****

**โครงงานจำลองการตกและการชนกันของตัวอักษรสามมิติ**

**แบบเรียลไทม์ด้วยคอมพิวเตอร์**

**(Real-time 3D Letters Drop & Collision Simulation)**

**อาจารย์ที่ปรึกษา**

อ. ดร. ณัฐพงศ์ ชินธเนศ

**จัดทำโดย**

นางสาวกรชนก จิรเวศยกุล รหัสนิสิต 5631001721

นายพัทธนันท์ อัครพันธุ์ธัช รหัสนิสิต 5631011021

นายเวศวรุศ งามดำรงเกียรติ รหัสนิสิต 5631086021

นิสิตชั้นปีที่ 3 คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา

Realtime Computer Graphics and Physics Simulation (2110594)

ภาคเรียนที่ 1 ปีการศึกษา 2558

จุฬาลงกรณ์มหาวิทยาลัย

****

**โครงงานจำลองการตกและการชนกันของตัวอักษรสามมิติ**

**แบบเรียลไทม์ด้วยคอมพิวเตอร์**

**(Real-time 3D Letters Drop & Collision Simulation)**

**อาจารย์ที่ปรึกษา**

อ. ดร. ณัฐพงศ์ ชินธเนศ

**จัดทำโดย**

นางสาวกรชนก จิรเวศยกุล รหัสนิสิต 5631001721

นายพัทธนันท์ อัครพันธุ์ธัช รหัสนิสิต 5631011021

นายเวศวรุศ งามดำรงเกียรติ รหัสนิสิต 5631086021

นิสิตชั้นปีที่ 3 คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา

Realtime Computer Graphics and Physics Simulation (2110594)

ภาคเรียนที่ 1 ปีการศึกษา 2558

จุฬาลงกรณ์มหาวิทยาลัย

**โครงงาน เรื่อง** โครงงานจำลองการตกและการชนกันของตัวอักษรสามมิติแบบเรียลไทม์ด้วยคอมพิวเตอร์ (3D Letters Drop & Collision Simulation)

**ผู้จัดทำโครงงาน** นางสาวกรชนก จิรเวศยกุล รหัสนิสิต 5631001721

นายพัทธนันท์ อัครพันธุ์ธัช รหัสนิสิต 5631011021

นายเวศวรุศ งามดำรงเกียรติ รหัสนิสิต 5631086021

**ปีการศึกษา** 2558

**บทคัดย่อ**

การจัดโครงงานในครั้งนี้มีวัตถุประสงค์เพื่อจำลองการตกและชนกันของตัวอักษรสามมิติแบบเรียลไทม์ด้วยคอมพิวเตอร์ โดยใช้ WebGL ในการสร้าง และมีภาษาหลักที่ใช้คือ JavaScript และโปรแกรมหลักที่ใช้ในการเขียนโปรแกรมคือ PHPStorm

คณะผู้จัดทำได้ดำเนินการทำตามขั้นตอนที่ได้วางแผนไว้ และได้นำเสนอเผยแพร่ผลงานโดยนำสื่อวีดีทัศน์ส่งขึ้น Youtube เพื่อให้บุคคลทั่วไปสามารถเข้าถึงได้ผ่านเครือข่ายอินเทอร์เน็ต

ผลการจัดทำโครงงาน พบว่าการจำลองตัวอักษรตกและชนกันแบบเรียลไทม์นั้นสามารถทำได้ แต่จะพบปัญหาเมื่อมีวัตถุบนฉากมากเกินไป กล่าวคือเว็บบราวเซอร์จะทำงานช้ามาก และอีกปัญหาหนึ่งคือการชนกันของตัวอักษรยังไม่ถูกต้องเท่าที่ควร อย่างไรก็ตาม หากมองในภาพรวมแล้ว ถือว่าผลของการทำโครงงานค่อนข้างสมบูรณ์ และบรรลุวัตถุประสงค์ที่วางไว้ครบถ้วน

**กิตติกรรมประกาศ**

โครงงานฉบับนี้สำเร็จได้ด้วยความกรุณาของ อ. ดร. ณัฐพงศ์ ชินธเนศ ซึ่งได้ให้คำปรึกษา ข้อชี้แนะ และความช่วยเหลือจนกระทั่งโครงงานสำเร็จลุล่วงไปได้ด้วยดี คณะผู้จัดทำขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

ขอขอบคุณบิดา มารดา ที่คอยให้กำลังใจเสมอมา และคอยดูแลเอาใจใส่เรื่องสุขภาพของคณะผู้จัดทำไม่เคยขาด

สุดท้ายนี้ คณะผู้จัดทำหวังเป็นอย่างยิ่งว่า โครงงานนี้จะเป็นประโยชน์ต่อการศึกษา และเป็นประโยชน์ต่อผู้ที่สนใจต่อ ๆ ไป

คณะผู้จัดทำ

**สารบัญ**

**บทที่ 1 บทนำ** 1

1.1 ความเป็นมาของโครงงาน 1

1.2 วัตถุประสงค์ 1

1.3 ขอบเขตของโครงงาน 1

1.4 ประโยชน์ที่คาดว่าจะได้รับ 2

**บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง** 3

2.1 Phong Reflection Model 3

2.2 Toon Shading Model 4

2.3 Physics Theory 5

2.3.1 องค์ประกอบในเว็กเตอร์และระบบพิกัดฉาก 5

2.3.2 ปริมาณการเคลื่อนที่ (Quantities of motion) 20

2.3.3 การเคลื่อนที่ในแนว 3 มิติ 20

2.3.4 กฎการเคลื่อนที่ของนิวตัน 23

2.3.5 แรงเสียดทาน 24

2.3.6 กฎการอนุรักษ์พลังงาน 25

2.3.7 การชนแบบไม่ยืดหยุ่น 27

**บทที่ 3 รายละเอียดของการพัฒนา** 28

3.1 ซอฟต์แวร์ที่ใช้ในการพัฒนา 28

3.1.1 Library Three.js 28

3.1.2 Library TextGeometry.js 28

3.1.3 Library FontUtils.js 28

3.1.4 Library Physi.js 28

3.1.5 Library Physijs\_Worker.js 28

3.1.6 Library Ammo.js 28

3.2 ขั้นตอนการดำเนินการ 28

3.3 การแบ่งหน้าที่การทำงาน 29

3.4 โครงสร้างของซอฟต์แวร์ 30

3.4.1 โครงสร้างโดยรวมของซอฟต์แวร์ 30

3.4.2 โครงสร้างไฟล์หลัก ‘index.html’ 30

3.4.3 ส่วนของฟังก์ชันที่ใช้สร้าง Control Panel 34

3.4.4 ส่วนของฟังก์ชันที่ใช้สร้างฉาก พื้น แสง กล้อง และเมฆ 36

3.4.5 ส่วนของฟังก์ชันที่ใช้สร้างตัวอักษร 38

3.4.6 ส่วนที่ใช้สร้าง Shading แบบต่าง ๆ ให้กับ Material 39

3.4.7 ส่วนประมวลผลเพื่อสร้างภาพเคลื่อนไหวอย่างต่อเนื่อง 40

**บทที่ 4 ผลการดำเนินงาน** 41

**บทที่ 5 ปัญหาและอุปสรรค** 42

**บทที่ 6 แนวทางการพัฒนาต่อในอนาคต** 43

**บทที่ 7 เอกสารอ้างอิง** 44

**บทที่ 8 ภาคผนวก** 45

วิธีการใช้งาน Control Pane 45

ลิงค์ไปยังวีดีทัศน์ของโครงงานใน Youtube.com 45

**บทที่ 1 บทนำ**

* 1. **ความเป็นมาของโครงงาน**

ทุกวันนี้ การจำลองภาพสามมิติได้เข้ามามีบทบาทในการดำรงชีวิตของมนุษย์มากขึ้น ดังจะเห็นได้จากการนำการจำลองภาพสามมิติมาใช้ในงานหลายๆ ด้าน เช่น การสร้างภาพยนต์อนิเมชัน การจำลองภาพของอวัยวะต่าง ๆ ในร่างกายเพื่อใช้ในทางการแพทย์ การจำลองภัยพิบัติต่าง ๆ เพื่อประเมินค่าความเสียหายที่อาจเกิดขึ้น การจำลองสภาพอากาศ จนไปถึงการใช้ในงานโฆษณาหรือการตกแต่งวีดีทัศน์ทั่วไปเพื่อเพิ่มความน่าสนใจ ด้วยเหตุนี้เอง คณะผู้จัดทำจึงจัดทำโครงงานนี้ขึ้น เพื่อจำลองการตกและการชนกันของตัวอักษรแบบเรียลไทม์ เพื่อที่จะได้เป็นประโยชน์ต่อผู้ที่ต้องการใช้งานตัวอักษรสามมิติเคลื่อนไหวแบบสมจริง ในงานต่าง ๆ ในอนาคต

* 1. **วัตถุประสงค์**

1. เพื่อสร้างภาพเคลื่อนไหวจำลองการตกและชนกันของตัวอักษรสามมิติแบบเรียลไทม์ โดยมีการใส่ Shading ให้กับวัตถุในแบบต่างๆ
2. เพื่อศึกษาการสร้างและเคลื่อนวัตุสามมิติในรูปแบบต่างๆ ด้วย WebGL และศึกษาการสร้างสิ่งแวดล้อมเพื่อให้วัตถุเหล่านั้นมีปฏิกิริยาต่อกันและกันเสมือนอยู่ในโลกความจริง

**1.3 ขอบเขตของโครงงาน**

1. ตัวอักษรสามมิติเป็นชื่อภาษาไทยของนิสิตในกลุ่มทั้งสามคน
2. ตัวอักษรสามมิติตกลงมาจากด้านบนของจอภาพตามกฎของฟิสิกส์ ด้วยแรงโน้มถ่วงที่สามารถปรับค่าได้โดยผู้ใช้
3. ตัวอักษรสามมิติสามารถชนกันได้ ตามกฎของฟิสิกส์
4. มีพื้นเป็นส่วนหนึ่งของสภาพแวดล้อม ที่วัตถุสามมิติสามารถหล่นลงมาถึงและตกกระทบ ทำให้ตัวอักษรสามมิติเคลื่อนที่ไปตามกฎของฟิสิกส์
5. มีแสงเป็นส่วนหนึ่งของสภาพแวดล้อม ที่ทำให้ตัวอักษรสามมิติสะท้อนแสงให้เห็นถึง shading ในแบบ Phong Reflection Model โดยค่าตัวแปรต่างๆ ของวัตถุ สามารถถูกปรับได้โดยผู้ใช้
6. มีแสงเป็นส่วนหนึ่งของสภาพแวดล้อม ที่ทำให้ตัวอักษรสามมิติสะท้อนแสงให้เห็นถึง shading ในแบบ Toon Shading Model โดยค่าสีของวัตถุสามารถปรับได้โดยผู้ใช้

**1.4 ประโยชน์ที่คาดว่าจะได้รับ**

1. ได้ความรู้เกี่ยวกับการการสร้างและเคลื่อนวัตุสามมิติในรูปแบบต่างๆ ด้วย WebGL และการสร้างสิ่งแวดล้อมเพื่อให้วัตถุเหล่านั้นมีปฏิกิริยาต่อกันและกันเสมือนอยู่ในโลกความจริง
2. เป็นประโยชน์ต่อผู้ที่ต้องการศึกษาหรือใช้งานตัวอักษรสามมิติเคลื่อนไหวแบบสมจริงในอนาคต

**บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง**

**2.1 Phong Reflection Model**

Phong reflection is a local illumination model devised by Bui Tuong Phong and can produce a certain degree of realism in three-dimensional objects by combining three elements - diffuse, specular and ambient lighting for each considered point on a surface. It has several assumptions - all lights are points, only surface geometry is considered, only local modelling of diffuse and specular exists, specular colour is the same as light colour, and ambient is a global constant.

Diffuse,

I_d = I_i k_d \cos \theta0 \le \theta \le \pi/2

where I_iis the brightness of the (point) light source. θ is the angle between the surface normal and the direction of the light source. k_dis the reflection coefficient.

For multiple light sources,

I_d = k_d \sum_{n} I_{i,n} (L_n \cdot N)

where *L* and *N* are unit vectors, L_nbeing the direction vector from the surface to the *n*th light source.

Specular (highlight producing),

I_s = I_i k_s \cos^n \Omega = I_i k_s (R \cdot V)^n

where *n* indicates the surface reflectivity, infinity would indicate a perfect mirror. Ω is the angle between the 'mirror' and the viewer. R is the specular direction and V is the actual view vector. This fails to produce radiosity, a flaw in Phong.

Combining diffuse and specular is sufficient for local accuracy. To mimic global conditions an ambient element is added to give general illumination, usually as a constant value

I_g = I_a k_a

Combining all three gives

I = I_a k_a + I_i (k_d (L \cdot N) + k_s (R \cdot V)^n)

assuming no reduction of light intensity with distance, which can be added if desired.

