ละชั้นของพีระมิดจะเก็บข้อมูลในแต่ละระดับความละเอียดของเวลา

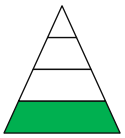
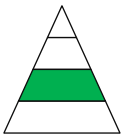
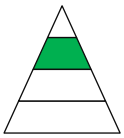
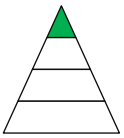
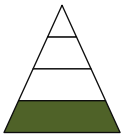
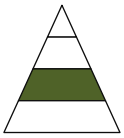
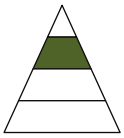
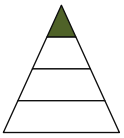
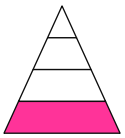
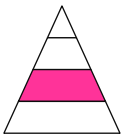
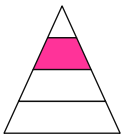
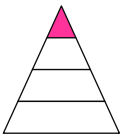
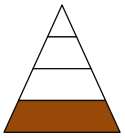
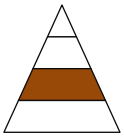
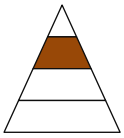
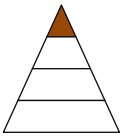
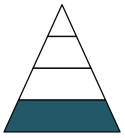
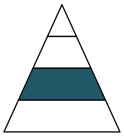
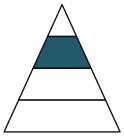
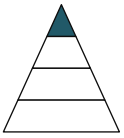
จากข้อมูลทั้ง 2 กลุ่มข้างต้น สรุปได้ว่า การสร้างพีระมิดข้อมูลเชิงเวลาสามารถสร้างได้ 2 รูปแบบ คือ สร้างพีระมิดข้อมูลเชิงเวลาตามระดับความละเอียดของเวลา ซึ่งรูปแบบนี้จะใช้ได้กับทั้งข้อมูลกลุ่มที่มีตำแหน่งไม่คงที่และกลุ่มที่มีตำแหน่งคงที่ และรูปแบบที่ 2 คือสร้างพีระมิดข้อมูลเชิงเวลาตามเกณฑ์ในการคัดเลือกข้อมูล ซึ่งรูปแบบนี้จะใช้กับข้อมูลกลุ่มที่มีตำแหน่งไม่คงที่เท่านั้น ซึ่งมีวิธีการสร้างพีระมิดข้อมูลเชิงเวลาดังนี้

การสร้างพีระมิดข้อมูลเชิงเวลาตามระดับความละเอียดของเวลา

การสร้างพีระมิดข้อมูลเชิงเวลาตามระดับความละเอียดของเวลาจะอาศัยแนวคิดของการแสดงข้อมูลหลายระดับความละเอียดในเชิงเวลามาสร้างพีระมิดข้อมูลเชิงเวลา โดยมีตัวอย่างการสร้างพีระมิดข้อมูลของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิดังตารางที่ 3.3 ซึ่งสมมติว่าข้อมูลถูก

เก็บมาในระดับชั่วโมง ดังนั้นชั้นล่างสุดของพีระมิดข้อมูลจะเป็นระดับชั่วโมง และชั้นบนของพีระมิดข้อมูลจะเป็นระดับวัน, เดือนและ ปี ตามลำดับ

ตารางที่ 3.3 ตารางแสดงตัวอย่างการสร้างพีระมิดข้อมูลของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ

ประเภทข้อมูล	**คอลัมน์ของข้อมูล**	**ฟังก์ชัน**	**ระดับความละเอียดของเวลา**			
			ชั่วโมง	**วัน**	**เดือน**	**ปี**
ข้อมูลแผ่นดินไหว	(Mag.) ระดับความรุนแรง	Greater than	 All Mag.	 Mag. > 4	 Mag. > 5	 Mag. > 6
ข้อมูลแผ่นดินไหว	(Mag.) ระดับความรุนแรง	Between	 All Mag.	 4<Mag.<5	 5<Mag.<6	 6<Mag.<7
ข้อมูลอุณหภูมิ	(Temp.) อุณหภูมิ	Average	 All Temp.	 Avg Temp.	 Avg Temp.	 Avg Temp.
ข้อมูลอุณหภูมิ	(Temp.) อุณหภูมิ	Maximum	 All Temp.	 Max Temp.	 Max Temp.	 Max Temp.
ข้อมูลอุณหภูมิ	(Temp.) อุณหภูมิ	Minimum	 All Temp.	 Min Temp.	 Min Temp.	 Min Temp.

จากตารางที่ 3.3 เป็นการสร้างพีระมิดข้อมูลของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ โดยข้อมูลทั้ง 2 ข้อมูลมีการเลือกใช้ฟังก์ชันในการกรองข้อมูลมากกว่า 1 ฟังก์ชัน ดังนั้นข้อมูลทั้ง 2 ข้อมูลจึงต้องมีพีระมิดข้อมูลมากกว่า 1 พีระมิดข้อมูล โดยแต่ละพีระมิดข้อมูลจะแทนฟังก์ชัน 1

ฟังก์ชัน และในแต่ละพีระมิดข้อมูลจะประกอบไปด้วยชั้นข้อมูลที่มีความละเอียดของเวลาต่าง ๆ คือระดับชั่วโมง, วัน, เดือน และปี จากตัวอย่างข้างต้นจะมีพีระมิดข้อมูลทั้งหมด 5 พีระมิด ดังนี้

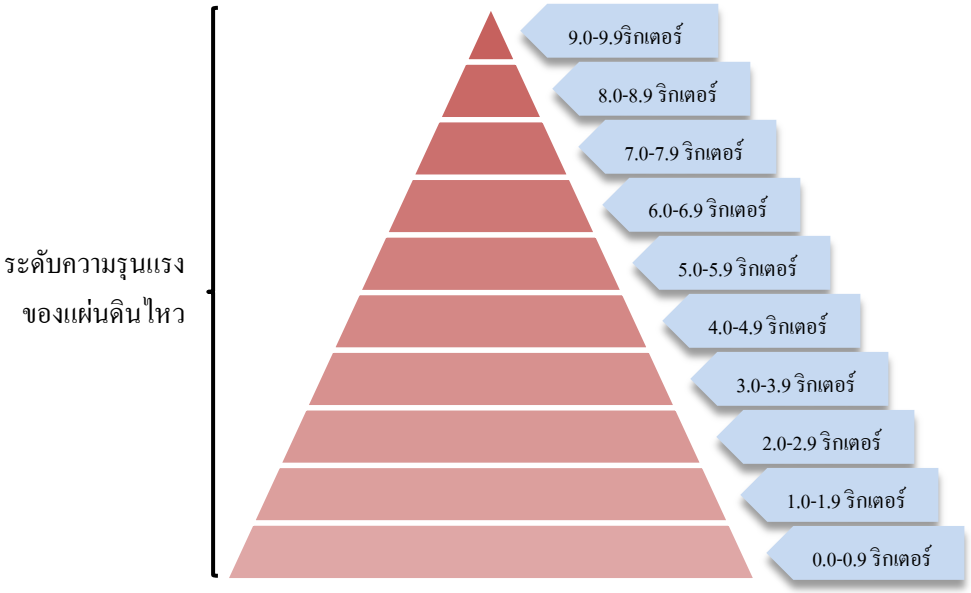
1. ข้อมูลแผ่นดินไหวมีการเลือกข้อมูลโดยใช้ระดับความรุนแรงของการเกิดแผ่นดินไหว และฟังก์ชันมากกว่า (Greater than) เป็นเกณฑ์ในการคัดเลือกข้อมูล โดยชั้นล่างสุดจะเป็นระดับชั่วโมง ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลที่เก็บมาทั้งหมด และในชั้นถัดขึ้นมาจะเป็นระดับวัน ข้อมูลที่อยู่ในชั้นนี้จะมีระดับความรุนแรงของแผ่นดินไหวมากกว่า 4 ริกเตอร์ และในชั้นถัดขึ้นมาจะเป็นระดับเดือน ข้อมูลที่อยู่ในชั้นนี้จะมีระดับความรุนแรงของแผ่นดินไหวมากกว่า 5 ริกเตอร์ และในชั้นบนสุดจะเป็นระดับปี ข้อมูลที่อยู่ในชั้นนี้จะมีระดับความรุนแรงของแผ่นดินไหวมากกว่า 6 ริกเตอร์
2. ข้อมูลแผ่นดินไหวมีการเลือกข้อมูลโดยใช้ระดับความรุนแรงของการเกิดแผ่นดินไหว และฟังก์ชันระหว่าง (Between) เป็นเกณฑ์ในการคัดเลือกข้อมูล โดยชั้นล่างสุดจะเป็นระดับชั่วโมง ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลที่เก็บมาทั้งหมด และในชั้นถัดขึ้นมาจะเป็นระดับวัน ข้อมูลที่อยู่ในชั้นนี้จะมีระดับความรุนแรงของแผ่นดินไหวระหว่าง 4-5 ริกเตอร์ และในชั้นถัดขึ้นมาจะเป็นระดับเดือน ข้อมูลที่อยู่ในชั้นนี้จะมีระดับความรุนแรงของแผ่นดินไหวระหว่าง 5-6 ริกเตอร์ และในชั้นบนสุดจะเป็นระดับปี ข้อมูลที่อยู่ในชั้นนี้จะมีระดับความรุนแรงของแผ่นดินไหวระหว่าง 6-7 ริกเตอร์
3. ข้อมูลอุณหภูมิมีการคำนวณข้อมูลโดยใช้อุณหภูมิ และฟังก์ชันเฉลี่ย (Avg.) เป็นเกณฑ์ในการคำนวณข้อมูล โดยชั้นล่างสุดจะเป็นระดับชั่วโมง ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลที่เก็บมาทั้งหมด และในชั้นถัดขึ้นมาจะเป็นระดับวัน ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลอุณหภูมิเฉลี่ยต่อวันของแต่ละสถานีวัดอุณหภูมิ และในชั้นถัดขึ้นมาจะเป็นระดับเดือน ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลอุณหภูมิเฉลี่ยต่อเดือนของแต่ละสถานีวัดอุณหภูมิ และในชั้นบนสุดจะเป็นระดับปี ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลอุณหภูมิเฉลี่ยต่อปีของแต่ละสถานีวัดอุณหภูมิ
4. ข้อมูลอุณหภูมิมีการคำนวณข้อมูลโดยใช้อุณหภูมิ และฟังก์ชันหาค่ามากสุด (Max.) เป็นเกณฑ์ในการคำนวณข้อมูล โดยชั้นล่างสุดจะเป็นระดับชั่วโมง ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลที่เก็บมาทั้งหมด และในชั้นถัดขึ้นมาจะเป็นระดับวัน ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลอุณหภูมิมากที่สุดต่อวันของแต่ละสถานีวัดอุณหภูมิ และในชั้นถัดขึ้นมาจะเป็นระดับเดือน ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลอุณหภูมิมากที่สุดต่อเดือน

ของแต่ละสถานีวัดอุณหภูมิ และในชั้นบนสุดจะเป็นระดับปี ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลอุณหภูมิมากที่สุดต่อปีของแต่ละสถานีวัดอุณหภูมิ

5. ข้อมูลอุณหภูมิมีการคำนวณข้อมูลโดยใช้อุณหภูมิ และฟังก์ชันหาค่าน้อยสุด (Min.) เป็นเกณฑ์ในการคำนวณข้อมูล โดยชั้นล่างสุดจะเป็นระดับชั่วโมง ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลที่เก็บมาทั้งหมด และในชั้นถัดขึ้นมาจะเป็นระดับวัน ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลอุณหภูมิน้อยที่สุดต่อวันของแต่ละสถานีวัดอุณหภูมิ และในชั้นถัดขึ้นมาจะเป็นระดับเดือน ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลอุณหภูมิน้อยที่สุดต่อเดือนของแต่ละสถานีวัดอุณหภูมิ และในชั้นบนสุดจะเป็นระดับปี ข้อมูลที่อยู่ในชั้นนี้จะเป็นข้อมูลอุณหภูมิน้อยที่สุดต่อปีของแต่ละสถานีวัดอุณหภูมิ

การสร้างพีระมิดข้อมูลเชิงเวลาตามเกณฑ์ในการคัดเลือกข้อมูล

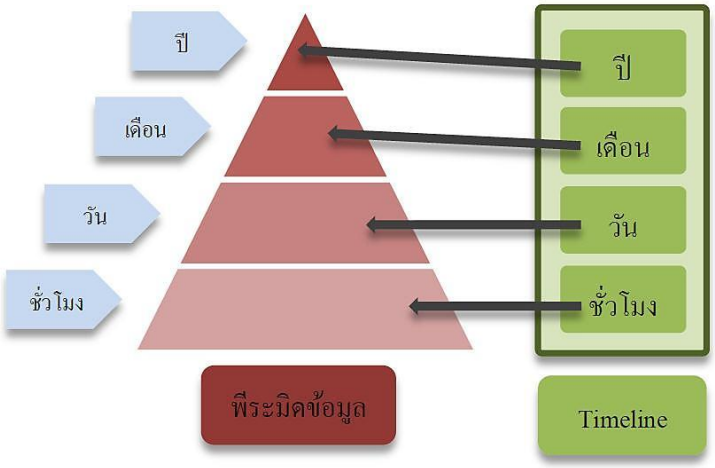
การสร้างพีระมิดข้อมูลเชิงเวลาตามเกณฑ์ในการคัดเลือกข้อมูลจะอาศัยแนวคิดของการแสดงข้อมูลหลายระดับความละเอียดมาสร้างพีระมิดข้อมูลเชิงเวลา โดยที่แต่ละชั้นของพีระมิดข้อมูลจะมีเกณฑ์ในการคัดเลือกข้อมูล ตัวอย่างเช่น การสร้างพีระมิดข้อมูลของข้อมูลแผ่นดินไหว จะใช้ระดับความรุนแรงของแผ่นดินไหวเป็นเกณฑ์ในการคัดเลือกข้อมูล โดยมีตัวอย่างการสร้างพีระมิดข้อมูลของข้อมูลแผ่นดินไหวดังภาพที่ 3.5



ภาพที่ 3.5 การสร้างพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล

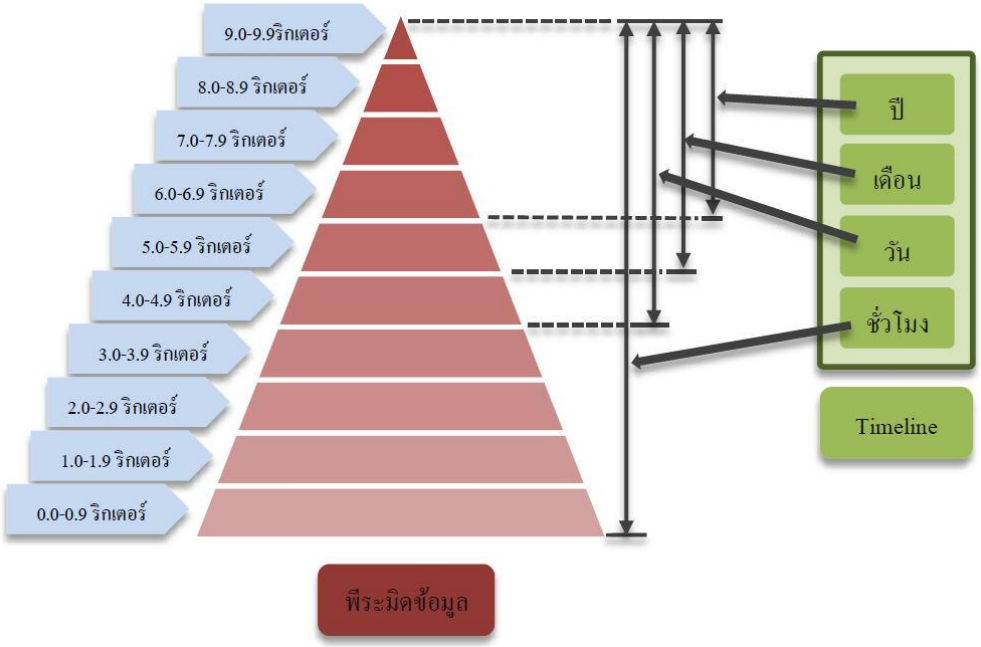
การเรียกใช้ชั้นข้อมูลของพีระมิดข้อมูลเชิงเวลา

ในการเรียกใช้ชั้นข้อมูลของพีระมิดข้อมูลตามความละเอียดของเวลา จะเรียกใช้ชั้นข้อมูลตามระดับความละเอียดของเวลาของ Timeline ตัวอย่างเช่น ถ้า Timeline กำลังแสดงอยู่ในระดับความละเอียดของเวลาเป็นปี ดังนั้นในการเรียกใช้ชั้นข้อมูลมาแสดงบนแผนที่จะต้องเป็นชั้นปีเช่นกัน หรือ ถ้า Timeline กำลังแสดงอยู่ในระดับความละเอียดของเวลาเป็นชั่วโมง ดังนั้นในการเรียกใช้ชั้นข้อมูลมาแสดงบนแผนที่จะต้องเป็นชั้นชั่วโมงเช่นกัน ดังภาพที่ 3.6



ภาพที่ 3.6 การเรียกใช้ชั้นข้อมูลของพีระมิดข้อมูลตามความละเอียดของเวลา

สำหรับการเรียกใช้ชั้นข้อมูลของพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล ในระดับความละเอียดของเวลาเป็นชั่วโมง, วัน, เดือน หรือปี ผู้ใช้งานจะสามารถเลือกได้ว่าจะดูข้อมูลในระดับความรุนแรงเท่าใด ตัวอย่างเช่น การเรียกใช้ชั้นข้อมูลของพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกของข้อมูลแผ่นดินไหว ในระดับชั่วโมง ผู้ใช้งานอาจจะเลือกดูข้อมูลการเกิดแผ่นดินไหวทุก ๆ ระดับความรุนแรงของแผ่นดินไหว, ในระดับวัน ผู้ใช้งานอาจจะเลือกดูข้อมูลการเกิดแผ่นดินไหวในระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 4 ริกเตอร์ขึ้นไป, ในระดับเดือน ผู้ใช้งานอาจจะเลือกดูข้อมูลการเกิดแผ่นดินไหวในระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 5 ริกเตอร์ขึ้นไป และในระดับปี ผู้ใช้งานอาจจะเลือกดูข้อมูลการเกิดแผ่นดินไหวในระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 6 ริกเตอร์ขึ้นไป เป็นต้น ดังภาพที่ 3.7



ภาพที่ 3.7 การเรียกใช้ชั้นข้อมูลของพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล

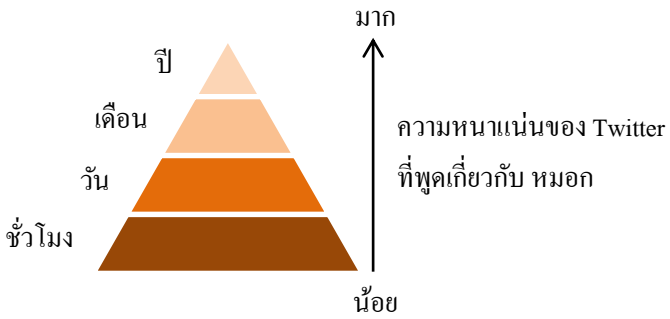
ตัวอย่างการเตรียมพีระมิดข้อมูลของข้อมูลอื่น ๆ

ในการเตรียมชั้นข้อมูลโดยการสร้างพีระมิดข้อมูลในข้างต้นสามารถนำแนวคิดของการสร้างพีระมิดข้อมูลของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิมาประยุกต์ใช้กับข้อมูลอื่น ๆ ได้ ดังตัวอย่างต่อไปนี้

ข้อมูลกลุ่มที่มีตำแหน่งไม่คงที่

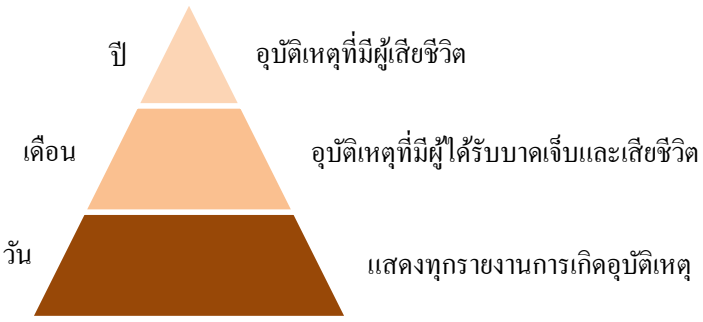
- ข้อมูลจาก Twitter เช่น การแสดงข้อมูล Twitter ของผู้ใช้ทั่วไปที่พูดคุยแสดงความคิดเห็นที่มีคำว่า หมอก ปรากฏอยู่ในข้อความ โดยจะแสดงข้อมูล Twitter ของผู้ใช้ทั่วไปตามตำแหน่งจีพีเอสที่ได้ส่งข้อความบนแผนที่ ซึ่งข้อมูลจาก Twitter จะมีเป็นจำนวนมากมายและจะมีข้อมูลได้ทุกตำแหน่งบนแผนที่ ดังนั้นข้อมูลจาก Twitter จึงจัดอยู่ในกลุ่มข้อมูลที่มีตำแหน่งไม่คงที่ โดยในการแสดงข้อมูลตามความละเอียดของเวลาอาจจะมีเกณฑ์บางอย่าง เช่น ความหนาแน่นของข้อมูลที่พูดถึงเรื่องหมอก เป็นต้น มาเลือกข้อมูลเพื่อแสดงบนแผนที่ในแต่ละความละเอียดของเวลา ดังนั้นในการสร้างชั้นพีระมิดข้อมูลจึงสร้างตามความหนาแน่นของข้อมูล Twitter โดยจะต้องหาความหนาแน่นของข้อมูลบนแผนที่ทั้งหมดก่อน แล้วจะทราบว่าบริเวณใดมีการพูดถึงเรื่องหมอกมากน้อยเพียงใด เช่น ในระดับความละเอียดของเวลาเป็นชั่วโมง อาจจะเลือก

บริเวณที่มีความหนาแน่นของการพูดถึงเรื่องหมอกน้อย ๆ ไปจนถึงบริเวณที่มีความหนาแน่นของการพูดถึงเรื่องหมอกมาก ๆ ส่วนในระดับความละเอียดของเวลาเป็นปี อาจจะเลือกเฉพาะบริเวณที่มีความหนาแน่นของการพูดถึงเรื่องหมอกมาก ๆ มาแสดงบนแผนที่ ดังภาพที่ 3.8



ภาพที่ 3.8 การสร้างชั้นพีระมิดข้อมูลของข้อมูลจาก Twitter

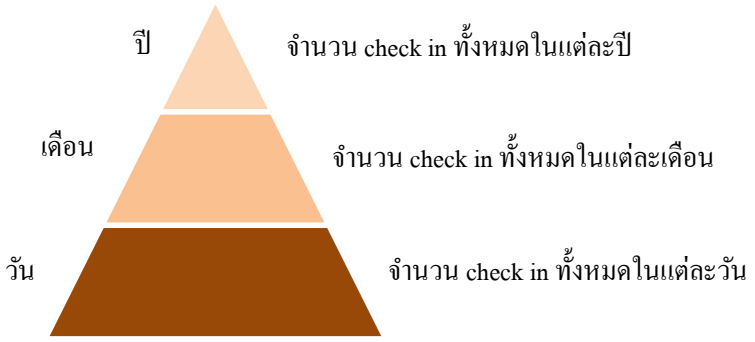
- ข้อมูลการรายงานการเกิดอุบัติเหตุทางจราจร ที่ได้จากสำนักงานตำรวจ เป็นการรายงานว่าตำแหน่งใดบนแผนที่เกิดอุบัติเหตุ ณ เวลาใดบ้าง ซึ่งข้อมูลการรายงานอุบัติเหตุทางจราจรจะมีเป็นจำนวนมากมายและจะมีข้อมูลได้ทุกตำแหน่งบนแผนที่ ดังนั้นข้อมูลการรายงานอุบัติเหตุทางจราจรจึงจัดอยู่ในกลุ่มข้อมูลที่มีตำแหน่งไม่คงที่ โดยในการแสดงข้อมูลตามความละเอียดของเวลาอาจจะมีเกณฑ์บางอย่าง เช่น ระดับความรุนแรงของอุบัติเหตุ เป็นต้น มาเลือกข้อมูลเพื่อแสดงบนแผนที่ในแต่ละความละเอียดของเวลา ดังนั้นในการสร้างชั้นพีระมิดข้อมูลจึงสร้างตามระดับความรุนแรงของอุบัติเหตุที่พิจารณาจากจำนวนผู้ได้รับบาดเจ็บและเสียชีวิต เช่น ในระดับความละเอียดของเวลาเป็นวัน อาจจะแสดงทุกรายงานการเกิดอุบัติเหตุ, ส่วนในระดับความละเอียดของเวลาเป็นเดือน อาจจะแสดงเฉพาะรายงานการเกิดอุบัติเหตุที่มีผู้ได้รับบาดเจ็บและเสียชีวิต และในระดับความละเอียดของเวลาเป็นปี อาจจะแสดงเฉพาะรายงานการเกิดอุบัติเหตุที่มีผู้เสียชีวิต ดังภาพที่ 3.9



ภาพที่ 3.9 การสร้างชั้นพีระมิดข้อมูลของข้อมูลการรายงานการเกิดอุบัติเหตุทางจราจร

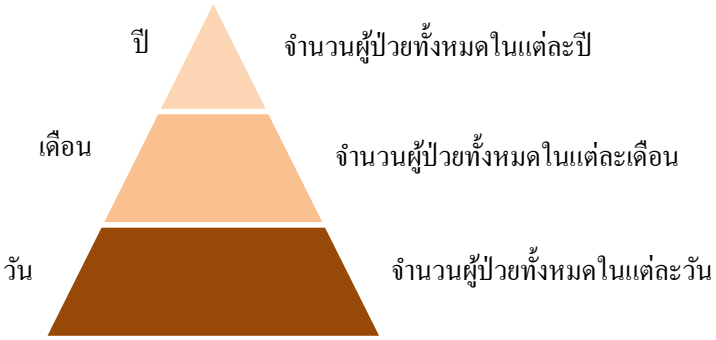
ข้อมูลกลุ่มที่มีตำแหน่งคงที่

- ข้อมูลการ check in ของโรงแรมทั่วโลก ซึ่งจำนวนโรงแรมทั่วโลกจะมีตำแหน่งและจำนวนคงที่ ดังนั้นข้อมูลการ check in ของโรงแรมทั่วโลกจึงจัดอยู่ในกลุ่มข้อมูลที่มีตำแหน่งคงที่ โดยในการแสดงข้อมูลตามความละเอียดของเวลาอาจจะมีการคำนวณข้อมูลการ check in ของแต่ละโรงแรมเพื่อเป็นตัวแทนข้อมูลของแต่ละโรงแรมในการแสดงผลบนแผนที่ในแต่ละระดับความละเอียดของเวลา เช่น ใช้จำนวนการ check in รวมทั้งหมด (SUM) ของโรงแรมเป็นตัวแทนข้อมูลของแต่ละโรงแรม โดยในระดับปี ในแต่ละโรงแรม อาจจะแสดงจำนวนการ check in ทั้งหมดตลอดทั้งปี, ในระดับเดือน ในแต่ละโรงแรม อาจจะแสดงจำนวนการ check in ทั้งหมดตลอดทั้งเดือน เป็นต้น ดังภาพที่ 3.10



ภาพที่ 3.10 การสร้างชั้นพีระมิดข้อมูลการ check in ของโรงแรมทั่วโลก

- ข้อมูลการรายงานโรคระบาด จากกรมควบคุมโรค ซึ่งจะเป็นการรายงานข้อมูลโรคระบาดจากสถานีอนามัยหรือโรงพยาบาลต่าง ๆ มายังกรมควบคุมโรค ซึ่งสถานีอนามัยและโรงพยาบาลจะมีตำแหน่งและจำนวนคงที่ ดังนั้นข้อมูลการรายงานโรคระบาดจึงจัดอยู่ในกลุ่มข้อมูลที่มีตำแหน่งคงที่ โดยในการแสดงข้อมูลตามความละเอียดของเวลาอาจจะมีการคำนวณข้อมูลจำนวนผู้ป่วยของแต่ละสถานีอนามัยและโรงพยาบาลเพื่อเป็นตัวแทนข้อมูลของแต่ละตำแหน่งในการแสดงผลบนแผนที่ในแต่ละระดับความละเอียดของเวลา เช่น ใช้จำนวนผู้ป่วยทั้งหมด (SUM) ของแต่ละสถานีอนามัยและโรงพยาบาลเป็นตัวแทนข้อมูลของแต่ละตำแหน่งโดยในระดับปี ในแต่ละสถานีอนามัยและโรงพยาบาล อาจจะแสดงจำนวนผู้ป่วยทั้งหมดตลอดทั้งปี, ในระดับเดือน ในแต่ละสถานีอนามัยและโรงพยาบาล อาจจะแสดงจำนวนผู้ป่วยทั้งหมดตลอดทั้งเดือน เป็นต้น ดังภาพที่ 3.11

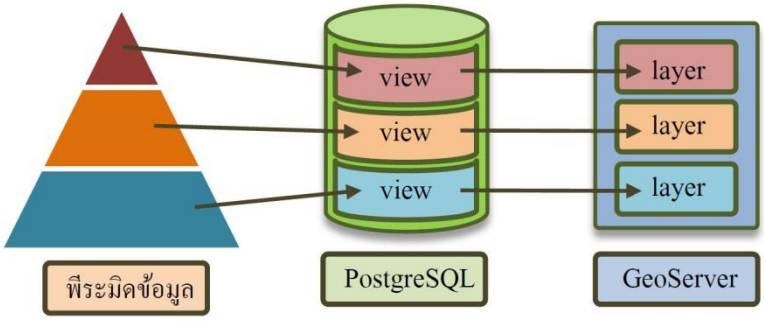


ภาพที่ 3.11 การสร้างชั้นพีระมิดข้อมูลการรายงานโรคระบาด

3.5.1.3 การสร้างชั้นข้อมูลใน GeoServer

เมื่อสร้างพีระมิดข้อมูลเรียบร้อยแล้ว ในขั้นตอนถัดไปคือการนำพีระมิดข้อมูลทั้งหมดที่ได้สร้างไว้มาเพิ่มลงในชั้นข้อมูลของ GeoServer โดยที่ชั้นข้อมูลของพีระมิดข้อมูล 1 ชั้นจะถูกสร้างเป็นชั้นข้อมูลใน GeoServer 1 ชั้นข้อมูล

การนำพีระมิดข้อมูลมาสร้างเป็นชั้นข้อมูลใน GeoServer มีขั้นตอนดังภาพที่ 3.12 โดยเริ่มจากนำแต่ละชั้นพีระมิดข้อมูลมาสร้างเป็น view ใน PostgreSQL ซึ่งแต่ละชั้นของพีระมิดข้อมูลจะถูกสร้างเป็น 1 view ใน PostgreSQL และในขั้นตอนถัดไปคือการนำพีระมิดข้อมูลทั้งหมดที่ถูกเก็บเป็น view มาเพิ่มลงในชั้นข้อมูลของ GeoServer โดยที่ view 1 view จะถูกสร้างเป็นชั้นข้อมูลใน GeoServer 1 ชั้นข้อมูล โดยขั้นตอนในการเพิ่มชั้นข้อมูลใน GeoServer สามารถศึกษาเพิ่มเติมได้ในภาคผนวกท้ายเล่มวิทยานิพนธ์



ภาพที่ 3.12 การสร้างชั้นพีระมิดข้อมูลการรายงานโรค

เมื่อเพิ่มชั้นพีระมิดข้อมูลลงในชั้นข้อมูลของ GeoServer ครบทุกพีระมิดข้อมูล จะถือว่าเป็นการเสร็จสิ้นขั้นตอนการเตรียมข้อมูลในส่วนของ Server ซึ่งข้อมูลที่ถูกเตรียมไว้ทั้งหมดนี้จะถูกเรียกใช้ในการแสดงผลบนแผนที่บนเว็บแอพพลิเคชันในลำดับถัดไป

3.5.2 ส่วนของ Client

ในส่วนของ Client นี้จะเป็นเว็บแอพพลิเคชันสำหรับการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา ซึ่งออกแบบเป็น Web Mapping Application โดยการพัฒนาระบบในส่วนของ Client จะประกอบด้วยการพัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลา เพื่อนำไปใช้ในหน้าเว็บแอพพลิเคชันในการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา โดยการพัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลาจะอธิบายในบทถัดไป

เนื้อความโดยสรุปของบทนี้ ได้นำเสนอการแบ่งประเภทข้อมูลตัวอย่าง แนวคิดในการสร้างพีระมิดข้อมูลเชิงเวลา แนวคิดในการสร้างพีระมิดข้อมูลเชิงพื้นที่และเวลา แนวคิดในการออกแบบ Timeline ที่รองรับหลายระดับความละเอียดของเวลา จากนั้นเป็นการออกแบบระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาโดยแบ่งเป็นส่วนของการเตรียมข้อมูลในฝั่ง Server ซึ่งได้อธิบายขั้นตอนการเตรียมข้อมูลไปแล้ว และส่วนของการแสดงข้อมูลบนเว็บแอพพลิเคชันในฝั่ง Client ซึ่งจะอธิบายไปลำดับถัดไป และในบทถัดไปจะกล่าวถึงการพัฒนาเครื่องมือสำหรับการเตรียมข้อมูล

บทที่ 4
การพัฒนาเครื่องมือสำหรับการเตรียมข้อมูล

หลังจากที่ได้อธิบายถึงแนวคิดในการออกแบบระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาไปในบทที่แล้ว ในบทนี้จะกล่าวถึงการพัฒนาเครื่องมือสำหรับการเตรียมข้อมูล ซึ่งประกอบไปด้วยเครื่องมือสำหรับนำข้อมูลเข้าฐานข้อมูลเชิงพื้นที่ และเครื่องมือสำหรับสร้างพีระมิดข้อมูลสำหรับข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ รวมถึงการประยุกต์ใช้เครื่องมือสำหรับการเตรียมข้อมูลชนิดอื่น

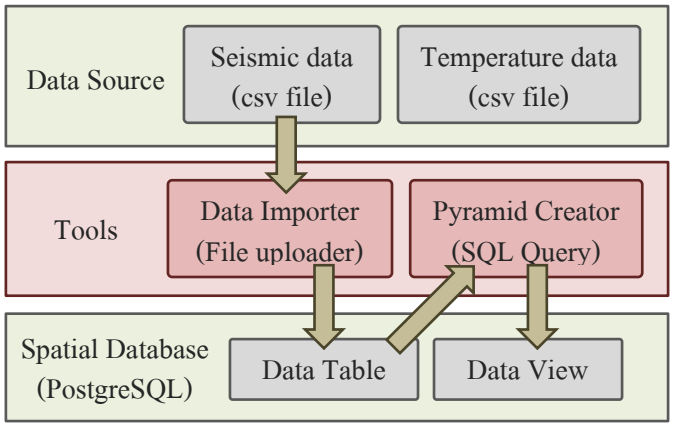
4.1 การพัฒนาเครื่องมือสำหรับการเตรียมข้อมูล

เครื่องมือสำหรับการเตรียมข้อมูลประกอบไปด้วยการนำข้อมูลเข้าฐานข้อมูลและการสร้างพีระมิดข้อมูล ซึ่งพัฒนาเป็นเว็บแอพพลิเคชันโดยใช้ภาษา php โดยมีโครงสร้างโดยรวมของเครื่องมือสำหรับการเตรียมข้อมูลดังภาพที่ 4.1 ซึ่งประกอบไปด้วย 3 องค์ประกอบ ดังนี้

องค์ประกอบส่วนที่ 1 คือ ข้อมูลที่จะนำเข้าฐานข้อมูล (Data Source) จะต้องอยู่ในรูปแบบของไฟล์ csv ที่มีรูปแบบคอลัมน์ภายในไฟล์ตามที่กำหนด โดยข้อมูลแต่ละข้อมูลจะแยกเป็นคนละไฟล์กัน เช่น ข้อมูลแผ่นดินไหวเก็บอยู่ในไฟล์ seismic.csv และข้อมูลอุณหภูมิเก็บอยู่ในไฟล์ temperature.csv เป็นต้น

องค์ประกอบส่วนที่ 2 คือ เครื่องมือที่พัฒนา (Tools) สำหรับนำข้อมูลนำข้อมูลเข้าฐานข้อมูล (Data Importer) เป็นส่วนสำหรับอัพโหลดไฟล์ข้อมูลเข้าสู่ฐานข้อมูล (File uploader) และเครื่องมือสำหรับสร้างพีระมิดข้อมูล (Pyramid Creator) โดยใช้ SQL Query เลือกข้อมูลจากตารางในฐานข้อมูลมาสร้างเป็นชั้นของพีระมิดข้อมูล จากนั้นจะนำทุกชั้นของพีระมิดข้อมูลมาสร้างเป็น view ในฐานข้อมูลเพื่อนำไปสร้างชั้นข้อมูลบน GeoServer ในลำดับถัดไป

