SISTEMA MULTIMEDIA QUE SIMULA ALGORITMOS DE CRIPTOGRAFÍA

MULTIMEDIA SYSTEM THAT SIMULATES CRYPTOGRAPHY ALGORITHMS

Meliza Contreras González
Benemérita Universidad Autónoma de Puebla
vikax68@gmail.com

Ana Patricia Cervantes Márquez
Benemérita Universidad Autónoma de Puebla
cervantes.patty@gmail.com

Claudia Zepeda Cortés
Benemérita Universidad Autónoma de Puebla
czepedac@gmail.com

Yolanda Moyao Martínez
Benemérita Universidad Autónoma de Puebla
ymoyao@cs.buap.mx

Beatriz Beltrán Martínez
Benemérita Universidad Autónoma de Puebla
bbeltranmtz@gmail.com

Rafael Gallardo García
Benemérita Universidad Autónoma de Puebla
rafaGallardo@gmail.com

Resumen

La criptografía es un área vital para el desarrollo de aplicaciones seguras. Sin embargo, resulta complejo realizar los algoritmos o conocer su efecto si se desconocen los principios básicos de funcionamiento por lo que, en este trabajo, mediante técnicas multimedia se ejemplifica mediante simulaciones el proceso de cifrado de forma didáctica.

Palabra(s) Clave: sistema multimedia, criptografía, didáctica.

Abstract

Cryptography is a vital area for the development of secure applications. However, it is complex to perform the algorithms or know their effect if the basic operating principles are unknown, so in this work using multimedia techniques, the encryption process is exemplified by simulations in a didactic way.

Keywords**:** *multimedia system, cryptography, didactics.*

1. Introducción

Dentro del desarrollo de cualquier proyecto se convierte en parte fundamental incorporar un modelo de Ingeniería de Software para crear un sistema funcional y que cumpla con todos los requerimientos de nuestro cliente.

Es importante mencionar que, de acuerdo, al modelo de proceso utilizado, la parte de análisis y diseño será la que nos proporcione la base de nuestro sistema, así que deberíamos tomar como premisa que un buen diseño nos proveerá de una implementación con menos errores, que cumpla con los requerimientos al 100%, y sobre todo no extenderá el tiempo calculado para el desarrollo de nuestro sistema.

Hoy día existen muchas herramientas para el análisis y diseño de requerimientos de un sistema, mas, sin embargo, el lenguaje UML (Arlow, 2012), que permite modelar los procesos de software de forma intuitiva, se posiciona como la mejor opción gracias a sus conceptos y a todos los diagramas que incorpora para realizar esta tarea.

También es importante que, al tener nuestro diseño completo y pasemos a la implementación, seamos capaces de optar por la mejor opción de lenguaje de programación, que brinde estabilidad, fiabilidad y funcionalidad dependiendo del tipo de sistema y sobre todo del requerimiento que el usuario indique.

Aunque existen nuevas tendencias y cada día los lenguajes de programación crecen y se desarrollan, la programación con PHP provee la compatibilidad con otros lenguajes de programación como lo es Html y MySQL (Cabezas, 2014), (De la Cruz, 2004), que en conjunto nos brindan una poderosa herramienta para la implementación de sistemas.

Por otro lado, Flash se ha convertido en uno de los lenguajes principales en los entornos multimedia, está presente en cualquier aplicación web y se ha desarrollado incluso para crear aplicaciones más grandes que involucran más que una simple animación con una línea de tiempo, con su lenguaje controlador ActionScript podemos crear clases, objetos, métodos, etc. Y así el alcance de esta herramienta se extiende incluso como el de un lenguaje diseñado exclusivo para programación.

Para el alumno muchas veces representa cierto nivel de dificultad la comprensión y asimilación del funcionamiento de los algoritmos criptográficos, es decir, cómo funcionan internamente las implementaciones de estos. La teoría asociada a las funciones criptográficas se basa en una serie de teorías matemáticas las cuales a su vez pueden ser complejas y de difícil absorción.

Con este sistema, se propone facilitar el proceso de abstracción de los algoritmos criptográficos para el estudiante con el objetivo de incentivar el desarrollo del área de seguridad informática que implica el desarrollo de teoría de números por lo que este sistema facilita el aprendizaje de éstos de una forma interactiva.

El trabajo está conformado con las siguientes secciones: en la sección dos se plantean los métodos que facilitan el análisis y diseño del sistema, en la sección tres de resultados se muestra la implementación y resultados, y finalmente la sección de discusión.

