
Älykkään reseptikirjan dokumentaatio

Release 0.1

Kimi Päivärinta, 351393, AIT 3 vk.

04.05.2016

1	Yleiskuvaus	1
2	Käyttöohje	3
3	Ohjelman rakenne	6
4	Algoritmit	7
5	Yleiskuvaus	8
6	Tiedostot	9
6.1	Raaka-aineet	9
6.2	Reseptit	10
6.3	Varastotilanne	10
7	Yleiskuvaus	12
8	Ohjelman puutteet	13
9	3 heikointa ja 3 parasta kohtaa	14
10	Yleiskuvaus	15
11	Yleiskuvaus	16
12	Arvio lopputuloksesta	17
13	Viitteet	18
14	Liitteet	19
15	Automaattisesti generoitu dokumentaatio	20
15.1	MainGUI	20
15.2	Search	24
15.3	Conversion	25
15.4	Recipe	26
15.5	Ingredient	28
15.6	IO	31
15.7	CustomErrors	32

16 Indices and tables	33
Python Module Index	34
Sisällysluettelo	35

Yleiskuvaus

Toteutin Älykkään reseptikirjan keskivaikeana. Ohjelmassa on graafinen käyttöliittymä, jossa voi selata tallennettuja reseptejä, raaka-aineita sekä varastossa olevia raaka-aineita. Näitä kaikkia on myös mahdollista muokata, sekä reseptejä on mahdollista luoda ohjelmassa. Käyttäjä voi myös tallentaa tekemänsä muutokset tiedostoihin tai halutessaan ladata tiedostosta alkuperäiset reseptit, raaka-aineet ja/tai varastotilanteen. Älykkään reseptikirjasta tekee sen haku-toiminnot. Reseptikirjalla voi hakea reseptit, joihin löytyy raaka-aineet varastosta tai jotka sisältävät esim. tiettyä raaka-ainetta.

Projekti täyttää keskivaikean toteutuksen vaatimukset. Lisätoiminnallisuuksien implementoimisen sijaan keskityin koodin hyvään luettavuuteen, kommentointiin sekä koodin automaattiseen dokumentointiin. Tämän dokumentin lopussa on koodista automaattisesti generoitu dokumentointi, jossa on kerrottu jokaisen oleellisen metodin toiminnallisuus. Näiden lisäksi tein myös kattavat yksikkötestit ohjelman tärkeimmille toiminnoille.

Ohjelman vaatimukset: * Python 3.x versio * PyQt5

Projektin toiminnallisuudet alla. Suurimmilta osin kuvaus on projektin vaatimuksista, mutta osaa kohtia on muokattu.

- Ohjelman käyttäjällä on jääkaappi ja muut kaapit täynnä ruoan raaka-aineita ja tarkat ajanmukaiset tiedot siitä kuinka paljon mitäkin ainetta on varastossa. Ohjelman avulla voidaan etsiä vain ne ruokalajit, jotka voidaan valmistaa käymättä kaupassa tai ruokalajit, jotka vaativat N puuttuvaa tai osittain puuttuvaa ainetta.
- Lisäksi voidaan hakea erityisesti reseptit jotka sisältävät tiettyjä aineita. Ruoka-aineisiin voidaan liittää merkintä siitä, että se sisältää jotakin allergisoivaa tekijää. (Esim maito, kerma jne sisältävät laktoosia) Ohjelmalla tulee voida rajoittaa haku niin, että tiettyssä haussa vältettäväksi halutut allergeenit jäävät pois.
- Jotkut raaka-aineet voidaan myös rakentaa itse reseptistä kuin ruokalajit. Esim jauhelihapihvin resepti sisältää lihamureketaikinaa jolla taas on oma resepti. Vastaavasti joulutortut tehdään voitaikinasta, joka sekin on itse valmistettavissa. Jos jääkaapissa ei ole voitaikinaa, ohjelman tulee yrittää koota taikina raaka-aineista. (Huomaa että tämä tekee raaka-aineista ja tuotetuista ruoista hyvin samankaltaisia...)
- Lienee itsestään selvää että jotkin aineet kuten munat tai sipulit ovat laskettavissa kappaleittain. Moni raaka-aine kuitenkin mitataan erilaisilla mitoilla. Kaapissa oleva jauho myös ostetaan kiloittain, mutta mitataan resepteissä desilitroissa. Tee ohjelmaasi luokka joka hoitaa kaikki muunnokset mittojen välillä. (vinkki: aineella on tiheys)
- Graafinen käyttöliittymä
- (Komentorivi käyttöliittymä, jossa osa toiminnoista, mutta testaaminen hyvin vähäistä.)
- Ohjelman käyttäjällä on jääkaappi ja muut kaapit täynnä ruoan raaka-aineita. Ohjelmalla voidaan pitää näistä kirjaa.
- Ohjelman avulla voidaan etsiä vain ne ruokalajit, jotka voidaan valmistaa käymättä kaupassa tai ruokalajit, jotka vaativat korkeintaan N puuttuvaa tai osittain puuttuvaa (riittää vain pieneen määrään) ainetta.
- Lisäksi voidaan hakea erityisesti reseptit jotka sisältävät tiettyjä aineita. Vaikkapa kala täytyy käyttää pois.

- Raaka-aineisiin tulee voida liittää merkintä siitä, että kyseinen raaka-aine sisältää jotakin allergisoivaa tekijää. (Esim maito, kerma jne sisältävät laktoosia) Ohjelmalla tulee voida rajoittaa reseptien haku niin, että kyseisessä haussa vältettäväksi halutut allergeenit jäävät pois.
- Jotkut raaka-aineet voidaan myös rakentaa itse reseptistä samoin kuin ruokalajit. Esim jauhelihapihvin resepti sisältää lihamureketaikinaa, jolla taas on oma resepti. Vastaavasti joulutortut tehdään voitaikinasta, joka sekin on itse valmistettavissa. Jos jääkaapissa ei ole valmista voitaikinaa, ohjelman tulee katsoa sen sijaan onko taikinan tekoon tarvittavat raaka-aineet tarjolla.
- Lienee itsestään selvää että jotkin aineet kuten munat tai sipulit ovat laskettavissa kappaleittain. Moni raaka-aine kuitenkin mitataan erilaisilla mitoilla. Kaapissa oleva jauho myös ostetaan kiloittain, mutta mitataan resepteissä desilitroissa. Tee ohjelmaasi luokka joka hoitaa kaikki muunnokset mittojen välillä.
- Raaka-aineet allergiamäärittämissä, ruokareseptit ja jääkaapin sisältö tulee tallentaa sopiviin tiedostoihin tekstimuodossa.
- reseptien interaktiivinen rakentaminen ja muokkaaminen käyttöliittymän kautta

Käyttöohje

Ohjelman vaatimukset: * Python 3.x * PyQt5

Ohjelma käynnistetään ajamalla mainGUI.py tiedosto python3:lla. Ohjelman käynnistyessä ladataan reseptit, raaka-aineet ja varastolistaus tiedostoista 'reseptit.txt', 'raaka_aineet.txt' ja 'varasto.csv'. Näiden tiedostojen struktuuri on ihmiselle helposti luettavissa sekä muokattavissa.

