

Vpeljava SCADA sistema s pomočjo orodja Ignition na proizvodnjo linijo

Predstavitev raziskovalnega seminarja

Opis projekta

- ▶ Namen projekta
 - ▶ Raziskati razloge za uporabo SCADA Sistema
 - ▶ SCADA (Supervisory Control and Data Acquisition)
 - ▶ Raziskati trende pri SCADA sistemih
 - ▶ Različne rešitve in orodja, ki so na voljo
 - ▶ Raziskava SCADA sistemov z izgradnjo testnega sistema na proizvodni liniji

Razlogi za uporabo SCADA sistema

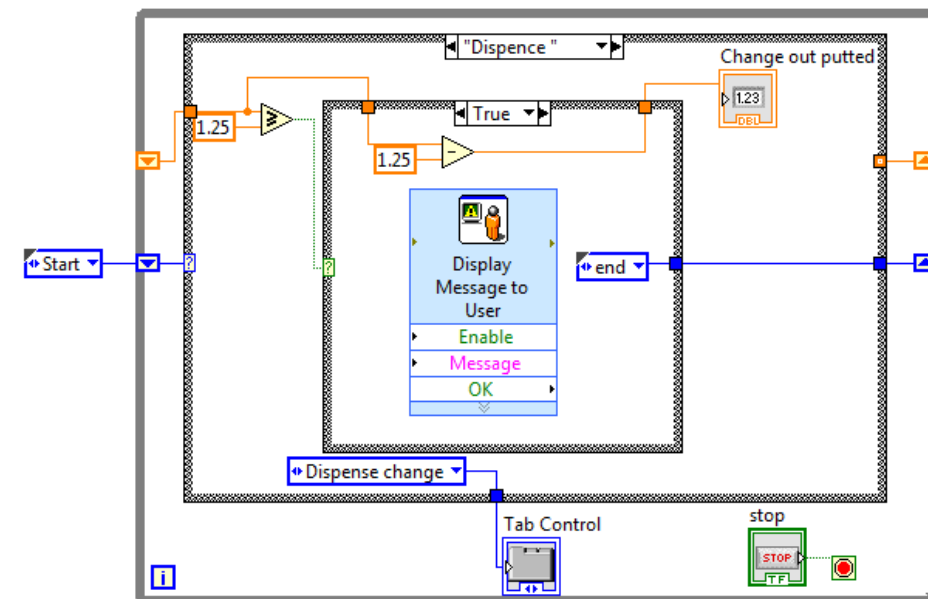
- ▶ Sledljivost izdelkom
- ▶ Nadzor nad kakovostjo
- ▶ Izboljšanje kakovosti
- ▶ Možnost nadaljnje analize podatkov pridobljenih s SCADA
- ▶ Pospešitev proizvodnega cikla

Rešitve za implementacijo SCADA sistema

- ▶ “Hišna” rešitev
 - ▶ Lahko je dvorezni meč
- ▶ Zunanje rešitve (orodja)
 - ▶ Ignition SCADA (Inductive automation)
 - ▶ LabView (National Instruments)
 - ▶ Intouch (Schneider Wonderware)
 - ▶ WinCC (Siemens)
 - ▶ RSVIEW (Rockwell)
 - ▶ Automation studio (B&R)

