



Universidad de Valladolid



**ESCUELA DE INGENIERÍAS
INDUSTRIALES**

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería de Organización Industrial

**Aplicación multiusuario con NetLogo.
Modelo de competencia en precios con
producto diferenciado.**

Autor:

Badás López, Guillermo

Tutor:

**Izquierdo Millán, Segismundo
Samuel**

**Departamento de Organización de
Empresas y CIM**

Valladolid, Octubre 2017

Resumen

El propósito de este Trabajo Fin de Grado es el de acercar, tanto a estudiantes como a docentes, una nueva metodología de aprendizaje de la economía y de modelos económicos, en especial, del modelo de competencia en precios con producto diferenciado. Esta nueva metodología esta soportada por la plataforma NetLogo, la cual es una plataforma o entorno de programación multiagente que posibilita el desarrollo de acontecimientos, simulaciones o fenómenos de diversa índole.

El objeto de este Trabajo Fin de Grado, por tanto, será el de exponer la nueva aplicación creada a partir del software NetLogo sobre el modelo de competencia en precios con producto diferenciado y cómo gracias a esta aplicación, se obtienen unos mejores resultados educacionales que a través de los ya conocidos métodos tradicionales.

Palabras clave

NetLogo, Aplicación, Multiusuario, Competencia, Precio

Abstract

The purpose of this End-of-Grade Work is to bring both students and teachers a new methodology of learning about economics and economic models, especially the Model of competition in prices with differentiated product. This new methodology is supported by the NetLogo platform, which is a platform or multi-agent programming environment that enables the development of events, simulations or various kinds of phenomena.

The purpose of this End-of-Grade Work will, therefore, be to expose the new application created from the NetLogo software on the Model of competition in prices with differentiated product and, as thanks to this application, better educational results are obtained than through the well-known traditional methods.

Keywords

NetLogo, Application, Multi-agent, Competition, Price



Índice

1. INTRODUCCIÓN.....	7
2. CAPÍTULO 2. FUNDAMENTOS TEÓRICOS.....	11
2.1 Economía.....	13
2.2 La microeconomía.....	14
2.2.1 Disyuntivas presentes.....	15
2.2.2 Precios y mercados.....	16
2.3 Mercados.....	16
2.3.1 Mercados competitivos y no competitivos.....	16
2.3.2 El precio de mercado.....	17
2.3.3 Dimensiones de un mercado.....	17
2.4 La oferta y la demanda.....	18
2.4.1 La oferta.....	18
2.4.2 La demanda.....	19
2.4.3 Oferta y demanda conjunta.....	21
2.5 Monopolios.....	22
2.5.1 Características de la competencia en un monopolio.....	24
2.6 Oligopolios.....	24
2.6.1 El equilibrio en un oligopolio.....	25
2.6.2 Modelo de Cournot.....	26
2.6.3 Modelo de Stackelberg.....	27
2.7 Competencia basada en precios.....	28
2.7.1 Modelo de Bertrand.....	28
2.7.2 Competencia en precios con producto diferenciado.....	30
3. CAPÍTULO 3. MANUAL DEL USUARIO.....	37
3.1 Primeros conceptos.....	39
3.2 NetLogo.....	39
3.2.1 Aspectos generales de NetLogo.....	41
3.3 HubNet.....	44
3.3.1 Aspectos de HubNet.....	45
3.4 Preparación de la experiencia.....	45
3.5 Manual para el usuario del “Modelo de competencia en precios con producto diferenciado”.....	49



Aplicación multiusuario con NetLogo

3.5.1 Descripción de la actividad.....	49
3.5.2 Interfaz del servidor.....	51
3.5.3 Interfaz del usuario.....	58
3.5.4 Pasos a seguir para el correcto desarrollo de una sesión.....	64
4. CAPÍTULO 4. MANUAL DEL PROGRAMADOR.....	69
4.1 Introducción.....	71
4.2 Manual para el programador de la actividad “Modelo de competencia en precios con producto diferenciado”	72
5. CAPÍTULO 5. PRESUPUESTO.....	91
5.1 Introducción.....	93
5.2 Perfiles, competencias y funciones del personal.....	94
5.2.1 Director del proyecto.....	94
5.2.2 Informático.....	94
5.2.3 Administrativo.....	94
5.3 Etapas del proyecto.....	95
5.3.1 Definición del proyecto.....	95
5.3.2 Estimación de recursos y tiempo.....	95
5.3.3 Desarrollo de la parte lógica.....	96
5.3.4 Desarrollo de las interfaces.....	96
5.3.5 Puesta en marcha.....	96
5.3.6 Documentación.....	97
5.4 Costes del proyecto.....	97
5.4.1 Costo de personal.....	98
5.4.2 Costo de amortización.....	101
5.4.3 Costo de materiales.....	103
5.4.4 Costes generales.....	103
5.5 Cálculo del coste total.....	104
6. CAPÍTULO 6. CONCLUSIONES.....	105
6.1 Conclusiones.....	107
ANEXO I. LICENCIA Y COPYRIGHT DE LA APLICACIÓN.....	113
BIBLIOGRAFÍA.....	115



1. Introducción

Los desarrollos y avances tecnológicos conseguidos por el hombre en las últimas décadas, han facilitado el desempeño de todo tipo de tareas en diversas áreas o materias. Todos estos avances posibilitan el desarrollo de las actividades de un modo más rápido, sencillo y con un menor coste en tiempo, dinero y esfuerzo, obviando la enorme eficiencia conseguida en relación a los procesos tradicionales.

Los avances tecnológicos han supuesto un giro en nuestras vidas, cambiando nuestras rutinas, hábitos y formas de vida. Estos, aparte de todo lo que posibilitan, han cambiado el modo en el que nos desenvolvemos en el día a día, alterando el modo en el que nos comunicamos, trabajamos, investigamos, descubrimos y aprendemos.

El área de la educación, comprendiendo esta tanto la tarea docente como la de aprendizaje, se ha visto enormemente beneficiada por todos estos avances tecnológicos posibilitando y facilitando dichas actividades. Gracias a la gran cantidad de medios tecnológicos disponibles en la actualidad, la tarea educativa cuenta con un amplio abanico de posibilidades y fuentes, los cuales se antojan de gran ayuda en dicha labor.

En la actualidad, los estudiantes cuentan con gran cantidad de medios y fuentes para que su labor didáctica se torne mucho más asequible y con una mayor profundidad y calado en cuanto a la adquisición de conocimientos se refiere. En este sentido, gozan del aprovechamiento de las innumerables fuentes de información con las que cuentan, y la rapidez del acceso a toda ellas, pudiéndolas contrastar de forma casi instantánea. Por tanto, en este aspecto este grupo se ha visto enormemente beneficiado.

En la vertiente del docente, estos también han visto cómo sus métodos lectivos se han visto acrecentados tanto cualitativamente como cuantitativamente, pasando de las clases magistrales como prácticamente la única modalidad de enseñanza, a ver como esta se podrían apoyar en material interactivo y en presentaciones audiovisuales, las cuales enriquecen de una manera notoria la tarea instructiva.

Si nos centramos en el ámbito de la ingeniería, esta, además de las clases magistrales en las cuales se instruye en la parte teórica de las asignaturas, necesita, por el modo de trabajo el cual desempeña, de los medios necesarios para formar en la parte práctica, la cual se antoja de mayor importancia y cuya formación en este sentido retorna una mayor adquisición de conocimientos. Esta afirmación, se ha visto reforzada en los últimos años por el actual modelo educativo imperante en la universidad española que obliga por ley a que todas



Aplicación multiusuario con NetLogo

1. Introducción

las asignaturas tengan una componente práctica importante, dejando en un segundo lugar, aun siendo todavía de mayor importancia, al examen teórico o examen escrito como tal.

En base a lo anterior y centrándonos un poco más en la especialidad en la cual estamos inmersos, Organización Industrial, surge la necesidad de cubrir el área práctica de esta especialidad de algún modo.

Al tratarse de dicha especialidad, es necesario apuntar a medios que faciliten la introducción de conceptos un tanto económicos, sobrevolando y nunca dejando de lado aspectos fundamentales del grado como son las ideas de la organización, la dirección, la logística, etc. Por ello, surge la necesidad de cubrir, de un modo lo más beneficioso para el acto lectivo, toda esa formación práctica que los alumnos de la especialidad de Organización Industrial necesitan.

Así pues, aunando el hecho de que en la actualidad la tecnología es la realidad a través de la cual se guía la sociedad, además de una herramienta que da un excelente soporte. Y del hecho por el cual se cubre una necesidad como es la de aportar medios e instrumentos prácticos para que los alumnos disfruten aprendiendo, en adición a que el cuerpo docente tenga un medio de calidad con el cual dichos alumnos aprendan del mejor modo, nace la motivación de la realización de este trabajo fin de grado.

De modo que con este trabajo de fin de grado se propone encontrar una manera de mejorar la docencia y enseñanza aprovechando las posibilidades tecnológicas disponibles para así remediar los inconvenientes que suelen surgir a los alumnos en materia económica, hecho que se torna de gran importancia en el presente grado. Y es calificado como inconveniente, dado que durante los primeros cursos del grado, apenas se introducen conceptos económicos en contraposición de los técnicos, por lo que así se cubre dicha necesidad.

En esencia, el objeto de este trabajo sería el de desarrollar e implementar una herramienta o instrumento docente que dé soporte a la instrucción de la asignatura de Entorno económico y estrategia y simule algunos de los problemas y modelos típicos que en esta asignatura se imparten.

La finalidad u objetivo, por tanto, será que los alumnos adquieran de una manera más comprensible y dinámica, los principios de las teorías económicas, entendiendo y asimilando el motivo por el cual se desarrollan y suceden diversos fenómenos económicos. Todo ello a través del desarrollo de actividades y mediante el juego con la herramienta obtenida mediante el programa NetLogo.



Aplicación multiusuario con NetLogo

1. Introducción

La aplicación desarrollada simularía un modelo de competencia en precios con producto diferenciado. Un modelo el cual se asemeja mucho a la realidad ya que las empresas que compiten en un determinado mercado lo hacen compitiendo en precios.

Por consiguiente, a través de la mencionada aplicación se pretende conseguir un acercamiento por parte de los alumnos a un modelo económico real y que este muestre las implicaciones teóricas que lleva consigo. Todo ello a través de ejercicios fácilmente comprensibles para que los alumnos puedan jugar, probar y experimentar a través de la herramienta.

Para llevar a cabo la realización del presente trabajo fin de grado, articularé este en varios pasos a seguir, los cuales son necesarios para alcanzar el objetivo fijado. Estos son:

- Análisis previo acerca de los modelos que mejor se ajusten a la realidad.
- Estudio y selección del mejor entorno de programación que consiga satisfacer las necesidades requeridas.
- Diseño conceptual de la aplicación informática.
- Realización de la aplicación a través de la plataforma de programación.
- Comprobación y verificación del programa.
- Realización de la memoria pertinente.

Así mismo, el presente documento se encuentra estructurado en seis capítulos o secciones bien diferenciados, a través de los cuales se desarrollan de manera independiente todos los conceptos que envuelven e incluyen tanto la selección del modelo teórico, como la selección de la plataforma de programación como los manuales pertinentes de la propia aplicación, así como un presupuesto de la realización del mismo. Como apoyo a toda la documentación referenciada en el presente documento, se adjunta la bibliografía empleada para la realización del mismo.

Por tanto, los capítulos en los cuales está estructurado el trabajo son los siguientes.

En esta primera parte se presenta e introduce el tema a desarrollar. En este se describe cual ha sido la motivación por la cual ha surgido este trabajo y que necesidades surge a cubrir. Se define de forma clara y concisa el objeto de estudio y los objetivos que se quieren conseguir con la finalización de este. Además se estructura el trabajo y se establecen las pautas a seguir para la realización del mismo.

En un segundo capítulo describiremos de una forma más teórica, las teorías económicas y los modelos económicos existentes. En adición escogeremos y



Aplicación multiusuario con NetLogo

1. Introducción

nos centraremos en el modelo a simular, que será el modelo de competencia en precios con producto diferenciado. Elegiremos y definiremos que formulación teórica se asemeja más a la realidad y la justificación de dicha elección.

Una vez documentado todo lo anterior, en el capítulo tercero nos centraremos en la plataforma de programación. En este capítulo expondremos el manual del usuario, es decir, explicaremos cual es el procedimiento a seguir por el usuario para la puesta en marcha del programa y para que la comprensión del mismo, así como para que la experiencia de uso sea satisfactoria. Expondremos y explicaremos las funciones de cada uno de los aspectos que conforman el programa, así como alguna sugerencia de uso.

En un cuarto capítulo nos centraremos en el manual del programador. En este capítulo atenderemos y expondremos como ha sido la confección de la aplicación y los pasos seguidos para desarrollarla. Esta sección se adjunta para que se pueda extender el conocimiento de cómo la misma ha sido desarrollada.

En el capítulo número cinco, se expondrá un estudio económico del coste de realizar dicho trabajo, aunando la realización del programa o aplicación, junto con la realización de la memoria y documentación correspondiente, a parte de todos los aspectos relativos al presente trabajo. Así pues, se presenta un presupuesto del mismo.

Por último, en el sexto capítulo se introducirán las conclusiones obtenidas de la realización del presente trabajo fin de grado.

Como ya he mencionado antes, al final se adjuntará un anexo relativo a la licencia establecida en la aplicación, y la bibliografía utilizada como soporte para la realización del proyecto.

Capítulo 2.

Fundamentos teóricos

En este segundo capítulo, nos centraremos en exponer y explicar de una forma general y conceptualista más que detallista, los conceptos y teorías fundamentales que rodean la economía, y en especial la microeconomía. Daremos una breve introducción sobre todos los conceptos básicos de esta, para posteriormente centrarnos en el modelo de competencia en precios con producto diferenciado, que será la pieza angular de este trabajo fin de grado, dado que todo ello girará en torno al mencionado modelo.

En un primer momento, describiremos que se entiende por teoría economía y que engloba, para posteriormente centrarnos en una de sus dos ramas, la microeconomía. De esta, expondremos elementos básicos como la oferta y la demanda, el mercado, etc. para finalmente entrar en detalle en los tipos de competencia, los equilibrios, los modelos de Cournot y Bernard y por último el ya mencionado modelo de competencia en precios con producto diferenciado, muy relacionado con el modelo de Bertrand.



2.1 Economía

Si atendemos a una definición propiamente dicha, el diccionario de la lengua española define a la economía como: “Ciencia social que estudia los métodos más eficaces para satisfacer las necesidades humanas materiales, mediante el empleo de bienes escasos”.¹

En un sentido más coloquial podríamos definir la economía como aquello que procura la satisfacción de unas necesidades por parte de una sociedad, bien sean materiales o no materiales, y que lleva a esta a la consecución de unas determinadas actividades productivas con el fin de satisfacer las mencionadas necesidades de los miembros involucrados. A través de estos procedimientos productivos, adquiriremos como resultado los bienes materiales o servicios necesarios.

Observando ambas definiciones, identificamos uno de los principales problemas que trata de resolver la economía, y es el de la satisfacción de unas necesidades principalmente ilimitadas mediante el empleo de unos recursos disponibles en cantidades limitadas. Esta es una de las principales disyuntivas que la economía, como ciencia, se enfrenta y trata de solventar.

Por tanto, la economía se encarga de gestionar unos recursos limitados, siendo conocedores de que, el empleo de dichos recursos para una finalidad, supone el encontrarse inutilizables para cualquier otra. Por ello es necesario que se definan correctamente cual son los objetivos primordiales a resolver y satisfacer, para así dar el mejor uso posible a dichos recursos.

Como cualquier otra ciencia, la economía trata de explicar los sucesos observados en la realidad. Estos sucesos se articulan y se explican a través de predicciones llamadas teorías. Estas últimas, se utilizan para dilucidar los fenómenos u observaciones estudiados a través de un conjunto de reglas y supuestos.

Así pues, las teorías económicas se utilizan con la finalidad de desarrollar predicciones. Estas, a su vez, sirven para formar modelos que permitan construir predicciones cuantitativas.

De este modo, se solventa en gran medida otro de los problemas que rodea a los fenómenos económicos, que es la incertidumbre. Ya no nos referimos únicamente a la incertidumbre que las teorías económicas traen consigo, sino a la incertidumbre que los diversos agentes o actores implicados en dichos fenómenos aportan al tratar o desempeñar un papel en el sistema. Por ejemplo,

¹ (Diccionario de la lengua española, 2017)



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

si dos empresas compiten en un determinado mercado, una de ellas estará fuertemente condicionada por las decisiones que tome la otra y viceversa. Por tanto, esto aunque no se puede solventar del todo, a través de los modelos descritos se puede obtener una aproximación de los resultados obtenidos en función de las decisiones del rival.

La teoría económica, como ya hemos introducido, se centra en el comportamiento y las interacciones de los agentes económicos y cómo funcionan las economías. De acuerdo con este enfoque, distinguiremos dos grandes ramas, la microeconomía y la macroeconomía.

La microeconomía se centra en la conducta de los individuos o unidades económicas. Véanse las empresas, los consumidores, los inversores, los trabajadores, etc.

Por el contrario, la macroeconomía se centra en las cantidades económicas agregadas o las economías nacionales, como es la tasa o nivel de crecimiento de la producción de un país, las tasas de desempleo, los tipos de interés, etc.

Sin embargo, en la actualidad, la separación o ramificación de la economía en estas dos vertientes no es tan marcada como lo era, dado que la macroeconomía necesita también del análisis de los mercados, área de la microeconomía. Por ello, en este sentido podríamos llegar a clasificar a la macroeconomía, en parte, como una extensión de la microeconomía.

2.2 La microeconomía

Como ya hemos descrito antes, la microeconomía se centra en la conducta de los individuos o unidades económicas. Esta trata de explicar el cómo y el porqué de las decisiones económicas adoptadas por las unidades individuales. Por ejemplo, describe cómo influyen las decisiones de las compras por parte de los consumidores en las variaciones de precios y como se deciden estas.

Anteriormente introducimos una de las principales disyuntivas presentes en la economía. Nos referimos a las limitaciones existentes en cuanto a las ilimitadas necesidades frente a los no tan ilimitados, más bien todo lo contrario, recursos. Un gran porcentaje de la microeconomía se basa en tratar estos límites.



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

2.2.1 Disyuntivas presentes

La microeconomía como rama de la economía que estudia la conducta de las unidades económicas, trata las disyuntivas que confronta y resuelve estas demostrando cual es la mejor manera de hacerlas frente.

El concepto de resolver las disyuntivas presentes de un mejor modo es una de las tareas fundamentales y primordiales de la microeconomía.

Se analizarán estas en base a tres agentes.

2.2.1.1 Consumidores

La limitación presente en estos agentes es la referida a la renta que pueden utilizar para adquirir bienes o servicios o inversiones futuras.

Por ello se estudia esta limitación en los consumidores dado que lo que buscan estos es en función de su renta disponible, sabiendo que es limitada, cuál sería la opción que maximizara sus beneficios, sean estos referidos a sus gustos, preferencias, deseos, etc.

2.2.1.2 Trabajadores

En este segundo grupo, se presentan tres posibles restricciones o disyuntivas. En un primer momento, los trabajadores han de decidir si trabajar inmediatamente o estudiar y formarse, posponiendo el trabajo para otro momento más futuro. En este sentido han de barajar el coste de oportunidad incurrido con cada una de las opciones, sin embargo deberán valorar el hecho de que una mayor formación conlleva una mayor retribución y viceversa.

En segundo lugar, deberán decidir en qué tipo de empresa desean trabajar, dado que hacerlo en las grandes empresas disminuye el riesgo incurrido además de que también lo hace el hecho de escalar peldaños en la organización. Por el contrario si deciden ser autónomos o trabajar por cuenta propia aumenta el riesgo pero también lo hace el posible ascenso o posición dentro de esta.

Por último en un tercer momento deberán escoger en la medida en que se les permita el número de horas a trabajar en una jornada laboral o en una semana laboral.

2.2.1.3 Empresas

En cuanto a este tercer y último grupo, las limitaciones a las que tiene que hacer frente se refieren al problema de decidir que producto o tipos de productos fabricarán en detrimento de otra gama o variedad. También se



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

encontrarán limitados en cuanto a los recursos financieros utilizados y a la producción o fabricación de bienes. Esta última limitación se puede paliar ampliando el tamaño de la fábrica, el número de empleados, etc. pero todo esto supone un coste y unas opciones las cuales hay que barajar y sopesar.

2.2.2 Precios y mercados

Otro de los asuntos primordiales que fundamentan y estudia la microeconomía es la labor del precio, tema que trataremos detalladamente durante el presente trabajo. En vista de las disyuntivas anteriores, todas ellas están referidas a los precios asumidos por los agentes.

La microeconomía por tanto se centra y estudia, en adición a lo anterior, la fijación y elección de los precios. Esta elección depende del tipo de planificación, bien sea centralizada o por parte del Estado o bien a través de economías de mercado en las cuales los precios son el producto de la interacción entre varios agentes.

2.3 Mercados

Los mercados propiamente dichos los conforman los compradores o consumidores, que son aquellos que adquieren los bienes y servicios, por ejemplo, y los vendedores o productores, que son aquellos que venden sus productos o servicios. Ambos dos conforman los mercados.

Los mercados, así pues, mediante la interrelación de consumidores y vendedores establecen los precios o valores de los bienes o servicios.

2.3.1 Mercados competitivos y no competitivos

Los mercados se pueden caracterizar en función de la competencia dentro de este. Por ello podemos clasificarlos en competitivos y no competitivos.

Los mercados perfectamente competitivos cuentan con numerosos consumidores y productores, por lo tanto, al encontrarnos con multitud de agentes y competidores dentro de este, no son lo suficientemente capaces como para influir y variar los precios de manera notoria. Esto ocurre debido a



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

que cada consumidor o vendedor únicamente posee o adquiere una pequeña parte del total existente por lo que las variaciones no afectan de manera global.

En cambio, en los mercados no competitivos o imperfectamente competitivos las decisiones de estos agentes influyen notablemente en el precio. Encontramos varios casos de mercados imperfectamente competitivos como son el caso de los monopolios, los oligopolios o la competencia monopolista.

2.3.2 El precio de mercado

En un mercado perfectamente competitivo existe un único precio a través del cual los consumidores y vendedores intercambian los bienes y servicios, denominado precio de mercado.

Sin embargo en los mercados que no existe competencia perfecta, el precio puede variar en función de las decisiones de cada empresa, a pesar de estar vendiendo el mismo producto. Esto puede deberse a que quieran atraer nuevos clientes con precios atractivos o en cambio que oferten sus productos a altos precios beneficiándose de una posición de marca o una clientela leal a esta.

2.3.3 Dimensiones de un mercado

Cuando hablamos de un mercado, conviene establecer y definir las dimensiones de este porque en función de eso, podremos conocer de un mejor modo el número de oferentes y demandantes que existirán y al que nos estamos dirigiendo.

Con la definición de las dimensiones del mercado nos referimos a los límites geográficos para saber con exactitud a la población que estamos llegando y la variedad de productos que incluiremos en nuestro mercado.

Definiendo y estableciendo estas dos pautas, las empresas, por su parte, conocerán y sabrán cuales son y serán los competidores en un determinado mercado además de la variedad de productos en los que compiten. Además, en función de su situación geográfica en la que compiten unos productos, estos tendrán más éxito, por sus características, en unos lugares que en otros.

2.4 La oferta y la demanda

La oferta y la demanda es uno de los temas centrales de la microeconomía. Mediante estos elementos se puede entender de un mejor modo como afectan los precios a las cantidades, ya sean las ofertadas o las demandadas y viceversa. O la relación entre oferta y demanda y la toma de decisiones en base a estas.

2.4.1 La oferta

La oferta expone las cantidades que, a un determinado precio y manteniendo los demás actores relacionados con estas constantes, los productores están dispuestos a vender.

En la siguiente figura, Figura 2.1, representamos la curva S de la oferta.

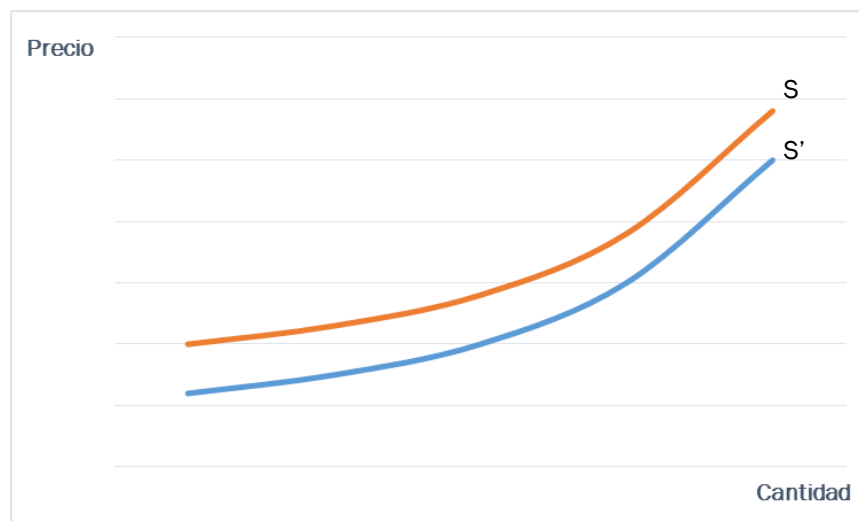


Figura 2.1. Curva de la oferta.

Como podemos apreciar, el precio al que se quiere vender viene situado en el eje de ordenadas, mientras que la cantidad lo está en el de abscisas.

