



ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

INGENIERÍA TÉCNICA EN INFORMÁTICA DE SISTEMAS

Curso Académico 2011/2012

Proyecto de Fin de Carrera

**FLEQ - Campeonatos online de preguntas
para motivar y mejorar el aprendizaje**

Autor: Arturo Moral Santiago

Tutor: Gregorio Robles



Esta obra se distribuye bajo la licencia **Creative Commons Reconocimiento 3.0 Unported**.
Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by/3.0/> o envíe una carta a Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

El conocimiento sin educación no es más que injusticia armada.

Quinto Horacio Flaco

Resumen

En la siguiente memoria describiremos un sistema, principalmente orientado a fines didácticos, con el que organizar y jugar campeonatos online de preguntas.

Basado en la prueba de concepto llevada a cabo en [1], el sistema se enfoca en ofrecer una herramienta educativa útil tanto para alumnos como para profesores.

Mientras que ayuda a los alumnos, convertidos en jugadores, a repasar los contenidos expuestos en clase de una forma efectiva, atractiva y divertida, ofrece a los profesores una potente herramienta con la que poder someter a auto evaluación a sus pupilos, sean el número que sean, sin tener que involucrarse más allá que en la puesta en marcha de un torneo. Elegir la fecha de comienzo de este, la categoría de las preguntas que se haga a los alumnos y el número de rondas de que consta serán unas de las pocas tareas en las que se verá implicado.

Los campeonatos se gestionarán a través de una aplicación web, donde los alumnos podrán apuntarse a un torneo previamente creado por su profesor y gestionar, por ejemplo, la hora a la que prefieren jugar una partida o la solicitud de revisión de una pregunta hecha durante una de sus partidas.

Las partidas se jugarán a través de un canal de chat donde un programa robot se encargará de hacer las preguntas oportunas a los jugadores, evaluar sus respuestas, dar pistas sobre la solución y conceder un punto al contrincante que antes acierte cada pregunta. Cuando uno de los dos jugadores obtenga la puntuación necesaria para ganar la partida, el mismo software robot se encargará de finalizar la partida, proclamar un vencedor y facilitar los datos necesarios al sistema para que se le sume un punto a la puntuación del ganador en la clasificación del torneo.

Tanto jugadores como profesores disponen también de una página web del proyecto donde encontrar documentación sobre el funcionamiento de todas las partes del sistema, instrucciones de uso, noticias y la opción de jugar allí mismo su partida sin tener que utilizar ningún otro programa ni abandonar la web.

Durante el proceso de desarrollo, el sistema ha sido probado por jugadores que, incluso en sus primeras etapas de madurez, han resaltado el espíritu competitivo que despierta en ellos. Se destaca especialmente la motivación personal que demuestran al conocer de antemano cuál será el tema de las preguntas que se realizarán en las partidas y prepararlas con ahínco para ser los vencedores. Si el jugador resulta vencido en una partida, preparará la próxima con más insistencia.

Índice

1. Introducción	5
2. Objetivos	7
2.1. Metodología empleada	8
3. Descripción informática	10
3.1. Arquitectura	10
3.2. Los subsistemas de software	11
3.3. El gestor de torneos	11
3.4. FLEQbot	12
3.5. Scripts de automatización de tareas	13
3.5.1. Python como lenguaje de scripting	14
3.5.2. Cron como planificador de tareas	15
3.6. Página web del proyecto	16
3.6.1. qwebirc	17
3.7. Control de versiones	18
4. Pruebas	20
5. Conclusiones	22
5.1. Logros principales alcanzados	22
5.2. Posibles trabajos futuros	23
5.2.1. Comunidad FLEQ	23
5.2.2. Internacionalización	24
5.2.3. Versión móvil	24
5.2.4. Integración como un módulo en Moodle	25
5.2.5. Estadísticas y minería de datos	25
5.3. Licencia	26
A. Eventos de una partida con FLEQbot	28
B. Encuestas de opinión a jugadores	29
B.1. Campeonato IARO y PTAVI	29
B.2. Primer mini campeonato	35

1. Introducción

La tarea del aprendizaje de ciertos conceptos que necesitan memorización en el colegio, el instituto o la universidad, en ocasiones es ardua para los alumnos. Demasiadas asignaturas, muchos conceptos distintos y no relacionados, tiempo que dividir a la hora de dedicarse a estudiar y aprender cada uno de ellos ...

¿Pero qué ocurriría si el estudiante se sintiese motivado por algo más allá que por el mero aprendizaje o aprobar la asignatura para olvidarse para siempre de lo preparado antes de una prueba de evaluación? Ante la respuesta a esa pregunta surgió la idea de introducir algo más allá de los métodos típicos de enseñanza y hacer que, tanto para profesores como para alumnos, todo sean beneficios.

Un grupo de profesores llevaba tiempo utilizando un software que, después de configurarse apropiadamente y lanzarse, se conectaba a un canal de chat y empezaba a hacer preguntas. Analizaba las respuestas recibidas, asignaba un punto al jugador que antes respondiese de forma correcta, facilitaba pistas si, pasado un tiempo, ningún jugador había acertado y, una vez que uno de los jugadores alcanzaba una cierta puntuación, le declaraba ganador de la ronda de preguntas.

El software funcionaba de maravilla y los alumnos respondían de forma muy positiva al método, pero tenía varios inconvenientes, como:

- La dependencia del profesor o alguien que lo manejase.
- La experiencia del usuario a la hora de utilizar IRC.
- No mantenía el estado de anteriores jornadas.

La tarea estaba clara, el esfuerzo invertido en crear un sistema que al menos solventase los inconvenientes anteriores merecería la pena y daría pie a que el método fuese utilizado por un mayor número de gente, profesores y alumnos, que aportasen experiencia sobre su uso y perfilaran aún más su funcionamiento hasta convertirlo en, ojalá, una herramienta didáctica presente en cualquier clase.

En [1] se dieron los primeros pasos para alcanzar el hito y, tras esa prueba de concepto, este proyecto de fin de carrera remata el trabajo.

Con la concepción de FLEQ (Free LibreSoft Educational Quizzbowl), los profesores ya no tienen excusas para utilizar este nuevo método de una manera fácil y potente para ellos y los alumnos, que se verán atraídos por los campeonatos de partidas que les enfrentarán unos con

otros respondiendo preguntas acerca de la asignatura.

En esta memoria se detallarán los requisitos que se tuvieron en cuenta a la hora de planificar el sistema, las tecnologías que se han utilizado para hacerlo realidad, los resultados de pruebas realizadas con alumnos reales externos al proyecto y los futuros trabajos relacionados con la idea que podrían llevarse a cabo.

2. Objetivos

Teniendo en mente las experiencias anteriores a su desarrollo, cuando una parte de las funciones implementadas eran realizadas manualmente por un profesor, se tiene claro que la motivación de los alumnos utilizando este método de repasar conceptos es alta. Los objetivos principales que se han marcado no ha sido tanto mejorar esa experiencia, que ya era demasiado buena, sino mejorar la forma de llegar a ella. Esto quiere decir, la forma en la que un profesor puede obtener con sus alumnos resultados iguales o mejores a los cosechados antes de esta solución y la forma en la que un alumno puede seguir disfrutando de una experiencia idéntica que, además, ahora aporta más funcionalidad.

Para definir los objetivos se ha tenido claro el problema al que nos enfrentamos, que es suplir las carencias de un método que ya se estaba poniendo en funcionamiento. Carencias como:

- La dependencia del profesor o alguien que lo manejase. El software no se ejecutaba autónomamente y tampoco se generaban de esta forma los archivos de configuración necesarios para cada tipo de partida.
- La experiencia del usuario a la hora de utilizar IRC, ya que no todos lo conocían y para algunos resultaba complicado el proceso de llegar a conectarse a un canal de cierta red IRC sin tener conocimientos previos.
- No mantenía el estado de anteriores jornadas. Aunque las partidas eran muy entretenidas, una vez terminadas todos volvían a empezar de cero, sin importar que previamente hubiesen ganado una o diez partidas de una misma temática. Eso hacía que reinase un ambiente de igualdad al comienzo de cada partida pero, por otra parte, no recompensaba correctamente a los jugadores que ganaban una partida, ofreciéndoles el mérito efímero de haber ganado pero sin reconocimiento posterior pasados los minutos o los días hasta la siguiente partida.

