



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

TEMA:

**Implementación de un sistema para la gestión de encuestas
en proyectos de vinculación de la Facultad de Ingeniería de
UCSG**

AUTOR:

López Alcívar, Diego Camilo

**Trabajo de titulación previo a la obtención del grado de
INGENIERO EN SISTEMAS COMPUTACIONALES**

TUTOR:

Ing. Salazar Tovar, César Adriano, Mgs

Guayaquil, Ecuador

09 de Marzo del 2018



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

CERTIFICACIÓN

Certificamos que el presente trabajo de titulación, fue realizado en su totalidad por **López Alcívar, Diego Camilo**, como requerimiento para la obtención del Título de **Ingeniero en Sistemas Computacionales**.

TUTOR

f. _____
Ing. Salazar Tovar, César Adriano, Mgs

DIRECTORA DE LA CARRERA

f. _____
Ing. Guerrero Yépez, Beatriz del Pilar, Mgs

Guayaquil, a los 09 días del mes de marzo del año 2018



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

DECLARACIÓN DE RESPONSABILIDAD

Yo, **López Alcívar, Diego Camilo**

DECLARO QUE:

El Trabajo de Titulación **Implementación de un sistema para la gestión de encuestas en proyectos de vinculación de la Facultad de Ingeniería de UCSG** previo a la obtención del Título de **Ingeniero en Sistemas Computacionales**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de nuestra total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 09 días del mes de marzo del año 2018

EL AUTOR

f. _____
López Alcívar, Diego Camilo



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

AUTORIZACIÓN

Yo, López Alcívar, Diego Camilo

Autorizo a la Universidad Católica de Santiago de Guayaquil a la **publicación** en la biblioteca de la institución del Trabajo de Titulación, **Implementación de un sistema para la gestión de encuestas en proyectos de vinculación de la Facultad de Ingeniería de UCSG**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 09 días del mes de marzo del año 2018

EL AUTOR

f. _____
López Alcívar, Diego Camilo

REPORTE URKUND

1.- DATOS SOBRE EL TRABAJO DE TITULACIÓN

PERÍODO: UTE B2017

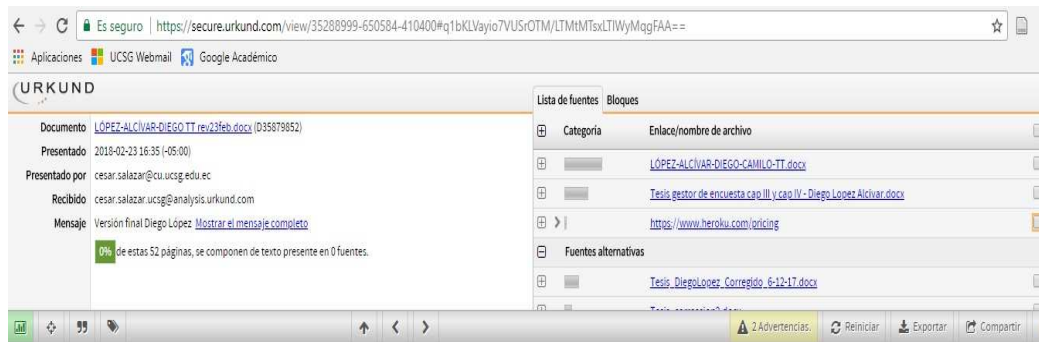
TÍTULO: Implementación de un sistema para la gestión de encuestas en proyectos de Vinculación de la Facultad de Ingeniería de UCSG.

ESTUDIANTE(S): Diego López Alcívar


2.- DESARROLLO

2.1 Observación sobre el porcentaje alcanzado

El trabajo de titulación fue revisado en el software antiplagio URKUND provisto por la UCSG con un porcentaje de 0% encontrándose dentro de los parámetros establecidos por parte de la Universidad, excluyéndose principalmente formatos y carátulas que aparecen en otros trabajos y en avances del mismo trabajo que fueron cargados previamente.



The screenshot shows the URKUND web interface. On the left, document details are displayed: 'Documento: LÓPEZ-ALCIVAR-DIEGO TT rev 23 feb.docx (D35879852)', 'Presentado: 2018-02-23 16:35 (-05:00)', 'Presentado por: cesar.salazar@cu.ucsg.edu.ec', 'Recibido: cesar.salazar.ucsg@analysis.orkund.com', and 'Mensaje: Versión final Diego López. Mostrar el mensaje completo'. A green box indicates '0% de estas 52 páginas, se componen de texto presente en 0 fuentes'. On the right, a 'Lista de fuentes' (List of sources) is shown with columns for 'Categoría' and 'Enlace/nombre de archivo'. Sources include 'LÓPEZ-ALCIVAR-DIEGO-CAMILLO-TT.docx', 'Tesis gestor de encuesta cao III y cao IV - Diego Lopez Alcivar.docx', 'https://www.heroku.com/poncing', and 'Tesis Diego Lopez, Corregido_6-12-17.docx'. The bottom status bar shows '2 Advertencias', 'Reiniciar', 'Exportar', and 'Compartir'.



UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL FACULTAD DE INGENIERÍA CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

TEMA:

Implementación de un sistema para la gestión de encuestas en proyectos de vinculación de la Facultad de Ingeniería de UCSG

AUTOR: López Alcívar,

AGRADECIMIENTO

En primero agradezco a mi madre porque desde el día que nací siempre ha estado apoyándome, animándome y guiándome en todas las adversidades y triunfos que se me han presentado a lo largo de mi vida.

Agradezco a mi padre que siempre ha sido un gran apoyo, continuamente he podido contar con él y siempre ha estado velando por mí.

Por último agradezco a todas las personas que han sido de apoyo que siempre me han brindado su ayuda y he podido contar con ellos. Muchas gracias a todos.

DIEGO CAMILO LÓPEZ ALCÍVAR

DEDICATORIA

A mi madre Maria Teresa Alcívar Avilés y mi padre Jacobo Camilo López Sánchez que han depositado su confianza en mí y siempre he podido contar con ellos les dedico con mucho amor y apego este pequeño esfuerzo y trabajo puesto para la realización de esta tesis. Son mi gran inspiración

DIEGO CAMILO LÓPEZ ALCÍVAR



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

TRIBUNAL DE SUSTENTACIÓN

f. _____

Ing. Beatriz del Pilar, Guerrero Yépez, Mgs
DIRECTORA DE LA CARRERA

f. _____

Ing. Alex, Almeida Campoverde, Mgs
DOCENTE DE LA CARRERA

f. _____

Ing. Galo Enrique, Cornejo Gómez, Mgs
OPONENTE

ÍNDICE GENERAL

RESUMEN	xi
ABSTRACT	xii
INTRODUCCION	1
CAPÍTULO I: MARCO REFERENCIAL.....	3
1.1 Fundamentación conceptual	3
1.1.1 Formulación del problema	3
1.1.2 Justificación	4
1.1.3 Delimitación	4
1.1.4 Pregunta de investigación.....	5
1.1.5 Objetivo general.....	5
1.1.6 Objetivos específicos.....	5
CAPÍTULO II: MARCO TEÓRICO	7
2.1 Antecedentes	7
2.1.1 Fundamentación teórica	8
2.1.1.1 Encuesta.....	8
2.1.1.1.1 Procesamiento de la información	9
2.1.1.1.2 Automatización de encuesta	10
2.1.1.2 Georreferenciación	10
2.1.1.3 Software	12
2.1.1.4 Angular framework	14
2.1.1.5 Single Page Application SPA.....	19
2.1.1.6 Bootstrap	19

2.1.1.7	Node.js	21
2.1.1.8	Express.....	21
2.1.1.9	JavaScript.....	22
2.1.1.10	TypeScript	23
2.1.1.11	Ionic Framework	23
2.1.1.12	MySQL.....	24
2.1.1.13	MySQL Workbench	25
2.1.2	Marco legal	26
CAPÍTULO III: METODOLOGÍA		29
3.1	Tipo de investigación	29
3.1.1	Enfoque metodológico	29
3.1.2	Paradigmas de investigación en Ingeniería de Software	30
3.1.2.1	Metodologías de desarrollo ágil.....	31
3.1.2.1.1	Programación Extrema eXtreme Programming XP.....	32
3.2	Diseño de investigación	33
3.2.1	Investigación no experimental	33
3.2.1.1	Diseño transversal o transeccional.....	35
3.3	Técnicas de recolección de información	36
3.3.1	Entrevista.....	37
3.3.1.1	Entrevista estructurada.....	37
3.4	Instrumentos de recolección de información	37
3.4.1	Cuestionario.....	38
3.5	Análisis de resultados	38

3.5.1	Entrevista 1	38
3.5.2	Entrevista 2.....	40
3.5.3	Entrevista 3.....	41
3.6	Selección de las herramientas de desarrollo	42
CAPÍTULO IV: PROPUESTA TECNOLÓGICA.....		46
4.1	Primera fase: Diseño y desarrollo del proyecto.....	46
4.1.1	Arquitectura de la solución.....	46
4.1.2	Diagrama de casos de uso	47
4.1.2.1	Diagrama de casos de uso para el sistema web	47
4.1.2.2	Diagrama de casos de uso para el aplicativo móvil.....	47
4.1.3	Actores del sistema web y aplicativo móvil	47
4.1.4	Descripción de los casos de uso	48
4.1.4.1	Descripción de los casos de uso – Sistema Web	48
4.1.4.2	Descripción de los casos de uso – Aplicativo Móvil.....	52
4.1.5	Modelo Entidad Relación.....	52
4.1.6	Descripción del diccionario de datos	53
4.1.7	Desarrollo del sistema web.....	53
4.1.8	Desarrollo del sistema móvil	53
4.2	Segunda fase: Implementación del sistema.....	53
4.2.1	Proceso de implementación.....	53
4.2.2	Requerimientos mínimos del sistema web y móvil	54
4.2.3	Análisis costo-beneficio	55
CONCLUSIONES		58

RECOMENDACIONES.....	59
GLOSARIO.....	60
REFERENCIAS.....	62
ANEXOS.....	68

ÍNDICE DE TABLAS

Tabla 1: Comparación de las metodologías tradicionales y las ágiles	32
Tabla 2: Comparación herramientas front-end	42
Tabla 3: Comparación de Bootstrap	43
Tabla 4: Comparación de Node.js.....	44
Tabla 5: Actor del sistema web	48
Tabla 6: Actor del sistema móvil	48

ÍNDICE DE FIGURAS

Figura 1: Resúmenes de G Suite.....	7
Figura 2: Atributos del software	13
Figura 3: Controladores	15
Figura 4: Controladores (2).....	15
Figura 5: Directivas	16
Figura 6: Directivas (2).....	16
Figura 7: Directivas (3).....	17
Figura 8: Directivas (4).....	17
Figura 9: Directivas (5).....	18
Figura 10: Vinculación de la UCSG	27
Figura 11: Diseños de la investigación no experimental	35
Figura 12: Comparación entre Ionic y Android Studio.....	45
Figura 13: Arquitectura de la solución.....	46
Figura 14: Diagrama de casos de uso - Sistema web.....	47
Figura 15: Diagrama de casos de uso - Aplicativo móvil	47
Figura 16: Modelo Entidad-Relación.....	52
Figura 17: Costos referenciales de servicio de nube	56
Figura 18: Costos referenciales de servicio de nube para base de datos....	56

ÍNDICE DE ANEXOS

Anexo 1: Entrevista 1	68
Anexo 2: Entrevista 2	69
Anexo 3: Entrevista 3	70
Anexo 4: Casos de uso sistema web	71
Anexo 5: Caso de uso sistema móvil	75
Anexo 6: Diccionario de datos	76
Anexo 7: Manual de usuario del sistema web.....	79
Anexo 8: Manual de usuario para instalación del aplicativo móvil	98
Anexo 9: Manual de usuario - Aplicativo móvil.....	103
Anexo 10: Manual técnico.....	107

RESUMEN

Para la realización de los proyectos de vinculación de la UCSG, la Facultad de Ingeniería necesita de una herramienta tecnológica que permita levantar información de los grupos objetivo. Por tal motivo, se planteó la implementación de un sistema para la gestión de encuestas, al que pueda acceder tanto en plataforma web como en dispositivo inteligente. La investigación preliminar utilizó la metodología cualitativa, el paradigma de programación formulativo, Programación Extrema como metodología de desarrollo ágil, la investigación no experimental con diseño transversal y la entrevista como instrumento de recolección de datos, la misma que se la aplicó a tres docentes de la Facultad y Universidad, de donde se obtuvieron los requerimientos funcionales y se trató algunos temas relacionados con la correcta elaboración de una encuesta. Con la información obtenida, se caracterizaron las necesidades y requerimientos que debía tener el sistema, determinándose las herramientas óptimas de desarrollo, con el fin de que la experiencia de usuario final sea agradable y que las funcionalidades del aplicativo web y móvil sean aprovechadas en su totalidad. Se diseñó e implementó el sistema gestor de encuestas para plataforma web y móvil para los usuarios finales, y finalmente se propusieron algunas recomendaciones a tomarse en consideración para el mejoramiento del mismo.

PALABRAS CLAVE: automatización, encuesta, georreferenciación, sistema web, aplicativo móvil, Angular, Bootstrap, Node.js, Ionic

ABSTRACT

In order to execute junction projects from UCSG, the Engineering Faculty needs a technological tool that could raise information from target groups. For this motive it has been suggested the implementation of a system for the poll management on which the user could have access in a web platform as through a smart device. The preliminary investigation used qualitative methodology, the paradigm of formulate programming, extreme programming like nimble development methodology, non-experimental investigation with cross design and interviewing for data harvest, the same interview was applied to three professors from the Faculty and University, on which the functional requirements where obtained and topics related with proper poll elaboration where treated. With the obtained information, the needs and requirements of the system where characterized, determining optimal development tools with the purpose of an enjoyable experience by the final user and a total functionality exploit of the web and mobile application. It has been designed and implemented the account management system for the web and mobile platform for the final users, and finally some conclusions and recommendations for the system enhancement where stated for consideration.

KEYWORDS: automation, poll, geo-referencing, web system, application, mobile, angular, bootstrap, Node.js, Ionic

INTRODUCCION

En la era moderna que se vive en la actualidad, las nuevas tecnologías juegan un papel importante para el día a día; el uso de estas nuevas herramientas es imprescindible para el desempeño de todo tipo de actividades en la vida cotidiana de cualquier individuo. Por eso el adaptarse a este nuevo entorno es ineludible.

Las encuestas son métodos de recolección de datos que es fundamental para cualquier tipo de investigación que se tenga que realizar. Las encuestas tienen una variedad de finalidades y dependiendo de cómo se la desarrolle pueden tener un gran impacto en la resolución de cualquier investigación.

El proyecto actual se engloba dentro del área de desarrollo de aplicaciones móviles y de web. La necesidad de este tipo de herramienta metodológica digital es de utilidad para equipos de investigación que requieran realizar encuestas donde se puede incluir la georreferenciación de los puntos encuestados.

La idea inicial se basa en desarrollar una aplicación tanto para dispositivos móviles como para la plataforma web, que permita a los usuarios diseñar, proponer, cuantificar y analizar encuestas propuestas como herramienta metodológica de investigación predeterminada con el fin de poder interpretar, validar, cuantificar las respuestas de las poblaciones objetivos para validar propuestas o determinar problemas o caracterizar poblaciones, además de incluir la georreferencia que es de importancia vital en el análisis de resultados.

Existen muchas herramientas para la aplicación de encuestas y la tabulación de resultados. Un ejemplo claro es la que ofrece el buscador Google. El problema que se quiere abordar es que la Carrera de Ingeniería en Sistemas Computacionales y la nueva carrera de Ciencias de la Computación, necesitan para sus proyectos de vinculación, una herramienta para generación de encuestas que registren la referencia geográfica de cada cuestionario aplicado. En el ámbito de investigación científica una de las

herramientas metodológicas más utilizada es la encuesta, es posible diseñar una aplicación lo suficientemente dinámica tanto para plataforma web como para móvil que pueda facilitar la aplicación de encuesta, como para la cuantificación, georreferenciación y el análisis de los resultados.

CAPÍTULO I: MARCO REFERENCIAL

La Universidad Católica de Santiago de Guayaquil UCSG, a través de sus proyectos de vinculación con la sociedad, propone e implementa programas, proyectos y acciones que permiten un acercamiento de la universidad con sectores de interés social, con el fin de mantener una relación directa con dichos sectores y poder trabajar en pro de su bienestar, generando procesos conjuntos de movilización para esos sectores comunitarios (UCSG, 2017a).

Para poner en marcha los proyectos de vinculación, cada facultad de la UCSG selecciona grupos estudiantiles para dichos proyectos, los mismos que necesitan todas las herramientas adecuadas para su trabajo de campo. Por tal motivo, la Carrera de Ingeniería en Sistemas Computacionales, debe tener una herramienta informática básica, que ayude en la toma de datos para realizar encuestas.

El proyecto que se desarrolla es la *Implementación de un sistema para la gestión de encuestas en proyectos de vinculación de la Facultad de Ingeniería de la UCSG*, mediante el cual, los estudiantes que forman parte de los proyectos de vinculación de la UCSG, podrán tener a su alcance una herramienta que simplifique la gestión de la información que se obtenga en el trabajo de campo, y ahorre tiempo en procesar dicha información para una toma de decisiones más efectiva.

1.1 Fundamentación conceptual

Para el inicio del proyecto, es indispensable describir el entorno del problema a solucionar, justificando las causas de su realización, delimitación del mismo, los objetivos, los fundamentos teóricos, el marco legal y el marco conceptual, temas que se abordan en los párrafos siguientes.

1.1.1 Formulación del problema

Herramientas para la aplicación de encuestas y la tabulación de resultados hay muchas empezando por la ofrecida por Google. El problema que se quiere abordar es que la Carrera de Sistemas necesita para sus proyectos de vinculación con la sociedad, una herramienta para toma de encuestas

que registren la referencia geográfica de cada cuestionario aplicado. En el ámbito de investigación científica, una de las herramientas metodológicas más utilizada es la encuesta, es posible diseñar una aplicación la suficientemente dinámica tanto para plataforma web como para móvil que pueda facilitar la aplicación de encuesta, como la cuantificación, georreferencia y el análisis de los resultados.

1.1.2 Justificación

El producto resultante de este trabajo de titulación responde a la necesidad de contar con una herramienta que facilite los medios y condiciones para el diseño, aplicación y análisis de encuestas en puntos estratégicos, proveyendo al usuario la rapidez y confiabilidad de los resultados.

Es de resaltar que su aplicabilidad es amplia y los beneficios que ofrece son innumerables. En el ámbito educativo, se convierte en una herramienta de soporte para el aprendizaje dentro del ámbito investigativo, ya que se puede registrar, mediante la georreferenciación, los puntos geográficos en donde se aplica una encuesta. Por tanto, profesores y estudiantes se beneficiarían de este aplicativo y podría ser un punto de partida para nuevas ideas e investigaciones relacionadas con este campo de estudio.

Finalmente, este estudio está enmarcado en la línea investigación y desarrollo de nuevos servicios y productos establecida por la carrera Ingeniería en Sistemas Computacionales dentro de su quehacer educativo

1.1.3 Delimitación

El proyecto se lo realizará para la Carrera de Ingeniería en Sistemas Computacionales de la Facultad de Ingeniería de la UCSG como herramienta a utilizar por los estudiantes que realicen proyectos de vinculación con la sociedad, auspiciados por la UCSG.

Dentro del alcance del proyecto se debe tomar hasta dónde se llegará tanto en la plataforma móvil como en la plataforma web. En la plataforma móvil se toma en cuenta lo siguiente:

- Ingreso de encuestas incluyendo datos de geo localización;
- En línea, para guardar datos directamente en la base de datos de la aplicación;
- Fuera de línea, para guardar datos en el dispositivo hasta que pueda conectarse y actualizar base de datos;

En cuanto a la plataforma web, el sistema realizará:

- Creación de encuestas con soporte de diferentes tipos de preguntas;
- Creación de encuestadores;
- Asignación de encuestas a proyectos,
- Tabulación de datos;
- Generación de informes estadísticos uso de barras, círculos y mapas;
- Resultados filtrados por proyectos, por preguntas, por datos generales, por sectores, por encuestadores.

1.1.4 Pregunta de investigación

El desarrollo e implementación de este proyecto obliga al planteamiento de la pregunta de investigación:

¿Es factible desarrollar e implementar un sistema gestor de encuestas para plataforma web como móvil que sea dinámica y que permita la aplicación de encuesta, la cuantificación, georreferenciación y el análisis de los resultados?

1.1.5 Objetivo general

Diseñar e implementar una aplicación de gestor de encuesta para plataforma web y móvil propuesto por la Carrera de Sistemas Computacionales en el 2018 para los proyectos de vinculación con la sociedad.

1.1.6 Objetivos específicos

- Caracterizar las necesidades y requerimientos que debe cumplir la herramienta que se va a desarrollar;

- Identificar las herramientas de software para desarrollar la aplicación de acuerdo con los requerimientos identificados;
- Diseñar e implementar el gestor de encuestas para la Carrera de Sistemas Computacionales.

CAPÍTULO II: MARCO TEÓRICO

En los párrafos siguientes, se hace una revisión pormenorizada de la literatura relacionada con el proyecto a desarrollar, para tener un contexto más claro de lo que se estudia.

2.1 Antecedentes

Para recolectar información de muestras determinadas, existen plataformas que ofrecen este servicio en la web. SurveyMonkey es una solución con suscripción profesional o gratuita que sirve para crear, enviar y analizar encuestas, que llega a las personas de las que se desea conocer información mediante las preguntas adecuadas; en su edición profesional, la eficiencia de la herramienta se refleja en la satisfacción del cliente (en tiempo real), compromiso de los empleados (cómo retener al mejor personal), planificación de eventos, análisis de educación y escuelas, realización de investigaciones de mercado, realizado desde cualquier dispositivo (SurveyMonkey, 2017).

Otra plataforma que ofrece herramienta de encuesta es “G Suite by Google Cloud”, que permite crear formularios de forma fácil y sencilla para encuestas, con aspecto profesional en tiempo real. El análisis de los resultados se los puede realizar con resúmenes automáticos generados por la misma herramienta (Google, 2017).



Figura 1: Resúmenes de G Suite. Tomado de Google (2017)

Los ejemplos de las herramientas de toma de encuesta antes analizadas, presentan algunos beneficios para los profesionales de la investigación. De lo que no disponen es de una georreferenciación de los puntos geográficos en donde se aplica una encuesta, por lo que el desarrollo e implementación de un sistema para la gestión de encuestas en proyectos de vinculación de la Facultad de Ingeniería de la UCSG, facilitará el análisis de la información en lugares estratégicos, y ofrecerá al usuario final confiabilidad y rapidez en el análisis de los resultados.

2.1.1 Fundamentación teórica

A continuación, se explican los conceptos utilizados para el desarrollo del proyecto de titulación. Se trabajará con Angular +2 y otros marcos de referencia para aplicaciones web, los lenguajes de programación TypeScript y JavaScript. Como base de datos se utilizará MySQL que es de código abierto, además de otras herramientas que se definirán para el funcionamiento del sistema.

2.1.1.1 Encuesta

De acuerdo a Bernal (2010) la encuesta es la más utilizada técnica de recopilación de datos, que se basa en la aplicación de un grupo de preguntas, preparadas para investigar y obtener información de una muestra o grupo de personas.

Por su parte Arias (2006) manifiesta que la encuesta es, asimismo, una técnica de recolección de información, aplicada hacia una persona o grupo de personas con la finalidad de conseguir datos sobre temas que tienen en común ese grupo o persona.

Por lo tanto, se entiende por encuesta a la técnica de recolección de datos, que es muy utilizada en investigación, para obtener información de una muestra o grupo de personas representativas que se estudia, sobre aspectos comunes en ellos, tales como costumbres, actitudes, opiniones o comportamientos. Se aplica un cuestionario de preguntas, el mismo que debe ser elaborado tomando en consideración los aspectos metodológicos de la investigación.

Según Behar Rivero (2008) para la aplicación de la encuesta se debe tener en cuenta el tamaño de la muestra, el mismo que dependerá de los procedimientos estadísticos que se utilicen para determinar los hallazgos y cómo se los utilizarán.

La información recogida a través de este instrumento, sólo es un conjunto de datos que no tiene ningún significado en relación al objetivo general y al contexto del estudio. Para que esta información adquiera el sentido que debe tener, se requiere que ésta pase por un proceso de organización, el respectivo análisis y su correspondiente interpretación, ya que constituirían el componente principal sobre el cual realizar el trabajo de investigación (Niño Rojas, 2011).

2.1.1.1 Procesamiento de la información

Luego de la recolección de la información, de acuerdo a Niño Rojas (2011) deberá realizar, por lo menos tres tareas: a) organización, b) codificación, y c) tabulación. En la *organización*, el investigador debe realizar un control de todo lo obtenido, estableciendo el tipo de información de la que se dispone, para poder seleccionarla, ordenarla y organizarla.

Luego de realizar el anterior proceso, se deberá clasificar la información en datos cualitativos y cuantitativos, para establecer una *codificación* y determinar si existen incongruencias o errores, ya que la información puede ser verbal (cualitativa) o numérica (cuantitativa); esta última deberá ser procesada a través de métodos estadísticos.

Sobre la *tabulación* de los datos, esto se refiere a la elaboración de tablas en las cuales se agrupan más fácilmente la información recogida y su contabilización para luego ser analizados. Al decir de Sabino (1998) referenciado por Niño Rojas (2011) se debe “confeccionar tablillas u hojas de tabulación donde figuran los códigos con base en los cuales se habrán de distribuir los datos y espacios para señalar, mediante signos convencionales, las unidades que se van contabilizando” (p. 102).

Al final de la tabulación, se procede a realizar el análisis de los resultados, para conocer las características, naturaleza y relaciones de los datos, con el

fin de llegar a una síntesis y establecer, por último, conclusiones, resultados y consecuencias de lo que se obtuvo a través de este instrumento.

2.1.1.1.2 Automatización de encuesta

Recolectar información a través de cuestionarios de preguntas a una determinada muestra que se estudia, es una actividad que varios o todos los sectores de la sociedad llevan a cabo, para realizar estudios que a cada sector le interesa. La encuesta es el medio más sencillo de retroalimentarse de la opinión del encuestado que es un usuario o consumidor de un servicio.

Con el avance de las TIC's, se hizo necesario mejorar este instrumento de recolección de datos, para que el encuestador pueda asegurar las distintas incidencias y respuestas a la encuesta mientras realiza su trabajo. Hoy en día, quienes realizan encuestas no necesariamente realizan trabajos de campo, sino que se sirven de la tecnología para mayor eficiencia en sus investigaciones. Por este motivo, existen cada vez más soluciones *online* que prestan este servicio para que, en la comodidad de su casa u oficina, el encuestado pueda responder las preguntas que el encuestador le envíe, sin que éste se mueva de su lugar de trabajo.

Las aplicaciones de encuestas conjugan lenguajes de programación, base de datos y servicios web para poder ofrecer al usuario formularios amigables y fáciles de manipular, para que sea llenado de forma rápida y con el mínimo de errores en las respuestas.

El tema de la automatización es el fundamento de este proyecto, ya que se busca proporcionar un aplicativo a la Facultad de Ingeniería, para sus proyectos de vinculación con la sociedad.

2.1.1.2 Georreferenciación

La información que todo investigador obligatoriamente necesita conocer sobre la muestra que estudia a través de la aplicación de encuestas, debe estar vinculada con el entorno en que aquella se desenvuelve; dicho entorno puede ser local (ciudad) o rural o comunitario, hacia donde se dirige la encuesta y en donde debe llevarse a cabo un trabajo de campo.

En el caso de que la encuesta que se aplique sea en un medio rural o comunidad rural, puede darse el caso de que los medios o herramientas tecnológicas no estén debidamente utilizados o quizás no exista una cultura tecnológica. Hoy el internet llega a poblados lejanos, pero no todas las personas que hacen uso del mismo están al tanto de sus beneficios, por lo que una herramienta en línea para la realización de una encuesta no sería adecuada en estos lugares.

Todo lo anterior se orienta hacia la comprensión de la necesidad de desarrollar una herramienta tecnológica para la toma de encuestas, que se realicen en la UCSG en sus proyectos de vinculación con la sociedad y en particular en la Facultad de Ingeniería. Dicha herramienta de automatización de toma de encuesta debe proporcionar la localización de los elementos (personas) que se encuentren en un espacio determinado (comunidades rurales o locales) a través de la georreferenciación, que se la define en el párrafo siguiente.

Según lo manifestado por Calero Clavijo (2006) la georreferenciación consiste “en ligar o relacionar una información geográfica a unos puntos en común, es decir poder llevar toda la información a unas coordenadas bases para que posteriormente otros datos se puedan unir al mismo y estos guarden la misma relación” (p. 6) por cuanto este proceso proporciona al investigador un entorno sobre el cual trabajar.

Otro concepto de georreferenciación se refiere a la utilización de las coordenadas de un mapa, en donde se ubican identidades cartográficas espacialmente. Cada uno de los componentes de un mapa poseen ubicación y extensión determinadas, y pueden ser ubicadas sobre la superficie terrestre o junto a ella (ArcGIS Resource Center, s.f.).

De los conceptos antes anotados, se puede decir que la georreferenciación es la asignación de un grupo de coordenadas a un grupo de puntos que se encuentran dentro de un determinado territorio, con el fin de que puedan definirse dentro de un mapa.

2.1.1.3 Software

Según Pressman (2010, p. 1) se entiende como software al:

Producto que construyen los programadores profesionales y al que después le dan mantenimiento durante un largo tiempo. Incluye programas que se ejecutan en una computadora de cualquier tamaño y arquitectura, contenido que se presenta a medida de que se ejecutan los programas de cómputo e información descriptiva tanto en una copia dura como en formatos virtuales que engloban virtualmente a cualesquiera medios electrónicos (p. 1).

Sommerville (2011) definió al software como “programas de cómputo y documentación asociada. Los productos de software se desarrollan para un cliente en particular o para un mercado en general” (p. 6).

Por otro lado, Ramírez & Weiss (1986) citados por Medina-Lozano, Villalobos-Salmerón, & Ordaz Celedón (2017) se define al software “como todos aquellos conceptos, actividades y procedimientos que dan como resultado la generación de programas para un sistema de computación (p. 4). Lo que se busca es el aumento de las posibilidades de que el software se lo realice a tiempo y sea efectivo en lo referente a costos, para que exista mayor eficiencia tanto en recursos como en el personal de trabajo.

De los conceptos anteriores se puede concluir que el software o programa de ordenador es un conjunto de aplicaciones que utilizan distintos lenguajes de programación para generar un producto que pueda controlar un equipo informático. El software es la parte lógica que tiene un equipo de cómputo a través de los cuales se pueden realizar tareas específicas, y que se las desarrolla para el cliente que lo solicita.

Los atributos de un buen software los propuso Sommerville (2011) y se muestran en la figura 2.

Características del producto	Descripción
Mantenimiento	El software debe escribirse de tal forma que pueda evolucionar para satisfacer las necesidades cambiantes de los clientes. Éste es un atributo crítico porque el cambio del software es un requerimiento inevitable de un entorno empresarial variable.
Confiabilidad y seguridad	La confiabilidad del software incluye un rango de características que abarcan fiabilidad, seguridad y protección. El software confiable no tiene que causar daño físico ni económico, en caso de falla del sistema. Los usuarios malintencionados no deben tener posibilidad de acceder al sistema o dañarlo.
Eficiencia	El software no tiene que desperdiciar los recursos del sistema, como la memoria y los ciclos del procesador. Por lo tanto, la eficiencia incluye capacidad de respuesta, tiempo de procesamiento, utilización de memoria, etcétera.
Aceptabilidad	El software debe ser aceptable al tipo de usuarios para quienes se diseña. Esto significa que necesita ser comprensible, utilizable y compatible con otros sistemas que ellos usan.

Figura 2: Atributos del software. Tomado de “Ingeniería de Software” por Sommerville, (2011). Madrid: Pearson Educación

De acuerdo a la figura 2, el software tiene cuatro características. En lo relacionado con el *mantenimiento*, se puede decir que el software debe ser desarrollado de forma que sea escalable, ya que con el paso del tiempo y el constante cambio de tecnología y de las necesidades de los clientes, aquel debe tener la posibilidad de adaptarse a los nuevos requerimientos.

El software tiene *confiabilidad y seguridad*, es decir que su desarrollo debe ofrecer a sus usuarios seguridad, confiabilidad y protección, ya que no su desarrollo no debe ocasionar daños físicos ni económicos, en el supuesto caso de que se presente algún fallo; además, de que deberá tener las debidas seguridades para evitar intrusiones no deseadas que ocasionen perjuicio a los usuarios del sistema.

La *eficiencia* del software se relaciona con el aprovechamiento de los recursos informáticos, esto es, ahorro en memoria y procesos, y está encaminada hacia una eficiente respuesta, procesamiento de los datos, memoria, entre otros.

En cuanto a la *aceptabilidad*, el software deberá ser diseñado y desarrollado a medida para el usuario que lo solicita. De este modo, el usuario podrá encontrar un producto que tenga usabilidad y compatibilidad con otros productos que maneje, además de ser comprensible y de fácil utilización.

2.1.1.4 Angular framework

Anteriormente llamado Angular 2, Angular es un nuevo framework de trabajo que tiene nuevos conceptos y formas de trabajo diferentes, desarrollado desde cero.

Angular usa un “sistema de inyección de dependencias jerárquico impulsando su rendimiento...” (campusMVP, 2017, párr. 2) en gran escala. Además, detecta cambios fundamentados en árboles unidireccionales, los mismos que también apoyan el rendimiento. Se conoce que este framework supera a AngularJS en su rendimiento.

Por ser una nueva herramienta y no una actualización de Angular JS, aprender Angular es largo y complicado para los programadores; está escrito en lenguaje de programación Typescript, cumpliendo con el estándar de JavaScript ECMAScript 6 (ES6 o ES2015). El nuevo framework ha sido reescrito desde sus inicios y presenta una serie de creaciones que no tiene su antecesor.

Entre los cambios que ofrece Angular, están los siguientes:

- Mejor rendimiento y orientación hacia dispositivos móviles, con bibliotecas que apoya el desarrollo de aplicaciones móviles de forma rápida. Se renderiza el código en los navegadores de forma diferente, que en las aplicaciones hacia dispositivos móviles;
- Mayor oportunidad de escoger lenguajes de programación. Se puede utilizar ECMAScript 5, ES6, TypeScript o Dart. La utilización de TypeScript, que es su lenguaje por defecto, es la mejor forma de escribir JavaScript;
- No existen más el \$scope de AngularJS y los controladores, los mismos que han sido cambiados por componentes web;

Controlador Angular 1.x

En JavaScript:

```
var myApp = angular
  .module("miModulo", [])
  .controller("productoController", function($scope) {
    var prods = { name: "Prod1", quantity: 5 };
    $scope.products = prods;
  });
```

Controlador Angular

En TypeScript:

```
import { Component } from '@angular/core';

@Component({
  selector: 'prodsdata',
  template: '<h3>{{prods?.name}}</h3>'
})

export class ProductoComponent {
  prods = {name: 'Prod1', quantity: 5};
}
```

Figura 3: Controladores. Tomado de "Las 10 principales diferencias entre AngularJS y Angular", por campusMVP (2017)

- Cambios en la sintaxis de directivas estructurales.

