

Chapter 16

Understand Auto Layout

iOS Screen Sizes



iPhone, 3G, 3GS,
iPod Touch Gen 1-4
320 x 480
(iOS 7 Not support)



iPad1,2, mini
768 x 1024
(9.7" / 7.9")



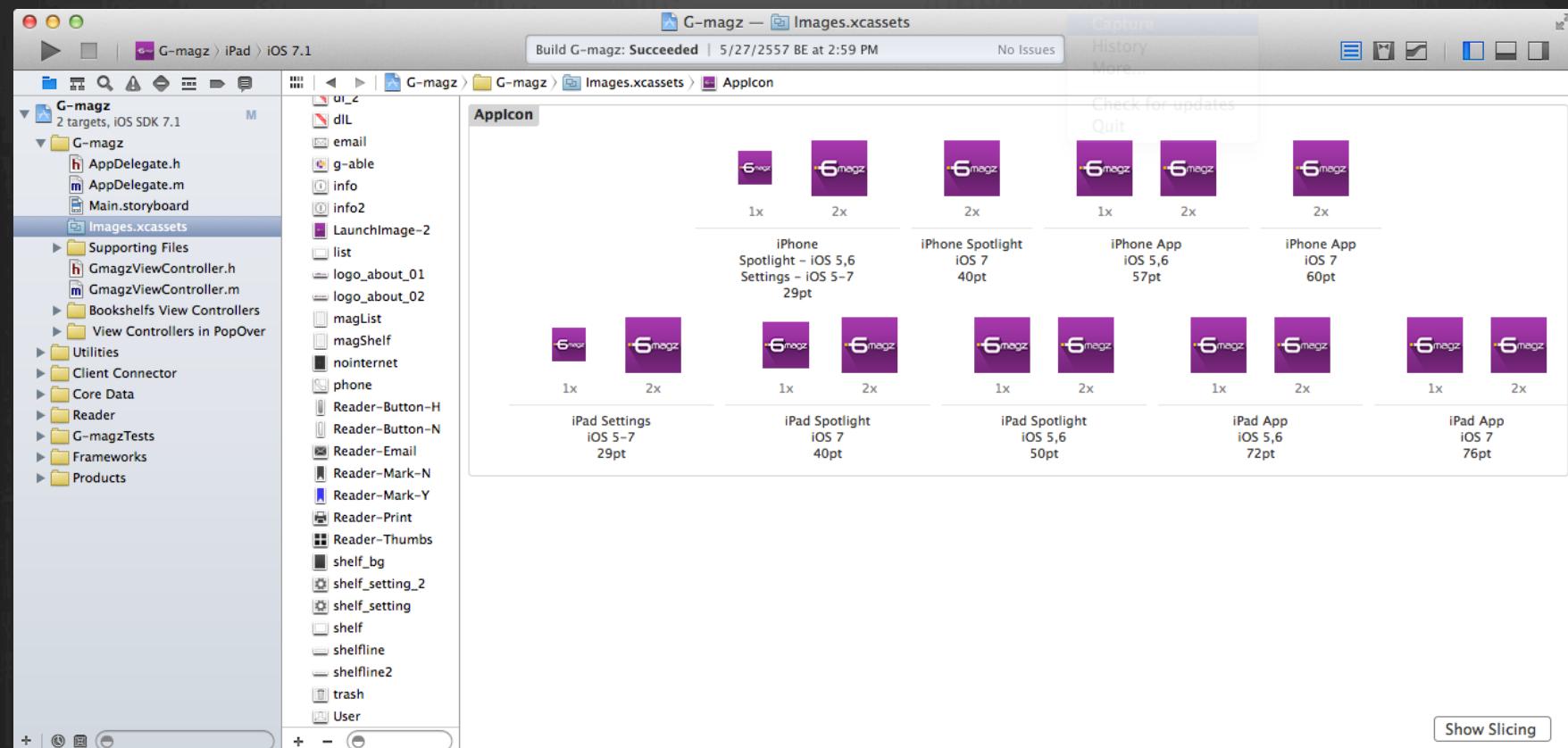
iPhone 4, 4s
640 x 960
(Retina 3.5")
still in the market



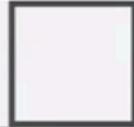
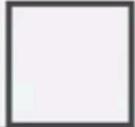
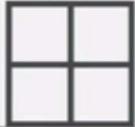
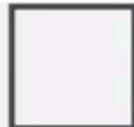
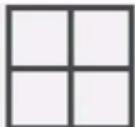
iPad 3,4, Air, mini, mini Retina
1536 x 2048
(Retina 9.7" / 7.9")

iPhone 5, 5s, 5c,
iPod Touch Gen 5
640 x 1136
(Retina 4")

Resources & Sizes



Points vs. Pixels in iOS

	Points	Pixels
Non Retina iPhone 1 point = 1 pixel		 320x480
iPhone 4 (Retina Screen) 1 point = 4 pixels		 640x960
iPhone 5 (Retina Screen) 1 point = 4 pixels		 640x1136

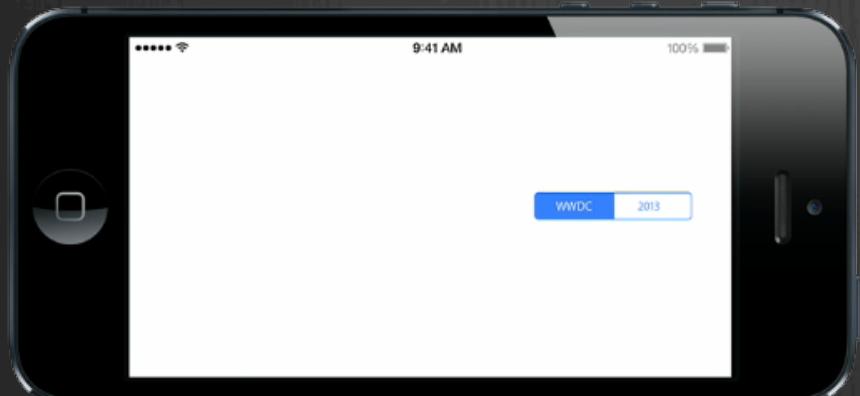
What is Auto Layout?

- ◉ Auto Layout เป็นระบบที่ใช้เพื่ออธิบาย Layout บน UI ของ iOS และ OS X โดยใช้ Constraint เป็นหลัก (Constraint-Based)
- ◉ โดย default นั้น Xcode จะใช้ตำแหน่ง x, y กำหนดตำแหน่งของ control บน view แต่ในตัวอย่าง ถ้าเราต้องการให้ปุ่มวางอยู่กลางจอชิดขวาเสมอไม่ว่า device จะอยู่ในแนวตั้งหรือแนวนอน การเขียน code เพื่อกำหนด x, y จะยุ่งยาก
- ◉ แนวทางการออกแบบ Auto Layout ก็คือ กำหนด Constraint ในลักษณะของคำอธิบาย เพื่อบอกว่า view นั้นสัมพันธ์ กับ view อื่นยังไง เช่น จากตัวอย่าง “button จะอยู่กลาง super view ในแนวตั้ง” และ “ชิดขอบขวาของ super view ในแนวนอน” เป็นต้น



Auto Layout Constraint

- ◉ เปรียบเทียบเหมือน Whiteboard และอธิบายด้วยภาษาที่เข้าใจง่าย
 - ◉ Control is centered vertically in its super view
 - ◉ Control is a fixed distance (padding) from the right of the super view
- ◉ เมื่อแปลงเป็น code
 - ◉ `Control.centerY =Superview.centerY`
 - ◉ `Control.right =Superview.right - <padding>`



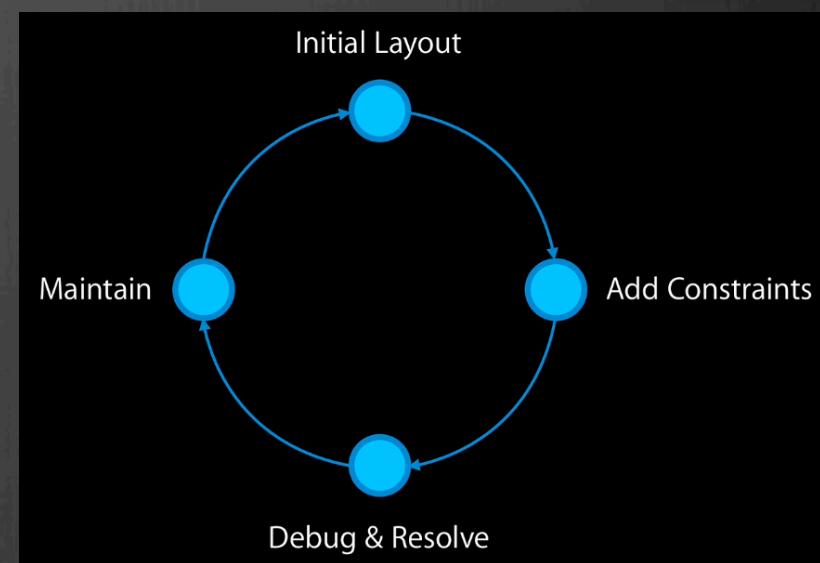
Describe the layout
with constraints, and
frames are calculated
automatically.

Why Auto Layout?

