



Patrón Iterator y Composite

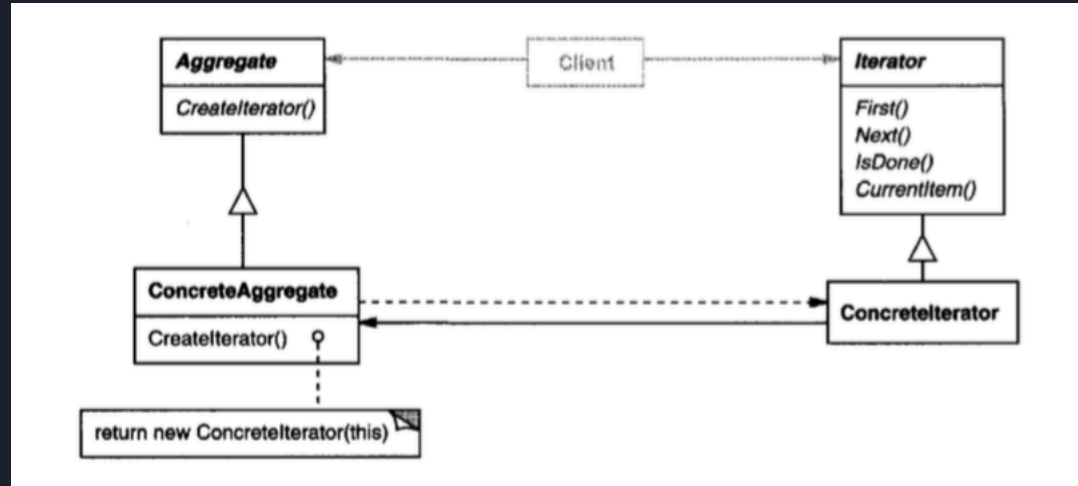
Equipo Los Qarlos	
Melissa Janet Treviño	A00816715
Raúl Axel Suárez	A01195895
José Manuel González	A01280106
José Carlos Villarreal	A01088593



Iterator

- El patrón del iterador cae bajo la categoría de **patrón de comportamiento**.
- Busca proporcionar una manera de **acceder secuencialmente a los elementos de un objeto acumulado o colección** sin exponer su estructura interna.
- Sirve para implementar diferentes formas de recorrer objetos de tipo colección sin modificar las operaciones de la clase.
- Asume la responsabilidad de acceso y recorrido de colecciones.

Iterator – Diagrama general

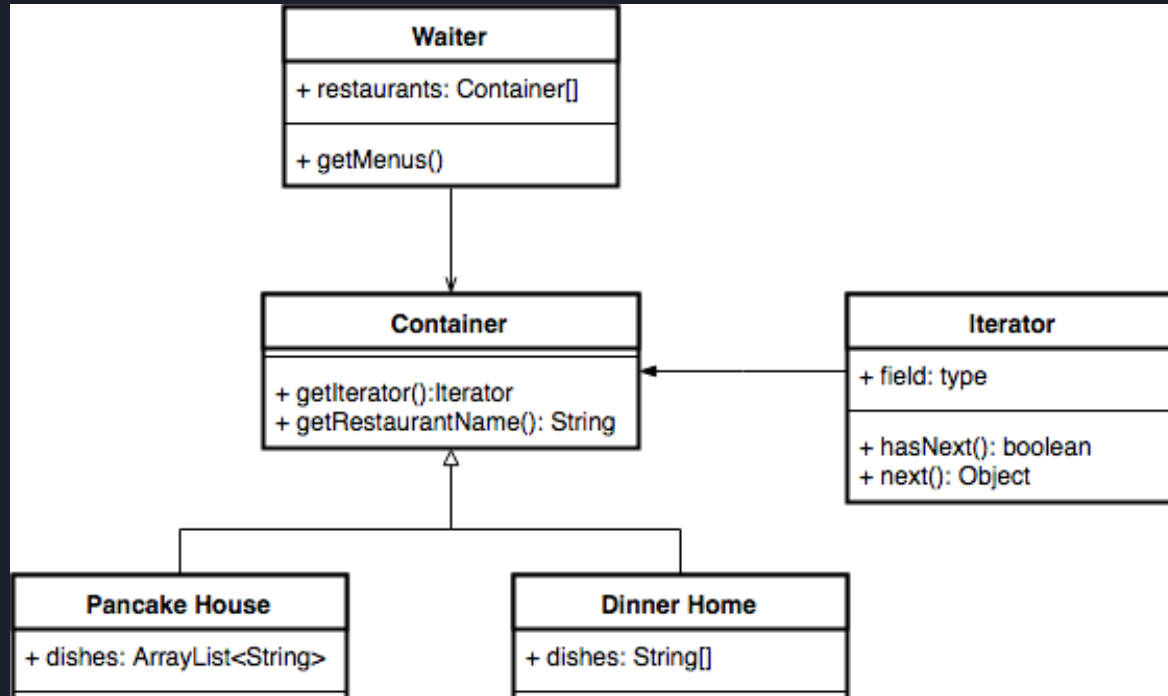




Iterator - Problema

- Dos compañías (Dinner Home y Pancake House) se unen para ofrecer sus platillos.
- Los menús tienen implementaciones diferentes.
- El mesero tiene que describir/imprimir los platillos que le son solicitados.
- Ninguna de las compañías está dispuesta a cambiar su implementación.

