

PROYECTO DE SOFTWARE 2016
DOCUMENTO DE REQUISITOS DE SOFTWARE



Proyecto: Ambiente de AutoEvaluación AAEEV 2.0

Alumnos

Alejandro Canosa

Nazareno Galvan

Agustín Di Stefano

Cliente: Sebastián Martins

1. Introducción

1.1. Objetivo

1.1.1. Propósito del documento

Este documento recoge la especificación de requisitos. Estos serán usados como guía para el desarrollo del sistema una vez que el cliente haya aceptado con la continuación del mismo

1.1.2. Audiencia

Este documento está dirigido al cliente y a los desarrolladores de la aplicación a producir.

1.2. Ámbito

1.2.1. Identificación del producto

El nombre del proyecto es Ambiente de AutoEvaluación 2.0 (referido como AAEV 2.0)

1.2.2. Que hace y no hace el producto

El producto consta del ingreso de estudiantes de diversas carreras a las materias correspondientes aprobado por los docentes de estas para evaluar su adquisición de conocimientos en las correspondientes cursadas. A su vez el producto permitirá a los docentes ingresar con sus datos y crear nuevos exámenes para estas materias previamente mencionadas. Este ambiente es exclusivo para evaluación y no contendrá los apuntes ni los programas de las materias.

1.2.3. Aplicaciones del software: beneficios, objetivos y metas

El objetivo es lograr que el alumno evalúe los conceptos y conocimientos adquiridos en clase para mejorar su rendimiento en la cursada, así como también refinar sus conocimientos a la hora de prepararse para un examen.

Para el lado del docente, permite analizar el rendimiento de los alumnos conforme a los contenidos de la materia que se evalúan y el programa de las mismas, para lograr conseguir una cátedra de mayor calidad.

1.3. Definiciones, acrónimos y abreviaturas:

Termino	Significado
AAEV	Ambiente de AutoEvaluación

1.4. Referencias

- [IEEE93] IEEE Std 830-1993 (Revision of IEEE Std 830-1984). Software Engineering Standards Committee of the IEEE Computer Society.

1.5. Vision general

1.5.1. Descripción del contenido del resto del documento

Se dividirá en 2 partes. La sección 2 describirá los factores generales que afectaran al producto y sus requerimientos. La sección 3 describirá los requisitos con profundo detalle

1.5.2. Organización del documento

Estas especificaciones de requisitos software siguen las recomendaciones del estándar [IEEE93] en cuanto a la organización de los distintos apartados así como el contenido de cada uno de ellos.

2. Descripción general

2.1. Perspectiva del producto

2.1.1. Independencia del producto

El Ambiente de AutoEvaluación es un producto independiente

2.1.2. Interfaces del sistema

El sistema debe interactuar correctamente con el servidor Apache 2 y la base de datos desarrollada en MySQL 5.5, el sistema operativo especificado es Debian Jessie

2.1.3. Limitaciones de memoria

No se establecen limitaciones en cuanto a la cantidad de memoria, tanto secundaria como primaria que el ambiente deba usar

2.1.4. Operaciones

2.1.4.1. Modos de operación de los distintos tipos de usuarios

Hay 3 tipos de usuarios con sus correspondientes privilegios:

- Administrador: Gestiona las cuentas de los docentes y las carreras
- Docente: Gestiona los exámenes de la materia, así como unidades y aceptar las inscripciones de alumnos a la materia

- Alumno: Realiza los exámenes creados previamente por el/los docente/s en el ambiente, para llevar a cabo esta operación debe estar previamente inscripto en la materia

El modo de operación es interactivo con la interfaz de usuario de la pagina, el resto de las operaciones que no sean interactivas se realizaran automáticamente por la pagina según la función deseada por el usuario

2.1.4.2. Funciones de respaldo del procesamiento de datos

Ninguna especificada

2.1.4.3. Operaciones de backup y recuperación

Ninguna

2.2. Funciones del producto

(Ver DFD)

2.3. Características de usuario

La aplicación va dirigida a docentes y alumnos en el ámbito académico. Por ende, se encamina a un usuario con conocimientos sobre la realización de exámenes escritos.

2.4. Limitaciones generales

El sistema solo debe tener instancias de evaluación, no soportara subida de apuntes ni comunicación entre alumnos y/o docentes.