Dejando a un lado la efectividad, la diversión y lo atractivo que ya era antes el método, los objetivos principales que se han marcado durante el desarrollo del sistema han sido dotarlo de facilidad de uso y casi total autonomía.

- Fácil de usar: la facilidad de uso tiene que ser un requisito indispensable en el sistema. Tanto el profesor, en su rol de administrador de torneos, como los alumnos, en el suyo como jugadores, deberían encontrar el menor número de trabas, a poder ser ninguna, a la

hora de aprender cómo se utiliza el sistema, de manejar las herramientas necesarias para utilizarlo y de volver a repetir el proceso cuantas veces deseen. Que tanto alumnos como profesores empleen más tiempo utilizando el sistema que aprendiendo cómo hacerlo es una meta principal en la facilidad de uso.

Hay que tener en mente que el sistema está enfocado a alumnos y profesores de cualquier nivel educativo y de cualquier área de conocimiento. Así, los conocimientos mínimos de cada usuario tendrán que ser tomados como conocimientos muy generales y abstraer la mayor parte de detalles técnicos.

- (casi) Autónomo: siendo este objetivo más enfocado al rol del profesor que al del alumno, se centra en ofrecer una herramienta potente. Que requiera un tiempo ínfimo en comparación con los resultados que va aportar a quien lo utiliza para crear un torneo y que pueda, de nuevo, abstraerle de toda la complejidad que maneja el sistema y darle las tareas lo más prefabricadas posibles.

Un usuario administrador de torneos se centrará principalmente en preparar un fichero de preguntas y respuestas, quizá el trabajo más laborioso pero no más que el que debe hacer para cualquier tipo de evaluación en la que pruebe a los alumnos. Una vez preparado el fichero, solo tendrá que definir la fecha de comienzo del torneo y el número de rondas de las que consta y, salvo para consultar el estado del ranking de jugadores o responder a la revisión de una pregunta propuesta por un alumno, su trabajo habrá acabado. Cuando el del sistema con los jugadores acaba de empezar.

- Ubicuo: como mejora al sistema antiguo, se pretende que los torneos puedan celebrarse en horarios fuera de clase. Que los alumnos puedan elegir a qué hora juegan una partida y ello se convierta en una actividad que, aunque puede ser comentada en clase, no requiera tiempo en el aula y haga a los jugadores independientes de las horas lectivas a la vez que hace al profesor disponer de un tiempo extra de formación a sus alumnos del que se encargará FLEQ.

2.1. Metodología empleada

Cuando se empezó a trabajar en el proyecto, preparando al principio la prueba de concepto [1], el hito a alcanzar era obtener un sistema funcional. Lo básico para poder construir sobre algo que funciona, algo que además de funcionar atrae y posee mucha más funcionalidad por encima de la básica.

Para ello se empezó trabajando en el primer puntal de la plataforma, un IRC bot que se encargase de llevar a cabo la parte más fundamental del sistema: las partidas.

Blitzed Trivia era el software que se había estado probando hasta ese momento por los profesores para llevar a cabo partidas. Y aunque el proyecto llevaba varios años sin actividad por parte de ningún desarrollador, tenía una característica primordial para que eso no nos afectase: es software libre.

Con la libertad para acceder al código de Blitzed Trivia y poder modificarlo, se actualizó primero para que pudiésemos utilizar las herramientas actuales de compilación con él — después de casi diez años sin modificarse, había código obsoleto que ya no cumplía con los estándares del lenguaje de programación en el que estaba escrito, C++.

Conseguida la tarea de tener funcionando el software para llevar a cabo partidas, se empezó a pensar de qué manera se podría llegar a un sistema que automatizase el uso de ese software y ofreciese la funcionalidad necesaria para organizar campeonatos. Así es como se decidió crear un gestor de torneos.

El gestor de torneos, a través de una interfaz web, ofrecía a los usuarios la funcionalidad para crear un torneo o participar en él según el rol que desempeñasen.

Hasta ese punto, la prueba de concepto demostró una funcionalidad mínima del sistema y un punto sobre el que apoyar la próxima fase. El sistema funcionaba pero no ofrecía autonomía completa a los administradores de torneos, que tenían que depender del administrador de la plataforma para llevar a cabo tareas esenciales para ellos como subir preguntas al sistema o definir las características de un torneo.

Con el proyecto en su nueva fase, se han centrado esfuerzos en hacerlo completamente autónomo para cualquier tipo de participante, jugador o administrador de torneos, así como dotar a ambos grupos de un lugar donde poder obtener información adicional sobre cómo jugar, de qué trata el proyecto, futuras actualizaciones . . . del que antes no disponían.

Con la concepción del sistema como tres partes diferenciadas: el FLEQbot para realizar las partidas, el gestor de torneos y la página web del proyecto, los participantes tienen a su disposición todos los recursos necesarios para usar el sistema con los objetivos planteados en su concepción.

3. Descripción informática

3.1. Arquitectura

Toda la arquitectura de FLEQ se sustenta sobre el puntal del software para realizar partidas en IRC, el FLEQbot. Este fue inicialmente desarrollado utilizando el software Blitzed Trivia y decide cómo todo el sistema se ordena alrededor para organizar los torneos, ya que FLEQbot en solitario solo puede utilizarse para jugar una única partida.

La siguiente ilustración muestra la arquitectura final del sistema creada alrededor de FLEQbot y que dota al proyecto de la capacidad de organizar torneos autónomos.

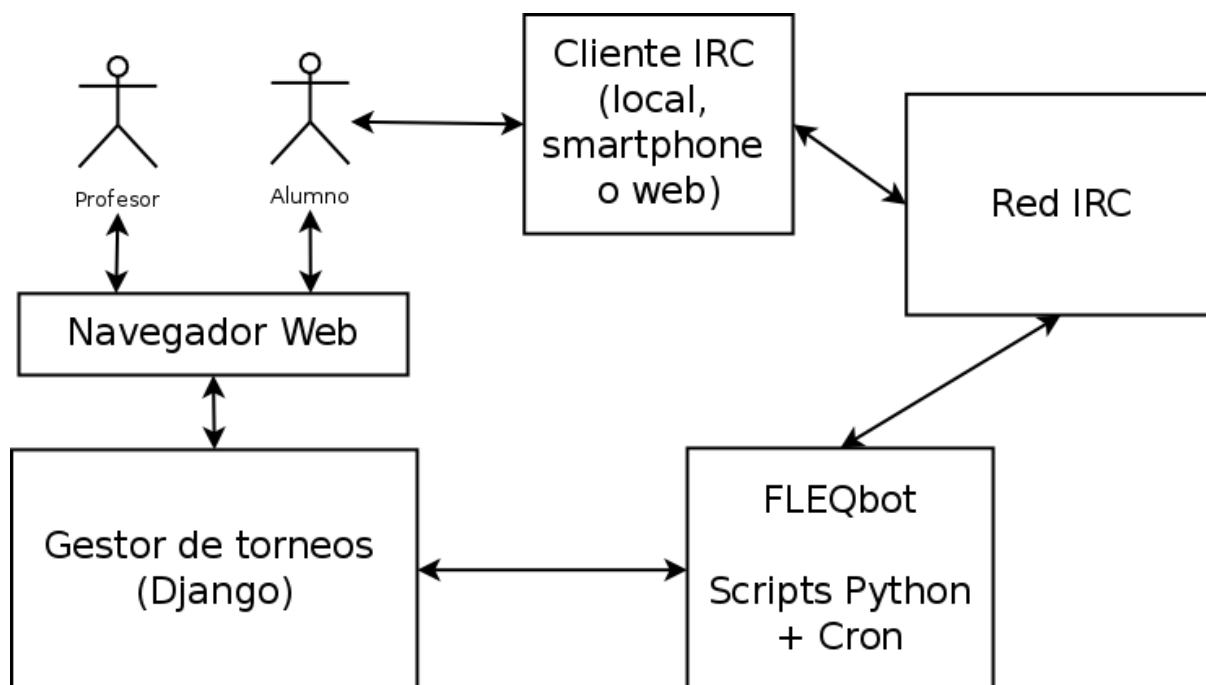


Figura 1: Arquitectura software creada alrededor de FLEQbot y que dota al sistema de autonomía.

Como puede verse, tanto el profesor como los alumnos acceden al gestor de torneos a través de un medio normalmente manejado por todos ellos con asiduidad, un navegador web.