Ohjelman käynnistyttyä aukeaa ikkuna, jossa on viisi välilehteä. Varasto, raaka-aineet ja reseptit -välilehdillä on listattuna kaikki ohjelmaan juuri ladatut kyseiset raaka-aineet/resepetit/varastotilanne. Näillä välilehdillä on myöskin mahdollista muokata näiden tietoja. Taulukon riviä klikkaamalla tulevat kyseisen rivin tiedot automaattisesti tekstikenttiin, josta niitä voi muokata. Jokaisella välilehdellä on lyhyt käyttöohje.

Tekstikenttiin pystyy syöttämään vain tiettyjä arvoja, kuten määrä kenttään vain desimaalilukuja. Lähes kaikissa kentissä on kuitenkin myös minimipituus vaatimus, joka käy ilmi vasta tallennettaessa. Huomioitavaa on, että muut tiedot tallentuvat, jos jossakin kentässä onkin virheellinen tieto.

Älykäs reseptikirja

Varasto Raaka-aineet Reseptit Haku Asetukset

Raaka-ainetta klikkaamalla tiedot tulevat alla oleviin kenttiin, jossa niiden muokkaaminen on mahdollista.

Nimi

Määrä

Yksikkö

Tallenna

Päivitä lista

	Nimi	Määrä	Yksikkö
1	Kala	800,0	g
2	Maito	35,0	l
3	Vehnajauho	2,0	kg
4	Pizzataikina	350,0	g
5	Jauheliha	2,0	kg
6	Tomaattimurska	400,0	g
7	Juusto	3,0	kg
8	Oregano	5,0	kg

Reseptilistaus päivitetty

Haku välilehdellä pystyy hakemaan reseptejä eri vaihtoehtoilta tai täysin ilman vaihtoehtoja. Vaikka hakuehdon tekstikentässä lukisi jotain, sitä ei käytetä ellei sitä vastaava checkbox ole ruksattuna.

Älykäs reseptikirja

Varasto Raaka-aineet Reseptit **Haku** Asetukset

Hakuehto on käytössä, vain jos hakuehdon checkboxi on valittuna. Jos mikään hakuehto ei ole valittuna, haetaan kaikki tiedetyt reseptit. Puuttuu N: reseptistä saa puuttua maksimissaan N raaka-ainetta. Löytyy N: reseptin raaka-aineista löydyttävä varastosta vähintään N kappaletta. HUOM! Haku ottaa myös huomioon raaka-aineiden määrät, heittoa saa olla 10%.

Nimi

Raaka-aine

Puuttuu N

Allergeeni

Löytyy N

☒ Nimi
☐ Raaka-aine
☐ Puuttuu N
☐ Ei allergeenia
☐ Löytyy N

	Nimi	Aika	Loppupulos	Allergeenit
1	Jauheliha pizza	40 Min	2,0 annos	Gluteeni, Laktoosi, Tomaatti

Reseptilistaus päivitetty

Asetukset välilehdellä käyttäjä pystyy ladata listoja uudelleen tiedostoista tai tallentamaan ne tiedostoon.

Älykäs reseptikirja

Varasto Raaka-aineet Reseptit Haku **Asetukset**

Lataa uudelleen Tallenna

<input type="text" value="Kaikki"/>	<input type="text" value="Kaikki"/>
<input type="text" value="Raaka-aineet"/>	<input type="text" value="Raaka-aineet"/>
<input type="text" value="Reseptit"/>	<input type="text" value="Reseptit"/>
<input type="text" value="Varastotilanne"/>	<input type="text" value="Varastotilanne"/>

Reseptilistaus päivitetty

Reseptit näkymä on kaikista ”monimutkaisin” ja siellä on eniten toiminnallisuksia. Näkymällä voidaan muokata sekä reseptin perustietoja, raaka-aineita sekä ohjeita. Raaka-aineita ja ohjeita pystyy lisätä sekä poistaa reseptille. Näiden lisäksi näkymällä on mahdollista luoda täysin uusi resepti painamalla ”uusi resepti” -painiketta. Tästä painikkeesta avautuu uusi ikkuna, johon annetaan reseptin tiedot. Reseptin ensimmäinen raaka-aine ja ohje annetaan dialogissa, loput lisätään reseptit näkymällä. Cancel painike peruuttaa luonnin, OK painike tallentaa reseptin, sikäli kun tiedot ovat oikeelliset.

Raaka-aineet ja ohjeet valitaan muokattavaksi myös klikkaamalla taulukosta riviä, kuten muillakin näkymillä. Huo-

mioitavaa on, että ”Lisää uusi” sekä ”poista valittu” toiminnot tallentavat muutokset heti.

Älykäs reseptikirja

Varasto Raaka-aineet **Reseptit** Haku Asetukset

Tällä näkymällä on mahdollista muokata reseptejä sekä niiden raaka-aineita ja ohjeita. Uusi resepti painikkeella voi myös luoda uuden reseptin.

Nimi: Kalaa
Aika: 35
Lopputulos: 4,0
Yksikkö: annos

Tallenna
Päivitä lista
Uusi resepti

Raaka-aine: Kala
Määrä: 5,0
Yksikkö: kg

Tallenna
Lisää uusi
Poista valittu

Nimi	Aika	Lopputulos	Allergeenit
1 Jauheliha pizza	40 Min	2,0 annos	Gluteeni, Laktoosi, Tomaatti
2 Kala pizza	65 Min	1000,0 g	Gluteeni, Tomaatti, Kala, Laktoosi
3 Kalaa	35 Min	4,0 annos	Kala

Nimi	Määrä	Yksikkö
1 Kala	5,0	kg

Ohje: jee

Tallenna
Lisää uusi
Poista valittu

Ohje: jee

Reseptilistaus päivitetty

Ohjelmasta on myös komentorivi versio, jossa on vain osa toiminnallisuuksista sekä sitä on testattu vähemmän. Sen saa toimimaan ajamalla main.py python3:lla.

Ohjelman rakenne

Algoritmit

Yleiskuvaus

Tiedostot

Ohjelma lukee ja tallentaa tekstitiedostoja. Raaka-aineilla, resepteillä ja varastolistauksella on kaikilla omat tiedostoformaattinsa, jotka ovat ihmisen helposti luettavissa sekä muokattavissa. Projektissa on mukana esimerkkiedostot 'raaka_aineet.txt', 'reseptit.txt' sekä 'varasto.csv'.

6.1 Raaka-aineet

Esimerkki ja tietuekuvaus:

```
INGREDIENTLIST

#Ingredient
Date           : 18.4.2015
Name           : Kala
Density        : 0.5
Allergen       : Kala
Recipe         : Jauheliha pizza

#Ingredient
Date           : 18.4.2015
Name           : Maito
Density        : 1.0
Allergen       : Laktoosi
Allergen       : Maito
```

Ekan rivin alussa oltava INGREDIENTLIST, muuten tiedostoa ei lueta.

#Ingredient kertoo uuden raaka-aineen alkamisesta, ei maksimimäärää.

