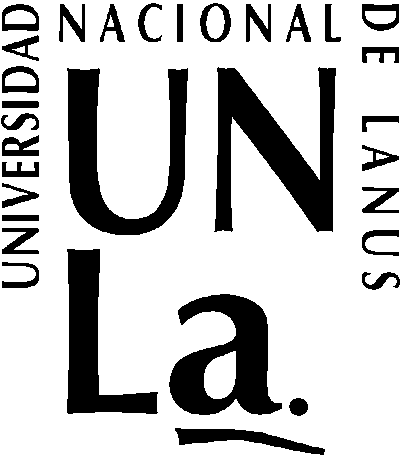
****

**Sistema Gestor de Corrección de TPs (SCTps)**

Javier Rodríguez

Diego Saavedra Griott

Laura Vallejos

Juan Ignacio Yone

**Cátedra Proyecto de Software**

**Licenciatura en Sistemas**

**2017**

Sistema Gestor de Corrección de Tps

Javier Rodríguez - Diego Saavedra Griott - Laura Vallejos - Juan Yone

Licenciatura en Sistemas – Departamento de Desarrollo Productivo y Tecnológico

Universidad Nacional de Lanús

*Extracto*—En este documento se presentaran las actividades realizadas en base al Modelo de Proceso para la entrega del primer hito. *(Abstracto)*

# **Introducción**

Este proyecto surgió ante la necesidad de crear un sistema que permitiera el gestionamiento de trabajos prácticos.

Constará de un sistema que será utilizado tanto por alumnos por profesores.

En donde los profesores tendrán la posibilidad de crear trabajos prácticos y se les permitirá a los alumnos realizar las correcciones. Constatando sus respuestas con las administradas como correctas por los profesores.

1. **Proceso de Selección de un Modelo de Ciclo de Vida del Producto**
   1. **Identificar los posibles MCVS:**

En esta sección daremos explicación de los motivos por los cuales optamos por seleccionar el ciclo de vida elegido para este proyecto.

Para comenzar es necesario dar a conocer los tres principales ciclos de vida que se suelen utilizar al trabajar. Detallar sus diferencias y ventajas respecto a los otros.

Ciclos de vida:

* Cascada: Descompone la actividad del proyecto en transiciones lineales dentro de las fases que lo comprenden. Cada una de ellas se realiza una única vez, y a continuación de la anterior. Facilita la división de las tareas y preveer los tiempos.
* Prototipado: Es utilizado en proyectos los cuales no se conoce con exactitud como desarrollarlo ya sea por ser proyectos con innovaciones importantes o porprever utilizar tecnologías poco conocidas lo cual impide iniciar un proyecto de forma lineal.

Al no conocer exactamente como desarrollar un producto suele recurrirse a un prototipo el cual es un producto parcial y provisional.

Su evaluación y desarrollo permitirán la definición de especificaciones más complejas y seguras para el producto definitivo.

Este ciclo de vida repite las fases dos veces, una para el prototipo y otra para el producto final.

* Espiral: Suele utilizarse en proyectos en los que se necesita la evaluación de más de un prototipo para asegurar la desaparición de incertidumbres y/o ignorancias.

El producto real puede considerarse como una sucesión de prototipos que progresan hasta alcanzar el producto deseado. En cada ciclo las especificaciones se van resolviendo paulatinamente.

|  |  |
| --- | --- |
|  |  |

* 1. **Seleccionar un modelo para el proyecto:**

Debido a lo antes explicado, para el actual proyecto procederemos a utilizar el ciclo de vida prototipado. En el cual definiremos especificaciones iniciales para desarrollar un prototipo el cual será un producto parcial y provisional.

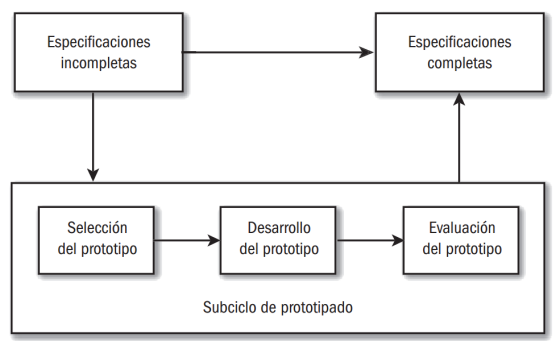


Figura 1. Ciclo de vida del prototipado.

