



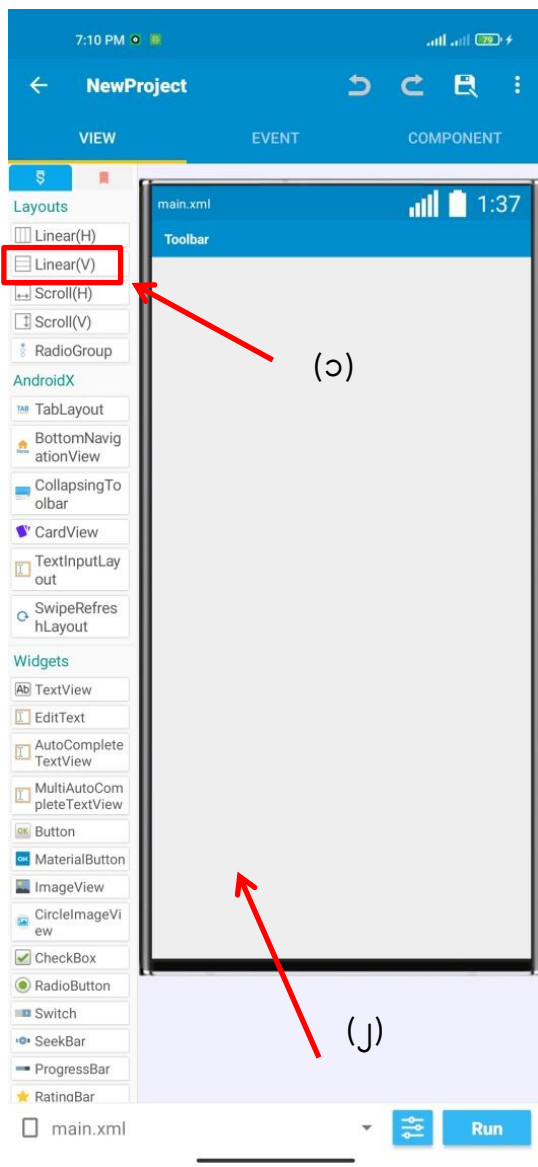
Sketchware Width & Height

Width/Height

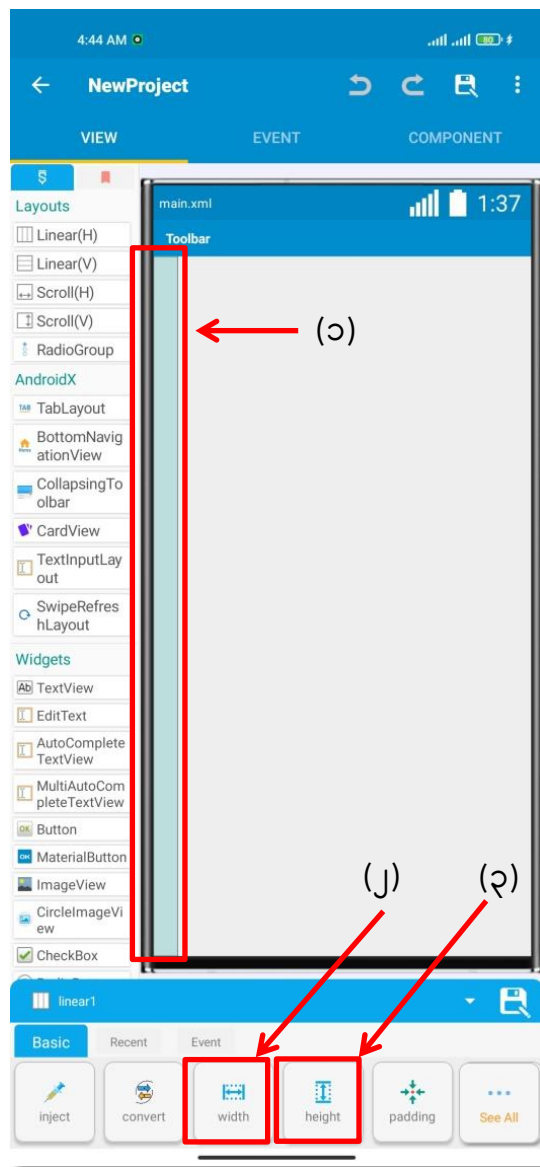
ဒီသင်ခန်းစာမှာ ကျတော် Width/Height ရဲ့ အလုပ်လုပ်ပုံကို ရှင်းပြပေးမှာ ဖြစ်ပါတယ်။ သေချာသဘောပေါက်အောင် လေ့လာစေချင်ပါတယ်။ Width/Height သင်ခန်းစာကို စပြီးမလေ့လာခင် ဘယ်ဟာကို width ခေါ်တာလဲ၊ ဘယ်ဟာကို height ခေါ်တာလဲဆိုတာကို အရင်သိထားဖို့လိုပါတယ်။ ပုံ (WH-1) တွင်ကြည့်ပါ။ ပုံတွင် A, B, C အမှတ်(၃) ခုရှိသည်ကို တွေ့ပါလိမ့်မည်။ အမှတ် (A) မှ (B) ကို **width** ဟု ခေါ်ပါတယ်။ အမှတ် (B) မှ (C) ကို **height** ဟု ခေါ်ပါတယ်။



