SKAISi MFN

Versioon: 1.0			
Käesolev dokument määrab kvaliteedi- ja mittefunktsionaalsed nõuded uutele infosüsteemidele ning nende dokumentatsioonile.			
Käesolevat dokumenti tuleb vaadata kui arenduste kvaliteedi- ja mittefunktsionaalsete nõuete põhidokumenti. Põhidokumendi ja viidatud dokumentide erisuste puhul tuleb lähtuda põhidokumendis kirjeldatust. Põhidokumendis viidatud Sotsiaalministeeriumi poolt koostatud dokumentide ja kolmandate osapoolte poolt koostatud dokumentide erisuste puhul tuleb lähtuda Sotsiaalministeeriumi poolt loodud dokumentides kirjeldatust.			
Kui mõnda nõuet ei ole võimalik või otstarbekas täita, tuleb selle mittetäitmise fakt ja põhjendus välja tuua pakkumuse esitamisel.			
Nõudeid tuleb järgida ka olemasolevate infosüsteemide versiooniuuendustel nii palju kui versiooniuuenduse käigus võimalik.			
Nõude nr.	**Nõude sisu**	**Seletused**	**Kostamise eest vastutaja**
1. Vastavus üldistele standarditele			
1.1	Lahendus peab olema kooskõlas riigi IT koosvõime raamistiku nõuetega.	https://www.mkm.ee/sites/default/files/riigi_it_koosvoime_raamistik.pdf	Arendaja
1.2	Lahenduse X-tee teenused peavad vastama RIA nõuetele.	https://www.ria.ee/ee/xtee-juhendid.html	Arendaja
1.3	Lahendus peab vastama Sotsiaalministeeriumi IT-profiilile.		Arendaja
1.4	Rakendus peab olema kirjutatud arvestades selle rakenduse poolt töödeldavatele andmetele määratud ISKE turvaklassi nõudeid.	SKAIS2 ISKE turvaklass on K2T2S2. https://www.ria.ee/ee/iske-kkk.html	Arendaja
1.5	Lahendus peab vastama veebide koosvõime raamistikule.	https://www.mkm.ee/sites/default/files/veebide_raamistik.pdf	Arendaja
1.6	Veebirakenduse kasutajaliides peab vastama vähemalt WCAG 2.0 tasemele AA.	http://www.w3.org/TR/WCAG20/	Arendaja
1.7	Veebipõhine kasutajaliides peab ühilduma täielikult standarditega HTML 5 ja CSS 3	Valideerimiseks kasutatakse vastavaid validaatoreid: http://validator.w3.org/ Kui on tegu olemasoleva süsteemi edasiarendusega, siis tuleb järgida olemasolevat HTML ja CSS versiooni. HTML valideerimisel arvestatakse sellega, et SKAIS infosüsteemides kasutusel olev Angular javascript raamistik lisab HTML atribuute, mis ei vasta HTML5 standardile. Sellest lähtuvalt eristatakse HTML valideerimisel Angular – ja HTML spetsiifilisi vigasid. Angular spetsiifilised HTML valideerumise vead ei kuulu parandamisele. Valideerimise tulemustest parema ülevaate saamiseks saab kasutada w3 validaatori filtreerimis funktsionaalsusi. CSS valideerimisel võetakse aluseks profiil CSS level 3 + SVG, brauseri tootjate spetsiifiliste (vendor prefix) css reegleid käsitletakse kui hoiatusi ja nende kasutamine on aktsepteeritav. Kui tekib vajadus vanemate brauserite toetamiseks kasutada mitte valideeruvat CSS-i siis selles lepitakse eraldi kokku.	Arendaja
1.8	ID-kaardiga allkirjastamisel on eelistatud veebipõhine digidoc-teenuste kasutamine.	Arendaja loodud lahenduse dokumentatsioonis (nt detailanalüüs vms) peab olema välja toodud digidoc-teekide versioonid ja kasutuskohad.	Arendaja
1.9	Veebirakendus peab probleemideta läbima OWASP ASVS baasil põhineva testi.	Kui pole arenduse eraldi kokku lepitud teisiti, siis on OWASP ASVS 3.0 tasemeks 2 (https://www.owasp.org/index.php/Category:OWASP_Application_Security_Verification_Standard_Project). Kinnise lähtekoodiga kommertstoote kasutamisel ei eeldata ligipääsu kinnisele lähtekoodile. Tellija poolset turvatestimist teostab kolmas sõltumatu pool. Selline esmane kolmanda poole turvatestimine tellitakse tellija finantseeringul. Ilmnenud vigade korral ja peale nende parandamist peab järeltestimise rahaliselt kompenseerima arendaja, kui tellija vastava nõudmise esitab.	Arendaja
1.10	Krüptoalgoritmide ja räsifunktsioonide kasutamisel tuleb järgida uusimat RIA kodulehel avaldatud krüptograafiliste algoritmide kasutusvaldkondade ja elutsükli uuringut, st lahendustes ei tohi kasutada uuringus väljatoodud ebaturvalisi algoritme ja võtmepikkuseid.	Arendaja loodud lahenduse dokumentatsioonis (nt detailanalüüs vms) tuleb välja tuua kasutatavad krüpto- ja räsialgoritmid, nende võtmepikkused, kasutuskohad, sh SSL sertifikaatide kasutuskohad. Värskeima uuringu leiab aadressilt https://www.ria.ee/ee/iske-dokumendid.html	Arendaja

1.11	Andmete edastus peab välisvõrgu liikluses olema kaitstud kasutades turvalisi ja üldteada andmeedastusprotokolle.		Arendaja
1.12	Infosüsteem peab kasutama serveri kellaaega ja ajatsooni.		Arendaja
1.13	Süsteemi edasiarendamisel/loomisel peab arvestama selle võimaliku laiendamisega nii andmemahtude, kui ka kasutajate arvu osas.		Arendaja
1.14	Rakendus peab olema tehniliselt tükeldatud vastavalt loogilisele jaotusele. Saadud osised peavad olema eraldi versioneeritavad ja paigaldatavad.	Näiteks, kui rakendusel on eraldi turvakontekstidega liidesed ametnikule ja kodanikule, peab rakendus olema jagatav kaheks eraldi liidesekomponendiks ning nende mõlema poolt kasutatavaks andmebaasiks.	Arendaja
1.15	Avalike e-teenuste loomisel peab arvestama valitsusasutusele kehtestatud visuaalse identiteedi stiilijuhisega.	https://www.valitsus.ee/et/eesmargid-tegevused/valitsusasutuste-visuaalse-identiteedi-stiilijuhis	Arendaja
1.16	Avalike e-teenuste loomisel peab arvestama valitsusasutustele kehtestatud iseteeninduskeskkonna raamistikuga.	https://www.mkm.ee/sites/default/files/iseteeninduskeskkondade_raamistik_08.07.2015.pdf https://www.mkm.ee/sites/default/files/iseteeninduskeskkondade_raamistiku_kasutatavuse_nouded_dets.pdf	Arendaja
2. Nõuded rakenduse arhitektuurile			