องค์ประกอบส่วนที่ 3 คือ ฐานข้อมูลเชิงพื้นที่ (Spatial Database) คือ PostgreSQL ซึ่งประกอบด้วยตาราง (table) สำหรับเก็บข้อมูลที่นำเข้ามา โดยข้อมูลแต่ละข้อมูลจะถูกเก็บเป็นคนละตาราง และวิว (view) สำหรับเก็บข้อมูลของแต่ละชั้นของพีระมิดข้อมูล

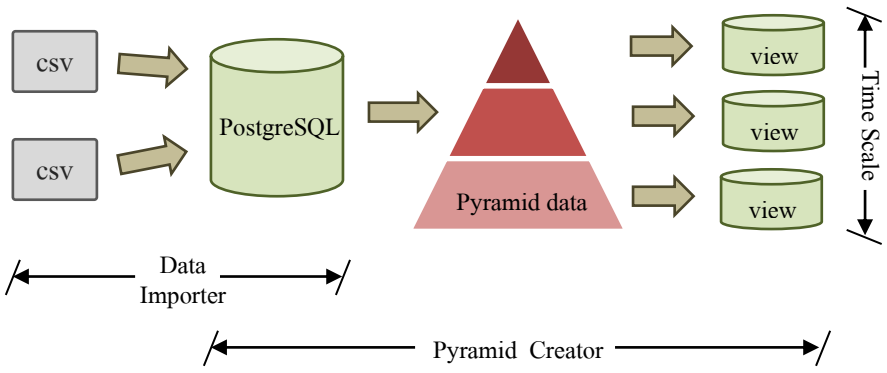


ภาพที่ 4.1 โครงสร้างของเครื่องมือสำหรับการเตรียมข้อมูล

เครื่องมือสำหรับการเตรียมข้อมูลที่พัฒนาขึ้นมามีการทำงานดังภาพที่ 4.2 โดยมีขั้นตอนการทำงานดังนี้

1. ใช้เครื่องมือสำหรับนำข้อมูลเข้าฐานข้อมูลเชิงพื้นที่ โดยจะมีปุ่มให้อัพโหลดข้อมูลเข้าสู่ฐานข้อมูล และข้อมูลที่จะอัพโหลดจะต้องอยู่ในรูปแบบของไฟล์ csv ที่มีรูปแบบภายในไฟล์ตามที่กำหนดไว้

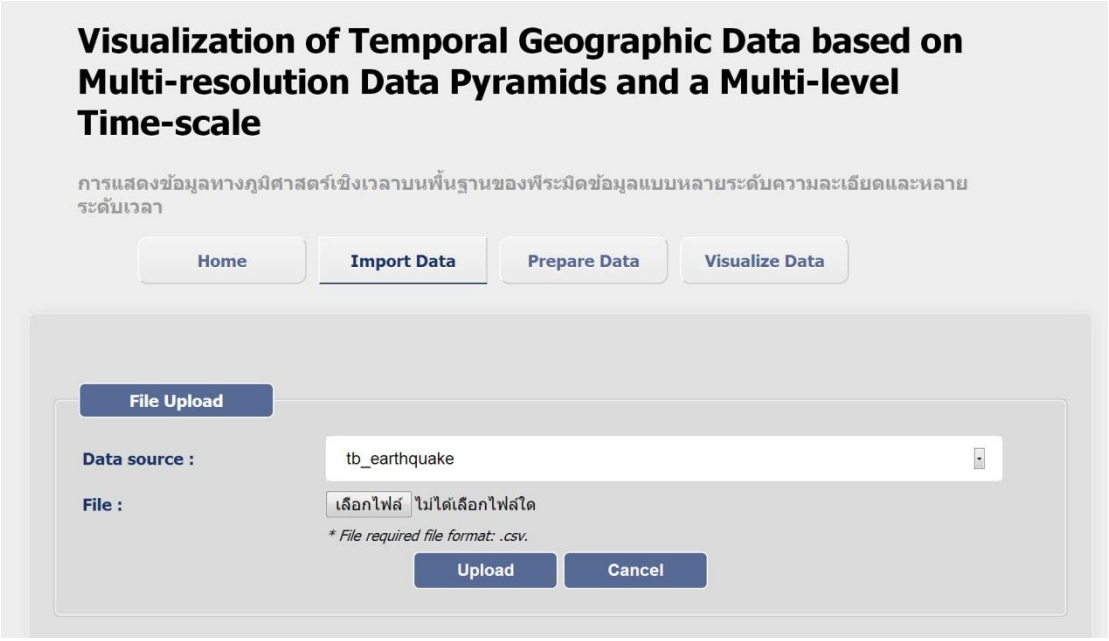
2. ใช้เครื่องมือสำหรับสร้างพีระมิดข้อมูล โดยผู้เตรียมข้อมูลสามารถตั้งค่าในการสร้างชั้นพีระมิดข้อมูลสำหรับแต่ละระดับความละเอียดของเวลาได้ จากนั้นเครื่องมือจะทำการสร้าง view ของแต่ละชั้นของพีระมิดข้อมูลเก็บไว้ในฐานข้อมูลให้อัตโนมัติ



ภาพที่ 4.2 การทำงานของเครื่องมือสำหรับการเตรียมข้อมูล

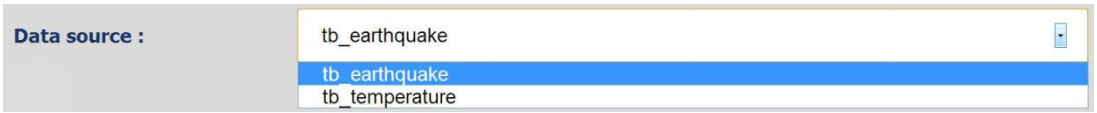
4.1.1 เครื่องมือสำหรับนำข้อมูลเข้าฐานข้อมูลเชิงพื้นที่

ในขั้นตอนการเตรียมข้อมูลจะเริ่มจากการนำข้อมูลเข้าฐานข้อมูลเชิงพื้นที่ PostgreSQL โดยผู้วิจัยได้พัฒนาเครื่องมือสำหรับนำข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิเข้าฐานข้อมูลบนเว็บแอพพลิเคชัน โดยมีหน้าเว็บดังภาพที่ 4.3



ภาพที่ 4.3 หน้าเว็บสำหรับนำข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิเข้าฐานข้อมูล PostgreSQL

ในหน้าเว็บนี้ประกอบไปด้วยส่วนสำหรับเลือกชนิดข้อมูลที่จะนำเข้าฐานข้อมูล และส่วนสำหรับเลือกไฟล์ข้อมูลที่จะนำเข้าฐานข้อมูล โดยเครื่องมือที่พัฒนาขึ้นมานี้รองรับเฉพาะข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ โดยผู้ใช้งานสามารถเลือกชนิดข้อมูลที่จะนำเข้าฐานข้อมูลได้ในส่วนของ Data source ดังภาพที่ 4.4



ภาพที่ 4.4 การเลือกชนิดข้อมูลที่จะนำเข้าฐานข้อมูล

ผู้ใช้งานสามารถเลือกไฟล์ข้อมูลที่จะนำเข้าฐานข้อมูลได้ในส่วน File โดยไฟล์ข้อมูลที่จะนำเข้าฐานข้อมูลจะต้องอยู่ในรูปแบบของไฟล์ csv ที่มีรูปแบบภายในไฟล์ดังนี้

File: ข้อมูลแผ่นดินไหว .csv

วันที่	เวลา	ศูนย์กลางการเกิดแผ่นดินไหว	ละติจูด	ลองจิจูด	ระดับความรุนแรงของแผ่นดินไหว	รายละเอียด

File: ข้อมูลอุณหภูมิ .csv

วันที่	เวลา	สถานีวัดอุณหภูมิ	ละติจูด	ลองจิจูด	อุณหภูมิ	รายละเอียด

เมื่อเลือกไฟล์ที่จะนำเข้าฐานข้อมูลแล้วกดปุ่ม Upload จากนั้นระบบจะอ่านไฟล์ที่นำเข้าไป แล้วเพิ่มข้อมูลลงฐานข้อมูล โดยภายในฐานข้อมูลจะมีตารางสำหรับเก็บข้อมูลแผ่นดินไหว (tb_earthquake) และข้อมูลอุณหภูมิ (tb_temperature) ซึ่งประกอบไปด้วยฟิลด์ต่าง ๆ ดังนี้

Table: tb_earthquake

eq_id	eq_datetime	eq_focus	eq_latitude	eq_longitude	geometry	eq_magnitude

Table: tb_temperature

eq_id	tm_datetime	tm_station_en	tm_latitude	tm_longitude	geometry	tm_temperature

ในการเพิ่มข้อมูลเข้าฐานข้อมูล ระบบจะตรวจสอบ Data source ว่าเลือกชนิดข้อมูลใด จากนั้นจะส่งข้อมูลไปยังไฟล์ (.php) ของข้อมูลแต่ละชนิดเพื่อเพิ่มข้อมูลเข้าฐานข้อมูล เช่น ถ้าต้องการนำข้อมูลแผ่นดินไหวเข้าฐานข้อมูล จะต้องเลือก Data source เป็น tb_earthquake จากนั้นระบบจะส่งไฟล์ข้อมูล (.csv) ไปยังไฟล์ uploadImport_earthquake.php เพื่อทำการเพิ่มข้อมูลเข้าฐานข้อมูล แต่ถ้าต้องการนำข้อมูลอุณหภูมิเข้าฐานข้อมูล จะต้องเลือก Data source เป็น tb_temperature จากนั้นระบบจะส่งไฟล์ข้อมูล (.csv) ไปยังไฟล์ uploadImport_temperature.php เพื่อทำการเพิ่มข้อมูลเข้าฐานข้อมูล เป็นต้น

ในหน้าหลักของการเพิ่มข้อมูลเข้าฐานข้อมูล จะมีการดึงชื่อตารางสำหรับเก็บข้อมูลจากตารางชื่อ tb_data_store ซึ่งเป็นตารางสำหรับเก็บรายชื่อตารางของข้อมูลที่ต้องการแสดงผล และในเมื่อกดปุ่ม Upload จะมีการตรวจสอบชนิดข้อมูลที่ต้องการนำเข้าฐานข้อมูล เพื่อเรียกไปยังหน้าสำหรับเพิ่มข้อมูลลงฐานข้อมูล โดยมีโค้ดตัวอย่างสำหรับการตรวจสอบชนิดข้อมูลที่ต้องการนำเข้าฐานข้อมูลและเรียกหน้าสำหรับเพิ่มข้อมูลลงฐานข้อมูล ดังตารางที่ 4.1

ตารางที่ 4.1 โค้ดตัวอย่างสำหรับการตรวจสอบชนิดข้อมูลที่ต้องการนำเข้าฐานข้อมูล

FILE: *Import/index.php*

```
<script type="text/javascript">
fnCheck(){
    var fm = document.fmupload;
    if(fm.select_table_name.value = = "tb_earthquake")          //earthquake data
    {
        fm.action = "uploadImport_earthquake.php";          // call upload earthquake data page
    }
    else if(fm.select_table_name.value = = "tb_temperature")    // temperature data
    {
        fm.action = "uploadImport_temperature.php";         // call upload temperature data page
    }
    fm.submit();
}
</script>
```

โดยในไฟล์สำหรับเพิ่มข้อมูลเข้าฐานข้อมูล (เช่น uploaadImport_earthquake.php) จะอ่านไฟล์ที่นำเข้ามาทีละบรรทัด และจะรวมคอลัมน์วันที่กับเวลาเป็นข้อมูลเดียวกัน จากนั้นจะนำข้อมูลทุกคอลัมน์มาสร้างเป็นอาร์เรย์ชื่อ $col_value โดย $col_value[1] คือ datetime และ $col_value[2], $col_value[3], … คือข้อมูลในคอลัมน์อื่น ๆ จากนั้นจะตรวจสอบข้อมูลที่จะนำเข้าฐานข้อมูลว่าซ้ำกับข้อมูลที่อยู่ในฐานข้อมูลหรือไม่ ถ้าข้อมูลที่จะนำเข้าไม่ซ้ำกับข้อมูลที่มีอยู่ในฐานข้อมูล ระบบจะเพิ่มข้อมูลใหม่ลงฐานข้อมูล โดยมีโค้ดตัวอย่างสำหรับการนำข้อมูลแผ่นดินไหวเข้าฐานข้อมูลดังตารางที่ 4.2

ตารางที่ 4.2 โค้ดตัวอย่างสำหรับการนำข้อมูลแผ่นดินไหวเข้าฐานข้อมูล

FILE: *Import/uploaadImport_earthquake.php*

```
$pathfile = $Fileuploadpath . $File_name;     // file name
$row = 0;                                     // row of data in file
$file = fopen($pathfile, "r");                // open file
while (($data = fgetcsv($file, 8000, ",")) !== FALSE)   // read file
{
    $num_column = count($data);           // count column in file
    $row++;
    for ($c=0; $c < $num_column; $c++)   // loop of column
    {
        $index = $c+1;                // index  of  '$col_value'
        if($index = = 2)              // set  datetime  format
        {
            $date = $data[$c-1];
            $time = $data[$c];
            $col_value[$index-1] = $date." ".$time;  // $col_value[1] : datetime
        }
        else if($index > 2)
        {
```

```
19                      $col_value[$index-1] = $data[$c];       // $col_value[2], $col_value[3], ... : other fields
20                  }
21          }
22
23          // check redundance event in db //
24          $sqlselect = "SELECT eq_id FROM tb_earthquake WHERE eq_datetime='$col_value[1]' AND
25                      eq_latitude='$col_value[3]' AND eq_longitude='$col_value[4]' AND
26                      geometry=PointFromText('POINT($col_value[4] $col_value[3])',4326) AND
27                      eq_magnitude='$col_value[5]' AND eq_detail='$col_value[6]'";
28          $result = pg_query($dbconn,iconv('tis-620','utf-8',$sqlselect)) or die("Cannot Query data" .
29                      pg_last_error());
30          if (!$result)                       // An error occured
31          {
32                  exit;
33          }
34          $n = pg_num_rows($result);     //amount of row of result
35
36          // insert new data from file
37          if($n= =0)
38          {
39                  $sql = "INSERT INTO tb_earthquake (eq_datetime, eq_focus, eq_latitude, eq_longitude,
40                          geometry, eq_magnitude, eq_detail) VALUES ('$col_value[1]', '$col_value[2]',
41                          '$col_value[3]',  '$col_value[4]', GeomFromText('POINT($col_value[4]
42                          $col_value[3])', 4326),'$col_value[5]', '$col_value[6]')";
43                  pg_exec($sql);
44          }
45  }
46  fclose($file);      // close file
```

การนำข้อมูลชนิดอื่นเข้าฐานข้อมูลเชิงพื้นที่

ในกรณีที่ต้องการนำข้อมูลชนิดอื่น ๆ เข้าฐานข้อมูล PostgreSQL จะต้องสร้างตารางใหม่สำหรับข้อมูลที่จะนำเข้าฐานข้อมูลก่อน โดยมีฟิลด์ต่าง ๆ ดังนี้

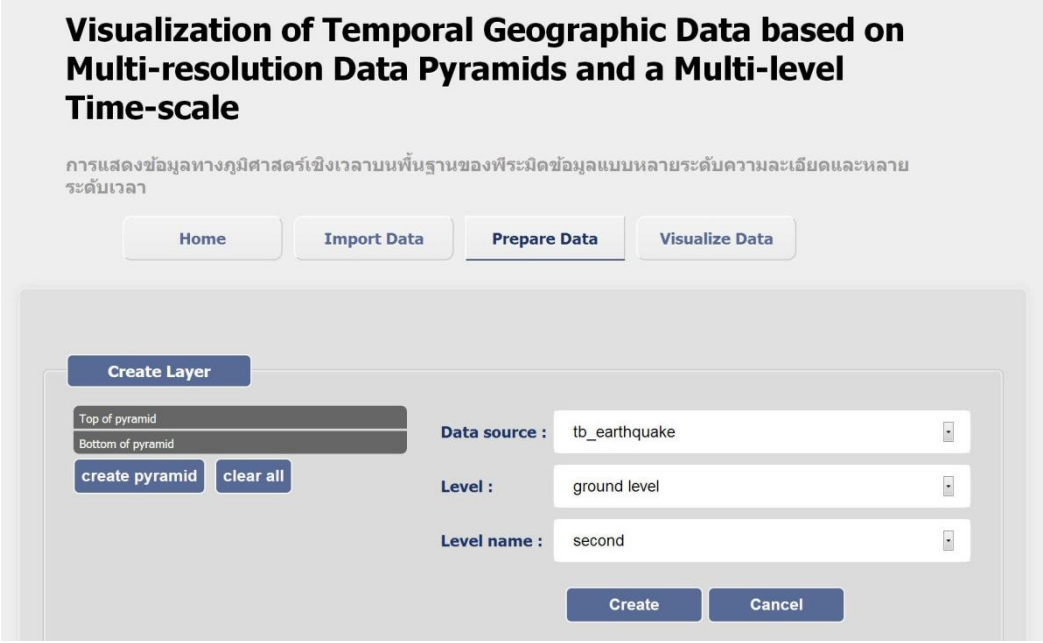
id	integer
datetime	timestamp with time zone
place name	character varying(254)
latitude	double precision
longitude	double precision
geometry	geometry
data	type of data

จากนั้นต้องเพิ่มรายละเอียดของข้อมูลชนิดใหม่ลงในตาราง tb_data_store (ds_id, ds_table_name, ds_data_name) โดยเพิ่มชื่อตารางของข้อมูลใหม่ที่ได้สร้างไปเมื่อสักครู่ และเพิ่มชื่อข้อมูลใหม่ที่ต้องการนำเข้าฐานข้อมูล

เมื่อสร้างฐานข้อมูลแล้ว ต้องสร้างไฟล์สำหรับเพิ่มข้อมูลเข้าฐานข้อมูลเหมือนกับไฟล์ uploaadImport_earthquake.php ในข้างต้น โดยเปลี่ยนแปลงโค้ดสำหรับการตรวจสอบข้อมูลซ้ำในบรรทัดที่ 24 และโค้ดสำหรับการเพิ่มข้อมูลเข้าฐานข้อมูลในบรรทัดที่ 39 จากนั้นต้องเพิ่มโค้ดสำหรับการตรวจสอบชนิดข้อมูลที่ต้องการนำเข้าฐานข้อมูลที่อยู่ในหน้าหลักของการเพิ่มข้อมูล (Import/index.php)

4.1.2 เครื่องมือสำหรับสร้างพีระมิดข้อมูล

หลังจากที่นำข้อมูลเข้าฐานข้อมูลแล้ว ในขั้นตอนถัดไปของการเตรียมข้อมูลคือการสร้างพีระมิดข้อมูล โดยผู้วิจัยได้พัฒนาเครื่องมือสำหรับสร้างพีระมิดข้อมูลของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิบนเว็บแอพพลิเคชัน โดยในหน้าเว็บดังภาพที่ 4.5

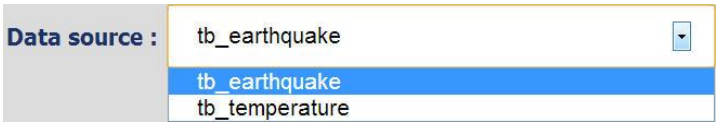


ภาพที่ 4.5 หน้าเว็บสำหรับการเตรียมข้อมูล

หน้าหลักของการสร้างพีระมิดข้อมูลจะประกอบไปด้วยส่วนสำหรับแสดงพีระมิดข้อมูลที่กำลังสร้าง ซึ่งจะปรากฏในส่วนซ้ายมือของภาพที่ 4.5 และส่วนสำหรับการสร้างชั้นพีระมิดข้อมูลดังที่แสดงในด้านขวามือของภาพที่ 4.5

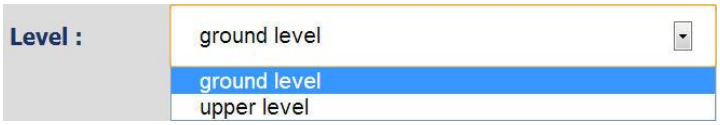
ในส่วนของการสร้างชั้นพีระมิดข้อมูลจะประกอบไปด้วย 3 ส่วนดังนี้

1. Data source จะแสดงรายชื่อตารางที่เก็บข้อมูล เพื่อให้ผู้ใช้เลือกชนิดข้อมูลที่จะนำมาสร้างพีระมิดข้อมูล ดังภาพที่ 4.6



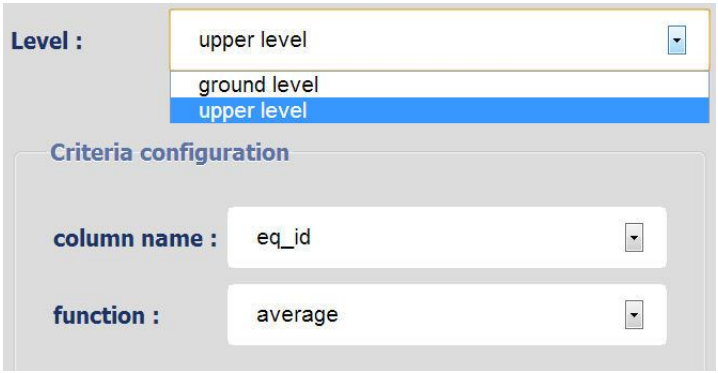
ภาพที่ 4.6 ส่วนแสดง Data source ของหน้าเว็บสำหรับการเตรียมข้อมูล

2. Level จะให้ผู้ใช้เลือกระหว่างชั้นพีระมิดชั้นล่างสุดหรือชั้นบนของพีระมิดข้อมูล ดังภาพที่ 4.7



ภาพที่ 4.7 ส่วนแสดง Level ของหน้าเว็บสำหรับการเตรียมข้อมูล

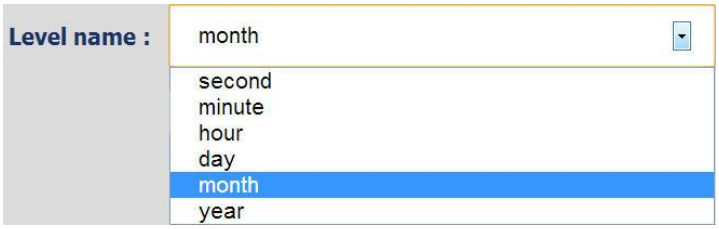
ในกรณีที่ต้องการสร้างชั้นล่างสุดของพีระมิดข้อมูล จะต้องเลือก ground level แต่ในกรณีที่ต้องการสร้างชั้นบนของพีระมิดข้อมูล จะต้องเลือก upper level แล้วจะมีส่วนสำหรับตั้งค่าในการสร้างชั้นพีระมิดดังภาพที่ 4.8



ภาพที่ 4.8 ส่วนแสดง Criteria configuration ของหน้าเว็บสำหรับการเตรียมข้อมูล

ในส่วนสำหรับตั้งค่าในการสร้างชั้นพีระมิด จะมีชื่อคอลัมน์และฟังก์ชันสำหรับเป็นเกณฑ์ในการสร้างชั้นพีระมิดข้อมูล โดยชื่อคอลัมน์จะไปดึงมาจากตารางในฐานข้อมูล และชื่อฟังก์ชันมีดังนี้ average, minimum, maximum, greater than, less than และ between สำหรับฟังก์ชัน greater than และ less than จะมี threshold ให้ผู้ใช้ระบุค่าตัวเลขที่ต้องการ และสำหรับฟังก์ชัน between ผู้ใช้จะต้องระบุค่าเป็นช่วงตัวเลขที่ต้องการ

3. Level name จะแสดงรายชื่อชั้นพีระมิดข้อมูลในระดับความละเอียดของเวลาต่าง ๆ เช่น second, minute, hour, day, month และ year เพื่อให้ผู้ใช้เลือกระดับความละเอียดของเวลาที่ต้องการสร้างชั้นข้อมูล ดังภาพที่ 4.9



ภาพที่ 4.9 ส่วนแสดง Level name ของหน้าเว็บสำหรับการเตรียมข้อมูล

เมื่อผู้ใช้กดปุ่ม Create จะเป็นการสร้างชั้นพีระมิดข้อมูลทีละชั้น โดยจะไปเรียกใช้ฟังก์ชัน createLayer() ซึ่งในฟังก์ชันนี้จะไปเรียกไฟล์ createLayer.php พร้อมส่งค่าพารามิเตอร์ที่ได้เลือกในข้างต้นไปเพื่อสร้างเป็นชั้นพีระมิดข้อมูลมาแสดงในส่วนซ้ายมือ เมื่อสร้างพีระมิดข้อมูลครบทุกชั้น ผู้ใช้จะกดปุ่ม create pyramid เพื่อสร้าง view ใน PostgreSQL โดยจะไปเรียกใช้ฟังก์ชัน createPyramid() ซึ่งในฟังก์ชันนี้จะไปเรียกไฟล์ createPyramid.php พร้อมส่งค่าพารามิเตอร์ที่ได้เลือกไปเพื่อสร้าง view สำหรับแต่ละชั้นของพีระมิดข้อมูล โดยในการสร้าง view จะใช้ sql query ข้อมูลจากฐานข้อมูลตามเกณฑ์ที่ต้องการของแต่ละชั้นพีระมิดข้อมูล โดยมีโค้ดตัวอย่างในการสร้าง view ของข้อมูลแผ่นดินไหวดังตารางที่ 4.3 และข้อมูลอุณหภูมิดังตารางที่ 4.4

ตารางที่ 4.3 โค้ดตัวอย่างสำหรับการสร้าง view ใน PostgreSQL ของข้อมูลแผ่นดินไหว

FILE: *Prepare/createPyramid.php (earthquake)*

```
1   if($function = = "greater")              // function: greater than
2   {
3           $view_sql = "CREATE VIEW $view_name AS SELECT * FROM $table WHERE $column >=
4           $threshold";
5   }
6   else if($function = = "less")            // function: less than
7   {
8           $view_sql = "CREATE VIEW $view_name AS SELECT * FROM $table WHERE $column <=
9           $threshold";
10  }
11  else if($function = = "between")         // function: between
12  {
13          $view_sql = "CREATE VIEW $view_name AS SELECT * FROM $table WHERE $column
14          between $from and $to";
15  }
```

ตารางที่ 4.4 โค้ดตัวอย่างสำหรับการสร้าง view ใน PostgreSQL ของข้อมูลอุณหภูมิ

FILE: *Prepare/createPyramid.php (temperature)*

```
1   if($function == "maximum")               // function: maximum
2   {
3               $view_sql = "CREATE VIEW $view_name AS
4                           SELECT tm_datetime, tm_station_en, tm_latitude, tm_longitude,
```

```
5                       MAX(tm_temperature) AS tm_temperature, geometry
6                       FROM tb_temperature
7                       GROUP BY tm_station_en, pyramid_level ";
8   }
9   else if($function == "minimum")       // function: minimum
10  {
11          $view_sql = "CREATE VIEW $view_name AS
12                      SELECT tm_datetime, tm_station_en, tm_latitude, tm_longitude,
13                      MIN(tm_temperature) AS tm_temperature, geometry
14                      FROM tb_temperature
15                      GROUP BY tm_station_en, pyramid_level ";
16  }
17  else if($function == "average")         // function: average
18  {
19          $view_sql = "CREATE VIEW $view_name AS
20                      SELECT tm_datetime, tm_station_en, tm_latitude, tm_longitude,
21                      AVG(tm_temperature) AS tm_temperature, geometry
22                      FROM tb_temperature
23                      GROUP BY tm_station_en, pyramid_level ";
24  }
```

การสร้างชั้นพีระมิดข้อมูลในระดับความละเอียดอื่น ๆ

ถ้าหากผู้ดูแลระบบต้องการสร้างพีระมิดข้อมูลในระดับความละเอียดอื่น ๆ เช่น ต้องการดูข้อมูลเป็นไตรมาส แทนที่จะดูข้อมูลเป็นรายวัน, รายเดือน หรือรายปี เป็นต้น ซึ่งการดูข้อมูลเป็นไตรมาสเสมือนเป็นการดูข้อมูลทีละ 3 เดือน ดังนั้นผู้ดูแลระบบจะต้องแก้ไขโค้ดในส่วนของ Level name ให้สามารถระบุได้ว่าจะเลือกดูข้อมูลทีละกี่วัน หรือดูข้อมูลทีละกี่เดือน หรือดูข้อมูลทีละกี่ปี และแก้ไขโค้ดในไฟล์ createPyramid.php สำหรับการสร้าง view โดยแก้ไข sql ในการ query ข้อมูลจากฐานข้อมูลให้ดึงข้อมูลเป็นช่วงเวลาที่ต้องการ เช่น query ข้อมูลทีละ 3 เดือน, query ข้อมูลทีละ 6 เดือน หรือ query ข้อมูลทีละ 10 ปี เป็นต้น

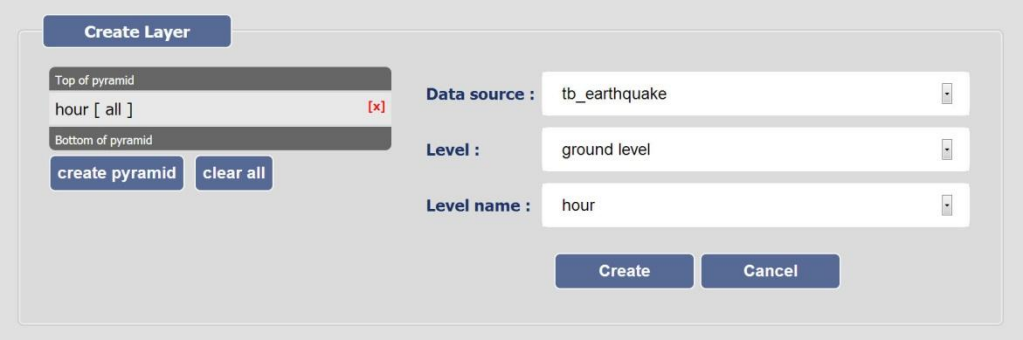
การสร้างฟังก์ชันในการสร้างพีระมิดข้อมูลฟังก์ชันอื่น

ถ้าหากผู้ดูแลระบบต้องการสร้างพีระมิดข้อมูลโดยใช้ฟังก์ชันอื่น ๆ เป็นเกณฑ์ในการสร้างชั้นข้อมูล เช่น ฟังก์ชันไม่เท่ากับ หรือ ฟังก์ชันหาผลรวม เป็นต้น ผู้ดูแลระบบจะต้องแก้ไขโค้ดในส่วนของ function โดยเพิ่มชื่อฟังก์ชันอื่น ๆ เข้าไปเป็นตัวเลือกเพิ่มเติม และแก้ไขโค้ดในไฟล์ createPyramid.php โดยเพิ่มฟังก์ชันอื่น ๆ และเขียน sql สำหรับฟังก์ชันที่เพิ่มเข้าไปใหม่

4.2 การใช้งานเครื่องมือสำหรับการเตรียมข้อมูล

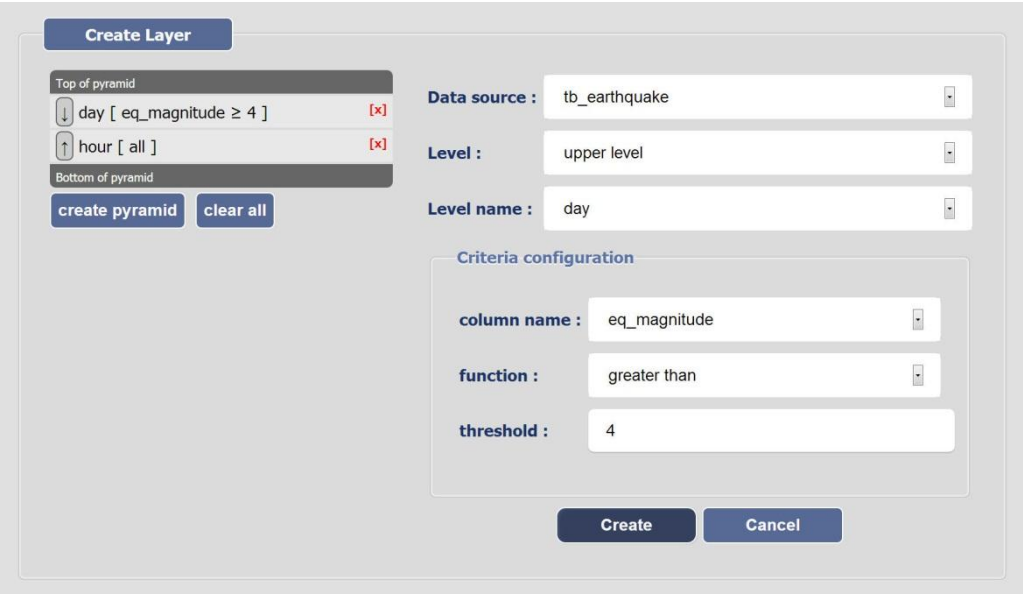
ในการสร้างชั้นข้อมูลของพีระมิดข้อมูลจะเริ่มสร้างจากชั้นล่างสุดของพีระมิดข้อมูล โดยให้เลือก level เป็น ground level และเลือกระดับความละเอียดของเวลาในชั้นล่างสุดที่ level name

ซึ่งในตัวอย่างนี้จะสร้างชั้นล่างสุดที่มีระดับความละเอียดของเวลาเป็นชั่วโมง ดังนั้นจึงเลือก level name เป็น hour จากนั้นกดปุ่ม create ดังภาพที่ 4.10



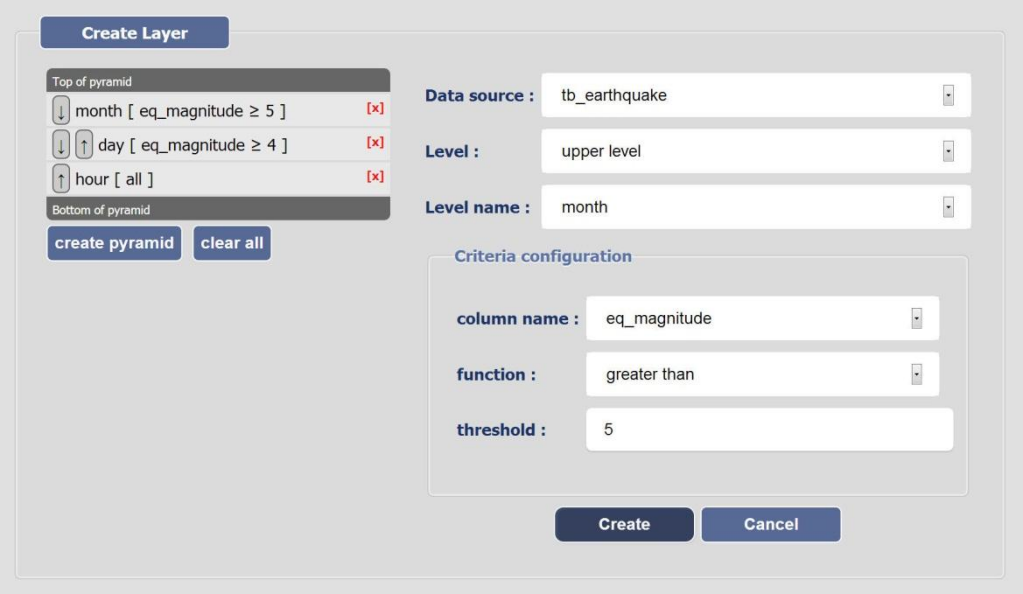
ภาพที่ 4.10 การสร้างชั้นล่างสุดของพีระมิดข้อมูล

จากนั้นจะเริ่มสร้างจากชั้นบนของพีระมิดข้อมูล โดยให้เลือก level เป็น upper level แล้วจะปรากฏในส่วนของ criteria configuration ขึ้นมา ซึ่งเป็นการเลือกคอลัมน์และฟังก์ชันในการคัดกรองข้อมูลดังที่ได้กล่าวในบทก่อนหน้า โดยในตัวอย่างเลือกนี้จะสร้างชั้นที่ 2 เป็นระดับวัน และจะเลือกข้อมูลที่มีระดับความรุนแรงของแผ่นดินไหวมากกว่า 4 ริกเตอร์มาเป็นข้อมูลในชั้นนี้ ดังนั้นจึงต้องเลือก level name เป็น day, column name เป็น eq_magnitude คือระดับความรุนแรงของแผ่นดินไหว และ function เลือกเป็น greater than พร้อมกับกำหนด threshold เป็น 4 จากนั้นกดปุ่ม create โดยระบบจะทำการสร้างพีระมิดข้อมูลในชั้นที่ 2 ขึ้นมาให้ซึ่งจะแสดงอยู่ทางด้านซ้ายมือ ดังภาพที่ 4.11