Iterator – Diagrama aplicado

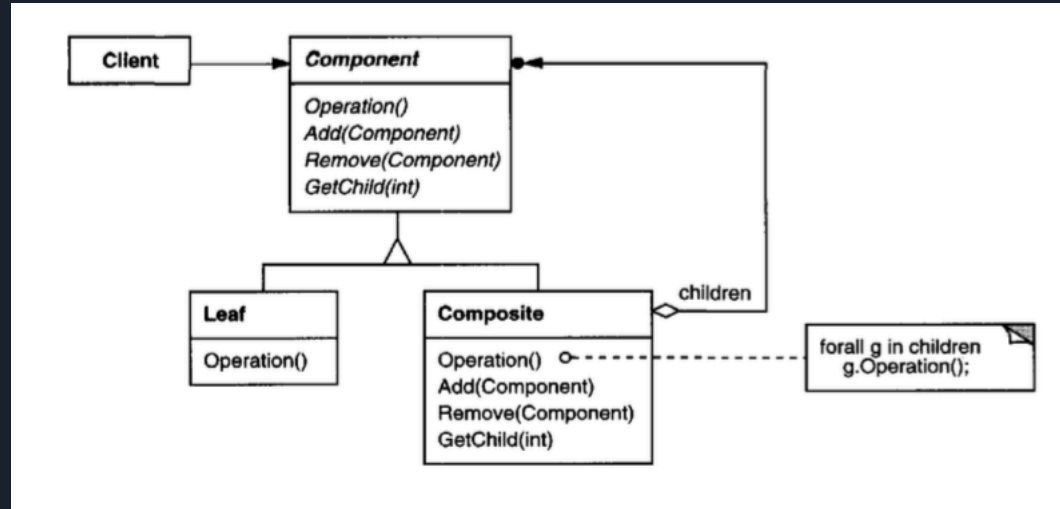




Composite

- Es un patrón estructural.
- Se utiliza cuando queremos tratar un grupo de objetos y objetos aislados de forma similar.
- Los objetos se representan en estructura de árbol para representar jerarquías completas por secciones.
- Crea una clase que contiene grupos de sus propios objetos y provee una manera de modificar cada grupo de objetos.
- El patrón permite que se remueva complejidad de la aplicación, enmascarando los diferentes tipos de objetos

Composite – Diagrama general

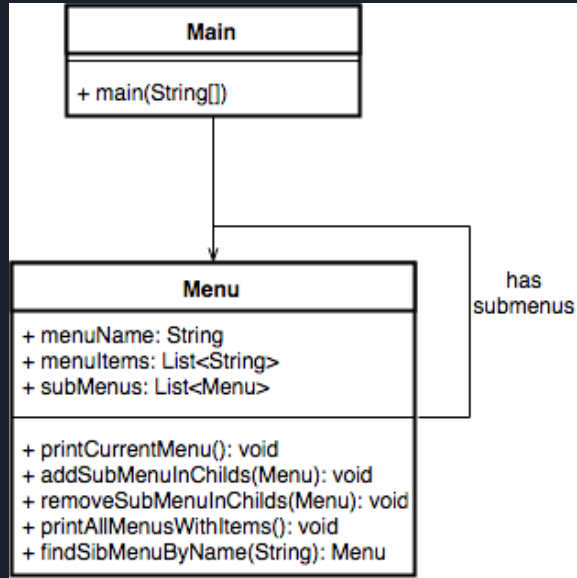




Composite - Problema

- Después de su unión, Pancake House y Dinner Home han decidido agregar nuevos platillos y menús para niños y postres.
- Cabe la posibilidad de que se tengan que agregar menús y elementos dentro de los menús en el futuro.

Composite – Diagrama aplicado





Referencias

Tutorials Point (s/f) “*Iterator Pattern*”. Recuperado el 12 de octubre de 2017 de

https://www.tutorialspoint.com/design_pattern/iterator_pattern.htm

Tutorials Point (s/f) “*Composite Pattern*”. Recuperado el 13 de octubre de 2017 de

https://www.tutorialspoint.com/design_pattern/composite_pattern.htm

Gamma, E., Helm, R., Johnson, R., & John Vlissides. (1995). *Elements of Reusable Object-Oriented Software*. Indianapolis, Indiana: Addison-Wiley.