2.1	Rakenduse, andmebaasi ja kolmanda osapoole komponentid peavad olema sellised, mille eluea lõpp (EOL) pole teadaolevalt vähem kui 2 aasta pärast.	Arendaja loodud lahenduse dokumentatsioonis (nt detailanalüüs vms) peab olema välja toodud kasutatavate komponentide nimetused ja versioonid. Versiooni eluea lõppu ei loeta võrdseks terve komponendi eluea lõpuks, st versiooni tugi võib aeguda, kui uus versioon on välja lastud.	Arendaja
2.2	Tulevase ja olemasolevate infosüsteemide platvormid (rakendusserver, andmebaas, kolmanda osapoole komponendid) ja topoloogia peab olema enne reaalse arenduse algust infosüsteemide halduse osakonna juhiga kooskõlastatud.	Süsteemi jõudlus peab vastama kokkulepitud topoloogial eelanalüüsi ja lähteülesande käigus välja toodud jõudlusnäitajatele.	Arendaja
2.3	Rakendusserver peab võimaldama töötamist andmebaasiserverist eraldi serveril.	-	Arendaja
2.4	Rakendusserver peab olema vajadusel klasterdatav aktiivklastris (kasutajasessioon ei tohi olla klastri node põhine).	-	Arendaja
2.5	Rakendust peab saama ilma ümberprogrammeerimata liigutada erinevate domeenide ja domeeni saitide vahel.	Lahendus ei tohi olla sisse kompileeritud absoluutseid URI-sid	Arendaja
2.6	Rakenduse liidesed peavad olema tõrkekindlad kolmandate osapoolte süsteemide vigade suhtes.	Väliste liidestatud süsteemide tõrke korral ei tohi süsteem hanguda, vaid väljastama mõistliku (võimalikult lühikese) aja jooksul ajakohase veateate. Võimalusel tuleb kasutada asünkroonseid liideseid.	Arendaja
2.7	Rakenduse konfiguratsiooniparameetrid tuleb ühte kohta kokku tuua nii, et nende muutmisel ei peaks rakendust uuesti kokku kompileerima (nt ühte tekstipõhisesse konfiguratsioonifaili, andmebaasi tabelisse).	Rakendus peab neid sealt ka kasutama (mitte kopeerima parameetreid käivitamisel kolmandatesse kohtadesse), logimise seaded võivad olla rakenduse konfiguratsioonifailist eraldi ühes lisakonfiguratsioonifailis (näit Log4net). Samuti on väga soovitatav eraldi konfiguratsioonifailis hoida arendaja ja administraatori vastutusala parameetrid. Infosüsteem peab olema seadistatav konfiguratsiooniparameetrite(de) abil. Konfiguratsioonifailiks ei saa lugeda faili, kus hoitakse lisaks konfiguratsioonile ka muud programmikoodi.	Arendaja
2.8	Rakenduse kompileerimine, saidi taaskäivitus ja konfiguratsiooni muutmine peavad toimuma mõistliku aja jooksul.	Rakenduse kompileerimine < 10min Mooduli taaskäivitus < 1min Konfiguratsiooni muutmine < 30s Kui rakendus vajab indekseeritud sisu ja see pole kättesaadav, siis peab rakendus väljastama selle kohta selge teate	Arendaja
2.9	Rakendus peab kasutama 64-bitist arvutiarhitektuuri kui ei ole kokku lepitud teisiti.	Suund on 64-bitiste rakenduste op süsteemide kasutamise poole.	Arendaja
2.10	Kõik andmed, andmebaasid, SQL skriptid ja rakendus peavad kasutama UTF-8 kodeeringut.	-	Arendaja
2.11	Failisüsteemi salvestamisel ei tohi ühte kausta tekkida üle 10000 faili.	Failid peab katalogiseerima kokkulepitud tunnuste alusel (nt aasta, kuu, kuupäev).	Arendaja
2.12	Rakenduse loomisel tuleb eelistada objektorienteeritud mudelit.	Eelistuse eiramine tuleb kooskõlastada projektijuhiga enne arendamise alustamist.	Arendaja

2.13	Ühest andmetabelist teise viitamisel tuleb kasutada väliseid võtmeid (Foreign key).	-	Arendaja
2.14	Kõik välised võtmed (Foreign Key) peavad olema indekseeritud.	Andmebaasis peab kasutama indekseid või muid meetmeid, et nõuded rakenduse jõudlusele oleksid täidetud ka tulevikus. (1, 3, 5 või 10 aasta pärast – vastavalt planeeritud kasutusajale).	Arendaja
2.15	Tuleb kasutada päringumuutujaid (Parameter Binding).	SQL päringute väljakutsumisel väljastpoolt andmebaasi, peab kasutama päringumuutujaid, et vältida SQL vahemälu fragmentseerumist (When calling SQL code from outside the database, Parameter Binding should be used to prevent SQL cache fragmentation)	Arendaja
2.16	Kõigis andmebaasi tabelites peab olema defineeritud üks primaarvõti. Andmebaasi objektide nimetused peavad olema sisulised ja andma aimu nende otstarbest.	Kasutada vastava andmebaasisüsteemi nimetamise parimaid praktikaid.	Arendaja
2.17	Andmebaasis defineeritakse üldjuhul kaks või enam kasutajat: • Rakenduse peakasutaja, kellena luuakse objektid ja skeemid. • Rakenduse piiratud õigustega kasutaja, kellena pöördub rakendusserver/rakendus. Objektide loomiseks vajalikud õigused ja ressursid on loetletud rakenduse dokumentatsioonis.	Need õigused, mis on vajalikud ainult rakenduse baasi loomiseks, on eraldi välja toodud ja tuleb peale installi ära võtta. Ei kehti teiste andmebaasisüsteemide korral, seal võib see tekitada mõttetut keerukust.	Arendaja
2.18	Failide hoidmise asukoht lepitakse igakord kokku. Kuid failid ja failide indeks peavad olema replikeeritavad teise serveriruumi.	Failide hoidmine klassikalises andmebaasis on kulukas ja seab kõrgendatud nõudmised ja piirangud andmebaasiserveritele. Lahenduse dokumentatsioonis tuleb ära tuua failide hoidmise asukoht.	Arendaja
2.19	Peab olema minimiseeritud vajadus, et haldur teeb haldustoiminguid otse baasis. St rakendusel peab olema haldusliides, mille kaudu rakenduse haldur saab teha tavapäraseid haldustoiminguid.	Halduri haldustoimingud lepitakse tellijaga kokku detailanalüüsi käigus.	Arendaja
2.20	Rakendus peab olema võimeline kasutama keskkonnamuutujaid (serverinimi, kuu, päev jne).	Näiteks logifailides.	Arendaja
2.21	Andmebaas peab toetama nii külm- kui ka kuumvaru (peegeldamist) teise serviruumi.	Ei tohi kasutada teenuseid, mis välistavad andmebaasi peegeldamist (nt "failstream").	Arendaja
2.22	Sorteerimisreeglistik peab olema Eesti tähestikule vastav. Tõusutundlikkus peab olema välja lülitatud. Accent peab olema sisse lülitatud.	Näiteks MS SQL puhul Estonian_CI_AS.	Arendaja