|  |  |
| --- | --- |
| **II. Proceso de Gestión del Proyecto**   * 1. **Establecer la matriz de actividades para el MCV**   El proyecto constara con la utilización de una maqueta interactiva, y un prototipo con algunas funcionalidades iniciales.   * 1. **Asignar los recursos del proyecto**   Recursos:  Recursos humanos:  El presente proyecto estará a cargo de Javier Rodríguez, Diego Saavedra Griott, Laura Vallejos y Juan Yone.  Recursos tecnológicos:  El sistema actual enfocado para ser utilizado en dispositivos con sistema Windows, en el futuro se proveerá la incorporación de nuevos sistemas para su uso.  El dispositivo en cuestión será desarrollado en el lenguaje Python 2.7, se utilizaran sockets para la comunicación entre cliente y servidor (Debian Jessie, Linux) .  Se utilizara MySQL v5.5 como gestor de bases de datos para almacenar los usuarios y los trabajos prácticos. | * 1. **Planificar la gestión del Proyecto**   Diagrama de Gantt:  El proyecto fue inicializado el día 18/08/2017 y su terminación está planificada para el día 10/11/2017.  La distribución de tarea fue pensada en base a estas fechas. En el diagrama de Gantt se podrán ver en detalle la planificación planteada para el proyecto.    Figura 2. Diagrama de Gantt.  Mapa de Actividades:  En la siguiente tabla figura el mapa de actividades diseñado en correlación con el ciclo de vida elegido para el proyecto: |