- Relational
- Dynamism
 - Metrics : iOS6 to iOS7, Multiple Screen Sizes, Rotate
 - Content : Localization
- Expressiveness : Relationship between view

Adopting Auto Layout

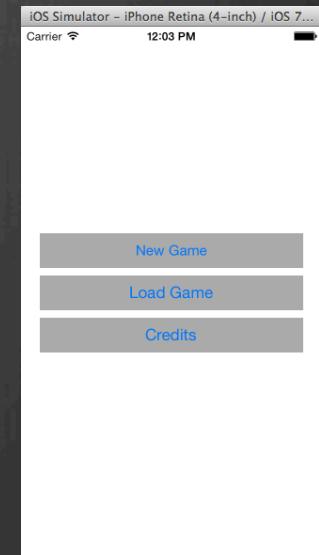
- ⦿ เราสามารถเพิ่ม Auto Layout ลงบน View ได้ 3 วิธี คือ
 - ⦿ ใช้ Control + Drag ระหว่าง view
 - ⦿ ใช้ Resolve Auto Layout Issue
 - ⦿ ใช้ Constraint property
- ⦿ เราสามารถ เพิ่ม Auto Layout ให้กับ view ที่ล่อนิดได้ไม่ต้องเพิ่มครั้งเดียว ทั้งหมด
- ⦿ ถ้าทำผิด เราสามารถลบ constraint ออก โดยไม่กระทบกับ view ที่เรา wrong ไว้ (หรือ undo ได้)



Lab 1/3 : Initial Layout

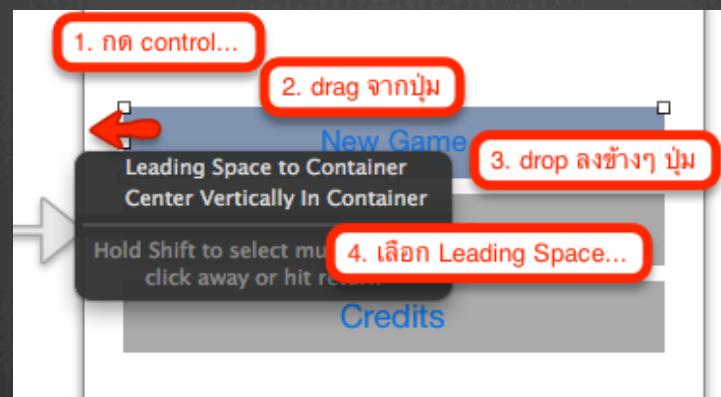
Add views, positioning, resizing (1/9)

1. จาก Xcode สร้าง project ใหม่โดยเลือก iOS > Application > Single View Application
2. ตั้งชื่อ project ว่า “AutoLayoutGame” และเลือก Devices เป็น iPhone
3. Click “Next” เลือก folder ที่จะ save project แล้ว click ปุ่ม “Create”
4. วางปุ่มลงบน view 3 ปุ่ม เปลี่ยน background color ของปุ่มเป็น “Light Gray Color” และ Align ตำแหน่ง ของปุ่มทั้ง 3 ให้อยู่กลางจอ
5. เปลี่ยน Title เป็น “New Game”, “Load Game”, และ “Credits”
6. Run โปรแกรมแล้วลองเปลี่ยนเป็นแนวอน จะเห็นว่า ตำแหน่งของปุ่มทั้ง 3 ไม่อยู่กลางจอ หรือลอง run ด้วย จอ 3.5” และ 4” จะเห็นว่า 3.5” ปุ่มไม่อยู่กลางจอ

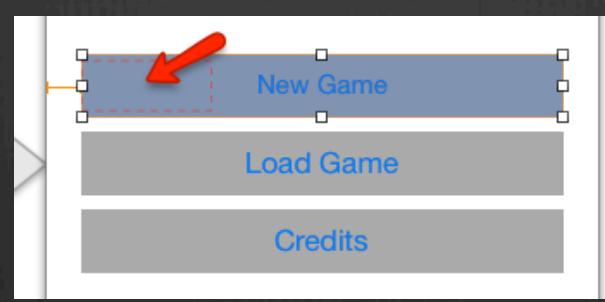


Task : Add Horizontal Space Constraint (2/9)

7. เพิ่ม constraint ให้กับปุ่ม “New Game” โดยกด control และ drag จากปุ่มไปยังพื้นที่ว่างระหว่าง “ปุ่ม” กับ “ขอบของ view ด้านซ้าย” จะมี popup เมนูขึ้นมา เลือก “Leading Space to Container”



สังเกตว่า Interface Builder จะแสดงกรอบสีเหลือง นั่นแสดงว่า Layout ยังไม่ครบ



Task : Add Horizontal Space Constraint (3/9)

8. เพิ่ม constraint “Horizontal Space” จากปุ่มไปยัง Super view โดยกด control ค้างไว้แล้ว drag จากปุ่มไปยังพื้นที่ว่างๆ ระหว่างปุ่มกับ view ด้านขวาเมื่อจะมีpopup เมนูขึ้นมา ให้เลือก “Trailing Space to Container”

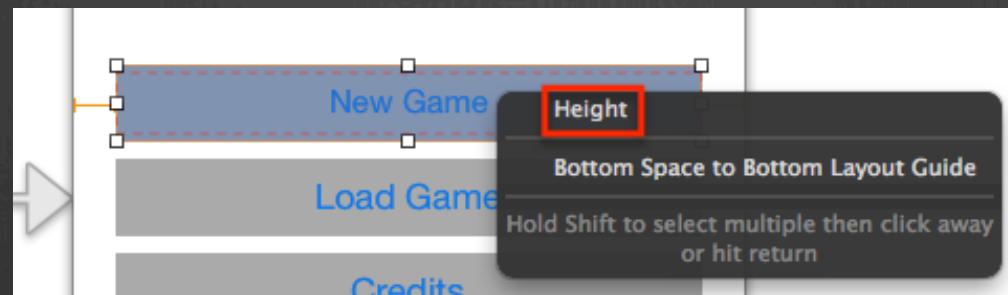


สังเกตุว่า กรอบสีเหลืองจะคลุมเกือบทั้งปุ่ม แสดงว่าขาด “Height Constraint”



Task : Add height constraint (4/9)

- เพิ่ม height constraint ให้กับปุ่ม “New Game” โดยกด control แล้ว drag จากปุ่มไปยังพื้นที่บนปุ่มเดิม จะมี popup เมนูแสดงขึ้นมา ให้เลือก “Height”

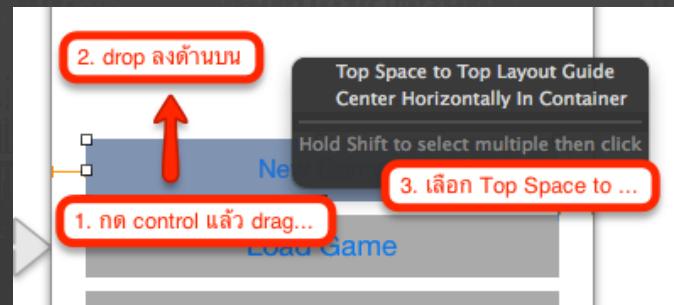


สังเกตว่า Interface Builder จะสร้าง constraint ขึ้นมาแต่ยังเป็นสีเหลืองอยู่ แสดงว่ายังไม่ครบ

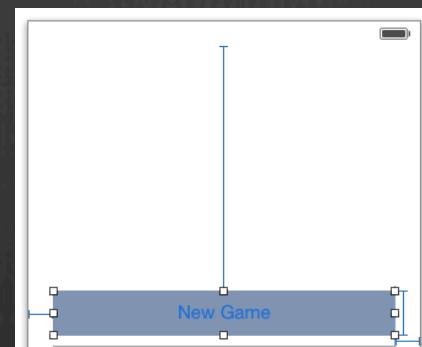


Task : Add Vertical constraint (5/9)

10. เพิ่ม vertical constraint ให้กับปุ่ม “New Game” โดยกด control แล้ว drag จากปุ่มไปยังพื้นที่ว่างด้านบนของปุ่ม จะมี popup เมนูแสดงขึ้นมา ให้เลือก “Top Space to Top Layout Guide”

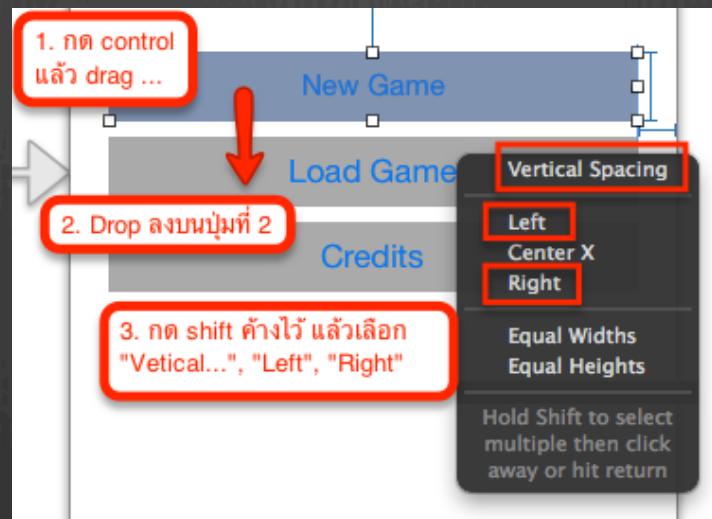


สังเกตุว่า ถ้า constraint ครบ เส้น Guide จะเป็นลีฟ้า



Task : Add Vertical Space constraint (6/9)

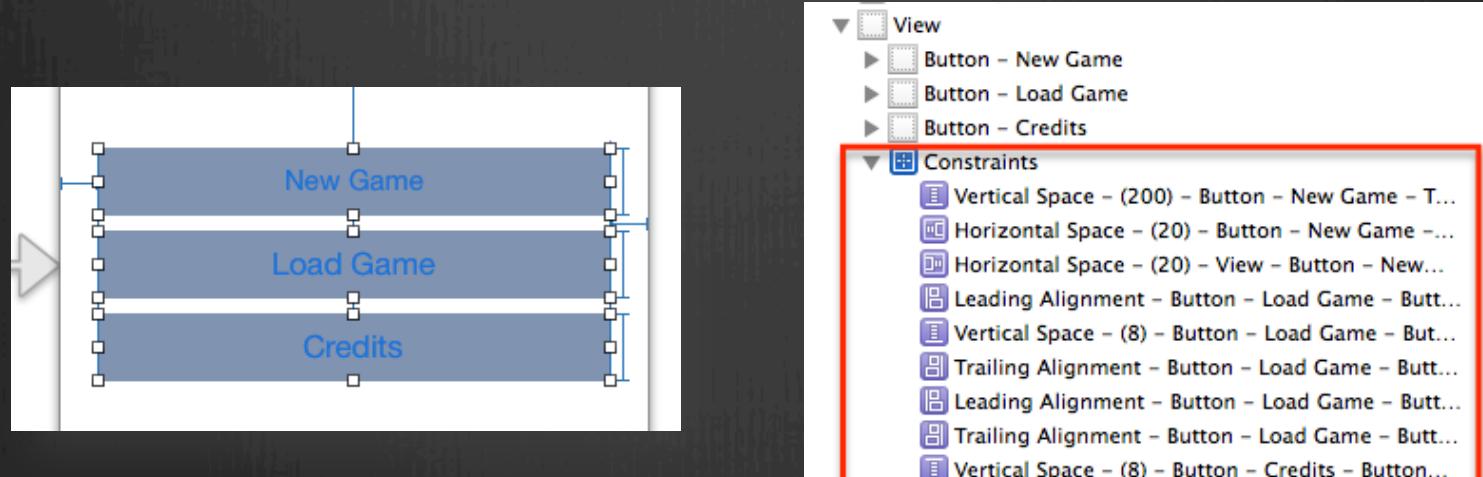
11. เพิ่ม constraint ระหว่างปุ่ม New Game ไปยัง Load Game โดยกด control ค้างไว้แล้ว drag จากปุ่ม New Game ไป drop บนปุ่ม Load Game (ปล่อยปุ่ม control) แล้วกด shift ค้างไว้แล้วเลือก “Vertical Spacing”, “Left” และ “Right” (จะทำทีละอันก็ได้)



