

DESCOMPOSICIÓN DEL DOMINIO

Clases Conceptuales —Candidatas—

Dominio: El dominio planteado por el problema del proyecto describe un escenario que conjuga procesos de enseñanza tradicional con la enseñanza asistida por software para facilitar el aprendizaje y fortalecimiento de habilidades sobre una variedad de herramientas tecnológicas. Esta descripción corresponde, entonces, al dominio educación, en particular, el dominio **EdTech** (Educational Technology) o Tecnología Educativa.

Clases: Análisis e identificación deliberada de clases.

Clases candidatas	Descripción	Atributos (Suficientes)
• Usuario	<i>Toda persona registrada en la aplicación. Superclase.</i>	<i>userFirstName, userLastName, userIDnum, userRole</i>
• Participante	<i>Usuario con el rol de estudiante o aprendiz del curso y que practica exámenes.</i>	
• Profesor	<i>Usuario con rol de impartir un curso.</i>	
• Administrador	<i>Usuario facultado para administrar usuarios y exámenes.</i>	
• Login	<i>Todo usuario tiene asociado un nombre de usuario y contraseña (credenciales) para iniciar sesión.</i>	<i>Login (correo o cédula), password, accessGranted</i>
• Categoría	<i>Cada examen se categoriza en un grupo de tecnologías específico.</i>	<i>catID, catDescription</i>
• Curso	<i>Cada examen práctico es parte de un curso presencial asociado.</i>	<i>courseID, courseName, courseDescription, startDate, endDate</i>
• Examen	<i>Componente esencial de práctica.</i>	<i>testMaxTime, testDate, testAborted, numQuestions</i>
• Pregunta	<i>Unidad mínima esencial de cada examen con estructura lógica propia.</i>	<i>questionID, questionText</i>
• Dominio	<i>Cada examen clasifica sus preguntas por dominios o temas.</i>	<i>domainID, domainName</i>
• Respuesta	<i>Toda pregunta tiene una respuesta asociada y con estructura lógica propia.</i>	<i>answerID, answerOptions, correctAnswer, Explanation isAnswerCorrect</i>

● Instrucción	<i>Cada examen tiene un conjunto de instrucciones a aceptar.</i>	<i>instructionsText, agreed (bool)</i>
● Historial	<i>Registro histórico de cada examen tomado por un estudiante que incluye total de respuestas aprobadas y reprobadas.</i>	
● Certificado	<i>Comprobante de participación emitido tras la finalización de un curso.</i>	
● Resultado	<i>Todo examen tiene criterios de evaluación que producen un resultado.</i>	
● Rol	<i>Todo usuario del sistema tiene un rol asociado.</i>	
● InstituciónEducativa	<i>Institución que imparte el curso y por ende emite el certificado; Escuela de Computación de la UCV o una dependencia interna de la Escuela de Computación.</i>	
● Temporizador	<i>Todo examen es cronometrado.</i>	
● SesiónExamen	<i>Cada intento de examen es una nueva sesión de examen.</i>	

Clases potencialmente candidatas que describen el dominio:

usuario, administrador, curso, examen, dominio, pregunta, respuesta, categoría, certificado

No todas las clases deben ser identificadas en esta etapa; solo las que tienen relación estrecha con el dominio; las que lo definen o enriquecen. Habría que debatir el resto de las clases para determinar su real potencialidad para describir el dominio y eliminar aquellas en redundancia.

Consideraciones aleatorias:

- La aplicación no ofrece o dicta cursos; solo apoya a cursos presenciales.
- Cada curso puede tener más de un tipo de examen.
- Un mismo examen pudiera ser tomado más de una vez (“la última vez que hizo cada tipo de examen”).
- A nivel de sistema, **“participante”**, **“administrador”**, **“profesor”** son roles y puede plantearse de distintas formas. El rol pudiera ser un atributo “userRole” de un usuario y en función de ese atributo se asignan las vistas del sistema. El rol pudiera ser una clase

asociada a cada usuario. O simplemente no se manejan roles y se mantienen clases separadas para cada rol.

- Inicialmente se planteó el uso de una superclase persona; sigue siendo válido, pero puede simplificarse y emplear “usuario”, que es el vocabulario que predomina en el planteamiento.
- El **inicio de sesión** se puede manejar desde una clase separada a la de User; cada usuario puede tener un objeto Login. Tiene beneficios separar la información personal y la del login; hay métodos que serán únicos para validar login y no necesariamente validan la información de un usuario. Solo se plantearía la estructura del login y no la implementación.
- El certificado no se emite o genera tras la completación de cada examen sino al completar todos los exámenes asociados al curso; implicaría llevar un registro/historial el cual también es sugerido por “la última vez que hizo cada tipo de examen”.
- El modelo del dominio **no debe llevar métodos en las clases; estrictamente/únicamente atributos.**
- Se puede enriquecer el modelo si los nombres de las clases se ajustan al vocabulario del dominio en vez de usar nombres genéricos. Ejemplo: en vez de “Domain”, que es muy genérico, se puede considerar “TestDomain” “ExamDomain”. En vez de Category para los cursos, puede considerarse “CourseCategory”. Creo que usar el vocabulario del dominio fortalece el modelo.
- El profesor: o se acepta como el administrador y se elimina como clase; colocándose solo como atributo a clase «curso» **o...** hay niveles de administrador.
- Se presume que «los encargados» son los administradores, que a su vez serían los profesores. *Aclarar implicación con punto anterior.*
- Las reglas y políticas representan una categoría de clase conceptual (*Craig Larman*, UML y Patrones) → ¿Las instrucciones que deben ser aceptadas antes de cada examen caen en este renglón?

Preguntas a resolver en próximas etapas:

1. ¿Cómo un administrador recibe los datos de un participante para hacer el registro?
2. Un profesor que dicta o ha dictado cursos ha decidido tomar uno de los cursos ofrecidos ¿cómo es el procedimiento si ya está registrado con el rol “profesor”?
3. ¿El participante tendrá acceso al sistema incluso luego de haber culminado el curso?
¿El acceso termina tan pronto finaliza el curso?

Reglas:

- La aplicación debe permitir a los administradores registrar usuarios.
- La aplicación debe permitir a los usuarios en general iniciar sesión.
- La aplicación debe permitir que se edite la información del perfil.
- El registro debe ser muy sencillo, solo información necesaria para generar una certificación de participación en el curso.
- Las preguntas se deben clasificar por dominios.

- Una pregunta puede pertenecer a uno o más dominios.
- Las preguntas serán de selección simple, con cuatro o más respuestas, donde una sola de ellas es correcta.
- Los exámenes no tendrán marcas negativas y se podrán terminar de dos maneras.
- La duración y el número de preguntas del examen debe ser escogido por el administrador al momento de crearlo.