

AI-sovelluksen heuristinen arviointi luottamuksen näkökulmasta

Heuristisen arvioinnin tarkoituksena on kustannustehokkaalla tavalla tarkastella tietojärjestelmän laatua tietyn kriteeristön (eli heuristiikan) avulla. Heuristisessa arvioinnissa voidaan tarkastella olemassa olevaa tai kehitteillä olevaa sovellusta erityisesti käyttäjän kokemuksen näkökulmasta. Olen kehittänyt KITE-projektissa heuristiikkoja tekoälyn *luotettavuuden* arvioimiseksi asiantuntijatyön kontekstissa (alla). Nämä ovat työn alla.

Laatimani heuristiikat pohjautuvat Jakob Nielsenin 1994 käytettävyysheuristiikkoihin, Nielsen Norman Groupin 'heuristics for complex application' -ohjeistukseen (Kaplan, 2021), Euroopan komission luotettavan tekoälyn arviointiin (2019) ja Microsoftin Amershi et al. (2019) ohjeistukseen ihmisen ja tekoälyn vuorovaikutuksen suunnittelusta. Tarkoitus on pohtia, miten tekoäly teknologiana ja toisaalta luotettavuuden näkökulma muuttaa näitä vakiintuneita heuristiikkoja: osa heuristiikoista lienee käyttökelpoisia myös tässä kontekstissa lähes sellaisenaan. On huomioitavaa, että heuristiikat voivat vaihdella käyttöliittymän mukaan (esimerkiksi äänikäyttöliittymä ja graafinen käyttöliittymä). Myös käyttökonteksti ja sovellusala vaikuttavat heuristiikkoihin.

HUOM: heuristiikat eivät ota kantaa datan laatuun, eheyteen, tai yksityisyyteen ja tietosuojaan vaikuttaviin seikkoihin: oletus on, että nämä seikat ovat tarkastettu tuotteen kehittämisvaiheessa, sillä ne ovat olennainen osa tekoälyteknologiaa.

Heuristiikat tekoälytuotteen luotettavuuden arviointiin asiantuntijatyössä (perustuen Nielsenin 1994 heuristiikkoihin) 1. Järjestelmän tilan näkyvyys ja käyttäjän kontrolli

Käyttäjän tulee pystyä seuraamaan ja kontrolloimaan sovelluksen tilaa ja toimintoja helposti ja nopeasti. Käyttäjällä tulee olla varmuus, että sovellus on käynnissä ja toimii tarkoituksenmukaisella tavalla. Anna käyttäjällä mahdollisuus opetella sovelluksen käyttöä ja tarvittaessa perua sovelluksen suorittama toiminto.

AI ja loppukäyttäjät: Tekoälysovellus voi olla upotettuna järjestelmään ja se voi toimia ikään kuin taustalla. Käyttäjän tulee ymmärtää, mitä tehtäviä sovellus suorittaa tai voisi suorittaa, sekä miten ja millaista tietoa sovellus hyödyntää. Motivoi ja rohkaise käyttäjää kokeilemaan tuotetta. Informoi käyttäjää sovelluksen epävarmuuksista. Voit esimerkiksi antaa käyttäjälle mahdollisuuden kysyä suosituksia halutessaan.

AI ja asiantuntijatyö: Käyttäjän kontrollin näkökulmasta on tärkeää, että käyttäjällä on mahdollisuus vaikuttaa sovellukseen ja sen käyttöön omasta kontekstista käsin. Luo toimenpiteitä, joilla käyttäjä voi tarvittaessa keskeyttää tai palauttaa prosessin, ja anna käyttäjälle mahdollisuus vaikuttaa tekoälyn toimintaan järjestelmän käytön eri vaiheissa esim. opettamalla järjestelmää. Käyttäjällä tulee olla mahdollisuus seurata niin tuotteen toimintaa kuin sen vaikutuksia ja mahdollisia riskejä. Esimerkiksi, anna käyttäjälle mahdollisuus valita älykkään automaation aste ja tarvittaessa palauttaa tiukempi kontrolli.

2. Järjestelmän ja tosielämän vastaavuus ja yhtenäisyys

Sovelluksen tulee olla ymmärrettävä käyttäjälle. Ymmärrettävyys pätee niin kirjalliseen ilmaisuun ja viestintään kuin visuaalisiin vihjeisiin käyttöliittymässä. Sovelluksen tulee olla yhteneväinen ja johdonmukainen olemassa olevien tuotteiden ja alan konventioiden kanssa.

AI ja loppukäyttäjät: Mukauta sovellus käyttökontekstiin ja rakenna ymmärrettävyyttä eritasoisille käyttäjille. Auta käyttäjää ymmärtämään, mitä uutta tekoälysovellus tuo tuttuihin toimintoihin. Selkeytä sovelluksen tarkoitus, esimerkiksi, korvaako tai vastaako sovellus aiemmin ihmisen toteuttamaa palvelua, ja onko käyttäjällä mahdollisuus valita ihmisen toteuttama palvelu tekoälysovelluksen sijaan.

