

GLUE-CODE

Tesztautomatizálás köztes rétegének fejlesztése



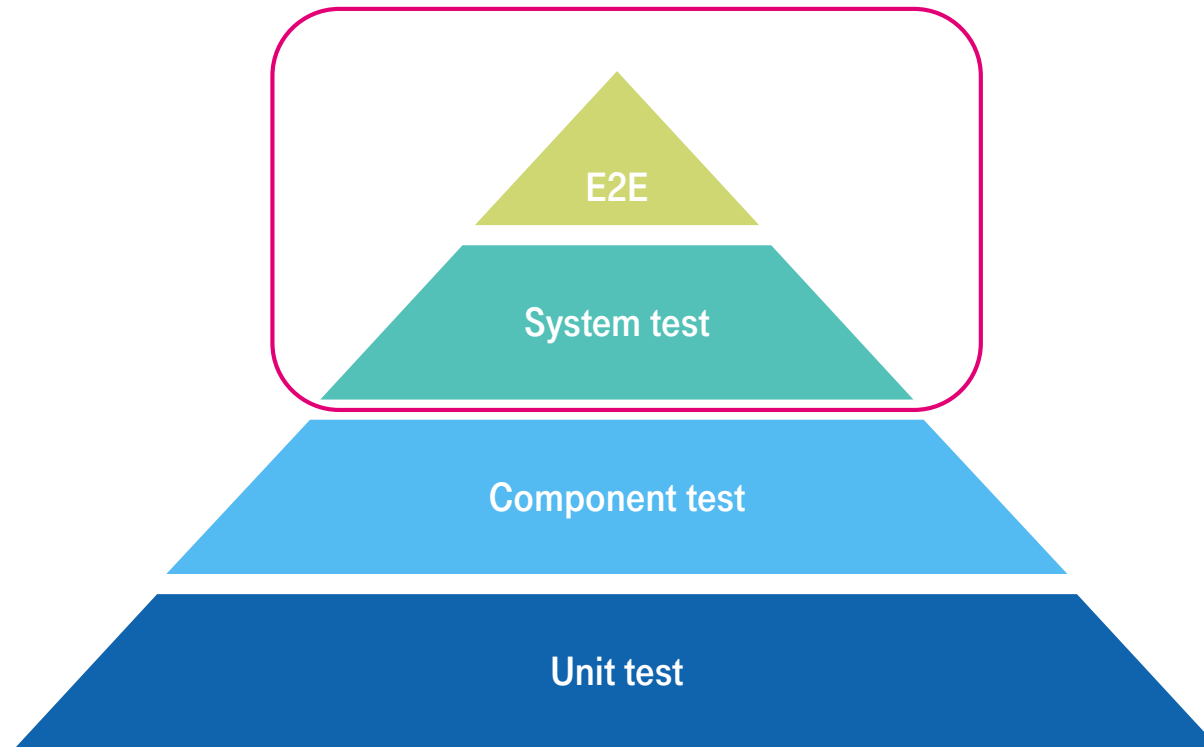
LIFE IS FOR SHARING.

**A KÖVETKEZŐKBEN BÁRMIT IS MONDOK,
AZ NEM A **TELEKOM** HIVATALOS
ÁLLÁSPONTJA, CSUPÁN A SZAKMAI
TAPASZTALATAIMAT ÉS PIACI
TRENDEKET OSZTOK MEG**



LIFE IS FOR SHARING.

BDD HELYE A TESZTELÉSI PIRAMISBAN



MI IS AZ A GLUE-CODE?

Test
Scenario



Application
under
Test

HOGYAN RAGASSZUNK - 1. FEJEZET

IMPERATÍV STÍLUS

Forgatókönyv: Az email címmel torteno bejelentkezes kivlasztasa

Amennyiben Az EMAIL belepesi mod gombra kattintunk

Akkor Az "Email alapú belépés" felirat megjelenik

Forgatókönyv: Bejelentkezes a belepesi modnak megfelelo azonosito es jelszo parossal

Amennyiben Beírom a(z) "email" beviteli mezőbe, "korta@gmail.com"

És Beírom a(z) "jelszó" beviteli mezőbe, "admin1234"

És Rákattintok a "Belépés" gombra

Akkor Az "Email címhez tartozó hibás jelszó" felirat megjelenik

HOGYAN RAGASSZUNK - 1. FEJEZET

IMPERATÍV STILUS

Test Scenario



Application
under
Test



HOGYAN RAGASSZUNK - 1. FEJEZET

IMPERATÍV STÍLUS

PRO

- Kevés glue code
- Egyértelmű forgatókönyv
- Gyors framework fejlesztés
- Kevés mondat – nem lehet újat készíteni

