



Facultad de Ciencias y Tecnologías Computacionales

Componente de mapas de calor según datos del censo de población

Trabajo de diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autor:

Alvaro Ernesto Gutiérrez Ortiz

Tutor:

MsC.Grethell Castillo Reyes

La Habana, 2022

Año 64 de la Revolución

DECLARACIÓN DE AUTORÍA

El autor del trabajo de diploma con título ***“Componente de mapa de calor según datos del censo de población”*** concede a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la investigación, con carácter exclusivo. De forma similar se declara como únicos autores de su contenido.

Para que así conste firman la presente a los ____ días del mes de _____ del año _____.

Alvaro Ernesto Gutiérrez Ortiz

MSc.Grethell Castillo Reyes

Firma del Autor

Firma del Tutor

DATOS DE CONTACTO

Tutora:

MsC.Grethell Castillo Reyes: Graduado de Ingeniero en Ciencias Informáticas en La Universidad de las Ciencias Informáticas en el año 2012, MsC En Informática Aplicada en el año 2015. Trabajador del centro GEYSED en el departamento Integración de soluciones, donde desempeñó el rol de Jefe de Proyecto de la Línea de Productos de Software Aplicativos SIG, Jefe de Departamento de Desarrollo de Componentes y Directora del Centro GEYSED. Actualmente se desempeña como Especialista A en Ciencias Informáticas en el Centro para la Representación y Análisis de Datos. Correo Electrónico: gcreyes@uci.cu

Autor:

Nombre y apellido: Alvaro Ernesto Gutiérrez Ortiz.

E-mail: alvaroego@estudiantes.uci.cu

AGRADECIMIENTOS

A mi esposa por todos sus consejos y su apoyo incondicional.

Quiero agradecer a mi familia, en especial a mi mamá por toda la confianza depositada en mí y por todo el apoyo que me han dado siempre en los momentos más difíciles de mi carrera.

A mi tutora Grethell por guiarme y preocuparse por mí y a todos los profesores que han contribuido a mi mejor formación como profesional.

A todos de corazón muchas gracias por acompañarme todos estos años.

DEDICATORIA

A mi esposa Daymara M. Garcia González por tenerme muchísima paciencia sobre todo en esta etapa, por siempre estar a mi lado luchando por un futuro mejor y una familia juntos, por lograr muchas metas y las que nos faltan por cumplir, por ser la mejor novia y esposa del mundo.

A mi mamá Iraida Ortiz Aguiar por haberme guiado, darme buenos consejos y orientarme de forma correcta, por la confianza que siempre depositó en mí, por darme tanto amor y ser la mejor mamá del mundo.

A mi hermana, mis sobrinos, mis abuelos, mis amigos y a todas las personas que me apoyaron y de una forma u otra forman parte de mi vida.

RESUMEN

Un Sistema de Información Geográfica (SIG) es un marco de trabajo para reunir, gestionar y analizar datos. El SIG muestra el conocimiento más profundo escondido en los datos, como patrones, relaciones y situaciones, ayudando al usuario a tomar decisiones más inteligentes. La presente investigación tiene como objetivo crear un componente para representar la concentración geográfica según los datos recopilados del último año del censo de población y vivienda realizada en Cuba, mediante mapas de calor, estos datos están actualmente guardados en una base de datos. La investigación está regida por la metodología AUP-UCI en el escenario cuatro: historias de usuarios. Para el desarrollo de este componente se seleccionaron varias herramientas y tecnologías como el framework Angular, la biblioteca OpenLayers para la parte la interacción con el mapa, y para el backend se utilizó el framework Express basado en NodeJS y el gestor de base de datos PostgreSQL. Las pruebas aplicadas al componente arrojaron resultados satisfactorios y se usaron como herramientas de prueba Jasmine y Karma. La aplicación permite la visualización de la concentración geográfica tanto por provincia como por municipio, de diferentes tipos de datos como la cantidad de habitantes, por grupo de edades

y por sexo.

PALABRAS CLAVE

Aplicación, componente, datos, mapa de calor, sistemas de información geográfica.

ABSTRACT

A Geographic Information System (GIS) is a framework for collecting, managing and analyzing data. GIS shows the deeper knowledge hidden in the data, such as patterns, relationships and situations, helping the user to make smarter decisions. This research aims to create a component to represent the geographic concentration according to the data collected from the last year of the population and housing census conducted in Cuba, by means of heat maps. These data are currently stored in a database. The research is guided by the AUP-UCI methodology in its fourth scenario, user stories. For the development of this component several tools and technologies were selected such as the Angular framework, the Open Layers library to interact with the maps, and for the backend the Express framework based on NodeJS and the PostgreSQL database manager were used. The tests applied to the component carry out successful results, Jasmine and Karma were used as testing tools. The application allows the visualization of the geographic concentration both by province and by municipality, of different types of data such as the number of the persons, by age group and by sex.