Directivas estructurales Angular 1.x:

```
<ul>
  <li ng-repeat="prod in productos">
    {{prod.name}}
  </li>
</ul>
```

Directivas estructurales Angular:

```
<ul>
  <li *ngFor="prod of productos">
    {{prod.name}}
  </li>
</ul>
```

Figura 4: Controladores (2). Tomado de "Las 10 principales diferencias entre AngularJS y Angular", por campusMVP (2017)

- Uso directo de elementos y eventos del modelo DOM Document Object Model o DOM ('Modelo de Objetos del Documento' o 'Modelo en Objetos para la Representación de Documentos'), por lo que

directivas integradas en Angular 1.x y las basadas en eventos ya no se utilizan;

En AngularJS:

```
<button ng-click="doSomething()">
```

En Angular simplemente tomas el evento estándar y lo envuelves entre paréntesis:

```
<button (click)="doSomething()">
```

Figura 5: Directivas. Tomado de “Las 10 principales diferencias entre AngularJS y Angular”, por campusMVP (2017)

- Cambio a [property] la directiva de datos enlazados en una sola dirección, es decir, que únicamente cambia hacia la vista, el enlace del código;

En Angular 1.x

```
<input ng-bind="prod.name"></input>
```

En Angular se utilizan simplemente corchetes sobre la propiedad estándar:

```
<input [value]="prod.name"></input>
```

Figura 6: Directivas (2). Tomado de “Las 10 principales diferencias entre AngularJS y Angular”, por campusMVP (2017).

- [(ngModel)] reemplaza a ng-model, esto es, en cuanto a enlaces de datos de doble dirección, para mejor rendimiento, control y seguridad, con ventajas en la gestión de formularios;

En Angular 1.x

```
<input ng-model="prod.name"></input>
```

En Angular la sintaxis equivalente sería:

```
<input [(ngModel)]="prod.name"></input>
```

Figura 7: Directivas (3). Tomado de “Las 10 principales diferencias entre AngularJS y Angular”, por campusMVP (2017)

- Cambio de sintaxis de la inyección de dependencias, que se realiza a través de constructores;

En Angular 1.x:

```
var myApp = angular
  .module("miModulo", [])
  .controller("productoController", function($scope, $http) {
    var prods = { name: "Queso", quantity: 5 };
    $scope.productos = prods;
  });
```

En Angular:

```
import { Injectable } from '@angular/core';
import { Http } from '@angular/http';
import { Producto } from 'app/shared/Producto.ts';

@Injectable()

export class ProductosService {
  constructor(private _http: Http) { }
  getProductos() {
    return [new Producto(1, 'Queso'),
            new Producto(2, 'Pan',
            new Producto(3, 'Verdura')
    ];
  }
}
```

Figura 8: Directivas (4). Tomado de “Las 10 principales diferencias entre AngularJS y Angular”, por campusMVP (2017)

- Cambio en sintaxis para routing, con el uso de `@RouteConfig{(...)}`.

En Angular 1.x y JavaScript:

```
var app = angular
    .module("miModulo", ["ngRoute"])
    .config(function ($routeProvider) {
        $routeProvider
            .when("/inicio", { templateUrl: "inicio.html", controller: "i
            .when("/Productos", { templateUrl: "productos.html", controll
        })
    })
    .controller("inicioController", function ($scope) {
        $scope.message = "Página de inicio";
    })
    .controller("productosController", function ($scope) {
        $scope.prods = ["Queso", "Pan", "Verdura"];
    })
```

En Angular y TypeScript:

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';

import { InicioComponent } from './inicio/inicio.component';
import { ProductoComponent } from './productos/producto.component';

const appRoutes: Routes = [
    {
        path: 'inicio',
        component: InicioComponent,
    },
    {
        path: 'productos',
        component: ProductoComponent,
    },
];

@NgModule({
    imports: [
        RouterModule.forRoot(appRoutes)
    ]
})
export class AppRoutingModule { }
```

Y en la página HTML:

```
<ul>
  <li><a routerLink="/inicio">Página de inicio</a></li>
  <li><a routerLink="/productos">Productos</a></li>
</ul>
```

Figura 9: Directivas (5). Tomado de “Las 10 principales diferencias entre AngularJS y Angular”, por campusMVP (2017)

- Las aplicaciones Angular cambiaron su modo de arranque.

Todas estas características han hecho que este framework permita a los programadores trabajar con JavaScript de forma que se puedan resolver problemas del desarrollo que se presentan en el lado del cliente.

2.1.1.5 Single Page Application SPA

De acuerdo a lo que manifestaron Ollivier & Gury (2016) Single Page Application SPA o aplicación de página única ofrece una aplicación web en base a códigos de aplicaciones robustas o de tiendas online. Se diferencia de un sitio web es en su estructura y la correlación que exista entre el servidor y el navegador web.

Se compone de una sola página, la navegación del usuario es en la propia página por intermedio de algunas vistas. El navegador es el encargado de la gestión la navegación entre las vistas y reduciéndolas, ya que el trío JavaScript-HTML-CSS lo permiten, además de recuperar la información y la unión de la misma después para la generación de la página a mostrar. El usuario tiene una mejor experiencia ya que la parte gráfica está en el cliente y la recarga de la página están limitadas a lo más mínimo (Guérin, 2014; Ollivier & Gury, 2016).

La tarea del servidor es presentar los datos, en similares condiciones como lo hacen las API web, además de ofrecer todos los recursos de la aplicación, aunque el servidor no envíe las páginas que se encuentren preparadas (Ollivier & Gury, 2016).

2.1.1.6 Bootstrap

Bootstrap es un entorno o marco de trabajo para el diseño de páginas o sitios web, utilizado para ofrecer al usuario un estilo de navegación sencillo, de forma que la interacción del usuario con el sitio sea fácil y orientada desde ordenadores hasta dispositivos móviles. Esta forma de visualización en los distintos ordenadores y a la vez en dispositivos se conoce como técnica de diseño de sitios web responsive.

Según el sitio Adobe (2017)

Bootstrap es un framework popular gratuito de HTML, CSS y JavaScript que se usa para desarrollar sitios web interactivos para dispositivos móviles (...). El framework incluye plantillas interactivas para CSS y HTML de botones, tablas, navegación, carruseles de imágenes (carousel) y otros elementos que puede usar en su página web. Además, incorpora varios plug-in de JavaScript que facilitan el desarrollo de sitios web interactivos increíbles... (párr. 1).

De acuerdo a Pavón Mestras (2013) Bootstrap es el framework de Twitter utilizado para desarrollar aplicaciones web, ligero y sencillo, “basado en los últimos estándares de desarrollo web HTML5, CSS3 y JavaScript/JQuery, plugins de jQuery para validar entrada de datos, visualización tablas, grafos, etc.” (p. 2). Tiene compatibilidad con los navegadores, diseño web responsive para visualizarse en dispositivos (móvil, pc, Tablet)

Entre sus características se señalan:

- Se pueden desarrollar interfaces adaptables a los navegadores tomando en cuenta las resoluciones y escalas de los dispositivos;
- Integración a librerías de Javascript;
- Diseño sólido por el uso de LESS y CSS3/HTML5;
- Framework de fácil integración por ser ligero;
- Variedad de layout previamente definidos
- Diseño responsive
- Documentación extensa y detallada (JaGonzalez.org, 2014; T. Rodríguez, 2012).

De lo antes anotado, se puede decir que este framework de trabajo, al ser una técnica responsive, le permite al desarrollador realizar su trabajo con mayor rapidez en el momento de crear el sitio web, por cuanto ya está creado el marco de trabajo y sólo se deberá adaptar a las necesidades del usuario.

2.1.1.7 Node.js

Node.js es un framework para la creación de sitios web, que se ejecuta del lado del servidor, basado en JavaScript; y se incluye también la posibilidad de la creación de servidores web.

Una descripción de este entorno de trabajo es que “Node.js es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente” (Fundación Node.js, 2017, párr. 1). El ambiente de paquetes de este entorno es el de mayor tamaño en cuanto a librerías open source, para creación de aplicaciones web escalables y rápidas.

Al decir de Laguna (2013) este framework “es un sistema del lado del servidor, para (...) crear aplicaciones web e incluso servidores web que responden a peticiones. Está creado sobre el motor JavaScript que lleva Chrome (V8), lo cual lo hace realmente rápido” (pp. iii–iv). Es adecuado para el desarrollo de aplicaciones que hagan uso de gran cantidad de información que se encuentra en tiempo real, ejecutadas indistintamente en todo tipo de dispositivos.

Como el significado de este framework lo indica, es de mucha utilidad en el momento de presentar información en tiempo real en distintos dispositivos, lo cual apoya el marco de trabajo del sistema para la automatización de las encuestas para los proyectos de vinculación de la Facultad de Ingeniería, por cuanto en el trabajo de campo que lleven a cabo los estudiantes involucrados, necesitarán de una aplicación web escalable y rápida, la cual estará basada en Node.js.

2.1.1.8 Express

De acuerdo a Laguna (2013) es un framework que sirve para la organización de la aplicación que se desarrolla, bajo el modelo de arquitectura modelo-vista-controlador, inspirado en un framework de Ruby (Sinatra). Según el mismo autor, la particularidad de mayor importancia en Express “es la

versatilidad que te ofrece a la hora de gestionar las rutas” (p. 73) de la aplicación que se desarrolle.

Otra definición de Express señala que es “una infraestructura de aplicaciones web Node.js mínima y flexible que proporciona un conjunto sólido de características para las aplicaciones web y móviles” (Express, s.f.).

La antes señalada fuente (Express, s.f.) menciona que “con miles de métodos de programa de utilidad HTTP y middleware a su disposición, la creación de una API sólida es rápida y sencilla” (Express, s.f., párrs. 2-3). Ofrece características básicas de aplicación web, que tienen características de Node.js.

Este framework es bastante sólido, puesto que con el tiempo han ido mejorándose sus versiones. A pesar de estas características, un aspecto negativo es su documentación (Laguna, 2013, p. 73).

Por lo que se analiza en las antes señaladas descripciones de Express, se puede decir que este framework, como complemento de Node.js, ofrecerá a la aplicación de automatización de encuestas, mayor ligereza cuando se gestionen las rutas de navegación en la aplicación.

2.1.1.9 JavaScript

Aunque JavaScript no es un lenguaje de programación propiamente dicho sino de rutinas (scripts), si es una programación ligera (scripting), que se lo puede interpretar por la gran mayoría de los navegadores web. Se lo utiliza para agregar funciones adicionales y efectos a las páginas web, en lenguaje HTML (J. Rodríguez, 2005).

Aunque es flexible, es también bastante compacto, y se ha elaborado bastantes herramientas sobre el núcleo de este lenguaje, permitiendo el desbloqueo funcionalidades adicionales con el menor esfuerzo posible. Entre estas funcionalidades se encuentran:

- APIs elaboradas dentro de cada navegador prácticas y de utilidad para la creación de contenido HTML, determinar estilo CSS, captura y

manipulación de video desde cámaras web en tiempo real de los usuarios, hasta la generación de gráficos 3D y modelos de sonidos;

- APIs de terceros, en donde los desarrolladores pueden adicionar mayor cantidad de funcionalidades de distintos proveedores de contenidos, en sus sitios.
- Frameworks y librerías de terceros aplicables al HTML del desarrollador, para la construcción y publicación rápida de aplicaciones y sitios web (MDN web docs, 2017).

2.1.1.10 TypeScript

TypeScript, de Microsoft, es un conjunto tipado para JavaScript, para compilar JavaScript nativo, un lenguaje de programación que ofrece elementos de la programación orientada a objetos POO. Proporciona herramientas de utilidad para aplicaciones robustas como son herencia, clases, módulos entre otros (Basu, 2013; Fernández Guerra, 2016).

Cuando se compila TypeScript se convierte ese código a JavaScript. La plataforma o navegador que sirva para ejecutar JavaScript no se entera de que el código originalmente fue desarrollado en TypeScript, ya que lo que se ejecuta el código JavaScript que resulta de compilar TypeScript (Fernández Guerra, 2016).

En consecuencia TypeScript ejecuta básicamente dos códigos: se desarrolla en TypeScript y ejecuta en JavaScript, y ofrece un conjunto de herramientas para la POO, con el consiguiente beneficio para el desarrollo de aplicaciones.

2.1.1.11 Ionic Framework

Ionic es un potente Framework front-end de Apache Cordova/Phonegap, utilizado en el desarrollo de aplicaciones para dispositivos móviles nativas que tienen alto rendimiento. Desarrollada para todo tipo de usuarios, incluso para los que desconozcan la navegación con app móviles.

Este framework puede desarrollar cualquier proyecto presentando el aspecto y la interfaz de una aplicación móvil. Trabaja muy bien con Bootstrap, ya que

tienen muchas similitudes; además ofrece un catálogo para animaciones con el fin de dar presentar elementos en forma sencilla (Programacion.net, 2017). También funciona adecuadamente con AngularJS por el uso de sus extensiones. Continuando con el modelo vista-controlador, este framework tiene controladores para la gestión de la vista con interacciones de la interfaz (Programacion.net, 2017).

Es importante la utilización de este framework ya que una particularidad de este proyecto, es su acceso a través de dispositivos móviles, por cuanto el trabajo de campo que se debe realizar para la toma de encuestas en los proyectos de vinculación será a través de estos dispositivos para la optimización del proceso.

Ionic tiene:

- Su propio paquete de íconos;
- Función de arrastre;
- Extensiones de AngularJS;
- Soporte de gestos;
- Transiciones automáticas;
- Set de componentes (Programacion.net, 2017).

2.1.1.12 MySQL

Es una base de datos open source, de mayor popularidad entre la comunidad de desarrolladores. Por su alto rendimiento, confiabilidad y fácil uso, es la base líder para desarrollos web; también es elegida como una base integrada (Oracle, s.f.).

Entre las razones que influyen para su elección para desarrollo, se encuentran las siguientes:

- Costo gratuito;
- Asistencia a través de una comunidad en la web y también con contratos de asistencia (precios asequibles);
- Más rápida que las otras bases de datos;

- Disponibilidad de funciones: compatibilidad con ASQ ANSI y con ACID, funciones SSL, integración con entornos de programación. Su actualización es más rápida que las demás bases de datos;
- En cuanto a su portabilidad, esta base tiene compatibilidad con gran parte de los sistemas operativos conocidos y, por lo general, la transferencia de los datos se realiza fácilmente entre sistemas;
- De fácil uso y administración por sus potentes y flexibles herramientas (Gilfillan, 2003).

La razón principal de la elección de esta herramienta de base de datos para el desarrollo de este proyecto para los proyectos de vinculación de la UCSG aplicados a la Facultad de Ingeniería, es por carácter gratuito y la versatilidad de sus funciones.

2.1.1.13 MySQL Workbench

Es una herramienta de desarrollo gráfico para modelado de diagramas entidad-relación para bases de datos y servidores MySQL, compatible con versiones posteriores de MySQL 5.5. La funcionalidad de MySQL Workbench se basa en:

- Desarrollo SQL: para creación y administración de conexiones a servidores de bases de datos, en donde se ejecutan consultas SQL con el uso del editor SQL que se encuentra incorporado;
- Se puede sincronizar el modelo que se está creando, con la base de datos real;
- Diseño o modelado de datos: para la creación del modelo de la base de datos y editar la base con el uso del editor de tablas, el mismo que facilita la edición de tablas, columnas, índices, disparadores, particiones, opciones, inserciones y privilegios, rutinas y vistas;
- Documentación de una base de datos que ya existe o migrar una base totalmente;
- Creación del script de creación de la base del esquema;
- Administración del servidor: para la administración de instancias del servidor MySQL con la administración de los usuarios, realización de

copia de seguridad y recuperación de datos, inspección de información de datos para auditoría, analizar el estado de la base de datos y controlar el rendimiento del servidor MySQL;

- Migración de datos: para la migración desde varias bases de datos a MySQL. La migración además es compatible de versiones anteriores de MySQL a las últimas versiones;
- Compatibilidad con MySQL Enterprise (MySQL, 2017; Uptodown Technologies, 2017).

Existen dos ediciones de esta base de datos: Community Edition, que es gratuita y la Commercial Edition, que ofrece características adicionales a las que ya posee. Para este proyecto se utiliza la Community Edition, ya que soporta la base de datos.

2.1.2 Marco legal

La vinculación en la UCSG se la entiende las acciones emprendidas por la Universidad para ofrecer a la comunidad y sobre todo a los sectores menos favorecidos por la sociedad, el conjunto de conocimientos que en esta institución de educación superior se imparten, con el fin de aportar al progreso de la calidad de vida de dichas comunidades (UCSG, 2017a).

La política del Subsistema de Vinculación tiene como política “propiciar, promover, sistematizar y conectar la gestión que se estructura al interior de la UCSG, (...) con (...): Ministerios, Gobiernos Autónomos Descentralizados (GAD), barrios y comunas; organizaciones de fin social no gubernamentales, gremios e instituciones de educación superior nacionales e internacionales” (UCSG, 2017c).



Figura 10: Vinculación de la UCSG. Tomado de “¿Cómo se vincula la UCSG?”, por UCSG (2017a)

La vinculación de la UCSG se la realiza a través del ofrecimiento y posterior cumplimiento de proyectos, programas y actividades de vinculación resultantes de una aproximación con los grupos o comunidades objetivo, en las cuales se aspira a construir lazos de confianza y formar líderes comunitarios que puedan orientar a sus comunidades en la participación y elaboración de procesos de mejora (UCSG, 2017a).

Las normativas en las cuales se sustentan los proyectos de vinculación se encuentran en el **artículo 82** del Reglamento de Régimen Académico Codificado y el **artículo 43** del Estatuto Universitario UCSG (UCSG, 2017b):

Artículo 82.- Vinculación con la sociedad y educación continua.- *La vinculación con la sociedad hace referencia a los programas de educación continua, investigación y desarrollo, y gestión académica, en tanto respondan a través de proyectos específicos a las necesidades del desarrollo local, regional y nacional.*

Las instituciones de educación superior deberán crear obligatoriamente instancias institucionales específicas para planificar y coordinar la vinculación con la sociedad, a fin de generar proyectos de interés público (Consejo de Educación Superior, 2013, p. 32)

Artículo 43.- Son deberes y competencias del Vicerrector o Vicerrectora de Vinculación, las que a continuación se describen:

- b) *Planificar, asesorar, organizar y coordinar la realización de proyectos y actividades (...) como cursos de vinculación con la colectividad, que serán guiados por el personal académico...*
- e) *Proponer al Rectorado la suscripción de Convenios para mantener vinculaciones con la colectividad, incluso de naturaleza académica, tecnológica, de artes y pedagógica.*
- f) *Facilitar la movilidad docente y estudiantil de grado y posgrado, y la relación en el desarrollo de sus actividades académicas, culturales, de investigación y de vinculación con la colectividad.*
- g) *Gestionar la interrelación entre la Universidad, las Facultades y las Carreras, con la colectividad, y de éstas con aquellas....*
- h) *Colaborar, con las Carreras, en la realización de los programas, seminarios, conferencias y cursos extracurriculares para la colectividad en general...*
- m) *Preparar, en coordinación con los Vicerrectorados, y poner a consideración del Rectorado, convenios referidos a vinculación con la colectividad, en materia educativa, cultural y científica, con otras Instituciones de Educación Superior nacionales o extranjeras, incluyendo a los sectores productivos y sociales... (UCSG, 2016).*

CAPÍTULO III: METODOLOGÍA

El desarrollo de este proyecto requirió el planteamiento de la metodología de la investigación y la metodología de desarrollo del sistema. Estos temas se tratan en los párrafos siguientes.

3.1 Tipo de investigación

Para la aplicación de la metodología de la investigación, se requiere detallar el enfoque metodológico, el paradigma de investigación y el tipo de metodología de desarrollo a utilizar para el sistema. En los párrafos siguientes se especifica cada uno de estos temas.

3.1.1 Enfoque metodológico

Realizar una investigación dentro de cualquier ámbito científico tiene como finalidad de originar conocimientos en los investigadores mediante la solución de un determinado problema, planteado al inicio de cada proyecto, por lo que la metodología de la investigación toma como referencia a todo el proceso que el investigador debe realizar para la solución del problema.

Parte de este proceso es la definición de los diseños de investigación que, al decir de Ugalde Binda & Balbastre-Benavent (2013) que se mueven alrededor de dos modelos: a) cuantitativo o racionalista, y b) cualitativo o naturalista.

Para Hernández Sampieri, Fernández Collado, & Baptista Lucio (2014) los enfoques de la investigación son: cuantitativo, cualitativo y mixto (cuantitativo-cualitativo). El enfoque metodológico adecuado para este estudio fue el enfoque cualitativo.

El *enfoque cualitativo* de la investigación “utiliza la recolección y análisis de los datos para afinar las preguntas de investigación o revelar nuevas interrogantes en el proceso de interpretación” (Hernández Sampieri et al., 2014, p. 7), teniendo en consideración que los datos cualitativos que se buscan en un estudio es una “información simbólica verbal, audiovisual o en forma de texto e imágenes” (p. 9).

Se utilizó este enfoque metodológico debido a que no se requiere de una medición numérica de una muestra determinada, ya que el sistema de automatización de encuesta fue desarrollado para la Facultad de Ingeniería y sus proyectos de vinculación con la sociedad, como instrumento para facilitar el levantamiento de información en los lugares en donde los estudiantes involucrados en los mencionados proyectos de vinculación deban trasladarse y recoger los datos necesarios de una población objetivo.

3.1.2 Paradigmas de investigación en Ingeniería de Software

Según manifestaron Glass, Vessey, & Ramesh (2002), existen tres paradigmas de la ingeniería de software: a) evaluativos, b) formulativos, c) descriptivos, que se sustentan en tres supuestos: a) supuesto ontológico, b) supuesto epistemológico, c) supuesto metodológico.

El desarrollo del sistema de automatización de encuestas para los proyectos de vinculación de la UCSG a desarrollarse en la Facultad de Ingeniería responde al paradigma formulativo, el mismo que, de acuerdo al supuesto ontológico, el software es un producto que se crea, sustentado en los requerimientos de los usuarios finales. En cuanto al supuesto epistemológico, el producto del software resulta de la creatividad del desarrollador, que deberá la metodología y el modelo necesario para que el software sea productivo; y, sobre el supuesto metodológico, se requiere de la elección de la metodología de desarrollo adecuada para la programación del sistema.

Además, el paradigma formulativo se divide en a) estándares/directrices de desarrollo, b) metodologías de desarrollo de software, c) modelos de desarrollo (Glass et al., 2002).

Dentro de las metodologías de desarrollo se encuentran las tradicionales y las ágiles, de las cuales se eligió las últimas por ser la importancia que le otorgan al producto final y al desarrollador, antes que a trámites de contratos y documentación extensa.

3.1.2.1 Metodologías de desarrollo ágil

Una metodología de desarrollo ágil es aquella que se caracteriza por su adaptación a los cambios constantes en el desarrollo de proyectos, con la finalidad de mejorar y ampliar el éxito del mismo (Pérez Esteso, s.f.).

Pérez Esteso (s.f.) sintetiza los principios de las metodologías ágiles en los siguientes puntos:

- Se da mayor prioridad al grupo de desarrollo y las interacciones que tienen entre ellos, que a las herramientas y procesos;
- Se da prioridad al software que a una consistente documentación;
- Es importante la participación de los clientes antes que al contrato de servicios del desarrollo;
- Adaptación a los cambios; no se requiere el seguimiento de una planificación realizada para el desarrollo.

Las metodologías ágiles son utilizadas para la solución de problemas que se presenten durante el desarrollo del sistema, por cambios a las necesidades de los clientes, que, en determinados momentos se hacen frecuentes y urgentes, de modo que el producto resultante pueda tener la calidad con la que ha sido solicitada (Orjuela Duarte & Rojas, 2008). En la tabla 1 se realiza una comparación entre las metodologías de desarrollo ágil y las metodologías tradicionales.

Tabla 1: Comparación de las metodologías tradicionales y las ágiles

METODOLOGÍAS AGILES	METODOLOGÍAS TRADICIONALES
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo.
Preparados para cambios durante el proyecto.	Cierta resistencia a los cambios.
Reglas de trabajo impuestas internamente (por el equipo).	Reglas de trabajo impuestas externamente.
Proceso menos controlado, con pocos principios.	Proceso mucho más controlado, con numerosas políticas/normas.
Flexibilidad en los contratos debido a la respuesta a cambios.	Existe un contrato prefijado.
El cliente es parte del equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones en determinadas etapas del proceso.
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio.	Grupos grandes y posiblemente distribuidos trabajando en diferentes tareas.
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos.

Nota fuente: Tomado de Orjuela Duarte & Rojas (2008) "Las metodologías de desarrollo ágil como una oportunidad para la ingeniería del software educativo" (p. 160)

Al decir de Pérez Esteso (s.f) los principios de las metodologías ágiles se pueden resumir y agrupar en cuatro categorías: a) retroalimentación, b) proceso continuo en lugar de por bloques, c) propiedad intelectual compartida, d) entendimiento compartido (párr. 5)

Para Orjuela Duarte & Rojas (2008), las metodologías de desarrollo ágil de mayor utilización son: Programación Extrema, Crystal y Scrum. Para la realización de este proyecto se utilizó Programación Extrema, por las características que esta metodología presenta en el momento del desarrollo, y que se presentan en los párrafos siguientes.

3.1.2.1.1 Programación Extrema eXtreme Programming XP

Según Pérez Esteso (s.f.) *eXtreme Programming XP* esta metodología de desarrollo ágil tiene como finalidad ampliar la productividad en el momento del desarrollo del proyecto de software. Prioriza las actividades que conducen a resultados eficientes y minimiza las demoras que se presenten en el entorno del desarrollo.

Programación Extrema es una metodología que da prioridad en favorecer las relaciones en el equipo de trabajo, favoreciendo el trabajo en equipo, al mismo tiempo que se preocupa por la capacitación de los desarrolladores en un entorno adecuado de trabajo. Como características principales tiene la

retroalimentación permanente usuario-equipo de trabajo, comunicación entre los miembros del equipo de desarrollo, soluciones simples de implementación, valentía para hacer frente a posibles cambios de requerimientos por parte de los clientes. Utilizada para proyectos en donde los requisitos y necesidades no están completamente definidos y precisan de cambios constantes (Orjuela Duarte & Rojas, 2008).

La selección de esta metodología para este proyecto de implementación se la determinó por su característica de la retroalimentación constante entre el usuario final y el desarrollador, y por las soluciones simples que ofrece.

3.2 Diseño de investigación

Existen algunos diseños de investigación. De acuerdo a lo manifestado por Arias (2006) el diseño de una investigación corresponde a las técnicas que el investigador aplica para solucionar el problema motivo de investigación, por lo que es importante elegir el que aplique al estudio y obtener los resultados esperados.

Entre estos, el antes señalado autor (Arias, 2006) señaló que existen: a) investigación documental o diseño documental, b) investigación de campo o diseño de campo, c) investigación experimental o diseño experimental, d) investigación no experimental o diseño no experimental (Hernández Sampieri et al., 2014).

Para este proyecto se escogió la investigación no experimental o diseño no experimental, que se define en los párrafos siguientes.

3.2.1 Investigación no experimental

