

Cambios:

02/06/2023

Receta.h

- Agregar un método de sobrescritura de **imprime()** en las clases: Receta, PlatoFuerte, Postre y Snack.
- Agregar la definición del constructor de Receta que recibe parámetros
- Agregar virtual a la clase padre: Receta, para aplicar polimorfismo:

```
56 virtual string cocinar(){
57     return "Cocinándose";
58 } //método donde se utilizará sobre escritura
59
60 virtual string imprime()=0;
61
62 };
```

Recetario.h

- Agregar métodos de búsqueda e impresión del recetario por tipo

```
58 //método que despliega las recetas de un solo tipo: Postre, Plato Fuerte o snack
59 void Recetario::mostrarRecetario(string _tipo){
60     cout<<"\n";
61     cout<<"Recetas tipo: "<<_tipo<<endl;
62     cout<<"\n";
63     for (int i=0;i<numReceta;i++){
64         if (recetas[i]->get_tipo()==_tipo)
65             cout<< recetas[i]->imprime();
66     }
67 }
68
69 //método que hace una búsqueda por nombre de las recetas
70 void Recetario::busqueda_nombre(string _nombre){
71     cout<<"\n";
72     cout<<"Receta: "<<_nombre<<endl;
73     cout<<"\n";
74     for (int i=0;i<numReceta;i++){
75         if (recetas[i]->get_nombre()==_nombre)
76             cout<< recetas[i]->imprime();
77     }
78 }
```

```
80 //método que hace una búsqueda por duración de preparación
81 void Recetario::busqueda_duracionPreparacion(){
82     float min;
83     float max;
84     cout<<"Ingresa el rango de tiempo en minutos \nMínimo: ";
85     cin>>min;
86     cout<<"Máximo: ";
87     cin>>max;
88     cout<<"\n";
89     for (int i=0;i<numReceta;i++){
90         if (recetas[i]->get_duracionPreparacion()>min && recetas[i]->get_duracionPreparacion()<max)
91             cout<< recetas[i]->imprime();
92     }
93 }
94
```

Main.cpp

- Añadir interacción con el usuario para que le vaya preguntando si agregar recetas, imprimir el recetario o buscar recetas.

Read me

- Añadir casos que harían que el proyecto deje de funcionar.

09/06/2023

Receta.h

- Agregar comentarios
- Eliminar método de cocinar ya que no es muy relevante ni de utilidad para el proyecto

Main.cpp

- Agregar una introducción del programa que se muestra al usuario
- Agregar código para que el usuario pueda ingresar más de una palabra en ingredientes e instrucciones

```
65     string _ingredientes;
66     cout<<"\n Agrega los ingredientes separados por espacios: "<<endl;
67     cin.ignore();
68     getline(cin, _ingredientes); //para que pueda recibir una línea de texto(palabras separadas por espacios)
```

Read me

- Añadir casos que harían que el proyecto deje de funcionar.

15/06/2023

Main.cpp

- Agregar código para que el usuario pueda ingresar más de una palabra en el nombre de la receta y modificar cuando se manda a llamar al método de búsqueda por nombre para que también pueda recibir más de una palabra.

```
58     if (respuesta1==1){
59         //Pregunta al usuario para que ingrese los datos de la receta
60         string _nombre;
61         cout<<"\n Ingresa el nombre de la receta: "<<endl;
62         cin.ignore();
63         getline(cin, _nombre); //para que pueda recibir varias palabras
64         cin>>_nombre;
```