Por tanto, vemos en la curva S de la Figura 2.1 como la cantidad ofertada por la empresa varía en función del precio al que esta lo quiera vender. La representación de la curva S' expone el hecho por el cual los costes de producción son menores, y por lo tanto el precio al que pueden ofertar la misma cantidad de producto es menor o en otras palabras, al mismo precio ofertan una mayor cantidad. Así pues, en este caso se observa como la curva se ha desplazado hacia la derecha. De este modo podemos afirmar como la cantidad ofertada no viene únicamente determinada por el precio sino, por ejemplo por



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

los costes de producción, los cuales pueden estar constituidos por los salarios de los trabajadores, las materias primas consumidas, etc.

En base a lo anterior podríamos por tanto definir la curva de la oferta mediante la ecuación:

$$Q_o = Q_o (P)$$

O dicho de otro modo, la cantidad en función del precio.

Observamos como la pendiente de esta gráfica es positiva, por ello podemos afirmar que cuanto mayor es el precio al cual quieren vender las empresas, mayor es la cantidad que estas producen e introducen en el mercado. En la misma línea podríamos afirmar que un aumento en el precio de venta de los productos puede ser un atractivo para nuevas empresas dispuestas a competir en este mercado y por tanto suponer una mayor cantidad de productos en el mercado.

2.4.2 La demanda

Si la oferta se refería a cuanto estaban dispuestos los productores a vender a un determinado precio, la demanda se encarga de la cantidad que los compradores están dispuestos a comprar a un determinado precio.

La Figura 2.2 muestra una curva de demanda.

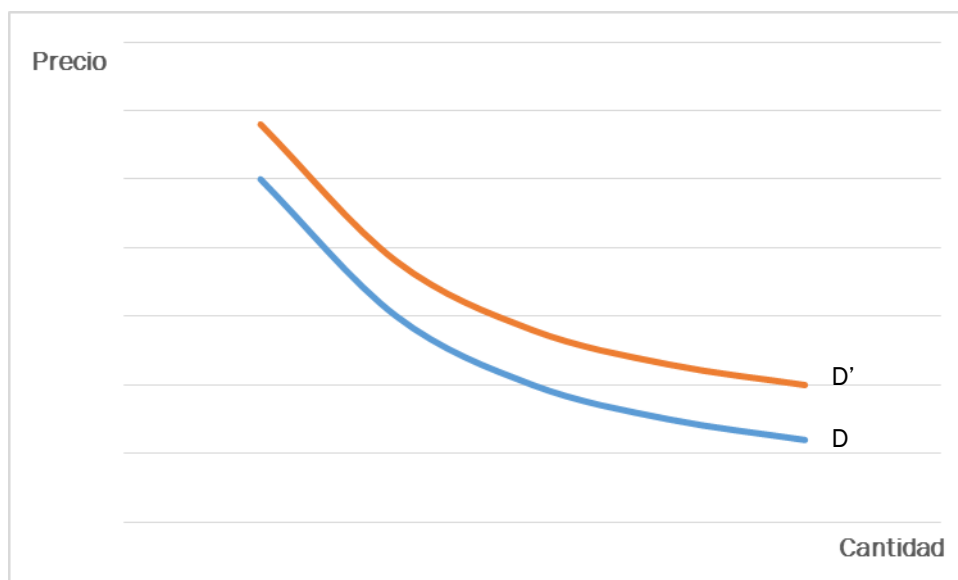


Figura 2.2. Curva de la demanda.



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

En este caso, la curva tiene una pendiente negativa. Esto evidencia el hecho por todos sabido y que todos aplicamos por el cual, los consumidores tienden a comprar mayor cantidad de un producto dado cuanto menor sea su precio. Así pues, se da el caso por el que se incita en cierto modo, a comprar una mayor cantidad de producto ya que, si la siguiente unidad de un determinado producto es más barata puede inducir al consumidor a que la adquiera. Además, un precio menor puede suponer que un determinado público que antes no podía adquirir dicho producto ahora si se lo pueda permitir.

La siguiente ecuación representa la relación existente para definir la curva de demanda:

$$Qd = Qd (P)$$

Como se puede comprobar en la ecuación, la cantidad demandada depende del precio al que este se compra. Sin embargo, como ya vimos en la definición de la oferta, la cantidad no depende únicamente del precio, sino que existen otros factores influyentes en ella. En esta ocasión, la cantidad demandada por el comprador dependerá de la renta disponible que este disponga para poder adquirir productos o servicios. Este es un punto muy a tener en cuenta dado que si el consumidor posee una mayor renta, esto repercutirá en que puede adquirir una mayor cantidad de productos.

Este hecho se puede ver reflejado en la Figura 2.2. En esta, se aprecia como la cantidad adquirida por un consumidor depende del precio al que se compre, curva D. Sin embargo, como ya hemos explicado, la cantidad demandada no depende únicamente del precio, sino que también lo hace en función de la renta disponible. Por ello, al aumenta la renta del consumidor, la curva se ve desplazada hacia la derecha, lo que se ejemplifica a través de la curva D'.

2.4.2.1 Bienes sustitutivos y complementarios

La cantidad demanda que presenta un producto, está condicionada en gran medida por el precio que este ostenta en el mercado. Así pues, cuanto mayor sea el precio de venta de ese producto, menor será la tasa de compra obtenida. En estos casos es donde aparecen los productos sustitutivos.

Los productos sustitutivos son aquellos productos que hacen las veces del producto origen debido a que sus características o los resultados obtenidos se asemejan en gran porcentaje al del original. Estos adquieren una mayor importancia cuando los productos originarios toman un elevado valor o precio.

En cambio, los bienes complementarios se caracterizan por, mientras que un tipo de producto alcanza un elevado valor, su complementario disminuye la cantidad demandada.

En este sentido, antes habíamos definido la alternativa de influencia de la renta en el desplazamiento de la curva de la demanda hacia la derecha. En vista de lo descrito en este punto podríamos defender la posición de que un aumento en un producto sustitutivo o un descenso del valor de un bien complementario desempeñarían el mismo efecto en la curva que un aumento de la renta disponible por parte del consumidor.

2.4.3 Oferta y demanda conjunta

En este apartado trataremos la superposición de ambas en un mismo gráfico y los efectos visibles que estos aportan. Ahora en el gráfico se analizarán simultáneamente el precio al cual lo productores están dispuestos a vender y los consumidores a comprar. La Figura 2.3 representa esta relación.

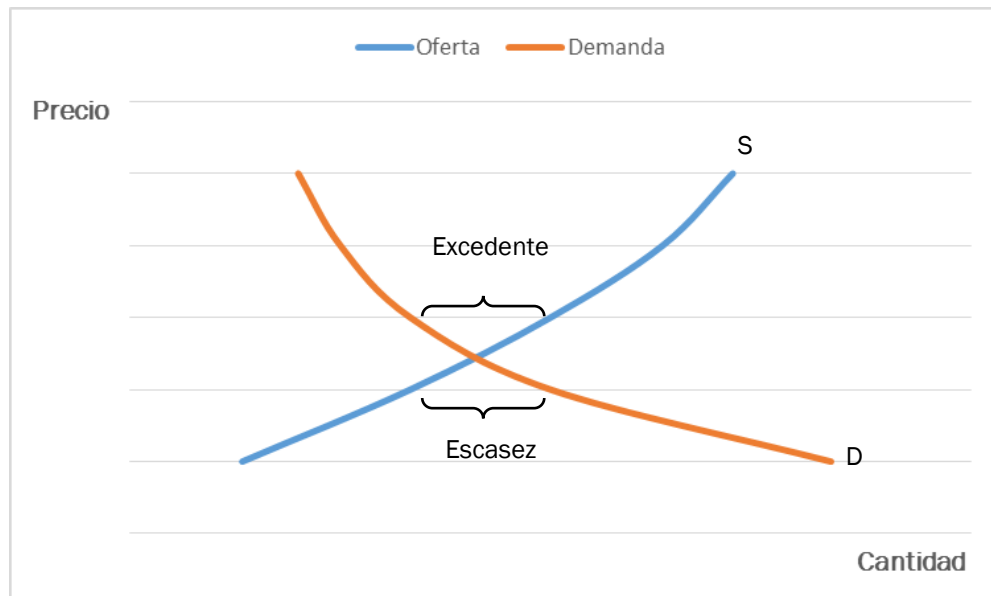


Figura 2.3. Oferta y demanda conjunta.

Atendiendo a la anterior figura, el punto donde las dos curvas se cortan resulta ser el punto de equilibrio. En este punto el precio coincide tanto para la cantidad ofertada como para la cantidad demandada. En este punto, por tanto, se sitúa el precio de equilibrio.

La tendencia natural del precio en un libre mercado es la de variar en busca de un equilibrio el cual alcanzará en algún momento debido a que esa es su finalidad. Esto ocurre por el denominado mecanismo del mercado. Una vez alcanzado el precio de equilibrio se mantendrá en esa posición dado que al no existir ni excedentes ni escasez, no existirán fuerzas que impulsen la variación



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

del precio. Aunque la tendencia es hacia el equilibrio, pueden existir mercados en los cuales esto no pase aún.

Las fuerzas que motivan el equilibrio del precio serán debidas al excedente de la cantidad o a la escasez de esta. Así bien definiremos ambas.

El excedente se forma cuando la cantidad ofertada al mercado supera la cantidad demandada o absorbida por este. Para evitar este fenómeno, los vendedores tratarán de disminuir el precio. Así, al disminuir el precio, la cantidad ofertada disminuirá y la demandada aumentará, por lo que se alcanzará el precio de equilibrio buscado.

En cambio la escasez supone que se demanda más cantidad de producto del que se oferta, así pues, los consumidores no podrán adquirir todos los productos que requieran. En esta situación, el precio trataría de aumentar. El resultado de esto sería que los productores aumentarían el precio en pos de la demanda existente y trataría de aumentar la producción para satisfacer las necesidades de todos los consumidores, y estos por su parte estarían dispuestos a pagar un precio mayor. De modo que volvemos de nuevo a la situación de equilibrio descrita.

Para que el modelo de oferta y demanda tenga validez y funcione con el nivel esperado, el mercado debe ser en gran medida competitivo, en otras palabras, que los agentes y actores involucrados en este no tenga gran poder en las elecciones de los precios y su influencia individual en este sea insignificativa. De cualquier otro modo, por ejemplo, en un mercado monopolístico, el precio no variará en función de la cantidad vendida. A continuación veremos esto con mayor detenimiento.

2.5 Monopolios

Comencemos este capítulo recordando que es un mercado en competencia perfecta, concepto que nos servirá para comparar con la nueva definición que expondremos en esta parte.

Un mercado en competencia perfecta es aquel que se caracteriza por albergar un gran número tanto de compradores o consumidores como de vendedores o productores, y por lo tanto, un individuo o conjunto de individuos de uno u otro grupo tiene el suficiente poder como para variar el precio en su favor. La fijación del precio, por tanto, vendrá determinado por el mercado y por sus efectos de oferta y demanda.



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

Por el contrario, un monopolio es aquel que se produce cuando en un mercado únicamente existe un vendedor, pero multitud de compradores. La empresa monopolista tiene como característica fundamental que la curva de demanda a la que tiene que hacer frente es justamente la curva de demanda de su empresa. De este modo, mediante esta curva relaciona el precio que cobra por la cantidad que está vendiendo. Además, al coincidir la curva de demanda del mercado con la propia de su empresa, se puede beneficiar de su posición monopolista y fijar el precio que más beneficios le reporte.

El monopolista se puede beneficiar de su posición y vender una menor cantidad de un producto a un precio mayor, cosa que en un mercado perfectamente competitivo no pasaría. En base a esto, existen en los países leyes antimonopolio, para intentar controlar este fenómeno de algún modo, y que las empresas no se beneficien de los consumidores.

La posición de privilegio que tienen las empresas monopolistas se puede acuñar como el poder del monopolio. Este hace referencia al hecho por el cual la empresa monopolista cobra un precio muy superior al coste marginal del producto. Así pues, un monopolista maximizará su beneficio conociendo perfectamente cuál es su coste de producción.

Sin embargo, el caso de monopolio propiamente dicho o puro se da con relativa poca frecuencia, ya que es muy difícil que solamente, en la actualidad, haya un único productor o vendedor de un producto. Se da con mayor frecuencia en el mercado un monopolio competitivo. Este, se asemeja al mercado perfectamente competitivo en que hay gran variedad de empresas compitiendo, y que no presenta barreras en la entrada de nuevos competidores. Por el contrario, se diferencia en que aunque a priori el producto, por características, funcionalidades, etc. es semejante, este no es el mismo ya que la diferencia radica en la marca, la calidad o la percepción que el cliente tiene sobre este.

Así pues, podríamos afirmar que una empresa tiene, en cierto modo una posición monopolista frente a otras si cuenta con la lealtad de un público, y por ello puede permitirse fluctuar mínimamente al alza sus productos. No obstante ha de ser cauta ya que si el precio impuesto por esta en su producto es mucho mayor que el de la competencia, los clientes leales a dicho producto pueden no estar dispuestos a pagar una elevada diferencia en el precio.

De modo que, entre empresas monopolistas, también existe competencia entre ellas por hacerse con el mercado y con una posición ventajosa frente a sus rivales.

En el siguiente punto describiremos las características de esta competencia.



2.5.1 Características de la competencia en un monopolio

Las características fundamentales de esta competencia monopolista serían:

- Los productos ofertados por las empresas monopolistas son productos sustitutivos unos de otros e incluso se podría decir que cuentan con gran facilidad para ser sustituidos, sin embargo no se consideran productos sustitutivos perfectos.
- Existe una gran disponibilidad tanto en la entrada de nuevas empresas que desean competir en un mercado, como aquellas que desean abandonarlo al verse obligadas por los malos resultados obtenidos. Es decir, no existen apenas barreras ni de entrada ni de salida al mercado.

2.6 Oligopolios

Comencemos este apartado definiendo que se entiende por oligopolio. Un oligopolio es un tipo de variedad de las estructuras existentes de mercado el cual se caracteriza porque solo compiten unas cuantas empresas entre sí, es decir, hay un pequeño número de oferentes, y además presenta grandes dificultades a la entrada de nuevos competidores. En cuanto al tipo de producto producido, puede ser distinto o puede ser del mismo tipo.

Para que las empresas oligopolísticas obtengan los mayores beneficios posibles, han de trazar una estrategia de actuación la cual vendrá determinada por las actuaciones, las decisiones y el análisis de las estrategias de sus rivales. Las estrategias de estas pueden definirse conjuntamente, lo cual repercute negativamente en el consumidor, y lo cual está prohibido por la ley, dado que las empresas han de competir entre ellas y no actuar conjuntamente para subir los precios a sus productos y enriquecerse ambas a costa de los consumidores. O pueden competir agresivamente incluso hasta llegando a incurrir en pérdidas para, de este modo, infligir el mayor daño posible a su rival y que esta se vea obligada a abandonar el mercado.

Las empresas que pertenecen a un modelo oligopolístico presentan unos altos beneficios debido a que muy pocas empresas ofertan todo un tipo de productos. Además, esto se ve reforzado por la dificultad que presenta entrar en un mercado oligopolístico debido a las grandes barreras existentes.

La gerencia de una empresa de estas características, aunque puede ofrecer grandes beneficios como ya hemos comentado, resulta bastante tedioso y



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

complicado de llevar, ya que los beneficios propios no dependen únicamente de tus decisiones, sino que las decisiones o actuaciones de tus competidores influyen notablemente en tus propios resultados. En adición, como comentamos con anterioridad, una actuación o decisión propia a priori puede parecer muy beneficiosa, sin embargo, si tus rivales toman las decisiones pertinentes, pueden incluso expulsarte del mercado porque finalmente tu decisión, tras a priori parecer la mejor, a posteriori resulta reportarte beneficios negativos y por tanto obligarte, en cierto modo, a abandonar el mercado. De modo que las decisiones tomadas han de tener en cuenta las posibles decisiones o actuaciones de tus contrincantes.

2.6.1 El equilibrio en un oligopolio

En todo mercado es interesante y de gran utilidad conocer cuáles serán las condiciones de equilibrio, es decir la cantidad y el precio.

Mientras en los mercados perfectamente competitivos, monopolios, etc. una vez encontrado y fijado el precio de equilibrio o la cantidad, en cierto modo, se podrían desentender de las elecciones tomadas por tus rivales dado que tu elección es una buena elección, en los oligopolios esto no puede ser posible.

Como ya hemos descrito, las decisiones en los oligopolios están relacionadas, por ello la fijación de un precio de equilibrio no significa que ese sea una buena opción. Lo mismo le ocurre a las rivales. Estas han de prever tus decisiones.

Ahora bien, caigamos en la cuenta de que significaba el equilibrio en, por ejemplo, el mercado de competencia perfecta. El equilibrio se calculaba ya que ese era el punto en el que el mercado en su conjunto, o mejor dicho, todas las empresas de ese mercado obtenían los mayores beneficios o resultados. En el caso del oligopolio, el equilibrio en esencia es lo mismo salvando alguna diferencia. El equilibrio de Nash describe esta particularidad.

El equilibrio de Nash es, por tanto, la conjunción de actuaciones o decisiones que cada empresa toma para obtener los mayores beneficios posibles en función de las actuaciones de sus rivales.

En función de esta afirmación se puede pensar que cada empresa tomará las decisiones que más le beneficien en función de sus competidoras. Pero para ello hay que conocer cuáles son las decisiones que tomarán las rivales o al menos prever de cuales se tratarán. Para esto, se recurre entre otras técnicas a las llamadas teorías de juegos.



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

Las teorías de juegos tratan de explicar situaciones complejas de una forma sencilla, entendible y clara. Podremos considerar como juego a todas aquellas decisiones estratégicas donde el resultado varíe en función de lo que ha elegido el individuo o en este caso jugador. Todos los juegos presentan tres elementos indispensables. Estos son: los jugadores, la estrategia a seguir y los rendimientos obtenidos.

Los jugadores son los individuos que piensan y ejecutan la decisión. Estos pueden escoger entre un abanico de posibilidades, las cuales cada una tiene su consecuencia. Las estrategias son las posibles opciones a tomar por los jugadores. Y por último los rendimientos son los resultados obtenidos mediante las distintas estrategias por los jugadores.

Los juegos pueden ser cooperativos o no cooperativos. La diferencia entre estos radica en si las estrategias tomadas por los jugadores pueden ser conjuntas y vinculantes o no. En los juegos no cooperativos, al no tomar decisiones vinculantes la incertidumbre es mucho mayor.

La estrategia óptima sería aquella que reporta el mayor beneficio posible, y esa sería la estrategia objetivo.

2.6.2 Modelo de Cournot

El modelo de Cournot es un modelo oligopólico que, basándose en que cada empresa genera un producto homogéneo y asumiendo, y considerando como fijo el nivel productivo del resto de competidoras, todas toman, simultáneamente, la decisión de la cantidad que van a producir.

Las características principales del modelo de Cournot se basan en que las empresas compiten en el mercado en cantidades, es decir, introducen en el mercado una determinadas cantidades de su producto, y en función de esta el resultado o beneficio puede ser mayor o menor, en vez de hacerlo en precios, como es a lo que estamos acostumbrados hoy en día. Y que la elección de las cantidades introducidas al mercado por cada empresa, ocurre en un tiempo simultáneo.

Sin embargo, la cantidad que debemos producir e introducir en el mercado dependerá en gran medida de las acciones que tomen nuestros contrincantes o rivales, ya que no debemos olvidar que el modelo de Cournot es un modelo oligopólico.

De modo que podemos saber cómo la función que maximiza los resultados o beneficios, y por tanto la cantidad que debemos producir, responde a una



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

función decreciente de la cantidad que nuestro rival introduzca o pensemos que introduzca en el mercado. Esta relación definida mediante una función es la llamada curva de reacción.

Si las dos empresas, a la hora de decidir qué cantidad producen, aciertan en su estimación sobre la cantidad producida por su rival, se llegaría al equilibrio de Cournot. Por ello se podría afirmar que el equilibrio de Cournot representa la cantidad que cada empresa ha de producir, en relación a la otra, para maximizar sus beneficios. Este punto, si lo representamos gráficamente, vendría a ser aquel en el cual las dos curvas de reacción se cruzan, como se puede observar en la Figura 2.4.

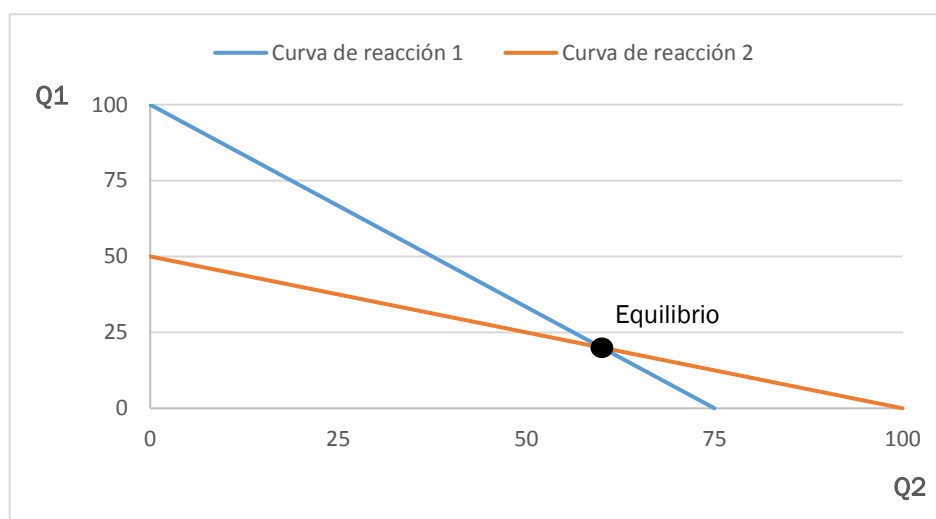


Figura 2.4. Equilibrio en curvas de reacción.

2.6.3 Modelo de Stackelberg

Como hemos visto, el beneficio obtenido por una empresa depende de la cantidad que estimemos que nuestro competidor introducirá en el mercado. Esta es una función decreciente, así pues cuanto mayor sea la cantidad producida por nuestro cliente, menor será la nuestra propia. Sin embargo, ¿y si pudiéramos elegir antes que nuestro competidor la cantidad que introducimos en el mercado?

Esta situación la trata el modelo de Stackelberg el cual se basa en que nuestra empresa fijaría la cantidad producida antes que el resto. ¿Esto qué repercusiones tiene? Pues lo cierto es que la solución obtenida gracias a este modelo, compitiendo en cantidades, beneficia enormemente a la empresa que primero decide. La empresa que elija primero, elegirá la cantidad de producción hasta el punto en el que el ingreso marginal igualándolo al coste marginal sea



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

de cero, es decir, que fabricar una unidad más tuviera un coste mayor que el ingreso obtenido por la venta de esa unidad. Por tanto el beneficio para la primera empresa es máximo.

Por el contrario, la segunda empresa tendría que actuar en función de lo que haya hecho la primera, y por tanto se tendrá que acomodar y producir en la medida que el mercado acepte toda la cantidad. Su beneficio vendrá determinado por su curva de reacción.

2.7 Competencia basada en precios

En este punto estudiaremos y haremos especial hincapié en el modelo que fundamentalmente nutre la aplicación que hemos desarrollado a través de la plataforma NetLogo, el modelo de competencia en precios con producto diferenciado. Introduzcamos todo lo que rodea a este modelo y como se llega a él.

Como ya hemos visto, las empresas en un oligopolio pueden competir en cantidades. Así pues, cada empresa fija una cantidad en base a una estimación sobre lo que harán las otras. Sin embargo, no es muy común que las empresas compitan en un mercado en función de la cantidad de producto que introduzcan en este, sino más bien en cuanto a la calidad, servicio, o principalmente en precio.

En un primer momento, expondremos y explicaremos el modelo de Bertrand, el cual trata y hace referencia a la competencia en precios con productos homogéneos. Aunque el modelo que nos atañe y nos importa para nuestra aplicación, se enfoca y trata de la competencia con diferenciación de producto, es aconsejable exponer el modelo de Bertrand dado que es la base, origen y fundamento para nuestra modelización.

2.7.1 Modelo de Bertrand

Este modelo fue expuesto en 1883 por Joseph Bertrand, prestigioso economista francés.

En un primer momento, encontramos como este modelo arroja varias semejanzas con el modelo de Cournot. Estas tratarían de que ambos modelos se encuadran para empresas que compiten con productos homogéneos y que



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

cuyas decisiones de puesta en el mercado de un producto, se hace de manera simultánea. Es decir, ninguna sabe de antemano la cantidad/precio de producto que introducirán en este. Sin embargo, la principal e importante diferencia entre estos es que, como ya vimos, Cournot se basa en la competencia en cantidades de producto puesta en el mercado, mientras que Bertrand lo hace en precios, hecho que a priori parece más real.

Por tanto, definiremos el modelo de Bertrand como un modelo oligopólico en el que cada empresa compite con el resto, en base a un mismo producto (producto homogéneo), y en el que cada una estipula un precio al que vender considerando este para sus competidoras constante y fijándolo simultáneamente.

