

Sissejuhatus tarkvaraarendusse

Martti Raavel

martti.raavel@tlu.ee

Teine loeng

- Meenutame eelmist loengut
- Koduste tööde tagasiside
- Issuest branchi loomine
- .gitignore
- Koodi kvaliteet
- Persoonad
- Kasutajalood
- Stsenariumid
- Kasutaja teekond
- Harjutus nõuete väljaselgitamiseks

Eelmine loeng

- Tarkvara
- Tarkvaraarendus
- Tarkvaraarenduse elutsükel
- Git
- Github
- Github issue

Kodustest töödest

Issuest branchi loomine 1

Test issue #4

Edit

New issue

Open

mrtrvl opened this issue 3 days ago · 0 comments



mrtrvl commented 3 days ago

This is a test issue



mrtrvl added **Homework** **Programmeerimine-I** labels 3 days ago



mrtrvl self-assigned this 3 days ago



mrtrvl closed this as completed 3 days ago



mrtrvl reopened this 2 minutes ago



Add a comment

Write

Preview

H

B

I

≡

<>

🔗

≡

≡

≡

📎

@

🗨️

↩️

📧

Notifications

Customize



Unsubscribe

You're receiving notifications because you modified the open/close state.

Issuest branchi loomine 2

Create a branch for this issue

×

Branch name

4-test-issue

Repository destination

Change branch source

TLUHK-RIF16-Martti/Martti-Raavel

What's next?