ภาพที่ 4.11 การสร้างชั้นที่ 2 ของพีระมิดข้อมูล

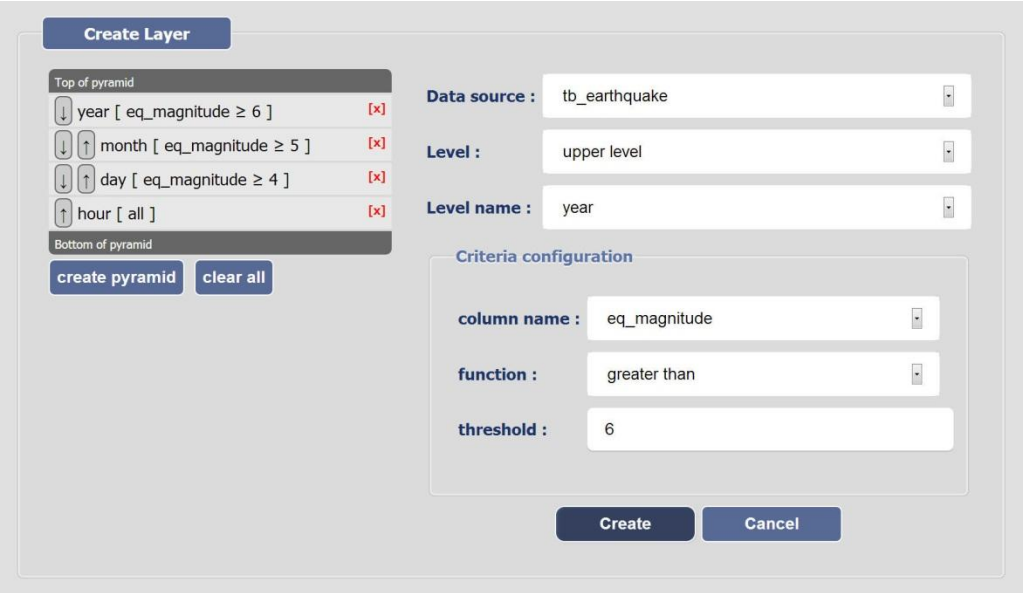
จากนั้นจะเริ่มสร้างจากชั้นที่ 3 ของพีระมิดข้อมูล ซึ่งเลือกเหมือนในการสร้างชั้นที่ 2 แต่เปลี่ยนในส่วน level name และ threshold ซึ่งในตัวอย่างนี้จะสร้างชั้นที่ 3 ในระดับความละเอียดของเวลาเป็นเดือน และจะเลือกข้อมูลที่มีระดับความรุนแรงของแผ่นดินไหวมากกว่า 5 ริกเตอร์มาเป็นข้อมูลในชั้นนี้ ดังนั้นจึงเลือก level name เป็น month และเลือก column name และ function ตามเดิม พร้อมกับกำหนด threshold เป็น 5 จากนั้นกดปุ่ม create ดังภาพที่ 4.12



ภาพที่ 4.12 การสร้างชั้นที่ 3 ของพีระมิดข้อมูล

จากนั้นจะเริ่มสร้างจากชั้นที่ 4 ของพีระมิดข้อมูล ซึ่งเลือกเหมือนในการสร้างชั้นที่ 2 แต่เปลี่ยนในส่วน level name และ threshold ซึ่งในตัวอย่างนี้จะสร้างชั้นที่ 4 ในระดับความละเอียดของเวลาเป็นปี ซึ่งเป็นชั้นบนสุดของพีระมิดข้อมูล และจะเลือกข้อมูลที่มีระดับความรุนแรงของแผ่นดินไหวมากกว่า 6 ริกเตอร์มาเป็นข้อมูลในชั้นนี้ ดังนั้นจึงเลือก level name เป็น year และเลือก column name และ function ตามเดิม พร้อมกับกำหนด threshold เป็น 6 จากนั้นกดปุ่ม create

เมื่อสร้างชั้นข้อมูลครบทุกชั้นแล้วให้ทำการกดปุ่ม create pyramid ในด้านซ้ายมือ เพื่อทำการยืนยันในการสร้างชั้นข้อมูล ดังภาพที่ 4.13 โดยระบบจะนำแต่ละชั้นของพีระมิดไปสร้างไปสร้าง view ในฐานข้อมูล PostgreSQL เพื่อนำไปสร้างเป็นชั้นข้อมูลบน GeoServer ในลำดับถัดไป



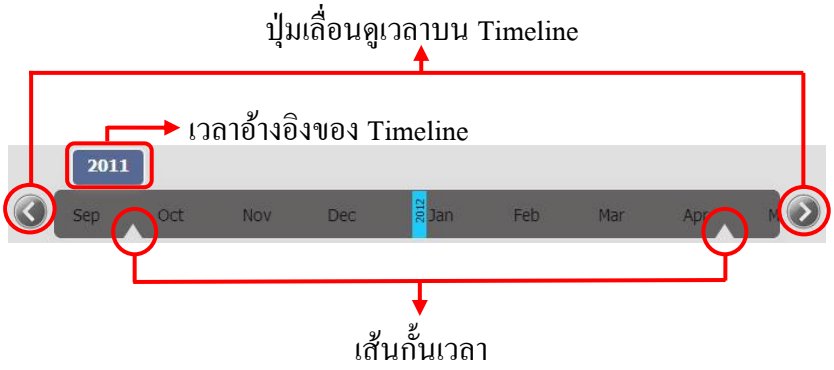
ภาพที่ 4.13 การสร้างชั้นที่ 4 ของพีระมิดข้อมูล

บทที่ 5
การพัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลา และการพัฒนาระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา

หลังจากที่ได้อธิบายถึงการพัฒนาเครื่องมือสำหรับการเตรียมข้อมูลไปในบทที่แล้ว ในบทนี้จะกล่าวถึงการพัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลาเพื่อนำไปใช้ในการพัฒนาระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา โดยในส่วนเนื้อหานั้นจะเริ่มต้นด้วยการพัฒนาเครื่องมือ Timeline การทำงานของเครื่องมือ Timeline และการนำเครื่องมือ Timeline ไปใช้ในการพัฒนาหน้าเว็บสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา และในส่วนท้ายจะอธิบายการพัฒนาระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา

5.1 การพัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลา

Timeline ที่พัฒนาขึ้นมาในงานวิจัยนี้รองรับหลายระดับความละเอียดของเวลา โดยจะพัฒนาเป็น JavaScript เพื่อสามารถนำไปใช้ทำงานร่วมกับเว็บแอพพลิเคชันทั่วไปได้ โดย Timeline ที่พัฒนาขึ้นมาเป็นดังภาพที่ 5.1



ภาพที่ 5.1 ตัวอย่าง Timeline ที่พัฒนาขึ้นมา

จากภาพตัวอย่าง Timeline ที่พัฒนาขึ้นมาจะประกอบไปด้วยส่วนประกอบต่าง ๆ ดังนี้

- เส้นกั้นเวลา มีไว้สำหรับให้ผู้ใช้งานเลือกช่วงเวลาในการดูข้อมูล ซึ่งจะมีเส้นกั้นเวลา 2 อันเอาไว้สำหรับเลือกเวลาเริ่มต้นและเวลาสิ้นสุด โดยการกดเลื่อนเส้นกั้นเวลาไปยังตำแหน่งเวลาที่ต้องการ
- ปุ่มเลื่อนดูเวลาบน Timeline มีไว้สำหรับให้ผู้ใช้งานเลื่อนเวลาที่แสดงบน Timeline ไปยังเวลาที่ต้องการ โดยสามารถเลื่อนเวลาไปมาได้ทั้งด้านขวาและซ้าย

- เวลาอ้างอิงของ Timeline จะเป็นเวลาของตัวที่อยู่ด้านซ้ายสุดของ Timeline มีไว้สำหรับอ้างอิงเวลาเมื่อซูม Timeline เข้าไปในระดับความละเอียดของเวลาที่มาก ๆ แล้วเมื่อต้องการซูมออกมายังระดับความละเอียดของเวลาที่น้อย ๆ จะสามารถทำได้โดยเลือกที่เวลาอ้างอิงของ Timeline

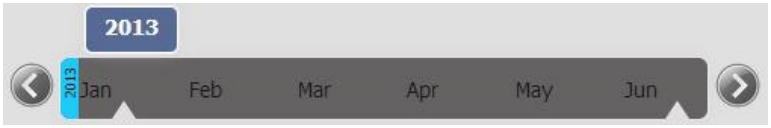
5.1.1 ฟังก์ชันในการทำงานของ Timeline

Timeline ที่พัฒนาขึ้นมามีฟังก์ชันหลัก ๆ ในการทำงานดังต่อไปนี้

1. ฟังก์ชันสำหรับสร้างเวลาในแต่ละระดับความละเอียดของเวลา
    - ในการสร้างเวลาในระดับความละเอียดของเวลาเป็นปี จะสร้างเวลาเป็นปีบน Timeline ดังนี้ 2000, 2001, 2002, 2003, … ดังภาพที่ 5.2

ภาพที่ 5.2 Timeline ในระดับความละเอียดของเวลาเป็นปี

    - ในการสร้างเวลาในระดับความละเอียดของเวลาเป็นเดือน จะสร้างเวลาเป็นเดือนบน Timeline ดังนี้ Jan, Feb, Mar, Apr, … ดังภาพที่ 5.3

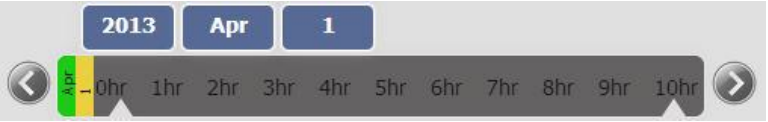


ภาพที่ 5.3 Timeline ในระดับความละเอียดของเวลาเป็นเดือน

    - ในการสร้างเวลาในระดับความละเอียดของเวลาเป็นวัน จะสร้างเวลาเป็นวันบน Timeline ดังนี้ 1, 2, 3, 4, … ดังภาพที่ 5.4

ภาพที่ 5.4 Timeline ในระดับความละเอียดของเวลาเป็นวัน

    - ในการสร้างเวลาในระดับความละเอียดของเวลาเป็นชั่วโมง จะสร้างเวลาเป็นชั่วโมงบน Timeline ดังนี้ 0hr, 1hr, 2hr, 3hr, … ดังภาพที่ 5.5



ภาพที่ 5.5 Timeline ในระดับความละเอียดของเวลาเป็นชั่วโมง

2. ฟังก์ชันในการดึงชั้นข้อมูลจาก GeoServer มาสร้างเป็นชั้นข้อมูลบน openLayers

ในการดึงชั้นข้อมูลจาก GeoServer จะต้องทราบช่วงเวลาที่ต้องการในการดึงข้อมูล โดย Timeline จะตรวจสอบช่วงเวลาที่เลือกอยู่ในขณะนั้น แล้วส่งช่วงเวลาไปดึงข้อมูลจาก GeoServer โดยใช้ SQL view ในการดึงชั้นข้อมูลตามช่วงเวลาที่ต้องการ ซึ่งก่อนการสร้างชั้นข้อมูลใหม่บนแผนที่ จะต้องลบชั้นข้อมูลเดิมที่เคยแสดงอยู่บนแผนที่ออกก่อน โดยมีโค้ดตัวอย่างของฟังก์ชัน createLayer() ซึ่งเป็นฟังก์ชันในการดึงชั้นข้อมูลจาก GeoServer มาสร้างเป็นชั้นข้อมูลบน openLayers เพื่อแสดงบนแผนที่ ดังตารางที่ 5.1

ตารางที่ 5.1 โค้ดตัวอย่างของฟังก์ชันในการดึงชั้นข้อมูลจาก GeoServer

FUNCTION: *createLayer*

```
1   map.removeLayer(layer_geoserver[0]);
2   layer_geoserver[0] = new OpenLayers.Layer.WMS(
3                           "query",
4                           "http://localhost:8081/geoserver/wms?viewparams =
5                                 where:where eq_datetime between '2013-01-01 000000'
6                                 and '2013-02-01 000000'",
7                           {layers: "VTGD:view_tb_earthquake_seq4_day_1",
8                                 transparent: "true", format: "image/png"},
9                           {visibility: true}
10                          );
11   map.addLayers([layer_geoserver[0]]);
```

- บรรทัดที่ 2 คือ การลบชั้นข้อมูลบนแผนที่ของ openLayers ซึ่งชั้นข้อมูลในตัวอย่างนี้แทนด้วยตัวแปร layer_geoserver[0]
- บรรทัดที่ 3-11 คือ การดึงชั้นข้อมูลจาก GeoServer มาสร้างเป็นชั้นข้อมูลบน openLayers
- บรรทัดที่ 4 คือ ชื่อ title ของชั้นข้อมูล
- บรรทัดที่ 5 คือ url ของ wms ของเครื่องที่เป็น GeoServer ในตัวอย่างนี้คือ http://localhost:8081/geoserver/wms ตามด้วย ?viewparams = where:where eq_datetime between '2013-01-01 000000' and '2013-02-01 000000' จะเป็นการดึงชั้นข้อมูลจาก GeoServer ในช่วงวันที่ 1 มกราคม 2556 ถึงวันที่ 1 กุมภาพันธ์ 2556

- บรรทัดที่ 7 คือ ชื่อ layer ใน GeoServer ที่ต้องการดึงข้อมูล ในตัวอย่างนี้คือ VTGD:view_tb_earthquake_seq4_day_1
- บรรทัดที่ 8 คือ parameter ในการสร้างชั้นข้อมูลแบบ WMS บน openLayers
- บรรทัดที่ 9 คือ ให้ชั้นข้อมูลนี้แสดงบนแผนที่อัตโนมัติเมื่อถูกเพิ่มบนแผนที่
- บรรทัดที่ 11 คือ เพิ่มชั้นข้อมูลลงบนแผนที่

3. ฟังก์ชันสำหรับตรวจสอบการเปลี่ยนแปลงของช่วงเวลาที่เลือกดูข้อมูล

Timeline จะมีการตรวจสอบการเปลี่ยนแปลงช่วงเวลาที่เลือกดูข้อมูลอยู่ตลอดเวลา โดยช่วงเวลาจะมีการเปลี่ยนแปลงได้ 2 วิธี คือ

1. เมื่อมีการเลื่อนเวลาบน Timeline ไปทางด้านซ้ายหรือขวา ช่วงเวลาที่เลือกอยู่ ณ ขณะนั้นจะเปลี่ยนแปลงตามเวลาบน Timeline ที่เปลี่ยนไปด้วย โดย Timeline จะตรวจสอบช่วงเวลาและเปลี่ยนแปลงช่วงเวลาที่เลือกให้ตรงตามเวลาบน Timeline อยู่ตลอดเวลา

2. เมื่อมีการเลื่อนเส้นกั้นเวลา ช่วงเวลาที่เลือกอยู่ ณ ขณะนั้นจะเปลี่ยนแปลงตามไปด้วย ในการตรวจสอบการเลื่อนของเส้นกั้นเวลาจะใช้ mouseup โดยเมื่อหยุดกดเลื่อนเส้นกั้นเวลา ฟังก์ชัน mouseup จะทำงานโดยจะไปเรียกใช้ฟังก์ชันในการดึงชั้นข้อมูลจาก GeoServer ตามช่วงเวลาที่เลือกใหม่ เพื่อแสดงชั้นข้อมูลบนแผนที่ตามช่วงเวลาใหม่ โดยมีโค้ดตัวอย่างสำหรับการตรวจสอบการกดเลื่อนเส้นกั้นเวลา ดังตารางที่ 5.2

ตารางที่ 5.2 โค้ดตัวอย่างสำหรับการตรวจสอบการกดเลื่อนเส้นกั้นเวลา

FILE: *vtgdTimeline-TimeChange-Control.js*

```
1  $("#tl_left-triangle").mouseup(function(){
2          setTimeout("createLayer()",100);
3  });
4  $("#tl_right-triangle").mouseup(function(){
5          setTimeout("createLayer()",100);
6  });
```

- บรรทัดที่ 1-3 คือ การตรวจสอบการเลื่อนของเส้นกั้นเวลาด้านซ้าย
- บรรทัดที่ 4-6 คือ การตรวจสอบการเลื่อนของเส้นกั้นเวลาด้านขวา
- บรรทัดที่ 2และ5 คือ การเรียกใช้ฟังก์ชัน createLayer() ซึ่งเป็นฟังก์ชันในการดึงชั้นข้อมูลจาก GeoServer ตามช่วงเวลาที่เลือกมาแสดงบนแผนที่

4. ฟังก์ชันสำหรับตรวจสอบการเลื่อนเวลาบน Timeline

Timeline จะมีการตรวจสอบการเลื่อนเวลาบน Timeline อยู่ตลอดเวลา โดยจะมีการเลื่อนเวลาอยู่ 2 รูปแบบ คือ เลื่อนเวลาไปทางซ้าย และเลื่อนเวลาไปทางขวา โดยการกดปุ่มเลื่อนดูเวลาทางด้านซ้ายหรือขวาของ Timeline ซึ่งในการกดปุ่มเลื่อนดูเวลาทางด้านซ้ายหรือขวา Timeline จะมีการทำงานอยู่ 3 อย่าง ดังนี้

1. เมื่อกดปุ่มเลื่อนดูเวลาทางซ้ายหรือขวา จะมีการเรียกใช้ฟังก์ชันสร้างเวลาใหม่เพื่อสร้างเวลาต่อจากเวลาที่แสดงอยู่บน Timeline ตัวอย่างเช่น จากภาพที่ 5.2 เป็น Timeline ในระดับความละเอียดของเวลาเป็นปี ซึ่งมีปีที่แสดงอยู่บน Timeline เป็น 2006-2013 เมื่อผู้ใช้งานกดปุ่มเลื่อนดูเวลาทางด้านซ้าย ฟังก์ชันสร้างเวลาจะสร้างเวลาเป็นปี 2005 2004 2003 … มาแสดงบน Timeline แต่ถ้าผู้ใช้งานกดปุ่มเลื่อนดูเวลาทางด้านขวา ฟังก์ชันสร้างเวลาจะสร้างเวลาเป็นปี 2014 2015 2016 … มาแสดงบน Timeline

2. ในขณะที่กดปุ่มเลื่อนดูเวลาบน Timeline จะมีการตรวจสอบเวลาอ้างอิงของ Timeline ให้ตรงกับเวลาที่แสดงอยู่ด้านซ้ายสุดของ Timeline อยู่ตลอดเวลา

3. เมื่อหยุดกดปุ่มเลื่อนดูเวลาบน Timeline จะทำการเรียกใช้ฟังก์ชันในการดึงชั้นข้อมูลจาก GeoServer เพื่อแสดงชั้นข้อมูลใหม่ตามช่วงเวลาและระดับความละเอียดของเวลาที่เลือก โดยจะใช้ฟังก์ชัน mouseup ในการตรวจสอบการหยุดกดปุ่มเลื่อนดูเวลาบน Timeline โดยเมื่อหยุดกดปุ่มเลื่อนดูเวลาบน Timeline ฟังก์ชัน mouseup จะทำงานโดยจะไปเรียกใช้ฟังก์ชันในการดึงชั้นข้อมูลจาก GeoServer ตามช่วงเวลาที่เลือก เพื่อแสดงชั้นข้อมูลบนแผนที่ตามช่วงเวลาใหม่ โดยมีโค้ดตัวอย่างสำหรับการตรวจสอบการกดปุ่มเลื่อนดูเวลาบน Timeline ดังตารางที่ 5.3

ตารางที่ 5.3 โค้ดตัวอย่างสำหรับการตรวจสอบการกดปุ่มเลื่อนดูเวลาบน Timeline

FILE: *vtgdTimeline-TimeChange-Control.js*

```
1  $("#btnLeft").mouseup(function(){
2          setTimeout("createLayer()",100);
3  });
4  $("#btnRight").mouseup(function(){
5          setTimeout("createLayer()",100);
6  });
```

- บรรทัดที่ 1-3 คือ การตรวจสอบการกดปุ่มเลื่อนดูเวลาบน Timeline ทางด้านซ้าย

- บรรทัดที่ 4-6 คือ การตรวจสอบการกดปุ่มเลื่อนดูเวลาบน Timeline ทางด้านขวา

บรรทัดที่ 2และ5 คือ การเรียกใช้ฟังก์ชัน createLayer() ซึ่งเป็นฟังก์ชันในการดึงชั้นข้อมูลจาก GeoServer ตามช่วงเวลาที่เลือกมาแสดงบนแผนที่

5. ฟังก์ชันสำหรับตรวจสอบการเปลี่ยนแปลงระดับความละเอียดของเวลา

Timeline จะมีการตรวจสอบการเปลี่ยนแปลงระดับความละเอียดของเวลาอยู่ตลอดเวลา โดยในการเปลี่ยนแปลงระดับความละเอียดของเวลาจะมีอยู่ 2 รูปแบบคือ

- การเปลี่ยนแปลงระดับความละเอียดของเวลาจากความละเอียดน้อยไปมาก (zoom in)
- การเปลี่ยนแปลงระดับความละเอียดของเวลาจากความละเอียดมากไปน้อย (zoom out)

เมื่อมีการเปลี่ยนแปลงระดับความละเอียดของเวลา ระบบจะไปเรียกใช้ฟังก์ชันในการสร้างเวลาใหม่ โดย Timeline จะเปลี่ยนระดับความละเอียดของเวลาเป็นระดับความละเอียดของเวลาตามที่ผู้ใช้งานเลือก และจะเรียกใช้ฟังก์ชันในการดึงข้อมูลจาก GeoServer ขึ้นมาใหม่เพื่อแสดงชั้นข้อมูลใหม่ตามช่วงเวลาและระดับความละเอียดของเวลาที่เลือก

5.1.2 การนำ Timeline ไปใช้งาน

ในการใช้งาน Timeline ที่พัฒนาขึ้นมานี้ ผู้ใช้งานจะต้องสร้างส่วนสำหรับแสดง Timeline บนหน้าเว็บแอพพลิเคชัน และต้องนำไฟล์ JavaScript ของ Timeline ทั้งหมดมาใส่บนหน้าเว็บแอพพลิเคชันด้วย โดยมีโค้ดตัวอย่างสำหรับการสร้างส่วนสำหรับแสดง Timeline บนหน้าเว็บแอพพลิเคชันดังตารางที่ 5.4

ตารางที่ 5.4 โค้ดตัวอย่างสำหรับการสร้างส่วนสำหรับแสดง Timeline

FILE: *index.php*

```
1  <div id="TL">
2      <table width="100%" border="0" cellspacing="0" cellpadding="0">
3          <tr>
4              <td width="30px" height="30px" align="left">
5                  <a onClick="" style="cursor:pointer;" id="btnLeft">;
6                      <img src="../../images/Left_grey.png" width="30px" height="30px"
7                      onMouseOver="this.src='../../images/Left_blue.png'"
8                      onMouseOut="this.src='../../images/Left_grey.png'"
```

```
 9                          onMouseDown="ClickLeft()"   onMouseUp="ClickLeft()" >
10                      </a>
11                  </td>
12                  <td width="5px"> </td>
13                  <td>
14                      <div id="time">
15                              <div id="showtime"></div>
16                      </div>
17                  </td>
18                  <td width="5px"> </td>
19                  <td  width="30px" height="30px" align="right">
20                      <a id="btnRight" onClick="" style="cursor:pointer;">
21                          <img src="../../images/Right_grey.png" width="30px" height="30px"
22                          onMouseOver="this.src='../../images/Right_blue.png'"
23                          onMouseOut="this.src='../../images/Right_grey.png'"
24                          onMouseDown="ClickRight()"   onMouseUp="ClickRight()">
25                      </a>
26                  </td>
27              </tr>
28          </table>
29      </div>
```

- บรรทัดที่ 1 คือ เป็นการสร้างส่วนสำหรับแสดง Timeline บนหน้าเว็บแอพพลิเคชัน
- บรรทัดที่ 2 คือ จะสร้างเป็นตารางโดยภายในตารางประกอบไปด้วย ปุ่มเลื่อนดูเวลาบน Timeline ด้านซ้าย, Timeline, ปุ่มเลื่อนดูเวลาบน Timeline ด้านขวา
- บรรทัดที่ 5-10 คือ การสร้างปุ่มเลือกดูเวลาบน Timeline ด้านซ้าย
- บรรทัดที่ 9 คือ เมื่อกดปุ่มเลื่อนดูเวลาบน Timeline ด้านซ้าย จะไปเรียกใช้ฟังก์ชัน ClickLeft() ซึ่งเป็นฟังก์ชันในการสร้างเวลาใหม่ในด้านซ้าย
- บรรทัดที่ 14-16 คือ การสร้างส่วนสำหรับแสดง Timeline
- บรรทัดที่ 20-25 คือ การสร้างปุ่มเลือกดูเวลาบน Timeline ด้านขวา
- บรรทัดที่ 24 คือ เมื่อกดปุ่มเลื่อนดูเวลาบน Timeline ด้านขวา จะไปเรียกใช้ฟังก์ชัน ClickRight() ซึ่งเป็นฟังก์ชันในการสร้างเวลาใหม่ในด้านขวา

เมื่อสร้างส่วนสำหรับแสดง Timeline บนหน้าเว็บแอพพลิเคชันแล้ว จากนั้นให้เรียกใช้ฟังก์ชันในการสร้าง Timeline ผ่าน JavaScript ดังตารางที่ 5.5

ตารางที่ 5.5 โค้ดตัวอย่างสำหรับการเรียกใช้ฟังก์ชันสำหรับสร้าง Timeline

FILE: *index.php*

```
1   <script type="text/javascript">
2           $(document).ready(function(){
3                   createTimeline();
4           });
5   </script>
```

ผลลัพธ์ที่ได้จากการสร้าง Timeline บนหน้าเว็บแอพพลิเคชัน แสดงได้ดังภาพที่ 5.6

ภาพที่ 5.6 การสร้าง Timeline บนหน้าเว็บแอพพลิเคชัน

การสร้าง Timeline ที่รองรับระดับความละเอียดของเวลาอื่น ๆ

ถ้าหากต้องการดูข้อมูลเป็นไตรมาส หรือดูข้อมูลทีละ 6 เดือน หรือดูข้อมูลทีละ 10 ปี จะต้องสร้างฟังก์ชันสำหรับสร้างเวลาในระดับความละเอียดของเวลาที่ต้องการเพิ่มเติม โดยต้องต้องเพิ่มโค้ดในไฟล์ vtgdTimeline.js โดยสร้างฟังก์ชันสำหรับสร้างเวลาในระดับความละเอียดอื่น ๆ บน Timeline เช่น หากต้องการแสดงข้อมูลทีละ 6 เดือน จะต้องสร้างเวลาที่แสดงบน Timeline เป็น Jan-June July-Dec หรือ หากต้องการแสดงข้อมูลทีละ 10 ปี จะต้องสร้างเวลาที่แสดงบน Timeline เป็น 1981-1990 1991-2000 2001-2010 2011-2020 2021-2030 เป็นต้น

5.2 การพัฒนาระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา

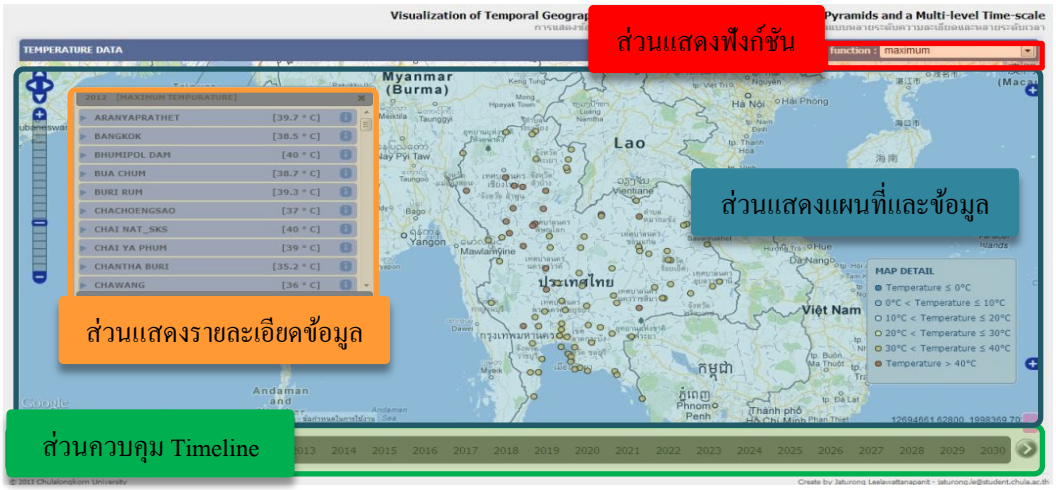
เมื่อเตรียมข้อมูลในส่วนของ Server และพัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลาเสร็จแล้ว ในลำดับถัดไปคือการนำข้อมูลมาแสดงผลบนระบบที่มี Timeline ที่รองรับหลายระดับความละเอียดเชิงเวลา โดยระบบจะพัฒนาเป็น Web Mapping Application สำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา ซึ่งมีรูปแบบ 2 รูปแบบ ดังนี้

1. Web Mapping Application สำหรับการสร้างพีระมิดข้อมูลตามความละเอียดของเวลา
2. Web Mapping Application สำหรับการสร้างพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล

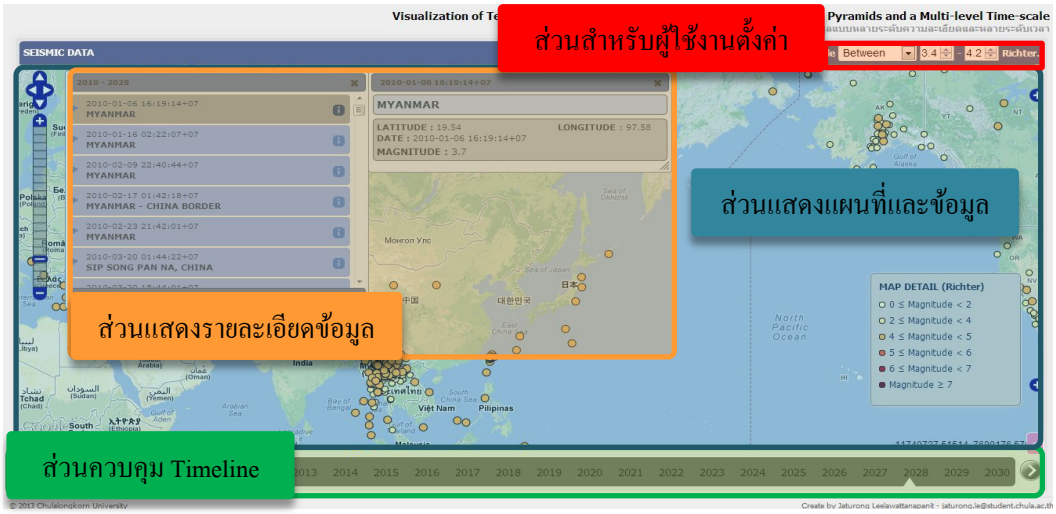
โดยมีโครงร่างของหน้าเว็บดังภาพที่ 5.7 ซึ่งมีส่วนประกอบ 4 ส่วนหลัก ๆ คือ

1. ส่วนแสดงแผนที่และข้อมูล ในส่วนนี้จะแสดงแผนที่และข้อมูลตามระดับความละเอียดของเวลาและช่วงเวลาที่เลือกดูข้อมูล
2. ส่วนแสดงรายละเอียดของข้อมูล ส่วนนี้จะแสดงรายละเอียดของข้อมูลของข้อมูลทั้งหมดที่แสดงอยู่บนแผนที่ ณ ขณะนั้น
3. ส่วนควบคุม Timeline ส่วนนี้จะเป็น Timeline ที่สามารถเปลี่ยนระดับความละเอียดของเวลาได้

4. ในส่วนนี้จะมีความแตกต่างกัน โดยสำหรับรูปแบบแรกจะแสดงชื่อฟังก์ชันที่ใช้ในการคัดเลือกหรือคำนวณข้อมูล สำหรับรูปแบบที่ 2 จะแสดงส่วนสำหรับให้ผู้ใช้งานตั้งค่าในการดูข้อมูลบนแผนที่



(ก)



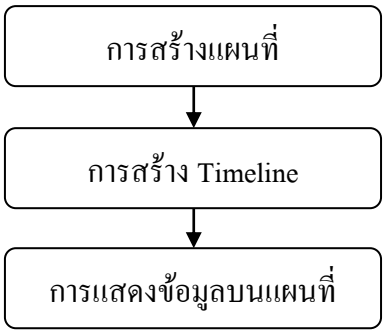
(ข)

ภาพที่ 5.7 โครงร่างของระบบต้นแบบ

(ก) สำหรับการสร้างพีระมิดข้อมูลตามความละเอียดของเวลา

(ข) สำหรับการสร้างพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูล

ในการพัฒนาระบบ Web Mapping Application นี้ได้แบ่งออกเป็น 3 ขั้นตอนหลัก ดังภาพที่ 5.8 โดยเริ่มจากการสร้างแผนที่ จากนั้นจะสร้างเครื่องมือ Timeline และในลำดับสุดท้ายจะนำชั้นข้อมูลมาแสดงผลบนแผนที่ โดยใช้ Timeline ในการเลือกดูข้อมูลในช่วงเวลาต่าง ๆ



ภาพที่ 5.8 ขั้นตอนการพัฒนาระบบ

โดยในหน้าเว็บแอพพลิเคชันหลักจะต้องสร้างส่วนสำหรับแสดงแผนที่ และส่วนสำหรับแสดง Timeline โดยมีโค้ดตัวอย่างสำหรับสร้างส่วนแสดงแผนที่และส่วนแสดง Timeline ดังตารางที่ 5.6

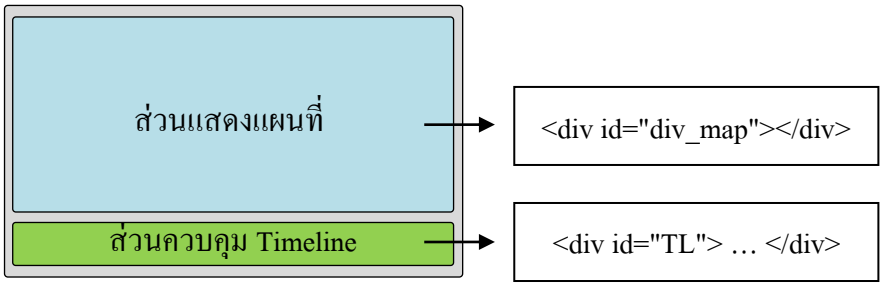
ตารางที่ 5.6 โค้ดตัวอย่างสำหรับสร้างส่วนแสดงแผนที่และส่วนแสดง Timeline

FILE: *index.php*

```
 1  <div id="div_map"></div>
 2  <div id="TL">
 3        <table width="100%" border="0" cellspacing="0" cellpadding="0">
 4        <tr>
 5              <td width="30px" height="30px" align="left">
 6                    <a onClick="" style="cursor:pointer;" id="btnLeft">
 7                          <img src="../../images/Left_grey.png" width="30px" height="30px"
 8                          onMouseOver="this.src='../../images/Left_blue.png'"
 9                          onMouseOut="this.src='../../images/Left_grey.png'"
10                          onMouseDown="ClickLeft()"   onMouseUp="ClickLeft()" >
11                    </a>
12              </td>
13              <td width="5px"> </td>
14              <td>
15                    <div id="time">
16                          <div id="showtime"></div>
17                    </div>
18              </td>
19              <td width="5px"> </td>
20              <td  width="30px" height="30px" align="right">
21                    <a id="btnRight" onClick="" style="cursor:pointer;">
22                          <img src="../../images/Right_grey.png" width="30px" height="30px"
23                          onMouseOver="this.src='../../images/Right_blue.png'"
24                          onMouseOut="this.src='../../images/Right_grey.png'"
25                          onMouseDown="ClickRight()"   onMouseUp="ClickRight()">
26                    </a>
27              </td>
28        </tr>
29        </table>
30  </div>
```

- บรรทัดที่ 1 คือ ส่วนสำหรับแสดงแผนที่
- บรรทัดที่ 2-30 คือ ส่วนสำหรับแสดง Timeline ซึ่งเหมือนกับ source code ในส่วนของการนำ Timeline ไปใช้งานที่อธิบายในหัวข้อก่อนหน้า

จากการสร้างส่วนสำหรับแสดงแผนที่และส่วนสำหรับแสดง Timeline จะได้โครงร่างของหน้าเว็บแอพพลิเคชันหลักดังภาพที่ 5.9



ภาพที่ 5.9 โครงร่างของหน้าเว็บแอพพลิเคชันหลัก

เมื่อสร้างส่วนสำหรับแสดงแผนที่ และส่วนสำหรับแสดง Timeline ในหน้าเว็บแอพพลิเคชันหลักเสร็จ ในลำดับถัดไปจะเป็นการสร้างแผนที่, การสร้าง Timeline และในลำดับสุดท้ายคือการนำข้อมูลมาแสดงบนแผนที่