Ejemplificando dos empresas en esta situación o atendiendo a este modelo nos podríamos preguntar qué precio habrán de marcar o que beneficio obtendrán por ello. Como solución a esa tesitura, nos tendríamos que percatar de que como el producto es homogéneo, el consumidor o cliente escogerá aquel bien que le resulte más barato, ya que en igualdad de condiciones la preferencia real es esa. Por tanto, la empresa que elija un precio menor, obtendrá toda la cuota de mercado y por consiguiente un beneficio máximo y dejará a su competidora sin ninguna venta. En otra vertiente, no encontramos la situación en la cual las dos empresas escojan el mismo precio. En esta situación, el cliente o consumidor le será indiferente adquirir uno u otro bien, debido a sus similitudes, por lo tanto la cuota de mercado se repartirá a la mitad aproximadamente.

Así pues, el punto de competitividad y objetivo por el que compiten ambas empresas será el de obtener la mayor cuota de mercado posible, y como hemos visto, esto se consigue buscando y eligiendo el menor precio posible. Sin embargo hay que caer en la cuenta de que las bajadas de precio tienen como límite el coste marginal, dado que si bajan por debajo de este, incurrirán en pérdidas. El mínimo precio al que podrán descender, vendrá determinado por el coste marginal, fijando este como límite y en el cual no obtendrán beneficio alguno.

Este punto (igualar el precio al coste marginal), será claramente un punto de equilibrio de Nash dado que, en principio, a pesar de que ambas empresas quieran subir sus precios para obtener algún beneficio, no existe incentivo alguno a subirlo, dado que si una la sube, perderá su cuota de mercado. Igualmente ocurriría en el caso de que bajará aún más sus precios, dado que si lo hace, si, obtendrá todo el mercado, pero como ya hemos dicho, perderá dinero al vender por debajo de coste. Así pues ninguna empresa tiene un incentivo real para modificar los precios en torno a ese punto.



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

Como resulta evidente, se podría pensar el por qué no sería posible fijar un precio conjunto o equilibrio de Nash en una cantidad más alta al coste marginal, la cual reporte beneficios a sus propietarios. Esta situación resultaría imposible por el hecho de que, cualquiera de las dos empresas podría bajar el precio para obtener mayor cuota de mercado y beneficio y este será un incentivo claro para hacerlo.

Por toda la anterior ejemplificación, el modelo de Bertrand ha sido puesto en duda dado que en igualdad de productos (homogéneos), el modelo de Cournot o el hecho de competir en cantidades tiene un resultado mejor para las empresas ya que pueden obtener beneficios, mientras que por Bertrand no. Además, hemos supuesto que en igualdad de precios la cuota de mercado se repartirá aproximadamente a la mitad, hecho que no está para nada fundamentado y que puede que no ocurra.

En adición a todo lo anterior, realmente, las empresas cuando compiten con bienes homogéneos lo primero que fijan o lo primero a lo que atienden es a su posible capacidad productiva, para posteriormente definir el precio al que pondrán dicho bien el mercado. De este modo se incurrirá en el ya mencionado equilibrio de Cournot y por tanto el beneficio no será nulo.

2.7.2 Competencia en precios con producto diferenciado

Tras haber realizado y expuesto de manera introductoria varios conceptos necesarios y útiles para el entendimiento de la economía, y tan adecuados e importantes para aprender y entender de un mejor modo tanto el funcionamiento como los resultados obtenidos en el juego, pasaremos a fundamentar teóricamente la piedra angular de este trabajo fin de grado, y es el modelo basado en competencia en precios con producto diferenciado.

Como ya hemos visto en los apartados anteriores, en un mercado se puede competir tanto en las cantidades introducidas en este, como en los precios a los que venden el bien. Por el momento, ya hemos descrito y expuesto la competencia surgida en un mercado cuando la variable de elección estratégica es el precio pero en el caso de que los productos o bienes introducidos fueran homogéneos.

Sin embargo, con frecuencia, los mercados oligopólicos poseen cierto nivel de diferenciación del bien o producto. Esta diferenciación no tiene por qué atender al aspecto físico o a las cualidades que poseen uno u otro. Puede ser el mismo producto que, sin embargo, se diferencie por la posición de marca, por los servicios que le acompañan o por simplemente por el sitio donde se adquiera.



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

Ahora bien, en este apartado, al tratar productos diferentes, la única diferencia ya no solo radica en el precio que el producto tenga, sino que ahora entran en juego muchos otros factores como el diseño, la calidad, las necesidades que satisface, las innovaciones que trae consigo, el rendimiento o la vida útil de este. Por ello, resulta evidente y lógico que las compañías compitan en precios más que en cantidades.

Para ejemplificar y exponer de un modo más sencillo este modelo, nos apoyaremos de dos funciones de demanda de dos empresas.

$$\text{Empresa 1: } Q_1 = 24 - 3 * P_1 + P_2$$

$$\text{Empresa 2: } Q_2 = 24 - 3 * P_2 + P_1$$

Las anteriores funciones expondrían la demanda de una de las empresas relacionada con el precio que tenga su competidora. Nótese que una mayor cantidad vendida por una empresa depende de un menor precio propio de venta, como ya hemos venido comentando. Además, se puede observar como un mayor precio de las competidoras supone una mayor cantidad de venta propia a precio de venta propio constante.

En cuanto a la elección o fijación de precios, estos irán orientados a la maximización de los beneficios. El beneficio de cada empresa dependerá de la cantidad vendida por el precio elegido menos el coste de fabricación:

$$B_1 = P_1 * Q_1 - C$$

La maximización de este dependerá en gran medida del precio que elijan los competidores. Sin embargo, la maximización del beneficio propio se consigue cuando no se obtienen beneficios adicionales por un aumento apenas notable del precio propio, independientemente del precio de los competidores. Así mismo, considerando los precios “competidores” como fijos, se obtendrían las curvas de reacción de las empresas. En el punto donde estas se crucen gráficamente, resultará el equilibrio de Nash. La representación gráfica de este suceso se muestra en la figura 2.5.

Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

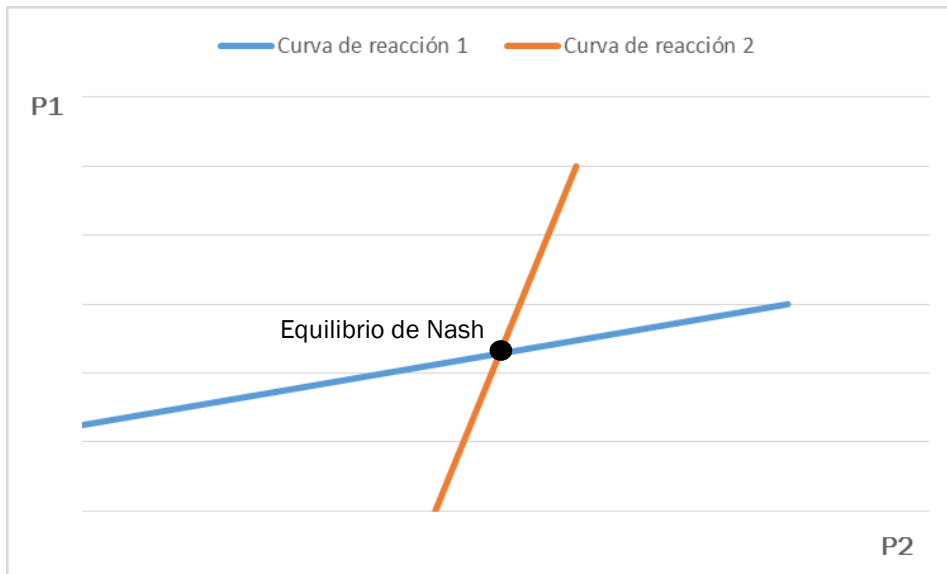


Figura 2.5. Equilibrio de Nash en curvas de reacción.

Recordemos que el punto en el cual se fija el equilibrio de Nash es aquel en el que las empresas no observan incentivos para modificar su precio, ya que en ese punto las empresas maximizan su beneficio en función de lo que hagan sus rivales.

Otro posible caso a presentar en cuanto a la elección de precios por parte de las empresas, se da cuando las dos empresas eligen un precio superior al del equilibrio de Nash. En este caso, ambas empresas obtendrán un mayor beneficio (maximización) que en la situación anterior. Esta situación se denomina colusión y viene reflejado en la Figura 2.6.

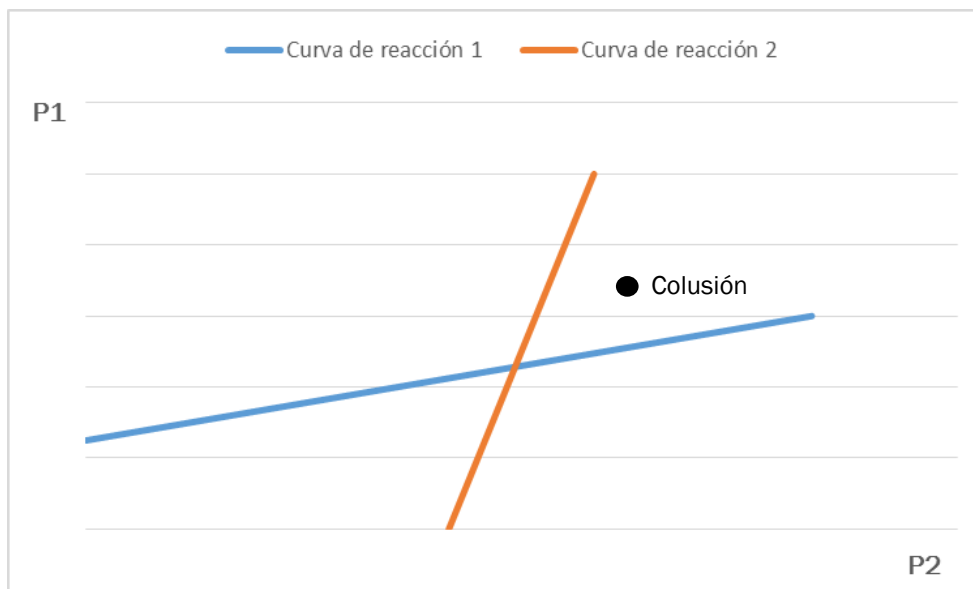


Figura 2.6. Colusión en curvas de reacción.



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

Como podemos observar en la figura anterior (Figura 2.9), los precios fijados por las empresas son mayores que en la situación de equilibrio de Nash, llevando consigo un aumento del beneficio.

En una tercera situación, podríamos comentar el hecho fruto de la fijación de precios de una empresa en un primer momento con la elección por parte de la otra en una situación posterior. En este caso, el resultado sería distinto al obtenido a través del modelo de Stackelberg, ya que en este, además de tratar como variable estratégica las cantidades de bienes o productos introducidos en el mercado, la ventaja era obtenida por aquella empresa que fijaba aquella cantidad en un primer momento. Sin embargo, si elegimos como variable estratégica el precio que tomará el bien en el mercado, la empresa que resultará beneficiada será aquella que fije el precio en un segundo momento. Esto es debido a que la segunda empresa, conocedora del precio que ha tomado la primera, puede intentar o barajar la posibilidad de fijar un precio inferior a esta y así obtener unos mayores beneficios al tener una mayor cuota de mercado.

Una de las razones por las que se ha decidido utilizar este modelo, el modelo de competencia en precios con producto diferenciado, como base de la aplicación conformada a través del programa NetLogo, es debido a que es un modelo que se aproxima mucho a la realidad existente en la actualidad, debido a que una de las variables competitivas por excelencia entre las empresas ha sido y será el precio del producto que fabrican y como optimizar este al máximo.

A lo largo de la historia, las compañías, de un mismo sector o que venden o producen un producto o bien similar, compiten entre ellas para obtener el mayor beneficio posible o la mayor cuota de mercado existente. Las empresas en la actualidad, compiten en muchos aspectos como pueden ser la calidad, la durabilidad, el aspecto físico, la manejabilidad o, como tradicionalmente ha sido, el precio.

El precio es un aspecto muy a tener en cuenta por parte de las empresas a la hora de lanzar un producto y requiere estar optimizado al máximo para así conseguir reportar beneficios a la empresa encargada de su manufactura. La búsqueda de la optimización del precio al que se vende un producto no solo viene condicionado por los costos de producción de este o la tasa de beneficios quieran obtener, sino que también está condicionado por los precios que el resto de compañías rivales o competidoras fije. Esto último es lo que estudia el presente modelo y es por ello, entre otras cosas, por lo que lo estudiamos y utilizamos como eje de la aplicación.

El precio es probablemente la variable estratégica más utilizada en la actualidad, ya que, a parte de las características del producto, el precio sigue



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

siendo algo fundamental. Es cierto que en un gran porcentaje existen determinados productos que aun teniendo un precio superior a otro, al estar amparado por una gran marca, o que la empresa productora goce de una posición de marca, tienen una mayor cuota de mercado. Hecho que ocurre también por aspectos como su calidad, o simplemente por preferencia.

Consideramos que este modelo es importante porque en gran medida el precio al que una empresa vende su producto está fuertemente condicionado por el resto de sus competidoras. Por tanto lo introducimos en la aplicación para que de este modo los alumnos gracias a la experiencia de uso, conozcan y adquieran los conocimientos que rodean dicho modelo y aprendan que no solo de las decisiones propias depende el éxito o fracaso de un producto.

Como segundo punto o aspecto por el cual consideramos este modelo para la aplicación, es porque es un modelo que los alumnos de la asignatura de Entorno Económico y Estrategia utilizan y aprenden en esta. Por tanto puede, gracias a esta aplicación, fortalecer y afianzar mejor los conocimientos adquiridos en las clases magistrales de la mencionada asignatura. Por ello, utilizaremos como funciones base para la aplicación funciones con las que ya están familiarizados debido a esta asignatura. En adición a todo lo anterior, estas funciones representan una aproximación bastante fiel a la realidad, y funcionan, en líneas generales, de un modo muy aproximado y coherente.

La función que utilizaremos será semejante a una de las que hemos utilizado en ejemplos previos para exponer este apartado 2.7.2 y es:

$$Q_i = X - C_1 * P_i + C_2 * \sum P_j$$

Esta función haría referencia a la cantidad que una empresa vendería a un mercado en función de su propio precio (en base a un coeficiente denominado como C_1) y de la suma de precios de sus competidores, también influenciado por un coeficiente C_2 . A todo esto, hay que añadir un coeficiente fijo, denominado como X en la función superior que influye en la cantidad final vendida por la empresa.

Al observar esta función, apreciamos como se trata de una función bastante intuitiva a la hora de identificar sus partes, y comprobamos como funciona de un modo muy satisfactorio. Esta función, como ya hemos comentado, reporta la cantidad introducida en el mercado por una empresa, y esta cantidad depende de dos aspectos fundamentales. El primero de ellos sería el precio propio el cual repercutirá negativamente en un mayor o menor grado cuanto mayor sea su valor. Este valor tiene ciertas limitaciones ya que si es excesivamente grande, incluso mayor que los demás coeficientes, reportará



Aplicación multiusuario con NetLogo

2. Fundamentos teóricos

una cantidad negativa, hecho que no tiene mucho sentido. No obstante, este fenómeno resulta difícil de reproducir ya que existen diversos coeficientes que influyen y limitan esta problemática. Además, se ha considerado introducir un coeficiente que influya en el precio elegido para, de este modo, obtener unos resultados mejores o más próximos a la solución buscada.

Por otro lado, la función tiene en cuenta el sumatorio de precios del resto de competidores, modificados también por un coeficiente. En este punto, existe también una limitación en cuanto al valor que tomarán. Esta limitación se da cuando el precio introducido por una empresa es enormemente alto. Atendiendo a la función, cuanto mayor sea el valor o precio que las empresas fijan, mayor será la cantidad introducida por la empresa principal. Este fenómeno es totalmente coherente dado que cuanto mayor precio alcance el producto de la competencia, mayor tasa de ventas tendrá nuestro propio producto. Sin embargo, la limitación encontrada en este punto se da cuando una empresa de las que se encasillan como competidoras, fija un precio enormemente alto. En este punto, si observamos la realidad, una empresa que fija un precio tan alto no vendería ningún producto, por lo que no se encuadraría, prácticamente, como rival. Sin embargo, este hecho en la función implica que nuestra empresa venderá una mayor cantidad de productos gracias al enorme precio que ha fijado la empresa rival, hecho que, como ya hemos mencionado, en la realidad no sucede.

En la aplicación desarrollada a través de NetLogo y la cuál es estudio y fundamento de este trabajo fin de grado, los valores de los diferentes coeficientes serán introducidos por el administrador, en nuestro caso, por el profesor de la asignatura, que determinará tanto coeficientes, como rondas y actuará como moderador o administrador de la experiencia. En base a los valores que el moderador desee fijar, los resultados obtenidos serán distintos en todos los casos, pero siempre fieles a la realidad. Los valores de los precios serán introducidos por los usuarios o alumnos. Estos introducirán el valor que consideren oportuno pudiendo inducir en algunos casos en las limitaciones o problemas mencionados. Sin embargo, para evitar o paliar en cierta medida que esto suceda, se fijaran de base en el programa un rango de valores para los cuales los resultados obtenidos sean coherentes y reales.

Capítulo 3.

Manual del usuario

En este tercer capítulo introduciremos y expondremos todos los aspectos necesarios para el correcto desarrollo de la experiencia a través del conocimiento y entendimiento de la aplicación utilizada, NetLogo, y del entorno soporte y necesario como es HubNet, el cual hace de nuestra aplicación, una herramienta de carácter participativa y colaborativa.

En este apartado daremos las pautas y conocimientos necesarios para el aprendizaje y entendimiento tanto de las plataformas utilizadas, NetLogo y HubNet, ya sea su origen y sus funciones y capacidades, así como de la exposición e instrucción de la aplicación creada.



3.1 Primeros conceptos

Como ya hemos introducido, en el presente capítulo daremos unas nociones básicas sobre las plataformas utilizadas para que, de un modo introductorio, los lectores puedan entender de una mejor forma tanto el porqué de la utilización de estas plataformas, así como el diseño y desarrollo llevado para la aplicación.

Por ello, en una primera parte de este capítulo, nos centraremos en la presentación de las herramientas, la obtención de estas, el porqué de su elección, las capacidades que presentan, etc. tanto de NetLogo como de HubNet.

En una segunda parte, y encuadrado como aspecto fundamental, expondremos el manual para el usuario propiamente dicho para, de este modo, instruir al lector en las funciones de la aplicación, su utilización y la valoración de los resultados. Este capítulo resultaría pues, como un libro de instrucciones de la aplicación desarrollada. A través de este, se describirá la aplicación, los elementos, su funcionalidad y las diferentes actividades para un correcto desarrollo y una excelente experiencia de uso.

Conviene mencionar que expondremos diferentes tipos de manuales, en concreto dos, ya que la aplicación que el usuario utilizará y con la cual desarrollara su experiencia, difiere de la aplicación que consideraremos como servidor, o aquella que el moderador o administrador utilizará.

3.2 NetLogo

Empezaremos por definir e introducir que es NetLogo.

NetLogo es una plataforma o entorno de programación multiagente el cual posibilita el desarrollo de acontecimientos, simulaciones o fenómenos de diversa índole o carácter, ya sean sociales, naturales... Se trata de un entorno que debido a su versatilidad, es utilizado por multitud de usuarios de diversa ocupación, ya sean investigadores que deseen desarrollar o simular algún fenómeno de cualquier carácter, profesores que necesiten de esta plataforma para enseñar a sus alumnos de una forma participativa los conocimientos adquiridos en las clases teóricas o incluso por los propios alumnos. Por tanto conviene destacar su sencillez a la hora de simular o incluso crear un modelo. Sin embargo, no por ello deja de ser una potente herramienta. Fue desarrollado



Aplicación multiusuario con NetLogo

3. Manual del usuario

en 1999 por el profesor y matemático Uri Wilensky. Actualmente sigue siendo desarrollada, versionándola con el paso del tiempo por el CCL.

NetLogo, es una herramienta particularmente interesante dado que permite hacer simulaciones con gran cantidad de agentes durante un determinado periodo de tiempo, e incluso para modelos especialmente complejos. A través de un código propio, se pueden ir dando pautas a los diversos agentes para que simulen la situación buscada de una forma tanto independiente, como colectiva como con el entorno. Gracias a las interacciones entre los agentes se pueden hacer estudios puntuales o relacionales simples o por el contrario se pueden sacar conclusiones de alto nivel o en el que están incluidos multitud de factores y agentes.

NetLogo se puede ejecutar de dos formas distintas.

La primera se trataría del método tradicional de descargar el programa e instalarlo en nuestro ordenador personal para poder simular, editar modelos ya existentes de la librería o incluso crear uno propio modelo. El programa es totalmente gratuito y se puede descargar a través de la página principal de NetLogo, o a través del siguiente enlace:

<https://ccl.northwestern.edu/netlogo/download.shtml>

Como segunda opción, se puede visualizar online, ejecutando un applet (componente que permite la ejecución de un programa en otro contexto). Sin embargo, a través de este método u opción, simplemente se pueden visualizar los modelos, pero nunca crearlos ni editarlos.

Por ello, para conseguir una experiencia más completa y plena y poseer un mayor abanico de posibilidades, sugerimos la instalación del paquete, recordando que este es totalmente gratuito, por lo que no tendría mayores consecuencias.

Los archivos generados y/o utilizados en Netlogo serían aquellos cuya extensión es .nlogo. La recomendación otorgada al lector consistiría en que todos los archivos y actividades referentes a Netlogo se guarden en la misma carpeta, para de este modo agilizar la búsqueda, e incluso por motivos puramente técnicos los cuales también afectan en cierto modo al funcionamiento del mismo y que posteriormente mencionaremos.



Aplicación multiusuario con NetLogo

3. Manual del usuario

3.2.1 Aspectos generales de NetLogo

En este apartado definiremos y daremos unas bases o nociones, sin entrar en mucho detalle, sobre el funcionamiento, posibilidades y pautas a seguir para realizar una actividad a través de este programa.

Comenzaremos exponiendo y explicando la barra de menú superior, la cual es similar a la de cualquier otro programa conocido. Ver Figura 3.1

Archivo Editar Herramientas Tamaño Pestañas Ayuda

Figura 3.1. Barra de Menu.

Como se puede apreciar en la Figura 3.1 y como hemos mencionado tiene muchas de las utilidades de cualquier programa, por lo que no entraremos en más detalle sobre estos. A excepción de la pestaña Herramientas en la cual existen varias posibilidades de configuración importantes dentro de NetLogo, como vemos en la Figura 3.2.

Herramientas	Tamaño	Pestañas	Ayuda
Preferences...			
Detener			
Monitor de Variables Globales			
Monitor de Tortugas			
Monitor de Parcelas			
Monitor de Enlaces			
Cerrar Todos los Monitores de Agentes			
Cerrar monitores para agentes muertos			
Ocultar Entrada de Comandos			Ctrl+Barra
Switch to 3D View			Ctrl+Mayús+T
Muestras de Colores Disponibles			
Editor de Formas para Tortugas			
Editor de Formas para Enlaces			
Modelador de Dinámicas de Sistemas			Ctrl+Mayús+D
Preview Commands Editor			Ctrl+Mayús+P
Analizador de Comportamiento			Ctrl+Mayús+B
Editor del Cliente HubNet			
Centro de Control del HubNet			Ctrl+Mayús+H

Figura 3.2. Pestaña de herramientas.



Aplicación multiusuario con NetLogo

3. Manual del usuario

Para todo lo demás, cerciorarse de la existencia de la opción de Ayuda, contenida en la barra superior (ver Figura 3.1) y en la cual se resuelve todas las dudas existentes en cuanto a funcionalidades del programa.

Ahora nos centraremos en el resto de opciones, esta vez sí, más específicas de NetLogo. En un primer momento, es decir, nada más ejecutar el programa nos encontraríamos con la siguiente ventana, representada en la Figura 3.3.

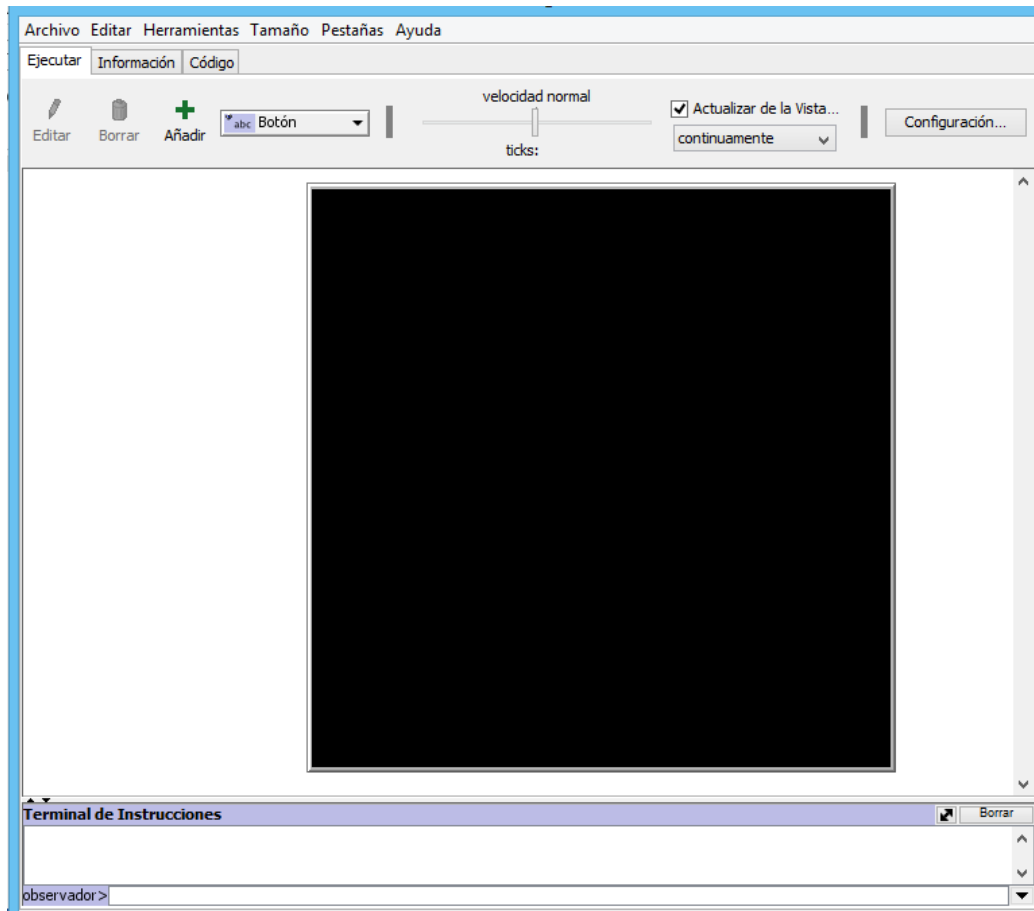


Figura 3.3. Ventana principal.