This is an empirical model, which is not based on physics' description of light interaction, but instead on physical observation. Phong observed that for very shiny surfaces the specular highlight was small and the intensity fell off rapidly, while for duller surfaces it was larger and fell off more slowly.

(\*) "color and ambient" are not the color of the model and the ambient lightning. "color" is the ambient color and "ambient" is the luminence of the ambient color. The object showed here is gray. But placed in a blue environnement. Thus, this image is very dangerous to consider and is quite inappropriate.

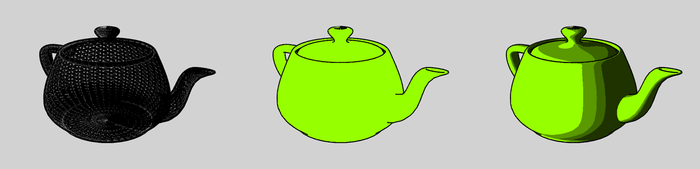
**2.2 Toon shading Model**

Toon shading or Cel shading is a type of non-photorealistic rendering designed to make 3-D computer graphics appear to be flat by using less shading color instead of a shade gradient or tints and shades. Cel-shading is often used to mimic the style of a comic book or cartoon. It is somewhat recent, appearing from around the beginning of the twenty-first century. The name comes from cels (short for celluloid), the clear sheets of acetate which are painted on for use in traditional 2D animation.

The cel-shading process starts with a typical 3D model. Where cel-shading differs from conventional rendering is in its non-photorealistic illumination model. Conventional (smooth) lighting values are calculated for each pixel and then quantized to a small number of discrete shades to create the characteristic flat look – where the shadows and highlights appear more like blocks of color rather than mixed in a smooth way.

Black "ink" outlines and contour lines can be created using a variety of methods. One popular method is to first render a black outline, slightly larger than the object itself. Backface culling is inverted and the back-facing triangles are drawn in black. To dilate the silhouette, these back faces may be drawn in wireframe multiple times with slight changes in translation. Alternatively, back-faces may be rendered solid-filled, with their vertices translated along their vertex normals in a vertex shader. After drawing the outline, back-face culling is set back to normal to draw the shading and optional textures of the object. Finally, the image is composited via Z-buffering, as the back-faces always lie deeper in the scene than the front-faces. The result is that the object is drawn with a black outline and interior contour lines. The term "cel-shading" is popularly used to refer to the application of this "ink" outlining process in animation and games, although originally the term referred to the shading technique regardless of whether the outline was applied.

The Utah teapot rendered using cel-shading:

[](https://en.wikipedia.org/wiki/File:Celshading_teapot_large.png)

1. The back faces are drawn with thick lines
2. The object is drawn with a basic texture
3. Shading

**2.3 Physics Theory**

**2.3.1 องค์ประกอบในเว็กเตอร์และระบบพิกัดฉาก**

เวกเตอร์เป็นเครื่องมือทางคณิตศาสตร์อย่างหนึ่ง ซึ่งดูเหมือนไม่มีความจำเป็นแต่เป็นเครื่องมืออย่างแรกที่ต้องใช้ อย่างไรก็ตามในการศึกษาทางฟิสิกส์เวกเตอร์เป็นเครื่องมือที่มีความสำคัญ เพื่อช่วยในอำนวยความสะดวกในการคำนวณ

**ระบบพิกัด (แกนอ้างอิง)**

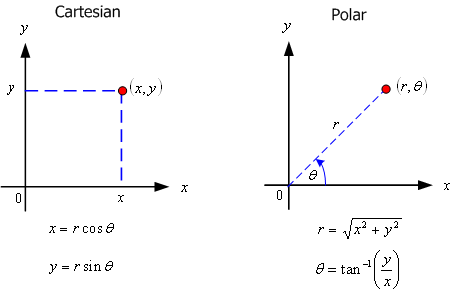
ระบบพิกัดมีความสำคัญเพื่อช่วยในการวัดมีความหมาย เช่นอีก 800 m ถึงอาคารวิทยาศาสตร์เป็นการบอกที่ไม่สมบูรณ์ เนื่องจากไม่ทราบว่าเริ่มต้นที่ตำแหน่งใด

ดังนั้นในการกำหนดระบบพิกัดต้องทราบ

1. จุดเริ่มต้น
2. ชนิดของระบบพิกัด (พิกัดฉาก ; พิกัดเชิงขั้ว ; พิกัดทรงกระบอก)
3. ทิศตามแกน

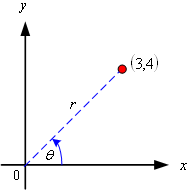
**ระบบพิกัดแบบมาตรฐานใน 2 มิติ**

ได้แก่ระบบพิกัดฉาก (Cartesian) และ ระบบพิกัดเชิงขั้ว (Polar)



รูปที่ 2.1 แสดงระบบพิกัดใน 2 มิติ

**ตัวอย่างที่ 2.1** จงหาระบบพิกัดเชิงขั้วของจุด  ดังรูปที่ 2.2

**วิธีทำ**  จากทฤษฎีพิธากอรัส

 =  =  = 

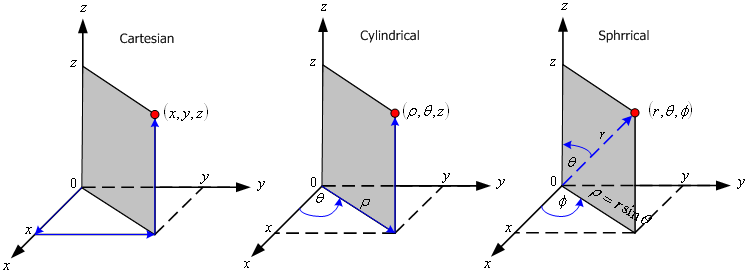
 = 

= 

รูปที่ 2.2

**ระบบพิกัดแบบมาตรฐานใน 3 มิติ**

ได้แก่ระบบพิกัดฉาก (Cartesian) และ ระบบพิกัดทรงกระบอก (Cylindrical) และระบบพิกัดทรงกลม (Spherical)



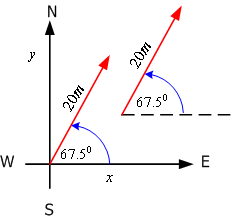
รูปที่ 2.3 ระบบพิกัดฉากใน 3 มิติ

**เวกเตอร์**

สเกลาร์ : ปริมาณที่มีเฉพาะขนาดอย่างเดียว

เวกเตอร์ : ปริมาณที่มีทั้งขนาดและทิศทาง

**ตัวอย่างที่ 2.2** นักศึกษาคนหนึ่งเดินทางไปทางทิศตะวันออกเฉียงไปทางเหนือได้ระยะทาง  ดังรูปที่ 2.4 ขวามมือ จงเขียนรูปแสดงการเดินทางในรูปของเวกเตอร์

 **วิธีทำ** เราสามารถเขียนให้อยู่ในรูปของเวกเตอร์ได้ดังรูปที่ 2.4 ทางซ้ายมือ เมือความยาวของลูกศรแทนขนาดของเวกเตอร์มีค่าเท่ากับ  (โดยใช้มาตราส่วน ) หัวลูกศรแสดงทิศทางของเวกเตอร์

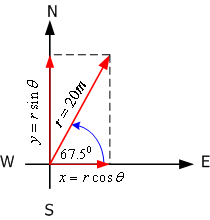
รูปที่ 2.4

**ข้อสังเกตุ** เวกเตอร์สามารถเลื่อนออกจากจุดเริ่มต้นได้ โดยขนาดและทิศทางคงที่

การเขียนเวกเตอร์ดังรูปที่ 2.4 เป็นการไม่สะดวกเราสามารถเขียนใหม่ได้เป็น  ทำมุม  กับแกน  ในหนังสือเล่มนี้จะแทนเวกเตอร์ด้วย 

**ส่วนประกอบของเวกเตอร์**

**ตัวอย่างที่ 2.3** นักศึกษาคนหนึ่งเดินทางไปทางทิศตะวันออกเฉียงไปทางทิศเหนือได้ระยะทาง  ดังรูปที่ 2.5 จงหาระยะทางที่เขาเดินทางได้ในทิศเหนือและทิศตะวันออก

 **วิธีทำ** เปลี่ยนจากระบบพิกัดเชิงขั้วเป็นระบบพิกัดฉากจากรูปที่ 2.5 จะได้

 = 

= 

=  ทิศเหนือ

 = 

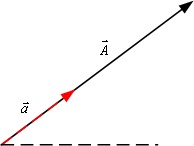
= 

=  ทิศตะวันออก

ส่วนของเวกเตอร์  บนแกน  และ  เราเรียกว่า **"ส่วนประกอบ"** ของเวกเตอร์ 

รูปที่ 2.5

**เวกเตอร์หนึ่งหน่วย** คือเวกเตอร์ที่มีขนาด 1 หน่วย และมีทิศทางตามเวกเตอร์ที่พิจารณา เช่น ให้  เป็นเวกเตอร์ที่มีขนาดเท่ากับ  และ  เป็นเวกเตอร์หนึ่งหน่วย ที่มีทิศเดียวกับเวกเตอร์  ดังรูปที่ 2.6



** = **

หรือเขียนใหม่ได้เป็น

 = 

รูปที่ 2.6

ดังนั้นในระบบพิกัดแกนมุมฉาก เวกเตอร์หนึ่งหน่วยแทนด้วย

 มีขนาด 1 หน่วยทิศตามแกน 

 มีขนาด 1 หน่วยทิศตามแกน 

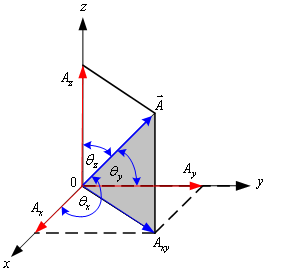
 มีขนาด 1 หน่วยทิศตามแกน 

โดยเวกเตอร์ทั้งสามตั้งฉากซึ่งกันและกัน และเวกเตอร์ทั้งสามจะเรียงกันในทิศทวนเข็มนาฬิกา

จากตัวอย่างที่ 2.3 คำตอบที่ได้สามารถเขียนใหม่ได้เป็น  เป็นการเขียนในรูปแบบมาตรฐานซึ่งมีความสะดวกมากเมื่อนำไปใช้ในการบวกและการคูณเวกเตอร์

รูปแแบบทั่ว ๆ ไปของเวกเตอร์เขียนได้ดังนี้  เมื่อ  คือส่วนประกอบบนแกน  ของ  ;  คือส่วนประกอบบนแกน  ของ  ;  คือส่วนประกอบบนแกน  ของ  เป็นเวกเตอร์ใน 3 มิติ

**ส่วนประกอบของเวกเตอร์ใน 3 มิติ**

 ถ้า  อยู่บนระบบพิกัดฉาก ; ;  โดยที่  ทำมุม ; ; กับแกน ; และ ตามลำดับ ดังรูปที่ 2.7

ส่วนประกอบของ  บนระนาบ  และ  คือ

 = 

= 

เพราะว่า  = 

แต่  = 

 = 

 = 

รูปที่ 2.7

ทิศทางของเวกเตอร์  หาได้โดยใช้โคไซน์บอกทิศ (direction cosine) โดยที่  ; ;  เมื่อ



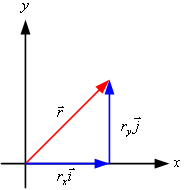
ดังนั้นส่วนประกอบของ  บนแกน ;  และ  เขียนใหม่ได้เป็น

 = 

= 

**การบวกเวกเตอร์**

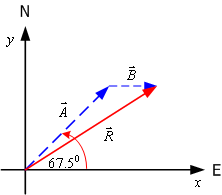
เวกเตอร์  สามารถหาได้จากส่วนประกอบของเวกเตอร์เข้าด้วยกันแสดงดังรูปที่ 2.8 เมื่อ 



การบวกเวกเตอร์ทำได้โดยการนำหางของเวกเตอร์ตัวที่สอง  ต่อเข้ากับหัวของเวกเตอร์อันแรก  ส่วน  ได้จากการลากจากหางของเวกเตอร์อันแรกไปยังหัวเวกเตอร์ตัวที่สอง

รูปที่ 2.8 แสดงการบวกเวกเตอร์

**ตัวอย่างที่ 2.4** จงหาระยะการกระจัดทั้งหมดของนักศึกษาคนหนึ่งซึ่งเดินทางไปทางทิศตะวันออกเฉียงไปทางเหนือได้ระยะทาง  จากนั้นเดินทางไปทางทิศตะวันออก  ดังรูปที่ 2.9



**วิธีทำ** จากตัวอย่างที่ 2.3 ระยะการกระจัดครั้งแรกคือ

 = 

ระยะการกระจัดครั้งที่ 2

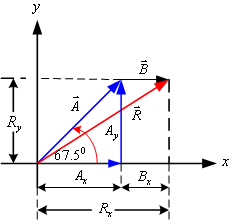
 = 

คำตอบที่ได้คือเวกเตอร์  ซึ่งเราเรียกว่า **ผลลัพธ์** ผลลัพธ์ที่ได้นี้สามารถหาได้ดังนี้