5.2.1 การสร้างแผนที่

ในการพัฒนาระบบจะเริ่มจากการสร้างแผนที่ โดยในการแสดงแผนที่และข้อมูลบนเว็บแอพพลิเคชันจะใช้ OpenLayers ซึ่งเป็น API tool ที่ใช้งานง่าย สำหรับแสดงชั้นข้อมูล โดยที่ชั้นล่างสุดของข้อมูลจะเป็น base map ซึ่งจะใช้ Google Maps API เป็นชั้นของแผนที่ และในชั้นบนของแผนที่จะเป็นชั้นข้อมูลมาซ้อนทับบนแผนที่ โดยมีโค้ดตัวอย่างสำหรับการสร้าง base map บนชั้นข้อมูลของ openLayers ดังตารางที่ 5.7

ตารางที่ 5.7 โค้ดตัวอย่างสำหรับการสร้าง base map บนชั้นข้อมูลของ openLayers

FILE: *vtgdMap.js*

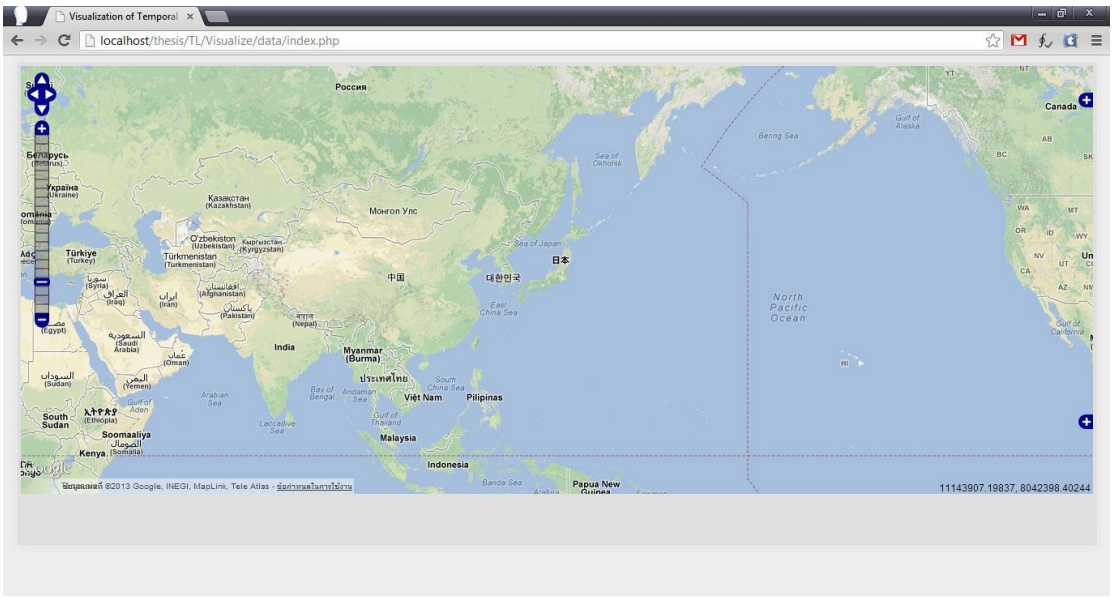
```
1  map = new OpenLayers.Map('div_map', {
2              projection: 'EPSG:900913',
3              layers: [
4                  new OpenLayers.Layer.Google(
5                          "Google Streets",
6                          {numZoomLevels: 20},
7                          {isBaseLayer: true}
```

```
 8                ),
 9                new OpenLayers.Layer.Google(
10                        "Google Physical",
11                        {type: google.maps.MapTypeId.TERRAIN}
12                ),
13                new OpenLayers.Layer.Google(
14                        "Google Hybrid",
15                        {type: google.maps.MapTypeId.HYBRID, numZoomLevels: 20}
16                ),
17                new OpenLayers.Layer.Google(
18                        "Google Satellite",
19                        {type: google.maps.MapTypeId.SATELLITE, numZoomLevels: 22}
20                )
21            ]
22    });
```

- บรรทัดที่ 1 คือ การเริ่มต้นสร้างชั้นแผนที่บน openLayers
- บรรทัดที่ 2 คือ กำหนด projection เป็น EPSG:900913 ซึ่งเป็นระบบพิกัดของ Google
- บรรทัดที่ 3 คือ การสร้างชั้นแผนที่ทั้งหมดบน openLayers
- บรรทัดที่ 4-19 คือ ชั้นแผนที่ทั้งหมดซึ่งมีทั้งหมด 4 ชั้นคือ Google Streets, Google Physical, Google Hybrid และ Google Satellite
- บรรทัดที่ 4-8 คือ ชั้นแผนที่ของ Google Streets ซึ่งกำหนดให้เป็น base map
- บรรทัดที่ 9-12 คือ ชั้นแผนที่ของ Google Physical ซึ่งเป็นชั้นแผนที่สร้างไว้เพิ่มเติมสำหรับให้ผู้ใช้งานสามารถเลือกชั้นแผนที่นี้เป็น base map ได้
- บรรทัดที่ 13-16 คือ ชั้นแผนที่ของ Google Hybrid ซึ่งเป็นชั้นแผนที่สร้างไว้เพิ่มเติมสำหรับให้ผู้ใช้งานสามารถเลือกชั้นแผนที่นี้เป็น base map ได้
- บรรทัดที่ 17-20 คือ ชั้นแผนที่ของ Google Satellite ซึ่งเป็นชั้นแผนที่สร้างไว้เพิ่มเติมสำหรับให้ผู้ใช้งานสามารถเลือกชั้นแผนที่นี้เป็น base map ได้

จากตัวอย่างการสร้าง base map จะมีการสร้างชั้นแผนที่ขึ้นมาโดยกำหนด projection เป็น EPSG:900913 คือเป็นระบบพิกัดของ Google และสร้างชั้นแผนที่ของ Google ขึ้นมา 4 ชั้นแต่กำหนดให้มี base map เพียงชั้นเดียวคือ ชั้นที่มีชื่อว่า Google Streets ซึ่งผู้ใช้งานสามารถเปลี่ยน base map เป็นชั้นแผนที่อื่น ๆ ได้ในภายหลัง

หน้าเว็บแอพพลิเคชันแสดงผลได้ดังภาพที่ 5.10 โดยจะมีเฉพาะในส่วนของแผนที่ ซึ่งยังไม่มีส่วนของการแสดง Timeline ซึ่งจะสร้างในลำดับถัดไป



ภาพที่ 5.10 หน้าเว็บแอพพลิเคชันหลักในส่วนของแผนที่

5.2.2 การสร้าง Timeline

เมื่อสร้างแผนที่เสร็จ ขั้นตอนถัดไปคือการสร้าง Timeline โดยจะนำเครื่องมือ Timeline ที่ได้สร้างไว้แล้วมาใช้งานร่วมกับการแสดงข้อมูลบนแผนที่ โดยผู้ใช้งานต้องนำไฟล์ JavaScript ของ Timeline ทั้งหมดไปใส่ในหน้าเว็บแอพพลิเคชันหลัก ดังตารางที่ 5.8

ตารางที่ 5.8 โค้ดตัวอย่างสำหรับการนำไฟล์ JavaScript ของ Timeline ทั้งหมดไปใส่ในหน้าเว็บแอพพลิเคชัน

FILE: *index.php*

```
1  <script type="text/javascript" src="../js/vtgdTimeline.js" charset="utf-8"></script>
2  <script type="text/javascript" src="../js/vtgdTimeline-Monitor.js" charset="utf-8"></script>
3  <script type="text/javascript" src="../js/vtgdTimeline-TimeChange-Control.js" charset="utf8">
4  </script>
5  <script type="text/javascript" src="../js/vtgdTimeline-TimeInterval-Control.js" charset="utf8">
6  </script>
```

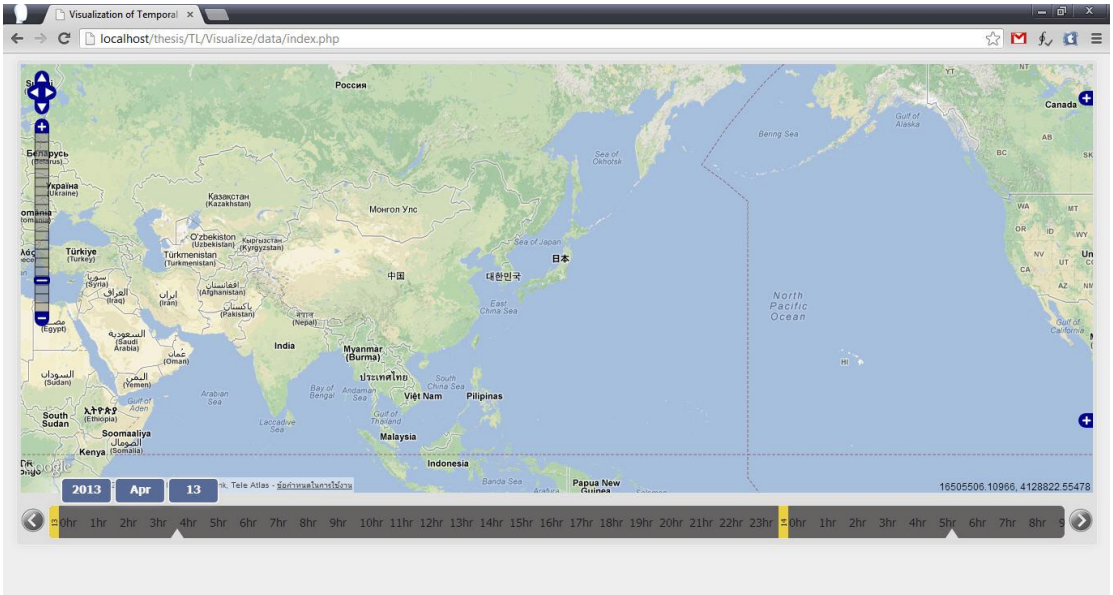
จากนั้นให้เรียกใช้ฟังก์ชันในการสร้าง Timeline ผ่าน JavaScript ดังโค้ดตัวอย่างในตารางที่ 5.9

ตารางที่ 5.9 โค้ดตัวอย่างของการเรียกใช้ฟังก์ชันในการสร้าง Timeline ผ่าน JavaScript

FILE: *index.php*

```
1  <script type="text/javascript">
2          $(document).ready(function(){
3                  createTimeline();
4          });
5  </script>
6
```

หน้าเว็บแอพพลิเคชันแสดงผลได้ดังภาพที่ 5.11 ซึ่งจะประกอบไปด้วยแผนที่และ Timeline สำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา ในลำดับถัดไปจะเป็นการนำชั้นข้อมูลมาแสดงบนเว็บแอพพลิเคชัน

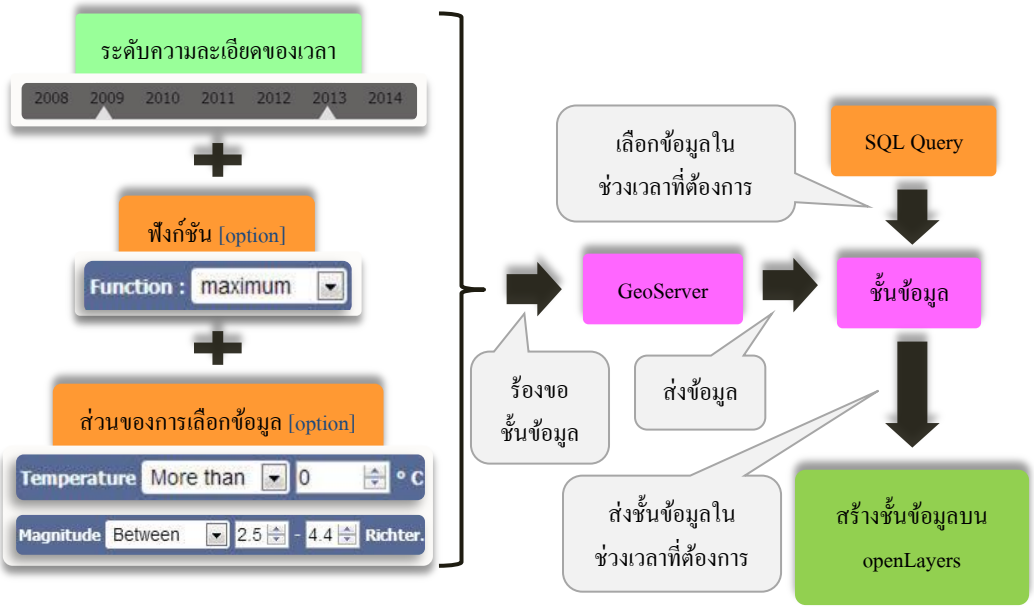


ภาพที่ 5.11 หน้าเว็บแอพพลิเคชันหลักในส่วนของแผนที่และ Timeline

5.2.3 การแสดงข้อมูล

ในการแสดงข้อมูลบนแผนที่บนเว็บแอพพลิเคชัน ข้อมูลที่นำมาแสดงจะเป็นชั้นข้อมูลที่ถูกเตรียมไว้ใน GeoServer โดยในการเลือกชั้นข้อมูลมาแสดงบนแผนที่ ระบบจะพิจารณาที่ระดับความละเอียดของเวลาที่กำลังแสดงอยู่ ณ ขณะนั้น และจะพิจารณาฟังก์ชันที่เลือกอยู่ ณ ขณะนั้น ประกอบกับพิจารณาค่าที่ผู้ใช้งานเลือกในส่วนของการเลือกข้อมูล จากนั้นระบบจะไปดึงชั้นข้อมูลใน GeoServer ที่ตรงกับระดับความละเอียดของเวลา, ฟังก์ชันที่เลือกไว้ และค่าที่ผู้ใช้งานเลือกเพื่อดูข้อมูล เมื่อได้ชั้นข้อมูลที่ต้องการแล้วจะใช้ SQL Query เลือกข้อมูลในช่วงเวลาที่ต้องการมาแสดง

บนแผนที่ จากนั้นนำชั้นข้อมูลที่มีช่วงเวลาที่ต้องการมาสร้างเป็นชั้นข้อมูลบน openLayers และนำชั้นข้อมูลมาแสดงบนแผนที่ ดังภาพที่ 5.12



ภาพที่ 5.12 การแสดงชั้นข้อมูล

ในการร้องขอข้อมูลจาก GeoServer เพื่อนำมาสร้างเป็นชั้นข้อมูลบน openLayers โดยจะสร้างชั้นข้อมูลได้หลังจากที่สร้าง base map ไว้แล้ว โดยมีโค้ดตัวอย่างดังตารางที่ 5.10

ตารางที่ 5.10 โค้ดตัวอย่างสำหรับการสร้างชั้นข้อมูลบน openLayers

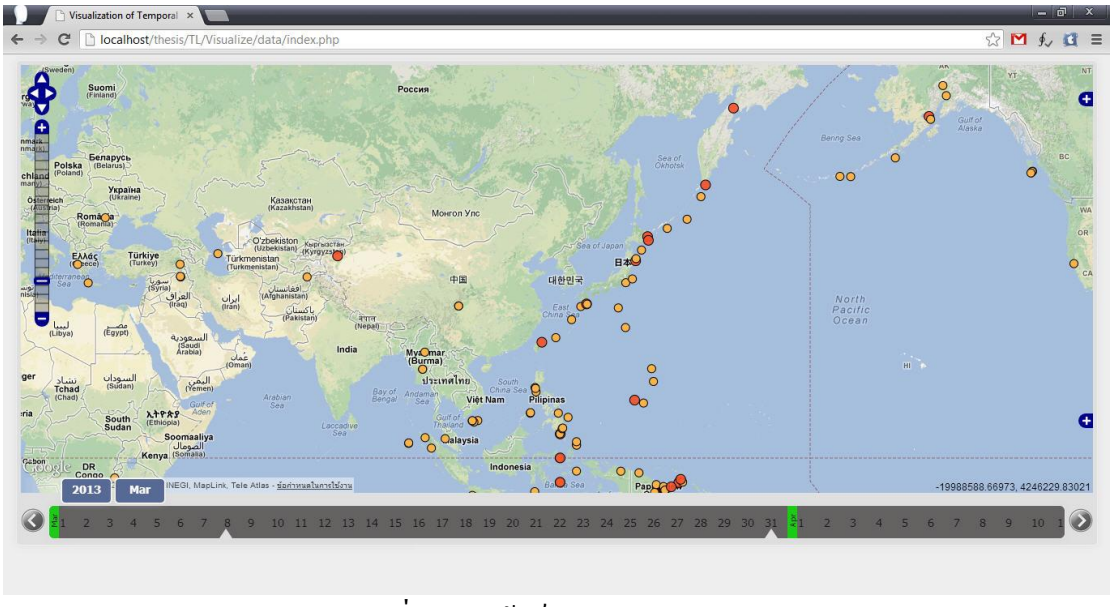
FILE: vtgdMap.js

```
1   //create basemap
2   layer = new OpenLayers.Layer.WMS(
3               "query",
4               "http://localhost:8081/geoserver/wms?viewparams=
5                           where:where eq_datetime between '2013-03-08 000000'
6                           and '2013-03-31 000000'",
7               {layers: "VTGD:view_tb_earthquake_seq4_day_1",
8                           transparent: "true", format: "image/png"},
9               {visibility: true}
10  );
11  map.addLayers([layer]);
```

- บรรทัดที่ 1 คือ source code ในส่วนของการสร้าง base map
- บรรทัดที่ 2 คือ การสร้างชั้นข้อมูลบน openLayers ใหม่ โดยจะสร้างต่อจากการสร้าง base map ซึ่งชั้นข้อมูลที่สร้างนี้เป็นแบบ WMS
- บรรทัดที่ 3 คือ ชื่อ title ของชั้นข้อมูล

- บรรทัดที่ 4 คือ url ของ wms ของเครื่องที่เป็น GeoServer ในตัวอย่างนี้คือ http://localhost:8081/geoserver/wms ตามด้วย ?viewparams = where:where eq_datetime between '2013-03-08 000000' and '2013-03-31 000000' จะเป็นการดึงชั้นข้อมูลจาก GeoServer ในช่วงวันที่ 8 มีนาคม 2556 ถึงวันที่ 31 มีนาคม 2556
- บรรทัดที่ 7 คือ ชื่อ layer ใน GeoServer ที่ต้องการดึงข้อมูล ในตัวอย่างนี้คือ VTGD:view_tb_earthquake_seq4_day_1
- บรรทัดที่ 8 คือ parameter ในการสร้างชั้นข้อมูลแบบ WMS บน openLayers
- บรรทัดที่ 9 คือ ให้ชั้นข้อมูลนี้แสดงบนแผนที่อัตโนมัติเมื่อถูกเพิ่มบนแผนที่
- บรรทัดที่ 11 คือ เพิ่มชั้นข้อมูลลงบนแผนที่

จากตัวอย่างการสร้างชั้นข้อมูลบน openLayers จะเป็นการร้องขอชั้นข้อมูลชื่อ VTGD:view_tb_earthquake_seq4_day_1 จาก http://localhost:8081/geoserver/wms โดยเลือกเฉพาะข้อมูลในช่วงวันที่ 8 มีนาคม 2556 ถึงวันที่ 31 มีนาคม 2556 และ GeoServer จะส่งชั้นข้อมูลที่ร้องขอมาในรูปแบบของรูปภาพ จากนั้นจะนำชั้นข้อมูลมาแสดงลงบนแผนที่ที่มี Timeline ในการเลือกดูข้อมูลในช่วงเวลาต่าง ๆ โดยมีหน้าเว็บแอพพลิเคชันแสดงผลดังภาพที่ 5.13



ภาพที่ 5.13 หน้าเว็บแอพพลิเคชันหลัก

เนื้อความโดยสรุปของบทนี้ ได้นำเสนอการพัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลา จากนั้นเป็นการพัฒนาระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา โดยใช้เครื่องมือ Timeline ที่ได้พัฒนาขึ้นมาในการแสดงข้อมูล ซึ่งในบทถัดไปจะเป็นการทดสอบระบบการนำเครื่องมือ Timeline ที่พัฒนาขึ้นมาใช้ในการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาจริง

บทที่ 6
การทดลองและวิเคราะห์ผลการทดลอง

หลังจากที่ได้พัฒนาเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลา และพัฒนาระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาไปในบทที่แล้ว ในบทนี้จะกล่าวถึงการทดสอบเครื่องมือ Timeline ที่ได้พัฒนาขึ้นมา โดยนำระบบที่พัฒนาร่วมกับเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลามาใช้ในการแสดงข้อมูลจริง โดยในส่วนเนื้อหานั้นจะเริ่มต้นด้วยการเตรียมข้อมูลที่จะใช้ในการทดสอบระบบ และจะอธิบายการทดสอบ จากนั้นจะอธิบายวิธีการวัดผลและประเมินผลการทดสอบระบบ ถัดไปจึงเป็นการแสดงผลการทดสอบระบบทั้งหมดรวมไปถึงการวิเคราะห์ผลการทดสอบระบบและสรุปผลการทดสอบระบบเป็นลำดับสุดท้าย

6.1 การเตรียมข้อมูลที่ใช้ในการทดสอบ

ข้อมูลที่จะใช้ในการทดสอบระบบจะประกอบด้วยข้อมูล 2 ประเภท คือ ข้อมูลประเภทที่มีตำแหน่งไม่คงที่ ซึ่งข้อมูลประเภทนี้จะมีจำนวนตำแหน่งข้อมูลที่แสดงบนแผนที่เป็นจำนวนมากมายตามจำนวนเหตุการณ์การเกิดแผ่นดินไหว ซึ่งเป็นผลให้เกิดข้อมูลล้นหน้าจอในการแสดงข้อมูลบนแผนที่ โดยจะใช้ข้อมูลแผ่นดินไหวเป็นตัวแทนของข้อมูลในกลุ่มนี้ และอีกข้อมูลประเภทคือข้อมูลประเภทที่มีตำแหน่งคงที่ ซึ่งข้อมูลประเภทนี้จะมีจำนวนตำแหน่งข้อมูลที่แสดงบนแผนที่ในแต่ละช่วงเวลาที่คงที่ แต่จำนวนข้อมูลในแต่ละตำแหน่งจะมีเป็นปริมาณมากตามช่วงเวลาที่เก็บข้อมูล ซึ่งเป็นปัญหาในการแสดงข้อมูลบนแผนที่ว่าจะแสดงข้อมูลหลายข้อมูล ณ ตำแหน่งเดียวกันอย่างไร โดยจะใช้ข้อมูลอุณหภูมิเป็นตัวแทนข้อมูลในกลุ่มนี้ ซึ่งในลำดับถัดไปจะอธิบายถึงแหล่งที่มาของข้อมูลทั้ง 2 ข้อมูล และการเตรียมพีระมิดข้อมูลของข้อมูลทั้ง 2 ข้อมูล

6.1.1 แหล่งที่มาของข้อมูล

6.1.1.1 ข้อมูลแผ่นดินไหว

ข้อมูลแผ่นดินไหวที่จะใช้ในการทดสอบระบบได้รวบรวมมาจากแหล่งข้อมูลออนไลน์ต่าง ๆ ที่ให้บริการข้อมูล ซึ่งมีทั้งหมด 3 แหล่งข้อมูลที่ได้รวบรวมมา ดังนี้

http://earthquake.usgs.gov/earthquakes/map/ เป็นเว็บไซต์ของสำนักงานธรณีวิทยาของสหรัฐอเมริกา (USGS) ที่ให้บริการข้อมูลการเกิดแผ่นดินไหวในประเทศสหรัฐอเมริกาและพื้นที่

ใกล้เคียงที่มีระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 0.1 ริกเตอร์ขึ้นไป และข้อมูลแผ่นดินไหวรอบโลกที่มีระดับความรุนแรงของแผ่นดินไหวมากกว่า 4.5 ริกเตอร์ขึ้นไป ล่าสุดย้อนหลัง 30 วัน โดยจะแสดงข้อมูลเป็นตารางอยู่บนหน้าเว็บไซต์ และสามารถดาวน์โหลดมาเป็นไฟล์ csv ได้

http://www.seismology.tmd.go.th/inside.html เป็นเว็บไซต์ของสำนักเฝ้าระวังแผ่นดินไหว กรมอุตุนิยมวิทยา ประเทศไทย ที่ให้บริการข้อมูลการเกิดแผ่นดินไหวในประเทศไทยและประเทศใกล้เคียง โดยจะแสดงข้อมูลเป็นตารางอยู่บนหน้าเว็บไซต์

http://www.tmd.go.th/earthquake_report.php เป็นเว็บไซต์ของกรมอุตุนิยมวิทยา ประเทศไทย ที่ให้บริการข้อมูลการเกิดแผ่นดินไหวในประเทศไทยและประเทศใกล้เคียง ในรอบ 15 วัน โดยจะแสดงข้อมูลเป็นตารางอยู่บนหน้าเว็บไซต์

ข้อมูลแผ่นดินไหวที่เก็บรวบรวมได้ทั้งหมดครอบคลุมช่วงเวลาตั้งแต่ปี 2473 -2556 โดยข้อมูลในช่วงปี 2473 – 2555 จะมีปริมาณน้อยโดยจะมีแต่บริเวณที่มีระดับความรุนแรงของแผ่นดินไหวสูง ๆ เนื่องจากแหล่งข้อมูลที่ให้บริการข้อมูลส่วนใหญ่จะแสดงข้อมูลย้อนหลังไม่เกิน 30 วัน และข้อมูลจะมีปริมาณมากในช่วงวันที่ 31 ธันวาคม 2555 ถึงเดือนมีนาคม 2556 เนื่องจากเป็นช่วงที่ทำการเก็บข้อมูล โดยข้อมูลในช่วงเวลานี้จะมีทุกวันในระดับชั่วโมง โดยจะเก็บข้อมูลทั้งหมดในรูปแบบของไฟล์ csv ที่ประกอบด้วยคอลัมน์ต่าง ๆ ดังนี้ วันที่ที่เกิดแผ่นดินไหว, เวลาที่เกิดแผ่นดินไหว, ศูนย์กลางการเกิดแผ่นดินไหว, ละติจูด, ลองจิจูด และระดับความรุนแรงของแผ่นดินไหว เพื่อนำเข้าฐานข้อมูลเชิงพื้นที่ PostgreSQL โดยในการนำข้อมูลเข้าฐานข้อมูลจะมีหน้าเว็บที่พัฒนาขึ้นสำหรับนำข้อมูลที่มีรูปแบบเป็นไฟล์ csv เข้าฐานข้อมูล สามารถศึกษาเพิ่มเติมได้ในภาคผนวกท้ายเล่มวิทยานิพนธ์ และมีตัวอย่างข้อมูลที่เก็บอยู่ในฐานข้อมูล ดังตารางที่ 6.1

ตารางที่ 6.1 ตารางแสดงตัวอย่างข้อมูลแผ่นดินไหวในฐานข้อมูลเชิงพื้นที่

eq_datetime	eq_focus	eq_latitude	eq_longitude	geometry	eq_magnitude
1930-05-05 20:45:58+07	MYANMAR (PEGU)	17.3	96.5	0101000020E61000000000000000205840CDCCCCCCCC4C3140	7.3
1930-12-04 01:51:51+07	MYANMAR	18.2	96.4	0101000020E61000009A99999991958403333333333333240	7.3
1933-05-16 08:12:28+07	NORTHERN SUMATRA, INDONESIA	7	96.5	0101000020E6100000000000000020584000000000000001C40	6.5
2007-11-15 17:28:00+07	MAE FA LUANG, CHIANG RAI	20.25	99.74	0101000020E61000008FC2F5285CEF5840000000000000403440	3.6
2010-04-03 18:23:27+07	WIANG PA PAO, CHIANG RAI	19.36	99.36	0101000020E6100000D7A3703D0AD758405C8FC2F5285C3340	2.2

6.1.1.2 ข้อมูลอุณหภูมิ

ข้อมูลอุณหภูมิที่จะใช้ในการทดสอบระบบได้มาจากกรมอุตุนิยมวิทยา ซึ่งข้อมูลจะอยู่ในรูปแบบของไฟล์ excel ที่ประกอบด้วยคอลัมน์ดังนี้ สถานีวัดอุณหภูมิ, วันที่วัดอุณหภูมิ และเวลาที่วัดอุณหภูมิ โดยในแต่ละสถานีวัดอุณหภูมิจะมีเวลาที่วัดอุณหภูมิทุก ๆ 3 ชั่วโมงคือ 1.00น, 4.00น, 7.00น, 10.00น, 13.00น, 16.00น, 19.00น และ 22.00น ซึ่งมีสถานีวัดอุณหภูมิทั้งหมด 119 สถานี โดยข้อมูลที่ได้มาอยู่ในช่วงตั้งแต่วันที่ 1 มกราคม 2555 ถึงวันที่ 31 ธันวาคม 2555 จากนั้นจะเก็บข้อมูลทั้งหมดในรูปแบบของไฟล์ csv ที่ประกอบด้วยคอลัมน์ต่าง ๆ ดังนี้ สถานีวัดอุณหภูมิ, วันที่วัดอุณหภูมิ, และอุณหภูมิ ณ เวลาต่าง ๆ คือ 1.00น, 4.00น, 7.00น, 10.00น, 13.00น, 16.00น, 19.00น และ 22.00น เพื่อนำเข้าฐานข้อมูลเชิงพื้นที่ PostgreSQL โดยในการนำข้อมูลเข้าฐานข้อมูลจะมีหน้าเว็บที่พัฒนาขึ้นสำหรับนำข้อมูลที่มีรูปแบบเป็นไฟล์ csv เข้าฐานข้อมูล ซึ่งมีตัวอย่างข้อมูลที่เก็บอยู่ในฐานข้อมูล ดังตารางที่ 6.2

ตารางที่ 6.2 ตารางแสดงตัวอย่างข้อมูลอุณหภูมิในฐานข้อมูลเชิงพื้นที่

tm_datetime	tm_station_en	tm_latitude	tm_longitude	geometry	tm_temperature
2012-01-01 01:00:00+07	BURI RUM	15.21666667	103.2333333	0101000020E61000005624CBEEE ECE594003910BEFEE6E2E40	21.8
2012-01-01 01:00:00+07	MAE SA RIANG	18.16666667	97.93333333	0101000020E61000007927B8BBB B7B5840B5FBB8AAAA2A3240	19.5
2012-01-01 01:00:00+07	NANG RONG	14.58333333	102.8	0101000020E61000003333333333 B3594097088EAAAA2A2D40	22
2012-01-01 01:00:00+07	LOM SAK	16.77361111	101.2494444	0101000020E6100000E70AA5E5 F64F5940B345B1600BC63040	23.5
2012-01-01 01:00:00+07	THA TUM	15.31666667	103.6833333	0101000020E610000023F197BBB BEB594036C43E2222A22E40	22
2012-01-01 01:00:00+07	CHIANG RAI	19.96138889	99.88138889	0101000020E6100000E566F2AC6 8F85840A94911951DF63340	19
2012-01-01 01:00:00+07	SU RIN_SKS	14.88333333	103.45	0101000020E6100000CDCCCCC CCCDC594030A2274444C42D40	21.1

6.1.2 การสร้างพีระมิดข้อมูล

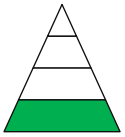
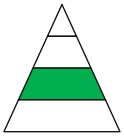
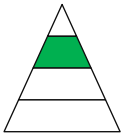
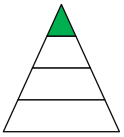
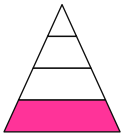
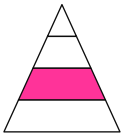
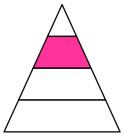
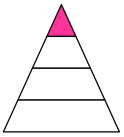
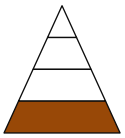
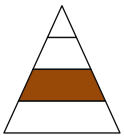
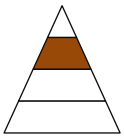
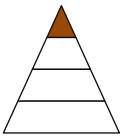
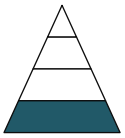
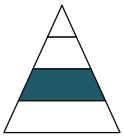
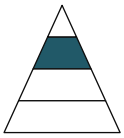
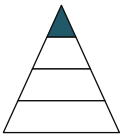
เมื่อเตรียมข้อมูลทั้ง 2 ข้อมูลไว้ในฐานข้อมูลเชิงพื้นที่แล้ว ในขั้นตอนถัดไปจะนำข้อมูลมาสร้างเป็นพีระมิดข้อมูลเชิงเวลา เพื่อนำไปแสดงผลในลำดับถัดไป โดยกระบวนการ ในการสร้างพีระมิดข้อมูลเชิงเวลาของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิมีดังนี้

ข้อมูลแผ่นดินไหว ที่เก็บมาจะมีการเกิดแผ่นดินไหวอยู่ทุกวันในระดับความรุนแรงของแผ่นดินไหวที่ต่าง ๆ กัน ซึ่งในแต่ละวันมีการเกิดแผ่นดินไหวขึ้นบ่อยครั้งจนเกือบจะเกิดขึ้นในระดับชั่วโมง ดังนั้นจึงสร้างชั้นล่างสุดของพีระมิดข้อมูลเป็นระดับชั่วโมงโดยนำข้อมูลแผ่นดินไหวทั้งหมดมาสร้างเป็นชั้นล่างสุดของพีระมิดข้อมูล จากนั้นจะทำการสร้างพีระมิดข้อมูลในชั้นบนที่มีระดับความละเอียดของเวลาเป็นวัน, เดือน และปี ตามลำดับ ซึ่งจะมีชั้นข้อมูลทั้งหมด 4 ชั้น

ข้อมูลอุณหภูมิที่ได้มาจะมีการเก็บอุณหภูมิของแต่ละสถานีวัดอุณหภูมิในทุก ๆ 3 ชั่วโมง ดังนั้นจึงสร้างชั้นล่างสุดของพีระมิดข้อมูลเชิงเวลาของข้อมูลอุณหภูมิเป็นระดับชั่วโมง โดยนำข้อมูลอุณหภูมิที่เก็บมาทั้งหมดสร้างเป็นชั้นล่างสุดของพีระมิดข้อมูล จากนั้นจะทำการสร้างพีระมิดข้อมูลในชั้นบนที่มีระดับความละเอียดของเวลาเป็นวัน, เดือน และปี ตามลำดับ ซึ่งจะมีชั้นข้อมูลทั้งหมด 4 ชั้น โดยจะใช้แนวคิดในการสร้างพีระมิดข้อมูลเชิงเวลาจากในบทก่อนหน้า

ในการสร้างพีระมิดข้อมูลจะต้องมีการเลือกใช้ฟังก์ชันในการคัดกรองข้อมูลมาสร้างชั้นพีระมิดข้อมูลในแต่ละชั้น โดยข้อมูลแผ่นดินไหว จะใช้ฟังก์ชันมากกว่าในการสร้างพีระมิดข้อมูล และสำหรับข้อมูลอุณหภูมิ จะใช้ฟังก์ชันในการสร้างพีระมิดข้อมูลทั้งหมด 3 ฟังก์ชัน คือ ฟังก์ชันหาค่าเฉลี่ย, ฟังก์ชันหาค่าต่ำสุด และฟังก์ชันหาค่ามากสุด ซึ่งการสร้างพีระมิดข้อมูลของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิจะได้ผลดังตารางที่ 6.3 จากนั้นจะนำพีระมิดข้อมูลทั้งหมดไปสร้างเป็นชั้นข้อมูลบน GeoServer โดยใช้ SQL view ซึ่งมีหน้าเว็บที่พัฒนาขึ้นสำหรับการสร้างพีระมิดเพื่อนำไปสร้างชั้นข้อมูลบน GeoServer สำหรับผู้เตรียมข้อมูล ซึ่งสามารถศึกษาเพิ่มเติมได้ในภาคผนวกท้ายเล่มวิทยานิพนธ์

ตารางที่ 6.3 ตารางแสดงการสร้างพีระมิดข้อมูลเชิงเวลาของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ

ประเภทข้อมูล	คอลัมน์ของข้อมูล	ฟังก์ชัน	ระดับความละเอียดของเวลา			
			ชั่วโมง	วัน	เดือน	ปี
ข้อมูลแผ่นดินไหว	(Mag.) ระดับความรุนแรง	Greater than	 All Mag.	 Mag. > 4	 Mag. > 5	 Mag. > 6
ข้อมูลอุณหภูมิ	(Temp.) อุณหภูมิ	Average	 All Temp.	 Avg Temp.	 Avg Temp.	 Avg Temp.
ข้อมูลอุณหภูมิ	(Temp.) อุณหภูมิ	Maximum	 All Temp.	 Max Temp.	 Max Temp.	 Max Temp.
ข้อมูลอุณหภูมิ	(Temp.) อุณหภูมิ	Minimum	 All Temp.	 Min Temp.	 Min Temp.	 Min Temp.

