

VIS-U

VERSIÓN 1.0

MANUAL TÉCNICO

Instituto Tecnológico de Costa Rica

Curso Computación y Sociedad

Primer Semestre, 2017 – Grupo 40 – Prof. Adriana Álvarez

Manual elaborado por:

Esteban Fonseca Montoya

Víctor Aguirre Brenes

TABLA DE CONTENIDOS

Objetivo General del Manual Técnico.....	3
Objetivos Específicos del Manual Técnico	3
Descripción de la aplicación	3
Requisitos para instalar la aplicación	3
Instalando la Aplicación	4
Ejecutando la aplicación	4
Requerimientos del Entorno de Desarrollo	5
Descripción de los Actores del Sistema.....	5
Implementación de la Aplicación Android	6
Dependencias del Proyecto	6
Listado de Paquetes del Código Programado (Java)	9
Listado de Clases por PAquete	11
Implementación de la Base de Datos.....	13
Diccionario de Tablas de la Base de datos	13
Modelo lógico de la Base de datos.....	14
Índices de la Base de Datos.....	15
Claves ForÁneas de la Base de Datos.....	15
Implementación de Servicios Web.....	16
Créditos	17

OBJETIVO GENERAL DEL MANUAL TÉCNICO

Ofrecer la información técnica necesaria para la instalación y el mantenimiento de la aplicación móvil y el servidor web utilizado para la adquisición de los datos utilizada por la misma.

OBJETIVOS ESPECÍFICOS DEL MANUAL TÉCNICO

- Determinar la estructura técnica y su diseño.
- Especificar los requerimientos de hardware y software.
- Detallar las tecnologías utilizadas para el desarrollo de la aplicación.

DESCRIPCIÓN DE LA APLICACIÓN

Vis-U es una aplicación móvil para dispositivos con sistema operativo Android que funciona como una plataforma para promocionar las carreras del Instituto Tecnológico de Costa Rica.

Actualmente la tecnología tiene un gran impacto en los medios de comunicación, un gran porcentaje de adolescentes tiene acceso a ella. Vis-U busca proporcionar una alternativa al material impreso utilizado por el Instituto Tecnológico de Costa Rica para proporcionar información a los visitantes y futuros estudiantes. De manera interactiva y lúdica se busca informar varios aspectos importantes de las carreras, tales como: descripción de la carrera, malla curricular, habilidades, campo laboral, opiniones de la carrera, material multimedia promocional (como imágenes y videos), entre otros.

Para la primera versión de la aplicación sólo se van a promocionar las carreras de Ingeniería en Mantenimiento Industrial, Ingeniería en Computación e Ingeniería Mecatrónica. No obstante, debido a que la información que la misma utiliza es obtenida a través de servicios web (web services), la incorporación de nuevas carreras y la actualización de las carreras actualmente existentes dentro de la aplicación será posible sin que sea necesario realizar modificaciones importantes dentro de la aplicación.

Por el momento el contenido de la aplicación está disponible en español.

REQUISITOS PARA INSTALAR LA APLICACIÓN

Para instalar esta aplicación se requiere un dispositivo móvil que cumpla las siguientes características:

- Teléfono inteligente (smartphone) con sistema operativo Android, versión 4.1 (Jelly Bean) o más reciente.
- El dispositivo debe tener al menos 512MB de memoria RAM y al menos 30MB de espacio de almacenamiento en la memoria interna.
- Se requiere de una conexión a internet para acceder a la información de las carreras.

INSTALANDO LA APLICACIÓN

Para instalar la aplicación en el dispositivo, deberá obtener el archivo APK instalable a través de alguno de los servicios de distribución utilizados por los desarrolladores (como Google Play) u otros canales de distribución.

Esta sección será actualizada conforme se incorporen canales de distribución para la obtención de esta aplicación.

