

1. Henkilötiedot

Y2 projekti: Rahan seuranta

Tiina Mikkola, 1024674

Kauppätieteiden kandidaatti, 2. vsk

24.2.2023

2. Yleiskuvaus ja vaikeustaso

Ohjelma, joka lukee tilitapahtumat tiedostosta (.xml, tai .csv), erottelee tulot ja menot toisistaan sekä tekee selkeän kuvaajan menojen jakautumisesta, esimerkiksi piirakka- tai pylväsdiagrammin.

Tilitapahtumatiedosto muokataan tarvittaessa .csv:hen (pdf → csv).

Toteutettavan ohjelman pitää automaattisesti osata laskea yhteen samaan kauppaan tehdyt ostokerrat. Käyttäjän pitää halutessaan pystyä myös ryhmittelemään (ja poistamaan ryhmittely) kauppvoja haluamansa nimikkeen alle, esimerkiksi ”ruokakaupat”, johon käyttäjä lisää haluamansa kaupat. Ohjelman ei tarvitse pystyä lisäämään kaikkia kauppvoja samanaikaisesti, vaan ohjelma voi olettaa käyttäjän lisäävän haluamansa kaupat yksitellen.

Käyttäjän pitää saada helposti sekä ryhmitelty että ryhmittelemätön kuvaaja rahan käytöstä. Halutessasi voit tehdä myös useamman luokittelun, esimerkiksi lasketaanko omien tilien väliset siirrot tai vuokranmaksun piirrettävän kuvaajan dataan mukaan.

Mikä työstä tekee yksilöllisen: kulujen lisäksi myös tulojen jakautumista kuvataan piirakkadiagrammilla.

