



Universidad de Buenos Aires
Facultad De Ingeniería
Año 2015 - Primer Cuatrimestre

75.59 - Técnicas de Programación Concurrente I

FECHA: 21/05/2015

PRIMER PROYECTO: ConcuDelivery

Apellido y Nombre	Padrón	Correo
Aceto, Ezequiel	84316	ezequiel.aceto@gmail.com
Opromolla, Giovanni	87761	giopromolla@gmail.com

[Objetivo](#)

[Análisis del problema](#)

[Casos de uso](#)

[Resolución](#)

[Tareas](#)

[Resolución](#)

[Interfaz de Línea de Comando](#)

[Diagramas](#)

[Diagramas de clases](#)

[Diagrama de transición de estados](#)

[Diagrama de un proceso genérico](#)

[Diagrama del proceso Recepcionista](#)

[Diagrama del proceso Cocinera](#)

[Diagrama general del funcionamiento de Procesos](#)

Objetivo

El objetivo de esta aplicación, ConcuDelivery, es simular el funcionamiento de una pizzería. La misma cuenta con recepcionistas de pedidos, cocineras, hornos, cadetas y una supervisora. En todo momento se puede observar el saldo de la caja, el estado de los pedidos, y las tareas que realiza cada actor.

Análisis del problema

Se plantea el desarrollo de una aplicación que permita simular una pizzeria que recibe pedidos debe prepararlos y luego entregarlos a sus clientes. La pizzeria cuenta con 5 entidades/actores que cumplen tareas distintas:

1. **Recepcionistas:** Son las encargadas de recibir los pedidos de los clientes y comunicarselos a las cocineras.
2. **Cocineras:** son las responsables de preparar la masa y meterlas en el horno.
3. **Hornos:** Su tarea es cocinar las distintas pizzas que van ingresando las cocineras, teniendo una demora en tiempo fijo.
4. **Cadetas:** Son las encargadas de entregar los pedidos listos en el domicilio de los clientes y cobrarlos para luego depositar el pago en la caja
5. **Supervisora:** Es la encargada de controlar periódicamente los ingresos en la caja.

Para que la simulación sea posible, cada una de las entidades debe poder comunicarse de forma concurrente con las demás entidades relacionadas. Para esto se utilizaron las distintas herramientas vistas en la materia 75.59 - *Técnicas de Programación Concurrente de la FIUBA*:

- A. Procesos
- B. Memoria Compartida
- C. Locks y Semáforos
- D. FIFO

Cada entidad es un Proceso, y se comunica con las demás por medio de un buffer FIFO, sincronizado con semáforos y haciendo uso de memoria compartida. Por lo tanto las Recepcionistas toman los pedidos y los cargan en el buffer de *pedidos a cocinar*, luego las Cocineras preparan el pedido y lo cargan en el buffer de *pedidos a hornear* para luego pasar al buffer de *pedidos listos*, donde las Cadetas irán tomando pedidos para entregar y cobrar su valor correspondiente. Por otra parte la Supervisora estará monitoreando periódicamente la recaudación de la caja.

Casos de uso

[especificación de los casos de uso de la aplicación]

Caso de Uso: Tomar Pedido
Descripción: Las recepcionistas van ingresando los pedidos para que sean cocinados.
Actores participantes: Recepcionista, Cocinera.
Pre-condiciones: -
Flujo Normal:
1- La simulación genera los pedidos.
2- La Recepcionista ingresa pedido si la cantidad de pedidos tomados es menor que dos veces la cantidad de cocineras disponibles.
3- Notifica a las Cocineras que hay un pedido nuevo.
4- Fin del caso de Uso
Post-condiciones: Recepcionista libre para ingresar nuevo pedido.

Caso de Uso: Preparar Pizza
Descripción: Las Cocineras preparan la masa y cocinan la pizza.
Actores participantes: Cocinera, Horno.
Pre-condiciones: Debe haber al menos un pedido tomado.
Flujo Normal:
1- La Cocinera lee el pedido tomado.
2- La Cocinera prepara la masa.
3- La Cocinera espera un horno libre.