2. Métodos

La metodología que se utilizará para desarrollar la aplicación será el modelo de proceso “cascada”. Este modelo divide el desarrollo del proyecto en 5 etapas que son el análisis, diseño, implementación, pruebas y mantenimiento. La característica principal de este modelo se basa en que no se puede avanzar a la siguiente etapa hasta haber completado la anterior incluyendo la documentación del proyecto de cada etapa.

En primera instancia abordaremos el análisis y diseño, se realizará el planteamiento del problema, inmediatamente los requerimientos para poder formular el diagrama de casos de uso su especificación y escenarios.

Posterior a esto se creará el modelo conceptual y el modelo de análisis que incluye diagramas VOPC, de secuencia y colaboración. Respecto a la parte de diseño, se diseñará el modelo E-R de la base de datos de la aplicación hasta el modelo relacional, es decir las tablas de nuestra base de datos que se implementara en la aplicación.

Se realizará la entrega del documento formal sobre estos puntos para su valoración, y pasaremos a la etapa de implementación (Chandrinos, 1998).

En esta etapa se modelará la interfaz de acuerdo a los requerimientos del sistema, y se creará la base de datos. Para esta etapa al final se mostrará la aplicación funcional junto con la respectiva documentación.

Una vez valorada esta etapa se realizarán las pruebas donde se revisará la aplicación y ante los fallos que presente pueden realizarse iteraciones un determinado número de veces en cada etapa hasta que esta sea aceptada.

El modelo en cascada ofrece la etapa de mantenimiento, sin embargo, por los fines de la aplicación se abordará como un posible trabajo a futuro.

Con esta metodología esperamos reducir el tiempo de desarrollo, al final obtener una aplicación íntegra que cumpla con la necesidad del usuario y con el mínimo de fallos.

El modelado de casos de uso, es la técnica más efectiva y a la vez la más simple para modelar los requisitos del sistema (Pressman, 2002) desde la perspectiva del usuario.

Los Casos de Uso se utilizan para modelar cómo un sistema o negocio funciona actualmente, o cómo los usuarios desean que funcione. No es realmente una aproximación a la orientación a objetos; es una forma de modelar procesos. Es, sin embargo, una manera muy buena de dirigirse hacia el análisis de sistemas orientado a objetos. Los casos de uso son generalmente el punto de partida del análisis orientado a objetos con UML.

Un caso de uso se modela para todos los procesos que el sistema debe llevar a cabo. Los procesos se describen dentro del caso de uso por una descripción textual o una secuencia de pasos ejecutados. En la figura 1 se muestra el diagrama de casos de uso del sistema.

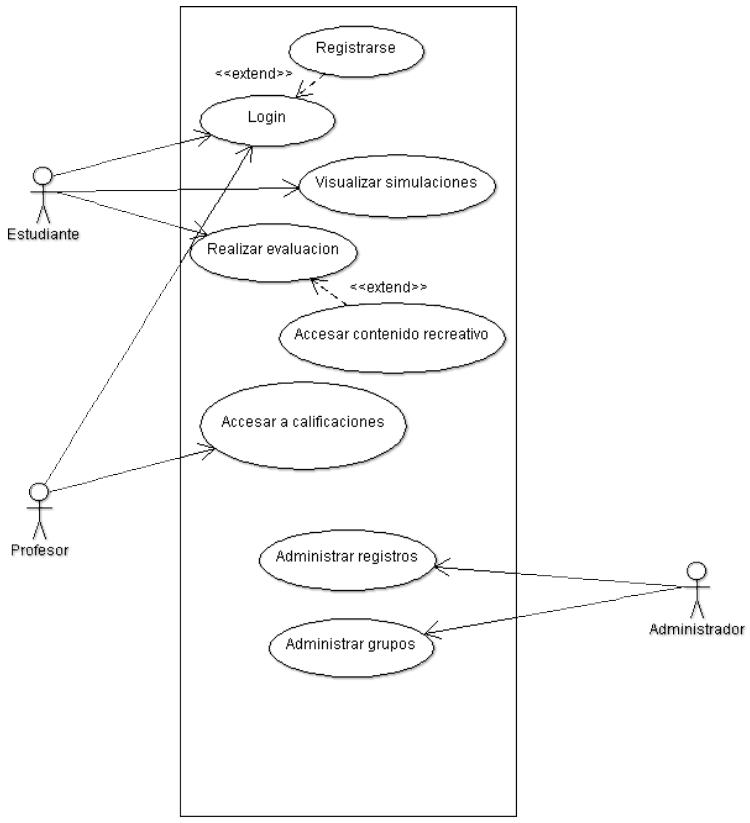

Figura 1 Diagrama de casos de uso de la aplicación.

