



EJERCICIOS PROPUESTOS

Ingeniería de Software

Integrantes del Equipo

Cruz González Daniel – Navarro López Rodrigo

Manuel – Rodríguez Balcazar David

Instituto Tecnológico de Celaya

6 de febrero de 2013

EJERCICIO 1. JUEGO DE ADIVINAR

Se requiere desarrollar una aplicación para un teléfono móvil donde participan dos jugadores cada uno desde su propio celular. Uno de los jugadores crea la partida y otro solo se conecta a ella. El juego consiste en que la aplicación le da una palabra a uno de los jugadores y este debe hacer un dibujo relacionado con la palabra de tal manera que al enviarle el dibujo al otro jugador, este trate de adivinar la palabra según el dibujo. Tiene tres intentos y si lo logra en el primer intento gana 100 puntos, si lo adivina en 2 intentos tiene 60 puntos y si lo adivina en el tercero gana 30 puntos, y si no adivina la palabra no gana ningún punto. Para terminar una partida el jugador que creo la partida debe darle en terminar ronda y con esto quien tenga mayor número de puntos es el ganador.

ACTORES.

Jugador A

Nombramos “Jugador A”, a la persona que inicia el juego o mejor dicho crea la partida a la cual se podrá unir otro jugador.

Jugador B

El “Jugador B” es la persona que una vez ya creada la partida, se podrá unir a la partida para jugar contra el jugador que creo la partida.

Sistema

El sistema es el encargado de dar palabras al azar y enviar los dibujos.

ESCENARIOS.

Nombre del escenario: Crear una partida

Actores participantes: Jugador A

Flujo de eventos:

- 1.- Supongamos que el Jugador A descarga el juego y se dispone a jugar pero él desea crear su propia partida
- 2.- El usuario A entra a la aplicación y en el menú principal accede a la opción “Crear partida”.
- 3.- Una vez que eligió crear una partida, la aplicación le pide que le dé un nombre a la partida, también le pregunta si quiere que sea una partida abierta o por contraseña.
- 4.- Elegir una partida abierta significa que cualquier jugador podrá ser el Jugador B, pero si elige con contraseña, la aplicación sólo dejara ser Jugador B al usuario que tenga la contraseña de la partida.
- 5.- El jugador A elige ya sea partida abierta o con contraseña y entonces la partida ha sido creada.

Nombre del escenario: Jugar

Actores participantes: Iniciado por el Jugador A

Jugador B

Flujo de eventos:

- 1.- El Jugador A espera a que algún usuario se una a la partida.
- 2.- Un Jugador B decide entrar a la aplicación y acceder a la opción de “Unirse a una partida”.
- 3.- La aplicación le pide al Jugador B ingresar el nombre de la partida y si resulta que la partida tiene contraseña, la aplicación también pedirá la contraseña para unirse a la partida.
- 4.- Si los datos son correctos el Jugador B se habrá unido a la partida.
- 5.- Los usuarios comienzan a jugar siendo el Jugador A quien sea el primero en dibujar.

CASOS DE USO.

Nombre del caso de uso: Crear una partida

Actores participantes: Iniciado por el Jugador A

Flujo de eventos:

- 1.- El Jugador A decide crear la partida por lo que accede a la opción de crear partida.
- 2.- Le da un nombre a la partida, así como también elige la forma en la que se conectará a la partida, ya sea con contraseña o acceso público.
- 3.- Elige la opción “Terminé de crear mi partida” y eso lo manda al menú principal en el cual ya deberá de estar la partida creada.

Nombre del caso de uso:	Unirse a una partida
Actores participantes:	Iniciado por el Jugador B Aceptado o no por el Jugador A
Flujo de eventos:	1.- El jugador B accede a la aplicación. 2.- Elige opción “Unirse a una partida” 3.- Ingresa el nombre de la partida. 4.- Ingresa contraseña. 5.- Accede a la partida

Nombre del caso de uso:	Turno de dibujar
Actores participantes:	Iniciado por el sistema. Jugador A y Jugador B
Flujo de eventos:	1.- El sistema asigna una palabra al azar al jugador de quien es el turno ya sea el Jugador A o B. 2.- El jugador hace un dibujo elusivo a la palabra. 3.- Elige la opción “Terminar y Enviar dibujo”. 4.- Espera respuesta del otro jugador.