4- La Cocinera ingresa la pizza al Horno.
5- Una vez terminada la cocción en el Horno, la pizza queda lista para ser entregada.
6- Fin del caso de Uso
Post-condiciones: Cocinera libre para cocinar nuevo pedido.

Caso de Uso: Entregar Pedido
Descripción: Las Cadetas reparten los pedidos que ya estan listos.
Actores participantes: Cadeta.
Pre-condiciones: Debe haber al menos un pedido listo.
Flujo Normal:
1- La Cadeta lee los pedidos listos.
2- La Cadeta toma el pedido listo para entregar.
3- La Cadeta entrega el pedido en el domicilio.
4- La Cadeta cobra el pedido.
5- La Cadeta deposita el cobro en la caja.
6- Fin del caso de Uso
Post-condiciones: Cadeta disponible para entregar otro pedido.

Caso de Uso: Supervisar Caja
Descripción: La Supervisora controla la recaudación en la caja temporalmente.
Actores participantes: Supervisora
Pre-condiciones: -

Flujo Normal:
1- La Supervisora solicita leer la recaudación de la caja.
2- La Supervisora lee la recaudación.
2- La Supervisora publica la recaudación
3- Fin del caso de Uso
Post-condiciones: Supervisora espera un tiempo para repetir tarea.

Resolución

Tareas

A continuación se listan las tareas a realizar para completar el desarrollo del proyecto:

1. Dividir el proyecto en procesos. El objetivo es lograr que la simulación esté conformada por un conjunto de procesos que sean lo más sencillos posible.
2. Una vez obtenida la división en procesos, establecer un esquema de comunicación entre ellos teniendo en cuenta los requerimientos de la aplicación. ¿Qué procesos se comunican entre sí? ¿Qué datos necesitan compartir para poder trabajar?
3. Tratar de mapear la comunicación entre los procesos a los problemas conocidos de concurrencia.
4. Determinar los mecanismos de concurrencia a utilizar para cada una de las comunicaciones entre procesos que fueron detectadas en el ítem 2. No se requiere la utilización de algún mecanismo específico, la elección en cada caso queda a cargo del grupo y debe estar debidamente justificada.
5. Realizar la codificación de la aplicación. El código fuente debe estar documentado.

Resolución

A continuación se detalla cómo se abordó cada tarea y la solución propuesta:

1. El proyecto se dividió en procesos como se detalló en la sección *Análisis del Problema* (Recepcionistas, Cocineras, Hornos, Cadetas y Supervisora) . La cantidad de entidades es un parámetro configurable en la ejecución de la simulación.
2. Los procesos que se comunican entre sí son:
 - a. Las Recepcionistas toman los pedidos para que las Cocineras los preparen. Comparten un buffer FIFO para los pedidos tomados.
 - b. Las Cocineras preparan los pedidos y los meten en los Hornos para cocinarlos. Comparten un buffer FIFO para los pedidos a hornear.
 - c. Los Hornos, una vez cumplido el tiempo de cocción, dejan listos los pedidos para ser entregados por las Cadetas. Comparten un buffer FIFO para los pedidos listos.
 - d. Las Cadetas, entregan y cobran los pedidos, guardando lo recaudado en la Caja.
 - e. La Supervisora consulta la Caja periódicamente para contabilizar las ganancias.

Interfaz de Línea de Comando

La ejecución de ConcuDelivery se puede realizar sin parámetros adicionales lo cual deriva en una ejecución con los siguientes parámetros como default:

- Recepcionistas 10
- Cadetas 2
- Cocineras 10
- Hornos 5
- Cant. de pedidos a simular 50
- Info. de Debug Apagada (Default)

Estos parámetros pueden ser modificados con los siguientes argumentos

- Recepcionista
 - -r (--recepcionista) <número positivo de recepcionistas>
- Cadetas
 - -a (--cadetas) <número positivo de cadetas>
- Cocineras
 - -c (--cocineras) <número positivo de cocineras>
- Hornos
 - -o (--hornos) <número positivo de hornos>
- Número de simulaciones
 - -s (--simulaciones) <número positivo de simulaciones a realizar>
- Habilitar información de Debug
 - -d (--debug)

Se puede consultar la ayuda del programa con los parámetros "--help" o "-h".