Tiedot kerrotaan yleensä [attribuutti] : [arvo] sekä joillain attribuuteilla on useampia arvoja, kuin 1, nekin ovat erotettu kaksoispisteellä.

Pakolliset attribuutit

Date: Raaka-aineen luontipäivä, oltava muodossa dd.mm.yyyy

Name: Raaka-aineen nimi, oltava yli kaksi merkkiä pitkä

Tiheys: Aineen tiheys (float)

Vapaaehtoiset attribuutit:

Allergen: Allergeeni tekstinä, voi olla useita

Recipe: Resepti

6.2 Reseptit

Esimerkki ja tietuekuvaus:

```

RECIPELIST

#Recipe
Date : 18.4.2016
Name : Jauheliha pizza
Time : 40
Outcome : 2.0 : annos
Ingredient : Jauheliha : 600.0 : g
Ingredient : Pizzataikina : 300.0 : g
Ingredient : Juusto : 0.1 : kg
Ingredient : Oregano : 5.0 : g
Ingredient : Tomaattimurska : 300.0 : g
Instruction : Ruskista jauheliha
Instruction : Kaada kaikki sekoitettuna uuniin
Instruction : Nauti kylman PepsiColan kanssa.

```

Ensimmäisen rivin alussa oltava RECIPELIST, muuten tiedostoa ei tunnisteta.

#Recipe kertoo uuden reseptin alkamisesta. Tiedostossa voi olla useita reseptejä.

Tiedot kerrotaan yleensä [attribuutti] : [arvo] sekä joillain attribuuteilla on useampia arvoja, kuin 1, nekin ovat erotettu kaksoispisteellä.

Pakolliset attribuutit:

Date: reseptin luontipäivä, oltava muodossa dd.mm.yyyy

Name: reseptin nimi, oltava yli 2 merkkiä pitkä

Time: Kauanko reseptin tekemiseen menee minuuteissa, kokonaisluku

Outcome: reseptin lopputuloksen määrä (float) ja yksikkö

Ingredient: raaka-aineen nimi, määrä (float) ja yksikkö. Raaka-aineiden maksimimäärää ei määritelty.

Instruction: Ohje, ohjeet tulee reseptille tässä järjestyksessä.

6.3 Varastotilanne

Csv-tiedosto, jossa soluerottimena puolipiste eikä tekstin ympärillä "" merkkejä. Esimerkki ja tietuekuvaus:

```

STORAGELIST
Kala;800.0;g
Maito;35.0;l
Vehnajauho;2.0;kg
Pizzataikina;350.0;g
Jauheliha;2.0;kg
Tomaattimurska;400.0;g
Juusto;3.0;kg
Oregano;5.0;kg

```

Ensimmäisellä rivillä oltava STORAGELIST tai tiedostoa ei lueta.

Rivin ensimmäisessä solussa raaka-aineen nimi

Rivin toisessa solussa määrä (float) desimaalipilkulla tai pisteellä

Rivin kolmannessa solussa määrän yksikkö

Yleiskuvaus

Ohjelman puutteet

3 heikointa ja 3 parasta kohtaa

Yleiskuvaus

Yleiskuvaus

Arvio lopputuloksesta

Viitteet

Liitteet

Automaattisesti generoitu dokumentaatio

15.1 MainGUI

Created on 27.4.2016

@author: Kimi Päivärinta

class `mainGUI.MainGUI`

Tämä luokka perii pyqt:n QMainWindow luokan sekä QT Designerilla luodun Ui_MainWindow luokan, joka on moduulissa GUIDesign. Ui_MainWindow luokka sisältää graafisen käyttöliittymän designin.

Luokka sisältää paljon metodeja, joilla tehdään muutoksia, kun käyttöliittymällä tapahtuu muutoksia. Metodit voidaan jakaa karkeasti osiin :init*: Luokan luomisen yhteydessä asetetaan nappuloiden toiminnallisuudet, ladataan tiedostoja ym. :populate*: Piirretään data johonkin tauluun :save*: Tallennetaan muuttunutta dataa :add*: Lisätään uusi ohje/raaka-aine/ym. :get*InDataListForTable: Metodit palauttavat niille annetun listan olennaisimmat tiedot "data" tyyppinä, joka voidaan antaa populateTableWith() metodille populoitavaksi

addNewRecipeIngredient ()

Tämä metodi tarkastaa ja tallentaa reseptinäkömällä raaka-aine tekstikentissä olevat tekstit uudeksi raaka-aineeksi reseptille. Oikean reseptin löytymiseksi hyödynnetään self.recipeToEdit muuttujaa.

Tallennuksen jälkeen metodi populoi uudelleen reseptin raaka-ainelistaus taulun.

addNewRecipeInstruction ()

Tämä metodi tarkastaa ja tallentaa reseptinäkömällä ohjetekstikentässä olevan tekstin uudeksi ohjeeksi reseptille. Oikean reseptin löytymiseksi hyödynnetään self.recipeToEdit muuttujaa.

Tallennuksen jälkeen metodi populoi uudelleen reseptin ohjelistaus taulun.

checkStateChanged (state)

Tämä metodi pitää huolen, ettei hakunäkymällä olevat "Puuttuu N" ja "Löytyy N" checkboksit ole valittuna samaan aikaan, koska se ei ole järin järkevä hakuvaihtoehto.

Tätä metodia kutsutaan vaihtamalla jomman kumman checkboksen tilaa.

clearRecipeEditLineEdits ()

Tämä metodi tyhjentää reseptin raaka-aine sekä ohjeen tekstikentät. Tätä metodia kutsutaan, kun reseptilistaus taulua klikataan, jotta kyseisissä tekstikentissä ei "vanhoja" tietoja.

deleteRecipeIngredient ()

Tämä metodi poistaa valitun reseptin valitun raaka-aineen.

Metodia kutsutaan painamalla poista painiketta. Oikea resepti ja raaka-aine selviää muuttujista self.recipeToEdit ja self.recipeIngredientToEdit

deleteRecipeInstruction ()

Tämä metodi poistaa valitun reseptin valitun ohjeen.

Metodia kutsutaan painamalla poista painiketta. Oikea resepti ja ohje selviää muuttujista `self.recipeToEdit` ja `self.recipeInstructionToEdit`

getIngredientContainersInDataListForTable (ingredientContainerList)

Muodostaa annettujen raaka-aineiden(container) tiedoista listojen listan, jotka on helppo populoida QTabletWidget tauluun.

getIngredientsInDataListForTable (ingredientList)

Muodostaa annettujen raaka-aineiden tiedoista listojen listan, jotka on helppo populoida QTabletWidget tauluun.

getRecipesInDataListForTable (recipeList)

Muodostaa annettujen reseptien tiedoista listojen listan, jotka on helppo populoida QTabletWidget tauluun.

initButtons ()

Tässä metodissa määritellään käyttöliittymän painikkeiden toiminallisuudet

initDialogLineEdits ()

Tässä metodissa alustetaan tarvittavat validaattorit sekä asetetaan nämä validaattorit käyttöön käyttöliittymän "Uusi resepti" dialogin tekstikenttiin

initLineEdits ()

Tässä metodissa alustetaan tarvittavat validaattorit sekä asetetaan nämä validaattorit käyttöön käyttöliittymän tekstikentille.

