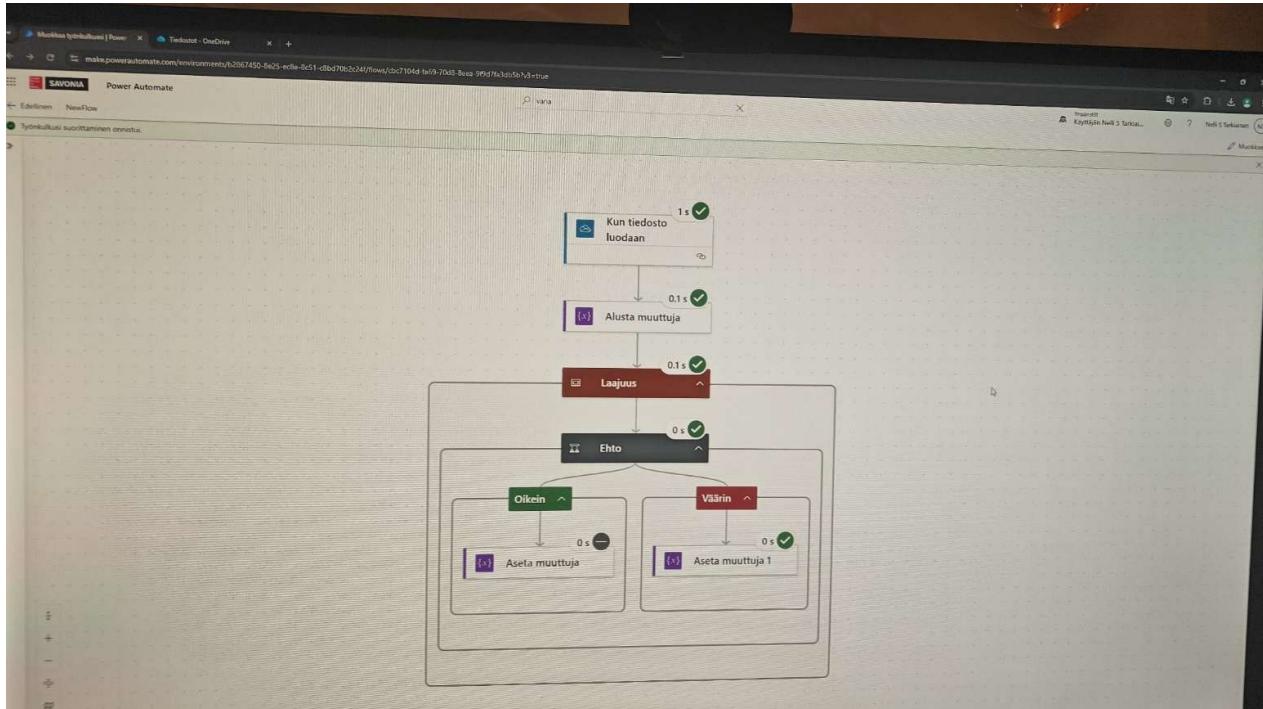


Nelli Tarkiainen

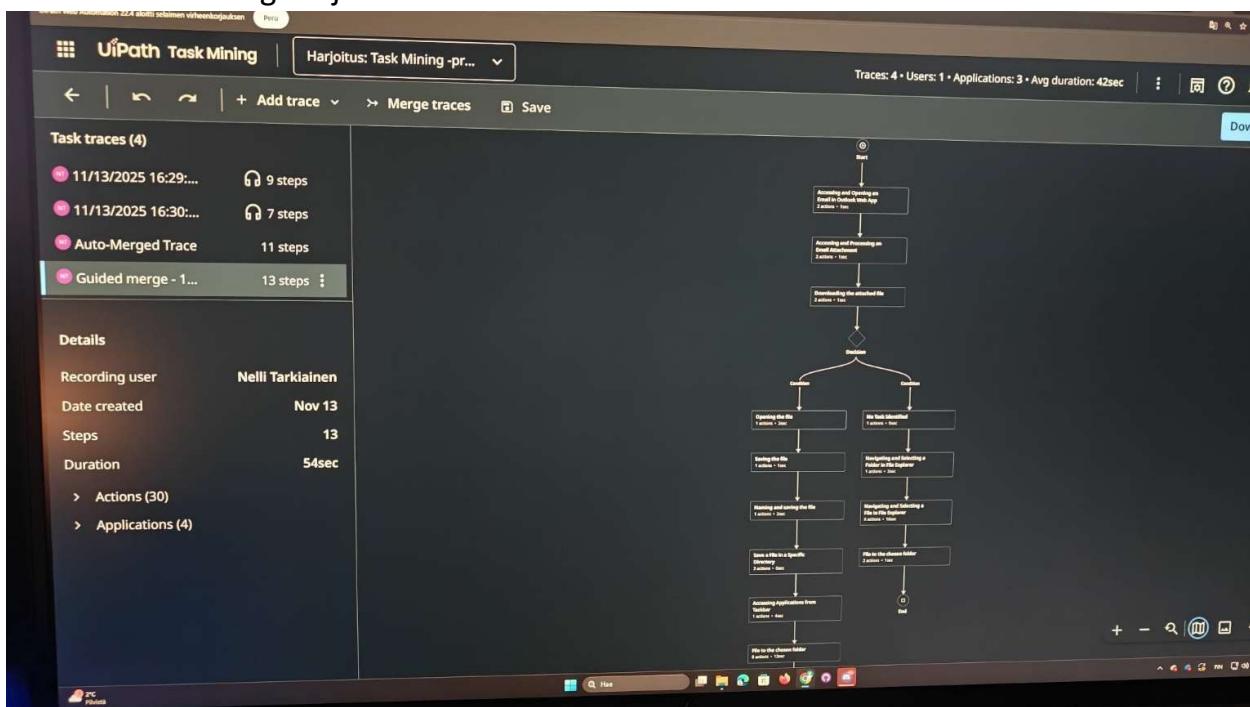
Huom. Tässä vain muutamia tuotoksia mitä olen tehnyt vuonna 2024-2025.

Prosessilouhintaa, ohjelmistorobotti ja tekoäly kurssilla tehtyjä joitakin harjoituksia

