

**INSTRUMENTO DE EVALUACIÓN:**

**CUESTIONARIO**

** CENTRO DE ENSEÑANZA TECNICA INDUSTRIAL**

*Organismo Público Descentralizado Federal*

**DATOS GENERALES**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CARRERA | | **Desarrollo de software** | | | | UAC | Proyecto Integrador de desarrollo de software I | | | | FECHA | | 3/9/25 | | |
| **\*\*\* Evidencia o producto de aprendizaje** | | | | | | | | **Tipo de evaluación** | | | **Finalidad o momento** | | | | |
| No. Parcial | Clave Producto | Descripción | | | | | | Auto-evaluación | Co- evaluación | Hetero- evaluación | Diagnóstica | | Formativa | Sumaria | |
| 123 |  | **Propuesta de proyecto** | | | | | | **C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.png**  **√** | C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.png | **C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.png** | **C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.png** | | **C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.png** | **√** | |
| **COMPETENCIAS A EVALUAR** | | | | | | | | | | **Ponderación Parcial** | | | **Calificación obtenida** | | |
| **Competencia** | | **Claves** | **Competencia** | | | **Claves** | **Competencia** | | **Claves** |
| C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.pngGenérica | |  | Disciplinar Básica Disciplinar Extendida | | | **C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.pngC:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.png** | Profesional Básica Profesional Extendida  **√** | | **C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.pngC:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.png** |  | | |  | | |
|  |  |  | |  |  |  |  |  |  |  |  |  |  | |  |
| **NOMBRE DEL ALUMNO** | | **Emmanuel Buenrostro Briseño** | | | | | **GRUPO** | **7F** | **REGISTRO** | **22300891** | **FIRMA** | |  | | |
| **NOMBRE DEL ALUMNO** | | **Emiliano Arzate Gutiérrez** | | | | | **GRUPO** | **7F** | **REGISTRO** | **22300929** | **FIRMA** | |  | | |

|  |  |
| --- | --- |
| **INSTRUCCIONES** | Las propuestas de proyecto deben de contestar TODAS las siguientes preguntas.  Puede ser contestando después de cada una o en forma de un solo texto. |

|  |  |
| --- | --- |
| **NOMBRE DE LA PROPUESTA** | **Desarrollo de una plataforma para entrenamiento en Olimpiada de Matemáticas** |

|  |  |
| --- | --- |
| 1 | Cambios en esta versión. |

|  |  |
| --- | --- |
| 2 | ¿Cuál es el problema o inquietud? (Problema, necesidad, inquietud, hobbies, etc.) |

Tener una manera de poder dar un proceso de seguimiento a un entrenamiento para la olimpiada de matemáticas, ya sea siendo entrenador o alguien que quiera entrenar por su cuenta. Incluyendo la organización y calificación de exámenes tanto oficiales como de práctica.

|  |  |
| --- | --- |
| 3 | ¿A quién afecta el problema? (O en el caso de que surja de un hobbie o inquietud a quien va dirigido) |

A las personas que están o se están preparando para la olimpiada de matemáticas.

|  |  |
| --- | --- |
| 4 | ¿Cuál es el contexto o entorno en donde se aplicará? (Usuarios, condiciones, recursos, etc) |

No hay condiciones y recursos necesarios además de un dispositivo electrónico e internet, ya que cualquier usuario con eso podría acceder, pero se espera que la mayoría de usuarios sean participantes o entrenadores de la Olimpiada Mexicana de Matemáticas.

|  |  |
| --- | --- |
| 5 | En un enunciado ¿Qué se propone? |

Una página web y una aplicación móvil con una recopilación de distintos problemas de olimpiadas de matemáticas, los cuales en la aplicación móvil podrán ser guardados offline, clasificados por tags, dificultad y calidad, en la cual puedes publicar problemas, soluciones o comentarios y que además te da recomendaciones de problemas en base a lo que te estas preparando y los problemas que haz hecho; además de un sistema de exámenes ya sea de practica u oficiales, donde los oficiales pueden ser calificados en la misma plataforma, asignando calificadores y asignando puntajes a cada problema.