KONTRA

- Technikai részletek
- Hosszú forgatókönyv
- Nem moduláris – monolit
- Nem agilis



HOGYAN RAGASSZUNK - 2. FEJEZET

NINCS FORGATÓKÖNYV, CSAK KÓD

```
scenario('add feature', function() {
    return action.open(URL)
        .then(action.render)
        .then(action.fill('#add_new_feature_input', 'new feature'))
        .then(action.click('#add_item_to_feature_list_btn'))
        .then(action.scenarioEnd(''))
})

scenario('delete feature', function() {
    return action.open(URL)
        .then(action.render)
        .then(action.clickItemInSelectList('#feature_list a', 'new feature'))
        .then(action.click('#delete_item_from_feature_list_btn'))
        .then(action.click('#delete_feature_dialog_btn'))
        .then(assert.selectList('#feature_list a', ["asfasdfasdf", "egy", "egyujfeature", "harom", "jjjjjjj", "proba"]))
        .then(action.scenarioEnd(''))
})

scenario('end', function() {
    return action.open(URL)
        .then(action.render)
        .then(action.scenarioEnd(''))
})
```



HOGYAN RAGASSZUNK - 2. FEJEZET

NINCS FORGATÓKÖNYV, CSAK KÓD



HOGYAN RAGASSZUNK - 2. FEJEZET

NINCS FORGATÓKÖNYV, CSAK KÓD

PRO

- Egyértelmű
- Moduláris

KONTRA

- Nincs „human readable” rész
- Technikai részletek a tesztben

HOGYAN RAGASSZUNK - 3. FEJEZET

DEKLARATÍV STÍLUS



HOGYAN RAGASSZUNK - 3. FEJEZET

DEKLARATÍV STÍLUS

Forgatókönyv: Az email címmel torteno bejelentkezés kiválasztása

Amennyiben Belepesi EMAIL belepesi_mod kiválasztása kerül

Akkor Megjelenik a(z) EMAIL belepesi modnak megfelelo azonositas kepernyo

Forgatókönyv: Bejelentkezés a belepesi modnak megfelelo azonosito es jelszo parossal

Amennyiben Belepes a(z) EMAIL belepesi modnak megfelelo Zsolt.email azonositoval es hibas jelszoval

Akkor A megfelelo hibauzenet jelenik meg a hibas jelszora vonatkozóan

HOGYAN RAGASSZUNK - 3. FEJEZET

DEKLARATÍV STÍLUS

PRO

- Egyértelmű forgatókönyv
- Moduláris
- Test-first
- Jól szeparálható szerepkörök
- Feature file
- Mondatok
- Osztály definíciók
- Kód

KONTRA

- Nehéz jó kódot írni
- Elszabadul a mondatok száma
- Nagy fegyelem kell hozzá
- Affinitás a kódoláshoz



MILYEN RAGASZTÓT HASZNÁLJUNK?

- SOLID alapelvek programozásból
- POM vagy repository
- Behave
 - TypeParser
- Scenario Outline
- Extra konfiguráció
- Teszt szelekciót támogató tag-ek

```
'labels' : {  
    'test_level' : TestLevel.SYSTEM.value,  
    'test_type' : [ TestType.FUNCTIONAL.value, TestType.REGRESSION.value ],  
    'test_case_id' : [ 'E2ETEST-T1666' ],  
    'project' : Project.RUBIK.value,  
    'systems' : [System.ONESHOP.value],  
    'test_object' : TestObject.ONESHOP.value,  
    'workflow' : [WorkFlow.BELEPES.value]  
}
```

```
class TelekomAdatokSzamlakuldesiMod ( Enum ):
```

```
    VEGYES = 'vegyes (elektronikus, papíralapú)'  
    ELEKTRONIKUS = 'elektronikus, e-számlás'  
    PAPIRALAPU = 'papíralapú, csekkes'
```

kerteszlatta27, 2 months ago • [E2ETEST-88]-szamlazasi-informaciok-modositasa

```
register_type( FolyoszamlaTipusa = TypeBuilder.make_enum( FolyoszamlaTipusa ) )  
register_type( SzamlazasiMod = TypeBuilder.make_enum( SzamlazasiMod ) )  
register_type( SzamlaFormatum = TypeBuilder.make_enum( SzamlaFormatum ) )  
register_type( FizetesiMod = TypeBuilder.make_enum( FizetesiMod ) )  
register_type( InvalidMezo = TypeBuilder.make_enum( InvalidMezo ) )  
register_type( TelekomAdatokSzamlakuldesiMod = TypeBuilder.make_enum( TelekomAdatokSzamlakuldesiMod ) )
```

```
@step( 'Kivalasztásra kerül a "{folyo_tipus:FolyoszamlaTipusa}" folyoszamla' )  
def step_impl( c, folyo_tipus ):  
    SzamlazasiAdatokModositasa.folyoszamla_kivalasztasa( folyo_tipus )
```

