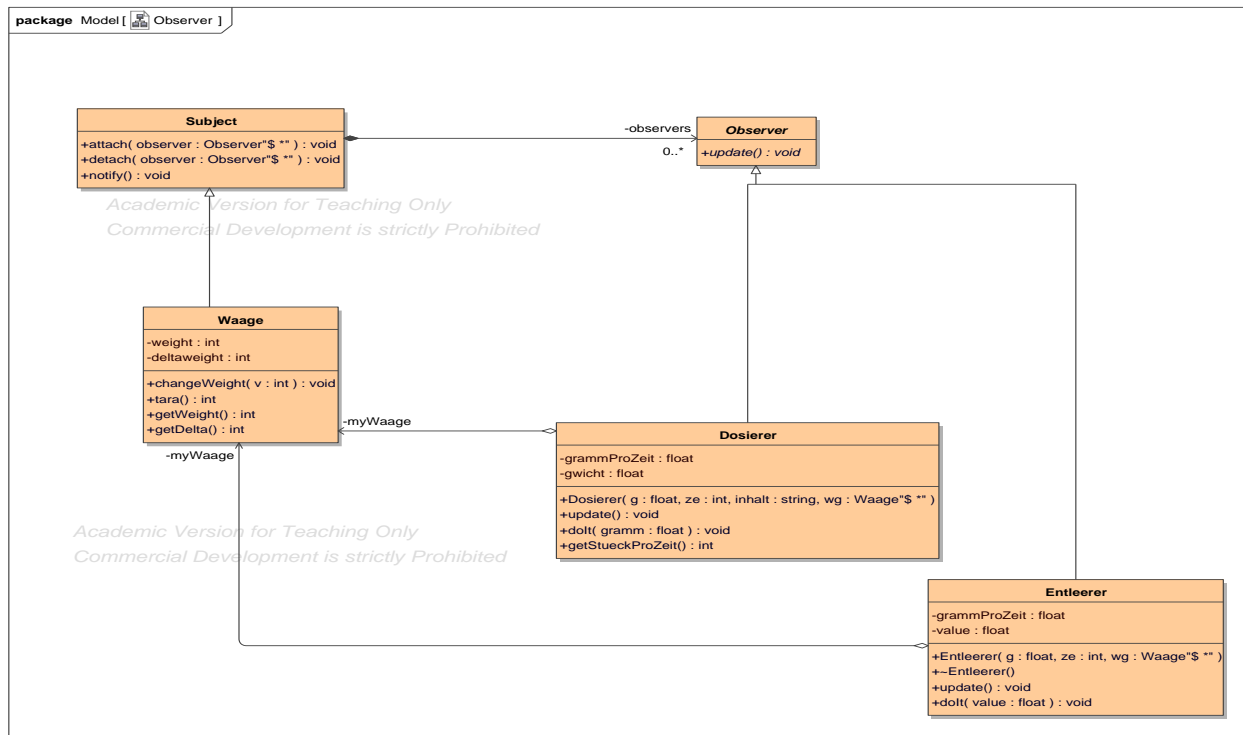


Date: Oct 5, 2021 1:09:03 PM

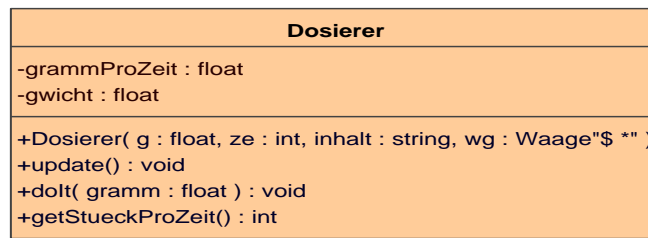
1 Klassen

[illegible]

