

Informatika Fakultatea

Informatika Ingeniaritzako Gradua

- Workflow-lengoaia ▪
- Eskuliburua ▪

Julen Rojo Raño

2020/06/24



Edukien taula

| | |
|--|---|
| 1. Eskuliburuaren helburua | 3 |
| 2. Softwarearen bizi-zikloen ezaugarriak | 3 |
| 2. DOT lengoaia..... | 3 |
| 3. Grafoaren formak eta koloreak | 5 |
| 4. Lengoaiaren deskribapena | 6 |
| 5. Adibideak..... | 8 |



1. Eskuliburuaren helburua

Dokumentu honen helburua *Workflow Editor* azpisisteman erabili behar den **workflow-lengoaia zehaztea** da. Proiektuko **prozesu sortzaileak workflow-ereduak sortzerako orduan jarraitu beharreko arauak dira**.

Erabiltzaileak edozein momentutan izan ahal dituen arazo eta kezka bertan konpontzeko gai izan beharko litzateke, bestela nire kontaktua hemen jartzen dut:

julenrojo12@gmail.com

2. Softwarearen bizi-zikloen ezaugarriak

Softwarearen bizi-zikloak, normalean, hurrengo elementuetaz osatuta daude:

- **Faseak:** errepikakorra izan daiteken iterazioa da, bere barruan aktibitateak eta haien erlazioak definitzen dira.
- **Aktibitateak:** jarduera multzo bat da, bertan jarduerak eta haien arteko erlazioak definitzen dira.
- **Jarduerak:** rol zehatz batek bete behar dituen ekintzak, ekintzak lan-produktuak edo artefaktuak izango dira normalean.
- **Artefaktuak:** jarduera baten emaitzak dira, lan-produktuak. Normalean, dokumentuak dira.
- **Galderak eta erabakiak:** bizi-zikloan zehar agertu ahal diren galderak eta erabakiak. Horiek bizi-zikloaren bide eta ekintzen gaineko eragina dute.

2. DOT lengoaia

Workflow-lengoaia DOTen azpi-lengoaia da, beraz, funtsezkoa da *DOT* lengoairen arauak ezagutzea eta ikastea.

Jarraian, laburki lengoaia horrek grafoak definitzeko erabiltzen duen gramatika deskribatuko da:

- **Digraph** (Grafo zuzendua): bere barruan nodoak eta ertzak definitzen dira.
 - Egitura:

```
digraph izena{  
    NODOAK ETA ERTZAK  
}
```



- **Grafoa:** grafoaren atributuak aldatzeko balio du.
 - Egitura:

```
graph [atributuak];
```
 - Atributu batzuk:
 - `fontsize`: letraren tamaina.
 - `fontname`: letra motaren izena.
 - `label`: grafoaren irudian agertuko den testua.
 - `Bgcolor`: atzeko kolorea.
- **Nodoa:** ertzen bidez konektatutako informazioa.
 - Egitura

```
izena [atributuak];
```
 - Atributu batzuk:
 - `shape`: nodoaren forma.
 - `label`: nodoaren irudian agertuko den testua.
 - `fillcolor`: nodoaren barruko kolorea.
- **Ertza:** bi nodoen arteko erlazioa da, gezia.
 - Egitura:

```
Nodo1 -> Nodo2 [atributuak];
```
 - Atributu batzuk:
 - `style`: estiloa.
 - `weight`: lodiera.

Beraz, grafo batek honako egitura izango du:

```
digraph grafoaren_izena{  
    graph[grafoaren atributuak];  
  
    Nodo1 -> Nodo2 [ertzaren atributuak];  
  
    Nodo1 [nodoaren atributuak];  
    Nodo2 [nodoaren atributuak];  
}
```



Hala ere, lengoaia honen xehetasun guztiak harrapatzeko, hurrengo dokumentua irakurtzea gomendatzen da:


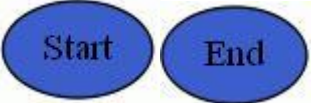

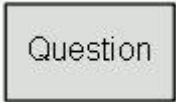
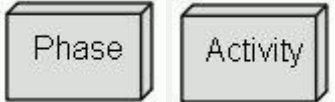


<https://www.graphviz.org/pdf/dotguide.pdf>

3. Grafoaren formak eta koloreak

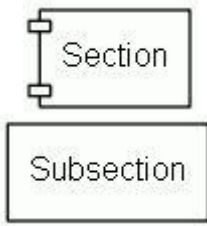
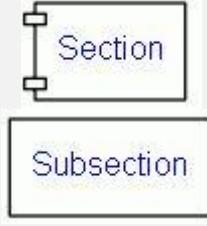
Grafoak izango dituen formak eta koloreak aldagarriak dira, *workflow*-eredua prozesatzeko orduan ez dute eraginik izango.