EJECUTANDO LA APLICACIÓN

Para ejecutar la aplicación, seleccione el ícono de la aplicación a través del menú de aplicaciones de su dispositivo Android. El título de la aplicación será Vis-U y el ícono de la aplicación es el siguiente:



Ilustración 1 - Ícono de Vis-U

Cuando se ejecute la aplicación aparecerá una pantalla de bienvenida (*splash screen*):



REQUERIMIENTOS DEL ENTORNO DE DESARROLLO

Para realizar labores de mantenimiento e implementación de la aplicación “Vis-U”, se deben tener las siguientes herramientas de software instaladas en los equipos a utilizar:

- Para la modificación del código fuente de la aplicación Android:
 - Alguno de los siguientes Entornos de Desarrollo Integral (IDE):
 - *Android Studio*, versión 2.0 en adelante.
 - *IntelliJ IDEA*, versión 2015 en adelante. Tanto la versión comunitaria (*Community Edition*) como la versión máxima (*Ultimate Edition*) son compatibles.
 - Kit de Desarrollo de Software (SDK) de Android:
 - Módulos de SDK de la versión 25 de la API de Android, correspondiente a la versión 7.1 (Nougat) de Android.
 - Opcionalmente, cualquier otro conjunto de módulos SDK de versiones anteriores de la API de Android (para las versiones anteriores de Android) que considere necesarias.
 - Herramienta de compilación Gradle, versión 2.2 en adelante. Estas herramientas se descargan automáticamente a través de los IDEs anteriores.
- Para la modificación de la base de datos utilizada por la aplicación:
 - Cualquier herramienta de gestión de bases de datos compatible con bases de datos *MySQL*, versión 5.4 en adelante: Por ejemplo, *MySQL Workbench* y *phpMyAdmin*.
 - Configuración de usuarios y contraseñas en el servidor web para su acceso a través de herramientas instaladas en el equipo. Consulte al administrador del dominio.
- Para la modificación de los servicios web:
 - Cualquier IDE compatible con *PHP*, versión 5.6.
 - *cPanel*.
 - Acceso al host remoto del servidor web. Consulte al administrador del dominio para obtener las credenciales necesarias.

DESCRIPCIÓN DE LOS ACTORES DEL SISTEMA

Nombre del Actor	Descripción
Usuario de la Aplicación	Utiliza la aplicación móvil para informarse acerca de las carreras universitarias a través de la aplicación.
Servidor Web Kivucr.com	Recibe las peticiones de datos de la aplicación y distribuye la información solicitada.
Administrador Servidor Web Kivucr.com	Agrega y modifica la información utilizada por la aplicación Vis-U a través del servidor web.

IMPLEMENTACIÓN DE LA APLICACIÓN ANDROID

A través de esta sección se brinda una explicación de las características del diseño del código fuente utilizado para la generación de la aplicación Android a través de un proyecto de aplicación utilizando alguno de los IDEs mencionados en la sección “Requerimientos de Entorno de Desarrollo”:

DEPENDENCIAS DEL PROYECTO

El archivo **build.gradle** del proyecto contiene un listado de las bibliotecas necesarias para compilar el proyecto y generar la aplicación de Android. Este listado se ubica en la sección “dependencies” del archivo:

Listado de bibliotecas de dependencias del proyecto

- JUnit:
junit:junit:xx.xx.xx
 - Versión 4.12 o más reciente.
 - Necesaria para crear unidades de prueba.
- Android MultiDex Support
com.android.support:multidex:xx.xx.xx
 - Versión 1.0.1 o más reciente.
 - Utilizada para la compilación.
- Android App Compatibility Support Library (AppCompat), v7
com.android.support:appcompat-v7:xx.xx.xx
 - Versión v7:25.3.1 o más reciente.
 - Permite la implementación compatible de funciones nativas de versiones más recientes de Android en versiones anteriores.
- RecyclerView Animation
jp.wasabeef:recyclerview-animators:xx.xx.xx
 - Versión 2.2.4 o más reciente.
 - Permite agregar animación a las vistas RecyclerView.
- Google GSON Library
com.google.code.gson:gson:xx.xx.xx
 - Versión 2.7 o más reciente.
 - Biblioteca con recursos para el manejo de objetos JSON.
 - Requerido para el procesamiento de los datos retornados por los servicios web.
- Glide
com.github.bumptech.glide:glide:xx.xx.xx
 - Versión 3.5.2 o más reciente.
 - Biblioteca con framework para el manejo de imágenes y demás contenido multimedia.
- OkHttp
com.squareup.okhttp3:okhttp:xx.xx.xx
 - Versión 3.4.1 o más reciente.