|  |  |
| --- | --- |
| 6 | ¿Cuál sería el objetivo general de la propuesta? (Puedes auxiliarte del punto anterior) |

Brindar recomendaciones y material para entrenar para la olimpiada de matemáticas, además de poder obtener experiencia real con distintos exámenes de práctica, ya sean simulaciones de exámenes reales, o un concurso organizado en la plataforma, los cuales podrán calificar los problemas entregados en los concursos.

|  |  |
| --- | --- |
| 7 | Descripción detallada de la propuesta.   * Descripción de bloques / módulos. * Descripción de usuarios. * Descripción las funciones y características que tendrá. |

El proyecto consiste en una plataforma accesible vía web y aplicación móvil, diseñada para mejorar la preparación de los participantes en la olimpiada de matemáticas. Funcionara como un repositorio masivo y organizado de problemas, incluyendo herramientas como un filtrado, un buscador, un sistema social, o un módulo de creación y calificación de exámenes.

Principalmente se tiene un ecosistema con los datos de la comunidad, donde cada estudiante puede filtrar problemas con precisión, discutir soluciones, guardar problemas en listas y poder realizar exámenes simulados en la misma plataforma. La versión móvil tiene además la capacidad de guardar ciertos problemas de manera offline.

**Login**

Existen cuentas de usuario, las cuales pueden ser creadas libremente por quien use la página, esta tiene relacionadas un username, un correo electrónico y una contraseña, el username es único para cada usuario. Al momento de iniciar sesión solo se ocupa el username y la contraseña.

**Repositorio de Contenidos**

En este modulo se encuentra la información (en una base de datos) de los problemas, estos serán obtenidos de distintos lugares, por ejemplo, concursos que hayan sucedido ya sea la IMO (International Mathematical Olympiad), la OMM (Olimpiada Mexicana de Matemáticas) o cualquier otro concurso, hay una gran cantidad de estos problemas recopilados en ingles en artofproblemsolving.com/community/c13\_contests, también puede haber problemas sin fuente que los usuarios pueden subir.

Cada problema cuenta con

* Enunciado
* Concurso de donde viene, si es que hay.
* Área, ya sea Algebra, Combinatoria, Geometría o Teoría de Números.
* Tipo de Repuesta, ya sea opción múltiple, de pura respuesta (un número o una cadena de caracteres), o de justificación, donde no solo importa a que respuesta llegaste si no como lo hiciste.
* En caso de que sea de opción múltiple o de pura respuesta se tendrá que ingresar la respuesta correcta.
* Tags, los tags significan distintos temas o técnicas que son importantes para la olimpiada de matemáticas, un problema se le asigna cierto tag si involucra ese tema, ejemplos de tags podrían ser “Principio Extremo”, “Cuadriláteros Cíclicos”, “Algoritmos”, etc.

Cada usuario puede marcar un problema como resuelto, esto servirá para que el usuario tenga un registro de cuales problemas ya hizo y no repetir problemas, además sirve para poder detectar el nivel del usuario en algunos temas en específico, o simplemente para que al momento de hacer algún examen de practica no le aparezca un problema que ya ha resuelto. Habrá un botón para marcar un problema como completado, pero para que este problema si quede marcado como completado se tendrá que cumplir, dependiendo del problema, lo siguiente: si el problema es de opción múltiple o de pura respuesta tendrá que ingresar su respuesta, y si es la respuesta correcta entonces se marca como completado, si el problema es de justificación entonces tendrás que poner una descripción de cual fue tu proceso para hacerlo, la cual va a quedar de forma publica para todos los usuarios como un comentario, en el cual te pueden dar feedback.

Al momento de marcar un problema como completado podrán añadir o agregar tags a cada problema o poder votar por la dificultad (del 0 al 9) y calidad de un problema (del 1 al 5), lo cual se promedia para poder mostrar esa información sobre cada problema.

Cuales tags existen solo podrá ser cambiado por el administrador, pero al momento de querer agregar un tag puede hacerse una solicitud de algún tag en específico, esto también va a funcionar para los concursos existentes.