También se puede consultar la versión del programa con "--version" o "-v".

Diagramas

Diagramas de clases

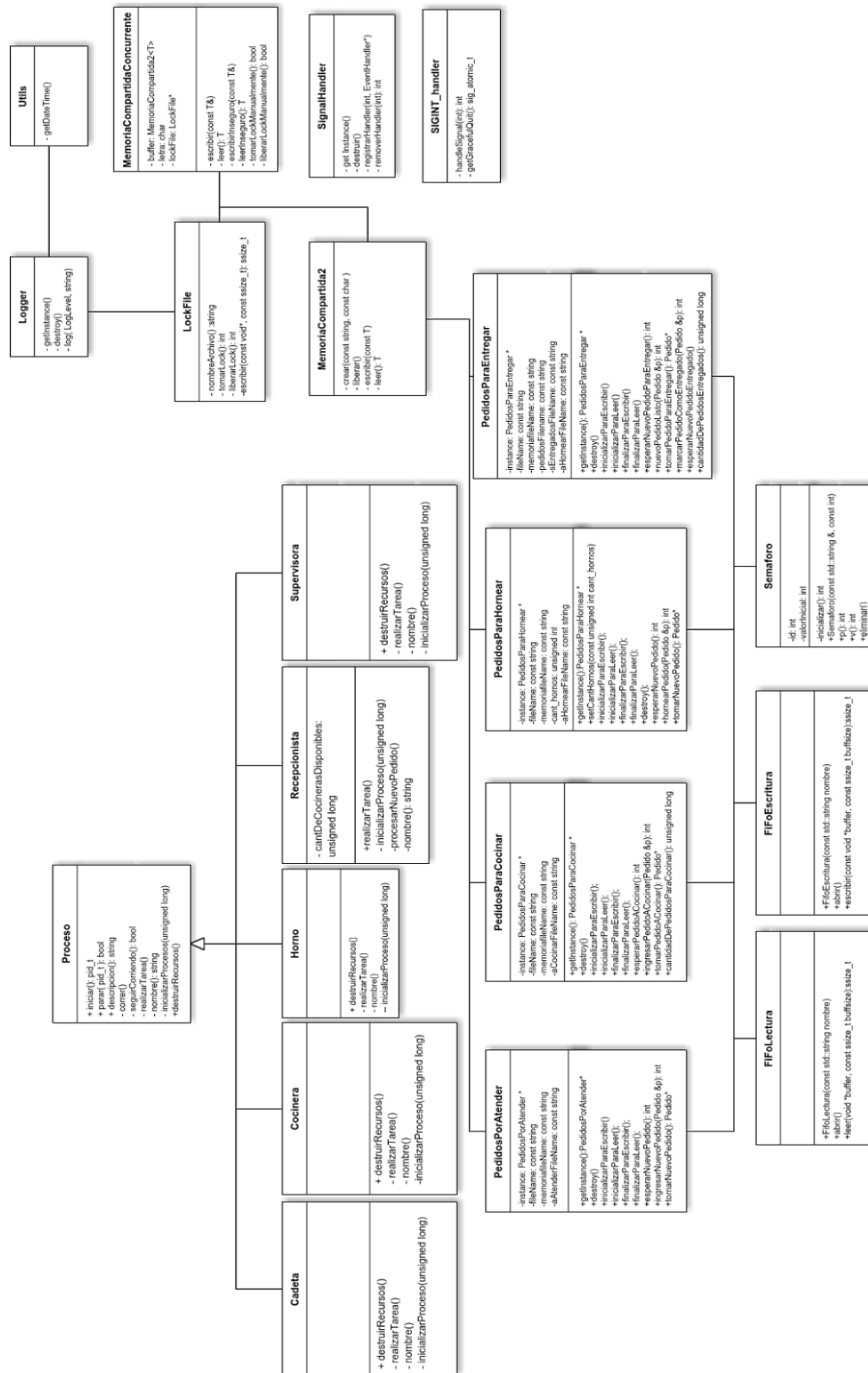