Para cada Caso de Uso se requieren las especificaciones siguientes:

- ***Nombre: "Registrarse"***: Mediante este caso de uso el estudiante se registrará en el sistema para que se le permita el acceso.

  Flujo de Eventos:

  - ✓ Precondiciones: El estudiante aún no está dado de alta en el sistema.
  - ✓ Flujo Principal: El estudiante elige la opción registrarse, el sistema registra la información y la almacena.
  - ✓ No existen flujos alternos
  - ✓ Flujos de excepción:
    - El estudiante no ingreso toda la información requerida, se le pedirá reingrese la información o terminar el caso de uso.
    - Que la información del registro ya se encuentre almacenada, se le notificará que ya ha realizado su registro y terminara el caso de uso.

- ***Caso de Uso: "Login"***. Con este caso de uso, el estudiante y el profesor podrán acceder al sistema y a su contenido.

  Flujo de Eventos:

  - ✓ Precondiciones: El alumno o profesor está registrado en sistema.
  - ✓ Flujo principal: El estudiante o profesor inician su sesión con su nombre de usuario y contraseña.
  - ✓ No existen flujos alternos.
  - ✓ Flujos de excepción: Error de usuario y/o contraseña el caso de uso no finalizará hasta que se ingresen los datos correctos o finalice por el usuario.

- ***Caso de uso: "Visualizar simulaciones"***. Con este caso de uso el estudiante podrá visualizar cada una de las simulaciones de los algoritmos que tiene cargados el sistema.

  Flujo de eventos:

  - ✓ Precondiciones: El estudiante cumplió el caso de uso "login".
  - ✓ Flujo principal:
    - El sistema pedirá al estudiante que elija la forma en que desea visualizar las simulaciones, como una presentación o bien por categoría llave privada o llave pública.
    - El estudiante elige visualizar las simulaciones como una presentación. Si la elección es:
      - ❖ Llave pública se ejecuta Llave pública (flujo alterno).
      - ❖ Llave privada se ejecuta Llave privada (flujo alterno).
    - La presentación de las simulaciones termina y termina el caso de uso.
  - ✓ Flujos Alternos
    - Llave publica:
      - ❖ El sistema muestra los algoritmos simulados que se encuentran en esa categoría. Algoritmo 1, Algoritmo 2.
      - ❖ El estudiante elige que algoritmo visualizar y termina el caso de uso.

- Llave privada
  - ❖ El sistema muestra los algoritmos simulados que se encuentran en esa categoría. Algoritmo 1, Algoritmo 2.
  - ❖ El estudiante elige que algoritmo visualizar y termina el caso de uso.

✓ No existen flujos de excepción

- ***Caso de uso: "Realizar evaluación".*** Mediante este caso de uso al terminar de visualizar las simulaciones de los algoritmos se le realizará una evaluación interactiva al estudiante y la calificación se almacenará en el sistema.

  Flujo de eventos:

  ✓ Precondiciones: Haber concluido el caso de uso "visualizar simulaciones".

  ✓ Flujo Principal:
  - Se mostrarán en pantalla preguntas acerca de las simulaciones de los algoritmos, el estudiante deberá elegir la respuesta correcta y seleccionar Continuar, Omitir, Regresar y/o Finalizar.
  - Conforme el estudiante vaya contestando las preguntas elegirá Continuar hasta llegar a la última pregunta.
  - Cuando el estudiante responde la última pregunta ejecuta Finalizar y el caso de uso termina.

  ✓ Flujos Alternos:
  - Omitir: El estudiante tendrá la oportunidad de omitir preguntas y regresar luego a esa pantalla.
  - Regresar: El estudiante tendrá la opción de retroceso para regresar a preguntas anteriores y modificar su respuesta o contestar preguntas que se omitieron.

  ✓ Flujos de excepción: Si el estudiante no omite preguntas solo da siguiente sin contestar, no se marcará como no contestada y no se notificará al estudiante que falta por responder.

- ***Caso de uso: "Acceso a contenido recreativo"***. Con este caso de uso el estudiante que ha realizado la evaluación y obtenido una buena calificación podrá visualizar videos recreativos.

  Flujo de Eventos:

  - ✓ Precondiciones: El alumno tendrá un promedio de evaluación minino de 8.
  - ✓ Flujo principal: Al finalizar cada simulación el alumno recibirá su calificación, si es mínima de 8, tendrá la opción para ver los videos recreativos, si decide no verlos se ejecuta FIN (flujo alterno). Termina caso de uso.
  - ✓ Flujo Alterno:
    - Fin: El estudiante elige no ver videos el sistema termina.
  - ✓ No existen flujos de excepción.

