

Arkitektura Koadernoa ModelEditor

1. Helburua

[Always address Sections 2 through 6 of this template. Other sections are recommended, depending on the amount of novel architecture, the amount of expected maintenance, the skills of the development team, and the importance of other architectural concerns.]

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

2. Arkitektura helburuak eta filosofia

[Describe the philosophy of the architecture. Identify issues that will drive the philosophy, such as: Will the system be driven by complex deployment concerns, adapting to legacy systems, or performance issues? Does it need to be robust for long-term maintenance?

Formulate a set of goals that the architecture needs to meet in its structure and behavior. Identify critical issues that must be addressed by the architecture, such as: Are there hardware dependencies that should be isolated from the rest of the system? Does the system need to function efficiently under unusual conditions?]

Sortutako editore grafikoa eta testu editorea erabiliz prozesuaren eredua sortzeko eta editatzeko aukera emango du. Sistema honen ardura prozesu ingeneriari rola izango du.

Azpisistemak hurrengo helburu nagusiak ditu:

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

3. Hipotesiak eta menpekotasunak

[List the assumptions and dependencies that drive architectural decisions. This could include sensitive or critical areas, dependencies on legacy interfaces, the skill and experience of the team, the availability of important resources, and so forth]

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 ereduen 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.

4. Betekizun arkitekturalki esanguratsuak

[Insert a reference or link to the requirements that must be implemented to realize the architecture.]

Sistemaren betekizunak Ikuspegia eta Betebeharren Espezifikazioa dokumentuetan definitu dira.

5. Erabakiak, mugak eta justifikazioak

[List the decisions that have been made regarding architectural approaches and the constraints being placed on the way that the developers build the system. These will serve as guidelines for defining architecturally significant parts of the system. Justify each decision or constraint so that developers understand the importance of building the system according to the context created by those decisions and constraints. This may include a list of DOs and DON'Ts to guide the developers in building the system.]

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

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

6. Mekanismo arkitekturelak

[List the architectural mechanisms and describe the current state of each one. Initially, each mechanism may be only name and a brief description. They will evolve until the mechanism is a collaboration or pattern that can be directly applied to some aspect of the design.]

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

7. Funtsezko abstrakzioak

[List and briefly describe the key abstractions of the system. This should be a relatively short list of the critical concepts that define the system. The key abstractions will usually translate to the initial analysis classes and important patterns.]

8. Geruzak edo arkitektura frameworka

[Describe the architectural pattern that you will use or how the architecture will be consistent and uniform. This could be a simple reference to an existing or well-known architectural pattern, such as the Layer framework, a reference to a high-level model of the framework, or a description of how the major system components should be put together.]

- **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:** Metodologiaren informazioa datu-basean gorde ahal izateko SQL kodea sortzen du.

9. Arkitektura ikuspegiak

[Describe the architectural views that you will use to describe the software architecture. This illustrates the different perspectives that you will make available to review and to document architectural decisions.]

Recommended views

- **Logical:** Describes the structure and behavior of architecturally significant portions of the system. This might include the package structure, critical interfaces, important classes and subsystems, and the relationships between these elements. It also includes physical and logical views of persistent data, if persistence will be built into the system. This is a documented subset of the design.
- **Operational:** Describes the physical nodes of the system and the processes, threads, and components that run on those physical nodes. This view isn't necessary if the system runs in a single process and thread.
- **Use case:** A list or diagram of the use cases that contain architecturally significant requirements.