Diagrama de transición de estados

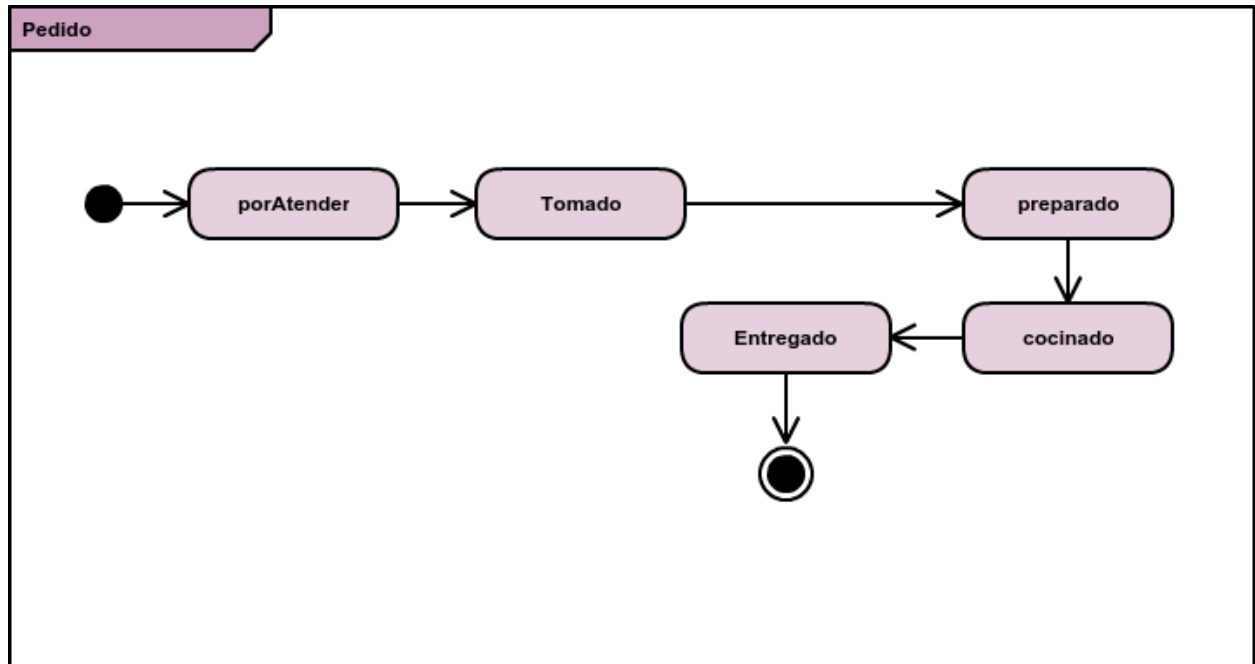
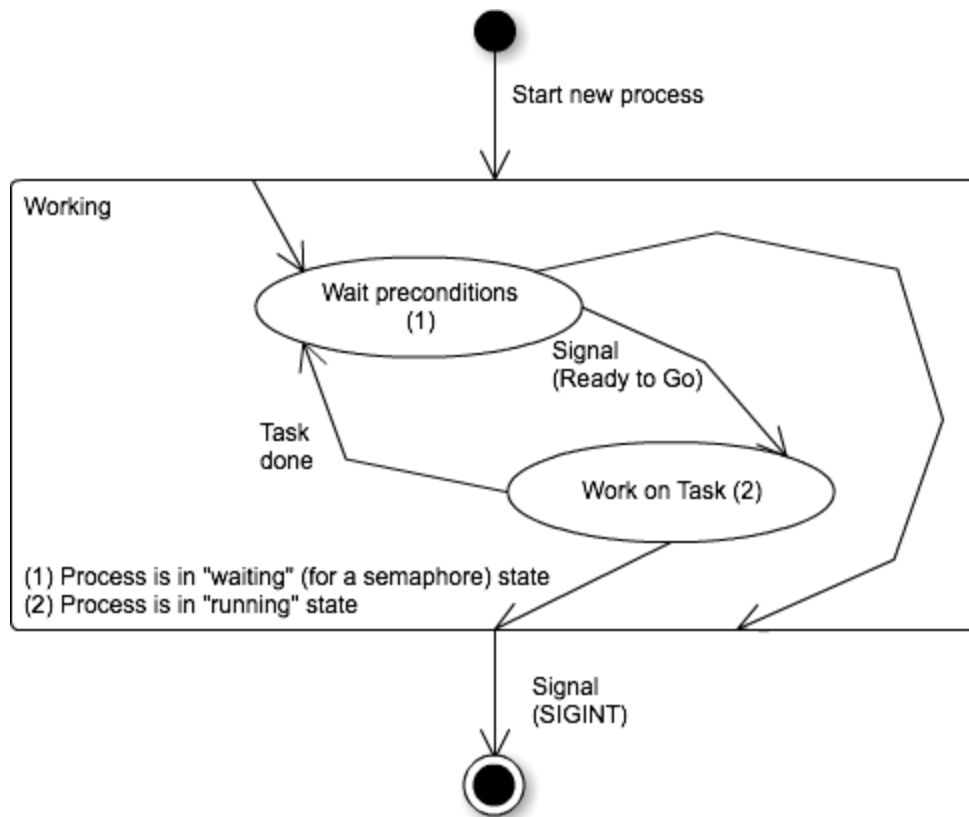