รูปที่ 2.9

1. **เขียนรูป** โดยการกำหนดมาตราส่วนแล้วนำมาเขียนรูปผลลัพธ์ที่ได้ หาได้โดยการวัดดังรูปที่ 2.9
2. **วิธีคำนวณ** โดยการแยกองค์ประกอบของ และ ลงบนบนแกน  และ

แกน  จากนั้นนำองค์ประกอบในแต่ละแกนมารวมกัน

 องค์ประกอบบนแกน  ของเวกเตอร์  หาได้จากการนำองค์ประกอบบนแกน  ของ  และ มาบวกกัน

 = 

= 

= 

ในทำนองเดียวกันองค์ประกอบบนแกน 

ของเวกเตอร์  หาได้จากการนำองค์ประกอบบนแกน  ของ  และ มาบวกกัน

 = 

= 

= 

รูปที่ 2.10

1. **ใช้เวกเตอร์หนึ่งหน่วย**

เมื่อ  = 

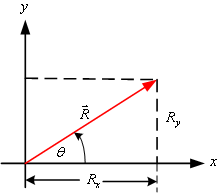
= 

จัดรูปใหม่

 = 

= 

จากคำตอบที่ได้ในข้อ ข. และ ข้อ ค. เราสามารถหาผลลัพธ์ได้โดยใช้พิธากอรัส และอาศัยฟังก์ชั่นทางตรีโกณมิติจากสามเหลี่ยมมุมฉากดังรูปที่ 2.11 จากวิธีคำนวณจะสังเกตุเห็นว่าเมื่อเขียนอยู่ในรูปของเวกเตอร์หนึ่งหน่วยทำให้เราสามารถหาผลลัพธ์ได้สะดวกยิ่งขึ้น

  = 

= 

= 

 = 

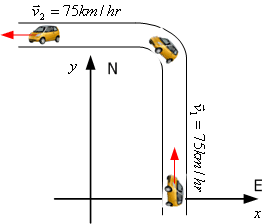
= 

=  กับแกน 

รูปที่ 2.11

นอกจากนี้ปริมาณเวกเตอร์ได้แก่ การกระจัด ความเร็ว ความเร่ง แรง โมเมนตัม ทอร์ค น้ำหนักเป็นต้น

**ตัวอย่างที่ 2.5** รถยนต์คันหนึ่งวิ่งไปทางทิศเหนือด้วยความเร็ว  จากนั้นเลี้ยวไปทางทิศตะวันตกและวิ่งด้วยความเร็ว  ดังรูปที่ 2.12 จงหาการเปลี่ยนแปลงความเร็วของรถยนต์

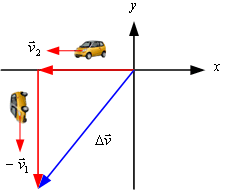


**วิธีทำ** จากรูปที่ 2.12 เมื่อนำมาเขียนในรูปของเวกเตอร์หนึ่งหน่วย

 = 

 = 

รูปที่ 2.12

 แต่อัตราการเปลี่ยนแปลงความเร็วคือ

 = 

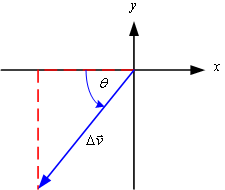
= 

นั่นคือการลบเวกเตอร์ เป็นการบวกเวกเตอร์ในทิศตรงกันข้ามคือ  เมื่อนำรูปมาเขียนใหม่ได้ดังรูปที่ 2.13 ดังนั่นอัตราเร็วเมื่อเขียนอยู่ในรูปของเวกเตอร์หนึ่งหน่วย

 = 

= 

รูปที่ 2.13

 เราสามารถหาขนาดและทิศทางของ  ได้ดังนี้

 = 

= 

= 

รูปที่ 2.14

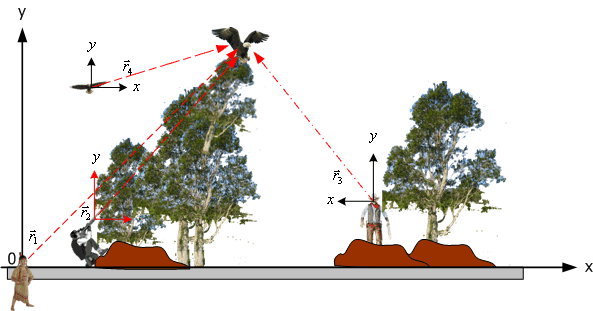
 = 

= 

=  กับแกน  ดังรูปที่ 2.14

**เวกเตอร์ตำแหน่ง**

เวกเตอร์ตำแหน่ง คือเวกเตอร์ทีบอกตำแหน่งของวัตถุเมื่อเทียบกับจุดใดจุดหนึ่งดังรูปที่ 2.15

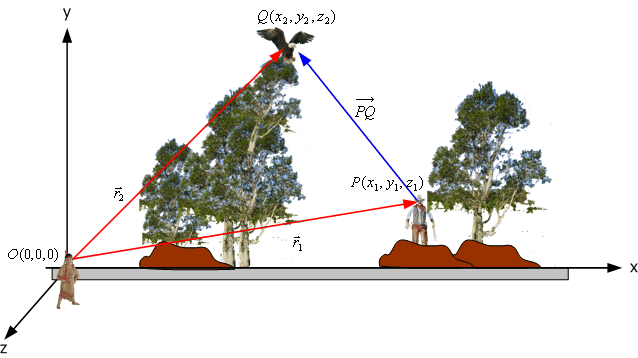


รูปที่ 2.15 แสดงเวกเตอร์ตำแหน่ง

จากรูปที่ 2.15 เด็กผู้หญิง นายพราน ผู้ชาย และนกตัวที่บิน สังเกตุนกตัวที่เกาะอยู่บนต้นไม้โดยเทียบกับตัวเอง เช่น  เป็นเวกเตอร์บอกตำแหน่งของผู้ชายเทียบกับนกตัวที่เกาะอยู่บนต้นไม้ หรือ  เป็นเวกเตอร์บอกตำแหน่งของนกตัวที่บินเทียบกับนกตัวที่เกาะบนต้นไม้

ตำแหน่งของนกที่เกาะอยู่บนต้นบนไม้จะอยู่ที่ตำแหน่งต่างกันเมื่อผู้สังเกต

ต่างกันดังนั้น ดังนั้นเพื่อหลีกเลี่ยงปัญหาดังกล่าวการบอกเวกเตอร์ตำแหน่ง จึงกำหนดจุดใดจุดหนึ่งเป็นจุดเปรีบยเทียบ ไม่ว่าใครจะเป็นผู้สังเกตก็ให้ใช้จุดเปรียบเทียบจุดเดียวกัน แสดงดังรูปที่ 2.16



รูปที่ 2.16 เวกเตอร์ตำแหน่งของจุด  และจุด 

จากรูปที่ 2.16 ให้  และ  เป็นเวกเตอร์ตำแหน่งของจุด (ผู้ชาย) อยู่ที่ตำแหน่ง  และ  (นกเกาะที่ต้นไม้) อยู่ที่ตำแหน่ง เมื่อเทียบกับจุด  (เด็กผู้หญิง) อยู่ที่ตำแหน่ง จะได้

 =  (มอง)

= 

= 

 =  (มอง )

= 

= 

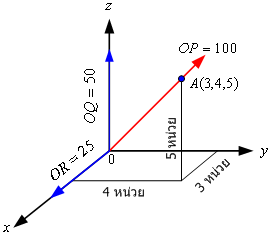
อาศัยการบวกเวกเตอร์โดยการเขียนรูปจะได้

 = 

 =  (มอง )

=  (ตำแหน่งสุดท้ายลบตำแหน่งตั้งต้น)

**ตัวอย่างที่ 2.6** จากรูปที่ 2.17 เวกเตอร์  มีขนาด 100 หน่วย โดยแนวของเวกเตอร์  ผ่านจุด  หน่วย เวกเตอร์ มีขนาด 25 หน่วยบนแกน  และเวกเตอร์  มีขนาด 50 หน่วยบนแกน  จงหา 

 **วิธีทำ** คำนวณโดยใช้ **เวกเตอร์หนึ่งหน่วย** (unite vector)

สมมติให้  อยู่ที่จุด 

 =  และ

 = 

รูปที่ 2.17

 = 

= 

 =  =  หน่วย

=  หน่วย

แต่  =  =  หน่วย

ดังนั้น  =  = หน่วย

 = 

 = 

 = 

คำนวณโดยใช้**โคไซน์บอกทิศ** (direction cosine)

 = 

= 

 =  =  หน่วย

=  หน่วย

เมื่อ  = 

 = 

แต่  =  =  หน่วย

 =  =  หน่วย

 =  =  หน่วย

จากโจทย์  =  หน่วย

ดังนั้น  =  หน่วย

 = 

 = 

 = 

**การคูณเวกเตอร์**

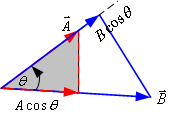
การคูณเวกเตอร์มี 2 ชนิดได้แก่

ก. ผลคูณที่ได้เป็นปริมาณสเกลาร์ เรียกว่าผลคูณแบบดอท (dot product) หรือผลคูณสเกลาร์ (scalar product)

ข. ผลคูณที่ได้เป็นปริมาณเวกเตอร์ เรียกว่าผลคูณแบบครอส (cross product) หรือผลคูณเวกเตอร์ (vector)

ให้  = 

 = 

**ผลคูณสเกลาร์** :

 = 

= 

(เพราะว่า  และ )

รูปที่ 2.18

เมื่อ  และ  คือขนาดของเวกเตอร์  และ 

 คือมุมระหว่างเวกเตอร์  และ  เมื่อ 

จากรูปที่ 2.18 เงาของ บน คือ  และ เงาของ บน คือ  นั่นคือผลคูณสเกลาร์คือการคูณเวกเตอร์กับเงาของอีกเวกเตอร์หนึ่ง

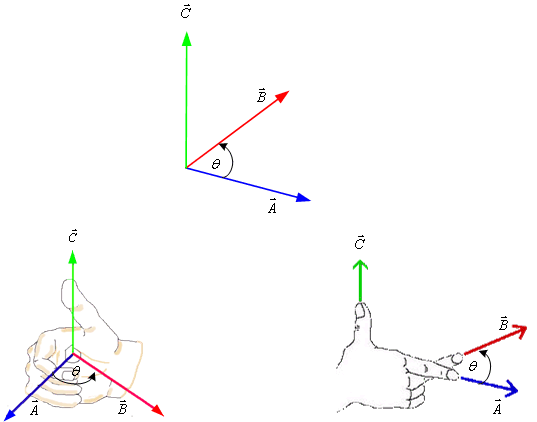
**ผลคูณเวกเตอร์** :

 = 

= 

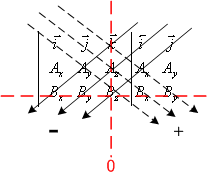
เมื่อ  คือเวกเตอร์ผลลัพธ์ทิศของเวกเตอร์  หาได้โดยใช้กฎมือขวาเมื่อ  แสดงดังรูปที่ 2.19 เพื่อความสะดวกในการคำนวณเราสามารถหาผลคูณแบบเวกเตอร์ได้โดยใช้เมตริกซ์

 = 



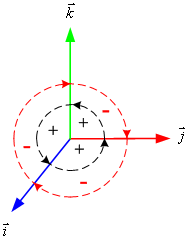
รูปที่ 2.19

 = 



=

การพิจาณาเครื่องหมายผลคูณเวกเตอร์พิจารณาโดยการตั้งแกนแบ่งครึ่งทั้งสองข้างให้เท่ากันค่าทางซ้ายเครื่องหมายลบ ค่าทางขวาเครื่องหมายบวกตามระบบแกน



หรือการพิจาณาเครื่องหมายผลคูณเวกเตอร์ดังรูปที่ 2.20 ทิศทวนเข็มเครื่องหมายบวก ทิศตามเข็มเครื่องหมายลบเมื่อ

 ;  ;  ;  ;  ; 

รูปที่ 2.20

**ตัวอย่างที่ 2.6** พิจารณาเวกเตอร์  และ จงหา

1. ผลคูณสเกลาร์
2. ผลคูณเวกเตอร์
3. มุมระหว่างเวกเตอร์ทั้งสอง

**วิธีทำ**

ก.  = 

= 

= 

ข.  = 

=  = 

= 

= 

ผลคูณแบบเวกเตอร์  มีขนาด  หน่วย ทิศตามแกน 

ค.  = 

 = 

= 

= 

= 

**สรุป**

ระบบพิกัดฉาก มีความสำคัญ และใช้ในการระบุตำแหน่ง

เวกเตอร์ แสดงตำแหน่งและทิศทาง

ส่วนประกอบเวกเตอร์ :  และ 

เวกเตอร์หนึ่งหน่วย : 

