

© Pekka Huttunen 2020

# Järjestelmäarkkitehtuurin määrittely

Janne Bragge  
21.2.2025

## **Kajaani Ammattikorkeakoulu**

PL 52, Ketunpolku 1  
87101 KAJAANI  
kajaanin.amk@kamk.fi

Copyright 2024 KAMK. All rights reserved.

<b>1</b>	<b>JOHDANTO .....</b>	<b>4</b>
1.1	JÄRJESTELMÄN YLEISKUVAUS .....	4
1.2	KAAVIOKUVAUS PROSESSISTA "KUITISTA KIRJANPIDON KIRJAUKSEKSI" .....	4
1.3	JÄRJESTELMÄN KÄYTTÄJÄT .....	5
1.4	JÄRJESTELMÄN KÄYTTÖ .....	5
1.4.1	<i>Mobiilisovellus</i> .....	5
1.4.2	<i>Taustajärjestelmä</i> .....	5
1.5	SOVELLUKSEN ASENNUS .....	6
1.5.1	<i>Mobiilisovellus</i> .....	6
1.5.2	<i>Taustajärjestelmä</i> .....	6
1.6	KERÄTTÄVÄT TIEDOT .....	7
<b>2</b>	<b>JÄRJESTELMÄN VAATIMUKSET .....</b>	<b>8</b>
<b>3</b>	<b>JÄRJESTELMÄARKKITEHTUURI .....</b>	<b>9</b>
3.1	KOMPONENTIT .....	10
3.2	KAAVIOKUVAUS ARKKITEHTUURISTA .....	10
	.....	10
<b>4</b>	<b>JÄRJESTELMÄN TIETOMALLI .....</b>	<b>11</b>
4.1	TIETOKANTARAKENNE .....	11
<b>5</b>	<b>JÄRJESTELMÄN RAJAPINNAT .....</b>	<b>12</b>
5.1	REST API .....	12
5.2	TIETOKANTARAJAPINTA .....	12
5.3	WEB-KÄYTTÖLIITTYMÄN RAJAPINTA .....	12
5.4	MOBIILISOVELLUKSEN ASENNUS JA POISTO .....	12
5.5	TAUSTAJÄRJESTELMÄN ASENNUS JA PÄIVITYS .....	12
<b>6</b>	<b>TIETOTURVA .....</b>	<b>13</b>
6.1	AUTENTIKOINTI JA KÄYTTÖOIKEUDET .....	13
6.2	TIETOJEN SALAUS .....	13
6.3	LOKITUS JA VALVONTA .....	13
6.4	DATAN EHEYS JA VARMUUSKOPIOT .....	13
6.5	TIETOSUOJAKÄYTÄNNÖT .....	14
<b>7</b>	<b>INTEGRAATIOTESTAUUS .....</b>	<b>15</b>
7.1	INTEGRAATIOTESTIT .....	15
7.1.1	<i>Integraatiotesti 1</i> .....	16
7.1.2	<i>Integraatiotesti 2</i> .....	17
7.1.3	<i>Integraatiotesti 3</i> .....	18
7.1.4	<i>Integraatiotesti 4</i> .....	19
7.1.5	<i>Integraatiotesti 5</i> .....	20