La dificultad indica que tanta experiencia o tiempo te debería tomar en promedio ese problema. Al momento de tratar de votar la dificultad de un problema habrá un link que te llevara a una explicación de la escala, la cual va desde un problema que podría resolver casi cualquier persona, hasta un problema que ningún estudiante de preparatoria del mundo podría resolverlo. Esta es muy subjetiva y depende mucho de que tanto has entrenado ciertos temas, por eso se va a promediar entre las opiniones de las personas que marquen como resuelto este problema.

La calidad es que tanto te gusto un problema, un 1 seria que no te gusto para nada intentar el problema, un 5 es que es un problema muy bonito que recomendarías a los demás.

Además de los problemas hay una sección de teoría, donde cada usuario puede subir posts con artículos con distinta información, como artículos, teoremas o técnicas. Estos posts de teoría pueden citar problemas que se encuentren en la plataforma.

**Posts**

En la pagina se pueden hacer posts, en cada post se puede dar likes, en el texto de los posts el usuario puede ingresar un texto el cual puede contener latex, que será mostrado con ayuda de latex.js, también se puede agregar un archivo .tex en vez del texto, y pueden agregarse a lo más 3 archivos aparte (que van a estar adjuntos no van a ser parte del texto del post), cada post tiene un id alfanumérico el cual es mostrado en la parte superior derecha del post.

En cada post pueden hacerse comentarios, y en cada comentario pueden hacerse comentarios. Los comentarios van a tener el mismo formato que un post normal, solo que van a aparecer como respuesta a el comentario que responde.

Los problemas están guardados como posts, pero los posts también podrían ser teoría, o en general cualquier cosa que un usuario quiera publicar. Al momento de crear un post se pide especificar si es un problema.

Los posts tendrán una privacidad, ya sea publica o que sea de un grupo en particular (los grupos están explicados después), los posts públicos serán mostrados en una sección de Comunidad, los cuales estarán del más reciente al más viejo.

Los problemas pueden ser filtrados mediante la calidad, dificultad y tags que tienen asignados, donde ser filtrado puede ser que cumpla ciertas características (las cuales puede ser que cumple todas o que cumpla alguna) o también pueden ser usados para ordenamiento.

**Buscador**

Se va a realizar un buscador por semántica entre los posts, para poder encontrar posts parecido pero que no se tengan que ver textualmente igual, para esto vamos a usar un modelo ya entrenado mediante la librería sentence\_transformers de Python, que lo que hace es convertir el texto de los posts en embeddings (vectores de números) y mide que tan parecidos son mediante similitud coseno. Para no tener que convertir cada problema en embedding cada que se quiere hacer una búsqueda se va a guardar el embedding para cada problema en una base de datos y se va a actualizar cada que se actualice el texto del post.

**Personalización**

Va a haber un perfil de usuario, el cual va a mostrar una descripción dada por el usuario, los problemas resueltos y los posts hechos (incluyendo problemas publicados) así como sus listas públicas y grupos a los que pertenece, además va a haber una opción para recomendación de problemas y que temas entrenar, en las cuales se va a mostrar su nivel de cada área, el cuál se explica más adelante como se obtiene. También se cuenta con una parte de configuración donde puede cambiar su descripción, así como si se quiere usar el programa de recomendaciones, entonces incluir para que concursos se está preparando y que al recomendarle problemas cuantos se le recomienden.

Al momento de hacer recomendaciones cada problema se marcará como resuelto, o si no se resuelve en cierto tiempo que cada usuario toma manualmente, pero se especifica que de preferencia sea una hora y media como no resuelto.

Para la recomendación de problemas, se tendrá lo siguiente: Para cada relación de área-dificultad, el porcentaje de problemas que noresuelve de manera correcta, pero para esto solo van a contar los problemas que se den en recomendaciones pasadas, si es un usuario que usa recomendaciones por primera vez ese porcentaje va a ser 0 por defecto. Y además se tendrá que porcentaje de problemas del concurso al que se está preparando el usuario son de área con cierta dificultad o más, entonces la importancia de cada par de (área, dificultad) va a ser 0.7 \* Porcentaje de problemas no resueltos + 0.3\* Porcentaje de problemas que son de esa área que son al menos de esa dificultad. La importancia de un área es la importancia de la primera vez que sale.