Hala ere, OpenUP metodologiako [webgunean](#) agertzen diren formak eta koloreak erabiltzea gomendatzen da, software proiektuen garatzaileekiko hurbiltasuna bilatuz.

Hona hemen, *OpenUP* metodologiaren itxura izateko erabili ahal diren atributuak:

| Elementu mota | Atributuak | Irudia |
|---|--|---|
| Grafoaren atzeko kolorea | bgcolor = lemonchiffon |  |
| Workflow eta azpi-workflowen hasiera eta bukaera nodoak | shape= ellipse, fillcolor = royalblue3, style = filled |  |
| Erabaki nodoa | shape=diamond, fillcolor = white, style = filled |  |
| Galdera nodoa | shape=box, fillcolor = #DCDCDC, style = filled |  |
| Fasea eta aktibitatea | shape=box3d, fillcolor = #DCDCDC, style = filled |  |
| Jarduera eta rola | shape= cds, fillcolor = #ffff80, style = filled |  |
| Artefaktua | shape=note, fillcolor = white, style = filled |  |



| | | |
|--|---|---|
| Sekzio eta azpi-sekzioa (txantilo gabe) | shape= component, fillcolor = white, style = filled EDO shape= box, fillcolor = white, style = filled |  |
| Sekzio eta azpi-sekzioa (txantiloiarekin) | shape= component, fillcolor = white, style = filled, fontcolor= blue EDO shape= box, fillcolor = white, style = filled, fontcolor= blue |  |

4. Lengoiaren deskribapena

Workflow-lengoiaren helburua, ikusgarritasun garbia izatea eta aldi berean inferentzia motorrak prozesatzeko prest egotea da. Beraz, garrantzitsua da ikusgarritasunaren eta negozio-logikaren arteko banaketa egitea, inferentzia motorrarekin zerikusia duena grafoaren iruditik kanpo geratuz.

Hori dela eta, nodoen izenaren eta bere *"label"* atributuaren arteko bereizketa egin da. Nodoren izenean inferentzia motorrak harrapatu behar duen informazioa jarriko da eta *"label"* atributuan, ostera, grafoaren irudian agertuko den testua.

Nodoaren izenak hurrengo egitura izango du beti:

```
"PROZESU_MOTA NODO_MOTA KODE_MULTZOA NODOAREN_IZENA | {BERTSIOA
edo ROLA edo KARDINALITATEA edo URLa}"
```

- PROZESU_MOTA:** egungo *workflow* edo azpi-*workflowaren* mota da. Hurrengo balioak izan ahal ditu:
 - Methodology: hasierako workflowa da, non bizi-zikloaren faseak agertzen diren.
 - Phase: fase baten azpi-workflowa da, non fasearen aktibitateak agertzen diren.
 - Activity: aktibitate baten azpi-workflowa da, non aktibitatearen barruan dauden jarduerak agertzen diren.
 - Task: jarduera baten azpi-workflowa da, non jardueran bete behar diren artefaktuak agertzen diren.
 - Artifact: artefaktu baten azpi-workflowa da, non artefaktuaren sekzioak agertzen diren.



- **NODO_MOTA:** nodoaren mota da. Hurrengo balioak izan ahal ditu:
 - Start: prozesu edo azpi-prozesuaren hasiera.
 - End: prozesu edo azpi-prozesuaren amaiera.
 - Phase: nodoak fase motako azpi-prozesua du.
 - Activity: nodoak aktibitate motako azpi-prozesua du.
 - Task: nodoak jarduera motako azpi-prozesua du.
 - Artifact: nodoak artefaktu motako azpi-prozesua du.
(Hurrengoak artefaktuen ekintzak adierazteko balio dute)
 - Section: nodoa idatzi beharreko sekzioa da.
 - SectionWT: (With Template) nodoa idatzi beharreko sekzioa da eta txantiloia du.
 - UploadGraph: nodoa igo beharreko grafoa da. Batzuetan, artefaktuak ez dira idatzi behar, eta irudi, grafo edo ereduak sortu behar dira.
 - Join: elkartze erromboa da.
 - Question: galdera da.
 - Decision: erabaki erromboa da.