Diagrama de un proceso genérico

Este diagrama representa de forma genérica los estados por los que pasa un proceso. Como se mencionó anteriormente, todos nuestros procesos heredan de una clase Proceso. La misma brinda la estructura básica para cada uno de los procesos particulares que implementamos.

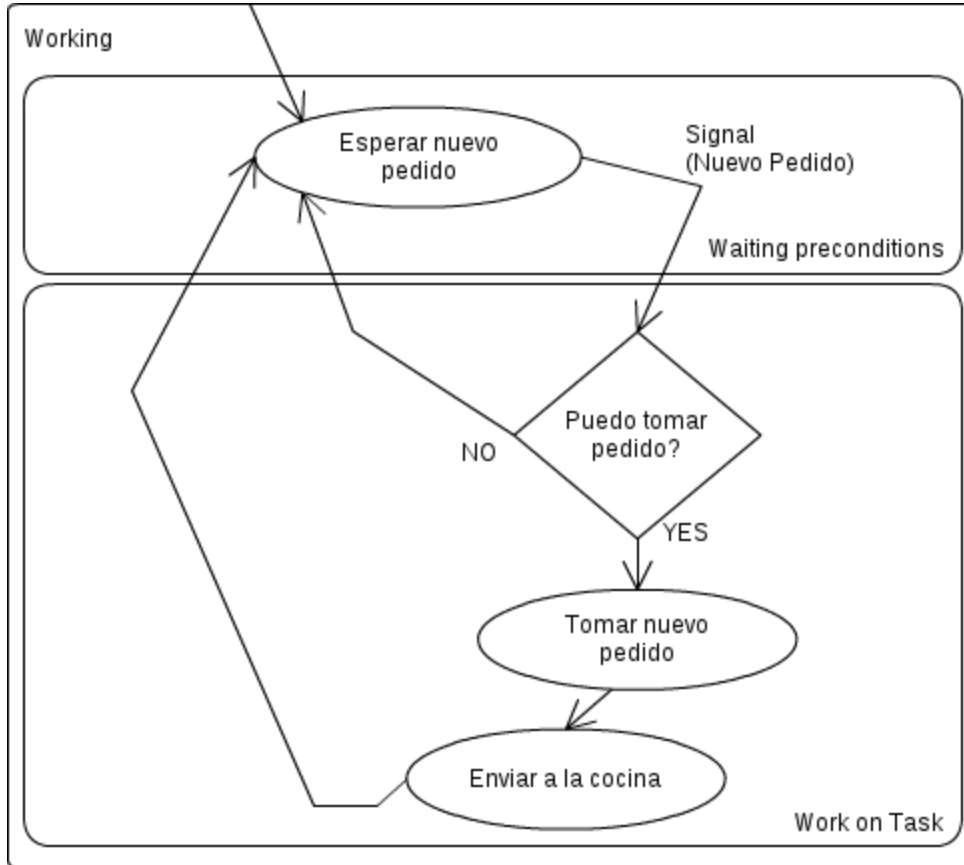


Cuando inicia el proceso está en el estado "Working" (trabajando). Este macro-estado está compuesto por dos estados, "Waiting preconditions" y "Work on Task". La idea detrás