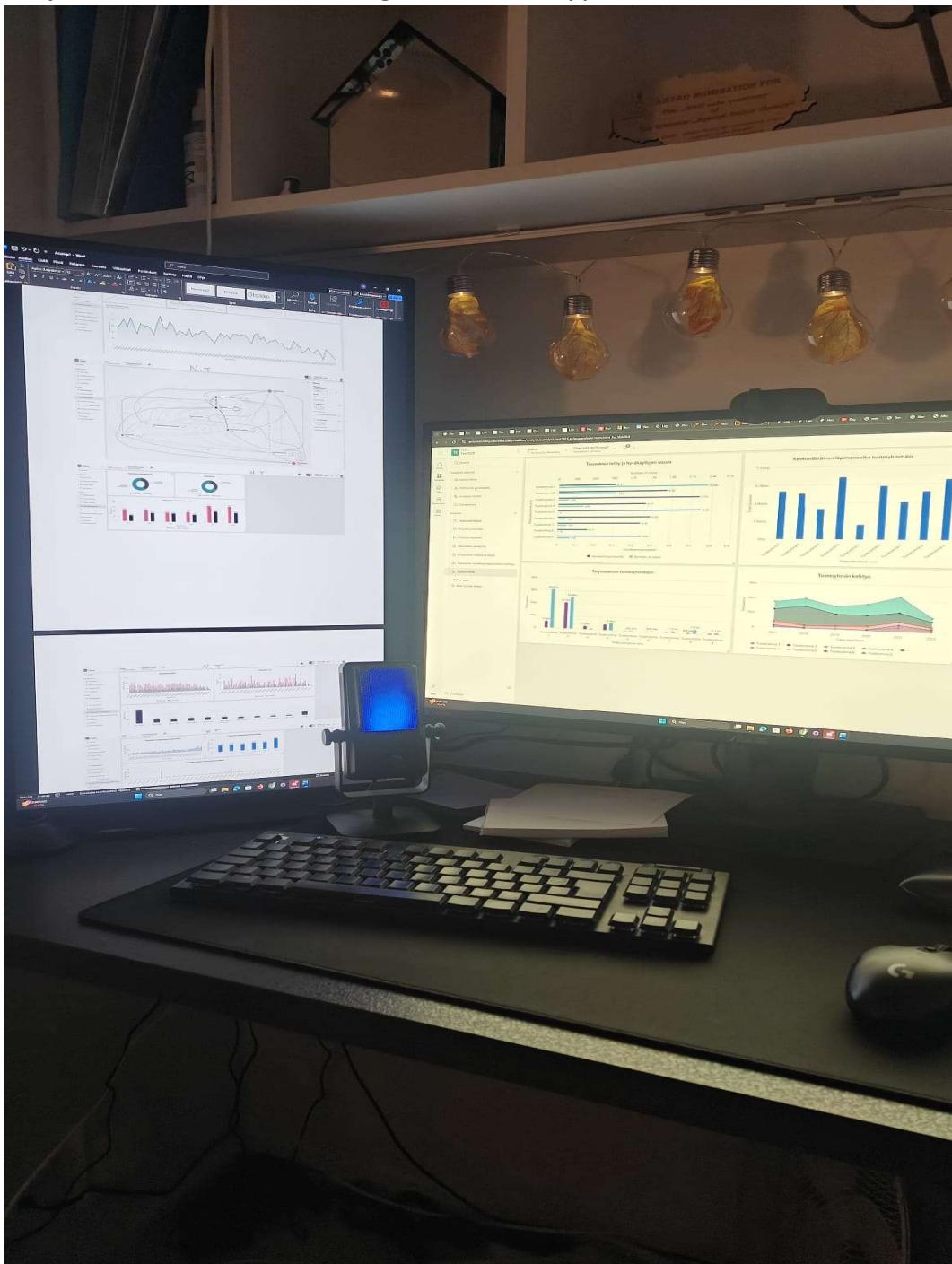
Microsoft Power Automate harjoitus: Luokittelu ilman AI Builderia



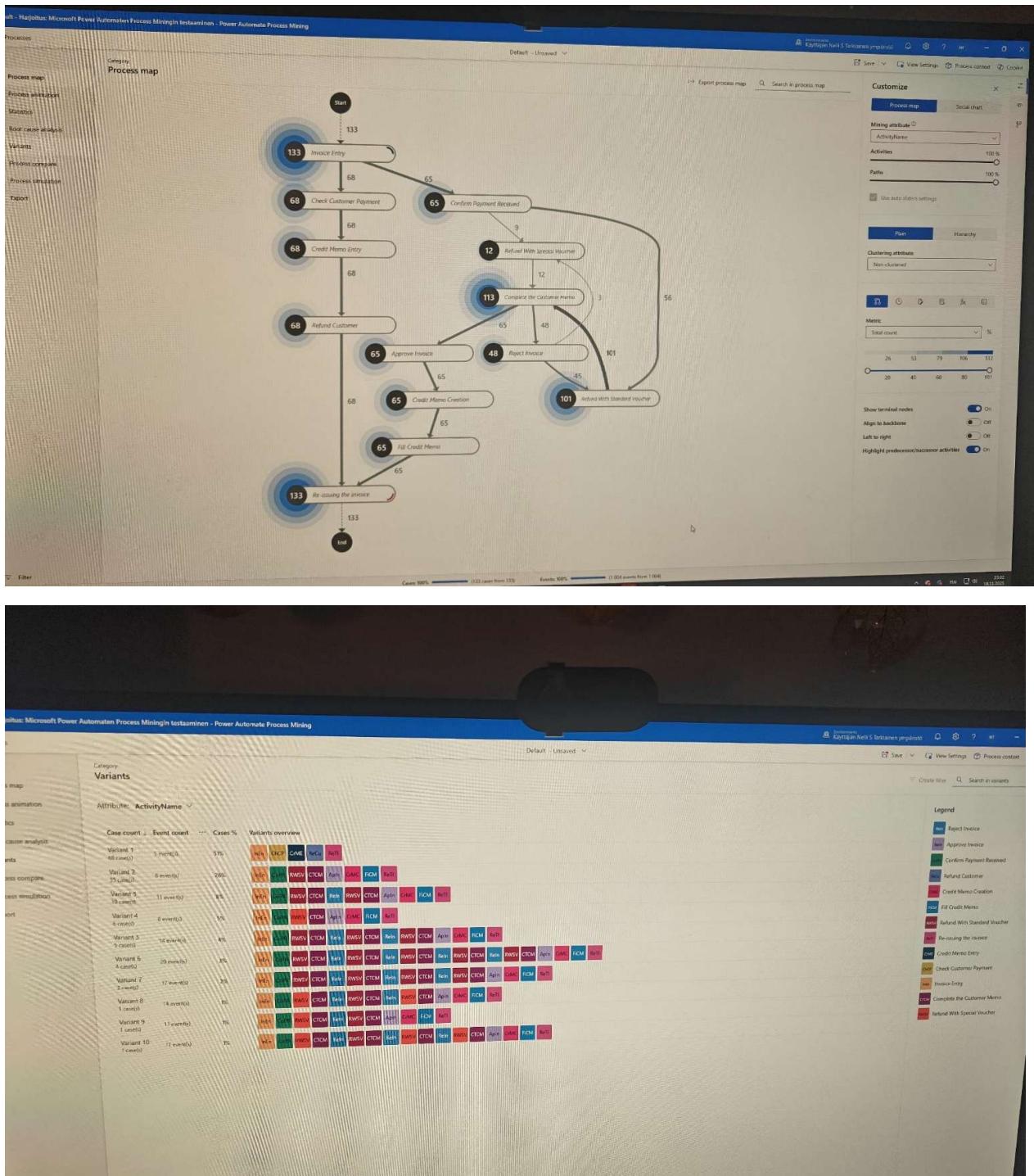
UiPath Task Mining harjoitus



Harjoitus - Aris Process Mining: Prosessianalyysin luominen



Harjoitus: Microsoft Power Automaten Process Mining testaaminen



Tiedonhallinta ja SQL harjoitustyö

Vaihe 1.

Harjoitustyön aihe: Verkkokauppatilausen hallintajärjestelmä

Tavoitteeni on mahdollisesti kehittää verkkokauppatilausen hallintajärjestelmä, joka mahdollistaa hallinnan asiakkaiden, tilausten, maksujen ja tuotteiden osalta.