ပုံ (WH-1)



ပုံ (WH-2)

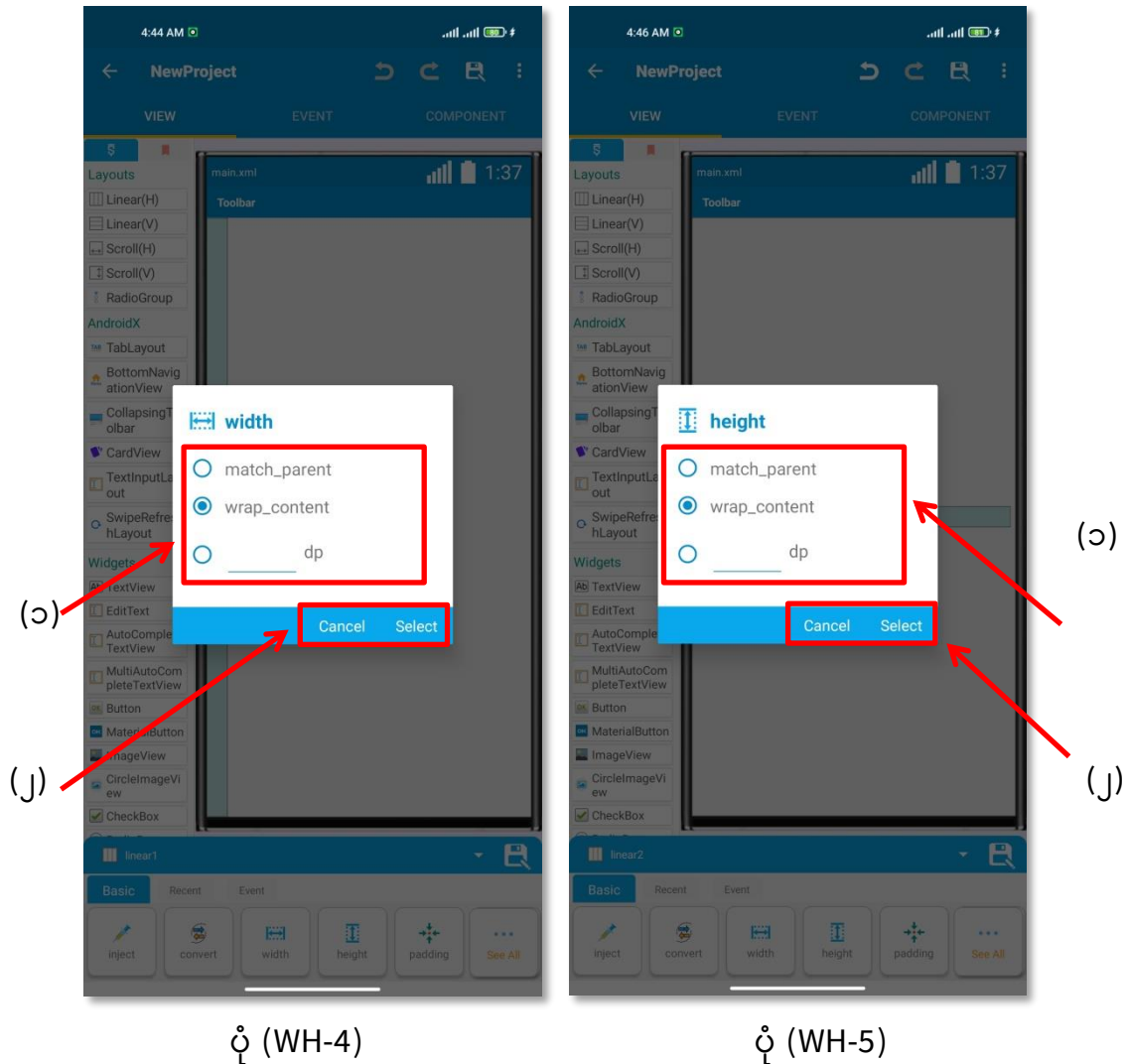


ပုံ (WH-3)