Observer



1.1 Dosierer



Klasse	
Klassenname	Dosierer
Dokumentation	
Abstrakt	false
Generalisierung	<ul style="list-style-type: none"> Observer InternalDevice
Attribute	<pre>private grammProZeit : float "" private myWaage : Waage "" private gewicht : float "" private inhalt : string ""</pre>
Operatoren	<pre>public Dosierer (g : float ze : int inhalt : string wg : Waage) : void public update () : void public dolt (gramm : float) : void public getStueckProZeit () : int</pre>

1.2 Entleerer

Entleerer
-grammProZeit : float -value : float
+Entleerer(g : float, ze : int, wg : Waage"\$ *") +~Entleerer() +update() : void +dolt(value : float) : void

Klasse	
Klassenname	Entleerer
Dokumentation	
Abstrakt	false
Generalisierung	<ul style="list-style-type: none"> • Observer • InternalDevice
Attribute	private grammProZeit : float "" private myWaage : Waage "" private value : float ""
Operatoren	public Entleerer (g : float ze : int wg : Waage) : void public ~Entleerer () : void public update () : void public dolt (value : float) : void

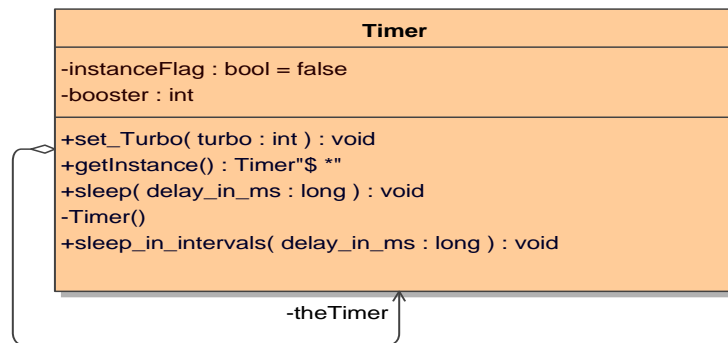
1.3 DeviceVerwalter

DeviceVerwalter
+DeviceVerwalter(zv : VorhandeneZutaten"\$ *") +rezeptSchrittZubereiten(zutat : string, menge : float) : void -setZutatenVerwalter(zv : VorhandeneZutaten"\$ *") : void -createDevices() : void

Klasse	
Klassenname	DeviceVerwalter
Dokumentation	Befuellt die Dosierer, verwaltet die Interne Hardware
Abstrakt	false
Generalisierung	
Attribute	public myDevices : InternalDevice "std::map<std::string, \$ *>" public myEntleerer : Entleerer "" private myMixer : Mixer "" private myStampfer : Stampfer "" private mySchuettler : Schuettler ""

	private myZutatenVerwalter : VorhandeneZutaten "" private theWaage : Waage ""
Operatoren	public DeviceVerwalter (zv : VorhandeneZutaten) : void public rezeptSchrittZubereiten (zutat : string menge : float) : void private setZutatenVerwalter (zv : VorhandeneZutaten) : void private createDevices () : void

1.4 Timer



Klasse	
Klassenname	Timer
Dokumentation	
Abstrakt	false
Generalisierung	
Attribute	private instanceFlag : bool "" private theTimer : Timer "" private booster : int ""
Operatoren	public set_Turbo (turbo : int) : void public getInstance () : Timer public sleep (delay_in_ms : long) : void private Timer () : void public sleep_in_intervals (delay_in_ms : long) : void • Doku: * ~Timer () { instanceFlag = false; };

1.5 Waage

Waage
-weight : int -deltaweight : int
+changeWeight(v : int) : void +tara() : int +getWeight() : int +getDelta() : int

Klasse	
Klassenname	Waage
Dokumentation	
Abstrakt	false
Generalisierung	<ul style="list-style-type: none">Subject
Attribute	private weight : int "" private deltaweight : int ""
Operatoren	public changeWeight (v : int) : void public tara () : int public getWeight () : int public getDelta () : int

1.6 CocktailZubereiter

CocktailZubereiter
+CocktailZubereiter(dv : DeviceVerwalter"\$ *") +cocktailZubereiten(rzpt : Recipe"\$ *") : bool