El sistema será desarrollado en el framework Django (versión 1.7.7), de Python (versión 2.7), usara MySQL 5.5 para la persistencia de datos y el servidor usado será Apache 2

2.5. Supuestos y dependencias

El alta de los usuarios depende de cada privilegio, ya que los docentes son registrados por el administrador y los alumnos tienen un registro propio que les dará el ambiente.

2.6. Requisitos futuros

Estadísticas de las notas de los alumnos por examen: para lograr comprender cuales conceptos son los más difíciles de absorber para los alumnos

Estadísticas de participación de alumnos en exámenes: para ver como es la participación de los alumnos en el ambiente y su interés por la refinación de conocimientos adquiridos en la materia

Estadísticas de notas por materia: para saber qué materia es la más difícil de comprender para los alumnos

3. Requisitos específicos

3.1. Requisitos funcionales

3.1.1. Requisito funcional 1: Gestionar Carreras (Actor: Administrador)

3.1.1.1. Introducción

El administrador de la página realizara el alta, baja y modificación de las carreras del ámbito académico universitario.

3.1.1.2. Datos Entrada

Las entradas de datos serán las siguientes si se realiza el alta:

- Nombre de la carrera
- Cantidad de años

En caso de modificación, la entrada será ya sea el nombre de la carrera, o la cantidad de años, dependiendo de que atributo se desee modificar.

En caso de baja:

- Selección de carrera registrada
- Respuesta al mensaje de confirmación (ver procesamiento)

3.1.1.3. Procesamiento

El sistema realizara las siguientes acciones luego de las entradas en caso de que se desee agregar una carrera:

- Validación de entradas (si la carrera ya existe o si los datos son erróneos), si es incorrecto, cancelar la operación e informar al usuario
- Registro en la base de datos de la carrera

En caso de modificar la carrera:

- Validación de entradas (si los datos son correctos y si en caso del nombre de la carrera existe una con ese valor ingresado)
- Actualización en la base de datos

En caso de eliminarse la carrera:

- Validación de entradas (si la carrera existe en la base de datos)
- Mensaje de confirmación que consultara a usuario si desea eliminar dicha carrera
- Actualización de la base de datos

3.1.1.4. Datos salida

El sistema brindará la siguiente información al ingresar, modificar o eliminar la carrera:

- Informe de éxito (en caso de que la operación se haya validado y completado)
- Informe de error (en caso de que la operación no se haya validado), indicando el error que hubo

3.1.2. Requisito funcional 2: Gestionar docentes (Actor: Administrador)

3.1.2.1. Introducción

El administrador de la pagina realizara el alta, baja o modificación de los docentes de la institución, ingresando sus datos en un formulario

3.1.2.2. Datos Entrada

En el caso de alta, el administrador ingresará los siguientes datos:

- Nombres del docente
- Apellidos
- E-mail
- DNI
- Usuario
- Contraseña

En el caso de modificación, se elegirá uno o varios de los campos que se usan en el alta, y se ingresara el nuevo valor de dicho atributo

En la baja, se lo seleccionara en la interfaz de usuario, y se confirmara su baja mediante un mensaje de confirmación (ver procesamiento)

3.1.2.3. Procesamiento

En el caso del alta, el sistema realizara las siguientes acciones:

- Validar datos (Si existe en el sistema, o si alguno de los datos fue ingresado incorrectamente)
- Agregar a la base de datos

En el caso de la modificación:

- Validar datos (Si alguno de los datos fue ingresado incorrectamente o en el caso de usuario si ya existe un usuario con ese mismo valor ingresado)
- Actualizar la base de datos con los valores ingresados

En el caso de la baja:

- Confirmar baja (mensaje que preguntara al usuario si de verdad desea eliminar ese docente)
- Validar baja (si dicho docente existe en la base de datos)
- Eliminar de la base de datos el docente indicado con sus dependencias

3.1.2.4. Datos salida

Para la adición, modificación o eliminación de docentes del sistema se encuentran las siguientes salidas:

- Informe de éxito de la operación (si se valido y completo la operación)
- Informe de error en la operación (si la operación no es válida), indicando que error hubo

3.1.3. Requisito funcional 3: Gestionar materias (Actor: Docente)

3.1.3.1. Introducción

El docente agrega, modifica o elimina una materia, ingresando los datos de esta en un formulario. Si la materia se elimina, los exámenes dentro de la misma también serán eliminados

3.1.3.2. Datos Entrada

Los datos para agregar una materia son:

- Nombre
- Año de carrera

Para modificarla pueden ser alguna o varias de las mencionadas en el alta.

