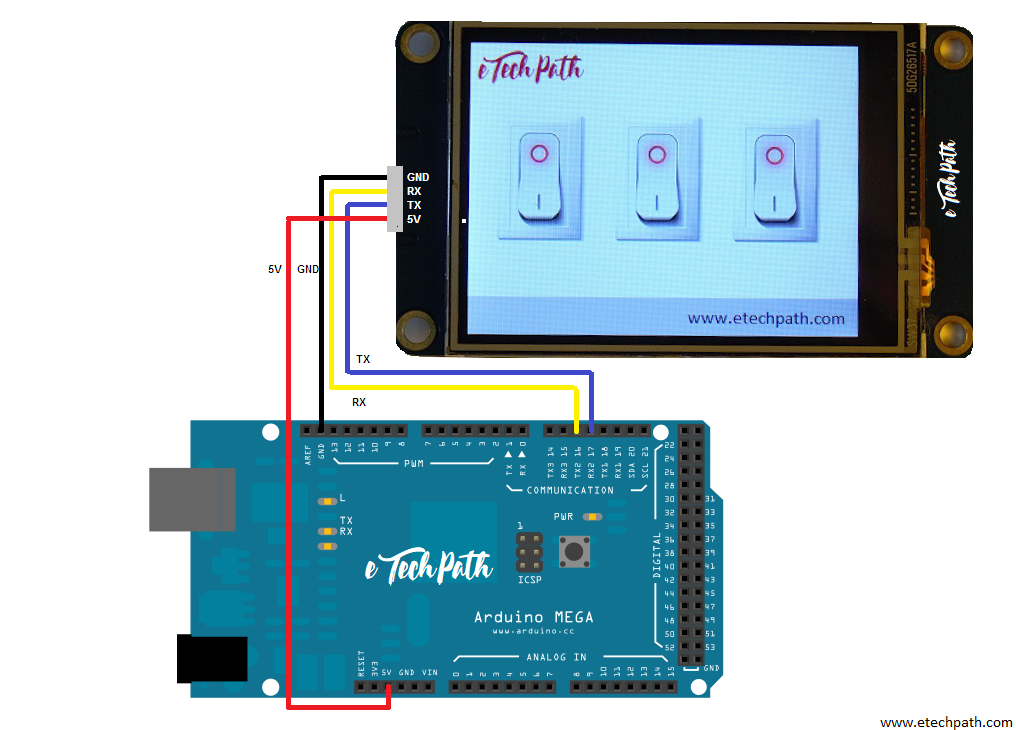
# **Vaja 0b** – Spoznavanje gradnikov Nextion editor-ja

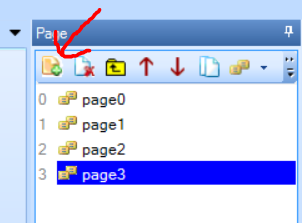
1. Na računalniku zaženemo **Nextion Editor** razvojno okolje.

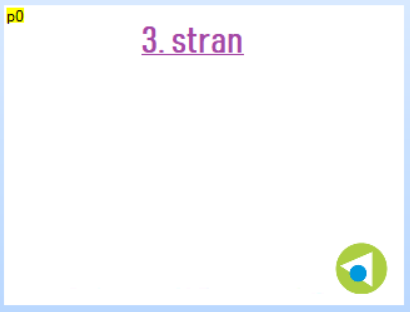
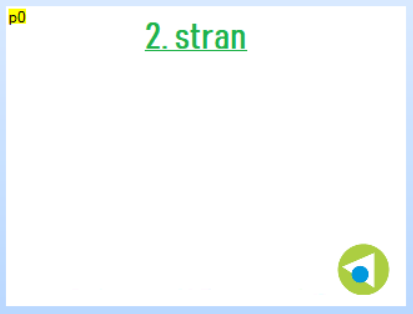
Pravilna povezava konektorjev Nextion zaslona z Arduino razvojno ploščo je prikazana na sliki:



