



UNIVERSIDAD DE BUENOS AIRES  
FACULTAD DE INGENIERÍA  
Año 2015 - 2<sup>do</sup> Cuatrimestre

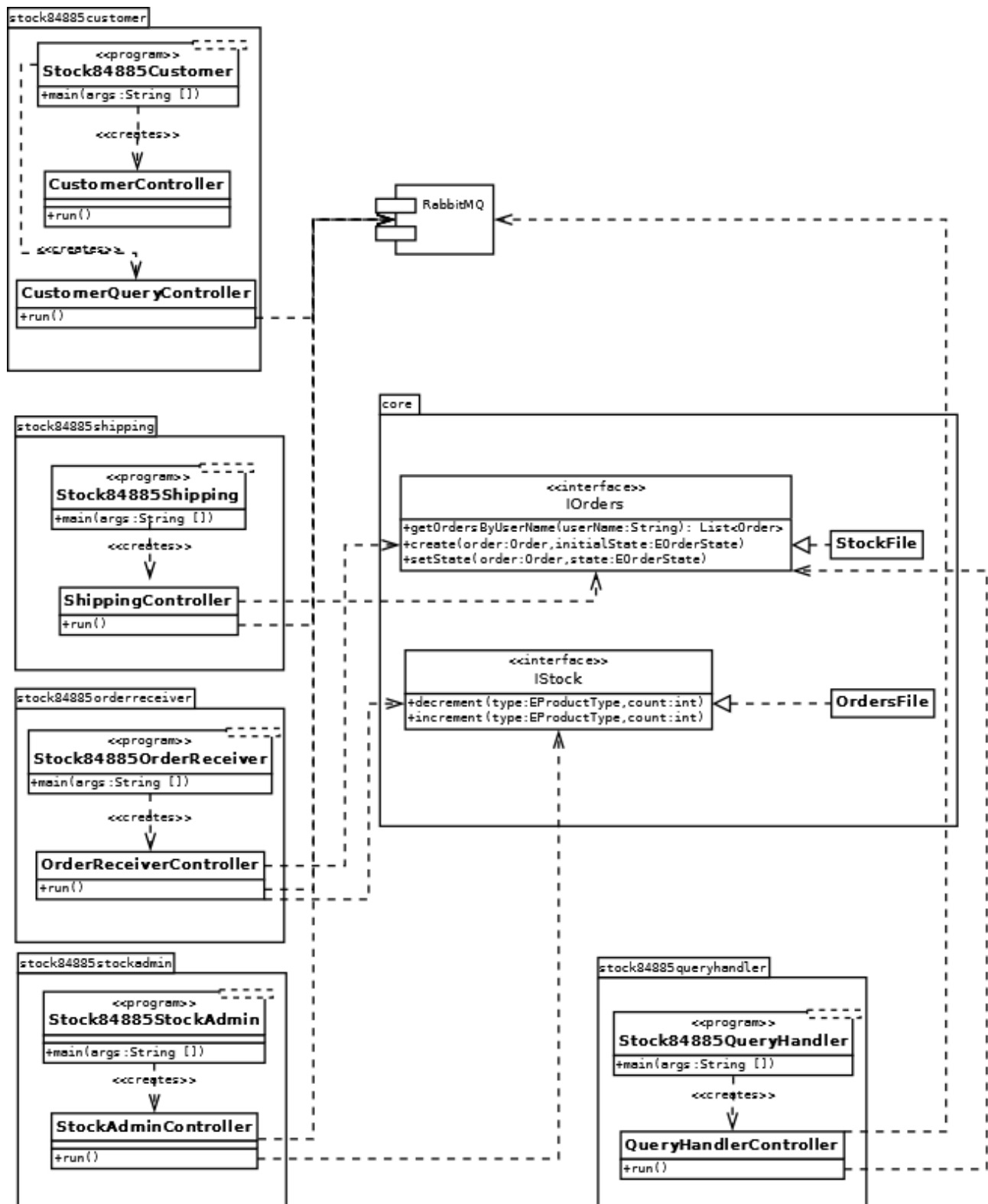
## **TALLER DE PROGRAMACIÓN III (75.61)**