- ***Nombre: "Acceso a calificaciones"***. El profesor tendrá el privilegio de consultar la lista de calificaciones de quien utilizo el sistema.

  Flujo de eventos:

  - ✓ Precondiciones: El profesor cumplió con el caso de uso "login".
  - ✓ Flujo principal:
    - El sistema le da las opciones de ver lista de calificaciones, Imprimir y Salir.
    - El profesor elige ver lista de calificaciones, si elige imprimir se ejecuta IMPRIME (flujo alterno), si elige Salir se ejecuta 2.3.2 SALIR. Termina caso de uso.
  - ✓ Flujo Alterno:
    - Imprime: El Profesor podrá imprimir la lista de los estudiantes y sus calificaciones configurando con las opciones de su navegador.
    - Salir: Cierra sesión del profesor y finaliza sistema.
  - ✓ Flujos de excepción: No se puede imprimir, el sistema dará la opción de guardar el archivo.

- ***Caso de Uso: "Administrar registros"***. Mediante este caso de uso el administrador podrá crear, modificar y eliminar registros del sistema.

  Flujo de Eventos:

  - ✓ Precondiciones: Que existan registros en la base de datos.
  - ✓ Flujo principal: El administrador tiene acceso a la base principal de datos donde puede realizar los movimientos requeridos si elige Crear se ejecuta CREAR (flujo alterno), si elige Modificar se ejecuta 2.3.2 MODIFICAR, si elige Eliminar se ejecuta 2.3.3 ELIMINAR.
  - ✓ Flujo Alterno:
    - Crear:
      - ❖ El administrador puede dar de alta a un nuevo estudiante.
      - ❖ El administrador brindará al estudiante su usuario y contraseña asignada.
    - Modificar: El administrador podrá modificar información de registros.
    - Eliminar: El administrador podrá eliminar registros del sistema.
  - ✓ No hay flujos de excepción.

- ***Caso de Uso: "Administrar Grupos"***. Descripción: Con este caso de uso el administrador podrá crear grupos si hay más de uno que toma la materia de criptografía e incluso con profesores diferentes.

  Flujo de eventos:

  - ✓ Ninguna Precondición
  - ✓ Flujo principal
    - El administrador elige crear grupos, si elige modificar grupos se ejecuta Modificar grupos (flujo alterno), si elige eliminar grupos se ejecuta 2.3.2 Eliminar grupos.
    - El administrador realiza búsquedas en la base de datos del sistema para agrupar estudiantes de una misma sección. Termina caso de uso.
  - ✓ Flujo Alterno:

- Modificar grupos: El administrador elige un grupo para modificar información.
- Eliminar grupos: El administrador elige un grupo para eliminar información.

✓ No hay flujos de excepción.

Diagrama Entidad–Relación

En el caso del modelado de información se realizó el modelo Entidad Relación (Somodevilla, 2011), (Ullman, 1998) que aparece en la figura 2.

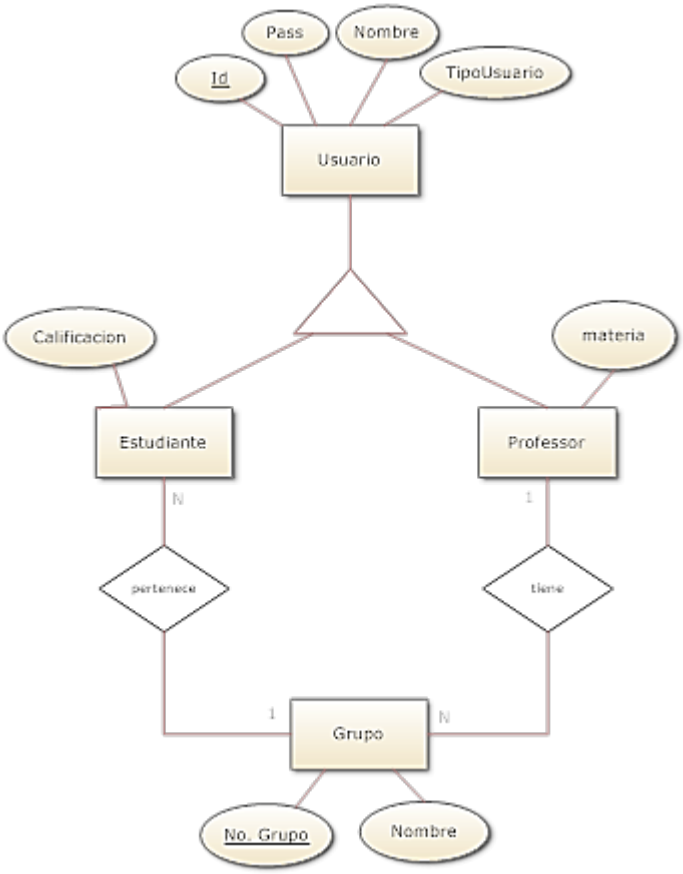

Figura 2 Diagrama E-R de la aplicación.