การบวกเวกเตอร์ :  เมื่อ  และ 

ผลคูณสเกลาร์ : 

ผลคูณเวกเตอร์ : = 

**2.3.2 ปริมาณการเคลื่อนที่ (Quantities of motion)**

1. ระยะทาง ( Destance ) หมายถึง ระยะที่วัตถุเคลื่อนที่ได้จริง ๆ โดยจะต้องมีตำแหน่งเริ่มต้นตำแหน่งสุดท้าย และเส้นทางการเคลื่อนที่ของวัตถุ สัญลักษณ์ s เป็นปริมาณสเกลาร์ มีหน่วยเป็น เมตร (m)

2. การกระจัด (displacement) หมายถึง ระยะทางที่สั้นที่สุดในการย้ายตำแหน่งจากจุกเริ่มต้นไปยังจุดสุดท้าย ของการเคลื่อนที่ในช่วงที่พิจารณา สัญลักษณ์ เป็นปริมาณเวกเตอร์ มีหน่วยเป็น เมตร (m)

3. อัตราเร็ว ( Speed ) หมายถึง ระยะทางที่วัตถุเคลื่อนที่ได้ในหนึ่งหน่วยเวลา เป็นปริมาณสเกลาร์ มีหน่วยเป็น เมตร / วินาที

อัตราเร็วเฉลี่ย หมายถึง อัตราส่วนระหว่างระยะทางทั้งหมดที่เคลื่อนที่ได้กับช่วงเวลาที่ใช้ในการเคลื่อนที่นั้น

4.ความเร็วเฉลี่ย หมายถึง ความเร็วระหว่างช่วงเวลาหนึ่งหรือการกระจัดที่วัดได้ในช่วงเวลาทั้งหมด ทิศของความเร็วที่เปลี่ยนไป จะอยู่ทิศการเปลี่ยนไปการขจัด

**2.3.3 การเคลื่อนที่ในแนว 3 มิติ**

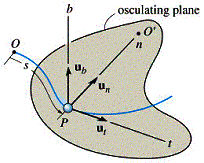
ถ้าหากว่าอนุภาคเคลื่อนที่ตามแนวเส้นโค้งในสามมิติ เราสามารถที่จะกำหนดพิกัดได้ดังต่อไปนี้

สำหรับเวคเตอร์ตามแนวเส้นสัมผัส เราสามารถที่จะสร้างขึ้นได้เพียงเวคเตอร์เดียว เพราะเส้นสัมผัสส่วนโค้งนั้นมีได้เพียงเส้นเดียว และเรากำหนดให้ทิศทางของเวคเตอร์นี้ต้องมีทิศทางไปตามการเคลื่อนที่ ของอนุภาคดังนั้นเราจึงสามารถหาเวคเตอร์ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image001.gifได้อย่างแน่นอนเพียงเวกเตอร์เดียว

สำหรับเวคเตอร์ตามแนวตั้งฉากนั้น เนื่องจากมีเวกเตอร์จำนวนไม่จำกัดที่สามารถสร้างขึ้นให้ตั้งฉากกับ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image002.jpgได้ เราจึงจะต้องสร้างเวกเตอร์ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image004.jpgให้ เป็นเวกเตอร์ที่ตั้งฉาก และมีทิศทางพุงเข้าสู่จุดศูนย์กลางความโค้งของเส้นโค้งนั้น ซึ่งแกนแกนนี้เราเรียก แกนตั้งฉากหลัก (Principal Normal) และให้ระลึกอยู่เสมอว่าทิศทางของ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image005.jpgนี้จะต้องชี้ไปในทิศที่เส้นโค้งนี้โค้งเข้าเสมอ

หลังจากที่เวกเตอร์ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image006.jpgและ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image007.jpgได้ถูกกำหนดแล้ว เราสามารถที่จะสร้างเวคเตอร์อีกตัวหนึ่งที่ตั้งฉากกับเวกเตอร์ทั้งสองได้จาก http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image009.jpgซึ่งแกน b นี้เราจะเรียกว่า binormal axis

ทั้งหมดที่ได้กล่าวมาทั้งสามข้อข้างบนนี้สามารถแสดงในรูปต่อไปนี้

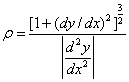


**การหารัศมีความโค้ง**

สำหรับเส้นโค้งทั่วไปที่เกิดการเคลื่อนที่ในระนาบ จะมีสมการของเส้นโค้งเป็น

http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image012.gif

เราจะได้รัศมีความโค้งของเส้นโค้งที่ตำแหน่งใดๆ บนเส้นนั้นเป็น



สรุปข้อสำคัญของการเคลื่อนที่ตามแนวเส้นโค้งโดยใช้ระบบพิกัดแนวตั้งฉากและแนวเส้นสัมผัส มีดังนี้

**การเลือกระบบพิกัด**

การเลือกระบบพิกัดแบบนี้ เราจะใช้ในกรณีที่เราทราบเส้นทางการเคลื่อนที่ที่แน่นอน (ทราบ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image015.jpg) เพื่อที่เราจะสามารถสร้างเวกเตอร์หนึ่งหน่วย (unit vector) ตามแนวตั้งฉากกับเส้นทางการเคลื่อนที่ (normal component) http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image016.gifและเวกเตอร์หนึ่งหน่วยตามแนวเส้นสัมผัส (tangential component) http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image017.gifที่จุดบนเส้นโค้ง เทียบต่อจุดศูนย์กลางความโค้งของเส้นโค้งแสดงการเคลื่อนที่ ในขณะเวลาที่กำหนด

เวกเตอร์ตามแนวเส้นสัมผัส จะต้องสัมผัสกับเส้นโค้งการเคลื่อนที่และทิศทางบวกจะชี้ไปในทิศทางการเคลื่อนที่

เวกเตอร์ตามแนวตั้งฉาก จะตั้งฉากกับเวกเตอร์ตามแนวเส้นสัมผัส และทิศทางบวกจะชี้ไปในทิศทางเข้าสู่จุดศูนย์กลางความโค้งของเส้นทางการ เคลื่อนที่

การใช้ระบบแกนแบบนี้จะมีข้อได้เปรียบในการศึกษาเพื่อกาค่าความเร็วและความ เร่งของอนุภาคที่จุดนั้น เพราะเราสามารถที่จะหาค่าความเร็วและความเร่งตามแนวเส้นสัมผัสและแนวตั้งฉาก ได้ตามสมการ

http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image018.gifhttp://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image019.gif

http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image020.gif

http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image022.jpgหรือ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image024.jpg

http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image025.gif

**ความเร็ว**

ความเร็วของอนุภาคจะต้องอยู่ในแนวเส้นสัมผัสกับเส้นโค้งเสมอ

ขนาดของความเร็วหรือ speed สามารถที่จะหาได้จากการหาอนุพันธ์ของระยะตามแนวเส้นโค้ง http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image026.gif

**ความเร่งตามแนวเส้นสัมผัส**

ความเร่งตาม แนวเส้นสัมผัสจะได้มาจากอัตราการเปลี่ยนแปลงของขนาดของความเร็วเทียบต่อเวลา โดยทิศทางของความเร่งนี้จะไปในแนวเดียวกันกับทิศทางของบวก s ถ้าหากว่าอนุภาคมีความเร็วเพิ่มขี้น และจะมีค่าเป็นลบถ้าหากว่าถ้าวัตถุเคลื่อนที่ไปตามเส้นโค้งแล้วมีความเร็วลดลง

ความสัมพันธ์ระหว่างความเร่งตามแนวเส้นสัมผัส http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image002_0000.jpgกับความเร็ว http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image003.gifและการเคลื่อนที่ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image004.gifจะใช้สมการเดียวกันกับการเคลื่อนที่เชิงเส้น

http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image029.jpg, http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image031.jpg

http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image032.jpgสำหรับในกรณีที่ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image033.jpgมีค่าคงที่ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image035.jpgเราสามารถที่จะอินทิเกรท สมการข้างบน เพื่อให้ได้

http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image036.gif

http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image037.gif

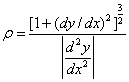
http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image038.gif

**ความเร่งตามแนวตั้งฉาก**

ความ เร่งตามแนวตั้งฉากมีค่ามาจากผลของการเปลี่ยนแปลงทิศทางของความเร็วของอนุภาค เทียบต่อเวลา ซึ่งความเร่งนี้จะมีทิศทางพุ่งเข้าสู่จุดศูนย์กลางความโค้งของเส้นโค้งการ เคลื่อนที่ของอนุภาคเสมอ นั่นคือจะมีทิศทางเดียวกับ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image001_0001.gif

ขนาดของความเร่งในแนวเส้นตั้งฉาก หาได้จาก http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image040.jpg

เมื่อเส้นโค้งมีสมการ http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image042.jpgค่ารัศมีความโค้ง http://www.rmutphysics.com/charud/scibook/dynamic1/index3_2_clip_image043.gifที่จุดใดบนเส้นโค้งนั้นหาได้จากสมการ



**2.3.4 กฎการเคลื่อนที่ของนิวตัน**

ไอแซก นิวตัน (Isaac Newton) นักฟิสิกส์ นักคณิตศาสตร์ นักดาราศาสตร์ ชาวอังกฤษเป็นผู้มีชื่อเสียงที่สุดในประวัติศาสตร์อังกฤษ นิวตันเกิดที่วูลส์ธอร์พแมน เนอร์ลิงคอนเชียร์ อังกฤษ ในปี ค.ศ. 2019 หนังสือชื่อ PhilosophiæNaturalis Principia Mathematica (เรียกกันโดยทั่วไปว่า Principiareble) เป็นรากฐานกฎกติกาพื้นฐานเกี่ยวกับการเคลื่อนที่ของวัตถุภายใต้แรงที่กระทำ (กฎว่าด้วยการเคลื่อนที่ 3 ข้อของนิวตัน) และทฤษฎีความโน้มถ่วงที่อธิบายว่าแรงซึ่งดึงดูดให้ผลแอปเปิลจากต้นตกสู่พื้น เป็นแรงชนิดเดียวกับที่ควบคุมการโคจรรอบโลกของพระอาทิตย์ นิวตันได้ศึกษาการเคลื่อนที่ของวัตถุและได้เสนอกฎการเคลื่อนที่สามข้อ กฎการเคลื่อนที่ทั้งสามข้อได้นำเสนออยู่ในหนังสือ Principia กฎการเคลื่อนที่ของนิวตัน (Newton’s laws of motion) สำหรับวัตถุ เป็นกฎกายภาพ (physical laws) ซึ่งเป็นกฎที่เกี่ยวกับพฤติกรรมของสสารที่เป็นจริงอยู่เสมออย่างไม่เปลี่ยน แปลง โดยเราไม่สามารถจะควบคุม ดัดแปลง หรือแก้ไขกฎแห่งความจริงได้

**กฎการเคลื่อนที่ข้อ 1 ของนิวตัน**

กฎการเคลื่อนที่ข้อ 1 ของนิวตัน (Newton’s first law of motion) หรือกฎของความเฉื่อย กล่าวว่า“วัตถุจะรักษาสภาวะอยู่นิ่งหรือสภาวะเคลื่อนที่อย่างสม่ำเสมอในแนวเส้นตรง นอกจากมีแรงลัพธ์ซึ่งมีค่าไม่เป็นศูนย์มากระทำ” คือ ถ้าวัตถุอยู่นิ่งก็ยังคงอยู่นิ่งเหมือนเดิม และถ้าวัตถุเกิดการเคลื่อนที่ก็จะเคลื่อนที่เป็นเส้นตรง ความเร็วคงที่ หรือความเร่งจะเป็นศูนย์ ซึ่งกรณีแรกจะเรียกว่า วัตถุอยู่ในสภาวะสมดุลสถิต (static equilibrium) และอีกกรณีหลังจะเรียกว่า วัตถุอยู่ในสภาวะสมดุลจลน์ (kinetic equilibrium) [7]มีสมการเป็นดังนี้ ∑F=0 Fคือ แรงลัพธ์ทั้งหมดที่กระทำกับวัตถุ

**กฎการเคลื่อนที่ข้อ 2 ของนิวตัน**

กฎการเคลื่อนที่ข้อ 2 ของนิวตัน (Newton’s second law of motion) บางทีเรียกว่า กฎความเร่ง กล่าวว่า“ความเร่งของอนุภาคเป็นปฏิภาคโดยตรงกับแรงลัพธ์ที่กระทำต่ออนุภาค โดยมีทิศทางเดียวกันและเป็นปฏิภาคผกผันกับมวลของอนุภาค” ดังนั้น อัตราส่วนของแรงกับความเร่งจะเป็นค่าคงที่ ซึ่งตรงกับมวลของวัตถุ เขียนเป็นความสัมพันธ์ได้ดังนี้ ∑F = ma

F คือ แรงลัพธ์ที่กระทำกับวัตถุ มีหน่วยเป็นนิวตัน (N)

m คือ มวลของวัตถุ มีหน่วยเป็นกิโลกรัม (kg)

a คือ ความเร่งมีหน่วยเป็นเมตรต่อวินาที2 (m/s2)