KEY WORDS

Application, component, data, heat map, geographic information systems.

TABLA DE CONTENIDO

INTRODUCCIÓN.....	1
CAPÍTULO I: FUNDAMENTOS Y REFERENTES TEÓRICO-METODOLÓGICOS SOBRE EL OBJETO DE ESTUDIO.....	6
I.1 Conceptos asociados al dominio de la investigación.....	6
I.2 Caracterización del objeto de estudio.....	9
I.2.1 Sistemas de Información Geográfica en el análisis estadístico de datos poblacionales.....	10
I.3 Metodología de desarrollo de software.....	12
I.4 Herramientas y tecnologías.....	15
I.4.1 JavaScript.....	15
I.4.2 OpenLayers v7.0.1.....	15
I.4.3 Visual Paradigm.....	15
I.4.4 Angular.....	16
I.4.4.1 Angular Cli.....	16
I.4.5 Nodejs y express.....	16
I.4.6 PostgreSQL 14.....	16
Conclusiones del capítulo.....	17
CAPÍTULO II: DISEÑO DE LA SOLUCIÓN PROPUESTA AL PROBLEMA CIENTÍFICO.....	18
II.1 Modelo conceptual.....	18
II.1.2 Descripción de la clase del modelo conceptual.....	19
II.2 Levantamiento de requisitos.....	20
II.2.1 Requisitos funcionales.....	20
II.2.3 Requisitos No Funcionales.....	21
II.3 Historia de Usuario.....	22
II.4. Estilo arquitectónico.....	26
II.5. Patrones de diseño.....	27
II.5.1. Patrones GRAPS.....	28
II.5.2. Patrones GOF.....	28
II.6. Diagrama de clase del diseño.....	29
II.7. Modelo de datos.....	29
II.8 Modelo de despliegue.....	30
Conclusiones del capítulo.....	31
CAPÍTULO III: VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.....	33
III.1 Modelo Implementación.....	33
III.2 Diagrama de componente.....	33
III.3 Estándares de codificación.....	34
III.3 Pruebas.....	35
III.3.1 Pruebas Unitarias.....	35
Conclusiones del capítulo.....	40
CONCLUSIONES FINALES.....	41
RECOMENDACIONES.....	42

REFERENCIAS BIBLIOGRÁFICAS.....	43
ANEXOS.....	48

ÍNDICE DE TABLAS

TABLA 1: DESCRIPCIÓN DE LAS FASES DE AUP-UCI [TOMADO DE: (BRITO MORALES, BRAVO, JIMÉNEZ 2019)].....13

TABLA 2: DESCRIPCIÓN DE LA CLASE DEL MODELO CONCEPTUAL.....19

TABLA 3: HU VISUALIZAR MAPA.....22

TABLA 4: HU ACERCAR ÁREA DEL MAPA (ZOOM IN).....23

TABLA 5: HU ALEJAR ÁREA DEL MAPA (ZOOM OUT).....23

TABLA 6: HU VISUALIZAR MAPA DE CALOR A PARTIR DE UN CRITERIO DETERMINADO.
.....24

TABLA 7: HU EXPORTAR IMAGEN DEL MAPA DE CALOR.....25

ÍNDICE DE FIGUR

FIGURA 1: SUBSISTEMA QUE INCLUYE UN SIG.....7

FIGURA 2: PRONÓSTICO DEL TIEMPO EN CUBA USANDO RADARES [TOMADO DE: (INSMET 2022)].	9
FIGURA 3: MAPA TEMÁTICO DENSIDAD POBLACIONAL DE COSTA RICA [TOMADO DE: (INEC 2022)].	11
FIGURA 4: MAPA DE CALOR DEL COVID-19 EN SANTA FE[TOMADO DE: (GOMEZ 2021)].	12
FIGURA 5: MODELO CONCEPTUAL.	19
FIGURA 6: VISUALIZAR MAPA.	23
FIGURA 7: ACERCAR ÁREA DEL MAPA (ZOOM IN).	23
FIGURA 8: VISUALIZAR MAPA DE CALOR A PARTIR DE UN CRITERIO DETERMINADO.	25
FIGURA 9: EXPORTAR IMAGEN DEL MAPA DE CALOR.	26
FIGURA 10: ARQUITECTURA MVC.	27
FIGURA 11: DIAGRAMA DE CLASES DEL DISEÑO.	29
FIGURA 12: MODELO DE DATOS.	30
FIGURA 13: DIAGRAMA DE DESPLIEGUE.	31
FIGURA 14: MODELO DE COMPONENTE.	34
FIGURA 15: PRUEBA UNITARIA CON JASMINE Y KARMA.	37
FIGURA 16. PRUEBA UNITARIA CON JASMINE Y KARMA.	38
FIGURA 17. PRUEBA UNITARIA CON JASMINE.	39
FIGURA 18: MAPA TEMÁTICO DENSIDAD POBLACIONAL POR COMUNA CIUDAD DE BUENOS AIRE.	48

Y

INTRODUCCIÓN

El Censo de población constituye una de las investigaciones estadísticas más importante que realiza un país y una de las más trascendentes en el campo social, teniendo en cuenta la utilidad de sus resultados para procesos de dirección, toma de decisiones e investigación en las temáticas que abarca (CEPAL 2019). El censo de población se define como el conjunto de operaciones que consisten en recopilar, resumir, valorar, analizar y publicar los datos de carácter demográfico, cultural, económico y social de todos los habitantes del país y de sus divisiones político-administrativas, referidos a un momento o período dado (Perales Miranda, Párraga Guachalla, Usnayo Sirpa 2021).