POM ÉS REPOSITORY

- Page Object Model:
 - Selenium
 - Mutató
- Repository
 - Ranorex
 - Teljes térkép



```
PassWord = POMELEMENT(  
    SelectorType.ST_ID,  
    'login-form-password'  
)  
  
BelepesMod = POMELEMENT(  
    SelectorType.ST_XPATH,  
    "//ui-link-button[contains(@data-test, '{belepes_mod}' ) ]",  
    is_dynamic = True  
)  
  
BelepesAzonositoInput = POMELEMENT(  
    SelectorType.ST_XPATH,  
    "//ui-input-text[contains(@data-test, '{belepes_mod}' ) ]//input",  
    is_dynamic = True  
)
```

SOLID ALAPELVEK

- **Single-responsibility principle**
 - Az egyik osztály megkéri a másikat
 - Senkinek semmi köze a belső adatstruktúrához

```
assert_map = {
    StickyCheck.HUSEGIDO           : cls.husegido_lekerdezes,
    StickyCheck.MOBIL_DIJCSOMAG    : cls.mobil_dijcsomag_nev_lekerdezes,
    StickyCheck.INTERNET_DIJCSOMAG : cls.internet_dijcsomag_nev_lekerdezes,
    StickyCheck.KESZULEK_NEV       : cls.keszulek_nev_lekerdezes,
    StickyCheck.KESZULEK_AR        : cls.keszulek_ar_lekerdezes,
    StickyCheck.KESZULEK_KEDV_AR   : cls.keszulek_kedvezmenyes_ar_lekerdezes,
    StickyCheck.KESZULEK_AR_ATVETELKOR : cls.atvetelkor_fizetendo_keszulek_ar_lekerdezes,
    StickyCheck.HAVONTA_FIZETENDO_AR : cls.havonta_fizetendo_osszeg_lekerdezes,
```

```
Sticky.check( {
    StickyCheck.MOBIL_HAVONTA_FIZETENDO_OSSZEG : 7600
} )
```


SOLID ALAPELVEK

- Open-closed
 - Nyitott a bővítésre (Enum, konfiguráció bővítés)
 - Zárt a kód módosításra
 - Öröklés – mutációk

```
class CimAttributumokEnum( Enum ):  
    TELEPULES       = "telepules"  
    KOZTERULET      = "kozterulet"  
    HAZSZAM         = "hazszam"  
    LAKAS           = "lakas"  
    ALLANDO_CIM     = "allando_cim"  
    TELEPITESI_CIM  = "telepitesi_cim"  
    POSTAZASI_CIM   = "postazasi_cim"  
    HRSZ            = "hrszt"  
    PF              = "postafio"
```

```
cim_attributumok_pom = {  
    CimAttributumokEnum.TELEPULES : CimPOM.Telepules,  
    CimAttributumokEnum.KOZTERULET : CimPOM.Kozterulet,  
    CimAttributumokEnum.HAZSZAM : CimPOM.Hazszam,  
    CimAttributumokEnum.LAKAS : CimPOM.Lakas,  
    CimAttributumokEnum.HRSZ : CimPOM.HRSZ,  
    CimAttributumokEnum.PF : CimPOM.PF  
}
```

```
@classmethod  
def cimadat_rogzitese( cls, cim_enum_nev = CimAttributumokEnum.TELEPULES.name, cim_ertekek = '' ):  
    """A cím formában egyetlen textboxba való címadat beírása történik.  
  
    Args:  
        cim_enum_nev ( TELEPULES, KOZTERULET, HAZSZAM, LAKASSZAM, HRSZ ): cim_attributumok_pom alapján melyik szelektort választjuk.  
        cim_ertekek ( string ): maga a címadat.  
    """  
    cls.cim_attributumok_pom[ cim_enum_nev ].scroll_clear_settext_wait( cim_ertekek )  
    cls.cimlista_elso_elemenek_kivalasztasa()
```