Entonces la lista de problemas que se va a sugerir va a ser 60% del área con más importancia, 30% de la segunda y 10% de la tercera.

Ahora para escoger los problemas de cada área se va a repetir el mismo proceso que al escoger las áreas, pero con los tags dando parejas de (tags, dificultad) para cada área, que se reparten de igual forma 60%,30%,10%. Entonces se van a tomar los problemas de ese tag con esa dificultad por orden de calidad, tomando en cuenta que no se pueden escoger problemas que el usuario tiene marcados como resueltos. De igual forma cada tag solo se considera la primera vez que aparece.

Se redondean los porcentajes de forma que si queden una cantidad entera de problemas.

**Listas**

Las listas son conjuntos de posts con un orden en específico, cada usuario puede crear sus propias listas, puede tener muchas solo que cada nombre puede aparecer solo una vez como lista de un usuario, usuarios distintos pueden tener listas con el mismo nombre, un usuario puede compartir listas con distintos usuarios para que le agreguen contenido.

Habrá una sección de listas en las cuales aparecen las listas que a creado o de la que es parte el usuario, además de un botón para crear una nueva lista.

En la interfaz de hacer la lista va a haber tres paneles, el de la izquierda en el cual se puede usar ya sea el filtro (el cual muestra puros problemas), o el buscador por semántica, también se puede agregar directamente con el id alfanumérico del post, se van a mostrar como bloques los posts que cumplan las condiciones ingresadas en cualquiera de esas tres formas, el panel del medio un lienzo en blanco en donde tienes que arrastrar los bloques y pueden ser reacomodados, esto mediante la librería Drag and Drop, para definir el orden, y en el panel de la derecha una vista de cómo se ve la lista.

**Grupos**

Habrá una sección de grupos, en la cuales se van a mostrar los grupos a los cuales pertenece el usuario, las invitaciones que tenga pendientes de distintos grupos a los que haya sido invitado, una opción de buscar algún grupo público y una opción de crear un grupo.

Cada grupo tiene nombre, logo, privacidad ya sea público, por medio de link o solo por invitación de un administrador, además se pueden asignar administradores, el usuario que cree el grupo es un administrador por defecto, los otros administradores no pueden sacar del grupo al administrador por defecto, además tiene una lista de participantes y una descripción, estos detalles pueden ser cambiados desde una sección de configuración de grupo.

En cada grupo los usuarios pueden compartir listas, problemas, posts u organizar algún concurso con las personas de ese grupo.

**Concursos**

Va a haber otra sección de concursos, en las cuales salen concursos que están por venir, concursos en los que a participado u organizado el usuario, o la opción de crear o organizar algún concurso. En la parte de concursos que organiza el usuario también se muestran los concursos donde tiene el rol de calificador.

En este módulo se pueden crear los exámenes de práctica, para cada examen se decide un formato dependiendo de cuantos problemas de solo respuesta y cuantos problemas de redacción se quieren, estos problemas pueden ser asignados manualmente, o se pueden generar aleatoriamente con los problemas de la plataforma, para esto pueden agregarse distintos filtros como las áreas, tags, o dificultad de cada problema.

Cada examen es configurado como un examen de práctica casual, el cual no tiene ninguna revisión y es usado para solo practicar con los problemas, o como un examen calificado, el cual es un poco más formal y pasa después por un proceso de calificación en su respectivo modulo. Tu puedes decidir quien aplica el examen, si tú mismo, otro usuario, un grupo, o público.

También se pueden hacer un proceso de varios exámenes, donde cada examen tenga una ponderación especifica.

Particularmente, puedes agregar problemas que se mantengan privados para que sean usados en el examen y después se hacen públicos.