- Biblioteca con cliente para los protocolos HTTP y HTTP/2.
- Android CardView Support
com.android.support:cardview-v7:xx.xx.xx
 - Versión 25.3.1 o más reciente.
 - Biblioteca con soporte para la vista CardView de Android.
- Google Play Authorization Services for Android
com.google.android.gms:play-services-auth:xx.xx.xx
 - Versión 10.2.1 o más reciente.
 - Biblioteca para integrar los servicios de Google Play.
- Android Support Library
com.android.support:support-v4:xx.xx.xx
 - Versión 25.3.1 o más reciente.
 - Similar a la biblioteca AppCompat; se utiliza junto con las otras bibliotecas de compatibilidad.
- Android LayoutManager for RecyclerView to Support Carousel view style
com.azoft.carousellayoutmanager:carousel:xx.xx.xx
 - Versión 1.1.5 o más reciente.
 - Permite agregar animación de carrusel a un RecyclerView a través de LayoutManager.
- Google YouTube API
com.google.apis:Google-api-services-youtube:v3-xxxxxx-xx.xx.xx
 - Versión v3-rev178-1.22.0 o más reciente.
 - Permite integrar reproducción de videos de YouTube a la aplicación.

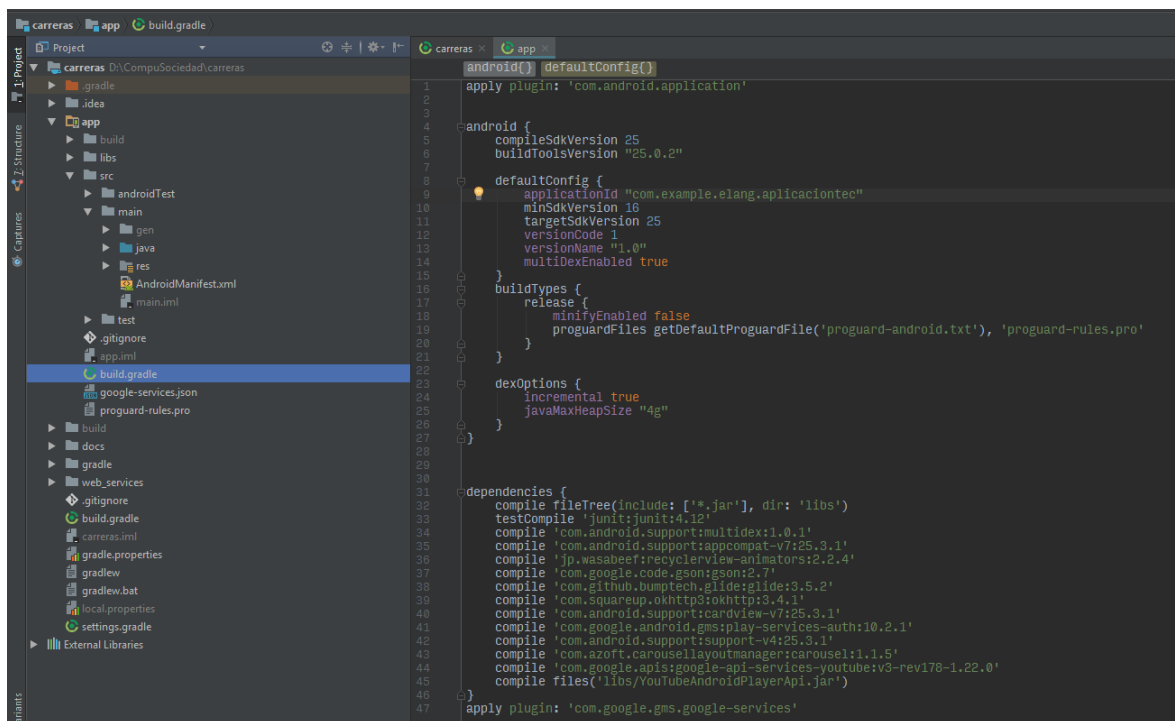
Se recomienda agregar las versiones más recientes disponibles de las dependencias para aprovechar cualquier nueva función incluida en estas versiones, excepto si ocurren problemas de compatibilidad.