Klasse	
Klassenname	CocktailZubereiter
Dokumentation	* Bereitet den Cocktail nach den Rezeptschritten zu.
Abstrakt	false
Generalisierung	
Attribute	private myDeviceVerwalter : DeviceVerwalter ""
Operatoren	public CocktailZubereiter (dv : DeviceVerwalter) : void public cocktailZubereiten (rzpt : Recipe) : bool

1.7 Schuettler

Schuettler
+Schuettler() +dolt(time : float) : void

Klasse	
Klassenname	Schuettler
Dokumentation	
Abstrakt	false
Generalisierung	<ul style="list-style-type: none"> • InternalDevice
Attribute	
Operatoren	public Schuettler () : void public dolt (time : float) : void

1.8 Mixer

Mixer
+Mixer() +dolt(time : float) : void

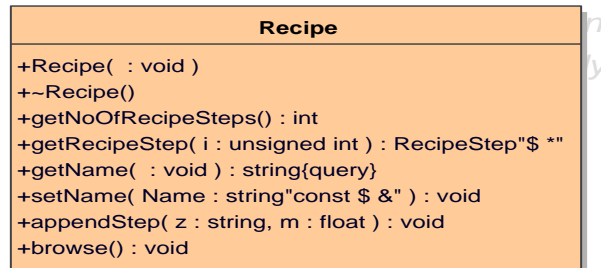
Klasse	
Klassenname	Mixer
Dokumentation	
Abstrakt	false
Generalisierung	<ul style="list-style-type: none"> • InternalDevice
Attribute	
Operatoren	public Mixer () : void public dolt (time : float) : void

1.9 MischbaresRezeptbuch

MischbaresRezeptbuch
+MischbaresRezeptbuch(zv : VorhandeneZutaten"\$ *") +browse() : void -setZutatenVerwalter(zv : VorhandeneZutaten"\$ *") : void

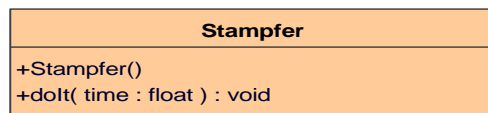
Klasse	
Klassenname	MischbaresRezeptbuch
Dokumentation	* Kennt die mischbaren Rezepte. Kann diese Ausgeben.
Abstrakt	false
Generalisierung	<ul style="list-style-type: none"> • RecipeBook
Attribute	private myZutatenVerwalter : VorhandeneZutaten "" private rezepte : Recipe "std::vector<\$ *>"
Operatoren	public MischbaresRezeptbuch (zv : VorhandeneZutaten) : void public browse () : void private setZutatenVerwalter (zv : VorhandeneZutaten) : void

1.10 Recipe



Klasse	
Klassenname	Recipe
Dokumentation	
Abstrakt	false
Generalisierung	
Attribute	private m_RecipeStep : RecipeStep "std::list<\$ *>" private m_Name : string ""
Operatoren	public Recipe (: void) : void public ~Recipe () : void public getNoOfRecipeSteps () : int public getRecipeStep (i : unsigned int) : RecipeStep public getName (: void) : string public setName (Name : string) : void public appendStep (z : string m : float) : void public browse () : void

1.11 Stampfer



Klasse	
Klassenname	Stampfer
Dokumentation	
Abstrakt	false
Generalisierung	<ul style="list-style-type: none"> • InternalDevice
Attribute	
Operatoren	public Stampfer () : void public dolt (time : float) : void

1.12 Subject

Subject
+attach(observer : Observer"\$ *") : void +detach(observer : Observer"\$ *") : void +notify() : void

Klasse	
Klassenname	Subject
Dokumentation	
Abstrakt	false
Generalisierung	
Attribute	private observers : Observer "std::vector<\$ *>"
Operatoren	public attach (observer : Observer) : void public detach (observer : Observer) : void public notify () : void