Nombre del caso de uso:	Turno de adivinar
Actores participantes:	Iniciado por sistema Jugador A y Jugador B
Flujo de eventos:	<p>1.- El sistema envía el dibujo al Jugador en turno.</p> <p>2.- El jugador revisa la imagen y escribe una palabra de lo que piense que es el dibujo.</p> <p>3.- Envía la palabra.</p> <p>4.- El otro jugador recibe la respuesta y decide si es correcta o incorrecta.</p> <p>5.- En caso de ser correcta acumula los puntos dados y si es incorrecta y falla en los 3 intentos, acaba el turno del jugador y no recibe puntos.</p>

Nombre del caso de uso:	Ver estadísticas del juego.
Actores participantes:	Iniciado por Jugador A o B
Flujo de eventos:	<p>1.- El jugador elige la opción ver historial de juego.</p> <p>2.- Ahí se muestra información como cantidad juegos ganados, jugados, perdidos, empatados y todos los puntos ganados.</p>

REQUERIMIENTOS FUNCIONALES.

- Un jugador no puede empezar a dibujar si no están los dos jugadores en la misma partida.
- Un jugador debe esperar a que el otro dibuje para poder adivinar.
- Cada jugador solo tendrá 3 intentos para adivinar la palabra.
- Si lo adivina en el primer intento el jugador gana 100 puntos.
- Si lo adivina en el segundo intento el jugador gana 60 puntos.
- Si lo adivina en el tercer intento el jugador gana 30 puntos.
- Si el jugador no adivina la palabra entonces su turno de adivinar termina y no gana ningún punto.
- Las palabras deben ser aleatorias.
- Un nombre de partida debe ser único y al tratar de agregar una partida nueva, el nombre deberá ser verificado para que no haya nombres de partidas iguales.

- Un jugador puede administrar máximo 5 partidas a la misma vez.
- Un jugador puede estar unido a 5 partidas a la misma vez.
- Al descargar la aplicación, el teléfono debe acceder a tu número telefónico para registrarte.

REQUERIMIENTOS NO FUNCIONALES.

- La aplicación requiere de internet para poder estar al día y registrar los movimientos en la cuenta del usuario.
- La aplicación debe cumplir con un tamaño.
- El tiempo de acceso a la aplicación debe de ser de acuerdo con lo establecido por las normas.
- La aplicación en si debe ser gratuita.

EJERCICIO 2: SALA DE DISCUSIÓN

Se requiere desarrollar un sistema que simule una sala de discusión virtual. Los usuarios pueden ver los temas a discutir. El tamaño de la sala tiene un máximo de 10 integrantes. Al creador de la sala o tema a discutir se le llama Manager. El manager nombra a un Moderador que se encarga de dar la palabra o turno. Los usuarios sólo pueden opinar cuando se les da la palabra y lo harán por medio de mensajes como una sala de chat. El Manager y el Moderador son los encargados de dar cierre al tema en discusión. En cualquier momento un usuario puede consultar el estado del sistema, por ejemplo los encuentros planeados y su información. Cualquier usuario puede entrar a la sala de discusión virtual mientras la sala no esté llena.

ACTORES.

Usuario

Son los usuarios que se encuentran en la sala de discusión y pueden opinar cuando se les asigne un turno.

Moderador

El moderador es un usuario pero este no opina, sino que sólo se da a la tarea de dar la palabra a los usuarios para hablar.

Manager

Es la persona que crea el grupo de discusión y da el tema a discutir.

ESCENARIOS.

Nombre del escenario: Registrarse

Actores participantes: Usuario, Moderador y Manager.