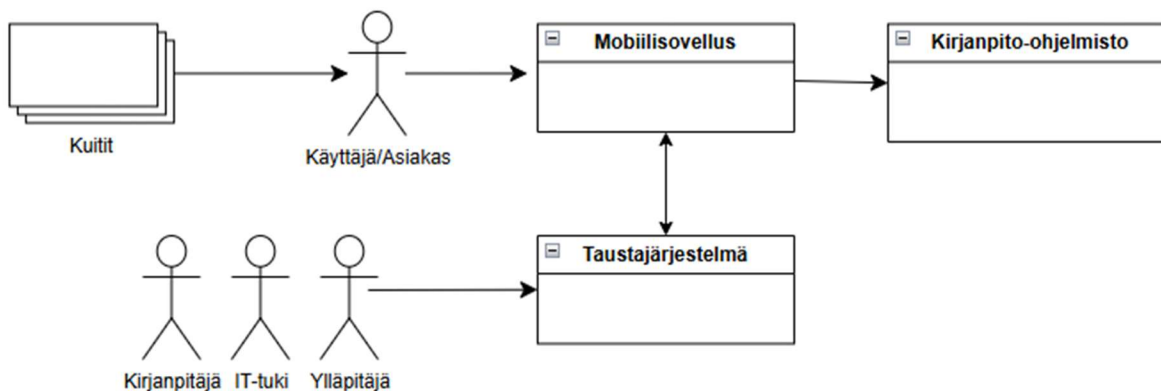
# 1 Johdanto

## 1.1 Järjestelmän yleiskuvaus

Tämä järjestelmä koostuu mobiilisovelluksesta ja taustajärjestelmästä. Mobiilisovelluksella yrityksen asiakkaat voivat kuvata kuitteja, joista sovellus poimii automaattisesti tiedot kulukorvauslomakkeelle tai matkalaskulle. Lisäksi järjestelmä integroituu suoraan kirjanpitosovellukseen, jolloin kuitista saadut tiedot voidaan välittää ja tallentaa automaattisesti ilman manuaalista syöttöä.

Järjestelmän tavoitteena on vähentää manuaalista kirjanpitotyötä, parantaa tehokkuutta ja varmistaa datan eheys automaattisella käsittelyllä.

## 1.2 Kaaviokuvaus prosessista ”kuitista kirjanpidon kirjaukseksi”



## 1.3 Järjestelmän käyttäjät

Käyttäjärooli	Kuvaus
Käyttäjä	Yrityksen asiakas, joka skannaa kuitteja mobiilisovelluksen avulla.
Ylläpitäjä	Ylläpitäjä, joka hallitsee sovelluksia ja käyttäjiä hallintanäkymässä.
Kirjanpitäjä	Käyttää taustajärjestelmää kuittidatan tarkasteluun sekä tarkistaa ja hyväksyy kirjanpitosovellukseen siirretyt tiedot.
IT-tuki	Vastuussa järjestelmän ylläpidosta ja vianhallinnasta

## 1.4 Järjestelmän käyttö

### 1.4.1 Mobiilisovellus

- **Kuittien skannaus:** Käyttäjä voi kuvata kuitin sovelluksen kameralla, ja mobiilisovellus poimii siitä tarvittavat tiedot automaattisesti.
- **Tietojen tarkistus:** Käyttäjä voi tarkistaa ja muokata tunnistettuja tietoja ennen tallennusta.
- **Tietojen lähetys:** Kuitin tiedot tallennetaan pilvipalveluun ja synkronoidaan kirjanpitosovellukseen.
- **Historiatiedot:** Käyttäjä voi tarkastella aiemmin skannattuja kuitteja ja hakea niitä eri hakukriteereillä.
- **Taustajärjestelmä** toimii kirjanpitäjän, it-tuon ja sovellusylläpitäjien väylänä järjestelmään ja sen asetuksiin roolien mukaisesti

### 1.4.2 Taustajärjestelmä

- **Kuittitietojen käsittely:** Mobiilisovelluksen poimimat tiedot validoidaan ja tallennetaan tietokantaan.

- **Hallintanäkymä:** Ylläpitäjät ja kirjanpitäjät voivat tarkastella kuitteja, ei kuitenkaan hyväksyä tai poistaa kuitteja.
- **Kirjanpitosovelluksen synkronointi:** Hyväksytyt kuitit siirtyvät automaattisesti kirjanpitojärjestelmään oikeille tileille. Kirjanpitäjä voi muokata, mikäli jotkin kuitit tarvitsevat erillistarkastuksen.

## 1.5 Sovelluksen asennus

### 1.5.1 Mobiilisovellus

Mobiilisovellus on ladattavissa **App Storesta ja Google Playsta**. Käyttäjä voi asentaa sovelluksen helposti noudattamalla seuraavia vaiheita:

- Etsi sovellus sovelluskaupasta.
- Lataa ja asenna sovellus mobiililaitteelle.
- Kirjaudu sisään tai luo käyttäjätili.