No se requiere la instalación previa de estas bibliotecas. Serán descargadas automáticamente por el IDE durante la compilación, siempre y cuando se disponga de una conexión a internet.

Debido a que la aplicación hace uso de los servicios de Google Play y Firebase, se debe incluir el plugin correspondiente en el archivo build.gradle. Para incluir el plugin, sólo es necesario agregar la siguiente línea al final del archivo:

```
apply plugin: 'com.google.gms.google-services'
```

La siguiente captura de pantalla tiene un ejemplo del contenido que deberá tener el archivo **build.gradle**:



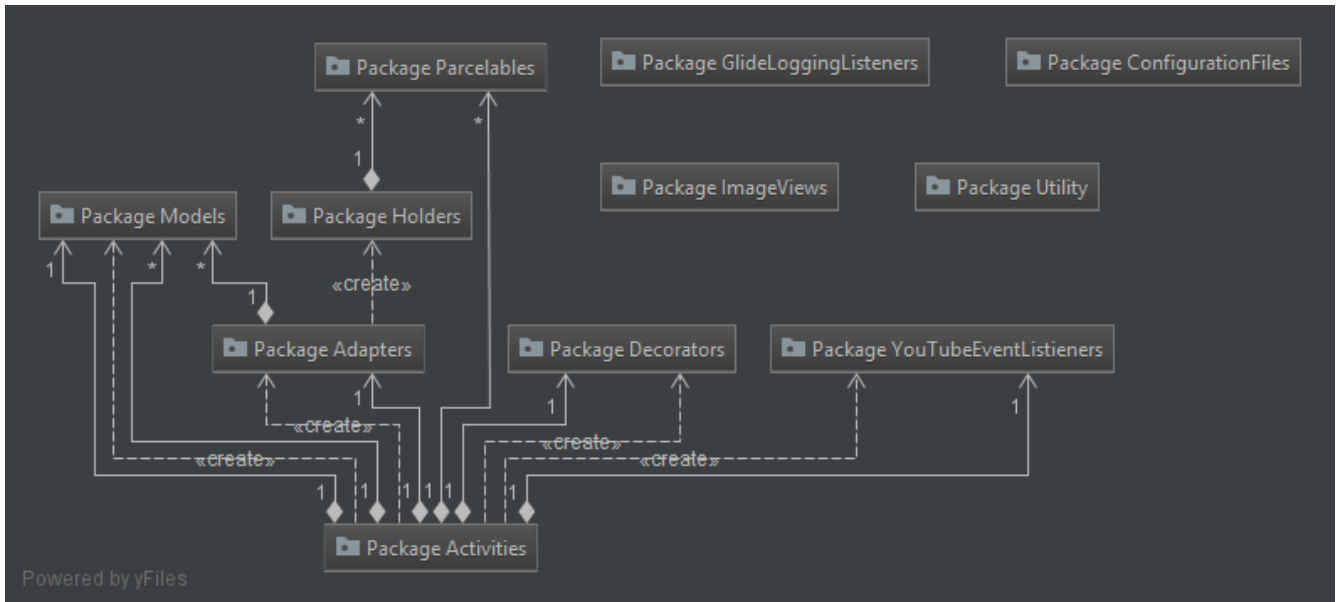
La única excepción a esta regla es la dependencia de la librería **YouTubeAndroidPlayerApi** (API de YouTube para Android), la cual debe ser adquirida por separado. El archivo JAR con esta API debe estar almacenado en la carpeta **libs** del proyecto.

