

# TUTORÍA EN CITAS VIRTUALES COMO PARTE DE LA FORMACIÓN DEL INGENIERO EN SISTEMAS COMPUTACIONALES

## TUTORING IN VIRTUAL APPOINTMENTS AS PART OF THE FORMATION OF THE COMPUTER ENGINEER

J. C. Rodríguez Campos<sup>1</sup>  
M. Rico Chagollan<sup>2</sup>  
E. García Herrera<sup>3</sup>

### RESUMEN

La tutoría a nivel educativo es un proceso de acompañamiento personalizado para completar algunas necesidades por parte del alumno, la tutoría tiene como finalidad ayudar al alumno a superar cualquier dificultad con el objetivo de reducir los índices de reprobación y disminuir las tasas de deserción escolar. Actualmente, en el Instituto Tecnológico Superior de Irapuato (ITESI) se cuentan con dos perfiles para el alumno, el académico y el personal, donde el docente tutor toma lugar como monitor de alumnos inscritos, los que llevan materias retrasadas y los que han reprobado materias. El objetivo del proyecto es realizar un software para el departamento de tutorías, en el cual permita agendar una entrevista y brindar el seguimiento tutorial, para realizar el monitoreo de manera más eficiente, se podrá visualizar información rápidamente de las materias en las cuales se encuentran en riesgo de reprobación o de ser necesario medir la probabilidad que tiene un alumno en riesgo y poder canalizarlo de forma correcta, con la ayuda del uso de tecnologías de diseño en css y html5 con una portabilidad hacia el enfoque móvil, para modelos de vista controlador.

### ABSTRACT

Tutoring at the educational level is a process of personalized accompaniment to complete some needs by the student, the tutoring is intended to help the student overcome any difficulty with the aim of reducing failure rates and reducing dropout rates. Currently at the Higher Technological Institute of Irapuato there are two profiles for the student, the academic and the staff, where the teacher tutor takes place as a monitor of enrolled students, those who have delayed subjects and those who have failed subjects. The objective of the project is to create a software for the tutoring department, in which it allows to schedule an interview and provide the tutorial follow-up, to carry out the monitoring more efficiently, it will be possible to quickly view information on the subjects in which they are in risk of failure or if it is necessary to measure the probability that a student is at risk and to be able to channel it correctly, with the help of the use of design technologies in css and html5 with a portability towards the mobile approach, for controller view models.

### ANTECEDENTES

Hoy en día en el ITESI, a pesar de que la tutoría es parte de la formación de los alumnos, uno de los problemas que se enfrentan como departamento, es que algunos de ellos no son localizados a tiempo y no se genera ese contacto directo, ya que, el único medio para agendar citas es informar a los alumnos a través del correo institucional y una gran parte de los alumnos no verifican su correo institucional, por lo tanto, no se dan cuenta de las asignaciones, lo que genera un conflicto y esto provoca que los tutores no realicen la labor

<sup>1</sup> Profesor de tiempo completo. Tecnológico Nacional de México / Instituto Tecnológico Superior de Irapuato.  
[juan.rc@irapuato.tecnm.mx](mailto:juan.rc@irapuato.tecnm.mx)

<sup>2</sup> Profesor de tiempo completo. Tecnológico Nacional de México / Instituto Tecnológico Superior de Irapuato.  
[mariana.rc@irapuato.tecnm.mx](mailto:mariana.rc@irapuato.tecnm.mx)

<sup>3</sup> Profesor de tiempo completo. Tecnológico Nacional de México / Instituto Tecnológico Superior de Irapuato.  
[eduardo.gh@irapuato.tecnm.mx](mailto:eduardo.gh@irapuato.tecnm.mx)

de forma adecuada, al no poder contactar a los alumnos a tiempo, como resultado no se puede dar el seguimiento apropiado. Cuando una entrevista es asignada, la asistencia es obligatoria para crear un expediente del alumno con el historial de cada sesión para identificar la problemática que presenta el alumno, con el fin de orientarlos y canalizarlos adecuadamente para evitar la deserción y disminuir el índice de reprobación.