Poisto tapahtuu poistamalla sovellus laitteelta normaalilla tavalla. Poistaminen ei vaikuta jo tallennettuihin kuitteihin, sillä tiedot ovat tallennettuna pilvipalveluun.

### 1.5.2 Taustajärjestelmä

#### 1.5.2.1 Kirjanpitotoimisto

Taustajärjestelmä toimii pilvipohjaisena SaaS-ratkaisuna, jota ylläpitää järjestelmäkehittäjä. Palvelu tarjoaa skaalautuvan ja jatkuvasti päivitettävän infrastruktuurin, joka mahdollistaa käyttäjille saumattoman integraation kirjanpitosovelluksiin ja reaaliaikaisen tiedonhallinnan ilman paikallisia asennuksia.

#### 1.5.2.2 Sovelluskehittäjä

Taustajärjestelmä voidaan asentaa pilvipalveluun (AWS, Azure) tai on-premise-palvelimelle.

Asennusprosessi:

- Asenna tarvittavat palvelinkomponentit (Node.js / Python, PostgreSQL, API-palvelut).

- Määritä tietokanta ja varmista yhteensopivuus kirjanpitosovelluksen kanssa.
- Ota käyttöön tietoturvatoinninnot, kuten TLS 1.3 -salauk ja OAuth 2.0 -autentikointi.

Päivitykset tehdään automaattisesti CI/CD-putken kautta, mikä takaa järjestelmän jatkuvan toiminnan ilman käyttökatkoja.

## 1.6 Kerättävät tiedot

- **Käyttäjätiedot:** Nimi, sähköposti, käyttäjärooli.
- **Kuititiedot:** Päivämäärä, summa, ALV-luokka, kuva kuitista.
- **Kirjanpitosetiedot:** Tiliöinti, hyväksyntätila, synkronointihistoria.
- **Lokitiedot:** Virheet, yhteysongelmat, käyttäjätoiminnot.

## 2 Järjestelmän vaatimukset

- **Automaattinen kuitintunnistus OCR-tekniikan avulla**, joka tukee useita kieliä ja eri kuittiformaatteja.
- **Turvallinen ja GDPR-yhteensopiva tietojen käsittely**, jossa tiedot salataan sekä siirron että tallennuksen aikana.
- **Pilvipohjainen taustajärjestelmä**, joka tukee suuria tietomääriä ja tarjoaa korkean saatavuuden ja skaalautuvuuden.
- **REST API -rajapinnat** tietojen siirtämiseen mobiilisovelluksen, taustajärjestelmän ja kirjanpitosovelluksen välillä.
- **Hallintapaneeli**, joka mahdollistaa käyttöstatistiikan seurannan, virhetilanteiden analysoinnin ja käyttäjähallinnan.
- **Sovelluksen asennus ja poisto** sekä automaattiset päivitykset.
- **Vikatilanteiden hallinta ja palautumismekanismit**, mukaan lukien virhelokit, käyttäjälmoitukset ja tukiticketointi.
- **Integraatio kirjanpitosovellukseen**, jotta kuitista poimitut tiedot voidaan viedä suoraan oikeille kirjanpitotileille.