☐ Open in codespace

☐ Checkout locally

☒ Open branch with GitHub Desktop

Beta

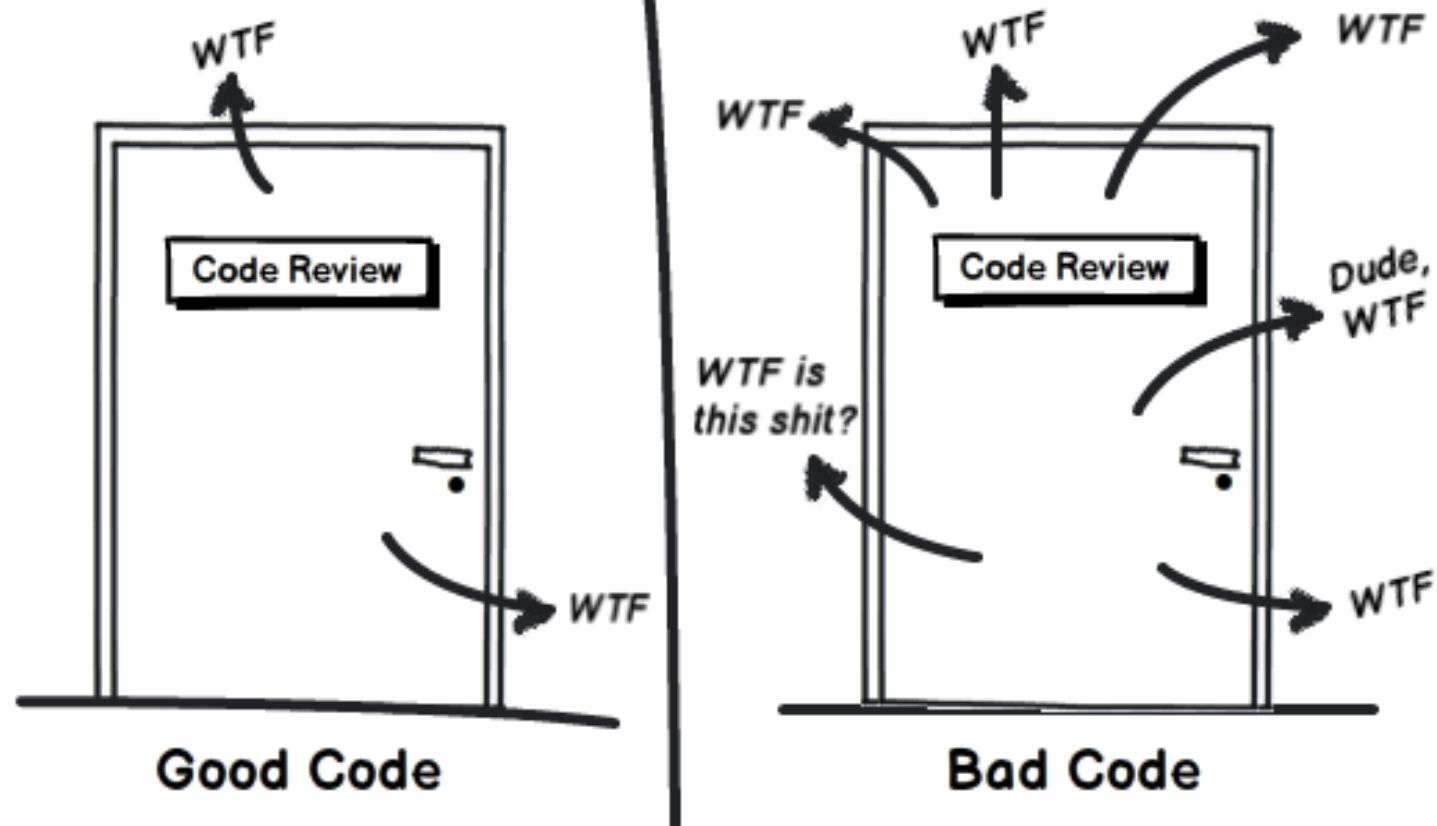
Share feedback

Create branch

.gitignore

`.gitignore` on konfiguratsioonifail, mida Git kasutab, et määrata, millised failid ja kataloogid tuleks versioonihaldussüsteemist välja jätta.

Code Quality Measurement: WTFs/Minute



Koodi kvaliteet 2

Koodi kvaliteet viitab koodibaasi üldisele kvaliteedile ja hooldatavusele. See hõlmab koodi erinevaid aspekte, sealhulgas selle

- loetavus (*Readability*)
- järjepidevus (*Consistency*)
- lihtsus (*Simplicity*)
- hooldatavus (*Maintainability*)
- töökindlus (*Reliability*)

Kvaliteetset koodibaasi on aja jooksul lihtsam mõista, muuta ja hooldada, mis võib säästa arendusmeeskondade aega ja ressursse.

Koodi loetavus (*Readability*)

Kood peaks olema kergesti loetav ja arusaadav. See peaks olema hästi struktureeritud ja organiseeritud, järjekindlate nimetamisreeglite, vahede ja taandega.

Koodi järjepidevus (*Consistency*)

Kood peaks olema stiililt ja lähenemisviisilt ühtne. See hõlbustab teistel arendajatel koodi mõistmist ja muutmist.

Koodi lihtsus (*Simplicity*)

Kood peaks olema lihtne ja arusaadav. See vähendab vigade tõenäosust ning muudab koodi aja jooksul muutmise ja hooldamise lihtsamaks.

Hooldatavus (*Maintainability*)

Kood peaks olema aja jooksul hooldatav. See tähendab, et nõuete muutumisel peaks seda olema lihtne muuta, parandada ja värskendada.

Töökindlus (*Reliability*)

Kood peaks olema töökindel ja prognoositav. See peaks töötama ootuspäraselt ja käsitlema vigu selliselt, et need ei põhjusta programmi töös tõrkeid.

Kuidas koodi kvaliteeti tagada?

- Järjepidev koodi vormindamine
- Kommentaarid ja dokumentatsioon
- Ühiktestimine
- Koodiülevaatused
- Vigade käsitlemine
- Koodi refaktoreerimine

Nõuete väljaselgitamine 1

Harjutame Programmeerimine I aines loodava rakenduse jaoks nõuete paikapanemist, prioritseerimist ja disaini.

Kuidas me seda teeme?

Nõuete väljaselgitamine 2

- Persoon
- Kasutajalood
- Stsenariumid
- Kasutaja teekond

Persoona 1

- Näidis-kasutaja, tüüpilise lõppkasutaja arhetüüp, kes esindab konkreetset kasutajagruppi;
- Luuakse kliendi paremaks mõistmiseks – mis teda motiveerib, mis ärritab, mida ta vajab?
- Ei asenda kasutajauuringuid, vaid tugineb nendel!
- Aitab disaini loomisel silmas pidada lõppkasutaja vajadusi, nt eristada olulisi funktsionaalsusi ebaolulistest.

Persoona 2

Tarkvaraarenduse ja kasutajakogemuse kujundamise kontekstis on **persoona** väljamõeldud tegelane, kes esindab teatud tüüpi kasutajat. **Persoona** luuakse uurimistöö ja andmete põhjal reaalse kasutajate kohta ning neid kasutatakse abistamiseks disaineritel ja arendajatel oma kasutajate mõistmiseks ja nende vajadustest, eesmärkidest ja käitumisest aru saamiseks.