AI ja asiantuntijatyö: Sovelluksen tulee integroitua osaksi erityistä asiantuntijatyön kontekstia, lainalaisuuksia, käsitteitä ja normeja. Selkeytä sovelluksen tarkoitus ja sen tuoma lisäarvo käyttäjälle, ja anna käyttäjälle mahdollisuus vaikuttaa tuotteen toimintaan esimerkiksi palautteen kautta. Varmista, että käyttöliittymän kieli on ymmärrettävä eritasoisille teknologian käyttäjille tarjoamalla erilaisia selityksiä ja täydentävää tietoa. Huomioi sovelluksen yhteneväisyys ja vuorovaikutus muiden käytössä tai samankaltaisten järjestelmien kanssa, esimerkiksi ennustejärjestelmien tai päätöksenteon tukijärjestelmien kanssa.

3. Virheiden estäminen, tunnistaminen ja korjaaminen

Käyttäjät olettavat sovelluksen opastavan käyttäjää. Sovelluksen tulee auttaa käyttäjää tunnistamaan mahdolliset virheet ja toimimaan virhetilanteissa. Virheilmoitusten tulee olla selkeitä ja ymmärrettäviä sekä tarjota opastusta virheen korjaamiseen.

AI ja loppukäyttäjät: Informoi käyttäjää selkeästi ja ymmärrettävästi sovelluksen mahdollisista epätarkkuuksista ja virheistä. Luo toimintoja, jotka auttavat käyttäjää vähentämään virheitä ja parantamaan tarkkuutta, esimerkiksi, personoimalla sovellusta. Tarjoa käyttäjälle tietoa tekoälyn toiminnasta ja luo tapoja parantaa järjestelmän tarkkuutta.

AI ja asiantuntijatyö: Tekoälyn tekemien virheiden tunnistaminen voi olla hankalaa, sillä virhe ei välttämättä näyttäydy kuin vasta käytön kautta. Yksittäinen virhe voi kumuloitua ja vääristää tulevia toimintoja. Auta käyttäjää ymmärtämään toimintojen seuraukset ja mahdolliset virhemarginaalit. Rakenna toimintoja, joilla jäljittää ja korjata mahdolliset virheet. On tärkeää, että käyttäjällä on todenmukainen ja ajantasainen ymmärrys tuotteen mahdollisuuksista ja rajoituksista. Informoi käyttäjää tuotteen riskeistä ja vääristymistä, rajoituksista ja puitteista. Tarjoa tietoa, jonka avulla käyttäjä voi opettaa sovellusta, ja rakenna toimintatapoja odottamattomiin ja yllättäviin tilanteisiin, esimerkiksi toiminnon keskeyttämiseen.

4. Selkeä ja tunnistamista korostava suunnittelu

Sovelluksen visuaalisuuden ja informaation tulisi olla tarkkaan harkittua. Sovelluksen toimintojen ja vaihtoehtojen tulee olla tunnistettavia ja pääteltävissä sen sijaan, että käyttäjän tulisi palata käyttöohjeisiin.

AI ja loppukäyttäjät: Opasta käyttäjää sovelluksen tarkoituksenmukaiseen käyttöön hyödyntämällä selkeitä visuaalisia tehosteita. Hyödynnä tuttuuden tunnetta käyttämällä vakiintuneita käyttöliittymäelementtejä mutta tuo selkeästi esiin tekoälyn rooli ja mahdollisuudet sovelluksessa.

AI ja asiantuntijatyö: Suuntaa käyttäjään huomiota kriittisiin tai epävarmoihin toimintoihin. Kiinnitä käyttäjän huomio tarkistettaviin, epätarkkoihin tai kriittisiin toimintoihin visuaalisilla ja selkeillä tehosteilla. Pyri minimoimaan käyttäjän kognitiivinen kuorma poistamalla ylimääräiset huomiota herättävät elementit käyttöliittymästä. Auta käyttäjää mukauttamaan sovellusta omien mieltymystensä mukaisesti, esimerkiksi sulkemalla tiettyjä osioita käyttöliittymästä, mikäli niille ei ole käyttöä.

5. Käytön ohjeistus ja käytön joustavuus

Sovelluksen käytön ohjeistuksen tulee olla helposti ja nopeasti löydettävissä ja opastaa käyttäjää toimintaan ymmärrettävällä, selkeällä ja tilannekohtaisella tavalla. Sovelluksen käytön tulee olla joustavaa ja tehokasta sekä aloitteleville että edistyneille käyttäjille.

AI ja loppukäyttäjät: Tarjoa erilaisille käyttäjille sopivaa tietoa ja selitettävyyttä ymmärrettävässä muodossa, mutta älä ylikuormita käyttäjää käyttöohjeilla. Anna käyttäjälle mahdollisuus personoida sovellusta omien tavoitteiden tai mieltymysten mukaan niin käytön alussa kuin käytön aikanakin. Tarjoa lisätietoa siitä kiinnostuneille.

AI ja asiantuntijatyö: Auta käyttäjää ymmärtämään sovelluksen oppivat mahdollisuudet ja tee selkeäksi käyttäjän mahdollisuus vaikuttaa sovellukseen esimerkiksi palautteen kautta tai sovellusta opettamalla. Informoi käyttäjää siitä, miten tekoäly toteuttaa tavoitteensa, esimerkiksi visualisoimalla tai viestimällä sovelluksen tarkkuudesta tai ajansäästöstä. Tue tehokasta ja täydentävää tiedonsaantia tarjoamalla käyttäjälle mahdollisuus helppoon ja nopeaan järjestelmätuen kontaktiin ja lisätietoon tarvittaessa.