## Lisa 1

## Hankelepingu 3-9/... - SKAIS2 Väikearendused 2021-3 tehniline kirjeldus

## Mõisted ja lühendid

|  |  |
| --- | --- |
| **Mõiste/Lühend** | **Kirjeldus** |
| SKA | Sotsiaalkindlustusamet |
| TEHIK | Tervise ja Heaolu Infosüsteemide Keskus |
| SKAIS1 | Sotsiaalkaitse infosüsteem (olemasolev) |
| SKAIS2 | Sotsiaalkaitse infosüsteem (hõlmab Ametnikurakendust ja Iseteenindust) |
| EBS | Majandustarkvara Oracle E-Business Suite |
| AVE | Abivahendite sooduseraldamise haldamise süsteem |

## Ülevaade

SKA kasutab ülesannete täitmiseks sotsiaalkaitse infosüsteemi, mis koosneb SKAIS1, EBS, AVE ja SKAIS2 rakendustest. SKAIS2s on praeguseks kasutusel elatisabiteenus, puude raskusastme tuvastamine, sotsiaaltoetuse teenus, peretoetused, isa täiendav vanemahüvitis ja abivahendite teenus. Sotsiaalkindlustusameti teenuste kasutajate jaoks on arendatud Sotsiaalkindlustusameti iseteenindus, kus kasutaja saab enda jaoks vajalikke toiminguid teha lihtsalt ja kiirelt. Iseteeninduses kuvatakse praegu isikuandmetega seotud infot, isikule maksete teostamise infot, perehüvitiste teenuseid ja alates oktoobrist abivahendite teenust. SKAISis olevate teenuste (va abivahendi teenuse) finantsarvestust, raamatupidamiskandeid ja väljamakseid teostatakse Oracle E-Business Suite majandustarkvaras. Loodud SKAIS2 süsteem tekitab igakuiselt suurel hulgal korduvaid hooldusjuhtumeid ning stabiilsus, jõudlus ja kasutajamugavus ei vasta kasutajate ootustele.

## Lepingu eesmärk

Lepingu eesmärgiks on realiseerida SKAIS2s olemasoleva funktsionaalsuse arendusvajadused, mis vähendavad hooldusjuhtumite mahtu ning tagavad kasutajasõbralikuma süsteemi ja muude täiendavate arendusvajaduste realiseerimine ning sellega seonduvad tööd (nt testimine, juurutamine jne).

## Tellitavad tööd

## Lepinguga tellitakse SKAIS2 funktsionaalsuse parandamist ja täiendamist ning SKAIS2 teenuste arendamisega seonduvad tööd (nt automaatestide loomine, andmete hägustamine ja muud arendustööd efektiivse ja kvaliteetse arenduse tagamiseks). Tööd tuleb üle anda hiljemalt 14.01.2022.

## Tööde loetelu tekib eelnevalt kokkulepitud backlogi alusel ja hallatakse jiras:

## Eelanalüüsi suunatud ülesanded (analüüsi task), mille tulemusel valmivad mahuhinnangutega kasutajalood (story’d) ja tehnilised eeldustööd (technical taskid);

## Arendusse suunatud kasutajalood (story’d) ja tehnilised eeldustööd (technical taskid) backlog staatuses vastavalt tööde järjekorrale (rank).

## Tööprotsess ja tulemid

## Järgnevalt täpsustatakse Täitjaga sõlmitud raamlepingus (nr 3-9/2307-1) kirjeldatud tingimusi tööprotsessi, dokumenteerimise, töökorralduse, testimise ja mittefunktsionaalsete nõuete osas.

## Tööprotsess

* Tööraamistik peab tuginema etapiviisilisele planeerimisele, võimaldades arendusmeeskonna prioriteete jooksvalt muuta;
* Tööraamistik peab võimaldama süsteemset ülevaadet iga kokkulepitud töö saavutustest;
* Tööraamistik peab võimaldama arendusmeeskonna tulemuslikkuse (kasvu) jälgimist ajas;
* Tööraamistik peab tagama, et arendusmeeskond tegeleb omal initsiatiivil enda tulemuslikkuse parandamisega ja parendusettepanekute esitamisega.
* Täpsem protsess lepitakse kokku tööde käigus.

## Dokumentatsioon

Vastavalt alapunktis 4 kirjeldatud skoobile tuleb töö käigus luua ja täiendada SKAIS2 dokumentatsiooni, milleks on minimaalselt:

* Funktsionaalsuse ja äriprotsesside kirjeldused kokkulepitud kujul (nt BPMN standard, diagrammid ja kirjeldused Confluence’is) koos ärireeglitega, mida peab olema hiljem võimalik täiendada järgnevates etappides lisanduva funktsionaalsuse alusel ning siduda tervikliku protsessi- ja ärireeglite kirjeldusega.
* Tehniline dokumentatsioon (nt paigaldusjuhendid, Confluence keskkonnas).
* Andmemudel.

Dokumentatsiooni täpsed liigid ja sisu formaat lepitakse kokku tööde käigus.