Käytettävät validaattorit ovat :min2char: vähintään 2 merkkiä pitkä :min1tomax10char: 1-10 merkkiä aakkosia :float2decimals: desimaaliluku, jossa 2 desimaalia :onlyint: ainoastaan kokonaislukuja

initSettingsAndLists ()

Tämä metodi alustaa tarvittavat muuttujat ja oliot sekä lataa reseptit, raaka-aineet ja varastotilanteen tiedoista listoihin.

initTablesAndLists ()

Tässä metodissa määritellään käyttöliittymän taulukkojen toiminnallisuudet, esim. kun taulukon riviä klikataan

initToEditVariables ()

Tässä metodissa alustetaan apumuuttujat, joilla pidetään kirjaa mikä rivi mistäkin taulukosta on valittuna. Tätä tietoa tarvitaan taulukon tietojen muuttamisessa, esim. reseptin nimen.

loadFromFileToList ()

Tämä metodi päättää kutsujan (`self.sender()`) perusteella mikä tiedosto tulee ladata uudelleen ja myöskin lataa sen.

Jos `self.sender()` on `None`, niin ladataan kaikki tiedostot. Muulloin kutsuja on esim. `self.buttonLoadRecipes`, jolloin ladataan reseptit tiedostosta.

openFileUTF8 (file)

Tämä metodi avaa halutun tiedoston UTF-8 enkoodauksella ja palauttaa sen tiedostokahvan.

Args:

file tiedoston polku

Returns:

Onnistuessa tiedostokahva avattuun tiedostoon

Epäonnistuessa False

populateAllMainTables ()

Tämä metodi kutsuu kaikkien päätaulukoiden (reseptit, varastotilanne, raaka-aineet) populointimetoja

populateIngredientsEditFields (mi)

Tämä metodi populoi raaka-ainenäkömällä olevat tekstikentät, joilla voi muokata raaka-aineen tietoja. Tätä metodia kutsutaan, kun raaka-ainelistaus taulua klikataan.

Rivin indeksi tallennetaan self.ingredientToEdit muuttujaan, jotta muutoksia tallennettaessa tiedetään mitä raaka-ainetta täytyy muokata.

Args:

mi mi muuttuja/olio, joka sisältää klikatun rivin ja kolumnin indeksin.

populateIngredientsTable ()

Tämä metodi populoi raaka-ainelistaus tauluun kaikki tiedetyt raaka-aineet

populateRecipesEditFields (mi)

Tämä metodi populoi reseptinäkömällä olevat reseptin tekstikentät sekä tyhjentää reseptin raaka-aine ja ohje muokkaustekstikentät sekä kutuu metodeja, joilla populoidaan reseptin raaka-aine ja ohjetaulut.

Tätä metodia kutsutaan, kun reseptilistaus taulua klikataan.

Rivin indeksi tallennetaan self.recipeToEdit muuttujaan, jotta muutoksia tallennettaessa tiedetään mitä reseptiä täytyy muokata.

Args:

mi mi muuttuja/olio, joka sisältää klikatun rivin ja kolumnin indeksin.

populateRecipesIngredientEditFields (mi)

Tämä metodi populoi reseptinäkömällä olevat raaka-aine tekstikentät, joilla voi muokata reseptin raaka-aineen tietoja. Tätä metodia kutsutaan, kun reseptin raaka-ainelistaus taulua klikataan.

Rivin indeksi tallennetaan self.recipeIngredientToEdit muuttujaan, jotta muutoksia tallennettaessa tiedetään mitä raaka-ainetta täytyy muokata.

Args:

mi mi muuttuja/olio, joka sisältää klikatun rivin ja kolumnin indeksin.

populateRecipesIngredientsTable ()

Tämä metodi populoi reseptinäkömällä reseptin raaka-aineet tauluun. Tätä metodia kutsutaan, kun reseptilistaus taulua klikataan.

Oikean reseptin löytämiseksi hyödynnetään self.recipeToEdit muuttujaa.

populateRecipesInstructionsEditFields (mi)

Tämä metodi populoi reseptinäkömällä olen ohje tekstikentät, joilla voi muokata reseptin ohjetta. Tätä metodia kutsutaan, kun reseptin ohjelistaus taulua klikataan.

Rivin indeksi tallennetaan self.recipeInstructionToEdit muuttujaan, jotta muutoksia tallennettaessa tiedetään mitä raaka-ainetta täytyy muokata.

Args:

mi mi muuttuja/olio, joka sisältää klikatun rivin ja kolumnin indeksin.

populateRecipesInstructionsTable ()

Tämä metodi populoi reseptinäkömällä reseptin ohjeet tauluun. Tätä metodia kutsutaan, kun reseptilistaus taulua klikataan.

Oikean reseptin löytämiseksi hyödynnetään self.recipeToEdit muuttujaa.

populateRecipesTable ()

Tämä metodi populoi reseptilistaus tauluun kaikki tiedetyt reseptit

populateSearchTable ()

Tämä metodi populoi datan hakunäkymän taulukkoon. Tätä metodia kutsuu hakunäkymän ”Hae” painike.

Hakunäkymällä on mahdollista valita nolla tai useampi hakuvaihtoehto, tämä metodi sisältää myös logiikan valittujen vaihtoehtojen tarkastamiselle sekä oikean hakutuloksen saamiseksi.

Haku hyödyntää Search luokan metodeja, jotka palauttavat listan resepteistä, jotka täyttävät hakukriteerin. Hakulogiikka on ns. iteratiivinen ja alussa Search luokan metodeille annetaan listana kaikki tunnetut reseptit, mahdollinen seuraava haku kuitenkin tehdään edellisen haun palauttamasta listasta, jolloin lopuksi listassa on enää kaikki hakutulokset täyttävät reseptit.

populateStorageEditFields (mi)

Tämä metodi populoi varastonäkymällä olevat tekstikentät, joilla voi muokata varastossa olevaa raaka-ainetta. Tätä metodia kutsutaan, kun varastolistauksessa klikataan riviä.

Rivin indeksi tallennetaan self.storageToEdit muuttujaan, jotta muutoksia tallennettaessa tiedetään mitä varasto raaka-ainetta täytyy muokata

Args:

mi mi muuttuja/olio, joka sisältää klikatun rivin ja kolumnin indeksin.

populateStorageTable ()

Tämä metodi populoi varastolistaus tauluun kaikki varastossa olevat raaka-aineet

populateTableWithData (table, data)

Tämä metodi populoi annettuun QTableWidgetItem tauluun annetun datan. Datan tulee olla muotoa esim. data = [['Nimi', 'Määrä'], ['Kala', 'Peruna'], [5,3]]

Taulun sisältö tyhjennetään aluksi, asetetaan asetetaan sarakkeiden ja rivien lukumäärät. Tämän jälkeen taulu populoidaan datalla, jonka jälleen kolumnien leveydet skaalataan sisällölle sopivaksi.

saveIngredientsEdit ()

Tämä metodi tarkistaa raaka-aineen perustietojen tekstikenttien arvot, jonka jälkeen muutokset tallennetaan raaka-aineeseen.

