

# Verkefni 5

Skilafrestur 1. nóvember, kl. 22:00

## 1. Markmið

Í þessu verkefni er aðallega unnið með tvennt, prófanir og villumeðhöndlun á gögnum. Eins og venjulega bætið þið við lausnina með því að forrita viðbótarnotkunartilvik. Þetta er næstsíðasta verkefnið og því ættu fá eða engin notkunartilvik að vera eftir við lok verkefnisins.

## 2. Framkvæmd og afurðir

### 2.1 Villumeðhöndlun á gögnum

Setja á skilyrði á tilviksbreytur í einum eða fleiri módel klösum (notið `javax.validation.constraints.*`). Grípa á villurnar í notendaviðmótinu og controller klasa og bregðast við þeim með leiðbeiningum til notanda. Farið var í efnið í viku 8 og 9. **Gerið lista yfir skrár (.java .jsp .html o.s.frv.) sem þið hafið breytt fyrir þennan lið. Látið fylgja með skýringar fyrir hverja skrá.**

Í mörgum skrá, t.d. model-klösum voru breytingarnar mjög sambærilegar og því teknar saman í eina skýringu.

Skrá	Lýsing
Model-klasur:	
Ad	Viðeigandi skilyrði sett á þær breytur sem notandi slær inn, skilyrðið NotEmpty var mikið notað. Tilgreint er hvaða villumeldingu úr messages.properties á að sýna.
Event	
InfoNord	
InfoBoard	
News	
Controller-ar:	
AdController	Sjá um að redirect-a aftur á sömu síðu ef einhverjar upplýsingar eru vitlausar, tapa ekki gögnunum á milli redirect-a svo notandi getur enn séð hvað hann sló inn og lagfært. Gögnin sem notandi sló inn (og eru vitlaus) eru geymd fyrir eitt redirect.
InformationController	
NewsController	
Templates:	
ad > form.html	Flash-message eru birt ef notandi slær inn eitthvað vitlaust.
ad > ad.html	
home > index.html	
information > information.html	
information > form.html	
information > boardForm.html	
news > form.html	
news > newsDetail.html	
event > form.html	
event > eventDetailhtml	
Viðbótar klasar:	
messages.properties	Villumeldingarnar sjálfar
AppConfig.java	Sér um að villumeldingar í messages.properties sé notaðar í staðinn fyrir default villumeldingar

## 2.2 Prófanir

Þið eigið að sýna að þið hafið framkvæmt nokkrar prófanir á forritinu ykkar. Farið verður í efnið í viku 9 og 10. Ekki þarf að skila neinni skýrslu fyrir prófanirnar, en þið þurfið að skjala prófanaklasana mjög vel og nota skráanöfnin í liðum a)-e)

- a) Athugið hvort forritið keyri og að einhver af controller klösunum verður til. Notið AssertJ (assertThat() o.fl. ) til að athuga lögmæti. Kallið test klasann SmokeTest.java
- b) Sendið URL á test web client og athugið hvort svarið er löglegt. Kallið test klasann HttpRequestTest.java
- c) Framkvæmið prófanir án þess að þurfa að kalla á þjóninn (server) en á laginu þar fyrir neðan þar sem Spring meðhöndlanr HttpRequest beiðnir og afhendir controllernum þau. Notið [@AutoConfigureMockMvc](#) Kallið test klasann ApplicationTest.java
- d) Framkvæmið prófanir á weblayer. Gerið svipað og í c) en notið @WebMvcTest. Sýnið hvernig þið prófið tvo mismunandi controller-a. Kallið test klasann WebLayerTest.java
- e) Framkvæmið prófanir á controller og service. Kallið test klasann WebMockTest.java

## 2.3 Forritið

a) **Sýnið í töflu** hvaða notkunartilvikum er lokið (tilgreinið verkefnisnúmer) og hverjum er ólokið.

Sjá skjalið Use Cases.

b) Haldið áfram að viðhalda issues í github og gerið nýtt milli-release.

c) Skoðið vel niðurstöður úr kóðarýni úr verkefni 4 og lagið klasana. **Gerið grein fyrir helstu lagfæringum.**

Dagsetningu bætt við í klasana newsController og newsService.

Engar breytingar gerðar í UserController og UserServiceImpl þar sem engar athugasemdir voru.

Engar breytingar gerðar í AboutController og AboutServiceImpl, eina athugasemdin sem var gerð var að engin comment eru í AboutServiceImpl, ástæðan fyrir því að það eru engin comment er að AboutServiceImpl hefur interface-ið AboutService sem hefur comment fyrir allt sem er implementað í AboutServiceImpl.

## 2.4 Verkaskipting innan teymis og frávik ef einhver eru

Hver og einn lýsir stuttlega hvað hann eða hún lagði af mörkum til framkvæmdar og/eða afurða.

Ef teymið gat ekki framkvæmt alla verkþætti sem beðið var um eða skilað afurðum, segið stuttlega frá því hér og segið hver ástæðan er.