Tietokantataulut ovat/voisivat olla

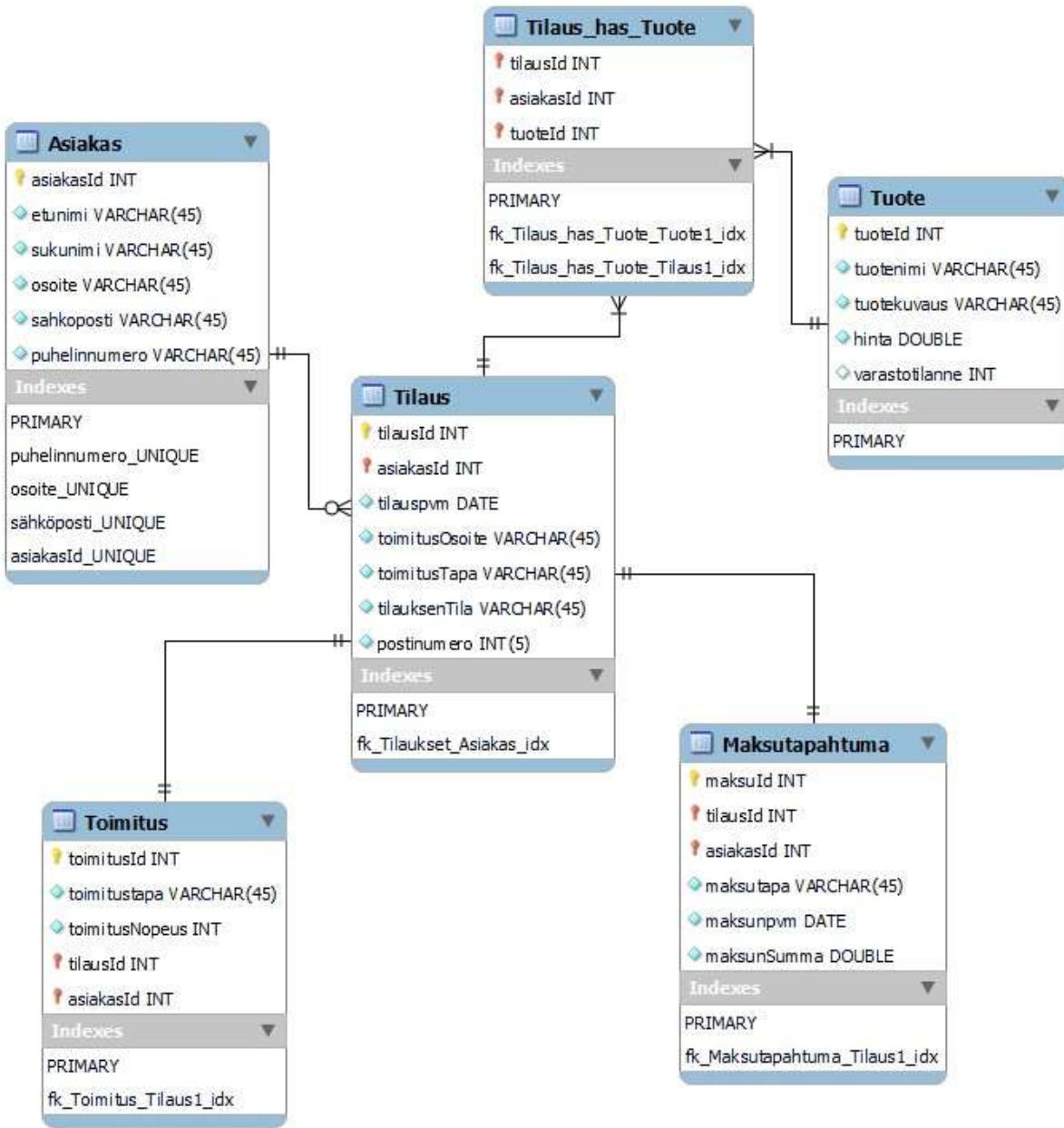
1. Asiakas-taulu
2. Tuote-taulu
3. Tilaus-taulu
4. Maksutapahtuma-taulu
5. Toimitustapa-taulu

Palautettu: torstai 25. tammikuu 2024

Tietokantaan tallennetaan asiakas, tuotteet ja tilaukset.

Vaihe 2

ER-kaavio



Taulukoiden väliset yhteydet:

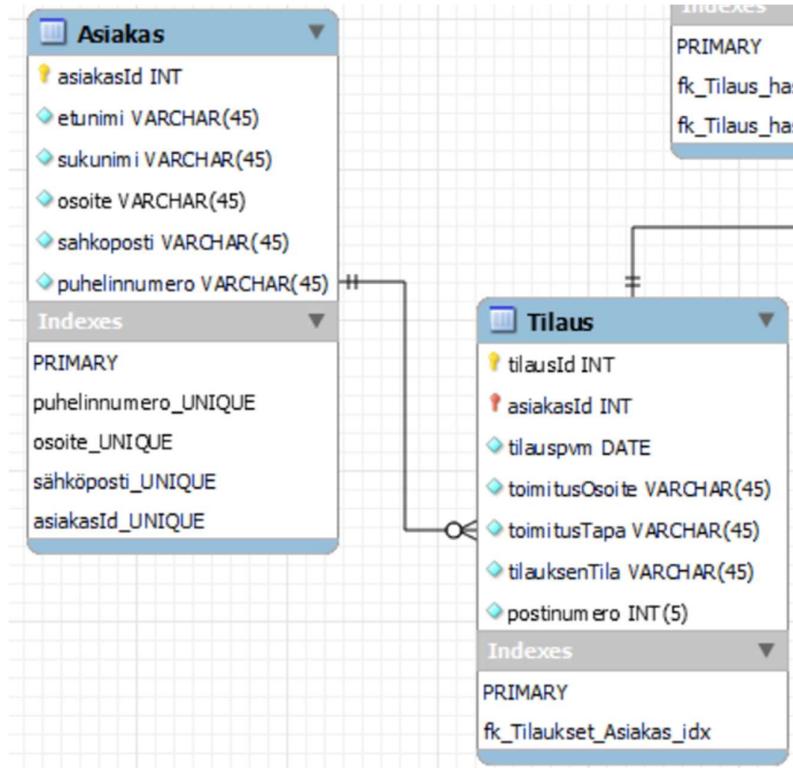
Asiakkaalla voi olla monta tilausta, mutta tilauksella ei voi olla montaa asiakasta.

Tilauksella voi olla monta tuotetta ja tuotteella voi olla monta tilausta.

Tilauksella voi olla yksi maksutapahtuma ja maksutapahtumalla voi olla vain yksi tilaus. Sama tilauksen ja toimituksen yhteydellä.

Asiakas-taulussa puhelinnumeron, osoitteen, sähköpostin ja asiakasId:n on oltava uniikit eli sarakkeessa ei saa olla samaa sisältöä jollakin toisella rivillä tämän taulun sisällä.

Keltaisella näkyvät pääavaimet ja punaisella vierasavaimet.