Oikea raaka-aine löydetään self.ingredientToEdit muuttujan avulla.

saveNewRecipe ()

Tämä metodi tarkistaa uusi resepti -dialogissa annetut tiedot ja valideja, niin luo uuden reseptin näillä tiedoilla ja lisää sen reseptilistaan. Tämän jälkeen reseptilistaus taulu populoidaan uudelleen.

saveRecipesEdit ()

Tämä metodi tarkistaa reseptin perustietojen tekstikenttien arvot, jonka jälkeen muutokset tallennetaan reseptiin.

Oikea resepti löydetään self.recipeToEdit muuttujan avulla.

saveRecipesIngredientEdit ()

Tämä metodi tarkistaa reseptin raaka-aineen tekstikenttien arvot, jonka jälkeen muutokset tallennetaan reseptiin.

Oikea resepti sekä raaka-aine löydetään self.recipeToEdit ja self.recipeIngredientToEdit muuttujien avulla.

saveRecipesInstructionsEdit ()

Tämä metodi tallentaa annetun ohjetekstin reseptille.

Oikea resepti sekä ohje löydetään self.recipeToEdit ja self.recipeInstructionToEdit muuttujien avulla.

saveStorageEdit ()

Tämä metodi tarkistaa varastossa olevan raaka-aineen perustietojen tekstikenttien arvot, jonka jälkeen muutokset tallennetaan raaka-aineeseen.

Oikea raaka-aine löydetään self.storageToEdit muuttujan avulla.

saveToFile ()

Tämä metodi päättää kutsujan (self.sender()) perusteella mikä tiedosto tulee tallentaa ja myöskin tallentaa sen.

Jos self.sender() on self.buttonSaveAll, niin ladataan kaikki tiedostot. Muulloin kutsuja on esim. self.buttonSaveRecipes, jolloin tallennetaan reseptit tiedostoon.

showCreateNewRecipeDialog ()

Tämä metodi luo uuden QDialog widgetin, jossa voi täyttää uuden reseptin tietoja. Dialogin graafinen ulkoasu on luotu QT Designerilla ja on tallennettuna moduuliin GUIrecipeDialog.

Jos dialogissa painetaan "OK" painiketta, kutsutaan metodia self.saveNewRecipe(), joka annetut tiedot ja tallentaa reseptin.

15.2 Search

Created on 19.4.2016

@author: Kimi Päiväranta

class search.Search

Tämän luokan metodeilla on mahdollista tehdä erilaisia hakuja. Ohjelma hyödyntää tätä luokkaa reseptien etsimisessä tietyillä hakuehdoilla

amountDifferenceMax10Perc (ingredientRecipe, ingredientStorage)

Apumetodi, jota ei ole tarkoitus kutsua luokan ulkopuolelta.

Tällä metodilla tarkastetaan onko varastossa tarpeeksi tiettyä raaka-ainetta. Jos resepti vaatii 1kg jauhelihaa, niin 900g kin kyllä riittää, joten erotus saa olla maksimissaan 10% vaaditusta määrästä.

Args:

ingredientRecipe Reseptissä oleva ingredientContainer olio

ingredientStorage Varastossa oleva ingredientContainer olio

Returns:

True Raaka-aineiden määrän erotus on alle 10% tarvittavasta eli reseptin määrästä

False Erotus on yli 10% tarvittavasta määrästä

howManyIngredientsFoundInStorage (recipe)

Apumetodi, jota ei ole tarkoitus kutsua luokan ulkopuolelta.

Metodi käy reseptin raaka-aineet läpi ja montako niistä on tarpeeksi varastossa, metodi siis tarkastaa myös määrän. Jos raaka-ainetta ei ole tarpeeksi varastossa, tarkastetaan voidaanko raaka-aine itse valmistaa reseptistä, jolloin kutsutaan tätä samaa metodia

searchForhRecipesNIngredientsInStorage (recipesList, N, storageList, NNotInStorage)

Tällä metodilla voidaan etsiä reseptejä, joihin löytyy vähintään N raaka-ainetta varastosta tai puuttuu maksimissaan N raaka-ainetta.

Args:

recipesList Lista kaikista reseptiolioista

N Monta raaka-ainetta löydyttävä/puututtava varastosta (int)

storageList Lista varastossa olevista raaka-aineolioista

NNotInStorage (Boolean) True arvolla N merkitsee sitä montako saa puuttua. False arvolla N merkitsee montako täytyy löytyä

Returns:

recipesFound Lista reseptio-olioista

searchFromList (*searchFor, searchList*)

Tällä metodilla etsitään olio listasta tiettyä oliota sen nimen perusteella. Vertailu tapahtuu stringeinä ja vaatii, että listassa olevilla olioilla on getName() metodi.

Args:

searchFor Etsittävä nimi (string)

searchList Lista olioista, joilla on getName() metodi

returns:

recipesFound Lista löydettyistä olioista. Tyhjä, jos ei löytynyt

searchIncludesIngredient (*ingredientStr, recipesList*)

Tällä metodilla etsitään reseptejä, jotka sisältävät tietyn raaka-aineen. Vertailutapahtuu stringeinä

Args:

ingredientStr Raaka-aineen nimi (string)

recipesList Lista reseptio-olioista

Returns:

recipesFound Lista reseptio-olioista. Tyhjä, jos ei löytynyt

searchNoAllergen (*allergenStr, recipesList*)

Tämmä metodilla etsitään reseptejä, jotka eivät sisällä tiettyä allergeeniä. Allergeenien vertailu tapahtuu stringeinä

Args:

allergenStr Allergeeni (string)

recipesList Lista reseptio-olioista

Returns:

recipesFound Lista reseptio-olioista. Tyhjä, jos ei löytynyt

15.3 Conversion

Created on 19.4.2016

@author: Kimi Päiväranta

class `conversion.Conversion`

Conversion luokka sisältää kaikki ohjelman tuntemat yksiköt sekä niiden "suhdeluvut", jotka mahdollistaa muunnokset. Tämän luokan avulla onnistuu yksikkömuunnokset.

convertFromTo (*amount, unitFrom, unitTo, density*)

Tekee yksikkömuunnoksen ja palauttaa muunnetun arvon. Jos ei muunnettavissa, palauttaa alkuperäisen arvon.