Interface Builder จะทำการสร้าง constraint เพื่อบอกว่า กำหนดให้มี Space ระหว่างปุ่ม “Load Game” และขนาดของ Left และ Right ของปุ่ม Load Game เท่ากับ New Game เสมอ

Task : Complete constraints (7/9)

12. เพิ่ม height constraint ให้กับปุ่ม Load Game เหมือนข้อ 9.
13. เพิ่ม Space และ Size constraint ระหว่างปุ่ม Load Game ไปยังปุ่ม Credits เมื่อข้อ 11
14. สังเกตว่า ถ้าเลือกปุ่มทั้ง 3 ปุ่ม เราจะเห็น constraint ทั้งหมดเป็นสีฟ้า และถ้าดูใน Document outline จะเห็นว่าเกิด Interface Builder จะสร้าง object ของ constraint ขึ้นมาให้เรา



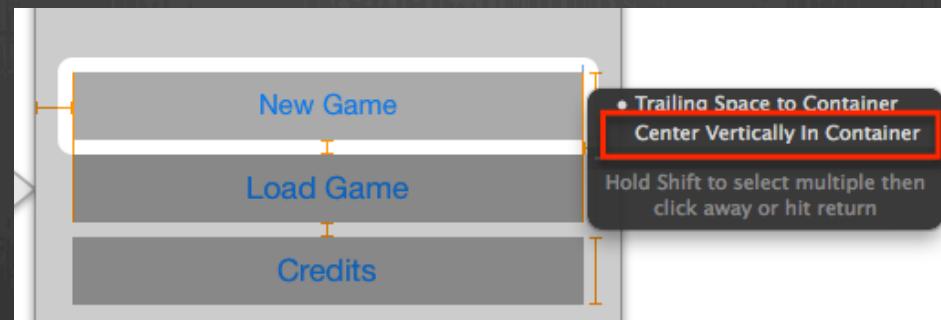
Task : Run & Test (8/9)

15. Run โปรแกรมแล้วทดลองเปลี่ยน Simulator เป็นแนวอน สังเกตว่าตำแหน่งของปุ่มยังเกาะอยู่กับขอบนิ่มได้อยู่กางจอ



Task : Fix Incorrect Constraint (9/9)

16. Click เลือกปุ่ม New Button และ click constraint ที่ยึดอยู่กับขอบบน แล้วลบออก โดยกดปุ่ม delete
17. กด control ค้างไว้แล้ว drag จากปุ่ม New Game ไปยังพื้นที่ว่างระหว่างปุ่มกับ view ทางด้านขวา หรือซ้ายของปุ่ม แล้วเลือกเมนู “Center Vertically in Container” ใน pop up เมนู

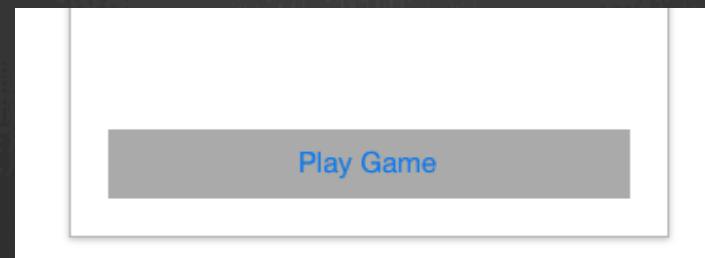


18. Run โปรแกรม และลองเปลี่ยน simulator เป็นแนวอน



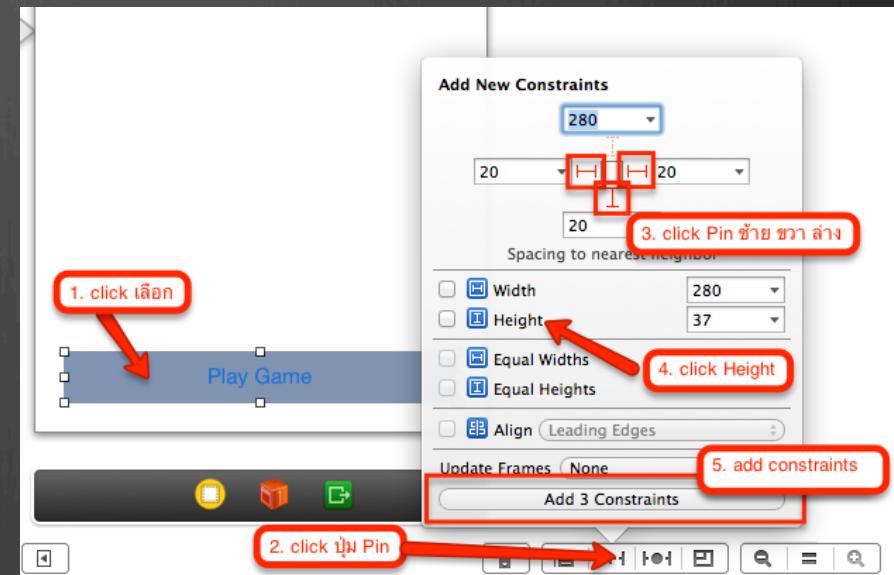
Lab 2/3 : Using Pin (1/9)

1. เปิดไฟล์ Images.xcassets และ click ขวาบนพื้นที่ว่าง ใต้ข้อความ Launch Image เลือกเมนู “Import...” และ Browse ไปที่ folder “../Resources/Day5 – Lab16/” เลือกรูปทั้ง 2 รูปแล้ว click ปุ่ม Open
2. เปิดไฟล์ Main.storyboard และ Embed view ใน Navigation control โดย click ที่ view และเลือกเมนู Editor > Embed In > Navigation Controller
3. สร้าง segue จากปุ่ม New Game ไปยัง View ใหม่ และกำหนด property “Identifier” ของ segue ว่า “segueNewGame”
4. เพิ่ม Button ลงบน View ใหม่ เปลี่ยน background color ของปุ่มเป็น “Light Gray Color” เปลี่ยน Title เป็น “Play Game” และวางตำแหน่งของปุ่มให้ชิดขอบล่างของจอ (เว้นช่องว่างตาม guideline ของ Interface Builder)



Task : Add Constraints by Pin (2/9)

5. Click ที่ปุ่ม Play Game แล้ว Click ที่ปุ่ม “Pin” ที่อยู่ด้านล่างของ Interface Builder แล้วเลือก Pin ตามรูป แล้ว click ปุ่ม “Add 4 Constraints”

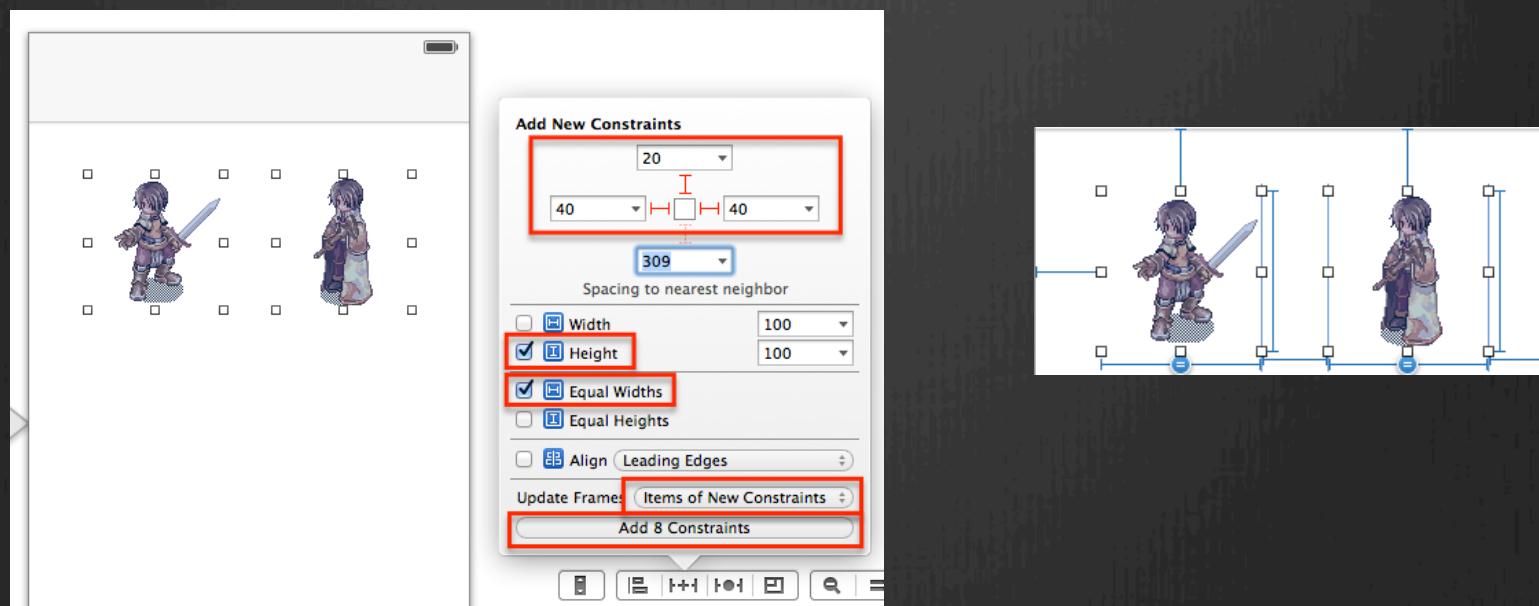


6. Run โปรแกรม สังเกตว่าเมื่อเราเปลี่ยน Simulator เป็นแนวอน ปุ่ม New Game จะหายของล่างเสมอ



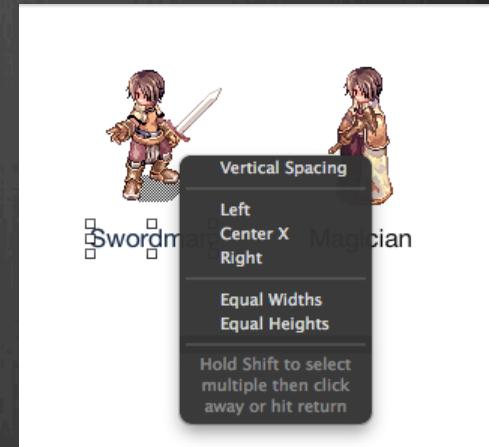
Task : Add Constraints by Pin (3/9)

7. เพิ่ม Button ลงบน View ใหม่ ลบ Title ออก และกำหนด property “Image” ของปุ่ม เป็น “Swordsman”
8. ทำซ้ำข้อ 7. แต่ใช้รูป “Magician”
9. เลือกรูปทั้ง 2 รูป และ click ที่ Pin เมนูบน Interface Builder จากนั้น กำหนดค่าต่างๆ ตามรูป และ click “Add 8 constraints” Interface Builder จะ add constraint ทั้งหมด ให้เรา



Task : Add Constraints Automatically (4/9)