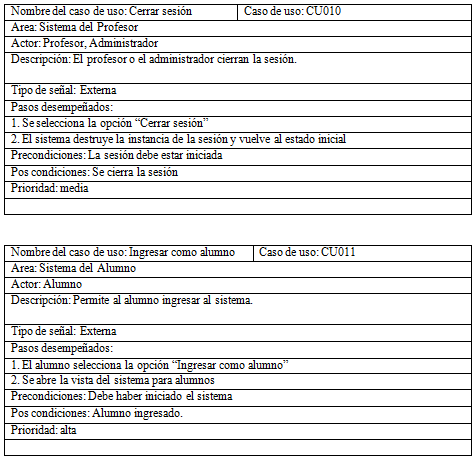
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Prototipo Maqueta | |  | Prototipo evolutivo | |  | |  | Instalación y Operación | | | |
|  | AP | ERI | DEI | VIU | ERS | DAC | C | | PP | IF | | OM | R |
| **Proceso de Selección de un MCVS** |  |  |  |  |  |  |  | |  |  | |  |  |
| Identificar los posibles MCVS | x |  |  |  |  |  |  | |  |  | |  |  |
| Seleccionar un modelo para el proyecto | x |  |  |  |  |  |  | |  |  | |  |  |
| **Proceso de Iniciación, Planificación y Estimación del Proyecto** |  |  |  |  |  |  |  | |  |  | |  |  |
| Establecer la matriz de actividades para el MCVS | x |  |  |  |  |  |  | |  |  | |  |  |
| Asignar recursos del proyecto | x | x | x | x | x | x | x | | x |  | |  |  |
| Definir el entorno del proyecto | x |  |  |  |  |  |  | |  |  | |  |  |
| Planificar la gestión del proyecto | x |  |  |  |  |  |  | |  |  | |  |  |
| **Proceso de Seguimiento y Control del Proyecto** | | | | | | | | | | | | | |
| Analizar riesgos | x | x | x | x | x | x | x | |  |  |  | |  |
| Realizar la planificación de contingencias |  | x | x | x | x | x | x | | x |  |  | |  |
| Gestionar el proyecto |  | x | x | x | x | x | x | | x |  |  | |  |
| Implementar el sistema de informes de problemas |  | x | x | x | x | x | x | | x |  |  | |  |
| Archivar registros |  | x | x | x | x | x | x | | x |  |  | |  |
| **Proceso de Gestión de Calidad del Software** | | | | | | | | | | | | | |
| Planificar la garantía de calidad del Software |  | x | x |  | x | x | x | |  |  | |  |  |
| Desarrollar métricas de calidad |  | x | x |  | x | x | x | |  |  | |  |  |
| Gestionar la calidad del software |  | x | x | x | x | x | x | | x |  | |  |  |
| Identificar necesidades de mejora de la calidad |  | x | x | x | x | x | x | | x |  | |  |  |
| **Proceso de Exploración de Conceptos** | | | | | | | | | | | | | |
| Identificar las ideas o necesidades | x | x |  |  | x |  |  | |  |  | |  |  |
| Formular las soluciones potenciales | x | x |  |  | x |  |  | |  |  | |  |  |
| Dirigir los estudios de viabilidad | x | x |  |  | x |  |  | |  |  | |  |  |
| Planificar la transición del sistemas | x | x |  |  | x |  |  | |  |  | |  |  |
| Refinar y Finalizar la idea o necesidad | x | x |  |  | x |  |  | |  |  | |  |  |
| **Proceso de Asignación del Sistema** | | | | | | | | | | | | | |
| Analizar las funciones del sistema |  |  | x |  |  | x |  | | x |  | |  |  |
| Desarrollar la arquitectura del sistema |  |  |  |  |  | x |  | |  |  | |  |  |
| Descomponer los requisitos del sistema |  |  |  |  |  | x |  | |  |  | |  |  |
| **Proceso de Análisis de Requisitos** | | | | | | | | | | | | | |
| Definir y Desarrollar los requisitos de software |  |  |  |  | x |  |  | |  |  | |  |  |
| Definir los requisitos de Interfaz |  | x |  |  |  |  |  | |  |  | |  |  |
| Priorizar e Integrar los requisitos del Software |  |  |  |  | x |  |  | |  |  | |  |  |
| **Proceso de Diseño** | | | | | | | | | | | | | |
| Realizar el diseño preliminar |  |  | x |  |  | x |  | |  |  | |  |  |
| Analizar el flujo de información |  |  |  |  |  | x |  | |  |  | |  |  |
| Diseñar la base de datos (si se aplica) |  |  |  |  |  | x |  | |  |  | |  |  |
| Diseñar las interfaces |  |  |  |  |  | x |  | |  |  | |  |  |
| Seleccionar o Desarrollar algoritmos (si se aplica) |  |  |  |  |  | x |  | |  |  | |  |  |
| Realizar el diseño detallado |  |  |  |  |  | x |  | |  |  | |  |  |
| **Proceso de Implementación e Integración** | | | | | | | | | | | | | |
| Crear los datos de prueba |  |  |  |  |  | x | x | |  |  | |  |  |
| Crear el código fuente |  |  |  |  |  |  | x | |  |  | |  |  |
| Generar el código objeto |  |  |  |  |  |  | x | |  |  | |  |  |
| Crear la documentación de operación |  |  | x |  |  | x | x | |  |  | |  |  |
| Planificar la integración |  |  |  |  |  | x | x | |  |  | |  |  |
| Realizar la integración |  |  |  |  |  |  | x | |  |  | |  |  |
| **Proceso de Instalación y Aceptación** | | | | | | | | | | | | | |
| Planificar la instalación |  |  |  |  |  |  |  | |  | x | |  |  |
| Distribuir el software |  |  |  |  |  |  |  | |  | x | |  |  |
| Instalar el software |  |  |  |  |  |  |  | |  | x | |  |  |
| Cargar la base de datos |  |  |  |  |  |  |  | |  | x | |  |  |
| Aceptar el software en el entorno de operación |  |  |  |  |  |  |  | | x | x | |  |  |
| Realizar las actualizaciones |  |  |  |  |  |  |  | |  | x | |  |  |
| **Proceso de Operación y Soporte** | | | | | | | | | | | | | |
| Operar el sistema |  |  |  |  |  |  |  | |  |  | | x |  |
| Proveer de asistencia técnica y consultas |  |  |  |  |  |  |  | |  |  | | x |  |
| Mantener el histórico de peticiones de soporte |  |  |  |  |  |  |  | |  |  | | x |  |
| **Proceso de mantenimiento** | | | | | | | | | | | | | |
| Realizar el mantenimiento correctivo |  |  |  |  |  |  |  |  | |  | | x |  |
| Reaplicar el ciclo de vida del software |  |  |  |  |  |  |  |  | |  | | x |  |
| **Proceso de retiro** | | | | | | | | | | | | | |
| Notificar al usuario |  |  |  |  |  |  |  |  | |  | |  | x |
| Conducir operaciones en paralelo (si se aplica) |  |  |  |  |  |  |  |  | |  | |  | x |
| Retirar el sistema |  |  |  |  |  |  |  |  | |  | |  | x |

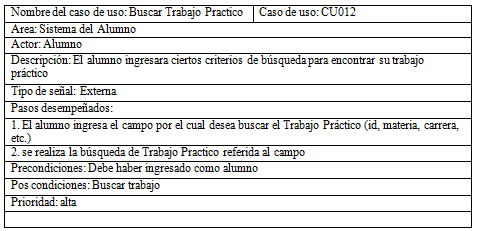
Figura 3. Mapa de actividades del Prototipo Evolutivo.