Flujo de eventos:

- 1.- La persona elige la opción Registrar.
- 2.- Llena un formulario de sus datos personales, elige un nombre de usuario y una contraseña.
- 3.- Llena un captcha.
- 4.- Se registra.
- 5.- Un correo es enviado para verificar sus datos.

Nombre del escenario: Discutir

Actores participantes: Iniciado por el Manager

Moderador

Usuario.

Flujo de eventos:

- 1.- El manager crea la partida o sala de discusión.
- 2.- Elige un tema y lo asigna.
- 3.- Elige un moderador el cual es uno de los usuarios que se unen a la sala.
- 4.- Los usuarios se van uniendo uno por uno.
- 5.- Discuten sobre el tema y al final el moderador o manager concluye la sala de discusión.

CASOS DE USO.

Nombre del caso de uso: Crear una sala de discusión

Actores participantes: Iniciado por el Jugador A

Flujo de eventos:

- 1.- El Jugador A decide crear la partida por lo que accede a la opción de crear partida.
- 2.- Le da un nombre a la partida, así como también elige la forma en la que se conectará a la partida, ya sea con contraseña o acceso público.
- 3.- Elige la opción "Terminé de crear mi partida" y eso lo manda al menú principal en el cual ya deberá de estar la partida creada.

Nombre del caso de uso:	Designar un moderador
Actores participantes:	Iniciado por el Manager Aceptado o no por el Moderador
Flujo de eventos:	1.- El Manager después de crear la sala virtual decide designar a un moderador. 2.- Observa la lista de usuarios que se han unido a la sala 3.- Manda una invitación a un usuario para convertirse en moderador. 4.- El Moderador acepta o no el cargo 5.- El Moderador asignado puede empezar a asignar la palabra.

Nombre del caso de uso:	Inicia discusión.
Actores participantes:	Iniciado por algún Usuario. Moderador
Flujo de eventos:	1.-Un usuario opina algo. 2.- El Moderador asigna el turno a otro usuario. 3.- El otro usuario opina sobre lo que opino el usuario anterior. 4.- Llegado a un acuerdo el Manager decide dar por concluida la sala de discusión.

Nombre del caso de uso:	Concluir Sala
Actores participantes:	Iniciado por Manager
Flujo de eventos:	1.- El Manager elige la opción de por terminada la sala de discusión. 2.- Todos los involucrados vuelven a ser usuarios normales. 3.- Muestra estadísticas como duración de la discusión.

REQUERIMIENTOS FUNCIONALES.

- Cuando se conecta al servidor, un usuario puede entrar o salir de un encuentro.
- Cada encuentro tiene un manager.
- El manager es el usuario que ha planificado el encuentro (el nombre del encuentro, la agenda del encuentro y el moderador del encuentro).
- Cada encuentro puede tener también un moderador designado por el manager.
- La misión del moderador es asignar los turnos de palabra para que los usuarios hablen.
- El moderador también podrá dar por concluido el encuentro en cualquier momento.
- En cualquier momento un usuario puede consultar el estado del sistema, por ejemplo los encuentros planeados y su información.

REQUERIMIENTOS NO FUNCIONALES.

- La aplicación requiere de internet para poder estar al día y registrar los movimientos en la cuenta del usuario.
- La aplicación debe cumplir con un tamaño.
- El tiempo de acceso a la aplicación debe de ser de acuerdo con lo establecido por las normas.
- La aplicación en si debe ser gratuita.

EJERCICIO 3: Categorías de Requisitos No Funcionales

Clasifica en base a su categoría los siguientes requisitos no funcionales de un proyecto que pretende crear gafas de realidad aumentada:

- Las gafas dispondrán de una gama de modelos para dar gusto a todo tipo de clientes.

Requisito de Usabilidad

- Los modelos de las gafas no pueden ser idénticos a modelos patentados por marcas dedicadas al diseño de anteojos.

Requisito Legal

- Las gafas se manejarán en dos líneas, con aumento y sin aumento, para que el usuario escoja según sus necesidades.