Width နဲ့ Height ကို ခွဲခြားတတ်သွားပြီးဆိုရင် ကျတော်တို့စလေ့လာကြည့်ကြရအောင်။

ကျတော်တို့ပထမဆုံး Skechware ကနေ Project အသစ်ကို ယူရပါမယ်။ ယူပြီးရင် ပုံ (WH-2) မြား (၁) နေရာတွင်ပြထားသည့် Linear (V) ကို ပုံ (WH-2) မြား (၂) နေရာသို့ Drag ဖိဆွဲပြီး ထည့်လိုက်ပါ။ ဆွဲထည့်ပြီးသွားရင် ပုံ (WH-3) မြား (၁) နေရာတွင် ပြထားသလို ရရှိလာမှာ ဖြစ်ပါတယ်။ ရရှိလာတဲ့ Linear (V) ရဲ့ width ရဲ့ တန်ဖိုးဟာ wrap_content ဖြစ်ပြီး heigh ရဲ့ တန်ဖိုးဟာ match_parent ဖြစ်ပါတယ်။

Linear တစ်ခု၏ width ကို သတ်မှတ်ရန်အတွက် ပုံ (WH-3) မြား (၂) နေရာကို နှိပ်လိုက်ရပါမယ်။ ပုံ (WH-4) တွင် ပြထားသလိုပေါ်လာမှာ ဖြစ်ပါတယ်။ height ကို သတ်မှတ်ရန်အတွက် ပုံ (WH-3) မြား (၃) နေရာကို နှိပ်ရပါမည်။ ပုံ (WH-5) တွင် ပြထားသည့်အတိုင်း ပေါ်လာမှာ ဖြစ်ပါတယ်။



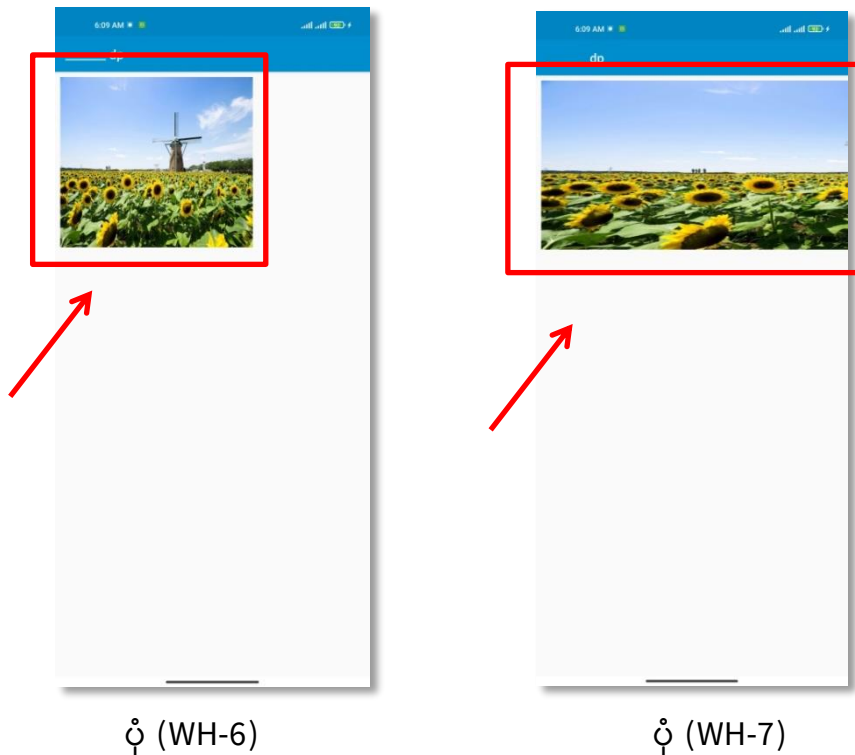
Width နဲ့ Height ရဲ့ တန်ဖိုးကို ကျတော်တို့အနေနဲ့ သတ်မှတ်ရာမှာ (၃) ခုရွေးချယ် လို့ရပါတယ်။ ပုံ (WH-4) နဲ့ ပုံ (WH-5) တို့ရဲ့ မြား (၁) နေရာမှာ ရွေးချယ်ရမှာ ဖြစ်ပါတယ်။ wrap_content နဲ့ match_parent ကို အောက်မှာ ထပ်ရှင်းပြပေးပါမယ်။