TRABAJO PRÁCTICO N.º 1  
TEMA: Concurrencia  
FECHA: 10/09/2015

Autor  
Darío Eduardo Ramos

Padrón  
84885

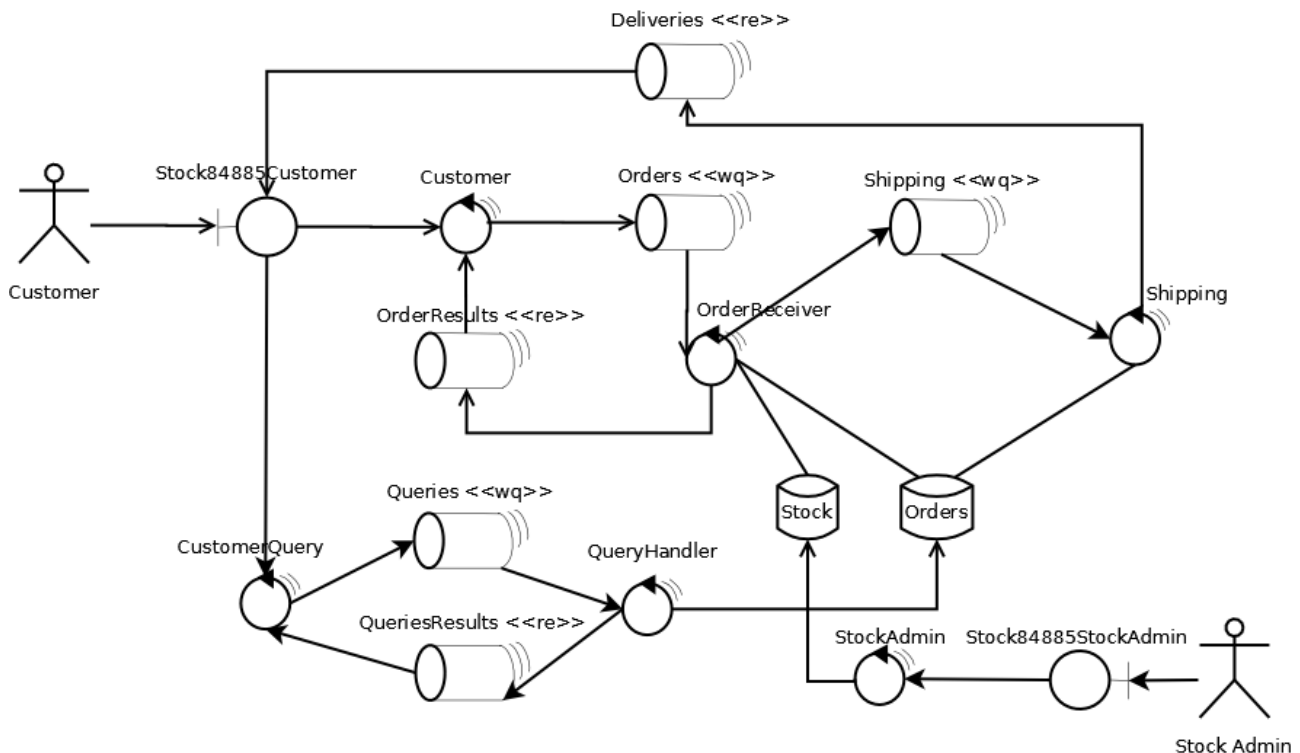
## Vista lógica: Diagrama de paquetes



Todos los programas usan RabbitMQ, y clases del Core que no se muestran por simplicidad (como `Ilogger` y `Configuration`). Se usan interfaces para `IOrders` e `IStock` a fin de poder reemplazar fácilmente los archivos por bases de datos.

Vista de procesos: Diagrama de secuencia

## Vista de despliegue: Diagrama de robustez



Estereotipos:

- re: Routing Exchange. Usa routing para enviar el mensaje a un receptor específico. Por ejemplo, Shipping lo usa para enviarle el pedido al cliente que lo pidió y a ningún otro.
- wq: Workers Queue. El productor pone el mensaje en el exchange, y lo toma cualquier consumidor (se usa el esquema por defecto, o sea Round Robin). Por ejemplo, lo usa el Customer para enviar su pedido y que lo tome cualquier OrderReceiver.

Sobre el diagrama:

Un cliente (**Customer**) envía un pedido (order) por la cola **Orders**, de donde la toma algún **OrderReceiver**. El mismo crea una instancia de Order y la guarda en **Orders** con estado inicial recibido. Luego, revisa el **Stock**, garantizando exclusión mutua, y le responde al **Customer** si el pedido fue rechazado porque no había stock, o aprobado. En ambos casos, se actualiza el estado del pedido.

Si el pedido es aprobado, es enviado a la cola **Shipping**, de donde la toma algún **ShippingController** y la envía por el exchange **Deliveries** al cliente que la pidió. Shipping sabe quién es ese cliente porque dicha información llegó en el pedido que recibió.

Por otro lado, un Customer puede consultar el estado de sus pedidos. Lo hace a través de **CustomerQuery**, quien envía el nombre del cliente por **Queries**, de donde lo toma **QueryHandler**. Éste revisa Orders, y manda la lista de pedidos con su estado al cliente vía **QueriesResults**.

Por último, el actor StockAdmin, a través del **controller homónimo**, puede consultar el stock de cualquier producto en cualquier momento, con exclusión mutua garantizada.