Los censos son la fuente primaria más importante y amplia de información estadística. Dada su periodicidad es posible comparar los cambios en el tiempo y entre países, ya que el levantamiento censal se realiza en fechas similares en casi toda América Latina y el Caribe (CEPAL 2022).

Los censos de población se fomentan por parte de las Naciones Unidas, que emiten recomendaciones internacionales para que puedan considerarse homogéneos en todo el mundo (Jiménez 2019). Se propone que esos recuentos se hagan al menos una vez cada diez años. Un censo permite ajustar y/o definir, implementar y evaluar planes, programas, políticas públicas y estrategias de desarrollo humano sostenible, económico y social en el ámbito nacional, departamental, sectorial, municipal y comunitario. Actualmente, esta información se almacena en bases de datos para su posterior análisis estadístico (ONEI 2014).

Para lograr el manejo de la información obtenida a través del censo de la población, una de las herramientas a utilizar son los Sistemas de Información Geográfica (SIG). Estos constituyen una plataforma tecnológica de utilidad para las agencias que trabajan con estadísticas, ofreciéndole un software apropiado para la gestión y administración de una base de datos geográfica. Son comúnmente usados para el análisis y la gestión de datos, arraigado en la ciencia geográfica (Buzai et al. 2016).

Una de las representaciones gráficas utilizadas en los SIG para el análisis de la información son los mapas de calor o *heatmaps*. Un mapa de calor es una representación gráfica de un elemento que utiliza un código de colores para reflejar diferentes estados de actividad. Habitualmente, el rojo representa a los valores altos y el azul a la actividad más baja. Los valores intermedios se reflejan con los colores que separan al rojo y azul en el espectro luminoso: naranja, amarillo y verde. Al tratarse de un recurso tan visual y contar con una representación colorimétrica tan natural y comprensible para cualquiera (las zonas calientes se muestran en rojo y las frías en azul) permite la fácil lectura e interpretación de los elementos o áreas de mayor interés e interacción para el usuario (González, Casañola 2019).

En este sentido, en el Centro de Representación y Análisis de Datos (CREAD) de la Facultad de Ciencias y

Tecnología Computacionales (CITEC) de la Universidad de las Ciencias Informáticas (UCI), se cuenta con una base de datos SQL Server¹ que almacena los datos recopilados en el último censo de población y viviendas realizado en Cuba. A través de estos datos es posible efectuar análisis estadístico, por ejemplo: desde la cantidad de personas por género, edad, grado de escolaridad, hasta la clasificación de las viviendas, cantidad de personas que reciben ayuda social o que la necesitan por sus condiciones de vida, cantidad de personas con determinado padecimiento de salud, entre otras clasificaciones. Actualmente, existen limitantes a la hora de consultar y analizar la concentración geográfica (por provincias, municipios y localidades) de estos datos, lo cual dificulta el proceso de toma de decisiones de forma rápida. Además, estos datos son usados por varios proyectos, entre ellos el SIG para las Tiendas Recaudadoras de Divisas (TRD), para la representación de la densidad poblacional en cada uno de los municipios del país. Sin embargo, actualmente, cada uno de los sistemas que requiere el uso y análisis de estos datos de forma temática, deben implementar un componente para la visualización de los mismos en dependencia de los requerimientos, lo cual se traduce en pérdida de tiempo y recursos humanos.

Luego de analizar lo antes planteado surge el siguiente **problema científico de la investigación** ¿Cómo contribuir al análisis de los datos recopilados a través del censo de población? Para dar solución al problema anteriormente planteado se propone como **objetivo general** Desarrollar un componente para representar los datos recopilados en el censo de población a través de mapas de calor.

Identificándose como **objeto de estudio** Desarrollo de Sistemas de Información Geográfica, enmarcado en el **campo de acción** Representación de mapas de calor en Sistemas de Información Geográfica.

Para facilitar la solución del problema a resolver se generan las siguientes **preguntas de la investigación**:

- ¿Qué metodología, herramientas y tecnologías utilizar para llevar a cabo el desarrollo de un componente para representar los datos recopilados en el censo de población a través de mapas de calor?
- ¿Qué características debe cumplir el desarrollo de un componente para representar los datos recopilados en el censo de población a través de mapas de calor para que cumpla con

¹ Microsoft SQL Server es un sistema de gestión de base de datos relacional, desarrollado por la empresa Microsoft Petkovic, Dusan. (2012). Microsoft SQL Server 2012 A Beginners Guide.

las necesidades de los clientes?