ပုံ (WH-4) နဲ့ ပုံ (WH-5) တို့ရဲ့ မြား (၁) နေရာမှ အောက်ဆုံးက “_____dp” ဆိုတာက Custom Number ကို မိမိစိတ်ကြိုက် သတ်မှတ်လို့ရတဲ့နေရာ ဖြစ်ပါတယ်။ နံပါတ်ကို 999 အများဆုံး ဖြည့်လို့ရနိုင်ပါတယ်။ Custom Number ထည့်လိုက်ခြင်းဟာ မိမိတို့ထည့်လိုက်တဲ့ Linear အရွယ်အစားကို ပုံသေသတ်မှတ်လိုက်တာ ဖြစ်ပါတယ်။ ကျတော်တော်တို့အနေနဲ့ Custom Number နဲ့ သတ်မှတ်လိုက်ခြင်းဖြင့် ဘယ်လိုအကျိုး သက်ရောက်မှုရှိတယ်ဆိုတာ အနည်းငယ် ရှင်းပြချင်ပါတယ်။

ဥပမာ - ကျတော်တို့ရဲ့ Phone Screen ဟာ width 490 dp နဲ့ height 1064 dp ရှိသည်ဆိုပါစို့။ ကျတော်တို့ View တစ်ခုရဲ့ width ကို 200 dp သတ်မှတ်လိုက်ပါက ထည့်လိုက်တဲ့ View ကို Phone Screen အတွင်းမှာ အဆင်ပြေစွာ မြင်တွေ့ရမည် ဖြစ်သော်လည်း 700 dp လောက်သတ်မှတ်လိုက်ပါက Phone Screen မှာရှိတဲ့ width 490 dp ထက် ကျော်သွားတဲ့အတွက်ကြောင့် view ရဲ့ size က Phone Screen ထက် ကြီးသွားမှာ ဖြစ်သည့်အတွက် အဆင်ပြေစွာ မြင်တွေ့ရမှာ မဟုတ်ပါဘူး။

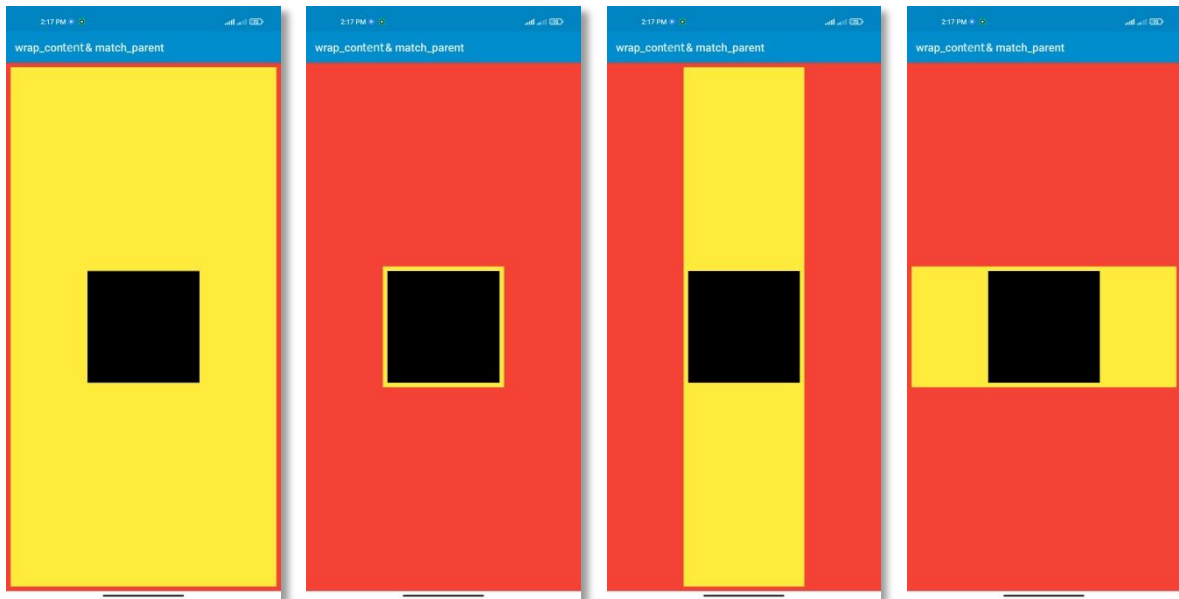
ပုံ (WH-6) သည် width ကို 300 dp သတ်မှတ်ထားသည့် အတွက် ပုံမှာ ပါဝင်သည့်အစိတ်အပိုင်းများကို မြင်တွေ့ရပါသည်။ ပုံ (WH-7) သည် width ကို 999 ပေးလိုက်ပါတယ်။ ဒါကြောင့် Phone Screen ရဲ့ width 490 dp ထက်ကျော်လွန် သွားပြီး ပုံမှာပါတဲ့ အစိတ်အပိုင်းအားလုံးကို မမြင်တွေ့ရတော့ပါ။ ဒီလောက်ဆိုရင် width သတ်မှတ်ခြင်းရဲ့ ရလဒ်ကို သဘောပေါက်မယ်ထင်ပါတယ်။