De acuerdo a lo manifestado por Hernández Sampieri et al., (2014) y Salinas (2010) la *investigación no experimental* se refiere a la no manipulación intencionada de variables de tipo independiente, sino que se realiza la observación de la evolución los hechos o grupos que suceden, en su ambiente natural, con el fin de analizar lo que en ellos sucede.

En los estudios no experimentales se realiza la observación del entorno preexistente sin que haya existido ningún tipo de manipulación por parte del

investigador. En estos estudios no se pueden manipular las variables independientes, en la cuales no se tiene control sobre ellas ni influenciar sobre ellas, puesto que ya ocurrieron, lo mismo que sus consecuencias (Hernández Sampieri et al., 2014).

Se escogió este diseño de investigación por cuanto se va a desarrollar un sistema en un entorno ya existente, que la necesidad de levantar información para los proyectos de vinculación a través de las encuestas que se realicen a los grupos objetivo requiere de que se brinde a los participantes en los proyectos, de herramientas que faciliten su trabajo.

Por lo tanto, este diseño de investigación no realiza ninguna modificación en lo que ya existe, que serían las encuestas a realizar, puesto que la planificación de encuestas manuales para los proyectos de vinculación ya se ha realizado y sigue realizándose sino que propone mejorar la propia estructura de la encuesta, con la misma información que se necesita levantar, sólo que facilitaría el trabajo del encuestador.

Existen algunos *tipos* de diseño no experimental. De acuerdo a Hernández Sampieri et al., (2014) se clasifican “por su dimensión temporal o el número de momentos o puntos en el tiempo en los cuales se recolectan datos” (p. 154), teniendo en consideración que se pueden presentar casos en que la investigación trate otros temas, como el nivel de una o más variables en determinado momento, la evaluación del hecho en determinado espacio de tiempo, o establecer el tipo de relación que existe entre las distintas variables en cierto momento, para lo que se determinaría un diseño transeccional o transversal con alcance descriptivo, exploratorio, explicativo o correlacional.

En otras ocasiones las investigaciones se pueden orientar en el estudio de la evolución de una o grupo de variables y sus relaciones, el análisis de las modificaciones que puede sufrir un proceso, evento al pasar el tiempo, por lo que el diseño será longitudinal (Hernández Sampieri et al., 2014).



Figura 11: Diseños de la investigación no experimental. Tomado de “Metodología de la Investigación”, por Hernández Sampieri et al., (2014). México: McGRAW-HILL / INTERAMERICANA EDITORES, S.A. DE C.V.

Para la selección del tipo de diseño no experimental se tomó en cuenta la evaluación del hecho en determinado espacio de tiempo, que es uno de los factores para su clasificación, de manera que el *diseño transeccional* fue el adecuado para la realización de este proyecto, ya que la investigación del problema se llevó a cabo en su propio lugar, es decir en la Facultad de Ingeniería. Su definición se puede revisar en el apartado siguiente.

3.2.1.1 Diseño transversal o transeccional

Al decir de Hernández Sampieri et al., (2014) los *diseños transversales o transeccionales* son estudios que sirven para reunir información en determinado momento, que es único, con el fin de puntualizar las variables y realizar el análisis correspondiente de cuál es su incidencia y relación que tienen entre ellas en un instante específico, y pueden ser descriptivos, exploratorios, y correlacionales-causales.

El estudio en cuestión es transeccional porque se va a tomar la opinión que tienen las tres personas a entrevistar (un delegado de Vinculación por las carreras de Computación e Ingeniería en Sistemas Computacionales de la Facultad de Ingeniería, para la planificación de los proyectos a realizarse entre los estudiantes y los grupos objetivos; a la Coordinadora de la Unidad de Titulación y docente a tiempo completo para conocer algunos requisitos que debería tener la aplicación; y, a un experto sicólogo de la UCSG, del departamento Comisión de Evaluación Interna CEI, sobre las características que debe tener una encuesta) para el definir detalles sobre la necesidad de desarrollar el sistema de automatización de encuesta para los proyectos de vinculación de la UCSG en la Facultad de Ingeniería.

El modelo de diseño transversal utilizado en esta investigación es el *diseño transeccional descriptivo* que son diseños que “indagan la incidencia de las modalidades, categorías o niveles de una o más variables en una población, son estudios puramente descriptivos” (Hernández Sampieri et al., 2014, p. 155). Al ser estudios descriptivos, lo que se busca es describir las variables (una o varias) de los objetos, grupos, contextos, etc.; puede darse el caso de que el investigador quiera detallar aspectos comparativos entre personas, grupos, comunidades, entre otros.

3.3 Técnicas de recolección de información

Para la recolección de datos en proyectos existen algunas técnicas que las utiliza el investigador para realizar su trabajo. De acuerdo al tipo de investigación se usan unas u otras.

Al decir de Muñoz Giraldo et al., (2001) citados por Bernal (2010), para el enfoque metodológico cualitativo existen algunas técnicas que puede utilizar el investigador para el levantamiento de información tales como: a) entrevista estructurada y no estructurada, b) observación sistemática y no sistemática, c) notas de campo, d) diarios, e) archivos, f) grabaciones en audio y video, entre otras.

Para la recolección de datos que sustenta el proyecto se utilizó la entrevista, ya que el sistema está destinado para un determinado fin, que es la de servir de herramienta para proyectos de vinculación, y las entrevistas realizadas fueron dirigidas hacia el delegado de Vinculación por las carreras de Computación e Ingeniería en Sistemas Computacionales de la Facultad de Ingeniería y comprendía aspectos técnicos de la aplicación; la segunda entrevista fue hecha a la Coordinadora de la Unidad de Titulación y docente a tiempo completo para conocer otros requisitos funcionales que debería tener el sistema; y, la tercera entrevista se la aplicó a un experto sicólogo de la UCSG, del departamento Comisión de Evaluación Interna CEI, sobre las características de una encuesta.

A continuación se plasma el concepto de entrevista y el tipo de entrevista utilizada para el proyecto.

3.3.1 Entrevista

Se conoce como entrevista, según lo manifestó Niño Rojas (2011, pp. 64–65) la “técnica, fundamentalmente de tipo oral, basada en preguntas y respuestas entre investigador y participantes, que permite recoger las opiniones y puntos de vista de dichos participantes o, eventualmente, según objetivos, intercambiar con ellos en algún campo” (p. 64). De este modo, el investigador puede descubrir características de los participantes para tener una idea clara de lo que necesitan para solucionar el problema que es motivo de estudio, en este caso, para el sistema de automatización de encuestas para los proyectos de vinculación de la Facultad de Ingeniería.

Esta técnica la utilizan investigadores o profesionales dedicados a las ciencias sociales en proyectos de corte cualitativo con la finalidad de conseguir datos que los ayude en la solución de problemas, pero además para tener cierta influencia en el comportamiento de otros individuos si es que se diera el caso (Niño Rojas, 2011).

Según Arias (2006) se conocen algunos tipos de entrevista, entre éstos se encuentran: a) estructurada o formal, b) no estructurada o informal, y, c) entrevista semi estructurada (pp. 73–74). La entrevista utilizada en la investigación fue la semi estructurada.

3.3.1.1 Entrevista estructurada

La entrevista estructurada contiene un conjunto de preguntas que el entrevistador va a realizar a la persona objetivo. Estas preguntas o cuestionario de preguntas podrán servir para que el entrevistador registre las respuestas dadas por el entrevistado y se lo puede considerar como un instrumento; además, como ayuda, también se puede utilizar una grabadora o videocámara (Arias, 2006, p. 73).

3.4 Instrumentos de recolección de información

El instrumento de recolección de datos de la técnica de la entrevista, que se utilizó en el levantamiento de información de este proyecto fue el

cuestionario y el grabador de sonido. En el párrafo siguiente se define el significado de cuestionario.

3.4.1 Cuestionario

Hernández Sampieri et al., (2014) manifestaron que el cuestionario es un “conjunto de preguntas respecto de una o más variables que se van a medir” (p. 217), constituyéndose posiblemente en el instrumento de mayor de mayor utilización para la recolección de datos. El cuestionario es un grupo de preguntas previamente elaboradas que se aplica a un entrevistado con el fin de medir variables y pueden aplicarse en varios campos de la ciencia.

Se elaboraron tres cuestionarios de preguntas para las entrevistas que se realizaron; el primero se lo aplicó al delegado de Vinculación por las carreras de Computación e Ingeniería en Sistemas Computacionales de la Facultad de Ingeniería para la planificación de los proyectos a realizarse entre los estudiantes y los grupos objetivos y comprendía aspectos técnicos de la aplicación; el segundo, a la Coordinadora de la Unidad de Titulación y docente a tiempo completo para conocer algunos requisitos que debería tener la aplicación; y, el tercer cuestionario se lo aplicó a un experto sicólogo de la UCSG, del departamento Comisión de Evaluación Interna CEI, sobre las características que debe tener una encuesta.

Los cuestionarios que se aplicaron para el levantamiento de los requerimientos se encuentran en la sección Anexos.

3.5 Análisis de resultados

Los resultados de las entrevistas se expresan en los apartados que siguen a continuación.

3.5.1 Entrevista 1

De la entrevista realizada al delegado de Vinculación por las carreras de Computación e Ingeniería en Sistemas Computacionales de la Facultad de Ingeniería, se determinaron los requerimientos funcionales de la aplicación:

- Para los proyectos a realizarse en la Facultad de Ingeniería, se necesita de un sistema de encuesta que registre algunos tipos de encuesta; podrían ser encuestas para alfabetización digital, conocimiento de office, entre otros, de acuerdo a la necesidad de la información que se quiera levantar.
- La aplicación deberá generar grupo de preguntas, tanto de tipo general (nombre, edad, sexo) y técnicas (uso de temas tecnológicos). Además, la aplicación deberá condicionar la respuesta a cualquier tipo de preguntas y finalizar las preguntas condicionadas.
- Se tomarán en consideración todos los tipos de preguntas que puede tener una encuesta (solo está habilitado opción múltiple, check, respuesta corta), por ejemplo: si la opción es otro/otra/otros/otras agregar un campo para escritura, tipo tabla donde cada ítem pueda ser valorado con opciones de calificación; todo esto en base a un modelo entregado por el docente. La aplicación deberá crear la encuesta de acuerdo a la necesidad, basándose en esos tipos de respuestas; además considerar numeración de las preguntas.
- La aplicación permitirá la creación de un único usuario administrador, que asigne roles a los estudiantes entrevistadores que formarán parte del proyecto, con un nombre, apellido y contraseña, más no por correo electrónico.
- Los usuarios no tienen permiso para ingresar a la aplicación y revisar los resultados de la tabulación de las encuestas. Tienen permiso de utilizar la aplicación móvil y realizar la encuesta que el administrador le haya asignado.
- En la tabulación de los datos, informes estadísticos, uso de barras; filtrado por encuesta individual, por grupo de preguntas y filtrado general en donde se presente toda la información resultante.
- En individual se podrá ver el usuario que hizo la encuesta y el tiempo que se demoró en realizar la encuesta.
- Además, se podrá ver la georreferenciación donde se realizó la encuesta. Las encuestas deben estar asociadas a un área de aplicación, esto es que cada encuesta debe tener un polígono del

área geográfica de aplicación de la encuesta delimitada a través de parámetros.

- Cada vez que aplique una encuesta desde la aplicación móvil debe reflejarse este ingreso en el polígono asociado a la encuesta.
- En cuanto a la aplicación móvil, se cargarán los proyectos asignados a los usuarios y en esos proyectos se encontrarán las encuestas.
- Cuando se ha terminado de llenar la encuesta, al presentar el botón enviar sucederán dos cosas: si tiene internet se enviarán esos datos con su ubicación a la aplicación web en donde se seguirán agregando los resultados, si no se tiene internet, se guardarán los datos en una base de datos integral hasta que tenga internet y se enviará la información a la aplicación web (ver Anexo 1).

3.5.2 Entrevista 2

De la entrevista realizada a la Coordinadora de la Unidad de Titulación se obtuvieron los siguientes resultados:

- Debe mantenerse un estándar de idioma en cada pantalla de la aplicación;
- Debe considerarse además un formato para mayúsculas, minúsculas y tildes en cada palabra;
- Hay que considerar un control de sesiones para seguridad, es decir, que no se pueda abrir otra sesión en otro browser con la misma dirección de acceso a la aplicación;
- Para realizar un seguimiento de los estudiantes, se debe considerar a qué hora empezó la encuesta y a qué hora terminó, mostrada en una etiqueta;
- Debe determinarse cuántas personas existen por zonas;
- Módulos u opciones que permitan crear grupos de trabajo y generar zonas de trabajo;
- Mayor variedad de tipos de preguntas;
- Reporte que filtre por fecha y tiempo de realización de la encuesta;
- Reporte de la cantidad de estudiantes que fueron a realizar la encuesta;

- Debe existir una pantalla de administración de encuesta, que permita definir la fecha de inicio y fin de la misma, para establecer la inactivación de la encuesta después de un tiempo límite;
- Diseño amigable al usuario final (ver Anexo 2).

3.5.3 Entrevista 3

Sobre la entrevista al experto psicólogo del departamento Comisión de Evaluación Interna CEI, sobre las características que debe tener una encuesta, se especificaron las siguientes:

- La entrevista deberá ser anónima, para garantizar la fiabilidad de las respuestas;
- Si la encuesta está ligada a la georreferenciación, debería conocerse quién la realizó y no conocer lo que contestó;
- Se recomienda que una encuesta no sea demasiado grande. Como máximo entre 20-25-26 preguntas; a más preguntas, la encuesta se vuelve poco confiable;
- El tiempo que se debería utilizar en realizar la encuesta con el número máximo sugerido de preguntas, sería entre 10 y 12 minutos;
- El tiempo de respuesta a una encuesta también depende del tipo de pregunta que se aplique. Por ejemplo, una persona puede no conocer cuánto dinero gana y cuánto gasta, por lo que emplearía tiempo en realizar esos cálculos;
- No deben existir preguntas abiertas, porque se orientaría hacia un análisis cualitativo;
- Deberían existir preguntas enganchadas, es decir, elaborar la pregunta en la que se responda si o no y, dependiendo si la respuesta es negativa, plantear todas las posibles opciones de respuesta;
- Antes de realizar una encuesta, es conveniente planificar uno o dos grupos focales, señalando una muestra pequeña, que permita analizar preguntas que pueden ser abiertas y abarcar todas las posibles respuestas (ver Anexo 3).

3.6 Selección de las herramientas de desarrollo

El análisis de las entrevistas determinó la utilización de las herramientas de desarrollo.

Para el *desarrollo web* se utilizó un vanguardista diseño de software en el cual maneja dos conceptos frontend y backend.

Para el *frontend* se utilizó como framework principal Angular + el mismo que sirvió para la creación del HTML, CSS, Typescript y Javascript En la tabla 2 se indican las características por las cuales se escogió esta herramienta.

Tabla 2: Comparación herramientas front-end

CARACTERÍSTICAS	Angular	Angular.js	Vue.js
Modelo de trabajo	MVC	MVW	MVC
Lenguaje de programación	JavaScript	JavaScript	PHP JavaScript
Licencia	MIT	MIT	New BSD (Berkeley Software Distribution) License
Paradigma de programación	Orientado a componentes	Orientado a componentes	Eventos orientados a objetos
Control de versiones	Git Subversion	Git Subversion	Git Subversion
Página oficial	angular.org	angularjs.org	vuejs.org
Documentación	Manipula DOM de una forma compatible con diferentes navegadores	Tutoriales y videotutoriales	Fácil de aprender si está migrando desde otros frameworks. Modelo de programación simple
Simplicidad	Curva de aprendizaje bastante empinada para los desarrolladores	Mayor complejidad en el aprendizaje	Aprendizaje menos complicado. Sólo necesita un tutorial sobre la sección de la documentación
Facilidad de uso	Crear y mantiene aplicaciones web de una sola página.	Impone varias implicaciones en la estructuración	Permite estructurar la aplicación de la forma que se desee
Velocidad y rendimiento	Utiliza sistema de inyección de dependencias jerárquico impulsando su rendimiento de forma asombrosa. Detección de cambios basados en árboles unidireccionales		Biblioteca de peso ligero que permite velocidad
Sintaxis	TypeScript	JavaScript	JavaScript

Fuente: Tomado de "Vue JS vs Angular: comparación entre líder y aspirante", por Ortego (2017)

Otra de las herramientas frontend es Bootstrap, otro framework que permite crear interfaces para el diseño de las pantallas. En la siguiente tabla se indica las características por las cuales se escogió esta herramienta.

Tabla 3: **Comparación de Bootstrap**

Características	Bootstrap	Skeleton	Foundation
Licencia	MIT	MIT	MIT
Diseño	Personalizable, variedad de diseños	No es tan atractivo	No es tan atractivo
Descripción	Es el marco de HTML, CSS y JS más popular para desarrollar primeros proyectos receptivos y móviles en la web.	Es una pequeña colección de archivos CSS que pueden ayudarlo a desarrollar rápidamente sitios que se ven hermosos en cualquier tamaño, ya sea una pantalla portátil de 17 "o un iPhone.	Puede prototipar rápidamente y crear sitios o aplicaciones que funcionen en cualquier tipo de dispositivo con Foundation, que incluye construcciones de disposición (como una cuadrícula totalmente receptiva), elementos y mejores prácticas. Rejilla receptiva
Pros sobre esta herramienta	Sensibilidad	Minimalista	Móvil primero
	Componentes de interfaz de usuario	Sensible	Fuente abierta
	Consistente	Sencillo	Semántico
	Excelentes documentos	Sintaxis ligera y limpia	Personalizable
	Flexible	Ligero	Rápido para prototipo
	HTML, CSS y marco JS		Ui simple
	Fuente abierta		Rápido
	Personalizable		Mejores prácticas
	Ampliamente utilizado		Configuración fácil
Clientes	Twitter spotify hootsuite bitly AdRoll Cloudmine CircleCi Chargify	LedgerNote Spirit Pixel emotion.me Krlío CampusBox-Api	Hoteltonight Customer.io CodeBrag Custora Create.it sportsy Opower
Integración	Udash IgniteUi Durandal Pingendon FlatUI Bootswatch MateralKit	No tiene integraciones	ZurbInk Codekit Durandal GridBox
GitHub Stats (17/02/2018)	122K	15.3K	27k

Fuente: Tomado de "Bootstrap vs. Skeleton vs. Foundation", por StackShare (2018)

Para el *backend*, la herramienta que se utilizó se encuentra del lado del servidor, como lo es Node.js, la misma que se encarga de levantar el servicio e interactuar con la base de datos. A continuación, se detalla por qué se seleccionó esta herramienta.

Como quedó especificado en el Marco Teórico, Node.js es un entorno en tiempo de ejecución multiplataforma, que ayuda a levantar todo el servicio. En la tabla 4 se indica características por las cuales se escogió esta herramienta.

Tabla 4: **Comparación de Node.js**

Características	Node.js	PHP
Sintaxis	Acceso a la interfaz de línea de comandos	Acceso a la interfaz de línea de comandos
Cambios de contexto	Tiene menos cambios de contexto, ya que junto con MongoDB conforma la pila MEAN y con esta se desarrolla en un solo lenguaje: JavaScript.	Con la pila LAMP, los desarrolladores deben dominar al menos dos lenguajes (PHP y SQL), además deben conocer HTML, CSS y JavaScript.
Contribuciones	Está creciendo más rápido que cualquier otra plataforma o lenguaje, principalmente a la filosofía de mantener un mínimo de módulos y realizar sólo un pequeño conjunto de tareas. Podría en un futuro sobresalir por la cantidad de contribuciones.	Esta es probablemente una de las áreas más importantes donde PHP aún sobresale por encima de Node.js. Hay sorprendentes aplicaciones de código abierto, por ejemplo, WordPress, toneladas de scripts libres, herramientas y libros.
Framework	MVC	CakePHP, Zend, Symfony y Yii (más populares)
Servidor web	Creado desde cero para aplicaciones de red y hay una serie de módulos básicos para escribir servidores web.	Para entornos de producción, PHP no se puede ejecutar por su cuenta. Una de las tecnologías más populares usadas con PHP son Apache y Nginx donde PHP es sólo un módulo del servidor web.
Rendimiento	Es rápido debido a su mecanismo de E/S no bloqueante y la tecnología del motor Google Chrome V8.	Es relativamente rápido, pero debido a su cuello de botella en el sistema de archivos, las bases de datos y solicitudes de terceros, fracasa.

Fuente: Tomado de “Breve comparación de Node.js y PHP”, por Cuesta (2014)

Para el *desarrollo móvil* se utilizó Ionic y, como se especifica en el Marco Teórico, ésta es una herramienta de la línea open source, utilizada para el desarrollo de aplicaciones híbridas que utilizan CSS, HTML5 y JS, que en un comienzo se la creó para dispositivos móviles y tablets; utiliza Sass y perfeccionado con Angular, y tiene variedades de componentes, que permiten a los desarrolladores realizar sus trabajos más fácilmente.

Se escogió esta herramienta para el desarrollo móvil porque Ionic utiliza Angular con el fin de crear un marco más adecuado para desarrollar aplicaciones robustas y seguras. Se fundamenta en las SDK utilizadas en la programación móvil de mayor popularidad, lo que permite que cualquier desarrollador que haya hecho una aplicación para iOS y Android lo entiendan.

Como se percibe en la figura 12, comparativa de Google Trends sobre el interés entre IONIC (desarrollo de aplicaciones híbridas) y Android Studio (desarrollo de aplicaciones nativas), Ionic ha ganado terreno en su uso en el último año e inicios de 2018.

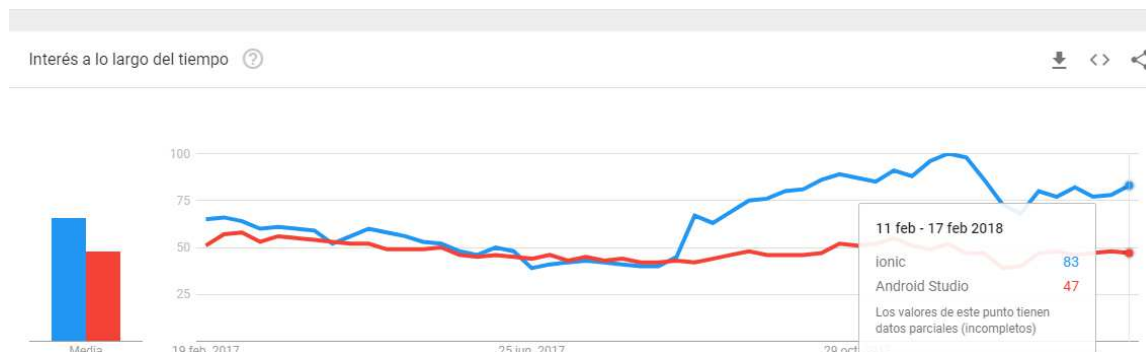


Figura 12: Comparación entre Ionic y Android Studio. Tomado de Google Trends (2018)

CAPÍTULO IV: PROPUESTA TECNOLÓGICA

Definidos los requerimientos funcionales de la aplicación para los proyectos a realizarse en la Facultad de Ingeniería, en los cuales se especificó exactamente cómo se requiere el sistema.

4.1 Primera fase: Diseño y desarrollo del proyecto

En los párrafos siguientes se presenta el diseño del proyecto, especificando la arquitectura de la aplicación, los diagramas de casos de uso con sus actores y su descripción, el modelo Entidad Relación, la descripción del diccionario de datos y el proceso del desarrollo del aplicativo web y móvil. Además, la fase de implementación del sistema.

4.1.1 Arquitectura de la solución

La arquitectura está dada por una arquitectura de tres capas. En la capa cliente se encuentra la aplicación del cliente, desarrollada en Angular; la capa de datos, con Node.js el framework para la creación de sitios web, que se ejecuta del lado del servidor; y, capa del servidor, en donde se encuentra la base de datos MySQL.

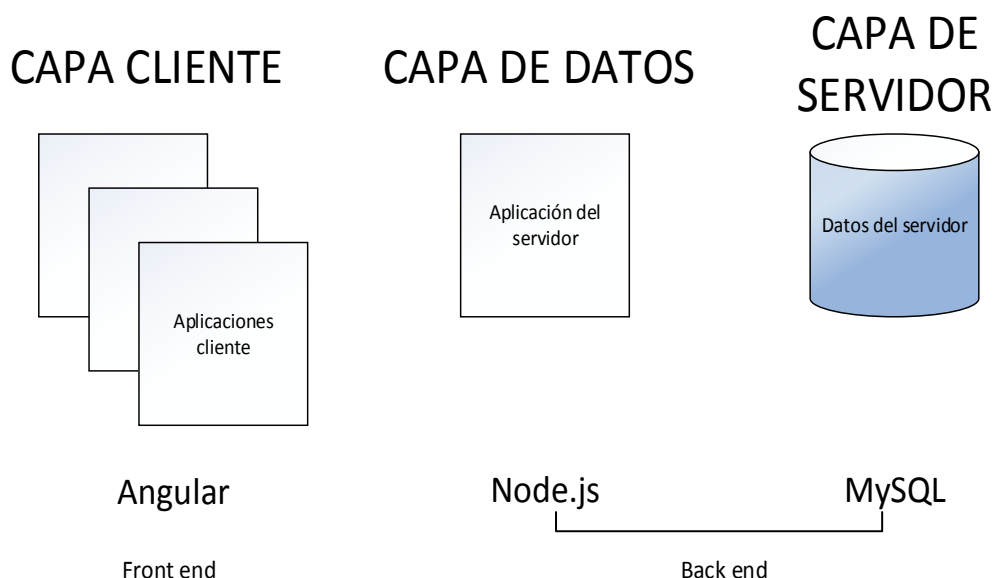


Figura 13: Arquitectura de la solución

4.1.2 Diagrama de casos de uso

Se diseñaron dos diagramas de casos de uso: uno para el sistema web y otro para el aplicativo móvil.

4.1.2.1 Diagrama de casos de uso para el sistema web

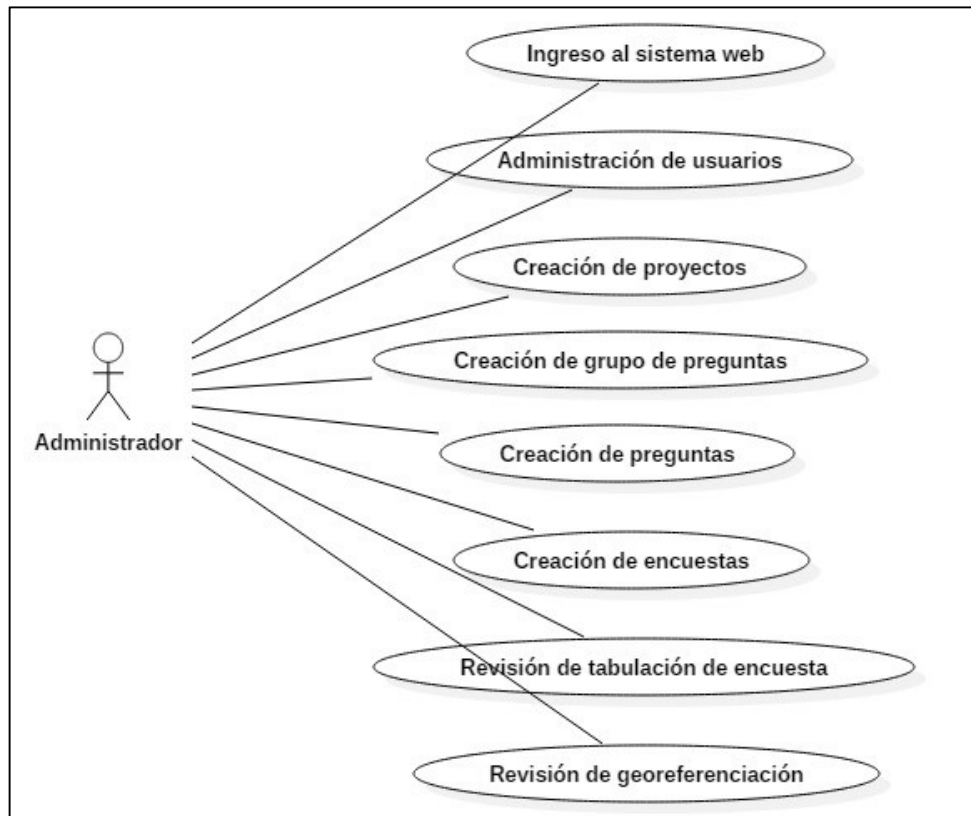


Figura 14: Diagrama de casos de uso - Sistema web

4.1.2.2 Diagrama de casos de uso para el aplicativo móvil

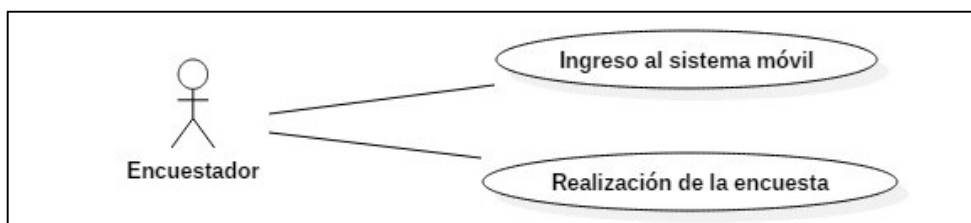


Figura 15: Diagrama de casos de uso - Aplicativo móvil

4.1.3 Actores del sistema web y aplicativo móvil

Los actores del sistema son: para el sistema web, el administrador, y para el aplicativo móvil, el encuestador.

En las tablas 5 y 6 se describen la interacción de los actores, tanto con el sistema web como con el aplicativo móvil.

Tabla 5: **Actor del sistema web**

Nombre	Administrador del sistema web
Descripción	Es el usuario del sistema web, que se encarga de la administración del sistema, es decir, crear usuarios, asigna roles de usuario (administrador-encuestador), quitar privilegios.
Tipo de actor	Primario

Tabla 6: **Actor del sistema móvil**

Nombre	Encuestador del sistema móvil
Descripción	Es el usuario del sistema móvil, que se encarga de realizar la encuesta creada en el dispositivo móvil por el usuario Administrador
Tipo de actor	Primario

4.1.4 Descripción de los casos de uso

La descripción de los casos de uso está dividida en dos partes: los casos de uso para el sistema web y para el aplicativo móvil.

4.1.4.1 Descripción de los casos de uso – Sistema Web