SOLID ALAPELVEK

- Interface segregation principle:
 - Rövid, tömör függvények célzott funkcionalitással
 - Ne ágyazzunk egymásba osztályokat fölöslegesen (SQL példa: kapcsoló tábla)
 - Spagettifikáció ellen

```
@classmethod
def kijelolt_szamlak_mentesenek_elokeszítése( cls, tipus = SzamlaTipusEnum.HAVI, altipus = SzamlaAltipusEnum.TELEKOM, index = None ):

    # sorszam és folyoszamla mindig 1 elemű tömb lesz, mivel a többi nincs lenyitva
    cls.sorszam = cls.szamla_sorszam_tipus_altipus( tipus = tipus.value, altipus = altipus.value ).get( strict_pom_handling = False ).
    cls.osszeg = cls.szamla_ar_tipus_altipus( tipus = tipus.value, altipus = altipus.value ).get( strict_pom_handling = False, index
    cls.folyoszla = cls.szamla_folyo_tipus_altipus( tipus = tipus.value, altipus = altipus.value ).get( strict_pom_handling = False ).

    if index != None:
        cls.osszeg = [ cls.osszeg ]

    cls.osszeg = [ SzamokKiolvasasa.szamok_szovegbol_olvasasa_int( osszeg ) for osszeg in cls.osszeg ]
```

TESZT ELŐKÉSZÍTÉSE

- Funkcionális egységek azonosítása
- Mondatok megtervezése
- Lehetséges paraméterek feltérképezése
- Feature file-ok megtervezése teszt tervezési technikák mentén
- Osztályok elkészítése funkcionális egységenként
- Lehetséges paraméterek enumerációba szervezése
- Üres függvények elkészítése, default paraméterekkel



TESZT ELŐKÉSZÍTÉSE



TESZT ELŐKÉSZÍTÉSE

Forgatókönyv vázlat: Jelszó indikátorok ellenőrzése

Amennyiben Megadom az új <jelszo> jelszot

Akkor A megadott jelszo nem felel meg a kovetkezo kriteriumnak <nok>

És A megadott jelszo megfelel a kovetkezo kriteriumnak <ok>

És A tovabblepes nem engedelyezett a hibas jelszo miatt

Példák: Kriteriumok

/jelszo	/ok	/nok
''	SZOKOZ	MIN8,SZAM,NAGYBETU,KISBETU
'a'	SZOKOZ,KISBETU	MIN8,SZAM,NAGYBETU
'aA '	KISBETU,NAGYBETU	MIN8,SZAM,SZOKOZ
'aA2'	SZOKOZ,KISBETU,NAGYBETU,SZAM	MIN8
'aA34d'	SZOKOZ,KISBETU,NAGYBETU,SZAM	MIN8
'8888888 '	MIN8	SZAM,NAGYBETU,KISBETU,SZOKOZ
'3 '	SZAM	NAGYBETU,KISBETU,SZOKOZ,MIN8
'A '	NAGYBETU	MIN8,SZAM,KISBETU,SZOKOZ

```
@step('A megadott jelszo nem felel meg a kovetkezo kriteriumnak {nok}')
def step_impl(c, nok ):
    for elem in nok.split(',') :
        Bejelentkezes.jelszo_erosseg_ellenorzes(indikator = elem, state = 'error')
```

```
@step('A megadott jelszo megfelel a kovetkezo kriteriumnak {ok}')
def step_impl(c, ok ):
    for elem in ok.split(',') :
        Bejelentkezes.jelszo_erosseg_ellenorzes(indikator = elem, state = 'ok')
```

```
@classmethod
def jelszo_erosseg_ellenorzes( cls, indikator = JelszoErossegIndikatorEnum.KISBETU, state = 'error' ):
    assert cls.jelszo_erosseg_indikator( indikator = indikator.value ).get_attribute( 'class' ) == state
```

```
jelszo_erosseg_indikator = LoginPOM.JelszoErossegIndikator
```

```
JelszoErossegIndikator = POMElement(
    SelectorType.ST_XPATH,
    "//ui-badge[contains(@data-test, '{indikator}' ) ]//div",
    is_dynamic=True
)
```

TESZT ELŐKÉSZÍTÉSE

Objektum

- Factory szükséges
- Nincs interferencia
- Könnyebb bővítés (workflow engine)

Classmethod

- Könnyű megérteni
- Szabályok betarthatóbbak
- Interferencia kockázata



TESZT ELŐKÉSZÍTÉSE

Objektum

- Factory szükséges
- Nincs interferencia
- Könnyebb bővítés (workflow engine)

Classmethod

- Könnyű megérteni
- Szabályok betarthatóbbak
- Interferencia kockázata



Q&A

- SOLID
- POM
- Behave
- Python
- Enum
- Test-firsts