2.23	Kui infosüsteemid saadavad e-kirju, peavad nad kasutama välist e-mailiserverit. Kirja saatmisel peab rakendus veenduma, et e-mailiserver võttis meili vastu. E-kirjade vormindamine peab järgima interneti standardeid (RFC 5322).	Saatja ja adressaadid, pealkiri ja sisu ei tohi olla rakendusse kodeeritud, vaid on muudetavad konfiguratsioonifaili kaudu. Genereeritud kirjade puhul peab tagama kirjade jälitatavuse (näiteks lisada X-päise kodeeritud kirje, milles on kirjeldatud, mis protsess/skriptifail/kasutaja kirja genereeris jms abistav info).	Arendaja
2.24	Konfiguratsiooniparameetrite nimed peavad olema sisulised. Kui see ei ole võimalik, siis peab kõrval olema seletus.	Näiteks : X-tee Turvaserver, mitte XTTS või viitenumber, mitte vk_seb jne	Arendaja
2.25	Infosüsteemides on eessüsteemid (front end; presentatsiooni kiht) ja tagasüsteemid (back end; äriloogika kiht) arhitektuuriliselt selgelt lahutatud.	Välise süsteemi tõrge tohib mõjutada ainult sellest otseselt sõltuvate kasutuslugude toimimist. Välise süsteemi taastumisel peab süsteem olema suuteline oma tööd jatkama taaskäivitamata.	Arendaja
2.26	Konfiguratsioonifailid peavad olema vastavalt rakendusserveri tüübile vaikimisi kaitstud failid	Näiteks IIS: *.config , *.resources Apache: *.conf, .htaccess. Arendaja peab välja tooma konfifailide listi, kui neid on mitu.	Arendaja
2.27	Rakenduse failid, mida kasutaja näha ei tohi, peavad olema vaikimisi kaitstud kaustades.	Näiteks: IIS: Bin,App_Code, App_Data, App_Browsers, App_GlobalResources, App_LocalResources, App_Themes, App_WebReferences	Arendaja
2.28	Konfiguratsiooniparameetrite taaskasutus. Erinevaid sama sisuga parameetreid ei tohi konfiguratsioonis eksisteerida.	Kõiki parameetreid tuleks konfiguratsioonis kirjeldada vaid korra, mitte nii, et igas lõigus kirjeldatakse samu asju uuesti.	Arendaja
2.29	Kõik rakenduse liidesed peavad olema võimelised töötama kõrgkäidetavalt.	Rakendustes tohib kasutada vaid masinapõhiseid teenuseid, mis lubavad kõrgkäideldavaid (klaster) lahendusi. Kõrgkäideldav lahendus on selline, mida saab samaaegselt käitada erinevates masinates.	Arendaja

2.30	Klientrakendus ei tohi pöörduda otse andmebaasi poole.	Tuleb kasutada rakendusservereid.	Arendaja
2.31	Keskkonnapõhised muutujad peavad olema konfiguratsioonifailist seadistatavad.	Näiteks WSDL ei tohi sisaldada viiteid arendusserveritele.	Arendaja
2.32	Rakenduses peab olema võimalik piirata ebaõnnestunud logimisi ajaühiku kohta (mobiil-ID, ID-kaart, paroolid) ühelt IP-aadressilt.	Eelistama peaks IP-aadressipõhist blokeeringut. Erandina tellijaga kokkuleppel võib kasutada captchat või konto lukustamist. Blokeeringute ajavahemikku ja logimiskatsete arvu peab saama konfiguratsioonifailist muuta. Allikas: https://iske.ria.ee/8_06/ISKE_kataloogid/7_Kataloog_M/M4/M_4.15	Arendaja
2.33	Rakenduse äriloogika tuleb realiseerida andmebaasist eraldi sõltumatus rakenduskihis.	Andmebaas ei tohi sisaldada äriloogikat, mis muudab andmetabelites olevaid/sinna kirjutatavaid andmeid, va trigerid, mis tekitavad logi.	Arendaja
2.34	Andmebaasis võib kasutada vaid ISO /IEC 9075 standardiga kaetud funktsionaalsusi. Lisaks ei tohi kasutada ka sama standardi osas 13 kirjeldatud funktsionaalsusi.	*Ei ole soovitav kasutada mingit platvormispetsiifilist lahendust, mille üleviimine mõnele muule andmebaasiplatvormile ei ole võimalik. *ISO/IEC 9075 osa 13 spetsifitseerib Javas kirjutatud programmimoodulite kasutamist andmebaasis.	Arendaja
2.35	Uniform resource identifier (URI) pikkus ei tohi ületada ühegi IS poolt toetatava brauseri maksimaalset lubatud väärtust.	Harilikult on piiriks 2000 tähemärki, kuid iga IS puhul tuleb seda eraldi järele uurida sõltuvalt IS komponentidest. Asjakohased viited: RFC 3986 ja RFC 7239.	Arendaja
2.36	SOAP teenuseid pakkuva rakenduse WSDL peab olema üles ehitatud nii, et see toetaks teenuste versioneerimist.	Näiteks: Alajaotis definitions/types/schema: * complexType defineerimisel tuleb sellele lisada any element.	Arendaja
2.37	Rakendus peab olema võimeline töötama koormusjaoturitega varustatud taristul.		Arendaja
2.38	Sidusinfosüsteemide mitte kättesaadavus ei tohi segada rakenduse töötamist. Sidusinfosüsteemidega andmevahetamisel tekkinud vead logitakse ja kasutajat hoiatatakse.	Sidussüsteemi tõrge tohib mõjutada ainult sellest otseselt sõltuvate kasutuslugude toimimist. Sidussüsteemi taastumisel peab süsteem olema suuteline oma tööd jatkama taaskäivitamata.	Arendaja
2.39	Kui ajastatult käivitatav taustatöö, ei ole mõeldud käima paralleelselt, peab selles olema realiseeritud kontrollmehhanism, mis tagab, et sama taustatööd ei ole võimalik käivitada uuesti enne, kui eelmisena käivitatud instants on oma töö lõpetanud.		Arendaja
2.40	Ühe tarkvarakomponendi raames ei tohi sama parameetri seadistamine toimuda rohkem kui ühes kohas.	Näiteks kui rakenduse komponent pöördub andmebaasi või veebiteenuse poole, siis selle pöördumise parameetrid peavad olema muudetavad vaid ühes kohas.	Arendaja
2.41	Uue toote arenduse ja olemasolevate infosüsteemide versiooniuuendustel kasutusele võetavate Tehnoloogiate ja standardite valik tuleb kooskõlastada Tellija poolse arhitektiga.		Arendaja
2.42	Rakenduse ühenduste (s.h. andmebaasi ja sidusinfosüsteemide ühendused) realiseerimisel tuleb kasutada ühenduste puulimist (connection pooling).	Implementeeritud peab olema vähemalt maksimaalsete ühenduste arvu piirang ja päringu aegumise aeg (request timeout). Rakenduse ühenduste tõrge tohib mõjutada ainult sellest otseselt sõltuvate kasutuslugude toimimist. Ühenduste taastumisel peab rakendus olema suuteline oma tööd jatkama taaskäivitamata. Tekkinud vead logitakse ja kasutajat hoiatatakse.	Arendaja