မိမိအနေနဲ့ width/height ကို 200/500 dp သတ်မှတ်လိုက်ပါက Phone Screen ဆိုဒ်သည် width 500 dp နဲ့ height 950 dp ရှိလျှင်အဆင်ပြေနိုင်ပြီး Phone Screen ဆိုဒ်သည် width 150 dp နဲ့ height 400 dp သာ ရှိမည်ဆိုလျှင် အဆင်ပြေနိုင်မည်မဟုတ်ပါ။ အဲဒီတော့ မိမိတော့ရေးသားမည့် Application သည် Screen Size ဘယ်လောက်ရှိတဲ့ Phone တွေမှာ သုံးမှာလဲ/ဘယ်နေရာမှာ ထည့်မှာလဲဆိုတာကို ကြိုစဉ်းစားပြီးသတ်မှတ် ပေးရမှာ ဖြစ်ပါတယ်။ ဥပမာ Scrol View ထဲကို ထည့်ပြီးသုံးမည်ဆိုရင်တော့ တစ်မျိုး စဉ်းစားပြီး မိမိလိုချင်တဲ့ ဒီဇိုင်းအနေအထားကို သတ်မှတ်နိုင်ပါတယ်။



ပုံ (WH-6)

ပုံ (WH-7)



ပုံ (WM-1)

ပုံ (WM-2)

ပုံ (WM-3)

ပုံ (WM-4)

wrap_content/match_parent

wrap_content နဲ့ match_parent ကို အနည်းငယ်ကြားဖြတ်အနေဖြင့် ရှင်းပြ ပါရစေ။ ကျတော် wrap_content နဲ့ match_parent ရဲ့ သဘောတရားကို ပုံ (WM-1)/ (WM-2)/ (WM-3)/ (WM-4) တို့ဖြင့် ဥပမာပြပေးထားပါတယ်။ ပုံတွင် ကျတော် Linear (၃) ခုကို အသုံးပြုထားပါတယ်။ အနီရောင်/ အဝါရောင်/ အနက်ရောင် တို့ဖြင့် ပြပေးထားသည်။ အဝါရောင်သည် အလုပ်လုပ်မည့် အရောင်ဖြစ်ပါသည်။ အနက်ရောင်ရဲ့ width နဲ့ height ကို 200 dp ပေးထားပါတယ်။

ပုံ (WM-1) သည် width နဲ့ height ကို match_parent ယူထားပါတယ်။ match_parent သည် ဘောင်အပြည့်ယူသည့်သဘောဖြစ်ပါသည်။

ပုံ (WM-2) သည် width နဲ့ height ကို wrap_content ယူထားတာဖြစ်ပါသည်။ wrap_content ရဲ့ သဘောကတော့ အတွင်းထဲမှာရှိသည့်အရာရဲ့ ဆိုဒ်နဲ့ ကွက်တိ အနေအထားကို ယူတာဖြစ်ပါတယ်။

ပုံ (WM-3) သည် width ကို wrap_content ယူထားပြီး height ကို match_parent ယူထားပါတယ်။

ပုံ (WM-4) သည် width ကို match_parent ယူထားပြီး height ကို wrap_content ယူထားတာဖြစ်ပါတယ်။

အခုဆိုရင် wrap_content နဲ့ match_parent ရဲ့သဘောတရားကို နားလည် လိမ့်မယ်ထင်ပါတယ်။