## Töökorraldus

Projekti töökorraldus toetub kodukorrale. Täitja peab tagama piisava kaasamise ja ülevaadete andmise Tellijale.

## Tööde üleandmine

Arendustööde lähtekoodi tarne toimub pideva integreerimise (continuous integration) teel.

Tööde järjekord ja UAT tagasiside andmise aeg lepitakse kokku tööde käigus taasesitamist võimaldavas vormis (Tellija ja Täitja poolt).

Tööd annab Täitja üle jooksvalt, vastavalt nende valmimisele – Tellija kontrollib seejärel töö nõuetekohasust ning annab vajadusel tähtaja töö parandamiseks. Töö loetakse vastuvõetuks alates selle Live paigaldusest peale edukat vastuvõtu testimist.

Täitja teab, et tellija võib igat üle antud funktsionaalsust paigaldada toodangusse ning täitja peab tagama, et lõplikult valmimata funktsiooni protsessid on võimalik toodangust välja lülitada (feature flags). Funktsioonide sisse-välja lülitamise võimekus lepitakse Tellija ja Täitja vahel kokku tööde käigus.

## Nõuded automaattestimisele

Loodav lähtekood peab olema kaetud 80% ulatuses ühiktestidega, 50% ulatuses integratsiooni/süsteemi/”end-to-end” automatiseeritud testidega. Testidega kaetakse serveripoolne ja kasutajaliidese kood. Kaetavuse kontrolliks ja raporteerimiseks seatakse Tellija koodihoidla keskkonnas (Gitlab) üles pideva integratsiooni töövoog, mis seadistatakse koostoimima SonarQube koodi ja testide kaetavuse analüsaatoriga.

Automaattestid peavad üle antavas lähtekoodis continuous-integration’i töövoo edukalt läbima. Samuti peab Täitja peab hoolt kandma, et varasemalt loodud testid jäävad töö lõpetamisel edukalt läbitavaks. Nõuete muutumisel peab Täitja viima testid vastavusse uute nõuetega või neid eemaldama, kui selliseid nõudeid enam ei eksisteeri.

Kasutajaliidese testimise osakaal kogu testimise mahust peab olema mõistlik (mitte ületades 30%), rakendades seda kriitilisele funktsionaalsusele (lepitakse tööde käigus kokku). 50% kasutajaliidese testimisest peab olema automatiseeritud ja korduvkasutatav tööde käigus kokkulepitud raamistikul (nt Selenium).

## Koormustestimised

### Iseteenindus peab toetama paralleelset minimaalselt 200 päringut (siinkohal on oodatav arendajapoolne hinnang, kas see on piisav päringute maht, vajadusel tõsta arvu).

### API teenustele ning keskmine päringu teenindamise aeg ei tohi olla suurem kui 2 sekundit.

### Nõude täitmise kontrolliks tuleb luua koormustestid. Koormustestide tegemine lepitakse Tellija Täitja poolt kokku tööde käigus.

### Nõudest eraldi tuleb vaadelda staatiliste failide allalaadimist – need tuleb http serveris puhverdada.

## Mittefunktsionaalsed nõuded

Mittefunktsionaalsete nõuete täitmisel tuleb lähtuda järgnevast:

* Uue funktsionaalsuse loomisel tuleb lähtuda raamlepingus (nr 3-9/2307-1) ja selle lisades kirjeldatud nõuetest.
  + Täpsemad mittefunktsionaalsed nõuded, millest lähtuda, asuvad Confluence keskkonnas: [Mittefunktsionaalsed nõuded arendustele](https://wiki.sm.ee/display/AV/SKAISi+MFN)

## Nõuded kiirusele ja latentsusele

Allpool kirjeldatud nõuded kehtivad ainult SKAIS2 sisemisele rakendusele ehk väliste süsteemide vasteaegasid ei loeta ajalise nõude sisse.

* Tühja veebivormi avanemine ei tohi kesta üle 2s.
* Andmetega veebivormi avanemine ei tohi kesta üle 4s.
* Menetluse jaoks andmete otsimine ei tohi kesta üle 5s.
* Menetluse andmete salvestamine ei tohi kesta üle 4s.
* Menetluse jaoks andmete värskendamine ei tohi kesta üle 4s.
* Üle interneti infosüsteemi kasutamisel võib lisanduda 2s viivitus.

Teenuse kvaliteedi tõstmiseks on poolel õigus teha teisele poolele ettepanekuid kiiruse ja latentsuse nõuete muutmiseks. Muudatuste korral kirjeldatakse kasutuslugude vastuvõtukriteeriumidesse mittefunktsionaalse nõudena.

## Tulemid

Valminud tööde tulemid antakse üle kord kuus tööde üleandmise-vastuvõtmise aktiga, tööde tulemid peavad sisaldama:

* Viiteid dokumentatsioonile Confluence’is;
* Viited tehtud töödele (Jira piletitena);
* Viiteid koodile GitLabis;
* Koodianalüüsi, mis sisaldab väljavõtet TEHIKu SonarQube keskkonnast, kust on näha testide kaetavuse protsent.