2.43	Rakenduse uuendustega kaasnevad andmebaasi muudatused tuleb automatiseerida.	Näiteks Liquibase või Flyway	Arendaja
3. Turvalisuse tagamisega seotud nõuded			
3.1	Sisemised rakendusliidese autentimised peab saama teha Active directory põhiselt.	Erandina tellija kooskõlastusel võib sellest loobuda ja kasutada vaid ID-kaardi ja mobiil-ID põhist autentimist. Isiku sertifikaatide kehtivust peab saama kontrollida vastu OCSP ja CRL-i (vastavalt vajadusele).	Arendaja
3.2	Kliendi ja serveri vahel peab autenditud kasutajasessioonide korral olema sessioon krüpteeritud HTTPS-protokolli kasutades.	-	Arendaja
3.3	SSL veebiserver peab kasutama turvalisi ja SSL/TLS versioone ja šifrikomplekte	https://www.ssllabs.com/ssltest/	Arendaja
3.4	Rakendus tohib kasutada vaid sessiooni küpsiseid. Muude küpsiste kasutamine on keelatud.	-	Arendaja

3.5	Kui andmebaasis olevate andmete ISKE tervikluse turvaosaklass on 2 või kõrgem, siis tuleb kõik klass 2 infot sisaldavad andmebaasi kirjed/tabelid versioneerida.	St kõik andmemuudatused peavad baasis säilima. Andmete muutmisel andmeid ei kustutata, vaid tehakse uus kirje uute andmetega. Vana muudetakse kehtetuks. Iga uus kirje peab sisaldama järgmist informatsiooni: *viide kirjele, mille ta kehtetuks muutis (kui on) *kasutaja, kes kirje lõi *kirje loomise aeg *sessiooni-ID (kui on olemas). Iga kehtetuks tunnistatud kirje peab omama järgmist informatsiooni: *kasutaja, kes kirje kehtetuks tunnistas *kirje kehtetuks tunnistamise aeg.	Ar en daja
3.6	Kui rakenduse poolt töödeldavate andmete konfidentsiaalsuse turvaosaklass on 2 või kõrgem, peab rakendusega kaasas olema lahendus, mis suudab toota toodangu andmetest testandmed, mis ei sisalda konfidentsiaalset informatsiooni.	Testandmed peavad säilitama kõik toodangu andmete omadused (pikkuse, tüübi) ja omavahelised suhted.	Ar en daja
3.7	Andmebaasis olevate rakenduse kontod peavad omama ainult minimaalselt rakenduse tööks vajalikke õiguseid.	Ei resource, dba, ANY ega muud sellist. Nõude täitmiseks vajalikud vahendid (skriptid) peavad kuuluma rakenduse juurde ja nende sisu peab olema kontrollitav. Kontodele vajalikud õigused peavad olema kirjeldatud rakenduse installijuhendis.	Ar en daja
3.8	Rakendusse ja andmetele tohib olla ligipääs vaid dokumenteeritud ja tellimuses kirjeldatud teid mööda ning dokumenteeritud autentimisprotseduure kasutades.	St rakendustes ega andmebaasides ei tohi olla ligipääsemiseks teisi võimalusi.	Ar en daja
3.9	Kõik paroolid ja salaküsimuste vastused peab rakendus salvestama vaid räsitud+soolatud. Kui räsimise asemel valitakse krüpteerimine, siis tuleb kirjeldadakrüptovõtme turvalise hoidmise protseduur.	Räsimine peab kasutama turvalist räsifunktsiooni (nt SHA2, SHA3, RIPEMD-160) ja kindlasti ka soola (salt). Sool peab olema andmebaasiüleselt unikaalne, piisavalt suure bitiarvuga ja (pseudo)random. Krüpteerimisel peab kasutama turvalisi algoritme (nt AES256) ja CBC, CRT vms režiimis. Kindlasti ei tohi kasutada ECB režiimi. Paroolid ja salaküsimuste vastused tohivad olla krüpteerimata kujul vaid ajutiselt serveri muutmälus. Krüpteerimata kujul ei tohi paroole salvestada (ka ajutiselt) ühelegi kettale. Arendaja loodud lahenduse dokumentatsioonis (nt detailanalüüs vms) peab olema ära toodud kasutatavad räsi ja krüptoalgoritmid, võtmepikkused ja nende kasutuskohad (vt nõue p 1.10)	Ar en daja
3.10	Rakendused, kuhu saavad ligi välised kasutajad, peavad võimaldama sisselogimist ID-kaardi ja mobiil-ID-ga. Paroolipõhist autentimist ei tohi kasutada.	Kui on vajalik ka parooliga logimine, peavad välised kasutajad autentima ennast spetsiaalse väliskasutajate jaoks mõeldud AD pihta. Kui parooliga autentimist tehakse alati samadelt üksikutelt IP-delt, siis tuleb lisaks paroolile kasutada ka IP-põhist ligipääsukontrolli. Rakendus ei tohi lubada kasutada nõrku paroole, peab võimaldama paroolide eelmääratud aegumist ja mitme valesisestuse (nt 5 korda) korral kontode lukustamist. Sertifitseerimiskeskuse dokument digisertifikaatide kasutamise kohta EV dokumentidel: https://sk.ee/upload/files/SK-CPR-ESTEID-ET-v5_0-20150101.pdf	Ar en daja
3.11	Mobiil-ID autenimise korral tuleb lisaks kasutaja telefoninumbrile küsida ka kasutaja isikukoodi.	Veebilehel kuvatav kontrollkood peab olema selgelt nähtav, sh ka nutitelefonidel ilma ekraanipilti kerimata.	Ar en daja
3.12	Rakendus ei tohi teostada X-tee päringut otse kasutajaarvutist.	Kasutajaarvutitest otse x-tee päringute tegemine on arvutivõrgu tasemel kinni.	Ar en daja
3.13	Veebipõhised välise veebilehega rakendused, mis on keskmise või kõrgema ISKE turbeastmega, peavad kasutama vahendeid kaitsmaks rakendust lubamatute päringute eest.	IIS puhul peab kasutama näiteks URL scan, apache puhul modsecurity või vastavat tööriista. Lubatud päringud on kõik päringud, mis ei ole detailanalüüsi käigus vastavalt kasutusjuhtudele ette nähtud. Kasutama peab whitelisting põhimõtet, mitte blacklisting.	Ar en daja
3.14	Kõigil rakendustel peab olema konfigureeritav kasutajasessiooni aegumise aeg.	Aeg peab olema muudetav koos teiste konfiguratsiooniparameetritega.	Ar en daja
3.15	Krüpteerimise ja/või räside arvutamise korral tuleb kasutada tugevaid algoritme.	Järgida tuleb uusimat RIA kodulehel avaldatud krüptograafiliste algoritmide kasutusvaldkondade ja elutsükli uuringut. Lubatud on: AES-256, Blowfish-256, RSA-2048, SHA-2, RIPEMD-160 või tugevamaid. Lahenduses tuleb välja tuua kõik krüptoalgoritmid, võtmepikkused ja kasutuskohad.	Ar en daja