6.2 การทดสอบเครื่องมือ Timeline

ในการทดสอบเครื่องมือ Timeline ที่รองรับหลายระดับความละเอียดของเวลาที่พัฒนาขึ้นมาในงานวิจัยนี้ จะทดสอบโดยการสร้างเว็บแอพพลิเคชันที่ทำงานร่วมกับ Timeline ที่รองรับหลายระดับความละเอียดของเวลาสำหรับแสดงข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ โดยมีขั้นตอนในการสร้างเว็บแอพพลิเคชันสำหรับแสดงข้อมูลแผ่นดินไหวและเว็บแอพพลิเคชันสำหรับแสดงข้อมูลอุณหภูมิดังนี้

สร้างหน้าเว็บแอพพลิเคชันหลัก โดยมีส่วนในการแสดงแผนที่ และส่วนแสดง Timeline และมีการนำไฟล์ JavaScript ของ Timeline ทั้งหมดมารวมเข้ากับหน้าเว็บแอพพลิเคชันหลัก

จากนั้นทำการสร้าง Timeline โดยการเรียกใช้ฟังก์ชันใน JavaScript ซึ่งมีโค้ดตัวอย่างสำหรับหน้าเว็บแอพพลิเคชันหลักดังตารางที่ 6.4

ตารางที่ 6.4 โค้ดตัวอย่างสำหรับหน้าเว็บแอพพลิเคชันในการแสดงข้อมูลเชิงพื้นที่

FILE: *index.php*

```
<html>
    <head>
        <script type="text/javascript" src="../js/vtgdTimeline.js" charset="utf-8"></script>
        <script type="text/javascript" src="../js/vtgdTimeline-Monitor.js"
         charset="utf-8"></script>
        <script type="text/javascript" src="../js/vtgdTimeline-TimeChange-Control.js"
        charset="utf-8"></script>
        <script type="text/javascript" src="../js/vtgdTimeline-TimeInterval-Control.js"
        charset="utf-8"></script>
        <script type="text/javascript">
            $(document).ready(function(){
                createTimeline();
            });
        </script>
    </head>
    <body>
        <div id="div_map"></div>
        <div id="TL">
            <table width="100%" border="0" cellspacing="0" cellpadding="0">
            <tr>
                <td width="30px" height="30px" align="left">
                    <a onClick="" style="cursor:pointer;" id="btnLeft">
                        <img src="../../images/Left_grey.png" width="30px" height="30px"
                        onMouseOver="this.src='../../images/Left_blue.png'"
                        onMouseOut="this.src='../../images/Left_grey.png'"
                        onMouseDown="ClickLeft()"   onMouseUp="ClickLeft()" >
                    </a>
                </td>
                <td width="5px"> </td>
                <td>
                    <div id="time">
                        <div id="showtime"></div>
                    </div>
                </td>
                <td width="5px"> </td>
                <td width="30px" height="30px" align="right">
                    <a id="btnRight" onClick="" style="cursor:pointer;">
                        <img src="../../images/Right_grey.png" width="30px" height="30px"
                        onMouseOver="this.src='../../images/Right_blue.png'"
                        onMouseOut="this.src='../../images/Right_grey.png'"
                        onMouseDown="ClickRight()"   onMouseUp="ClickRight()">
                    </a>
                </td>
            </tr>
            </table>
        </div>
    </body>
</html>
```

- บรรทัดที่ 3-9 คือ เป็นการนำไฟล์ JavaScript ของ Timeline ทั้งหมดมาใส่ในหน้าเว็บหลัก
- บรรทัดที่ 10-14 คือ การเรียกใช้ฟังก์ชัน createTimeline() ซึ่งเป็นฟังก์ชันในการสร้าง Timeline ที่พัฒนาขึ้นมา
- บรรทัดที่ 17 คือ ส่วนสำหรับแสดงแผนที่
- บรรทัดที่ 18-46 คือ ส่วนสำหรับแสดง Timeline ซึ่งเหมือนกับ source code ในส่วนของการนำ Timeline ไปใช้งานที่อธิบายในหัวข้อก่อนหน้า

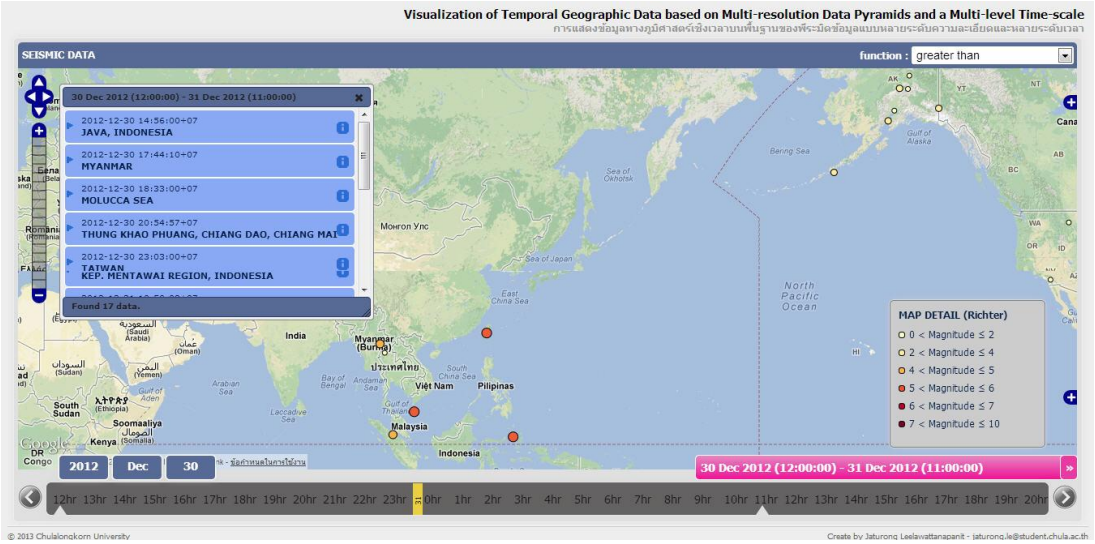
สำหรับในส่วนของการสร้างแผนที่ จะประกอบไปด้วยการสร้าง base map โดยมีโค้ดตัวอย่างการสร้าง base map บน openLayers ดังตารางที่ 6.5

ตารางที่ 6.5 โค้ดตัวอย่างสำหรับการสร้าง base map บน openLayers

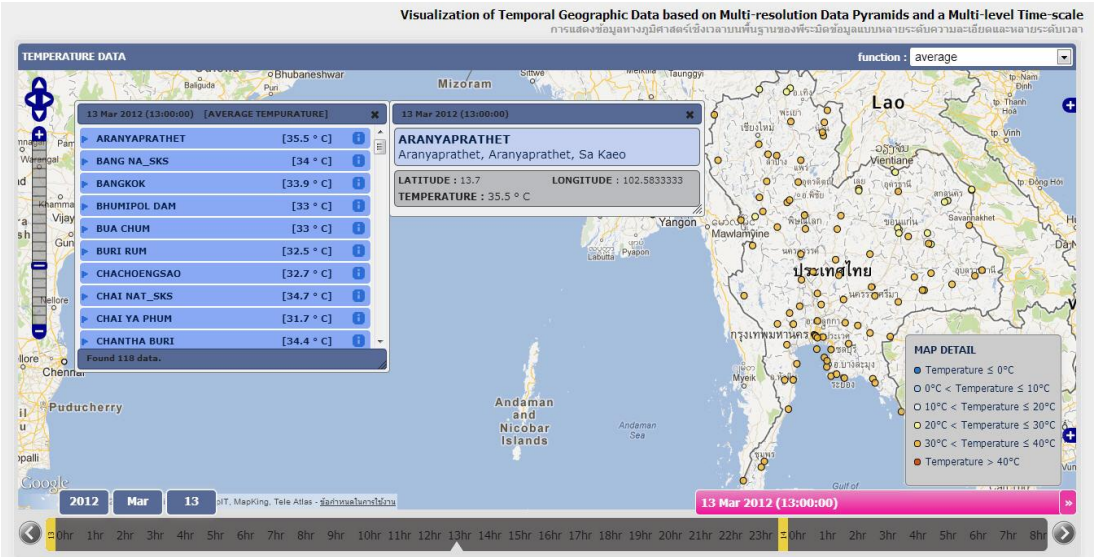
FILE: *vtgdMap.js*

```
1  map = new OpenLayers.Map('div_map', {
2              projection: 'EPSG:900913',
3              layers: [
4                          new OpenLayers.Layer.Google(
5                                 "Google Streets",
6                                 {numZoomLevels: 20},
7                                 {isBaseLayer: true}
8                          ),
9                          new OpenLayers.Layer.Google(
10                                "Google Physical",
11                                {type: google.maps.MapTypeId.TERRAIN}
12                         ),
13                         new OpenLayers.Layer.Google(
14                                "Google Hybrid",
15                                {type: google.maps.MapTypeId.HYBRID, numZoomLevels: 20}
16                         ),
17                         new OpenLayers.Layer.Google(
18                                "Google Satellite",
19                                {type: google.maps.MapTypeId.SATELLITE, numZoomLevels: 22}
20                         )
21                   ]
22 });
```

และในการสร้างชั้นข้อมูลจะทำโดยการดึงข้อมูลมาจาก GeoServer ที่ได้เตรียมไว้ในส่วนก่อนหน้าในช่วงเวลาต่าง ๆ มาสร้างเป็นชั้นข้อมูลบน openLayers แล้วแสดงชั้นข้อมูลบนแผนที่ โดยมีหน้าเว็บแอพพลิเคชันสำหรับแสดงข้อมูลแผ่นดินไหวดังภาพที่ 6.1 และหน้าเว็บแอพพลิเคชันสำหรับแสดงข้อมูลอุณหภูมิดังภาพที่ 6.2



ภาพที่ 6.1 หน้าเว็บแอพพลิเคชันสำหรับแสดงข้อมูลแผ่นดินไหว



ภาพที่ 6.2 หน้าเว็บแอพพลิเคชันสำหรับแสดงข้อมูลอุณหภูมิ

6.3 วิธีการวัดและประเมินผลการทดลอง

วิธีการวัดผลและประเมินผลการทดสอบเครื่องมือ Timeline นี้ จะทดสอบประสิทธิภาพของเครื่องมือโดยการสร้างเว็บแอพพลิเคชันสำหรับแสดงข้อมูลแผ่นดินไหว และเว็บแอพพลิเคชันสำหรับแสดงข้อมูลอุณหภูมิที่ทำงานร่วมกับ Timeline ที่รองรับหลายระดับความละเอียดของเวลา โดยจะทดสอบประสิทธิภาพการทำงานของ Timeline ในการเปลี่ยนระดับความละเอียดของเวลา และทดสอบประสิทธิภาพของระบบในการควบคุมปริมาณข้อมูลบนแผนที่ในแต่ละระดับความ

ละเอียดของเวลาสำหรับทั้งข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ ในลำดับถัดไปจะเป็นผลการทดสอบระบบ โดยใช้วิธีวัดและประเมินผลที่ได้นำเสนอไปในหัวข้อนี้

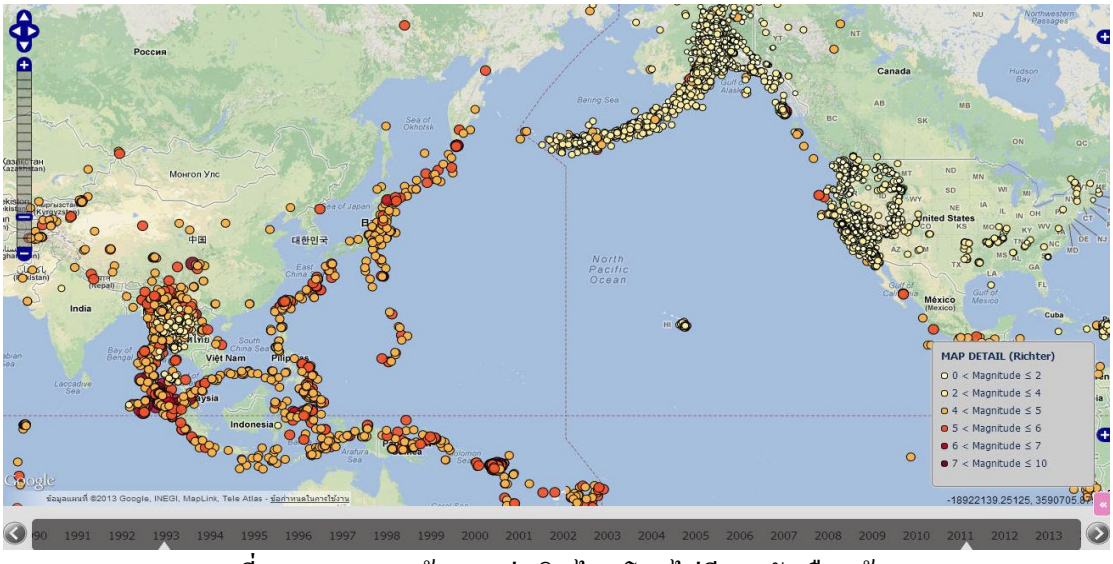
6.4 ผลการทดลองและวิเคราะห์ผลการทดลอง

ในหัวข้อนี้ จะนำเสนอผลการทดสอบระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิในระดับความละเอียดของเวลาต่าง ๆ โดยสมมติว่าข้อมูลเก็บมาในระดับชั่วโมง ดังนั้นในชั้นล่างสุดของพีระมิดข้อมูลจะเก็บข้อมูลในระดับชั่วโมง และชั้นบนของพีระมิดข้อมูลจะเก็บข้อมูลในระดับวัน, เดือน และปี ตามลำดับ จากนั้นจะวิเคราะห์ผลการทดสอบระบบ โดยผลการทดสอบระบบมีดังต่อไปนี้

6.4.1 ข้อมูลแผ่นดินไหว

ผลการแสดงข้อมูลแผ่นดินไหวโดยไม่มีการคัดกรองข้อมูล

ผลการแสดงข้อมูลแผ่นดินไหวโดยไม่มีการคัดกรองข้อมูลโดยการคัดเลือกหรือคำนวณข้อมูลมาแสดงบนแผนที่เป็นดังภาพที่ 6.3 ซึ่งจะเห็นได้ว่าข้อมูลที่แสดงบนแผนที่มีอยู่เป็นปริมาณมาก และซ้อนทับกันซึ่งจะทำให้ไม่สามารถแสดงข้อมูลให้เป็นที่เข้าใจของผู้ใช้งานได้

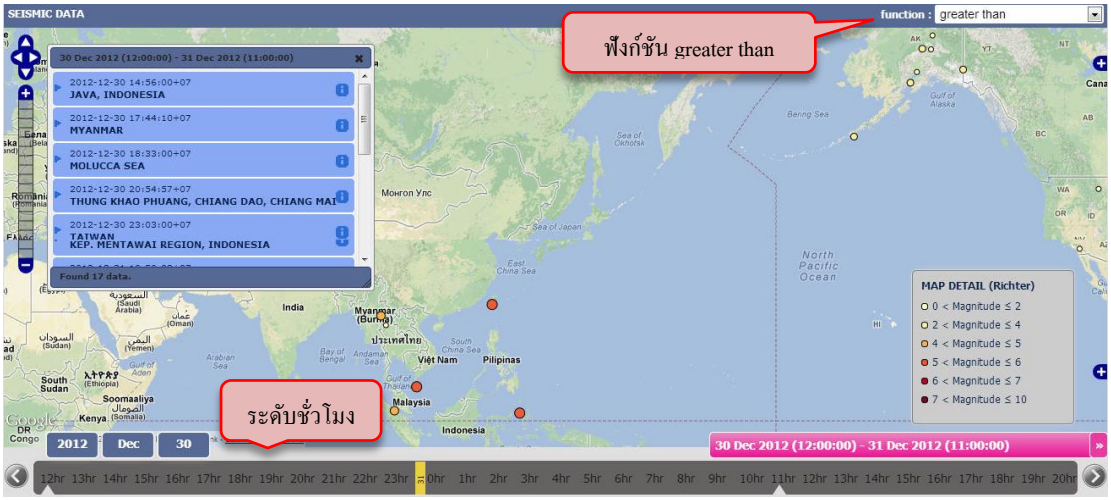


ภาพที่ 6.3 การแสดงข้อมูลแผ่นดินไหวโดยไม่มีการคัดเลือกข้อมูล

ผลการแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลเชิงเวลาตามระดับความละเอียดของเวลา

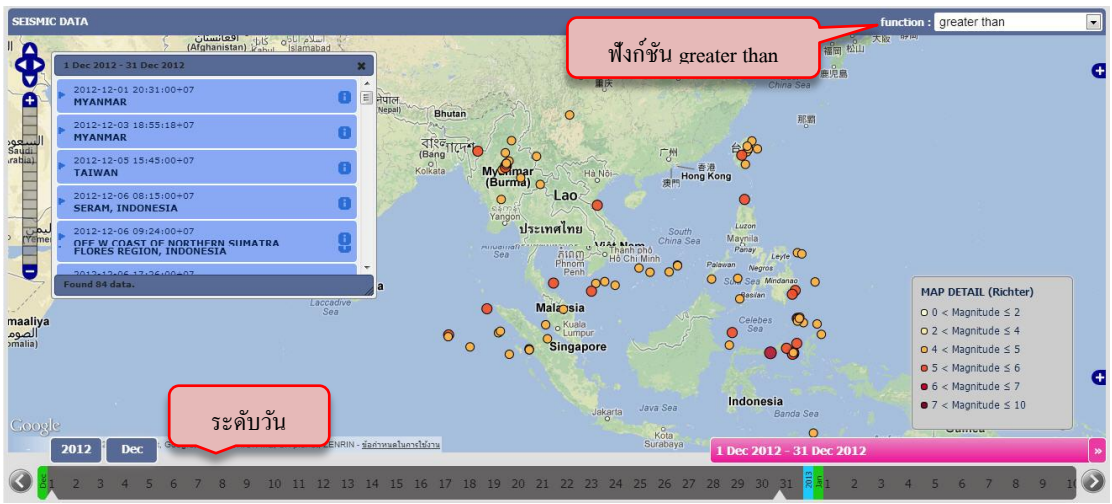
- **ฟังก์ชัน Greater than**

ความละเอียดของเวลาในระดับชั่วโมง ซึ่งเป็นชั้นของข้อมูลที่เก็บมาจากแหล่งข้อมูลในระดับชั่วโมง โดยจากภาพที่ 6.4 จะเห็นว่าปริมาณข้อมูลบนแผนที่มีอยู่อย่างกระจาย แต่มีปริมาณข้อมูลไม่มากเกินไป



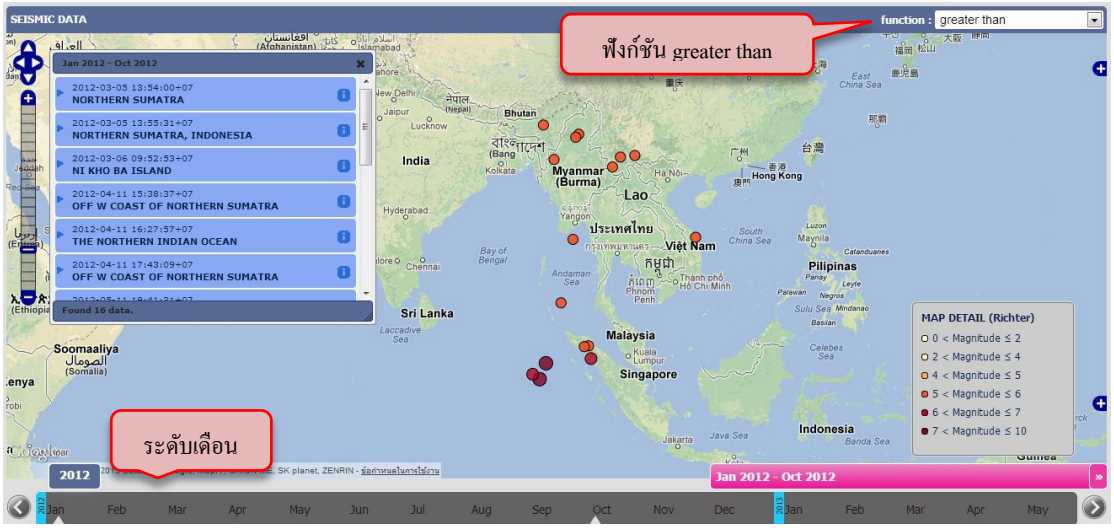
ภาพที่ 6.4 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับชั่วโมงโดยใช้ฟังก์ชัน Greater than

ความละเอียดของเวลาในระดับวัน จากภาพที่ 6.5 จะเห็นได้ว่าปริมาณข้อมูลมีมากขึ้นเมื่อเทียบกับในระดับชั่วโมง แต่มีปริมาณข้อมูลไม่มากเกินไป โดยจะแสดงข้อมูลแผ่นดินไหวที่มีความรุนแรงของแผ่นดินไหวตั้งแต่ 4 ริกเตอร์ขึ้นไป



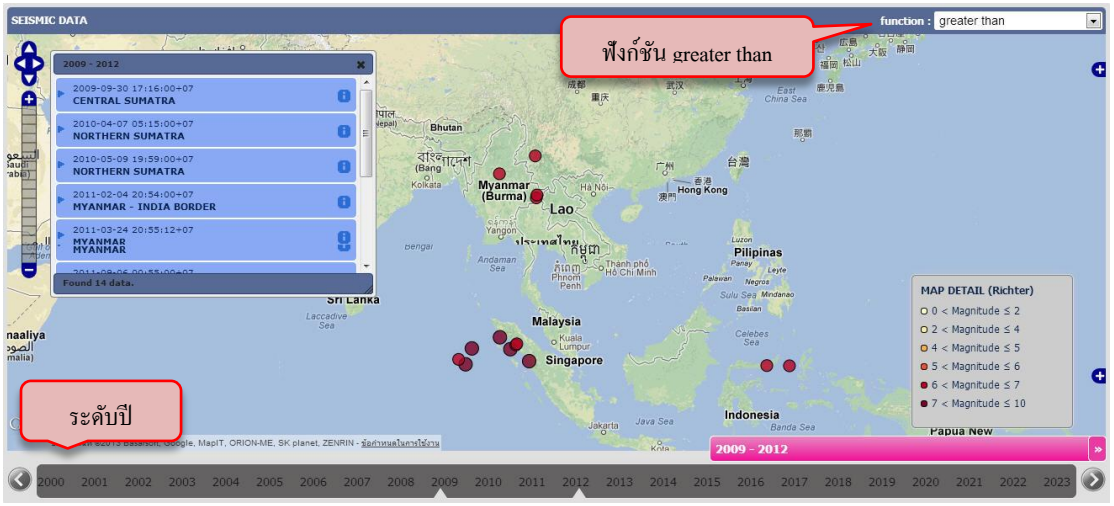
ภาพที่ 6.5 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับวันโดยใช้ฟังก์ชัน Greater than

ความละเอียดของเวลาในระดับเดือน จากภาพที่ 6.6 จะเห็นว่าปริมาณข้อมูลที่แสดงบนแผนที่ไม่มากจนเกินไป โดยจะแสดงข้อมูลแผ่นดินไหวที่มีความรุนแรงของแผ่นดินไหวตั้งแต่ 5 ริกเตอร์ขึ้นไป



ภาพที่ 6.6 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับเดือนโดยใช้ฟังก์ชัน Greater than

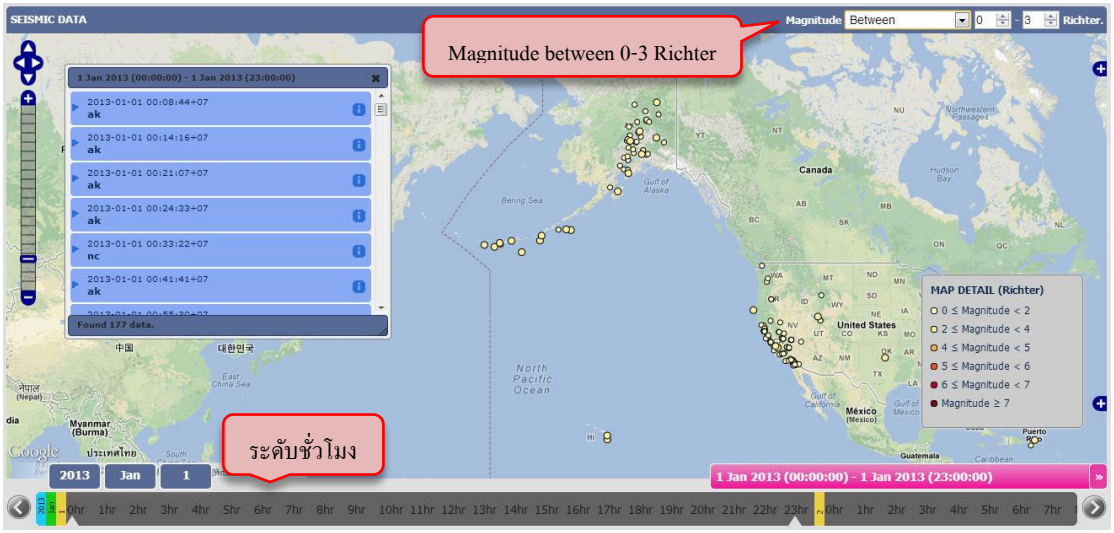
ความละเอียดของเวลาในระดับปี จากภาพที่ 6.7 จะเห็นว่าปริมาณข้อมูลที่แสดงบนแผนที่ไม่มากจนเกินไป โดยจะแสดงข้อมูลแผ่นดินไหวที่มีความรุนแรงของแผ่นดินไหวตั้งแต่ 6 ริกเตอร์ขึ้นไป



ภาพที่ 6.7 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับปีโดยใช้ฟังก์ชัน Greater than

ผลการแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลเชิงเวลาตามเกณฑ์ในการคัดเลือกข้อมูล

ความละเอียดของเวลาในระดับชั่วโมง และเลือกระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 0 -3 ริกเตอร์ ($0 \leq Mag. \leq 3$) จากภาพที่ 6.8 จะเห็นว่าปริมาณข้อมูลที่แสดงบนแผนที่ไม่มากเกินไป เนื่องจากผู้ใช้ได้เลือกระดับความรุนแรงของแผ่นดินไหวที่เหมาะสมกับความละเอียดเวลา

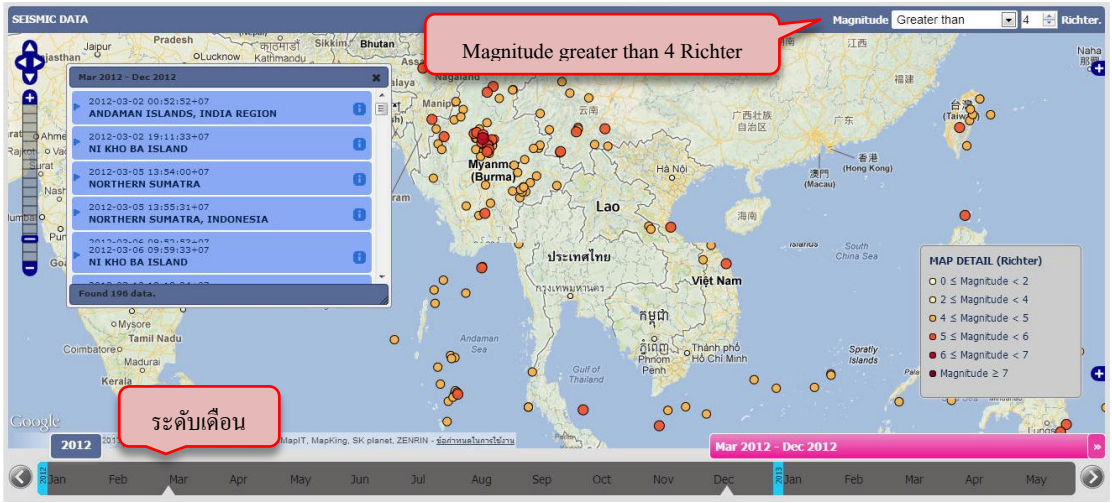


ภาพที่ 6.8 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับชั่วโมงโดยเลือกความรุนแรงตั้งแต่ 0-3 ริกเตอร์

ความละเอียดของเวลาในระดับวัน และเลือกระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 3 ริกเตอร์ขึ้นไป ($Mag. \geq 3$) จากภาพที่ 6.9 จะเห็นว่าปริมาณข้อมูลที่แสดงบนแผนที่ไม่มากเกินไปเช่นกัน เนื่องจากผู้ใช้ได้เลือกระดับความรุนแรงของแผ่นดินไหวที่เหมาะสมกับความละเอียดเวลา

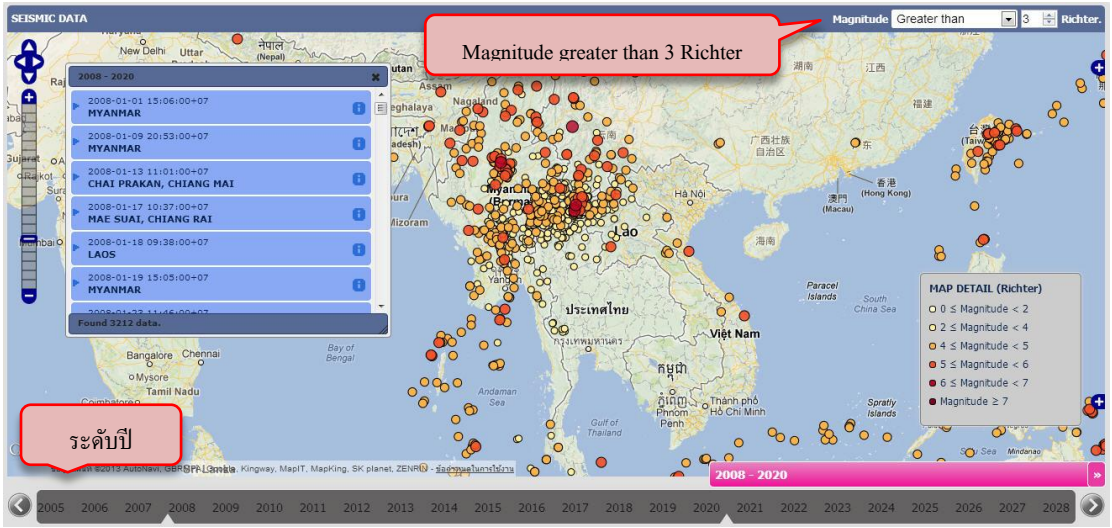
ภาพที่ 6.9 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับวันโดยเลือกความรุนแรงตั้งแต่ 3 ริกเตอร์ขึ้นไป

ความละเอียดของเวลาในระดับเดือน และเลือกระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 4 ริกเตอร์ขึ้นไป (Mag. ≥ 4) จากภาพที่ 6.10 จะเห็นว่าปริมาณข้อมูลที่แสดงบนแผนที่ยังเหมาะสมกับขนาดหน้าจอ เนื่องจากความรุนแรงของแผ่นดินไหวที่เลือกมีความเหมาะสมกับระดับเวลา



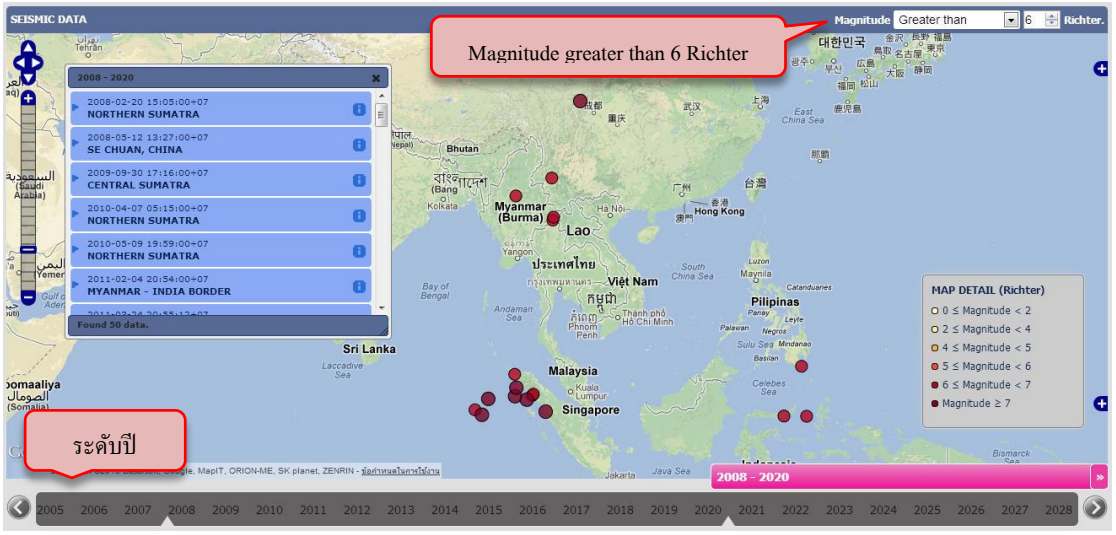
ภาพที่ 6.10 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับเดือนโดยเลือกความรุนแรงตั้งแต่ 4 ริกเตอร์ขึ้นไป

ความละเอียดของเวลาในระดับปี และเลือกระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 3 ริกเตอร์ขึ้นไป (Mag. ≥ 3) จากภาพที่ 6.11 จะเห็นว่าปริมาณข้อมูลที่แสดงบนแผนที่มีมากเกินไป เนื่องจากความรุนแรงของแผ่นดินไหวที่เลือกไม่เหมาะสมกับระดับเวลา



ภาพที่ 6.11 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับปีโดยเลือกความรุนแรงตั้งแต่ 3 ริกเตอร์ขึ้นไป

ความละเอียดของเวลาในระดับปี และเลือกระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 6 ริกเตอร์ขึ้นไป (Mag. ≥ 6) จากภาพที่ 6.12 จะเห็นว่าปริมาณข้อมูลที่แสดงบนแผนที่มีความเหมาะสมกับขนาดหน้าจอมากกว่าเลือกระดับความรุนแรงของแผ่นดินไหวตั้งแต่ 3 ริกเตอร์ขึ้นไป เนื่องจากมีการเลือกระดับความรุนแรงของแผ่นดินไหวที่มากขึ้น



ภาพที่ 6.12 การแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับปีโดยเลือกความรุนแรงตั้งแต่ 6 ริกเตอร์ขึ้นไป

สรุปผลการแสดงข้อมูลแผ่นดินไหว

จากผลการทดสอบระบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาของข้อมูลแผ่นดินไหว เมื่อเปรียบเทียบระหว่างการแสดงข้อมูลแผ่นดินไหวทั้งหมดโดยไม่มีการคัดเลือกข้อมูลตามระดับความละเอียดของเวลา และการแสดงข้อมูลแผ่นดินไหวที่มีการคัดเลือกข้อมูลตามระดับความละเอียดของเวลา จะเห็นได้ว่าถ้าไม่มีการคัดเลือกข้อมูลจะทำให้ข้อมูลแสดงบนแผนที่จะมีปริมาณมากเกินไปซึ่งไม่สามารถแสดงข้อมูลให้เป็นที่เข้าใจของผู้ใช้งานได้ และเมื่อมีการเปลี่ยนระดับความละเอียดของเวลาจากชั่วโมงเป็นวัน, จากวันเป็นเดือน และจากเดือนเป็นปี จะเห็นว่าข้อมูลมีการเปลี่ยนแปลงตามระดับความละเอียดของเวลาและมีปริมาณไม่มากเกินไป เนื่องจากมีการคัดเลือกข้อมูลมาแสดงในแต่ละระดับความละเอียดของเวลา

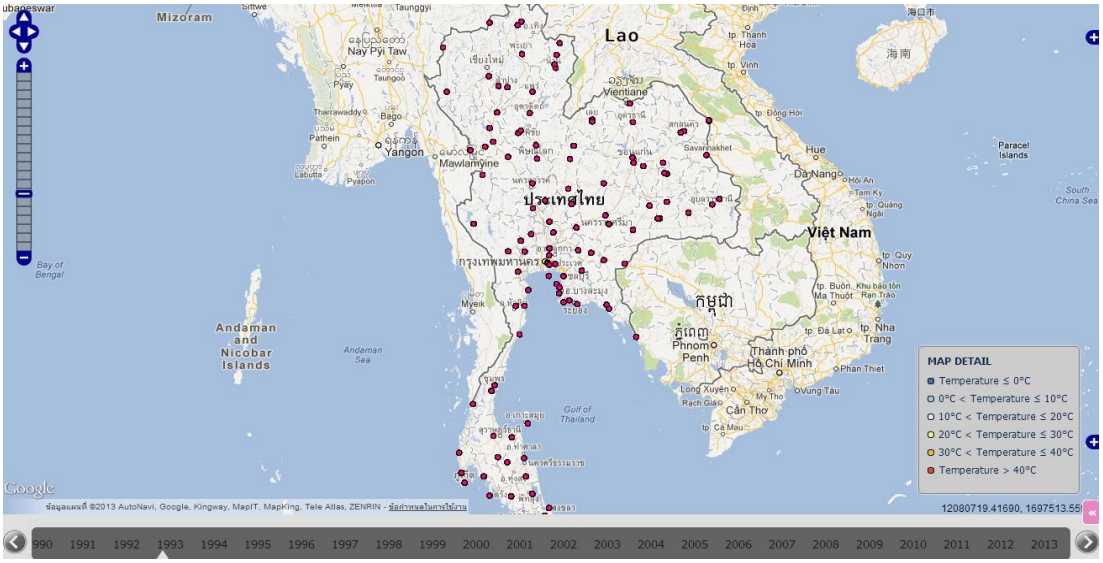
ในการแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลเชิงเวลาตามเกณฑ์ในการคัดเลือกข้อมูล ผู้ใช้งานสามารถเลือกระดับความรุนแรงของแผ่นดินไหว ดังนั้นข้อมูลที่แสดงบนแผนที่จะมีปริมาณมากหรือน้อยขึ้นอยู่กับระดับความรุนแรงของแผ่นดินไหวที่ผู้ใช้เลือก โดยในการแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลเชิงเวลาตามเกณฑ์ในการคัดเลือกข้อมูล จะมีความยืดหยุ่นในการดูข้อมูลกว่าการแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลเชิงเวลาตามระดับความละเอียดของ