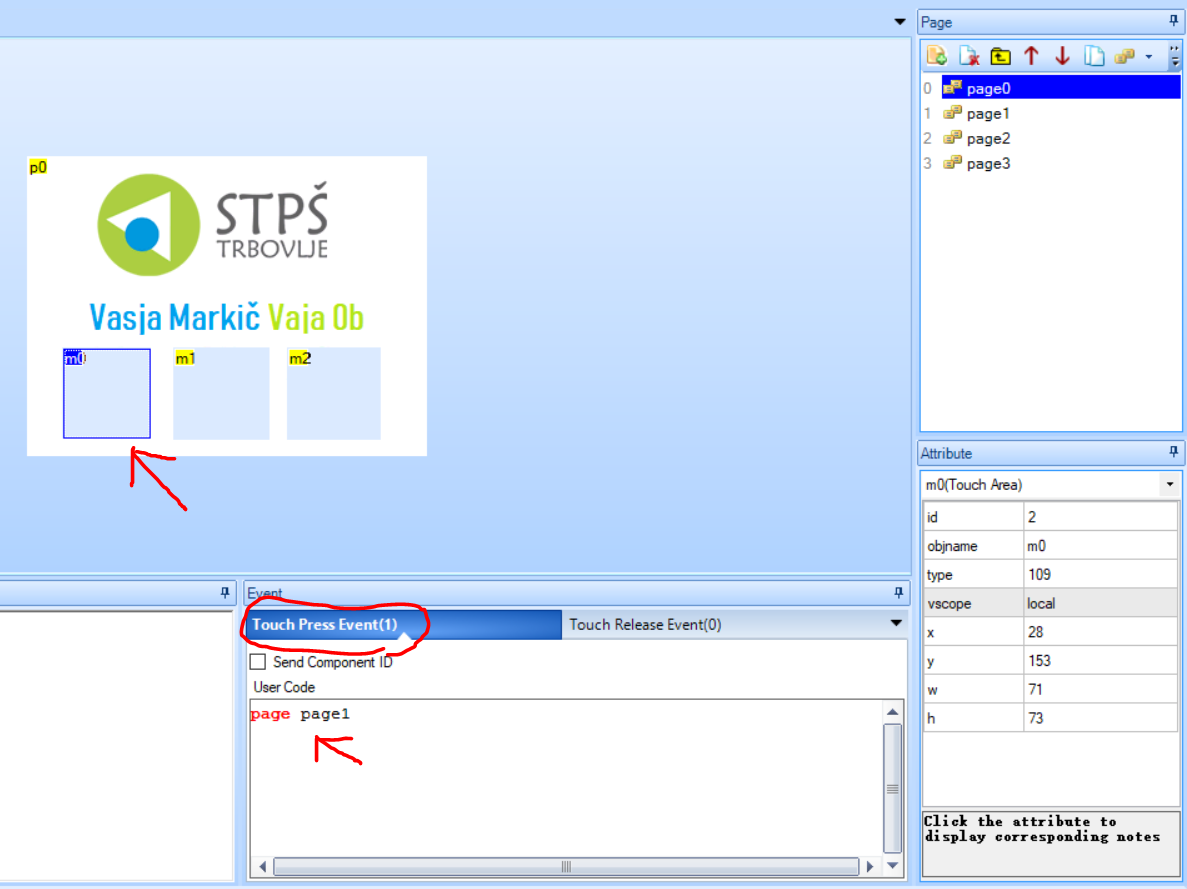
Izbrani Nextion zaslon je 2.4'', kar pomeni, da je 320 x 240 px oziroma tudi 240 x 320 px.

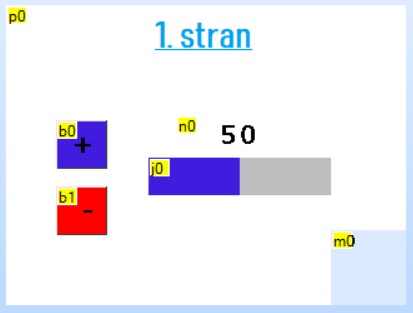
## Naloga

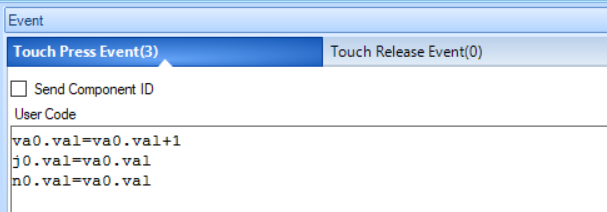
1. **Ustvarite projekt** - Izberemo *New* (zavihek *File*) in shranite ime projekta **vaja0\_ime\_priimek.HMI**. Izberite ustrezni model zaslona (*Device* 🡪*basic, NX3224T024; DISPLAY 🡪 90 Horizontal).* Kliknemo *Ok.*
2. **V **slikarju** izdelajte sliko za prvo stran; z logom šole, vašim imenom in priimkom ter št. vaje in tremi različnimi gumbi (primer na sliki spodaj). Pazite, da sliko na disku shranite kot velikosti platna 320×240 točk ter .png končnico.
3. V **Nextion Editorju** vašo sliko kot ozadje prenesete na začetno stran tako, da v levem oknu »Toolbox« kliknete gradnik *Picture*, v spodnjem »Picture« oknu kliknete **+** ter izberete na disku vašo shranjeno sliko. Sedaj v desnem spodnjem oknu »Atribute« gradnika p0(Picture) kliknete na prazno polje poleg **pic** in izberete vašo prednaloženo sliko ozadja. Ker je to vaša prva slika, ima **pic** vrednost 0.
4. Vse tri številke oz. gumbe bomo pokrili z *Hotspot* polji. V »Toolbox« izberemo *Hotpsot* in pokrijemo vsak gumb posebej s tremi pravokotnimi območji, ki nam omogoča ta gradnik. To naredimo z miško – spremenimo pozicijo polja in ga tudi ustrezno povečamo/pomanjšamo. Izdelamo torej tri *Hotspot* točke. Te točke imajo **objektna** imena **m0**, **m1** in **m2**.
5. V desnem zgornjem oknu »Page« dodamo še tri nove strani s klikom na gumb *Add.*
6. V slikarju izdelajte še 3 ozadja za vse tri strani (page1, page2 in page3), ki vsebujejo oštevilčen naslov strani ter logo šole, ki bo imel kasneje funkcijo gumba za vrnitev na začetno stran. Shranite vse tri slike na disk in jih uvozite v Nextion Editor (glej c razdelek) ter jih namestite na vsako stran tako, da v oknu »page« najprej izberete ustrezno stran in dodate gradnik *Picture* in v polju pic ustrezno ozadje.