3.16	Autenditud sessiooni tunnust ei tohi ainult lihtsa küpsisega lahendada.	Sessiooni ei tohi olla võimalik üle võtta sessioonitunnuse kopeerimisega ühest arvutist teise.	Ar en daja
3.17	AD või AAM-i autentimise kasutamisel peab rakendus kasutama ka AD või AAM-i kontoga kaasnevaid piiranguparameetreid.	Näiteks: konto on lukus, parool aegunud, konto aegunud, paroolipoliitika jne.	Ar en daja
3.18	Tagada tuleb rollide lahusus. Halduritel ei tohi olla võimalik muuta ega näha rakenduse konfiguratsiooni.	Administraatoril, halduril ja tavakasutajal on erinevad tööülesanded. Rollide/õiguste kirjeldus peab lähtuma detailanalüüsist ja kasutusjuhtudest.	Ar en daja
3.19	ID-kaardiga autentimisel, peab rakendus suutma vastu võtta ID-kaardi sertifikaati ka päises.	Proxy tugi	Ar en daja
3.20	Kui kasutajaid hallatakse ka rakenduses ja autenditakse AD või AAM-i vahendusel, tuleb sisselogimisel kõigepealt kontrollida kasutaja olemasolu rakenduses ja alles siis pöörduda AD või AAM-i poole.	Eesmärk vähendada AD ja AAM-i koormust.	Ar en daja
3.21	Kui rakenduse tervikluse turvaosaklass on T3, peavad tõestusväärtust omavad andmed olema kas ajatembeldatud, digiallkirjastatud või digitembeldatud.	Milline lahendus valitakse tuleb kokku leppida tellijaga. Täpsustuseks vt ISKE nõue HT.34.	Ar en daja

3.22	Kui lahendus peaks kasutama ajatempli teenust, siis tuleks eelistada Guardtime lahendust.	Ajatempli kasutamise vajadus lepitakse eraldi kokku Tellija IT juhiga ja infoturbejuhiga. See sõltub Tellija keskse ajatempli kasutamise võimalustest.	Ar en daja
3.23	Kui rakenduse tervikluse turvaosaklass on T3, peavad tõestusväärtust omavad andmed olema krüptoaheldatud, et tagada et tõestusväärtusega andmeid ei saaks märkamatult kustutada.	Täpsustuseks vt ISKE nõue HT.10. Krüptoahela kasutamise vajadus lepitakse eraldi kokku Tellija infrastrktuuri juhiga ja infoturbejuhiga. See sõltub Tellija keskse krüptoahela kasutamise võimalustest.	Ar en daja
3.24	Kui rakenduses on S3 salastuse astmega andmeid, peavad need olema nii transpordi ajal ja ka salvestatult alati krüpteeritult.	Täpsustuseks vt ISKE nõue HT.37 .	Ar en daja
3.25	Rakendus ja selle komponendid peavad võimaldama kasutada keskkondade lahusust	Arendaja arendab arenduskeskkonnas ja annab tarne üle tellijale paigalduspakkidena. Tellija paigaldab selle testkeskkonda ja testib ning seejärel paigaldab tarne toodangu keskkonda. Reaalseid andmekogu andmeid tohib töödelda üksnes toodangu keskkonnas.	Ar en daja
3.26	Rakendus peab võimaldama hõlpsalt välja vahetada aegunud ja ebaturvalise krüptoalgoritmi.	Krüptograafiat kasutav rakenduskood ei tohi nimeliselt välja kutsuda krüptograafilisi algoritme, vaid peaksid seda tegema vahendavate vaheteekide kaudu üldiste funktsioonide järgi (nt krüpteerimine, dekrüpteerimine, signeerimine, signatuuri verifitseerimine jne). Dokumentatsioon peab kajastama üldist kirjeldust, kuidas vajadusel ebaturvaline krüptoalgoritm välja vahetada.	Ar en daja
3.27	Rakenduse andmebaasi krüpteerimisega seotud andmeväljad peavad olema muudetava pikkusega.	Andmebaasides kasutatavad krüpteerimisfunktsioonidest tingitud lisaväljad peaksid olema muudetava pikkusega, et formaati muutmata saaks kasutada teistsuguste parameetritega krüpteerimisalgoritme.	Ar en daja
4. Logimine, debuggimine, testimine			
4.1	Rakendusel peab olema masinloetav testleht (JSON, XML).	Testlehe kättesaadavus erinevatest arvutivõrkudest peab olema konfigureeritav. Testleht peab uuendama ennast lehe pärimisel. Testleht peab sisaldama custom builtrakenduse versiooni numbrit, standardsed komponendid (veebiserver, andmebaas, CMS'id jms) ei tohi oma versioone reeta. Samuti peab testlehel olema infot rakenduse (vajadusel tema erinevate osade) ja tema kõigi väliste liideste staatuse kohta (töötab, ei tööta). Rakenduse, andmebaasi ja liideste töökorda kontrollitakse testpäringute teel, mis tuleb tellijaga kokku leppida eelanalüüsi käigus. Testleht peab oma konfiguratsiooni võtma rakenduse üldisest konfiguratsioonist (baasistring, välised ühendused).	Ar en daja
4.2	Rakenduse kõik üleantavad versioonid peavad enne tellijale üle andmist olema testitud.	Testitulemused tuleb edastada tellijale koos rakenduse üleandmisega.	Ar en daja
4.3	Rakendus peab logima kasutaja edukat ja ebaedukat autentimist ja sessiooni lõpetamist, kasutaja IP ja autentimismeetodit (ID-kaart, mobiil-ID vms), eduka autendi puhul tuleks logida ka kasutaja isikukood ja mobiil-ID puhul telefoninumber	Logima peab ka autentimise ebaõnnestumise koos põhjusega (vale parool, aegunud konto jne). Logida tuleks IP-aadress, meetod ja kui võimalik kasutajatunnus (mobiil-ID puhul telefoni number, ID-kaardi puhul isikukood). Kui rakendus kasutab kasutajate autentimiseks AAM-i või TARA, siis leppida projektijuhiga eraldi kokku autentimise detailsus ehk mis kajastatakse AAM-is või TARA-s ja mis rakenduses.	Ar en daja
4.4	Erinevate logifailide kirjeid peab olema võimalik seotud komponentide logidega loogiliselt kokku viia.	Näiteks timestamp või mingi (*request*)ID abil.	Ar en daja
4.5	Rakendus peab suutma logida kõiki X-tee teenuste kaudu liikuvaid andmeid. Peab olema võimalus logimist sisse-välja lülitada.	Vajalik eelkõige debuggimiseks ja toodangu keskkonna probleemide lahendamiseks.	Ar en daja
4.6	Kui rakenduse ISKE konfidentsiaalsus turvaosaklass on 2 või kõrgem, peab rakendus logima kõiki konfidentsiaalsus klassiga 2 või kõrgemate andmete loomist, muutmist (sh kustutamist) ja vaatamist.	Isikuandmete töötlemisel lähtub täitja turvameetmetest vastavalt IKS §-le 43.	Ar en daja
4.7	Kui rakenduse ISKE tervikluse turvaosaklass on 2 või kõrgem peab rakendus logima kõiki tervikluse klassiga 2 andmete loomist ja muutmist (sh kustutamist).	Isikuandmete töötlemisel lähtub täitja turvameetmetest vastavalt IKS §-le 43.	Ar en daja