เวลา เนื่องจากผู้ใช้งานสามารถเลือกระดับความรุนแรงของแผ่นดินไหวในแต่ละระดับความละเอียดของเวลาได้

6.4.2 ข้อมูลอุณหภูมิ

ผลการแสดงข้อมูลอุณหภูมิโดยไม่มีการคัดกรองข้อมูล

ผลการแสดงข้อมูลอุณหภูมิโดยไม่มีการคัดกรองข้อมูลโดยการคัดเลือกหรือคำนวณข้อมูลมาแสดงบนแผนที่เป็นดังภาพที่ 6.13 ซึ่งในแต่ละสถานีวัดอุณหภูมิจะไม่สามารถดูข้อมูลอุณหภูมิได้ เนื่องจากมีการเก็บข้อมูลอุณหภูมิเป็นราย 3 ชั่วโมง ดังนั้นในแต่ละวันแต่ละสถานีวัดอุณหภูมิจะมีข้อมูลอุณหภูมิอยู่หลายอุณหภูมิ ซึ่งในแต่ละปีแต่ละสถานีวัดอุณหภูมิจึงมีข้อมูลอุณหภูมิอยู่มากมายเช่นกัน ดังนั้นถ้าดูข้อมูลอุณหภูมิในระดับปี จะทำให้ไม่รู้ว่าจะต้องนำข้อมูลอุณหภูมิของเวลาใดมาแสดงบนแผนที่ จึงมีความจำเป็นที่จะต้องหาตัวแทนข้อมูลอุณหภูมิในแต่ละสถานีวัดอุณหภูมิ เช่น อุณหภูมิเฉลี่ยตลอดทั้งปี, อุณหภูมิสูงสุดตลอดทั้งปี หรืออุณหภูมิต่ำสุดตลอดทั้งปี เป็นต้น ซึ่งได้นำเสนอผลการแสดงข้อมูลอุณหภูมิจากแนวคิดดังกล่าวในลำดับถัดไป



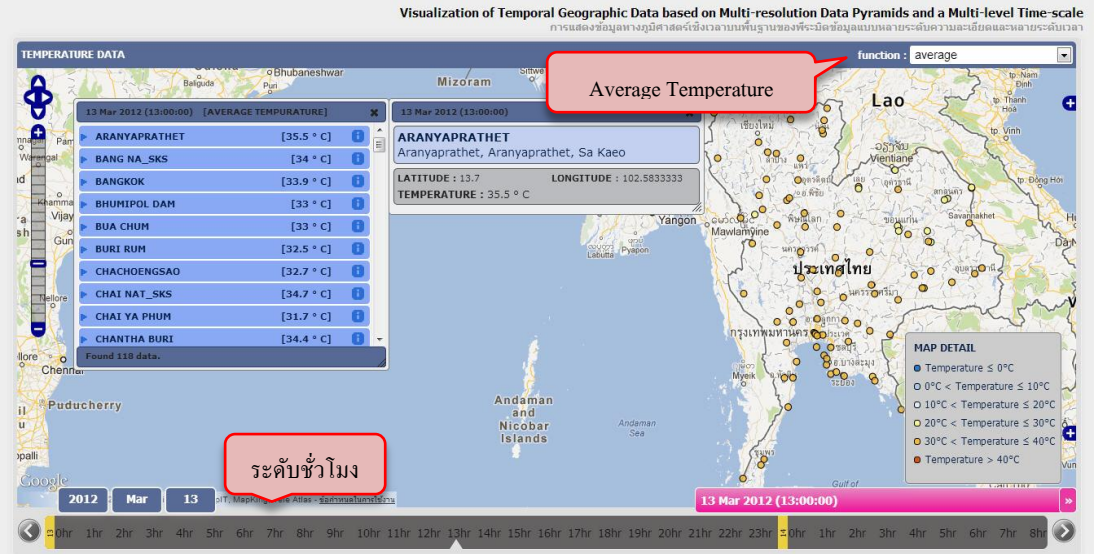
ภาพที่ 6.13 การแสดงข้อมูลอุณหภูมิโดยไม่มีการคำนวณข้อมูล

ผลการแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลเชิงเวลาตามระดับความละเอียดของเวลา

- **ฟังก์ชัน Average**

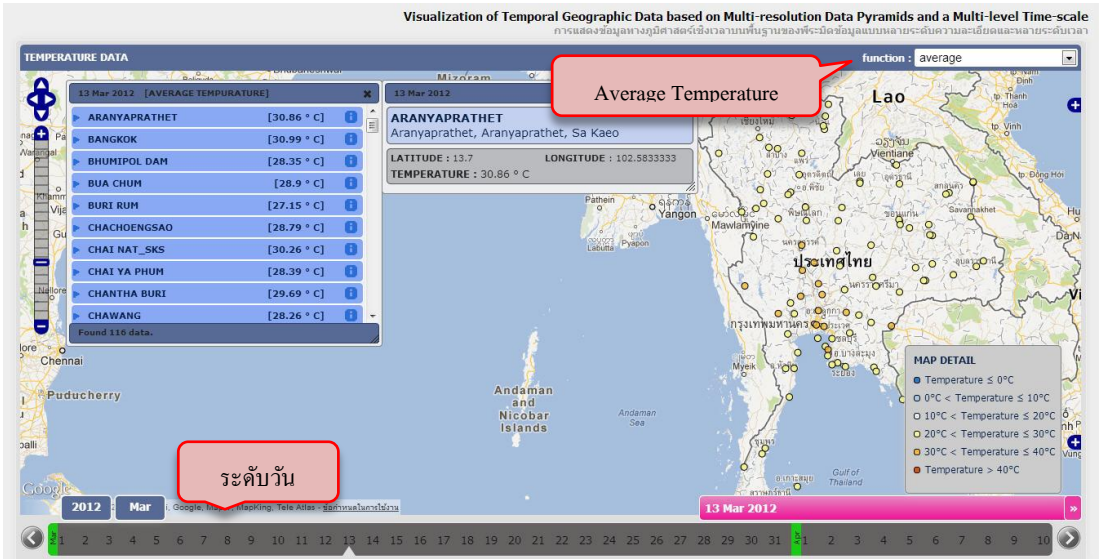
ความละเอียดของเวลาในระดับชั่วโมง จากภาพที่ 6.14 จะเห็นได้ว่าจำนวนตำแหน่งข้อมูลบนแผนที่มีเท่ากันกับการแสดงข้อมูลอุณหภูมิโดยไม่มีการคำนวณข้อมูล แต่ในระดับชั่วโมงจะ

เป็นข้อมูลที่เก็บมาในระดับชั่วโมง ดังนั้นในแต่ละตำแหน่งข้อมูลจะมีข้อมูลอยู่เพียงค่าเดียว คือ อุณหภูมิที่วัดได้ในแต่ละชั่วโมงของแต่ละสถานีวัดอุณหภูมิ



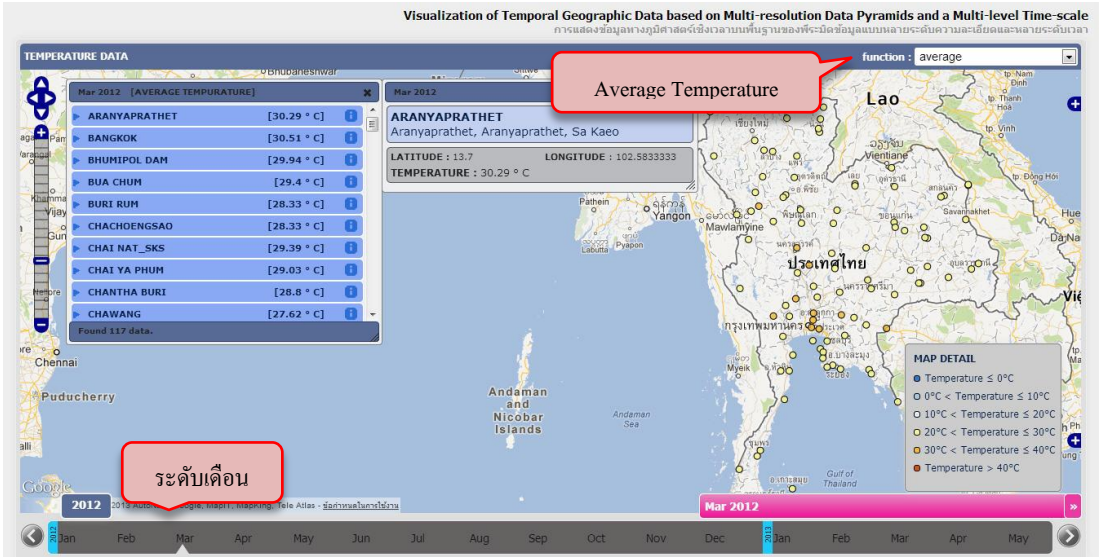
ภาพที่ 6.14 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับชั่วโมงโดยใช้ฟังก์ชัน Average

ความละเอียดของเวลาในระดับวัน จากภาพที่ 6.15 เมื่อเปรียบเทียบจำนวนตำแหน่งกับในระดับชั่วโมงจะเห็นว่ามีจำนวนตำแหน่งข้อมูลเท่าเดิม แต่เนื่องจากมีการคำนวณอุณหภูมิเฉลี่ยในแต่ละตำแหน่งข้อมูล ดังนั้นในแต่ละตำแหน่งข้อมูลจึงมีข้อมูลอุณหภูมิเพียงค่าเดียวคืออุณหภูมิเฉลี่ยในแต่ละวันของแต่ละสถานีวัดอุณหภูมิ



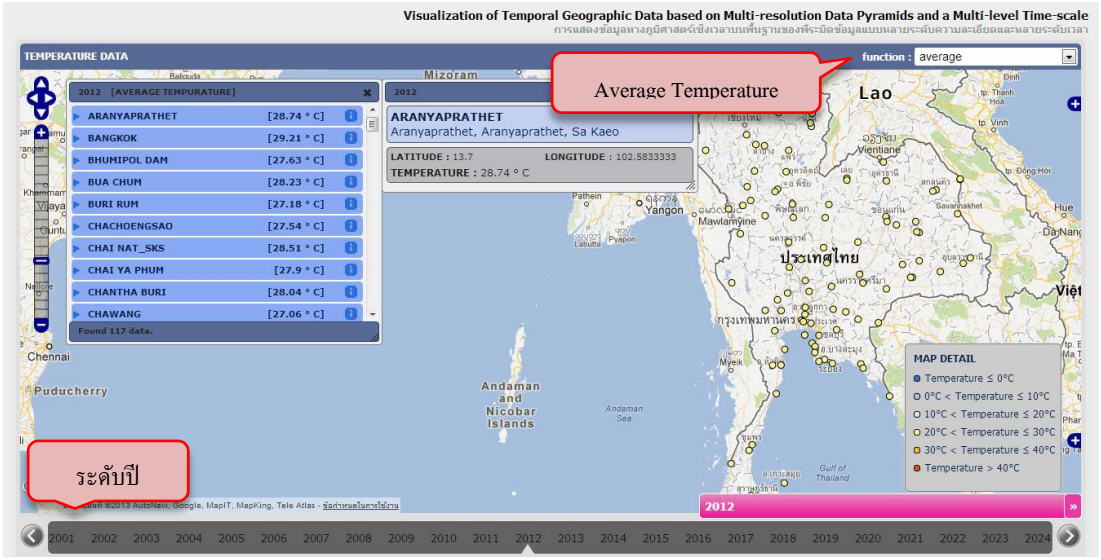
ภาพที่ 6.15 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับวันโดยใช้ฟังก์ชัน Average

ความละเอียดของเวลาในระดับเดือน จากภาพที่ 6.16 เป็นการแสดงข้อมูลอุณหภูมิเฉลี่ยในระดับเดือน ซึ่งมีจำนวนตำแหน่งข้อมูลเท่ากับในระดับวันและชั่วโมง แต่ข้อมูลในแต่ละตำแหน่งจะเป็นข้อมูลอุณหภูมิเฉลี่ยในแต่ละเดือน ดังนั้นในแต่ละตำแหน่งจึงมีข้อมูลเพียงค่าเดียว



ภาพที่ 6.16 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับเดือนโดยใช้ฟังก์ชัน Average

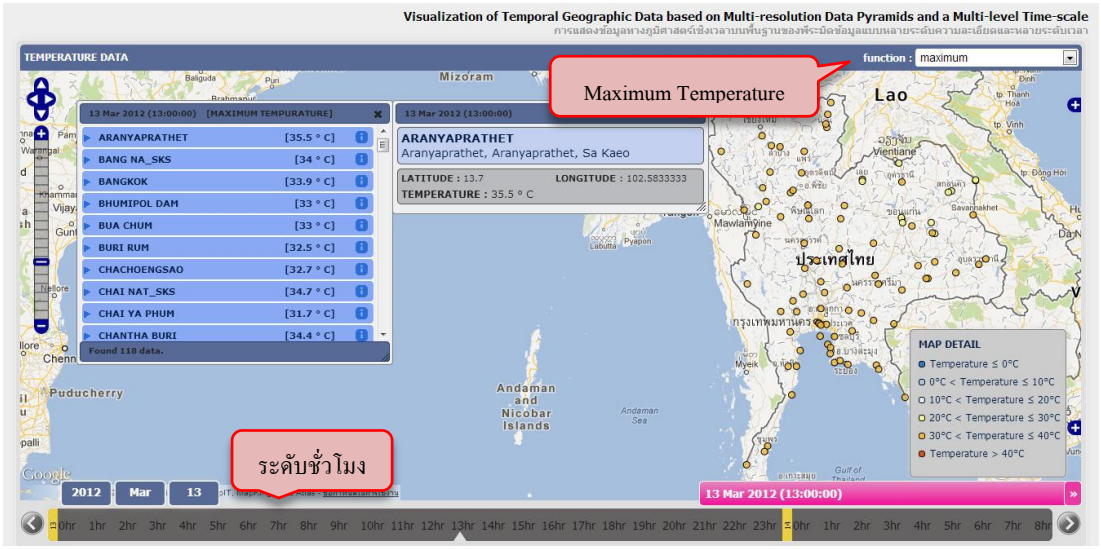
ความละเอียดของเวลาในระดับปี จากภาพที่ 6.17 เป็นการแสดงข้อมูลอุณหภูมิเฉลี่ยในระดับปี ซึ่งมีจำนวนตำแหน่งข้อมูลเท่ากับในระดับเดือน, วันและชั่วโมง แต่ข้อมูลในแต่ละตำแหน่งจะเป็นข้อมูลอุณหภูมิเฉลี่ยในแต่ละปี ดังนั้นในแต่ละตำแหน่งจึงมีข้อมูลเพียงค่าเดียว



ภาพที่ 6.17 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับปีโดยใช้ฟังก์ชัน Average

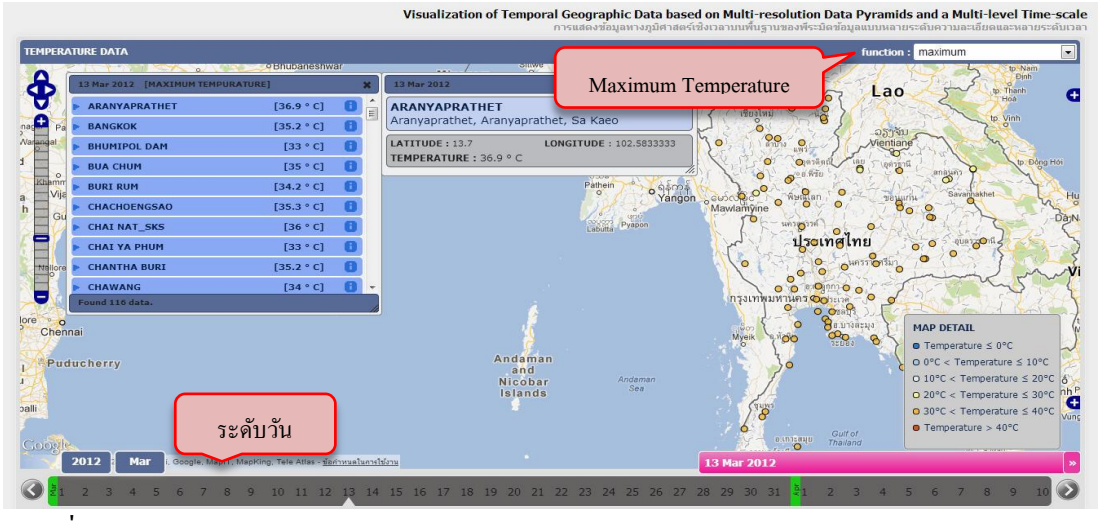
- **ฟังก์ชัน Maximum**

ความละเอียดของเวลาในระดับชั่วโมง จากภาพที่ 6.18 เป็นการแสดงข้อมูลอุณหภูมิที่เก็บมาในระดับชั่วโมง ดังนั้นในแต่ละตำแหน่งข้อมูลจะมีข้อมูลเพียงค่าเดียว ทำให้สามารถดูข้อมูลอุณหภูมิบนแผนที่ได้



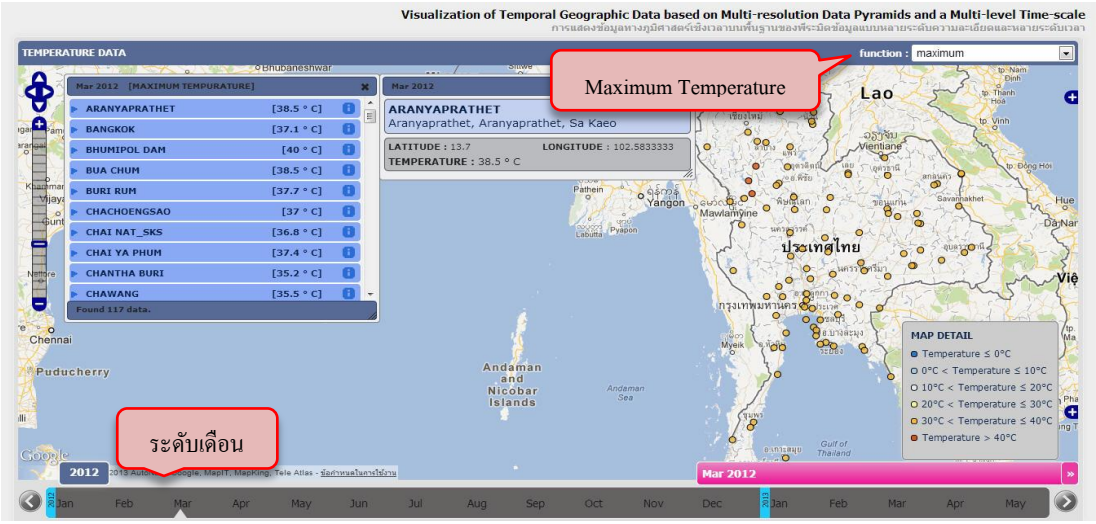
ภาพที่ 6.18 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับชั่วโมงโดยใช้ฟังก์ชัน Maximum

ความละเอียดของเวลาในระดับวัน จากภาพที่ 6.19 เป็นการแสดงข้อมูลอุณหภูมิที่มากที่สุดของทุกสถานีวัดอุณหภูมิในแต่ละวัน ดังนั้นในแต่ละตำแหน่งข้อมูลจึงมีข้อมูลอุณหภูมิเพียงค่าเดียวคืออุณหภูมิที่มากที่สุดในแต่ละวันของแต่ละสถานีวัดอุณหภูมิ



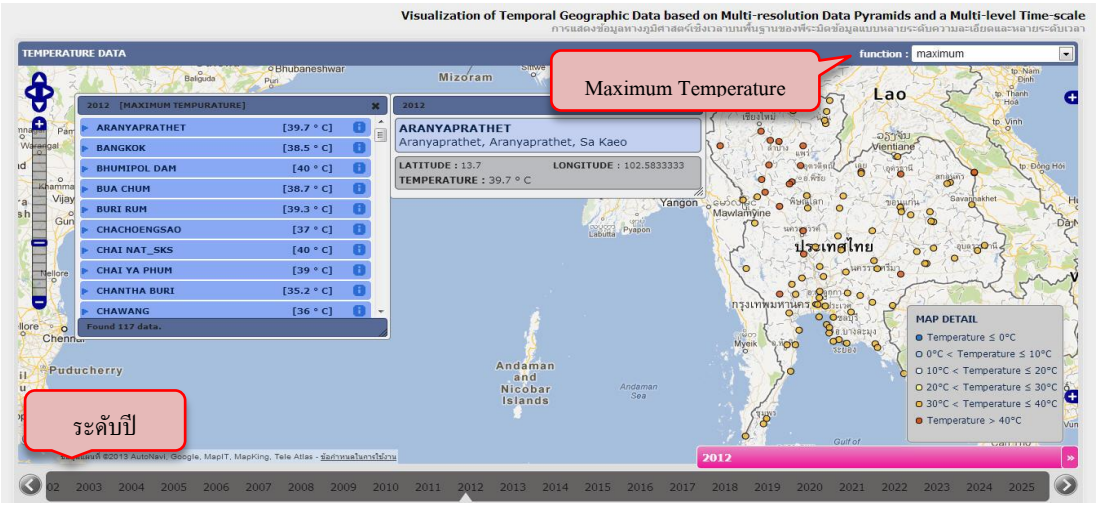
ภาพที่ 6.19 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับวันโดยใช้ฟังก์ชัน Maximum

ความละเอียดของเวลาในระดับเดือน จากภาพที่ 6.20 เป็นการแสดงข้อมูลอุณหภูมิที่มากที่สุดในระดับเดือน ซึ่งมีจำนวนตำแหน่งข้อมูลเท่ากับในระดับวันและชั่วโมง แต่ข้อมูลในแต่ละตำแหน่งจะเป็นข้อมูลอุณหภูมิที่มากที่สุดในแต่ละเดือน ดังนั้นในแต่ละตำแหน่งจึงมีข้อมูลเพียงค่าเดียว



ภาพที่ 6.20 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับเดือนโดยใช้ฟังก์ชัน Maximum

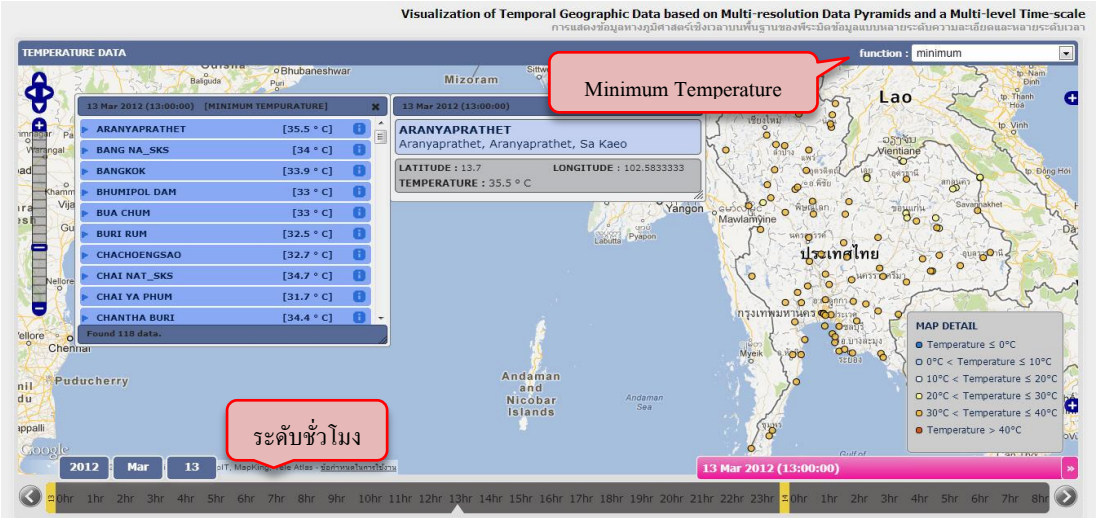
ความละเอียดของเวลาในระดับปี จากภาพที่ 6.21 เป็นการแสดงข้อมูลอุณหภูมิที่มากที่สุดในระดับปี ซึ่งมีจำนวนตำแหน่งข้อมูลเท่ากับในระดับเดือน, วันและชั่วโมง แต่ข้อมูลในแต่ละตำแหน่งจะเป็นข้อมูลอุณหภูมิที่มากที่สุดในแต่ละปี ดังนั้นในแต่ละตำแหน่งจึงมีข้อมูลเพียงค่าเดียว



ภาพที่ 6.21 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับปีโดยใช้ฟังก์ชัน Maximum

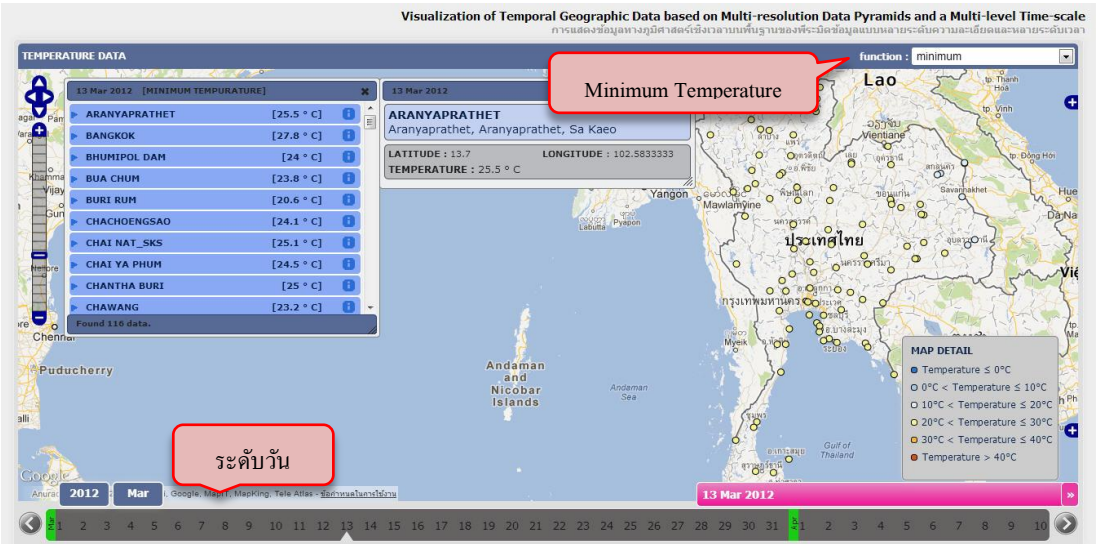
- **ฟังก์ชัน Minimum**

ความละเอียดของเวลาในระดับชั่วโมง จากภาพที่ 6.22 เป็นการแสดงข้อมูลอุณหภูมิที่เก็บมาในระดับชั่วโมง ดังนั้นในแต่ละตำแหน่งข้อมูลจะมีข้อมูลเพียงค่าเดียว ทำให้สามารถดูข้อมูลอุณหภูมิบนแผนที่ได้



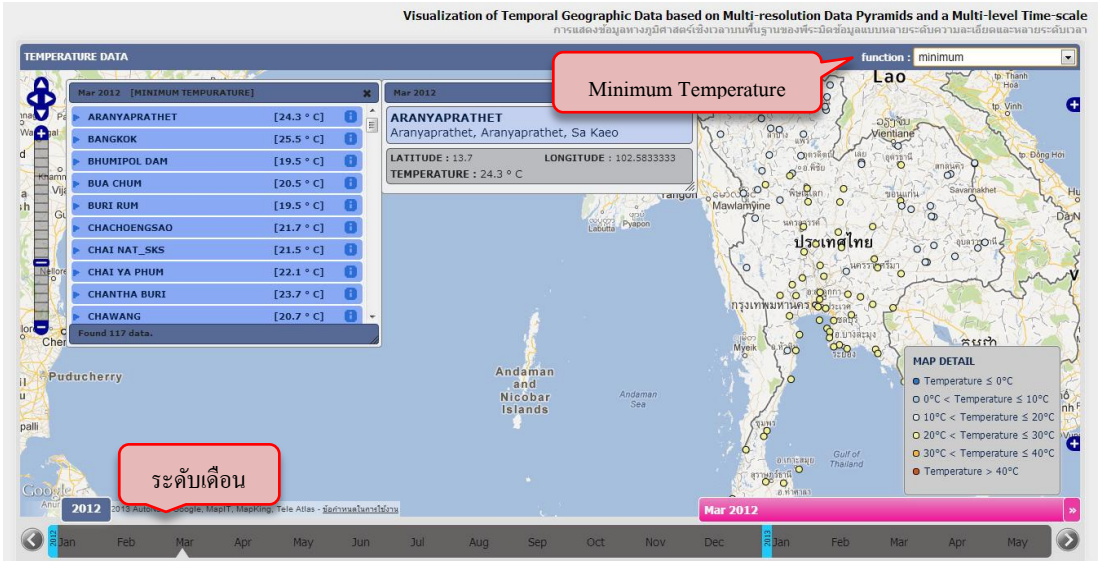
ภาพที่ 6.22 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับชั่วโมงโดยใช้ฟังก์ชัน Minimum

ความละเอียดของเวลาในระดับวัน จากภาพที่ 6.23 เป็นการแสดงข้อมูลอุณหภูมิที่น้อยที่สุดของทุกสถานีวัดอุณหภูมิในแต่ละวัน ดังนั้นในแต่ละตำแหน่งข้อมูลจึงมีข้อมูลอุณหภูมิเพียงค่าเดียวคืออุณหภูมิที่น้อยที่สุดในแต่ละวันของแต่ละสถานีวัดอุณหภูมิ



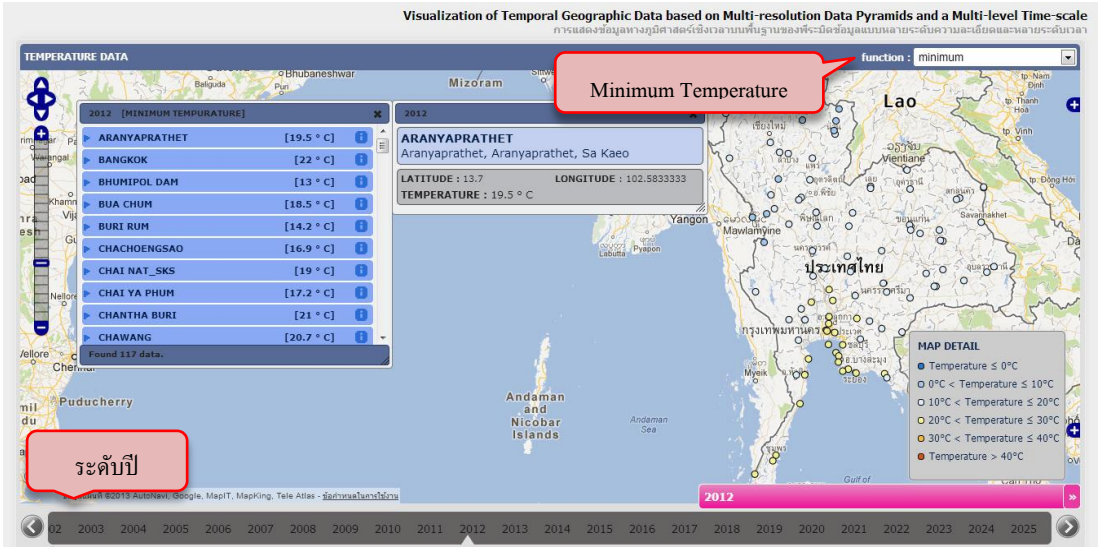
ภาพที่ 6.23 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับวันโดยใช้ฟังก์ชัน Minimum

ความละเอียดของเวลาในระดับเดือน จากภาพที่ 6.24 เป็นการแสดงข้อมูลอุณหภูมิที่น้อยที่สุดในระดับเดือน ซึ่งมีจำนวนตำแหน่งข้อมูลเท่ากับในระดับวันและชั่วโมง แต่ข้อมูลในแต่ละตำแหน่งจะเป็นอุณหภูมิที่น้อยที่สุดในแต่ละเดือน ดังนั้นในแต่ละตำแหน่งจึงมีข้อมูลเพียงค่าเดียว



ภาพที่ 6.24 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับเดือนโดยใช้ฟังก์ชัน Minimum

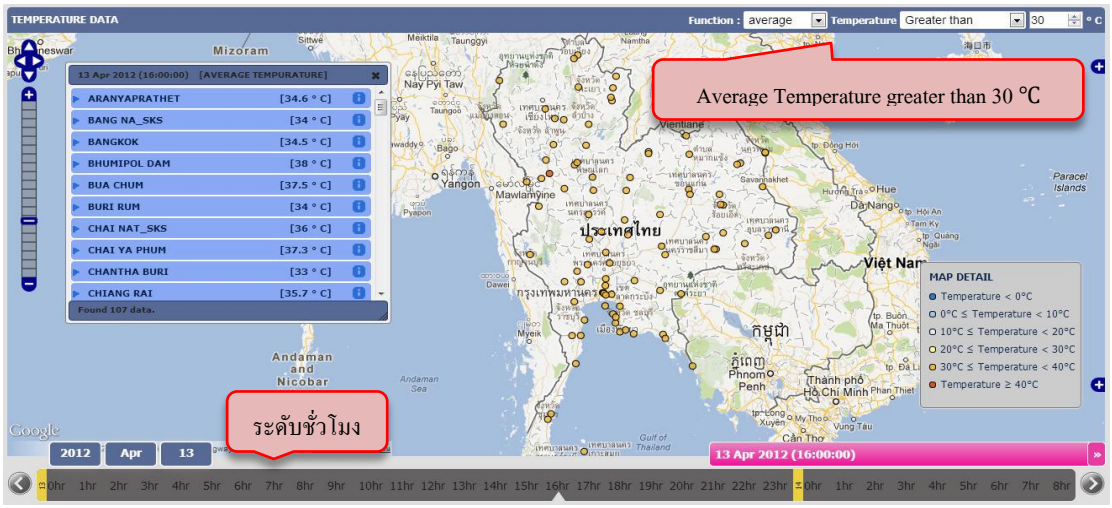
ความละเอียดของเวลาในระดับปี จากภาพที่ 6.25 เป็นการแสดงข้อมูลอุณหภูมิที่น้อยที่สุดในระดับปี ซึ่งมีจำนวนตำแหน่งข้อมูลเท่ากับในระดับเดือน, วันและชั่วโมง แต่ข้อมูลในแต่ละตำแหน่งจะเป็นอุณหภูมิที่น้อยที่สุดในแต่ละปี ดังนั้นในแต่ละตำแหน่งจึงมีข้อมูลเพียงค่าเดียว



ภาพที่ 6.25 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามระดับความละเอียดของเวลาในระดับปีโดยใช้ฟังก์ชัน Minimum

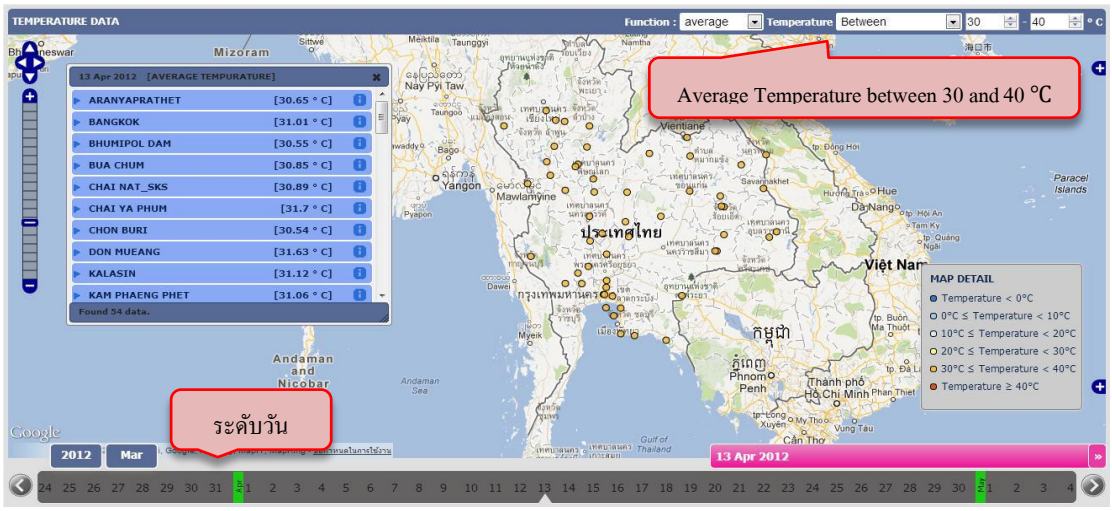
ผลการแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลเชิงเวลาตามเกณฑ์ในการคัดเลือกข้อมูล