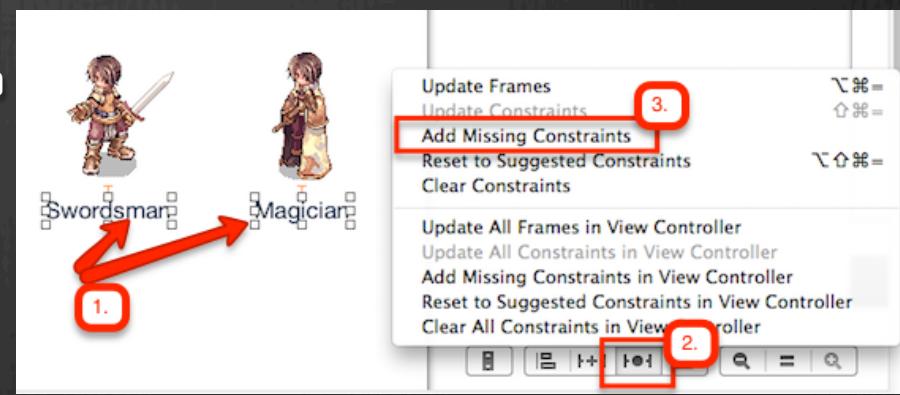
10. เพิ่ม Label ลงบน view 2 ตัว เปลี่ยน title เป็น “Swordsman” และ “Magician” และจัดตำแหน่งให้ label แต่ละอยู่ใต้ button ตรงกลางพอดี



11. สร้าง constraint ให้กับ Label Swordsman เพื่อให้ตำแหน่งของ label ออยู่ใต้ button และอยู่ตรงกลาง button เสมอ โดยกด control แล้ว drag จาก label ไปยัง button และเลือกเมนู “Vertical Spacing”

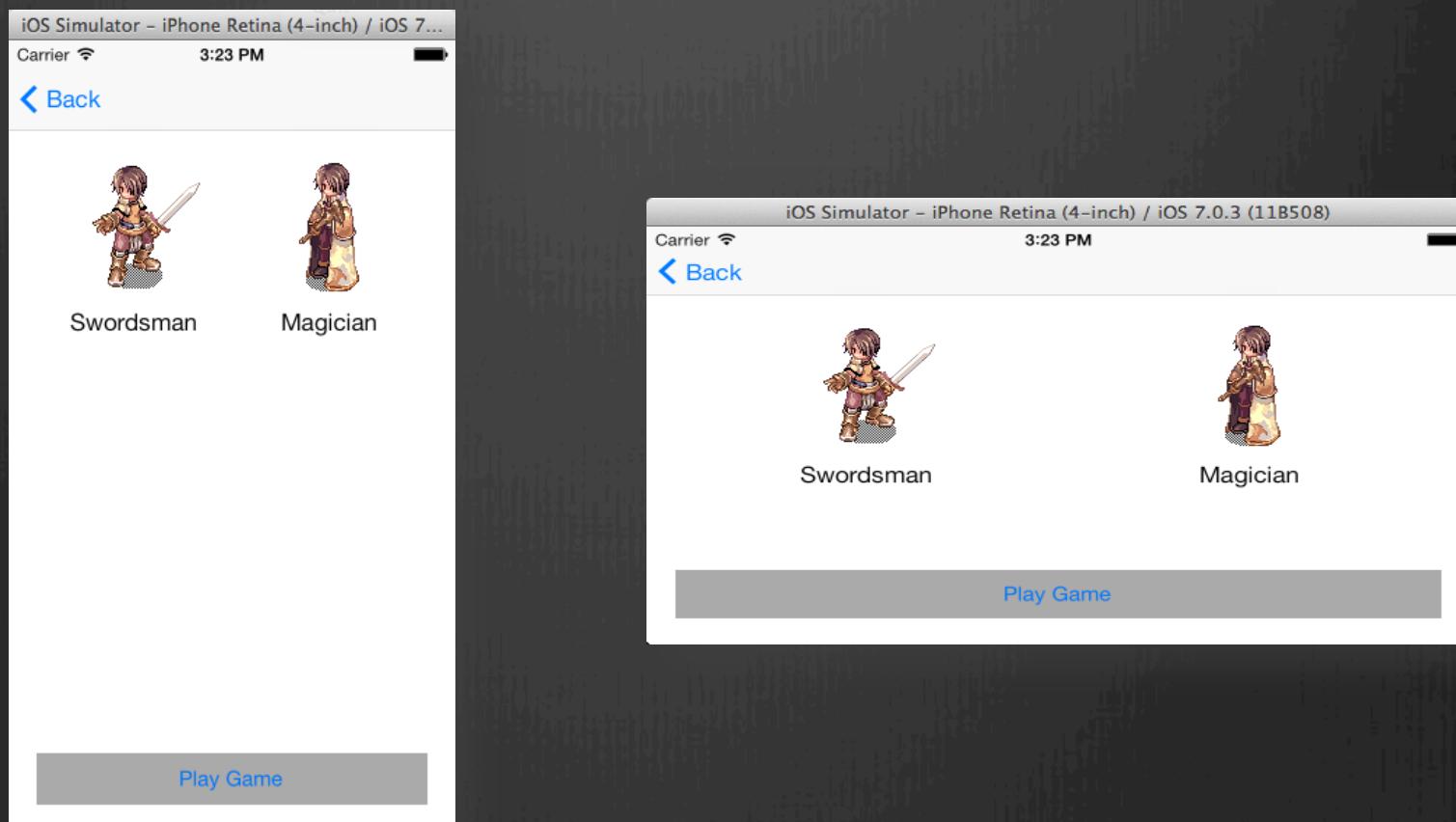
12. ทำซ้ำข้อ 11. สำหรับ label ที่ 2 เราจะได้ constraint สำหรับกำหนด space ระหว่างปุ่มกับ Label แต่ยังไม่มี constraint สำหรับกำหนดให้ label อยู่ตรงกลางปุ่มเสมอ

13. click เลือก label ทั้ง 2 ตัว และ click ที่ปุ่ม “Resolve Auto Layout Issues” และเลือกเมนู “Add Missing Constraints” Interface Builder จะเพิ่ม constraint ที่ขาดไปให้อัตโนมัติ โดยเดาจาก constraint ที่เราเพิ่มไว้ก่อนแล้ว



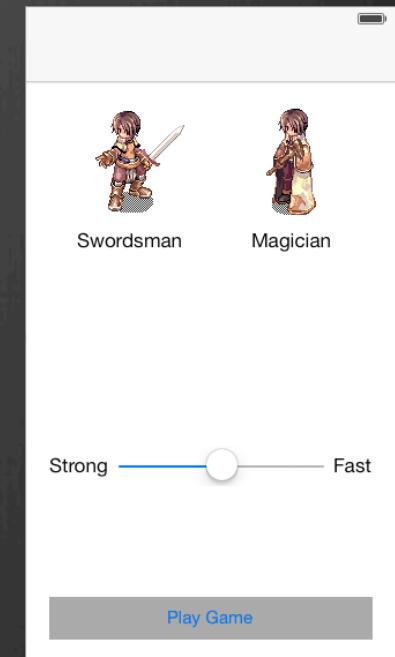
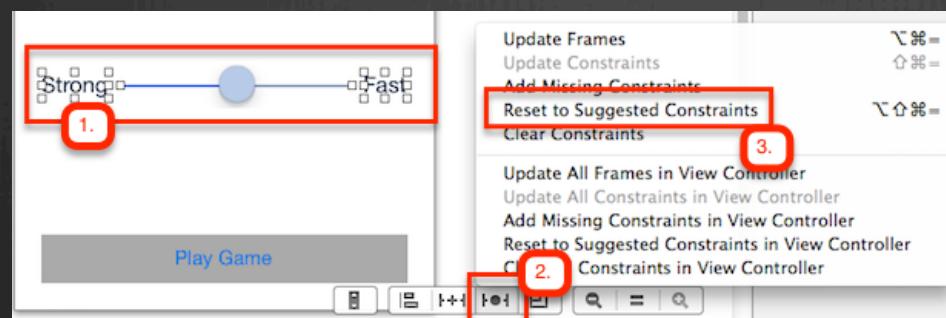
Task : Run & Test (5/9)

14. Run โปรแกรมแล้วทดลองเปลี่ยน Simulator เป็นแนวอน

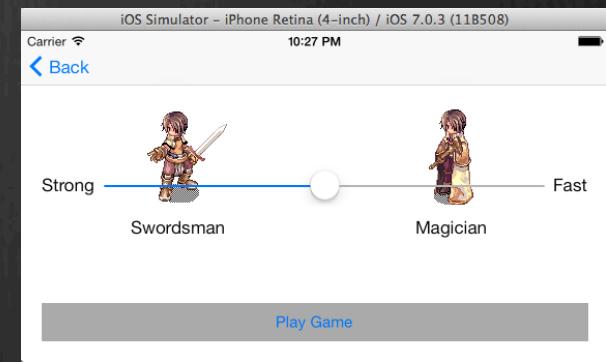


Task : Add more Views (6/9)

15. เพิ่ม Label และ Slide Bar ลงบน view ตามรูป โดยให้ view ทั้ง 3 ตัวอยู่ในระนาบแนวอนต์รอกัน และวางตำแหน่งค่อนลงมาด้านล่างของ view
16. เลือก view ทั้ง 3 ตัวแล้ว click ที่เมนู “Resolve Auto Layout Issues” แล้วเลือก “Reset to Suggested Constraints”