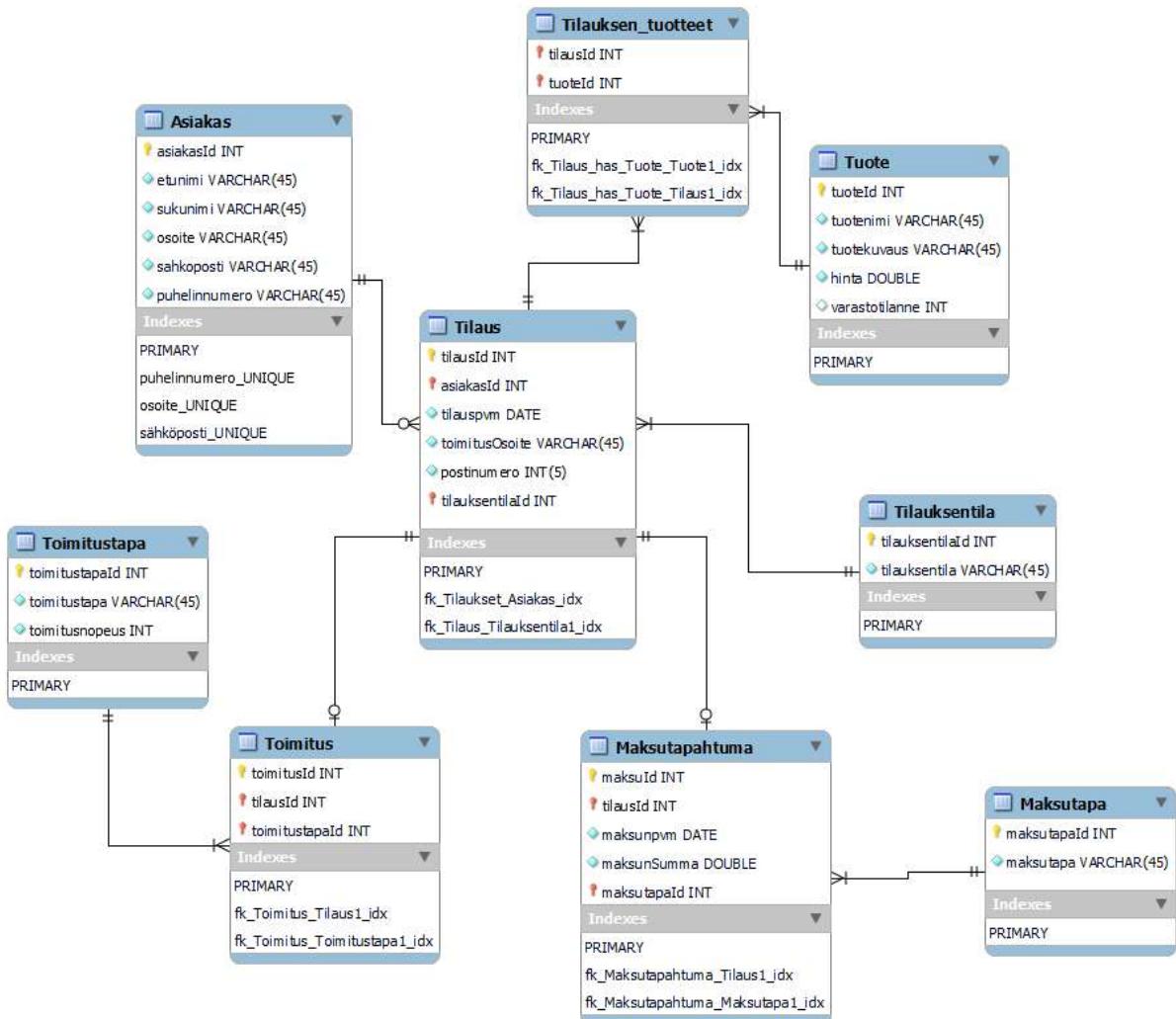


Asiakkaalla ei ole pakko olla tilausta (tyhjä pikku pallura).

Palautettu: sunnuntai 18. helmikuu 2024

MUUTOKSET, PÄIVITETTY ER-KAAVIO 23.2.2024

- Normalisoitu maksu ja toimitus
- Toimitustapoja toimituksella voi olla monta erilaista sekä maksutapoja voi olla maksutapahtumalla monta (harakanvarpaat) erilaista, mutta toimituksia ja maksutapahtumia tilauksella voi olla vain yksi.



Viite-eheyssääntöjä:

- ON DELETE NO ACTION / ON UPDATE NO ACTION: Kun ensisijaisen avaimen arvoa (asiakasId) muutetaan tai poistetaan, siihen liittyviä rivejä ei muuteta tai poisteta.
- ON DELETE CASCADE / ON UPDATE CASCADE: Kun ensisijaisen avaimen arvoa (tilausId) muutetaan tai poistetaan, siihen liittyvät rivit (esim. maksutapahtumat) poistetaan tai päivitetään vastaavasti.
- ON DELETE SET NULL / ON UPDATE SET NULL: Kun ensisijaisen avaimen arvoa (tilausId) muutetaan tai poistetaan, siihen liittyvät ulkopuoliset avaimet (esim. maksutapahtumat) asetetaan NULL-arvoiksi.
- ON DELETE RESTRICT / ON UPDATE RESTRICT: Estää muutoksen tai poiston toimenpiteen, jos on olemassa liittyviä rivejä toisessa taulussa.

Asiakkaan ja tilauksen välinen suhde:

Tilaus-taululla on asiakasId-sarake, joka viittaa Asiakas-taulun asiakasId-sarakkeeseen. Tämä määrittelee suhteen asiakkaan ja hänen tilaustensa välillä.

Tilaus ja tilauksen tuotteiden välinen suhde:

Tilaus_tuotteet-taululla on sekä tilausId- että tuotId-sarakkeet, jotka viittaavat vastaavasti Tilaus- ja Tuote-tauluihin. Tämä määrittelee suhteen tilauksen ja siihen liittyvien tilauksen tuotteiden välillä.

Tilaus ja maksutapahtuman välinen suhde:

Maksutapahtuma-taululla on tilausId-sarake, joka viittaa Tilaus-taulun tilausId-sarakeeseen. Tämä määrittelee suhteen tilauksen ja siihen liittyvien maksutapahtumien välillä.

Vaihe 3

TIETOKANTA

CRUD-toiminnot Asiakas-taululle:

C – CREATE

- luodaan uusi asiakas

```
2 •    INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero)
3      VALUES ('Marjatta', 'Mallikas', 'Mannila 123', 'marjatta@mallikas.com', '123456789');
```

utput			
Action Output			Message
#	Time	Action	
1	18:05:48	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VAL...	1 row(s) affected

R – READ

- näyttää kaikki asiakkaat

4 • SELECT * FROM Asiakas;

The screenshot shows the MySQL Workbench interface. At the top, there is a toolbar with various icons for editing, filtering, and exporting data. Below the toolbar is a "Result Grid" table with the following data:

asiakasId	etunimi	sukunimi	osoite	sahkoposti	puhelinnumero
1	Nelli	Tarkiainen	Juu23	teletappi@gmail.com	0403939274
2	Juuso	Rinkeli	Hermannia44	vaahtokarkki@hotmail.com	+35840193872
3	Herkko	Hermann	Herkkola11	herksutin@gmail.com	0402940192
4	Noora	Pietikäinen	Puijo11	norksutin@gmail.com	04002910392
5	Sebastian	Tiilikka	Yuhkiotie10	sebulius@gmail.com	0402937462
6	Marjatta	Mallikas	Mannila 123	marjatta@mallikas.com	123456789
*	NULL	NULL	NULL	NULL	NULL

Below the table, the "Output" section shows the query history:

#	Time	Action	Message
1	18:05:48	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1 row(s) affected
2	18:07:02	SELECT * FROM Asiakas	6 row(s) returned

U – UPDATE

- Päivittää asiakkaan (jonka asiakasId on 1 eli minun) osoitteen ja sähköpostin tiedot

The screenshot shows the MySQL Workbench interface. The "Output" section shows the following query:

```
5 • UPDATE Asiakas
6     SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com'
7     WHERE asiakasId = 1;
```

Below the query, the "Output" section shows the query history:

#	Time	Action	Message
1	18:05:48	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1 row(s) affected
2	18:07:02	SELECT * FROM Asiakas	6 row(s) returned
3	18:10:09	UPDATE Asiakas SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com...' 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0

Muutos/Päivitys

The screenshot shows the MySQL Workbench interface. The "Output" section shows the following query:

```
5 • UPDATE Asiakas
6     SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com'
7     WHERE asiakasId = 1;
```

Below the query, the "Output" section shows the query history:

#	Time	Action	Message
1	18:05:48	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1 row(s) affected
2	18:07:02	SELECT * FROM Asiakas	6 row(s) returned
3	18:10:09	UPDATE Asiakas SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com...' 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
4	18:10:56	SELECT * FROM Asiakas	6 row(s) returned

D – DELETE

- Poistaa asiakkaan (jonka asiakasId on 6 eli Marjatta Mallikkaan, jonka loin alussa)

8 • `DELETE FROM Asiakas WHERE asiakasId = 6;`

Output

Action Output	#	Time	Action	Message
1 18:05:48 INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1	18:05:48	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1 row(s) affected
2 18:07:02 SELECT * FROM Asiakas	2	18:07:02	SELECT * FROM Asiakas	6 row(s) returned
3 18:10:09 UPDATE Asiakas SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com...' ...	3	18:10:09	UPDATE Asiakas SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com...' ...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
4 18:10:56 SELECT * FROM Asiakas	4	18:10:56	SELECT * FROM Asiakas	6 row(s) returned
5 18:14:02 DELETE FROM Asiakas WHERE asiakasId = 6	5	18:14:02	DELETE FROM Asiakas WHERE asiakasId = 6	1 row(s) affected

Result Grid

asiakasId	etunimi	sukunimi	osoite	sahkoposti	puhelinnumero
1	Nelli	Tarkainen	Vaaltoarkki 33	onni@orava.com	0403939274
2	Juuso	Rinkeli	Hermannia44	vaaltoarkki@hotmail.com	+35840193872
3	Herkko	Hermann	Herkkola11	herksutin@gmail.com	0402940192
4	Noora	Pietikäinen	Puijo11	norksutin@gmail.com	04002910392
5	Sebastian	Tiilikka	Yuhkiotie10	sebulius@gmail.com	0402937462
*	HULL	HULL	HULL	HULL	HULL

Asiakas 3 ×

Output

Action Output	#	Time	Action	Message
1 18:05:48 INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1	18:05:48	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1 row(s) affected
2 18:07:02 SELECT * FROM Asiakas	2	18:07:02	SELECT * FROM Asiakas	6 row(s) returned
3 18:10:09 UPDATE Asiakas SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com...' ...	3	18:10:09	UPDATE Asiakas SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com...' ...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
4 18:10:56 SELECT * FROM Asiakas	4	18:10:56	SELECT * FROM Asiakas	6 row(s) returned
5 18:14:02 DELETE FROM Asiakas WHERE asiakasId = 6	5	18:14:02	DELETE FROM Asiakas WHERE asiakasId = 6	1 row(s) affected
6 18:15:32 SELECT * FROM Asiakas	6	18:15:32	SELECT * FROM Asiakas	5 row(s) returned

Jos asiakkaalla on liittyviä tilauksia Tilaus-taulussa, alla oleva SQL-lauseke ei suoriudu ja tietokanta palauttaa virheen. Tämä tapahtuu, koska viittausrajoitus estää asiakkaan poiston, joka on kytketty tilauksiin.

9 • `DELETE FROM Asiakas WHERE asiakasId = 1;`

Output

Action Output	#	Time	Action	Message	Duration /
1 18:05:48 INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1	18:05:48	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1 row(s) affected	0.000 sec
2 18:07:02 SELECT * FROM Asiakas	2	18:07:02	SELECT * FROM Asiakas	6 row(s) returned	0.000 sec
3 18:10:09 UPDATE Asiakas SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com...' ...	3	18:10:09	UPDATE Asiakas SET osoite = 'Vaaltoarkki 33', sahkoposti = 'onni@orava.com...' ...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
4 18:10:56 SELECT * FROM Asiakas	4	18:10:56	SELECT * FROM Asiakas	6 row(s) returned	0.000 sec
5 18:14:02 DELETE FROM Asiakas WHERE asiakasId = 6	5	18:14:02	DELETE FROM Asiakas WHERE asiakasId = 6	1 row(s) affected	0.000 sec
6 18:15:32 SELECT * FROM Asiakas	6	18:15:32	SELECT * FROM Asiakas	5 row(s) returned	0.000 sec
7 18:18:09 INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	7	18:18:09	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1 row(s) affected	0.000 sec
8 18:18:12 SELECT * FROM Asiakas	8	18:18:12	SELECT * FROM Asiakas	6 row(s) returned	0.000 sec
✖ 9 18:27:11 DELETE FROM Asiakas WHERE asiakasId = 1	9	18:27:11	DELETE FROM Asiakas WHERE asiakasId = 1	Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails	0.016 sec

Syntyi virhe: DELETE FROM Asiakas WHERE asiakasId = 1

Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails
 ('harjoitustyö`.`tilaus`, CONSTRAINT `fk_Tilaus_Asiakas` FOREIGN KEY (`asiakasId`) REFERENCES `asiakas`(`asiakasId`) ON DELETE NO ACTION ON UPDATE NO ACTION)

Eli virhe syntyi sen takia, koska asiakkaalla, jonka asiakasId on 1 (minä) on jo olemassa tilaus (alhaalla oleva kuva). Asiakas, jonka asiakasId oli 6, voitiin poistaa (Marjatta Mallikas), koska hänenne ei ollut tilausta tehtynä.

```
10 •  SELECT * FROM Tilaus;
11
```

Tilaus 5 x

Output

#	Time	Action	Message
3	18:10:09	UPDATE Asiakas SET osoite = 'Vaanaja 33', sahkoposti = 'onni@orava.co...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
4	18:10:56	SELECT * FROM Asiakas	6 row(s) returned
5	18:14:02	DELETE FROM Asiakas WHERE asiakasId = 6	1 row(s) affected
6	18:15:32	SELECT * FROM Asiakas	5 row(s) returned
7	18:18:09	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) V...	1 row(s) affected
8	18:18:12	SELECT * FROM Asiakas	6 row(s) returned
9	18:27:11	DELETE FROM Asiakas WHERE asiakasId = 1	Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails
10	18:35:12	SELECT * FROM Tilaus	3 row(s) returned

Toinen esimerkki, jossa viittausrajoitus estää tilauksen lisäämisen ilman olemassa olevaa asiakasta:

Tilaus-tauluun liittyy viittaus Asiakas-tauluun asiakasId-sarakkeen avulla. Tällöin yritettäessä lisätä tilausta ilman olemassa olevaa asiakasta, tietokanta estää lisäyksen viittausrajoitteen vuoksi.

```
12 •  INSERT INTO Tilaus (asiakasId, tilauspvm, toimitusOsoite, postinumero, tilauksentilaId)
13     VALUES (10, '2024-03-03', 'Pisara 22', '70100', 1);
14
```

Output

#	Time	Action	Message
1	18:05:48	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1 row(s) affected
2	18:07:02	SELECT * FROM Asiakas	6 row(s) returned
3	18:10:09	UPDATE Asiakas SET osoite = 'Vaanaja 33', sahkoposti = 'onni@orava.com...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
4	18:10:56	SELECT * FROM Asiakas	6 row(s) returned
5	18:14:02	DELETE FROM Asiakas WHERE asiakasId = 6	1 row(s) affected
6	18:15:32	SELECT * FROM Asiakas	5 row(s) returned
7	18:18:09	INSERT INTO Asiakas (etunimi, sukunimi, osoite, sahkoposti, puhelinnumero) VA...	1 row(s) affected
8	18:18:12	SELECT * FROM Asiakas	6 row(s) returned
9	18:27:11	DELETE FROM Asiakas WHERE asiakasId = 1	Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails
10	18:35:12	SELECT * FROM Tilaus	3 row(s) returned
11	18:44:51	INSERT INTO Tilaus (asiakasId, tilauspvm, toimitusOsoite, postinumero, tilauksentilaId)	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ...