Los jugadores utilizarán un cliente IRC, disponible para muchas plataformas diferentes, para acceder a la red IRC que se utilizará para jugar la partida.

Los scripts en Python y el planificador de tareas Cron se encargarán de recoger los datos necesarios tanto del gestor de torneos como de los logs generados por FLEQbot al terminar cada partida para hacer que el sistema mantenga su funcionamiento automático.

3.2. Los subsistemas de software

Los subsistemas de software sobre los que se sustenta el sistema y que soportan al software desarrollado se basan en soluciones de software libre:

- Linux: La distribución de GNU/Linux elegida es Debian en su versión estable actual, Squeeze.
- Apache: El popular servidor web de código abierto se encarga de proporcionar el acceso web tanto al gestor de torneos como al sitio web de FLEQ. Además de servir archivos como el código fuente del software del proyecto.
- SQLite: La ligera pero potente base de datos aporta facilidad a la hora de acceder a todos los datos almacenados en el gestor de torneos.
- Python: Es el lenguaje elegido tanto para el desarrollo del gestor de torneos (a través del framework Django) como para los scripts de automatización de tareas.

3.3. El gestor de torneos

Para el desarrollo del gestor de torneos se ha elegido Django, un framework de desarrollo web escrito en Python. El gestor es accesible desde la dirección <http://quizbowl.trivial.libresoft.es> y ofrece una interfaz web fácil de utilizar para los participantes.

Las funciones que el gestor ofrece a los participantes son:

- Registro en el sistema. Necesario para administradores de torneos y jugadores.
- Almacenamiento de preguntas para las partidas. Tarea que deberá realizar un administrador de torneos cada vez que quiera subir nuevas preguntas.
- Creación de nuevos torneos. Por el momento, solo permitido a administradores de torneo.
- Participación en un torneo. Permitido para participantes con ambos roles.
- Chequeo de detalles de torneo, incluido ranking de jugadores. Que podrá ser utilizada por jugadores y administradores de torneo.
- Chequeo de detalles de partida. Accesible para ambos roles.
- Elección de hora de preferencia para próxima partida. Opción únicamente disponible para jugadores y que les permitirá elegir una hora de preferencia de comienzo de partida que,

si llega a ser común entre los dos participantes de una partida, será la que prevalezca sobre una hora por defecto a la que todas las partidas comenzarán si no se encuentra consenso.

- Tramitación de solicitudes de revisión de pregunta. Que se diferencia si se accede desde un perfil de jugador, que es quien abre la solicitud y consulta su estado, o un perfil de administrador de torneo, que será quien aporte el texto de la solución.

El jugador deberá facilitar su identificador, el número que aparece antes del enunciado de cada pregunta durante la partida, a la hora de solicitar la revisión de dicha pregunta.

3.4. FLEQbot

Un fork de Blitzed Trivia, el software de partidas de preguntas a través de IRC escrito en C++, que nos permite configurar muchos de sus parámetros para adaptarlos al modo de juego elegido para las partidas.

Una vez lanzado junto con un archivo de preguntas y uno de configuración, el bot IRC es autónomo y el encargado de:

- Establecer los permisos de los usuarios del canal
- Invitar a los jugadores de una partida
- Hacer preguntas
- Facilitar pistas sobre la respuesta a la pregunta
- Comprobar si las respuestas de los jugadores son correctas
- Adjudicar puntos al jugador que acierte antes una pregunta
- Declarar vencedor al jugador que acierte antes un número determinado de preguntas

El formato del archivo de texto de preguntas, escrito en texto plano, que maneja FLEQbot y que es proporcionado por el script correspondiente antes de lanzarlo es el siguiente:

Cada línea contendrá una pregunta donde los campos entre corchetes son opcionales y el asterisco significa que ese campo puede repetirse ninguna o muchas veces.

```
categoría[, categoría]*;pregunta;respuesta1;[respuesta2;[respuesta3;[respuesta4;]]]
```

- categoría: Indica la(s) categoría(s) de la pregunta. Al menos deberá tener una y separarse con comas si son más.

- pregunta: El texto completo de la pregunta.
- respuesta1: La respuesta a la pregunta anterior. Será la que se muestre al jugador.
- respuesta2: Es opcional y una respuesta alternativa a la pregunta. Si se incluye, parte de ella se utilizará para dar pistas a los jugadores.
- respuesta3, respuesta4: También son opcionales y respuestas alternativas a la pregunta.

Un ejemplo de archivo con tres preguntas es:

```
acertijos;;¿De qué color era el caballo Blanco de Santiago?;Blanco;
acrónimos,memes;;¿Qué significa LOL?;Laughing Out Loud;Laugh Out Loud;Lots Of
Laughs;
arte,cultura general,personajes;;¿Cuál era el nombre de la mujer del pintor
Salvador Dalí?;Gala;Gala Dalí;Gala Eluard Dalí;Gala Éluard Dalí;
```

Aunque es posible configurar FLEQbot para conectarse a cualquier red IRC abierta, i.e. que sea accesible para cualquier usuario y sin restricciones que provoquen que el software no funcione correctamente, durante todo el proceso de desarrollo y pruebas se ha utilizado Freenode, la red IRC orientada al software libre, por ofrecer esta una alta disponibilidad y no tener ninguna restricción contra el uso de nuestro bot IRC.

Una vez ejecutado el software, este se conectará a un canal IRC, creado con un nombre aleatorio. Establecerá los permisos del canal para evitar que nadie se una a él salvo usuarios invitados y procederá a invitar a los participantes de la partida. En diagrama del apéndice A. describe esta secuencia de eventos.

3.5. Scripts de automatización de tareas

La combinación del uso de scripts con un planificador de procesos hacen que el sistema sea realmente automático.

Aunque algunos procesos se ejecuten por la acción de un participante a través del gestor de torneos, la mayoría de los procesos que mantienen el sistema funcionando son realizados, cuando llega su momento, por scripts.

3.5.1. Python como lenguaje de scripting

La elección de Python como lenguaje de scripting permite una muy buena integración con el framework de desarrollo web, Django, que utiliza el mismo lenguaje. Debido a ello es casi trivial acceder a funciones del gestor de torneos desde el exterior.

A continuación se resume la función de cada script que actúa en el sistema:

tournament_creator.py

Es el encargado de decidir si, el día de su fecha de comienzo, un torneo sigue adelante. La única condición para que no lo haga es que el número de jugadores inscritos en él sea menor de dos (este juego no es el solitario). Si el torneo no cumple la condición anterior, informa mediante una notificación por correo electrónico al administrador del torneo y al jugador, en el caso de que hubiese. Si, por el contrario, el torneo sigue adelante, informa a los jugadores inscritos en él de que este va a dar comienzo y crea las rondas correspondientes definiendo su fecha de comienzo y de fin.

round_creator.py

Se ocupa de comprobar qué rondas comienzan ese día e informar mediante correo electrónico a sus jugadores si participarán en ellas. La única condición para que un jugador no participe en las partidas de una ronda es que el número de jugadores del torneo sea impar. En este caso, un jugador por ronda elegido al azar permanece en ronda de meditación, obteniendo un punto de partida, y espera a la siguiente ronda, si la hubiese. El script empareja a los jugadores participantes de la ronda y crea las partidas convenientes, utilizando para ello las reglas del sistema suizo de competición.

player_reminder.py

Treinta minutos antes del comienzo de cada partida, envía un recordatorio a los jugadores a través de correo electrónico.

game_launcher.py

Toma nota de todas las partidas cuya hora de comienzo sea dentro de tres minutos o menos y prepara los archivos necesarios para que cada una de ellas se ejecute. Crea un archivo de preguntas seleccionando aquellas de la base de datos etiquetadas con la categoría de la que trate el torneo de la partida. Confecciona el archivo de configuración necesario para lanzar FLEQbot incluyendo los

datos necesarios y particulares de cada partida.

Por último, y después de almacenar los anteriores archivos en un directorio único para cada partida, lanza el script *game_watchdog.py*, que seguirá con el proceso de ejecución de las partidas.

game_watchdog.py

Haciendo uso de los archivos generados por el script anterior, lanza FLEQbot y aguarda a que termine su ejecución (signo de que la partida ha finalizado).

Después de parsear el archivo de log de la partida, lo almacena en el campo correspondiente de la base de datos y suma un punto de partida al ganador.

