Kuntalaisen rajapinta sähköiseen asiointiin

Kalle Launiala, ProtonIT Oy

kalle.launiala@protonit.net, +358 44 5575665

Esityksen jäsennys

- Käyttäjän tunnistaminen sähköisessä asioinnissa
  - Käyttäjän tunnistaminen vs. Käyttäjän rooli
- Käytännön nykytilanne
  - Epäselvät roolit, kotien digitalisoituminen
- Roolikohtaiset palvelut
  - Roolien tunnistaminen palveluissa
- Teknisen arkkitehtuurin migraatio roolien hallintaan
  - Käyttäjän omien tietojensa ja palveluiden hallinnan mahdollistaminen arkkitehtuurikontrollin mahdollistamana

Käyttäjän tunnistaminen sähköisessä asioinnissa

Käyttäjän ja roolin eriyttäminen

Käyttäjän tunnistaminen

- Tunnus ja salasana
- Oman laitteen ja kotikoneen tilin tunnus
- Oman laitteen iTunes/Google/Windows Live-tili
  - Vahva kytkös; ostetut appsit/palvelut liittyvät näihin
- Pankkitunnukset
- Sähköinen henkilökortti

= Kuka käyttäjä (järjestelmän mielestä) on

Kuntalaisen/kansalaisen rooli

- Rooli on ATK-järjestelmän ulkopuolinen ns. hyväksyttävä tosiasia

- Esim: Yhteisesti sovittu, etuun oikeuttava asema
  - Opiskelija
  - Työtön
  - Eläkeläinen
  - Yrittäjä

- Esim: Yksilöiden välisen perhesuhteen tai sopimuksen tulos
  - Huoltajan suhde alaikäiseen lapseen
  - Omaisen hoitosuhde hoidettavaan

= Rooli määrittää, mistä käyttäjä saa päättää ja kenen puolesta

Käytännön nykytilanne

Nykyiset järjestelmät, laitteet ja palvelut

Nykytilanne – epäselvät roolit

- Tunnistautumisessa usein yksi taso heikko tai vahva – ristiriita tarpeessa
  - Vahva on teknisesti luotettava ja helpommin ”oikein” toteutettu
  - Vahvalla tunnistautumisella sopimukset voivat olla juridisesti pitäviä
  - Miksi pitää käyttää vahvaa tunnistusta vain katsoakseen omia tietojaan?

- Heikolla tunnistautumisella ei saa tehdä mitään
  - Käyttäjä kuitenkin käyttää yksityisiä tietojaan ”heikolla tilillä”

- Vahva tunnistautuminen on kankea käyttää – ei ole käytännön realismia
  - ”Henkkarit tarvitaan joka paikassa asioidessa”
  - Nykyisissä ”ei-digitaalisissa” palveluissa samalla ”tiukkuustasolla” puhelimitse ei voitaisi palvella käytännössä lainkaan

- Käyttäjän kulloinenkin rooli ja päätöksenteko epäselvää yhtään monimutkaisemmassa asiassa

Nykytilanne – kotien digitalisoituminen

- Kotien digitalisoituminen + henkilökohtaiset laitteet
- Laitekohtaiset tilit ohjaavat henkilökohtaiseen tiliin
- Jaettu kotikone käyttää samaa usein yhtä kirjautumistiliä
  - Käyttäjien henkilökohtaisia tilejä ei voida hyödyntää
  - Jopa selainpohjaisessa tunnistuksessa on tietosuoja-ongelmia ”kirjaudu ulos” ja ”tyhjennä välimuisti”

Roolikohtaiset palvelut

Tunnistautumisen ja roolin erottaminen

Erotetaan tunnistaminen roolista

- Tekninen käyttäjän tunnistus sisältämään vahvuus, roolin tunnistaminen
  - Käyttäjä voi itse päättää, mitä tietoja millä tasolla näkee
  - Vahvuus-arvoa käytetään roolikohtaisissa vaatimuksissa
  - Sovellusarkkitehtuurissa liitetään roolin käsittely tunnistautumisen yhteyteen

- Päätöksenteko/allekirjoitus asianmukaisella tasolla
  - Asian valmistelu käyttäjän päässä vapaasti
  - Päätöksen/sopimuksen tekeminen selkeyttää roolin
  - Roolikohtainen päätös ja asia vaikuttaa vahvuustasoon

- Ulottuu sujuvasti myös tiedon näkymiseen roolista riippuen
  - Esim: kun hoidetaan edunvalvojana toisen henkilön asioita, vaaditaan vahva tunnistautuminen jo tietojen katseluunkin

Mahdollistaa nykyisten puhelinpalveluiden joustavuuden ja juridisesti pitävät sopimukset

