Arkitektura Koadernoa ModelEditor

There is guidance within this template that appears in a style named InfoBlue. This style has a hidden font attribute that allows you to toggle whether it is visible or hidden in this template. Use the Microsoft® Word® menu **Tools > Options > View > Hidden Text** check box to toggle this setting. There is also an option for printing: **Tools > Options > Print**.

# Helburua

Dokumentu honek deskribatu egiten ditu proiektuan zehar, eta zehazki ModelEditor azpisistemaren atalean, filosofia, erabakiak, baldintzak, justifikazioak eta garrantzizkoa izan daitekeen edozer.

# Arkitektura helburuak eta filosofia

Sortutako editore grafikoa eta testu editorea erabiliz prozesuaren eredua sortzeko eta editatzekoa aukera emango du. Sistema honen ardura prozesu ingenieri rolak izango du.

Azpisistemak hurrengo helburu nagusiak ditu:

* Metodologien informazioa editatu EPF Composer erabiliz.
* Metodologien informazioa editatu editore grafikoa erabiliz.
* Metodologien informazioa editatu testu editorea erabiliz.
* Metodologien informazioa datu-basean gordetzeko SQL kodea sortu.

# Hipotesiak eta menpekotasunak

Metaereduen erabilerak malgutasuna ematen du proiektuaren hurbilpena aldatzeko edo etorkizunean beste hurbilpen batzuk probatzeko.

Definitutako garapen prozesuak software proiektuen elaborazioa gidatu eta kontrolatuko du. Prozesua aldatzeak sistemaren portaera eta datu-basea automatikoki aldatzea ekarriko du.

IO-System azpisistemaren arkitektura ModelEditor azpisistemarekin erlazionatuta dago. Izan ere, ModelEditor-ekin definitutako ereduaren datuak erabili beharko ditu. Hala ere, dependentzia ez da ProWF proiektuan bezain zuzena, ModelEditor sistemaren arkitektura abstraktuagoa delako.

Sistemarako tresna moduan Eclipse erabiltzen denez honekiko dependentzia izango du. Gainera, Java bertsioekiko dependentzia ere egongo da. Adibidez, EPF Composerekin derrigorrez Java 8 32 bit erabili behar da.

# Betekizun arkitekturalki esanguratsuak

Sistemaren betekizunak Ikuspegia eta Betebeharren Espezifikazioa dokumentuetan definitu dira.

# Erabakiak, mugak eta justifikazioak

ProMeta proiektuan sistemaren abstrakzio-maila igotzea erabaki da. Metodologien informazioa workflow lengoaian edo BPMNn eskuz definitu ordez, metaeredu eta ereduen bidez gordetzen da. Horrela, ereduaren informazioa berrerabil daiteke eta beste ereduetara eraldaketak egin. Ereduaren informazioa zuzenean kodea sortzeko ere erabil daiteke, beste metaereduren beharrik gabe.

Hurbilpen honek ziurtatzen du etorkizunean aurreko aukeretako edozein edo beste batzuk aukeratu ahal izatea. Izan ere, BPMNren eta workflow-lengoaiaren metaereduak izanik, eraldaketa egitea ez litzateke horrenbeste lan izango. Gainera, eredutik datu-baserako SQL kodea edo webgunerako HTML kodea sortzeko aukera ere egongo litzateke.

# Mekanismo arkitekturalak

* **Metodologiak editatu:** Metodologiak editatu EPF Composer erabiliz edo editore grafiko eta testuala erabiliz.
* **Metodologien webgunea sortu:** Metodologien webgunea sortu EPF Composer erabiliz.
* **Metodologien datu-basea sortu:** Metodologien informazioa datu-basean gorde SQL aginduekin. Ondoren, informazio hori IO-System azpisistemak erabiliko du.

# Geruzak edo arkitektura frameworka

* **EPF Composer:** Metodologiak modu grafikoan editatzeko aukera ematen du. Metodologiaren webgunea sortu daiteke bertatik.
* **Formatu Aldatzailea:** Metodologiaren formatua aldatzen du XMItik UMAra editore grafikoak erabili ahal izan dezan.
* **Editore Grafikoa:** Metodologiak modu grafikoan editatzeko aukera ematen du, metodologiaren jatorrizko ikonoak erabiliz.
* **Testu Editorea:** Metodologiak testu bidez editatzeko aukera ematen du, metaeredutik sortutako gramatika erabiliz.
* **Editore Sinkronizatzailea:** Metodologiaren informazioa editore batekin aldatutakoan bestea ere aldatzeaz arduratzen da.
* **Kode Sortzailea:** Metodologien informazioa datu-basean gorde ahal izateko SQL kodea sortzen du.