### **Tutoría**

La tutoría se define como el acompañamiento y apoyo docente de carácter individual, basada en una atención personalizada que favorece una mejor comprensión de los problemas que enfrenta el alumno, por parte del profesor, en lo que se refiere a su adaptación al ambiente universitario, a las condiciones individuales para un desempeño aceptable durante su formación y para el logro de los objetivos académicos que le permitirán enfrentar los compromisos de su futura práctica profesional (Asociación Nacional de Universidades e Instituciones de Educación Superior [ANUIES], 2000).

### **Software**

Instrucciones (programas de cómputo) que cuando se ejecutan proporcionan las características, función y desempeño buscados; estructuras de datos que permiten que los programas manipulen en forma adecuada la información, información descriptiva tanto en papel como en formas virtuales que describen la operación y uso de los programas (Pressman, 2003).

### **Lenguaje de programación**

Un lenguaje de programación es un lenguaje formal que proporciona una serie de instrucciones que permiten a un programador escribir secuencias de órdenes y algoritmos a modo de controlar el comportamiento físico y lógico de una computadora con el objetivo de que produzca diversas clases de datos. A todo este conjunto de órdenes y datos escritos mediante un lenguaje de programación se le conoce como programa (Vaca, 2011).

### **Base de datos**

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso (Pérez y Gardey, 2016).

### **Visual Code**

Visual Studio Code es una distribución del repositorio Code - OSS con personalizaciones específicas de Microsoft liberadas bajo una licencia de producto tradicional de Microsoft. Combina la simplicidad de un editor de código con lo que los desarrolladores necesitan para su ciclo básico de edición-compilación-depuración. Proporciona un completo soporte de edición de código, navegación y comprensión junto con una ligera depuración, un rico modelo de extensibilidad e integración ligera con las herramientas existentes. No genera costo el acceso a sus funciones (Sintes, 2020).

### **Aplicación Móvil**

Son componentes descargados desde el internet para desempeñar una función que el usuario requiera o solicite (Carrasco, 2015).

Son contenidos de internet que cubren las necesidades de una Sociedad, proporciona: comunicación, entretenimiento, productividad y otras utilidades (Cuello y Vittone, 2013).

Son herramientas con características especiales, orientadas para dispositivos pequeños como: tabletas o teléfonos inteligentes (Yeeply, 2017).

### **Apps nativas**

Una aplicación nativa es la que se desarrolla de forma específica para un determinado sistema operativo, llamado Software Development Kit o SDK. Cada una de las plataformas, Android, iOS o Windows Phone tienen un sistema diferente.

Cuando se habla de desarrollo móvil casi siempre se refiere a aplicaciones nativas. La principal ventaja con respecto a los otros dos tipos, es la posibilidad de acceder a todas las características del hardware del móvil: cámara, GPS, agenda, dispositivos de almacenamiento y otras muchas. Esto hace que la experiencia del usuario sea mucho más positiva que con otro tipo de apps.

Además, las aplicaciones nativas no necesitan conexión a internet para que funcionen. La descarga e instalación de estas apps se realiza siempre a través de las tiendas de aplicaciones (app store de los fabricantes).

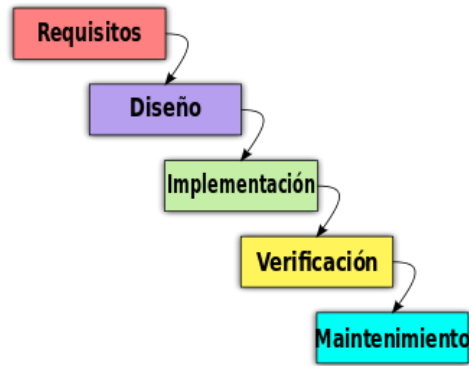
### **Ciclo de Vida**

El ciclo de vida del software está constituido por el conjunto de las etapas que preceden y las que siguen de la programación. Los métodos y técnicas de la ingeniería del software se inscriben dentro del marco delimitado por el ciclo de vida del software, y, más concretamente, por las diferentes etapas que se distinguen (Pressman, 2010).