4.8	Kui rakenduse andmete konfidentsiaalsuse turvaosaklass on 3, siis ka kõiki administraatorite ja haldurite poolt tehtavaid andmete vaatamised (ka otse baasis) tuleb logida.	Lahendus peab tagama, et administraatorid/haldurid ei saa andmete vaatamise logimist ise (ka tavakasutajate logimist) deaktiveerida või logisid kustutada/muuta. Võib tellijaga kokku leppel nõudest loobuda kui andmed on krüpteeritud.	Ar en daja
4.9	Kui rakenduse andmete tervikluse turvaosaklass on 3, siis ka kõiki administraatorite ja haldurite poolt tehtavaid andmete muudatused (ka otse baasis) tuleb logida.	Lahendus peab tagama, et administraatorid/haldurid ei saa andmete muutmise logimist ise (ka tavakasutajate logimist) kinni keerata või logisid kustutada/muuta. Tellijaga kokkuleppel võib nõudest loobuda, kui andmed on kaitstud digiallkirja, digitempli või välise osapoole ajatempliga.	Ar en daja
4.10	Andmete loomise/vaatamise/muutmise /kustutamise tegevused peab logima.	Logid peavad asetsema tsentraalses logiserveris	Ar en daja

4.11	Andmebaasi logidest saadetakse reaalajas koopia failisüsteemi logisse ja seal logis peab kajastuma ka logimisfunktsionaalsuse aktiveerimise ja deaktiveerimise info (aeg, kasutaja jms)	Failisüsteemi logide eesmärk on koguda logid ühtesesse logihaldussüsteemi, et neid krüptoaheldada ja aegtembeldada.	Arendaja
4.12	Rakendusega peab olema kaasas skript jõudlustestide tegemiseks.	Jõudlustestide täpne kirjeldus tuleb kokku leppida detailanalüüsi käigus. Arendaja peab koos rakendusega tarnima skripti ja vajalikud tarkvaralised vahendid kokkulepitud jõudlustestide läbiviimiseks. Jõudlustestide läbiviimine ei tohi nõuda tellijalt omapoolset tarkvara arendamist, skriptide kirjutamist või litsentside ostmist.	Arendaja
4.13	Rakendus peab logima kõiki rakenduses tekkivaid tehnilisi vigu.	Logi sisaldab minimaalselt vea tekkimise aega, veakoodi, veakirjeldust (stack trace, traceback vms), võimalusel kasutaja andmeid, HTTP-, GET- ja POST-parameetrid ja nende väärtusi.	Arendaja
4.14	Vea ja süsteemilogid peavad olema vähemalt failisüsteemi tekstifailis üldtuntud vormingus, lisaks võib ka andmebaasis hoida.	Kui logitakse mitmesse kohta, siis vajadusel peab saama ühte või teise kohta logimist välja lülitada. Logi peab olema lihtsalt masintöödeldav ja tuntud vormingus. Näiteks syslog, syslog-ng, XML, CSV.	Arendaja
4.15	Failisüsteemi logimise korral peavad logid olema ka katalogiseeritud (näiteks kuupäeva või liigi järgi) ja üldtunnustatud faililaiendiga (näiteks .log, .txt, .xml), logi peab olema roteeruv, et ei tekiks liiga suuri faile (nt 5MB). Logifailide seadistamisel peab olema failinime /kaustatee nimedes võimalik kasutada keskkonnamuutujaid (kuupäev, masinanimi jne).	Ei tohi esinda olukorda, kus ühte kausta tekib rohkem kui 1000 faili.	Arendaja
4.16	Logimis parameetreid peab saama muuta rakendust taaskäivitamata.	Näiteks log4j konfiguratsiooni failis "monitoring-interval".	Arendaja
4.17	Arhitektuuriline lahendus peab olema 75% ulatuses kaetud komponenditestidega (*unit test*).		Arendaja
4.18	Arhitektuuriline lahendus peab olema 50% ulatuses kaetud automatiseeritud integratsiooni ja vastuvõtutestidega (*integrationtest, api end-to-end test*).		Arendaja
4.19	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 kokkulepitud raamistikul (nt Selenium)		Arendaja
5. Nõuded rakenduse lähtekoodile			
5.1	Lähtekoodi kommentaarid peavad kõigis lahenduse kihtides (rakenduse enda kood, andmebaas, jne) olema kirjutatud inglise keeles.	**NB!** Nõuet ei arvestata arendustarkvara poolt automaatselt genereeritavate koodilõikude puhul – neid ei ole vaja tõlkida. Samuti ei rakendata nõuet kolmandate osapoolte poolt toodetud lähtekoodile – nt igasugu erinevad lahtise koodiga koodilõigud jms. Kui on tegu olemasoleva süsteemi edasiarendusega, siis peaks kommentaarides kasutama eelnevalt kasutatud keelt.	Arendaja
5.2	Muutujate, tüüpide ja funktsioonide nimed peavad olema sisulised ja andma aimu nende otstarbest.	Parim praktika	Arendaja
5.3	Koodis kasutatavad konstandid ja lühendid tuleb kirjutada suurte tähtedega.	Parim praktika. Nt Identifikaator --> ID. Front-End reeglid on kirjeldatud Front-end arendusreeglid lehel	Arendaja
5.4	Koodis kasutatavaid konstante ei tohi selle kasutamise kohta väärtusena hardcodeda – need tuleb defineerida muutujatena ja kasutada läbi nende.	-	Arendaja
5.5	Koodis defineeritud andmetüübid peavad olema nimetava käände ainsuses. Kõik andmemassiivid tuleb nimetada nimetava mitmuses (st igasugu collectionid, arrayd, jms).	N:Isik; Menetlus; jne. Andmebaaside struktuurikirjeldustes/andmemudelis ei tohi kasutada täpitähti.	Arendaja
5.6	Andmetabelites sisalduvad võõrvõtmed peavad nime järgi seostuma tabeli ja väljaga millele need viitavad.	Kasutada tuleb konkreetse andmebaasisüsteemi nimetamise parimaid praktikaid. Nt kui on tegu tabelitega 'Isikud' ja 'Autod', siis seos 'isiku autod' oleks: Isikud.ID=Autod.Isik_ID	Arendaja
5.7	Andmebaasi väljade pikkused tuleb kirjeldada sümbolites, mitte baitides.	Selle asemel, et eraldada väljale x baiti, tuleb eraldada x tähemärki. (Instead of allocating x bytes of storage for the field, x chars of storage must be allocated).	Arendaja

5.8	Kui kokku pole lepitud teisiti, siis Java rakenduse kood peab olema kirjutatud vastavalt "Google Java Style Guide" dokumendile: https://google.github.io/styleguide/javaguide.html		Arendaja