En esta, tendríamos todo lo necesario para desarrollar una aplicación cualquiera. Empecemos por ir definiendo apartado por apartado.

En un primer momento, nos encontraremos en la parte superior tres pestañas, las cuales, son Ejecutar, Información y Código (ver Figura 3.4).

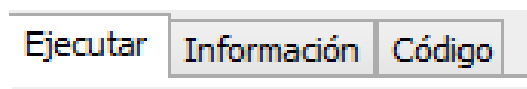


Figura 3.4. Pestañas de función.



Aplicación multiusuario con NetLogo

3. Manual del usuario

En la pestaña Ejecutar, nos encontramos con la pantalla que se puede visualizar en la Figura 3.3. En esta pestaña es donde se van a situar todos los botones, deslizadores, graficas, etc. Estos elementos se pueden situar en cualquier posición deseada. También es el lugar en donde se encuentra el visor 2D (cuadro negro) en el que se representan muchas de las simulaciones. Este se puede configurar a través del botón superior llamado Configuración, o clicando sobre él en el botón derecho.

Las posibilidades de adición en esta pestaña vienen representadas en la Figura 3.5.

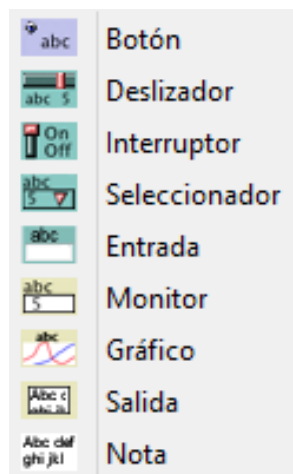


Figura 3.5. Elementos.

En esta Figura 3.5 podemos observar todas las posibilidades existentes, las cuales serán necesarias para desarrollar o posibilitar nuestra experiencia.

Recordar que para una mayor información sobre las posibilidades de configuración del programa o las especificaciones de algunos de los controles o botones, existen varios métodos de consulta como son la Ayuda, situada en la parte superior de la ventana, o diferentes páginas web o video tutoriales, las cuales podrás consultar en la bibliografía de este trabajo.

Por último, en el cuadro inferior denominado como Terminal de Instrucciones (ver Figura 3.3) se pueden introducir diversas instrucciones o código el cual ejecuta acciones visibles en el visor 2D para las tortugas (agentes) y parches. Gracias a este terminal, podemos ejecutar e introducir todo tipo de instrucciones las cuales no son necesarias de introducir como código (se verá más adelante) y es de gran ayuda ya que facilita y agiliza las acciones de búsqueda y manipulación para los agentes o parches.

La pestaña de Información (ver Figura 3.4), se utiliza para que el creador del programa o aplicación exponga y explique al usuario qué es la aplicación con la que va a trabajar, cómo funciona, cómo se usa, temas sobre los que



Aplicación multiusuario con NetLogo

3. Manual del usuario

percatarse... En definitiva, es una especie de manual del programa para el usuario para mostrarle y orientarle sobre cómo este funciona y que sugerencias de uso se le exponen.

Por último, la pestaña de Código está destinada para albergar todas las instrucciones o código necesario para que funcione el programa. En esta pestaña se genera el programa, sin embargo, el usuario básico que utilice la aplicación, no necesitará acceder a esta pestaña, evitando así que pueda modificar el contenido de esta e incurriendo en algún error de programación. Mencionar que NetLogo provee un diccionario, dentro de la pestaña de ayuda, en el cual se explican muchas de las instrucciones utilizadas.

3.3 HubNet

Como ya dijimos en el apartado anterior, NetLogo es una plataforma o entorno de programación multiagente el cual posibilita el desarrollo de acontecimientos, simulaciones o fenómenos de diversa índole o carácter, ya sean sociales, naturales...

NetLogo facilita HubNet, un módulo a través del cual se posibilita la participación de varios usuarios del programa o aplicación creada. Es decir, a través de HubNet, múltiples jugadores o usuarios pueden realizar la experiencia simultáneamente.

Muchas veces el programa creado en NetLogo se destina a la representación o simulación de un fenómeno o actividad y para ello es probable que no sea necesario la participación de múltiples usuarios, sino simplemente la adición de unas condiciones de partida y, a través de las reglas o instrucciones definidas, obtenemos el resultado esperado. Sin embargo las posibilidades o productos obtenidos con NetLogo aumentan gracias al módulo HubNet ya que de este modo podemos realizar simulaciones participativas para representar, por ejemplo un mercado de valores o una subasta, al disponer de múltiples usuarios o participantes tomando las decisiones que cada uno crea correctas y así obtener unos resultados totalmente diferentes ronda tras ronda. Por tanto, de esta herramienta contenida en NetLogo nos beneficiaremos enormemente y será, así pues, el modo de conectar a todos los usuarios que queramos para representar nuestra actividad.

Por tanto podríamos decir que HubNet añade una dimensión más a nuestro programa facilitando y posibilitando que las simulaciones o experiencias no se



Aplicación multiusuario con NetLogo

3. Manual del usuario

ciñan únicamente a unas cuantas reglas, sino a las decisiones cambiantes de los participantes, los cuales pueden modificar sus decisiones ronda tras ronda en vista a unos resultados

HubNet está diseñado para la realización de simulaciones de carácter o con interacción social.

3.3.1 Aspectos de HubNet

La arquitectura de HubNet se basa en la relación existente denominada cliente-servidor. Esto quiere decir que existe un servidor el cual es referente al propio programa de NetLogo, es decir, que se ejecuta el programa desde este último, y el cual ejerce como administrador de la relación. Por tanto cuando se ejecuta un programa en NetLogo de HubNet, nos referiremos al primero como el servidor. Así pues, el servidor proporciona la aplicación para que, de este modo, los usuarios o participantes de la experiencia puedan conectarse y participar de ella a través de HubNet.

Para que cada participante pueda desempeñar la actividad correctamente a través de HubNet, precisa de un ordenador en el cual exista instalado el paquete NetLogo y que además esté conectado a una red.

3.4 Preparación para la experiencia

En este apartado expondremos los pasos a seguir para iniciar la aplicación creada en este presente trabajo de modo correcto y sencillo para los usuarios.

Primeramente, antes de empezar a realizar cualquier acción, se sugiere que de un modo introductorio la persona que desempeñe las funciones de administrador de la aplicación, realice diversas pruebas y prácticas para mostrar la funcionalidad del programa, y que de este modo, el resto de los participantes estén más familiarizados con este y sepan por donde comenzar o cómo deben actuar.

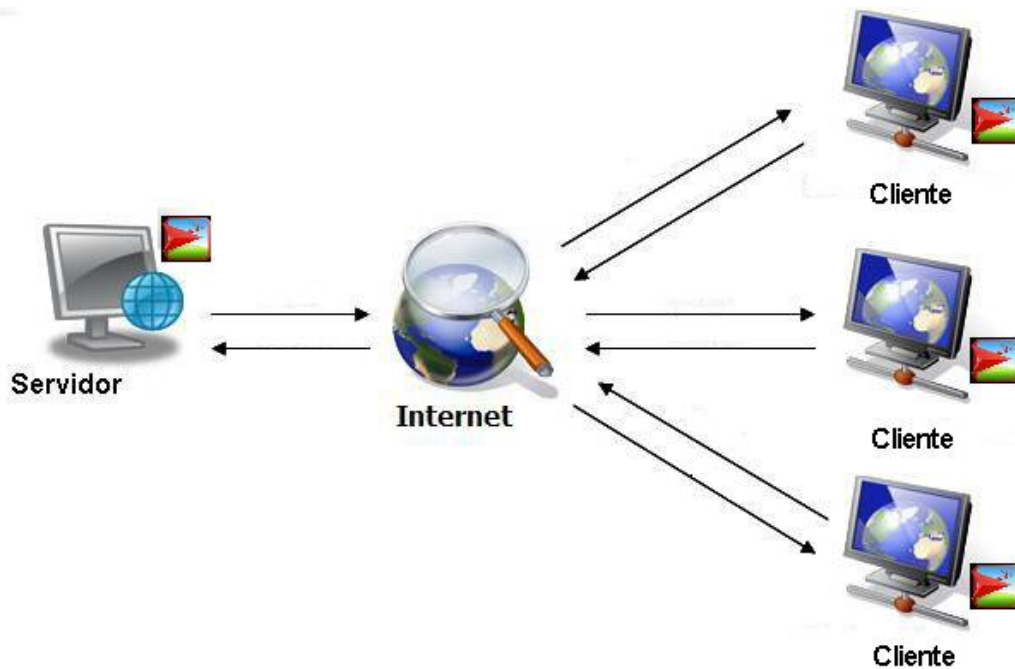


Figura 3.6. Relación cliente-servidor.²

El primer paso en esta preparación de la actividad será iniciar nuestro modelo (Modelo de competencia en precios con producto diferenciado) por parte de la persona que tomará el rol de administrador y servidor de la aplicación. Una vez iniciada la aplicación, NetLogo automáticamente despliega una ventana (Figura 3.7) para iniciar una nueva sesión de HubNet y en la que el administrador introducirá el nombre de la sesión, en nuestro caso, Modelo de competencia en precios con producto diferenciado. Este nombre será el que se le asocie a la actividad, y de este modo los usuarios que participen en ella lo verán e identificarán. Mencionar que el check (*Broadcast server location*) ha de estar clicado.

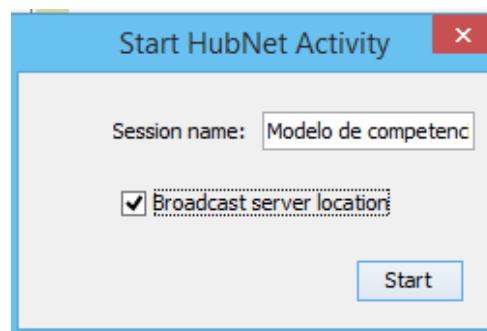


Figura 3.7. Ventana *Start HubNet Activity*

² Fuente: <http://majusanchezv.blogspot.com.es/2016/05/111-caracteristicas-de-la-computadora.html>



Aplicación multiusuario con NetLogo

3. Manual del usuario

Una vez clicado el botón Start, NetLogo abrirá de forma automática el Centro de Control de HubNet en el ordenador del administrador, Figura 3.8, para visualizar los participantes vinculados a la actividad, interactuar con ellos y seleccionar algunas opciones disponibles para un distinto desarrollo de la actividad.

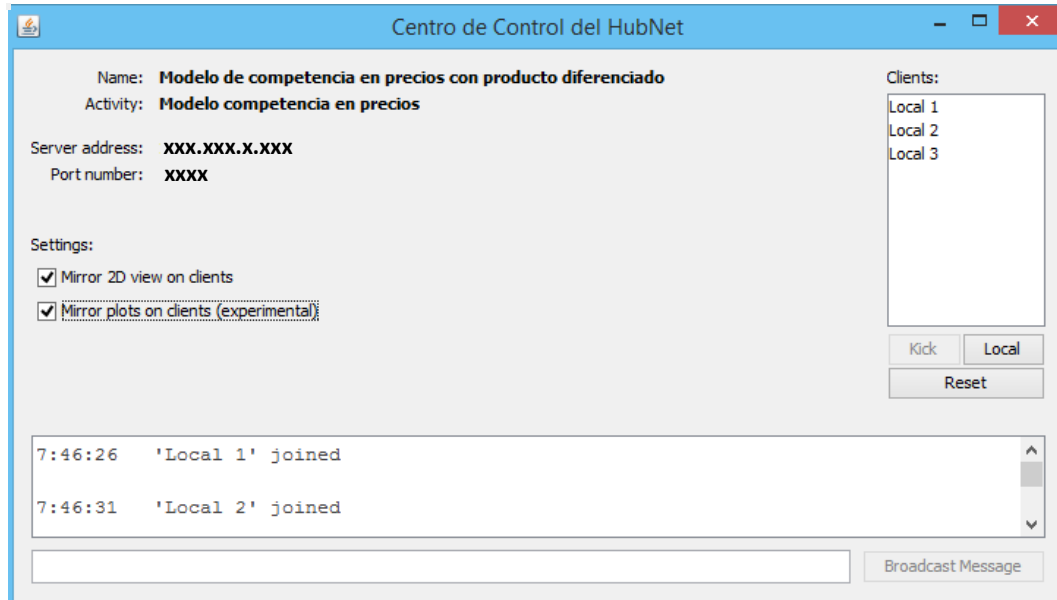


Figura 3.8. Ventana Centro de Control del HubNet.

Gracias al Centro de Control de HubNet, podemos interactuar con el módulo de HubNet. Como se puede apreciar en la Figura 3.8, este nos muestra el nombre el cual hemos dado a la sesión, el nombre de la actividad o programa ejecutado, la dirección del servidor y el número del puerto. Además podemos editar varias opciones. Si clicamos la casilla de *Mirror 2D view on clients*, la función obtenida sería que los clientes pudieran visualizar el visor 2D o “cuadro negro” en su propio interfaz. Mientras que la casilla *Mirror plots on clients* lo que permite es reportar y rellenar las gráficas de los interfaces de cliente y que así estos puedan recibir y ver dicha información. No solo sería necesario que dicha casilla estuviese seleccionada, sino que las gráficas del interfaz de cliente y del servidor o administrador deben presentar el mismo nombre, para que estas se muestren relacionadas.

En el cuadro superior derecho se muestran los usuarios conectados al servidor. Como se puede ver se pueden crear clientes locales, mediante el botón con ese mismo nombre, especialmente útil cuando lo que simplemente se quiere hacer



Aplicación multiusuario con NetLogo

3. Manual del usuario

es probar la actividad. También se pueden borrar participantes ya vinculados mediante el botón *Kick* o se puede resetear borrando así a todos.

Para finalizar con el Centro de Control de HubNet, existe un cuadro situado en la parte baja de la pantalla la cual informa de las entradas y salidas de usuarios a la sesión e incluso puede funcionar a modo de chat enviando a través de él mensajes a todos los participantes.

Todo lo anteriormente mencionado son las funciones que el administrador de la actividad debe realizar para poner en funcionamiento la actividad y permitir a los participantes poder conectarse.

Por su parte, los usuarios o participantes que deseen conectarse a la aplicación, deberán iniciar el *HubNet Client* en sus ordenadores. Una vez arrancado este módulo, les aparecerá la ventana de la Figura 3.9.

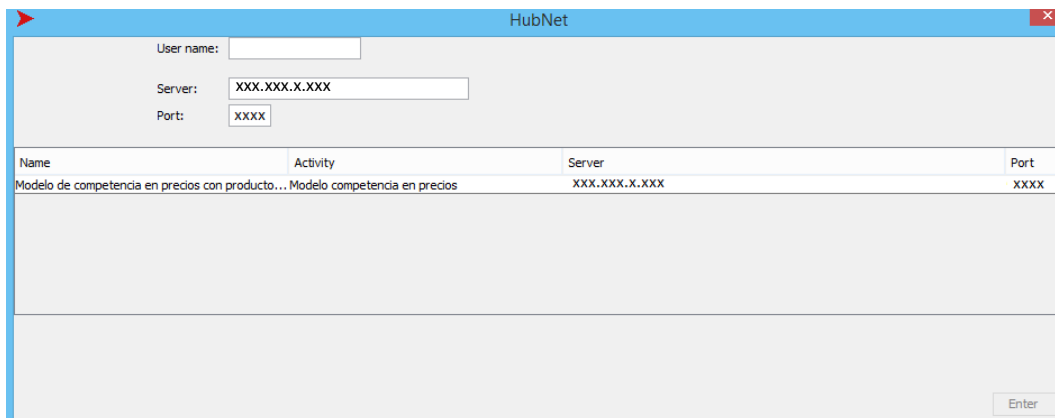


Figura 3.9. Ventana *HubNet Client*.

En esta ventana, los participantes deberán introducir el nombre deseado o asignado (*User name*) y seleccionar la sesión en la que deseen participar. Una vez seleccionada la sesión deberán pulsar *Enter* y ya podrán comenzar con la actividad. En el caso de que a los clientes no les aparezca la actividad o sesión deseada, tendrán la posibilidad de introducir manualmente el servidor y el puerto.



3.5 Manual para el usuario del “Modelo de competencia en precios con producto diferenciado”

En este apartado expondremos el manual para el usuario de la actividad desarrollada en este trabajo fin de grado, que es el “Modelo de competencia en precios con producto diferenciado”.

En un primer momento expondremos de una forma breve cual es el objetivo de la actividad y en qué consiste. Acto seguido, explicaremos de una manera más concisa los dos tipos de interfaces existentes, como ya hemos introducido anteriormente, es decir, el interfaz del usuario y el del servidor o administrador. Por último, daremos una guía en la que paso a paso se expone cual será el mejor desarrollo de la actividad.

3.5.1 Descripción de la actividad

Esta actividad está diseñada siguiendo el Modelo de competencia en precios con producto diferenciado, y por tanto, se desarrolla a través de esta aplicación para explicarlo de un modo más interactivo, más entretenido y de esta forma, conseguir afianzar más los nuevos conocimientos aprendidos. Este modelo explica el comportamiento seguido por las diferentes organizaciones existentes en un mercado cuya competencia es imperfecta, en concreto, en un mercado oligopolístico. Este, como ya sabemos, es característico por presentar pocos competidores o empresas que ofertan unos tipos de productos distintos o el del mismo tipo. Estos, por tanto, son muy buenos sustitutivos entre sí. Además es característico porque presenta grandes barreras de entrada a nuevos competidores. Un factor diferencial de este mercado es que las decisiones tomadas por las empresas competidoras en cantidades o en precio ofertado o puesto en el mercado, influyen de manera notoria a la hora de tomar nuestras propias decisiones.

La actividad desarrollada en este trabajo está orientada a multitud de participantes, por lo que se pueden conformar desde duopolios hasta mercados oligopolísticos en los que están presentes multitud de competidores. La teoría de esta aplicación es simple, cada participante o usuario piensa primero, y envía después al mercado los productos “fabricados” por su empresa al precio que considere oportuno, teniendo en cuenta siempre la función de costes de su fabricación. En base a este precio, el programa diseñado, en función de las diversas fórmulas de demanda, establece la cantidad que enviará cada empresa, así como la cantidad total ofertada en el



Aplicación multiusuario con NetLogo

3. Manual del usuario

conjunto del mercado y los beneficios obtenidos por cada empresa individual en función del propio precio establecido así como el del resto de competidoras.

Para aclarar el proceso seguido introduciremos los pasos siguientes:

- Cada participante de la actividad representa a una empresa ofertante.
- Toda empresa presenta una función de costes en base a la producción de su bien o producto. En nuestro caso, todas las empresas poseerán la misma función de costes, editada en parte por el administrador de la sesión.
- Los usuarios elegirán el precio al que quieren enviar su producto fabricado al mercado. Estos tendrán que tener en cuenta la función de costes introducida.
- Una vez realizada e introducido por parte de todos los usuarios el precio considerado como oportuno, se obtienen los valores resultado.
- El beneficio reportado por el programa será, como bien sabemos, la diferencia entre el producto obtenido del precio elegido y la cantidad producida, menos el coste de producción de todas las unidades.
- Al finalizar la sesión, se poseerá de una cantidad de información que nos permitirá discutir y comentar los resultados obtenidos durante la práctica, y así valorar cual eran las posibles mejores decisiones.

Para una mayor información del modelo utilizado en esta aplicación, puede verse el capítulo 2 en el cual se expone dicho modelo, e incluso toda la información o aspectos teóricos que rodean a esta práctica.

3.5.2 Interfaz del servidor

La misión de este presente apartado será la de exponer al lector las posibilidades y funciones de los diferentes aspectos que rodean al interfaz servidor o administrador, desde botones a *sliders*, pasando por monitores o gráficas. A parte de introducir los aspectos ya mencionados, se pretende mostrar cuál es su función y por qué son necesarios para controlar el desarrollo de la sesión.

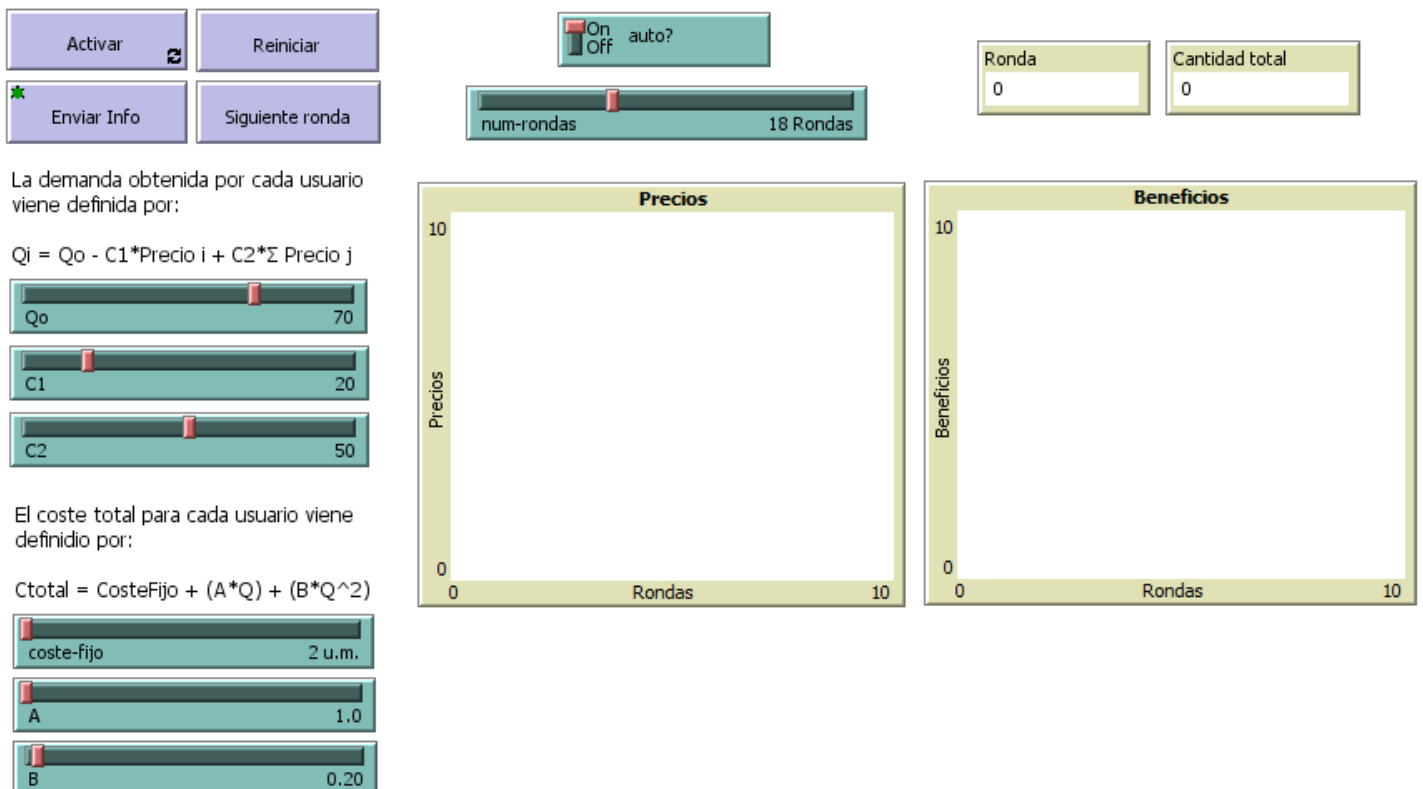


Figura 3.10. Interfaz del servidor.

En la Figura 3.10 se puede observar el interfaz del servidor. Este interfaz será el que utilice el administrador de la aplicación para introducir los parámetros que considere oportunos para la sesión, es decir, aquellos coeficientes que modifican las funciones de demanda y coste total de producción. Además, a través de este interfaz, podrá visualizará los resultados del cómputo de las decisiones de los participantes, es decir, los resultados presentes en el mercado.

A continuación pasaremos a describir los diferentes elementos presentes en esta interfaz.



Aplicación multiusuario con NetLogo

3. Manual del usuario

3.5.2.1 Botones

Los botones proporcionan una buena manera de controlar el modelo. Típicamente, un modelo tendrá al menos un botón de *Setup* para de este modo poder situar el mundo en un nivel o estado inicial, y uno de activación para hacer que el modelo avance o se ejecute. Algunos modelos tendrán botones adicionales que desarrollen otras acciones. En nuestro caso presentaremos cuatro botones diferentes.