Args:

amount Muunnettava määrä (float)
unitFrom Alkuperäinen yksikkö
unitTo Haluttu yksikkö
density aineen tiheys, tarvitaan vain kun muunnetaan massa<->tilavuus (float)

Returns:

Muunnetun yksikön (float)

convertMassToMass (*amount, unitFrom, unitTo*)

Muuttaa massa yksikön massa yksiköksi, esim. g->kg

convertMassToVolume (*amount, unitFrom, unitTo, density*)

Muuttaa massa yksikön tilavuus yksiköksi, esim. kg -> l. Hyödyntää kaavaa $v = m / r$

convertVolumeToMass (*amount, unitFrom, unitTo, density*)

Muuttaa tilavuus yksikön massa yksiköksi, esim. l -> kg. Hyödyntää kaavaa $m = v * r$

convertVolumeToVolume (*amount, unitFrom, unitTo*)

Muuttaa tilavuus yksikön tilavuus yksiköksi, esim. l -> m3

isValidUnit (*unit*)

Tällä metodilla voidaan tarkistaa onko sille parametrina annettu yksikkö ohjelman tuntema

Returns:

Tunnettu True

Ei tunnettu False

15.4 Recipe

Created on 19.4.2016

@author: Kimi Päivärinta

class `recipe.Recipe`

Luokka reseptejä varten. Tämä luokka pitää sisällään kaikki reseptille ominaiset attribuutit sekä tarvittavat metodit niiden arvojen muuttamiseksi.

Attributes:

self.date Luontipäivä

self.name Reseptin nimi

self.time Reseptin tekemiseen menevä aika minuuttina (int)

self.instructions Ohjeet (str[])

self.outcomeSize Reseptin lopputuloksen koko, esim. 4 (kg)

self.outcomeUnit Reseptin lopputuloksen yksikkö, esim (4) kg

self.ingredients Raaka-aineet (object[])

Returns: Attribuuttien muuttamiseen käytettävät metodit (set* & add* & delete/remove*) palauttavat True, jos muutos onnistuu

Raises: Kaikki attribuuttien asettamiseen käytettävät metodit (set* & add*) heittävät SetAttributeErrorin, jos attribuutin arvon asetus epäonnistuu.

addIngredientContainer (*ingredientContainer*)

Validoi, että lisättävä raaka-aine on IngredientContainer-olio sekä lisää raaka-aineen(Container) self.ingredients[] listaan

addInstruction (*instruction*)

Validoi, että ohje on yli 2 merkkiä pitkä ja lisää sen self.instruction[] listaan

deleteIngredient (*index*)

Poistaa reseptiltä raaka-aineen. Argumenttina annetaan ohjeen sijainti listassa (index)

deleteInstruction (*index*)

Poistaa reseptiltä ohjeen. Argumenttina annetaan ohjeen sijainti listassa (index)

getAllergensDistinctGUI ()

Palauttaa reseptien raaka-aineiden stringinä pilkulla erotettuna. Allergeeni esiintyy listassa vain kerran, vaikka se olisi monessa raaka-aineessa.

getDate ()

Palauttaa reseptin luontipäivän

getIngredients ()

Palauttaa reseptin raaka-aine oliot listana

getIngredientsGUI ()

Palauttaa raaka-aineiden nimet listana

getIngredientsStr ()

Palauttaa reseptin raaka-aineet stringinä, hyödyntää raaka-aine luokan __str__() metodia

getInstructions ()

Palauttaa ohjeet listana

getInstructionsStr ()

Palauttaa ohjeet stringinä, jokainen ohje omalla rivillä ja edessä ohjeen järjestysnumero eli järjestys listassa

getName ()

Palauttaa reseptin nimen

getOutcomeSize ()

Palauttaa lopputuloksen floattina

getOutcomeSizeGUI ()

Palauttaa lopputuloksen stringinä desimaalipilkulla

getOutcomeStr ()

Palauttaa reseptin lopputuloksen desimaalipilkulla muodossa "<määrä> <yksikkö>"

getOutcomeUnit ()

Palauttaa lopputuloksen yksikön

getTime ()

Palauttaa reseptin tekemiseen menevän ajan inttinä

getTimeGUI ()

Palauttaa reseptin tekemiseen menevän ajan stringinä

getTimeStr ()

Palauttaa reseptin tekemiseen menevän ajan stringinä, jonka lopussa on " Min"

setDate (*date*)

Validoi päivämäärän ja asettaa sen stringinä: self.date

setName (*name*)

Validoi, että nimi on yli 2 merkkiä pitkä ja asettaa sen: self.name

setOutcomeSize (*outcomeSize*)

Muuttaa desimaalipilkun pisteeksi ja settaa määrän floattina: self.outcomeSize

setOutcomeUnit (*outcomeUnit*)

Validoi, että yksikkö on ohjelman tuntema ja asettaa sen: self.outcomeUnit

setTime (*time*)

Asettaa reseptin tekemiseen menevän ajan (min) inttinä: self.time

15.5 Ingredient

Created on 19.4.2016

@author: Kimi Päiväranta

class ingredient.Ingredient

Raaka-aine luokka. Tämä luokka sisältää perustiedot raaka-aineesta, varastossa ja resepteissä olevat "raaka-aineet" sisältävät tämän olion.

Attributes:

self.date Luontipäivä

self.name Nimi

self.density Tiheys yksikkömuunnoksia varten (float)

self.allergens Allergeenit (str[])

self.recipe Mahdollinen resepti (object)

self.recipeLoaded Kertoo onko raaka-aineen resepti ladattu. None = Ei reseptiä, False = Resepti on, mutta oliota ei ladattu, True = Olio ladattu

Returns: Attribuuttien muuttamiseen käytettävät metodit (set* & add* & delete/remove*) palauttavat True, jos muutos onnistuu

Raises: Attribuuttien asettamiseen käytettävät metodit (set* & add*) heittävät SetAttributeErrori:n, jos validointi epäonnistuu

addAllergen (*allergen*)

Validoi, että allergeeni on yli 2 merkkiä pitkä ja lisää sen raaka-aineen allergeeni listaan

getAllergens ()

Palauttaa allergeenit listana

getAllergensGUI ()

Palauttaa allergeenit stringinä pilkulla ja välilyönnillä erotettuna

getAllergensStr ()

Palauttaa allergeenit stringinä pilkulla erotettuna sekä alkussa teksti "Allergeenit: "

getDate ()

Palauttaa luontipäivän

getDensity ()

Palauttaa tiheyden floattina

getDensityGUI ()

Palauttaa tiheyden stringinä desimaalipilkulla

getName ()
 Palauttaa nimen

getRecipe ()
 Palauttaa resepti olion, jos se on asetettu ja ladattu

getRecipeGUI ()
 Palauttaa reseptin nimen stringinä

getRecipeLoaded ()
 Palauttaa self.recipeLoaded arvon. None = Ei reseptiä, True = Resepti oli ladattu, False = Reseptiä ei vielä ladattu

getRecipeStr ()
 Palauttaa "Resepti: " + reseptin nimi, jos resepti on asetettu ja ladattu

loadRecipe (recipesList)
 Etsii nimen perusteella reseptilistasta raaka-aineelle halutun reseptin ja asettaa olion: self.recipe

Args:

recipesList Lista kaikista resepteistä

Returns:

Onnistuessa True

Ei ladattavaa None

Raises:

SetAttributeError Reseptiä ei löytynyt

removeAllergens ()
 Poistaa kaikki raaka-aineen allergeenit

removeRecipe ()
 Asettaa self.recipe = None sekä self.recipeLoaded = None

setDate (date)
 Validoi päivämäärän ja asettaa sen: self.date

setDensity (density)
 Muuttaa desimaalipilkun pisteeksi, muuntaa tiheyden float luvuksi ja asettaa sen: self.density

setName (name)
 Validoi, että nimi on yli 2 merkkiä pitkä ja asettaa sen self.date

setRecipe (recipe)
 Raaka-aineet luetaan sisälle ennen reseptejä, joten reseptin oliota ei todennäköisesti ole vielä olemassa. Validoi, että resepti on yli kaksi merkkiä pitkä sekä asettaa halutun reseptin nimen stringinä: self.recipe sekä asettaa self.recipeLoaded = False

class ingredient.IngredientContainer

Tämä luokka sisältää viittauksen raaka-aine olioon ja tämän lisäksi omat attribuutit määrästä sekä yksiköstä. Tätä luokkaa hyödynnetään varastolistauksen sekä reseptien raaka-aineiden tallentamisessa.