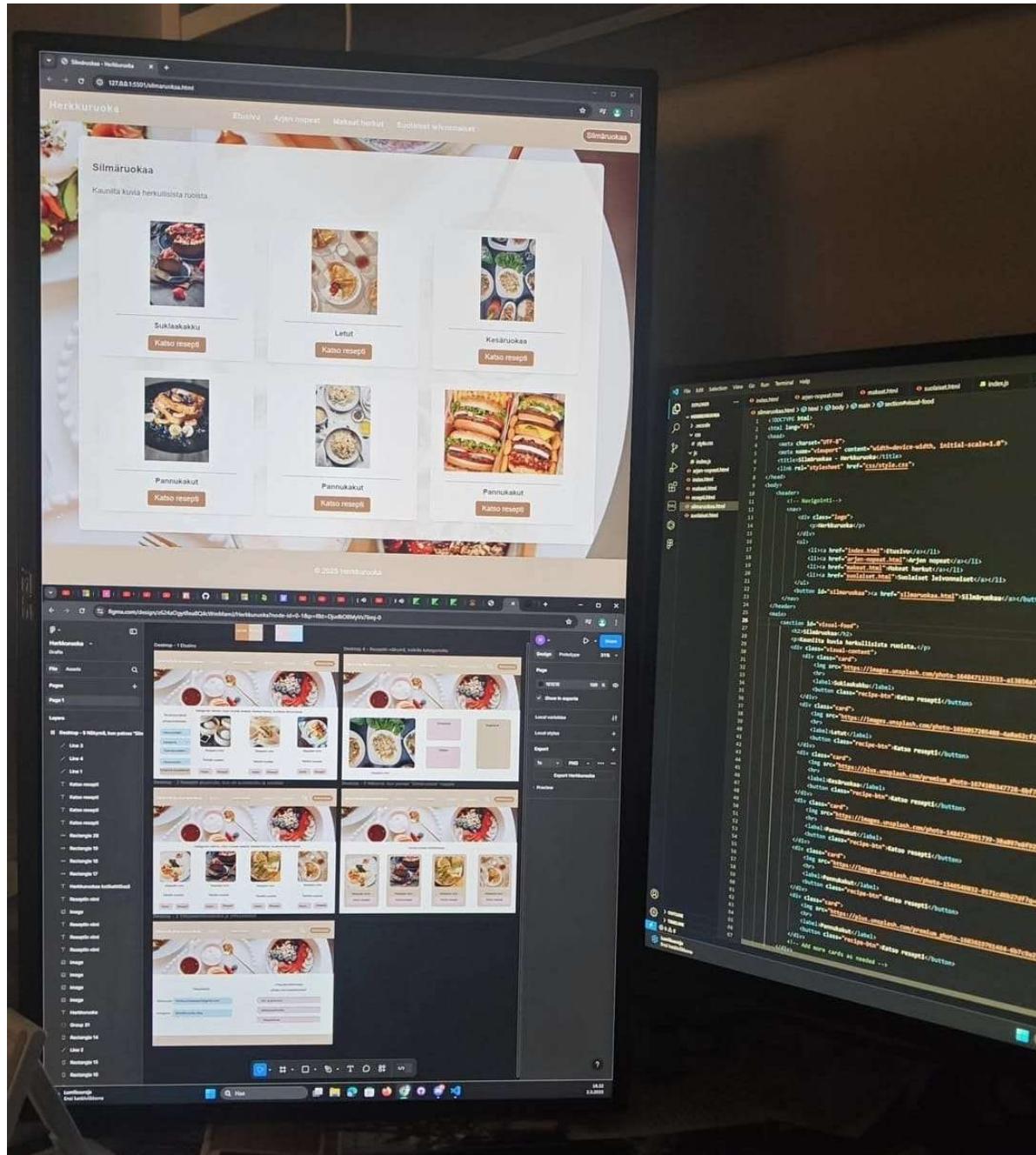
Virhe: INSERT INTO Tilaus (asiakasId, tilauspvm, toimitusOsoite, postinumero, tilauksentilaId)
VALUES (10, '2024-03-03', 'Pisara 22', '70100', 1)

Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails
(`harjoitustyö`.`tilaus`, CONSTRAINT `fk_Tilaukset_Asiakas` FOREIGN KEY (`asiakasId`) REFERENCES `asiakas` (`asiakasId`) ON DELETE NO ACTION ON UPDATE NO ACTION)

eli tietokanta palauttaa virheen, koska asiakasId 10 ei ole olemassa Asiakas-taulussa.

Palautettu 3.3.2024

Herkkuoka-reseptisivu:



Linkki Figmaan:

<https://www.figma.com/design/z624aOgyt8eaBQ4cWmMamJ/Herkkuruoka?node-id=0-1&t=rxgbkYzx7ENISTbv-1>

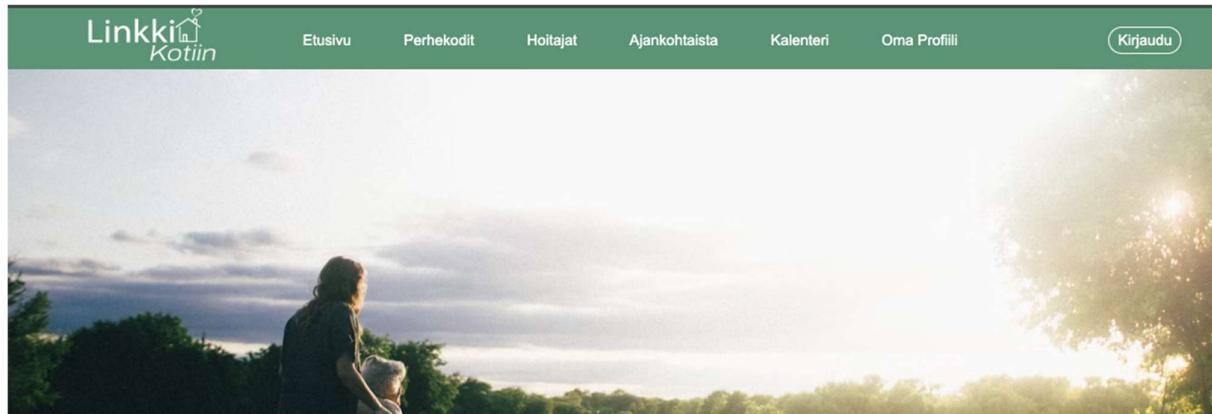
Voin näyttää koodin pyydettäessä livenä 😊.

