

Programa de Aprendizaje Digital

HACKATHON 2023

Carlos Palomeque de la Torre
Francisco Gómez Rosillo
Mauro Agustín Cervantes García
Josue Hernandez Chavez

INTRODUCCIÓN

“El conocimiento es poder” Frase acuñada por Sir Francis Bacon, donde la interpretamos cómo una afirmación de que en la educación y la acumulación de saberes, se halla la verdadera posibilidad del ser humano de ser influyente, de cambiar, de crecer e incluso de tener éxito profesional. Con esto queremos llegar al punto de la deserción académica, un pueblo ignorante es un pueblo perdido, y nosotros queremos revertir esto creando una página que sea accesible para todos, que no necesite tener acceso a internet y donde puedas aprender a tu ritmo.

En Querétaro

En 2020, en Querétaro el **grado promedio de escolaridad de la población** de 15 años y más de edad es de **10.5**, lo que equivale a poco más de primer año de bachillerato.



A nivel nacional, la población de 15 años y más tiene **9.7 grados de escolaridad** en promedio, lo que significa un poco más de la secundaria concluida.

De cada 100 personas de 15 años y más...

- 4** no tienen ningún grado de escolaridad.
- 45** tienen la educación básica terminada.
- 23** finalizaron la educación media superior.
- 28** concluyeron la educación superior.

Fuente: INEGI. Censo de Población y Vivienda 2020.

Programa de Aprendizaje Digital

El programa de Aprendizaje Digital tiene como objetivo proporcionar a los residentes de Querétaro y San Juan del Río las habilidades y conocimientos necesarios para participar de manera efectiva en la sociedad. El aprendizaje digital se refiere a la capacidad de utilizar herramientas tecnológicas y recursos en línea de manera competente y segura. A medida que la tecnología se vuelve más presente en la vida cotidiana, es esencial que las personas tengan las habilidades necesarias para aprovechar al máximo sus beneficios.

Cursos en línea

Recursos interactivos como tutoriales en video, guías en línea y simulaciones que permitan a los participantes practicar las habilidades digitales en un entorno seguro. Estos recursos estarán disponibles en nuestra plataforma.

Módulos Específicos

Secundaria y Preparatoria con las materias básicas como Español, Matemáticas, Ciencias Naturales e Inglés.

Evaluaciones y Certificaciones

Implementar evaluaciones periódicas para medir el progreso de los participantes y otorgar certificados de finalización a aquellos que completen con éxito los cursos. Esto para ayudar a motivar a los participantes, reconocer sus logros y que todo su esfuerzo tenga validez oficial y puedan utilizar ese certificado ya sea para obtener un mejor empleo o continuar con sus estudios en alguna institución de estudios superiores.

Las certificaciones buscamos sean oficiales y respaldadas por alguna institución federal (RVOE).

8. ¿Qué instituciones pueden otorgar un RVOE del tipo superior?

Las instancias facultadas para otorgar el RVOE son las siguientes:

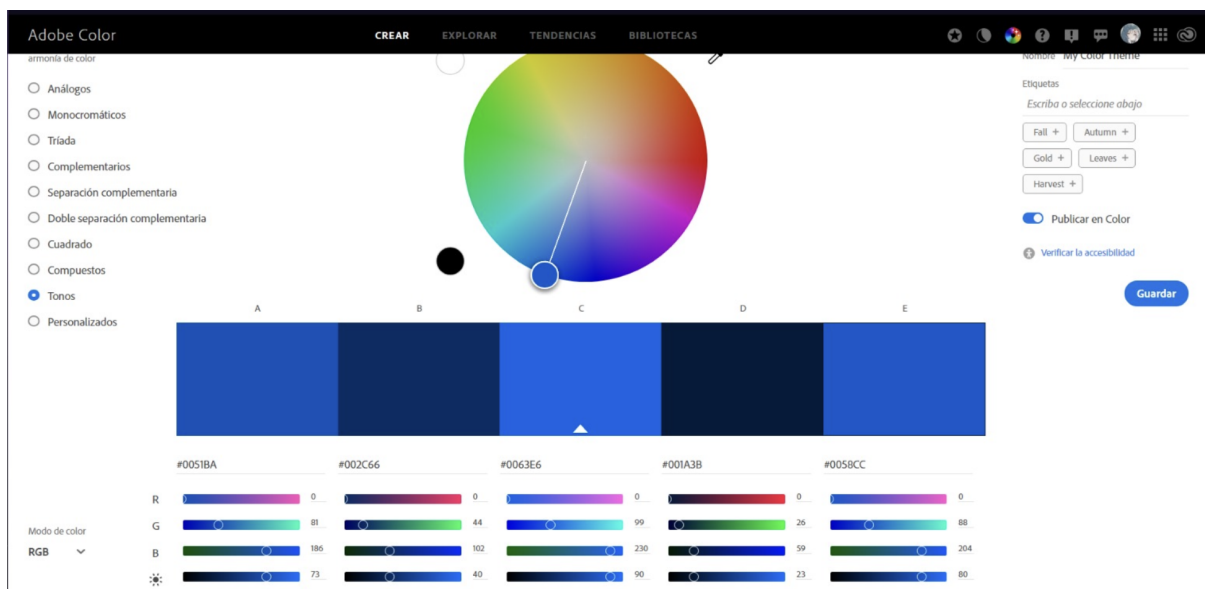
- Universidad Autónoma De Baja California Sur (UABCS)
- Universidad Autónoma De Ciudad Juárez (UACJ)
- Universidad Autónoma De Coahuila (UACOH)
- Universidad Autónoma De Chihuahua (UACH)
- Universidad De Guadalajara (UDG)
- Universidad Autónoma Del Estado De Hidalgo (UAEH)
- Universidad Autónoma Del Estado De México (UAEMEX)
- Universidad Autónoma Del Estado De Morelos (UAEMOR)
- Universidad Autónoma De Nuevo León (UANL)