Requisito de Usabilidad

- Las gafas se actualizarán descargando los archivos necesarios desde internet.

Requisito de Soporte

- El sistema operativo de las gafas no sufrirá errores que requieran tareas de reinicio o reseteo de interacción física con el usuario, pues solo tiene un botón.

Requisito de Confiabilidad

- Todas las aplicaciones que se diseñen para el reloj deben escribirse en Android.

Requisito de Implementación

- Las gafas cuentan con una entrada microUSB para su sincronización con un equipo.

Requisito de Soporte

- Las gafas puede sincronizarse en cualquier Windows basado en la tecnología NT y Apple Mac OS.

Requisito de Empaquetado.

- Las gafas tardarán en iniciar el sistema como máximo 30 segundos.

Requisito de Desempeño

- Cualquier usuario familiarizado con dispositivos Android encontrará relativamente fácil de usar las gafas.

Requisito de Usabilidad

EJERCICIO 4: Horarios del Hospital

Un hospital pequeño ha llevado registro y programando horarios para todos sus empleados (de tiempo completo, de medio tiempo, regulares y temporales) manualmente. De vez en cuando el hospital tiene problemas con la programación de los horarios y el personal no está disponible por varias razones. Otras veces, hay más personas de las que se necesitan. Este problema no ha representado un obstáculo en los negocios del hospital. Hace poco, el hospital recibió fondos para duplicar su tamaño y poder atender a más personas, por lo que la programación de horarios debe ser mucho mejor. La administración del hospital decide evaluar la opción de crear un sistema que les ayude con esta situación.

Encuentre actores, escenarios, casos de uso y requisitos funcionales y no funcionales del sistema de programación de horarios; teniendo en cuenta que habrá un encargado de llevar registro de todos los trabajadores del hospital.

Actores

- Encargado
- Empleados
- Sistema

Escenarios

Nombre	DarDeAlta
Actores Involucrados	Encargado Sistema Empleados
Flujo de eventos	1. Un empleado nuevo llega con el encargado 2. El encargado registra al empleado en el sistema

Nombre	AsignarHorario
Actores Involucrados	Encargado Sistema Empleados
Flujo de eventos	1. El encargado busca empleados sin horario asignado 2. El encargado asigna un horario al empleado

Nombre	DarDeBaja
Actores Involucrados	Encargado Sistema Empleados
Flujo de eventos	1. Un empleado que dejará de trabajar en el hospital llega con el encargado. 2. El encargado busca el registro del empleado 3. El da de baja al empleado.

Nombre	RevisarListaEmpleados
Actores Involucrados	Encargado Sistema
Flujo de eventos	1. El encargado desea ver la lista de empleados 2. El sistema muestra las listas de empleados laborando y no laborando

Casos de uso

Nombre	RegistrarEmpleado
Actores Involucrados	Encargado Empleados Sistema
Flujo de eventos	1. El encargado entra a la opción de registro 2. El sistema carga el formulario de registro 3. El encargado llena el formulario en base a los datos del empleado 4. El sistema registra al empleado
Condiciones de Entrada	El sistema acepta el formulario recién llenado.
Condiciones de salida	El sistema registra a un nuevo empleado.
Calidad de requisitos	El sistema tarda en registrar al nuevo empleado menos de 5 segundos.

Nombre	ComprobarDisponibilidad
Actores Involucrados	Sistema
Flujo de eventos	1. El sistema recibe orden de asignar horario a un empleado 2. El sistema revisa el registro del empleado 3. Verifica si no tiene un horario asignado 4. Asigna horario.
Condiciones de Entrada	El sistema abre el registro del empleado.
Condiciones de salida	<ul style="list-style-type: none"> El sistema verifica que el usuario no tiene horario y se lo asigna. O El sistema verifica que el usuario ya tiene horario y pregunta al encargado qué debe hacer.
Calidad de requisitos	El sistema puede demorar en encontrar al empleado buscado. El sistema no debe tardar en revisar la disponibilidad de un empleado más de 3 segundos.