## 3 Järjestelmäarkkitehtuuri

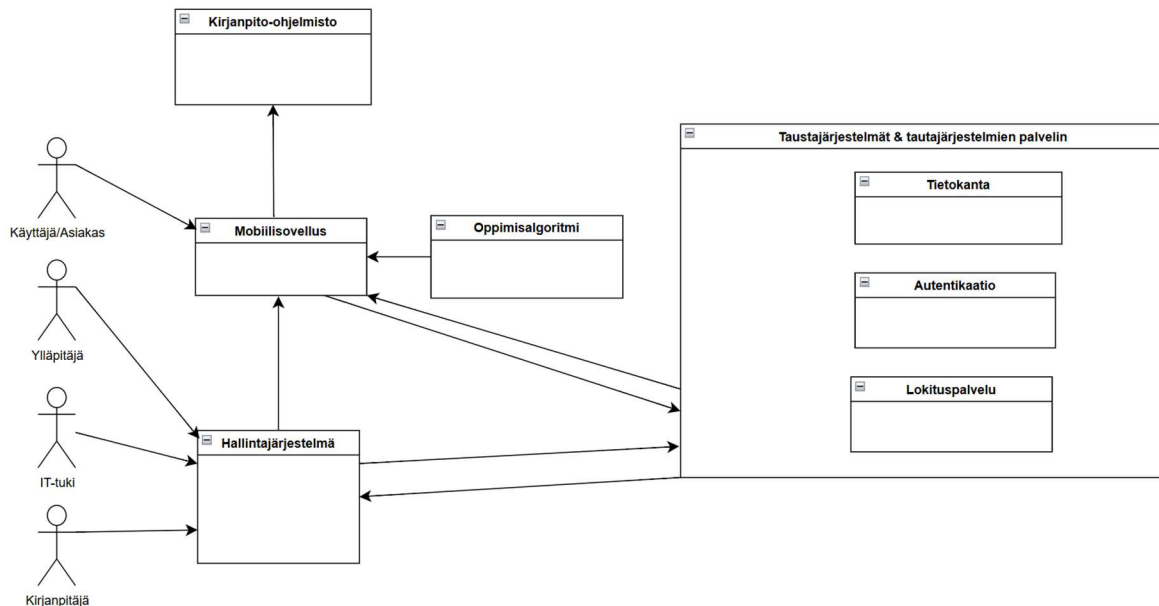
Järjestelmä koostuu neljästä pääkomponentista:

- **Mobiilisovellus** (React Native, oppimisalgoritmi, offline-tuki)
- **Taustajärjestelmä** (Node.js/Python-palvelin, PostgreSQL-tietokanta, mikropalveluarkkitehtuuri)
- **Hallintanäkymä/-järjestelmä** (React.js-pohjainen web-käyttöliittymä, käyttöoikeuksien hallinta)
- **Kirjanpitojärjestelmän integraatio** (API-rajapinta taloustietojen siirtoa varten)

### 3.1 Komponentit

Nimi	Tyyppi	Kuvaus
Mobiilisovellus	React Native	Skannaa kuitit ja lähettää tiedot taustajärjestelmään.
Oppimisalgoritmi	PyTorch	Tunnistaa kuitin tiedot ja muuntaa ne tekstiksi.
Taustajärjestelmän palvelin	Node.js / Python	Palvelinohjelmisto, joka vastaanottaa ja prosessoi datan.
Tietokanta	PostgreSQL	Tallentaa käyttäjien ja kuitintunnistuksen tiedot.
Hallintajärjestelmä	React.js	Hallintanäkymä sovellusten ja tilastien hallintaan.
Kirjanpito-ohjelmisto	API-Integraatio	Automaattinen tietojen siirto kirjanpitojärjestelmään
Autentikaatio	OAuth 2.0 / JWT	Käyttäjien tunnistautuminen ja käyttöoikeuden hallinta
Lokituspalvelu	Lokiohjelmisto	Virhelokien ja järjestelmän suorituskyvyn seuranta