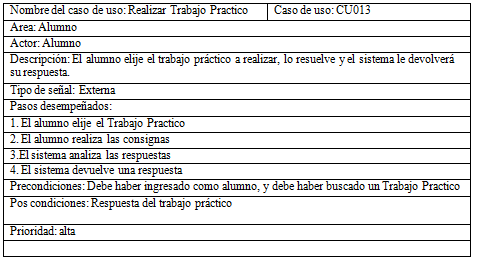
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tabla referencias:**   |  |  | | --- | --- | | Inicial | Significado | | AP | Análisis Preliminar | | ERI | Especificación requisitos de interfaz | | VIU | Validación Interfaz Usuario | | ERS | Especificación Requisitos de Software | | DAC | Diseño de Arquitectura del Software | | C | Codificación | | PP | Prueba Prototipo | | IF | Implementación Final | | OM | Proceso de operación y mantenimiento | | R | Retiro |   Figura 4. Tabla de referencias del mapa de actividades.   * 1. **Análisis de riesgos**   Los riesgos encontrados durante el análisis inicial del presente proyecto se encuentran detallados en la tabla que figura a continuación, detallando su descripción, su impacto, su probabilidad y la acción a seguir:  **TotalPuntos de función no ajustados: 196**  52+49+18+80=**199**  Calculo Nivel Total de Influencia (TDI)   |  |  | | --- | --- | | Caracteristica | Cantidad | | Comunicación de datos | 4 | | Procesamiento distribuido de datos | 2 | | Rendimiento | 3 | | Configuraciones fuertemente utilizadas | 3 | | Frecuencia de transacciones | 3 | | Entrada de datos on- line | 3 | | Eficiencia del usuario final | 4 | | Actualizaciones Online | 0 | | Procesamiento complejo | 3 | | Reusabilidad | 2 | | Facilidad de instalacion | 3 | | Facilidad de operación | 5 | | Instalacion de distintos lugares | 3 | | Facilidad de cambio | 3 | | Total TDI | 41 |   Figura II: Calculo Nivel total de Influencia  **Total TDI: 41**  **Factor de Ajuste:** (41\*0,01)+0,65= 1.06  **Puntos de Función**: 1.06\*199 = 210.94  **Líneas de código:**210.94\*50(aprox)= 10547  **Esfuerzo/Horas**: 210.94/(1/8)= 1687.52  **Horas por persona**: 1687.52/4= 421.88  **Distribución de tiempo:**  4 hs\* 20 días =80 horas por mes  421.88/80= **5 meses y 2 días aproximadamente**  **Esfuerzo/Horas**: 210.94/(1/8)= 1687.52  **Horas por persona**: 1687.52/4= 421.88  **Distribución de tiempo:**  4 hs\* 20 días =80 horas por mes  421.88/80= **5 meses y 2 días aproximadamente** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | # | Descripción | Impacto en el proyecto | Probabilidad | Acción a Seguir | | 1 | Problema de incompatibilidades de los archivos utilizados | El usuario no podrá realizar el trabajo práctico pedido. | Media | Intentar utilizar un solo tipo de archivo para la gestión de trabajos prácticos | | 2 | Problema a la hora de comparar los Strings. | El sistema no podrá reconocer el resultado con claridad | Alta | Se optó por armar trabajos prácticos con el formato de múltiple choice. |   Figura 5. Tabla de riesgos detallados.  Cálculo aproximado:   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | Tipo de componente | C Baja | Cant | C Media | Cant | C Alta | Cant | Total | | Entrada externa | 3 | 10 | 4 | 4 | 6 | 1 | 52 | | Salida externa | 4 | 8 | 5 | 2 | 7 | 1 | 49 | | Consulta externa | 3 | 2 | 4 | 3 | 6 | 0 | 18 | | Archivo logico interno | 7 | 0 | 10 | 2 | 15 | 4 | 80 | | Archivo interfaz externo | 5 | 0 | 7 | 0 | 10 | 0 | 0 |   Figura 6: Estimacion Puntos de Funcion no ajustados  **III. Proceso de Pre-Desarrollo**   * 1. **Proceso de Exploración de Conceptos**   Identificar las ideas o las necesidades:   * Comprobación de respuestas sin intervención del docente, rápida y simple. Almacenamiento en servidor de trabajos prácticos (enunciados y respuestas). * Vistas diferenciadas para alumnos y docentes, sin y con usuario respectivamente.   1. **Formular Soluciones potenciales**:   La idea original consistía de una aplicación web por medio de la cual, tanto docente como alumno tuvieran acceso. La misma fue descartada tras la primera entrevista con el cliente, donde se planteó la necesidad de desarrollar un dispositivo software del tipo cliente/servidor, donde se aprovechen los recursos del usuario.  Debido a esto y tras una segunda entrevista y maqueta, se decidió generar una aplicación cliente que diferencie sus vistas a través de un login. El mismo debe ser con contraseña en el caso de administradores o docentes, y como invitado en el caso de los alumnos. Pasada esta etapa, la interfaz varía dependiendo del tipo de usuario:   * Administrador:   + Es bienvenido con un home donde se muestran las peticiones de acceso pendientes (con sus respectivos datos).   + Tiene tres botones: crear, eliminar y resetear usuario (este último permite cambiarle la contraseña a un usuario en caso de olvido previa validación de su identidad). * Docente:   + Tiene 4 botones: crear, eliminar y visualizar trabajos prácticos y darse de baja . Al crear se le piden primero los datos de identificación del TP (título, materia y carrera), luego se le pide completar el primer enunciado y su respuesta, dándole la opción de terminar aquí o bien continuar para agregar más. |