Para eliminar una materia, se la seleccionara en la interfaz de usuario.

3.1.3.3. Procesamiento

En el caso del alta, el sistema realizara las siguientes acciones:

- Validar datos (Si existe en el sistema, o si alguno de los datos fue ingresado incorrectamente)
- Agregar a la base de datos

En el caso de la modificación:

- Validar datos (Si alguno de los datos fue ingresado incorrectamente o en el caso de nombre de materia si ya existe una con ese mismo valor ingresado, o que su carrera sea válida)
- Actualizar la base de datos con los valores ingresados

En el caso de la baja:

- Confirmar baja (mensaje que preguntara al usuario si de verdad desea eliminar esa materia)
- Validar baja (si dicha materia existe en la base de datos)
- Eliminar de la base de datos la materia indicada con sus dependencias

3.1.3.4. Datos salida

Para la adición, modificación o eliminación de materias del sistema se encuentran las siguientes salidas:

- Informe de éxito de la operación (si se valido y completo la operación)
- Informe de error en la operación (si la operación no es válida), indicando que error hubo

3.1.4. Requisito funcional 4: Gestionar examen (Actor: Docente)

3.1.4.1. Introducción

En este caso de uso, el docente realizara el alta, baja o modificación de un examen, durante la cual ya sea en el alta o en la modificación, se validaran y se realizara la gestión de opciones de las preguntas, asi como el sistema de puntuación del examen mismo

3.1.4.2. Datos Entrada

Para realizar el alta de un examen, el docente deberá ingresar:

- La cantidad de preguntas del examen
- La puntuación total del examen
- La descripción del examen (si se desea)
- La gestión de opciones del examen a crear

3.1.4.3. Procesamiento

En el caso del alta, el sistema realizara las siguientes acciones:

- Validar datos (Si existe en el sistema, o si alguno de los datos tanto del examen como de sus opciones fue ingresado incorrectamente)
- Agregar a la base de datos

En el caso de la modificación:

- Validar datos (Si alguno de los datos fue ingresado incorrectamente)
- Modificar opciones
- Actualizar la base de datos con los valores ingresados

En el caso de la baja:

- Confirmar baja (mensaje que preguntara al usuario si de verdad desea eliminar ese examen)
- Validar baja (si dicho examen existe en la base de datos)
- Eliminar de la base de datos el examen indicado con sus dependencias

3.1.4.4. Datos salida

Para la adición, modificación o eliminación de exámenes del sistema se encuentran las siguientes salidas:

- Informe de éxito de la operación (si se valido y completo la operación)
- Informe de error en la operación (si la operación no es válida), indicando que error hubo

3.1.5. Requisito funcional 5: Aceptar Alumnos (Actor: Docente)

3.1.5.1. Introducción

El docente tiene a su disposición la lista de alumnos que desean ingresar a la materia para realizar los exámenes, es decisión del usuario si rechaza o no a los solicitantes.

3.1.5.2. Datos Entrada

El docente recibirá la siguiente información:

- La lista de los alumnos que solicitan sus datos (podrá ver en detalle los datos de ellos)
- La materia a la que desean inscribirse

3.1.5.3. Procesamiento

El docente tiene 2 caminos, aceptar o rechazar la solicitud de ingreso.

Para aceptar:

- Se selecciona el alumno y se lo da de alta eligiendo la opción “Aceptar” (esto actualizará la base de datos)
- Se selecciona el alumno y se lo rechaza eligiendo la opción “Rechazar” (esto eliminara el registro de inscripción de la base de datos)

3.1.5.4. Datos salida

Debido a los 2 casos posibles del requisito, las salidas serán distintas para cada caso:

En caso de aceptar:

- Informe de éxito si la operación fue completada sin errores
- Alumno registrado en la materia, posibilitándole hacer exámenes de la misma
- Datos del alumno eliminados de la lista de solicitudes

En caso de rechazar:

- Informe de éxito si la operación fue completada sin errores
- Datos del alumno eliminados de la lista de solicitudes

3.1.6. Requisito funcional 6: Exportar examen (Actor: Docente)

3.1.6.1. Introducción

El docente puede exportar el examen previamente creado a un archivo con formato reutilizable, para usarlo luego a su decisión

3.1.6.2. Datos Entrada

Las entradas de datos serian:

- El examen seleccionado
- La opción de exportación

3.1.6.3. Procesamiento

Se realizaran los siguientes procesos para satisfacer el requisito:

- Conversión del examen a un archivo
- Descarga del archivo convertido con el examen

3.1.6.4. Datos salida

Las salidas correspondientes al requisito son:

- El archivo descargado con el examen

- Informe de éxito en caso de que no haya habido errores en la operación
- Informe de error en caso de que haya habido errores en la operación

3.1.7. Requisito funcional 7: Registrarse (Actor: Alumno)

3.1.7.1. Introducción

El alumno registrara sus datos en el sistema completando un simple formulario de registro que le brindara el ambiente.

3.1.7.2. Datos Entrada

Los datos de entrada serian:

- Nombres
- Apellidos
- Mail
- DNI
- Usuario
- Contraseña

3.1.7.3. Procesamiento

Se realizaran los siguientes procesos para satisfacer el requisito:

- Ingreso de datos del usuario
- Validación de datos del usuario ingresados
- Registrar usuario en la base de datos

3.1.7.4. Datos salida

Las salidas correspondientes al requisito son:

- Alumno registrado en la base de datos (en caso de éxito)
- Informe de éxito en caso de que los datos ingresados sean validos y el registro se haya completado
- Informe de error en caso de que los datos ingresados no sean validos o el registro no pudo completarse

3.1.8. Requisito funcional 8: Inscribir a materia (Actor: Alumno)

3.1.8.1. Introducción

El alumno seleccionara una de las materias de la carrera para enviarle una solicitud de ingreso a los/el docente/s de las mismas que ellos aceptaran o rechazaran (ver Requisito funcional 5)

3.1.8.2. Datos Entrada

Los datos de entrada serán:

- Datos del alumno

- Datos de la materia

3.1.8.3. Procesamiento

Se realizaran los siguientes procesos para satisfacer el requisito:

- Actualizar la base de datos (registrar el ingreso incompleto del alumno en la materia)
- Agregar alumno a la lista de solicitudes pendientes de la materia

3.1.8.4. Datos salida

Las salidas correspondientes al requisito son:

- Registro de los datos de la solicitud en la base de datos
- Alumno agregado a la lista de solicitudes pendientes de la materia
- Informe de éxito al alumno
- Informe de error al alumno

3.1.9. Requisito funcional 9: Realizar examen (Actor: Alumno)

3.1.9.1. Introducción

El Alumno, una vez aceptada la solicitud en la materia, elige uno de los exámenes realizados por los docentes y comienza su autoEvaluación en el sistema, al final de dicha se le indicara su nota y cantidad de preguntas correctas e incorrectas

3.1.9.2. Datos Entrada

Los datos de entrada serán:

- El alumno que realizará el examen
- El examen propiamente dicho
- Las respuestas del alumno al mismo

3.1.9.3. Procesamiento

Se realizaran los siguientes procesos para satisfacer el requisito:

- Traer examen de la base de datos
- Alumno responde el examen
- El sistema corrige el examen e indica los aciertos y errores del alumno, su nota final y si aprobó o no
- El sistema registra la nota obtenida y actualiza la cantidad de intentos de ese alumno al hacer el examen

3.1.9.4. Datos salida

Las salidas correspondientes al requisito son:

- Examen realizado
- Resultados del examen (nota, condición de aprobación, aciertos y errores)
- Intentos del alumno en ese examen
- Actualización de la base de datos en nota e intentos

3.2. Requisitos de interfaz externa

3.2.1. Interfaces de usuario

General:

- Saludo de bienvenida al sistema y formulario para datos de ingreso al sistema (usuario y clave), así como una opción para registrarse que aclare que es solo para alumnos

Administrador:

Comprenderá las siguientes interfaces:

- Gestión de carreras, donde se realiza el alta, baja o modificación de las mismas
- Gestión de docentes, donde se realiza el alta, baja o modificación de los mismos