Siirtyminen roolikohtaisiin palveluihin

- Nykyjärjestelmien modernisoivassa ylläpidossa tunnistetaan roolit
  - Järjestelmäkohtaisesti ensin ”kaikki ominaan”
  - Sujuva askellus yhdistää asteittain – esim. tilojen varaaminen on kaikkien kaupunkilaisten palvelu
  - Todelliset julkiset roolit saadaan eri viranomaisrekistereistä
    - Opintotuki, työttömyystuki jne..
    - Haut voidaan tehdä nykyisten lakien puitteissa

- Sähköisen asioinnin palvelukirjastoon roolit mukaan
  - Käytettävissä olevat sähköiset palvelut suodattuvat käyttäjän roolin mukaan
  - Kirjastot voivat olla avoimia, koska rooli vahvistetaan päätöksentekohetkellä

- Mahdollistaa yksityisen sektorin liittymisen palveluihin
  - Tiedot käyttäjäkohtaisissa varastoissa (= tunnistautumisen takana)
  - Päätökset roolikohtaisesti pankkitunnuksin ”hyväksytkö tämän muutoksen”
  - Yhtenäinen luotettava tapa hoitaa asioita – mahdollistaa mikroyritysten palvelut

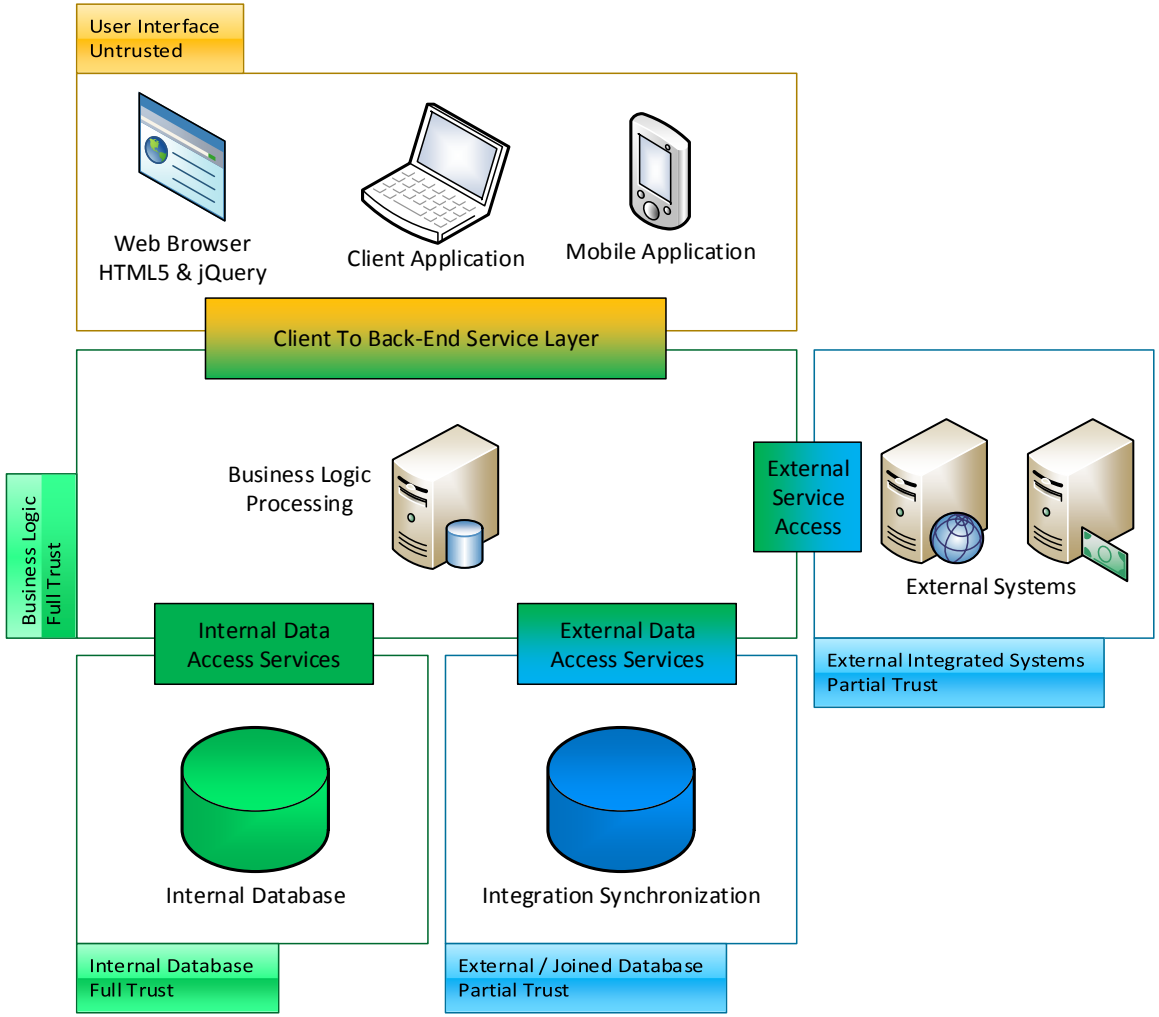
Migraatio roolien hallintaan

Tekninen nykytilanne, looginen migraatio

Nykytilanne

- Käyttäjän tekninen tunnistaminen tapahtuu käyttöliittymä-rajapinnassa
- Prosessointi on täydellisen tietoinen, kuka on aktiivinen käyttäjä
- Tietojen haut tehdään aktiivisen käyttäjän ”where” filteröinnillä tavalla tai toisella: **Käyttäjien tietoja ei sekoiteta!**
- Prosessointi päättelee käyttäjän roolin tapauskohtaisesti
- Rooli ”hallitaan” järjestelmän määrittely/speksitasolla
- Rooli ”hallitaan” useimmin pelkästään edellyttämällä teknisesti vahvaa tunnistautumista