|  |  |
| --- | --- |
| * Alumno:   + Su home es la búsqueda de TPs, a la izquierda completa cualquiera de los datos (id, profesor o materia) y a la derecha se le muestran los resultados de la búsqueda.   + Una vez elegido un TP, el alumno elige una respuesta del enunciado (de momento únicamente elección múltiple) y puede o pedir corrección hasta ahí o bien continuar con el resto de las preguntas( en cada respuesta se le permite pedir correcion al sistema).   El dispositivo en cuestión será desarrollado en el lenguaje Python 2.7, se utilizaran sockets para la comunicacion entre cliente y servidor ( Debian, Jessie, Linux) y MySQL v5.5 como gestor de bases de datos para almacenar los usuarios y los trabajos prácticos.   * 1. **Refinar y Finalizar la idea o necesidad:**   Como resultado de dos entrevistas presenciales con el cliente y su devolución sobre la maqueta preliminar, se puede estipular que el objetivo o necesidad del dispositivo software es facilitar al alumno una herramienta de autocorrección para utilizar a la hora de prácticas o estudio.  La misma debe permitir al docente visualizar, crear y eliminar trabajos prácticos y sus respectivas respuestas; al alumno buscar un trabajo práctico y corroborar sus respuestas contra el servidor sin interacción del docente, ya sea solo una pregunta o todas; por último el administrador se encargará de manejar las peticiones de registro de usuarios.  A su vez, desde el punto de vista de seguridad la aplicación debe limitar las vistas de los usuarios, los cuales serán provistos por un administrador únicamente aplicable a los docentes, los alumnos ingresarán libremente. | **IV. Procesos Orientados al Desarrollo del Software**  **I. Proceso de Requisitos**  Requisitos del Software:   * Registración de docentes y administradores en el sistema   Datos:   * + - Mail     - Nombre     - Apellido     - Usuario     - Contraseña * Autentificación de docentes y administradores en el sistema   Datos:   * + - Usuario     - Contraseña * Incorporación de acceso como invitado (alumno) * Creación de trabajos prácticos   Datos:   * + - Titulo     - Materia     - Carrera * Creación de consignas y respuestas verdaderas y falsas de los trabajos prácticos   Datos:   * + - Enunciados     - Respuesta V     - Respuesta F     - Respuesta F * Acceso a lista de trabajos prácticos anteriores realizados por los docente   Acceso para invitados a los trabajos prácticos a resolver. |