### 3.2 Kaaviokuvaus arkkitehtuurista

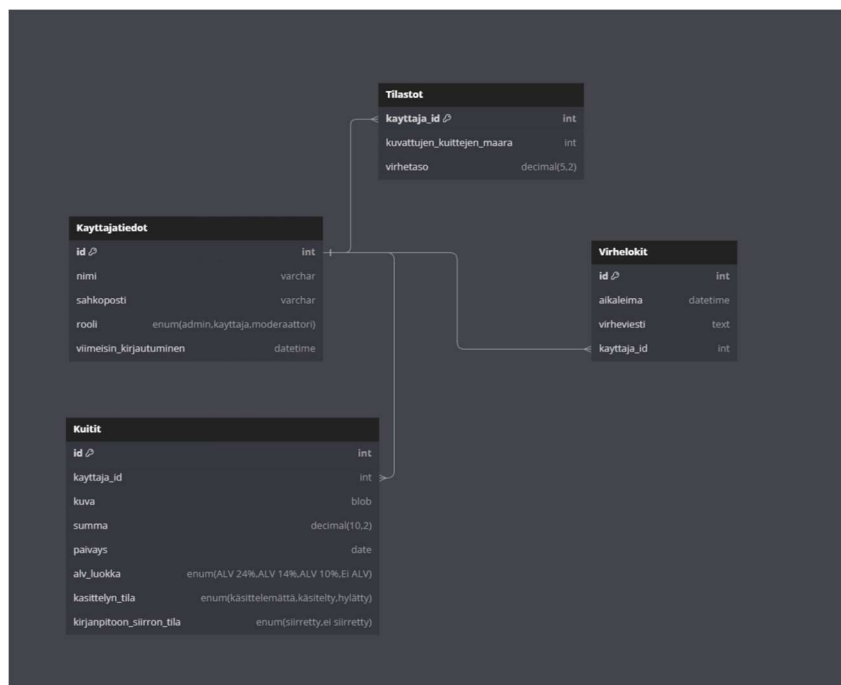


## 4 Järjestelmän tietomalli

Järjestelmä käyttää SQL-pohjaista relaatiotietokantaa kuittidatan, käyttäjätietojen ja hallintalokin tallentamiseen.

### 4.1 Tietokantarakenne

- **Käyttäjätiedot:** ID, nimi, sähköposti, rooli, viimeisin kirjautuminen.
- **Kuitit:** ID, käyttäjä-ID, kuva, summa, päiväys, ALV-luokka, käsittelyn tila, kirjanpitoon siirron tila.
- **Statistiikka:** Käyttäjäkohtainen tilasto kuvattujen kuittien määrästä ja virhetasosta.
- **Virhelokit:** Tapahtumaloki järjestelmän virheistä ja häiriöistä.



## 5 Järjestelmän rajapinnat

### 5.1 REST API

Tarjoaa CRUD-toiminnot kuitteihin, käyttäjiin ja hallintanäkymän tietoihin liittyen. Tukee OAuth 2.0 -autentikointia.

### 5.2 Tietokantarajapinta

SQL-pohjainen yhteys tietokantaan, jossa kuitit ja käyttäjätiedot tallennetaan. Käytössä ovat normaalimuotoiset tietomallit ja indeksit suorituskyvyn optimoimiseksi.

### 5.3 Web-käyttöliittymän rajapinta

Hallintapaneelin ja palvelimen välinen API, joka mahdollistaa datan noutamisen, analysoinnin ja käyttöoikeuksien hallinnan.

### 5.4 Mobiilisovelluksen asennus ja poisto

Mobiilisovellus voidaan asentaa ja poistaa sovelluskaupasta (App Store / Google Play). Sovellus tukee automaattisia päivityksiä ja offline-käyttöä.

### 5.5 Taustajärjestelmän asennus ja päivitys

Palvelu voidaan asentaa pilviympäristöön (AWS, Azure) tai on-premise-palvelimille. Päivitykset tehdään automaattisesti CI/CD-putken kautta ilman käyttökatkoja.

## 6 Tietoturva

Tietoturva on keskeinen osa järjestelmän suunnittelua ja toteutusta. Seuraavat mekanismit on otettu käyttöön järjestelmän tietoturvan varmistamiseksi:

### 6.1 Autentikointi ja käyttöoikeudet

- Käyttäjäidentiteetin hallintaan käytetään **OAuth 2.0 / JWT**-pohjaista autentikointia.
- Käyttöoikeudet määritellään **roolipohjaisen pääsynhallinnan (RBAC)** avulla.
- Ylläpitäjät voivat hallita käyttöoikeuksia hallintanäkymän kautta

### 6.2 Tietojen salaus

- Kaikki järjestelmään syötetyt tiedot salataan **AES-256**-salauksella ennen tallennusta tietokantaan.
- Tiedonsiirto mobiilisovelluksen, palvelimen ja kirjanpitosovelluksen välillä tapahtuu **TLS 1.3**-salauksella.

### 6.3 Lokitus ja valvonta

- Järjestelmässä on **lokituspalvelu (ELK Stack / AWS CloudWatch)**, joka tallentaa kaikki järjestelmän tapahtumat ja virheet.
- Epänormaalista toiminnasta tai tunkeutumisyrityksistä ilmoitetaan **hälytysjärjestelmän** kautta ylläpidolle reaaliajassa.