de este funcionamiento es que cada proceso tiene dos estados preponderantes. Uno en el cual espera que se cumpla cierta precondición para trabajar, y otro en el cual está trabajando en la tarea definida.

Si en algún momento el proceso recibe una señal de corte, el mismo pasa a un estado detenido, y solo puede ser relanzado si se inicia un nuevo proceso.

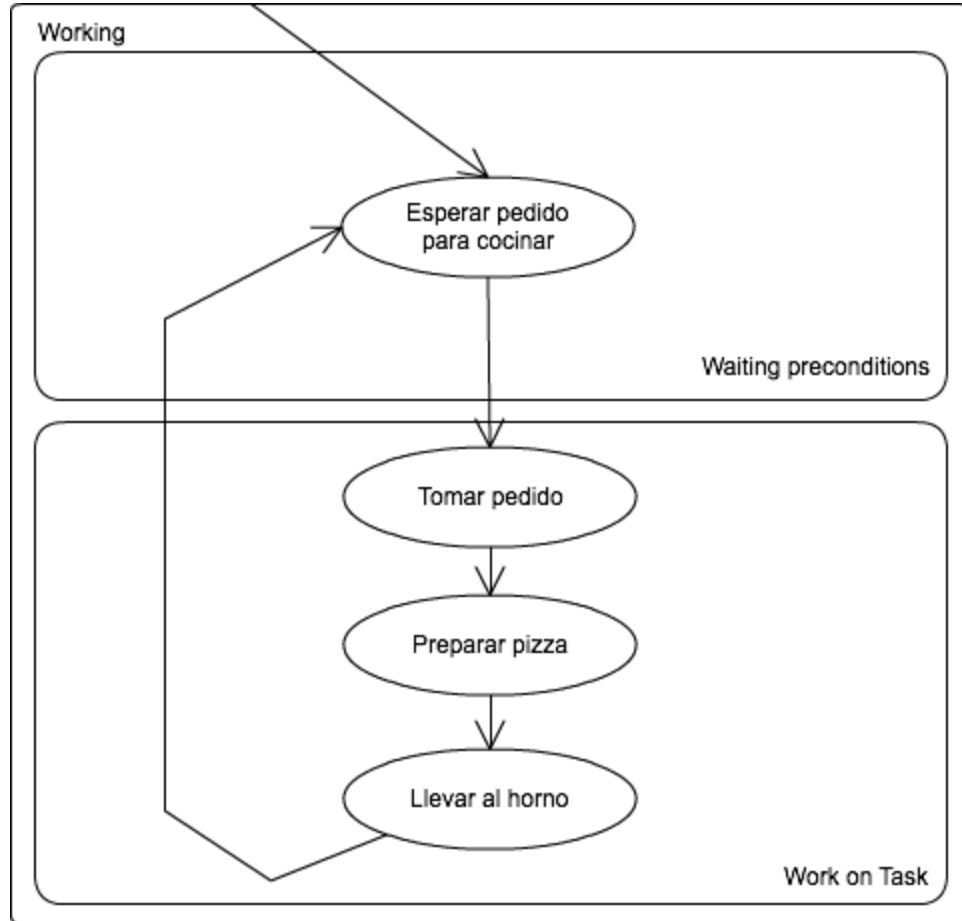
Los siguientes diagramas de estado hacen foco en lo que sucede en los estados "Wait preconditions" y "Work on Task"