ถ้าแรงลัพธ์ (F) กระทำกับวัตถุอันหนึ่ง จะทำให้วัตถุมีความเร่ง (a) ในทิศทางเดียวกันกับทิศทางของแรง ซึ่งแรงลัพธ์ (F) ที่กระทำกับวัตถุ จะเท่ากับผลคูณระหว่างมวล (m) และความเร่ง (a) ของวัตถุ จะสรุปได้ว่า “แรงลัพธ์คงที่ที่กระทำกับวัตถุ ซึ่งมีมวลคงที่ วัตถุนั้นจะมีความเร่งคงที่ในทิศทางของแรงที่กระทำนั้น”

**กฎการเคลื่อนที่ข้อ 3 ของนิวตัน**

กฎการเคลื่อนที่ข้อ 3 ของนิวตัน (Newton’s third law of motion) กล่าวว่า“ทุกแรงกิริยา (action) ย่อมมีแรงปฏิกิริยา (reaction) ซึ่งมีขนาดเท่ากัน แต่มีทิศตรงข้ามกันเสมอ” กฎข้อนี้เรียกว่า กฎของกิริยาและปฏิกิริยา (Law of action and reaction) แรงกิริยาและแรงปฏิกิริยา หมายถึง แรงกระทำและแรงกระทำตอบ โดยเป็นแรงซึ่งกระทำต่อมวลที่ต่างกันและเกิดขึ้นพร้อมกันเป็นคู่เสมอ โดยที่มวลอาจไม่สัมผัสกันและถือว่าแรงหนึ่งแรงใดเป็นแรงกิริยาและแรงปฏิกิริยาก็ได้ เขียนเป็นความสัมพันธ์ได้ดังนี้ FA = -FR

**2.3.5 แรงเสียดทาน**

|  |  |
| --- | --- |
|  | แรงเสียดทาน (frictional force) คือ แรงที่ต้านการเคลื่อนที่ของวัตถุซึ่งเกิดขึ้นระหว่างผิววัตถุกับพื้นที่  สัมผัสและมีทิศตรงกันข้ามกับทิศการเคลื่อนที่ของวัตถุเสมอ แรงเสียดทานแบ่งเป็น 2 ชนิด คือ  1. แรงเสียดทานสถิตเป็นแรงเสียดทานที่เกิดขึ้นในขณะที่วัตถุอยู่นิ่งจนถึงเริ่มต้นเคลื่อนที่ มีความสัมพันธ์     ตามสมการ  http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/math7.png  http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/fs.gif= แรงเสียดทานสถิต หน่วยเป็นนิวตัน  http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/n.gif= แรงปฏิกิริยาในแนวตั้งฉากกับผิวสัมผัส หน่วยเป็นนิวตัน  http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/us.gif เรียกว่า สัมประสิทธิ์ความเสียดทานสถิต ไม่มีหน่วย  2. แรงเสียดทานจลน์เป็นแรงเสียดทานขณะวัตถุกำลังเคลื่อนที่ด้วยความเร็วคงตัว ซึ่งจะมีค่าน้อยกว่าแรง      เสียดทานสถิต มีความสัมพันธ์ตามสมการ  http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/math8.png  http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/fk.gif= แรงเสียดทานจลน์ หน่วยเป็นนิวตัน http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/n.gif= แรงปฏิกิริยาในแนวตั้งฉากกับผิวสัมผัส หน่วยเป็นนิวตัน http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/uk.gif เรียกว่า สัมประสิทธิ์ความเสียดทานจลน์ ไม่มีหน่วย **กราฟแรงเสียดทาน**  http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/pic_con3.png  เมื่อออกแรง F กระทำกับวัตถุ วัตถุยังคงอยู่นิ่งไม่ขยับเพราะมีแรงเสียดทานสถิต http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/fs.gifต้านอยู่  แต่เมื่อออกแรงมากขึ้นเรื่อยๆจนกระทั่งแรง F มีค่าเท่ากับ http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/fsmax.gif วัตถุจะอยู่ในสภาพกำลังจะเคลื่อนที่ซึ่งเป็นจุดที่มีแรงเสียดทานสูงสุด (ที่ตำแหน่งA) และเมื่อเราออกแรงมากขึ้นวัตถุจะเคลื่อนที่โดยมีแรงเสียดจลน์ http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/fk.gifเกิดขึ้น แต่จะมีค่าน้อยกว่า http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/fsmax.gif  **ลักษณะของแรงเสียดทาน**  1. ไม่ขึ้นกับจำนวนพื้นที่ผิวสัมผัส  2. ไม่ขึ้นกับความเร็วที่วัตถุเคลื่อนที่ และมีทิศตรงข้ามการเคลื่อนที่ของวัตถุ  3. ขึ้นอยู่กับแรงที่วัตถุกดพื้นในแนวตั้งฉาก หรือ แรงปฏิกิริยาของ พื้นในแนวตั้งฉาก  4. ขึ้นกับพื้นผิวสัมผัส เช่น ขรุขระ หรือเรียบ 5. สำหรับผิวสัมผัสคู่หนึ่ง http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/us.gifมีค่ามากกว่า http://www.cpn1.go.th/media/thonburi/lesson/02_ForceMassNewtonlaw/images/uk.gifเสมอ |

**2.3.6 กฎการอนุรักษ์พลังงาน**

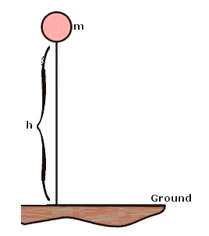
วัตถุที่กำลังเคลื่อนที่โดยมีอัตราเร็ว จะเรียกว่าวัตถุนั้นมี **พลังงานจลน์** (kinetic energy : Ek ) พลังงานจลน์ในวัตถุจะมากน้อยเท่าไรขึ้นกับมวล (m) และอัตราเร็ว (v) ของวัตถุ โดยมีความสัมพันธ์ดังนี้

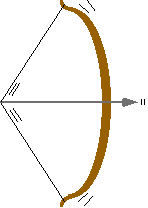
http://fieldtrip.ipst.ac.th/backend/images/resources/siampark/content_pic/11-1.png

พลังงานจลน์ของวัตถุจะเปลี่ยนไปถ้ามีงานที่เกิดจากแรงลัพธ์ซึ่งไม่เป็นศูนย์กระทำต่อวัตถุพลังงานในวัตถุที่ขึ้นกับตำแหน่งหรือสภาวะของวัตถุ เช่น พลังงานของวัตถุเมื่ออยู่ที่สูง พลังงานในสปริงที่ถูกยืดออกหรือหดสั้นกว่าปกติเหล่านี้ เรียกว่า **พลังงานศักย์** (potential energy) วัตถุที่มีพลังงานศักย์ พร้อมที่จะทำงานต่อไปได้ ถ้าเราออกแรงเท่ากับน้ำหนักวัตถุ(mg) ยกวัตถุมวล m ขึ้นไปสูงจากระดับเดิม (ซึ่งอาจถือว่าเป็นระดับอ้างอิง) เป็นระยะ h ถ้าปล่อยมือ วัตถุจะตกลงมาด้วยแรงดึงดูดของโลก (mg)  วัตถุขณะอยู่ที่สูงจากระดับอ้างอิงนี้จะมีพลังงานศักย์ที่เรียกว่า **พลังงานศักย์โน้มถ่วง** (gravitational potential energy : Ep ) โดยพลังงานศักย์โน้มถ่วงของวัตถุหาได้จากสมการ

http://fieldtrip.ipst.ac.th/backend/images/resources/siampark/content_pic/11-2.png

     โดย h  =  ความสูงจากระดับอ้างอิงซึ่งถือว่ามีพลังงานศักย์เป็นศูนย์ และการทำงานโดยออกแรงทำให้วัตถุมีพลังงานศักย์เพิ่มขึ้นนั้น ไม่ขึ้นกับเส้นทางการเคลื่อนที่ แต่ขึ้นกับระดับความสูงเพียงอย่างเดียว



[](http://fieldtrip.ipst.ac.th/backend/images/resources/siampark/content_pic/11-5.gif)

ส่วนการดึงสปริงให้ ยืดออกหรือหดจากตำแหน่งสมดุลจะทำให้เกิดพลังงานศักย์สะสมอยู่ในสปริงที่ สามารถดึงหรือผลักวัตถุที่ติดกับสปริงให้เคลื่อนที่ไปได้ พลังงานศักย์ที่สะสมอยู่ในสปริงนี้ เรียกว่าพลังงานศักย์ยืดหยุ่น(elastic potential energy)

ขณะวัตถุตกจากที่สูงลงมา แต่ละขณะวัตถุมีทั้งพลังงานจลน์และพลังงานศักย์ ผลรวมของพลังงานจลน์และพลังงานศักย์เรียกว่า **พลังงานกลรวมของวัตถุ** เมื่อมีแต่แรงโน้มถ่วงโดยไม่มีแรงเสียดทานหรือแรงอื่นมากระทำต่อวัตถุแล้ว **พลังงานกลรวมของวัตถุ ณ ตำแหน่งใด ๆ จะมีค่าคงตัวเสมอ** ซึ่งเป็น **กฎการอนุรักษ์พลังงานกล** ถ้าปล่อยวัตถุจากจุดหยุดนิ่งจากที่สูง วัตถุจะมีพลังงานศักย์โน้มถ่วงแต่ไม่มีพลังงานจลน์ และขณะวัตถุกำลังตกลงมาด้วยแรงโน้มถ่วง พลังงานศักย์โน้มถ่วงของวัตถุจะลดลงเท่ากับพลังงานจลน์ของวัตถุที่เพิ่มขึ้น ในทางกลับกัน ถ้าโยนวัตถุขึ้นจากพื้นด้วยความเร็ว ขณะวัตถุเคลื่อนที่สูงขึ้น พลังงานงานจลน์ของวัตถุจะลดลงเท่ากับพลังงานศักย์โน้มถ่วงที่เพิ่มขึ้น



     ในสถานการณ์จริง ส่วนมากผลรวมของพลังงานจลน์และพลังงานศักย์มีค่าไม่คงตัว เพราะมีแรงเสียดทานไปต้านการเคลื่อนที่ของวัตถุ งานของแรงเสียดทานจะทำให้พลังงานกลของระบบส่วนหนึ่งเปลี่ยนไปเป็นพลังงาน อื่นเช่นความร้อนและเสียง แต่เมื่อรวมพลังงานส่วนนี้เข้ากับพลังงานกลแล้ว พลังงานรวมจะมีค่าคงตัว ซึ่งเป็นไปตาม **กฎการอนุรักษ์พลังงาน**(law of conservation of energy) ที่กล่าวว่า **พลังงานรวมของระบบจะไม่สูญหาย แต่เปลี่ยนจากพลังงานหนึ่งไปเป็นอีกพลังงานหนึ่ง**

**2.3.7 การชนแบบไม่ยืดหยุ่น**

1. **การชนแบบไม่ยืดหยุ่น (Inelastic Collision)** คือ การชนแบบสูญเสียพลังงานจลน์ไปบางส่วน พลังงานที่สูญเสียไปบางส่วนอาจจะเปลี่ยนไปเป็นเสียง, แสง, เปลี่ยนรูปทรง การชนโดยทั่วไปจะเป็นการชนแบบนี้
2. **การชนแบบไม่ยืดหยุ่นโดยสมบูรณ์ (Completely Inelastic Collision)** คือ การชนแบบสูญเสียพลังงานจลน์ไปมากที่สุด โดยภายหลังการชนวัตถุจะติดกันไป

## การคำนวณการชนแบบไม่ยืดหยุ่นและการชนแบบไม่ยืดหยุ่นโดยสมบูรณ์

1. กฎการอนุรักษ์โมเมนตัม





สำหรับการชนแบบไม่ยืดหยุ่นโดยสมบูรณ์ จะได้ว่า



1. ผลรวมของพลังงานจลน์ก่อนชน > ผลรวมของพลังงานจลน์หลังชน เสมอ





**บทที่ 3 รายละเอียดของการพัฒนา**

**3.1 ซอฟต์แวร์ที่ใช้ในการพัฒนา (Software)**

**3.1.1 Library Three.js**

เป็น javascript library ใช้ในการสร้างภาพสามมิติ (3D Computer Graphic Simulation) บน web browser โดย Three.js ใช้ WebGL

**3.1.2 Library TextGeometry.js**

เป็นส่วนต่อขยายของ Three.js ใช้ในการสร้างวัตถุที่เป็นข้อความ เป็นตัวอักษรสามมิติ ที่หล่นมาจากก้อนเมฆ

**3.1.3 Library FontUtils.js**

เป็น class สำหรับการดำเนินการที่เกี่ยวกับ text ใน Three.js ซึ่งทำให้สามารถใช้ font ที่ต้องการได้

**3.1.4 Library Physi.js**

เป็น javascript physic engine ที่ถูกสร้างคลุม ammo.js ซึ่งช่วยให้การเรียกใช้ได้ง่ายขึ้น และช่วยให้ใช้ interface แบบ Three.js ง่ายขึ้น และสามารถทำการจำลอง physic simulation ได้โดยใช้งานเป็น Thread ผ่าน Web worker