1.13 RecipeBook

RecipeBook
+RecipeBook(: void) +-RecipeBook() +getNumberOfRecipes() : int +getRecipe(i : unsigned int) : Recipe"\$ *" +deleteRecipe(i : unsigned int) : bool

Klasse	
Klassenname	RecipeBook
Dokumentation	* @class RecipeBook @brief RecipeBook manages a collection of recipes A RecipeBook knows a list of recipes You can ask for the number of recipes, query recipes by index or delete a recipe
Abstrakt	false
Generalisierung	
Attribute	private m_Recipe : Recipe "std::list<\$ *>" <ul style="list-style-type: none"> Doku: * This is the list of recipes that the RecipeBook manages
Operatoren	public RecipeBook (: void) : void <ul style="list-style-type: none"> Doku: * @brief Konstruktor that creates a RecipeBook @return A pointer to the created RecipeBook

	<p>This method creates a RecipeBook. It uses temporarily an object of the class Initialisierer to initialize the list of recipes. After that the Initialisierer is destroyed.</p> <pre>public ~RecipeBook () : void</pre> <ul style="list-style-type: none"> Doku: * @brief Destructor that deletes the RecipeBook <pre>public getNumberOfRecipes () : int</pre> <ul style="list-style-type: none"> Doku: * @brief Returns the number of recipes @return The number of recipes in the RecipeBook <pre>public getRecipe (i : unsigned int) : Recipe</pre> <ul style="list-style-type: none"> Doku: * @brief fetches the i-th recipe @param [in] i a positive integer for the i-th recipe @return A pointer to the requested recipe object, NULL for invalid i <p>This method returns the i-th recipe; where i is an integer from 0 to n-1 where n is the number of recipes in the RecipeBook</p> <pre>public deleteRecipe (i : unsigned int) : bool</pre> <ul style="list-style-type: none"> Doku: * @brief Deletes the i-th recipe from the RecipeBook @param [in] i a positive integer for the i-th recipe @return Boolean that indicates whether the deletion was successful <p>This method deletes the i-th recipe from the RecipeBook. And returns true if the deletion was successful (valid i). For values of i that are invalid (i < 0 or i >= n) the function return false.</p>
--	--

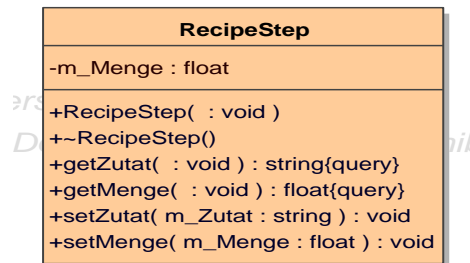
1.14 VorhandeneZutaten

VorhandeneZutaten	
-DEBUG : bool"const \$" = false{readOnly}	
-anzahlDosierer : int	
+VorhandeneZutaten(: void)	
+~VorhandeneZutaten()	
+browse(: void) : void	
+getZutat(i : int) : string	
+getAnzahlVorhandeneZutaten() : int	
-DummyZutatenEinfuegen() : void	
-ZutatenDateiEinlesen(myfile : string) : void	

Klasse	
Klassenname	VorhandeneZutaten
Dokumentation	<p>*</p> <p>Liest Zutaten aus Liste ein. Merkt sich in der Liste die Zutaten + Aggregatzustand.</p>

Abstrakt	false
Generalisierung	
Attribute	private zutaten : string "std::vector<\$>" private DEBUG : bool "" private anzahlDosierer : int ""
Operatoren	public VorhandeneZutaten (: void) : void public ~VorhandeneZutaten () : void public browse (: void) : void public getZutat (i : int) : string public getAnzahlVorhandeneZutaten () : int private DummyZutatenEinfuegen () : void private ZutatenDateiEinlesen (myfile : string) : void

1.15 RecipeStep



Klasse	
Klassenname	RecipeStep
Dokumentation	
Abstrakt	false
Generalisierung	
Attribute	private m_Zutat : string "" private m_Menge : float ""
Operatoren	public RecipeStep (: void) : void <ul style="list-style-type: none"> Doku: * Konstruktor public ~RecipeStep () : void public getZutat (: void) : string public getMenge (: void) : float public setZutat (m_Zutat : string) : void public setMenge (m_Menge : float) : void