LISTADO DE PAQUETES DEL CÓDIGO PROGRAMADO (JAVA)

El código fuente programado en Java de la aplicación se encuentra distribuido en los siguientes paquetes, los cuales están contenidos dentro del paquete de la aplicación **cr.ac.tec.vis_u**. La siguiente tabla realiza un listado de los paquetes creados por los desarrolladores de la aplicación para la organización del código fuente de la aplicación:

Nombre relativo del paquete	Nombre absoluto del paquete	Descripción
Activities	cr.ac.tec.vis_u.Activities	Contiene las clases asociadas a las Activities de la aplicación.
Adapters	cr.ac.tec.vis_u.Adapters	Contiene las clases con los adaptadores para la asociación de contenido con las Activities de la aplicación.
ConfigurationFiles	cr.ac.tec.vis_u.ConfigurationFiles	Contiene las clases con información de configuración por defecto de la aplicación.
Decorators	cr.ac.tec.vis_u.Decorators	Contiene las clases con elementos de software decoradores de los Layouts utilizados por las Activities
GlideLoginListeners	cr.ac.tec.vis_u.GlideLoginListeners	Contiene las clases observadoras para el registro de las operaciones realizadas durante el uso de la aplicación.
Holders	cr.ac.tec.vis_u.Holders	Contiene las clases que especifican rutinas a ejecutar durante la interacción con la aplicación (ej: solicitudes a servicios web).
ImageViews	cr.ac.tec.vis_u.ImageViews	Contiene clases encargadas de la visualización de las imágenes en las subsecciones de imágenes.
Models	cr.ac.tec.vis_u.Models	Contiene los modelos (clases de objetos con propiedades) utilizados por la aplicación.
Parcelables	cr.ac.tec.vis_u.Parcelables	Contiene las clases Parcelables utilizada para la transición de las clases modelo entre Activities.
Utility	cr.ac.tec.vis_u.Utility	Contiene clases encargadas de facilitar la conectividad de la aplicación a internet.
YouTubeEventListeners	cr.ac.tec.vis_u.YouTubeEventListeners	Contiene las clases relacionadas con el uso de la API de YouTube.

El siguiente diagrama UML contiene las dependencias entre paquetes de Java de la aplicación:



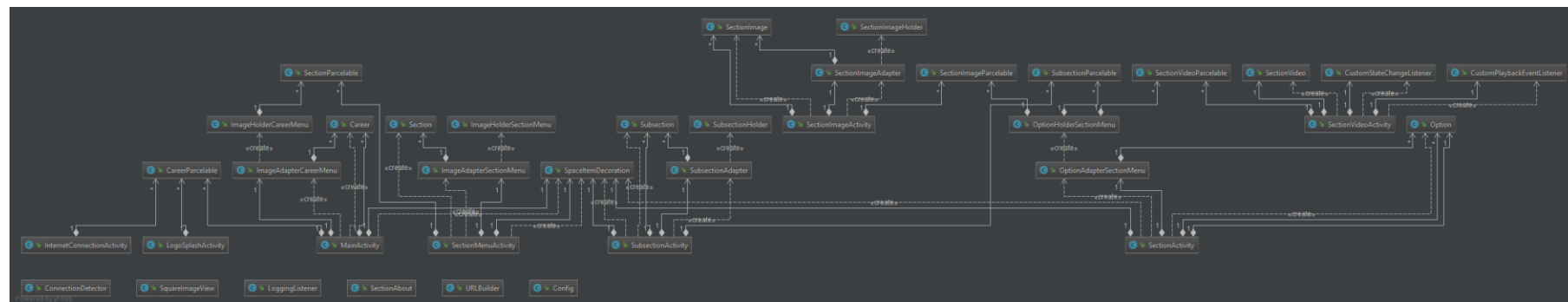
LISTADO DE CLASES POR PAQUETE

La siguiente tabla contiene un listado de las clases creadas por los desarrolladores de la aplicación, clasificadas según la tabla donde están contenidas:

Nombre del Paquete	Nombre de la Clase	Descripción
Activities	InternetConnectionActivity	Clase de la Activity InternetConection (mensaje de error en conexión a Internet).
	LogoSplashActivity	Clase de la Activity LogoSplash (splash screen).
	MainActivity	Clase de la Activity principal con el listado de las carreras.
	SectionAbout	Clase de la Activity SectionAbout (Acerca de...)
	SectionActivity	Clase de la Activity Section, con la información relacionada a una sección de la carrera.
	SectionImageActivity	Clase de la Activity SectionImage con las imágenes de una sección de la carrera.
	SectionMenuActivity	Clase de la Activity SectionMenu con el listado de las secciones de la carrera.
	SectionVideoActivity	Clase de la Activity SectionVideo con los vídeos de YouTube de una sección de la carrera.
	SubsectionActivity	Clase de la Activity Subsection con los elementos de información adicional de una sección de la carrera.
Adapters	ImageAdapterCareerMenu	Clase adaptadora de las imágenes usadas por el listado de carreras.
	ImageAdapterSectionMenu	Clase adaptadora de las imágenes usadas por la información de una sección de la carrera.
	OptionAdapterSectionMenu	Clase adaptadora de las imágenes usadas por el menú de selección de las secciones de la carrera.
	SectionImageAdapter	Clase adaptadora de las imágenes mostrada para la subsección de imágenes de las secciones.
	SubsectionHolder	Clase adaptadora de las tarjetas de contenido con información adicional a una sección.
ConfigurationFiles	Config	Contiene la llave de la API de YouTube usada por la aplicación.
Decorators	SpaceltemDecoration	Clase para determinar las dimensiones de objetos interactivos en pantalla.
GlideLoggingListener	LoggingListener	Clase para llevar registro de actividades de la aplicación.
Holders	ImageHolderCareerMenu	Clase que define las subrutinas de las interacciones del menú de selección de carreras.
	ImageHolderSectionMenu	Clase que define las subrutinas de las interacciones del menú de selección de secciones.
	OptionHolderSectionMenu	Clase que define las subrutinas de las interacciones dentro de la sección de información de la carrera.

	SectionImageHolder	Clase que define las subrutinas de las interacciones con la subsección de imágenes adicionales de la sección.
	SubsectionHolder	Clase que define las subrutinas de las interacciones del contenido de la subsección de información adicional de la sección de la carrera.
ImageViews	SquareImageView	Clase para la generación de las Views para mostrar las imágenes de la subsección de imágenes de la sección de la carrera.
Models	Career	Modelo de información de la carrera.
	Option	Modelo de información de la sección seleccionada.
	Section	Modelo de información de la sección.
	SectionImage	Modelo de información de una imagen de la subsección de imágenes de la sección.
	SectionVideo	Modelo de información de un video de la subsección de videos de la sección.
	Subsection	Modelo de información de la subsección de información adicional de una sección.
Parcelables	CareerParcelable	Parcel para transición de objetos Career entre Activities.
	SectionImageParcelable	Parcel para transición de objetos SectionImage entre Activities.
	SectionParcelable	Parcel para transición de objetos Section entre Activities.
	SectionVideoParcelable	Parcel para transición de objetos SectionVideo entre Activities.
	SubsectionParcelable	Parcel para transición de objetos Subsection entre Activities.
Utility	ConnectionDetector	Clase para detección de cambios en la conexión a internet de la aplicación.
	URLBuilder	Constructor de las direcciones URL para realizar consultas a los servicios web.
YouTubeEventListeners	CustomPlaybackEventListener	Clase para controlar la reproducción de videos de YouTube en la aplicación.
	CustomStateChangeListener	Clase para detectar eventos al reproducir videos de Youtube en la aplicación.

El siguiente diagrama UML contiene una representación visual de las dependencias entre las clases Java de la aplicación:



IMPLEMENTACIÓN DE LA BASE DE DATOS

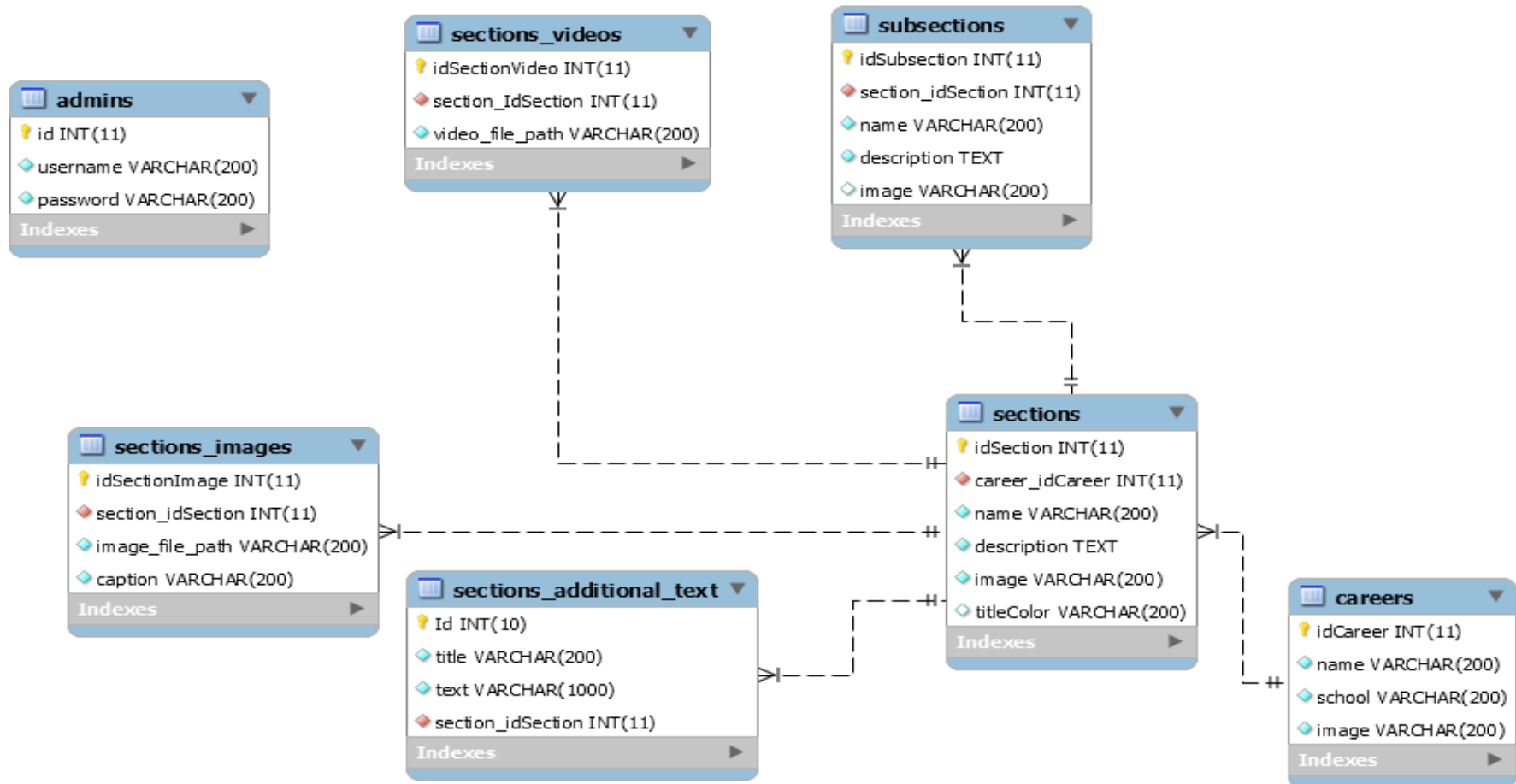
En esta sección se da una explicación del esquema implementado para la base de datos utilizada por la aplicación:

DICCIONARIO DE TABLAS DE LA BASE DE DATOS

Nombre de la tabla	Descripción
admins	Tabla que contiene información de los administradores de la base de datos.
careers	Tabla que contiene un registro de todas las carreras universitarias que serán mostradas en Vis-U.
sections	Tabla que contiene un registro de las secciones de información de cada carrera registrada en la tabla careers.
subsections	Tabla que contiene un registro de datos adicionales relacionados con las secciones registradas en la tabla sections.
sections_additional_text	Tabla que contiene un registro de texto descriptivo adicional de las secciones registradas en la tabla sections.
sections_images	Tabla que contiene las direcciones URL de las imágenes que están relacionadas con las secciones registradas en la tabla sections.
sections_videos	Tabla que contiene los identificadores de los videos en YouTube relacionados con las secciones registradas en la tabla sections.

