

Universidad de Alcalá

Paradigmas Avanzados de Programación

ENUNCIADO PECL3

Cundy Crosh Soga –
Cloud Integration



Versión: 1 (última revisión, 20 de abril de 2023).

Índice

	Pág.
1. El Videojuego Base.....	2
1.1. Modificaciones para la PECL3	2
1.2. Incorporando Cloud	2
2. Trabajo a realizar	3
2.1. Modificaciones de puntuación (obligatoria)	3
2.2. Parte Cloud (obligatoria).....	3
2.3. Comparativa Clouds (extra)	3
3. Operaciones limitadas y prohibidas con Scala.....	4
4. Aclaraciones Cloud.....	4
5. Prerrequisitos de entrega	5
6. Documentación a entregar	5
7. Forma de entrega.....	6
8. Defensa	6
9. Evaluación	6

1. El Videojuego Base

El juego “Cundy Crosh Soga” que se propone para la PECL3 se basa en el popular juego de móviles “Candy Crush Saga”, un videojuego multidioma desarrollado por la compañía británica King, originalmente disponible como una aplicación de Facebook y adaptado para los sistemas operativos Android, iOS y Windows Phone, que se lanzó el 12 de abril de 2012 (fuente: Wikipedía).

Se tomará, como base para la PECL3, la solución propuesta en Scala para la PECL2. Sobre ese proyecto base, se piden unas modificaciones y una conectividad Cloud.

1.1. Modificaciones para la PECL3

Sobre la base del enunciado de la PECL2 se propone la contabilización de un sistema de puntuación basado en las siguientes reglas:

- Por cada bloque (normal o especial) borrado – 1 punto.
- Por cada 10 bloques borrados – 1 punto adicional.
- Por cada bloque especial bomba (fila o columna) seleccionado – 5 puntos.
- Por cada bloque especial TNT seleccionado – 10 puntos.
- Por cada bloque especial rompecabezas – 15 puntos.
- Modificador final por modo de juego: modo fácil (x1) y modo difícil (x2).

Por ejemplo,

1	2	TNT	1	3	2	B-Fila	1	1	1	3	4	(modo fácil)
---	---	-----	---	---	---	--------	---	---	---	---	---	--------------

*El borrado de estos 12 bloques (por selección de la bomba de fila) generaría: 12 (por cada bloque individual borrado) + 1 (por cada 10 bloques borrados) + 5 (por selección de la bomba fila) * 1 (modo fácil) = 18 puntos.*

El juego finaliza, al igual que en la PECL2, cuando al usuario no le quedan vidas para continuar. Se guardará la fecha y hora de finalización y la duración de la partida (en segundos).

Al finalizar una partida, se solicitará el nombre que el usuario quiere que figure en la tabla clasificatoria. Al estilo de las tradicionales máquinas recreativas. Será una cadena de texto alfanumérica sin restricciones (tampoco se requiere que sea única, puede duplicarse sin necesidad de control).

1.2. Incorporando Cloud

Los datos de cada partida finalizada (nombre, puntuación, fecha y duración) se enviarán al Cloud para su almacenamiento en una Base de Datos (o similar).

Se dispondrá de una URL pública en la que mostrar un visor Web en la que mostrar la tabla clasificatoria. Cada fila mostrará los datos de una partida, ordenadas por fecha. Se podrán incluir, opcionalmente, filtros de búsqueda y formas de ordenación alternativas sobre las diferentes columnas disponibles.



Ilustración 1. Conceptualización de la integración Cloud de la PECL3

2. Trabajo a realizar

2.1. Modificaciones de puntuación (obligatoria)

Este apartado es obligatorio, y consiste en la implementación del juego utilizando las técnicas básicas de programación funcional con Scala.

- 1) Implementar las modificaciones, sobre la PECL2, de puntuación del Juego empleando Scala (con IntelliJ IDEA) para actualizar el estado del juego conforme a las especificaciones.

2.2. Parte Cloud (obligatoria)

De forma avanzada, se piden integraciones en Cloud para poder recopilar, almacenar y mostrar los resultados de la clasificación de las partidas. Para este apartado, se seleccionará 1 proveedor Cloud (a elección de los alumnos). Detalles:

- 2) Recepción de los datos de cada partida y almacenamiento en una Base de Datos (o similar).
- 3) Creación del visor web, que recupere los datos de la Base de Datos y muestre la tabla clasificatoria.
 - De forma básica, un visor web (HTML) con una tabla en la que cada fila mostrará los datos de una partida, ordenadas por fecha. Se podrán incluir, opcionalmente, filtros de búsqueda y formas de ordenación sobre las diferentes columnas disponibles.
 - Hay libertad de diseño de la interfaz, siempre y cuando se respeten las funcionalidades solicitadas.

Nota: Ver apartado “4. Aclaraciones Cloud” para más detalles.

2.3. Comparativa Clouds (extra)

Como apartado extra se pide realizar una comparativa opcional entre varias clouds. Es un apartado completamente opcional, y sirve para obtener puntos extra.

- 4) Comparativa Clouds
 - Para la parte obligatoria, se selecciona 1 proveedor Cloud. Para este apartado opcional se pide realizar la integración con otros proveedores Cloud diferentes. O, dentro del mismo proveedor, con otro/s módulo/s diferente/s, siempre y cuando los componentes a utilizar sean significativamente diferentes a los ya utilizados.
 - Se puede modificar, para este apartado extra, lo que se desee del juego en Scala si fuera necesario para adaptar la funcionalidad a las Clouds comparadas.

Nota: Ver apartado “4. Aclaraciones Cloud” para más detalles.