Nombre	AsignarNuevoHorario
Actores Involucrados	Sistema Encargado Empleado
Flujo de eventos	1. El sistema se encuentra con que un empleado ya tienen un horario asignado 2. El sistema pregunta al empleado si desea cambiar el horario asignado de dicho empleado.

	3. El sistema cambia el horario.
Condiciones de Entrada	El sistema encuentra un horario asignado
Condiciones de salida	<ul style="list-style-type: none"> El sistema cambia el horario del empleado. ○ El sistema no cambia el horario del empleado.
Calidad de requisitos	El sistema tarda en registrar al nuevo empleado menos de 5 segundos.

Nombre	AvisarSobreBajaFutura
Actores Involucrados	Sistema Encargado
Flujo de eventos	<ol style="list-style-type: none"> El sistema encuentra con una fecha próxima de baja de empleado temporal. El sistema avisa al encargado sobre esta futura baja, para que prevenga la situación.
Condiciones de Entrada	El sistema se percata de una fecha cercana de baja.
Condiciones de salida	El sistema manda un aviso al encargado.
Calidad de requisitos	El aviso se recibe cada vez que se enciende el sistema o se consulta la zona de avisos.

Nombre	AvisarBacante
Actores Involucrados	Sistema Encargado Empleado
Flujo de eventos	<ol style="list-style-type: none"> El sistema se encuentra con un horario sin personal asignado. El sistema manda aviso al encargado El encargado decide ver la lista de empleados El encargado asigna un empleado al horario vacío.
Condiciones de Entrada	El sistema reconoce un horario sin empleado asignado.
Condiciones de salida	El sistema manda un aviso al encargado
Calidad de requisitos	El aviso se recibe cada vez que se abre una bacante, se enciende el sistema o se consulta la zona de avisos.

Nombre	AvisarCambioDeTurno
Actores Involucrados	Sistema Encargado Empleado
Flujo de eventos	<ol style="list-style-type: none"> El sistema se percata de que el turno de uno de los empleados de medio tiempo está por terminar. El sistema informa al encargado. El sistema muestra al empleado que suplirá al actual y las horas de entrada y salida de este.

Condiciones de Entrada	El sistema encuentra un turno por terminar.
Condiciones de salida	El sistema manda aviso al encargado
Calidad de requisitos	El aviso se recibe cada vez que se abre terminará o se termina un turno, o se consulta la zona de avisos.

Requisitos Funcionales

- El sistema debe revisar los horarios de todos los empleados al iniciarse, con el fin de ver si es necesario mandar algún aviso al encargado.
- El sistema dará de baja de los empleados que dejen de trabajar en el hospital, pero conservará el registro de estos pasándolos a una lista de ex empleados.
- El sistema no siempre preguntará antes de cambiar un horario ya asignado.
- El sistema será capaz de mostrar la lista de empleados, la dividirá en dos secciones (empleados actuales y ex empleados); mostrando de cada empleado actual su status (laborando o descansando).
- El sistema mandará un aviso cada vez que se percate de una vacante, una baja programada (empleados temporales) o el final del turno de algún empleado (empleados de medio tiempo).

Requisitos No Funcionales

- El sistema debe ser capaz de ejecutarse en Windows 7 (el sistema operativo que se maneja en la computadora del encargado) y en Windows 8, en caso de que se decida actualizar el sistema operativo.
- El sistema debe tener un reloj y calendario internos.
- El tiempo de registro de un empleado no puede tardar más de 5 segundos.
- Los avisos de vacantes se mostrarán con color de letra azul, los avisos de término de turno con verde, y los avisos de bajas próximas con rojo.
- El encargado del sistema debe estar familiarizado con programas parecidos a Excel, pues el registro de empleados se lleva a cabo en una interfaz parecida.
- En caso de fallas, el sistema puede reiniciarse para continuar trabajando.
- El sistema debe ser capaz de correr con fluidez en una computadora con 4gb de memoria RAM.
- El sistema es de uso exclusivo para el hospital.