Käyttäjän reaalimaailman roolia ei ole hallittu käytännössä lainkaan!



Teknisen arkkitehtuurin laajennus

Pelkästään käyttäjän tunnistaminen

- Tekninen tunnistus
  - Aktiivinen käyttäjä
- Roolin vaatimukset koodattu toteutukseen
  - Määrittelyjen mukaan

Käyttäjällä ei kontrollia omaan rooliinsa!

Käyttäjän roolin tunnistaminen

- Tekninen tunnistus
  - Aktiivinen käyttäjä
  - + Tunnistuksen luotettavuustaso
- Roolin vaatimukset koodattu toteutukseen
  - Määrittelytasolla tarkennus rooliin
  - Roolin hallinta mahdollista hallituilla ”if”-rakenteilla

Tuodaan rooli hallittavaksi myös käyttäjätasolle ja käyttöliittymään!

Arkkitehtuurin laajentamisen tehokas hallinta

- Laajennetaan ”aktiivinen käyttäjä”-kontekstia
  - On olemassa jo jokaisessa järjestelmässä
- Voidaan hallita modulaarisella ADM-automaatio-arkkitehtuurilla
  - Tarpeen mukaisen koodigenerointi-automaation tukemana
  - ”Kuten käsityönä toteutettaisiin laajennukset best-practice tasolla”
  - Täysin analoginen kooditason ”if”-rakenteiden ja niitä vastaavien määrittelymuutosten kanssa
  - Käytännössä ”if” rakenteet supistuvat tiedon näyttämiseen ja tallennuksen eriyttämiseen eli ”käyttäjän päätöksentekoon”
- Käyttäjä voi itse päättää, haluaako esim. nähdä tietonsa ”Google”-accountin turvatason tunnistautumisen kautta
  - Voidaan toteuttaa sujuva ”vahva tunnistautuminen” kesken session
  - Usean yhtaikaisen tunnistuksen hallinta järjestelmän toimesta
  - Selkeyttää käyttäjän kokemusta ”katselu” ja ”päätöksenteko” roolien osalta

Mahdollistaa kertakirjautumisen koko järjestelmäverkostoon hallittavasti!