Un botón puede contener cierto código de NetLogo, o por el contrario, puede tener una llamada a una instrucción definida en la pestaña de Código. Ese código se ejecuta una vez el botón es clicado.

Existen dos tipos de botones. Los primeros son aquellos que se clican una vez y ejecutan la regla una vez. Por el contrario existen otro tipo de botones los cuales ejecutan su código de manera continuada. Estos últimos pueden dejar de ejecutarse si vuelven a ser clicados. Sin embargo, cabe mencionar que si los botones tienen una llamada a un procedimiento, y este procedimiento para en cuanto acaba la regla, el botón lo hará del mismo modo

Conviene mencionar que los botones pueden tener una función genérica, como ejecutar el programa, borrar variables, etc. o pueden estar referidos a los agentes o tortugas.

Por lo general, un botón se llamará del mismo modo que la ejecución o acción que realice, dado que de este modo se interpreta rápidamente cuál es su función.

Como ya hemos dicho, en nuestro caso introduciremos cuatro botones diferentes, Figura 3.11.



Figura 3.11. Botones del interfaz servidor.

A continuación expondremos brevemente cual es la función de cada uno de nuestro botones:



Aplicación multiusuario con NetLogo

3. Manual del usuario

- **Activar:** La función de este botón es arrancar o ejecutar el programa. Este botón, como se puede observar en la Figura 3.11 es de tipo *Forever* es decir, su ejecución es continua, y esta acaba cuando se vuelve a clicar el botón o cuando la ejecución del procedimiento se detiene.
- **Reiniciar:** La misión de este botón es la de situar al programa en el estado inicial, es decir, retornar todas las variables, gráficas y funciones a cero o a su valor por defecto. Su función está destinada para la ejecución de una nueva sesión desde la posición inicial.
- **Enviar Info:** Este botón pretende enviar toda la información referente a la funcione de coste a los participantes. Debe ser pulsado siempre y cuando las condiciones de estas varíen en el transcurso de la sesión. Este botón está referido a los agentes como tal.
- **Siguiente ronda:** Tras ser pulsado, este botón lo que pretende es avanzar de ronda haciendo los cálculos necesarios en función de los valores que tengan los participantes en ese preciso momento. Una vez realizados los cálculos, muestra los resultados obtenidos.

3.5.2.2 Monitores

Los monitores son elementos proporcionados por NetLogo en los que, a través de ellos, podemos observar diferentes valores para así poder controlar diferentes variables, y en definitiva, poder controlar el transcurso de la sesión. Los monitores, al igual que los botones, pueden contener cierto código de NetLogo, o por el contrario, puede tener una llamada a una instrucción definida en la pestaña de Código. Gracias a los monitores podemos saber el valor que tiene o devuelve cierta variable interesante para nuestro estudio del modelo. Los monitores se caracterizan por devolver o mostrar valores numéricos que han sido obtenidos tras la ejecución de una instrucción o un procedimiento y gracias a ellos podemos visualizar y controlar los comportamientos de nuestra aplicación. Los monitores se actualizan automáticamente multitud de veces por segundo.

En nuestra interfaz servidor, como podemos ver en la Figura 3.12, presentamos y utilizamos dos monitores diferentes para así controlar el desarrollo de la sesión.

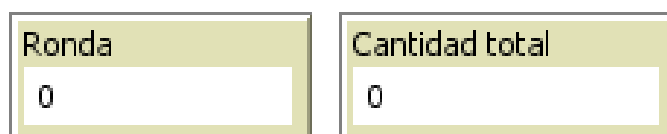


Figura 3.12. Monitores del interfaz servidor.



Aplicación multiusuario con NetLogo

3. Manual del usuario

- **Ronda:** Este monitor nos ayuda a conocer en que ronda de la sesión nos encontramos, y por consiguiente, las rondas transcurridas durante la experiencia. La ronda avanzará mediante la pulsación del botón “**Siguiente ronda**” o mediante un procedimiento automático (siempre y cuando se active) el cual avanza de ronda en el momento en el que todos los usuarios hayan enviado un precio al mercado.
- **Cantidad total:** Este otro monitor nos devuelve la cantidad total ofertada al mercado por parte de las empresas como resultado de la introducción de los diferentes precios elegidos por los usuarios. Es decir, es la cantidad total de bienes o productos ofertados al mercado por parte de todas las empresas competidoras. Esta cantidad está influenciada por los precios elegidos por los usuarios a la hora de introducir un bien al mercado.

3.5.2.3 Deslizadores

Los deslizadores son variables globales que son accesibles para todos los agentes. Son muy utilizados en el desarrollo de los modelos debido a que proporcionan una forma rápida de escoger o cambiar una variable sin tener que recodificar los procedimientos o instrucciones cada vez que se requiera una modificación de estos. En vez de tener que hacer todo lo anterior, gracias a la versatilidad que ofrecen estos elementos, el usuario, en este caso el administrador de la aplicación, puede elegir los valores que desee simplemente deslizando la barrita al valor que el estime oportuno.

En nuestro modelo, hemos utilizado deslizadores para elegir las diferentes cantidades de las constantes de las funciones, entre otras cosas, de un modo rápido y sencillo.

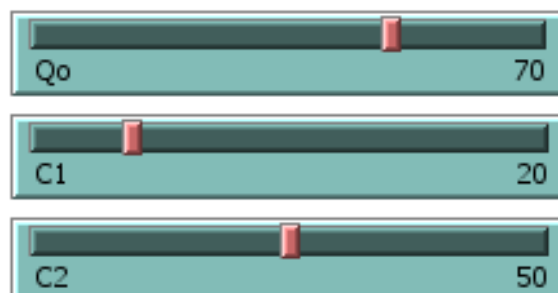


Figura 3.13. Deslizadores para las constantes de la función de demanda del interfaz servidor.

En la Figura 3.13 podemos observar los deslizadores que proveen de valor a las constantes de la función de demanda. De este modo, como ya hemos dicho, el administrador de la experiencia puede elegir el valor de las constantes

existentes en la función de demanda definida. La función de demanda recordemos que es:

$$Q_i = X - C_1 * P_i + C_2 * \sum P_j$$

En la Figura 3.14 observamos los deslizadores correspondientes a la función de costes de nuestro modelo.

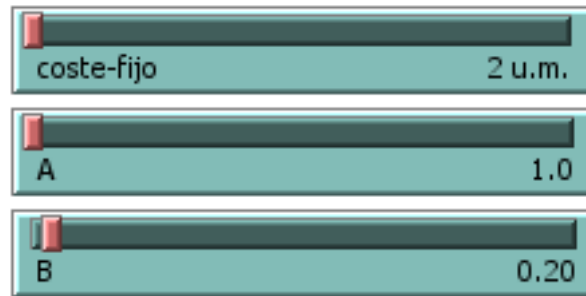


Figura 3.14. Deslizadores para las constantes de la función de producción del interfaz servidor.

Como podemos ver en esta figura anterior, se definen los valores de las constantes de la mencionada función de costes. Recordemos que esta es:

$$C_{Total} = CosteFijo + (A * Q_i) + (B * Q_i^2)$$

Como último deslizador presente en nuestro interfaz servidor, tenemos el seleccionador del número de rondas que queremos definir como duración total de la sesión o número de rondas deseadas para desarrollar la experiencia (ver Figura 3.15).



Figura 3.15. Deslizador del número de rondas.

3.5.2.4 Interruptores

Los interruptores son elementos o representaciones visuales de las variables globales de tipo verdadero/falso, true/false, 1/0. Para esos botones se debe definir un valor para cada una de las dos posibilidades en las que se puede situar.

Nuestra interfaz servidor posee un interruptor representado en la Figura 3.16. Este interruptor denominado como “Auto?” pretende activar o desactivar la función de paso de ronda automáticamente. Esto quiere decir que si está activo

(On) una vez que todos los usuarios hayan introducido el precio que consideren oportuno, se pasará a la siguiente ronda mostrando todos los datos, representaciones, etc. de la ronda pasada. De este modo, no sería necesario que el administrador pulsara en botón “**Siguiente ronda**” para avanzar esta. En el caso contrario (interruptor modo Off) será necesario clicar el botón de “**Siguiente ronda**” y este, por tanto, no será un proceso automático.

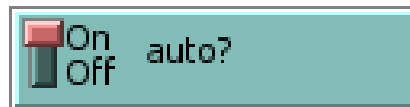


Figura 3.16. Interruptor para la función automática del avance de ronda.

3.5.2.5 Gráficas

Los gráficos son elementos de NetLogo que tienen como función mostrarnos o enseñarnos los datos que el modelo o programa ha generado. Estos, a través de las diferentes instrucciones seguidas por el programa, traducen estos resultados y dibujan o describen las gráficas, ya sea punto a punto, modo en el cual se van dibujando ronda tras ronda, o generales, en función de los datos obtenidos.

En nuestro interfaz servidor tenemos presentes dos gráficas, una denominada **Precios** y otra llamada **Beneficios** (Figura 3.17).

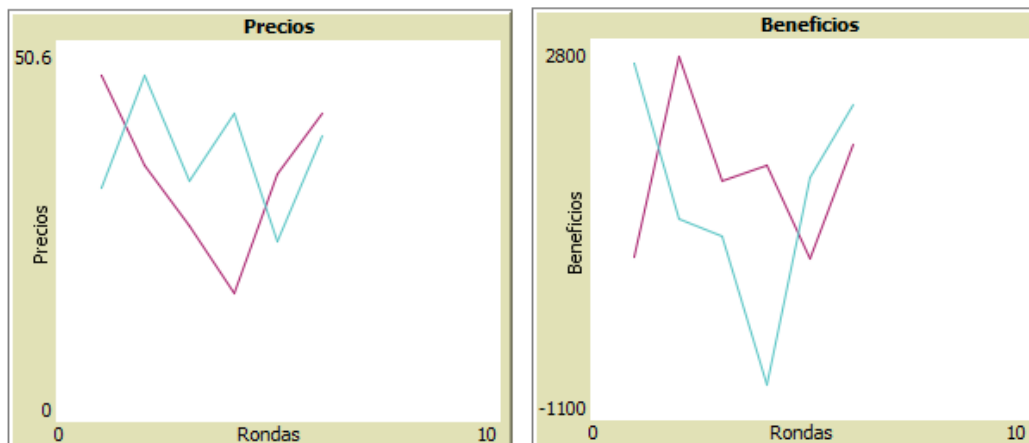


Figura 3.17. Gráficas de precios y beneficios.

Estas gráficas se caracterizan por ir formándose y desarrollándose ronda tras ronda en función de los valores que han tomado los participantes en cada turno. Explicaremos cual es el significado de cada una:

- **Precios:** Esa gráfica representa y dibuja los precios a los que los usuarios venden u ofertan sus bienes en una determinada ronda. Se caracteriza por

Aplicación multiusuario con NetLogo

3. Manual del usuario

tener en una misma gráfica todos los precios tomados o escogidos por todos los participantes ronda tras ronda, para así dar una mejor aproximación o perspectiva de la tendencia tomada y dar una idea de que elección futura será la más conveniente en vista a estos resultados y tendencias.

- **Beneficios:** Esta gráfica tiene una función similar a la anterior, sin embargo en este caso, la variable significativa es la variable beneficio. Por tanto, representa el beneficio obtenido por cada empresa, en cada ronda, en función del precio al que oferta su producto al mercado.

Para la representación de la Figura 3.17, lo hemos ejemplificado para un modelo de duopolio, es decir para un mercado en el que solo compiten dos empresas. Como se puede observar en la Figura 3.17, vemos como cumple a la perfección la función de demanda establecida en la que, para un precio superior de una empresa con respecto a otra, el beneficio esperado será menor. Ronda tras ronda esto se va cumpliendo, y cada una, intenta igualar su precio a la otra ya que de este modo su beneficio será igual de alto.

Vemos también como para cada usuario se ha asignado un color distinto con el fin de identificar fácilmente la diferencia entre ellos y ver de qué usuario se trata. El color asignado en la gráfica, coincidirá con el color asignado al usuario en el visor 2D. Todo esto facilita la identificación de los usuario para así conocer fielmente que datos corresponden con que usuario y así pues, facilitar la discusión posterior sobre estos.

En este apartado, convendría mencionar que existen otras cuatro gráficas que no se muestran como tal, ya que se encuentran ocultos detrás de las gráficas anteriores, Figura 3.18.

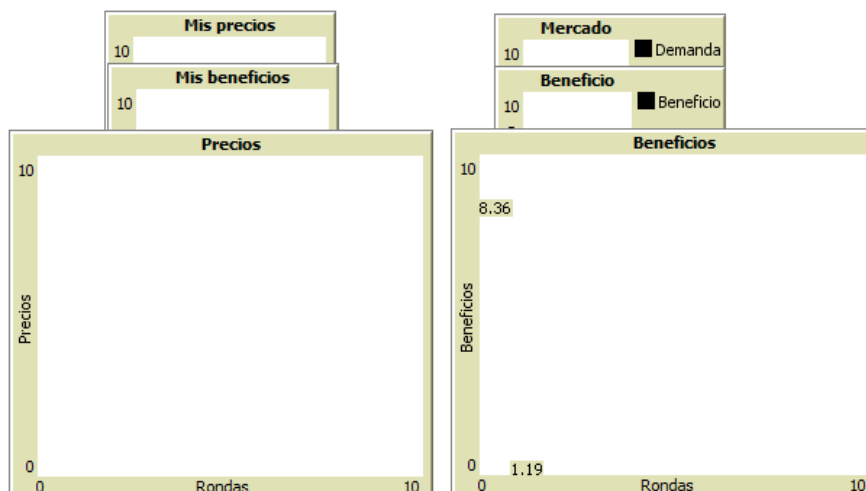


Figura 3.18. Graficas ocultas en el interfaz servidor.

Esto es debido a que, para que estas gráficas ocultas aparezcan en el interfaz de usuario, han de ser creadas en el interfaz servidor primero. Esto es motivado simplemente requerimientos de programación y transmisión de datos del propio programa NetLogo. La funcionalidad de estas gráficas las trataremos en el siguiente apartado donde nos centraremos en las funciones y elementos del interfaz de usuario.

3.5.3 Interfaz del usuario

En este apartado nos centraremos en exponer los elementos presentes en el interfaz de usuario, así como sus funcionalidades y propiedades para dar una idea de uso a los participantes que tomen este rol. En esta sección no entraremos tan en detalle en las funcionalidades que ofrecen los diferentes elementos ya que esta descripción ya ha sido abordada en los apartados previos.

Empezaremos mostrando una imagen a modo de esquema para que todos aquellos participantes que tomen el rol de usuario y utilicen este interfaz, tengan así una primera toma de contacto con este. En la Figura 3.19 podemos ver el interfaz del usuario.

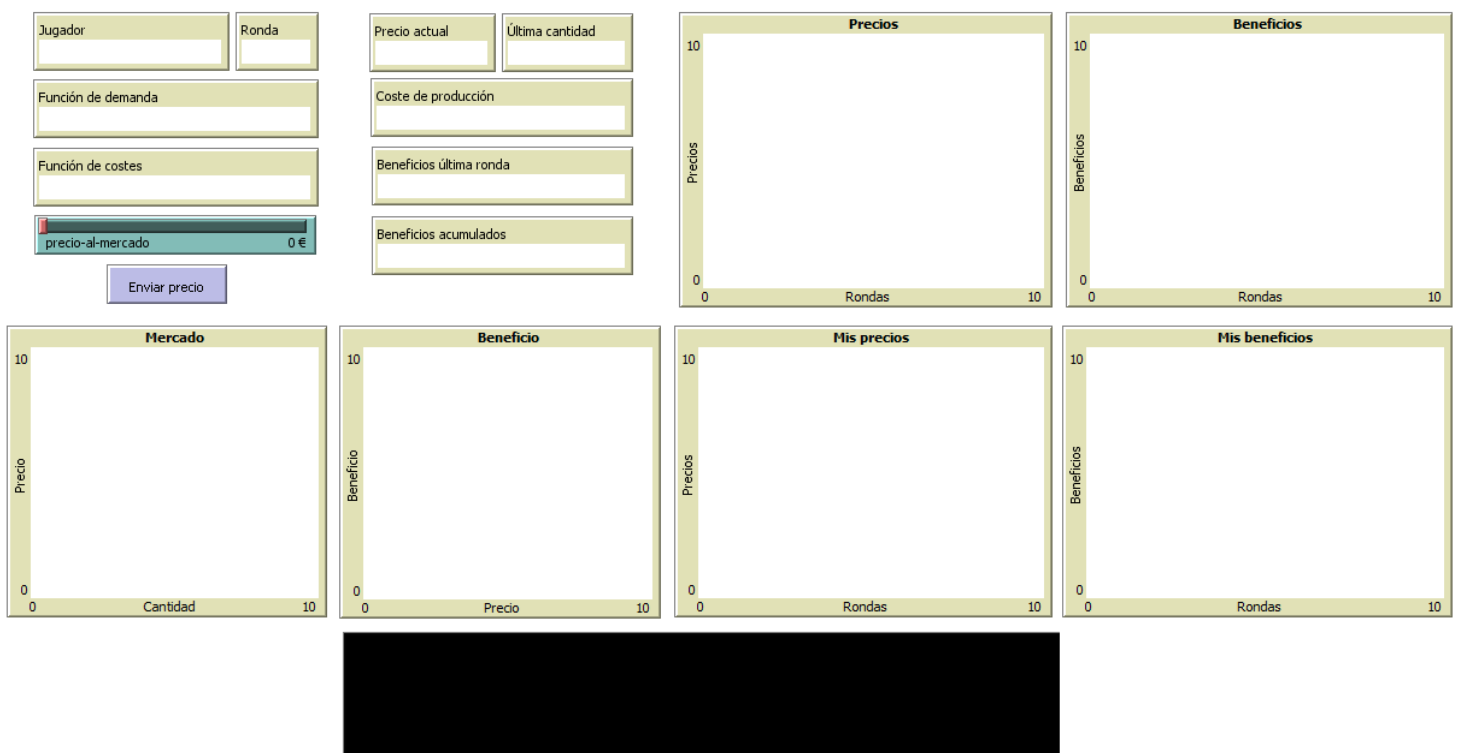


Figura 3.19. Interfaz de usuario.



Aplicación multiusuario con NetLogo

3. Manual del usuario

Como podemos observar en la Figura 3.19, el interfaz de usuario está muy poblado de gráficas y monitores y mucho menos de botones. Esto es así dado que los botones tienen una función más de configuración o acción (tarea reservada para el administrador de la aplicación) mientras que las gráficas tienen el objetivo de aportar información para que de este modo el usuario sepa con mayor exactitud la elección del precio al que quiere ofertar su empresa en la siguiente ronda, y así obtener un beneficio mayor.

A continuación pasaremos a describir los diferentes elementos presentes en esta interfaz.

3.5.3.1 Botones

En este interfaz solo encontraremos un botón dado que, como hemos expuesto en la introducción del apartado 3.5.3, los botones tienen una función más destinada a ejecutar acciones o configurar parámetros, cosa que en nuestra aplicación y para nuestros usuarios no tiene cabida. Como vemos en la Figura 3.20, el único botón que tenemos sirve para enviar el precio al que cada usuario quiere que su empresa venda u oferte los bienes o productos producidos al mercado.

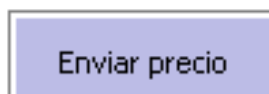


Figura 3.20. Botón *Enviar precio*.

3.5.3.2 Monitores

Los monitores, como ya hemos explicado previamente, tienen la función de proporcionar la información reportada por los diferentes valores o variables para así poder controlar de un mejor modo el transcurso de la sesión y las variables interesantes.

En nuestro modelo se adjuntan diferentes monitores, todos ellos para arrojar la mayor información posible al usuario y que este sepa que decisiones ha de tomar en el futuro basándose en las que tomo en el pasado o presente. Los monitores presentados en nuestro interfaz de usuario aparecen en la Figura 3.21.

Aplicación multiusuario con NetLogo

3. Manual del usuario

The screenshot displays a user interface with several monitors (input fields and a slider) arranged in two columns. The left column contains: 'Jugador' (Player) and 'Ronda' (Round) input fields; 'Función de demanda' (Demand function) and 'Función de costes' (Cost function) input fields; and a 'precio-al-mercado' (price-to-market) slider set to 0 €. The right column contains: 'Precio actual' (Current price) and 'Última cantidad' (Last quantity) input fields; 'Coste de producción' (Production cost) input field; 'Beneficios última ronda' (Last round profit) input field; and 'Beneficios acumulados' (Accumulated profit) input field. A 'Enviar precio' (Send price) button is located below the slider.

Figura 3.21. Monitores del interfaz usuario.

Expondremos los monitores presentados.

- **Jugador:** En este monitor se presenta el nombre de usuario elegido por el participante a la hora de unirse a la sesión. Este es útil ya que a través de este nombre se podrá identificar tanto al usuario en sí, como a sus decisiones durante el transcurso de la experiencia.
- **Ronda:** Este monitor nos informa de la ronda presente en la que se encuentra la sesión o dicho de otro modo, las rondas transcurridas desde el inicio de esta.
- **Función de demanda:** Como su propio nombre indica, nos ofrece y muestra la función de demanda elegida por el administrador y aquella que seguirá el mercado durante la sesión. Esta función es idéntica para todas las empresas.
- **Función de costes:** Nos ofrece la función de costes del proceso productivo elegida por el administrador. Es decir, aquella que gravará el precio elegido por los usuarios y mermará su beneficio. Esta función es idéntica para todas las empresas
- **Precio actual:** En este monitor se muestra el precio que envió el usuario en la presente ronda y en el cual se situarán todos los resultados mostrados.
- **Última cantidad:** Cantidad total ofertada por todas las empresas la ronda anterior a la presente en función de los precios a los que quisieron vender o introducir sus productos al mercado.



Aplicación multiusuario con NetLogo

3. Manual del usuario

- **Coste de producción:** Este monitor muestra el coste de producción incurrido en la ronda anterior en función del precio que se escogió y de la función de costes de producción vigente.
- **Beneficio última ronda:** Se muestran los beneficios obtenidos en la última ronda por parte de la empresa, motivados por el precio propio escogido y el precio escogido por la competencia.
- **Beneficios acumulados:** Este monitor presenta el sumatorio de todos los beneficios propios a lo largo de todas las rondas de la presente sesión.

3.5.3.3 Deslizadores

Es este interfaz, solo contamos con la presencia de un deslizador. Este deslizador, representado en la Figura 3.21, se denomina **precio-al-mercado**. La función de este es aportar funcionalidad, versatilidad y facilidad a la hora de elegir el precio al que el usuario quiere introducir los bienes producidos por su empresa al mercado. Además, gracias a la utilización de deslizadores para esta funcionalidad, se limita los rangos de precios posibles de introducción, y así se controla de un mejor modo el correcto funcionamiento de la práctica. Esta limitación o elección del rango de posibles precios a introducir está motivado por el hecho el cual, al seguir la función de demanda utilizada en nuestro modelo, si un usuario trata de escoger un precio extremadamente alto, falsea el resultado del resto de participantes ya que tendrían unos altos beneficios al vender todos sus productos a costa del alto precio de este último. Sin embargo la realidad resulta distinta ya que si una empresa vende extremadamente caro su producto, la realidad es que los clientes no adquieren ni uno de sus productos y miran al resto de empresas. Pero hay que mencionar que no miran al resto de empresas del mismo modo a todas, sino que vuelven a filtrar por precio. Esto último en el programa no sucede. Por tanto para paliar esta deficiencia del modelo, se ha propuesto limitar los rangos de precios y que esos estén contenidos y sean coherentes.

3.5.3.4 Gráficas

Como ya hemos comentado, los gráficos son elementos de NetLogo que tienen como función mostrarnos o enseñarnos los datos que el modelo o programa han generado. Este apartado resulta muy interesante para este interfaz ya que el usuario necesita de mucha información, tanto pasada como presente, de sus actuaciones y de las actuaciones de sus competidores para definir su estrategia competitiva y así obtener los mayores beneficios posibles.

Conscientes de esta máxima, en la interfaz de usuario de la aplicación se han incluido multitud de gráficas para mostrar a cada responsable de cada

empresa datos reales de varias magnitudes valoradas, para que así puedan trazar una estrategia de un modo más sencillo.

En la Figura 3.19 se ilustran las gráficas adjuntas de un modo general.

A continuación pasaremos a definir cada una de las gráficas, para que el lector sepa con exactitud cuál es la función de cada una de estas.

En un primer momento nos encontramos con las gráficas de la Figura 3.22.

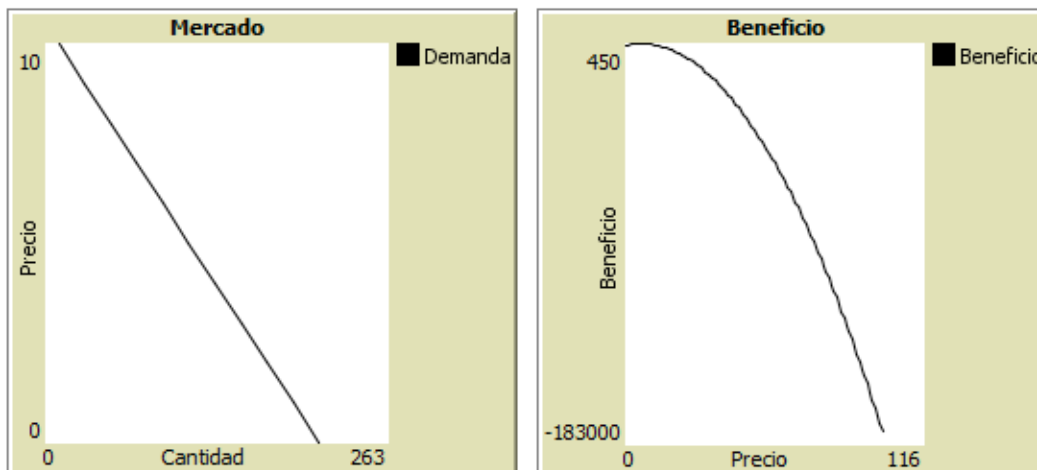


Figura 3.22. Gráficas de mercado y beneficio.