Por último, comprueba si la partida que ha terminado era la última del torneo y, si es así, envía un correo electrónico a jugadores y administrador informando de que el torneo en el que participaban ha finalizado.

Además de las notificaciones descritas en el funcionamiento de cada script, el sistema también notificará a los participantes de nuevos eventos cada vez que la hora de comienzo de una partida haya sido modificada (jugadores), una nueva solicitud de revisión haya sido abierta (administradores) o una revisión haya sido tramitada (jugadores).

3.5.2. Cron como planificador de tareas

Cron, el planificador de tareas que puede encontrarse en la mayoría de sistemas tipo Unix, es el encargado de lanzar los scripts del apartado anterior a las horas oportunas para que realicen su trabajo.

A continuación se detalla a qué hora y con qué frecuencia se lanza cada tarea:

00:01h - cada día - tournament_creator.py

La primera tarea que se ejecutará en el día. Esto es debido a que las demás dependen de ella en el caso de que el torneo no haya sido creado aún.

00:10h - cada día - round_creator.py

Si el torneo se acaba de crear, esta tarea se encargará de crear las partidas correspondientes a la primera ronda. Es por eso por lo que se distancia nueve minutos de la anterior tarea, para darle a su predecesora a terminar.

09:30h, 15:30h, 21:30h - cada día - player_reminder.py

Recordatorios a los jugadores media hora antes del comienzo de la partida.

09:57h, 15:57h, 21:57h - cada día - game_launcher.py

Encargada de lanzar la partida a las horas del día predeterminadas de comienzo de juegos.

3.6. Página web del proyecto

La página web del proyecto, accesible a través de la url `http://trivial.libresoft.es`, está programada utilizando HTML y una hoja de estilo para darle formato.

Su estructura es simple y directa para ofrecer un recurso de fácil manejo tanto para nuevos participantes como para los ya experimentados.

Tanto jugadores como administradores de torneo podrán encontrar información útil en sus diferentes secciones:

Qué es FLEQ Un espacio donde nuevos participantes pueden aprender sobre el proyecto. Leyendo acerca de qué se trata, cuáles son sus funciones ...

Reglas Donde tanto nuevos participantes como aquellos con experiencia que quieren repasarlas pueden ver un listado de las reglas que rigen en cada una de las partes de FLEQ.

Cómo jugar Una completa colección de *how-tos* donde se explica paso a paso cómo utilizar cada parte del juego.

Esta sección está orientada tanto a jugadores como administradores de torneo y, si un participante novel la recorre en orden, podrá llegar a aprender todo lo necesario para el uso de la herramienta.

Jugar La sección donde los jugadores que quieran acceder tanto a su partida como a cualquiera de los canales de Freenode pueden hacerlo sin tener que utilizar más medios que su navegador web.

Noticias Un lugar donde incluir las actualizaciones y noticias relacionadas con el proyecto. Los participantes registrados pueden consultarla asiduamente para estar al día del estado del proyecto y los nuevos participantes pueden leer las últimas noticias para tener una idea del timeline del sistema.

FAQ Una sección donde recoger las respuestas a las preguntas que más frecuentemente se hacen los participantes.

Aunque ya recoge casi una decena de respuestas a preguntas frecuentes, esta sección se irá ampliando con otras nuevas a medida que el sistema sea probado por más participantes y, de sus experiencias, se extraiga qué dudas en común suelen tener.

Uno de los atractivos que ofrece la página a los jugadores es *qwebirc*, el cliente web IRC que pueden encontrar en la sección *Jugar*.

3.6.1. *qwebirc*

Con la inclusión del cliente IRC *qwebirc* en el apartado *Jugar* de la página web, se da la posibilidad a los jugadores de llevar a cabo sus partidas sin tener que utilizar ningún otro recurso aparte de un navegador web capaz de utilizar Javascript.

Esto permite a usuarios poco experimentados con el uso de IRC acceder al chat en muy pocos pasos y sin la necesidad de instalar ni configurar un cliente IRC en su ordenador.

El *qwebirc* ofrece un cliente web muy fácil de utilizar y, aunque carece de opciones para mantener el estado de una sesión a otra (almacenar nick, password o canales a los que conectarse al comienzo) será suficiente para que, al principio, los jugadores no tengan como excusa la dificultad de usar el IRC para no participar en un torneo.



The image shows a web browser window titled "Connection details" with a light blue background. The main heading is "Connect to freenode IRC". Below this, there are several input fields and checkboxes:

- Nickname: [text input field]
- Channels: [text input field containing "#fleq"]
- Auth to services:
- Username: [text input field]
- Password: [text input field]
- reCAPTCHA: A box containing the word "influence" in a bold, lowercase font and "bytmncic" in a stylized, lowercase font. Below this box is a [text input field] and a [Reload] button.
- Audio captcha:
- [Connect] button

Figura 2: Pantalla de login de *qwebirc*, a través de la cual, rellenando los campos correspondientes, los jugadores acceden a una red IRC (Freenode en este caso).

```
[Welcome to FLEQ!]  
[18:37] <m0rti[0]> hola!  
[18:37] <g0to[0]> hola  
[18:37] <g0to[0]> listo para la partida?  
[18:37] <m0rti[0]> listo  
[18:38] <@FLEQbot> Welcome to FLEQ. A new game is starting (6  
questions in database). The game will commence in 15 seconds. First  
player to hit 3 points on score, wins!  
[18:38] <m0rti[0]> esto empieza  
[18:38] <g0to[0]> a por ello  
[18:38] <@FLEQbot> 77: ¿Qué parte del cuerpo distingue  
principalmente y en la mayoría de los casos a un mono y un simio?  
[18:38] <g0to[0]> la cola  
[18:38] <@FLEQbot> 1 points to g0to, who gave the correct answer La  
cola  
[18:38] <@FLEQbot> - Score -  
[18:38] <@FLEQbot> g0to[1], m0rti[0]  
[18:38] <m0rti[0]> uff, eres rápido  
[18:38] <g0to[0]> ;)
```

Figura 3: Captura de un momento de una partida llevada a cabo a través de *qwebirc*.

Después de las experiencias llevadas a cabo con jugadores, se observa que muchos, animados si se les guía en el proceso de instalación y configuración, dan el paso de utilizar un cliente IRC local y explotar su potencia no solo para jugar sus partidas, sino para utilizar cada vez más de vez en cuando el IRC y participar en los canales que en Freenode tienen a su disposición.

3.7. Control de versiones

Aunque no se empezó a utilizar hasta el pasado mes de octubre, el código fuente del proyecto y algunos archivos adicionales son respaldados por un sistema de control de versiones.

El software elegido para este propósito es Git y las principales características que su uso aporta son:

- Control de versiones no centralizado. Lo que permite mantener repositorios locales e independientes de máquinas remotas.
- Acceso al *log* de cambios desde que se creó el repositorio. Pudiendo acceder a las modificaciones que se han hecho de un archivo desde que se incluyó en el sistema de control de versiones hasta su estado actual.
- Fácil de compartir con otros desarrolladores que, sin en un futuro quieren unirse al proyecto, solo tendrán que descargar su copia del repositorio.

Actualmente se utilizan dos repositorios independientes, uno para el control de cambios en el gestor de torneos y otro para FLEQbot.

En un futuro, gracias al uso de Git, pueden aplicarse herramientas de minería de datos en ambos repositorios y obtener información sobre el proceso de desarrollo del proyecto.

4. Pruebas

Una de las ventajas que se han tenido durante el desarrollo de esta última fase del sistema ha sido poder contar con el feedback de las pruebas a las que el sistema se ha sometido utilizando jugadores externos a su desarrollo.

Tanto al terminar la prueba de concepto de la fase anterior como en un grado de madurez media de esta última fase, el sistema fue probado por un grupo de usuarios que lo usó y proporcionó el feedback necesario para saber si se estaba avanzando en la buena dirección y qué partes necesitaban modificarse si así no era.

Los usuarios de la primera prueba fueron algunos miembros del grupo de investigación LibreSoft, quienes se ofrecieron a jugar las partidas de un mini torneo que duró una tarde.

El sistema se encontraba a punto para la entrega de la prueba de concepto, así que las gestiones de creación del torneo, lanzamiento de partidas y creación de rondas e información a los usuarios de los pasos a seguir para utilizar el sistema fueron realizadas por mí mismo manualmente.