La descripción de cada uno de los casos de uso para el sistema web se encuentra en la sección anexos, para los cuales se elaboraron tablas que contienen información relacionada a su descripción, actores involucrados, precondiciones, flujo normal del proceso, restricciones y post condiciones (ver Anexo 1: *Entrevista 1*)

ENTREVISTA AL ENCARGADO DE VINCULACIÓN POR LAS CARRERAS DE COMPUTACIÓN E INGENIERÍA EN SISTEMAS COMPUTACIONALES DE LA FACULTAD DE INGENIERÍA

1. ¿Por qué cree usted necesaria la implementación de una aplicación de encuesta para para la realización de proyectos en la Facultad?

.....

2. Dentro de los requisitos funcionales necesarios para el desarrollo de la aplicación web ¿qué tipo de preguntas deberá considerarse en el desarrollo de la aplicación?

.....

3. ¿Qué tipos de respuestas se deberá manejar la encuesta para desarrollarla en el sistema?

.....
.....

4. ¿Qué debería tenerse en cuenta para el manejo de los posibles usuarios del sistema?

.....
.....

5. En cuanto a la interfaz del sistema ¿cuál deberá ser la secuencia de exploración?

.....
.....

6. Similar a los requisitos para el sistema web, ¿qué deberá realizar la aplicación móvil?

.....
.....

Anexo 2: *Entrevista 2*

ENTREVISTA A DOCENTE DE TIEMPO COMPLETO DE LA CARRERA DE CIENCIAS DE LA COMPUTACIÓN E INGENIERÍA EN SISTEMAS COMPUTACIONALES DE LA FACULTAD DE INGENIERÍA

1. ¿Qué requerimientos de forma debería tener la aplicación?

.....
.....

2. ¿Qué otros requerimientos funcionales considera apropiados para la aplicación?

.....
.....

3. ¿Qué tipo de preguntas no deberían ser consideradas en la encuesta?

.....
.....

**ENTREVISTA AL SICÓLOGO DEL DEPARTAMENTO COMISIÓN DE
EVALUACIÓN INTERNA CEI**

1. ¿Qué características deben tener las preguntas de la encuesta?

.....
.....

2. ¿Qué tiempo promedio debería durar una encuesta?

.....
.....

3. ¿Cuántas preguntas en promedio debería tener una encuesta?

.....
.....

4. ¿Qué tipos de preguntas no deberían ser incluidas en una encuesta?

.....
.....

5. ¿Qué consideraciones deberían tomarse en cuenta para elaborar las preguntas de la encuesta?

.....
.....

Anexo 4).

4.1.4.2 Descripción de los casos de uso – Aplicativo Móvil

Como se observa en la Figura 15, existen dos casos de uso para el aplicativo móvil. De igual forma que para los casos de uso del sistema web, la descripción se encuentra en la sección anexos (ver Anexo 5).

4.1.5 Modelo Entidad Relación

La figura 16 muestra el modelo Entidad-Relación del sistema web.

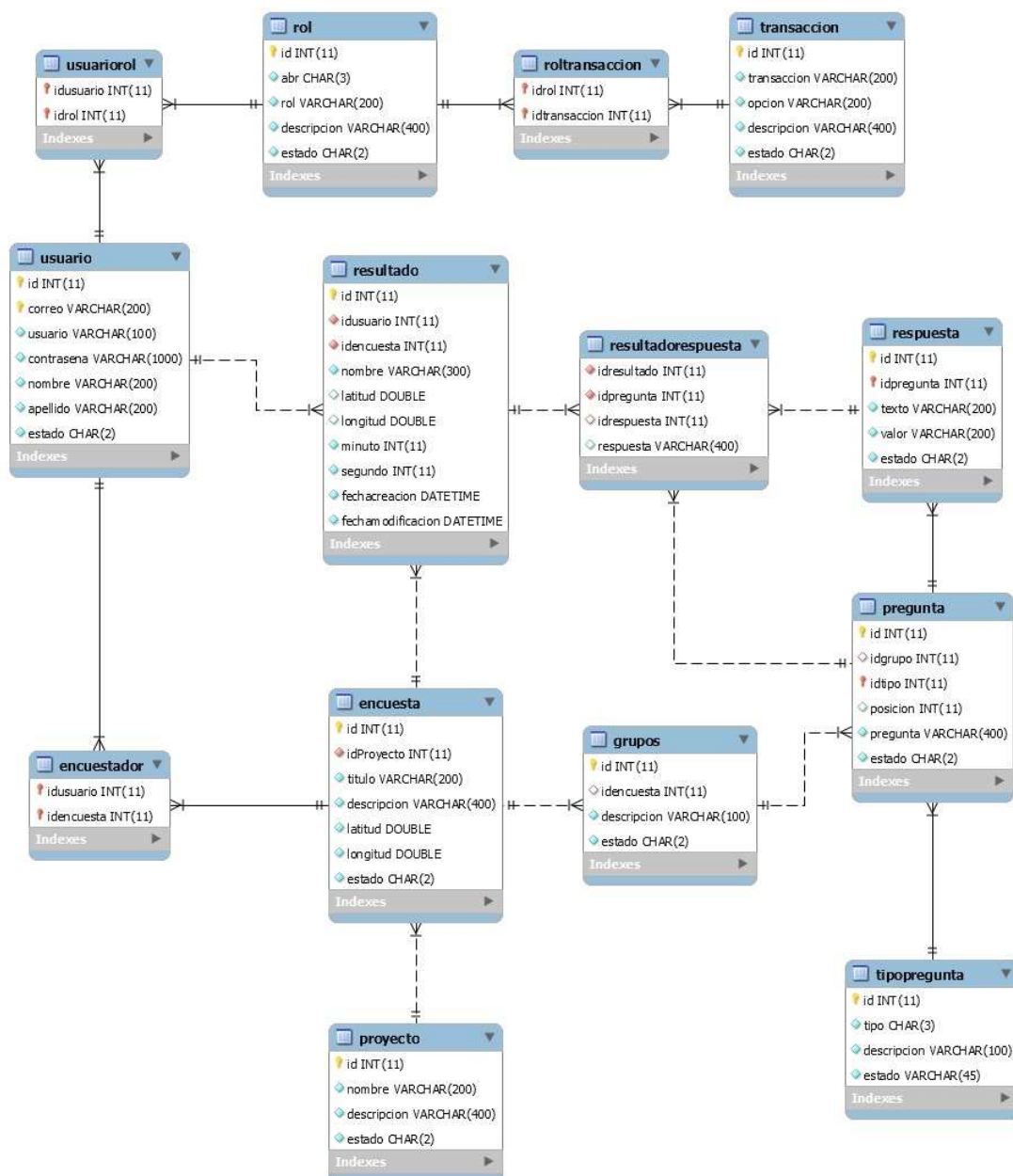


Figura 16: Modelo Entidad-Relación

4.1.6 Descripción del diccionario de datos

El diccionario de datos presenta las características de los datos utilizados en el sistema, que sirvió para la especificación de los requerimientos. Las tablas se encuentran en la sección anexos (ver Anexo 6).

4.1.7 Desarrollo del sistema web

Para el desarrollo del sistema web se necesitó de la selección de las herramientas más óptimas que permitan la entrega de un producto eficiente para cubrir los requerimientos de la Facultad de Ingeniería en cuanto a la ejecución de proyectos.

En la búsqueda de una herramienta que tenga las características antes señaladas y a través de la investigación realizada, se determinó que Angular es la herramienta de vanguardia front end, que permite un mejor rendimiento hacia los dispositivos móviles.

4.1.8 Desarrollo del sistema móvil

Para el desarrollo móvil se utilizó Ionic, que es un marco de trabajo para generar aplicaciones móviles de alto rendimiento, que funcionan en los más conocidos ambientes del mercado. Este framework es ideal para aplicaciones móviles, nativas, de alto rendimiento, y desarrollada para usuarios de todo tipo, incluidos los que no conozcan cómo se navega en app móviles.

4.2 Segunda fase: Implementación del sistema

La implementación del sistema web para automatización de encuestas para proyectos en la Facultad de Ingeniería se lo explica en los párrafos siguientes.

4.2.1 Proceso de implementación

El proceso de implementación del sistema web se lo realizó por etapas, las que se detallan en los párrafos siguientes.

Revisión del hardware y software disponibles

El sistema web va a quedar alojado en una plataforma en la nube llamada heroku, que brinda el servicio de alojamiento de aplicaciones.

Método de implementación

Heroku utiliza git, que es un sistema de control de versiones, que permite de forma más organizada, subir actualizaciones, diseñado para tener el mejor control de los cambios que se van realizando en el aplicativo.

Por medio del git se va a subir el proyecto a la nube del heroku y por este mismo medio se puede ir subiendo las actualizaciones a la nube.

Puesta en operación

La fecha prevista para la entrada en vigencia del sistema está prevista para el mes de marzo de 2018. Hasta su puesta en ejecución, se deben tener en cuenta algunos aspectos: a) capacitación al usuario final, b) interfaces finales, c) manual de usuario.

Interfaces

Se mantendrán las interfaces que se han desarrollado para el sistema, tanto para web como para móvil, luego de la implementación.

Perfiles o roles

El proyecto fue desarrollado por un solo programador, el mismo que adoptó los roles de un equipo de desarrollo: analista, programador, tester. El usuario final fue el usuario clave del proyecto, con el cual se probó el sistema para determinar si los requerimientos se cumplieron.

4.2.2 Requerimientos mínimos del sistema web y móvil

Para el aplicativo web, los requerimientos son los siguientes:

Requisitos de software

- Sistema operativo (Windows, Linux, Mac OS);
- Acceso a Internet;

- Navegador Web. Se recomienda google Chrome o Mozilla Firefox (últimas versiones).

Requisitos de hardware

Un ordenador que posea mínimamente las siguientes características:

- Mainboard Biostar J3060NH o versiones parecidas;
- Procesador Dual Core 3.00 GHz;
- RAM de 2 GB;
- Modem con servicio de internet de 10 Mbps;
- Monitor genérico, mouse, teclado.

Para el aplicativo móvil, los requerimientos son los siguientes:

Requisitos de software

Se debe contar con sistema operativo Android que maneje una versión 4.0 o superior.

Requisitos de hardware

Teléfono inteligente que soporte las versiones de los sistemas operativos indicadas en los requisitos de software y que conste, además, de:

- 1 MB de almacenamiento en la memoria del celular;
- 1 GB de memoria RAM;
- Procesador igual o superior al Cortex A7

4.2.3 Análisis costo-beneficio

El proyecto no tiene costo alguno para la Facultad, ya que el desarrollo del sistema se lo realizó con herramientas open source. Para su implementación, se utilizó un servicio de la nube (heroku) con un nivel de disponibilidad y rendimiento limitado para el alojamiento del aplicativo web, el cual cuenta con un 1 GB en la base de datos para almacenar información. Después de superar ese tamaño de espacio, la UCSG deberá gestionar el

financiamiento de mayor cantidad de espacio en la nube, para el correcto funcionamiento del aplicativo.

Los costos referenciales del servicio de la nube, se los puede analizar en la figura 17.

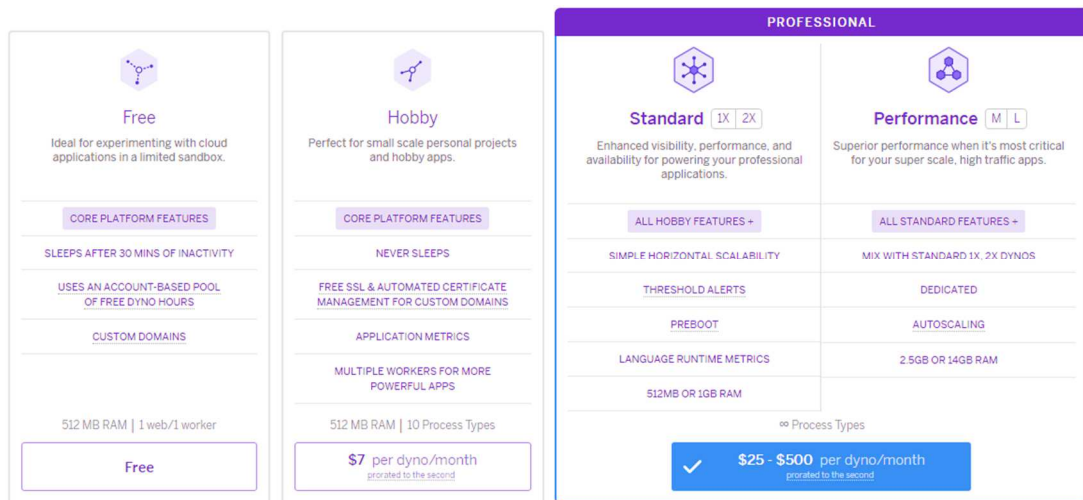


Figura 17: Costos referenciales de servicio de nube. Tomado de “Simple, flexible pricing. Plans for the needs of every app”, por Heroku, (s/f).

A nivel de base de datos para mayor capacidad, los costos del servicio se los visualiza en la figura 18.

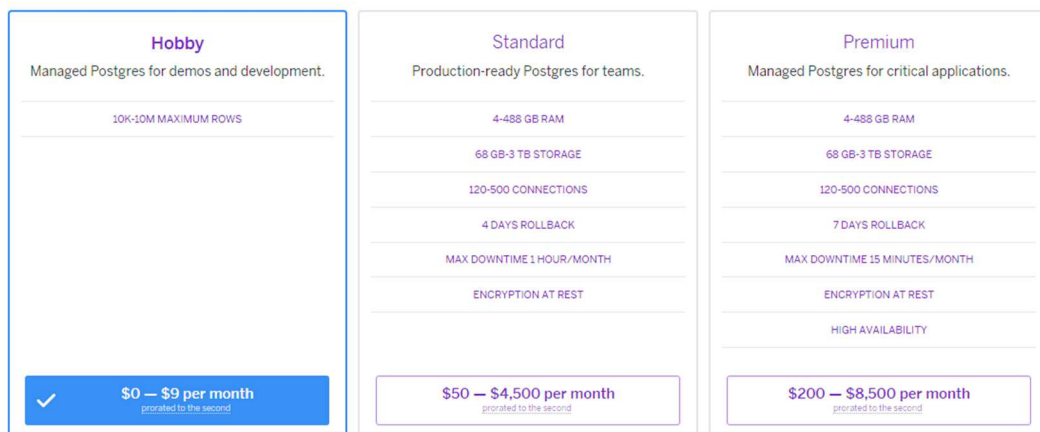


Figura 18: Costos referenciales de servicio de nube para base de datos. Tomado de “Simple, flexible pricing. Plans for the needs of every app”, por Heroku, (s/f).

Para el aplicativo móvil, no tiene ningún costo, por el hecho de que fue desarrollado con herramientas open source y será como parte del proyecto,

para que pueda ser utilizado en cualquier móvil que cumpla con los requerimientos mínimos indicado en los requisitos para el dispositivo móvil.

CONCLUSIONES

De los resultados del estudio realizado para este proyecto tecnológico, se puede afirmar que sí fue factible desarrollar e implementar el sistema gestor de encuestas para plataforma web como móvil, dinámica y que permita cuantificar la información recogida, la ubicación del lugar en donde se encuentra la encuesta mediante la georreferenciación y permitir el análisis de los resultados. Tanto el sistema web como el aplicativo móvil cubren las necesidades que tiene la Facultad de Ingeniería de contar con un sistema que agilice procesos en la realización de proyectos que requieran de la aplicación de encuestas.

Para la implementación del sistema se requirió de la identificación de las necesidades del usuario final de la aplicación, entendiéndose que el proyecto debe cubrir con el levantamiento de información personal y de relevancia de los grupos objetivos que son parte de los proyectos que se ejecutan en la Facultad de Ingeniería.

La elección de las herramientas de desarrollo fue realizada luego de una profunda investigación, llegando a determinarse que Angular fue la mejor opción y la que mejores resultados ofrece en el momento de escoger los lenguajes de programación, por su rendimiento y porque está orientado hacia dispositivos móviles, que es la segunda parte del proyecto de automatización de encuestas.

Con la especificación de las herramientas de desarrollo se diseñó y desarrolló el sistema gestor de encuestas para la Facultad de Ingeniería, el mismo que tiene las funcionalidades especificadas por el usuario final de la aplicación para la realización de los proyectos.

RECOMENDACIONES

Para optimizar el aplicativo web se sugiere que, como una medida de seguridad, se considere un control de sesiones, de manera que no se pueda ingresar al sistema en otro navegador con la misma dirección.

También, es recomendable que en la aplicación se considere la creación de grupos de trabajo y generación de zonas de trabajo, con el fin de llevar un control de los estudiantes que realicen las encuestas y realizar su seguimiento.

Además, el sistema web debe tener una pantalla que permita administrar la encuesta, para que ésta tenga un tiempo límite de realización. En la pantalla debe existir la opción de establecer la fecha y hora de inicio y fin de la encuesta que se va a realizar.

En cuanto a las preguntas, se debería incluir otro tipo de preguntas y eliminar las que son de respuesta abierta.

Mayor cantidad de reportería, que incluya reportes que filtren por fecha y hora de realización de la encuesta, y cantidad de estudiantes que se asignaron a la zona de la encuesta.

El diseño del aplicativo puede ser mejorado para que sea más amigable al usuario final.

GLOSARIO

API: La interfaz de programación de aplicaciones, abreviada como API del inglés: “Application Programming Interface”, es un conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

ECMA: Se trata de un acrónimo de “European Computer Manufacturers Association”, una organización internacional basada en membresías de estándares para la comunicación y la información.

Framework: Entorno de trabajo o marco de trabajo es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

HTTP: Son las siglas de “Hypertext Transfer Protocol” es un protocolo de transferencia donde se utiliza un sistema mediante el cual se permite la transferencia de información entre diferentes servicios y los clientes que utilizan páginas web.

ISO 9000/9001: Normas sobre calidad y gestión de calidad, establecidas por la Organización Internacional de Normalización.

ISO/IEC 15504 (SPICE): Conocido como Software “Process Improvement Capability Determination”, abreviado SPICE, en español, «Determinación de la Capacidad de Mejora del Proceso de Software» es un modelo para la mejora, evaluación de los procesos de desarrollo, mantenimiento de sistemas de información y productos de software.

MVC o Modelo-Vista-Controlador es un modelo de arquitectura de software que, utiliza Vistas, Models y Controladores y mediante este modelo se separa la lógica de la aplicación de la lógica de la vista en una aplicación. Utiliza elementos gráficos básicos y sistemas empresariales; la mayoría de los frameworks modernos utilizan MVC (o alguna adaptación del MVC) para

la arquitectura, como Ruby on Rails, Django, AngularJS y otros (Hernández, s.f., párr. 1).

SASS Syntactically Awesome StyleSheets es una extensión de CSS que le aporta características elegantes y robustas. Usa variables, reglas CSS anidadas, importación de hojas de estilos manteniendo la compatibilidad con CSS, contiene funciones que le permiten operar valores y colores, usa librerías y directivas de control, se generan archivos CSS (Librosweb, 2017).

SW-CMM: El Modelo de Madurez de la Capacidad para el desarrollo de Software.

REFERENCIAS

- Adobe. (2017). Diseño de sitios web adaptativos con Bootstrap. Recuperado el 26 de noviembre de 2017, a partir de <https://helpx.adobe.com/es/dreamweaver/using/bootstrap.html>
- ArcGIS Resource Center. (s.f.). Georreferenciación y sistemas de coordenadas. Recuperado el 1 de diciembre de 2017, a partir de <http://resources.arcgis.com/es/help/getting-started/articles/026n0000000s000000.htm>
- Arias, F. (2006). *El Proyecto de Investigación* (Quinta). Caracas: Episteme.
- Basu, S. (2013). Comenzando con TypeScript. Recuperado el 2 de diciembre de 2017, a partir de <https://code.tutsplus.com/es/tutorials/getting-started-with-typescript--net-28890>
- Behar Rivero, D. (2008). *Metodología de la investigación*. Shalom. Recuperado a partir de https://drive.google.com/file/d/0B00rfQ9umQIAOV9RWkxBdkh3OEK/view?pref=2&pli=1&usp=embed_facebook
- Bernal, C. A. (2010). *Metodología de la investigación, administración, economía humanidades y ciencias sociales* (Tercera). Bogotá: Pearson Educación de Colombia Ltda.
- Calero Clavijo, R. D. (2006). Metodología para la georeferenciación en el software Arcview 3.2 con ayuda de la extensión Santitools. Recuperado a partir de http://www.era-mx.org/Estudios_y_proyectos/Estudios/Analco2007-FLACSO/Reportes/SIG/ARC-VIEW3x/GEOREFERENCIACION.pdf
- campusMVP. (2017). Las 10 principales diferencias entre AngularJS y Angular. Recuperado el 1 de diciembre de 2017, a partir de <https://www.campusmvp.es/recursos/post/las-10-principales-diferencias-entre-angularjs-y-angular.aspx>

- Consejo de Educación Superior. regimen_academico.pdf, Pub. L. No. 051, RPC-SE-13 Reglamento de Régimen Académico (2013). Recuperado a partir de http://www.snaa.gov.ec/wp-content/themes/institucion/dw-pages/Descargas/regimen_academico.pdf
- Cuesta, A. (2014). Breve comparación entre Node.js y PHP. Recuperado el 10 de febrero de 2018, a partir de <http://mycyberacademy.com/breve-comparacion-entre-node-js-y-php/>
- Fernández Guerra, E. (2016). Introducción a TypeScript. Recuperado el 2 de diciembre de 2017, a partir de <http://www.desarrolloweb.com/articulos/introduccion-a-typescript.html>
- Gilfillan, I. (2003). *MySQL*. Madrid: Anaya Multimedia.
- Glass, R. L., Vessey, I., & Ramesh, V. (2002). Research in software engineering: an analysis of the literature. *Information and Software Technology*, 44(8), 491–506. [https://doi.org/10.1016/S0950-5849\(02\)00049-6](https://doi.org/10.1016/S0950-5849(02)00049-6)
- Google. (2017). Formularios [Encuesta online]. Recuperado el 24 de noviembre de 2017, a partir de https://gsuite.google.com/intl/es-419/products/forms/?utm_source=google&utm_medium=cpc&utm_campaign=latam-ROSA-all-es-dr--all-all-trial-r-latam-1002240-LUAC0000094&utm_content=text-ad-none-any-DEV_c-CRE_192291629477-ADGP_BKWS%20%7C%20UBM%20~%20Google%20Forms-KWID_43700014274613348-kwd-161937949974-userloc_20261&utm_term=KW_crear%20google%20formul%C3%A1rio-ST_crear%20google%20formul%C3%A1rio&gclid=CjwKCAiAo9_QBRACEiwASknDwWiL5Cv0f3FMUpX6UPqNX4UepseDBRvImrg5HN0OMLZwjJr3Q3iHVhoCh4QQA_vD_BwE&gclid=COaHztbS19cCFYcghwodXRkA0g

- Google Trends. (2018). Google Trends. Recuperado el 17 de febrero de 2018, a partir de <https://trends.google.es/trends/explore?q=ionic,%2Fm%2F0vpp538>
- Guérin, B.-A. (2014). *ASP.NET en C# con Visual Studio 2013: diseño y desarrollo de aplicaciones web*. Cornellà de Llobregat: ENI.
- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2014). *Metodología de la investigación* (Sexta). México: McGRAW-HILL / INTERAMERICANA EDITORES, S.A. DE C.V.
- Hernández, U. (s.f.). MVC (Model, View, Controller) explicado. Recuperado el 24 de noviembre de 2017, a partir de <https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>
- Heroku. (s/f). Simple, flexible pricing. Plans for the needs of every app. Recuperado el 21 de febrero de 2018, a partir de <https://www.heroku.com/pricing>
- JaGonzalez.org. (2014). 6 Características Que Hacen Tan Popular A Bootstrap. Recuperado el 27 de noviembre de 2017, a partir de <https://jagonzalez.org/6-caracteristicas-que-hacen-tan-popular-a-bootstrap/>
- Librosweb. (2017). Capítulo 1. Características (Sass, el manual oficial). Recuperado el 3 de diciembre de 2017, a partir de https://librosweb.es/libro/sass/capitulo_1.html
- MDN web docs. (2017). Fundamentos de JavaScript. Recuperado el 6 de diciembre de 2017, a partir de https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/JavaScript_basics
- Medina-Lozano, A., Villalobos-Salmerón, J. M., & Ordaz Celedón, M. A. (2017). Software educativo de costo. *Boletín Redipe*, 6(2), 206–213.

- MySQL. (2017). MySQL Workbench. Recuperado a partir de <https://downloads.mysql.com/docs/workbench-en.pdf>
- Niño Rojas, V. M. (2011). *Metodología de la investigación: diseño y ejecución* (Primera). Bogotá: Ediciones de la U. Recuperado a partir de <http://site.ebrary.com/id/10559875>
- Ollivier, S., & Gury, P.-A. (2016). *AngularJS desarrolle hoy las aplicaciones web de mañana*. Cornellà de Llobregat: ENI. Recuperado a partir de http://sfx.urv.cat/urv?url_ver=Z39.88-2004&url_ctx_fmt=info:ofi/fmt:kev:mtx:ctx&ctx_enc=info:ofi/enc:UTF-8&ctx_ver=Z39.88-2004&rfr_id=info:sid/sfxit.com:azbook&sfx.ignore_date_threshold=1&rft.object_id=22400000000000433
- Oracle. (s.f.). MySQL: La base de datos de código abierto más popular del mercado. Recuperado el 3 de diciembre de 2017, a partir de <https://www.oracle.com/es/mysql/index.html>
- Orjuela Duarte, A., & Rojas, M. (2008). Las Metodologías de Desarrollo Ágil como una oportunidad para la Ingeniería del Software educativo, 5(2). Recuperado a partir de <https://revistas.unal.edu.co/index.php/avances/article/view/10037>
- Ortego, D. (2017). Vue JS vs Angular: comparación entre líder y aspirante. Recuperado el 21 de febrero de 2018, a partir de <https://openwebinars.net/blog/vue-vs-angular/>
- Pavón Mestras, J. (2013). Bootstrap 3.0. Recuperado a partir de <https://www.fdi.ucm.es/profesor/jpavon/web/26-Bootstrap.pdf>
- Pérez Esteso, M. (s.f.). Programación Extrema: Qué es y principios básicos. Recuperado el 20 de diciembre de 2017, a partir de <https://geekytheory.com/programacion-extrema-que-es-y-principios-basicos>

- Pressman, R. S. (2010). *Ingeniería del software: un enfoque práctico* (Séptima). México: McGRAW-HILL INTERAMERICANA EDITORES, S.A. DE C.V.
- Programacion.net. (2017). Ionic, el framework para crear apps web con HTML5. Recuperado el 3 de diciembre de 2017, a partir de http://programacion.net/articulo/ionic-_el_framework_para_crear_apps_web_con_html5_1034
- Rodríguez, J. (2005). Definición de JavaScript. Recuperado el 28 de noviembre de 2017, a partir de <https://www.gestiopolis.com/definicion-javascript/>
- Rodríguez, T. (2012). Bootstrap from Twitter. Recuperado el 27 de noviembre de 2017, a partir de <https://www.genbetadev.com/frameworks/bootstrap>
- Salinas, P. J. (2010). *Metodología de la investigación científica*. Venezuela. Recuperado a partir de http://botica.com.ve/PDF/metodologia_investigacion.pdf
- Sommerville, I. (2011). *Ingeniería de software*. Madrid: Pearson Educación de México, S.A. de C.V.
- StackShare. (2018). Bootstrap vs Skeleton vs Foundation. Recuperado el 17 de febrero de 2018, a partir de <https://stackshare.io/stackups/bootstrap-vs-foundation-vs-skeleton>
- SurveyMonkey. (2017). SurveyMonkey. Toma de decisiones más inteligentes. Recuperado el 24 de noviembre de 2017, a partir de <https://es.surveymonkey.com/>
- UCSG. Estatuto de la Universidad Católica de Santiago de Guayaquil 2016, § Del Vicerrector o Vicerrectora de Vinculación, Nominación, Requisitos y Atribuciones (2016). Recuperado a partir de <http://www2.ucsg.edu.ec/transparencia-de-la-informacion/leyes-estatutos-reglamentos-y-resoluciones-administrativas/estatutos/649--358/file.html>

UCSG. (2017a). ¿Cómo se vincula la UCSG? [Educativo]. Recuperado el 24 de noviembre de 2017, a partir de <http://www2.ucsg.edu.ec/vinculacion/como-se-vincula-la-ucsg.html>

UCSG. (2017b). Vicerrectorado de Vinculación. Áreas de intervención de la vinculación en la UCSG [Educativo]. Recuperado el 4 de diciembre de 2017, a partir de <http://www.ucsg.edu.ec/vinculacion/areas-de-intervencion-de-la-ucsg.html>

UCSG. (2017c). Vicerrectorado de Vinculación. Política General [Educativo]. Recuperado el 4 de diciembre de 2017, a partir de <http://www.ucsg.edu.ec/vinculacion/politica-general.html>

Ugalde Binda, N., & Balbastre-Benavent, F. (2013). Investigación cuantitativa e investigación cualitativa: buscando las ventajas de las diferentes metodologías de investigación. *Revista de Ciencias económicas*, 31(2), 179–187. Recuperado a partir de <https://revistas.ucr.ac.cr/index.php/economicas/article/viewFile/12730/11978>

Uptodown Technologies. (2017). MySQL Workbench 6.3.4 (32-bit) [Bases de datos]. Recuperado el 3 de diciembre de 2017, a partir de <https://mysql-workbench.uptodown.com/windows>

ANEXOS

Anexo 1: *Entrevista 1*

ENTREVISTA AL ENCARGADO DE VINCULACIÓN POR LAS CARRERAS DE COMPUTACIÓN E INGENIERÍA EN SISTEMAS COMPUTACIONALES DE LA FACULTAD DE INGENIERÍA

7. ¿Por qué cree usted necesaria la implementación de una aplicación de encuesta para para la realización de proyectos en la Facultad?

.....
.....

8. Dentro de los requisitos funcionales necesarios para el desarrollo de la aplicación web ¿qué tipo de preguntas deberá considerarse en el desarrollo de la aplicación?

.....
.....

9. ¿Qué tipos de respuestas se deberá manejar la encuesta para desarrollarla en el sistema?

.....
.....

10. ¿Qué debería tenerse en cuenta para el manejo de los posibles usuarios del sistema?

.....
.....

11. En cuanto a la interfaz del sistema ¿cuál deberá ser la secuencia de exploración?

.....
.....

12. Similar a los requisitos para el sistema web, ¿qué deberá realizar la aplicación móvil?

.....
.....

ENTREVISTA A DOCENTE DE TIEMPO COMPLETO DE LA CARRERA DE CIENCIAS DE LA COMPUTACIÓN E INGENIERÍA EN SISTEMAS COMPUTACIONALES DE LA FACULTAD DE INGENIERÍA

4. ¿Qué requerimientos de forma debería tener la aplicación?

.....
.....

5. ¿Qué otros requerimientos funcionales considera apropiados para la aplicación?

.....
.....

6. ¿Qué tipo de preguntas no deberían ser consideradas en la encuesta?

.....
.....

**ENTREVISTA AL SICÓLOGO DEL DEPARTAMENTO COMISIÓN DE
EVALUACIÓN INTERNA CEI**

6. ¿Qué características deben tener las preguntas de la encuesta?

.....
.....

7. ¿Qué tiempo promedio debería durar una encuesta?

.....
.....

8. ¿Cuántas preguntas en promedio debería tener una encuesta?

.....
.....

9. ¿Qué tipos de preguntas no deberían ser incluidas en una encuesta?

.....
.....

10. ¿Qué consideraciones deberían tomarse en cuenta para elaborar las preguntas de la encuesta?

.....
.....

Anexo 4: **Casos de uso sistema web**

Anexo 4.1: *Caso de uso: Ingreso al sistema web*

ATRIBUTO	DESCRIPCIÓN
Código	CU-1
Nombre	Ingreso al sistema web
Autor	Diego López
Fecha	
Descripción El usuario Administrador ingresa al sistema para la creación de proyectos y encuestas	
Actores Administrador	
Precondiciones Haber ingresado al sistema web	
Flujo Normal Administrador: El administrador ingresa al sistema web El administrador revisa las opciones del sistema El administrador realiza el proceso El administrador cierra el sistema	
Flujo Alternativo	
Restricciones El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones El Administrador ingresa al sistema web para crear proyectos y encuestas	

Anexo 4.2: *Caso de uso Administración de usuarios*

ATRIBUTO	DESCRIPCIÓN
Código	CU-2
Nombre	Caso de uso Administración de usuarios
Autor	Diego López
Fecha	
Descripción El Administrador se encarga de la administración de los usuarios: crear usuarios, otorgar y quitar privilegios (encuestador y administrador)	
Actores Administrador	
Precondiciones Haber ingresado al sistema web	
Flujo Normal Administrador: El Administrador ingresa al sistema con su usuario y contraseña El Administrador crea el/los usuario/s con los privilegios correspondientes El Administrador sale del sistema	
Flujo Alternativo	
Restricciones El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones El Administrador crea los usuarios del sistema con sus privilegios	

Anexo 4.3: Caso de uso Creación de proyectos

ATRIBUTO	DESCRIPCIÓN
Código	CU-3
Nombre	Creación de proyectos
Autor	Diego López
Fecha	
Descripción El Administrador crea proyectos para la Facultad, para la generación de una encuesta	
Actores Administrador	
Precondiciones Haber ingresado al sistema web	
Flujo Normal Administrador: El Administrador ingresa al sistema con su usuario y contraseña El Administrador ingresa al módulo de proyectos El Administrador crea el proyecto con su nombre correspondiente	
Flujo Alternativo	