LinkkiKotiin

Asiakasohjauksen parantaminen ikääntyneiden perhehoidossa

Tämän projektin tarkoituksena oli luoda demosovellus perhehoitokotien päivittäisen toiminnan yksinkertaistamiseksi. LinkkiKotiin-sovellus helpottaa perhehoitokotien ja hoitajien varaamista. Hoitajien synkronoidun kalenterin avulla reaalialkaiset saatavuustiedot jaetaan hyvinvointialueen työntekijöiden kanssa. Lisäksi alusta sisältää käyttäjien rekisteröinti- ja kirjautumistoiminnot, ja siinä on erilliset sivut eri käyttäjärooleille, kuten hoitokotien ylläpitäjille, hoitajille ja asiakkaille. Se tarjoaa myös ammattitason suodatusvaihtoehtoja perhehoitokoteille ja hoitajille, mikä varmistaa, että yksilöt löytävät sopivimmat hoitoratkaisut erityistarpeidensa perusteella.

Tämä projekti oli osa Ohjelmistotekniikka 2 -kurssia ja se kehitettiin yhteistyössä neljän muun projektitiimin jäsenen kanssa. Logon loi projektitiimin jäsen, joka oli muotoilija. Itse olin projektissa Scrum Masterina.



Perhekodit

Tällä sivulla voit vapaasti selata perhekoteja tai hakea niitä erilaisin kriteerein ja suodattimin.

Haku

Suodattimet

sii

Hae

Perhekodin k

Perhekoti Sili

Esteetön:

Sillinjärvi

 Kyllä Ei Ei välttä Sillinjärvi pp . kk . vvvv

Eläimlä:

 Kyllä Ei Ei välttä Varkaus Kiuruvesi

Lapsia:

 Kyllä Ei Ei välttä Joroinen Joensuu

Nollaa päävärinäätöt

Nollaa suodattimet



Perhekoti Tuomiranta

Kuopio
Paikkoja vapaana

Lue lisää



Perhekoti Sili

Sillinjärvi
Paikkoja vapaana

Lue lisää



Perh...

Ei v...

Li...



Hoitajat

Tällä sivulla voit vapaasti selata hoitajia tai hakea niitä erilaisin kriteerein ja suodattimin

Haku

Suodattimet

Hae

Sukupuoli:	
<input type="checkbox"/> Nainen	<input type="checkbox"/> Suomi
<input type="checkbox"/> Mies	<input type="checkbox"/> Englanti
	<input type="checkbox"/> Ruotsi
	<input type="checkbox"/> Ranska
	<input type="checkbox"/> Saksa

Aikupvm:
pp.kk.vvvv
Nollaa suodattimet

Loppupvm:
pp.kk.vvvv
Nollaa päivämäärät



Anna Annanen

Ikä: 31
Kielet: Suomi, Englanti
[Vapalta aikoa](#)

[Lue lisää](#)



Mikko Mehiläinen

Ikä: 45
Kielet: Suomi, Ruotsi, Englanti
[Vapalta aikoa](#)

[Lue lisää](#)

Tiedotteet

Uusi päivitys saatavilla

Linkkikotiin-sovellukseen on julkaistu uusi päivitys, joka sisältää parannuksia käytettävyteen.

Perhekotien haku laajentunut

Perhekotien hakutoiminto on nyt saatavilla useammille alueille. Käy tutustumassa!

Hoitajien profiilit päivitetty

Hoitajien profilisivut on päivitetty vastaamaan käyttäjien toiveita.

[Lisää uusi](#)

Dokumentit

[Ohjeet](#)

[Sopimus](#)

[Ehdot](#)

[Lisää uusi](#)

Chat

Hei! Kuinka voin auttaa sinua tänään?

heyhey!

Kiitos viestistäsi! Palaamme asiaan pian.

Kirjoita viesti...

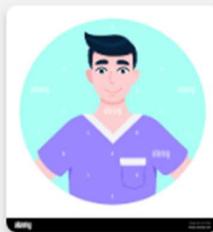
Lähetä

Nimi:

Sukupuoli:

Ikä:

Kuvaus:



Poista kuva

Poista profili

**Palvelukielet
(pilkuilla erotettuna):**

**Erityisosaaminen
(pilkuilla erotettuna):**

**Email
(muokkaus ylläpidon kautta):**

Puhelinnumero:

Kiertävä hoitaja:

Profiilikuvan URL:

Tallenna

Peruuta

Lisää perhekoti

Hae tapahtumia... Kuukausi Kaikki Lisää tapahtuma

Edellinen Seuraava

maaliskuu 2025

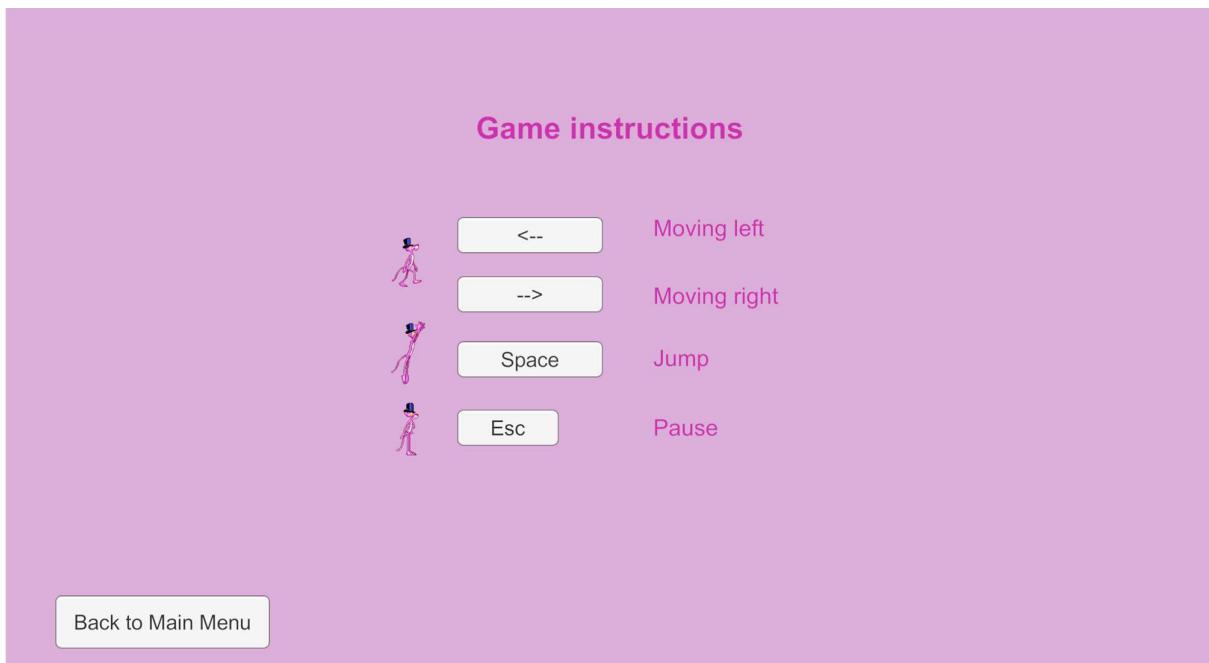
Maanantai	Tiistai	Keskiviikko	Torstai	Perjantai	Lauantai	Sunnuntai
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26 Palaveri	27 Katseimointi	28	29	30
31						



Voin näyttää koodin pyydettäessä livenä 😊 .