LabView

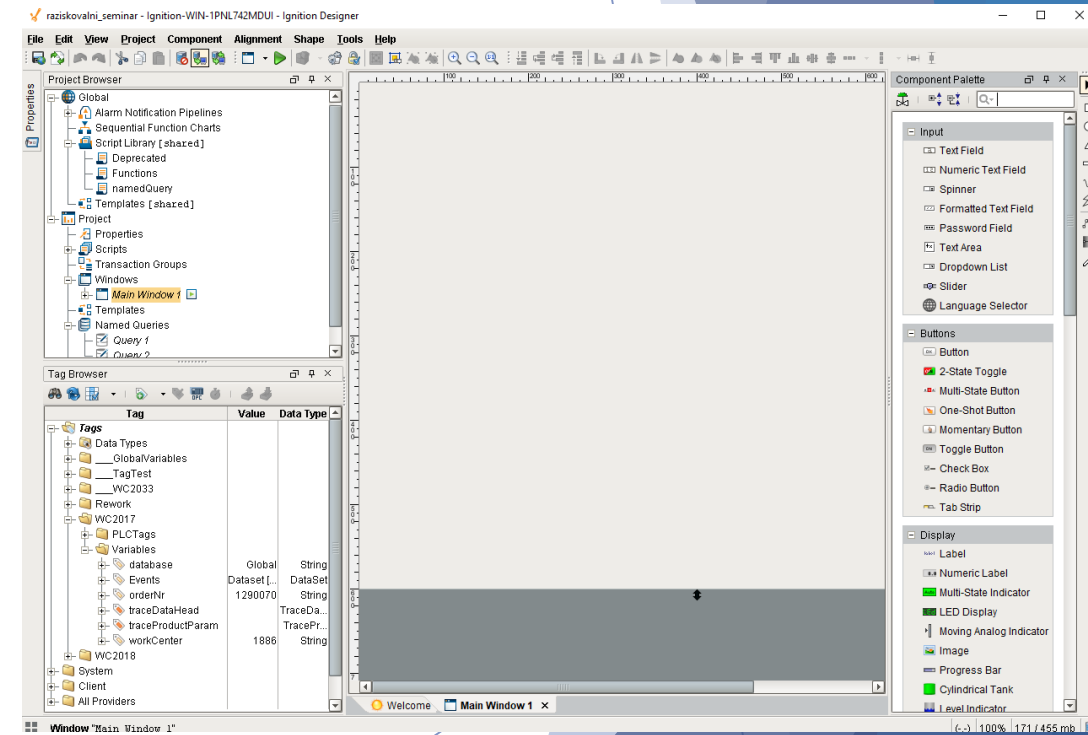
- ▶ Grafični programski jezik (NI kode)
 - ▶ Kar se da enostaven za uporabo neprogramerjem
 - ▶ Kar se da težek za uporabo programerjem
- ▶ Namen jezika enostavno zajemanje podatkov iz laboratorijskih naprav
 - ▶ Ni namenjen gradnji celotnega sistema
 - ▶ Vendar pa polna različica to omogoča
- ▶ Licenca letno
 - ▶ Polna različica 4000€
 - ▶ Profesionalna različica 7000€
 - ▶ Pogonjanje programa brezplačno na katerem koli računalniku



Ignition SCADA



- ▶ <https://inductiveautomation.com/scada-software/>
- ▶ Obsežna dokumentacija
- ▶ Možno brezplačno testiranje s podaljševanjem licence
 - ▶ Licenca glede na en server (okoli 10.000\$)
- ▶ Neodvisen od platform (napisan v Javi)
- ▶ Dobra podpora (tudi forum)
- ▶ Vključeni gonilniki za naprave + OPC UA server (licenca)
- ▶ Enostavna komunikacija z bazo
- ▶ Programiranje v Jython (Python)
- ▶ Uporablja ga veliko znanih podjetij



Hišna rešitev

- ▶ Težka za implementacijo
 - ▶ Zahteva ekipo programerjev
 - ▶ Začetek iz 0
- ▶ Popolna svoboda
 - ▶ Ni omejitev programskih orodij
- ▶ Po navadi manj stroškov z licenciranjem
- ▶ Primer
 - ▶ Back-end Java Spring
 - ▶ Front-end Angular

Implementacija SCADA sistema

- ▶ Namen raziskati in bolje spoznati SCADA sisteme
- ▶ Uporaba Ignition
- ▶ Proizvodna linija z delavnimi mesti, ki delujejo po enakem principu z izjemami

Shema sistema



- ▶ PLCji in prikazovalniki na liniji
- ▶ Lokalna podatkovna baza
 - ▶ Z njo se povezuje tudi MES sistem
 - ▶ Preslikava v globalno podatkovno bazo
- ▶ Splošno delovno mesto
- ▶ Druga delovna mesta