### 6.4 Datan eheys ja varmuuskopiot

- Järjestelmä käyttää **tietokannan replikaatiota** varmistaakseen jatkuvan toiminnan ja datan saatavuuden.

- Tiedot varmuuskopioidaan **päivittäin** ja varmuuskopiot säilytetään turvallisesti eri palvelinsijainneissa.

## 6.5 Tietosuojakäytännöt

- Järjestelmä noudattaa GDPR-asetuksia varmistaakseen käyttäjien tietosuojan.
- Käyttäjillä on oikeus pyytää tietojensa poistamista järjestelmästä.

## 7 Integraatiotestaus

Testataan eri komponenttien tai moduulien yhteensopivuutta ja toiminnallisuutta yhdessä. Se varmistaa, että yksittäiset osat kommunikoivat oikein keskenään ja että järjestelmä toimii odotetulla tavalla.

### 7.1 Integraatiotestit

- Rajapintojen testaus: Tarkistetaan, että eri moduulit, palvelut tai järjestelmät vaihtavat tietoa oikein.
- Tietovirtojen varmistaminen: Testataan, että data kulkee oikealla tavalla eri järjestelmäosien välillä.
- Virheenkäsittelyn testaus: Selvitetään, kuinka järjestelmä reagoi odottamattomiin tilanteisiin, kuten virheelliseen syötteeseen tai verkko-ongelmiin.
- Reaalimaailman käyttötilanteiden simulointi: Varmistetaan, että järjestelmän eri osat toimivat yhdessä odotetulla tavalla.

### 7.1.1 Integraatiotesti 1

.	Testitapaus 1 (ylätaso)
<b>Tunnus ID</b>	ITC1
<b>Nimi</b>	AI Receipt Data Extraction
<b>Testitapauksen kuvaus</b>	Testataan tekoälyn kykyä poimia kuitista tiedot (päivämäärä, summa, ALV-luokka jne.) ja tallentaa ne tietokantaan.
<b>Testiympäristö ja järjestelmien väliset integraatiot</b>	<p>Tässä testitapauksessa testataan mobiilisovelluksen, tekoälypalvelun ja taustajärjestelmän integraatiota. Testin aikana seuraavat järjestelmät ovat vuorovaikutuksessa keskenään:</p> <ol style="list-style-type: none"> <li>1) Mobiilisovellus -&gt; Tekoälyalgoritmi</li> <li>2) Tekoälyalgoritmi -&gt; Taustajärjestelmä</li> <li>3) Taustajärjestelmä -&gt; Mobiilisovellus</li> </ol>
<b>Testausmenetelmä</b>	<ol style="list-style-type: none"> <li>1. Käyttäjä lataa kuvan kuitista mobiilisovellukseen</li> <li>2. Sovellus lähettää kuvan tekoälypalvelulle analysoitavaksi</li> <li>3. Tekoäly poimii kuitista tarvittavat tiedot ja palauttaa ne taustajärjestelmään</li> <li>4. Tiedot tallennetaan tietokantaan ja näytetään mobiilisovelluksessa käyttäjälle</li> <li>5. Tarkistetaan, että kaikki tiedot ovat oikein ja vastaavat alkuperäistä kuittia</li> </ol>
<b>Hyväksymiskriteerit</b>	<p>Hyväksymiskriteerit:</p> <p>Kaikki alatason testit tehty hyväksytysti:</p> <ul style="list-style-type: none"> <li>- A1: Kaikki kuitista poimitut tiedot ovat oikein tietokannassa (päivämäärä, summa, ALV-luokka jne.).</li> <li>- A2: Vähintään 80 % kuiteista prosessoidaan ilman virheitä tekoälyn toimesta</li> <li>- A3: Käyttäjä voi tarkistaa ja muokata tietoja sovelluksessa</li> </ul>