- ¿Cómo desarrollar un componente para representar los datos recopilados en el censo de población a través de mapas de calor de manera que contribuya al proceso de toma de decisiones en el centro CREAD?
- ¿Cómo probar el desarrollo de un componente para representar los datos recopilados en el censo de población a través de mapas de calor con el fin de garantizar el correcto funcionamiento del mismo?

Para facilitar la solución del problema a resolver se generan las siguientes **tareas de la investigación**:

1. Estudio de los conceptos asociados al marco teórico de la investigación.
2. Caracterización de los sistemas que permitan la visualización de los mapas de calor.
3. Estudio, definición y análisis de la metodología y herramientas a utilizar para guiar el desarrollo del componente para representar los datos recopilados en el censo de población a través de mapas de calor en SIG.
4. Identificar las principales funcionalidades con las que contará el módulo a desarrollar.
5. Modelar la solución propuesta a partir de la metodología y herramientas definidas.
6. Implementar la solución propuesta.
7. Realizar pruebas a la solución propuesta.

Una vez cumplidas de manera satisfactoria las tareas de la investigación se esperan como posibles resultados:

- Componente de mapas de calor según datos del censo de población.
- Artefactos de ingeniería para la documentación del desarrollo.
- Documento de tesis.

Para el desarrollo de la investigación se emplearon varios **métodos científicos** que permiten tener una idea más detallada del resultado a obtener:

- **Histórico-Lógico**: Estudia la trayectoria real de los fenómenos y acontecimientos en el transcurso de su historia. Además, investiga las leyes generales del funcionamiento y desa-

rollo de los fenómenos. Este método se usa para estudiar la evolución de los conceptos asociados a los SIG, censo de la población y mapas de calor que permite la definición de términos propios.

- **Analítico-Sintético:** El análisis permite la descomposición de un todo complejo en sus partes y cualidades. La síntesis, por otro lado, establece la unión entre las partes, previamente analizadas y posibilita descubrir relaciones y características generales entre los elementos de la realidad. Este método se utiliza para la evaluación de soluciones que respondan al problema y permite realizar una valoración crítica y detallada de cada una de ellas.

Métodos empíricos:

Modelación: Representación de las características del sistema, así como las relaciones entre los objetos y funcionalidades del mismo.

Observación: Posibilita obtener conocimiento acerca del funcionamiento de los sistemas existentes en la actualidad relacionados con los SIG en representación de los mapas de calor.

Entrevista: Se emplea en encuentros con el cliente para definir las funcionalidades del componente de mapa de calor, identificando a la vez particularidades necesarias para su desarrollo.

La presente investigación se desglosa en los siguientes capítulos:

Capítulo 1: Se describen los conceptos por los cuales se rige el dominio de la investigación y se realiza un análisis de los componentes de mapa de calor más conocidos nacional e internacionalmente. Se analizan las herramientas, metodologías y técnicas utilizadas para la solución del problema planteado en la investigación. Además, se valoran las soluciones actuales que de alguna manera ofrecen respuesta al problema en cuestión.

Capítulo 2: En este capítulo se diseña la propuesta de solución según la metodología empleada. Se describen los requisitos funcionales y no funcionales con las historias de usuarios de los requerimientos que debe cumplir el sistema, se define el diseño de la arquitectura, los

patrones del diseño, el diagrama de clase del diseño, modelo de datos y el diagrama de despliegue.

Capítulo 3: En este capítulo se abordan los aspectos relacionados con la construcción de la aplicación, dentro del cual se destacan los procesos de implementación y de prueba para su validación.

CAPÍTULO I: FUNDAMENTOS Y REFERENTES TEÓRICO-METODOLÓGICOS SOBRE EL OBJETO DE ESTUDIO

Para lograr una mayor comprensión del alcance de la investigación, en el presente capítulo se exponen los fundamentos teóricos asociados al dominio del problema planteado, se realiza un análisis del estado de la temática en la comunidad científica. Se selecciona la metodología, herramientas y tecnologías que serán utilizadas para dar cumplimiento a la propuesta de solución.

I.1 Conceptos asociados al dominio de la investigación

Censo de población:

El Censo de población y viviendas se define como el conjunto de operaciones consistentes en reunir, elaborar, analizar y publicar datos demográficos, económicos y sociales correspondientes a todos los habitantes de un país, así como a las viviendas en que residen, referidos a un momento determinado o a ciertos períodos dados. Esta operación va dirigida a todas las personas que residen en viviendas, ya sean viviendas familiares o alojamientos (Jiménez 2019).