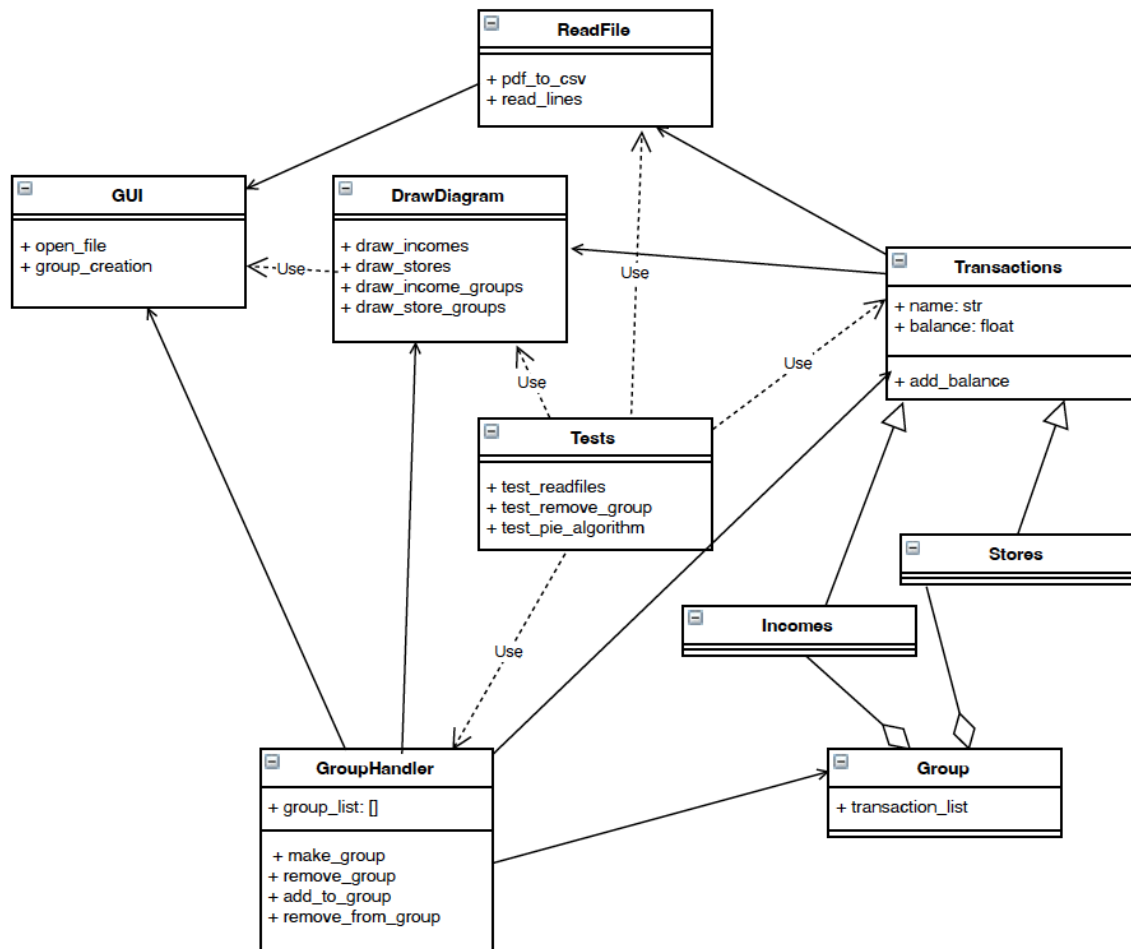
Suunniteltu vaikeustaso: keskivaikea

- käyttöliittymä ja rahankäytön visualisointi graafisena
- yksikkötestit edes jollekin osalle ohjelmaa

3. Käyttötapauskuvaus ja käyttöliittymän luonnos

1. Ohjelma kommunikoi käyttäjän kanssa graafisen käyttöliittymän avulla.
2. Käyttäjä käynnistää ohjelman, ja lataa tilitapahtumatiedoston graafisessa käyttöliittymässä. ReadFiles-luokka alkaa lukemaan annetun tiedoston rivejä ja syöttämään tietoja Transactions-luokalle, joka alustaa Transactions-olioita.
3. Käyttäjä voi ryhmitellä kauppvoja ja tuloja kommunikoiden graafisen käyttöliittymän kanssa. Tällöin GroupHandler-luokka saa tiedon annetusta ryhmästä, siihen lisättävistä kaupoista, ja liittää Transactions-oliot Group-olioon (sekä luo uuden Group-olion, mikäli tätä ei ollut jo olemassa).
4. Käyttäjä voi myös käskä poistamaan tietyn Group-olion, tai poistamaan yksittäisiä Transactions-olioita Group-oliosta. GroupHandler suorittaa nämä pyynnöt.
5. Käyttäjä voi myös pyytää graafisessa käyttöliittymässä piirakkadiagrammin tulostusta. Tällöin DrawDiagram tulostaa pyydettyä kuvaajan. Tulostettava piirakkakaavion riippuu käyttäjän antamasta käskystä (menot/tulot, ryhmitelty/ryhmittelemätön).

4. Ohjelman rakennesuunnitelma



5. Tietorakenteet

Luokat Group ja GroupHandler sisältävät dynaamisia listoja, joissa olioita (Group-luokka sisältää listan Transactions-olioista, GroupHandler-luokka sisältää listan Group-olioista).

Group-olioihin pitää pystyä lisäämään/sieltä pitää pystyä poistamaan Transactions-olioita, joten dynaaminen tietorakenne on välttämätön.

ReadFile-luokassa csv- tai xml-tiedostoa luetaan riveittäin merkkijonoina.

6. Tiedostot ja tiedostformaattit

Tiedostformaattit on .xml, .csv ja .pdf, joista viimeinen muokataan ennen lukemista .csv-muotoon. Tiedostot sisältävät riveittäin tilitapahtumat. Yksittäinen tilitapahtuma (rivi) sisältää tiedon, kirjauspäivästä, määrästä, selityksestä, viitteestä sekä saajan tiedoista. Tiedostot ovat esimerkiksi seuraavanlaisia:

PDF-tiedosto:

Kirjaus- päivä Arvopäivä	Määrä, euroa	Selitys Maksunsaaja/maksaja Viesti	Viite	Arkistointitunnus Saajan tilinumero Saajan pankin BIC
	+311,76	Saldo 31.12.2022		
31.12.2022	-48,00	TILISIIRTO VISMA PAYMENTS OY VIESTI TaxIdNb 0313471-7 Lopullinen saaja: Helsingin Yliopisto	00000000300709366813	20221231593156048277 A FI76 5000 0120 4838 20 OKOYFIHH