- Universidad Autónoma De Querétaro (UAQRO)
- Universidad De Quintana Roo (UQROO)
- Universidad Autónoma De Yucatán (UADY)
- Instituto Politécnico Nacional (IPN)
- Universidad Nacional Autónoma De México (UNAM)

DISEÑO

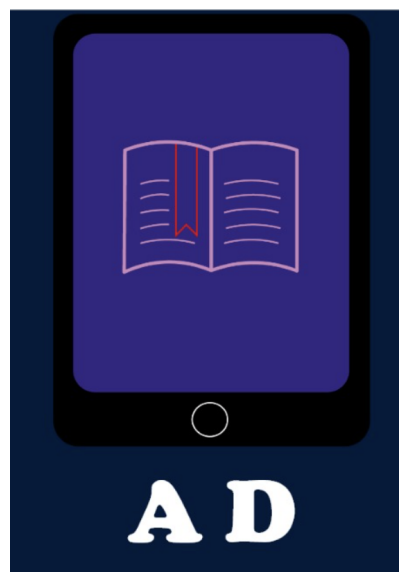
Programa de Aprendizaje Digital

“AD”

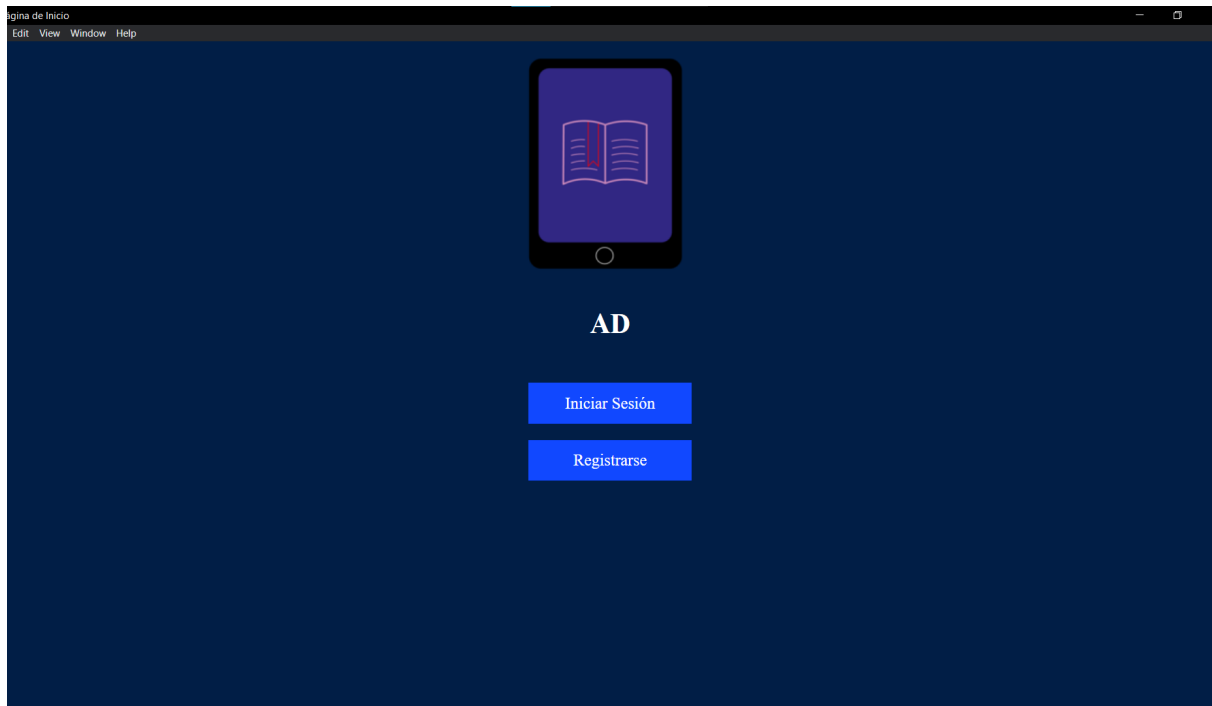
Para la base de nuestra plataforma utilizaremos la siguiente paleta de colores:



Nuestro Logo



Inicio de sesión/registro



Registro

The screenshot shows a web browser window titled 'registro'. The browser's menu bar includes 'Edit', 'View', 'Window', and 'Help'. The main content area has a dark blue background. In the top left corner, there is a small white icon of a tablet displaying an open book. In the top right corner, the text 'Registro' is displayed in white. Below this, there is a white registration form with the following fields and labels: 'Nombre:', 'Apellidos:', 'CURP:', 'Nivel de Estudio:' (with a dropdown menu showing 'Primaria'), 'Correo o Teléfono:', and 'Contraseña:'. At the bottom of the form is a blue button with white text labeled 'Registrarse'.

Inicio de sesión

Inicio Sesión

Iniciar Sesión

CURP:

Contraseña:

Iniciar Sesión



Grados

Usuarios

Validar cursos

Porcentajes

Primaria: 00%

Secundaria: 00%

Preparatoria: 00%

#69F57F

#FC7C0D

Grados

Primaria

Registrarse

Secundaria

#F21A0C

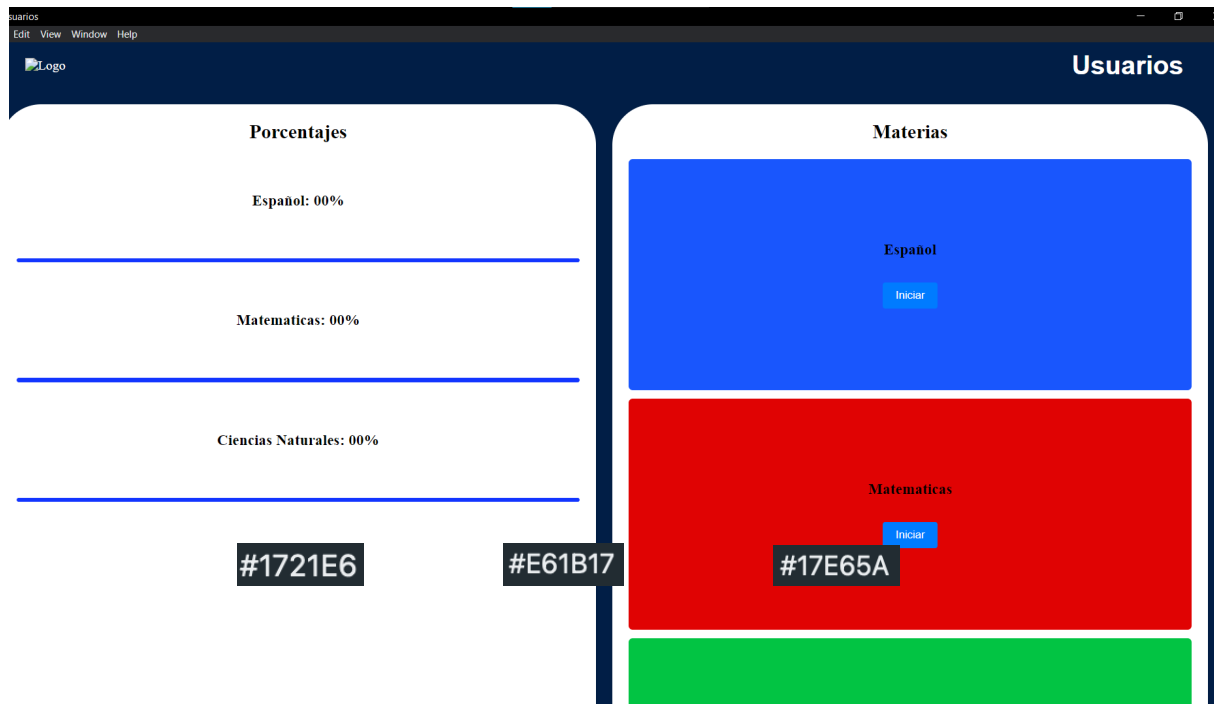
Registrarse



Perfil de usuario



Materias de cada módulo



Validación de progreso y certificación

Formulario de Preguntas

1. Tarjetas o papeletas que sirven para organizar los datos encontrados e información más importante de materiales consultados en una investigación.

- ☐ a) Lluvia de ideas
- ☐ b) Fichas de trabajo
- ☐ c) Resumen
- ☐ d) Cita textual

2. Reducción de un texto a sus partes esenciales, en el cual debe respetarse el orden de las ideas principales y palabras del autor.

- ☐ a) Lluvia de ideas
- ☐ b) Fichas de trabajo
- ☐ c) Resumen
- ☐ d) Cita textual

3. Son las que recuperan un fragmento de un texto exactamente con las mismas palabras que usó su autor, van entre comillas.

- ☐ a) Lluvia de ideas
- ☐ b) Fichas de trabajo
- ☐ c) Resumen
- ☐ d) Cita textual

4. Documentos que especifican normas para regular las actividades de los miembros de una comunidad, cimientan las bases para la convivencia.

- ☐ a) Reglamento
- ☐ b) Leyes y derechos
- ☐ c) Declaración de los Derechos del niño y adolescente
- ☐ d) Constitución

Sección de evaluación

Inicio

localhost:3000/solicitud/64e0e974ddd2e164ca606e3b

¿Cómo actualizar / ... Página principal de... Kick Ass iLovePDF | Herrami... Ducky Script - el le... AlternativeTo - Cro... Visión General del... Función async - Jav... Editar PDF - Editor... JavaScript | MDN