ความละเอียดของเวลาในระดับชั่วโมง และเลือกอุณหภูมิเฉลี่ย ที่มากกว่าหรือเท่ากับ 30 ℃ (Avg(Temp.) ≥ 30℃) จากภาพที่ 6.26 เป็นการแสดงข้อมูลอุณหภูมิเฉลี่ยในระดับชั่วโมง โดยเลือกแสดงเฉพาะสถานีวัดอุณหภูมิที่มีอุณหภูมิเฉลี่ยตั้งแต่ 30℃ ขึ้นไป ซึ่งจะเห็นได้ว่ามีบางสถานีวัดอุณหภูมิที่ไม่ได้ถูกแสดงบนแผนที่เนื่องจากอุณหภูมิไม่เป็นไปตามเกณฑ์ที่เลือกไว้



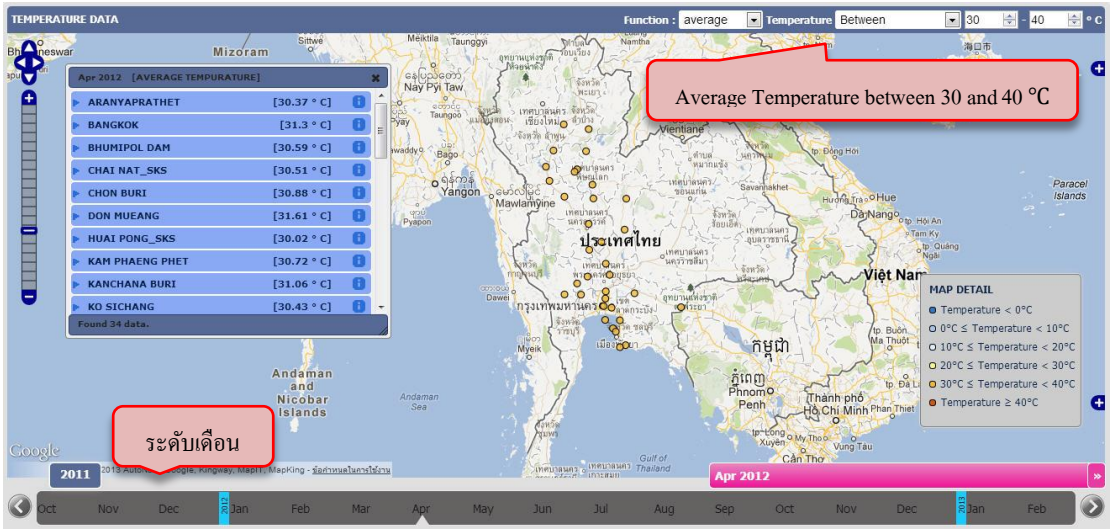
ภาพที่ 6.26 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับชั่วโมงโดยเลือกอุณหภูมิเฉลี่ยตั้งแต่ 30 องศาเซลเซียสขึ้นไป

ความละเอียดของเวลาในระดับวัน และเลือกอุณหภูมิเฉลี่ย ที่อยู่ระหว่าง 30 ℃ และ 40℃ (30℃ ≤ Avg.(Temp.) < 40℃) จากภาพที่ 6.27 เป็นการแสดงข้อมูลอุณหภูมิเฉลี่ยในระดับวัน โดยเลือกแสดงเฉพาะสถานีวัดอุณหภูมิที่มีอุณหภูมิเฉลี่ยตั้งแต่ 30℃ - 40℃



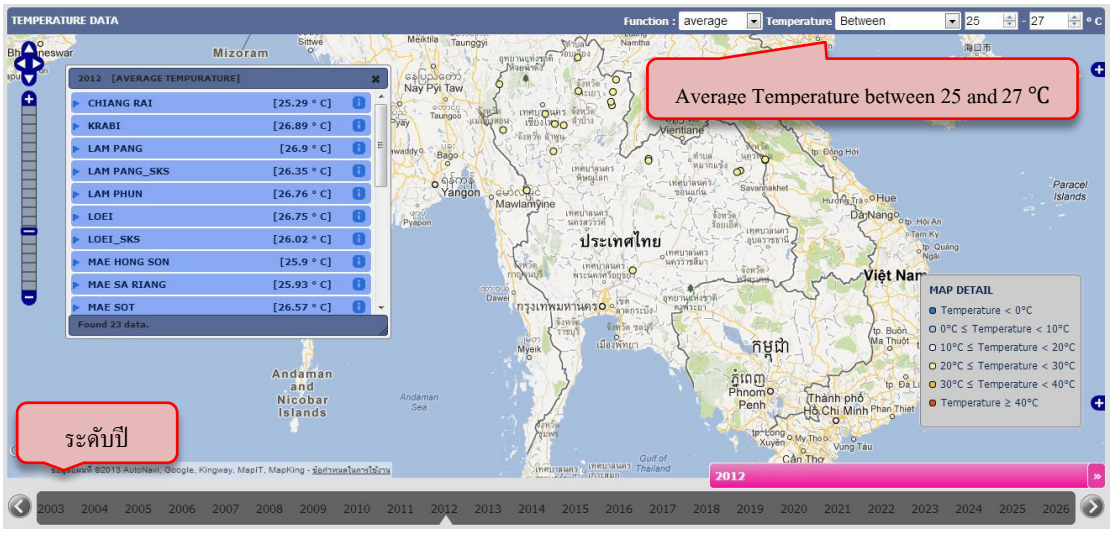
ภาพที่ 6.27 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับวัน โดยเลือกอุณหภูมิเฉลี่ยตั้งแต่ 30 - 40 องศาเซลเซียส

ความละเอียดของเวลาในระดับเดือน และเลือกอุณหภูมิเฉลี่ย ที่อยู่ระหว่าง 30 ℃ และ 40℃ (30℃ ≤ Avg.(Temp.) < 40℃) จากภาพที่ 6.28 เป็นการแสดงข้อมูลอุณหภูมิเฉลี่ยในระดับเดือน โดยเลือกแสดงเฉพาะสถานีวัดอุณหภูมิที่มีอุณหภูมิเฉลี่ยตั้งแต่ 30℃ - 40℃



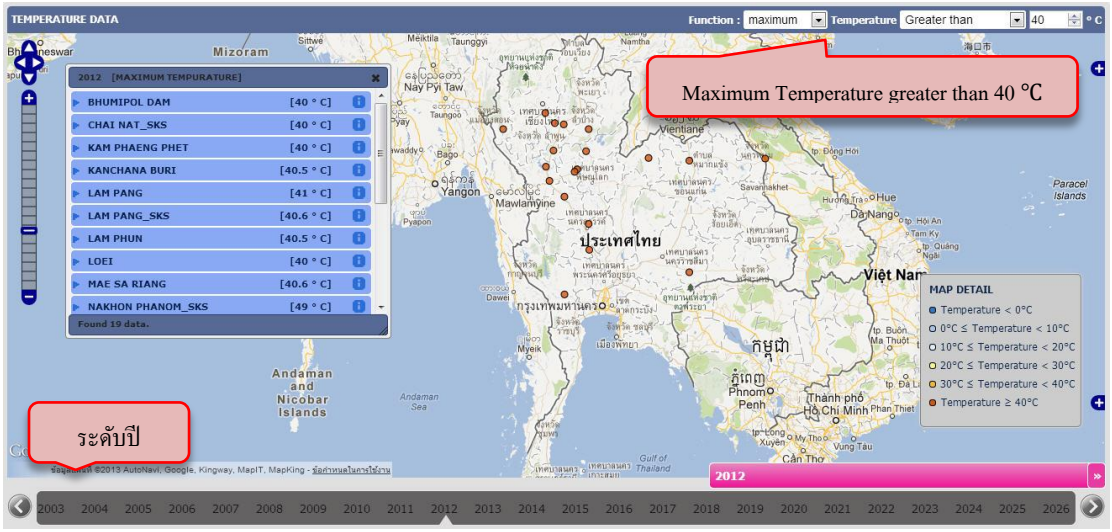
ภาพที่ 6.28 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับเดือนโดยเลือกอุณหภูมิเฉลี่ยตั้งแต่ 30 - 40 องศาเซลเซียส

ความละเอียดของเวลาในระดับปี และเลือกอุณหภูมิเฉลี่ย ที่อยู่ระหว่าง 25 ℃ และ 27℃ (25℃ ≤ Avg.(Temp.) ≤ 27℃) จากภาพที่ 6.29 เป็นการแสดงข้อมูลอุณหภูมิเฉลี่ยในระดับปี โดยเลือกแสดงเฉพาะสถานีวัดอุณหภูมิที่มีอุณหภูมิเฉลี่ยตั้งแต่ 25℃ - 27℃



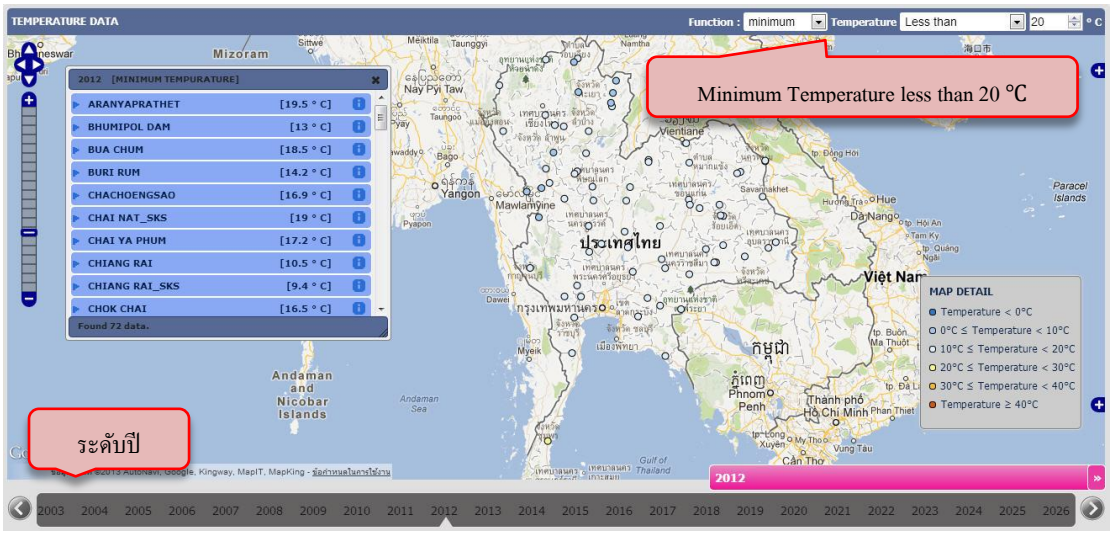
ภาพที่ 6.29 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับปี โดยเลือกอุณหภูมิเฉลี่ยตั้งแต่ 25 - 27 องศาเซลเซียส

ความละเอียดของเวลาในระดับปี และเลือกอุณหภูมิสูงที่สุด ที่มากกว่าหรือเท่ากับ 40 ℃ (Max(Temp.) ≥ 40℃) จากภาพที่ 6.30 เป็นการแสดงข้อมูลอุณหภูมิสูงสุดในระดับปี โดยเลือกแสดงเฉพาะสถานีวัดอุณหภูมิที่มีอุณหภูมิสูงสุดตั้งแต่ 40℃ ขึ้นไป



ภาพที่ 6.30 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับปี โดยเลือกอุณหภูมิสูงที่สุดตั้งแต่ 40 องศาเซลเซียสขึ้นไป

ความละเอียดของเวลาในระดับปี และเลือกอุณหภูมิต่ำที่สุด ที่น้อยกว่าหรือเท่ากับ 20 ℃ (Avg.(Temp.) ≤ 20℃) จากภาพที่ 6.31 เป็นการแสดงข้อมูลอุณหภูมิต่ำสุดในระดับปี โดยเลือกแสดงเฉพาะสถานีวัดอุณหภูมิที่มีอุณหภูมิต่ำสุดที่น้อยกว่า 20℃



ภาพที่ 6.31 การแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลตามเกณฑ์ในการคัดเลือกข้อมูลในระดับปี โดยเลือกอุณหภูมิต่ำที่สุดที่น้อยกว่า 20 องศาเซลเซียส

สรุปผลการแสดงข้อมูลอุณหภูมิ

สำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาของข้อมูลอุณหภูมิเมื่อเปรียบเทียบระหว่างการแสดงข้อมูลอุณหภูมิทั้งหมดโดยไม่มีการคำนวณข้อมูลตามระดับความละเอียดของเวลา และการแสดงข้อมูลอุณหภูมิที่มีการคำนวณข้อมูลตามระดับความละเอียด จะเห็นว่าถ้าไม่มีการคำนวณข้อมูลอุณหภูมิจะทำให้ไม่ทราบว่าแต่ละตำแหน่งแสดงข้อมูลอุณหภูมิเท่าไรอยู่ เนื่องจากในแต่ละตำแหน่งมีข้อมูลอยู่เป็นจำนวนมาก และเมื่อมีการเปลี่ยนระดับความละเอียดของเวลาจากชั่วโมงเป็นวัน, จากวันเป็นเดือน และจากเดือนเป็นปี จะเห็นว่าข้อมูลมีการเปลี่ยนแปลงตามระดับความละเอียดของเวลา ซึ่งในแต่ละฟังก์ชันจะมีการแสดงข้อมูลที่แตกต่างกันตามฟังก์ชันที่แสดง สำหรับฟังก์ชัน Average ในแต่ละตำแหน่งจะแสดงอุณหภูมิเฉลี่ยตามระดับความละเอียดของเวลาต่าง ๆ, สำหรับฟังก์ชัน Maximum ในแต่ละตำแหน่งจะแสดงอุณหภูมิที่มากที่สุดตามระดับความละเอียดของเวลาต่าง ๆ และสำหรับฟังก์ชัน Minimum ในแต่ละตำแหน่งจะแสดงอุณหภูมิที่ต่ำที่สุดตามระดับความละเอียดของเวลาต่าง ๆ ซึ่งในการคำนวณจะทำให้ในแต่ละตำแหน่งสถานีวัดอุณหภูมิมีจำนวนข้อมูลอุณหภูมิเหลือเพียง 1 ค่า จะทำให้สามารถแสดงข้อมูลอุณหภูมิบนแผนที่ได้

ในการแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลเชิงเวลาตามเกณฑ์ในการคัดเลือกข้อมูลผู้ใช้งานสามารถเลือกช่วงอุณหภูมิที่ต้องการได้ โดยจะเลือกแสดงบางสถานีวัดอุณหภูมิที่มีอุณหภูมิที่คำนวณได้อยู่ในช่วงอุณหภูมิที่ต้องการ โดยในการแสดงข้อมูลแผ่นดินไหวจากพีระมิดข้อมูลเชิงเวลาตามเกณฑ์ในการคัดเลือกข้อมูล จะมีความยืดหยุ่นในการดูข้อมูลกว่าการแสดงข้อมูลอุณหภูมิจากพีระมิดข้อมูลเชิงเวลาตามระดับความละเอียดของเวลา เนื่องจากผู้ใช้งานสามารถเลือกช่วงอุณหภูมิได้ โดยจะแสดงอุณหภูมิเฉพาะสถานีวัดอุณหภูมิที่มีอุณหภูมิอยู่ในช่วงที่ต้องการได้

จากการแสดงข้อมูลทั้ง 2 ข้อมูล จะเห็นว่า Timeline ของระบบสามารถเปลี่ยนระดับความละเอียดของเวลาได้ และในแต่ละระดับความละเอียดของเวลาจะมีปริมาณข้อมูลที่แสดงบนแผนที่ไม่มากเกินไปเนื่องจากมีการควบคุมปริมาณข้อมูลที่แสดงในแต่ละระดับความละเอียดของเวลา และในลำดับถัดไปจะเป็นการสรุปผลการทดสอบระบบต้นแบบที่ได้พัฒนาขึ้นมา

6.5 สรุปผลการทดลอง

จากการทดสอบระบบต้นแบบเพื่อวัดประสิทธิภาพในการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาทั้งในด้านการรองรับหลายระดับความละเอียดของเวลาของ Timeline และการควบคุมปริมาณข้อมูลที่แสดงบนแผนที่ในแต่ละระดับความละเอียดของเวลา ของข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ สรุปได้ว่าเครื่องมือ Timeline ที่พัฒนาขึ้นในงานวิจัยรองรับหลายระดับความละเอียดของ

เวลา โดยสามารถเปลี่ยนระดับความละเอียดของเวลาจากระดับความละเอียดของเวลาหนึ่งไปเป็นอีกระดับความละเอียดของเวลาหนึ่งได้ และในการแสดงข้อมูลแผ่นดินไหวและอุณหภูมิในแต่ละระดับความละเอียดของเวลา จะมีการควบคุมปริมาณข้อมูลที่แสดงบนแผนที่ตามระดับความละเอียดของเวลา อีกทั้งผู้ใช้งานยังสามารถเลือกดูข้อมูลโดยตั้งค่าตามเกณฑ์ที่ต้องการได้

บทที่ 7
สรุปผลการวิจัย อภิปรายผลและข้อเสนอแนะ

งานวิจัยนี้นำเสนอวิธีการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาโดยพัฒนาเครื่องมือสำหรับการเตรียมข้อมูล และพัฒนาระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาที่มี Timeline ที่รองรับการเปลี่ยนระดับความละเอียดของเวลา โดยประยุกต์ใช้เทคนิคหลายระดับความละเอียดของข้อมูลเชิงพื้นที่ และระบบที่พัฒนายังมีการคัดกรองข้อมูลที่แสดงบนแผนที่ในแต่ละระดับความละเอียดของเวลาโดยการคัดเลือกหรือคำนวณข้อมูลเพื่อหลีกเลี่ยงปัญหาข้อมูลล้นแผนที่ซึ่งจะทำให้ไม่สามารถแสดงข้อมูลให้เป็นที่เข้าใจของผู้ใช้งานได้ ซึ่งปัจจุบันนี้ยังไม่มีงานวิจัยใดทำมาก่อน ระบบต้นแบบที่พัฒนาสามารถทำงานได้ครบทุกฟังก์ชันที่ได้ออกแบบไว้ดังที่ได้นำเสนอผลการทดลองและวิเคราะห์ผลการทดลองในบทที่ผ่านมา โดยสามารถสรุปผลการวิจัยทั้งหมดได้ดังนี้

7.1 สรุปและอภิปรายผลการวิจัย

ในหัวข้อนี้จะสรุปและอภิปรายงานวิจัยทั้งหมดตั้งแต่จุดเริ่มต้นของการวิจัยตามด้วยการออกแบบและพัฒนาระบบไปจนถึงสรุปผลการทดลอง ดังนี้

งานวิจัยนี้ เน้นที่ปัญหาของการขาดการพัฒนาการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาบนแผนที่บนเว็บแอพพลิเคชัน แม้ว่าจะมีเครื่องมือ Timeline สำหรับช่วยแสดงข้อมูลในเชิงเวลาแล้ว แต่เนื่องจากข้อมูลทางภูมิศาสตร์เชิงเวลามีเป็นปริมาณมาก ดังนั้นการแสดงข้อมูลปริมาณมากนี้อาจจะทำให้ข้อมูลล้นแผนที่และไม่สามารถแสดงข้อมูลให้เป็นที่เข้าใจของผู้ใช้งานได้ จากปัญหาที่กล่าวมานี้จึงเป็นแรงจูงใจของงานวิจัยในการหาวิธีและเครื่องมือสำหรับแสดงข้อมูลทางภูมิศาสตร์ที่เกี่ยวเนื่องกับเวลาในปริมาณมาก ๆ โดยไม่มีการแสดงข้อมูลบนแผนที่ที่ยุ่งเหยิง

ในการแสดงข้อมูลเชิงพื้นที่นั้นมีเทคนิคที่ช่วยในการจัดการกับข้อมูลในปริมาณมาก ๆ โดยสร้างเป็นพีระมิดข้อมูลเพื่อช่วยลดปริมาณข้อมูลที่แสดงบนแผนที่ในแต่ละระดับความละเอียดของพื้นที่ โดยในแต่ละระดับความละเอียดของพื้นที่จะแทนด้วย 1 ชั้นของพีระมิดข้อมูล ซึ่งเทคนิคนี้เรียกว่า Multi-resolution ซึ่งในงานวิจัยนี้ได้ประยุกต์ใช้เทคนิค Multi-resolution ของข้อมูลเชิงพื้นที่มาใช้กับข้อมูลเชิงเวลาในการพัฒนา Timeline ที่รองรับหลายระดับความละเอียดของเวลา โดยให้แต่ละชั้นของพีระมิดข้อมูลแทนระดับความละเอียดของเวลาต่าง ๆ เช่น ระดับชั่วโมง, วัน, เดือน และปี เป็นต้น อีกทั้งระบบยังรองรับการคัดกรองข้อมูลที่แสดงบนแผนที่ในแต่ละระดับความละเอียดของเวลาเพื่อหลีกเลี่ยงการแสดงข้อมูลบนแผนที่ที่ยุ่งเหยิง โดยใช้การคัดเลือกหรือคำนวณข้อมูลในแต่ละระดับความละเอียดของเวลา

จากการทดสอบระบบต้นแบบเพื่อวัดประสิทธิภาพในการแสดงข้อมูลทางภูมิศาสตร์เชิงเวลาทั้งในด้านการรองรับหลายระดับความละเอียดของเวลาของ Timeline และการควบคุมปริมาณข้อมูลที่แสดงบนแผนที่ในแต่ละระดับความละเอียดของเวลา โดยทดสอบกับข้อมูลตัวอย่าง 2 ข้อมูลคือ ข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ ซึ่งมีความแตกต่างกันในด้านจำนวนตำแหน่งข้อมูลที่แสดงบนแผนที่ ซึ่งสำหรับข้อมูลแผ่นดินไหวจะเป็นผลให้เกิดความยุ่งเหยิงในการแสดงข้อมูลบนแผนที่ แต่สำหรับข้อมูลอุณหภูมิจะเป็นปัญหาในการแสดงข้อมูลบนแผนที่ว่าจะแสดงข้อมูลหลายข้อมูล ณ ตำแหน่งเดียวกันอย่างไร โดยผลการทดสอบสรุปได้ว่า เครื่องมือ Timeline ที่พัฒนาขึ้นในงานวิจัยรองรับหลายระดับความละเอียดของเวลาโดยสามารถเปลี่ยนระดับความละเอียดของเวลาจากระดับความละเอียดของเวลาหนึ่งไปเป็นอีกระดับความละเอียดของเวลาหนึ่งได้ และในการแสดงข้อมูลแผ่นดินไหวในแต่ละระดับความละเอียดของเวลา ข้อมูลที่แสดงบนแผนที่จะมีปริมาณไม่มากเกินไปเนื่องจากมีการคัดเลือกข้อมูลมาแสดงในแต่ละระดับความละเอียดของเวลา และในการแสดงข้อมูลอุณหภูมิในแต่ละระดับความละเอียดของเวลา ข้อมูลที่แสดง ณ แต่ละตำแหน่งจะเป็นข้อมูลที่ผ่านการคำนวณข้อมูลในแต่ละระดับความละเอียดของเวลา เพื่อหาตัวแทนข้อมูลมาแสดง ณ แต่ละตำแหน่ง

7.2 ข้อจำกัดและข้อเสนอแนะ

ระบบต้นแบบที่พัฒนาขึ้นมารองรับหลายระดับความละเอียดของเวลาซึ่งมีการควบคุมปริมาณข้อมูลที่แสดงในแต่ละระดับความละเอียดของเวลา แต่ไม่รวมถึงหลายระดับความละเอียดในเชิงพื้นที่ กล่าวคือ เมื่อมีการเปลี่ยนระดับความละเอียดเชิงพื้นที่ ข้อมูลที่แสดงบนแผนที่จะไม่เปลี่ยนแปลง ซึ่งการรองรับทั้งหลายระดับความละเอียดเชิงเวลาและหลายระดับความละเอียดเชิงพื้นที่ คือจะกรองข้อมูลตามระดับความละเอียดของเวลาแล้วจะกรองข้อมูลอีกตามระดับความละเอียดของพื้นที่ตามที่ได้อธิบายไปในส่วนของแนวคิดในการแสดงข้อมูลหลายระดับความละเอียดในเชิงพื้นที่และเวลา ซึ่งเป็นแนวทางในการพัฒนาต่อไปในอนาคต

ในขั้นตอนการสร้างพีระมิดข้อมูล ผู้เตรียมข้อมูลจะต้องสร้างพีระมิดข้อมูลโดยใช้เครื่องมือที่ได้พัฒนาขึ้นมาสำหรับสร้างพีระมิดข้อมูล โดยระบบจะสร้าง view ของแต่ละชั้นพีระมิดข้อมูลเก็บไว้ในฐานข้อมูล PostgreSQL จากนั้นผู้เตรียมข้อมูลต้องนำ view ทั้งหมดสร้างชั้นข้อมูลบน GeoServer ผ่านอินเตอร์เฟสของ GeoServer ทีละชั้นข้อมูลซึ่งได้อธิบายวิธีการเพิ่มชั้นข้อมูลใน GeoServer ในภาคผนวกท้ายเล่มวิทยานิพนธ์ ซึ่งถ้าหากลดขั้นตอนที่ให้ผู้เตรียมข้อมูลนำ view มาสร้างเป็นชั้นข้อมูลบน GeoServer โดยการพัฒนาให้ระบบสามารถสร้างชั้นข้อมูลบน GeoServer ได้อัตโนมัติโดยไม่ต้องให้ผู้เตรียมข้อมูลเพิ่มชั้นข้อมูลเอง จะทำให้ผู้เตรียมข้อมูลไม่จำเป็นต้องมี

ความรู้เบื้องต้นเกี่ยวกับการสร้างชั้นข้อมูลบน GeoServer และสามารถเตรียมข้อมูลได้สะดวกและรวดเร็วขึ้น

รายการอ้างอิง

[1] André, P., Wilson, M. L., and schraefel, m.c. Continuum: Designing timelines for hierarchies, relationships and scale. User Interface Software and Technology, pp. 101-110. Rhode Island, 2007.

[2] Christopherson, C. E., and Thomsen, R. W. Encounter Geosystems: Interactive Explorations of Earth Using Google Earth. 10th ed. New Jersey: Prentice Hall, 2009.

[3] Google. Basics of the time slider - Google Earth Help[Online]. 2012. Available from: http://support.google.com/earth/bin/answer.py?hl=en&answer=183758[2012, October 29]

[4] Google. Time and Animation - Keyhole Markup Language - Google Developers[Online]. 2012. Available from: https://developers.google.com/kml/documentation/time [2012,October 29]

[5] Histros Inc. myHistro[Online]. (n.d.). Available from: http://www.myhistro.com/[2008, January 8]

[6] Johnson, I., and Wilson, A. The timemap project: Developing time-based GIS display for cultural data. Journal of GIS in Archaeology 1 (April 2003) : 123-135.

[7] Lendl, C., and Purgathofer, P. Challenges in Interactive Time-Based Information Visualization. 5th International Workshop on Digital Approaches in Cartographic Heritage, Vienna, Austria, 2010.

[8] MIT. Jewish History[Online]. 2006. Available from: http://www.simile-widgets.org/timeline/examples/religions/jewish-history.html[2012,October 29]

[9] MIT. SIMILE Timeline[Online]. 2006. Available from: http://www.simile-widgets.org/timeline[2012,October 29]

[10] MIT. The Life of Monet[Online]. 2006. Available from: http://simile.mit.edu/timeline/examples/monet/monet.html[2012,October 29]

[11] Mitchell, T. Web Mapping Illustrated. 1st ed. Sebastopol: O' Reilly Media, 2005.

[12] Mnemograph LLC. Timeglider[Online]. (n.d.). Available from: http://timeglider.com/ [2012,October 29]

[13] Open Geospatial Consortium, Inc. OGC Standards and Supporting Documents[Online]. 1994. Available from: http://www.opengeospatial.org/standards/[2012,October 29]

[14] Open Source Geospatial Foundation. OpenLayers[Online]. (n.d.). Available from: http://openlayers.org[2012,October 29]

[15] Pearson., D. Astronomy Timeline[Online]. 2006. Available from: http://www.astronomer.me.uk/logs/by-timeline/[2012,October 29]

[16] ProPublica. Timeline Setter[Online]. (n.d.). Available from: http://propublica.github.com/timeline-setter/[2012,October 29]

[17] Pumphrey., M. GeoServer[Online]. 2010. Available from: http://www.geoserver.org [2012,October 29]

[18] The PostgreSQL Global Development Group. PostgreSQL[Online]. 1996. Available from: http://www.postgresql.org/[2012,October 29]

[19] TimeMap Open Source Consortium. timemap[Online]. (n.d.). Available from: https://code.google.com/p/timemap/[2012,October 29]

[20] USGS. Earthquake Facts and Statistics[Online]. 2012. Available from: http://earthquake.usgs.gov/earthquakes/eqarchives/year/eqstats.php[2012, November 20]

[21] Wernecke, J. The KML Handbook. 1st ed. Addison-Wesley, 2009.

ภาคผนวก

ภาคผนวก ก

การติดตั้ง GeoServer บน Windows

การติดตั้ง GeoServer บน Windows สามารถดาวน์โหลดตัวติดตั้งซอฟต์แวร์ และติดตั้งซอฟต์แวร์ผ่าน Windows GUI ได้โดยไม่ต้องติดตั้งผ่าน Command line ให้ยุ่งยาก ซึ่งมีขั้นตอนการติดตั้งดังนี้

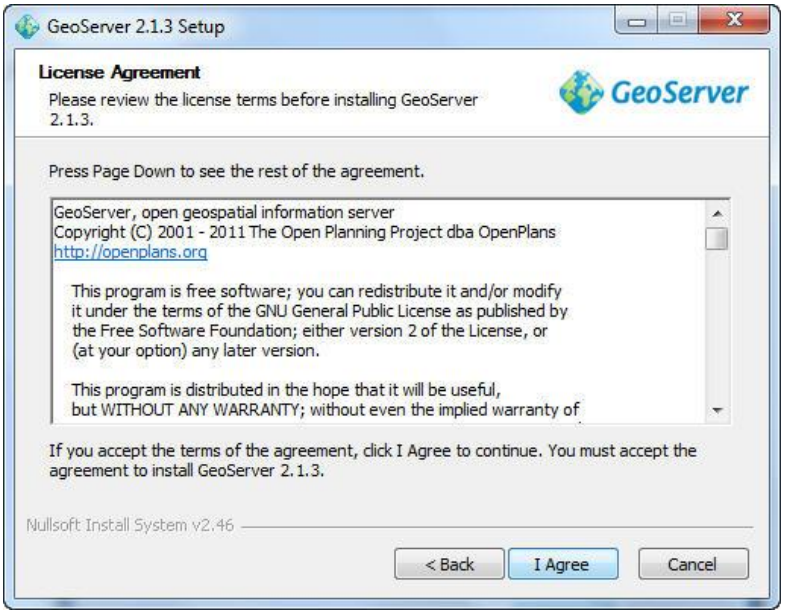
1. ดาวน์โหลดตัวติดตั้งได้ที่ http://geoserver.org/display/GEOS/Download
2. เลือกเวอร์ชันที่ต้องการดาวน์โหลด โดยเลือกที่ Stable โดยเวอร์ชันที่จะติดตั้งคือ GeoServer 2.1.3 ซึ่งปัจจุบันมีเวอร์ชันที่ใหม่กว่า
3. เลือก Windows Installer เพื่อดาวน์โหลดซอฟต์แวร์มาไว้ที่เครื่อง จากนั้นทำการติดตั้งซอฟต์แวร์ ดังภาพที่ ก.1

ภาพที่ ก.1 การดาวน์โหลดซอฟต์แวร์

4. หน้า Welcome to the GeoServer 2.1.3 Setup Wizard ให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป ดังภาพที่ ก.2

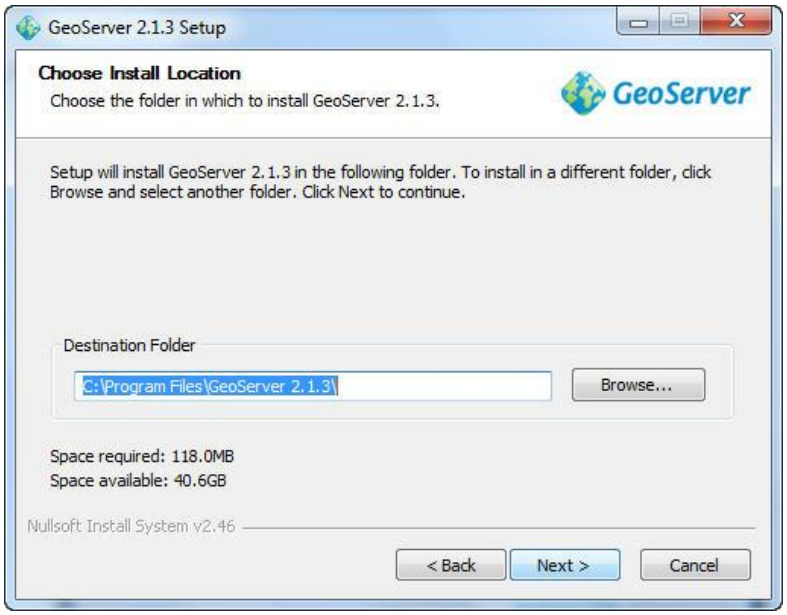
ภาพที่ ก.2 ขั้นตอนการติดตั้งซอฟต์แวร์ GeoServer

5. หน้า License Agreement ให้กดปุ่ม I Agree เพื่อยอมรับเงื่อนไขของซอฟต์แวร์ ดังภาพที่ ก.3



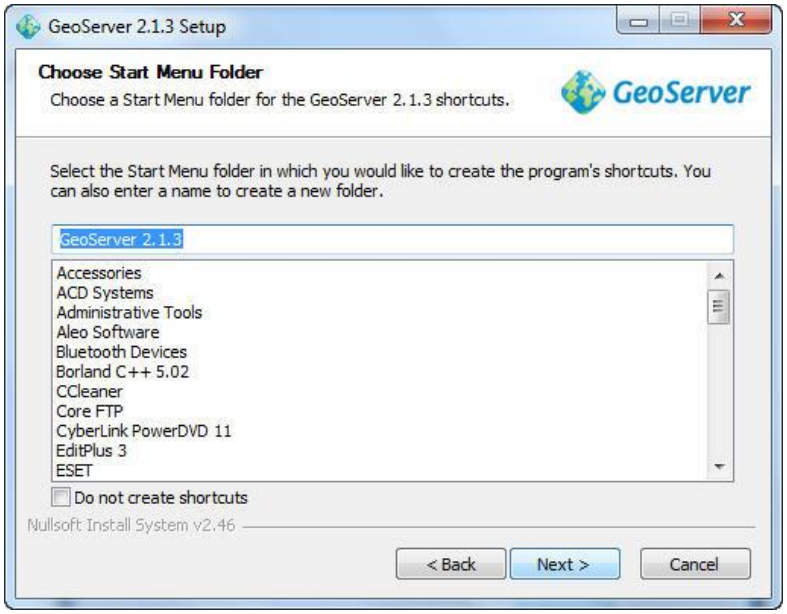
ภาพที่ ก.3 แสดงรายละเอียดเงื่อนไขของซอฟต์แวร์

6. หน้า Choose Install Location ให้เลือกปลายทางที่ต้องการติดตั้งโปรแกรม โดยค่าเริ่มต้นปลายทางที่ติดตั้งจะเป็น C:\Program Files\GeoServer 2.1.3\ หากต้องการเปลี่ยนปลายทางที่ติดตั้ง ให้กด Browse แล้วเลือกปลายทางที่ต้องการ เมื่อเลือกปลายทางเสร็จสิ้น ให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป ดังภาพที่ ก.4