Attributes:

self.ingredient Raaka-aine olio

self.quantity Raaka-aineen määrä

self.unit Määrän yksikkö

Returns: :Attribuuttien muuttamiseen käytettävät metodit (set* & add* & delete/remove*) palauttavat True, jos muutos onnistuu

Raises: :Attribuuttien asettamiseen käytettävät metodit (set* & add*) heittävät SetAttributeError:n, jos validointi epäonnistuu

getAllergens ()

Palauttaa allergeenit listana

getAllergensStr ()

Palauttaa allergeenit stringinä pilkulla erotettuna sekä alkussa teksti "Allergeenit: "

getDensity ()

Palauttaa raaka-aineen tiheyden floattina

getIngredient ()

Palauttaa raaka-aine olion

getName ()

Palauttaa raaka-aineen nimen

getQuantity ()

Palauttaa määrän floattina

getQuantityStr ()

Palauttaa määrän stringinä, desimaalipilkulla

getRecipe ()

Palauttaa reseptiolion, jos raaka-aineella on

getRecipeStr ()

Palauttaa reseptin nimen stringinä, jos raaka-aineella on

getUnit ()

Palauttaa määrän yksikön

hasRecipe ()

Palauttaa True, jos raaka-aineella on tallennettu resepti, muuten false

setIngredient (ingredient, ingredientsList)

Etsii halutun raaka-aineen annetusta raaka-ainelistasta nimen perusteella sekä asettaa sen: self.ingredient

Attributes:

ingredient Etsittävä raaka-aine (string)

ingredientsList Lista kaikista raaka-aineista

Returns:

Onnistuessa True

Raises: :SetAttributeError epäonnistuessa

setQuantity (quantity)

Muuttaa desimaalipilkun pisteeksi ja asettaa määrän floattina: self.quantity

setUnit (unit)

Validoi, että yksikkö on ohjelman tuntema sekä asettaa sen: self.unit

15.6 IO

Created on 19.4.2016

@author: Kimi Päivärinta

class `IO.IO`

Input output luokka, jolla ladataan raaka-aineet, reseptit ja varastolistaus tiedostoista. Tämän lisäksi nämä voidaan tallentaa tiedostoihin annetusta listasta.

loadIngredients (*inputLines*)

Tämä metodi lukee raaka-aineet tiedostosta ja palauttaa ne listana.

Args:

inputLines tiedoston kahva (fileIO), josta tiedot luetaan.

Returns:

ingredientList Raaka-aine oliot listana

successCount Montako luettiin onnistuneesti sisään

errorCount Montako jäi virheeseen luettaessa

Raises:

CorruptedFileError Jos kohdataan tuntematon tiedostotyyppi

loadRecipes (*inputLines, ingredientsList*)

Tämä metodi lukee reseptit tiedostosta ja palauttaa ne listana.

Args:

inputLines tiedoston kahva (fileIO), josta tiedot luetaan.

ingredientList Tunnetut raaka-aineet listana

Returns:

recipesList Resepti oliot listana

successCount Montako luettiin onnistuneesti sisään

errorCount Montako jäi virheeseen luettaessa

Raises:

CorruptedFileError Jos kohdataan tuntematon tiedostotyyppi

loadRecipesForIngredients (*ingredientsList, recipesList*)

Tällä metodilla voidaan ladata kaikkien raaka-aineiden reseptit. Käytännössä tämä metodi kutsuu raaka-aineen loadrecipe() metodia

Args:

ingredientList Raaka-aine oliot listana

recipesList Reseptio oliot listana

loadStorage (*inputLines, ingredientsList*)

Tämä metodi lukee varastolistauksen tiedostosta ja palauttaa ne listana.

Args:

inputLines tiedoston kahva (fileIO), josta tiedot luetaan.

ingredientList Tunnetut raaka-aineet listana

Returns:

storageList Raaka-aine oliot listana
successCount Montako luettiin onnistuneesti sisään
errorCount Montako jäi virheeseen luottaessa

Raises:

CorruptedFileError Jos kohdataan tuntematon tiedostotyyppi

saveIngredients (*fileName, ingredientsList*)

Tämä metodi tallentaa raaka-aineet tiedostoon ohjelman luettavassa muodossa. Tiedot tallennetaan ensin temp.ing tiedostoon, jonka jälkeen kyseinen tiedosto nimetään uudelleen halutun nimiseksi.

Args:

fileName Tiedostonimi, johon raaka-aineet tallennetaan
ingredientList Raaka-aine oliot listana

saveRecipes (*fileName, recipesList*)

Tämä metodi tallentaa reseptit tiedostoon ohjelman luettavassa muodossa. Tiedot tallennetaan ensin temp.rec tiedostoon, jonka jälkeen kyseinen tiedosto nimetään uudelleen halutun nimiseksi.

Args:

fileName Tiedostonimi, johon reseptit tallennetaan
recipesList Resepti oliot listana

saveStorage (*fileName, storageList*)

Tämä metodi tallentaa varastossa olevat raaka-aineet tiedostoon ohjelman luettavassa muodossa. Tiedot tallennetaan ensin temp.sto tiedostoon, jonka jälkeen kyseinen tiedosto nimetään uudelleen halutun nimiseksi.

Args:

fileName Tiedostonimi, johon varaston raaka-aineet tallennetaan
storageList Varaston raaka-aine oliot listana

15.7 CustomErrors

Created on 19.4.2016

@author: Kimi Päivärinta

exception customErrors.**CorruptedFileError**

Korruptoituneen tiedoston exception

exception customErrors.**SetAttributeError**

Käytetään, kun olion attribuutin asettamisessa tapahtuu virhe

Indices and tables

- `genindex`
- `modindex`
- `search`

c

`conversion`, [25](#)
`customErrors`, [32](#)

i

`ingredient`, [28](#)
`IO`, [31](#)

m

`mainGUI`, [20](#)

r

`recipe`, [26](#)

s

`search`, [24](#)

A

addAllergen() (ingredient.Ingredient method), 28
addIngredientContainer() (recipe.Recipe method), 27
addInstruction() (recipe.Recipe method), 27
addNewRecipeIngredient() (mainGUI.MainGUI method), 20
addNewRecipeInstruction() (mainGUI.MainGUI method), 20
amountDifferenceMax10Perc() (search.Search method), 24