**3.1.5 Library Physijs\_Worker.js**

เป็นตัวที่คำนวณ physic simulation ที่เกิดจากการแบ่งงานเป็นหลายๆ ชิ้น

**3.1.6 Library Ammo.js**

เป็นกลไกทางฟิสิกส์ที่เราใช้ในการเล่น Canvas และนอกจากนี้ยังใช้ในการทำให้ text หล่น ซึ่งถูกใช้งานโดย Physi.js

**3.2 ขั้นตอนการดำเนินการ**

1. ทำการหาตัวอย่างการสร้างภาพเคลื่อนไหว 3 มิติ แบบ real-time ด้วย WebGL ในเว็บไซต์ต่างๆ เพื่อใช้เป็นส่วนหนึ่งในการคิดโครงงานใหม่
2. เสนอหัวข้อโครงงานที่คิดเสร็จแล้วแด่อาจารย์ประจำวิชา
3. เริ่มวางแผนการทำโครงงาน โดยการจัดตารางเวลาให้เหมาะสม
4. หาข้อมูลความรู้เพื่อทำโครงงาน
5. ทำส่วนของฉากและตัวอักษร
6. ใส่ shading แบบต่างๆ ให้ตัวอักษร
7. เพิ่มส่วนฟิสิกส์เข้าสู่ฉากและตัวอักษร โดยใช้ Physi.js ทำให้วัตถุสามารถเคลื่อนที่และมีปฏิกิริยาต่อกันตามกฎของฟิสิกส์
8. เพิ่มแถบเครื่องมือสำหรับปรับค่าต่าง ๆ ของ shading และแรงโน้มถ่วง โดยใช้ dat.gui.js
9. ทำการตกแต่งโดยการใส่เมฆเข้าสู่ฉาก
10. ทำการบันทึกภาพเคลื่อนไหวของหน้าจอเพื่อแสดงผลงานที่เสร็จสิ้นแล้ว
11. ทำรายงานและทำการเขียนข้อมูลลงแผ่นซีดี
12. เสร็จสิ้นการทำงาน

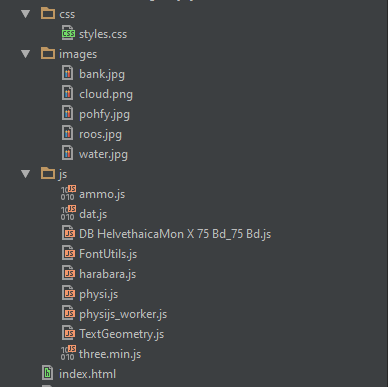
**3.3 การแบ่งหน้าที่การทำงาน**

|  |  |
| --- | --- |
| **ชื่อสมาชิก** | **หน้าที่ที่รับผิดชอบ** |
| นางสาวกรชนก จิรเวศยกุล | - ส่วนของฟังก์ชันที่ใช้สร้าง Control Panel  - ส่วนที่ใช้สร้าง Shading แบบ Toon Shading |
| นายพัทธนันท์ อัครพันธุ์ธัช | - ส่วนของฟังก์ชันที่ใช้สร้างฉาก พื้น แสง กล้อง และเมฆ  - ส่วนของฟังก์ชันที่ใช้สร้างตัวอักษร  - ส่วนที่ใช้สร้าง Shading แบบ Phong Shading |
| นายเวศวรุศ งามดำรงเกียรติ | - คิดโครงสร้างโดยรวมของซอฟต์แวร์  - ส่วนประมวลผลเพื่อสร้างภาพเคลื่อนไหวอย่างต่อเนื่อง |

**3.4 โครงสร้างของซอฟต์แวร์**

**3.4.1 โครงสร้างโดยรวมของซอฟต์แวร์**

โครงสร้างของซอฟต์แวร์ในชั้นแรกสุด ประกอบไปด้วยส่วนประกอบต่างๆ ดังภาพ

1. โฟลเดอร์ ‘css’ ใช้ในการเก็บรูปแบบการแสดงผลของหน้าเว็บ

2. โฟลเดอร์ ‘images’ ใช้ในการเก็บรูปภาพที่ต้องใช้ในการแสดงผลภาพสามมิติ

3. โฟลเดอร์ ‘js’ ใช้ในการเก็บไฟล์ JavaScript ซึ่งเป็น libraries ต่างๆ ที่สำคัญ ซึ่งต้องใช้ในการเขียนเว็บไซต์เพื่อสร้างภาพเคลื่อนไหวสามมิติพร้อมการจำลองการเคลื่อนที่ทางฟิสิกส์

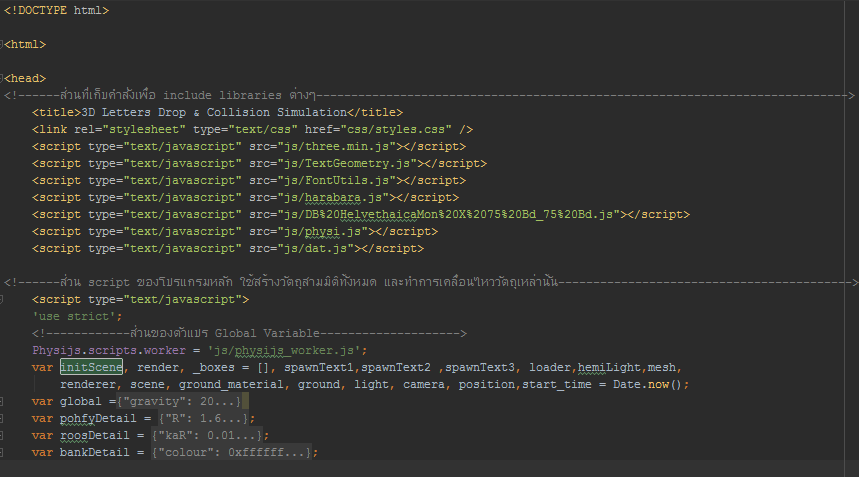
4. ไฟล์ index.html เป็นไฟล์หลักที่ใช้ในการสร้างวัตถุสามมิติทั้งหมด และเป็นส่วนที่ใส่ shading แบบต่างๆ ให้กับวัตถุ รวมถึงเป็นตัวกลางที่เรียกใช้ libraries ต่างๆ เพื่อใช้ในการจำลองทางฟิสิกส์ทั้งหมดด้วย

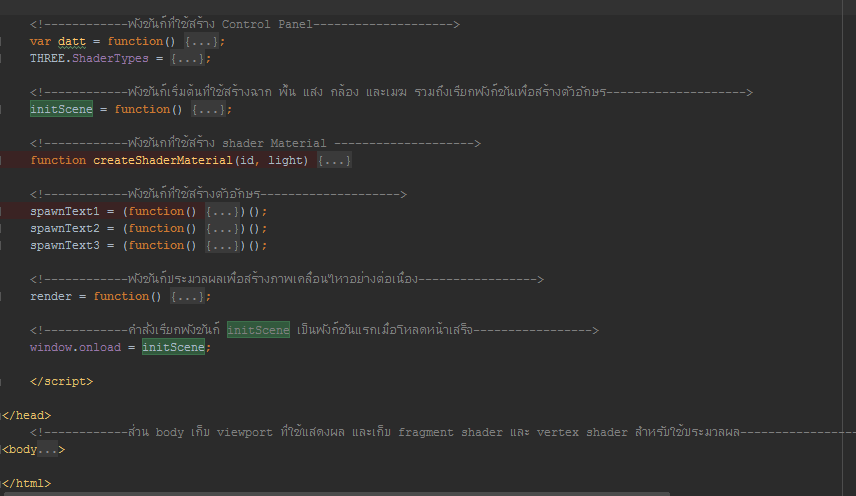
**3.4.2 โครงสร้างไฟล์หลัก ‘index.html’**

ประกอบไปด้วย 5 ส่วนหลักๆ คือ

1. ส่วนที่เก็บคำสั่งเพื่อ include libraries ต่างๆ
2. ส่วน script ของโปรแกรมหลัก ใช้สร้างวัตถุสามมิติทั้งหมด และทำการเคลื่อนไหววัตถุเหล่านั้น
   1. ฟังก์ชันที่ใช้สร้าง Control Panel
   2. ฟังก์ชันเริ่มต้นที่ใช้สร้างฉาก พื้น แสง กล้อง และเมฆ รวมถึงเรียกฟังก์ชันเพื่อสร้างตัวอักษร
   3. ฟังก์ชันที่ใช้สร้าง shader Material
   4. ฟังก์ชันที่ใช้สร้างตัวอักษร
   5. ฟังก์ชันประมวลผลเพื่อสร้างภาพเคลื่อนไหวอย่างต่อเนื่อง
   6. คำสั่งเรียกฟังก์ชัน initScene เป็นฟังก์ชันแรกเมื่อโหลดหน้าเสร็จ
3. ส่วน body เก็บ viewport ที่ใช้แสดงผล และเก็บ fragment shader และ vertex shader สำหรับใช้ประมวลผล

สามารถดูภาพโครงสร้างจริงในไฟล์ index.html ได้ในหน้าต่อไป





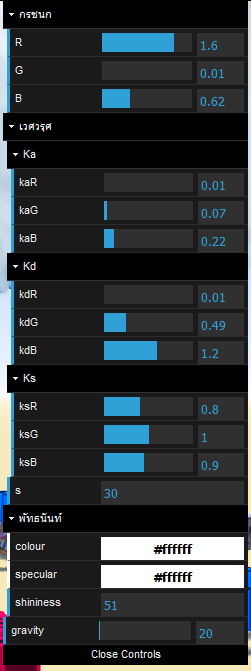
**3.4.3 ส่วนของฟังก์ชันที่ใช้สร้าง Control Panel**



datt เป็นฟังก์ชันที่ใช้สำหรับสร้าง Control Panel ซึ่งใช้ library dat.GUI.js ในการสร้าง โดยมีโครงสร้างต่างๆ ดังต่อไปนี้

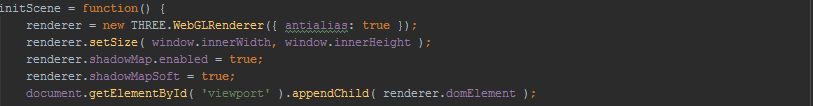
1. poh เป็นตัวแปรสำหรับสร้างส่วนควบคุมของตัวอักษร ‘กรชนก’ โดยมีองค์ประกอบคือ
   1. ส่วนที่ใช้ควบคุมสีค่า R (Red)
   2. ส่วนที่ใช้ควบคุมสีค่า G (Green)
   3. ส่วนที่ใช้ควบคุมสีค่า B (Blue)
2. roos เป็นตัวแปรสำหรับสร้างส่วนควบคุมของตัวอักษร ‘เวศวรุศ’ โดยมีองค์ประกอบคือ
   1. ส่วนที่ใช้ควมคุมสีของ ambient (Ka) แบ่งเป็น R (Red) G (Green) B (Blue)
   2. ส่วนที่ใช้ควมคุมสีของ diffuse (Kd) แบ่งเป็น R (Red) G (Green) B (Blue)
   3. ส่วนที่ใช้ควมคุมสีของ specular (Ks) แบ่งเป็น R (Red) G (Green) B (Blue)
   4. ส่วนที่ใช้ควมคุมระดับของ specular (ใช้อักษรย่อคือ s )
3. bank เป็นส่วนที่ใช้สำหรับสร้างส่วนควบคุมของตัวอักษร ‘พัทธนันท์’ โดยมีองค์ประกอบคือ
   1. colour ใช้ควบคุมสีของวัตถุ
   2. specular ใช้ควบคุมสีของแสงสะท้อน specular ของวัตถุ
   3. shininess ใช้ควบคุมระดับความเปล่งประกายของวัตถุ
4. gravity เป็นส่วนควบคุมค่าแรงโน้มถ่วงในการจำลองทางฟิสิกส์ มีค่าได้ตั้งแต่ 0-2000

ฟังก์ชัน datt จะถูกสั่งทำงานในฟังก์ชัน initscene ซึ่งจะให้ผลดังภาพต่อไปนี้

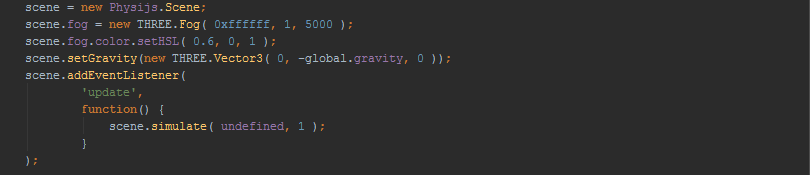


**3.4.4 ส่วนของฟังก์ชันที่ใช้สร้างฉาก พื้น แสง กล้อง และเมฆ**

1. คำสั่งสำหรับสร้าง Renderer โดยมีการปรับให้มีเงาตกกระทบและทำให้เงาดูนุ่มสวยงาม

****

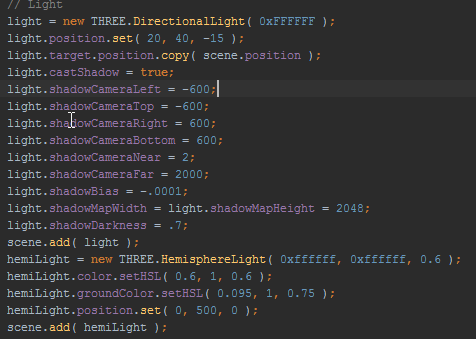
1. ส่วนของฉาก ถูกสร้างด้วย Physijs เพื่อที่จะสามารถใช้เป็นฉากในการ simulate การเคลื่อนที่แบบฟิสิกส์ด้วย Physi.js ได้ต่อไป โดยมีการกำหนดค่า gravity ให้กับฉากและมีการใส่ fog ให้กับฉากด้วย