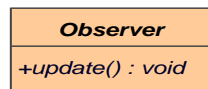
1.16 CocktailPro

CocktailPro
<pre>#waehle() : int #demo() : void +CocktailPro(argc : int, param : char"\$ * *") +start() : void</pre>

Klasse	
Klassenname	CocktailPro
Dokumentation	<p>*</p> <p>Gesamtsystem @class CocktailPro</p> <p>@brief CocktailPro diese Klasse leitet die Einfuehrung in das Programm</p> <p>CocktailPro startet den Prozess und in dem wird die Rezeptnummer ausgewaehlt.</p>
Abstrakt	false
Generalisierung	
Attribute	<p>protected theCocktailZubereiter : CocktailZubereiter ""</p> <ul style="list-style-type: none"> Doku: * Objekt von CocktailZubereiter <p>protected theDeviceVerwalter : DeviceVerwalter ""</p> <ul style="list-style-type: none"> Doku: * Objekt von DeviceVerwalter <p>protected theMischbaresRezeptbuch : MischbaresRezeptbuch ""</p> <ul style="list-style-type: none"> Doku: * Objekt von MischbaresRezeptbuch <p>protected theZutatenVerwalter : VorhandeneZutaten ""</p> <ul style="list-style-type: none"> Doku: * Objekt von VorhandeneZutaten
Operatoren	<p>protected waehle () : int</p> <ul style="list-style-type: none"> Doku: * @brief waehle() dient zum Rezeptnummerauswahl @return die ausgewaehlte Rezeptnummer oder -1 zum Beenden <p>und gibt die erste Einleitung aus.</p> <p>protected demo () : void</p> <ul style="list-style-type: none"> Doku: * @brief demo() macht das gleiche wie Methode start() @see start() <p>aber die ausgewaelte Rezeptnummer ist festgestellt bzw. ist(0). d.h. es wird den Rezeptnamen gezeigt und das Cocktail wird vorbereitet.</p>

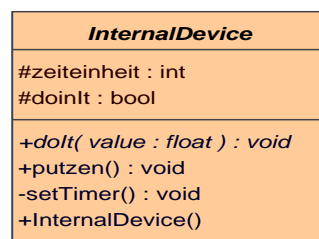
	<pre>public CocktailPro (argc : int param : char) : void</pre> <ul style="list-style-type: none"> Doku: * @brief Konstruktor erzeugt objekte von CocktailPro @return ein Zeiger auf das erzeugten Objekt @param argc ist die Anzahl der Befehlszeilenoptionen @param turbo Die einzelnen Befehlszeilenoptionen befinden sich im diesem Array. <p>da werden die im protected Variabeln mit dynamischen Objekte intialisiert.</p> <pre>public start () : void</pre> <ul style="list-style-type: none"> Doku: * @brief diese Methode startet den Prozess <p>Diese Methode fragt den Benutzer nach einer Rezeptnummer ,die er auswaehlen soll um den Prozess weiterzufuehren und kontrolliert die ausgewaehlte Nummer, ob es sinnvoll ausgewaehlt oder nicht das Cocktail wir auch durch eine aufgerufte Methode vorbereitet.</p>
--	--

1.17 Observer



Klasse	
Klassenname	Observer
Dokumentation	
Abstrakt	true
Generalisierung	
Attribute	
Operatoren	public update () : void

1.18 InternalDevice



Klasse	
Klassenname	InternalDevice
Dokumentation	* Abstraktion von Dosierern und Verarbeitungseinheiten

CocktailPro 2015

Abstrakt	true
Generalisierung	
Attribute	protected zeiteinheit : int "" protected doInt : bool "" protected myTimer : Timer ""
Operatoren	public dolt (value : float) : void public putzen () : void private setTimer () : void public InternalDevice () : void