### 1. Creación de Contenidos Interactivos

* **1.1 Crear interfaz de creación de contenidos**
  + Crear formulario para que los profesores puedan ingresar preguntas de tipo test, completar con palabras, código, etc. (HTML, CSS, Bootstrap, Angular).
* **1.2 Agregar soporte multimedia**
  + Permitir que los profesores suban videos, imágenes, y enlaces a artículos en las actividades (HTML, CSS, JavaScript, Angular).
* **1.3 Crear plantillas para las actividades**
  + Crear plantillas de preguntas interactivas prediseñadas (HTML, SCSS, PHP).

### 2. Gamificación de Clases

* **2.1 Implementación de sistema de puntos**
  + Desarrollar sistema de puntos que permita a los estudiantes ganar recompensas según su rendimiento (PHP, JavaScript, Angular).
* **2.2 Implementación de recompensas y niveles**
  + Implementar niveles de progreso según puntos acumulados (CSS, JavaScript, PHP).
* **2.3 Crear tablas de clasificación**
  + Desarrollar vista para mostrar el ranking de estudiantes (HTML, CSS, Bootstrap, Angular, JavaScript).

### 3. Plataforma de Evaluación en Tiempo Real

* **3.1 Crear cuestionarios interactivos**
  + Desarrollar formularios que permitan a los estudiantes realizar exámenes con corrección automática (HTML, CSS, JavaScript, Angular).
* **3.2 Implementación de retroalimentación instantánea**
  + Desarrollar lógica para dar feedback inmediato sobre respuestas correctas e incorrectas (JavaScript, PHP, Laravel).

### 4. Desafíos y Retos

* **4.1 Crear sistema de retos semanales/mensuales**
  + Implementar interfaz para organizar y visualizar retos de manera periódica (HTML, CSS, JavaScript, Angular).
* **4.2 Crear sistema de competiciones**
  + Desarrollar sistema para crear competiciones entre grupos o estudiantes (PHP, JavaScript, Laravel).

### 5. Comunicación entre Estudiantes y Profesores

* **5.1 Sistema de notificaciones**
  + Implementar un sistema de notificaciones para mantener a los usuarios informados sobre nuevas actividades, resultados, etc. (JavaScript, PHP, Laravel, Angular).

### 6. Personalización de Perfiles de Usuarios

* **6.1 Crear perfiles de usuario**
  + Desarrollar funcionalidad para que los estudiantes creen y personalicen sus perfiles (HTML, CSS, SCSS, PHP).
* **6.2 Personalización para profesores**
  + Crear una sección para que los profesores personalicen sus perfiles y los contenidos que proporcionan (PHP, Angular).
* **6.3 Visualización de logros y progreso**
  + Mostrar los logros y progreso de los estudiantes en sus perfiles (JavaScript, Angular, PHP).

### 7. Accesibilidad Multidispositivo

* **7.1 Diseño responsivo de la interfaz**
  + Crear una interfaz responsiva y adaptativa para diversos dispositivos (HTML, CSS, Bootstrap, SCSS).
* **7.2 Compatibilidad entre dispositivos**
  + Asegurarse de que la plataforma sea accesible desde PC, tabletas y móviles (HTML, CSS, SCSS, Bootstrap).

### 8. Soporte para Diversos Idiomas

* **8.1 Implementación de selección de idioma**
  + Crear una funcionalidad para cambiar el idioma de la plataforma (Angular, PHP).
* **8.2 Traducción de contenidos**
  + Desarrollar un sistema para traducir los contenidos de la plataforma a varios idiomas (PHP, Angular).

### 9. Informes Detallados para Profesores

* **9.1 Crear panel de informes**
  + Desarrollar panel para que los profesores vean informes detallados del rendimiento de sus estudiantes (HTML, CSS, PHP, Angular).
* **9.2 Generación de informes personalizados**
  + Implementar la generación de informes descargables en formatos como PDF o CSV (PHP, Laravel, JavaScript).

### 10. Infraestructura y Seguridad

* **10.1 Implementación de registro de usuarios**
  + Crear sistema de autenticación y registro de usuarios (PHP, Laravel, Angular).
* **10.2 Configuración de almacenamiento en AWS**
  + Configurar AWS S3 para el almacenamiento de archivos multimedia (AWS SDK, PHP).
* **10.3 Seguridad y protección de datos**
  + Implementar medidas de seguridad para proteger los datos de los usuarios (PHP, Laravel).
* **10.4 Implementación de copias de seguridad**
  + Crear sistema de copias de seguridad automáticas en la base de datos (AWS, PHP, Laravel).

### 11. Integración de Backend

* **11.1 Configuración de base de datos**
  + Configuración de la base de datos en Laravel para gestionar usuarios, contenidos, puntuaciones, etc. (Laravel, MySQL).
* **11.2 API para comunicación entre Frontend y Backend**
  + Desarrollar API RESTful para permitir la comunicación entre el frontend (Angular) y el backend (Laravel) (PHP, Laravel, Angular).

### 12. Pruebas y QA

* **12.1 Pruebas de usabilidad**
  + Realizar pruebas de usabilidad de la interfaz (todos los desarrolladores, UX).
* **12.2 Pruebas de integración**
  + Realizar pruebas de integración entre las diferentes tecnologías y servicios (todos los desarrolladores).
* **12.3 Pruebas de carga y rendimiento**
  + Realizar pruebas de carga para garantizar que la plataforma pueda manejar un gran número de usuarios simultáneos (AWS, Laravel).

### 13. Documentación

* **13.1 Documentación técnica**
  + Crear documentación detallada sobre el código y la arquitectura de la plataforma (Markdown, PHP, Laravel).
* **13.2 Manual de usuario**
  + Crear manual de usuario para estudiantes y profesores (HTML, CSS).