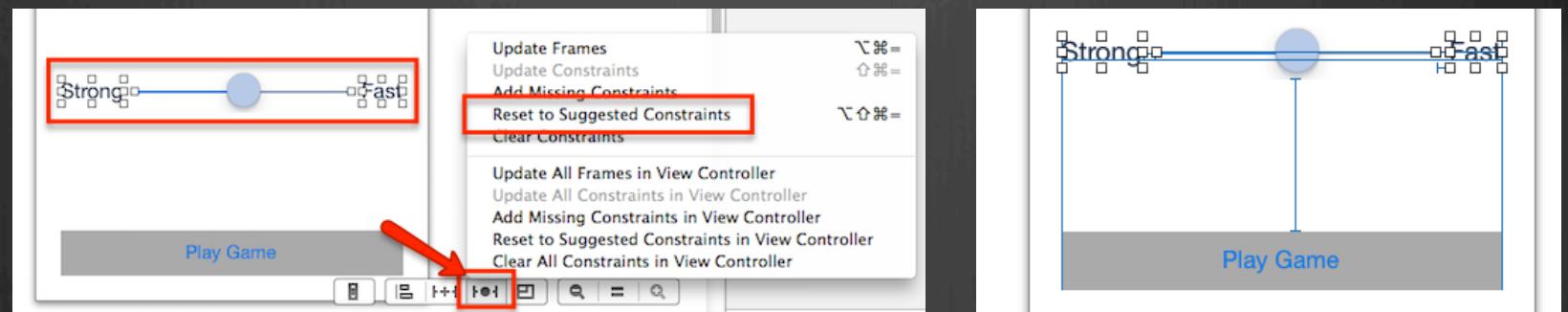


17. Run โปรแกรม จะเห็นว่าเมื่อเปลี่ยน simulator เป็นแนวอน Slides จะไปทับปุ่มทั้ง 2 ปุ่ม

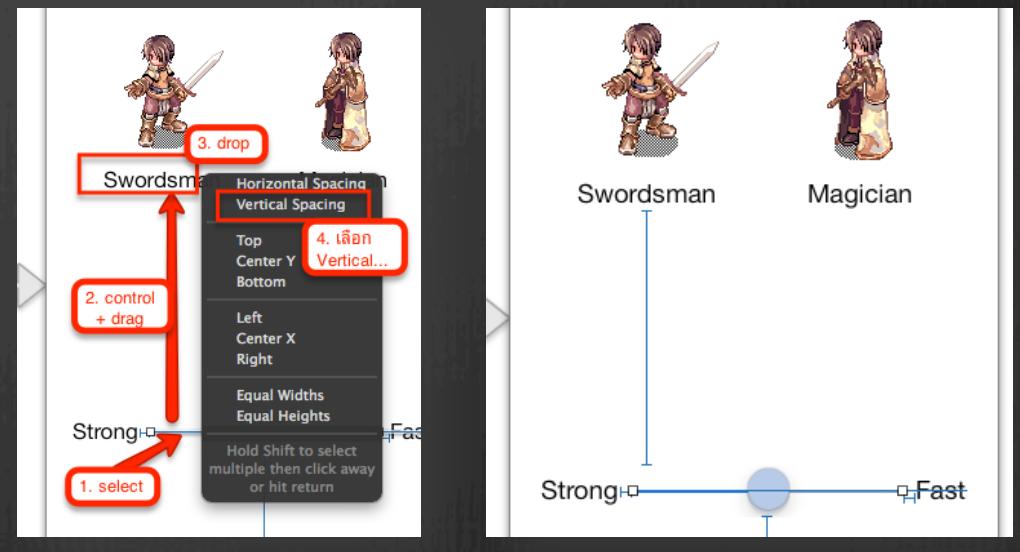


Task : Fix with Auto Layout (7/9)

18. Click เลือก Label และ Slide Bar และ click ที่เมนู “Resolve Auto Layout Issues” และเลือกเมนู “Reset to Suggested Constraints”

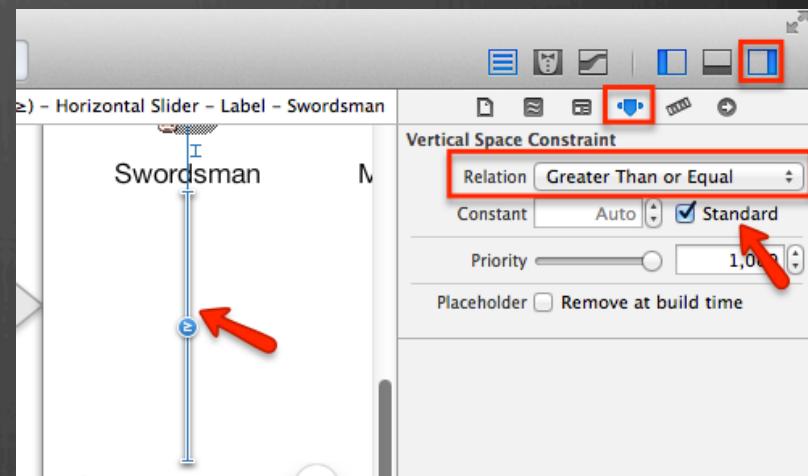


19. Click เลือก Slide Bar กด control ค้างไว้แล้ว drag จาก Slide Bar ไปยัง Label “Swordsman” แล้วเลือกเมนู “Vertical Spacing”

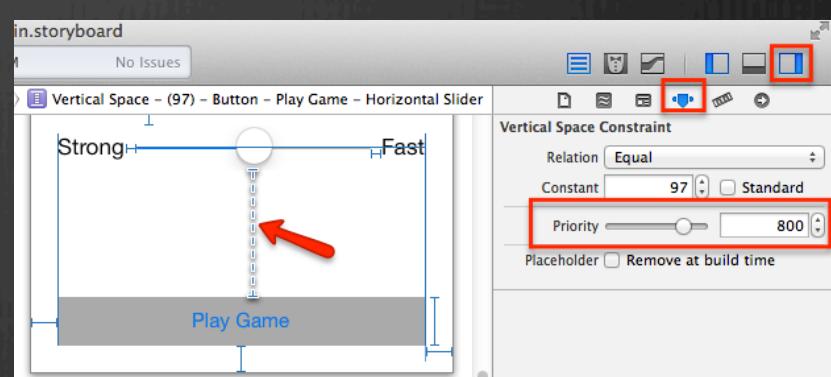


Task : Fix with Auto Layout (8/9)

20. Click ที่ Vertical Spacing Constraint ระหว่าง Slide Bar กับ Label Swordsman ที่ได้จากข้อ 19 แล้วเปิด Attributes inspector
- เปลี่ยน property “Relation” เป็น “Greater Than or Equal”
 - เลือก check box “Standard” ด้านข้าง “Constraint” เป็น true

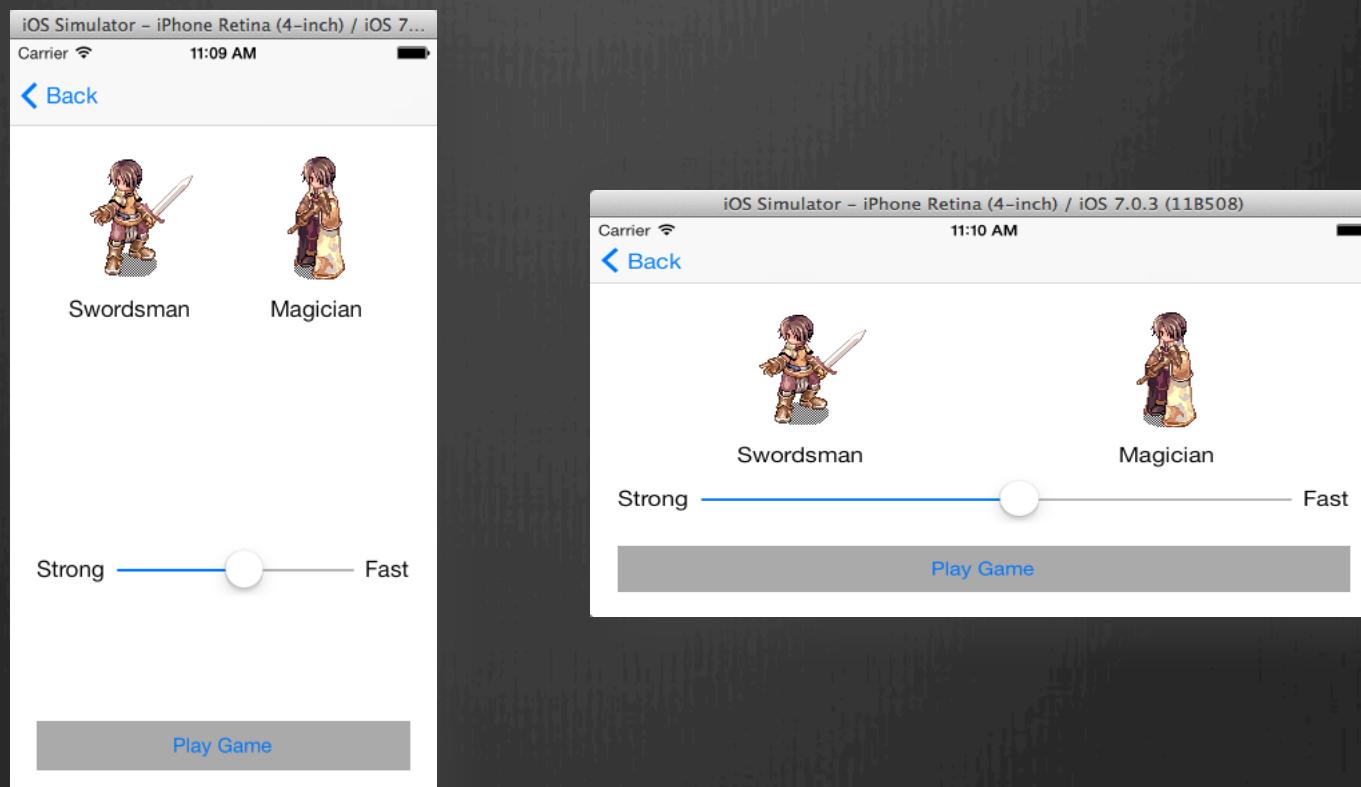


21. Click ที่ Vertical Spacing Constraint ระหว่าง Slide Bar กับปุ่ม “Play Game” แล้วกำหนด property “Priority” เป็น 1



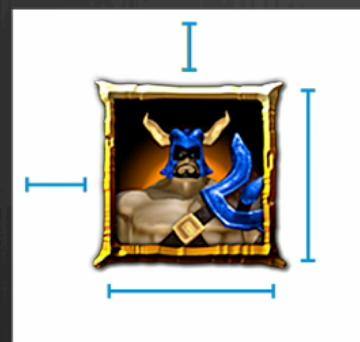
Task : Fix with Auto Layout (9/9)

22. Run โปรแกรม แล้วทดลองเปลี่ยน Simulator เป็นแนวอน สังเกตว่า Slide Bar จะปรับระยะ โดยไม่ทับบน label Swordsman

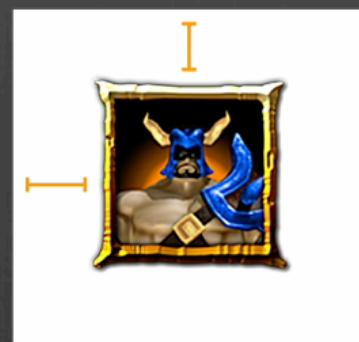


Debug & Resolve

- เมื่อ View มี Constraint แล้วเกิดค่าของ size หรือ position เปลี่ยน Interface Builder จะไม่แก้ค่าของ constraint ให้อัตโนมัติ
- ตัวอย่างการเกิด Intermediate State



Complete. But not what we want.