Los jugadores de la segunda prueba fueron los alumnos de las asignaturas Protocolos para la Transmisión de Audio y Vídeo en Internet e Información Audiovisual en Redes de Ordenadores, de las titulaciones Grado en Ingeniería en Sistemas Audiovisuales y Multimedia e Ingeniería de Telecomunicación respectivamente.

Los alumnos fueron animados por el profesor de las asignaturas, que les ofreció unas pautas mínimas sobre cómo acceder al sistema y, durante el desarrollo del campeonato, todos fueron asistidos a través de un foro de debate donde se resolvieron todas sus dudas.

Aunque el proyecto en ese momento carecía de una página web y las tareas de automatización para el administrador de sistemas aún no estaban terminadas completamente, se observó que, aunque se trataba de un número de jugadores mayor que en la primera prueba y con una duración del torneo de varios días, los alumnos seguían manteniendo su motivación y respondían muy positivamente al método con esfuerzos por mantenerse altos en el ranking de jugadores y batir a sus compañeros de clase en las partidas.

En el apéndice B se pueden encontrar los resultados de una encuesta de satisfacción que los jugadores de ambas pruebas rellenaron después del acontecimiento.

Cabe destacar que la mayoría de jugadores ha reportado un problema concreto durante las dos sesiones de prueba. Este problema, que en realidad no lo es y al que se da respuesta en la sección *FAQ* de la página web, lo encuentran al responder algunas preguntas y ver que en la pantalla de su cliente IRC, aunque su respuesta a la pregunta hecha por FLEQbot es correcta y aparece

antes que la de su contrincante, el punto se le otorga a su oponente.

El error no es tal ya que el cliente IRC imprime la respuesta del usuario local sin chequear si ha llegado al servidor ni cuando, lo que provoca que aunque el jugador vea en su cliente su respuesta antes que la de su compañero, es la otra la que ha llegado antes y, por lo tanto, la que se ha llevado el punto de la pregunta.

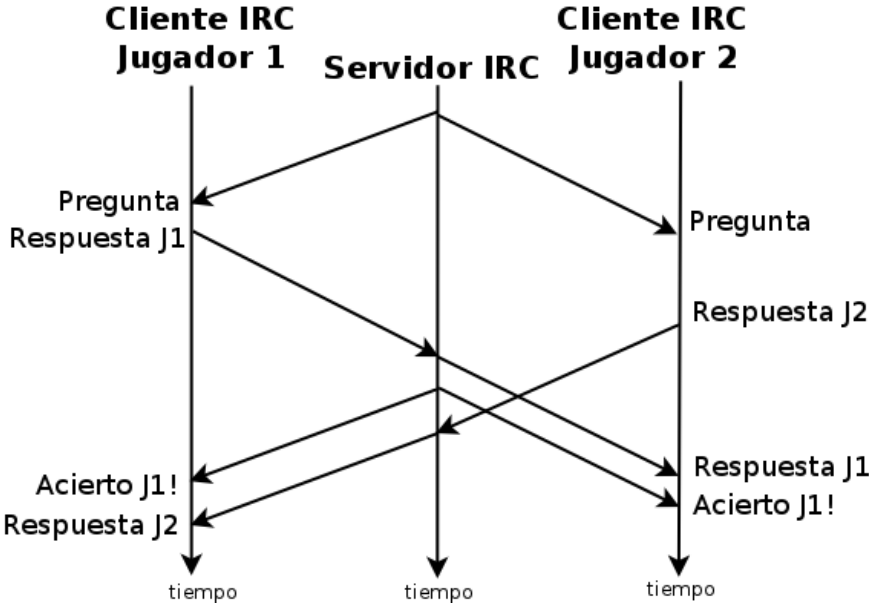


Figura 4: Efecto de falsa respuesta correcta recibida.

5. Conclusiones

Aún cambiando la manera en la que los estudiantes se implican en este nuevo método de enseñanza, su motivación se ha mantenido.

Cuando el sistema no existía y el profesor era el encargado de realizar todas las tareas de configuración y puesta en marcha de una partida, los estudiantes solían intervenir en esas partidas en horario de clase. Normalmente, siendo guiados por el profesor, que podía asistir in situ los problemas o cuestiones que tuviesen en ese momento. Pero tras implicar a los alumnos en el uso de FLEQ, haciéndoles más independientes y proactivos (ellos aprenden cómo funciona el sistema, cómo involucrarse, cómo participar ...) han respondido tan bien como anteriormente y no han dejado a un lado la actividad por ser más lo que tuviesen que entregar que lo que obtuviesen a cambio.

Aunque aún no se ha realizado ninguna prueba con profesores ajenos al proyecto e involucrados en áreas no relacionadas con la informática, el sistema ofrece un herramienta para ellos incluso más fácil de manejar que para estudiantes, por lo que se espera que su aceptación entre los docentes sea alta y que el boca a boca, no solo entre profesores sino de alumnos que lo han probado en otras asignaturas al resto de profesores, beneficie a su rápida expansión.

Tras la prueba de concepto expuesta en [1] el sistema ha evolucionado y ha pasado con creces su prueba de madurez.

El método está listo para su uso y, si es aceptado por la comunidad educativa, puede incluirse como una herramienta más de aprendizaje a través de Internet como han sido incluidas, no hace mucho, los foros de debate o los blogs.

5.1. Logros principales alcanzados

Como se ha indicado en el apartado anterior, el hecho de haber mantenido la atención de los estudiantes que ahora se involucran ellos mismos en la actividad sin tener a un profesor que les asista, es un gran logro.

El sistema ha aportado funcionalidad a la vez que ha aumentado dificultad, pero los jugadores han seguido motivados igualmente.

El sistema ofrece una herramienta que descarga de un gran trabajo a los profesores en comparación con la cantidad de alumnos a los que puede llegar a dirigirse.

Una vez preparadas las preguntas que se quieren realizar en los torneos que se organicen, en unos minutos un profesor podrá organizar torneos para cientos de jugadores dejando que sean ellos los que lleven a cabo toda la tarea. Si algún alumno quiere solicitar la revisión de una pregunta, sabrá que tiene el apoyo del profesor para solventarlo, pero el jugador se siente en

todo momento independiente y autosuficiente y en rara vez necesita algo que vaya más allá que sus propios conocimientos.

La obtención de un sistema casi del todo autónomo y en el cual los roles de los participantes no necesiten de nadie más para poder involucrarse es uno de los principales objetivos cumplidos. En comparación con la prueba de concepto expuesta en [1], el sistema ha conseguido que los administradores de un torneo puedan realizar ellos mismos todas las tareas relacionadas con la gestión y administración de sus torneos, aportándoles una independencia total del administrador de la plataforma.

5.2. Posibles trabajos futuros

Enfocado a un número tan grande de usuarios, las posibilidades futuras del proyecto son muchas.

Cada universidad, cada clase, cada alumno o profesor tendrá sus propias demandas sobre el rumbo que debería tomar el sistema para adaptarse a sus necesidades. Demasiadas para poder describirlas una a una, aunque a continuación se detallan algunas de las más generales y que podrían dar pie a que el sistema llegase a adoptarse por un número de usuarios muchísimo mayor gracias a ellas.

5.2.1. Comunidad FLEQ

Con el sistema funcionando y disponible como software libre, un gran paso para el proyecto sería crear una comunidad alrededor de él que no solo contribuya en el desarrollo del software, sino a mejorarlo en todos los sentidos.

Algunas de las herramientas con las que esta comunidad podría contar serían:

Un foro de debate a disposición tanto de jugadores como de desarrolladores. Con diferentes secciones donde aportar ideas y sugerencias, promocionar futuros torneos, compartir ficheros de preguntas ...

Una forja para desarrolladores. Podría utilizarse alguna de las más famosas, como Github o Sourceforge, o incluso aprovechar el uso de Git y una de las interfaces web que ofrece para crear una propia del proyecto que aglutinase el software que se vaya aportando a él.

Un sistema de clasificaciones permanente que convirtiese el juego en una competición constante. Que cada jugador acumule puntos en un ranking general, no solo en el campeonato que disputa, y que motive a los jugadores a participar en el mayor número de campeonatos para conseguir el mayor número de puntos y posicionarse en lo más alto de la tabla.