MODELO LÓGICO DE LA BASE DE DATOS

El siguiente diagrama contiene una representación visual del modelo lógico de la base de datos:



ÍNDICES DE LA BASE DE DATOS

Tabla	Nombre del índice	Estructura de datos	Clave Primaria	Valores Únicos	Empaquetado	Columna(s)	Cotejamiento	Permite valores nulos
admins	PRIMARY	BTREE	Sí	Sí	No	id	A	No
careers	PRIMARY	BTREE	Sí	Sí	No	idCareer	A	No
sections	PRIMARY	BTREE	Sí	Sí	No	idSection	A	No
	Carrera_idCarrera	BTREE	No	No	No	career_idCareer	A	No
sections_additional_test	PRIMARY	BTREE	Sí	Sí	No	Id	A	No
	section_idSection	BTREE	No	No	No	section_idSection	A	No
sections_images	PRIMARY	BTREE	Sí	Sí	No	idSectionImage	A	No
	section_sectionId	BTREE	No	No	No	section_idSection	A	No
section_videos	PRIMARY	BTREE	Sí	Sí	No	idSectionVideo	A	No
	section_sectionId	BTREE	No	No	No	section_idSection	A	No
subsections	PRIMARY	BTREE	Sí	Sí	No	idSubsection	A	No
	section_idSection	BTREE	No	No	No	section_idSection	A	No

CLAVES FORÁNEAS DE LA BASE DE DATOS

Tabla	Nombre de la clave	Columnas	Nombre de la tabla origen	Columna origen	Acción por actualización de clave de origen	Acción por borrado de clave de origen
sections	FK_Career	career_idCareer	careers	idCarrer	CASCADE	CASCADE
sections_additional_text	FK_SectionsAddText_Sections	section_idSection	sections	idSection	CASCADE	CASCADE
sections_images	FK_sectionImage_section	section_idSection	sections	idSection	CASCADE	CASCADE
sections_videos	FK_sectionVideo_section	section_idSection	sections	idSection	CASCADE	CASCADE
subsections	FK_section	section_idSection	sections	idSection	CASCADE	CASCADE

Los servicios web realizan protocolos SOAP para intercambiar datos entre la aplicación. Los intercambios de datos son enviados y recibidos en estructura JSON. Estos JSON son interpretados por el servicio web para así ejecutar las peticiones solicitadas por la aplicación.

Para realizar el envío de datos el servicio web accede a la base de datos por medio de consultas con las cuales obtiene la información requerida para ser enviada en un JSON.

A continuación, se detallan los scripts que conforman este servicio web, dichos scripts fueron realizados PHP. Utilizando la funcionalidad PHP: **mysqli** para establecer conexión y consultas con la base de datos de MySQL.

Los scripts se encuentran ubicados en la carpeta **AMCareer**. El nombre de la base de datos es **AMCareerDB**.

Nombre	Dependencia	Funcionalidad
connector.php	Ninguna	Por medio del usuario, servidor, base de datos, usuario y contraseña realiza una conexión con la base de datos.
career.php	connector.php	Obtiene los nombres de las carreras por medio de la tabla careers y las retorna por medio de un JSON
section.php	connector.php	Obtiene el nombre, descripción, la imagen y el color del título por medio de la tabla sections. Utilizando el identificador (career_idCareer) de la carrera.
sectionAdditionalTexts.php	connector.php	Obtiene el título y el texto de la tabla section_additional_text utilizando el identificador de la sección(section_idSection).
sectionImages.php	connector.php	Obtiene imágenes de la tabla sections_images. Utilizando el identificador (section_idSection).
sectionSubsections.php	connector.php	Obtiene el nombre y la descripción de la carrera de la table subsection. La información la obtiene por medio del identificador section_idSection.
sectionVideos.php	connector.php	Selecciona los videos de la aplicación de la tabla sections_videos por medio del identificador section_idSection.

Desarrolladores originales de la aplicación:

- Rafael Zamora
- Kevin Bejarano
- Ernesto Lang

Diseñador Gráfico:

- Pablo Blandón

Colaboradores:

- Adriana Álvarez Figueroa
- Esteban Fonseca Montoya
- Víctor Aguirre Brenes

Agradecimientos Especiales:

- Tecnológico de Costa Rica (TEC)



© 2017 – Tecnológico de Costa Rica – Todos los Derechos Reservados