### **METODOLOGÍA**

En este proyecto es desarrollado como apoyo al proceso de tutorías dentro de la carrera de Ingeniería en Sistemas Computacionales, al desarrollar una aplicación progresiva que permita llevar un control tutorial de los alumnos, al mejorar el acompañamiento individual y grupal y así fortalecer el programa de tutorías en la coordinación. Durante el desarrollo de la aplicación el alumno podrá visualizar el desarrollo de cada una de las etapas del ciclo de vida, brindando al alumno un impacto para su formación académica de cómo se debe llevar a cabo una metodología y adaptando sus conocimientos previos para desarrollar nuevas habilidades y poder implementarlas en un futuro hacia el ámbito laboral. Tomando en cuenta que también podrá conocer los procesos metodológicos apropiados para el desarrollo del software a la medida.

Para la realización de este proyecto se implementa el modelo de cascada, a veces llamado ciclo de vida clásico, sugiere un enfoque sistemático y secuencial para el desarrollo de software, que comienza con la especificación de requerimientos por parte del cliente y avanza a través de planeación, modelado y construcción y despliegue, para concluir con el apoyo del software terminado. Como se muestra en la Figura 1.



**Figura 1.** *Ciclo de Vida en Cascada*

Se ha decidido utilizar la metodología mencionada con anterioridad al ser considerada adecuada por el tiempo posible y permitir al usuario final hacer uso del sistema es un tiempo relativamente corto, tomando en cuenta tres partes importantes para su desarrollo: Programación, laboratorio de usabilidad y propuestas de mejora de la aplicación, como se muestra en la Tabla 1.

**Tabla 1.** *Actividad y etapas del ciclo*

Actividad	Etapas
Requisitos	Programación
Diseño	
Implementación	Laboratorio de usabilidad
Verificación	
Mantenimiento	Propuesta de mejora de la aplicación

Como primer punto se iniciará la programación, la cual consiste en la interfaz del calendario para la realización de citas, posteriormente, cuando se haya finalizado seguirán los laboratorios de usabilidad y por último se harán ciertas observaciones a los posibles cambios o mejoras que se puedan hacer en la app.

Para determinar el proceso de abordar el problema y lograr el objetivo planteado, se dispone de un tiempo determinado para bosquejar un prototipo de la interfaz de calendario y el seguimiento de las pruebas de usabilidad que tendrá, algunas consideraciones que surgen al momento del análisis pre-desarrollo son:

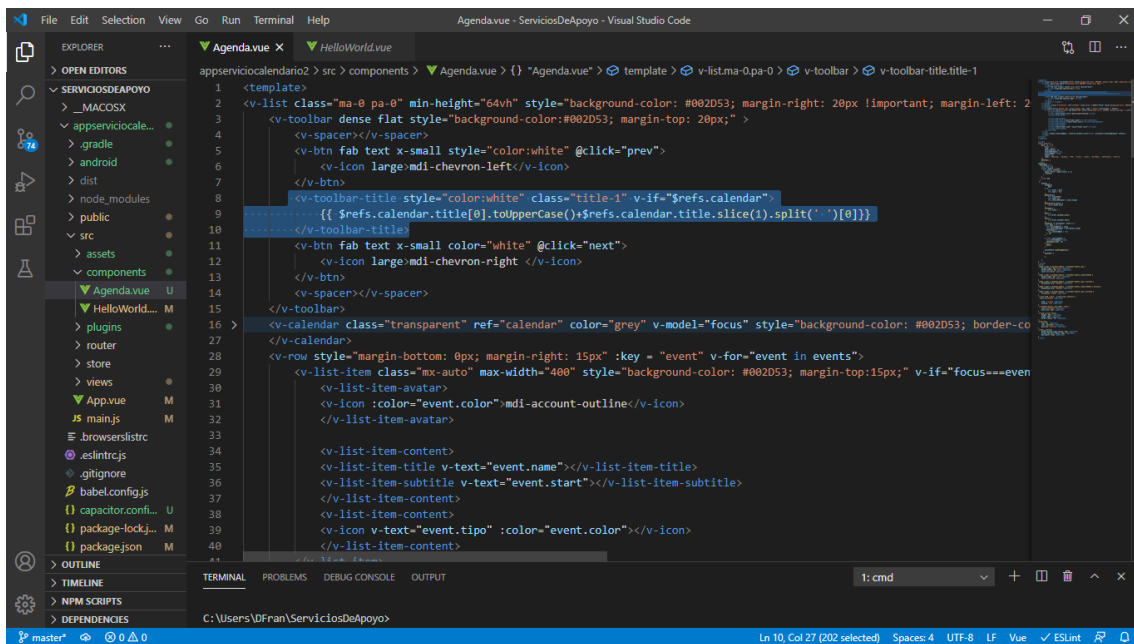
- Mostrar una interfaz llamativa e intuitiva, además de ser legible y clara.
- Cumplir las necesidades que tendrá el usuario al checar su contenido.
- Trabajar con interfaces independientes como paneles de navegación y mutuamente excluyentes como consultas de agenda.
- Evitar la ambigüedad de ejecución (que se abran 2 o más ventanas de la misma instancia)
- Reducir el consumo de recursos y del tiempo en contraste con el uso de los navegadores modernos en páginas web o en el portal escolar.

Con base en el análisis previo del calendario dentro de la aplicación servicios de apoyo de tutorías, se desarrollará en lenguaje de programación Vue.js con HTML 5 y JavaScript en un entorno de que maneja las herramientas y editor de Visual Code. Tomando en cuenta lo anterior para un calendario este deberá ser de manera intuitiva y legible de tal forma que se opta por usar dichas herramientas de desarrollo, además de utilizar diseños de la plantilla creada en Adobe XD para el desarrollo gráfico de la vista calendario que mejoraran el diseño, haciendo uso de los modelos a seguir para desarrollar en móviles, que por la estructura de programación orientada a objetos dichos objetos son de HTML.

Por consiguiente, al ser una aplicación híbrida todo el diseño se basa en el modelo vista controlador que maneja una página web por el mismo modo se pretende usar este modelo para evitar así ambigüedades y conseguir la reducción de memoria y tiempos.

- **Programación**

En esta parte se empieza a programar la parte asignada de la aplicación la cual consiste en la interfaz de un calendario en el cual se mostrarán los eventos que se tengan marcados en este mismo e indicando información de este evento, como se muestra en Figura 2.



```
1 <template>
2 <v-list class="ma-0 pa-0" min-height="64vh" style="background-color: #002D53; margin-right: 20px !important; margin-left: 2
3 <v-toolbar dense flat style="background-color:#002D53; margin-top: 20px;" >
4 <v-spacer></v-spacer>
5 <v-btn fab text x-small style="color:white" @click="prev">
6 <v-icon large>mdi-chevron-left</v-icon>
7 </v-btn>
8 <v-toolbar-title style="color:white" class="title-1" v-if="$refs.calendar">
9   {{ $refs.calendar.title[0].toUpperCase()+$refs.calendar.title.slice(1).split(' ')[0]}}
10 </v-toolbar-title>
11 <v-btn fab text x-small color="white" @click="next">
12 <v-icon large>mdi-chevron-right </v-icon>
13 </v-btn>
14 </v-spacer></v-spacer>
15 </v-toolbar>
16 <v-calendar class="transparent" ref="calendar" color="grey" v-model="focus" style="background-color: #002D53; border-co
17 </v-calendar>
18 <v-row style="margin-bottom: 0px; margin-right: 15px" :key = "event" v-for="event in events">
19 <v-list-item class="mx-auto" max-width="400" style="background-color: #002D53; margin-top:15px;" v-if="focus===even
20 <v-list-item-avatar>
21 <v-icon :color="event.color">mdi-account-outline</v-icon>
22 </v-list-item-avatar>
23 <v-list-item-content>
24 <v-list-item-title v-text="event.name"></v-list-item-title>
25 <v-list-item-subtitle v-text="event.start"></v-list-item-subtitle>
26 </v-list-item-content>
27 <v-list-item-content>
28 <v-icon v-text="event.tipo" :color="event.color"></v-icon>
29 </v-list-item-content>
30 </v-list-item>
31 </v-row>
32 </template>
```