El mejor posicionado mensualmente, por ejemplo, podría recibir algún tipo de premio como recompensa y así motivar aún más a los demás a alcanzar su puesto y a él a mantenerlo.

5.2.2. Internacionalización

En relación directa con la idea de crear una comunidad, hacer que el sistema sea accesible internacionalmente podría aumentar en gran medida el número de jugadores.

Aunque el primer paso sería traducir al inglés toda la interfaz, en un futuro la idea podría atraer a jugadores de todas las nacionalidades que no dominen ni el español ni el inglés y necesiten una versión en su idioma para poder jugar u ofrecer el juego a sus conocidos.

Como parte de la comunidad, se podrían utilizar herramientas de traducción colaborativa y que cualquiera, sin necesidad de conocimientos de programación, pueda ayudar en la tarea de traducción de las diferentes partes del sistema.

5.2.3. Versión móvil

Que los participantes de un torneo puedan acceder a una versión del sistema para dispositivos móviles aumentaría aún más el número de ellos interesados en participar a la vez que haría más fáciles las cosas para los que ya utilizan el acceso a través de un ordenador.

La aproximación más sencilla para llevar a cabo la idea sería adaptar tanto la interfaz del gestor de torneos como la página web del proyecto para que se muestren correctamente en un dispositivo móvil (pantalla de reducidas dimensiones). Añadir, si hiciese falta, elementos que hiciesen más fácil la introducción de datos o la navegación (en ausencia de teclado y ratón). Y verificar que todo ello es compatible con todos o la mayoría de los navegadores de dispositivos móviles actuales.

Para el acceso al chat a la hora de jugar las partidas se podría recomendar uno de los clientes IRC para las distintas plataformas móviles que existen actualmente.

La opción más complicada pero más atractiva para el usuario sería empaquetar todo el sistema en una aplicación, donde los jugadores tengan acceso tanto al gestor de torneos como al cliente IRC. Se le podría añadir una parte donde residan las secciones más importantes de la página web del proyecto, como noticias o instrucciones sobre cómo jugar.

Como plataforma elegida para desarrollar esta aplicación, Android puede facilitar todos los recursos necesarios. Como medio de intercambio de datos entre la aplicación móvil y las diferentes partes del sistema se utilizaría JSON.

5.2.4. Integración como un módulo en Moodle

Moodle, la plataforma de enseñanza a través de Internet, cada día cuenta con más usuarios y entre sus posibilidades, tanto para estudiantes como para profesores, cuenta con módulos dedicados para, por ejemplo, foros, cuestionarios, encuestas, wiki . . .

La futura integración de FLEQ como uno de estos módulos y que los campeonatos online formasen parte del desarrollo de un curso a través de Moodle es una idea atractiva para no considerarla.

5.2.5. Estadísticas y minería de datos

Algunos profesores pueden conformarse con el feedback obtenido con la tabla de clasificación después de un campeonato, pero después de toda la información que contienen los logs de las partidas, resulta muy atractivo hacer un poco de minería en ellos y extraer datos más allá de quién es el ganador de cada partida.

Entre los datos que se podrían obtener están:

- Preguntas que más veces han sido acertadas
- Preguntas que más veces han sido acertadas antes de la pista
- Preguntas que más veces han sido acertadas después de la pista
- Preguntas que menos veces han sido acertadas
- Preguntas que menos veces han sido acertadas por los jugadores en las posiciones más altas del ranking
- Preguntas que más veces han sido acertadas por los jugadores en las posiciones más bajas del ranking
- Tiempo medio de duración por partida
- Tiempo medio de acierto de cada pregunta correctamente contestada
- Número medio de preguntas realizadas en cada partida
- Número de partidas no jugadas por ausencia de uno o ambos jugadores
- Horario de comienzo de partida más repetido

De este resumen, un profesor podría tomar nota de qué preguntas resultan más fáciles y más complicadas para los alumnos y suprimirlas o modificarlas en la base de datos. De igual forma, podría tener estos resultados en cuenta a la hora de realizar otro tipo de evaluación para los alumnos como, por ejemplo, un test escrito de preguntas.

5.3. Licencia

Una de las conclusiones más claras a las que hemos llegado es que sin la ayuda del software libre este proyecto no se hubiese podido llevar a cabo o ni siquiera haber nacido.

Si el autor de Blitzed Trivia no hubiese dotado a su proyecto de las libertades del software libre, el puntal principal de FLEQ no hubiese podido serlo y no podemos más que devolver el favor con la misma moneda. Todo el software desarrollado para este proyecto y del que él depende se distribuye bajo licencias de software libre con la esperanza de que un día también sea la llama que alguien pueda utilizar para realizar su idea sin miedo a quemarse.

Referencias

- [1] Arturo Moral Santiago. *Campeonatos de Trivial a través de IRC*. Trabajo de Fin de Máster, Universidad Rey Juan Carlos, 2011.
- [2] Gregorio Robles, Jesús M. González-Barahona y Arturo Moral. *A synchronous on-line competition software to improve and motivate learning: Organizing (almost fully) automated quizzbowls with FLEQ*. 2011
- [3] Mark Lutz. *Learning Python 3rd Edition*. O'Reilly, 2007.
- [4] Wikipedia
<http://www.wikipedia.org/>
- [5] Sitio web del proyecto Django
<https://www.djangoproject.com/>
- [6] Los sitios web *Stack Overflow*, *Super User* y *Server Fault* de la red de portales de soporte *Stack Exchange*
<http://stackexchange.com/sites>
- [7] Wikibook sobre \LaTeX
<http://en.wikibooks.org/wiki/LaTeX>

A. Eventos de una partida con FLEQbot

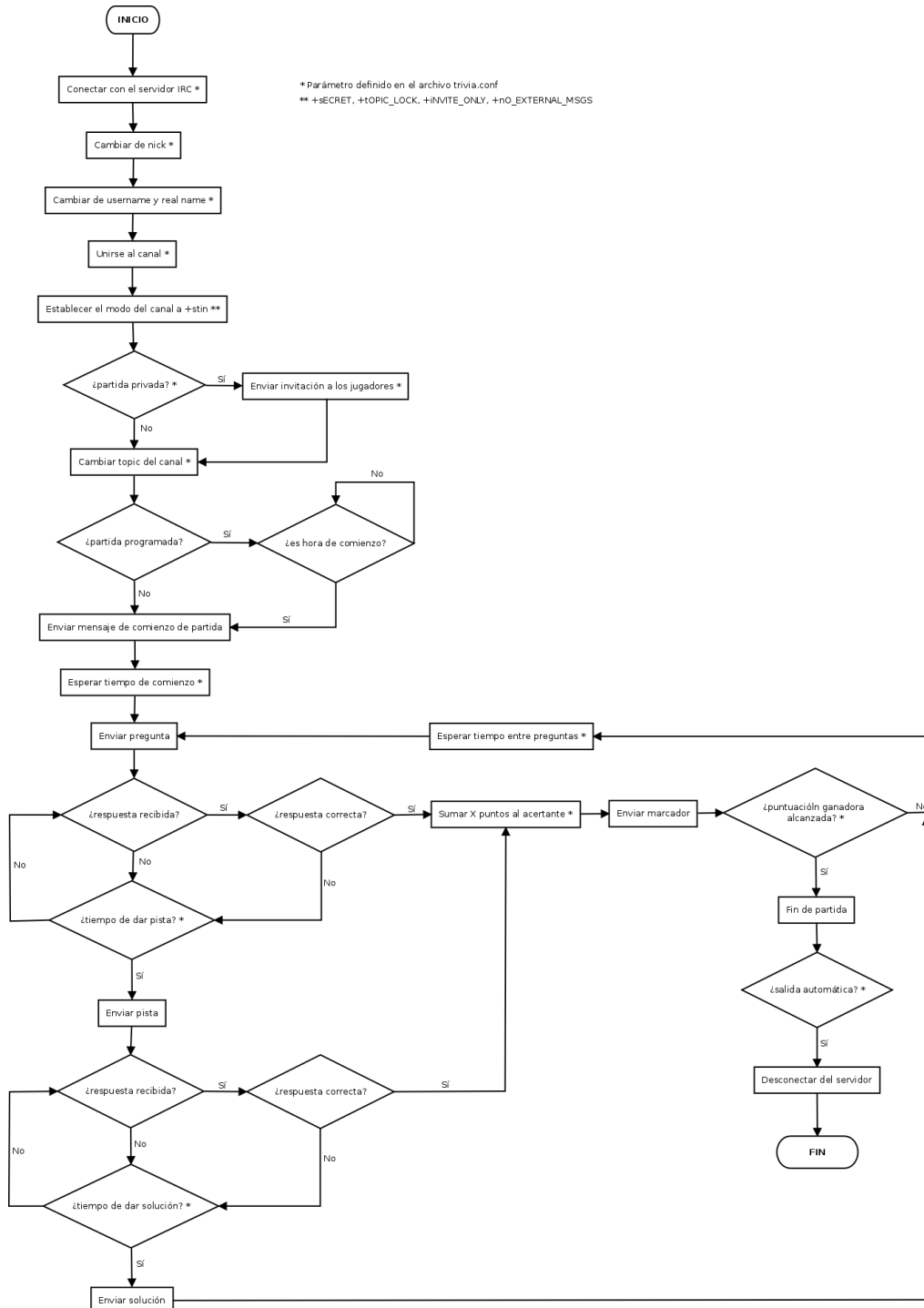


Figura 5: Diagrama del flujo de los eventos que se dan en una partida.