## 7.1.2 Integraatiotesti 2

.	Testitapaus 2 (Alatso)
<b>Tunnus ID</b>	ITC1A1
<b>Nimi</b>	AI Receipt Data Extraction A1
<b>Testitapauksen kuvaus</b>	Testataan tekoälyn kykyä poimia kuitista tiedot (päivämäärä, summa, ALV-luokka jne.) ja tallentaa ne tietokantaan.
<b>Testiympäristö ja järjestelmien väliset integraatiot</b>	1) Varmista, että mobiilisovellus on asennettuna testilaitteelle. 2) Tarkista, että tekoälypalvelu ja taustajärjestelmä ovat toiminnassa. 3) Varmista, että tietokantayhteys on aktiivinen ja lokitus on päällä virheiden analysoimiseksi.
<b>Testausmenetelmä</b>	Vaihe 1: Käyttäjä lataa kuittikuvan mobiilisovellukseen Avaa mobiilisovellus ja siirry kuittiskannaus-näkymään. Kuvaa testikuitin sovellukseen  Vaihe 2: Tekoäly analysoi kuitin Varmista, että tekoälypalvelu vastaanottaa ja käsittelee pyynnön. Tekoäly poimii kuitista: päivämäärän, summan, ALV-luokan, myyjän tiedot  Vaihe 3: Sovellus tallentaa tiedot Taustajärjestelmä vastaanottaa JSON-vastauksen ja tallentaa tiedot SQL-tietokantaan  Vaihe 4: Käyttäjä tarkistaa tiedot sovelluksessa Mobiilisovellus tekee GET-pyyynnön kuitin tietojen hakemiseksi taustajärjestelmästä
<b>Hyväksymiskriteerit</b>	Hyväksymiskriteerit:  - Tekoäly poimii kuitin tiedot oikein ja palauttaa ne taustajärjestelmään. - SQL-kysely suoritetaan ilman virheitä, ja tiedot tallentuvat tietokantaan. - Sovellus näyttää kuitin tiedot oikein, ja käyttäjä voi muokata niitä tarvittaessa.

### 7.1.3 Integraatiotesti 3

.	Testitapaus 3
<b>Tunnus ID</b>	ITC6
<b>Nimi</b>	User Registration and Login with Mobile Certificate and Bank Authentication
<b>Testitapauksen kuvaus</b>	Testataan, että käyttäjä voi rekisteröityä ja kirjautua sisään onnistuneesti mobiilivarmenteella tai pankkitunnistautumisella. Tämä integraatiotesti varmistaa, että mobiilisovellus, taustajärjestelmä, tietokanta ja ulkoinen tunnistuspalvelu toimivat saumattomasti yhdessä käyttäjän vahvistuksessa.
<b>Testiympäristö ja järjestelmien väliset integraatiot</b>	Tässä testitapauksessa testataan mobiilisovelluksen, taustajärjestelmän, ulkoisen tunnistuspalvelun ja tietokannan integraatiota rekisteröinti- ja kirjautumisprosessissa.
<b>Testausmenetelmä</b>	<ol style="list-style-type: none"> <li>1. Käyttäjä valitsee tunnistautumistavan rekisteröityessä</li> <li>2. Käyttäjä tunnistautuu ulkoisen palvelun kautta</li> <li>3. Käyttäjä kirjautuu sisään tunnistautumistavalla</li> <li>4. Käyttäjä pääsee omaan profiiliinsa</li> </ol>
<b>Hyväksymiskriteerit</b>	<p>Hyväksymiskriteerit:</p> <p>Kaikki alatasen testit tehty hyväksytysti:</p> <ul style="list-style-type: none"> <li>- Käyttäjä voi rekisteröityä onnistuneesti mobiilivarmenteella tai pankkitunnistautumisella.</li> <li>- Ulkoinen tunnistuspalvelu palauttaa käyttäjän tiedot oikein taustajärjestelmälle.</li> <li>- Käyttäjä voi kirjautua sisään valitsemallaan tunnistautumistavalla ilman virheitä.</li> <li>- Taustajärjestelmä palauttaa käyttäjän tiedot ja kirjautumistokenin oikein ja käyttäjä päätyy omaan profiiliin.</li> <li>- Käyttäjän profiilitiedot näkyvät sovelluksessa ilman virheitä ja ne on vahvistettu</li> </ul>