Try to fix it and it not ready, we called Intermediate State

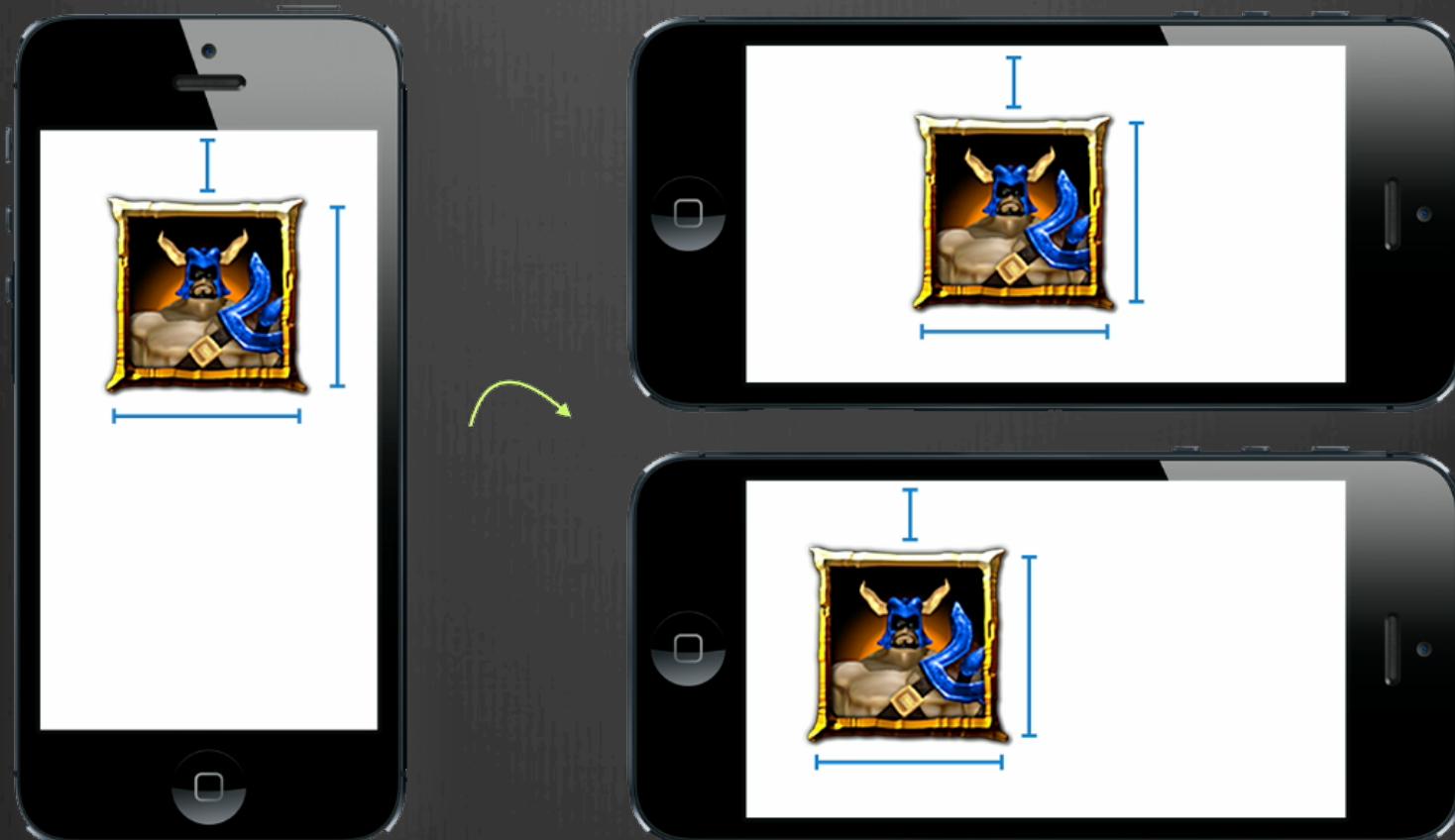


Problem fixed.

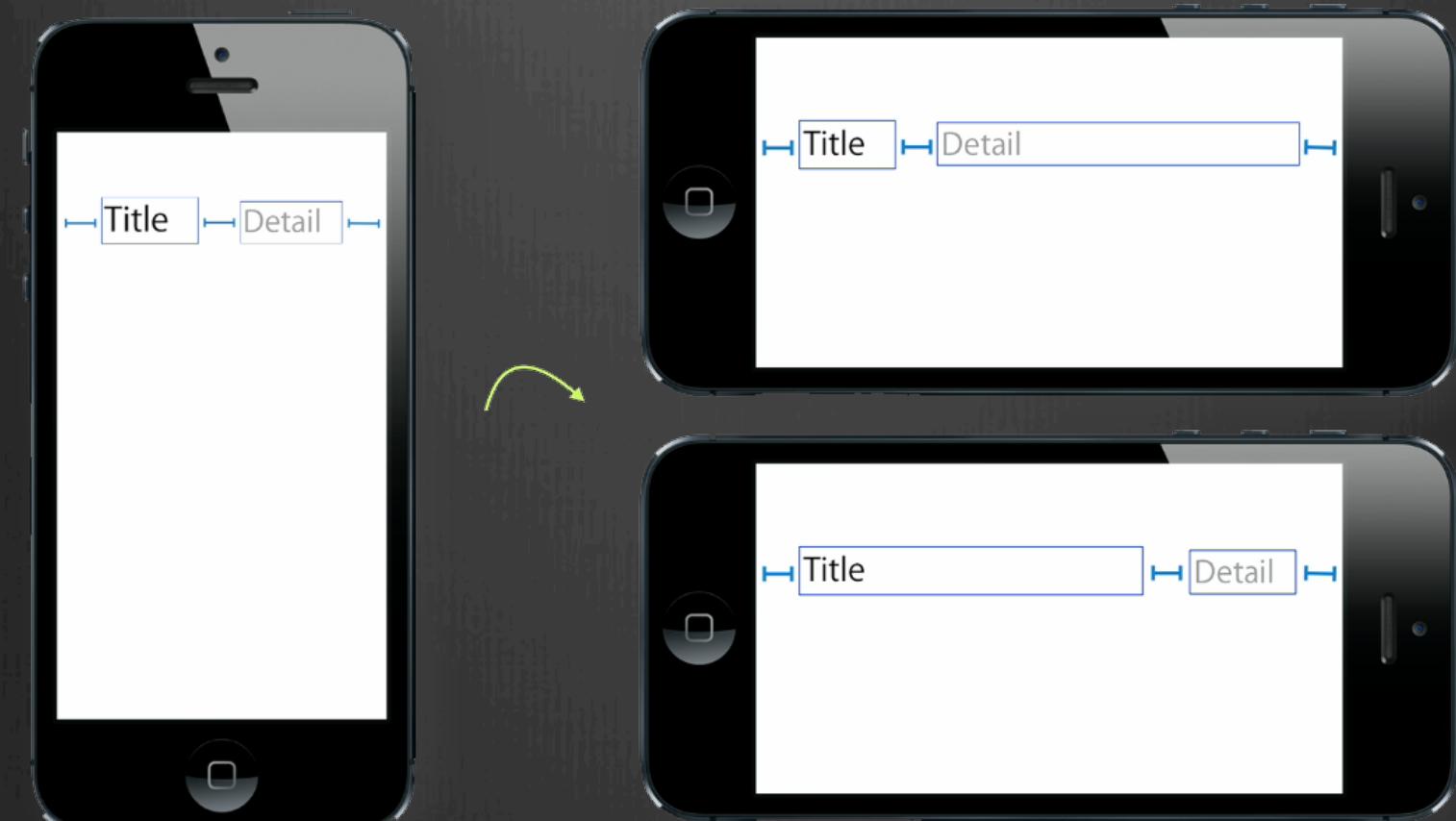
Intermediate State Types

- ⦿ Ambiguous Frames : Not enough information
- ⦿ Conflicting Constraints : Too much information
- ⦿ Misplaced Views : Mismatched position and size