|  |  |
| --- | --- |
| Requisitos de usuario:   1. El usuario podrá realizar peticiones de acceso al sistema 2. El usuario(docente) podrá crear trabajos prácticos a resolver 3. El usuario(docente) podrá suministrarle a esos trabajos prácticos consignas y respuestas pertinentes 4. El usuario(docente) podrá indicar si esas respuestas son verdaderas o falsas 5. El usuario(docente) podrá indicar a que carrera y materia pertenece el trabajo practico 6. El usuario(administrador) podrá registrar peticiones de acceso al sistema 7. El usuario(administrador) podrá modificar y eliminar los usuarios de tipo docentes del sistema 8. El alumno podrá buscar el trabajo práctico indicado por el docente. 9. El alumno podrá responder el trabajo práctico. 10. El alumno deberá visualizar cuales respuestas fueron las correctas y cuales incorrectas.   Requisitos Funcionales:   1. Proceso de control de las respuestas suministradas tanto por el docente como por el alumno   Requisitos no Funcionales:   1. El docente no participará en el proceso de corrección de los trabajos prácticos 2. El sistema deberá restringir el acceso a la respuestas por medio de validaciones de usuario     Figura 7. Diagrama de Secuencias: Iniciar Sesión (Profesor), creación de trabajos prácticos, muestra y eliminación de trabajos prácticos, recuperación de contraseña y generación de la orden de baja.    Figura 9. Análisis de casos de uso: Se modela a los actores que interactúan con el sistema y los casos de uso que afectan. | **II. Procesos de Diseño**  Se optó por la programación Orientada a Objetos y desarrollamos los correspondientes diseños y diagramas de dicho paradigma.  Figura 5. Diagrama de Secuencias: Registro y Creación de usuario.  Figura 6. Diagrama de Secuencias: Iniciar sesión (Administrador)  Figura 8. Diagrama de Secuencias: Ingresar como Alumno, búsqueda de trabajos prácticos y su realización.  Figura 10. Diagrama de Casos de uso: Modela los casos de uso y todos los actores que interactúan con cada caso de uso.  Escenario de casos de uso:  En las siguientes tablas se ofrece información específica de cada caso de uso. |







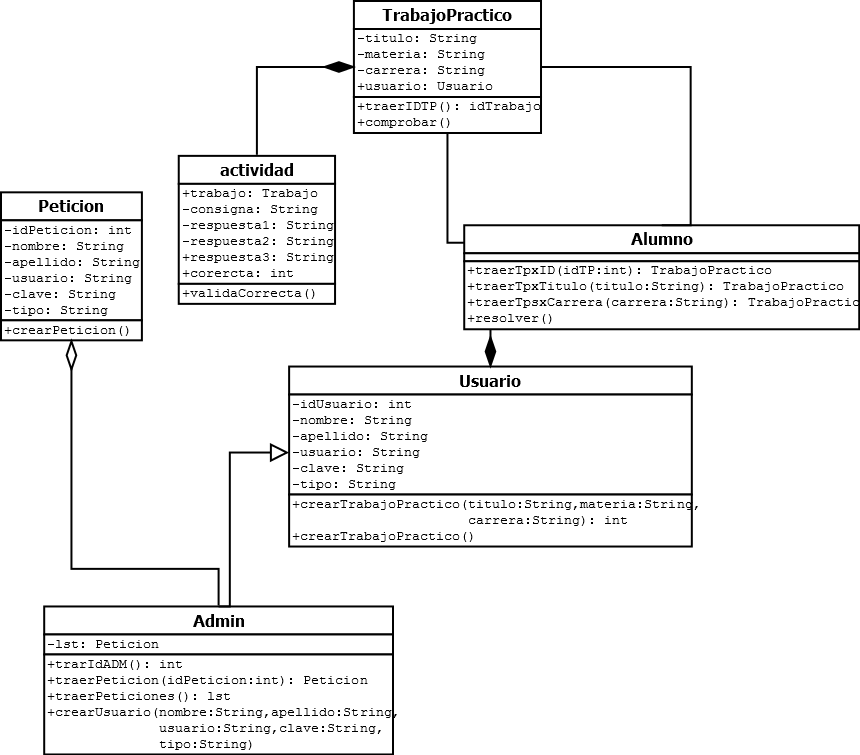


Figura 11. Diagrama de Clases: Modela las relaciones entre las clases del sistema.

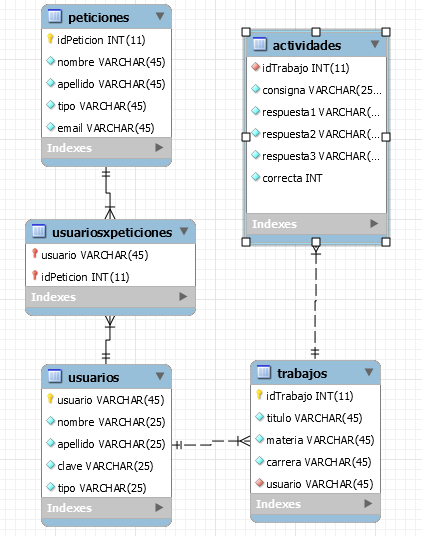


Figura 12. Diagrama de Entidad-Relación: Modelado de la base de datos utilizada.