## 7.1.4 Integraatiotesti 4

.	Testitapaus 4
<b>Tunnus ID</b>	ITC8
<b>Nimi</b>	Application Version Management
<b>Testitapauksen kuvaus</b>	Testataan, että järjestelmä rekisteröi sovelluksen uusimmat versiot hallintapaneelissa. Tämä integraatiotesti varmistaa, että mobiilisovellus, taustajärjestelmä ja versionhallintajärjestelmä toimivat saumattomasti yhdessä.
<b>Testiympäristö ja järjestelmien väliset integraatiot</b>	Tässä testitapauksessa testataan mobiilisovelluksen, taustajärjestelmän ja versionhallintajärjestelmän integraatiota sovelluksen päivitysten seurannassa
<b>Testausmenetelmä</b>	<ol style="list-style-type: none"> <li>1. Sovellus lähettää versionumeronsa taustajärjestelmälle ensimmäisen käynnistyksen yhteydessä.</li> <li>2. Taustajärjestelmä tarkistaa version versionhallintajärjestelmästä.</li> <li>3. Jos sovellusversio on ajan tasalla, käyttäjä voi jatkaa sovelluksen käyttöä normaalisti.</li> <li>4. Jos sovellusversio on vanhentunut, käyttäjälle näytetään ilmoitus sovelluksen päivittämisestä.</li> <li>5. Tarkistetaan, että sovelluksen versiotieto tallentuu järjestelmän hallintapaneeliin.</li> </ol>
<b>Hyväksymiskriteerit</b>	<p>Hyväksymiskriteerit:</p> <p>Kaikki alatason testit tehty hyväksytysti:</p> <ul style="list-style-type: none"> <li>- Taustajärjestelmä rekisteröi käyttäjän sovellusversion onnistuneesti.</li> <li>- Versionhallintajärjestelmä tunnistaa, onko käyttäjän sovellus ajan tasalla.</li> <li>- Käyttäjä saa ilmoituksen, jos sovellusversio on vanhentunut.</li> <li>- Hallintapaneelissa näkyvät kaikki käytössä olevat sovellusversiot ja niiden käyttäjämäärät</li> </ul>

## 7.1.5 Integraatiotesti 5

.	Testitapaus 5
<b>Tunnus ID</b>	ITC5
<b>Nimi</b>	Data Security
<b>Testitapauksen kuvaus</b>	Testataan, että käyttäjän syöttämät tiedot tallennetaan ja käsitellään turvallisesti. Tämä integraatiotesti varmistaa, että mobiilisovellus, taustajärjestelmä ja tietokanta toimivat saumattomasti yhdessä ja että tiedot pysyvät salattuina ja suojattuina koko prosessin ajan.
<b>Testiympäristö ja järjestelmien väliset integraatiot</b>	Tässä testitapauksessa testataan mobiilisovelluksen, taustajärjestelmän ja tietokannan integraatiota turvallisen datan käsittelyn osalta.
<b>Testausmenetelmä</b>	Käyttäjä syöttää tietoja mobiilisovelluksessa. Sovellus lähettää tiedot taustajärjestelmään salattuna (SSL/TLS). 1. Taustajärjestelmä käsittelee datan ja salaa sen ennen tallennusta. 2. Tiedot tallennetaan salattuna tietokantaan. 3. Käyttäjä pyytää tietojensa katselua sovelluksessa. 4. Taustajärjestelmä hakee ja purkaa salatut tiedot ja lähettää ne sovellukseen. 5. Käyttäjä voi tarkistaa, että tiedot vastaavat alkuperäisiä syötettyjä tietoja.
<b>Hyväksymiskriteerit</b>	Hyväksymiskriteerit:  Kaikki alatasen testit tehty hyväksytysti: - Kaikki käyttäjän syöttämät tiedot siirretään vain salattuna (SSL/TLS). - Taustajärjestelmä ei tallenna dataa selväkielisenä, vaan ainoastaan salatussa muodossa. - Käyttäjä voi hakea omia tietojaan, ja ne palautuvat oikein ja salauksen purku toimii. - Tietokantaan tallennettu data ei ole luettavissa ilman salausta.