3. Operaciones limitadas y prohibidas con Scala

Mismas limitaciones que para la PECL2

Se debe tener en cuenta que, para la realización de los programas en Scala, se deben usar vals para la definición de variables, objetos inmutables y métodos sin efectos colaterales. Con limitadas excepciones, y previa autorización de los profesores, se podrán usar vars, objetos mutables y efectos colaterales.

En la medida de lo posible, se utilizarán listas. En el caso de utilización de otros elementos (Sets o Maps), deberá justificarse por qué no ha podido utilizarse listas.

Operaciones **prohibidas** (puede usarse funciones que hagan esa funcionalidad, pero deberán implementarse con funciones propias del alumno):

- Utilización del operador de concatenación de listas (::: o ++).
- Operaciones de las listas, como reverse(), length(), last() e isEmpty().
- Otras operaciones, como filter(), flatten(), flatMap() y operaciones derivadas de reverse (como, por ejemplo, reverseMap(), reverse_:::).
- No se pueden usar bucles (for, while y similares) para recorrer elementos.
- Colecciones mutables.

4. Aclaraciones Cloud

Advertencia – *Nota de renuncia: Los profesores de la asignatura no se hacen responsables de los gastos producidos por los alumnos en las plataformas Cloud. Es responsabilidad del alumno detectar si las acciones que está realizando conllevan algún tipo de gasto.*

Debe utilizarse cualquier proveedor Cloud de los “típicos” introducidos en la asignatura (Azure, Amazon Web Services o Google Cloud), si bien para la parte opcional pueda optarse por otros proveedores alternativos. Aunque como recomendación para evitar gastos indeseados, se fomenta el uso de los créditos de estudiante de Azure o los módulos gratuitos de Google Cloud.

Nota: El entorno local (o localhost) no es aplicable como proveedor Cloud.

Nota (II): Para evitar gastos (monetarios o de recursos) adicionales (si los hubiera), solo se exige mantener en ejecución los entornos Cloud durante la Defensa de la PECL, si así lo solicita el profesor para ver el resultado de la integración durante la propia Defensa.

5. Prerrequisitos de entrega

Es requisito indispensable haber entregado las actividades semanales en la plataforma gamificada (<http://plataformagamificacion.cc.uah.es/pap>), en fecha y hora anunciada cada semana en las sesiones de Laboratorio.

En concreto, las actividades a entregar son:

- Tarea 5.1 – Ejercicios I
- Tarea 5.2 – Ejercicios II
- Tarea 5.3 – Ejercicios III

Según la Guía Docente de la Asignatura, se otorgará la calificación de "*No presentado*" al alumno que, habiendo optado por el procedimiento de evaluación continua, no se presenten a la evaluación de todos los instrumentos de Evaluación. Por tanto, la no presentación de todas las entregas semanales supondrá un "*No Presentado*" en la convocatoria actual.

Nota: No se admiten entregas semanales manifiestamente erróneas para cumplir con los prerrequisitos de la PECL.

6. Documentación a entregar

La entrega de la PECL deberá constar de:

- Proyecto de IntelliJ IDEA con la solución codificada.
 - 1x Proyecto de implementación básica + 1 integración Cloud.
 - (opcional) Nx Proyecto de implementación extra (si se modificara algo) (1 proyecto por cada integración comparada).
- Memoria PDF de la parte Cloud, con la información relevante (pasos y/o comandos), capturas y los detalles clave para la configuración del entorno/proveedor Cloud y despliegue de los módulos necesarios.
 - Si se genera/utiliza código, se añadirá como anexo.
 - (opcional) Si se realiza la parte extra de la PECL, se deberá incluir un apartado en Memoria PDF por cada proveedor Cloud elegido (con los detalles específicos de cada implementación), así como un apartado en memoria PDF (máximo 5 páginas) que realice una comparativa razonada entre todas las plataformas Cloud utilizadas.
- Breve video comentando la ejecución
 - 1x Video modificación juego + envío de datos al Cloud.
 - 1x Video de integración Cloud + visualización de datos.
 - (extra) 1x Video de las otras integraciones Cloud
 - Los videos no superarán los 3 minutos de duración. Deben participar ambos miembros de la pareja de trabajo. Se comentarán los detalles clave de la implementación y demostración de ejecución.

7. Forma de entrega

El trabajo se realizará en parejas. Se entregará mediante el correspondiente Buzón de Entrega disponible en BlackBoard. La fecha tope de entrega y fecha de Defensa se encuentra disponible en BlackBoard. Al ser un trabajo en pareja, solamente es necesario que uno de los miembros haga la entrega.

Nota: No se admitirán entregas enviadas al correo electrónico (o mensajería interna de BlackBoard) de los profesores. Únicamente se permite la interacción con el Buzón de entrega.

8. Defensa

La defensa de las prácticas será en la fecha y la forma indicada por el/la profesor/a del laboratorio, pudiendo ser esta oral o escrita. En el caso de no contestarse correctamente a las cuestiones presentadas, la práctica presentada podrá considerarse como suspensa.

9. Evaluación

La nota máxima a la que puede acceder se pondera en función de:

- Calidad del material entregado por el alumno.
- Documentación de seguimiento presentada durante su realización
- Realización e implementación de la práctica.
- Defensa e implementación de las modificaciones solicitadas.
- Manejo del software utilizado para el desarrollo de la práctica.
- Código desarrollado.

La puntuación de cada apartado es el siguiente:

Sección	Nota
Modificaciones de puntuación (obligatoria)	2 pts. Máx.
Integración Cloud (obligatoria)	5 pts. Máx.
Comparativa Cloud (extra)	(opcional) Máximo 3 puntos extra. De esto, se puede conseguir un máximo de 0,5 pts. por cada Cloud extra comparada.
Documentación entregada	3 pts. Máx.
Defensa	1x, 1/2x o 0x a la nota de la PCL