La gráfica titulada como **Mercado**, nos muestra la función de demanda propiamente dicha. Como ya sabemos, esta función se caracteriza por tener una tendencia descendente, en lo que a la evolución de la gráfica se refiere, y viene definida en función de los precios y las cantidades. Esta gráfica se dibuja gracias a la definición de la función introducida por el administrador, y los diferentes precios seleccionados por los usuarios para competir en el mercado con sus productos. Se trata de una gráfica específica para cada usuario ya que valora, manteniendo constantes los precios elegidos por los competidores en la ronda anterior, que cantidad se introducirá en el mercado para que diferente elección de precio. Aunque en nuestra aplicación, toda la oferta producida por las empresas e introducida en el mercado será captada, esta gráfica es de especial importancia para conocer cuál es la demanda existente en función de los precios escogidos por todas las empresas.

Por otro lado, la gráfica titulada como **Beneficio** nos muestra cual es el beneficio obtenido en función de los precios. Dicho de otro modo, nos devuelve los beneficios posibles en función de un rango diferente de precios. Esta gráfica se construye calculando los beneficios que se pueden obtener en función de

los precios escogidos por las empresas. Esta gráfica, al igual que la anterior, es una gráfica específica para cada participante, para arrojarles una idea de que precios han de escoger para obtener unos beneficios mayores. Conviene mencionar que representa la función en base a los precios escogidos en la ronda anterior de modo que, aunque dan una idea general del estado del mercado, en la siguiente ronda cambiará. Está, por tanto, más orientada a mejorar la decisión propia del precio de oferta al mercado de nuestro producto si la competencia volviera a elegir el mismo precio de venta al mercado.

En la Figura 3.23 mostramos otras dos gráficas presentes en este interfaz.

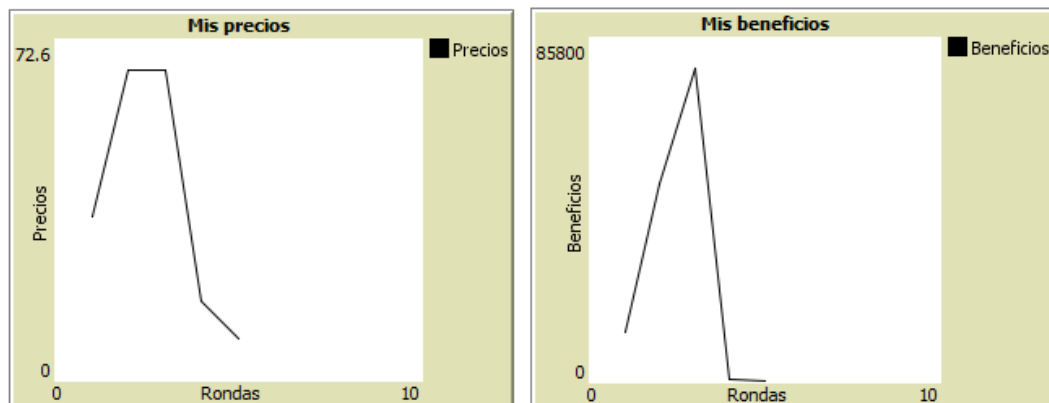


Figura 3.23. Gráficas de los precios y beneficios propios del usuario.

Las dos gráficas ilustradas en la Figura 3.23, exponen datos propios de cada empresa, conformados ronda tras ronda en función de sus elecciones.

La gráfica nombrada como **Mis precios**, muestra el histórico de precios escogidos por cada empresa ronda tras ronda hasta la actualidad.

Por el contrario la gráfica **Mis beneficios**, expone el histórico de los beneficios obtenidos en cada ronda en función de la elección de los precios de cada empresa y de sus competidoras.

Ambas gráficas son distintas y personalizadas para cada empresa.

Conviene mencionar que para que los datos de estas gráficas se pueden mostrar a cada empresa o no en función de lo que considere el administrador. Simplemente basta con que el administrador de la sesión marque o no la casilla de *Mirror plots on clients* en el Centro de control de Hubnet cuando está configurando la sesión. En caso de que este la seleccione, las gráficas de la Figura 3.23 se rellenarán con los datos propios.



Aplicación multiusuario con NetLogo

3. Manual del usuario

Por último, en este interfaz de usuario también se han introducido las gráficas presentes en el interfaz del servidor, con el fin de dar una idea general de lo que está pasando y que todos y cada uno de los participantes conozcan de primera mano cuál es su posición en el mercado con respecto al resto de competidores.

3.5.4 Pasos a seguir para el correcto desarrollo de una sesión

En este último apartado de este capítulo, ofreceremos unas pautas a seguir para que la experiencia obtenida a través de nuestra aplicación sea completa y de gran valía.

Como ya sabemos, nuestra aplicación trata el Modelo de competencia en precios con producto diferenciado. De modo que vamos a exponer la correcta utilización de este programa para ver la esencia de este modelo.

Los primeros pasos corresponden al usuario que ejercerá de administrador de la sesión. Este configurará, dirigirá y controlará todo lo relacionado con la aplicación y experiencia que él desea que suceda. A través del interfaz de servidor realizará todas estas acciones. Por otro lado, todas las decisiones tomadas por el administrador, repercutirán en los participantes, que mediante el interfaz de usuario, participarán de dicha experiencia acorde a esos parámetros definidos. Los pasos que habrá que seguirse para conseguir una experiencia plena serán:

Lo primero de todo es que todos los usuarios tengan a su disposición un ordenador con el paquete de NetLogo instalado (hay otras posibilidades, pero esta es la más recomendable). Cerciorándose que esto es así, los participantes que tomen el rol de usuario, y por tanto, que participaran en la actividad, dirigiendo, organizando y eligiendo los precios que enviarán al mercado a través del interfaz usuario, deberán abrir el módulo de Hubnet. Por otro lado, el usuario que tome el rol de administrador, deberá abrir el programa propiamente dicho con la herramienta NetLogo. Este, como ya hemos dicho, configurará los parámetros siguiendo los pasos definidos en el apartado 3.4.

Una vez abierto el programa por parte de todos los participantes, configurados todos los parámetros y unidos y listos todos para empezar la actividad, lo primero que ha de hacer el administrador es clicar el botón **Activar** en el interfaz servidor. Este debe permanecer clicado durante todo el desarrollo de la sesión, ya que es el botón que ejecuta y pone en marcha todos los procedimientos. Al ser un botón de tipo *Forever*, como ya hemos dicho, este permanecerá automáticamente clicado hasta que el administrador decida. Acto seguido de realizar esta acción, se podrá comprobar en el interfaz del servidor como en el



Aplicación multiusuario con NetLogo

3. Manual del usuario

visor 2D negro van apareciendo representaciones con diferentes colores de los diferentes usuarios que van a participar de la experiencia. En la Figura 3.24 se representa este fenómeno.



Figura 3.24. Representación de cada usuario.

Debajo del icono, como se puede apreciar en la imagen superior, aparecen tres valores. El primero es el nombre del usuario (para poder identificarlo rápidamente), el segundo, esta vez un valor numérico, representa los beneficios acumulados por este usuario a lo largo de las rondas completadas, es decir, la suma de todos los beneficios de las rondas hasta la última. Y por último aparece el último precio al que envió sus productos al mercado (el precio elegido en la ronda anterior). Como vemos en la Figura 3.24, los valores de estas son cero, dado que la simulación no habría comenzado.

La actividad se puede pausar o interrumpir pulsando el botón **Activar**, y se volverá a ejecutar donde estaba volviéndolo a clicar.

Primeramente, antes de empezar la sesión como tal, e iniciar la primera ronda, el administrador debe establecer los valores de diferentes coeficientes introducidos en las dos funciones existentes (función de demanda y función de costes). Como ya hemos dicho, la definición de estos coeficientes lo realizara deslizando el elemento deslizador y eligiendo el valor que considere oportuno para las variables Q_0 , C_1 , C_2 , **Coste-fijo**, **A** y **B**. Ahora ya se puede comenzar la experiencia. Comentar que para cerciorarnos de que no exista ningún problema que se ocasione en el futuro, sería recomendable pulsar el botón **Reiniciar** para que todas la variables y contadores estuvieran en su valor inicial definido por defecto.

Una vez todo esté definido, se instará a los usuarios encargados de sus empresas a seleccionar a través del deslizador **precio-al-mercado**, el precio que consideren oportuno para la primera ronda y una vez hecho esto, pulsen el botón **Enviar precio**. Solamente se podrá enviar un único precio por ronda, por tanto no se podrá rectificar la decisión. Esta acción de solicitar a los participantes que introduzcan su precio puede hacerse de la forma tradicional, es decir, mediante el uso de la voz del administrador, o por el contrario se puede hacer mediante el Centro de control HubNet a través de un simple mensaje.

Aplicación multiusuario con NetLogo

3. Manual del usuario

El precio al que cada empresa envía sus productos se reflejará en el monitor **Precio actual** para mostrar al usuario cual ha sido su elección. El precio que escoja cada empresa para vender sus productos, se traducirá en una cantidad producida por estos. Esta cantidad tendrá una serie de costes, de acuerdo a la función de costes establecida. Como resulta obvio, los costes incurridos por la producción penalizará el beneficio obtenido por la empresa. Como ya hemos dicho, el mercado acepta toda la oferta existente, por tanto el beneficio de las empresas dependerá de la función de demanda y de la función de costes.

Cada vez que un usuario envía el precio al que quiere vender en el mercado, su representación en el visor 2D cambiará de icono. Esto tiene el fin de conocer con exactitud en qué posición se hayan cada una de las empresas (ver figura 3.25).

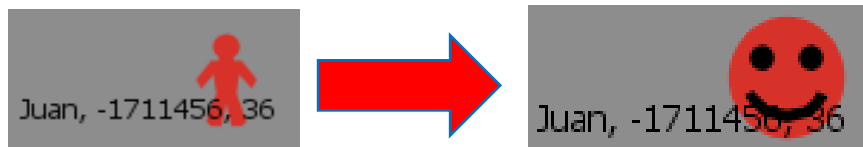


Figura 3.25. Cambio de icono una vez enviado el precio.

En el momento en el que todos los usuarios hayan seleccionado y enviado su elección existen dos opciones. Una sería la resultante de un avance de ronda automático. Para que esto sea posible el interruptor **Auto?** ha de estar en modo *On*. Si este está en dicho modo, una vez que todos los usuarios hayan enviado sus cantidades, la ronda avanzará de un modo automático. La segunda opción consiste en que el administrador, cuando sepa que todos los usuarios han introducido su elección, pulse el botón **Siguiente ronda**. Se recomienda la primera de las opciones porque de este modo se está seguro de que todos los participantes han enviado su elección. Además de la rapidez que aporte a la sencillez.

Una vez se avance de ronda se rellenarán de datos todos los monitores, gráficas... También se cambiará de nuevo el icono referente al usuario a su estado inicial. Comentar que es posible que en la primera ronda algunas gráficas no muestren datos representados. Esto es debido a que la representación que se hace en la gráfica es de un punto unitario, de modo que hasta que no se avanza a la siguiente ronda (ronda 3) no se traza la línea entre los dos puntos anteriores (ronda 1 y 2).

Acto seguido se realizará una nueva selección de precio, pero esta vez ya con datos y pruebas que puedan ayudar a postularse por uno u otro precio. Este



Aplicación multiusuario con NetLogo

3. Manual del usuario

proceso se repetirá sucesivamente y así tantas rondas como haya definido el administrador.

Si en algún momento se quieren cambiar las funciones de demanda y de costes, el administrador debe reasignar los valores a las variables y pulsar el botón **Enviar info** para que a los usuarios les llegue las nuevas reglas de juego. También se pueden reestablecer los datos iniciales sin más que pulsar el botón **Reiniciar**.

3.5.4.1 Sugerencias y observaciones para una mejor experiencia

Con el fin de obtener unos resultados más fieles a la realidad, se recomienda a los usuarios que realicen sesiones de 15 a 20 rondas. Cuando finalicen la sesión, que comiencen otra habiendo variado las funciones de demanda y costes.

También se puede probar como y de qué manera afecta que los participantes conozcan y tengan los datos de graficas como la de **Mercado** o la de **Beneficios** a la hora de establecer sus precios. Si estas les ayudan en la definición y consiguen beneficios más altos o más óptimos. O incluso en una misma sesión, probar un número determinado de rondas sin ninguna ayuda, es decir sin mostrar dichas gráficas y a continuación, concederles dicha ayuda para que estimen los precios apoyados en algo.

Es cierto que esto último, es decir, varias las condiciones de la partida (funciones de coste y funciones de demanda) se pueden hacer en el transcurso de una sesión. Sin embargo, se recomienda no hacerlo ya que los usuarios aprenderán y afinarán más el precio óptimo al que vender al mercado tras unas cuantas rondas. De modo que si se varían las condiciones en mitad de la partida, aunque es posible, creará el desconcierto en estos. Por tanto reiteramos que si se quiere varias las condiciones de la simulación, se haga desde el inicio de la sesión.

También recomendamos que si finalmente se van a realizar cambios en las condiciones de simulación, el administrador advierta al resto de participantes ya que, aunque existen monitores en el interfaz de usuario que muestran las actuales funciones de demanda y coste, estos pueden no percatarse.

Capítulo 4.

Manual del programador

Este presente capítulo está dedicado a la exposición y explicación del diseño, desarrollo y construcción de la parte lógica de la aplicación. En otras palabras, se expone cómo se han codificado las distintas instrucciones necesarias para el correcto funcionamiento del mismo.

Detallaremos instrucción por instrucción todos los procedimientos, exponiendo para qué sirven o cuál es su finalidad. De este modo, el lector conocerá de primera mano cómo se ha conformado este proyecto.



4.1 Introducción

Como ya hemos introducido, la finalidad de este presente capítulo es la de mostrar y enseñar al lector cual y como se ha hecho la presente aplicación. El objetivo buscado con la introducción de este capítulo en el proyecto, es dar una idea sobre cual han sido los procedimientos utilizados, y el porqué de su utilización y función. Además de exponerlos, se busca que el lector conozca de primera mano y entienda todos estos procedimientos para que, en la medida que él desee, pueda modificar ciertos parámetros, funciones o valores y así darle la forma que el desee.

Ahora daremos unas pautas para un mejor entendimiento del manual.

Con el fin de aportar claridad en la exposición de los diferentes procedimientos que a continuación expondremos, estos vendrán expresados en letra cursiva. Mientras que los títulos o nombres de estos lo harán en letra negrita. La finalidad de este método es que el lector diferencie de un modo rápido y sencillo el procedimiento en sí, de las posibles explicaciones sobre estos.

Los comentarios que aparezcan en los procedimientos, vendrán precedidos por un punto y coma (;). Estos comentarios no forman parte del código, sino más bien son aclaraciones o comentarios, como el propio nombre indica, para hacer entender al lector el código de un mejor modo. Por otra parte se puede tratar de código que se ha “comentado” para que no sea ejecutado en ese momento, y así tenerlo para a posteriori, si se desea, introducir la funcionalidad que ejecutaría.

Todos los procesos introducidos en NetLogo presentan una misma o al menos similar forma. Estos procesos poseen todos un nombre para su identificación y llamada en, por ejemplo, un botón. Estos vendrán introducidos por un *to*. A continuación, se escriben las instrucciones como tal. Y, finalmente, se cierra el procedimiento con un *end*. Un ejemplo de esto anteriormente descrito sería:

To nombre

Líneas de código (Instrucciones)

(...)

end

Un procedimiento o código se puede ejecutar mediante un botón. Este aporta una buena forma de administrar el código. Un botón puede almacenar código. Por tanto, puede albergar un procedimiento o, por el contrario, guarda la



Aplicación multiusuario con NetLogo

4. Manual del programador

llamada del procedimiento para ejecutarlo una vez accionado este. También en los botones se pueden configurar otro tipo de cosas como de qué tipo son o a quien están dirigidos. Para ello hay que hacer clic derecho en él y editarlo.

Para editar o modificar el rango de valores de un deslizador, lo que deberemos hacer es clicar el propio deslizador con el botón derecho y editarlo.

Además, comentar que el propio NetLogo posee una serie de comandos primitivos que se podrán utilizar en la medida de la necesidad de estos. Para consultarlos y conocer su funcionalidad, se puede consultar, en la pestaña de ayuda, el diccionario de NetLogo en el que aparecen todos estos comandos.

4.2 Manual para el programador de la actividad “Modelo de competencia en precios con producto diferenciado”

En este apartado se presenta la guía para el lector que quiera tomar la función de programador de la presente aplicación. De este modo, gracias a este apartado, conocerá el programa y el código utilizado para modificar este como considere oportuno.

El código que a continuación se presentará, es el código que hace posible que la aplicación desarrollada funcione con total normalidad. Todas las funciones y posibilidades de ambos interfaces están descritos por el siguiente código. Cerciorarse de que todo el código que introduciremos estará contenido en la pestaña de Código.

En primer lugar, como en toda aplicación realizada con NetLogo, habrá que declarar y definir las variables globales de la aplicación. Estas variables denominadas globales (o *globals*), son un tipo de variables las cuales son necesarias y utilizadas por todo el programa para realizar o dar paso a la posible realización de los diferentes cálculos. Es decir, no son concernientes a un solo procedimiento. Nuestras variables globales declaradas son:



globals

```
[  
  rondas           ;; Número de rondas transcurridas.  
  cantidad-total  ;; Cantidad total en el mercado en función de los  
                  precios elegidos por cada usuario.  
  precio-total    ;; Suma de todos los precios introducidos por los usuarios.  
  lista-color     ;; Lista de colores disponibles para los usuarios.  
]
```

El siguiente comando definido es el comando *Breed*. La función de este es introducir o describir la “especie” de la que son pertenecientes las tortugas o agentes.

```
breed [students student]      ;; Nombre asignado al agente.
```

Como podemos ver, el comando viene definido por dos términos entre corchetes. El primero de ellos define el nombre del grupo del que son miembros los agentes, es decir, se trata del conjunto de los individuos (plural). Por el contrario, el segundo de los dos términos se refiere al nombre individual que recibe el agente. De este modo se podrá saber si un procedimiento va dirigido al conjunto de los agentes (plural) o por el contrario a cada agente por separado.

A continuación declararemos las variables concernientes a cada agente individual. Estas son propias de cada agente, por lo que serán distintas para cada uno de ellos.



Aplicación multiusuario con NetLogo

4. Manual del programador

students-own

[

user-id ;; Identificador único para cada usuario (nombre).

precio-provisional ;; Variable auxiliar.

precio-venta ;; Precio al cual el usuario vende en el mercado.

cantidad-venta ;; Cantidad vendida por cada usuario al mercado para el precio de venta seleccionado por este.

Ctotal ;; Coste total incurrido por el usuario para el precio de venta introducido.

beneficios ;; Beneficios obtenidos hasta el momento.

beneficio-ronda ;; Beneficio obtenido por cada usuario en la ronda actual.

enviado? ;; Variable booleana que indica si el usuario ha seleccionado un precio en la ronda actual.

]

El procedimiento **startup** referencia unas funciones preparatorias que son necesarias para la correcta inicialización del programa. Estas son de carácter automático, por lo que no es necesario pulsar ningún botón ni realizar ninguna acción para ejecutarlas. Para una mejor explicación dividiremos las instrucciones y explicaremos cada una punto por punto.

to startup ;; Procedimiento de ejecución inicial automática.

hubnet-reset

;; Gráfico único para cada usuario ;;

- 1.- *__hubnet-make-plot-narrowcast "Mis precios"*
- __hubnet-make-plot-narrowcast "Mis beneficios"*
- __hubnet-make-plot-narrowcast "Mercado"*
- __hubnet-make-plot-narrowcast "Beneficio"*

- 2.- *ask patches [set pcolor gray]*

crear-lista

end



Aplicación multiusuario con NetLogo

4. Manual del programador

- 1.- Estas instrucciones tienen la función de realizar gráficos específicos para cada participante en función de sus propios datos.
- 2.- Mediante esta instrucción, cambiamos el color de los *patches*, es decir, de las "casillas del visor 2D de negro a gris y realizamos una llamada al procedimiento **crear-lista**.

En el botón denominado **Reiniciar**, se ejecuta un proceso denominado con el mismo nombre. Como ya sabemos del capítulo anterior, este botón realiza (a través del proceso) la vuelta a los valores definidos por defecto.

```
to reiniciar           ;; Procedimiento para reiniciar la interfaz del usuario
                        (Llamada botón "Reiniciar").
1.-   set rondas 0
2.-   ask students
      [
        reiniciar-variables
        actualizar-interfaz-clientes
      ]
3.-   borrar-graficos
      enviar
end
```

- 1.- Torna las rondas a cero.
- 2.- Se pregunta/pide a los usuarios que realicen los procesos dentro definidos.
- 3.- Se realiza una llamada al proceso **borrar-graficos** y a **enviar**.



Aplicación multiusuario con NetLogo

4. Manual del programador

Reiniciar-variables es un proceso que, como su propio nombre indica, reinicia las variables tanto globales como específicas de cada usuario.

to reiniciar-variables ;; Procedimiento para reiniciar las variables usuario.

```
hubnet-broadcast "precio-al-mercado" 10
set precio-provisional 0
set precio-venta 0
set cantidad-venta 0
set Ctotal 0
set beneficios 0
set beneficio-ronda 0
set enviado? false
set label (word user-id " , " precision beneficios 2 " , " precio-venta)
set cantidad-total 0
set precio-total 0
end
```

Borrar-gráficos es un proceso que tiene la función de borrar todos los gráficos presentes en ambos interfaces (servidor y usuario).

to borrar-graficos ;; Procedimiento para borrar todos los gráficos.

```
1.- set-current-plot "Mis precios"
ask students [__hubnet-clear-plot user-id]
set-current-plot "Mis beneficios"
ask students [__hubnet-clear-plot user-id]
set-current-plot "Mercado"
ask students [__hubnet-clear-plot user-id]
set-current-plot "Beneficio"
ask students [__hubnet-clear-plot user-id]
2.- clear-all-plots
end
```



Aplicación multiusuario con NetLogo

4. Manual del programador

1.- Estos gráficos, al tratarse de gráficos específicos para cada participante, han de ser borrados uno a uno. Por tanto, y para poder realizar esta acción, hay que definir primero el nombre del gráfico al que nos estamos refiriendo, y acto seguido la acción de borrar como tal.

2.- Esta instrucción es una de las que se llaman primitivas. Su función es borrar todas las gráficas del interfaz servidor.

El proceso denominado **activar**, se activa a través del botón del interfaz servidor que tiene el mismo nombre. Este ejecuta las instrucciones comprendidas entre corchetes cada 0.1 segundos.

1.-

```
to activar           ;; Procedimiento de activación (Llamada botón "Activar").  
  
  every 0.1          ;; Ejecuta cada 0.1 segundos el comando entre corchetes.  
  
  [listen-to-clients] ;; Obtiene los comandos y datos aportados por los  
  usuarios.  
  
  if auto?           ;; Desarrollo automático de la sesión si dicha  
                     ;; funcionalidad está activada. No sería necesario pulsar  
                     ;; el botón "Siguiete ronda".  
  
  [  
    if not any? students with [enviado? = false]  
    and (rondas < num-rondas) [siguiete-ronda]  
  ]  
  
end
```

1.- Este comando comprueba si el interruptor denominado como **Auto?** esta en modo *On*. De ser así, ejecuta las instrucciones comprendidas entre corchetes. La funcionalidad de estas, como ya hemos visto es que las rondas avancen automáticamente si todos los participantes han enviado su cantidad. Como vemos llama al proceso **siguiete-ronda** que ahora definiremos.



Aplicación multiusuario con NetLogo

4. Manual del programador

El botón denominado en el interfaz servidor como **Siguiente ronda**, hace una llamada al proceso denominado con el mismo nombre. El objetivo de este es pasar o avanzar de ronda. Este acto deberá hacerse una vez se hayan enviado todas los precios por parte de todos los participantes. Lo que hace es avanzar de ronda, pero primeramente realiza los cálculos correspondientes y devuelve los valores “resueltos” a los usuarios mostrándoselos en los diferentes elementos dedicados a ello.

```
to siguiente-ronda      ;; Procedimiento que avanza de ronda. (Llamada
                          botón "Siguiente ronda" o procedimiento
                          automático)

1.-  set rondas (rondas + 1 )
      ask students

      [
2.-  realizar-calculos
      ]

3.-  dibujar-graficos

end
```

1.- Esta instrucción incrementa en uno las rondas existentes.

2.- Llamada al proceso **realizar-calculos**.

3.- Llamada al proceso **dibujar-graficos**.