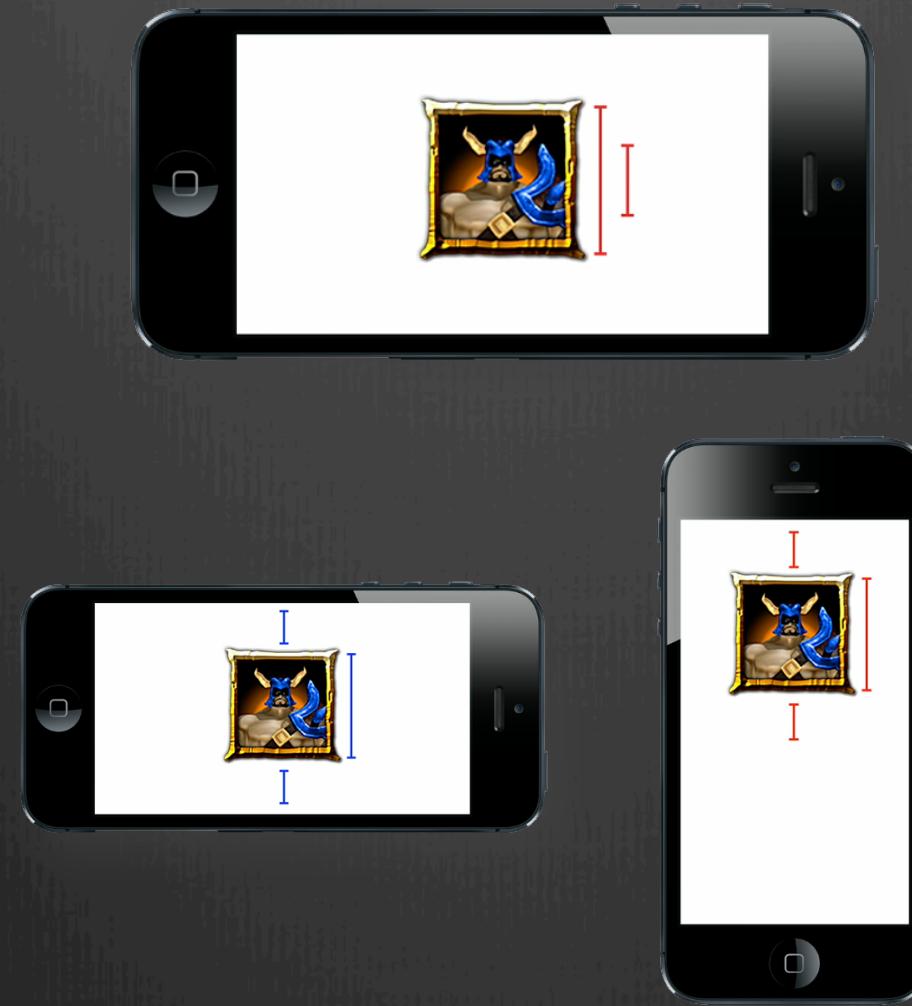
Ambiguous Frames



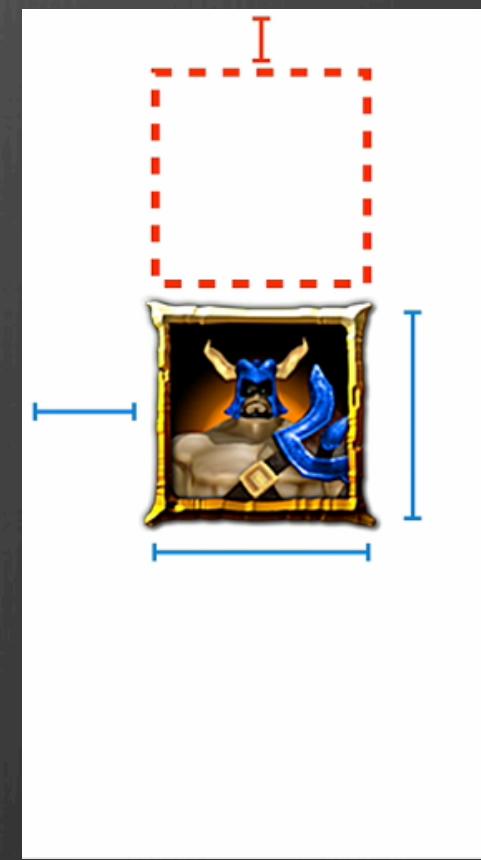
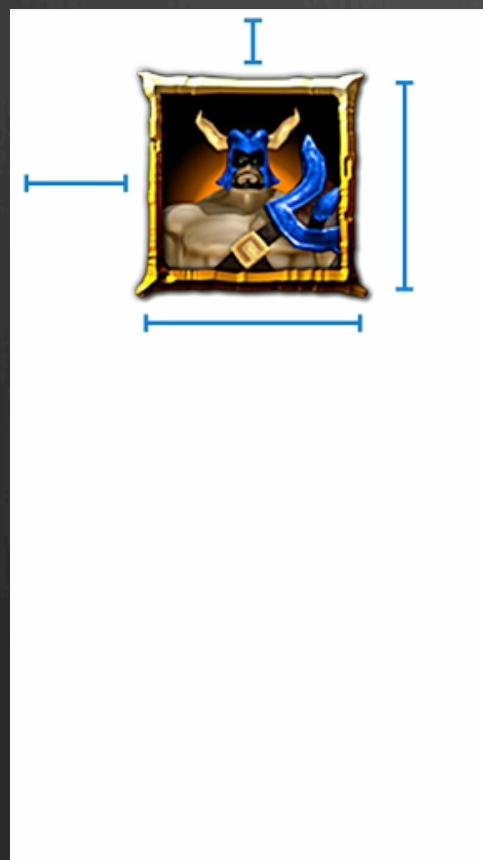
Ambiguous Frames



Conflicting Constraints

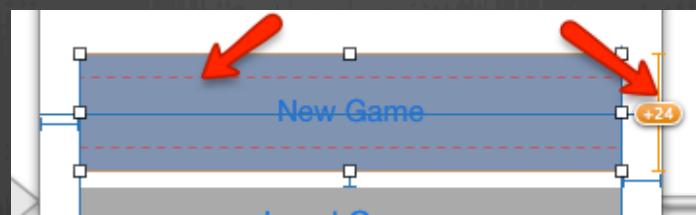


Misplaced Views



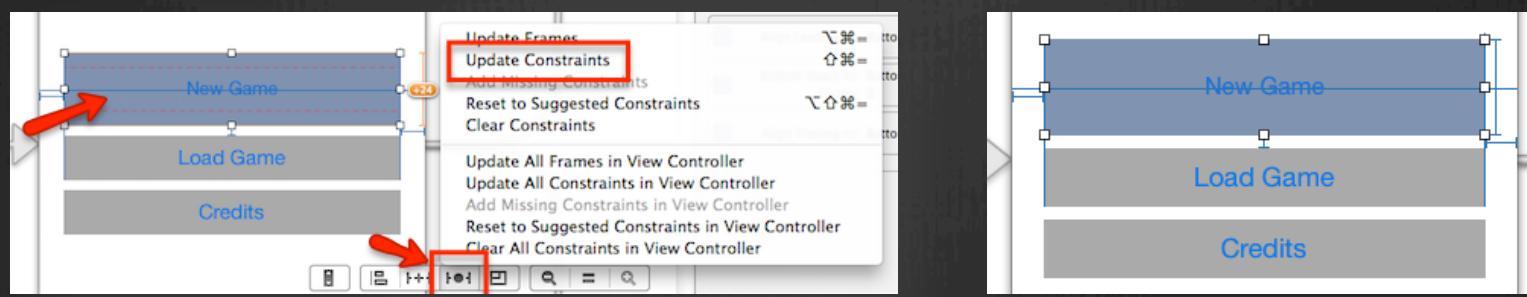
Lab 3/3 : Debug & Resolve (1/8)

1. เปิดไฟล์ Main.storyboard และเพิ่มความสูงของปุ่ม New Game สังเกตว่าปุ่มจะมีปัญหา Misplace ซึ่ง Interface Builder จะบอกว่า ขนาดของปุ่มต่างกับ Height ใน constraint อยู่กี่ point



ทางแก้ไข 2 ทาง คือ ทำให้ขนาดตรงกับ constraint หรือทำให้ constraint ตรงกับขนาด ในกรณีนี้เราต้องการเปลี่ยนขนาด เพราะฉะนั้นจะต้อง update constraint ให้ตรงกับขนาด

2. Click ที่ปุ่ม “Resolve Auto Layout Issues” และเลือก “Update Constraints”



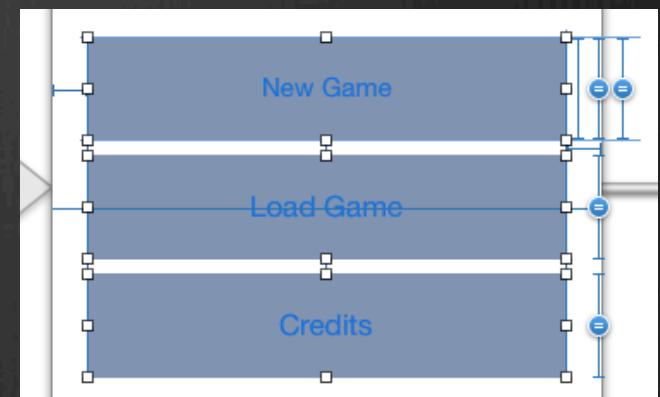
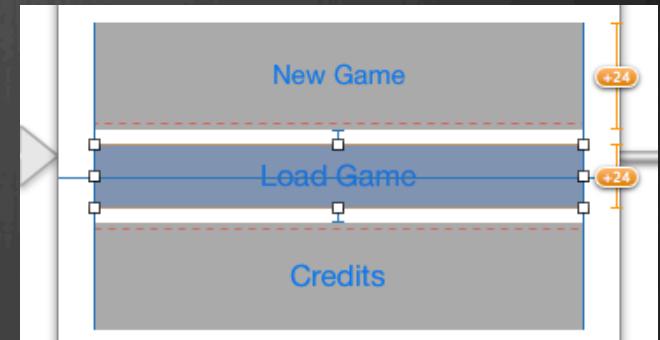
Task : Make Buttons Same Size (2/8)

3. เพิ่มความสูงของปุ่ม Credits ให้เท่ากับปุ่ม New Game ด้วยวิธีสร้าง Equal Height Constraint โดยกด control แล้ว drag จากปุ่ม New Game ไปยังปุ่ม Credits แล้วเลือก “Equal Heights” สังเกตว่าจะเกิดปัญหา Conflict เพราะปุ่ม Credits มี height constraint อยู่แล้ว แต่เราเพิ่ม height constraint เข้าไปอีก
4. ลบ height constraint ของปุ่ม Credits ออก 1 ตัว โดย click constraint ตัวที่มีตัวเลข แล้วกด delete สังเกตว่า Conflict หายไป แต่ยังคงเกิดปัญหา Misplace
5. เปลี่ยนขนาดของปุ่ม ให้ตรงกับ constraint โดย click ที่ “Resolve Autolayout Issues” และเลือก Update Frames



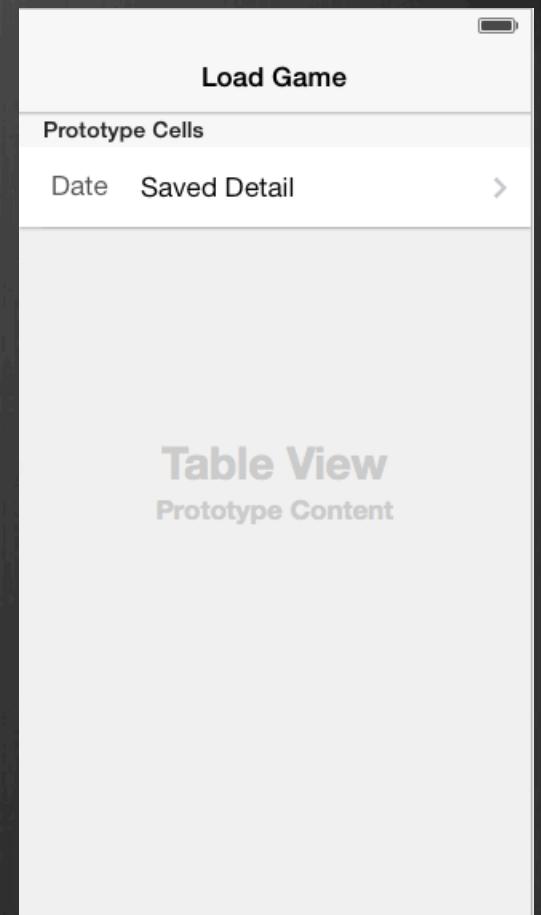
Task : Resolve all Misplace (3/8)

6. Click ที่ปุ่ม Load Game และลบ height constraint ออก (เพื่อหลีกเลี่ยง conflict)
7. เพิ่ม equal height constraint จากปุ่ม New Game ไปยังปุ่ม Load Game โดยกด control ค้างไว้แล้ว drag จากปุ่ม New Game ไปยัง Load Game และเลือกเมนู “Equal Heights” ก็จะเกิด Misplace แต่ไม่ได้เกิดที่ Load Game ตัวเดียว แต่เกิดกับปุ่มอื่นด้วย
8. Click ที่เมนู “Resolve Autolayout Issues” และเลือก “Update All Frames in View Controller” เพื่อแก้ปัญหา Misplace ทุก view ให้ view controller นี้ สังเกตว่าขนาดและตำแหน่งของปุ่มทั้ง 3 จะถูกต้อง