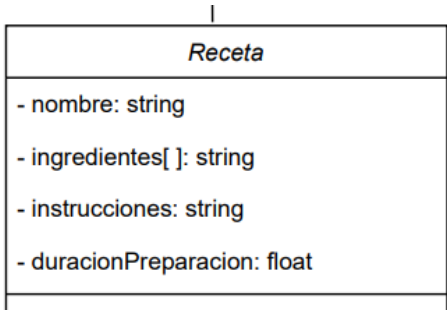
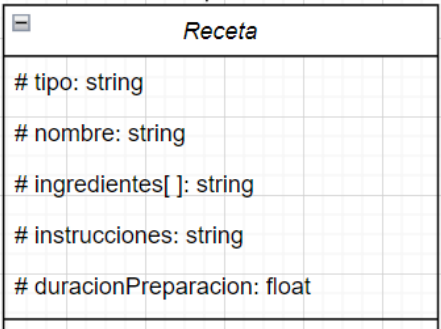
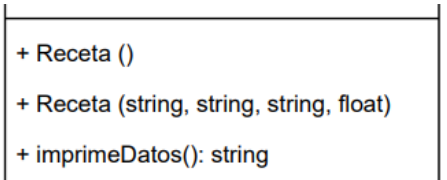
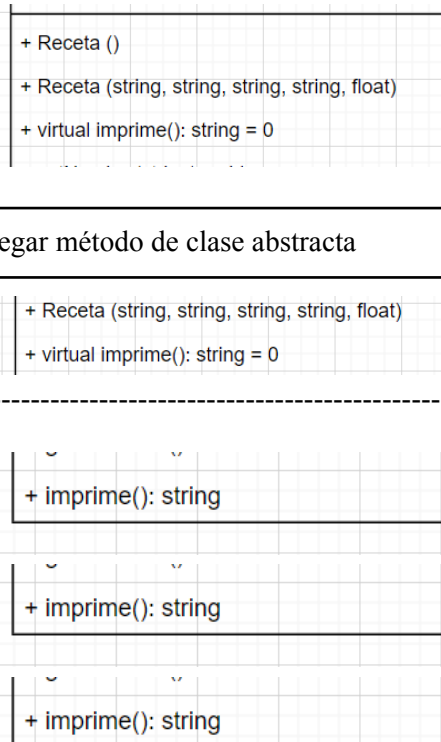
```

126     string nomb;
127     cout<<"Nombre de la receta: ";
128     cin.ignore();
129     getline(cin, nomb); //Puede recibir más de una palabra
130
131     recetario.busqueda_nombre(nomb);

```

Diagrama UML

- Hacer modificaciones al diagrama UML, agregar atributos y métodos

Anterior	Corrección
 <pre> classDiagram class Receta { - nombre: string - ingredientes[]: string - instrucciones: string - duracionPreparacion: float } </pre>	 <pre> classDiagram class Receta { # tipo: string # nombre: string # ingredientes[]: string # duracionPreparacion: float } </pre>
Tipo: private	Tipo protected Agregar atributo de tipo
 <pre> classDiagram class Receta { + Receta () + Receta (string, string, string, float) + imprimeDatos(): string } </pre>	 <pre> classDiagram class Receta { + Receta () + Receta (string, string, string, string, float) + virtual imprime(): string = 0 } class Imprime1 { + imprime(): string } class Imprime2 { + imprime(): string } class Imprime3 { + imprime(): string } </pre>
	Agregar método de clase abstracta

	<ul style="list-style-type: none"> - Agregar método donde se utiliza sobre escritura en la clase padre: Receta con virtual - Agregar método imprime() donde se utiliza sobre escritura en las clases hijas sin virtual
<pre> class Recetario { - receta: Receta[] + Recetario() + AgregaReceta(Receta): void + MostrarRecetario(): string + busqueda_nombre(): string + busqueda_ingredientes(): string + busqueda_duracionPreparacion(): string } </pre>	<pre> class Recetario { - receta: Receta[] - numReceta: int + Recetario() + agregaPostre(string, string, string, float, int): void + agregaPlatoFuerte(string, string, string, float, string): void + agregaSnack(string, string, string, float, string): void + mostrarRecetario(): void + mostrarRecetario(string): void + busqueda_nombre(string): void + busqueda_duracionPreparacion(): void } </pre>
	<ul style="list-style-type: none"> - Agregar atributo de numRecetas - agregaPostre, agregaPlatoFuerte, agregaSnack - Agregar un mostrarRecetario(string) para aplicar sobrecarga

Main.cpp, Receta.h, Recetario.h

- Agregar comentarios con el formato según se pide.