Docente:

Comprenderá las siguientes interfaces:

- Gestión de materias, donde se realiza el alta, baja o modificación de las mismas
- Consulta y aceptación de solicitudes, donde se ve la lista de alumnos en espera de la aceptación y se decide si aceptarlas o no
- Gestión de exámenes, donde se realiza el alta, baja o modificación de los mismos

Alumno:

Comprenderá las siguientes interfaces:

- Registro al sistema, donde el usuario deberá ingresar sus datos
- Consulta e inscripción a materias, donde se muestran las materias de la carrera del alumno
- Consulta de exámenes por materia, donde aparecen los exámenes según la materia seleccionada
- Búsqueda de exámenes por materia, donde aparecen los exámenes según la materia ingresada en el buscador
- Interfaz de autoevaluación, donde aparecerá el examen con sus respectivas preguntas

3.2.2. Interfaces de hardware

Ninguna especificada.

3.2.3. Interfaces de software

Ninguna especificada.

3.2.4. Interfaces de comunicaciones

- Email para notificaciones

3.3. Requisitos de ejecución

Tener instalado el servidor Apache 2 y la base de datos MySQL versión 5.5 en la maquina que administre el servidor

3.4. Restricciones de diseño

3.4.1. Acatamiento de estándares

- Especificación de Requisitos Software: IEEE Std. 830/1983
- Proyecto: IEEE Std. 1074/1989

3.4.2. Limitaciones de hardware

Ninguna especificada

Índice

<u>1.- Introducción</u>	<u>Pág. 2</u>
<u>1.1 Objetivo</u>	<u>Pág. 2</u>
<u>1.2 Ámbito</u>	<u>Pág. 2</u>
<u>1.3 Definiciones, Siglas y Abreviaturas</u>	<u>Pág. 2</u>
<u>1.4 Referencias</u>	<u>Pág. 3</u>
<u>1.5 Visión general</u>	<u>Pág. 3</u>
<u>2.- Descripción General</u>	<u>Pág. 3</u>
<u>2.1 Perspectiva del producto</u>	<u>Pág. 3</u>
<u>2.2 Funciones del producto</u>	<u>Pág. 4</u>
<u>2.3 Características del Usuario</u>	<u>Pág. 4</u>
<u>2.4 Limitaciones Generales</u>	<u>Pág. 4</u>
<u>2.5 Supuestos y dependencias</u>	<u>Pág. 4</u>
<u>2.6 Requisitos futuros</u>	<u>Pág. 4</u>
<u>3.- Requisitos específicos</u>	<u>Pág. 5</u>
<u>3.1 Requisitos funcionales</u>	<u>Pág. 5</u>
<u>3.1.1 Requisito funcional 1</u>	<u>Pág. 5</u>
<u>3.1.2 Requisito funcional 2</u>	<u>Pág. 6</u>
<u>3.1.3 Requisito funcional 3</u>	<u>Pág. 7</u>
<u>3.1.4 Requisito funcional 4</u>	<u>Pág. 7</u>
<u>3.1.5 Requisito funcional 5</u>	<u>Pág. 8</u>
<u>3.1.6 Requisito funcional 6</u>	<u>Pág. 9</u>
<u>3.1.7 Requisito funcional 7</u>	<u>Pág. 10</u>
<u>3.1.8 Requisito funcional 8</u>	<u>Pág. 10</u>
<u>3.1.9 Requisito funcional 9</u>	<u>Pág. 11</u>

<u>3.2 Requisitos de interfaz externa</u>	<u>Pág. 12</u>
<u>3.2.1 Interfaces de usuario</u>	<u>Pág. 12</u>
<u>3.2.2 Interfaces de hardware</u>	<u>Pág. 12</u>
<u>3.2.3 Interfaces de software</u>	<u>Pág. 12</u>
<u>3.2.4 Interfaces de comunicaciones</u>	<u>Pág. 12</u>
<u>3.3 Requisitos de ejecución</u>	<u>Pág. 12</u>
<u>3.4 Restricciones de diseño</u>	<u>Pág. 13</u>
<u>3.4.1 Acatamiento de estándares</u>	<u>Pág. 13</u>
<u>3.4.2 Limitaciones hardware</u>	<u>Pág. 13</u>