Käyttäjän omien tietojensa hallinta

Arkkitehtuuritason kontrollin luontainen laajentaminen

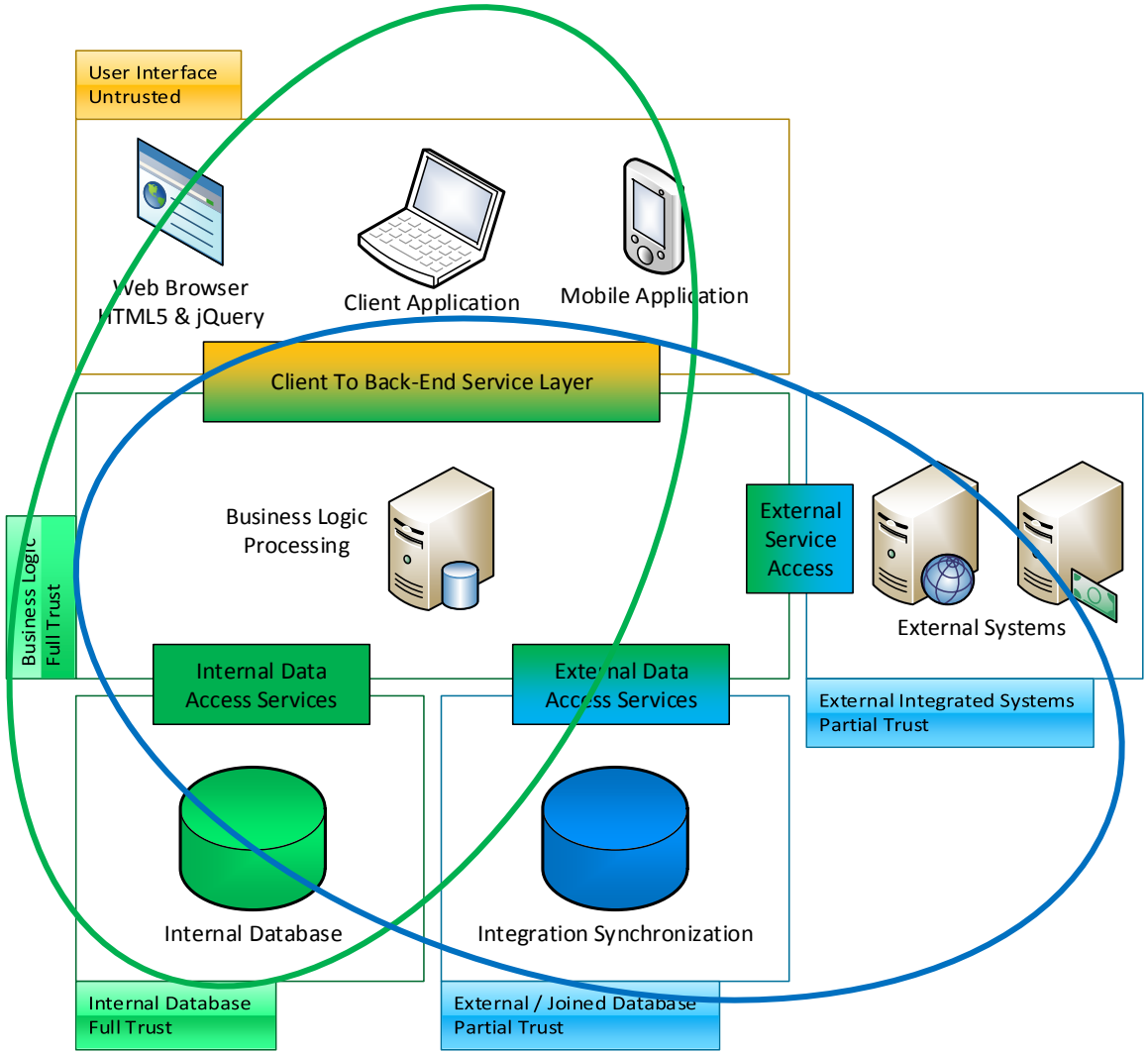
Käyttäjän kontrollin lisääminen

- Arkkitehtuurin laajennus realisoi prosessoinnin käyttäjäkohtaisessa kontekstissa

- Käyttäjälle voidaan tarjota mahdollisuus mukauttaa omaa kontekstiaan
  - Digitaaliset moduulit ja palvelut hyväksytyistä lähteistä
  - Koskee myös ns. teknisiä core-palveluita

Tiedon tallennus voidaan tehdä käyttäjän kontrolloimalla tavalla!

Integraatiot voidaan tehdä käyttäjän kontrolloimalla tavalla ja käyttäjän valitsemiin järjestelmiin!



Materiaalin loppu

Muistilista ja ”työ-yhteenvetoa” tämän jälkeen...

Toteutus-Yhteenveto-Muistilista

- Käyttäjän tunnistaminen
  - Tekninen tunnistaminen = autentikointi
  - Roolin/valtuuksien tunnistaminen = autorisointi
  - Käyttäjän oman roolinsa ymmärtäminen

- Nykytilanne
  - Ongelma: vahva tunnistautuminen ja roolit menevät järjestelmissä sekaisin
  - Käyttäjäkohtainen hallinta käytännössä
  - Teknisesti kirjautumistunnus/laite identifioi teoriassa
  - Käytännössä: jaetut selaimet, jaetut laitteet
  - Reality check
    - sujuva asiointi edellyttää automaattista tunnistautumista
    - Autorisointi/päätökset edellyttävät tosiaikaista vahvaa valtuutusta

- Käyttäjän oman informaationsa/datansa hallinta
  - Avoimet rajapinnat != avoin data
  - Prosessoinnin autorisointi
    - Edellyttää hallintaroolia
  - Teoriassa ”voi olla useita totuuksia samasta datasta ns. Appikohtainen siilo”
  - Käytännössä: käyttäjälle on yksi totuus, eli luotettava prosessointi ”reaalidataa vasten”

- Toteutus
  - Auditoidut arkkitehtuurirakenteet, kuten nytkin ”edellytetään tämän speksin mukaista”
  - Best practice voidaan saada suoraan toimittajalta; esim. Microsoftin maaorganisaation avustamana