Restricciones El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones El Administrador crea un proyecto para la Facultad con las opciones que requiere	

Anexo 4.4: Caso de uso Creación de encuestas

ATRIBUTO	DESCRIPCIÓN
Código	CU-4
Nombre	Creación de encuestas
Autor	Diego López
Fecha	
Descripción El Administrador crea una encuesta para aplicarla de acuerdo a la necesidad del proyecto	
Actores Administrador	
Precondiciones Haber ingresado al sistema web Haber creado el proyecto	
Flujo Normal Administrador: El Administrador ingresa al sistema web con su usuario y contraseña El Administrador crea el proyecto	
Flujo Alternativo	
Restricciones El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones El Administrador crea la encuesta	

Anexo 4.5: Caso de uso Creación de grupo de preguntas

ATRIBUTO	DESCRIPCIÓN
Código	CU-5
Nombre	Creación de grupo de preguntas
Autor	Diego López
Fecha	
Descripción El Administrador crea el grupo de preguntas de tipo general (nombre, edad, sexo) y técnicas (utilización de temas tecnológicos)	
Actores Administrador	
Precondiciones Haber ingresado al sistema web Haber creado el proyecto Haber creado la encuesta	
Flujo Normal Administrador: El Administrador ingresa al sistema con su usuario y contraseña El Administrador crea el proyecto El Administrador crea la encuesta	
Flujo Alternativo	
Restricciones El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones El Administrador crea el grupo de preguntas para la encuesta	

Anexo 4.6: Caso de uso Creación de preguntas

ATRIBUTO	DESCRIPCIÓN
Código	CU-6
Nombre	Creación de preguntas
Autor	Diego López
Fecha	
Descripción El Administrador crea las preguntas para el grupo de preguntas que tendrá la encuesta del proyecto de la Facultad	
Actores Administrador	
Precondiciones Haber ingresado al sistema web Haber creado el proyecto Haber creado la encuesta Haber creado el grupo de preguntas	
Flujo Normal Administrador: El Administrador ingresa al sistema con su usuario y contraseña El Administrador crea el proyecto El Administrador crea la encuesta El Administrador crea el grupo de preguntas	
Flujo Alternativo	
Restricciones El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones El Administrador crea las preguntas para la encuesta	

Anexo 4.7: Caso de uso Revisión de tabulación de encuestas

ATRIBUTO	DESCRIPCIÓN
Código	CU-7
Nombre	Revisión de tabulación de encuestas
Autor	Diego López
Fecha	
Descripción	
El Administrador revisa la tabulación de los resultados de las encuestas ingresadas al sistema	
Actores	
Administrador	
Precondiciones	
Haber ingresado al sistema web Haber ingresado al módulo de tabulación y estadísticas	
Flujo Normal	
Administrador: El Administrador ingresa al sistema con su usuario y contraseña El Administrador ingresa al módulo de tabulación y estadísticas El Administrador revisa informe de encuestas El Administrador revisa estadísticas gráficas El Administrador sale del sistema	
Flujo Alternativo	
Restricciones	
El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones	
El Administrador revisa el informe de encuestas y estadísticas gráficas	

Anexo 4.8: Caso de uso Revisión de georreferenciación

ATRIBUTO	DESCRIPCIÓN
Código	CU-8
Nombre	Revisión de georreferenciación
Autor	Diego López
Fecha	
Descripción	
El Administrador localiza la encuesta mediante la georreferenciación a través del mapa de la aplicación web.	
Actores	
Administrador	
Precondiciones	
Haber ingresado al sistema web Haber ingresado al módulo de tabulación y estadísticas Haber ingresado a la opción informe detallado de las encuestas	
Flujo Normal	
Administrador: El Administrador ingresa al sistema con su usuario y contraseña El Administrador ingresa al módulo detalle de encuestas El Administrador revisa la ubicación de la encuesta mediante la georreferenciación y ubica en el mapa el lugar en el que se realiza la encuesta	
Flujo Alternativo	
Restricciones	
El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones	
El Administrador localiza la ubicación de la encuesta a través de la georreferenciación mediante el mapa de ubicación.	

Anexo 5: **Caso de uso sistema móvil**

Anexo 5.1: *Caso de uso Ingreso al sistema móvil*

ATRIBUTO	DESCRIPCIÓN
Código	CU-1
Nombre	Ingreso al sistema móvil
Autor	Diego López
Fecha	
Descripción El usuario Encuestador ingresa al sistema para la elaboración de la encuesta	
Actores Encuestador	
Precondiciones Haber ingresado al sistema web	
Flujo Normal Administrador: El encuestador ingresa al sistema web El encuestador revisa las opciones del sistema	
Flujo Alternativo	
Restricciones El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones El Encuestador ingresa al sistema web	

Anexo 5.2: Realización de encuesta

ATRIBUTO	DESCRIPCIÓN
Código	CU-2
Nombre	Realización de encuesta
Autor	Diego López
Fecha	
Descripción El usuario Encuestador ingresa al sistema para realizar la encuesta	
Actores Encuestador	
Precondiciones Haber ingresado al sistema web	
Flujo Normal Administrador: El encuestador ingresa al sistema web El encuestador realiza la encuesta El encuestador sale el sistema	
Flujo Alternativo	
Restricciones El usuario se encuentra bloqueado No hay red para ingresar al sistema El sitio web se encuentra en mantenimiento El hardware no funciona correctamente	
Post Condiciones El Encuestador ingresa al sistema web para realizar la encuesta	

Anexo 6: **Diccionario de datos**

Anexo 6.1: *Tabla usuario*

NOMBRE DE LA TABLA: usuario				
DESCRIPCIÓN: permite el ingreso de los usuarios que van a utilizar el sistema				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de usuario
correo	Varchar(200)	PK	X	Correo de usuario
usuario	Varchar(200)		X	Nick de usuario
contrasena	Varchar(1000)		X	Contraseña de usuario
nombre	Varchar(200)		X	Nombre de usuario
apellido	Varchar(200)		X	Apellido de usuario
estado	Char(2)			Estado de usuario

Anexo 6.2: *Tabla usuariorol*

NOMBRE DE LA TABLA: usuariorol				
DESCRIPCIÓN: se asigna un identificador único de usuario e identificador único de rol				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
idusuario	Int(11)	PK	X	Id de rol de usuario
idrol	Int(11)	PK	X	Id de rol

Anexo 6.3: *Tabla rol*

NOMBRE DE LA TABLA: rol				
DESCRIPCIÓN: se determinan los privilegios que van a tener los usuarios en el sistema				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de rol
abr	Char(3)		X	
rol	Varchar(200)		X	Rol de usuario
descripcion	Varchar(400)		X	Descripción de rol
estado	Char(2)			Estado de rol

Anexo 6.4: *Tabla roltransaccion*

NOMBRE DE LA TABLA: roltransaccion				
DESCRIPCIÓN: se asigna un identificador único de usuario e identificador único de transacción				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
idrol	Int(11)	PK	X	Id de rol
idtransaccion	Int(11)	PK	X	Id de transacción de rol

Anexo 6.5: *Tabla transaccion*

NOMBRE DE LA TABLA: transaccion				
DESCRIPCIÓN:				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de transacción
transaccion	Varchar(200)		X	Tipo de transacción
opcion	Varchar(200)		X	Opción de transacción
descripcion	Varchar(400)		X	Descripción de transacción
estado	Char(2)			Estado de transacción

Anexo 6.6: *Tabla resultado*

NOMBRE DE LA TABLA: resultado				
DESCRIPCIÓN: registra el detalle de las encuestas a realizarse				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de resultado
idusuario	Int(11)	FK	X	Id de usuario
idencuesta	Int(11)	FK	X	Id de encuesta
nombre	Varchar(300)		X	Nombre del encuestador
latitud	Double		X	Latitud de resultado
longitud	Double		X	Longitud de resultado
minuto	Int(11)		X	Minuto de realización
segundo	Int(11)		X	Segundo de realización
fechacreacion	Datetime		X	Fecha de creación
fechamodificacion	Datetime		X	Fecha de modificación

Anexo 6.7: *Tabla resultadorespuesta*

NOMBRE DE LA TABLA: resultadorespuesta				
DESCRIPCIÓN: registra los resultados de la encuesta que se van a tabular				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
idresultado	Int(11)	FK	X	Id de resultado
idpregunta	Int(11)	FK	X	Id de pregunta
idrespuesta	Int(11)		X	Id de respuesta
respuesta	Varchar(400)		X	Respuesta de resultado

Anexo 6.8: *Tabla respuesta*

NOMBRE DE LA TABLA: respuesta				
DESCRIPCIÓN: registra la respuesta de la encuesta				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de respuesta
Idpregunta	Int(11)	PK	X	Id de pregunta
texto	Varchar(200)		X	Texto de respuesta
valor	Varchar(200)		X	Valor de respuesta
estado	Char(2)			Estado de respuesta

Anexo 6.9: *Tabla encuesta*

NOMBRE DE LA TABLA: encuesta				
DESCRIPCIÓN: registra los datos informativos de la encuesta a realizar				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de encuesta
idProyecto	Int(11)	FK	X	Id de proyecto
titulo	Varchar(200)		X	Título de encuesta
descripcion	Varchar(400)		X	Descripción de encuesta
latitud	Double		X	Latitud de encuesta
longitud	Double		X	Longitud de encuesta
estado	Char(2)			Estado de encuesta

Anexo 6.10: *Tabla grupos*

NOMBRE DE LA TABLA: grupos				
DESCRIPCIÓN: se guardan los datos de los grupos de preguntas				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de grupos
Idencuesta	Int(11)	FK	X	Id de encuesta
descripcion	Varchar(100)		X	Descripción de grupos
estado	Char(2)			Estado de grupos

Anexo 6.11: *Tabla pregunta*

NOMBRE DE LA TABLA: pregunta				
DESCRIPCIÓN: se registran los tipos de preguntas de la encuesta				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de pregunta
idgrupo	Int(11)	FK	X	Id de grupo
idtipo	Int(11)	PK	X	Id de tipo de pregunta
posición	Int(11)		X	Se guarda la respuesta marcada de pregunta
pregunta	Varchar(400)		X	Pregunta
estado	Char(2)			Estado de pregunta

Anexo 6.12: *Tabla encuestador*

NOMBRE DE LA TABLA: encuestador				
DESCRIPCIÓN: se registra el tipo de encuestador y el tipo de encuesta a realizar				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
idusuario	Int(11)	PK	X	Id de encuestador
idencuesta	Int(11)	PK	X	Id de encuesta

Anexo 6.13: *Tabla proyecto*

NOMBRE DE LA TABLA: proyecto				
DESCRIPCIÓN: se registra el proyecto a crear				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de proyecto
nombre	Varchar(200)		X	Nombre de proyecto
descripcion	Varchar(400)		X	Descripción de proyecto
estado	Char(2)			Estado de proyecto

Anexo 6.14: *Tabla tipopregunta*

NOMBRE DE LA TABLA: tipopregunta				
DESCRIPCIÓN: se registra el tipo de pregunta a realizar en la encuesta				
CAMPO	TIPO DE DATO	CAMPO CLAVE	NOT NULL	DESCRIPCIÓN
id	Int(11)	PK	X	Id de tipo de pregunta
tipo	Char(3)		X	Tipo de pregunta
descripcion	Varchar(100)		X	Descripción de pregunta
estado	Char(45)			Estado de pregunta

MANUAL DE USUARIO DEL SISTEMA WEB PARA AUTOMATIZACIÓN DE ENCUESTAS PARA LOS PROYECTOS DE VINCULACIÓN DE LA FACULTAD DE INGENIERÍA

Introducción

¿Qué es el sistema para automatización de encuestas para proyectos de Vinculación?

Es un sistema para el levantamiento de información de grupos objetivo necesarios en los proyectos de vinculación de la UCSG, a través del cual se realicen encuestas con varios tipos de preguntas, que puedan ser localizadas en determinada área geográfica mediante la georreferenciación.

¿Cuál es el objetivo de este manual de usuario?

El objetivo de este manual es servir de apoyo para el usuario final para el manejo del sistema, permitiendo que se conozcan las características del mismo y su funcionalidad.

Requerimientos básicos

Se necesitan los siguientes requerimientos para el sistema:

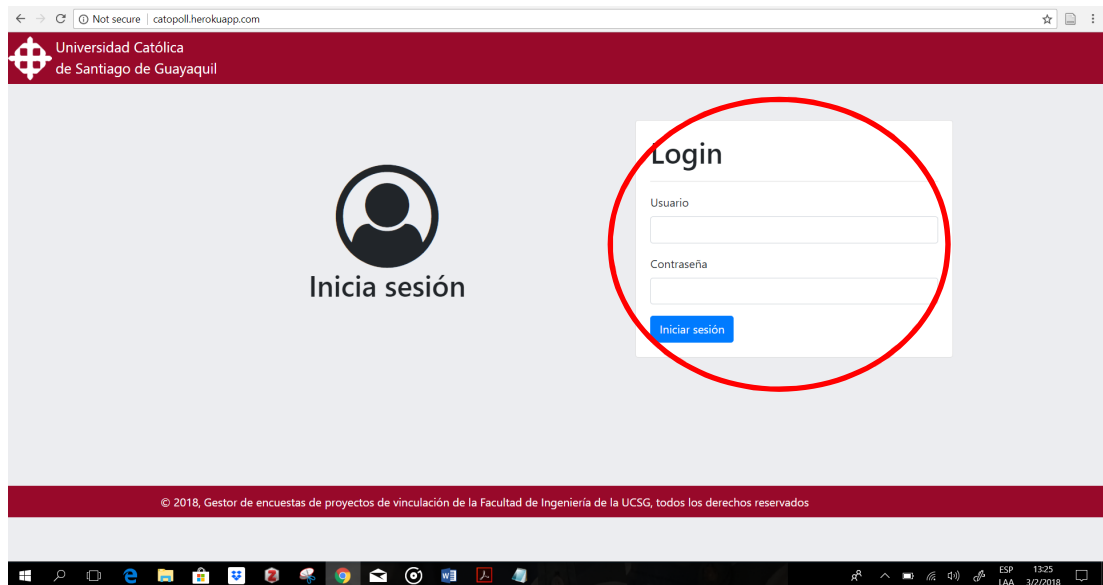
De software: sistema operativo (Windows, Linux, Mac OS), acceso a Internet, navegador Web. Se recomienda google Chrome o Mozilla Firefox (últimas versiones).

De hardware: Un ordenador que posea mínimamente las siguientes características: mainboard Biostar J3060NH o versiones parecidas, procesador Dual Core 3.00 GHz, RAM de 2 GB, Modem con servicio de internet de 1 Mbps, monitor genérico, mouse, teclado.

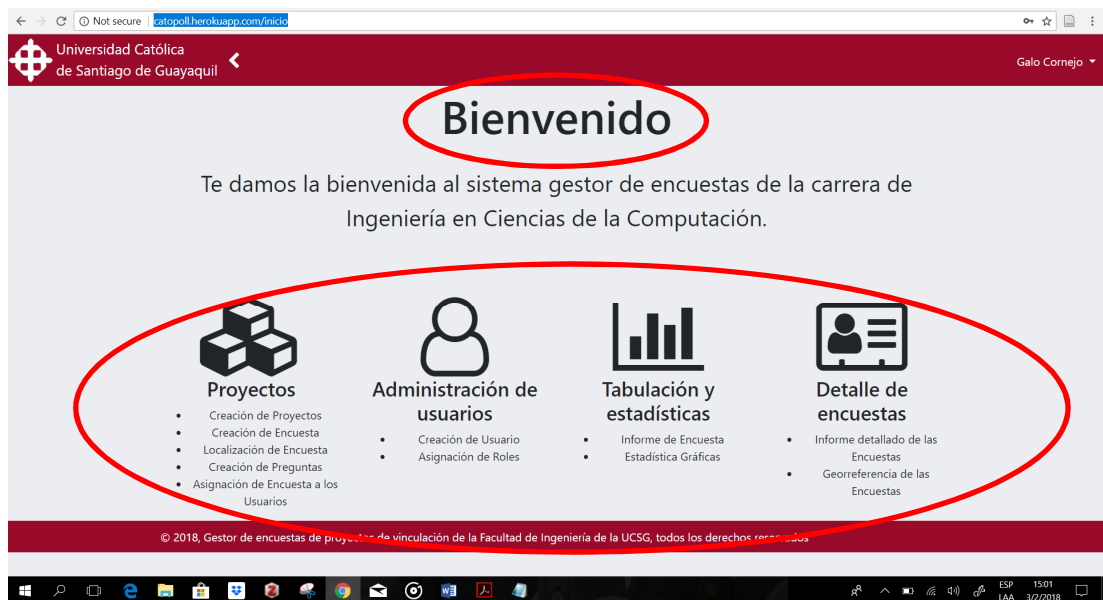
Acceso a la aplicación

Para acceder al sistema web, en el navegador se deberá acceder a la dirección: <http://catopoll.herokuapp.com/>

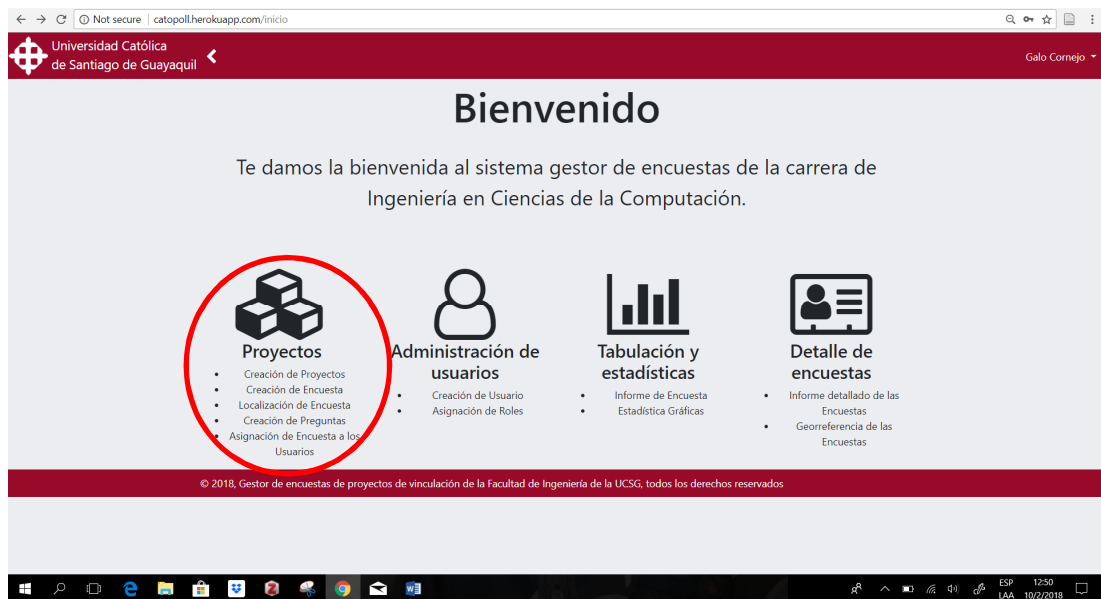
La primera pantalla que se abrirá es la de ingreso al sistema, en la cual para el inicio de sesión se requiere del usuario y contraseña.



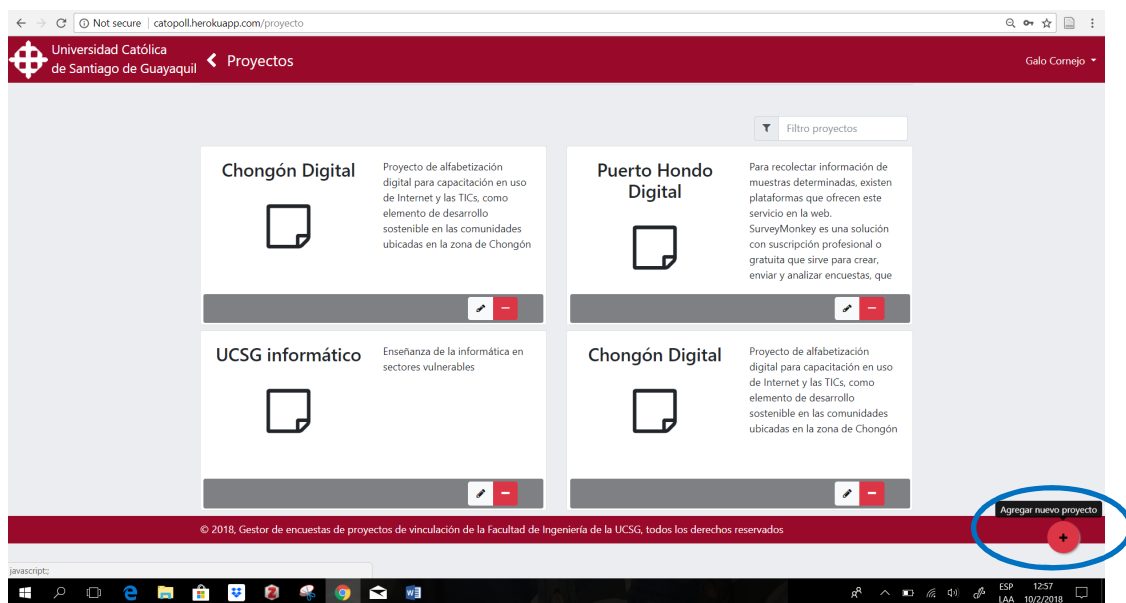
Una vez iniciada la sesión, se accede a la pantalla Bienvenido, en la cual se pueden visualizar los módulos del sistema web: Proyectos, Administración de usuarios, Tabulación y estadísticas y Detalle de encuestas, cada uno con sus especificaciones.



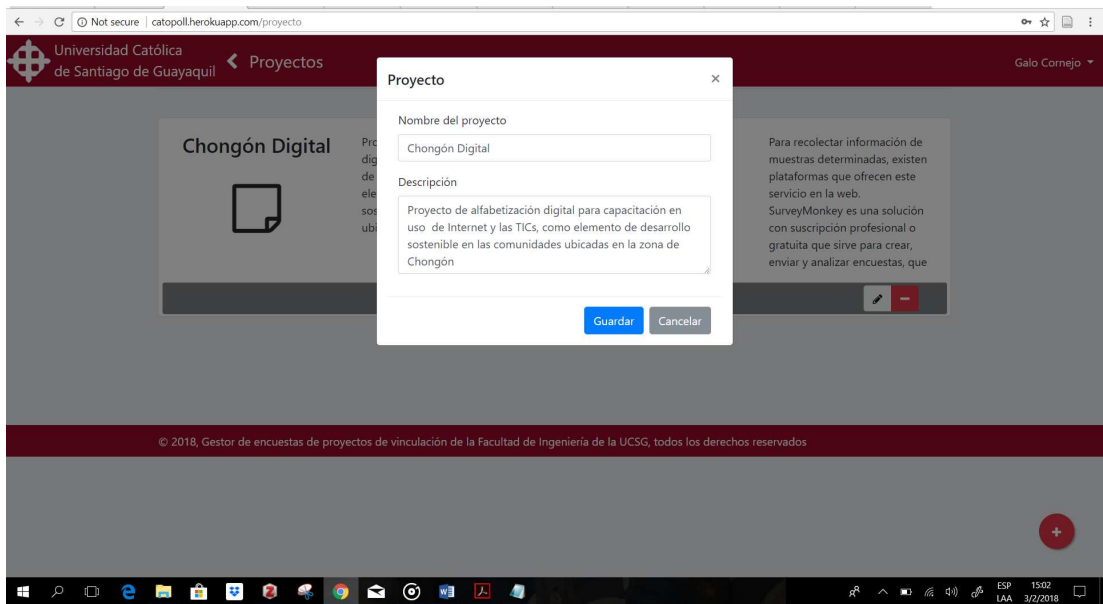
El primer módulo es el de Proyectos. En este módulo se encuentran las siguientes opciones: Creación de Proyectos, Creación de Encuesta, Localización de Encuesta, Creación de Preguntas y Asignación de Encuesta a los usuarios.



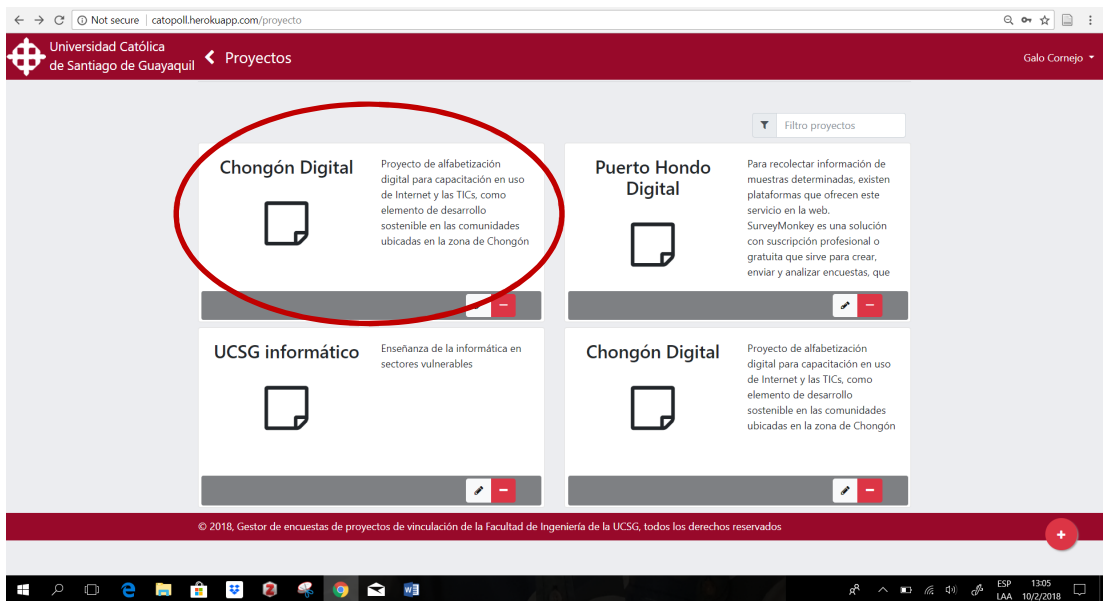
En el módulo Proyectos se visualizan los proyectos creados y se Agrega un nuevo proyecto, luego de dar clic en el signo +



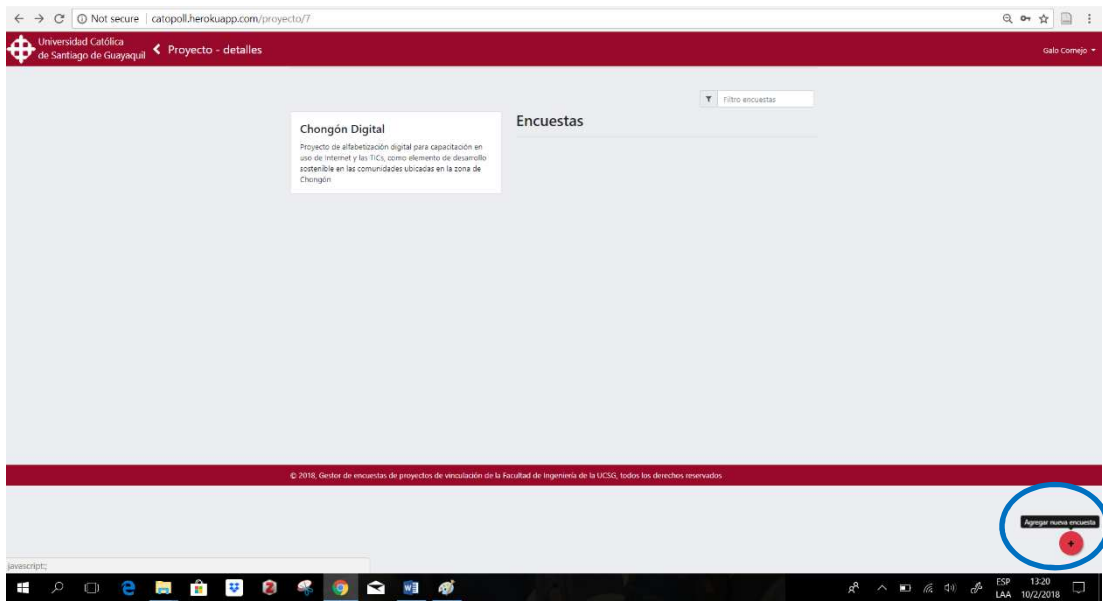
Aparece la ventana para la creación del proyecto, en la cual se requiere la especificación del nombre y la descripción del mismo. Se guardan los cambios y se crea el proyecto.



Para la creación de las encuestas, se ingresa al Proyecto que se creó.

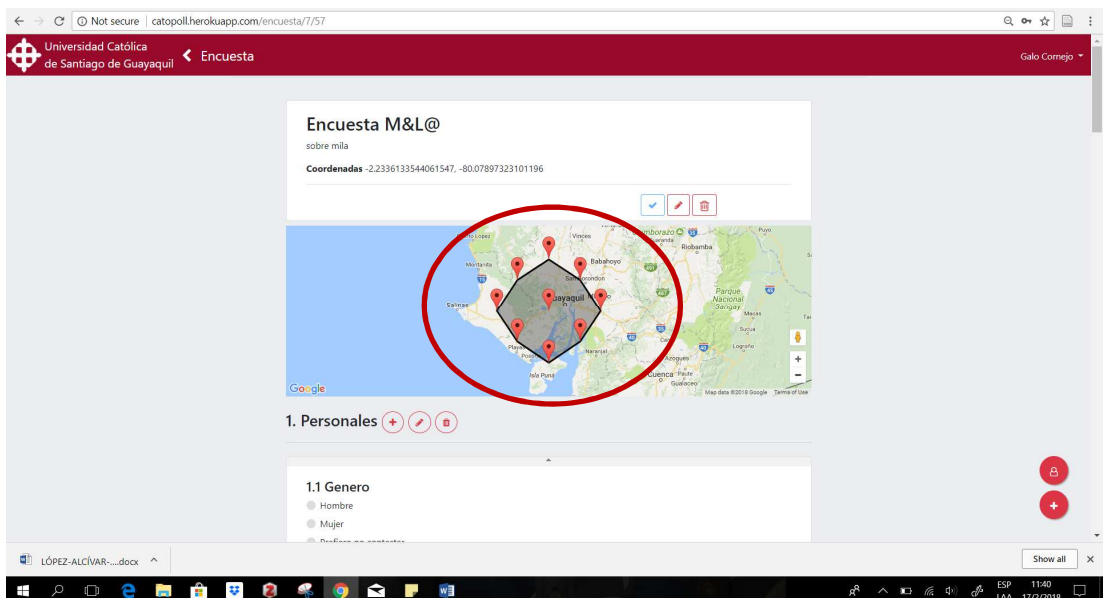


Dentro del módulo se crea la encuesta, mediante la opción Agregar nueva encuesta, luego de dar clic en el signo +

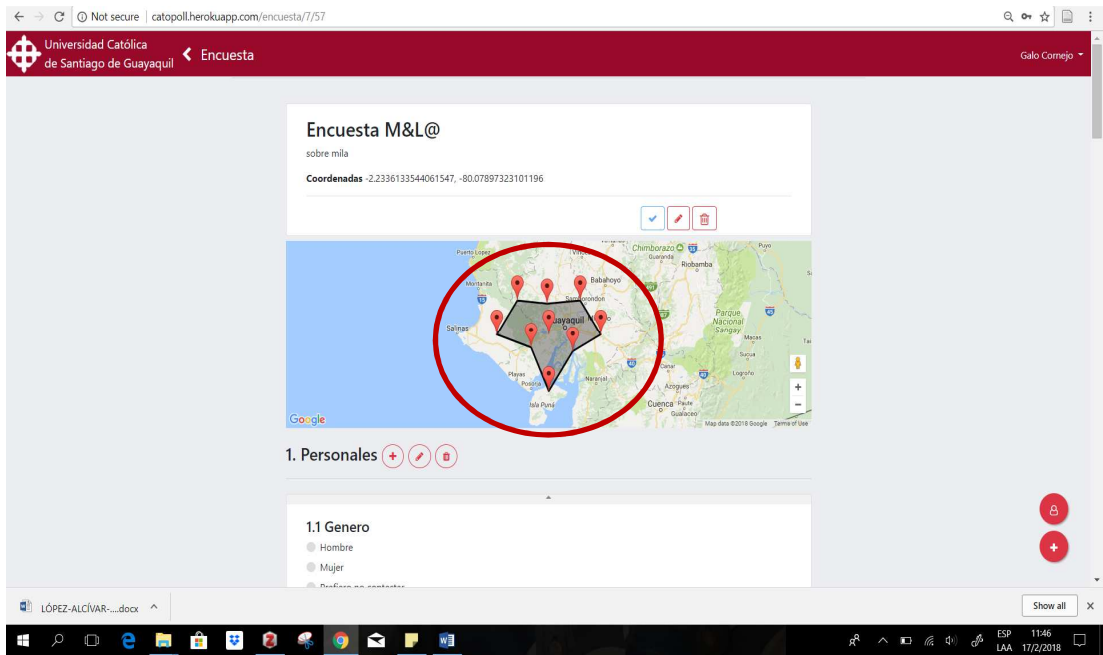


Para crear la encuesta, se debe colocarle un título y una descripción sobre el propósito de la encuesta en los campos sugeridos. Se realiza la georreferenciación en el buscador de Google, que presenta las coordenadas de la localización en donde se va a realizar la encuesta. Se guardan los cambios.

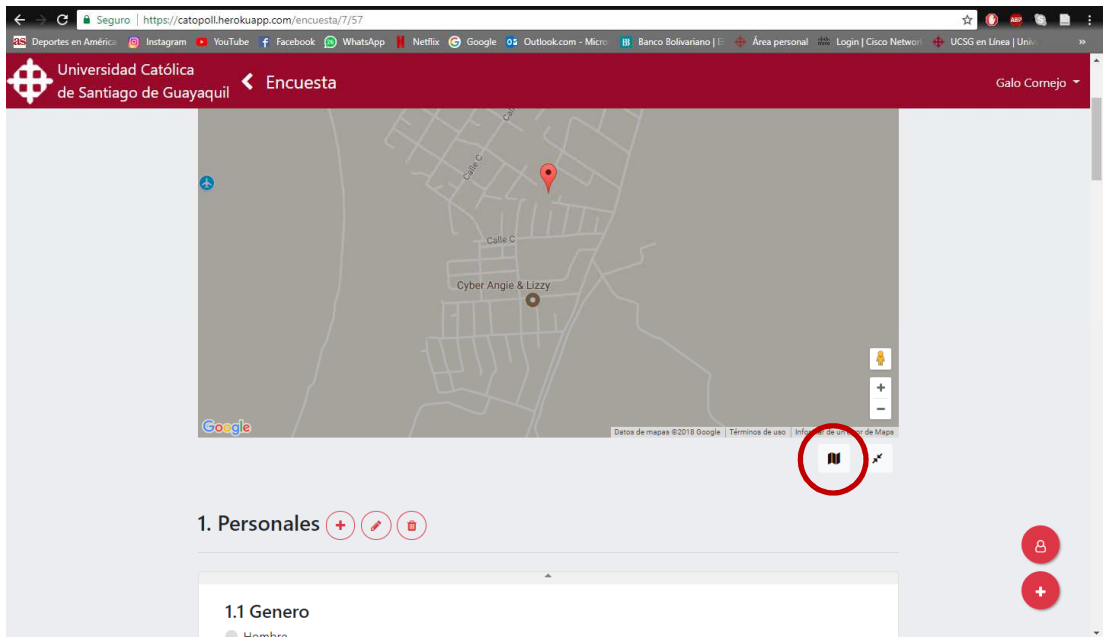
Se genera un polígono de ocho puntos del área en donde se va a realizar la encuesta.



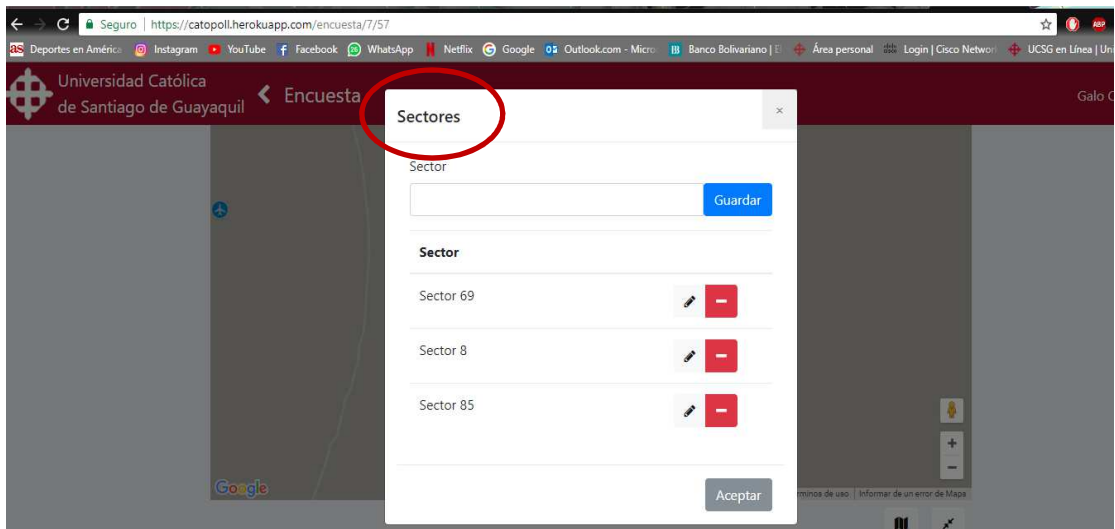
Si se necesita especificar con mayor precisión el área de la encuesta, se puede mover el mouse para determinarla.



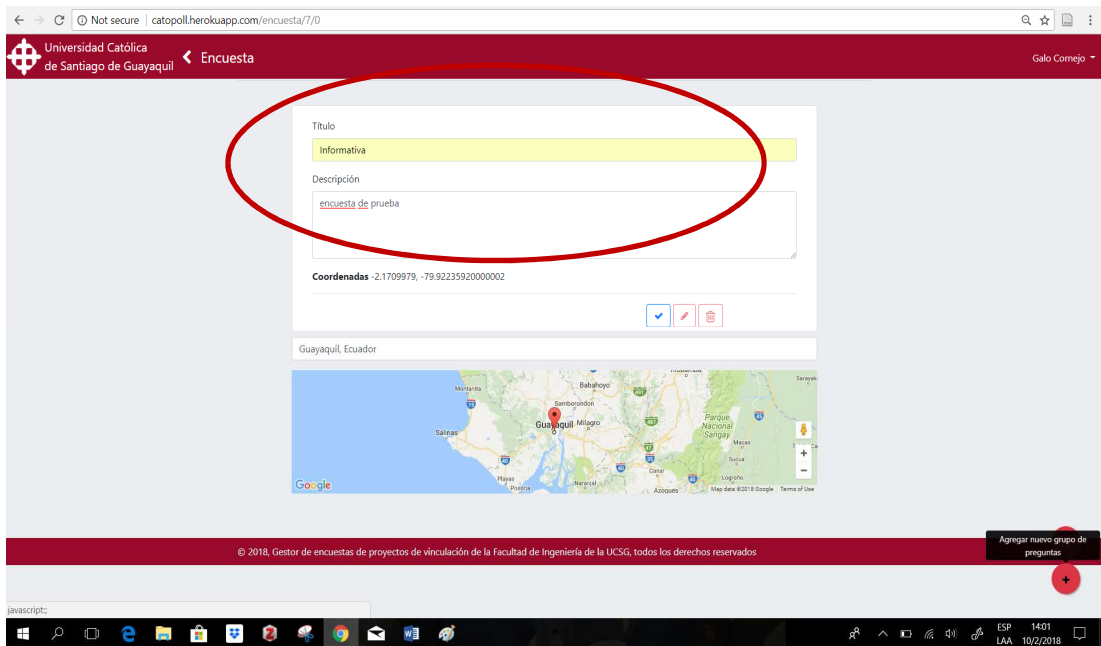
Dentro del polígono de ocho puntos se establecen los sectores para asignar la encuesta a los estudiantes. Para esto se da clic en el botón bajo el mapa de ubicación.



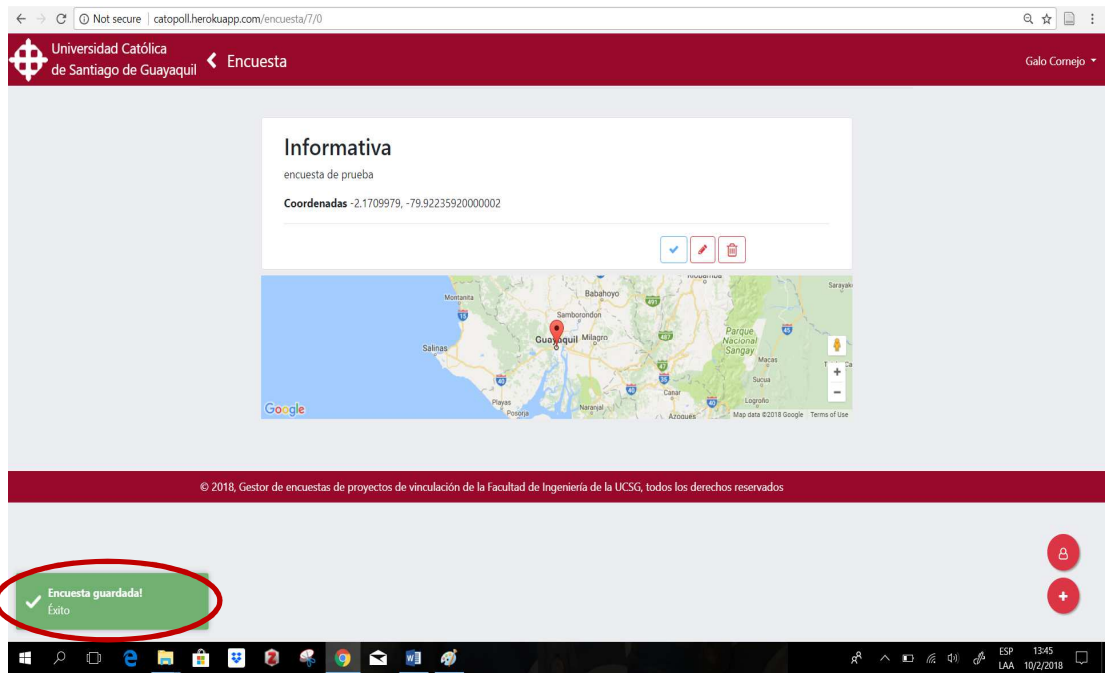
Se despliega una pantalla donde se podrá ingresar la cantidad de sectores que hay dentro del polígono.



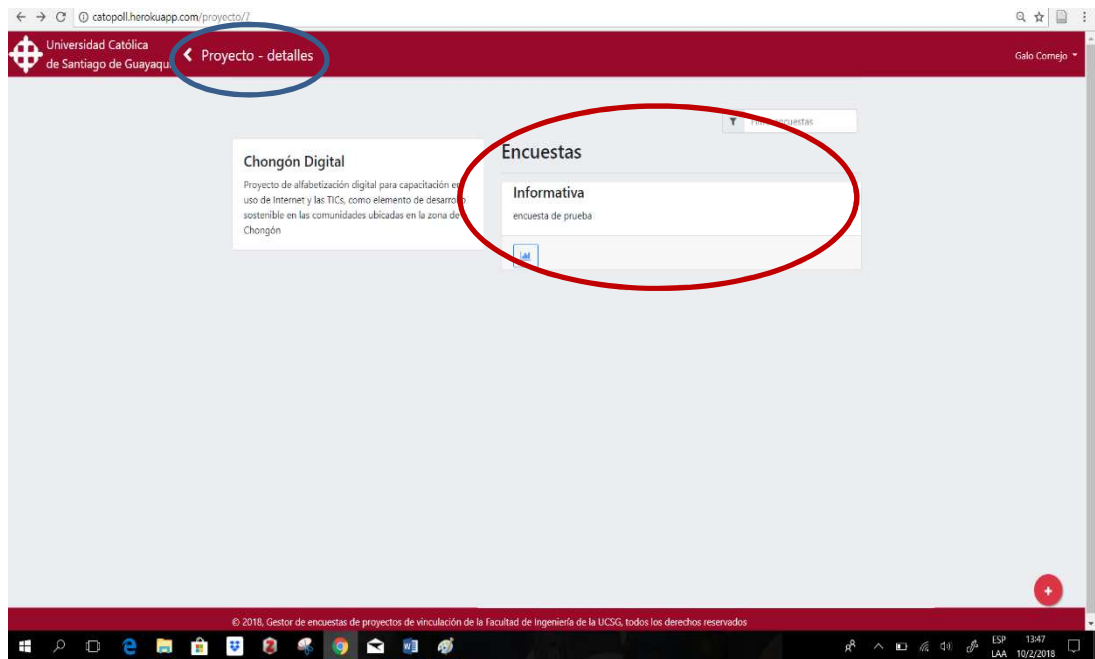
Se puede además modificar los campos de título o descripción, o en su defecto eliminar la encuesta.



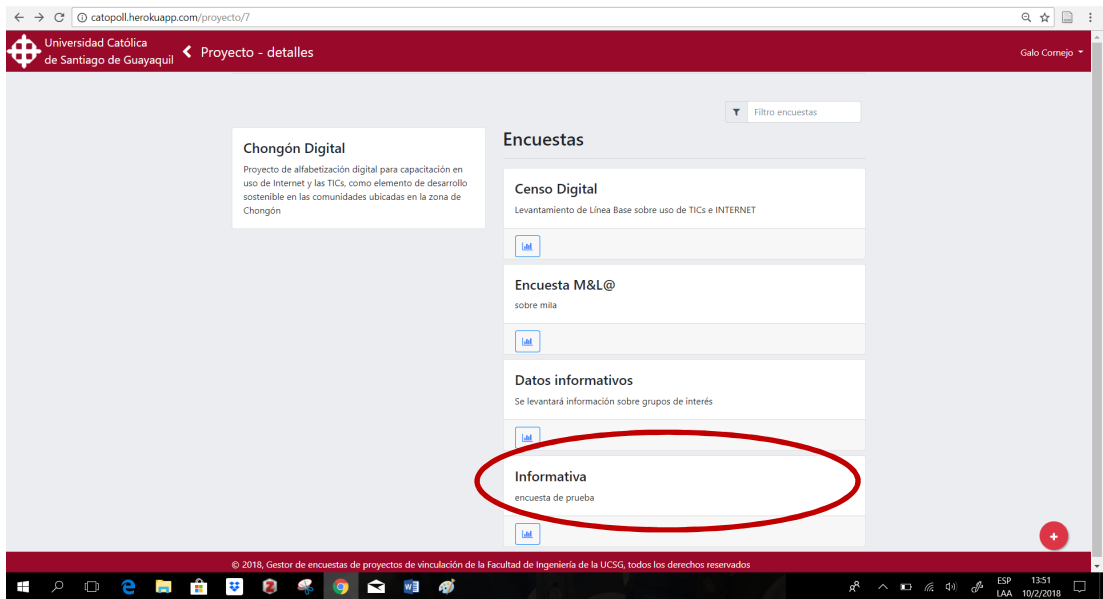
Se guardan los cambios, en caso de realizarlos.



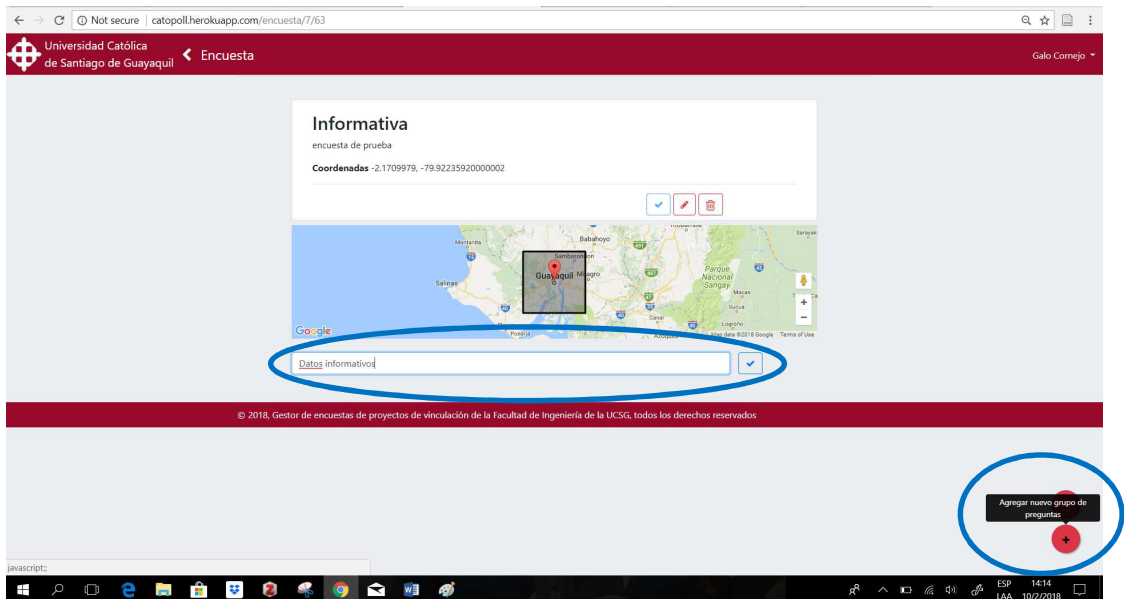
Creada la encuesta, esta se refleja en el módulo Proyectos.



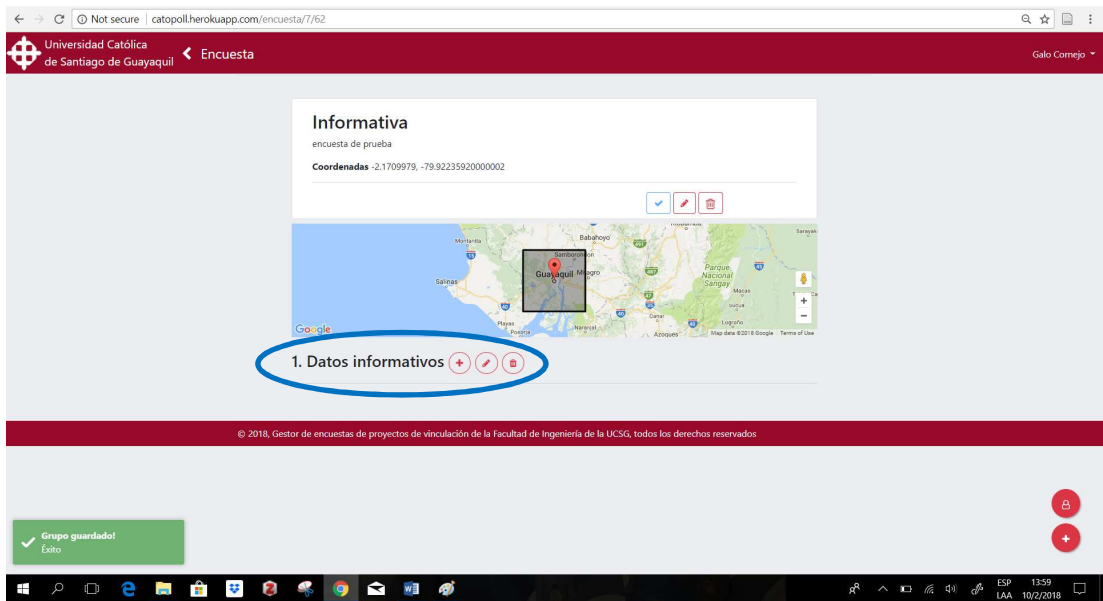
Se pueden crear la cantidad de encuestas que se requieran para el Proyecto.



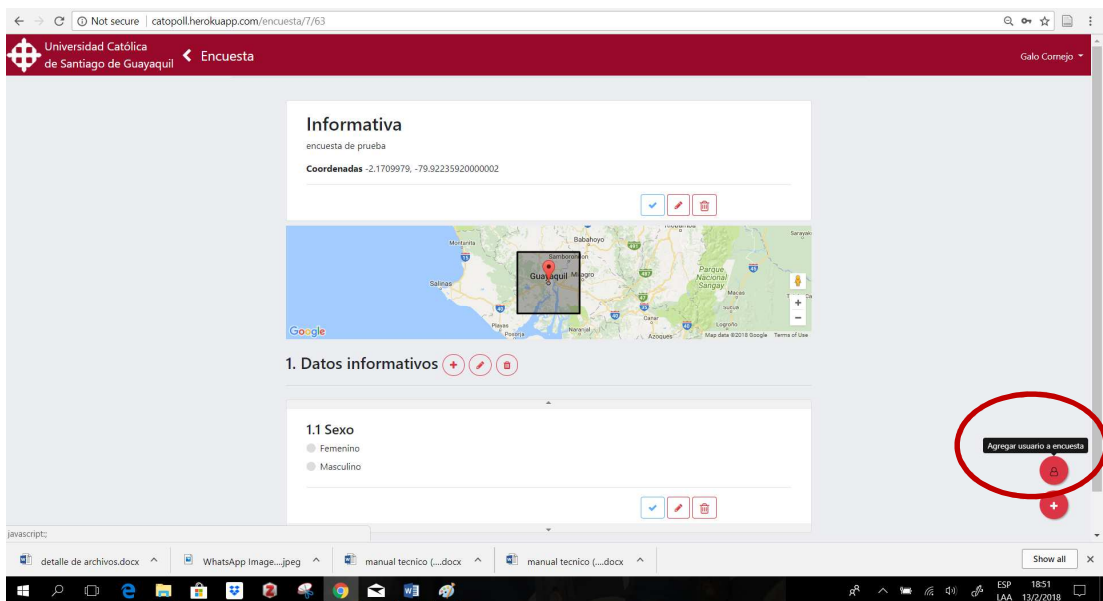
Creada la encuesta del Proyecto, se accede a ella para crear el Grupo de preguntas, mediante la acción del signo +, el mismo que adiciona a la ventana un cuadro de texto para colocar el título o nombre del grupo de preguntas a crear. Se guardan los cambios en la opción ✓.



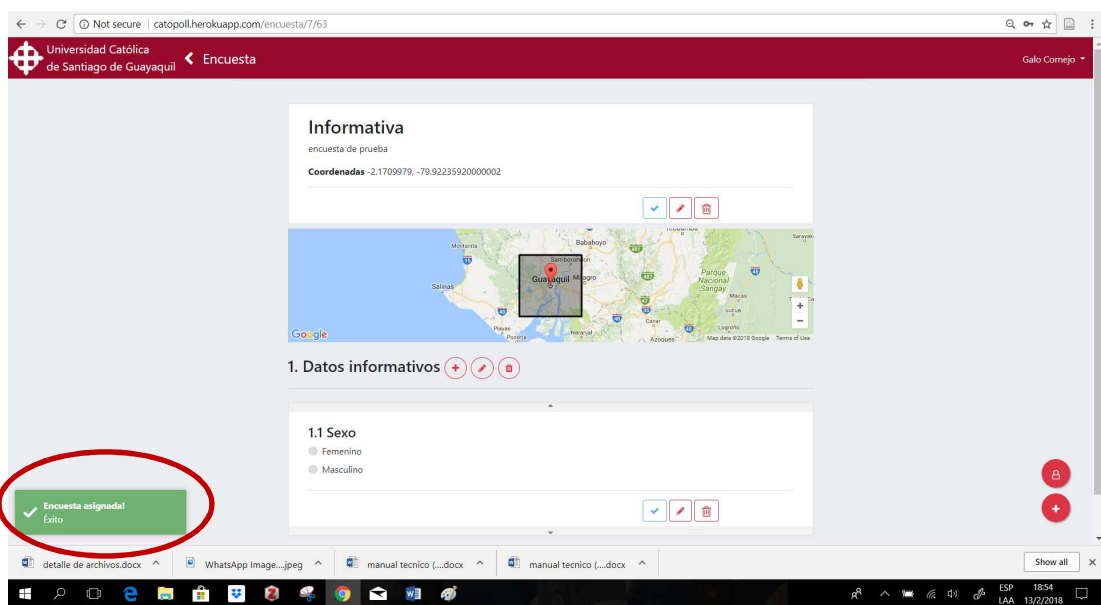
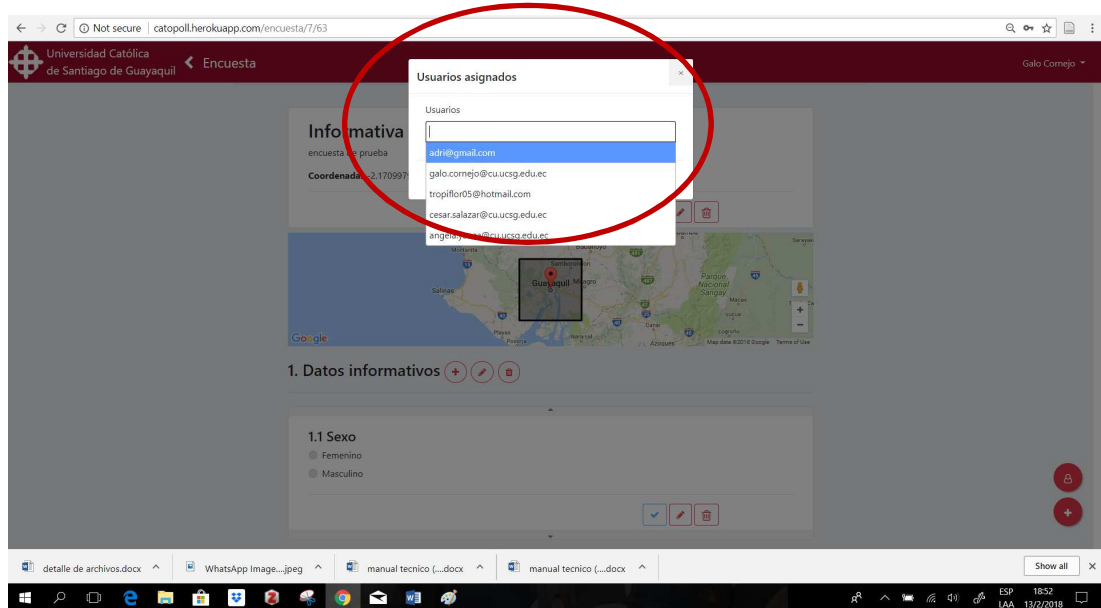
Aparece el grupo de encuestas creado, el mismo que puede ser editado o eliminado.



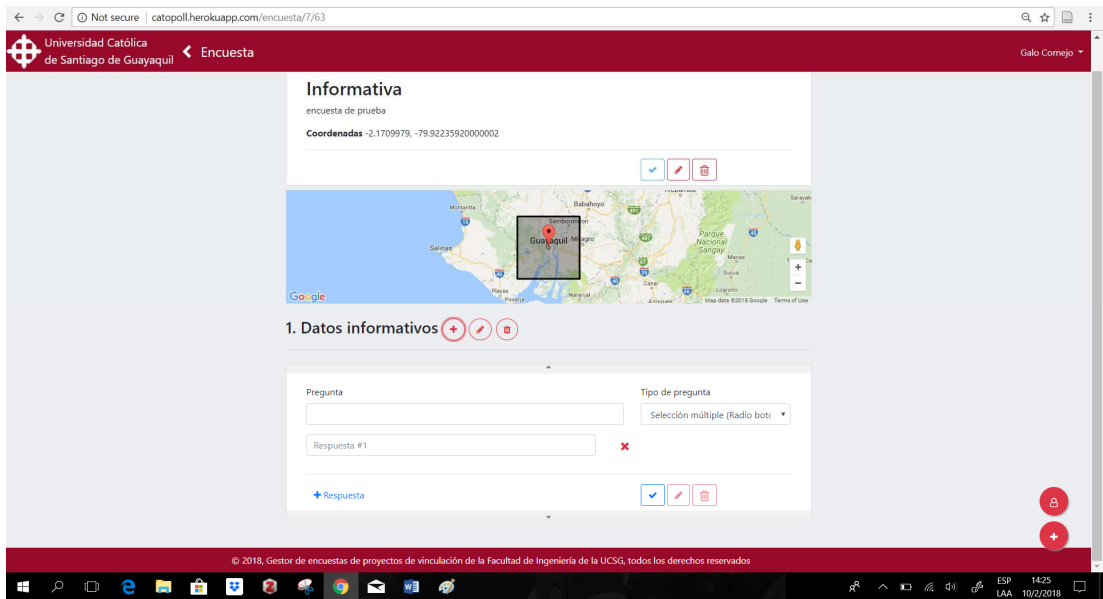
Se procede a asignar la encuesta a la persona encargada



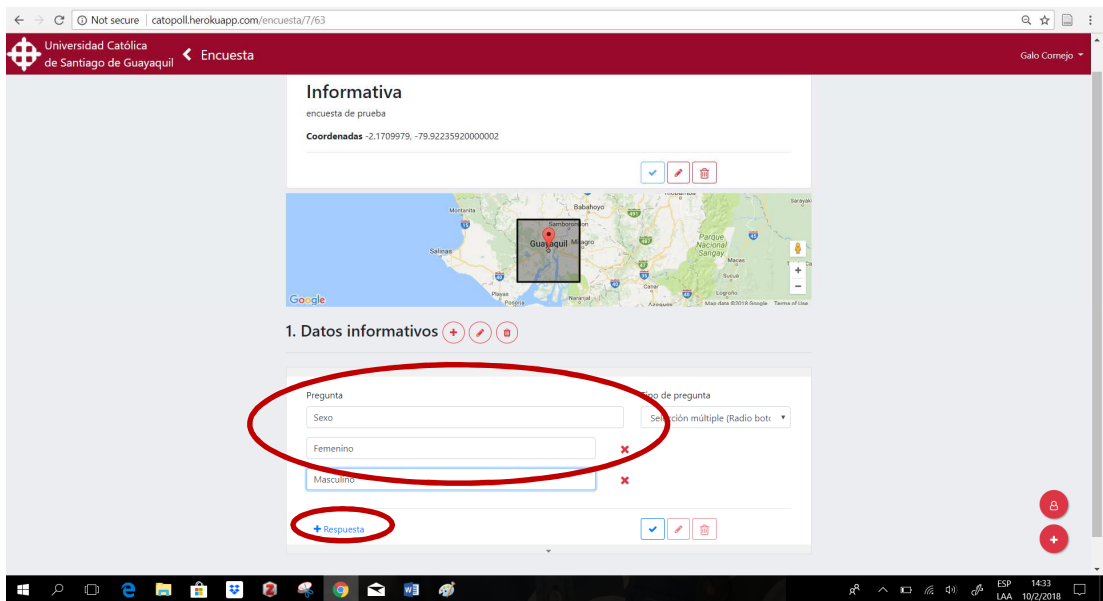
Se despliega una ventana para asignar al encuestador. Se guardan los cambios.



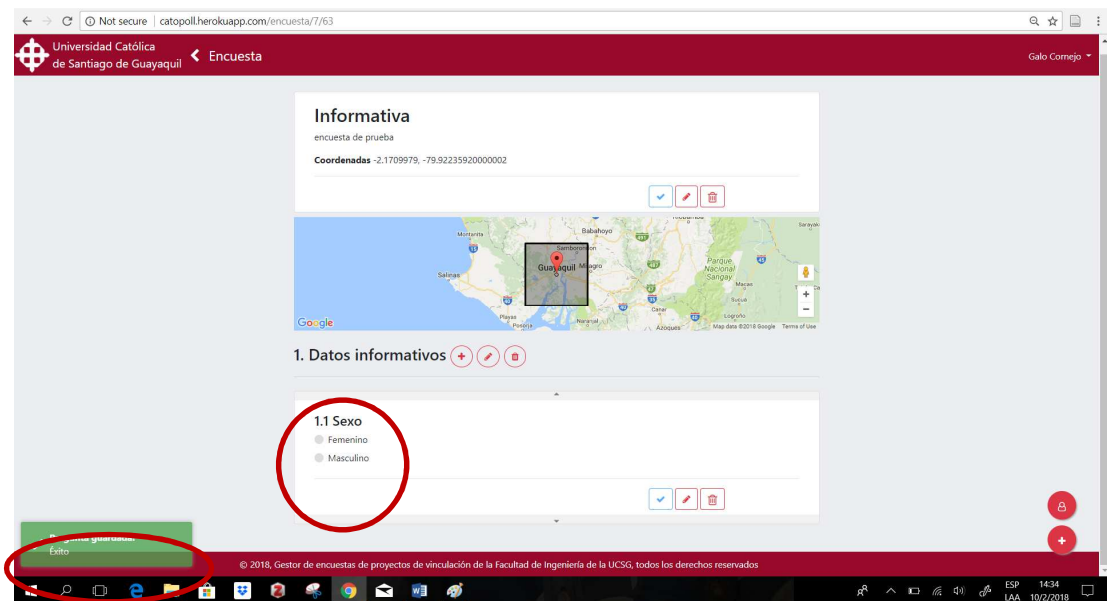
Se procede a crear los tipos de preguntas, cuando se da clic en el signo +, el mismo que despliega una ventana con cuadros de texto, que servirán para detallar la pregunta y el tipo de respuesta al tipo de pregunta.



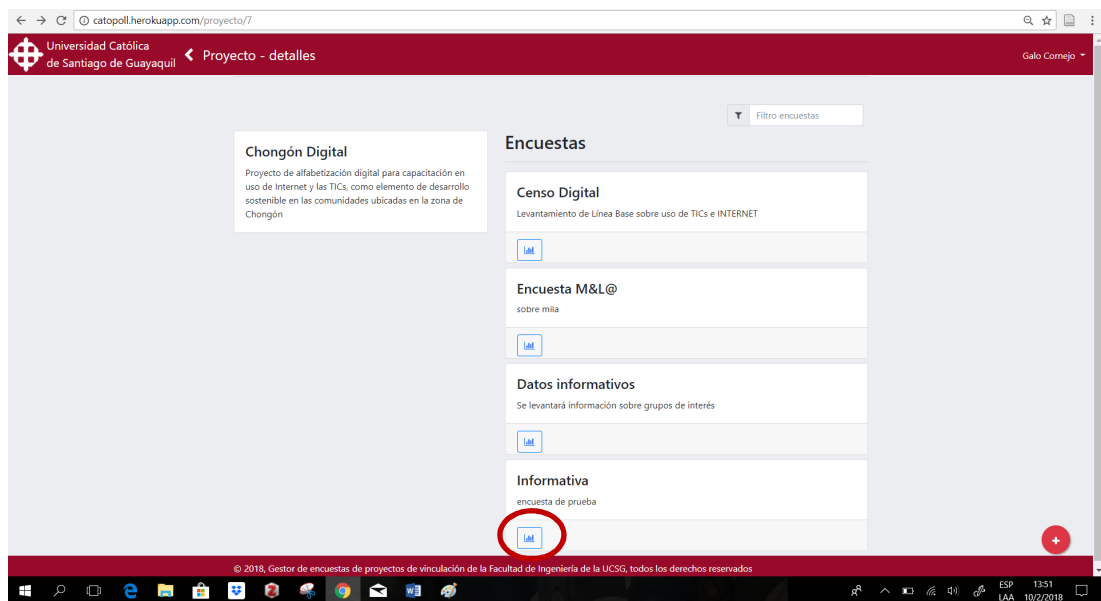
Si las respuestas son múltiples se agregan otras en la opción +Resposta y se guardan los cambios. Se pueden editar las respuestas o eliminarlas.



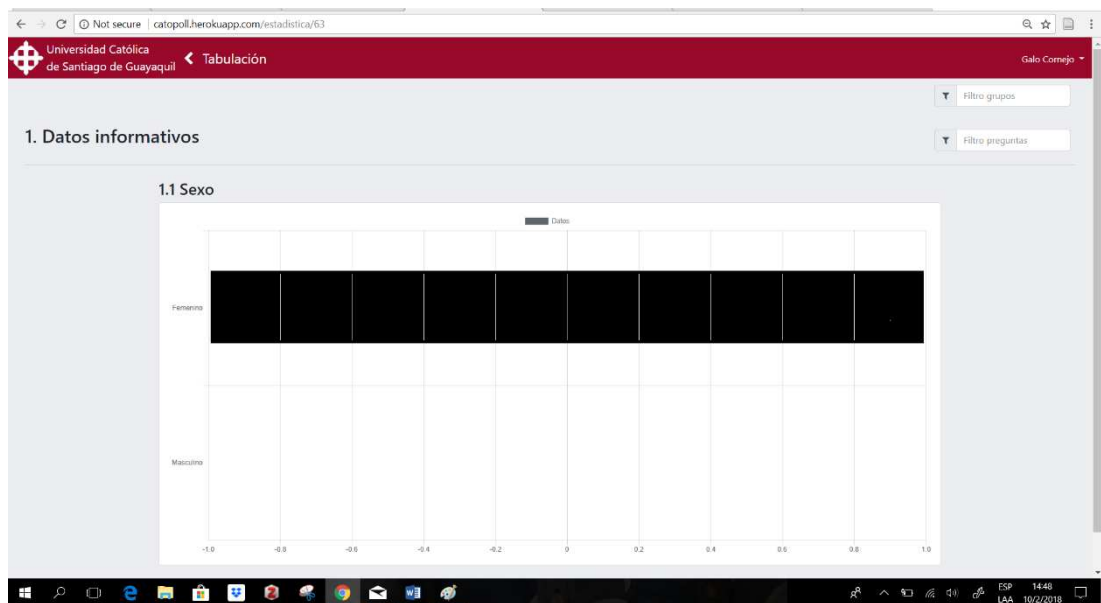
Se crea el tipo de pregunta y se muestra.



Si se necesitan revisar las estadísticas de la encuesta, se accede a la opción correspondiente, ubicada debajo de la encuesta



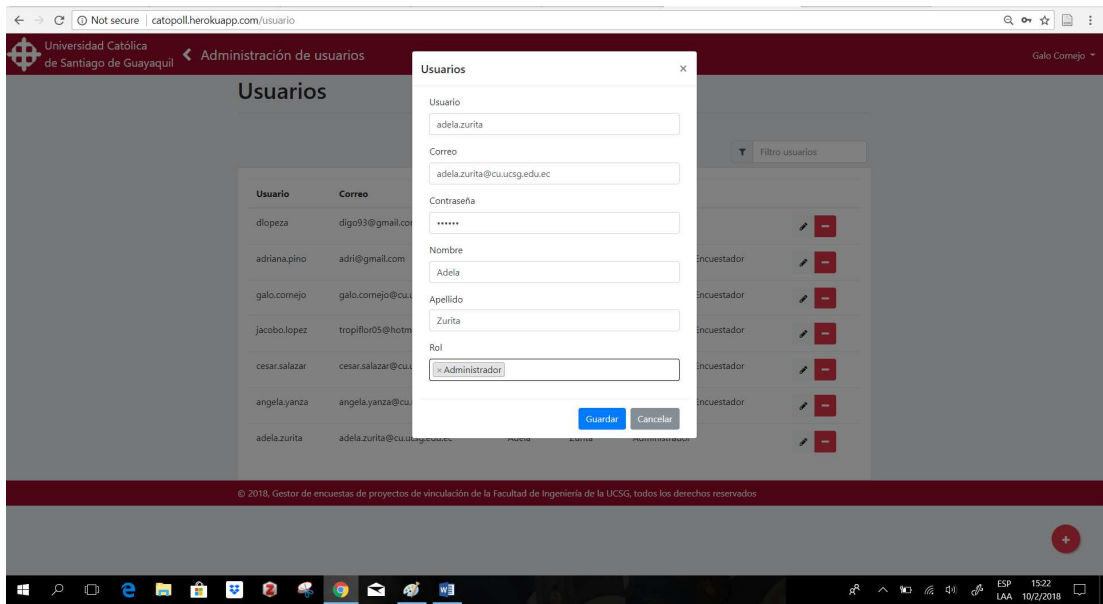
Los resultados se presentan por cada una de las preguntas de la encuesta.



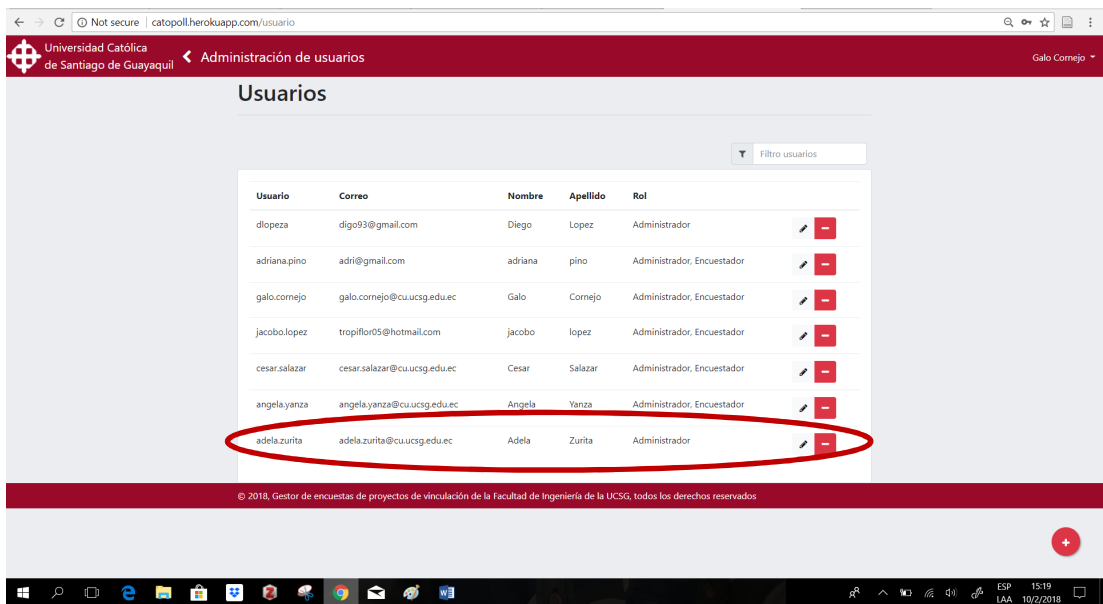
El siguiente módulo es el de Administración de usuarios, en el cual se crean los usuarios que tendrán acceso al sistema y sus roles de Administrador o Encuestador.

The screenshot shows a web application dashboard for 'Universidad Católica de Santiago de Guayaquil'. The page is titled 'Bienvenido' and contains a message: 'Te damos la bienvenida al sistema gestor de encuestas de la carrera de Ingeniería en Ciencias de la Computación.' Below the message are four menu items: 'Proyectos', 'Administración de usuarios', 'Tabulación y estadísticas', and 'Detalle de encuestas'. The 'Administración de usuarios' menu item is circled in red. The browser address bar shows 'catopoll.herokuapp.com/inicio'.

Se ingresa a la opción y se crea el usuario, al cual se le asigna su rol en la ventana que aparece con los datos más importantes a llenarse en los cuadros de texto. Para realizar esta acción, se da clic en el +. Se despliega una ventana en la cual se necesita se llenen los cuadros de texto con el nombre y apellido del usuario, correo electrónico y el rol que se le asigna: Administrador o Encuestador.



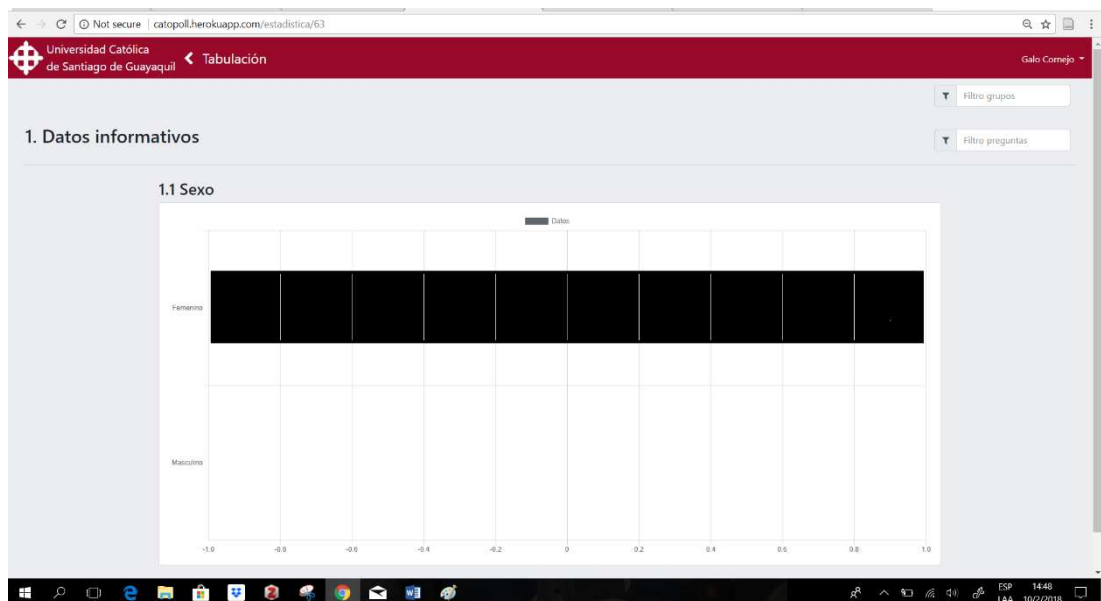
Usuario creado. En caso de requerirse, se pueden editar los datos o eliminar el usuario.



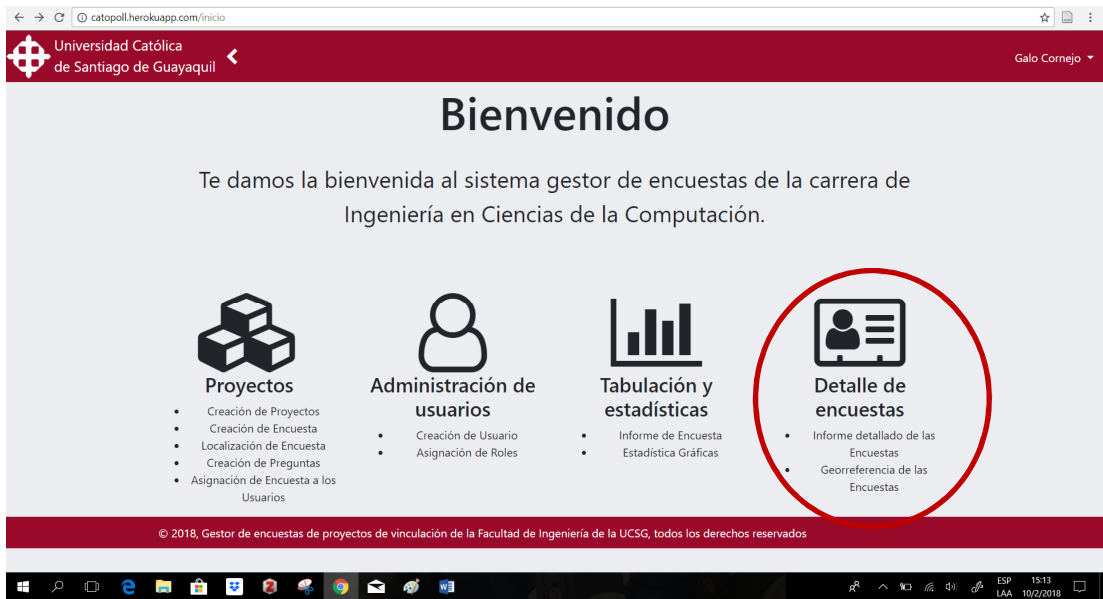
El siguiente módulo es Tabulación y estadísticas, en el cual se pueden observar todas las encuestas creada para el Proyecto.



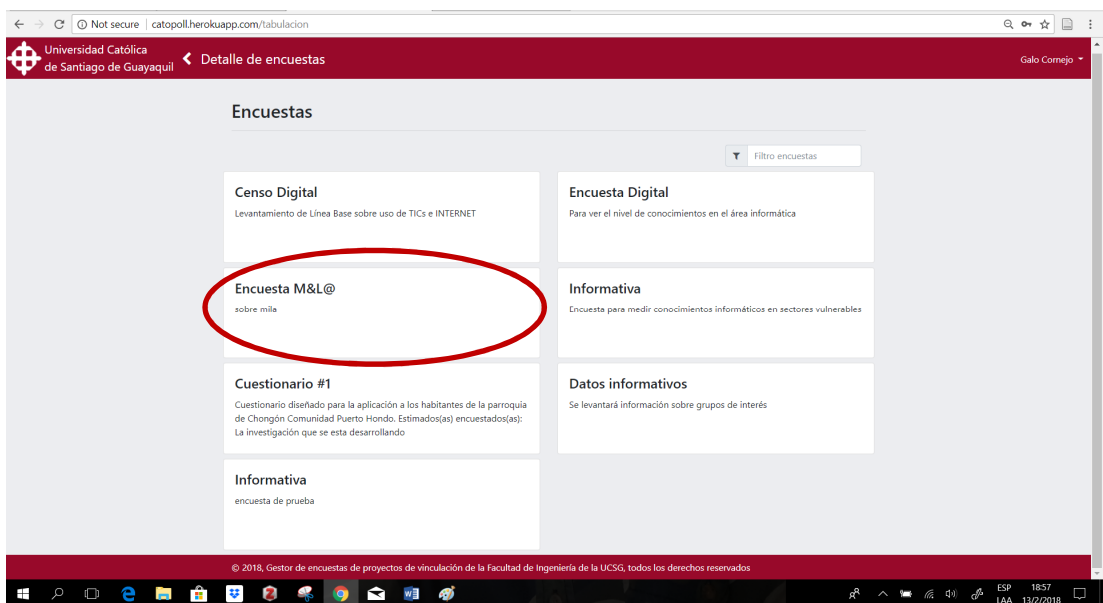
En esta opción se encuentra la especificación de las estadísticas de cada una de las preguntas de la encuesta creada para el proyecto. La pantalla es la misma que aparece cuando se accede a la opción ubicada debajo de la encuesta.



El módulo cuarto del sistema web es Detalle de encuestas, en el cual se encuentra un informe detallado de las encuestas y la georreferencia de las mismas.



Se accede al módulo y se busca la encuesta para su verificación.



Dentro de la encuesta seleccionada, aparecen todos los usuarios de la aplicación que han realizado las encuestas y el tiempo empleado en ellas. La columna Ver permite acceder a la localización de la encuesta y al detalle de las repuestas de la misma.

Universidad Católica de Santiago de Guayaquil < Detalle Galo Cornejo

Encuesta M&L@

No.	Usuario	Encuestado	Tiempo	Ver
1	adriana pino		0:52	<input checked="" type="checkbox"/>
2	adriana pino		0:17	<input type="checkbox"/>
3	Cristhian Lopez		0:46	<input type="checkbox"/>
4	adriana pino		0:48	<input type="checkbox"/>
5	adriana pino		0:29	<input type="checkbox"/>

Se muestran las respuestas de la encuesta.

Universidad Católica de Santiago de Guayaquil < Respuesta Galo Cornejo

Encuesta M&L@

sobre mila

Coordenadas -2.1593506563779705, -79.92778247326049

Encuestado:

1. Personales

1.1 Genero

- Hombre
- Mujer
- Prefiero no contestar

1.2 Edad

- 15-18
- 19-22
- mayor de 22

MANUAL DE USUARIO PARA INSTALACIÓN DEL APLICATIVO MÓVIL

Introducción

¿Cuál es el objetivo de este manual de usuario?

El objetivo de este manual es servir de apoyo al usuario final para la instalación de la aplicación móvil en el dispositivo inteligente.

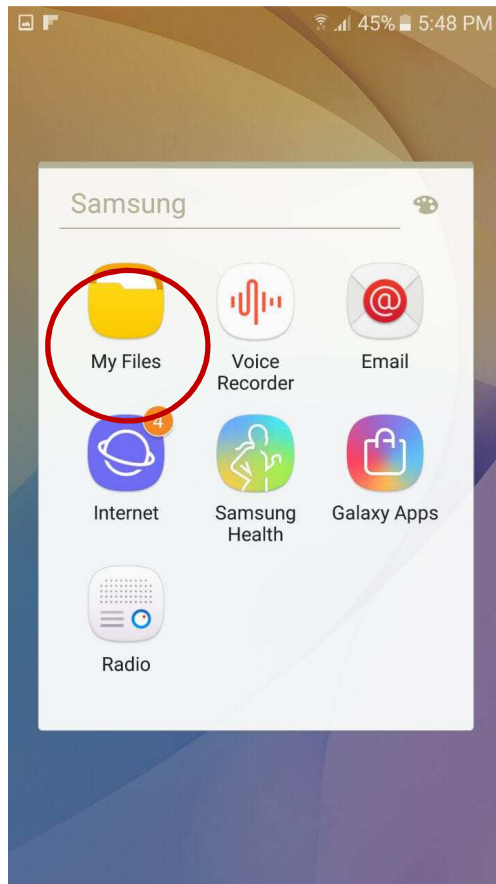
Requerimientos básicos

Se necesitan los siguientes requerimientos para el aplicativo móvil:

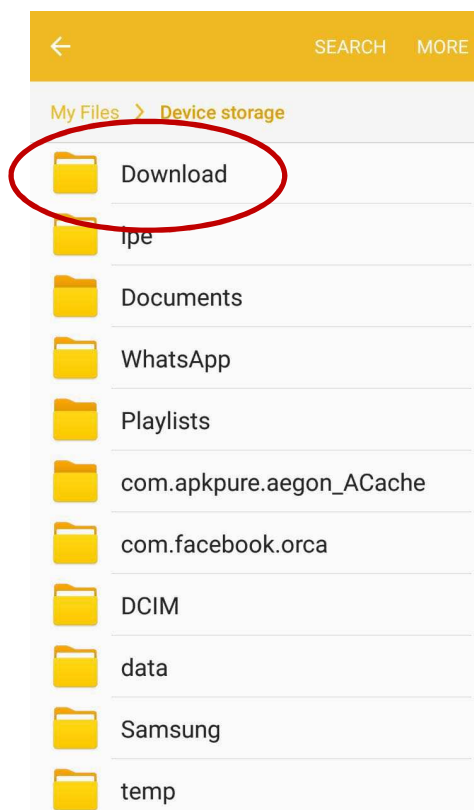
- Dispositivo móvil inteligente que soporte versiones del sistema operativo Android desde 2.3 en adelante;
- Memoria interna de 1 Mb, para la instalación del aplicativo.

Acceso a la aplicación

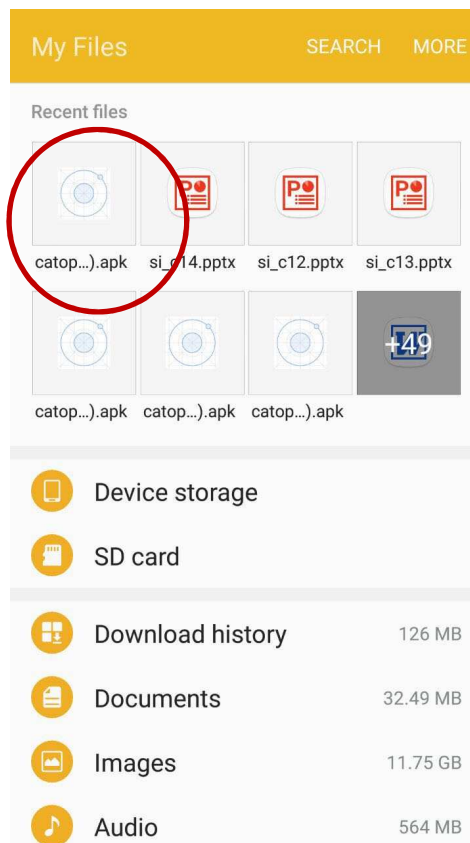
Se accede a la opción Mis archivos



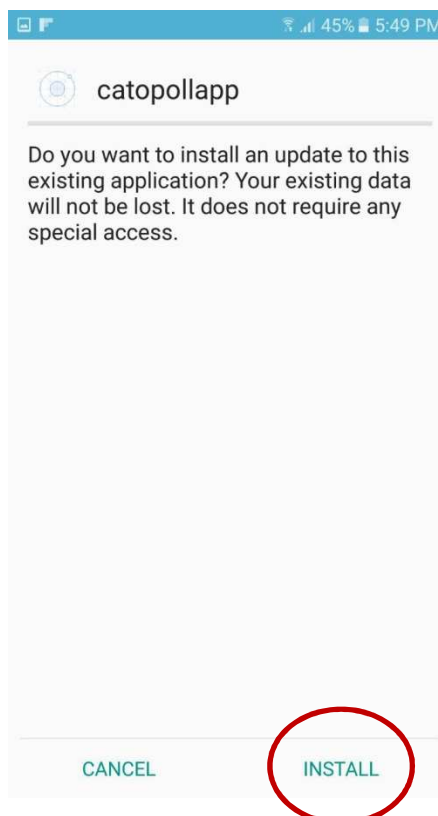
Dentro de esa opción, se accede a Descargas



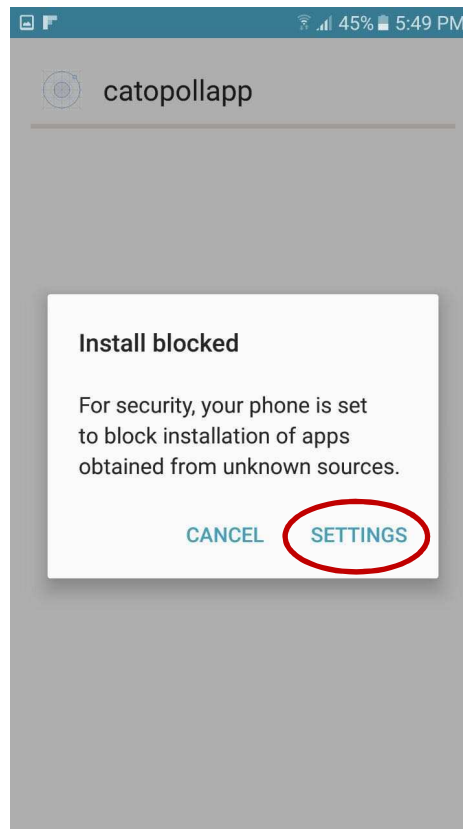
Archivos recientes. Se da clic en el archivo de la aplicación.



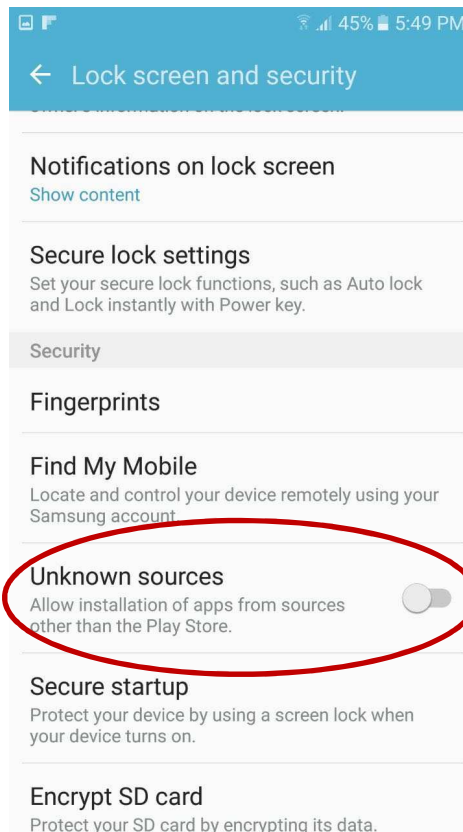
Se descarga el archivo y se procede a su instalación



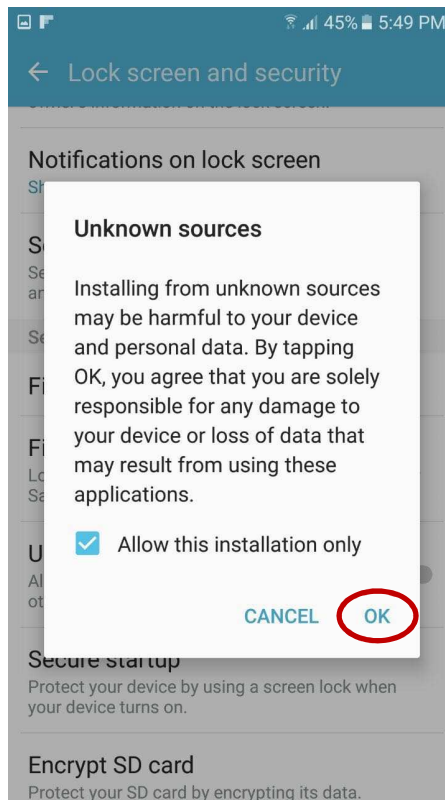
Como es una aplicación de terceros, se requiere autorización en Ajustes.



Se habilita la opción de Aplicación de terceros.



Se autoriza la instalación.



MANUAL DE USUARIO DEL SISTEMA MÓVIL PARA REALIZACIÓN DE ENCUESTAS PARA LOS PROYECTOS DE VINCULACIÓN DE LA FACULTAD DE INGENIERÍA

Introducción

¿Qué es el aplicativo móvil para realización de encuestas para proyectos de Vinculación?

Es un aplicativo para la realización de encuestas de grupos objetivo, necesarios en los proyectos de vinculación de la UCSG, que puedan ser localizadas en determinada área geográfica mediante la georreferenciación.

¿Cuál es el objetivo de este manual de usuario?

El objetivo de este manual es servir de apoyo para el usuario final para el manejo de la aplicación, permitiendo que se conozcan su funcionalidad.

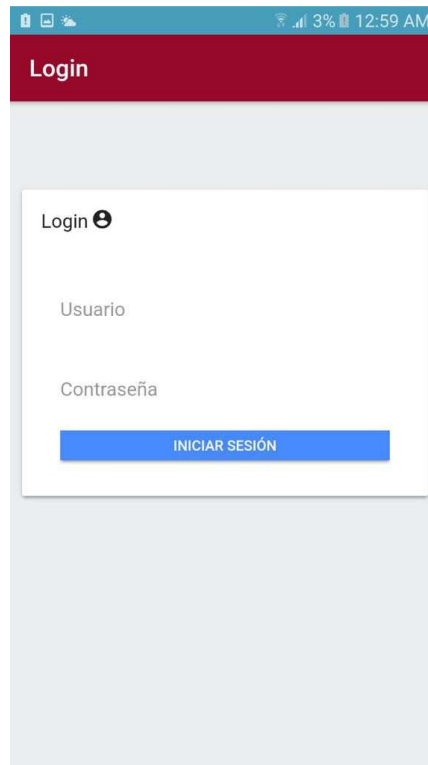
Requerimientos básicos

Requisitos de software: se debe contar con un sistema operativo (Android, IOS, Windows phone) en caso de Android que sea superior a la versión 2.3.

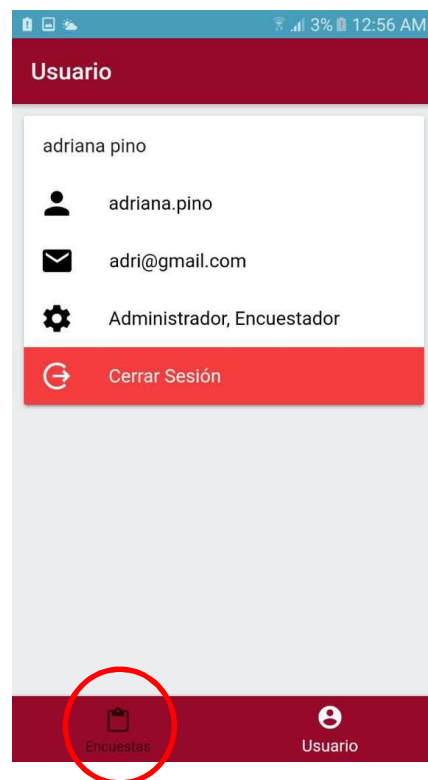
Requisitos de hardware: un teléfono inteligente que soporte las versiones indicadas en los requisitos de software.

Acceso a la aplicación

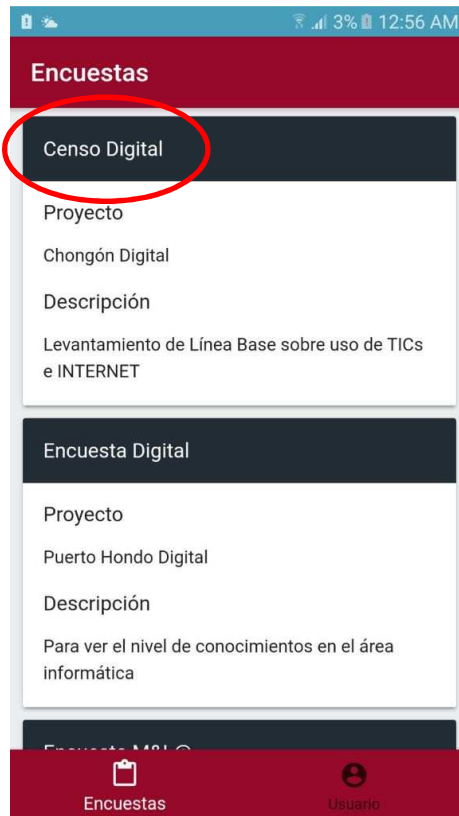
La aplicación debe estar instalada en el dispositivo inteligente para abrirla e iniciar sesión con el usuario y contraseña.



Iniciada la sesión, se presenta la información del usuario, su correo y su rol en la encuesta. Se procede a ingresar a la pantalla de las encuestas.



En la pantalla de encuestas, se procede a seleccionar la encuesta a realizar. En dicha pantalla se muestra el nombre del Proyecto y su descripción.



Se procede a realizar la encuesta y se envía la información recolectada.



Cuando se envía la información, se guarda.



En la pantalla Respuestas, se registran las respuestas de la encuesta que se han grabado. Si hay internet se guardan en la nube, si no, se almacenan en el dispositivo.



MANUAL TECNICO DEL SISTEMA WEB PARA AUTOMATIZACIÓN DE ENCUESTAS PARA LOS PROYECTOS DE VINCULACIÓN DE LA FACULTAD DE INGENIERÍA

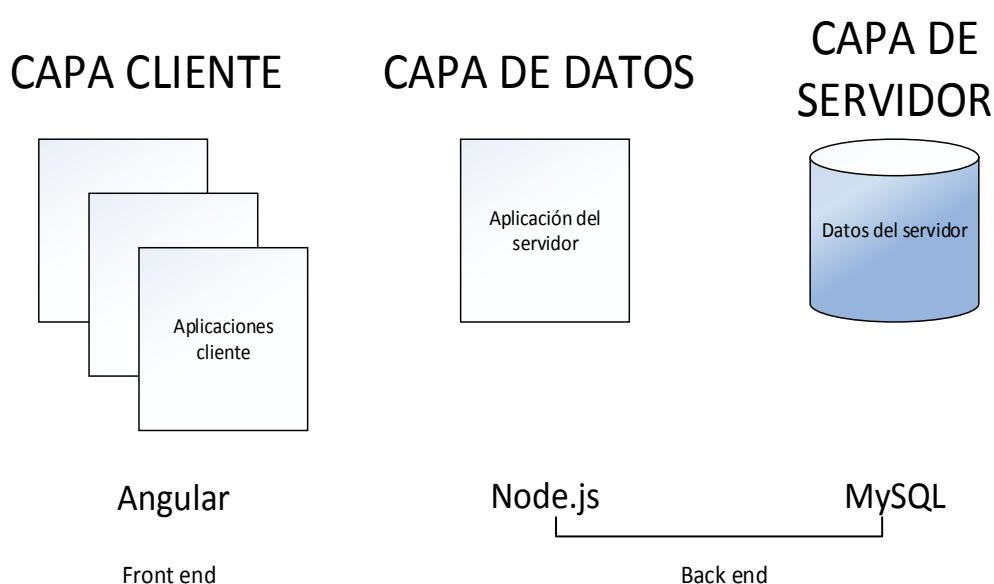
Introducción

Con la finalidad de ofrecer un mejor conocimiento al lector de la lógica con que se ha desarrollado las aplicaciones, se elaboró este documento; cabe indicarse que cada programador tiene su propio estilo desarrollo, por lo que se consideró necesaria esta documentación.

En este manual se define la arquitectura de software, el uso de las herramientas tecnológicas, instalación, configuración, su utilización en este proyecto, el desarrollo de los aplicativos web y móvil.

Este documento no pretende ser una guía de aprendizaje para el desarrollo o entendimiento de las herramientas utilizadas en esta arquitectura, sino documentar su aplicación en desarrollo del sitio. Para un mayor detalle de la utilización de alguna herramienta o aprendizaje de la misma se recomienda consultar con el autor.

Arquitectura de software



Instalación de aplicaciones

Para el levantamiento del ambiente de desarrollo se deben instalar ciertos programas, siendo necesario contar con acceso a internet para la descarga de instalador. Cabe mencionarse que todas las herramientas aplicadas para este proyecto son de la línea open source.




Se inicia con el levantamiento del entorno de desarrollo para aplicaciones web con Angular, para esto se necesita instalar Node.js, la misma que incluye NPM (Node Package Manager), necesaria para instalar dependencias tanto del entorno de desarrollo como dependencias del aplicativo en sí. También se requiere instalar Git, herramienta utilizada en el desarrollo para el manejo de versiones; además se instalará el editor de texto Atom y el framework Ionic para el desarrollo del aplicativo móvil. A su vez se requiere de la instalación del motor de base de datos Mysql y un cliente para ejecutar los scripts de base de datos, en este caso Mysql Workbench.

Instalación del Node.js

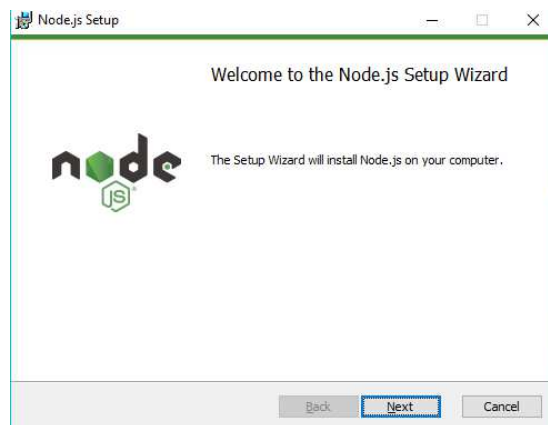
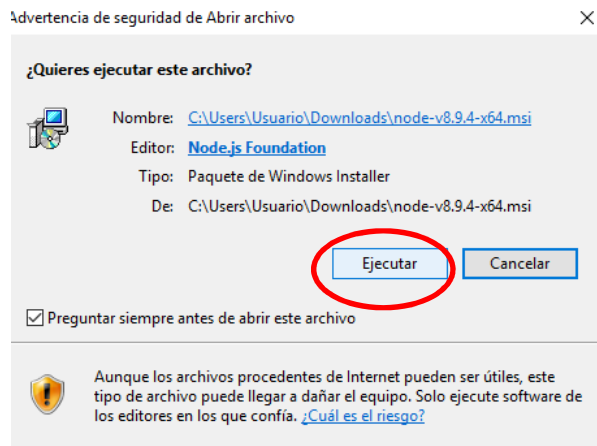
Para instalar Node.js se accede directamente a esta ruta: <https://nodejs.org/en/download/>

Downloads
Latest LTS Version: 8.9.4 (includes npm 5.6.0)

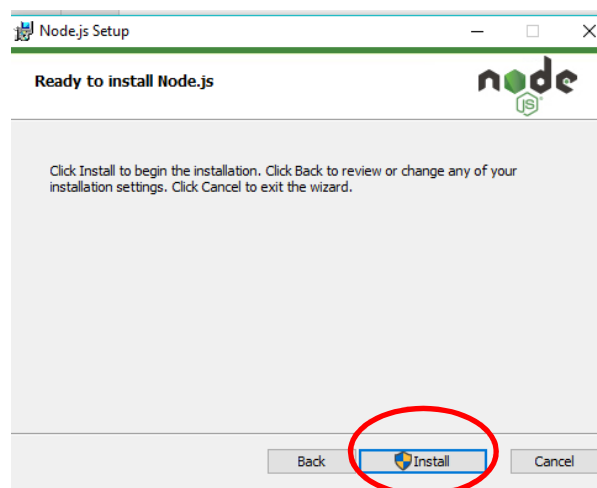
Download the Node.js source code or a pre-built installer for your platform, and start developing today.

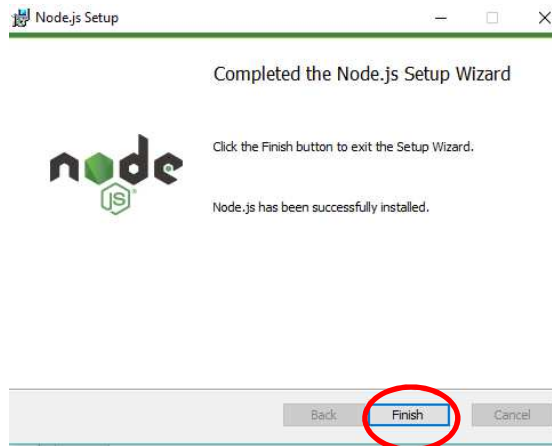
LTS Recommended For Most Users	Current Latest Features		
 Windows Installer <small>node-v8.9.4-win10.msi</small>	 macOS Installer <small>node-v8.9.4.pkg</small>	 Source Code <small>node-v8.9.4.tar.gz</small>	
Windows Installer (.msi)	32-bit	64-bit	
Windows Binary (.zip)	32-bit	64-bit	
macOS Installer (.pkg)	64-bit		
macOS Binary (.tar.gz)	64-bit		
Linux Binaries (x86/x64)	32-bit	64-bit	
Linux Binaries (ARM)	ARMv6	ARMv7	ARMv8
Source Code	node-v8.9.4.tar.gz		

Después de descargar el instalador, se continúa con su instalación.



Después de seguir con las configuraciones predeterminada para la instalación del Nodejs se da clic en instalar y se finaliza la instalación.





El siguiente paso es abrir la terminal e instalar todas las dependencias en la siguiente ruta.