1. Na vsaki strani dodajte še *Hotspot,* ki bo prekrilo vse tri logotipe.
2. Vsakemu *Hotspot-u* sedaj dodajte funkcijo *Touch Press Event* tako, da ustrezno v okno »Event« napišete ukaz za premik po strani *page*, poleg pa še ime strani, na katero naj skoči, primer:



1. Ko ustrezno povežete vse *Hotspot-e* s stranmi, kliknete na *Compile.* Če program ne javi napak v oknu »Output«, potem kliknete na *Debug* ter tako simulirate delovanje zaslona.
2. Za delo z ostalimi gradniki potrebujemo **generirati vsaj eno pisavo.** Kliknemo *Tools 🡪 Font Generator* in izberemo usterzno višino v px, obliko pisave ter ime, pod katero jo bomo shranili. Kliknemo *Generate font.* Font prav tako shranimo na disk in ga dodamo projektu (*Add generated font? –* kliknemo *Yes).*
3. V Nextion Editor se postavimo na **page1**. Tukaj ustvarimo dve tipki z gradnikom *Button*. V polje **txt** okna »Atribute« gradnika ustrezno dopolnite, da bo v prvi tipki znak +, v drugi pa - . Tipki ustrezno spremenite barve (**bco** in **bco2**).
4. Dodajmo še tri gradnike: *Number, Progress bar* in *Variable* (**objname** so n0, j0 in v0)*.* Vsem trem nastavimo srednjo vrednost **val** na 50. Pazite – spremenljivke so pod glavnim oknom, v posameznih okvirčkih.
5. S tipko + bomo *progress bar* ter tudi številčno vrednost v polju povečali za 1, s pritiskom na tipko – pa odšteli za 1. Ker ta proces sprožita dotika na teh dveh tipkah, bomo kodo vpisali v gradnika **b0** in **b1** v zavihku *Touch Press Event.*



Vrednost spremenljivke **va0** ob pritisku na tipko + povečamo za ena. Ta nova vrednost se v drugi in tretji vrstici kode prenese tako na objekt **j0** (progress bar) in **n0** (polje s številom). Podpičij tukaj ne pišemo, vsaka vrstica je nov dogodek oz. *Event.*

Podobno napišite kodo pri tipki – in preverite, če v simulaciji program deluje (*Debug).*

1. Če spremenite vrednosti in preklapljate med strani, boste opazili, da se vrednosti resetirajo na 50. Razlog za to je atribut **vscope*,*** katerega vrednost morate ustrezno spremeniti za vse objekte, ki so deležni dogodka na dotik.
2. Vašo datoteko .HMI (koda), .zi (font) in zapakirane (ZIP) slike, ki ste jih uporabili za kreiranje grafičnega vmesnika naložite na GITHUB portal. Če še niste še registrirani, se prijavite in uredite svoje profilne podatke. Nato ustvarite nov *repository* z ustreznim imenom vašega projekta. Repozitor poimenujte **Nextion projects**, vanj pa naložite tako .HMI, .zi kot .ZIP datoteke. Ustrezno komentirajte besedilna polja preden potrdite prenos datotek, saj le ta vam služijo kot razlago vašega dela!
3. V kolikor imate Nextion zaslon ter mikroSD kartico z adapterjem, lahko naložite vaš program: Kliknete *Compile*, nato *File 🡪 Open build folder.* V oknu skopirajte ustrezno .tft datoteko na mikroSD kartico (mora biti prazna!) in tako vstavite mikroSD kartico v režo (izklopljenega) zaslona. Zaslon proklopite, počakajte da se strojna koda naloži, zaslon izklopite, izvlecite mikroSD kartico ven iz reže ter ponovno priklopite zaslon. Deluje? ☺