Constituye la investigación estadística más importante y costosa que realizan la mayoría de los países, usualmente con periodicidad decenal, requiriéndose, por tanto, una cuidadosa preparación de algunos años para ejecutarlo con eficiencia máxima y obtener el mejor aprovechamiento posible de los recursos y los resultados obtenidos. La información de un censo es obtenida por declaración de las personas que constituyen el objeto del mismo (Campos, Zambrano 2020).

Sistema de Información geográfica (SIG):

Una de las herramientas informáticas que se utilizan en el análisis de la información recopilada en un censo de población y viviendas, son los SIG. Es una integración organizada de hardware, software y datos geográficos diseñada para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente referenciada con el fin de resolver problemas complejos de planificación y de gestión (Vanegas 2014).

Según la (de empresarios de Andalucía 2014) los SIG *“son softwares específicos que permiten a los usuarios crear consultas interactivas, integrar, analizar y representar de una forma eficiente cualquier tipo de información geográfica referenciada asociada a un territorio, conectando mapas con bases de datos”*.



Figura 1: Subsistema que incluye un SIG

Mapa de calor:

El mapa de calor es un método de visualización popular en matrices bidimensionales (2D) donde los colores son los principales elementos estéticos que se asignan a los datos (Wilkinson, Friendly 2009).

Las visualizaciones de mapas de calor normalmente van acompañadas de un reordenamiento de filas y columnas, por agrupación jerárquica, de modo que las características con patrones similares se agrupen estrechamente y se puedan identificar fácilmente a partir de los colores en los mapas de calor (Gu, Eils, Schlesner 2016).

A través de una representación visual basada en un código de colores de fácil lectura e interpretación, el mapa de calor web de una página muestra qué elementos o áreas geográficas presentan más interés e interacción para el usuario. Los mapas de calor basan su forma de representación en la termografía (técnica que permite calcular temperaturas a distancia, con exactitud y sin necesidad de contacto físico con el objeto a estudiar) y utiliza una jerarquía de colores en la que los puntos de más interés, se identifican con las gamas de colores

cálidos (rojo, naranja, amarillo), mientras que las zonas de menos interés están representadas por las gamas de colores fríos (verde, azul, turquesa y otros) (Schmittgen, Livak 2008).

Es una técnica de representación geográfica usada en la exploración de conjunto de datos masivos para visualizar la intensidad de un fenómeno, además de brindar una representación visual interpretable de patrones de distribución espacial complejos (Rodríguez Cárdenas, Duarte Jiménez 2019).

A continuación, la Figura 2 muestra el uso de un mapa de calor generador por INSMET (Instituto de Meteorología de la República de Cuba), centro de pronóstico del tiempo mediante los radares:

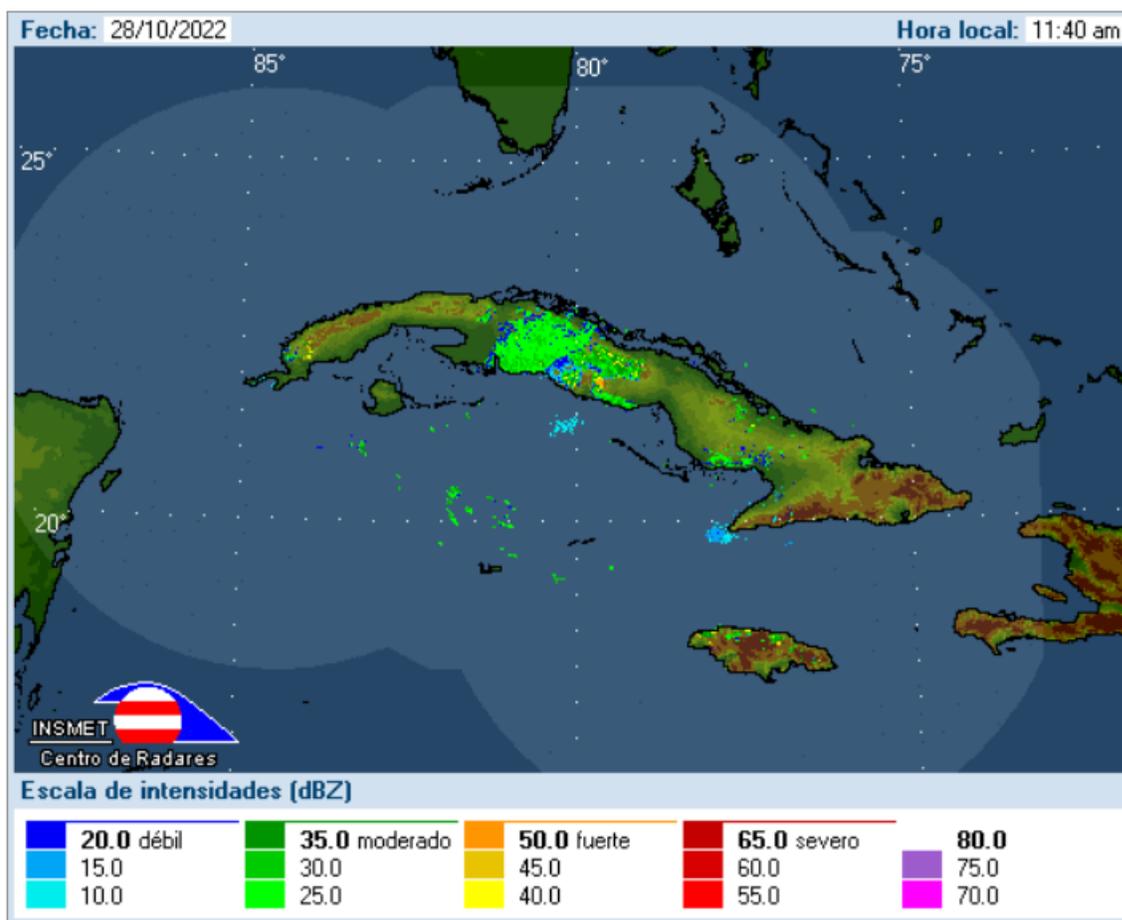


Figura 2: Pronóstico del tiempo en Cuba usando radares [Tomado de:(INSMET 2022)]

I.2 Caracterización del objeto de estudio

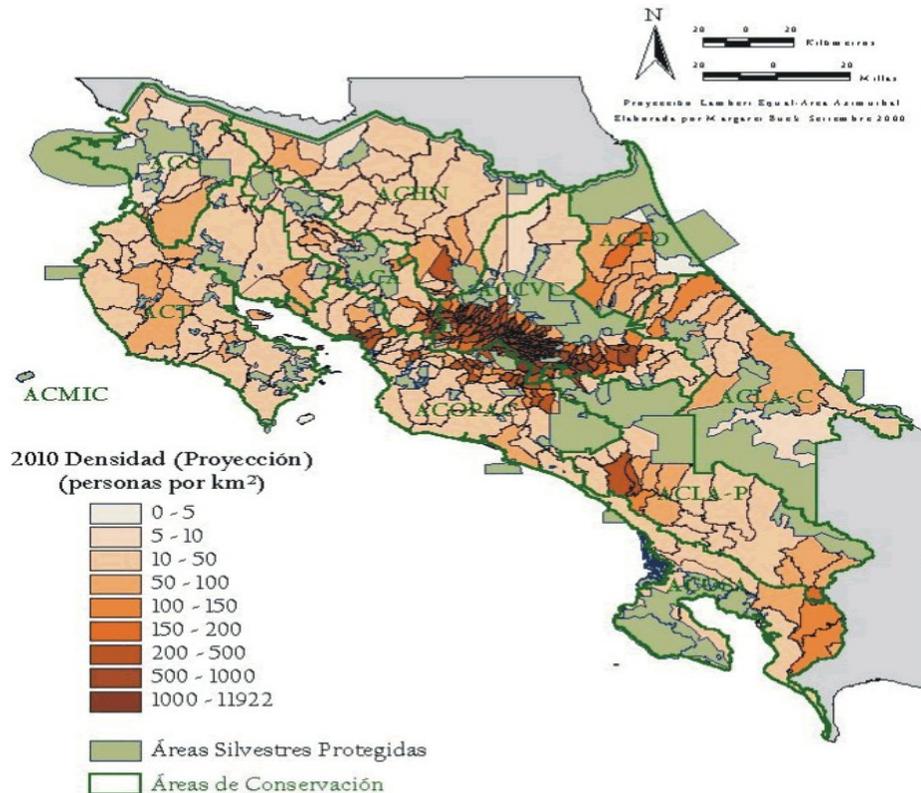
Los SIG tienen un amplio campo de aplicación en diferentes aspectos como sociales-culturales, económicos y

ambientales que conducen a la toma de decisiones de una manera más eficaz. El término SIG, es de compleja definición, dadas sus capacidades técnicas, analíticas y su carácter multipropósito. En la actualidad está ampliamente difundido tanto en la geografía como en otras ciencias, en especial, en aquellas vinculadas con la planificación territorial y la resolución de problemas sociales, económicos, productivos y ambientales (Delgado, García 2017).

I.2.1 Sistemas de Información Geográfica en el análisis estadístico de datos poblacionales.