B. Encuestas de opinión a jugadores

B.1. Campeonato IARO y PTAVI

Fecha de fin del torneo: 18/09/2011

Número de jugadores: 14

Número de rondas/partidas por jugador: 3

Número de encuestas completadas: 6

1. El uso del portal me parece fácil e intuitivo		
Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	■□□□□□□□□□	16.67%
Ni de acuerdo ni en desacuerdo	■□□□□□□□□□	16.67%
De acuerdo	■□□□□□□□□□	16.67%
Muy de acuerdo	■□□□■□□□□□	50%

2. La posibilidad de cambio de hora de la partida de manera consensuada me ha parecido acertada		
Muy en desacuerdo	■□□□□□□□□□	16.67%
En desacuerdo	■□□□□□□□□□	16.67%
Ni de acuerdo ni en desacuerdo	■□□□□□□□□□	16.67%
De acuerdo	■□□□□□□□□□	16.67%
Muy de acuerdo	■□□■□□□□□□	33.33%

3. Los avisos a través de e-mail de partidas y demás información durante el campeonato me han parecido acertados		
Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	□□□□□□□□□□	0%
De acuerdo	■□□■□□□□□□	33.33%
Muy de acuerdo	■□□■□□■□□□	66.67%

4. ¿Qué añadirías, suprimirías o modificarías en el portal de gestión de torneos?

- El sistema de consenso de hora de juego; dar la posibilidad de conocer las prioridades de la otra persona y poder modificarlas una vez introducidas si no han coincidido en una primera elección (o añadir comentarios para que ambos participantes puedan concretar directamente en el portal).
- Saber el resto de partidas que se disputan (aunque esto es superfluo).
- Modificaría el sistema de puntuación a uno en el que se reflejen las respuestas acertadas, no solo las partidas ganadas.
- Me gustaria que hubiera habido mas partidas entre los jugadores ya que si perdias una de las tres rondas era dificil poder ganar el campeonato.
- Introduciria mensajes de ayuda como un boton de help
- Todas las horas disponibles, otra hora fija que no sea las 22:00 y mayor margen de tiempo antes de la partida.
- Quizá una interfaz un poquito más atractiva. Pero a efectos del torneo y de aplicaciones de este, no eché en falta nada.

5. Ha sido durante el campeonato cuando he utilizado por primera vez IRC

Sí 100 %
 No 0 %

6. Aprender el uso básico de IRC (acceder a Freenode, unirse a un canal, utilizar comandos...) no me ha supuesto demasiado esfuerzo

Muy en desacuerdo 0 %
 En desacuerdo 0 %
 Ni de acuerdo ni en desacuerdo 16.67 %
 De acuerdo 33.33 %
 Muy de acuerdo 50 %

7. El uso de IRC para jugar las partidas me parece acertado

Muy en desacuerdo 16.67 %
 En desacuerdo 0 %
 Ni de acuerdo ni en desacuerdo 16.67 %
 De acuerdo 33.33 %
 Muy de acuerdo 33.33 %

8. ¿Qué sistema alternativo propondrías para llevar a cabo las partidas?

- este sistema me parece apropiado para el fin de la actividad.
 - Alguno en el que la latencia fuera similar para todos
 - Cualquier red social, messenger, chat... Al fin y al cabo todo son sistemas de chat en tiempo real.. Me parece bien. Quizá whatsapp, o alguna aplicación móvil que sea muy usada ahora (aunque IRC lo tengamos ya para el móvil), pero se trata de ir siguiendo un poco lo que más utiliza la gente en general..

9. Me gusta la idea de dar una pista sobre la respuesta a una pregunta antes de darla por no contestada

Muy en desacuerdo	■□□□□□□□□□	16.67 %
En desacuerdo	□□□□□□□□□□	0 %
Ni de acuerdo ni en desacuerdo	□□□□□□□□□□	0 %
De acuerdo	□□□□□□□□□□	0 %
Muy de acuerdo	■□□□□□□□□□	83.33 %

10. Me gustaría que se añadiesen otras modalidades de juego además del uno contra uno, como juego por equipos o todos contra todos

Muy en desacuerdo	□□□□□□□□□□	0 %
En desacuerdo	□□□□□□□□□□	0 %
Ni de acuerdo ni en desacuerdo	□□□□□□□□□□	0 %
De acuerdo	■□□□□□□□□□	16.67 %
Muy de acuerdo	■□□□□□□□□□	83.33 %

11. El ritmo del juego me parece apropiado: tiempo dado para responder, tiempo entre preguntas, duración de la partida

Muy en desacuerdo	□□□□□□□□□□	0 %
En desacuerdo	□□□□□□□□□□	0 %
Ni de acuerdo ni en desacuerdo	■□□□□□□□□□	16.67 %
De acuerdo	■□□□□□□□□□	16.67 %
Muy de acuerdo	■□□□□□□□□□	66.67 %

12. Después de probarlo me he quedado con ganas de seguir jugando más rondas, competir con más gente y subir en la clasificación del campeonato

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	■□□□□□□□□□	16.67%
Ni de acuerdo ni en desacuerdo	□□□□□□□□□□	0%
De acuerdo	■□□□□□□□□□	50%
Muy de acuerdo	■□□□□□□□□□	33.33%

13. ¿Qué añadirías, suprimirías o modificarías en el sistema de partidas en IRC?

- Una segunda invitación de repesca por si algún participante ha llegado 1 minuto tarde o ha tenido algún problema con la conexión, por ejemplo.
- Que las partidas admitiesen "público"(usuarios conectados a la sala donde se disputa la partida pero que sólo pueden mirar)
- Nada.
- Mejoraría el sistema de puntuar.
- Quizás en ciertas respuestas sería bueno dar tres respuestas posibles a elegir
- suprimiría por supuesto la LATENCIA!!

14. Indica el cliente IRC que has utilizado para las partidas

- Cliente para Windows (Pchat), cliente web y cliente para iOS (MYIRC)
- PChat
- Bermejo7
- Freenode webchat
- Chustaman
- cliente local (pmoreno!?)

15. El campeonato en general me ha parecido un buen método de repaso y aprendizaje

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	□□□□□□□□□□	0%
De acuerdo	■□□□□□□□□□	16.67%
Muy de acuerdo	■□□□□□□□□□	83.33%

16. Recomendaría el uso de campeonatos como método de repaso en otras asignaturas de la carrera

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	■ ■ □ □ □ □ □ □ □ □	33.33 %
De acuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33 %
Muy de acuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33 %

17. Me gusta la elección del sistema suizo como método de emparejamiento de rivales en cada ronda (enfrentamiento de jugadores por sus puntuaciones)

Muy en desacuerdo	■ ■ □ □ □ □ □ □ □ □	16.67 %
En desacuerdo	■ ■ □ □ □ □ □ □ □ □	16.67 %
Ni de acuerdo ni en desacuerdo	□ □ □ □ □ □ □ □ □ □	0%
De acuerdo	■ ■ □ □ □ □ □ □ □ □	16.67 %
Muy de acuerdo	■ ■ ■ ■ ■ □ □ □ □ □	50%

18. Añade tu comentario libre sobre las cuestiones anteriores o cualquier tema relacionado

- Me gustaria que hubiera una eleccion de la hora de juego de las partidas al principio del campeonato para poder emparejar a la gente en relacion con la disponibilidad horaria que hayan elegido.