```
C:\Users\Usuario\Desktop\CATOPOLL> npm install
```

```
C:\Users\User\Desktop\CATOPOLL\catopollapp\src>cd ..
C:\Users\User\Desktop\CATOPOLL\catopollapp>npm install
> node-sass@4.5.3 install C:\Users\User\Desktop\CATOPOLL\catopollapp\node_modules\node-sass
> node scripts/install.js
Downloading binary from https://github.com/sass/node-sass/releases/download/v4.5.3/win32-x64-57_binding.node
Download complete
Binary saved to C:\Users\User\Desktop\CATOPOLL\catopollapp\node_modules\node-sass\vendor\win32-x64-57_binding.node
Caching binary to C:\Users\User\AppData\Roaming\npm-cache\node-sass\4.5.3\win32-x64-57_binding.node
> uglifyjs-webpack-plugin@0.4.6 postinstall C:\Users\User\Desktop\CATOPOLL\catopollapp\node_modules\uglifyjs-webpack-plugin
> node lib/post_install.js
> node-sass@4.5.3 postinstall C:\Users\User\Desktop\CATOPOLL\catopollapp\node_modules\node-sass
> node scripts/build.js
Binary found at C:\Users\User\Desktop\CATOPOLL\catopollapp\node_modules\node-sass\vendor\win32-x64-57_binding.node
Testing binary
Binary is fine
added 595 packages in 224.078s
```

Se pone a correr el servicio con el siguiente comando.

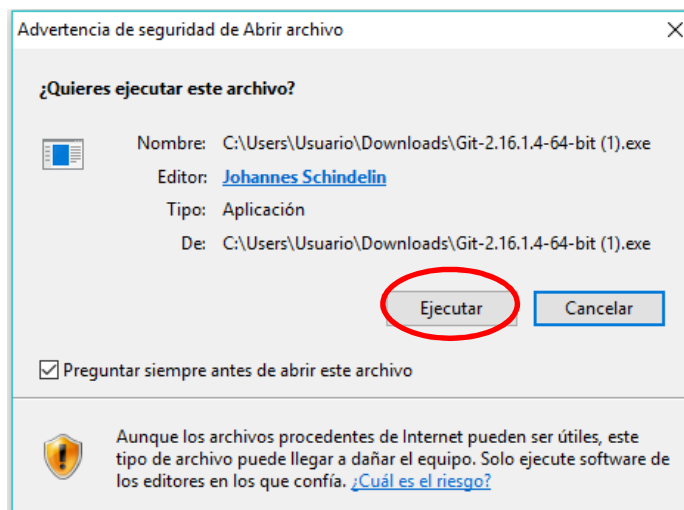
```
C:\Users\Usuario\Desktop\CATOPOLL> npm run start:dev
```

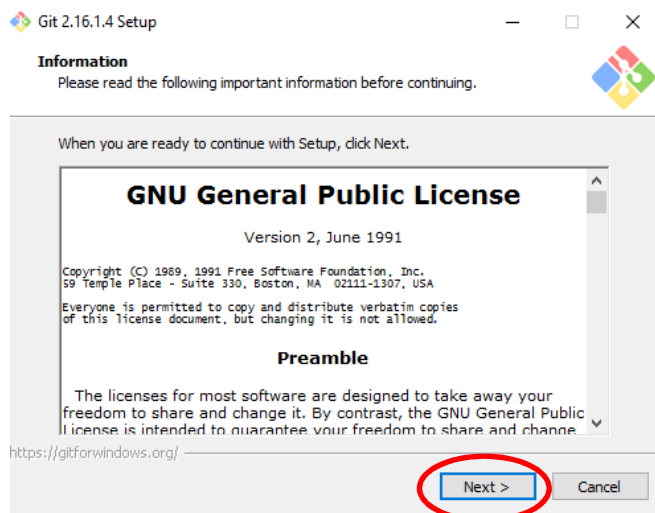
```
ca. Símbolo del sistema
49% building modules 328/385 modules 57 active ..ode_modules\node_modules\moment\locale\X-p
49% building modules 328/386 modules 58 active ..les\core-js\modules\property
49% building modules 329/386 modules 57 active ..les\core-js\modules\property
49% building modules 330/386 modules 56 active ..les\core-js\modules\property
49% building modules 331/386 modules 55 active ..les\core-js\modules\property
49% building modules 332/386 modules 54 active ..les\core-js\modules\property
49% building modules 333/386 modules 53 active ..les\core-js\modules\property
50% building modules 334/386 modules 52 active ..les\core-js\modules\property
50% building modules 335/386 modules 51 active ..les\core-js\modules\property
50% building modules 336/386 modules 50 active ..les\core-js\modules\property
50% building modules 337/386 modules 49 active ..les\core-js\modules\property
50% building modules 338/386 modules 48 active ..les\core-js\modules\property
50% building modules 339/386 modules 47 active ..les\core-js\modules\property
50% building modules 340/386 modules 46 active ..les\core-js\modules\property
51% building modules 341/386 modules 45 active ..les\core-js\modules\property
51% building modules 342/386 modules 44 active ..les\core-js\modules\property
51% building modules 343/386 modules 43 active ..les\core-js\modules\property
51% building modules 344/386 modules 42 active ..les\core-js\modules\property
51% building modules 345/386 modules 41 active ..les\core-js\modules\property
51% building modules 346/386 modules 40 active ..les\core-js\modules\property
51% building modules 347/386 modules 39 active ..les\core-js\modules\property
51% building modules 348/386 modules 38 active ..les\core-js\modules\property
51% building modules 349/386 modules 37 active ..les\core-js\modules\property
52% building modules 350/386 modules 36 active ..les\core-js\modules\property
52% building modules 351/386 modules 35 active ..les\core-js\modules\property
52% building modules 352/386 modules 34 active ..les\core-js\modules\property
52% building modules 353/386 modules 33 active ..les\core-js\modules\property
52% building modules 354/386 modules 32 active ..les\core-js\modules\property
52% building modules 355/386 modules 31 active ..les\core-js\modules\property
52% building modules 356/386 modules 30 active ..les\core-js\modules\property
52% building modules 357/386 modules 29 active ..les\core-js\modules\property
52% building modules 358/386 modules 28 active ..les\core-js\modules\property
53% building modules 359/386 modules 27 active ..les\core-js\modules\property
53% building modules 360/386 modules 26 active ..les\core-js\modules\property
53% building modules 361/386 modules 25 active ..les\core-js\modules\property
53% building modules 362/386 modules 24 active ..les\core-js\modules\property
53% building modules 363/386 modules 23 active ..les\core-js\modules\property
53% building modules 363/387 modules 24 active ..poll\node_modules\moment\loca
```

Instalación del Git

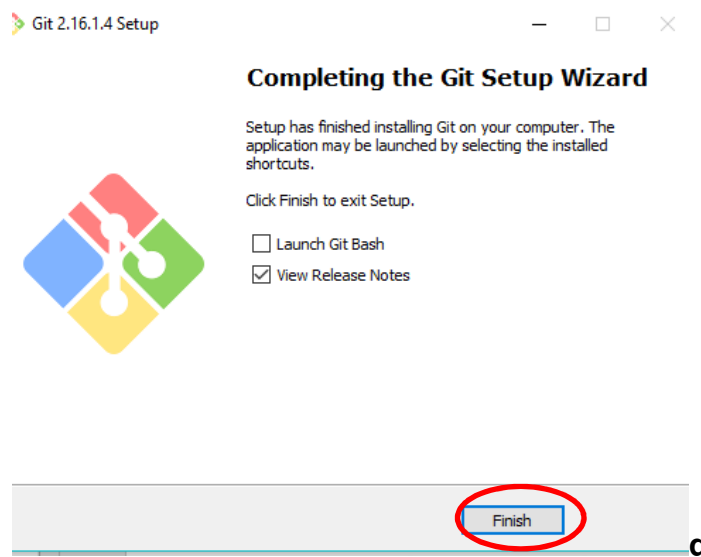
Para instalar Git se accede directamente a esta ruta: <https://git-scm.com/downloads>

Después de descargar el instalador se continúa con la instalación del mismo y se ejecuta el archivo. Se siguen las instrucciones de instalación.





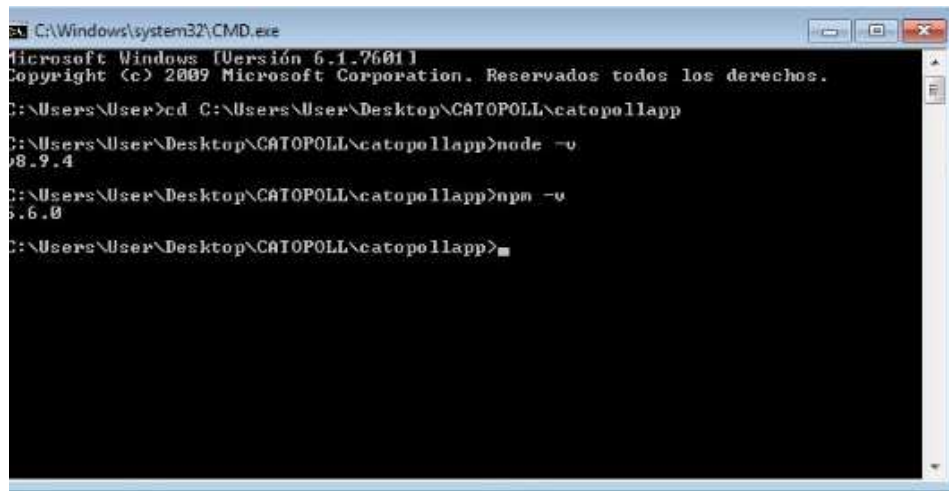
Después de seguir con las configuraciones predeterminadas para la instalación del Nodejs se da clic en instalar. Se finaliza la instalación.



Después de terminar de instalar Git, se comprueba que todo esté instalado y con las últimas versiones. Se lo realiza con el siguiente

```
C:\Users\Usuario\Desktop\CATOPOLL> node -v
```

comando:



```
C:\Windows\system32\CMD.exe
Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\User>cd C:\Users\User\Desktop\CATOPOLL\catopollapp
C:\Users\User\Desktop\CATOPOLL\catopollapp>node -v
v8.9.4
C:\Users\User\Desktop\CATOPOLL\catopollapp>npm -v
5.6.0
C:\Users\User\Desktop\CATOPOLL\catopollapp>
```

El siguiente paso es poner los repositorios dentro del proyecto con el

```
git clone https://github.com/angular/quickstart my-proj cd CATOPOLL
```

comando

Dentro de la ruta de carpeta del proyecto se coloca el siguiente comando para poder instalar el manejador de paquetes de nodejs dentro del proyecto.

```
C:\Users\Usuario\Desktop\CATOPOLL> npm install
```

Se sigue con el comando

```
C:\Users\Usuario\Desktop\CATOPOLL> npm start
```

El comando npm start primero compila la aplicación, luego vuelve a compilar y ejecuta el servidor lite. Tanto el compilador como el servidor miran los cambios de archivos.

Con esto queda levantado lo servicios para el futuro desarrollo del aplicativo.

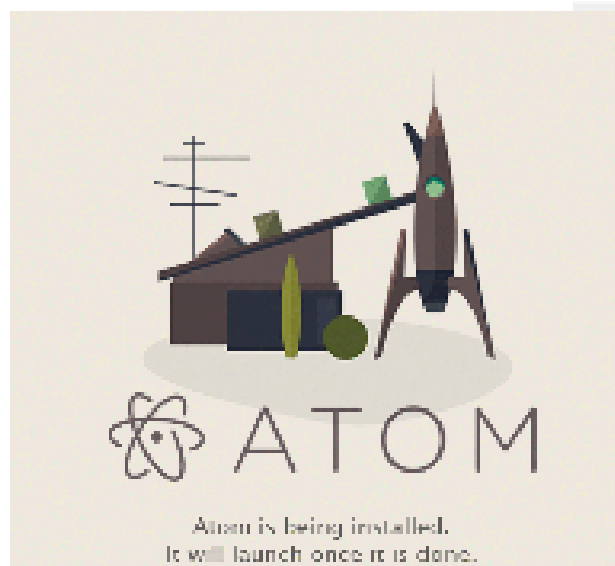
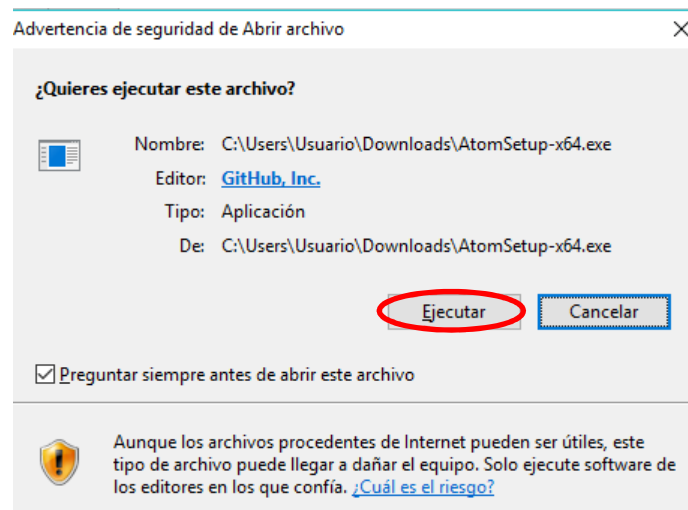
Instalación de Atom

Atom es un editor de código de fuente de código abierto para Windows con soporte para plug-ins escritos en Node.js y control de versiones Git integrado, desarrollado por GitHub.

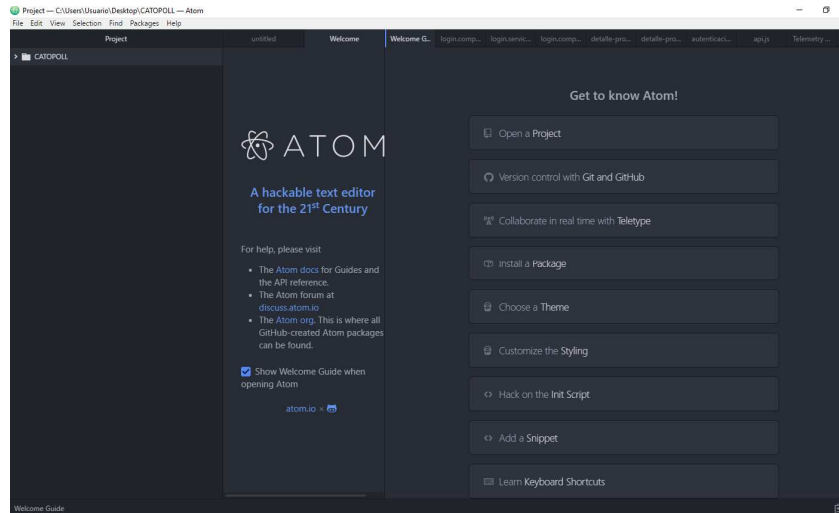
Se usará este editor de texto para el desarrollo del aplicativo web y móvil. Cabe incluir que cualquier editor de texto hubiera servido para el desarrollo de la aplicación; si se escogió este editor fue por el soporte que tiene con el Node.js.

Para instalar atom podemos ir directamente a esta ruta: <https://atom.io/>

Después de descargar el instalador, se continúa con su instalación.



Después de seguir con las configuraciones predeterminadas para la instalación del Atom, se puede iniciar el programa.



Instalación del IONIC

Se digita el siguiente comando en la terminal:

```
C:\Users\Usuario\Desktop\CATOPOLL>npm install -g ionic  
[.....] | fetchMetadata: sill resolveWithNewModule ar
```

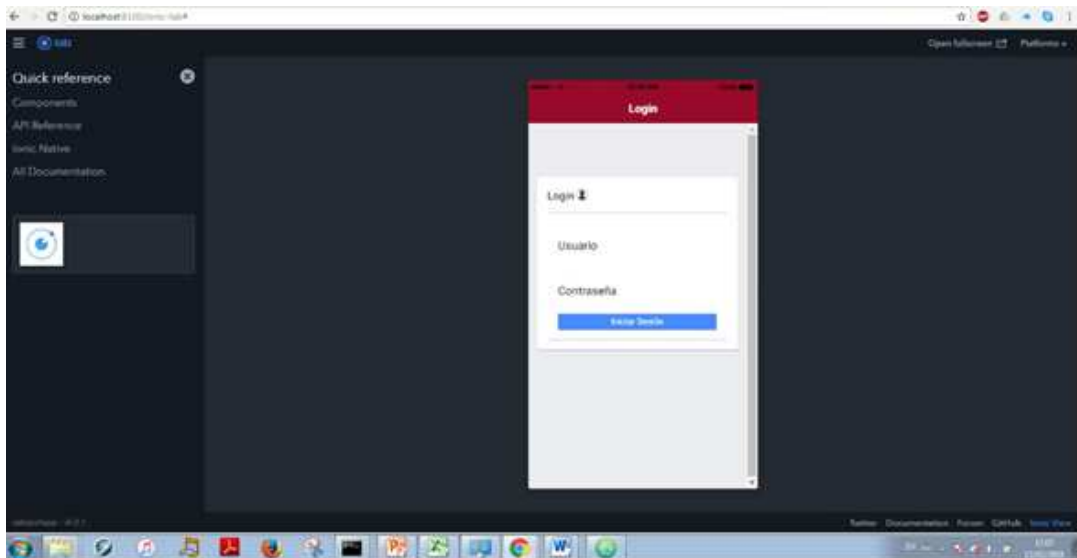
Cuando termine la instalación

```
+ ionic@3.19.1  
added 253 packages in 32.142s  
C:\Users\Usuario\Desktop\CATOPOLL>
```

Se comprueba que se instaló correctamente con el siguiente comando:

```
C:\Users\Usuario\Desktop\CATOPOLL>ionic -v  
3.19.1  
C:\Users\Usuario\Desktop\CATOPOLL>
```

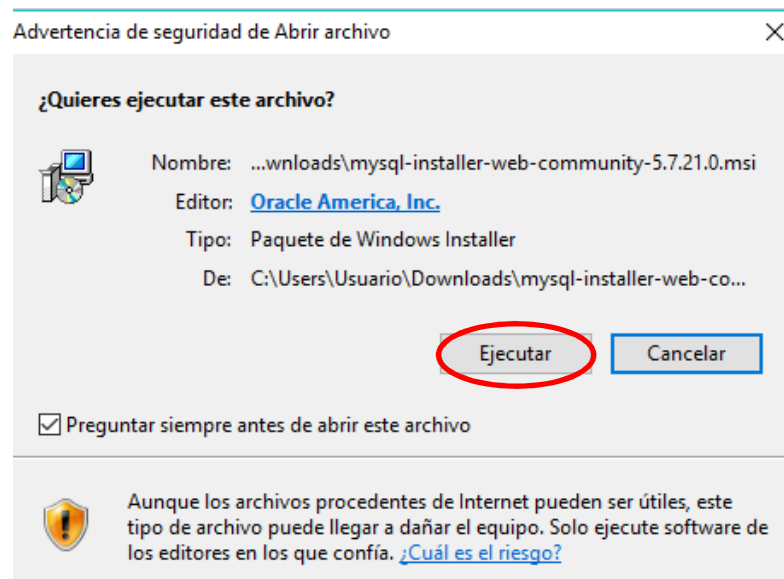
El ambiente IONIC para el aplicativo móvil se lo visualiza a continuación:



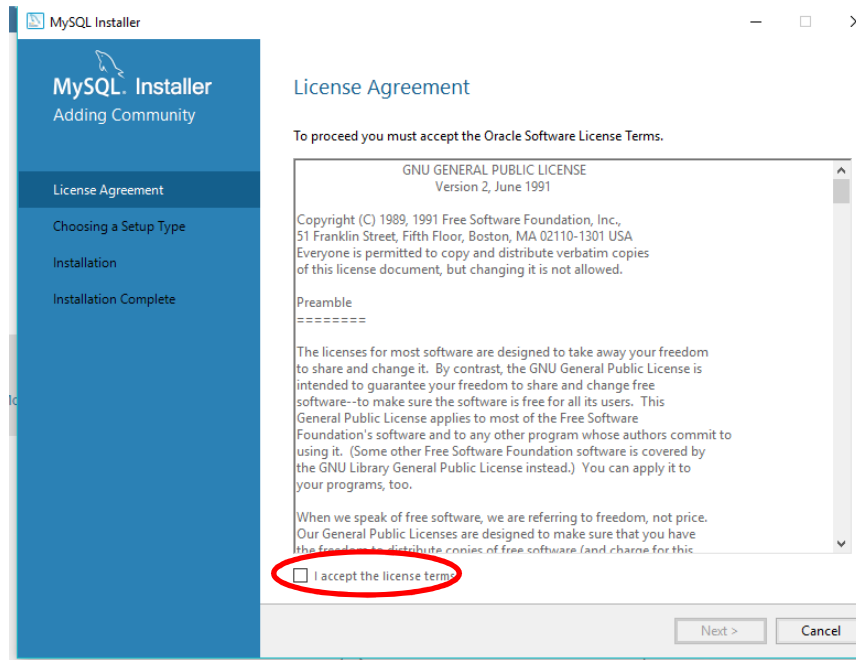
Instalación de Mysql y Workbench

Se puede descargar el instalador de Mysql en:

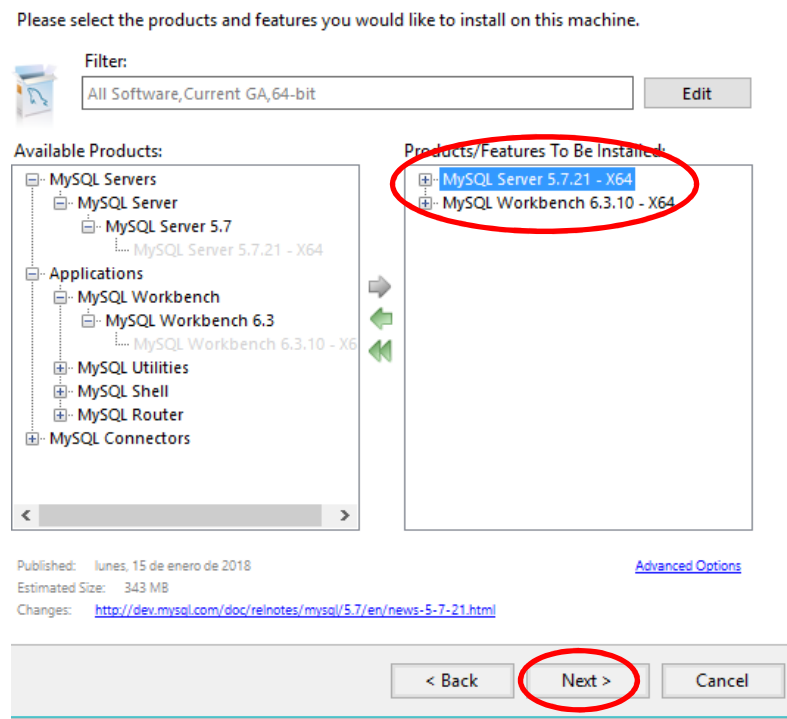
<https://dev.mysql.com/downloads/file/?id=474802>



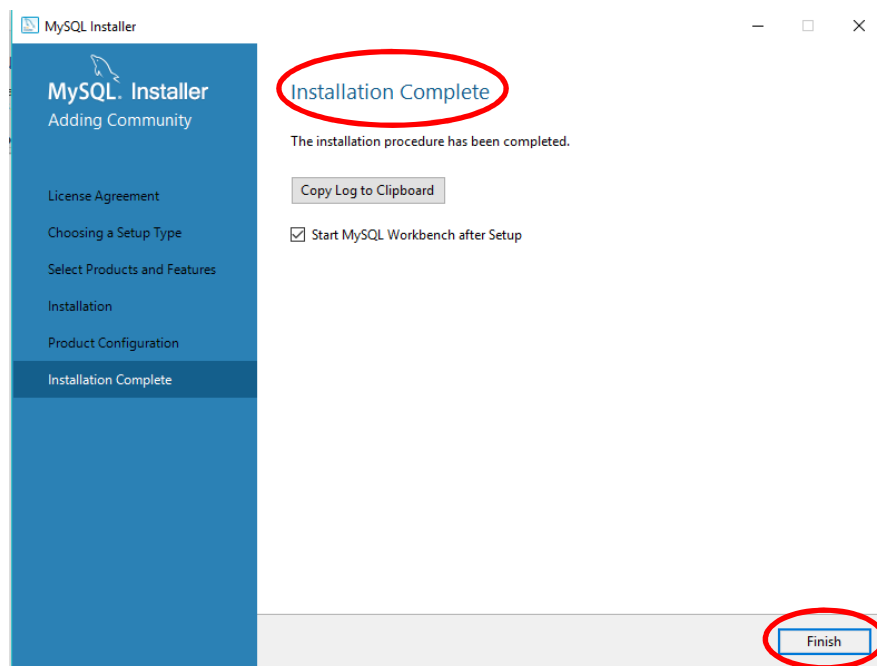
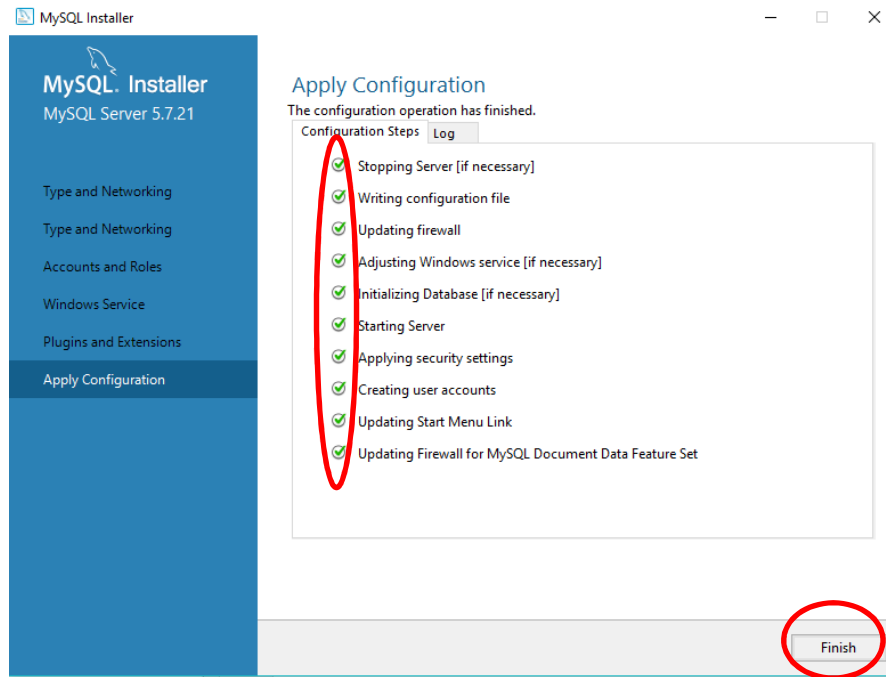
Se continúa con la configuración respectiva para su correcta instalación. Se aceptan las condiciones de la licencia.



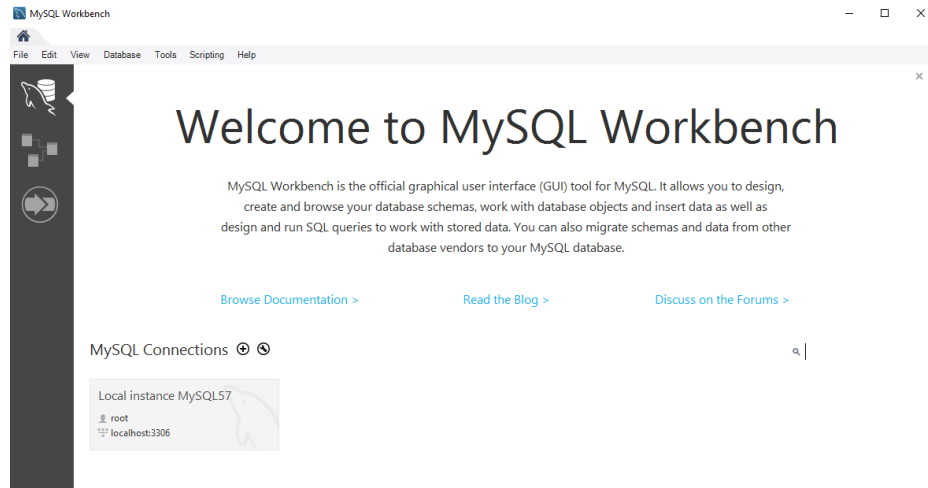
Se seleccionan las dos herramientas que se va a necesitar.



Después de terminar con las descargas, se continúa con la configuración de Mysql. Se finaliza la instalación.



Una vez finalizada la instalación y configuración predeterminada, se podrá utilizar el motor de base de datos Mysql y para la ejecución de los scripts el Workbench.



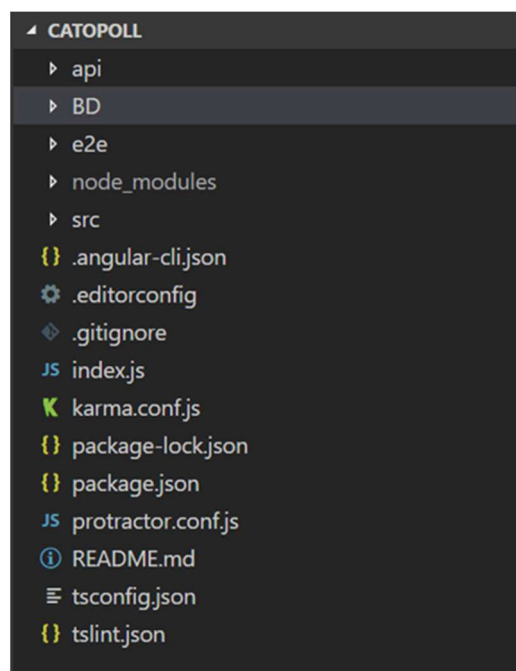
De esta manera queda levantado el ambiente para el desarrollo de los aplicativos web y móvil.

Funcionamiento del aplicativo web

Para describir el funcionamiento del aplicativo web, se requiere definir y especificar el sistema de archivos que se va a utilizar.

Sistema de archivos

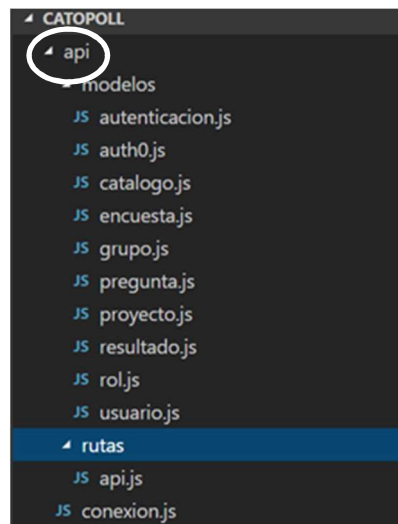
Se enlistan todos los archivos y carpetas que corresponden al código fuente del sistema vistos desde la raíz.



A continuación se explican los elementos más importantes del sistema de archivos.

Api

La carpeta api/ es propia del backend del sistema, básicamente es por donde pasa la comunicación del frontend con la base de datos y viceversa.



BD

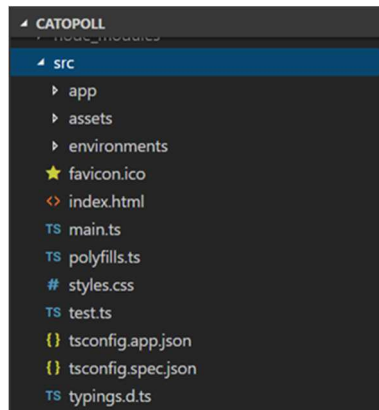
La carpeta BD/ contiene los scripts básicos de base de datos, los scripts de creación de tablas, de procedimientos almacenados e insert de catálogos base para el funcionamiento del sistema.

node_modules

Contiene todas las dependencias usadas para el funcionamiento del aplicativo (Bootstrap, jQuery, Expressjs, Mysql, etc) y se instalan al momento de realizar el **npm install**.

src

Corresponde a archivos exclusivos de Angular; en estos directorios y subdirectorios se encuentran las vistas (archivos html y css), los controladores, servicios y modules (todos archivos TypeScript y componentes funcionales del framework).



index.js

En este archivo es donde se encuentran todas las configuraciones correspondiente al servidor ExpressJs que va a alojar el sitio (se detallará en el siguiente apartado).

package.json

Este archivo es propio de un repositorio de Node Package Manager y se enlistan todas las configuraciones del mismo así como sus dependencias. Cabe destacar que las dependencias listadas aquí son las que se instalan al momento de ejecutar el comando “npm install”.

Configurando el servidor (ExpressJs) – archivo index.js

La página web usa ExpressJs como servidor. En este apartado se visualiza el código necesario para la configuración de este servidor.

```
index.js 1.98 KB
1  const express = require('express');
2  const path = require('path');
3  const http = require('http');
4  const bodyParser = require('body-parser');
5
6  var cors = require('cors');
7
8  // Se configuran las rutas del archivo api
9  const api = require('./api/rutas/api');
10
11 //Con esta función forzamos a que el tipo de conexión se https
12 const forceSSL = function () {
13   return function (req, res, next) {
14     if (req.headers['x-forwarded-proto'] !== 'https') {
15       return res.redirect(
16         ['https://', req.get('Host'), req.url].join('')
17       );
18     }
19     next();
20   }
21 }
22
23
```

```

24 const app = express();
25
26 var conexion = require('./api/conexion');
27
28 //Configuramos para que el servidor acepte CORS - Cross-origin Request
29 //Que nos va a servir para poder acceder al api desde el app móvil
30 app.use(cors());
31
32 //Configuramos la función forceSSL para el protocolo Https
33 app.use(forceSSL());
34
35 //Configuramos la librería body-parser que nos va ayudar en las llamadas get y post que utilizamos
36 app.use(bodyParser.json());
37 app.use(bodyParser.urlencoded({ extended: false }));
38
39 //Servimos el directorio dist que es donde se aloja el fuente de Angular 2+ ya compilado
40 //Practicamente es la estructura de archivos que va a ser leída desde el cliente (navegador)
41 app.use(express.static(path.join(__dirname, 'dist')));
42
43 // configuramos la ruta /api
44 app.use('/api', api);

```

```

45
46 //Redirigimos las peticiones raíz hacia nuestro index.html
47 //donde estará nuestro SPA
48 app.get('*', (req, res) => {
49   res.sendFile(path.join(__dirname, 'dist/index.html'));
50 });
51
52 //Configuramos el puerto, en caso de que el puerto no este definido por una variable de ambiente
53 //el puerto será 5000
54 const port = process.env.PORT || '5000';
55 app.set('port', port);
56
57 //Iniciamos la conexión hacia la base de datos
58 conexion.init();
59
60 //Creamos el servidor tipo Express con todas las configuraciones hechas
61 const server = http.createServer(app);
62
63 //Ponemos el servidor a escuchar y enviamos un mensaje por consola
64 server.listen(port, () => console.log(`La magia sucede en el puerto:${port}`));

```

El código está comentado de tal manera que se describe paso a paso lo que se está haciendo. A continuación, un breve repaso.

De la línea 1-6 se definen las dependencias que se utilizarán para la configuración del servidor. Desde la línea 12-21 se define una rutina para forzar la utilización de un protocolo Https y un certificado de seguridad tipo SSL. En la línea 24 se instancia expressjs, y a partir de aquí se va a realizar toda la configuración correspondiente al servidor.

En la línea 33 se llama a la rutina definida anteriormente para la utilización del protocolo Https y el certificado SSL; si en el ambiente no se tiene un certificado instalado se puede omitir esta línea.

Se comienza la configuración en la línea 6, que es donde se configura CORS. Esto se lo hace para que los diferentes tipos de llamadas que se van a hacer a la API puedan ser accedidas desde fuera del servidor, es decir para que sean accedidas desde el aplicativo móvil.

En la línea 36-37 se configura la librería body-parser que es la que se utilizará para capturar y enviar las respuestas tipo REST (para este sistema se utilizará solo get y post).

En la línea 44 se configuran las rutas definidas en el archivo /api/rutas/api.js.

En las líneas 54-55 se configura el puerto en el que va a escuchar el servidor, si el puerto no está configurado en una variable de ambiente por defecto escuchará en el puerto 5000.

En la línea 61 se crea el servidor a partir de las configuraciones hechas y finalmente en la línea 64 el servidor se pone a escuchar las peticiones.

Enrutamiento de backend

En el apartado anterior se detalló la configuración general del servidor y se mencionaron ciertas consideraciones acerca de “rutas”, específicamente se especificó que en la línea 44 se configuraron las rutas. Es importante tener un mayor detalle de esta configuración.

```
43 // configuramos la ruta /api
44 app.use('/api', api);
45
46 //Redirigimos las peticiones raíz hacia nuestro index.html
47 //donde estará nuestro SPA
48 app.get('*', (req, res) => {
49   res.sendFile(path.join(__dirname, 'dist/index.html'));
50 });
```

En estas líneas se están configurando las rutas que va a manejar la aplicación; en la línea 48-50 se programa el servidor para que cada vez que haya una petición tipo GET devuelva el archivo index.html. Se hace este proceso debido a que las aplicaciones de Angular son SPA (Single Page Application), prácticamente la página master de la aplicación se encuentra en este archivo (index.html) y de aquí se enrutan a todas las subpáginas contenidas en el frontend.

Mientras, en la línea 44, se configura de tal manera que todas las peticiones que contengan después de la url base “/api” (Ejemplo: “http://localhost:5000/api/catalogo/all/fa”) llamen a las rutas de la API (se detallará más a fondo esto en el siguiente apartado).

Archivo api/rutas/api.js

El archivo api.js contiene todas las rutas a las cuales el frontend llamará según la función a realizar. Como se especificó antes para estas llamadas se utilizaron dos tipos: GET y POST. A continuación un ejemplo sencillo de ambas.

```
41 // USUARIOS
42 router.get('/usuarios/all/:params', (req, res) => {
43     usuario.getUsuarios(req.params.params, res);
44 });
45 router.get('/usuarios/porid/:params', (req, res) => {
46     usuario.getPorId(req.params.params, res);
47 });
48 router.post('/usuarios/', (req, res) => {
49     usuario.mantenimiento(req.body, res);
50 });
```

En el código anterior se observan tres peticiones (dos tipo get y una tipo post). El concepto que sigue este archivo es: si la petición cumple con el tipo (get o post) y la ruta, se procede a realizar las acciones que están dentro del arrow function, para el caso de las tres rutas, todas realizan una acción definida por la clase **usuario**. Se aprecia mejor con la tabla a continuación:

Rutabase/subruta	Tipo	Subruta	Acción a realizar
http://localhost:5000/api/ usuarios/all/1	GET	/usuarios/all/:params	usuario.getUsuarios(req.params.params, res);
http://localhost:5000/api/ usuarios/porid/2	GET	/usuarios/porid/:params	usuario.getPorId(req.params.params, res);
http://localhost:5000/api/ usuarios	POST	/usuarios	usuario.mantenimiento(req.body, res):

Seguridad basada en Token (JWT)

Para poder proteger las rutas explicadas anteriormente se utilizó una seguridad por **token**, con la utilización de una librería llamada Json Web Token. Bajo los lineamientos de trabajo que se sigue con esta librería se dice que se tienen dos tipos de rutas: rutas públicas y rutas privadas.

Las rutas públicas van a poder acceder sin ningún tipo de control y para las privadas siempre se va a pedir un token de seguridad de sesión, es decir cada vez que se quiera acceder a estas rutas las peticiones deberán enviar un token que fue previamente generado.

```

52 //...FIN DE RUTAS PÚBLICAS
53 // ---B R O K E R---
54 router.use(function (req, res, next) {
55
56     // Obtenemos el token desde el header de la petición
57     var token = req.headers['x-access-token'];
58
59     if (token) {
60
61         jwt.verify(token, secret, function (err, decoded) {
62             if (err) {
63                 return res.json({ success: false, err: -1, mensaje: 'Error al decodificar token.' });
64             } else {
65                 req.decoded = decoded;
66                 next();
67             }
68         });
69     } else {
70         return res.status(403).send({
71             success: false,
72             mensaje: 'No token provided.'
73         });
74     }
75 });
76 // INICIO DE RUTAS PRIVADAS ...

```

En el archivo `api/rutas/api.js` a partir de la línea 54 hasta la 75 se programó lo anteriormente citado. Este código corresponde a un middleware, es decir que todas las peticiones que se hacen a las rutas privadas deben pasar por aquí, algo similar a una aduana. Definir rutas públicas y privadas es algo tan simple como poner las rutas antes del middleware si se quiere que estas sean públicas, o poner las rutas después del middleware si se requiere que estas sean privadas.

Dicho lo anterior, se va a explicar la lógica más a detalle. Cuando se hace una llamada a una ruta privada se captura el token que llega en el header de la petición (línea 57), luego se valida si el token existe (línea 59); en caso de que no exista, se envía un error diciendo que el token no fue enviado y en caso de que exista lo que se debe hacer es decodificarlo y validarlo (esta lógica es propia de la librería JWT). En caso de que la verificación sea correcta, se procede a realizar la acción que corresponde a la ruta; caso contrario devuelve un mensaje de error.

Funcionamiento del frontend

Se ha explicado en apartados anteriores que el frontend se basa en una arquitectura SPA. Básicamente solo se carga una página master y sobre esta página se renderizan otras subpáginas según el requerimiento del

usuario. Para saber qué subpágina se debe mostrar, se tienen que definir rutas. Estas rutas están dadas por el archivo `src/app/rutas/app.routing.ts`.

app.routing.ts

En este archivo se define la navegación del frontend; aquí se determinan las rutas y los componentes que se deben renderizar, cuando la ruta del navegador coincida con la definida para cada componente.

```
17 const appRoutes: Routes = [  
18   // RUTAS PRIVADAS  
19   { path: 'usuario', component: UsuarioComponent, canActivate: [RouteActivatorService] },  
20   { path: 'inicio', component: IndexComponent, canActivate: [RouteActivatorService] },  
21   { path: 'proyecto', component: ProyectoComponent, canActivate: [RouteActivatorService] },  
22   { path: 'proyecto/:id', component: DetalleProyectoComponent, canActivate: [RouteActivatorService] },  
23   { path: 'encuesta/:idproyecto/:idencuesta', component: EncuestaComponent, canActivate: [RouteActivatorService] },  
24   { path: 'estadistica', component: EstadisticaComponent, canActivate: [RouteActivatorService] },  
25   { path: 'estadistica/:idencuesta', component: EstadisticaDetalleComponent, canActivate: [RouteActivatorService] },  
26   { path: 'tabulacion', component: TabulacionComponent, canActivate: [RouteActivatorService] },  
27   { path: 'tabulacion/:idencuesta', component: TabulacionDetalleComponent, canActivate: [RouteActivatorService] },  
28   { path: 'respuesta/:idencuesta/:idrespuesta', component: RespuestaComponent, canActivate: [RouteActivatorService] },  
29  
30   // RUTAS PÚBLICAS  
31   { path: '', component: LoginComponent },  
32   { path: '404', component: Page404Component },  
33  
34   // SI LA RUTA NO COINCIDE CON NINGUNA DE LAS ANTERIORES SE LLAMA AL LOGIN  
35   { path: '**', redirectTo: '' }  
36 ];  
37  
38 export const routing = RouterModule.forRoot(appRoutes);  
39
```

Todas las rutas se enlistan en un arreglo de objetos del tipo “Route”. Un objeto tipo “Route” recibe tres parámetros:

Parámetro	Tipo	Descripción	Obligatorio
Path	String	La ruta url en el navegador	Si
Component	Component	La clase tipo componente donde está la página que se quiere renderizar	Si
canActivate	Service	Es el service que se ejecuta cada vez que se quiere acceder a esta página, para este caso es aquí donde se valida la sesión y que un usuario no pueda ingresar si no ha hecho inicio de sesión	No

Las rutas públicas no van a ser validadas por el service `RouteActivatorService` mientras que las privadas si pasarán por esta validación por seguridad.

Services

Los services son clases que Angular recomienda usar para manejar datos, y realizar operaciones comunes que puedan ser utilizadas desde varios componentes. Por ejemplo, en el punto anterior se decía que se utilizó el service `RouteActivatorService` para validar la sesión cada vez que se accede a una página privada; esta validación no es propia de una página en particular, la misma validación se la utilizó en todas las páginas privadas, debido a esto esta validación se la usó en un service.

Funcionamiento base del aplicativo

Angular es un framework de frontend que maneja una arquitectura MVC (Modelo Vista Controlador), es decir que para cada pantalla del aplicativo web se tendrán siempre estos tres elementos para su funcionamiento básico.

A continuación, se va a explicar de una manera muy técnica el funcionamiento básico de las funcionalidades más importante del sistema.

Login

Frontend	
Archivo	Descripción
<code>login.component.html</code>	Contiene la estructura html de la vista
<code>login.component.css</code>	Contiene los estilos css propios del componente
<code>login.component.ts</code>	Modelo y controlador de la vista
<code>login.service.ts</code>	Service que será utilizado desde el controlador para la llamada a nuestra ruta API REST

Backend	
Archivo	Descripción
<code>api/rutas/api.js</code>	Donde se aloja la ruta llamada desde el <code>login.service.ts</code>
<code>api/modelos/autenticacion.js</code>	Modelo de acceso a datos, archivo que contiene las rutinas que son ejecutadas después

Una vez listados todos los archivos que intervienen en el proceso de inicio de sesión, se va a mostrar el código de cada archivo, el que se encuentra relacionado con el proceso de Login.

Usuario

Frontend	
Archivo	Descripción
usuario.component.html	Contiene la estructura html de la vista
usuario.component.css	Contiene los estilos css propios del componente
usuario.component.ts	Modelo y controlador de la vista
usuario.service.ts	Service que será utilizado desde el controlador para la llamada a nuestra ruta API REST

Backend	
Archivo	Descripción
api/rutas/api.js	Donde se aloja la ruta llamada desde el usuario.service.ts
api/modelos/usuario.js	Modelo de acceso a datos, archivo que contiene las rutinas que son ejecutadas después

Estos son los archivos que intervienen en la creación y administración de usuario

Home

Frontend	
Archivo	Descripción
home.component.html	Contiene la estructura html de la vista
home.component.css	Contiene los estilos css propios del componente
home.component.ts	Modelo y controlador de la vista
home.service.ts	Service que será utilizado desde el controlador para la llamada a nuestra ruta API REST

Backend	
Archivo	Descripción
No aplica debido a que esta página no requiere de una consulta a la base de datos	

Estos son los archivos que intervienen en el funcionamiento de la página principal del sistema, por la cual se podrá acceder a las demás páginas del aplicativo.

Encuesta

Frontend	
Archivo	Descripción
encuesta.component.html	Contiene la estructura html de la vista
encuesta.component.css	Contiene los estilos css propios del componente
encuesta.component.ts	Modelo y controlador de la vista
encuesta.service.ts	Service que será utilizado desde el controlador para la llamada a nuestra ruta API REST

Backend	
Archivo	Descripción
api/rutas/api.js	Donde se aloja la ruta llamada desde el encuesta.service.ts
api/modelos/encuesta.js	Modelo de acceso a datos, archivo que contiene las rutinas que son ejecutadas para las consultas y mantenimientos.
api/modelos/grupo.js	Modelo de acceso a datos, archivo que contiene las rutinas que son ejecutadas para las consultas y mantenimientos de los grupos de preguntas en la encuesta.
api/modelos/pregunta.js	Modelo de acceso a datos, archivo que contiene las rutinas que son ejecutadas para las consultas y mantenimientos de las preguntas en la encuesta.
api/modelos/sector.js	Modelo de acceso a datos, archivo que contiene las rutinas que son ejecutadas para las consultas y mantenimientos de los sectores en la encuesta.

Estos son los archivos que intervienen en la creación de las encuestas: tipo de preguntas, grupos de preguntas, asignación de encuestadores y sectores de encuesta.

Proyectos

Estos son los archivos que intervienen en la creación de Proyectos en los cuales dentro se crearan las encuestas.

Estadística

Frontend	
Archivo	Descripción
estadistica.component.html	Contiene la estructura html de la vista
estadistica.component.css	Contiene los estilos css propios del componente
estadistica.component.ts	Modelo y controlador de la vista
estadistica.service.ts	Service que será utilizado desde el controlador para la llamada a nuestra ruta API REST
Backend	
Archivo	Descripción
api/rutas/api.js	Donde se aloja la ruta llamada desde el estadistica.service.ts
api/modelos/ resultado.js	Modelo de acceso a datos, archivo que contiene las rutinas que son ejecutadas para la consultas en la página y generación de encuestas desde el aplicativo.

Estos son los archivos que intervienen en la creación de los marcos gráficos para la generación de estadísticas y para enviar una encuesta desde el aplicativo móvil.

Funcionamiento del aplicativo móvil

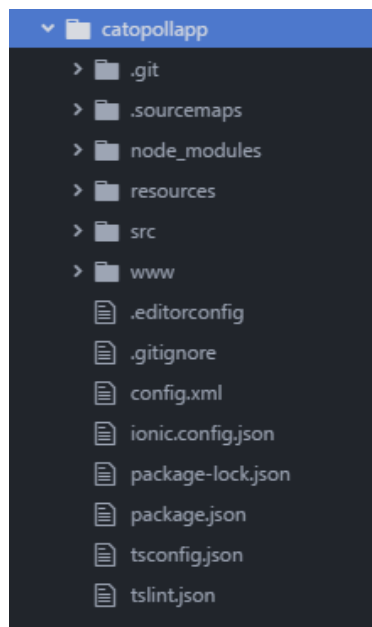
El aplicativo movil esta hecho en Ionic Framework el mismo que usa como base a angular, por lo que sigue la misma metodologia de desarrollo y se maneja con el mismo lenguaje. A su vez Ionic Framework a través de

Frontend	
Archivo	Descripción
proyecto.component.html	Contiene la estructura html de la vista
proyecto .component.css	Contiene los estilos css propios del componente
proyecto .component.ts	Modelo y controlador de la vista
proyecto .service.ts	Service que será utilizado desde el controlador para la llamada a nuestra ruta API REST
Backend	
Archivo	Descripción
api/rutas/api.js	Donde se aloja la ruta llamada desde el proyecto.service.ts
api/modelos/ proyecto.js	Modelo de acceso a datos, archivo que contiene las rutinas que son ejecutadas para el mantenimiento y consulta de proyectos

cordova hace la respectiva generación del apk para su instalación en Android.

Dicho lo anterior se deja claro ciertos conceptos que son propios de Ionic Framework.

1. Angular es un framework web basado en componentes, en la página web se definía componentes para ser usados como páginas, en Ionic se generan **páginas** (page) que practicamente son componentes de angular.
2. En Ionic Framework los providers son los services, practicamente solo cambian de nombre en este framework.



Src

Corresponde a archivos exclusivos de Angular; en estos directorios y subdirectorios se encuentran las vistas (archivos html y css), los controladores, servicios y modules (todos archivos TypeScript y componentes funcionales del framework).

Src/pages

En esta carpeta están todas las páginas generadas para la navegación en el aplicativo.

Src/providers

En esta carpeta se alojan todos los providers (services) manejados por la aplicación.

Src/providers/global-config.ts

Este es un archivo esencial en la app aquí vamos a configurar la url base de la api.

```
1 import { Injectable } from '@angular/core';
2
3 @Injectable()
4 export class GlobalConfigProvider {
5
6     getUrlBase(){
7         // return 'http://localhost:5000';
8         return 'https://catopoll.herokuapp.com';
9     }
10
11 }
```



DECLARACIÓN Y AUTORIZACIÓN

Yo, **López Alcívar, Diego Camilo** con C.C: # **0924163827** autor del trabajo de titulación: **Implementación de un sistema para la gestión de encuestas en proyectos de vinculación de la Facultad de Ingeniería de UCSG** previo a la obtención del título de **Ingeniero en Sistemas Computacionales** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, **09 de marzo de 2018**

f. _____

Nombre: **López Alcívar, Diego Camilo**

C.C: 0924163827



REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TÍTULO Y SUBTÍTULO:	Implementación de un sistema para la gestión de encuestas en proyectos de vinculación de la Facultad de Ingeniería de UCSG		
AUTOR	Diego Camilo, López Alcívar		
TUTOR	Ing. César Adriano, Salazar Tovar, Mgs		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Facultad de Ingeniería		
CARRERA:	Carrera de Ingeniería en Sistemas Computacionales		
TITULO OBTENIDO:	Ingeniero en Sistemas Computacionales		
FECHA DE PUBLICACIÓN:	09 de marzo de 2018	No. DE PÁGINAS:	126
ÁREAS TEMÁTICAS:	Hardware, Software, Redes y Comunicaciones		
PALABRAS CLAVES/ KEYWORDS:	automatización, encuesta, georreferenciación, sistema web, aplicativo móvil, Angular, Bootstrap, Node.js, Ionic		
RESUMEN/ABSTRACT (150-250 palabras):			
<p>Para la realización de los proyectos de vinculación de la UCSG, la Facultad de Ingeniería necesita de una herramienta tecnológica que permita levantar información de los grupos objetivo. Por tal motivo, se planteó la implementación de un sistema para la gestión de encuestas, al que pueda acceder tanto en plataforma web como en dispositivo inteligente. La investigación preliminar utilizó la metodología cualitativa, el paradigma de programación formulativo, Programación Extrema como metodología de desarrollo ágil, la investigación no experimental con diseño transversal y la entrevista como instrumento de recolección de datos, la misma que se la aplicó a tres docentes de la Facultad y Universidad, de donde se obtuvieron los requerimientos funcionales y se trató algunos temas relacionados con la correcta elaboración de una encuesta. Con la información obtenida, se caracterizaron las necesidades y requerimientos que debía tener el sistema, determinándose las herramientas óptimas de desarrollo, con el fin de que la experiencia de usuario final sea agradable y que las funcionalidades del aplicativo web y móvil sean aprovechadas en su totalidad. Se diseñó e implementó el sistema gestor de encuestas para plataforma web y móvil para los usuarios finales, y finalmente se propusieron algunas recomendaciones a tomarse en consideración para el mejoramiento del mismo.</p>			
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
CONTACTO CON AUTOR:	Teléfono: +593-4-0996788888	E-mail: diego-lo-93@hotmail.com	
CONTACTO CON LA INSTITUCIÓN (COORDINADOR DEL PROCESO UTE):	Nombre: Ing. Yanza Montalván, Ángela Olivia, Mgs.		
	Teléfono: +593-4-2206950 ext 1020		
	E-mail: angela.yanza@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
Nº. DE REGISTRO (en base a datos):			
Nº. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			