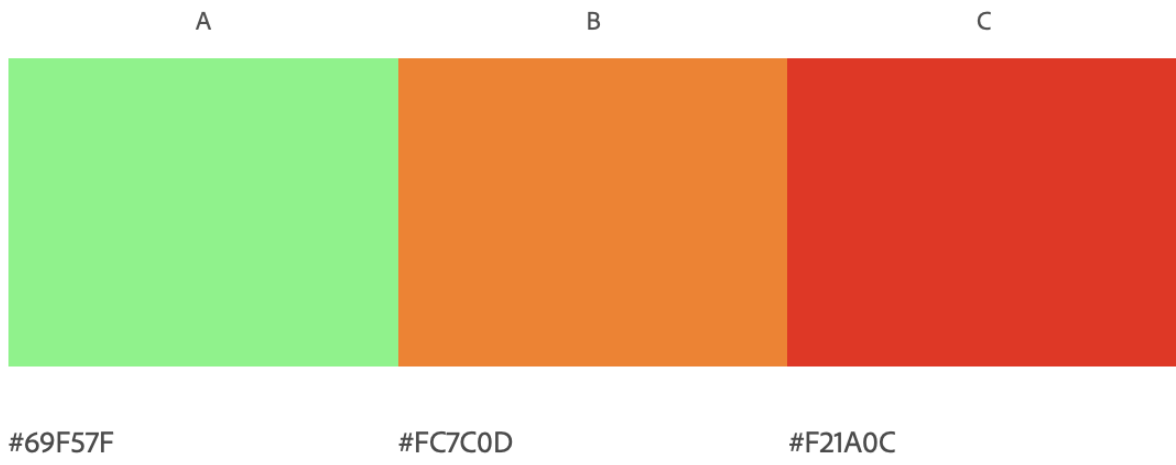
Aprendizaje Digital Cerrar sesión

Información del usuario y validación

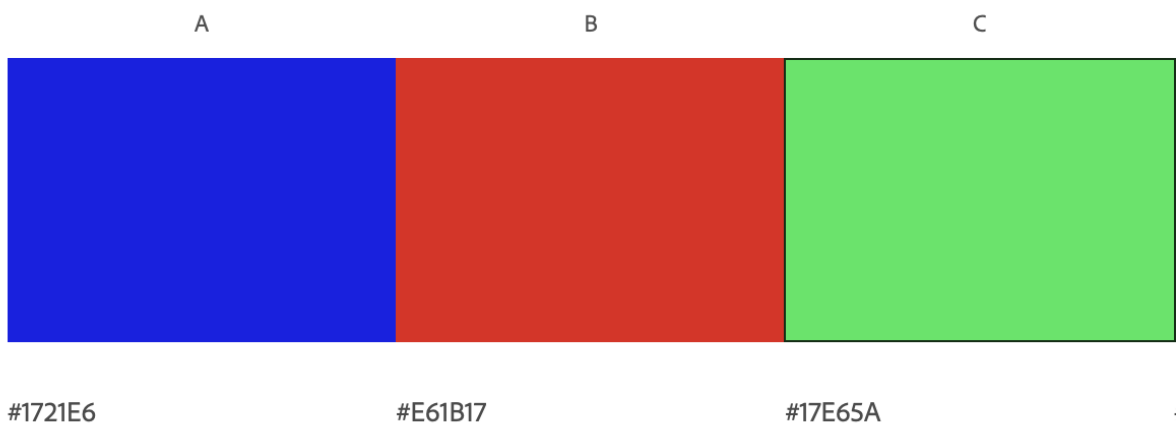
Validar curso Rechazar validación

Nombre:	Fecha de solicitud:
<input type="text" value="nombre apellidos"/>	<input type="text" value="Sat Aug 19 2023 10:06:12 GMT-0600 (hora estándar central)"/>
CURP:	Promedio:
<input type="text" value="numerosaleatorios"/>	<input type="text"/>

Paleta de colores para "Grados", cada grado tiene su código de color en la base de la imagen.



Paleta de colores para las "Materias", cada materia tiene su código de color debajo de la imagen.



FUNCIONAMIENTO Y CODIFICACIÓN

La plataforma Aprendizaje digital está constituida por dos partes, una aplicación de escritorio multiplataforma y un servicio web. En conjunto crean un entorno de educación íntegro.

Aplicación de escritorio:

La aplicación de escritorio fue elaborada con NodeJs gracias al framework ElectronJs. La arquitectura monolítica de dicha aplicación nos permite tener todo el contenido académico precargado para que no se requiera acceso a internet. La construcción de la aplicación permite que cualquier curso pueda ser cargado mediante ficheros .json, ampliando mucho más su rango y niveles de estudio.

Estructura de ficheros .json.

La estructura de los ficheros es específica y se requiere de su construcción exacta, pues de esta depende la carga correcta del curso. "tema actual" indica cual es el tema más reciente que el usuario ha consultado, permitiendo así un rastreo de las actividades del usuario. Las etiquetas "secundaria" permiten separar los niveles académicos para su correcta renderización. Cada propiedad deberá tener dentro "progreso" para mostrarlo junto al nivel académico. También tendrán un "url" para obtener todos los recursos necesarios para mostrar un cuestionario o un tema.

Renderizado de elementos.

El programa se encarga de leer mediante el fileSystem el archivo .json correspondiente al curso e indicado por su propiedad "url". Lo preferible sería que la aplicación utilice ese proceso para cargar sus elementos y cursos, volviendo así

más fácil la carga de cualquier curso o tema. Todos los arreglos del archivo `registroTemario.json` serían recorridos por su método `.forEach()` obteniendo así cada elemento para mostrarlo en pantalla.

Renderizado y manejo de pantallas

ElectronJs nos permite crear una ventana de navegador (`BrowserWindow`), y mediante JavaScript (NodeJs) podemos cambiar la vista o archivo html que se desea que la ventana cargue. Esto permite crear una navegación muy parecida a la de un navegador ordinario. Utilizando el módulo `ipcMain` y `ipcRenderer` podemos comunicar el JavaScript de NodeJs y el JavaScript de la ventana de ElectronJs. El objeto `window` es el controlador de la ventana y su método `.loadFile()` permite indicar la url del archivo a renderizar en dicha ventana. Dentro de la carpeta de recursos se tiene la carpeta `app`, la cual contiene los archivos `.js` que hacen el cambio de url. En los mismos hay una función con el nombre de la página a mostrar, la cual recibe como parámetro el objeto de la ventana, y el cual puede pasar a otra función para la modificación correspondiente.