- **KODE_MULTZOA:** informazioa datu-basean gordetzeko erabiliko diren identifikatzaile numerikoak. 7 identifikatzaile izango dira beti.

method_id phase_id activity_id task_id role_id artifact_id section_number

- method_id: metodologia identifikatzeko zenbakia.
- phase_id: fasea identifikatzeko zenbakia.
- activity_id: aktibitatea identifikatzeko zenbakia.
- task_id: jarduera identifikatzeko zenbakia.
- role_id: rola identifikatzeko zenbakia.
- artifact_id: artefaktua identifikatzeko zenbakia.
- section_number: sekzioaren zenbakia (float)

GARRANTZITSUA:

- Workflowan agertzen diren lehenengo fase, lehenengo aktibitate eta lehenengo jarduerak 1 kodea izan behar dute, bigarren jarduera/aktibitate/faseak 2 kodea eta horrela gainerakoekin. Honela, workflowaren elementuen arteko hierarkia finkatzeko eta web-aplikaziotik jardueren arteko ordena interpretatzeko.
 - Identifikatzailearen bat behar ez denan 0 jarriko da.
 - Workflow eta azpi-workflowen arteko lotura egiteko ezinbesteko da azpi-workflowaren START nodoak bere goragoko nodoaren kode multzo berdina izatea.
 - Sekzioaren zenbakia *float* motako da, sekzio eta azpi-sekzioak bereizteko. Adibidez: sekzio bat → 2, azpi-sekzio bat → 2.1
- **NODOAREN_IZENA:** datu-basean gordeko den nodoaren izena (karaktere kate bat baino gehiago izan daiteke).



- “|” **ikurra**: ikur honen ostean nodo batzuek behar duten informazio gehigarria sartuko da. Ikur honen ostean doana hutsik jarri daiteke informazio gehigarrik ez denean behar.
 - **BERTSIOA**: metodologiaren START eta END nodoetan agertuko da bakarrik, workflow-ereduaren bertsioa adierazteko. Adibidez: Pre-Alpha, Alpha, Beta...
 - **ROLA**: jardueren (task) kasuan agertuko da bakarrik, rolaren izena adierazteko
 - **KARDINALITATEA**: JOIN motako nodoen kasuan agertuko da, bere kardinalitatea adieraziko da, adibidez: bi sarrera, irteera bat → 2 1
 - **URLa**: SectionWT motako nodoetan agertuko da bakarrik, sekzioaren txantiloia non gordetzen den adierazteko (path)

5. Adibideak

Atal honetan, workflow-lengoaia hobeto ulertzeko balio duten adibide batzuk agertuko dira.

- Hasierako workflowaren Start eta End nodoen izenen adibideak:

```
"Methodology Start 1 0 0 0 0 0 0 OpenUp Lifecycle | Alpha"
```

```
"Methodology End 1 0 0 0 0 0 0 OpenUp Lifecycle | Alpha"
```

- Workflow eta azpi-workflow baten arteko lotura:

```
Workflow: "Methodology Phase 1 1 0 0 0 0 0 Inception |"
```

```
Azpi-workflow: "Phase Start 1 1 0 0 0 0 0 Inception |"
```

- Aktibitate baten azpi-workflowan agertutako jarduera nodo baten izenaren adibidea:

```
"Activity Task 1 1 1 1 1 0 0 Define Vision | Analyst"
```

- Jarduera baten azpi-workflowan agertutako artefaktu nodo baten izenaren adibidea:

```
"Task Artifact 1 1 4 1 3 1 0 Architecture Notebook |"
```

- Artefaktu baten azpi-workflowan agertutako sekzio nodo baten izenaren adibidea:

```
"Artifact SectionWT 1 1 4 1 3 1 1 Purpose | Templates/ArchitectureNotebook/Purpose.txt"
```