Persoona 3

Persoona sisaldab tavaliselt teavet kasutaja demograafia (näiteks vanus, sugu ja amet), samuti üksikasjad nende huvide, motivatsiooni ja käitumise kohta. **Persoona** võib sisaldada ka teavet kasutaja eesmärkide, pettumuste ja valupunktide kohta, samuti üksikasju nende eelistatud suhtlemismeetodite kohta tarkvara või rakendusega.

Persoona 4

Persoonasid kasutatakse empaatia tekitamiseks ning disainerite ja arendajate abistamiseks, mis põhinevad kasutaja vajadustel ja eelistustel. Mõistes eri tüüpi kasutajate vajadusi ja käitumist, saavad disainerid luua liideseid ja kogemusi, mis on kohandatud nende sihtrühmale, mille tulemuseks on tarkvara, mis on kaasahaaravam, kasutatavam ja tõhusam.

Persoona 5

Persoona loomiseks kogub meeskond tavaliselt teavet selliste uurimismeetodite abil nagu:

- küsitlused;
- intervjuud;
- reaalseste kasutajate vaatlus.

Persoonade jaotumine

- **Proto-persoona:** ei tugine uutel uuringutel. Eesmärk on kiiresti luua visand või ülevaade olemasolevatest eeldustest!
- **Kvalitatiivne persoona:** tugineb väikesemahulistel uuringutel (nt intervjuu, vaatlus). Eesmärk on klientide süvitsi mõistmine!
- **Kvantitatiivne persoona:** tugineb laiaulatuslikel uuringutel (nt statistika, küsitlused). Eesmärk on andmete isikustamine!

Kasutajalood 1

Kasutajalugu on lihtne ühelauseline tarkvararakenduse funktsiooni või funktsionaalsuse kirjeldus, mis on kirjutatud lõppkasutaja vaatenurgast. Kasutajalood on levinud tehnika, mida kasutatakse agiilses tarkvaraarenduses, et kirjeldada rakenduse nõudeid nii, et neid on lihtne mõista ja prioriteete seada.

Kasutajalood 2

Registreeritud kasutajana tahan oma parooli lähtestada, et saaksin parooli unustamise korral oma kontole tagasi juurdepääsu

Kasutajalugude kirjutamisel

- **Tuvastage kasutaja:** alustage kasutaja või rolli tuvastamisest, kelle vaatenurgast lugu on kirjutatud. See võib olla konkreetne kasutaja või kasutajate rühm, kellel on sarnane vajadus.
- **Kirjeldage toimingut.** Järgmisena kirjeldage toimingut, mida kasutaja soovib teha. See peaks olema lihtne ja selge kirjeldus selle kohta, mida kasutaja teha tahab.
- **Nimetage kasu:** lõpuks märkige kasu, mida kasutaja saab selle toimingu sooritamiseest. See peaks kirjeldama, miks kasutaja soovib toimingu sooritada ja mida ta loodab saavutada.

Kasutajalood 3

1. Külaliskasutajana tahan, et saaksin tooteid sirvida, et leida midagi, mida osta.
2. Registreeritud kasutajana soovin, et saaksin oma ostukorvi tooteid lisada, et saaksin jälgida, mida osta tahan.
3. Registreeritud kasutajana soovin, et mul oleks võimalik oma kaupade eest tasuda, et saaksin ostu lõpule viia.
4. Registreeritud kasutajana soovin näha oma tellimuste ajalugu, et saaksin näha, mida olen varem ostnud.
5. Registreeritud kasutajana soovin, et saaksin kirjutada tootearvustusi, et saaksin oma kogemusi teiste kasutajatega jagada.
6. Registreeritud kasutajana soovin, et saaksin oma konto andmeid muuta, et saaksin oma andmed ajakohasena hoida.

Stsenaarium 1

- Persoona-põhine kirjeldus toote/teenuse kasutamisest;
- Ei keskendu ärilistele või tehnoloogilistele nüanssidele ega disainile, vaid kirjeldab kasutajakogemust!
- Sisaldab:
 - tegija
 - motivatsioon
 - eesmärk või kavatsus
 - tegevus
 - tulemus või lahendus

Stsenaarium 2

Tarkvara arendamisel viitab stsenaarium konkreetsele kasutusjuhtumile või olukorrale, millega kasutaja tarkvaraga suheldes võib kokku puutuda. Stsenaariume kasutatakse selleks, et kirjeldada, kuidas kasutaja tarkvaraga suhtleb ja milliseid samme nad peavad konkreetse ülesande või eesmärgi täitmiseks tegema.