****

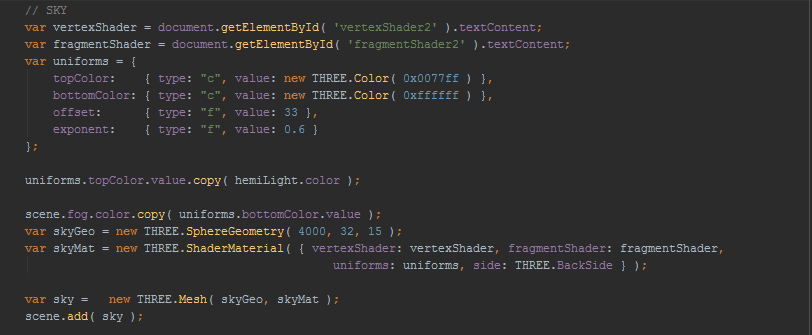
1. เพิ่มกล้องให้กับฉาก โดยจะเป็นมุมมองที่ผู้ใช้เห็น

****

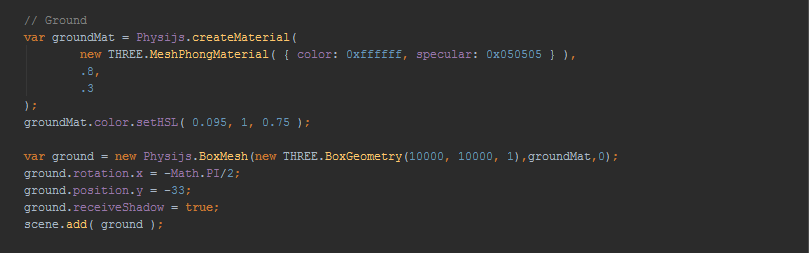
1. เพิ่มแสงให้กับฉาก

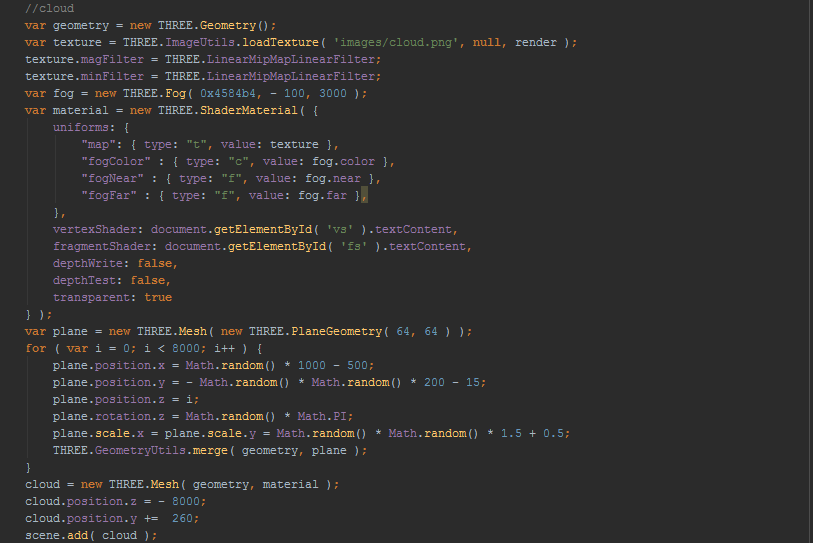


1. ใส่ท้องฟ้าให้กับฉาก



1. สร้างพื้นให้กับฉาก โดยมีการกำหนดค่าความเสียดทานของพื้น = 0.8 และค่าสัมประสิทธิ์การคืนตัว =0.3

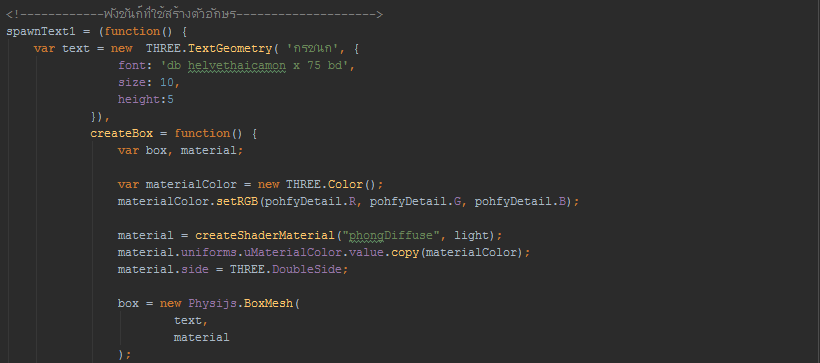


1. สร้างก้อนเมฆให้กับฉากจำนวน 8000 ก้อน
2. สั่ง spawnText ทั้งสาม เพื่อเริ่มสร้างตัวอักษร ที่เป็นชื่อของสมาชิกทั้งสามคน จากนั้นสั่งฟังก์ชัน datt(); เพื่อสร้าง control panel จากนั้นสั่ง requestAnimationFrame( render ) เพื่อเริ่มการ animate โดยใช้ฟังก์ชัน render



**3.4.5 ส่วนของฟังก์ชันที่ใช้สร้างตัวอักษร**

1. สร้างตัวอักษรด้วย Three.Geometry โดยใช้ font DB HelvethaicaMon x 75 Bd และสร้าง wrapper เป็น BoxMesh กล่าวคือ ให้เปรียบเสมือนว่า Text นั้นมีโครงสร้างทางฟิสิกส์ที่เปรียบเสมือนกล่อง และจะมีการเคลื่อนที่เหมือนกล่องสี่เหลี่ยมทุกประการ

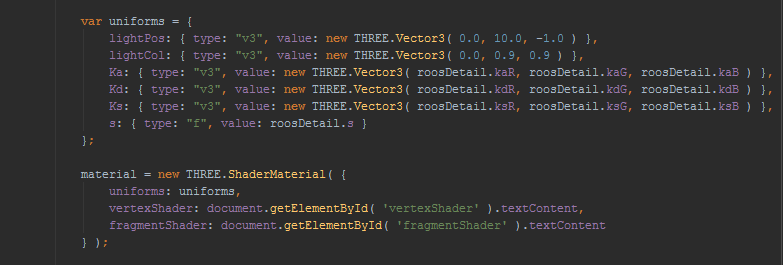
****

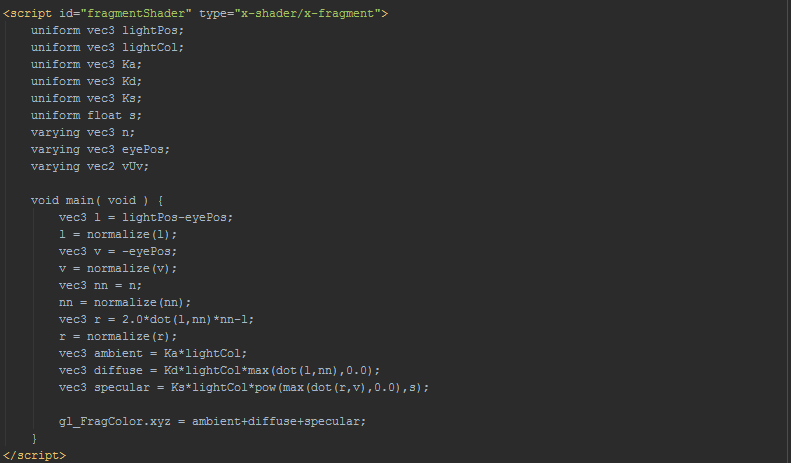
1. ทำการสุ่มตำแหน่งเริ่มต้นของตัวอักษร ว่าจะให้ตกจากจุดไหนบนฟ้า จากนั้นระบุให้ตัวอักษรสามารถ cast shadow ได้ แล้วจึงเพิ่มตัวอักษรเข้าไปในฉาก จะเห็นว่ามีการ setTimeout ทั้งนี้เพื่อให้มีการสร้างตัวอักษรในทุก ๆ 1 วินาที

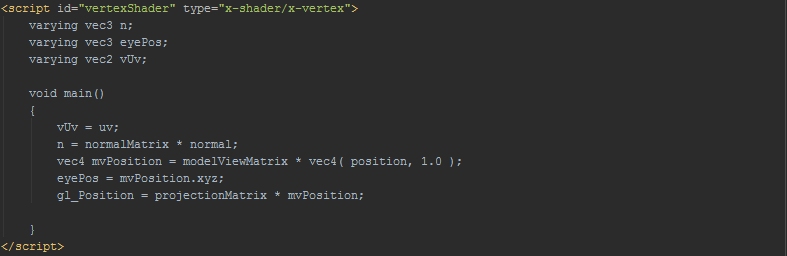


**3.4.6 ส่วนที่ใช้สร้าง Shading แบบต่างๆ ให้กับ Material**

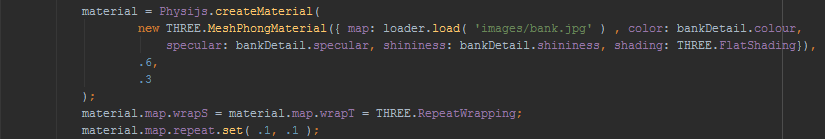
1. Phong Shading เพื่อใช้สำหรับใส่ Shading ให้กับตัวอักษร “เวศวรุศ” มีการปรับค่า ka kd และ ks ด้วย control panel ซึ่งจะเห็นได้เมื่อเปิด page นี้ในเว็บบราวเซอร์







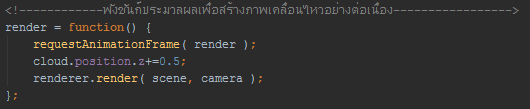
1. Phong Shading เพื่อใช้สำหรับใส่ Shading ให้กับตัวอักษร “พัทธนันท์” ใช้ MeshPhongMaterial จาก Library three.js ซึ่งมีความสามารถในการ map texture ให้กับวัตถุด้วย