Diagrama del proceso Recepcionista



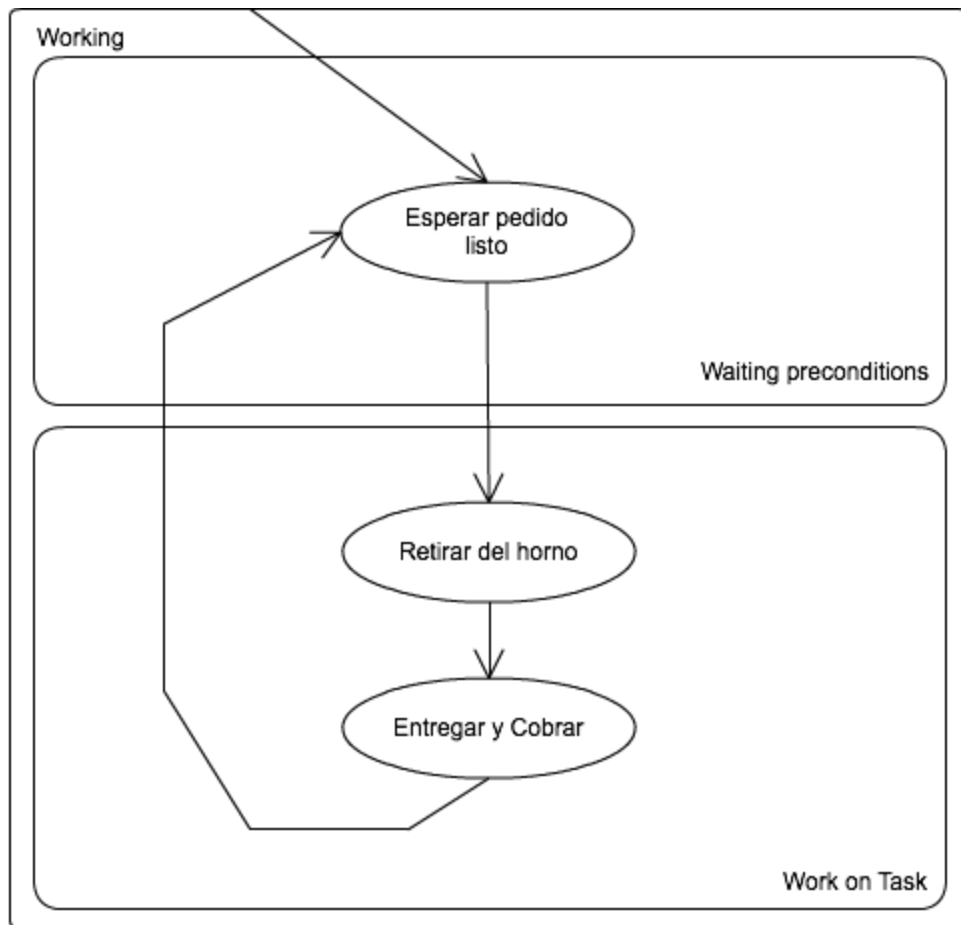
Las precondiciones hacen que el proceso espere una señal de “Nuevo Pedido” para pasar del estado “En espera” a “Corriendo”. Una vez que el proceso se encuentra corriendo, se ejecuta la lógica de negocios de la recepcionista. Si el pedido puede ser tomado, es tomado por la recepcionista y luego de ser procesado se envía a la cocina.

Diagrama del proceso Cocinera



“Esperar pedido para cocinar” espera a que la bandeja de pedidos donde las Recepcionistas dejan los pedidos, tenga al menos un pedido. Luego una Cocinera toma ese pedido, lo prepara y lo lleva al horno.

Diagrama del proceso Cadeta



En “Esperar pedido lista” se espera a que los pedidos estén cocinados. Luego una cadeta los toma y procede a entregarlo y cobrarlo.

Diagrama general del funcionamiento de Procesos