Figura 2. Desarrollo de la programación

En esta Figura 3, se muestra la parte del calendario con el diseño encargado, al momento de programarlo se utiliza para visualizar los resultados el navegador en el local host y este permite ver los cambios en tiempo real cuando se modifica el código.

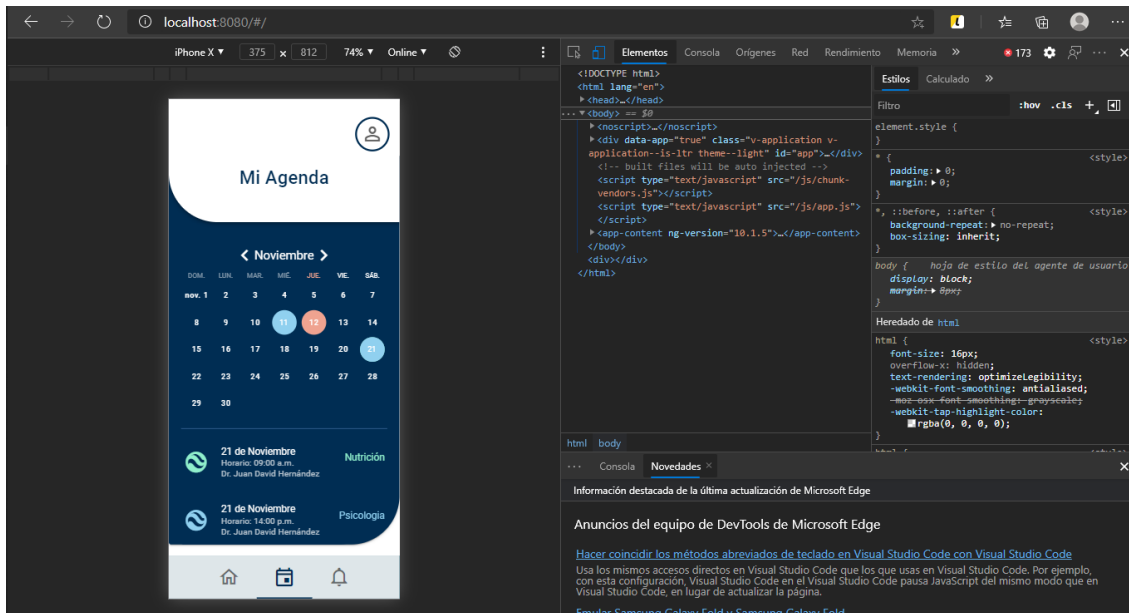


Figura 3. Diseño del calendario y visualización en tiempo real en local host

• **Laboratorio de usabilidad**

Esta etapa se basa en la evaluación del prototipo para realizar las pruebas necesarias con las cuales se podrán detectar los errores que pudieran darse en el momento de la ejecución. Para ello, se utilizarán los formatos de las pruebas de usabilidad, donde se describen los aspectos a evaluar y las distintas tareas que serán efectuadas por los participantes. A continuación, se presentan las fases correspondientes de dicho laboratorio:

Fase 1. Selección de los participantes

- En esta sección se dan a conocer los requisitos previos que deben de tener los participantes para que puedan acceder a las pruebas llenando una consulta previa de conocimientos sobre el tema o aplicativo.