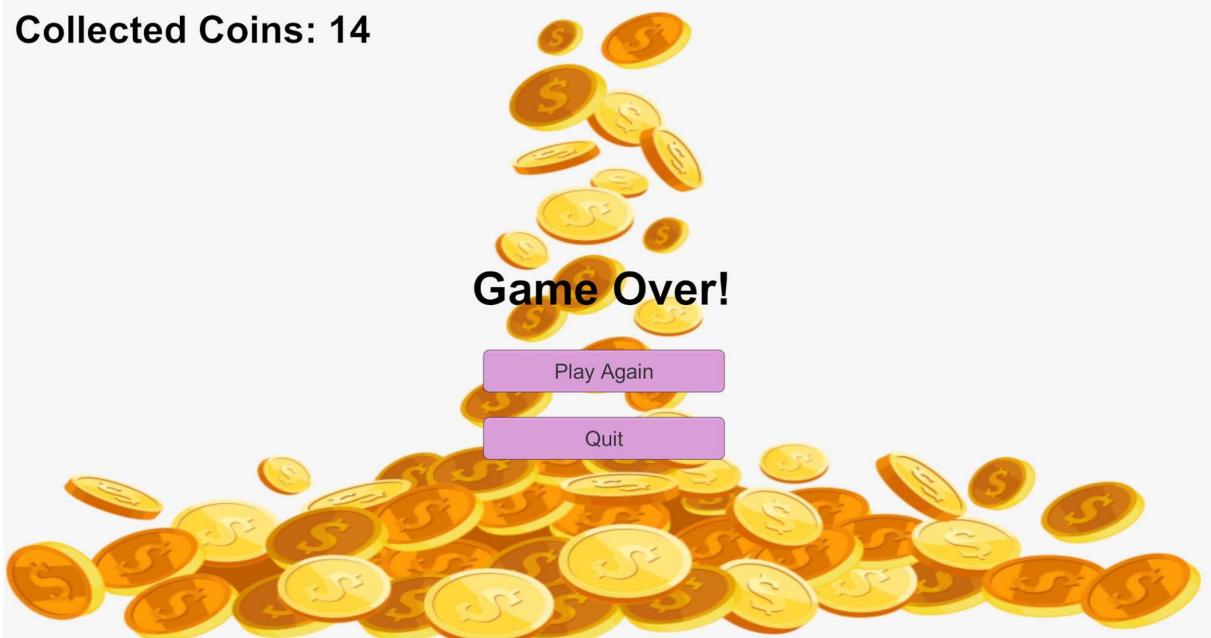
Pinkki Pantteri tasohyppelypeli







Collected Coins: 14



Pelin suunnitteludokumentti

Pelin nimi: Vaaleanpunainen pantteri kolikkotasopeli

Pelin genre: 2D Tasohyppelypeli

Kohdealusta: Windows, Mac, Linux

Pelimoottori: Unity

Ohjelmointikieli: C# (Visual Studio Codessa)

1. Pelin yleiskuvaus

Vaaleanpunainen pantteri kolikkotasopeli on nopeatempoinen 2D-tasohyppelypeli, jossa pelaaja ohjaa Vaaleanpunaista pantteria keräämään kolikoita rajatussa ajassa samalla välttellen putoamista ruudulta. Peli on suunniteltu yksinkertaisilla mekaniikoilla, jotka haastavat pelaajan tarkkuutta ja reaktiokykyä.

2. Pelimekaniikat

Pelaajan ohjaus:

- Liikkuminen vasemmalle/oikealle: Nuolinäppäimet ← ja →
- Hyppy: Välilyönti

Pelin säännöt:

- Pelaaja ohjaa Vaaleanpunaista pantteria ja kerää kolikoita kasvattaakseen pistemääränsä.
- Pelissä on jatkuvasti laskeva ajastin. Kun aika loppuu, peli päättyy.
- Jos Vaaleanpunainen pantteri putoaa ruudulta, peli päättyy välittömästi ja pelaaja näkee omat pisteesä.
- Pelialue sisältää alustoja, jotka samalla auttavat ja haastavat pelaajan liikkumista.

Voitto- ja häviöehdot:

- Peli päättyy, kun:
 - Ajastin saavuttaa nollan.
 - Pelaaja putoaa ruudulta.
 - Pelin päättyvä tulosnäytöllä esitetään pelaajan saavuttama pistemääriä.
-

3. Pelin rakenne ja scenet

Päävalikko scene:

- Näyttää pelin nimen/logon.
- Päävalikosta peli käynnistyy Play-napista

- Quit-nappi sulkee pelin.

Ohje scene:

- Sisältää ohjeet, miten pantteri voi liikkua + Esc pääsee menuun.

Peli scene:

- Varsinainen pelialue, jossa Vaaleanpunainen pantteri kerää kolikoita.
- Näytöllä näkyy ajastin ja kerättyjen kolikoiden määrä.
- Kolikkoon osuminen toistaa ääniefektin.
- Esc-näppäin vie pausesceneen.

Pause scene:

- Pelaaja voi valita jatkaako peliä – Peli jatkuu Continue-napista (pisteet, aika ja pantterin sijainti tallentuu = peli jatkuu siitä mihin peli jäi viimeksi).
- Pelaaja voi aloittaa pelin alusta (em. tiedot nollaantuu = peli alkaa alusta).
- Pelaaja voi sulkea pelin Quit-napista.

Game Over scene:

- Näyttää pelaajan saavuttaman pistemäärän.
- Play Again-napista peli käynnistyy uudelleen.
- Quit-nappi sulkee pelin.

4. Grafiikka ja äännet

Grafiikka:

- Netistä haettuja Sprite Sheetejä, joita on muokattu kuvaeditorissa.
- Animaatiot toteutettu Unityssä.

Äännet:

- Jokaisessa scenessä eri taustamusiikki.
- Kolikon kerääminen toistaa ääniefektin.

5. Kehitystyökalut

- **Pelimoottori:** Unity
- **Ohjelmointi:** C# (Visual Studio Code)

- **Grafiikan muokkaus:** Kuvankäsittelyohjelma mm. Paint
-

6. Mahdolliset tulevat parannukset

- Health bar Vaaleanpunaiselle pantterille ja mahdollisesti vihollisille.
 - Enemmän vihollisia tai esteitä.
 - Hirviötä, jotka tuottavat damagea jos niihin osuu + mahdollisesti voivat ampua joitain vihreitä mönjää.
 - Erikoiskykyjä Vaaleanpunaiselle pantterille.
 - Pistetaulukko parhaiden tulosten tallentamiseen.
-

7. Yhteenvedo

Vaaleanpunainen pantteri kolikkotasopeli on yksinkertainen, mutta viihdyttävä 2D-tasohypypeli, joka on suunniteltu nopeisiin pelisessioihin. Yksinkertaiset kontrollit, aikarajoitteet ja kolikkojen keräämiseen keskittyvä pelimekanika tekevät pelistä hauskan kokemuksen, jota voidaan laajentaa tulevaisuudessa uusilla sisällöillä.

3D-ankkapeli

Tämän pelin tavoitteena on kerätä koiranpentuja, jotka ovat piilotettuna ympäri karttaa. Samalla, kun pelaaja yrittää suorittaa tehtävänsä niin häntä jahtaa hirviö, joka osuessaan lopettaa pelin. Pelaajan tulee siis yrittää selviytyä samalla, kun hän yrittää pelastaa koiranpentuja. Huom. tällä hetkellä projektin näyttää valitettavasti tältä, koska uusien Unity päivitysten mukana on ilmentynyt erroreita projektiini ja minun pitäisi se uudestaan rebuildata, koska se ei pysty vanhan Unityn version takia renderöitymään sellaiseksi niin kuin se alun perin oli.

Paina mitä tahansa
näppäimistöä aloittaaksesi peli