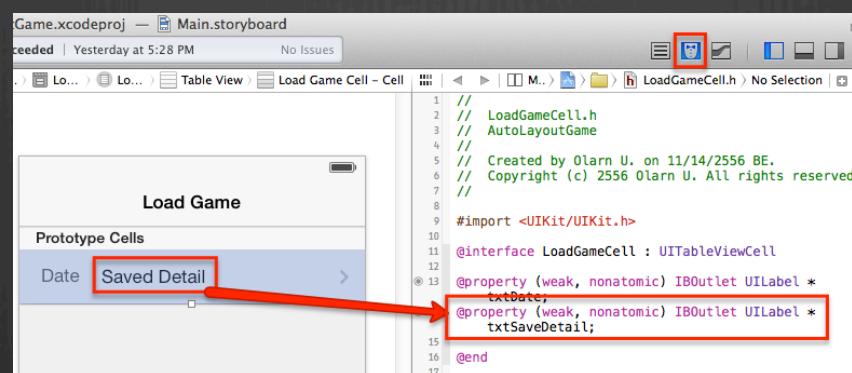
Task : Resolve all Misplace (4/8)

9. เพิ่ม Table View Controller ลงบน storyboard และเพิ่ม Label ลงบน Cell ของ Table ตามรูป
10. Click ที่ cell ของ Table View เปิด Attributes inspector และกำหนด property “Identifier” เป็น “Cell”
11. สร้าง Segue จากปุ่ม “Load Game” มาอยู่ Table View Controller โดยกด control ด้านไว้แล้ว drag จากปุ่ม “Load Game” มาอยู่ Table View Controller และเลือก “push”
12. Click ที่ Segue ใหม่ที่ได้ เปิด Attribute inspector และกำหนด “Identifier” เป็น “segueLoadGame”
13. เปิด Finder และ browse ไปที่ path “../Resources/Day5 - Lab16/” และ drag ไฟล์ทั้งหมด (ยกเว้นรูป .png) มา drop ลงบน project ใน Project Navigation



Task : Resolve all Misplace (5/8)

14. Click ที่ Table View Controller เปิด Identity Inspector ใน Inspector Pane และเปลี่ยน class ของ Table View เป็น “LoadGameViewController”
15. Click ที่ Cell ของ Table View และเปิด Identity Inspector ใน Inspector Pane และเปลี่ยน class ของ Cell เป็น “LoadGameCell”
16. เปลี่ยน editor mode เป็น Assistant editor โดยให้ code ด้านขวาเป็นไฟล์ “LoadGameCell.h” และผูก delegate ของ label บน Cell ไปยัง property “txtDate” และ “txtSaveDetail” ที่มีอยู่แล้ว



17. Run โปรแกรมแล้วเปิดไปที่ Load View จะเห็นว่าข้อความบน label ถูกตัดออกไปโดยที่ขนาดของ Label ไม่ขยายตาม



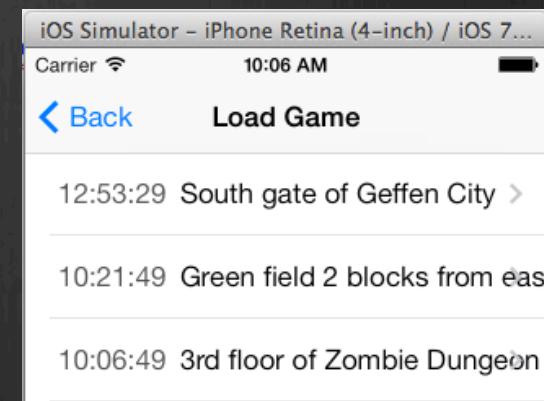
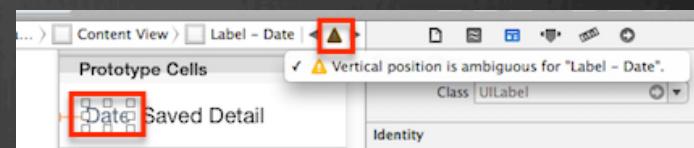
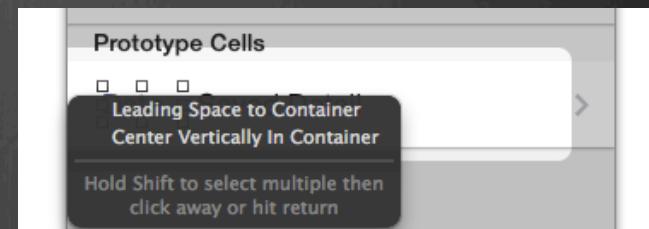
Task : Resolve all Misplace (6/8)

18. กด control ค้างไว้แล้ว drag จาก label “Date” ไปยังขอบด้านซ้ายระหว่าง Label และขอบของ cell แล้วเลือก “Leading Space to Container”

สังเกตว่า ที่มุมขวาบนของ editor ใน Interface Builder จะมีเครื่องหมายสามเหลี่ยมลีเหลือง ถ้า click ดูจะเห็นข้อความที่บอกว่า มีอะไรไม่ถูกต้อง

19. Click ที่ label “Date” แล้ว click ที่ปุ่ม “Resolve Autolayout Issues” แล้วเลือก “Add Missing Constraints in Load Game Cell”

20. Run โปรแกรม สังเกตว่า Label ของ Date ไม่ถูกตัดแล้ว แต่ Label “Saved Detail” ขาดจากขอบ



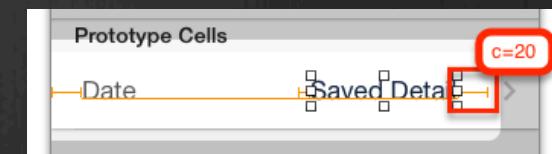
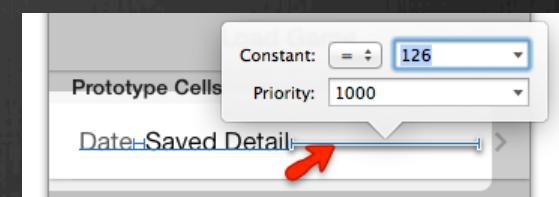
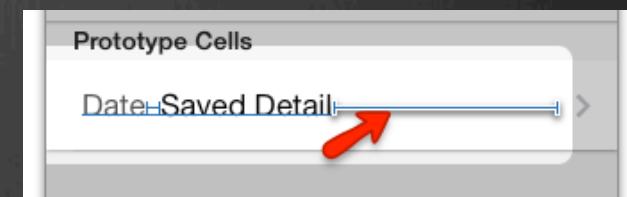
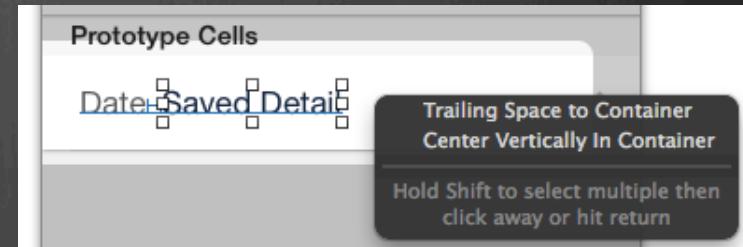
Task : Resolve all Misplace (7/8)

21. เพิ่ม padding ของ label “Saved Detail” กับขอบขวาเพื่อ label ถูกตัดล้ำ content ยาวเกินไป โดย กด control แล้ว drag จาก Saved Detail ไปยังพื้นที่ว่างด้านขวาของ label และเลือก “Trailing Space to Container”

ถ้า Run ตอนนี้จะเห็นว่า Save Detail โดนตัดแล้ว แต่ข้อความสั้นเกินไป นั่นเป็นเพราะ Interface Builder จะกำหนดค่าโดยอัตโนมัติจาก frame ที่เรา design ไว้ ทำให้ขนาดของ constraint มากเกินไป

22. Double click ที่ Horizontal constraint และกำหนดค่า Constraint เป็น 20

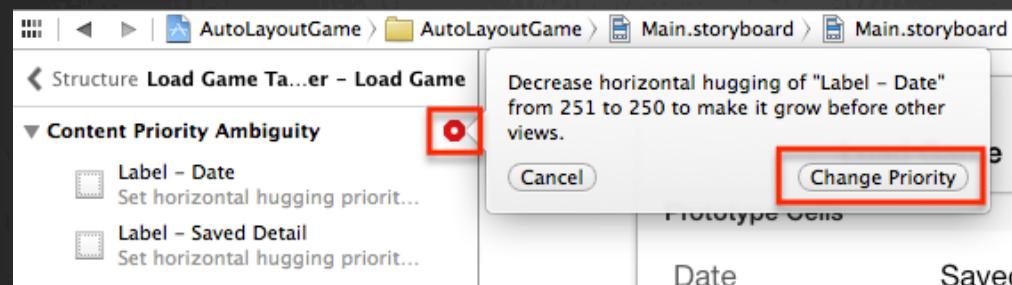
เมื่อกำหนดค่าเป็น 20 Autolayout จะปรับ padding ตามรูป แต่ก็จะเกิดปัญหา Ambiguous ตามมา เพราะ Interface Builder ไม่รู้ว่า ถ้า label 2 ตัวเกิดมี content เกินทั้งคู่ จะเอาตัวไหนเป็นหลัก



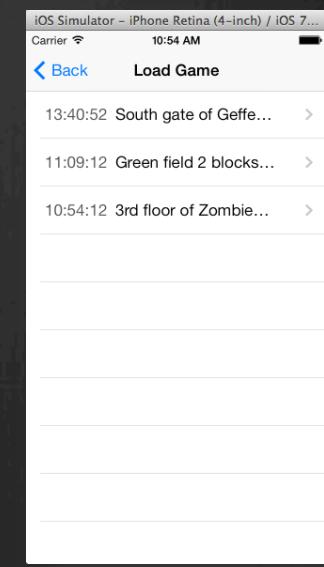
Task : Resolve all Misplace (8/8)



23. ที่ Document Outline จะสังเกตว่ามี icon สีแดงอยู่ ถ้า click ที่ icon สีแดง Interface Builder จะแสดง error ให้เห็น พร้อมทั้ง hint และ resolve guideline ให้เห็น ซึ่งในตัวอย่างเป็นเพราะ Label ทั้ง 2 ตัวมีค่า “content hugging” เท่ากัน
24. Click ที่ icon สีแดง ในหน้าที่ 2 แล้วเลือก “Change Priority”



25. Run โปรแกรมเพื่อดูผลลัพธ์



Topics not covered

- Auto Layout by code (NSLayoutConstraint)
- Auto Layout Virtual Format
 - [NSLayoutConstraint constraintsWithVisualFormat:@"
options:0
metrics:nil
views: NSDictionary]
- Things that can go wrong (when using Auto Layout by code)

References & More

- Constraints over different View Hierarchy.
- WWDC 2013 Session Video – Taking Control of Auto Layout in Xcode 5 <http://goo.gl/ZqZnXO>
- More Auto Layout Videos in WWDC 2012
<https://developer.apple.com/videos/wwdc/2012/>
 - Introduction to Auto Layout for iOS and OS X
 - Best Practices for Mastering Auto Layout
 - Auto Layout by Example