- La experiencia ha estado muy bien.

Las instrucciones de ingreso y demás eran un poco enrevesadas para lo sencillo que era en realidad entrar al IRC y al Trivial, y al principio faltó un poco de información clara sobre horas y días o forma de organizar el torneo, pero una vez dentro las partidas han estado muy curiosas, y Arturo ha estado muy atento a preguntas, necesidades, problemas, curiosidades, etc. También me parece interesante haber empezado a manejar un poco el IRC, que hasta ahora no conocía.

Como método de repaso, el picarnos entre nosotros siempre es un incentivo más al estudio que simplemente leer diapositivas.

El mayor problema para las partidas en mi opinión ha sido la latencia: como son preguntas de respuesta rápida, una mala conexión puede dejarte en desventaja.

- Una forma rápida y sencilla de mandar mensajes a tu rival para establecer la hora estaría muy bien

- Las horas de abajo, añadiendo algo del tiempo del día que estuvimos haciendo pruebas, ya que ahí me dí de alta, instalé cliente local y jugué..

19. Indica el tiempo aproximado en horas que te ha llevado participar en el campeonato, incluyendo repasar, darse de alta y las partidas

- 3.00

- 2.50

- 6.00

- 4.00

- 150.00

- 4.00

B.2. Primer mini campeonato

Fecha de torneo: 17/06/2011

Número de jugadores: 4

Número de rondas/partidas por jugador: 1

Número de encuestas completadas: 3

1. El uso del portal me parece fácil e intuitivo		
Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	■ ■ ■ ■ ■ ■ ■ □ □ □	66.67%
De acuerdo	□□□□□□□□□□	0%
Muy de acuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33%

2. No me importa que mis reclamaciones sobre alguna cuestión en una partida sean atendidas a través del portal en lugar de hacerse en persona o utilizando el correo electrónico		
Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33%
De acuerdo	■ ■ ■ ■ ■ ■ ■ □ □ □	66.67%
Muy de acuerdo	□□□□□□□□□□	0%

3. ¿Qué añadirías, suprimirías o modificarías en el portal de gestión de torneos?

- Más información relativa al tipo de torneo, número de jugadores y de rondas a jugar, resultado final numérico de cada eliminatoria, etc. Y más información durante el proceso de negociación de la hora de la partida.

- Aun está bastante verde. Por ejemplo, le añadiría menús para navegar más rápidamente.

También estaría bien que la sala en la que se disputa la partida apareciera en el portal, al menos las de cada usuario (que ya que está registrado...).

Se podría añadir una parte de estadísticas, como con el número de partidas ganadas, perdidas, etc, el face to face, comparativas entre gente, etc. Y si se pudiera analizar las respuestas del IRC mejor. Algo como número de promedio de preguntas acertadas por partida, número de respuestas dadas (correctas e incorrectas), etc

- Quizá tener un menú con las diferentes opciones (datos propios y de las próximas partidas). Además de añadir una página web amigable a los ojos.

4. El uso de IRC me parece fácil incluso para un usuario novel

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	■□□□□□□□□□	33.33%
Ni de acuerdo ni en desacuerdo	■□□□□□□□□□	33.33%
De acuerdo	□□□□□□□□□□	0%
Muy de acuerdo		

5. El uso de IRC para jugar las partidas me parece arcaico (ni siquiera tiene emoticonos de colores)

Muy en desacuerdo	■□□□□□□□□□	33.33%
En desacuerdo	■□□□□□□□□□	66.67%
Ni de acuerdo ni en desacuerdo	□□□□□□□□□□	0%
De acuerdo	□□□□□□□□□□	0%
Muy de acuerdo	□□□□□□□□□□	0%

6. Me gusta la idea de dar una pista sobre la respuesta a una pregunta antes de darla por no contestada

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33 %
De acuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33 %
Muy de acuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33 %

7. Me gustaría que se añadiesen otras modalidades de juego además del uno contra uno, como juego por equipos o todos contra todos

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	□□□□□□□□□□	0%
De acuerdo	■ ■ ■ ■ ■ ■ ■ ■ ■ ■	100%
Muy de acuerdo	□□□□□□□□□□	0%

8. El ritmo del juego me parece apropiado (tiempo dado para responder, tiempo entre preguntas, duración de la partida)

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33 %
De acuerdo	■ ■ ■ ■ ■ ■ ■ ■ □ □	66.67 %
Muy de acuerdo	□□□□□□□□□□	0%

9. Después de probarlo me he quedado con ganas de seguir jugando más rondas, competir con más gente y subir en la clasificación del campeonato

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	□□□□□□□□□□	0%
De acuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33 %
Muy de acuerdo	■ ■ ■ ■ ■ ■ ■ ■ □ □	66.67 %

10. ¿Qué añadirías, suprimirías o modificarías en el sistema de partidas en IRC?

- No se debería dar por buena tan sólo la primera respuesta correcta recibida, sino también aquellas respuestas correctas que son recibidas dentro de un período de tiempo razonable. Esto podría paliar los efectos de la conexión a Internet que tenga cada usuario.

En relación a esto, también sería bueno que un jugador no pueda ver las respuestas del otro, ya que el primero podría haber respondido con alguna pequeña errata, dándose la respuesta por mala, y obteniendo una pista importantísima el otro jugador, que bien podría no tener ni idea de la respuesta en un principio.

También debería limitarse el número de respuestas incorrectas de un jugador ante una determinada pregunta, puesto que en caso de no saber la solución, se puede responder por prueba y error hasta obtener la respuesta correcta.

- Lo más importante es que las preguntas y respuestas estén bien hechas. El otro día, había preguntas que había que acertar exáctamente (ej. Richard Stallman) pero en otras no (ej. GPL2 valía con poner v2 ¿?). En las preguntas donde dicen que des el nombre de dos cosas, parece que sólo vale con una. Las ayudas también habría que cuidarlas porque a veces despistan.

- Falta limar algunas asperezas, como respuestas que funcionen correctamente con diferentes opciones: Richard Stallman o Stallman, por ejemplo.

11. Me gusta la elección del sistema suizo como método de emparejamiento de rivales en cada ronda (enfrentamiento de jugadores por sus puntuaciones)

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	■ ■ ■ ■ ■ ■ ■ □ □ □	66.67%
De acuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33%
Muy de acuerdo	□ □ □ □ □ □ □ □ □ □	0%

12. El uso de alertas a través de e-mail me parece correcto y no excesivo

Muy en desacuerdo	□ □ □ □ □ □ □ □ □ □	0%
En desacuerdo	■ ■ ■ □ □ □ □ □ □ □	33.33%
Ni de acuerdo ni en desacuerdo	□ □ □ □ □ □ □ □ □ □	0%
De acuerdo	■ ■ ■ ■ ■ ■ ■ □ □ □	66.67%
Muy de acuerdo	□ □ □ □ □ □ □ □ □ □	0%

13. Utilizando los contenidos de la asignatura como preguntas, aprobaría que una parte de mi nota en esa asignatura dependiese de mi clasificación en un torneo

Muy en desacuerdo	□□□□□□□□□□	0%
En desacuerdo	□□□□□□□□□□	0%
Ni de acuerdo ni en desacuerdo	■ ■ ■ ■ ■ ■ ■ ■ ■ ■	100%
De acuerdo	□□□□□□□□□□	0%
Muy de acuerdo	□□□□□□□□□□	0%

14. Añade tu comentario libre sobre las cuestiones anteriores o cualquier tema relacionado

- Aprobaría que una actividad de este tipo sirviese para subir nota o como propuesta al alumno para que repase de cara a un examen, pero nunca que fuese parte de la evaluación ya que entran en juego muchas variables que el alumno y el profesor no puede controlar: retardos en la red, velocidad de tecleo, memorización de las preguntas-respuestas almacenadas en la base de datos, ayudas que un alumno pueda obtener de otros compañeros que ya hayan jugado su ronda y conozcan preguntas ya realizadas, etc.
- Sobre la última cuestión, aclarar que si la nota se pone en base a que se conteste más rápido a una pregunta me parece injusto. Que alguien tarde más en responder no significa que no sepa. Si se quiere hacer esto las preguntas también deberían estar muy bien, aunque supongo que eso ya lo tenéis en cuenta.