Zahteve delovanja na posamezni postaji

► Obvezne

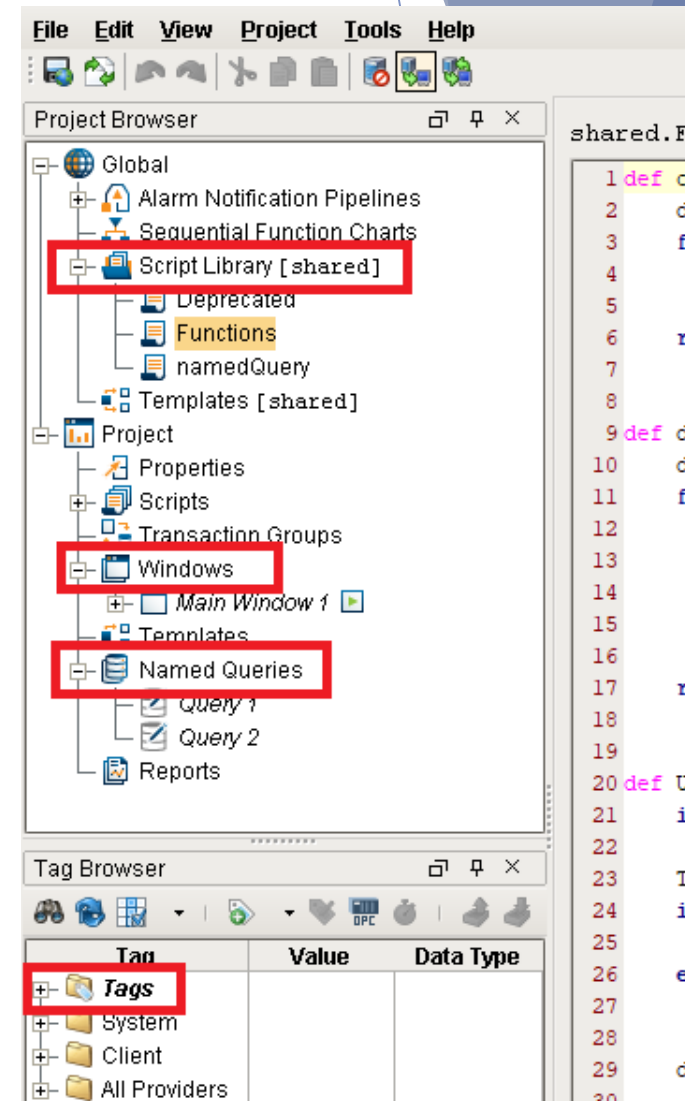
- 1) Dovoliti izdelku opravljanje operacije na trenutnem delovnem mestu ali opravljanje operacije zavrniti
- 2) Shraniti procesne parametre, ki so nastali tekom opravljanja operacije na tem delovnem mestu
- 3) Oceniti ali zabeležiti uspešnost opravljanja operacije na tem delovnem mestu
- 4) Zabeležiti delovno mesto na katerega naj polizdelek nadaljuje po opravljeni operaciji na tem delovnem mestu
- 5) Poskrbeti, da stroj pravilno deluje v odvisnosti od tipa izdelka na katerem opravlja operacijo

► Opcijske - le za operaterja linije

- I. Pregledovanje dokumentacije tipa izdelka, ki je trenutno v izdelavi na delovnem mestu
- II. Pregledovanje zgodovine trenutnega delovnega mesta
- III. Pregledovanje zgodovine izdelka, ki je trenutno v obdelavi na delovnem mestu

Načrtovanje in izvedba delovanja SCADA sistema

- ▶ Okno
- ▶ Zadovoljitev zahtev 1 - 4
 - ▶ s protokolom ob prihodu izdelka na delovno mesto
 - ▶ Sodelovanje s programerji stroja
 - ▶ Ob spremembi značke PLC-ja izvedba funkcije v Pythonovi skripti
 - ▶ Klic imenovane poizvedbe za poizvedovanje po bazi
- ▶ Zadovoljitev zahteve 5
 - ▶ podobno a le v redkih primerih
- ▶ Zadovoljitev opsijskih zahtev
 - ▶ Dokumentacija
 - ▶ Dostop preko URL-ja s pomočjo web browser komponente (plačljivo)
 - ▶ Pregled zgodovine
 - ▶ Imenovane poizvedbe ob pritisku gumba



Zaključek

- ▶ Podrobno spoznavanje in raziskovanje SCADA sistemov
- ▶ V primeru morebitnih vprašanj, le-te prosim zastavite
 - ▶ Hvala