Registro de avance.

Cada que un usuario termina un curso el avance deberá ser registrado en `registroTemario.json`, para así tener control de los avances. Este registro de avance puede ser utilizado para la solicitud de validación del curso.

Solicitudes http.

Las solicitudes http al servicio se realizan utilizando la función `fetch` nativa de JavaScript mediante una función intermedia para agilizar el proceso de petición. Esta función recibe un url a la que se realizará la petición y el cuerpo de la petición. Respecto a las cabeceras de dichas solicitudes, algunas requieren de la configuración de la cabecera "Authorization" con el token obtenido por la solicitud de acceso, sin embargo todas requieren de la configuración de "Content-Type" como "x-www-form-urlencoded".

Registro y acceso.

La aplicación y todo su contenido está disponible y no requiere registro alguno. El registro y acceso están disponibles en el caso de que el usuario quiera solicitar la validación de sus cursos. En este caso se requiere de realizar un registro de información relevante: CURP, nombre completo, máximo nivel de estudios, un método de contacto, y una contraseña. Para el acceso solo es requerido CURP y password. Estas dos consultas no requieren de la configuración de cabecera "Authorization". Como respuesta de la solicitud de acceso se recibe un token web tipo json (jwt) el cual se configura como "tokenAuth" en localStorage.

Servicio Web:

El servicio web esta tambien elaborado con NodeJs utilizando el framework de ExpressJs, Respecto a módulos los requeridos fueron mongoose (para establecer la conexión a la base de datos), express-session (para controlar las sesiones de administradores), body-parser (controlar el cuerpo de las peticiones por API), cors (permitir peticiones de origen cruzado), jsonwebtoken (generación y autenticación de tokens de usuario), crypto-js (encriptar contraseñas para almacenamiento en base de datos) y ejs (motor de vistas). Estructuralmente el servicio se compone de dos partes, las rutas de API y las rutas de administrador (server-site-rendering).

Estructuración app.js

El archivo app.js dentro de la carpeta principal es el encargado de correr el servicio, En este archivo se mandan llamar todas las dependencias y los modelos de base de datos. La constante "app" es la que controla el servicio y mediante su método .use() se agregan las dependencias y se configuran las rutas API. Después de la carga de dependencias y la carga de las rutas de administración se configura el motor de

visitas con `.set()`, y se pone a correr el servicio con el método `.listen()`.

Rutas API

Dentro de la carpeta `router` se encuentra el archivo `api.js` en el que se crean todas las rutas propias de la API se llaman también algunas dependencias y los modelos de base de datos. Se declara al inicio una función `"sessionVerify()"` tipo `middleware` para la validación de token. Posteriormente se configuran las rutas mediante el módulo `Router` de `express`. Finalmente se exporta el objeto `"ruta"`.

Modelos de base de datos:

La base de datos utilizada fue `MongoDB`, una base de datos `NoSQL` orientada a documentos. En la carpeta `"models"` dentro de la carpeta principal se encuentran los archivos `.js` donde se llaman los módulos `"model"` y `"Schema"` para la creación de nuevos modelos y se exportan para su uso dentro del resto de archivos. El archivo `"solicitudes.js"` contiene la definición del esquema solicitando los siguientes datos: `usuarioId`, `estado`, `revisado`, `fechaSolicitud`, `fechaRevision` y `reporte`. En `"usuarios.js"` se solicitan los siguientes datos: `curp`, `nombre`, `apellidos`, `nivelEstudios`, `contacto`, `password`.

Bibliografía

de Educación Pública, S. (s/f). *Reconocimiento de Validez Oficial de Estudios (RVOE)*. gob.mx. Recuperado el 19 de agosto de 2023, de <https://www.gob.mx/sep/acciones-y-programas/reconocimiento-de-validez-oficial-de-estudios-rvoe>

Educación. Querétaro. (s/f). Org.mx. Recuperado el 19 de agosto de 2023, de <https://cuentame.inegi.org.mx/monografias/informacion/queret/poblacion/educacion.aspx?tema=me&e=22>

Querétaro. (s/f). Data México. Recuperado el 19 de agosto de 2023, de <https://www.economia.gob.mx/datamexico/es/profile/geo/queretaro-992201?redirect=true>

(S/f). Gob.mx. Recuperado el 19 de agosto de 2023, de https://planeacion.sep.gob.mx/Doc/Atlas_estados/queretaro.pdf