CSV-tiedosto:

```
"Kirjauspäivä";"Arvopäivä";"Määrä EUROA";"Laji";"Selitys";"Saaja/Maksaja";"Saajan tilinumero";"Saajan pankin BIC";"Viite";"Viesti";"Arkistointitunnus"  
"2022-12-31";"2022-12-31";"-48,00";"106";"TILISIIRTO";"VISMA PAYMENTS OY";"FI7650000120483820";"OKOYFIHH";"ref=0000000300709366813";"TaxIdNb 0313471-7 Lopullinen saaja: Helsingin Yliopisto";"20221231/593156/048277"  
"2023-01-02";"2023-01-02";"416,00";"521";"ETUUS";"KELA/FPA";"";"";"ref=";"Viesti: ASUMISTUKI Maksajan viite: AY5 20230102K05001009717";"20221229/5UTH00/132049"  
"2023-01-02";"2023-01-02";"268,23";"524";"OPINTOTUKI";"KELA/FPA";"";"";"ref=";"Viesti: OPINTOTUKI Maksajan viite: OT5 20230102K03001020932";"20221229/5UTH00/275618"  
"2023-01-02";"2023-01-02";"-35,90";"162";"PKORTTIMAKSU";"MOB.PAY*HELSINGIN S HELSINKI";"";"";"ref=";"Viesti: 401046*****5313 OSTOPVM 221230MF NRO 74541092365058561083275 VARMENTAJA 400";"20230102/5EQE02/330345"
```

7. Algoritmit

Transactions-luokassa summataan kunkin Transactions-olion (esimerkiksi samaan kauppaan tehtyjen ostosten) määrä yhteen.

Piirakkadiagrammin piirtämistä varten lasketaan sektorien koot. Mikäli piirakkadiagrammi piirretään ilman ryhmittelyä, lasketaan yksittäisen Transactions-olion osuus diagrammista:

$$A_i = \frac{TransactionBalance_i}{\sum TransactionBalance}$$

Mikäli halutaan ryhmitelty piirakkadiagrammi, lasketaan yksittäisen Group-olion sisältämien Transactions-olioiden summan osuus diagrammista:

$$A_i = \frac{GroupBalance_i}{\sum GroupBalance}$$

8. Testaussuunnitelma

test_readfiles:

Testaa, osaako ReadFiles lukea tiedoston rivit oikein ja antaa Transaction-oliolle oikeat tiedot. ReadFiles-luokan pitää siis tunnistaa oikeanlainen tiedostorivi, pilkkoa se osiin oikeista kohdista sekä antaa oikeat tiedot Transactions-luokalle Transactions-olioiden alustamista varten.

ReadFiles-luokan täytyy osata myös hylätä virheellinen tiedosto, ja käsitellä oikein rivejä, joista puuttuu esimerkiksi maksun viite.

test_remove_group:

Testaa, että Transactions-oliot ei poistu, jos GroupHandler-luokka poistaa jonkin Group-olion. GroupHandlerin täytyy löytää oikea Group-olio sekä poistaa se niin, ettei Transactions-oliot poistu.

test_pie_algorithm:

Testaa, että algoritmi laskee piirakkadiagrammin pinta-alat oikein.

9. Kirjastot ja muut työkalut

PyQt: graafista käyttöjärjestelmää varten.

Kirjasto PDF-tiedostojen muokkaamiseksi CSV-muotoon (esim. tabula-py).

10. Aikataulu

27.2 klo 13: projektisuunnitelman esitys

24.3 mennessä:

ReadFiles- ja Transactions-luokkien ohjelmointi. Tavoite saada vaaditut tiedostotyypit avattua, (muokattua), luettua sekä annettua oikeat tiedot Transactions-olioiden alustamista varten. Yksikkötesti test_readfiles tehtynä.

24.3 projektin välipalautus 1 (yllä mainitut luokat tehtynä)

14.4 mennessä:

Luokkien Group ja GroupHandler ohjelmointi. Testi test_remove_group tehtynä.

14.4 projektin välipalautus 2 (test_readfiles tehtynä)

5.5 mennessä:

Graafinen käyttöjärjestelmä + DrawDiagram tehtynä. Testi test_pie_algorithm tehtynä.

5.5 - 12.5:

Hienosäätöä, testailua.

12.5 projektin palautus

17.5 projektin demo

11. Kirjallisuusviitteet ja linkit

Stack Overflow: <https://stackoverflow.com>

Python Official Documentation: <https://docs.python.org>

PyPI: <https://pypi.org/project/tabula-py/>

NCL, pie charts: <https://www.ncl.ac.uk/webtemplate/ask-assets/external/maths-resources/statistics/data-presentation/pie-charts.html>