Fase 2. Presentación del diseño

- Es necesario dar a conocer el diseño para realizar dichas pruebas de usabilidad.

Fase 3. Asignación de tareas

- Para esta fase se ocupan una serie de tareas que permiten al participante realizar las evaluaciones en las que se encuentran el reconocimiento del sistema donde se pide al cliente observe la interfaz sin llegar a usarla solo que comente a vista previa su opinión, después de esto se pide que el usuario se ponga en distintos escenarios que usaría el sistema además se dan llenados de formularios y medición de tiempos.

**RESULTADOS**

Como se muestra en la Figura 4, es parte del desarrollo para el funcionamiento de la aplicación móvil, su manejo de la estructura, datos y el consumo de APIs. Cabe mencionar que, durante la programación se trabajó Vuetify, el cual tiene una gran similitud a Bootstrap, ya que, son estilos propios ya definidos para la ayuda de visualización de la aplicación y se obtuvieron aprendizajes nuevos, como implementación de la herramienta, uso de los estilos, edición y creación de estilos.

```

17 <v-calendar ref="calendar" color="#EFA398" v-model="focus" style="background-color: #002D53; border-color: #002D53; overflow-x: hidden;" :ev
18 <template id="date" v-slot:day="{date}">
19 <template v-for="event in eventsMap[date.split(' ')[0]]">
20 <v-menu :key="event.index" v-model="selectedOpen">
21 <template v-slot:activator="{on}">
22 <center>
23 <v-chip class="event-pos" id="selectP" :color="event.color" small v-ripple v-on="on" v-text="event.start.substr(8,9)" @click=
24 {{event.index}}
25 </v-chip>
26 </center>
27 </template>
28 </v-menu>
29 </template>
30 </template>
31 </v-calendar>
    
```

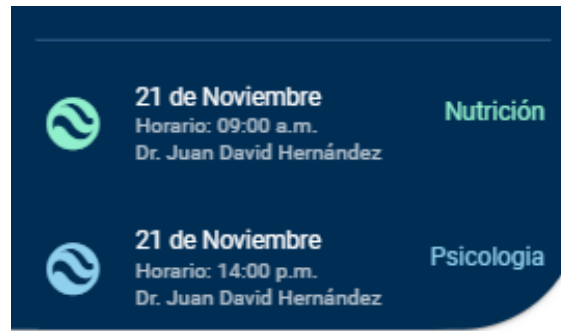
**Figura 4.** Codificación de estilos y edición de la aplicación

En la siguiente Figura 5, se muestra un calendario donde podremos desplazarnos por los diferentes meses del año y dentro de este se muestran los diferentes eventos marcados de forma circular en colores para especificar eventos en los meses. Para desplazarse por el calendario se selecciona alguna de las flechas para cambiar de mes. El calendario muestra los eventos que se tienen en el mes posicionado y son marcados de forma circular y de color; mostrando los siguientes resultados:



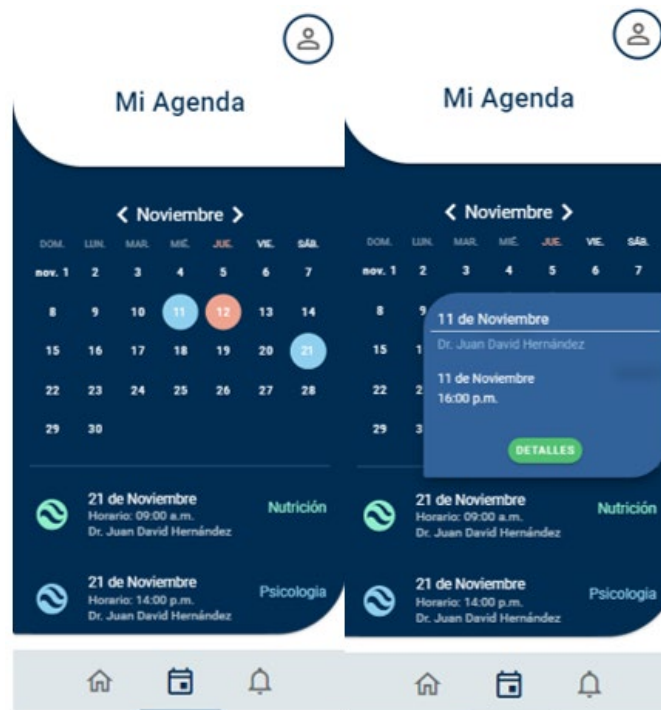
**Figura 5.** Desplazamiento del calendario y marcadores para eventos