5.9	Java koodi valideerimiseks kasutatakse tellija SonarQube paigalduses seadistatud reeglistiku	TEHIK-us on automaatseks koodivalideerimiseks kasutusel SonarQube (https://www.sonarqube.org/). Ennem üleandmist tuleb veenduda, et koodis puuduvad: 1. Turbedefektid 2. Blokeerivad ja kriitilised vead Mõistlik on koodivalideerimine automatiseerida Gitlabi või Jenkinsi abil. Sõltub milline lahendus on projektis kasutusel.	Arendaja
5.10	Kasutuses mitteolev kood tuleb rakenduse lähtekoodist kõrvaldada.	-	Arendaja
5.11	Arendamisel kasutatakse DRY ja SOLID printsiipe	http://en.wikipedia.org/wiki/Don%27t_repeat_yourself http://en.wikipedia.org/wiki/SOLID_(object-oriented_design)	Arendaja
5.12	Üleantavas koodis ei tohi olla paroole, mida on kasutatud arenduse käigus	Kehtib ka siis, kui need on välja kommenteeritud. Kõik sellised paroolid tuleb asendada fraasiga “<password>“.	Arendaja
5.13	Rakenduste lähtekoodi tasemel ei tohi olla ühtegi sisse kodeeritud parameetrit, väljade nimetust, veateadet.	Eelpoolmainitu haldamine toimub failis või andmebaasis.	Arendaja
6. Andmekvaliteet ja standardid			
6.1	Rakendus peab võimalikult palju informatsiooni eeltäitma automaatselt (kirje sisestamise kuupäev, kasutaja nimi jne).	Välja arvatud logimisvormi lahtrid autentimisel	Arendaja
6.2	Tegevusalade andmete sisestamisel, kuvamisel ja hoidmisel tuleb lähtuda Vabariigi Valitsuse 10. Jaanuari 2008. a määrusest nr 11 "Klassifikaatorite süsteem" ja kasutada EMTAK infosüsteemis kehtivat klassifikaatorit.		Arendaja
7. Kasutajaliides			
7.1	Kasutajaliidese kõik disainiotsused peavad olema kooskõlastatud tellijaga enne nende realiseerimist	-	Arendaja
7.2	Veebipõhine kasutajaliides peab olema kasutatav enamlevinud veebibrauseritega, sh nutiseadmetel (Android, IOS, Windows Phone)	Minimaalselt Internet Explorer, Mozilla Firefox, Chrome ja Safari arenduse testimise hetkel tootja poolt toetatud versioonid. Täpsemad nõuded dokumendis "Front-end arendusreeglid"	Arendaja
7.3	Rakenduse värviskeem ja logo kasutamine peab vastama Tellija ametlikule visuaalsele identiteedile (CVI) ja disainijuhistele (UIG).	Kui tegemist on struktuurfondide projektiga on lisaks nõutud ka vastav SF sümboolika. Tellija ametlikud CVI esitluspõhjad, logo kasutusjuhend ja kõik logod (ka jpg-na) küsida tellijalt. Iseteeniduse väljanägemine tuleb vastaval RIA stiiliraamatule. Ametnikurakenduse väljanägemine vastavalt ametnikurakenduses kehtestatud stiiliraamatule.	Arendaja
7.4	Kasutajaliidese kõik osad ja teated peavad olema eestikeelsed.	Kui soovitakse juurde eraldi ka muid keeli, siis see on spetsifitseeritud hankedokumentides	Arendaja
7.5	Avalikuks kasutamiseks tehtav rakendus peab olema graafiliselt eskaleeruv ja mugavalt kasutatav kõigi enamlevinud arvutite monitoride resolutsioonidega.	Toetatud peavad olema vähemalt resolutsioonid: 1920x1200, 1920x1080, 1680x1050, 1600x1200, 1440x900, 1360x768, 1280x1024, 1280x960, 1280x800, 1280x768, 1152x864, 1024x768, 1024x600. Ühegi nimetatud resolutsiooni korral ei tohi tekkida lehe ülest horisontaalset kerimisriba. Mahukate andmekogumite väikestel ekraanidel kuvamiseks üks lahendus võib olla komponendi sisene kerimine x ja y teljel. Täpsed lahendused lepitakse kokku töö käigus ja vastavalt vajadusele.	Arendaja
7.6	Sisemiseks kasutamiseks tehtav rakendus peab olema graafiliselt eskaleeruv ja mugavalt kasutatav järgmiste monitoride resolutsioonidega: 1024x768, 1280x1024, 1680x1050, 1920 x 1080, 1920x1200.	Ühegi nimetatud resolutsiooni korral ei tohi tekkida lehe ülest horisontaalset kerimisriba. Mahukate andmekogumite väikestel ekraanidel kuvamiseks üks lahendus võib olla komponendi sisene kerimine x ja y teljel. Täpsed lahendused lepitakse kokku töö käigus ja vastavalt vajadusele.	Arendaja
7.7	Hüpikaknaid (pop-up) ei tohi kasutada.	Silmas on peetud uusi veebilehtiseja aknaid avavaid hüpikaknaid	Arendaja
7.8	Kasutajaliides peab alati küsima kinnituse andmete kustutamise ja massmuutmiste kohta kui pole kokku lepitud teisiti.	-	Arendaja

7.9	Rakenduse kasutamisel tekkinud veale peab kasutajaliides vastama kasutajale eestikeelse kasutajasõbraliku veateatega, mis sisaldab soovituslikult ka vea koodi.	Veateated peavad olema sellised, mis võimaldavad IT-abil võimalikult lihtsalt tuvastada vea olemuse ja asukoha. Kui kasutaja kasutab süsteemi mõnes võõrkeeles siis peavad veateated olema selles keeles	Ar en daja
7.10	Kasutajaliides peab olema ilma rakenduse koodi muutmata tõlgitav teise keelde, v.a kui ei ole kokkulepitud teisiti.	Uue keele lisamine peab olema teostatav konfiguratsiooni failist või administreerimisliidesest.	Ar en daja
7.11	Rakenduse kasutajaliides peab teavitama kasutajat ette sessioon aegumisest.	Ette teavitamise aeg peab olema konfigureeritav.	Ar en daja
7.12	Kui vormile sisestatakse mahukaid andmevälju peab kasutajaliides kokku lepitud ajavahemike järel salvetama välja sisu, et sessiooni aegumisel või võrgu katkestuse korral juba sisestatud andmed ei kaoks.	Kui vorm koosneb paljudest väiksest andmeväljadest (nt taotlus), siis jagatakse vorm etappideks ning salvestatakse vastava etapi lõpus.	Ar en daja
7.13	Sisestusvormidel andmete sisestamisel peab saama väljade vahel vastavalt äriloogikale liikuda klaviatuuri abil tabulaatoriga.	Tabuleerimise järjekord tuleb HTML struktuurist. Pigem vältida käsitsi *tabindex*-ite seadmist.	Ar en daja
7.14	Interaktiivsete vormide puhul (näiteks faili üleslaadimine), ei tohiks lehe värskendamisega tegevust korrata (faili taas üles laadida, andmeid saata, avaldust esitada).	-	Ar en daja
7.15	Kui päring võtab aega kauem kui 3 sekundit, peab kasutaja saama visuaalse teate, et süsteem tegeleb päringu läbiviimisega.	Ikoon peab muutuma liivakellaks ja/või kuvatakse teade: päringut sooritatakse või muu tellijaga kokkulepitud indikaator.	Ar en daja