**Calificación**

Cuando organizas un examen puedes asignar ciertos usuarios como calificadores de ciertos problemas de justificación y de solo respuesta, la parte de solo respuesta es para darle una checada que alguna respuesta este bien solo que este en un formato distinto al esperado, los problemas de opción múltiple se califican automáticamente, estos usuarios tienen que aceptar, un usuario puede calificar muchos problemas, y un problema puede tener muchos calificadores (pero un usuario solo puede calificar exactamente un problema). También se define para cada problema un puntaje máximo sobre el cual calificar, lo usual es 7 puntos. Los puntajes de un problema son enteros, pero la ponderación de cierto examen no necesariamente es entera.

Habrá un chat general para los organizadores y calificadores del concurso, además habrá un chat para cada problema en el cual estarán los calificadores específicos de ese problema y los organizadores. El chat general este hecho para cualquier tema que pueda salir al organizar el concurso, el chat por problema para particularmente dos cosas, discutir criterios de calificación, y discutir discrepancias al momento de calificar.

Cada persona que va a calificar puede ver las redacciones del problema que va a calificar, entonces asigna una calificación en base al criterio decidido. Si califican varias personas y hay una discrepancia, donde discrepancia significa que no sea exactamente el mismo puntaje, en los puntajes se manda un mensaje automático al chat de ese problema, para incitar una discusión sobre ese puntaje y llegar a una decisión final.

Después o durante la calificación, el organizador del examen decide qué información mostrar, ya sea si solo la de cada usuario, la de todos, o la de nadie.

**Aplicación Móvil**

Refleja las funcionalidades de la pagina web, pero con la capacidad de guardar problemas o listas de manera offline.

|  |  |
| --- | --- |
| 8 | Lista de aplicaciones similares |

Art of Problem Solving artofproblemsolving.com/community/c13\_contests

|  |  |
| --- | --- |
| 9 | Argumentos de cumplimiento de: Viabilidad, Aplicabilidad, Accesibilidad, Usabilidad |

Viabilidad: El enfoque se centra en una base de datos bien diseñada, y una buena interfaz de usuario.

Aplicabilidad: Se cumple una necesidad, que es la capacidad de filtrar problemas de manera muy específica, y de generar un lugar donde puedes tomar control de tu proceso olímpico.

Accesibilidad: Al además contar con posible guardado de problemas offline, hace que se pueda usar desde distintos lugares.

Usabilidad: Tendrá una interfaz de usuario limpia e intuitiva.

|  |  |
| --- | --- |
| 10 | ¿Qué tecnologías se utilizarán y para qué? |

Se usará Oracle Cloud Infrastructure (OCI) como proveedor de nube, ya que el su nivel gratuito, Always Free Tier, permite desplegar el proyecto a un costo prácticamente nulo.

Para el backend se usará Python con Django y estará en contenedores Docker. Esto porque Django tiene un montón de funcionalidades ya integradas, y Python porque es un lenguaje versátil, además de tener librerías para el Embedding, en contenedores Docker para tener una buena escalabilidad, se ejecutarán en Oracle Kubernetes Engine (OKE).

La base de datos será PostgreSQL ya que permite manejar datos complejos y es eficiente, se ejecuta con el servicio OCI Database with PostgreSQL.

Para el frontend se usará React para el sitio web y React Native para la aplicación móvil, ya que permiten funcionar de manera similar y crear interfaces de usuario rápidas e interactivas. Además, se usará Latex.js para el renderizado de latex de forma clara en ambos lugares.

Para el buscador vamos a usar Python con la librería sentence\_transformers la cual incluye un modelo ya entrenado para poder convertir los problemas a vectores.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EVALUADOR** | | | | |
| C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.pngC:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.png**Docente Titular de la UAC Otro Docente** | | **NOMBRE:** | C:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.pngC:\Users\DEPART~1\AppData\Local\Temp\msohtmlclip1\01\clip_image005.png**Propio Estudiante Otro Estudiante** | **REGISTRO:** |
| **OBSERVACIONES:** |  | | | |