Cuando se realiza un evento dentro del calendario, el resultado obtenido se muestra en la Figura 6, en la que se detalla en forma breve con quien se tiene la cita y el día de la misma.



**Figura 6.** Resultados de los eventos próximos.

Los resultados de la interfaz con estos elementos mencionados aparecen de la siguiente manera, donde muestra en la parte superior el calendario marcado con eventos y en la parte inferior los eventos próximos, Figura 7.



**Figura 7.** Resultado de la interfaz final con los eventos y detalles de la entrevista

## CONCLUSIONES

El desarrollo de este proyecto permitió a los alumnos del Instituto Tecnológico Superior de Irapuato (ITESI), de la carrera de Ingeniería en Sistemas Computacionales, pudieran obtener habilidades técnicas y humanísticas que le permitan afrontar retos en el ambiente laboral, habilitándolos para desarrollarse en el cumplimiento de objetivos y solución de problemáticas reales que pueden presentarse en el ambiente profesional, cabe mencionar que por motivos de contingencia las pruebas de usabilidad no se han efectuado, pero por parte de las habilidades obtenidas por el desarrollo de la interfaz que serán de gran ayuda para futuros



proyectos e integrando conceptos a los alumnos en procesos de la elaboración y seguimiento de una metodología de un software.

#### BIBLIOGRAFÍA

Asociación Nacional de Universidades e Instituciones de Educación Superior (2000). *Programas Institucionales de Tutorías*. ANUIES

Carrasco, S. (2015). *Análisis de la aplicación de tecnología móvil en las empresas*. [Tesis de Licenciatura, Universidad Politécnica de Valencia] <https://riunet.upv.es/bitstream/handle/10251/57229/TFC%20Silvia%20Carrasco.pdf?sequence=1>. España: Universidad Politécnica de Valencia.

Cuello, J. y Vittone, J. (2013). *Disigning Mobile Apps* (1<sup>st</sup> Ed.). <https://books.google.com.ec/books?id=nQBJAQAAQBAJ&printsec=frontcover&dq=Cuello,+J.,+%26+Vittone,+J.+App+design+book&hl=en&sa=X&ved=0ahUKEwj08-TC3OXlAhWiv1kKHccNADcQ6AEILzAB#v=onepage&q&f=false>

Peréz, J. y Gardey, A. (2016). *Definición de Base de datos*. Definición. DE. <https://definicion.de/base-de-datos/>

Pressman, R. (2003). *Ingeniería del Software, un enfoque práctico*. Nueva York: McGraw-Hill.

Pressman, R. (2010). *Ingeniería del software, un enfoque práctico*. México: McGrawHill.

Pressman, R. (2013). *Ingeniería del software. Un enfoque práctico*. México: McGraw-Hill Education.

Sintes, B. (2020). *Visual Studio Code*. MCLibre. <https://www.mclibre.org/consultar/informatica/lecciones/vsc.html>

Vaca, C. (2011). *Paradigmas de Programación*. Universidad de Valladolid. <https://www.infor.uva.es/~cvaca/asigs/docpar/intro.pdf>

Yeeply. (21 de marzo de 2017). Qué son las aplicaciones nativas, web e híbridas? [blog] Yeeply. <https://www.yeeply.com/blog/tipos-de-app-y-para-que-sirven/>