Aplicación multiusuario con NetLogo

4. Manual del programador

El proceso *listen-clients* era un proceso que si recordamos se le llamaba desde el proceso *activar*. Se trata de un proceso estándar del módulo Hubnet. A través de él, se obtienen los datos y comandos aportados por los usuarios. Mediante este también se controlan acciones como la entrada de nuevos participantes a la sesión o la salida de estos. Esto se controla a partir de las llamadas a otros procedimientos.

```
to listen-to-clients
  while [ hubnet-message-waiting? ]
  [
    hubnet-fetch-message
    ifelse hubnet-enter-message?
1.- [ create-new-student hubnet-message-source ]
    [ ifelse hubnet-exit-message?
2.- [ remove-student hubnet-message-source ]
3.- [ execute-command hubnet-message-tag ]
  ]
]
end
```

1.- Se realiza la llamada al proceso *create-new-student*, el cual crea un nuevo participante.

2.- Se realiza la llamada al proceso *remove-student*, el cual elimina al participante que desee salir de la sesión.

3.- Se realiza la llamada al proceso *execute-command*.



Aplicación multiusuario con NetLogo

4. Manual del programador

Execute-command tiene el objetivo de obtener todos los datos suministrados por cada participante.

```
to execute-command [command]
```

- 1.- *if command = "precio-al-mercado" ;; Se actualiza la variable precio al mercado cada vez que se ajusta la barra deslizador.*

```
[ask students with [user-id = hubnet-message-source]
```

```
[if (enviado? = false)
```

```
[ set precio-provisional hubnet-message ]
```

```
]
```

```
]
```

- 2.- *if command = "Enviar precio" ;; Cuando se pulsa el botón "Enviar precio" se envía al mercado el precio seleccionado en la barra.*

```
[ask students with [user-id = hubnet-message-source] ;; Solo se puede enviar precios al mercado una vez por ronda.
```

```
[if (enviado? = false)
```

```
[ set precio-venta precio-provisional
```

```
hubnet-send user-id "Precio actual" precision precio-venta 2
```

```
set shape "face happy"
```

```
set enviado? true
```

```
]
```

```
]
```

```
]
```

```
end
```




Aplicación multiusuario con NetLogo

4. Manual del programador

1.- Con esta condición lo que se pretende es fijar un precio provisional. Es decir, en el momento en el que los participantes eligen un precio (a través del deslizador propio) la variable denominada *precio-provisional* adquiere un valor. Este se actualiza cada vez que el participante desliza o cambia el precio. Sin embargo esto sucede si aún no se ha clicado el botón **Enviar precio**.

2.- Esta condición comprueba que se haya pulsado el botón **Enviar precio**. Si se cumple, es decir, si clican el botón, el valor de la variable antes definida como *precio-provisional*, será el de la nueva variable denominada **precio-al-mercado**. El valor de esta variable será necesario para la realización de los próximos cálculos. Después de realizar y conferir el valor de la variable provisional a la definitiva, realiza los cambios de iconos para las simulaciones de los usuarios en el visor 2D (aparece el icono de la cara) y fija el valor de una variable denominada *enviado?* a *true*. Lo que se busca con esto último es que la condición primera no se pueda cumplir y por tanto el valor que marca el deslizador en ese momento no afecte a la variable y lo varía sin tener que ser así.

En este apartado conviene mencionar que si en la ronda posterior, el usuario no clicca el botón **Enviar precio**, el precio que se enviará al mercado dicha ronda será el último precio enviado. Resulta indiferente que se haya deslizado la barra o no. Ya que lo que realmente ejecuta la acción es la pulsación del botón, y no el movimiento del deslizador que solo varía una variable provisional.

Remove-student es el nombre que se le da al proceso por el cual se eliminan a los usuarios que desean abandonar la sesión. Mediante este procedimiento se borran todas las variables que tenían asignadas.

```
to remove-student [id]    ;; Procedimiento de eliminación de los usuarios
                             que abandonen la sesión.

  ask students with [user-id = id] [ die ]

end
```



Aplicación multiusuario con NetLogo

4. Manual del programador

Create-new-student es el nombre del proceso el cual añade un nuevo participante a la sesión. En este, se le asignan diferentes características y especificaciones necesarias para su identificación.

```
to create-new-student [id]           ;; Procedimiento de creación de un
                                       nuevo usuario.
1.-  let p one-of patches with [ count neighbors = 8 and not any? turtles-here
   and not any? neighbors4 with [ any? turtles-here ] ]
2.-  ifelse p = nobody
     [
       user-message "Todos los espacios en esta vista están completos. No se
       puede agregar más usuarios."
       stop
     ]
3.-  [ create-students 1
     [ set user-id id
       set shape "person"
       set color item 0 lista-color
       set lista-color but-first lista-color
       set label-color black
       setxy [pxcor] of p + 0.5 [pycor] of p
       hubnet-send user-id "Jugador" user-id
       reiniciar-variables
       actualizar-interfaz-clientes
     ]
   ]
end
```



Aplicación multiusuario con NetLogo

4. Manual del programador

1.- En esta parte del proceso se declara una variable auxiliar p que nos ayudará a posicionar la imagen o representación del participante en el visor 2D. La función que cumple esta variable auxiliar es la de seleccionar un *patch* vacío donde poder situar el icono o representación del participante, sin tener ningún otro cerca.

2.- En esta parte de la instrucción se comprueba mediante una condición si lo anteriormente dicho se cumple o no. Si no lo cumple, entra en esta condición y muestra el mensaje descrito.

3.- Si finalmente, existe una posición libre capaz de dar cabida y cumplir las anteriores condiciones, se creará un nuevo participante. A este nuevo usuario se le asignaran diferentes variables como la forma inicial, su color representativo... Como el color será algo representativo y por tanto se busca que sea único, se comprobará que esto sea así, eliminando cada color asignado de la lista de colores disponibles. Al final de este proceso se presenta llamadas a otros.

Realizar-cálculos será el proceso clave de este programa. En él se ejecutarán las instrucciones que traducen las variables de entrada, principal y únicamente el precio enviado por cada empresa, en la salida deseada o esperada.

to realizar-cálculos ;; Procedimiento de realización de los cálculos.

- 1.- `set precio-total sum [precio-venta] of students`
- 2.- `set cantidad-venta (Qo - (C1 * precio-venta) + (C2 * (precio-total - precio-venta)))`
- 3.- `set Ctotal (coste-fijo + (A * cantidad-venta) + (B * (cantidad-venta ^ 2)))`
- 4.- `set cantidad-total sum [cantidad-venta] of students`
- 5.- `set beneficio-ronda (precio-venta * cantidad-venta) - Ctotal`
- 6.- `set beneficios beneficios + beneficio-ronda`
- 7.- `set label (word user-id " , " precision beneficios 2 " , " precio-venta)`
- 8.- `set enviado? false`

realizar-graficos

actualizar-interfaz-clientes

end



Aplicación multiusuario con NetLogo

4. Manual del programador

A continuación se expondrá todas y cada una de las instrucciones de este proceso:

1.- En esta instrucción lo que se busca es introducir en la variable *precio-total*, la suma de todos los precios. En otras palabras, esta instrucción busca aunar todo el sumatorio de precios presente y posteriormente utilizado en la función de demanda. El valor del sumatorio de todos los precios se almacena en dicha variable.

2.- Se almacena el valor de la función de demanda en la variable *cantidad-venta*. En este momento se busca identificar la cantidad de producto que ha vendido u ofrecido al mercado cada empresa. Cerciorarse de que en esta instrucción es donde utilizamos la variable anterior utilizada, *precio-total*.

3.- En esta instrucción se calcula el coste de producción. Este viene definido por el valor de la variable *cantidad-venta* anteriormente definida. En función de esta, que previamente depende de los precios a los que las empresas deciden colocar su producto en el mercado, se obtiene el valor de coste de producción introducido en la variable *Ctotal*.

4.- Se calcula la cantidad total colocada por todas las empresas en el mercado en función de los precios escogidos. Se realiza la suma de todas las cantidades.

5.- En esta instrucción se almacena en la variable *beneficio-ronda* el resultado del cálculo del beneficio de cada empresa en función de su precio de venta, principalmente, y de su coste de producción.

6.- A continuación, en la variable *beneficios* almacenamos el beneficio acumulado gracias al sumatorio de todos beneficios anteriores más el que ha habido en la presente ronda.

7.- La función de esta línea de código consiste en actualizar las variables presentes en la representación gráfica del usuario contenida en el visor 2D. Se actualizan los valores existentes en beneficio y precio por los actuales.

8.- Se modifica la variable *Enviado?* a valor *false* para que de este modo vuelva a estar operativa para poder ser usada en el procedimiento ***execute-command***.

Acto seguido de esta última línea de código, se realizan dos llamadas a dos procedimientos más los cuales han de realizarse a la hora de realizar los cálculos.

Tras realizar los cálculos, es necesario actualizar diversos parámetros de cada usuario. Por tanto, esta es la función que desempeña el proceso ***actualizar-interfaz-clientes***. Este, como su propio nombre indica, actualiza y envía la información resultante de los cálculos realizados a los usuarios,



Aplicación multiusuario con NetLogo

4. Manual del programador

mostrándoselos en los diferentes elementos destinados para ello, como son los monitores o las gráficas. Como veremos en el procedimiento siguiente, se modifica la representación del usuario en el visor 2D, y se envía la información a los monitores deseados cuyos nombres aparecen comprendidos entre las comillas, y los valores que tomarán estos a continuación, en esas mismas líneas de código. En algunos casos añadiremos la precisión o dígitos que queremos que presente el monitor.

```
to actualizar-interfaz-clientes      ;; Procedimiento que envía la información
                                     actualizada a los usuarios.

  set shape "person"

  hubnet-send user-id "Jugador" user-id

  hubnet-send user-id "Ronda" rondas

  hubnet-send user-id "Beneficios acumulados" precision beneficios 2

  hubnet-send user-id "Beneficios última ronda" precision beneficio-ronda 2

  hubnet-send user-id "Precio actual" precision precio-venta 2

  hubnet-send user-id "Última cantidad" precision cantidad-venta 2

  hubnet-send user-id "Coste de producción" precision Ctotal 2

  enviar

end
```

También, como vemos, se realiza la llamada al proceso **enviar**. Este procedimiento se ejecuta o bien como vemos en la llamada de este procedimiento, o por el contrario cuando el administrador de la sesión clicla el botón **Enviar info**. Como sabemos de apartados anteriores, la función de este procedimiento es actualizar y enviar la información de las nuevas funciones tanto de costes como de demanda que se han definido. Pueden ser o bien las que ya estaban vigentes, o pueden ser unas nuevas definidas por el administrador.

```
to enviar      ;; Procedimiento que envía la función de costes a los
                 usuarios.

  hubnet-broadcast "Función de costes" (word "Ctotal = " coste-fijo " + " A "
  *Q" " + " B " *Q^2")

  hubnet-broadcast "Función de demanda" (word "Qi = " Qo " - " C1 " *Precio
  i" " + " C2 " *Σ Precio j")

end
```



Aplicación multiusuario con NetLogo

4. Manual del programador

Ya se han mostrado como se realizan los cálculos, es decir, a través de que procesos y con qué instrucciones, además de la forma de enviar y mostrar todos esos resultados en monitores. Sin embargo, falta el hecho de completar con la información las gráficas y dibujar los trazos que las comprenden. De esto se encarga el proceso denominado **dibujar-graficos**.

Explicaremos los apartados definidos en el código, aparentes en la página siguiente.

1.- Seleccionamos el gráfico “Beneficio”. Acto seguido consultamos a cada cliente, ya que este gráfico será propio para cada cliente. Definimos una variable auxiliar Z . Esta variable ira tomando todas las distintas posibilidades de precios a elegir. Después, eliminamos el gráfico previo para que no se sobrescriban los resultados. Introducimos la condición reflejada en el código, la cual mientras se cumpla, ejecutará el código contenido en su interior. Como ya hemos mencionado, como la variable Z tomará el valor de las distintas posibilidades de precios disponibles, estas no serán menores de 0 y mayores a 100, ya que no se pueden introducir precios mayores a 100 en la aplicación. Por tanto, mientras esta condición se cumpla, dibujara la gráfica correspondiente. Por ultimo incrementará el valor de la variable auxiliar en uno.

2.- La misma estructura de código posee la gráfica “Mercado”. Esta tiene todos los mismos aspectos que la gráfica anterior. Como resulta evidente, la condición y la función que traza la gráfica es distinta, ya que estas serán las propias para la gráfica “Mercado”. La condición introducida hace referencia a que la cantidad vendida al mercado no puede ser nunca negativa. Por tanto, mientras esta condición se cumpla, se ira representando la gráfica para los distintos valores que tome Q .

3.- En la gráfica “Mis precios” la estructura seguida es bastante sencilla. En esta se llama a la gráfica, y simplemente para cada usuario, representa los distintos precios escogidos en cada ronda.

4.- La gráfica de “Mis beneficios” se representa de igual forma que la gráfica de “Mis precios”.



to dibujar-graficos

```
1.- set-current-plot "Beneficio"

ask students[

let Z 0

__hubnet-clear-plot user-id

while [ Z <= 100 ]

[

__hubnet-plotxy user-id Z ((Z * (Qo - (C1 * Z) + (C2 * (precio-total - precio-
venta )))) - (coste-fijo + A * (Qo - (C1 * Z) + (C2 * (precio-total - precio-venta ))) + B
* ((Qo - (C1 * Z) + (C2 * (precio-total - precio-venta ))) ^ 2))

set Z (Z + 1 )

]

]

2.- set-current-plot "Mercado"

ask students [

let Q 0

__hubnet-clear-plot user-id

while [ (Qo - (C1 * Q) + (C2 * (precio-total - precio-venta ))) >= 0 ]

[

__hubnet-plotxy user-id (Qo - (C1 * Q) + (C2 * (precio-total - precio-venta))) Q

set Q (Q + 1 )

]

]

3.- set-current-plot "Mis precios"

ask students [__hubnet-plotxy user-id rondas precio-venta]

4.- set-current-plot "Mis beneficios"

ask students [__hubnet-plotxy user-id rondas beneficio-ronda]

end
```



El proceso **realizar-graficos** tiene el objetivo de dibujar y realizar los gráficos, como su propio nombre indica, pero esta vez de las gráficas que representan la evolución de cada participante. Es decir, ronda tras ronda han de ir actualizándose y preguntando a cada usuario por el precio que ha enviado y el beneficio que ha obtenido, y representando estos datos en las gráficas destinadas a ello en función de cada ronda.

to realizar-graficos

;; Representa los precios enviados por cada jugador en un mismo gráfico

1.-

```
set-current-plot "Precios"
```

```
create-temporary-plot-pen user-id
```

```
set-plot-pen-color color
```

```
plotxy rondas precio-venta
```

;; Representa los beneficios obtenidos por cada jugador en un mismo gráfico ;;

2.-

```
set-current-plot "Beneficios"
```

```
create-temporary-plot-pen user-id
```

```
set-plot-pen-color color
```

```
plotxy rondas beneficio-ronda
```

```
end
```

1.- En esta parte del código, escogemos el gráfico que vamos a necesitar o con el que deseamos realizar acciones, en nuestro caso "Precios". Una vez seleccionado, desarrollamos una plumilla de carácter temporal para cada uno de los participantes y la nombramos del mismo modo que tenga cada participante. Definimos un color para cada plumilla. Este color, como resulta evidente, será, a efectos de practicidad, el mismo que tenga asignado cada usuario, para definir y conocer de qué usuario se trata. Acto seguido a todas las definiciones, asignamos al eje horizontal (X) el valor de las rondas, y al eje vertical (Y) el de los precios.

2.- El procedimiento es el mismo que el tomado en la gráfica anterior, a diferencia de que en este caso, el eje horizontal (X) tomará el valor de las rondas, y al eje vertical (Y) el de los beneficios.



Aplicación multiusuario con NetLogo

4. Manual del programador

Para finalizar este capítulo, nos falta por mencionar el proceso que asigna los colores a cada participante. Este es el proceso **crear-lista**.

```
to crear-lista                ;; Procedimiento para crear la lista de colores
                                que se les asignarán a los usuarios.

  set lista-color (list red orange pink lime sky magenta
                        cyan turquoise green brown blue
                        red orange pink lime sky magenta
                        cyan turquoise green brown blue
                        red orange pink lime sky magenta
                        cyan turquoise green brown blue)

  ;; NOTA: Solamente podrá haber tantos clientes como colores en la lista.
  ;;
  ;;   En la lista actual hay 33 colores, suficientes para cualquier actividad
  ;;
end
```

Mención especial tiene la nota que aparece en él, la cual dice: *Solamente podrá haber tantos clientes como colores en la lista. En la lista actual hay 33 colores, suficientes para cualquier actividad.*

Capítulo 5.

Presupuesto

El tema a tratar en este presente capítulo es concerniente a detallar la inversión y recursos, ya sean materiales o humanos, necesarios para desarrollar una aplicación docente de este calibre. En él describiremos qué ha sido necesario para su desarrollo, así como la duración, materiales, mano de obra y costos incurridos. Tras este análisis se buscará hacer el balance y se ofrecerá un presupuesto veraz sobre la inversión que sería necesaria realizar por parte de cualquier entidad u organización interesada en esta aplicación.



5.1 Introducción

El presupuesto que en este tema se presentará, es concerniente al desarrollo de la aplicación docente realizada con el programa de NetLogo, Modelo de competencia en precios con producto diferenciado.

La aplicación ha sido desarrollada motivada por petición del departamento de Organización de Empresas y CIM de la Escuela de Ingenierías Industriales de la Universidad de Valladolid, más en concreto para la asignatura de Entorno económico y estrategia.

La aplicación se ha desarrollado de acuerdo a los requerimientos y especificaciones que el cliente, en este caso el profesor titular de dicha asignatura solicitó.

Los coste a los que se va a incurrir en este proyecto, como veremos, son muy distintos de los costes de otro tipo de proyectos como mecánicos o de edificación en los que una gran componente del coste o encarecimiento de estos es referente a los materiales o materias primas empleadas. En nuestro caso, la mayoría de los costes en los que caeremos, son relativos a costos de estudio, evaluación, concepción, desarrollo y prueba de la aplicación propiamente dicha. De la anterior afirmación se deriva que para conseguir este propósito es necesario contar con un equipo cualificado y competente para poder desarrollar el proyecto de la mejor forma posible.

A modo de guía, expondremos brevemente los apartados que recorreremos por este capítulo.

Como primer punto, describiremos las competencias y perfiles que necesita el proyecto para poder desarrollarse del mejor modo posible. En este punto describiremos que perfiles son más idóneos, que competencias debe poseer el equipo humano y que funciones desarrollaran a lo largo del proyecto.

A continuación, definiremos el alcance y duración del proyecto, para conocer con una mayor certeza en que momento necesitaremos que actividades. Gracias a la foto que nos reporta tener definido esto, podremos definir y calcular con mayor precisión el tiempo empleado en cada parte del proyecto, y así calcular los costes incurridos en él. Por último, a parte de los costes directos en forma de esfuerzo y tiempo empleado, calcularemos otros costes como son materiales, de amortización o indirectos, para llegar al presupuesto final de la realización de esta aplicación.



5.2 Perfiles, competencias y funciones del personal

Para el mejor desarrollo de este proyecto, se necesitan tres perfiles distintos, aportando las competencias propias acordes a su cualificación, y buscando esos matices necesarios en cada fase del proyecto.

5.2.1 Director del proyecto

Las funciones que la persona con este rol tendrá que desempeñar serán, en un primer momento, guiar y llevar las conversaciones, negociaciones, y acuerdos de todos los aspectos necesarios para la concepción del proyecto. Tendrá que recopilar toda la información y requerimientos del cliente de forma clara, para cumplir con el objetivo solicitado y deseos del cliente.

Acto seguido, deberá transmitir fielmente y de acuerdo a las aclaraciones del cliente, toda la información que recibió al resto del equipo. Deberá saber el número de personas o el esfuerzo que la tarea tendrá y saber qué momento del proyecto so más críticos para emplear una mayor carga de trabajo.

Una vez hecho esto, deberá coordinar el trabajo de todo el equipo a lo largo del proyecto, haciendo un seguimiento del avance del proyecto y conociendo de cerca el día a día de este. Deberá estar en contacto activo tanto con los trabajadores, conociendo los problemas que puedan surgir o los requerimientos que estos le hagan, y con el cliente informándole periódicamente del avance del proyecto y de las partes completadas.

5.2.2 Informático

La función principal de este, como no podía ser de otra manera, será desarrollar la aplicación de acuerdo a los requerimientos del director de proyectos. Realizará todo el desarrollo relativo a la aplicación informática como tal, desde el diseño conceptual, al diseño más propiamente técnico.

5.2.3 Administrativo

Su objetivo será el de realizar toda la parte relacionada con la gestión documental del proyecto. Es decir, la definición, escritura, la adaptación, etc. para generar todos los informes necesarios así como la memoria del proyecto.



Aunque como vemos se tratan de tres perfiles muy distintos, un Ingeniero de Organización Industrial podría desempeñar los tres perfiles y realizar las tres funciones gracias a la extraordinaria versatilidad que su cualificación le otorga. Sin embargo, a lo largo de este capítulo hablaremos de cada perfil por separado para definir mejor que competencias desempeñaría cada uno de ellos.

5.3 Etapas del proyecto

En este apartado dividiremos el proyecto en diferentes etapas y expondremos las tareas concernientes a cada etapa.

El presente proyecto podemos dividirlo en seis etapas o fases. En cada una de estas fases expondremos que tareas se han de realizar y que esfuerzo o carga de trabajo necesitan para ser completadas.

5.3.1 Definición del proyecto

En esta primera fase se constituye e inicializa el proyecto. En esta etapa, el director del proyecto, entabla las relaciones con el cliente a fin de conocer los requerimientos y especificaciones que este solicita para después poderlas implantar en el proyecto propiamente dicho. El administrativo realizará mientras tanto la organización y preparación de los documentos agregando las especificaciones para transmitir estas al resto del equipo con la claridad suficiente y sin ocasionar malentendidos, así como las líneas de actuación a seguir.

5.3.2 Estimación de recursos y tiempo

Toda vez que todo el equipo conoce que hay que hacer, se procede a definir el cómo y con qué medios o recursos se deben realizar o cuales son necesarios para ello. En este, se valorarán todas las opciones a emplear, así como los materiales que se necesitarán, el tiempo que se empleará, etc. para así establecer un calendario de entregas o entregables para la correcta información y supervisión por parte del cliente. En base a todo esto se le ofrecerá un presupuesto aproximado del conjunto del proyecto que el cliente



Aplicación multiusuario con NetLogo

5. Presupuesto

deberá validar. Cada persona involucrada en el proyecto, deberá participar en esta fase aportando que necesita en cada momento, cuando tardará en realizar sus tareas, etc. En definitiva, lo recomendable es que todos y cada uno de los actores empleados en realizar y desempeñar el proyecto deberán aportar sus ideas o pensamientos para dar más claridad y fiabilidad al alcance y presupuesto.

5.3.3 Desarrollo de la parte lógica

En esta etapa se diseñará el área conceptual del proyecto. Habiendo sido elegidas las alternativas a tomar y el programa a utilizar, el informático deberá realizar una primera maquetación o conceptualización de lo que será la herramienta en si y como esta será, siempre todo bajo la supervisión del director del proyecto. Después de esta conceptualización, se pasará a construir el programa.

5.3.4 Desarrollo de las interfaces

En esta fase el informático desarrollará las dos interfaces presentes (servidor y usuario) y necesarias para la aplicación, siempre con la ayuda y supervisión de director del proyecto. Esta parte será importante porque en ella se diseña lo que finalmente verá y utilizará el cliente por lo que estéticamente debe ser buena. Esta fase puede realizarse en forma correlativa con la anterior ya que ambas son concernientes al desarrollo de la aplicación, diferenciando a parte lógica de la parte “visual” y más funcional. Para que la ejecución en paralelo sea satisfactoria, el director del proyecto deberá analizar que tareas pueden hacerse simultáneamente y cuáles no, y diferencias que tareas son más críticas o urgentes que otras.

5.3.5 Puesta en marcha

Para esta fase, se deberá instalar en la red local del cliente el programa necesario, así como en todos los ordenadores que necesite. Será recomendable dar una pequeña guía o mostrar la experiencia en ejecución al cliente para que vaya familiarizándose con el programa, vea de lo que es capaz, y valide que tiene todo lo que exigjó.



5.3.6 Documentación

En la última fase del proyecto, se realizará la gestión documental. Esta constará de la realización de los manuales para el servidor y el usuario, así como manuales del programador, y toda la demás información necesaria. El administrativo será el encargado de realizar esta fase, aunque será necesario el resto del equipo para asesorar a este, sobre todo el director del proyecto, que más que asesorarle, le controlará y supervisará que el trabajo se realice del modo correcto. Esta supervisión será necesaria para comprobar que lo que se introduce en el manual está presente en el programa y viceversa.

5.4 Costes del proyecto

En el presente apartado abordaremos los costes incurridos en la realización de las diferentes etapas del proyecto.

Estos costes, aunque serán de todo tipo, los que prevalecerán y representarán el mayor porcentaje del total será los costes atribuidos al personal. Sin embargo, también ha existido un uso de material informático importante, con su correspondiente uso y desgaste.

Los costes en los que incurrirá el proyecto y que detallaremos a continuación son:

- Costo de personal
- Costo de amortización
- Costo de materiales
- Costes generales

En estos cuatro grupos agruparemos todos los costes o gastos incurridos desde el inicio del proyecto al fin de este.