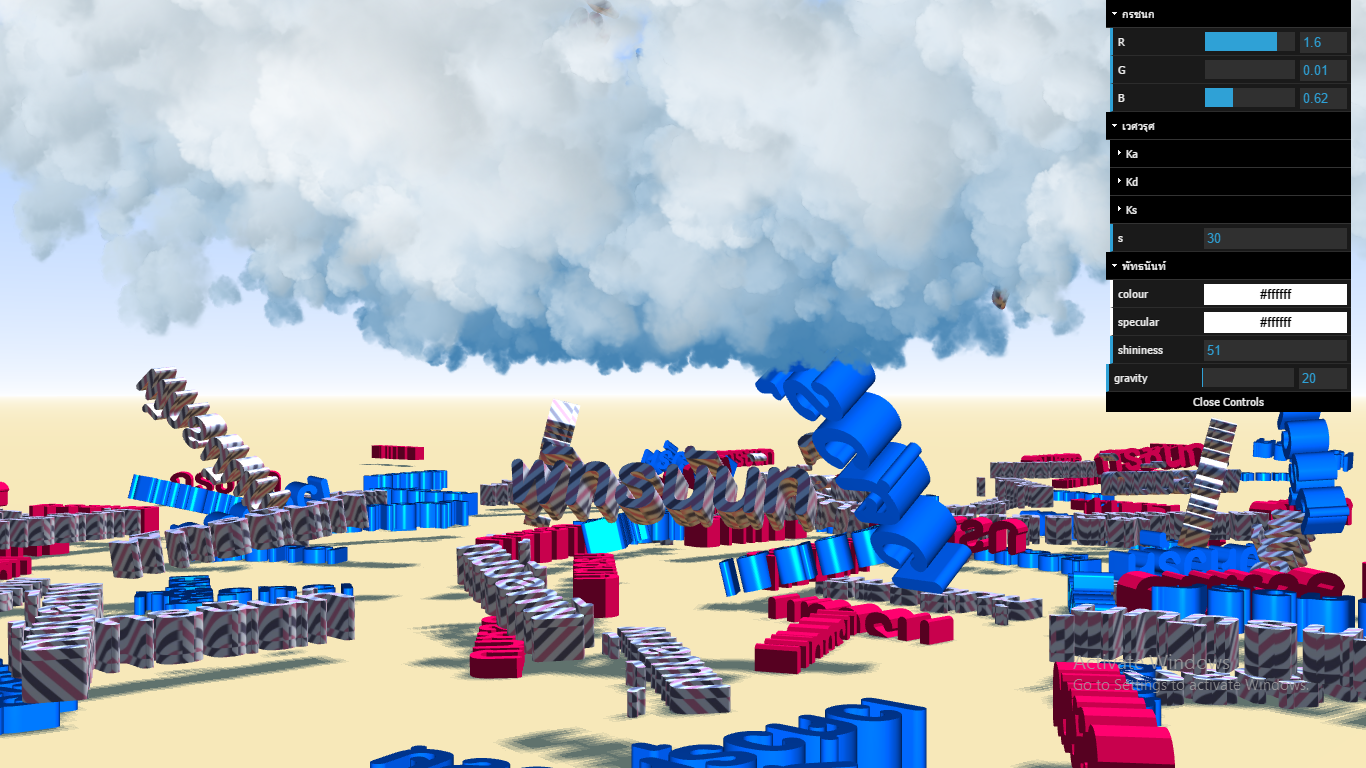
1. Toon Shading เพื่อใช้สำหรับใส่ Shading ให้กับตัวอักษร “กรชนก” มีขั้นตอนการคำนวณดังนี้
   1. คำนวณทิศทางของแสงที่ตกมากระทบมายังตัวอักษร จากนั้นจึงทำการ normalize แสงออกมาก
   2. นำค่า Normal ทำการ normalize
   3. จากนั้นคำนวณหา diffuse จากสูตร max(dot(N,L),0.0) โดยที่ N คือ Normal ที่ normalize แล้ว และ L คือ Directional Light ที่ normalize แล้วเช่นกัน
   4. ทำการเช็คค่า Diffuse ที่ได้ว่าค่าอยู่ในช่วงใด เนื่องจาดเรามี Threshold 2 ระดับ สมมติว่าเป็น T1 และ T2 หาก Diffuse > T1 ให้ Diffuse เป็นค่าหนึ่ง แต่หาก Diffuse มากกว่า T2 แต่น้อยกว่า T1 ให้เป็นอีกค่าหนึ่ง และหาก diffuse มีค่าต่ำกว่า T1 และ T2 ก็ให้เป็นอีกค่าหนึ่ง ซึ่งจะทำให้เรามีการไล่ระดับสีเพียงแค่ 3 สี
   5. เมื่อได้ค่า diffuse ใหม่เป็นที่เรียบร้อยแล้วจึงคำนวณค่าสีที่ปรากฏบนตัวอักษร ณ ตำแหน่งนั้นๆ จากสูตร Kd\*Material Color\*Direction Light Color\*Diffuse

**3.4.7 ส่วนประมวลผลเพื่อสร้างภาพเคลื่อนไหวอย่างต่อเนื่อง**

มีการ render ภาพความเคลื่อนไหวตาม libraries สำหรับวัตถุทุกชิ้นในฉาก แต่มีก้อนเมฆที่เขียนการเคลื่อนที่ด้วยตนเอง



**บทที่ 4 ผลการดำเนินงาน**



*ภาพที่ได้จากการ simulation แบ่งออกได้เป็นสามส่วน คือ ส่วน Foreground ส่วน Background และส่วน Control Panel โดย*

1. **Background** จะเป็นพื้นสีเนื้อ และท้องฟ้าสีฟ้าอ่อน
2. **Foreground** จะมีกลุ่มก้อนเมฆสีฟ้าเข้มลอยอยู่ โดยกลุ่มก้อนเมฆเหล่านี้จะมีลักษณะเคลื่อนที่เข้ามาใกล้ กล้องมากขึ้นเรื่อย ๆ และมีตัวอักษรแบบต่าง ๆ ตกลงมาจากด้านบนเรื่อย ๆ เช่นกัน ซึ่งระหว่างการตกลงมานี้ก็อาจมีการชนระหว่างตัวอักษรด้วยกันเอง หรือชนกับพื้นเมื่อตกถึงพื้น ตามหลักการทางฟิสิกส์
3. **Control Panel** อยู่บริเวณมุมบน ขวาของหน้า สามารถปรับค่าตัวแปรต่าง ๆ ที่มีผลต่อการ simulation ได้

**บทที่ 5 ปัญหาและอุปสรรค**

1. **การตก และการชนกันของตัวอักษรสามมิติมีความแม่นยำต่ำ**

ทั้งนี้เป็นเพราะเราใส่ Physics ให้ ตัวอักษรเปรียบเสมือนกล่องสี่เหลี่ยม ไม่ได้สนใจส่วนโค้ง ส่วนเว้า ร่องลึก ตามสภาพความเป็นจริงของตัวอักษร ทำให้การชนกันมีความถูกต้องแม่นยำค่อนข้างต่ำ

1. **ไม่สามารถรองรับตัวอักษรจำนวนมากเกินไปได้**

เนื่องจากคณะผู้จัดทำ ไม่ได้มีการลบ object ตัวอักษรออกจากฉากเลย ทำให้ตัวอักษรจะเพิ่มขึ้นเรื่อย ๆ และเมื่อถึงจุดหนึ่ง การ simulation จะเริ่มไม่ Real-time และเว็บบราวเซอร์จะค้างในที่สุด ทั้งนี้เป็นเพราะข้อจำกัดทางด้าน Hardware ของเครื่องคอมพิวเตอร์เอง และความสามารถในการเขียน WebGL ของคณะผู้จัดทำยังมีไม่เพียงพอด้วย

1. **การสะท้อนและเงายังไม่สมจริง**

ไม่มีเงาระหว่างวัตถุด้วยกันเอง และยังไม่มีการสะท้อนซึ่งกันและกันระหว่างวัตถุที่มีความมันวาวกับวัตถุชนิดอื่น

1. **การจำลองก้อนเมฆยังไม่สมบูรณ์**

จะเห็นได้ว่า ตัวอักษรสามมิติจะอยู่ข้างหลังก้อนเมฆเสมอ ไม่สามารถทะลุก้อนเมฆได้ ทั้งนี้เป็นเพราะการวางตำแหน่งก้อนเมฆยังไม่ถูกต้อง และมีโอกาสเป็นไปได้ที่การจำลองก้อนเมฆแบบนี้ไม่เหมาะกับงานชิ้นนี้

**บทที่ 6 แนวทางการพัฒนาต่อในอนาคต**

การจำลองวัตถุ 3 มิตินั้นมีความสำคัญมากขึ้นเรื่อย ๆ ไม่ว่าจะนำไปใช้ภาพยนตร์ หรือเป็นการจำลองข้อความคำพูดแบบ 3 มิติ เพื่อใช้ในโอกาสต่าง ๆ ก็ล้วนต้องอาศัยความถูกต้องแม่นยำทั้งสิ้น จึงอยากที่จะพัฒนาให้โปรแกรมของเรามีความถูกต้องแม่นยำตามหลักการทางฟิสิกส์มากขึ้น โดยอาจทำได้โดยจากเดิมที่จำลองให้ตัวอักษรทั้งชุดแทนด้วยกล่องสี่เหลี่ยมกล่องเดียว ก็เปลี่ยนเป็น compound object คือประกอบไปด้วยกล่องหลายๆ กล่องต่อกันให้เป็นรูปร่างของตัวอักษรที่ถูกต้อง และต้องแยกตัวอักษรแต่ละตัวออกจากกันด้วยเพื่อความสมบูรณ์ทางฟิสิกส์ นอกจากนี้ คณะผู้จัดทำเห็นว่าควรพัฒนาโปรแกรมให้มีฟีเจอร์พิเศษเพิ่มขึ้น เช่น ให้ผู้ใช้สามารถป้อนตัวอักษรที่ต้องการจะใช้ได้ ปรับรูปแบบตัวอักษรได้หลากหลายแบบ รวมไปถึงสามารถปรับค่าคงที่ต่าง ๆ ได้โดยไม่มีข้อจำกัด

**บทที่ 7 เอกสารอ้างอิง**

**หลักการและทฤษฎี**

**Phong shading**. เข้าถึงได้จาก : http://graphics.wikia.com/wiki/Phong\_shading

**Cel shading**. เข้าถึงได้จาก : https://en.wikipedia.org/wiki/Cel\_shading

**เวกเตอร์**. เข้าถึงได้จาก : https://www.google.co.th/url?sa=t&rct=j&q=&esrc=s&source= web&cd=7&ved=0ahUKEwiYm4SP-d7JAhUTI44KHe2MDr0QFgg0MAY&url=http% 3A%2F%2Fwww.neutron.rmutphysics.com%2Fphysicsboard%2Fforum%2Findex.php%3Faction%3Ddlattach%3Btopic%3D832.0%3Battach%3D3809&usg=AFQjCNHKhUPeXIfcFTcDi-G8V0fF4JU3Vg&sig2=WXpXhKPUIV0F3h\_Aa8vjoQ&cad=rja

**การเคลื่อนที่ในแนวสามมิติ**. เข้าถึงได้จาก : http://www.rmutphysics.com/charud/scibook/ dynamic1/index3\_2.html

**กฎการเคลื่อนที่ของนิวตัน.** เข้าถึงได้จาก : https://th.wikipedia.org/wiki/กฎการเคลื่อนที่ของนิวตัน

**แรงเสียดทาน.** เข้าถึงได้จาก : http://www.cpn1.go.th/media/thonburi/lesson/02\_ForceMas sNewtonlaw/content3.html

**กฎการอนุรักษ์พลังงาน.** เข้าถึงได้จาก : http://fieldtrip.ipst.ac.th/intro\_sub\_content.php? content\_id=18&content\_folder\_id=191

**กฎการอนุรักษ์โมเมนตัมของระบบ.** เข้าถึงได้จาก : https://www.google.co.th/url?sa=t &rct=j&q=&esrc=s&source=web&cd=3&ved=0ahUKEwi-7svvw97JAhVOBI4K HawGCyoQFggnMAI&url=http%3A%2F%2Fwww.mtm.in.th%2Facademic%2Fmyfile%2F200114154227\_1.doc&usg=AFQjCNGQF6PzSbd4Io7IoMns3bOOMfJLQQ&sig2=kIIXvTXDShTZ09-PcF0rcg&cad=rja

**Libraries ต่างๆ และวิธีการใช้งาน**

**three.js.** เข้าถึงได้จาก : threejs.org/

**physi.js,** **physijs\_worker.** เข้าถึงได้จาก : https://github.com/chandlerprall/Physijs

**ammo.js**. เข้าถึงได้จาก : https://github.com/kripken/ammo.js/

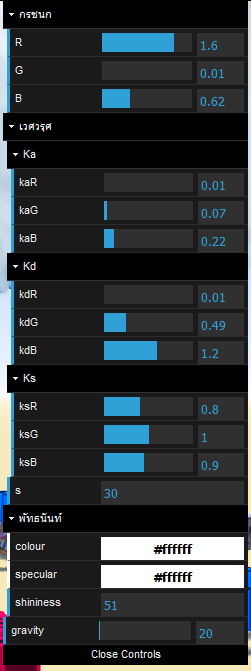
**งานที่นำมาดัดแปลง หรือคัดลอกบางส่วน เพื่อนำมาใช้ในโครงงาน**

**ท้องฟ้าและพื้น** จากhttp://threejs.org/examples/webgl\_lights\_hemisphere.html

**ก้อนเมฆ** (เฉพาะรูปก้อนเมฆและกระบวนการสร้าง การเคลื่อนที่ของก้อนเมฆคณะผู้จัดทำเขียนเอง) จาก http://mrdoob.com/lab/javascript/webgl/clouds/

**บทที่ 8 ภาคผนวก**

**วิธีการใช้งาน Control Pane**

****

เลื่อนเพื่อปรับค่าสีต่างๆ ของ Kd (diffuse) ของตัวอักษรเวศวรุศ

เลื่อนเพื่อปรับระดับ Gravity ของพื้น

คลิกเลือกสี specular ของตัวอักษรพัทธนันท์

ใส่เลขเพื่อปรับระดับ shininess ของตัวอักษรพัทธนันท์

คลิกเลือกสีตัวอักษรพัทธนันท์

เลื่อนเพื่อปรับค่าสีต่างๆ ของ Ks (specular) ของตัวอักษรเวศวรุศ

ใส่เลขเพื่อปรับระดับความ specular ของตัวอักษรเวศวรุศ

เลื่อนเพื่อปรับค่าสีต่างๆ ของ Ka (ambient) ของตัวอักษรเวศวรุศ

เลื่อนเพื่อปรับค่าสีเขียวของตัวอักษรกรชนก

เลื่อนเพื่อปรับค่าสีน้ำเงินของตัวอักษรกรชนก

เลื่อนเพื่อปรับค่าสีแดงของตัวอักษรกรชนก

**ลิงค์ไปยังวีดีทัศน์ของโครงงานใน Youtube.com**

https://www.youtube.com/watch?v=DjOggD4zt-o