1. Sistema de Información Geográfica aplicado a la Dirección de Estadística y Censo de la Ciudad autónoma de Buenos Aires (Andrade, de Sabalain 2001). El SIG aplicado a la Dirección de Estadística y Censo de la Ciudad autónoma de Buenos Aires surge en el marco de un proyecto financiado por el Banco Mundial. Su principal objetivo consistió en disponer de una base cartográfica digital a nivel de manzana, para posibilitar la optimización de las tareas de gestión de información estadística que competen a la Dirección de Estadística de la ciudad. Ver figura en el Anexo 1.
2. Mapas temáticos del Instituto Nacional de Estadísticas y Censos de Costa Rica. En el sitio web del Instituto Nacional de Estadísticas y Censos de Costa Rica (INEC 2022) se encuentran publicados gran variedad de mapas temáticos en formato PDF² relacionados con la distribución de la población en el país, teniendo en cuenta determinado parámetro, por ejemplo la tasa de migración por localidad, porcentaje de viviendas individuales, densidad de población por localidad, entre otros. Aunque en estos sistemas el análisis se realiza a través de mapas temáticos por colores y no de mapas de calor, como es el interés de esta investigación, resulta importante resaltar que se toman como base para definir las principales variables del censo de población a visualizar.

² PDF (Portable Document Format) es un formato de almacenamiento para documentos digitales independientes de la plataforma de software o hardware.



*Fuente: Censo de Población 1984 (Proyecciones), INEC, Áreas de Conservación y ASP- SINAC.

Figura 3: Mapa temático densidad Poblacional de Costa Rica [Tomado de:(INEC 2022)]

3. Sistema de información geográfica para la gestión estadística de la salud de Cuba. Con el objetivo de facilitar la gestión estadística de salud se confecciona una aplicación de un SIG, el cual permite cartografiar y hacer diferentes tipos de análisis de importantes indicadores de salud: morbilidad, mortalidad, demográficos, recursos y servicios (Fernández Núñez 2006).
4. Análisis de las características espacio-temporales de la pandemia del Covid-19 en la provincia de Santa Fe y particularmente en sus dos principales áreas metropolitanas: Rosario y Santa Fe, a partir de un análisis de mapas de calor utilizando SIG (Gomez 2021).

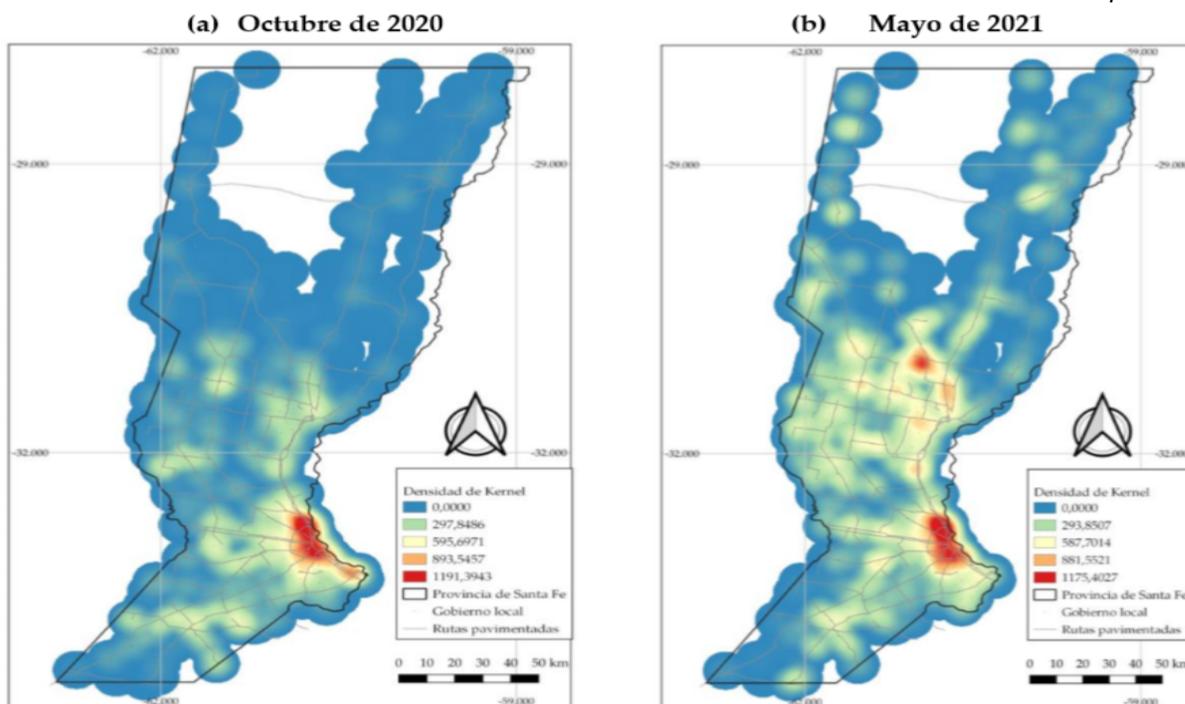


Figura 4: Mapa de calor del COVID-19 en Santa Fe [Tomado de: (Gomez 2021)]

El estudio de los diferentes sistemas homólogos, permitió adquirir conocimiento de cómo estos analizan y visualizan los diferentes datos del censo de población. Se llega a la conclusión de implementar un componente que brinde un servicio donde cualquier sistema que desee tener una toma de decisión de forma rápida sobre la concentración geográfica de estos datos pueda consumir este servicio.