7.16	Esilehel (sisselogimata) ja ka pärast kasutaja sisselogimist peab olema lihtne võimalus teavitada kasutajat muudatustest või probleemidest. Teavitus peab olema halduri poolt lihtsalt lisatav ja olema kasutajale märgatav.	Näiteks võimalikud teavituses: mingi süsteemi osa on vigane, tuli mingi uus funktsionaalsus, vahetage oma parool, uuendage isikuandmeid jne.	Ar en daja
7.17	Infosüsteem peab funktsionaalse vea (näiteks kohustuslikkude väljade täitamata jätmisel) korral kasutajale kuvama kasutajasõbraliku veateate. Veateated peavad olema hallatavad.		Ar en daja
7.18	Päringu vastusena kuvatud tabeli veerge on võimalik andmete/teksti tähestikulises järjekorras sorteerida.		Ar en daja
7.19	Vormide täitmisel peab kasutaja saama ülevaate kohustuslikest andmeväljadest enne vormi täitmist	Vastavalt RIA stiiliraamatule tähistatakse kohustuslikud väljad tekstiga. Kui vormil on kohustuslikke väljasid on rohkem kui mittekohustuslikke siis tähistatakse hoopiski mittekohustuslikud. Kui ametniku rakendus ei kasuta RIA stiiliraamatut siis tähistatakse kohustuslikud väljad tärniga. Oluline on ka meeles pidada, et lähtuvalt WCAG nõuetest https://www.w3.org/TR/WCAG20-TECHS/H90.html peab iga vormi alguses olema legend, mis selgitab kohustuslikke väljasid tähistavat sümbolit. Nt: *tähistab kohustuslikke väljasid	Ar en daja
7.20	Rakenduse andmeväljade mõisted peavad olema üheselt identifitseeritavad, korrektses eesti keeles (ilma kirjavigadeta) ja vajadusel sisaldama selgitavat teksti. Abiinfo (kasutusjuhendid) peab olema kättesaadav rakenduse toimimise erinevatel etappidel.	Korrektne keel ja õigekirjareeglite järgimine on sisuhaldajate ülesanne.	Ar en daja
7.21	Kasutajaliides peab vastama ka dokumendis "Front-end arendusreeglid" kirjeldatud reeglitele	Front-end arendusreeglid	Ar en daja
8. Dokumentatsioon			
8.1	Kogu rakenduse dokumentatsioon peab olema kirjutatud eesti keeles.	Erandiks võivad olla kolmanda osapoole komponentide (mis pole kirjutatud tellija jaoks) dokumentatsioon. Samuti võib erandiks olla välispooltega seotud projektid. Erandid tuleb kooskõlastada tellijaga enne dokumentatsiooni koostamist	Ar en daja

8.2	Lahendus kirjeldatakse RIHA määruse nõuete kohaselt.	https://www.riigiteataja.ee/akt/12933746?leiaKehtiv#para6	Arendaja / Projektijuht / Tellija RIHA haldur
8.3	Rakenduse dokumentatsioon peab sisaldama paigaldusjuhiseid, varundatavate komponentide kirjeldust, kasutajate kasutusjuhendeid, peakasutajate ja administraatorite kasutusjuhendeid, andmemudeleid, arhidektuurilisi mudeleid, süsteemitehnilisi kirjeldusi, nõudeid riistavarale, krüptoalgoritme ja võtmepikkuseid, SSL sertifikaatide kasutuskohti jms	Dokumentatsioon peab olema versioneeritud, muutmiskuupäevadega, autori nimedega, korrektse keelekasutusega, selge struktuuriga. Dokumentatsiooni detailsus peab olema piisav, et sõltumatu kolmas tehnliste IT baasteadmistega isik suudaks dokumendist vajalikke järeldusi teha (st dokument peab olema arusaadav sellele isikule, kuid näiteks paigaldusjuhise järgi toimetades ei pea ta ebaõnnestunud tarnele teostama veaanalüüsi).	Arendaja
8.4	Rakenduse dokumentatsioon peab sisaldama tabelite-andmete-logide mahu kasvu arvestuslikku hinnangut rakenduse sihipärase kasutamise korral ettenähtud arvu kasutajate poolt. (MB/GB kuus /aastas).	Esialgne kirjete mahu hinnang peab tulema lähteülesandest, ning täpsustuma eel ja detailanalüüsi käigus. Mahuhinnang peab sisaldama ka logide säilitamise, arhiveerimise tähtaegu.	Arendaja
8.5	Iga uue versiooniga peab alati välja tooma versiooni muudatuse kirjeldused (release notes).	Release notes peab kajastama kõiki muudatusi eelmise ja uue versiooni vahel.	Arendaja
8.6	Rakenduse dokumentatsioon peab vastama ka dokumendis "Nõuded infosüsteemi dokumentatsioonile" kirjeldatud nõuetele	Nõuded infosüsteemi dokumentatsioonile	Arendaja
9. Versioonihaldus			
9.1	Kõik rakenduse testimiseks, koolituseks või implementeerimiseks üle antavad tarkvarapaketid peavad olema versioneeritud. Kasutama peab Tellija versioonihalduse repositooriumi.	Arendajale antakse selleks õigused Tellija versioonihalduse repositooriumi, kus ta peab hoidma oma erinevaid versioone. Versioonihalduse repositooriumi juurdepääsutaotlus esitatakse Tellija kasutajatoele läbi projektijuhi.	Arendaja
9.2	Nii arendamisel kui ka hooldusleping ute korral kasutatakse Tellija veahalduse keskkonda.	Arendajale antakse selleks õigused Tellija veahalduse keskkonda. Juurdepääsutaotlus esitatakse Tellija kasutajatoele läbi projektijuhi.	Arendaja
10. Paigalduspaketi kooste			
10.1	Tarnitava lahenduse koosseisus üle antava lähtekoodiga peavad kaasas olema kirjeldused sellest paigalduspaketi koosteks.	Näiteks võib lahenduse paigalduspaketi koosteprotsess ette näha, et käivitada tuleb rida shell-käske või võivad lahenduse koosseisus olla valmis (Gradle, ..) koosteskriptid või mis iganes muu moodus paigalduspaketi tekitamiseks. Eelistatud on kasutada GitLab skripte.	Arendaja
10.2	Kooste kirjelduste alusel valmiv paigalduspakett tohib sisaldada ainult minimaalse rakenduse käitamiseks vajamineva failikomplekti.	Näiteks: kompileeritavate keelte puhul ei tohi sisaldada lähtekoodi, kui see pole vajalik rakenduse käitamiseks.	Arendaja
10.3	Kooste kirjelduste alusel valmivat paigalduspaketti peab olema võimalik liigutada erinevate masinate vahel.	Näiteks ei tohi tekitada olukorda, kus rakenduse jooksutamiseks uues serveris tuleb see tingimata just sealsamas kokku kompileerida.	Arendaja
10.4	Administraatoril peab olema võimalus andmebaasi muudatuste skriptide sisus veenduda.		Arendaja