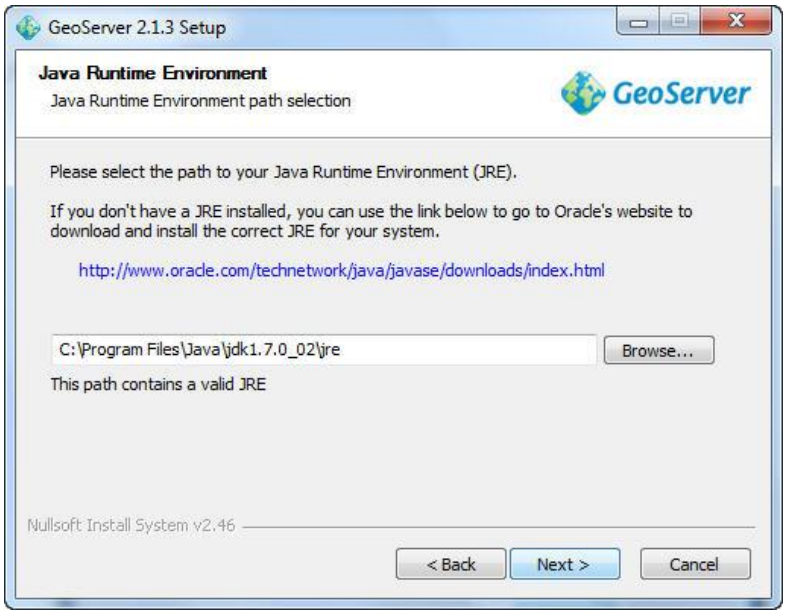
ภาพที่ ก.4 เลือกปลายทางการติดตั้งซอฟต์แวร์ GeoServer

7. หน้า Choose Start Menu Folder ให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป ดังภาพที่ ก.5



ภาพที่ ก.5 แสดงชื่อโฟลเดอร์ของซอฟต์แวร์ที่ติดตั้ง

8. หน้า Java Runtime Environment ให้เลือก path ที่ติดตั้ง Java Runtime Environment จากนั้นให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป (ถ้ายังไม่ได้ติดตั้ง JRE และ JDK ให้ไปติดตั้ง JRE และ JDK ก่อน แล้วค่อยมาติดตั้ง GeoServer) ดังภาพที่ ก.6

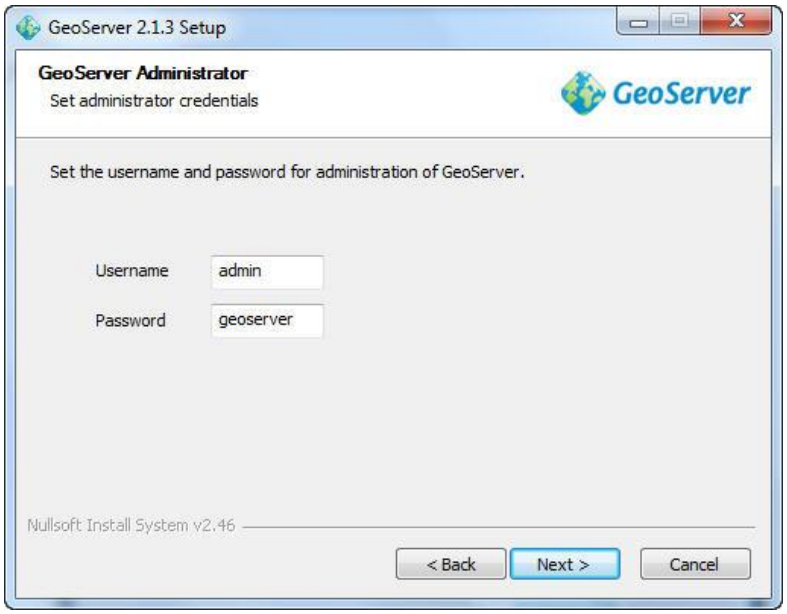


ภาพที่ ก.6 เลือก path ที่ติดตั้ง Java Runtime Environment

9. หน้า GeoServer Data Directory ให้เลือก Default Data Directory สำหรับกรณีที่ติดตั้งครั้งแรก แต่ถ้าเคยติดตั้งมาก่อนแล้วให้เลือก Existing data directory แล้วเลือก path ที่เก็บ data_dir จากนั้นให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป ดังภาพที่ ก.7

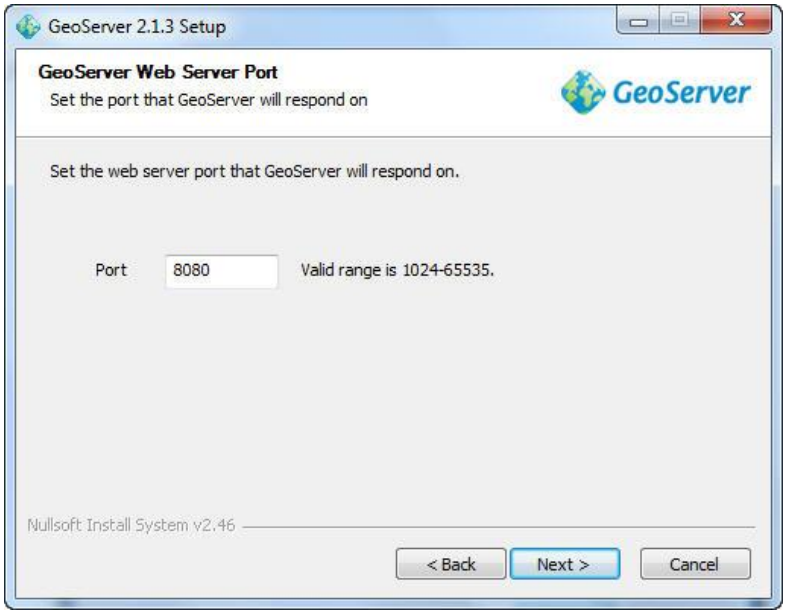
ภาพที่ ก.7 เลือก directory ของข้อมูล

10. หน้า GeoServer Administrator จะให้ใส่ Username และ Password สำหรับใช้ในการเข้าสู่ระบบ Web Administration Interface ของ GeoServer โดยมีค่าเริ่มต้นคือ admin / geoserver จากนั้นให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป ดังภาพที่ ก.8



ภาพที่ ก.8 กำหนดชื่อผู้ใช้และรหัสผ่าน

11. หน้า GeoServer Web Server Port ให้เลือก port ซึ่งใช้สำหรับการเข้าจัดการใน Web Administration Interface ซึ่งตามค่าเริ่มต้น port คือ 8080 จากนั้นให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป ดังภาพที่ ก.9



ภาพที่ ก.9 ระบุ port ของ GeoServer

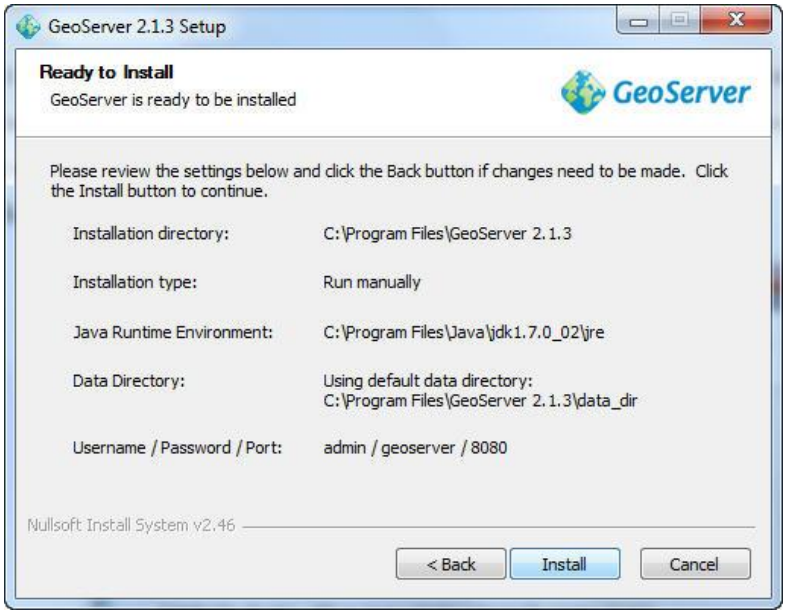
12. หน้า Type of Installation ให้เลือกชนิดของการติดตั้ง

- Run manually คือผู้ดูแลระบบจะต้อง Start service และ Stop service ของ GeoServer สำหรับเปิดและปิดการใช้งาน GeoServer เอง และผู้ดูแลระบบสามารถเห็นการให้บริการข้อมูลของ Server ผ่าน Command line ได้
- Install as a service คือ Service จะรันแบบ Background Service ซึ่งผู้ดูแลระบบจะไม่สามารถเห็นการให้บริการข้อมูลของ Server

(แนะนำให้เลือก Run manually) จากนั้นให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป ดังภาพที่ ก.10

ภาพที่ ก.10 เลือกชนิดของการติดตั้ง

13. หน้า Ready to Install ให้กดปุ่ม Install เพื่อทำการติดตั้งซอฟต์แวร์ ดังภาพที่ ก.11



ภาพที่ ก.11 แสดงรายละเอียดการติดตั้งซอฟต์แวร์

14. สิ้นสุดขั้นตอนการติดตั้งซอฟต์แวร์ GeoServer ให้กดปุ่ม Finish เพื่อเสร็จสิ้นการติดตั้งซอฟต์แวร์ GeoServer ดังภาพที่ ก.12

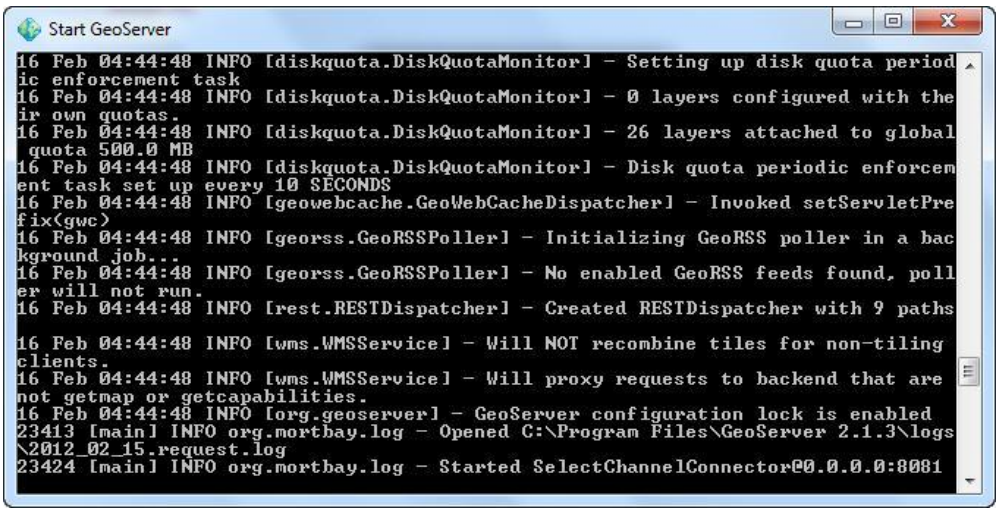
ภาพที่ ก.12 แสดงหน้าจอขั้นตอนสิ้นสุดการติดตั้งซอฟต์แวร์ GeoServer

การเรียกใช้งาน GeoServer

1. ถ้าตอนติดตั้งเลือก Install as a service จะสามารถเข้าหน้า Web Administration Interface ได้เลย แต่ถ้าเลือก Run manually จะต้องไป Start service ของ GeoServer ก่อน โดยไปที่ Start Menu เลือก GeoServer 2.1.3 และเลือก Start GeoServer ดังภาพที่ ก.13

ภาพที่ ก.13 แสดงการ Start service GeoServer

เมื่อ Start GeoServer แล้วจะมี Command line ขึ้นมา ให้รอการ Start GeoServer จนเสร็จก่อน ดังภาพที่ ก.14 เมื่อ Start GeoServer สำเร็จไม่ต้องปิดหน้าต่าง Command line นี้



ภาพที่ ก.14 แสดงหน้าต่างการ Start GeoServer

2. สามารถเข้าหน้า Web Administration Interface ได้โดยเปิด Browser แล้วใส่ url ตามรูปแบบนี้

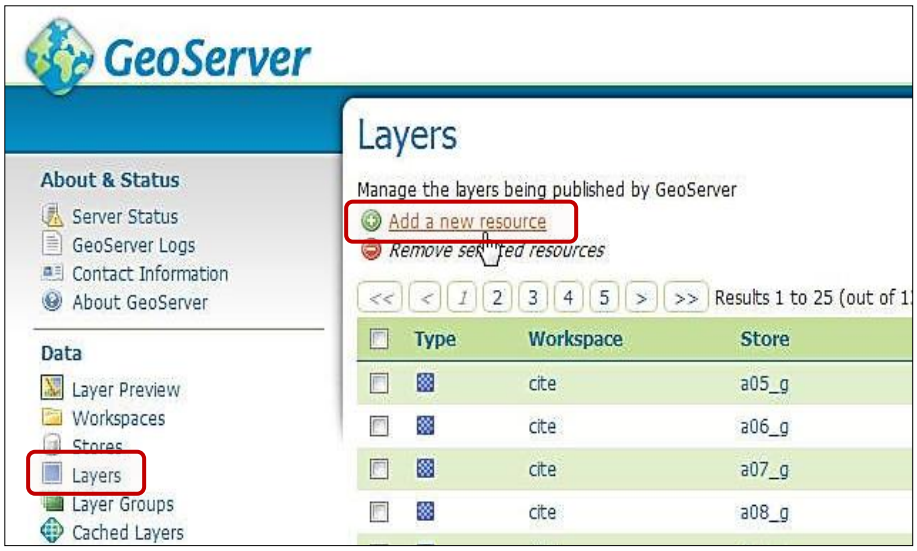
http://[SERVER_URL]:[PORT]/geoserver/ เช่น http://localhost:8080/geoserver/ จะปรากฏหน้า Web Administration Interface ของ GeoServer ใน Browser ดังภาพที่ ก.15

ภาพที่ ก.15 Web Administration Interface ของ GeoServer

ภาคผนวก ข
การเพิ่มชั้นข้อมูลใน GeoServer

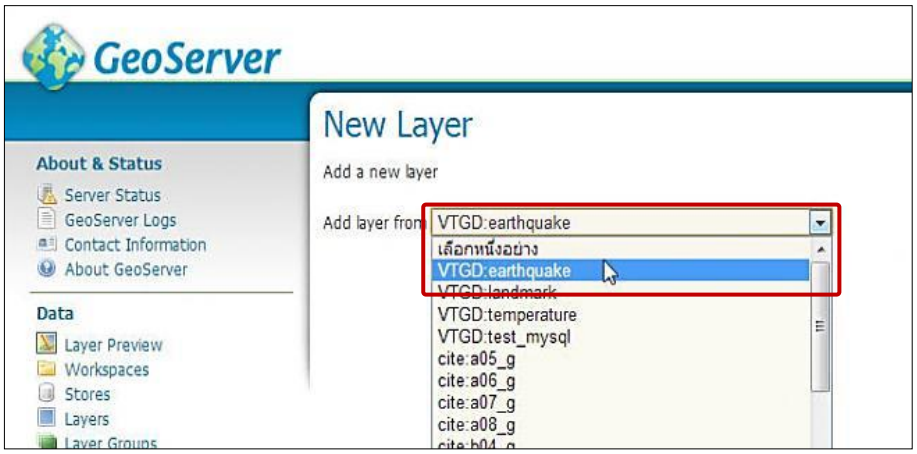
การเพิ่มชั้นข้อมูลใน GeoServer มีขั้นตอนดังต่อไปนี้

1. สร้าง Layer ใหม่ โดยเลือกที่ Layers ในแถบเมนูด้านซ้ายมือ จากนั้นเลือก Add a new resource ดังภาพที่ ข.1



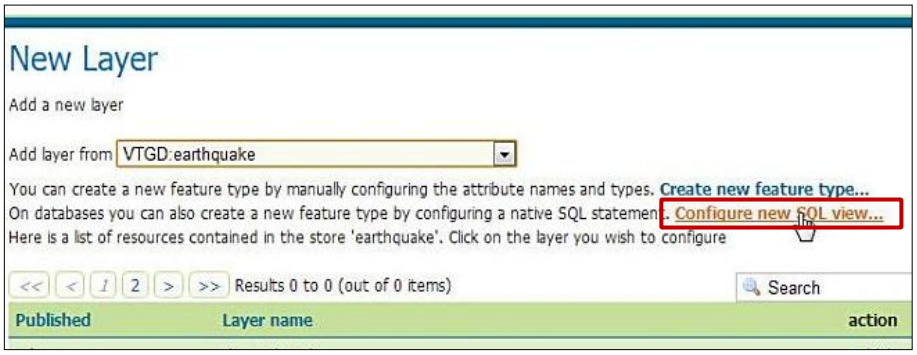
ภาพที่ ข.1 การสร้าง Layer ใหม่ใน GeoServer

2. เลือกฐานข้อมูลที่จะนำมาสร้าง Layer ใหม่ โดยในตัวอย่างนี้จะเลือกฐานข้อมูลแผ่นดินไหวดังภาพที่ ข.2



ภาพที่ ข.2 การเลือกฐานข้อมูลที่จะนำมาสร้าง Layer ใน GeoServer

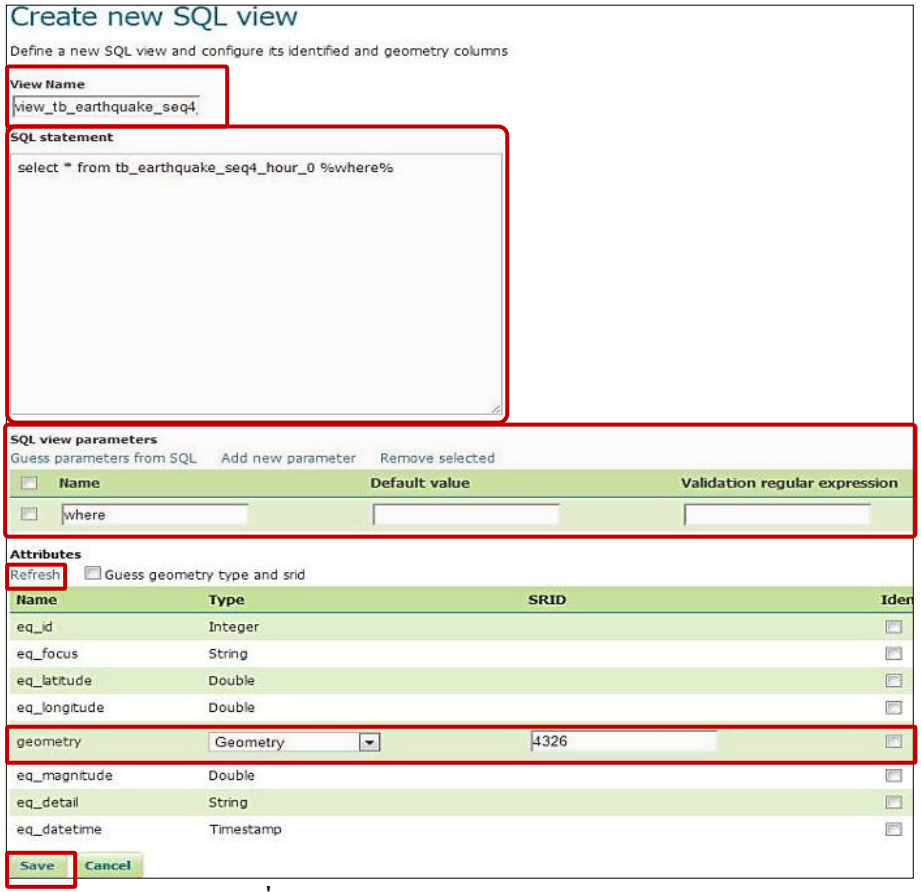
3. เลือก Configure new SQL view ที่อยู่ในทางขวามือเหนือชั้นข้อมูลต่าง ๆ เพื่อให้ Layer ที่สร้างขึ้นมาใหม่สามารถเลือกดูข้อมูลเฉพาะบางช่วงเวลามาแสดงได้ ดังภาพที่ ข.3



ภาพที่ ข.3 การเข้าสู่ Configure new SQL view ใน GeoServer

4. กำหนดรายละเอียดของ SQL view ที่สร้างขึ้นมาใหม่

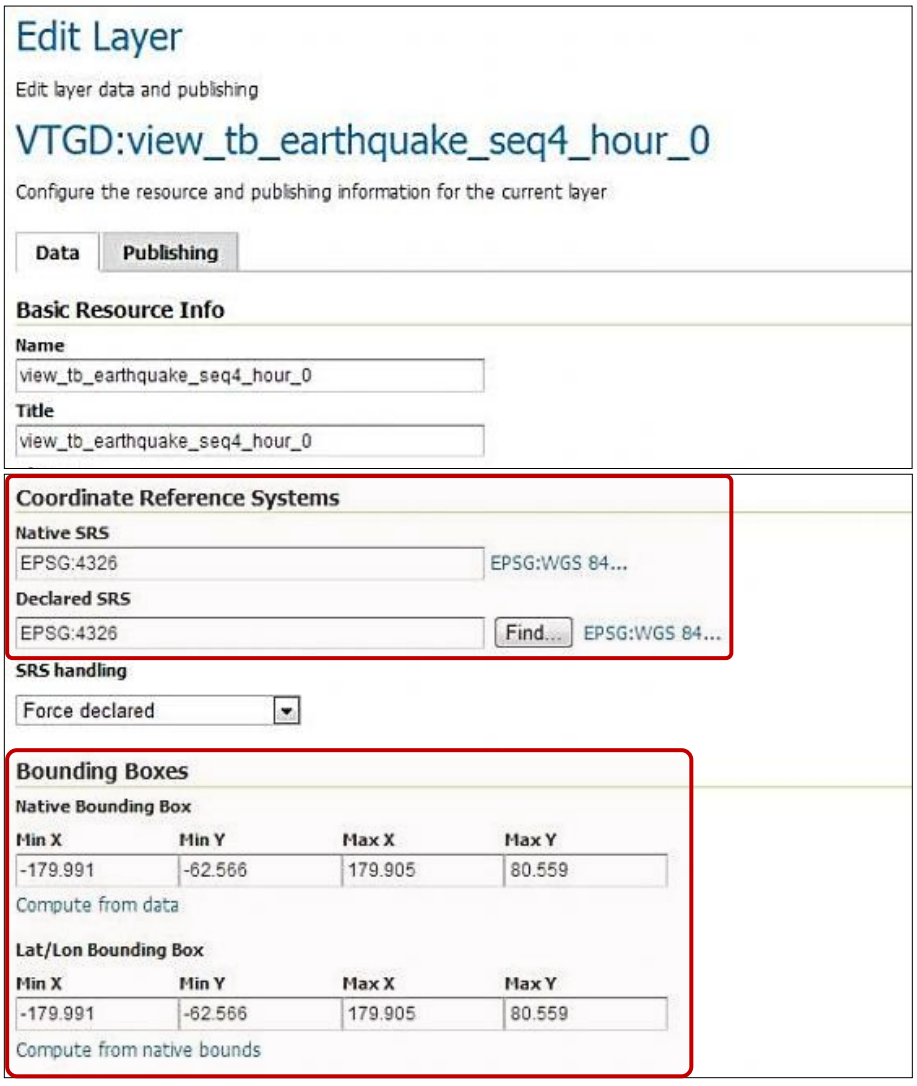
   - กำหนด View Name, SQL statement
   - ในส่วนของ SQL view parameters ให้เลือก Guess parameters from SQL
   - ในส่วนของ Attributes ให้เลือก refresh และเปลี่ยน SRID ของ geometry เป็น 4326 จากนั้นกดปุ่ม save ดังภาพที่ ข.4



ภาพที่ ข.4 การสร้าง SQL view ใน GeoServer

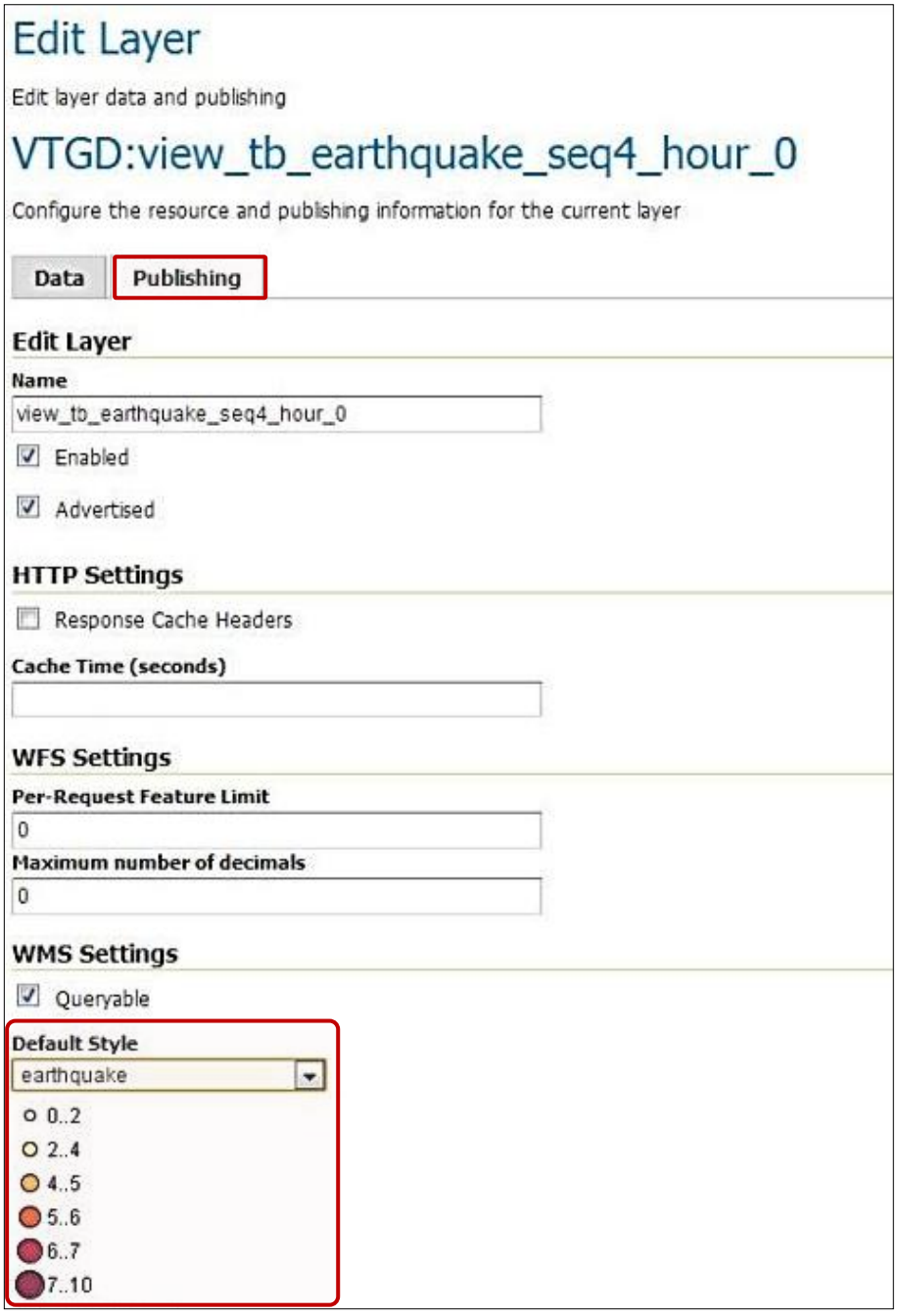
5. เพิ่มรายละเอียดของ Layer ที่สร้างขึ้นมาใหม่ ดังภาพที่ ข.5

   - ในส่วนของ Coordinate Reference Systems กำหนด Declared SRS เป็น EPSG:4326
   - ในส่วนของ Bounding Boxes ให้กด Compute from data และกด Compute from native bounds



ภาพที่ ข.5 การเพิ่มรายละเอียดของ Layer ใน GeoServer

6. กำหนดลักษณะรูปร่างของข้อมูลที่แสดงบนแผนที่ โดยเลือก Tab Publishing จากนั้นเลือกลักษณะที่ต้องการในส่วน Default Style (สามารถข้ามขั้นตอนนี้ได้ โดยจะใช้ลักษณะเริ่มต้นที่กำหนดไว้ให้) จากนั้นกดปุ่ม save ดังภาพที่ ข.6



ภาพที่ ข.6 การกำหนดลักษณะรูปร่างของข้อมูลที่แสดงบนแผนที่ใน GeoServer

ภาคผนวก ค
ระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา

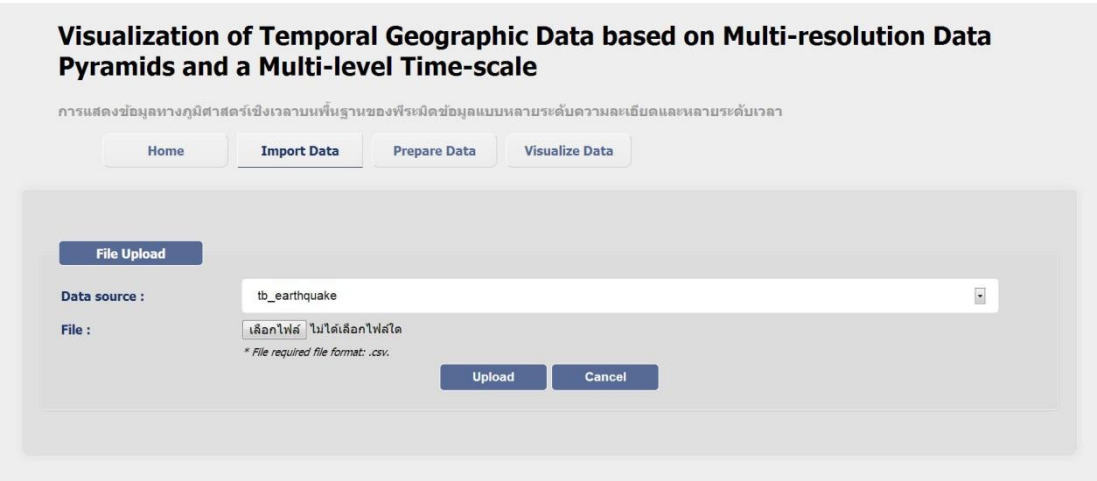
ระบบต้นแบบสำหรับแสดงข้อมูลทางภูมิศาสตร์เป็นระบบที่พัฒนาขึ้นมาในงานวิจัยนี้ สำหรับแสดงข้อมูลแผ่นดินไหว และข้อมูลอุณหภูมิ ซึ่งพัฒนาเป็น Web Mapping Application โดยมีหน้าหลักทั้งหมด 4 หน้า คือ

1. หน้าหลัก (Home)
2. หน้าสำหรับนำข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิเข้าฐานข้อมูล (Import Data)
3. หน้าสำหรับการเตรียมข้อมูล (Prepare Data)
4. หน้าสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา (Visualize Data)

1. หน้าหลัก

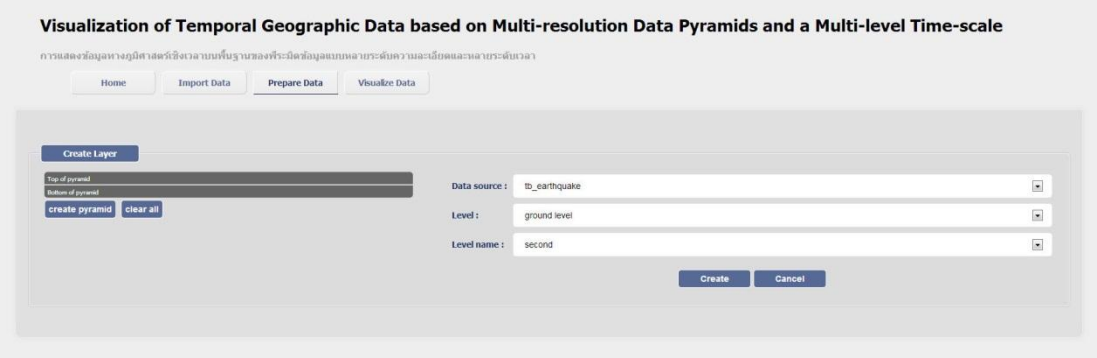
ภาพที่ ค.1 หน้าหลักของระบบต้นแบบ

2. หน้าสำหรับนำข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิเข้าฐานข้อมูลเชิงพื้นที่



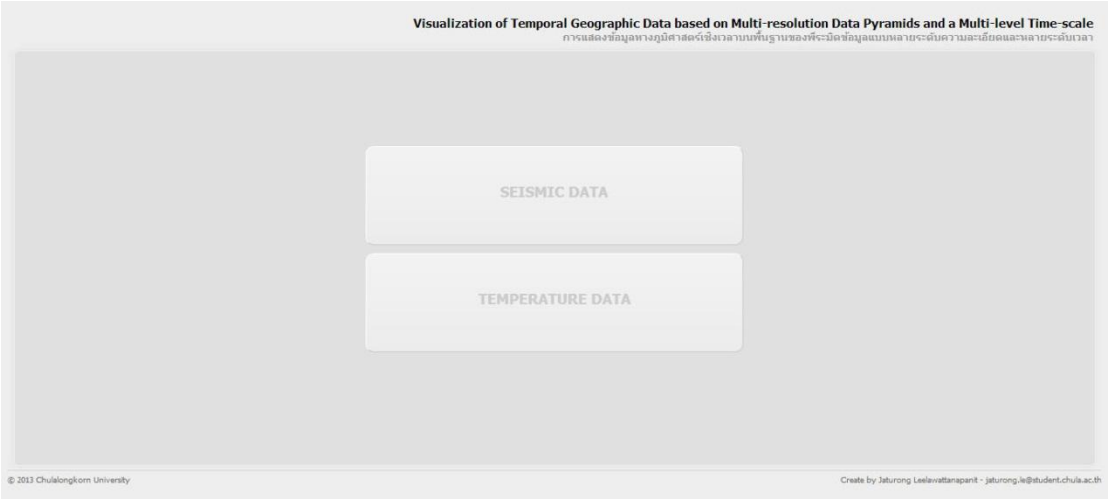
ภาพที่ ค.2 หน้าสำหรับนำข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิเข้าฐานข้อมูล PostgreSQL

3. หน้าสำหรับการเตรียมข้อมูล



ภาพที่ ค.3 หน้าสำหรับการเตรียมข้อมูล

4. หน้าสำหรับแสดงข้อมูลทางภูมิศาสตร์เชิงเวลา



ภาพที่ ค.4 หน้าสำหรับแสดงเตรียมข้อมูลทางภูมิศาสตร์

ในการแสดงข้อมูลทางภูมิศาสตร์จะมีการสร้างเว็บแอพพลิเคชันสำหรับแสดงข้อมูลแผ่นดินไหวและข้อมูลอุณหภูมิ ซึ่งผู้ใช้งานสามารถเลือกดูการแสดงข้อมูลทั้ง 2 ข้อมูลได้จากในหน้านี้ โดยรายละเอียดในการแสดงข้อมูลทั้ง 2 ข้อมูลสามารถอ่านได้ในเล่มวิทยานิพนธ์

ภาคผนวก ง

ไฟล์ JavaScript ของ Timeline

ในงานวิจัยนี้ได้พัฒนา Timeline เป็น JavaScript โดยมีไฟล์ JavaScript ที่เป็นการทำงานของ Timeline ทั้งหมด 4 ไฟล์ คือ

1. vtgdTimeline.js
2. vtgdTimeline-Monitor.js
3. vtgdTimeline-TimeChange-Control.js
4. vtgdTimeline-TimeInterval-Control.js

ในการนำ Timeline นี้ไปใช้งานกับเว็บแอพพลิเคชัน จะต้องนำไฟล์ทั้ง 4 ไฟล์ไปใส่ในเว็บแอพพลิเคชันที่จะพัฒนาด้วย โดยแต่ละไฟล์รายละเอียดดังนี้

1. vtgdTimeline.js

ไฟล์นี้ประกอบไปด้วยในส่วนต้นเป็นการกำหนดตัวแปรที่จะใช้ในการสร้าง Timeline และในส่วนถัดไปเป็นการสร้าง Timeline ในระดับความละเอียดของเวลาต่าง ๆ

2. vtgdTimeline-Monitor.js

ไฟล์นี้เป็นตรวจสอบการเปลี่ยนแปลงของเวลาบน Timeline และส่วนของการเชื่อมต่อกับการดึงชั้นข้อมูลบน GeoServer และส่วนของการแสดงรายละเอียดของข้อมูลที่แสดงอยู่บนแผนที่ ซึ่งผู้ใช้งานสามารถเปลี่ยนแปลงการตั้งค่าได้ในส่วนนี้

3. vtgdTimeline-TimeChange-Control.js

ไฟล์นี้เป็นตรวจสอบการเปลี่ยนแปลงของช่วงเวลาในการดูข้อมูล

4. vtgdTimeline-TimeInterval-Control.js

ไฟล์นี้เป็นการสร้างเส้นกั้นเวลาที่ใช้ในการเลือกช่วงเวลาของข้อมูล

ประวัติผู้เขียนวิทยานิพนธ์

นายจตุรงค์ ลีลาวัฒนพาณิชย์ เกิดเมื่อวันที่ 12 กันยายน 2531 สำเร็จการศึกษาในระดับมัธยมศึกษาจากโรงเรียนศรัทธาสมุทร จ.สมุทรสงคราม เข้าศึกษาต่อในระดับอุดมศึกษาที่คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยมหิดล ในปีการศึกษา 2550 และสำเร็จการศึกษาในปีการศึกษา 2553 หลังสำเร็จการศึกษาระดับปริญญาตรีได้เข้าศึกษาต่อในระดับปริญญามหาบัณฑิตในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2554