Stsenaarium 3

Tüüpiline stsenaarium võib sisaldada kasutaja kirjeldust, nende eesmärke või samme, mida nad peavad nende eesmärkide saavutamiseks võtma, ja kogu asjakohast teavet tarkvara või keskkonna kohta, kus seda kasutatakse.

Näiteks võib e-kaubanduse veebisaidi stsenaarium kirjeldada kasutajat, kes soovib toodet osta, sealhulgas samme, mida nad vajavad toote leidmiseks, nende ostukorvi lisamiseks ja kassas protsessi lõpuleviimiseks.

Näidis-stsenaarium

Tegija	Tiina..
Motivatsioon	..soovib õppida, kuidas oma ettevõttele ise veebileht luua
Eesmärk	Ta tahab leida koolitust, mis sobiks hinnalt, oleks suunatud algajatele ja toimuks sobivatel kuupäevadel
Tegevus	Tiina valib koolituste nimekirjast talle sobiva kategooria, milleks on IT. Seejärel sorteerib ta koolitusi hinna ja toimumisaja alusel ning tutvub nende sisuga
Tulemus	Ta leiab nimistust sobiva koolituse ja registreerib osalejaks

Stsenaariumi loomisel

- Väldi tehnoloogia või disaini liigset kirjeldamist – stsenaariumi eesmärk on olla disaini inspiratsiooniks, mitte fikseeringuks!
- Ära kirjelda persoonat ja tema eesmärke ülemäärase detailsusega, keskendu vaid olulisele!

Kasutaja teekond 1

- Alternatiiv / laiendus stsenaariumile, loob tervikliku ülevaate kasutajakogemusest.
- Esmalt määratakse persoona ja stsenaarium, samuti eesmärgid.
- Teisena on kaardistuse keskmes kogemuse visualiseering, mis jaotatakse mitmeks etapiks.
- Kolmanda osa moodustavad tähelepanekud ja võimalused edasiseks ning vastutajad.

Kasutaja teekond 2

Kasutaja teekond, mida tuntakse ka kui kasutaja teekonnakaarti või kliendi teekonnakaarti, on tööriist, mida kasutatakse tarkvaraarenduses, et visualiseerida kasutaja kogemus toote või teenusega suheldes. See on sisuliselt kasutaja kogemuste samm-sammuline kaart, alates kasutaja esialgsest teadlikkusest tootest või teenusest, sellega suhtlemisest ja lõpuks otsuseni kasutamise jätkamisest või selle lõpetamisest.

Tüüpiliselt võib sisaldada järgmiseid etappe

- **Teadlikkus:** kasutaja saab tootest või teenusest teada reklaami, sotsiaalmeedia või suusõnalise info kaudu.
- **Uuringud:** kasutaja viib läbi uuringuid toote või teenuse kohta lisateabe saamiseks, sealhulgas arvustusi lugedes, veebisaiti külastades või videot vaadates.
- **Registreerumine või ost:** kasutaja otsustab toote või teenuse kasutamise.
- **Onboarding:** kasutajat juhendatakse toote või teenuse kasutusele võtmisel.
- **Kasutamine:** kasutaja kasutab toodet või teenust regulaarselt, puutudes kokku kasutajakogemust paremaks muutvate omaduste ja funktsionaalsutega.
- **Tugi:** kasutajal võib tekkida toodete või teenusega probleeme või probleeme ning otsib arendaja või tugimeeskonna tuge.
- **Uuendamine või katkestamine:** kasutaja otsustab, kas jätkata toote või teenuse kasutamist, tellimuse uuendamist või kasutamise katkestamist.

Kasutaja teekond 3

Kaardistades kasutaja teekonna sel viisil, saavad tarkvaraarendajad ja disainerid tuvastada valdkonnad, kus saab parandada kasutajakogemust, näiteks lihtsustades *onboardingu* protsessi, täiustades toote funktsioone või pakkudes paremat tuge.

Moodustame grupid (võib ka individuaalselt)

Harjutus

Kodune töö

- [ESLint](#) rakendamine Programmeerimine I kodutöödes