Ahora bien, para determinar con exactitud cuál será el coste de personal incurrido y la amortización de las herramientas, primero debemos conocer y determinar cuál serán las tasas a abonar y/o amortizar. Además, para ello es necesario conocer y establecer cuál será el calendario laboral, es decir, que días el/los empleados deberán trabajar y cuáles no, así como las horas que empleará al día. En base a esto último, se podrá conocer con exactitud qué cantidad de tiempo y cuanto se podrá emplear para desarrollar el proyecto.



Aplicación multiusuario con NetLogo

5. Presupuesto

Gracias a toda esta información, podremos concretar las horas efectivas de las que se precisará y el alcance que tendrá. Todo ello para conseguir el coste incurrido por gastos de personal y de amortizaciones.

Para mostrar todo esto hemos tabulado toda la información necesaria para resolver esta cuestión en la tabla 8.1.

	Días
Año	365
Semana laboral	261
Fines de semana	104
Días vacaciones	22
Días festivos	12
Media días baja por enfermedad	42
Formación y otros	7
Total efectivo	178
Total horas/año disponibles (8h/día)	1.422

Tabla 8.1. Horas anuales disponibles.

5.4.1 Costo de personal

Como ya hemos mencionado en anteriores apartados, aunque los trabajos o tareas las hemos descompuesto en función de diferentes perfiles los cuales serían necesarios para tratar o desarrollar este proyecto, a la hora de calcular el presupuesto propiamente dicho, no lo haremos en función de estos. Como somos conocedores de que las diferentes funciones a realizar en el mencionado proyecto, aunque quizá existan perfiles más especializados, un Ingeniero de Organización Industrial es capaz de desarrollar todas y cada una de las funciones o tareas necesarias. Por tanto, en la hoja de presupuestos, y como veremos en la Tabla 8.2, en cuanto a costes de personal solo incurriremos en el coste de contratación de un Ingeniero en Organización Industrial.



Aplicación multiusuario con NetLogo

5. Presupuesto

REMUNERACIÓN	Ingeniero en Organización Industrial
<i>Nomina bruta (Recién graduado)</i>	20.000,00 €
<i>Seguridad Social c/Empresa (35%)</i>	7.000,00 €
Total	27.000,00 €
Total Coste Horario	18,99 €/h

Tabla 8.2. Coste de contratación.

Toda vez que tenemos calculada la tasa horaria a cobrar por el profesional encargado del desarrollo del proyecto, podemos valorar el coste del personal para el desarrollo de este.

Para obtener los datos numéricos referentes a los tiempos y duraciones de las diferentes fases del proyecto, nos hemos apoyado en referencias a proyectos de semejante magnitud o similares en cuanto a contenido, para así reflejar fielmente el posible alcance y no dar una fecha errónea o azarosa. Como resulta evidente, para obtener el coste, multiplicaremos el coste horario a percibir por el número de horas empleadas en el desarrollo del proyecto.

Aunque todo el trabajo lo realizará un Ingeniero de Organización Industrial, seguiremos descomponiendo las horas referentes a las diferentes tareas para los tres posibles perfiles escogidos, con el fin de conseguir un mejor ajuste. El presupuesto, como ya hemos dicho, será real y referente al individuo indicado, sin embargo en cuanto a descripción de las tareas, y descomposición de estas en horas y tiempos, será conveniente seccionarlos por los diferentes perfiles posibles.

En la Tabla 8.3 descompondremos en tiempos y expondremos las diferentes tareas a desarrollar a lo largo de todo el proyecto referentes al Director del proyecto. Se incluirán, como no puede ser de otro modo, las horas acaecidas en torno a las labores de supervisión ejercidas por este representante. Como veremos, el esfuerzo mayor se concentrará en la definición del proyecto.



Aplicación multiusuario con NetLogo

5. Presupuesto

ETAPAS DEL PROYECTO	Director del proyecto
<i>Definición del proyecto</i>	65 h
<i>Estimación de recursos y tiempo</i>	20 h
<i>Desarrollo de la parte lógica</i>	10 h
<i>Desarrollo de las interfaces</i>	7 h
<i>Puesta en marcha</i>	5 h
<i>Documentación</i>	5 h
Total horas	112 h

Tabla 8.3. Dedicación del director del proyecto en cada etapa.

En la Tabla 8.4 detallaremos el trabajo realizado por el Informático, centrándonos en la descomposición de las tareas y la asignación de las horas a cada una de ellas. Como veremos, el esfuerzo mayor se concentrará en la construcción del proyecto, referido a la parte lógica especialmente.

ETAPAS DEL PROYECTO	Informático
<i>Definición del proyecto</i>	-
<i>Estimación de recursos y tiempo</i>	15 h
<i>Desarrollo de la parte lógica</i>	210 h
<i>Desarrollo de las interfaces</i>	25 h
<i>Puesta en marcha</i>	18 h
<i>Documentación</i>	8 h
Total horas	276 h

Tabla 8.4. Dedicación del informático en cada etapa.

En la Tabla 8.5 detallaremos el trabajo realizado por el Administrativo, centrándonos en la descomposición de las tareas y la asignación de las horas a cada una de ellas. Como veremos, el esfuerzo mayor se concentrará en la documentación del proyecto.



ETAPAS DEL PROYECTO	Administrativo
<i>Definición del proyecto</i>	10 h
<i>Estimación de recursos y tiempo</i>	5 h
<i>Desarrollo de la parte lógica</i>	-
<i>Desarrollo de las interfaces</i>	-
<i>Puesta en marcha</i>	-
<i>Documentación</i>	120 h
Total horas	135 h

Tabla 8.5. Dedicación del administrativo en cada etapa.

Por último, en la Tabla 8.6 representaremos la conjunción de las tablas anteriores, traduciendo los tiempos empleados o a emplear en cada etapa por los costes en los que se incurrirá. Para ello, multiplicaremos el coste horario del trabajo por cada hora empleada.

ETAPAS DEL PROYECTO (COSTE)	Director de proyecto	Informático	Administrativo	Coste por etapa
<i>Definición del proyecto</i>	1.234,43 €	-	189,91 €	1.424,34 €
<i>Estimación de recursos y tiempo</i>	379,82 €	284,87 €	94,96 €	759,65 €
<i>Desarrollo de la parte lógica</i>	189,91 €	3.988,14 €	-	4.178,05 €
<i>Desarrollo de las interfaces</i>	132,94 €	474,78 €	-	607,72 €
<i>Puesta en marcha</i>	94,96 €	341,84 €	-	436,80 €
<i>Documentación</i>	94,96 €	151,93 €	2.278,94 €	2.525,82 €
Total horas	2.127,01 €	5.241,56 €	2.563,81 €	9.932,38 €

Tabla 8.6. Coste por etapa y rol.

5.4.2 Costo de amortización

En este apartado detallaremos los costes incurridos con motivo del uso de los diversos materiales necesarios para la consecución del proyecto. Estos costes, son los denominados costes de amortización. Los costes de amortización generados en este proyecto serán los referentes al uso de un ordenador personal con el hardware y el software que se especificará y que son necesarios y adecuados para el correcto desarrollo de la actividad.



Aplicación multiusuario con NetLogo

5. Presupuesto

El cálculo del coste de amortización que realizaremos de este producto (un ordenador), vendrá de realizar una amortización a tres años. La forma de amortización será de tipo lineal.

El cálculo de la cuota de amortización diaria que obtendremos y reflejaremos en la Tabla 8.7 vendrá determinada y calculada en función del precio total del ordenador y los diversos softwares empleados. Sumaremos la totalidad de los precios, y como lo amortizaremos a tres años, lo dividiremos entre los tres años para saber la cuota anual. Como una cuota anual no nos aporta gran cantidad de información, hallaremos la cuota diaria de amortización. Con esta cuota diaria, y sabiendo los días que utilizaremos los equipos informáticos, calcularemos que cantidad de la amortización total se adjuntará al proyecto y por tanto, aparecerá en el presupuesto de este.

EQUIPO INFORMÁTICO	Coste
<i>ORDENADOR con procesador Intel® Core™ i7-4500U, Tarjeta gráfica AMD Radeon™ R5 M240, Memoria RAM de 4GB y 750GB de Disco Duro</i>	650,00 €
<i>Microsoft® Windows 8.1</i>	34,99 €
<i>Microsoft® Office 2016</i>	194,99 €
<i>Total</i>	879,98 €
<i>Amortización diaria</i>	1,65 €

Tabla 8.7. Desglose de la amortización diaria.

En la Tabla 8.8 expondremos el coste total de amortización de los equipos informáticos en función de la utilización del proyecto. Para ser más exactos y como las tareas están descompuestas por perfiles, continuaremos con esta descomposición para reflejar más fielmente el nivel de utilización del equipo por parte de los diferentes roles.

Rol	Horas de utilización	Amortización horaria	Amortización debida al rol
<i>Director del proyecto</i>	112,00 h	0,21 €/h	23,11 €
<i>Informático</i>	276,00 h	0,21 €/h	56,94 €
<i>Administrativo</i>	135,00 h	0,21 €/h	27,85 €
<i>Total Costes de Amortización</i>			<i>107,90 €</i>

Tabla 8.8. Coste de amortización.



5.4.3 Costo de materiales

En este apartado detallaremos los costes incurridos por el uso de diversos materiales necesarios para el desarrollo del proyecto. En la Tabla 8.9 detallamos estos costes.

MATERIALES	Coste
<i>Material de oficina</i>	20,00 €
<i>Encuadernación y edición</i>	50,00 €
<i>Total</i>	70,00 €

Tabla 8.9. Coste de materiales.

5.4.4 Costes generales

En este apartado detallaremos los costes generales. Como costes generales nos referimos a todos aquellos costes indirectos y que no son consumidos por el proyecto pero que sin embargo son necesarios para su consecución. Véase, por ejemplo, alquileres, electricidad, comunicaciones... Todos estos costes se encontrarán tabulador en la Tabla 8.10.

CONCEPTO	Coste
<i>Alquiler de una oficina</i>	1.634,38 €
<i>Telecomunicaciones (Conexiones)</i>	196,13 €
<i>Electricidad</i>	98,06 €
<i>Otros</i>	150,00 €
<i>Total</i>	2.078,56 €

Tabla 8.10. Costes generales.



5.5 Cálculo del coste total

En este último apartado agrupamos todos los costes y ofrecemos el presupuesto total para la realización del presente proyecto. Estos se agruparán y mostrarán por concepto. Para la obtención de este presupuesto, a la suma de todos los costes anteriormente reflejados, hay que añadir el beneficio industrial deseado, en nuestro caso de un 15% y el IVA correspondiente (21%). Por tanto, este sería el presupuesto final a destinar para la realización del proyecto:

CONCEPTO	Coste
<i>Personal</i>	9.932,38 €
<i>Amortizaciones</i>	107,90 €
<i>Materiales</i>	70,00 €
<i>Costes generales</i>	2.078,56 €
<i>Presupuesto de ejecución</i>	12.188,84 €
<i>Beneficio industrial (15%)</i>	1.828,33 €
<i>Precio sin IVA</i>	14.017,17 €
<i>IVA (21%)</i>	2.943,61 €
PRECIO TOTAL DEL PROYECTO	16.960,77 €

Tabla 8.1. Desglose del presupuesto.

Capítulo 6.

Conclusiones

En este último capítulo recogeremos las conclusiones que de este proyecto se han podido obtener. Valoraremos si la aplicación cumple con los objetivos y requerimientos fijados en un primer momento. Revisaremos si los objetivos marcados, traducidos en una primera planificación, generaron las expectativas depositadas.

Se valorará de un especial modo si la herramienta o aplicación posee un valor inherente y si tendrá una buena y amplia utilidad final.



6.1 Conclusiones

En este presente capítulo trataremos de valorar si el proyecto cumple con los objetivos fijados en un primero momento, y en qué grado este los resuelve. Para ello, lo primero que tendremos que realizar será recordar qué objetivos nos marcamos y qué finalidad tendría esta aplicación.

Como describimos en la introducción, el objeto de este trabajo sería el de desarrollar e implementar una herramienta o instrumento docente que dé soporte a la instrucción de la asignatura de Entorno económico y estrategia y simule algunos de los problemas y modelos típicos que en esta asignatura se imparten. Todas estas acciones estarían encaminadas para que los alumnos adquieran de una manera más comprensible y dinámica, los principios de las teorías económicas, entendiendo y asimilando el motivo por el cual se desarrollan y suceden los diversos fenómenos económicos.

Para conseguir lo anterior, en un primer momento habría que definir cuestiones básicas de la economía, como son la oferta, la demanda, el mercado, el equilibrio en este... así como cuestiones un poco más específicas del propio proyecto, como son los oligopolios o el modelo de competencia en precios con producto diferenciado. Todas estas actuaciones están definidas dentro del marco teórico del proyecto. Por ello, la primera parte más teórica del proyecto, busca dar solución a las preguntas teóricas que al lector se le crean, y al tratarse de una aplicación de carácter docente, introducir y definir conocimientos teóricos económicos.

En una segunda parte, se presentó la realización de la herramienta o aplicación de dos formas y para los dos roles (usuario y programador). La aplicación multiusuario realizada con NetLogo, proporciona un gran servicio, especialmente para asignaturas de ámbitos económicos como lo es la asignatura de Entorno económico y estrategia, debido al carácter eminentemente instructivo de la plataforma. Un punto muy a favor de esta aplicación, es su versatilidad en el uso y utilización. Esto es así, debido a que el profesor puede utilizar la herramienta, en un primer momento para dar una visión aproximada de los modelos, que posteriormente de forma teórica van a estudiarse, o por el contrario la puede utilizar y mostrar a sus alumnos, después de las explicación magistral de los modelos teóricos, para refutar y poder corroborar que las experiencias teóricas son ciertas, y así mejorar la comprensión de estas.

Conviene mencionar también la ingente cantidad de trabajo que optimiza debido a que, el tratamiento de los datos de otro modo más “tradicional”, resultaría tediosa. Esta aplicación posibilita el procesamiento de datos así



Aplicación multiusuario con NetLogo

6. Conclusiones

como las conclusiones esperadas de un modo instantáneo, y para un buen número de participantes.

Para la consecución de este proyecto, ha sido necesaria la realización de una aplicación informática mediante el software NetLogo, además de la necesidad de la utilización del módulo que permite trabajar y participar de esta a un mayor número de participantes, HubNet.

En la introducción de esta memoria, se fijó y definió una hoja de ruta o planificación a seguir por el proyecto para la correcta o mejor consecución de este. Las etapas definidas y cumplidas para conseguir nuestra meta fueron:

- **Análisis previo acerca de los modelos que mejor se ajusten a la realidad**

Como primera tarea a ejecutar, debimos de informarnos y documentarnos de cual serían los modelos actuales que mejor se ajusten a la realidad.

Debían conocerse los modelos disponibles y posibles a estudiar o sobre los que centrar el proyecto, para así seleccionar de un mejor modo, el mejor modelo para la herramienta, o al menos aquel que ofreciera unos resultados más próximos a la realidad imperante.

- **Estudio y selección del mejor entorno de programación que consiga satisfacer las necesidades requeridas**

Toda vez que el modelo haya sido seleccionado y escogido, deberemos estudiar cómo este modelo puede ser traducido a una plataforma interactiva y como está será capaz de reproducir fielmente los resultados que el modelo debe reportar.

Tras consultar diversa documentación e informarnos de las diferentes posibilidades al respecto, la solución escogida fue la de utilizar un sistema basado en agentes, dado que es un sistema muy acorde a los requerimientos establecidos y, en adición, se pose multitud de información y documentación al respecto.

Habiendo elegido como metodología los sistemas basados en agentes, acto seguido se debía escoger de entre las múltiples posibilidades de softwares, aquella que encajara y se ejecutara del modo más semejante al modelo teórico posible. La herramienta ideal y que encajaba a la perfección con nuestras pretensiones era NetLogo.

NetLogo, como ya sabemos, es una plataforma o entorno de programación multiagente el cual posibilita el desarrollo de acontecimientos, simulaciones o fenómenos de diversa índole o carácter, ya sean sociales, naturales... Se trata de un entorno que debido a su versatilidad, es utilizado por multitud de



Aplicación multiusuario con NetLogo

6. Conclusiones

usuarios de diversa ocupación, ya sean investigadores que deseen desarrollar o simular algún fenómeno de cualquier carácter, profesores que necesiten de esta plataforma para enseñar a sus alumnos de una forma participativa los conocimientos adquiridos en las clases teóricas o incluso por los propios alumnos.

Ahora la pregunta que nos podríamos hacer sería ¿por qué NetLogo?

En la actualidad, existen multitud de herramientas y aplicaciones para dar solución a multitud de problemas y dificultades que la vida nos presenta. Los ingenieros no pueden limitarse a usar estas aplicaciones o herramientas, sino que deben ser los que las diseñen y construyan, o simplemente rediseñarlas y reconstruirlas para dar nuevas soluciones. Con NetLogo, esto es posible. Esto es así puesto que, cada individuo es más o menos capaz de coger un modelo existente, y transformarlo y rediseñarlo dando lugar a un modelo más o menos complejo, con el fin de ofrecer unas nuevas posibilidades.

En adición a esto, hemos escogido NetLogo por las siguientes razones:

- 1.- El lenguaje de NetLogo. Se buscaba un lenguaje de programación que no ofreciera muchas dificultades de aprendizaje y uso. El lenguaje que utiliza NetLogo es un lenguaje declarativo el cual no entraña demasiada dificultad a la hora de su interiorización y uso.
- 2.- Software gratuito. NetLogo no precisa de pago alguno para su utilización y se puede encontrar a libre disposición en su página web.
- 3.- Amplia ayuda. Cuenta con una gran y activa red de usuarios existentes en diversas comunidades, páginas web, foros, etc. lo que proporciona un gran soporte en caso de duda. También resulta posible ponerse en contacto con la Universidad de Northwestern (universidad donde se desarrolla la plataforma) en caso de duda mayor.
- 4.- Amplia información. NetLogo cuenta con amplia cantidad de información sobre su uso, así como manuales, video-tutoriales y una gran cantidad de modelos los cuales pueden ser consultado o probados para verificar el uso o funcionalidad de NetLogo.
- 5.- Librería de modelos. Como ya hemos dicho, NetLogo cuenta con una amplia biblioteca de modelos lo que nos brinda la posibilidad de utilizar estos como base para futuras modelizaciones.

Por tanto, tras haber seleccionado el programa (NetLogo), fue necesario conocer su lenguaje de programación así como sus amplias funcionalidades y



Aplicación multiusuario con NetLogo

6. Conclusiones

posibilidades. Este aprendizaje fue llevado a cabo a través de los diferentes manuales y videos existentes en su página web.

- Diseño conceptual de la aplicación informática

Una vez que se interiorizó y conoció el uso de la herramienta, sus funcionalidades y posibilidades, así como su lenguaje de programación, se debía diseñar el modelo y todos los aspectos relacionados. El diseño conceptual es una de las fases más importantes del desarrollo de un proyecto, debido a que es la etapa donde se definen los objetivos que deberán cumplir.

Se pueden distinguir dos fases distintas pero muy relacionadas dentro del diseño conceptual. Una sería la selección de los modelos a implantar, mientras que la otra haría referencia al diseño conceptual propiamente dicho. Está claramente relacionadas debido a que la segunda necesita imperiosamente de la primera para formarse.

Mencionar que en esta etapa fue necesario conocer las limitaciones que el software presentaba a la hora de diseñar la aplicación, debido a que una vez conocidas estas, se podrían buscar y encontrar soluciones a estos problemas.

- *Selección del modelo adecuado*

En esta fase debíamos estudiar el modelo y seleccionar aquellos fundamentos económicos o aquella conjunción de problemas que queremos observar, analizar y aprender a través de esta aplicación.

El modelo que necesitábamos y que ha sido estudiado ha sido el Modelo de competencia en precios con producto diferenciado. La base teórica de este modelo se encuentra descrita en el Capítulo 2: Fundamentos teóricos, de este presente proyecto.

- *Diseño conceptual de la aplicación informática*

En esta fase del desarrollo del proyecto, se trazaron las líneas de actuación a seguir para la correcta consecución del proyecto, y en especial, la buena trazabilidad de la aplicación. Se ajustaron los objetivos a cumplir con las necesidades o requerimientos necesarios.

- Realización de la aplicación a través de la plataforma de programación

El trabajo de esta etapa concierne a la implementación del modelo generado en la fase anterior. Se debía traducir fielmente las soluciones tomadas en la



Aplicación multiusuario con NetLogo

6. Conclusiones

fase anterior, así como las características del modelo económico al código o lenguaje del programa.

Mediante el lenguaje de programación se debía afinar en un alto grado el modelo teórico para así obtener los resultados esperados. Por ello, fue necesario un estudio de las posibilidades que se tenían para traducir fielmente nuestros deseos.

- Comprobación y verificación del programa

Esta etapa tenía como objetivo la comprobación y revisión del programa. En ella se revisó el programa, así como su funcionamiento, para de este modo depurar los posibles errores o problemas existentes. Se revisó el código con la finalidad de aclararlo lo más posible para el lector o futuro editor de este. Se probó que la aplicación funcionara correctamente en diferentes ordenadores y equipos.

En definitiva, se revisó tanto la parte más funcional del programa, como el código, depurando y solventando cualquier posible error presentado. También se trató de afinar la aplicación haciéndola más atractiva y más clara para los lectores, usuarios y posibles editores de este.

- Realización de la memoria pertinente

En esta etapa se realizó todo lo referente a la redacción y escritura de todo el Trabajo Fin de Grado. A modo de memoria, se escribió y anotó todo lo referente a la aplicación desarrollada.

El resultado de esta etapa es el presente documento. Este está estructurado y seccionado en seis apartados. En estos apartados se introduce la aplicación y el objeto del Trabajo Fin de Grado, se dan unas nociones básicas sobre economía y en especial se fundamenta teóricamente el modelo sobre el que se apoya la aplicación. A continuación se adjunta un manual para el usuario, diferenciando a este en función del rol que tomará durante la experiencia (administrador o usuario). Acto seguido se expone el manual del programador en el que se explica cómo se ha hecho la aplicación en sí. Por último se expone un pequeño presupuesto sobre los costes de realización del proyecto.



Anexo I

Licencia y copyright de la aplicación.

La aplicación Modelo de competencia en precios creada a través de la plataforma NetLogo, está basada en el programa llamado Oligopolios, el cual presenta como licencia la *GNU GENERAL PUBLIC LICENSE* versión 2.

Esta establece que el software es libre. Por ello, se puede redistribuir y/o modificar bajo los términos y condiciones que dicha licencia menciona. Como esta licencia permite lo anterior descrito, y además admite su nueva implantación o la de versiones posteriores de la misma, para nuestra aplicación estableceremos la ***GNU GENERAL PUBLIC LICENSE* versión 3**. Esta enmarca nuestra aplicación como libre, y por tanto puede ser modificada y redistribuida.

La finalidad de introducir y elegir esta licencia atiende a cumplir el objetivo que esta aplicación y proyecto tiene, y es el de servir como herramienta docente para todos los usuarios que lo deseen. Por ello se establece una licencia abierta, para que cualquiera pueda utilizar esta aplicación como método lectivo, o por el contrario, desee profundizar en ella o modificarla introduciendo nuevas funcionalidades.

El copyright de la aplicación será **Copyright (C) 2017 Guillermo Badás López**.

Para una mayor información sobre los términos y condiciones que dicha licencia establece, se podrá consultar esta a través del enlace descrito en el apartado bibliográfico.



Bibliografía

Anticona, A. J. (07/2017). *Competencia de productos diferenciados*. Monografías. Recuperado de <http://www.monografias.com/trabajos96/competencia-productos-diferenciados/competencia-productos-diferenciados.shtml>

Diccionario de la lengua española (06/2017). Recuperado de <http://dle.rae.es/>

GitHub (10/2017). *NetLogo Issues*. Recuperado de <https://github.com/NetLogo/NetLogo/issues/>

GNU General Public License (10/2017). Recuperado de <https://www.gnu.org/licenses/gpl-3.0.en.html>

Izquierdo, L. R. (07/2017). *NetLogo Quick Guide*. Recuperado de <http://luis.izqui.org/resources/NetLogo-4-0-QuickGuide.pdf>

Izquierdo, S. S. (2011). *Pequeño manual de introducción a HubNet*. Valladolid, España: Insisoc.

Kreps, D. M. (1994). *Curso de teoría microeconómica*. Madrid, España: MacGraw-Hill.

Merino, J. (2007). *Un programa para economía experimental*. Valladolid, España: Universidad de Valladolid.

NetLogo Users Group (09/2017). Recuperado de <https://groups.yahoo.com/neo/groups/netlogo-users/>

Nicholson, W. (8ª Edición). (2006). *Teoría microeconómica. Principios básicos y ampliaciones*. Madrid, España: Thomson Editores Spain.

Payette, N. (07/2017). *Unofficial features*. GitHub, Inc. Recuperado de <https://github.com/NetLogo/NetLogo/wiki/Unofficial-features>

Pindyck, R. S., Rubinfeld, D. L. (7ª Edición). (2009). *Microeconomía*. Madrid, España: Pearson Educación, S.A.

Poza, D. J. (07/2017). *Manual de NetLogo en español*. Insisoc. Recuperado de <https://sites.google.com/site/manualnetlogo/home>



Aplicación multiusuario con NetLogo



Sánchez Vera, M. J. (09/2017). *Características de la computadora cliente-servidor*. Blogger. Recuperado de <http://majusanchezv.blogspot.com.es/2016/05/111-caracteristicas-de-la-computadora.html>

Wilensky, U. (09/2017). *NetLogo*. Northwestern. Recuperado de <https://ccl.northwestern.edu/netlogo/>