C

checkStateChanged() (mainGUI.MainGUI method), 20
clearRecipeEditLineEdits() (mainGUI.MainGUI method), 20
Conversion (class in conversion), 25
conversion (moduuli), 25
convertFromTo() (conversion.Conversion method), 25
convertMassToMass() (conversion.Conversion method), 26
convertMassToVolume() (conversion.Conversion method), 26
convertVolumeToMass() (conversion.Conversion method), 26
convertVolumeToVolume() (conversion.Conversion method), 26
CorruptedFileError, 32
customErrors (moduuli), 32

D

deleteIngredient() (recipe.Recipe method), 27
deleteInstruction() (recipe.Recipe method), 27
deleteRecipeIngredient() (mainGUI.MainGUI method), 20
deleteRecipeInstruction() (mainGUI.MainGUI method), 20

G

getAllergens() (ingredient.Ingredient method), 28
getAllergens() (ingredient.IngredientContainer method), 30

getAllergensDistinctGUI() (recipe.Recipe method), 27
getAllergensGUI() (ingredient.Ingredient method), 28
getAllergensStr() (ingredient.Ingredient method), 28
getAllergensStr() (ingredient.IngredientContainer method), 30
getDate() (ingredient.Ingredient method), 28
getDate() (recipe.Recipe method), 27
getDensity() (ingredient.Ingredient method), 28
getDensity() (ingredient.IngredientContainer method), 30
getDensityGUI() (ingredient.Ingredient method), 28
getIngredient() (ingredient.IngredientContainer method), 30
getIngredientContainersInDataListForTable() (mainGUI.MainGUI method), 21
getIngredients() (recipe.Recipe method), 27
getIngredientsGUI() (recipe.Recipe method), 27
getIngredientsInDataListForTable() (mainGUI.MainGUI method), 21
getIngredientsStr() (recipe.Recipe method), 27
getInstructions() (recipe.Recipe method), 27
getInstructionsStr() (recipe.Recipe method), 27
getName() (ingredient.Ingredient method), 28
getName() (ingredient.IngredientContainer method), 30
getName() (recipe.Recipe method), 27
getOutcomeSize() (recipe.Recipe method), 27
getOutcomeSizeGUI() (recipe.Recipe method), 27
getOutcomeStr() (recipe.Recipe method), 27
getOutcomeUnit() (recipe.Recipe method), 27
getQuantity() (ingredient.IngredientContainer method), 30
getQuantityStr() (ingredient.IngredientContainer method), 30
getRecipe() (ingredient.Ingredient method), 29
getRecipe() (ingredient.IngredientContainer method), 30
getRecipeGUI() (ingredient.Ingredient method), 29
getRecipeLoaded() (ingredient.Ingredient method), 29
getRecipesInDataListForTable() (mainGUI.MainGUI method), 21
getRecipeStr() (ingredient.Ingredient method), 29
getRecipeStr() (ingredient.IngredientContainer method), 30

getTime() (recipe.Recipe method), 27
 getTimeGUI() (recipe.Recipe method), 27
 getTimeStr() (recipe.Recipe method), 27
 getUnit() (ingredient.IngredientContainer method), 30

H

hasRecipe() (ingredient.IngredientContainer method), 30
 howManyIngredientsFoundInStorage() (search.Search method), 24

I

Ingredient (class in ingredient), 28
 ingredient (moduuli), 28
 IngredientContainer (class in ingredient), 29
 initButtons() (mainGUI.MainGUI method), 21
 initDialogLineEdits() (mainGUI.MainGUI method), 21
 initLineEdits() (mainGUI.MainGUI method), 21
 initSettingsAndLists() (mainGUI.MainGUI method), 21
 initTablesAndLists() (mainGUI.MainGUI method), 21
 initToEditVariables() (mainGUI.MainGUI method), 21
 IO (class in IO), 31
 IO (moduuli), 31
 isValidUnit() (conversion.Conversion method), 26

L

loadFromFileToList() (mainGUI.MainGUI method), 21
 loadIngredients() (IO.IO method), 31
 loadRecipe() (ingredient.Ingredient method), 29
 loadRecipes() (IO.IO method), 31
 loadRecipesForIngredients() (IO.IO method), 31
 loadStorage() (IO.IO method), 31

M

MainGUI (class in mainGUI), 20
 mainGUI (moduuli), 20

O

openFileUTF8() (mainGUI.MainGUI method), 21

P

populateAllMainTables() (mainGUI.MainGUI method), 21
 populateIngredientsEditFields() (mainGUI.MainGUI method), 22
 populateIngredientsTable() (mainGUI.MainGUI method), 22
 populateRecipesEditFields() (mainGUI.MainGUI method), 22
 populateRecipesIngredientEditFields() (mainGUI.MainGUI method), 22
 populateRecipesIngredientsTable() (mainGUI.MainGUI method), 22

populateRecipesInstructionsEditFields() (mainGUI.MainGUI method), 22
 populateRecipesInstructionsTable() (mainGUI.MainGUI method), 22
 populateRecipesTable() (mainGUI.MainGUI method), 22
 populateSearchTable() (mainGUI.MainGUI method), 23
 populateStorageEditFields() (mainGUI.MainGUI method), 23
 populateStorageTable() (mainGUI.MainGUI method), 23
 populateTableWithData() (mainGUI.MainGUI method), 23

R

Recipe (class in recipe), 26
 recipe (moduuli), 26
 removeAllergens() (ingredient.Ingredient method), 29
 removeRecipe() (ingredient.Ingredient method), 29

S

saveIngredients() (IO.IO method), 32
 saveIngredientsEdit() (mainGUI.MainGUI method), 23
 saveNewRecipe() (mainGUI.MainGUI method), 23
 saveRecipes() (IO.IO method), 32
 saveRecipesEdit() (mainGUI.MainGUI method), 23
 saveRecipesIngredientEdit() (mainGUI.MainGUI method), 23
 saveRecipesInstructionsEdit() (mainGUI.MainGUI method), 23
 saveStorage() (IO.IO method), 32
 saveStorageEdit() (mainGUI.MainGUI method), 23
 saveToFile() (mainGUI.MainGUI method), 24
 searchForIngredientsInStorage() (search.Search method), 24
 Search (class in search), 24
 search (moduuli), 24
 searchFromList() (search.Search method), 25
 searchIncludesIngredient() (search.Search method), 25
 searchNoAllergen() (search.Search method), 25
 SetAttributeError, 32
 setDate() (ingredient.Ingredient method), 29
 setDate() (recipe.Recipe method), 27
 setDensity() (ingredient.Ingredient method), 29
 setIngredient() (ingredient.IngredientContainer method), 30
 setName() (ingredient.Ingredient method), 29
 setName() (recipe.Recipe method), 27
 setOutcomeSize() (recipe.Recipe method), 28
 setOutcomeUnit() (recipe.Recipe method), 28
 setQuantity() (ingredient.IngredientContainer method), 30
 setRecipe() (ingredient.Ingredient method), 29
 setTime() (recipe.Recipe method), 28
 setUnit() (ingredient.IngredientContainer method), 30

showCreateNewRecipeDialog() (mainGUI.MainGUI
method), [24](#)