Una vez diseñado el diagrama de Bases de Datos y los casos de uso, se puede plantear la elaboración del sistema multimedia que se muestra en la siguiente sección.

3. Resultados

En este caso el sistema multimedia que simula algunos de los más elementales algoritmos de criptografía fue implementado sobre una plataforma PC con sistema Windows XP. Para lograr la implementación de la aplicación se requiere un servidor web local, en este caso se empleó WampServer 2.0, que provee de las aplicaciones necesarias como es la herramienta MySQL (Welling, 2005) para gestionar la base de datos, y herramientas para lograr la comunicación de la aplicación del usuario con esta.

La interfaz gráfica se diseñó con una plantilla CSS y Html, la conexión requerida con MySql se logra con código PHP insertado dentro de la página Html (Posadas, 2005), toda la animación multimedia dentro de la interfaz se realizó con Flash.

En la página de inicio el usuario podrá iniciar sesión, si el tipo de usuario es un profesor su Id y Contraseña serán provistos por el Administrador, en caso de que sea un estudiante deberá registrarse primero para obtener su Id y contraseña, el cuerpo de la página principal se reserva para insertar banners o cualquier información requerida por el usuario final de nuestro sistema, como se observa en la figura 3.

Figura 3 Página de Inicio.

En el caso del estudiante deberá registrarse por única vez, para darse de alta en la base de datos, como se muestra en figura 4.

Figura 4 Registro del estudiante.

Ahora el usuario ya puede iniciar sesión con su Id y contraseña para que pueda ingresar al contenido multimedia, se le mostrará un menú principal donde el estudiante podrá seleccionar la simulación del algoritmo (Maiorano, 2009) que quiera aprender, en este caso en la figura 5 se muestra el cifrado César, en la figura 6 Vigenere y en la figura 7 el algoritmo de Hill.

Figura 5 Simulación Algoritmo de Cesar.

Figura 6 Simulación Algoritmo de Vigenere.

Figura 7 Simulación Algoritmo de Hill.

Como se observa en la simulación de los algoritmos se muestran los caracteres para realizar el cifrado, y se genera el texto cifrado con la explicación de los desplazamientos de acuerdo al algoritmo presentado y posteriormente se muestra a partir del texto cifrado los pasos inversos que se requieren para obtener el texto de origen. Finalmente en la figura 8, se muestra la evaluación obtenida con cada algoritmo trabajado.

Figura 8 Interfaz de evaluación.

4. Discusión

La ventaja de los sistemas multimedia es que logran en los estudiantes una experiencia de gamificación que permite que se sientan cómodos y puedan facilitar más sus mecanismos de aprendizaje.

Como trabajo futuro se puede plantear ahora hacer simulaciones del algoritmo RSA y curvas elípticas.

5. Bibliografía y Referencias

[1] Maiorano, A., Criptografia Tecnicas de desarrollo para profesionales, Buenos Aires, Argentina 2009: Alfaomega.

[2] Arlow,,Jim., Neustadt,I., UML and the Unified Process, Great Britain 2002: Addison Wesley.

[3] Cabezas, L. M. PHP 5, España 2004: ANAYA Multimedia.

[4] De la Cruz, D., Zumbado C., Flash, PHP y MySQL, España 2004: ANAYA Multimedia.

[5] Posadas, M., Introducción al lenguaje XML, España 2000: Grupo Eidos.

[6] Pressman, Roger. 2002. Ingeniería de Software un enfoque práctico Quinta Edición. Madrid España: McGraw Hill.

[7] Sommerville, Ian. "Ingeniería de Software" Séptima Edición. Madrid, España: Addison Wesley.

[8] Somodevilla G. Maria J. 2011 "Diseño de Bases de Datos". Trabajo no publicado.

[9] Ullman, Jeffrey. Widom, Jennifer. "Introducción a los sistemas de Bases de Datos". México 1999: Prentice Hall.

[10] Welling Luke. Thomsom, Laura. "Desarrollo Web con PHP y MySQL", España 2005: ANAYA Multimedia.

[11] Chandrinos, K. V., & Trahanias, P. E., Web-based Information Systems ERCIM Workshop Proceedings: http://www.ercim.org/publication/ws-proceedings/DELOS6/. 1998.