I.3 Metodología de desarrollo de software

Durante la creación de un software es necesario utilizar una metodología que se encargue de guiar el proceso de desarrollo; como todos los procesos no son iguales, no existe una única metodología, sino varias que se ajustan a las características de cada producto que se realice.

Existen metodologías ágiles y tradicionales. Las metodologías tradicionales pueden ser empleadas para guiar el proceso de desarrollo de proyectos grandes o pequeños, aunque son más apropiadas para proyectos grandes que por su importancia requieren una fuerte planificación. Las metodologías ágiles son apropiadas para guiar proyectos de poco volumen que requieran una rápida implementación (Ramírez Martín, Rodríguez Donatien 2009).

La UCI desarrolló una versión de la metodología de desarrollo de software AUP (Proceso Ágil Unificado), con el fin de crear una metodología que se adapte al ciclo de vida definido por la actividad productiva de la

universidad. Esta versión decide mantener para el ciclo de vida de los proyectos la fase de Inicio, pero modificando el objetivo de la misma y se unifican las restantes fases de la metodología de desarrollo de software AUP en una sola, nombrada Ejecución y agregándose también una nueva fase llamada Cierre (Sánchez 2015). A continuación, se muestra una tabla con las fases de la metodología AUP-UCI:

Tabla 1: Descripción de las fases de AUP-UCI [Tomado de: (Brito Morales, Bravo, Jiménez 2019)]

Fases AUP	Fases Variación AUP-UCI	Objetivos de las fases (Variación AUP-UCI)
1-Inicio	1-Inicio	Durante el inicio del proyecto se llevan a cabo las actividades relacionadas con la planeación del proyecto. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no el proyecto.
2-Elaboración 3-Construcción 4-Transición	2- Ejecución	En esta fase se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtienen los requisitos, se elaboran la arquitectura y el diseño, se implementa y se libera el producto.
	3-Cierre	En esta fase se analizan tanto los resultados del proyecto como la ejecución y se realizan las actividades formales de cierre del proyecto.

La metodología de software AUP-UCI a partir de que el modelado de negocio propone tres variantes a utilizar en los proyectos, como son: CUN (Casos de uso del negocio), DPN (Descripción de proceso de negocio) o MC (Modelo conceptual) y existen tres formas de encapsular los requisitos, los cuales son: CUS (Casos de uso del sistema), HU (Historias de usuario), DRP (Descripción de requisitos por proceso), surgen cuatro escenarios para modelar el sistema en los proyectos, los cuales son:

- Escenario No 1: Proyectos que modelen el negocio con CUN solo pueden modelar el sistema con CUS.
- Escenario No 2: Proyectos que modelen el negocio con MC solo pueden modelar el sistema con CUS.
- Escenario No 3: Proyectos que modelen el negocio con DPN solo pueden modelar el sistema con DRP.
- Escenario No 4: Proyectos que no modelen negocio solo pueden modelar el sistema con HU.

A partir del análisis e investigación efectuada, la metodología AUP-UCI es apropiada para proyectos pequeños, permitiendo disminuir las probabilidades de fracaso, por ser un producto de fácil uso utilizando cualquier herramienta. Con la adaptación de AUP que se propone para la actividad productiva de la UCI se logra estandarizar el proceso de desarrollo de software. Se logra hablar un lenguaje común en cuanto a fases,

disciplinas, roles y productos de trabajos. Se redujo a 1 la cantidad de metodologías que se usaban y de más de 20 roles en total que se definían se redujeron a 11. De igual manera, también es la metodología que más utilidad tiene en la UCI por ser una versión desarrollada en dicha institución (Sánchez 2015).

Se seleccionó la metodología AUP-UCI en su escenario 4 porque emplea un equipo de desarrollo pequeño, en este caso un desarrollador, los requisitos del sistema están bien definidos y el cliente va a estar en todo el proceso de desarrollo de la aplicación.

I.4 Herramientas y tecnologías

I.4.1 JavaScript

JavaScript es un lenguaje de programación o de secuencias de comandos que permite crear contenido de actualización dinámica, controlar multimedia, animar imágenes e implementar funciones complejas en páginas web. Es la tercera capa del pastel de las tecnologías web estándar (Hashemi, Tahir, Rasheed 2022).

I.4.2 OpenLayers v7.0.1

OpenLayers es una biblioteca de mapas web JavaScript de código abierto para crear aplicaciones de mapas web, es una biblioteca modular de alto rendimiento y con una amplia variedad de funciones para mostrar e interactuar con mapas y datos geoespaciales (OpenLayers 2022).

Permite la creación de mapas dinámicos en cualquier página web. Puede mostrar mosaicos de mapas, datos vectoriales y marcadores cargados desde cualquier fuente. OpenLayers ha sido desarrollado para promover el uso de información geográfica de todo tipo. Es completamente gratuito, de código abierto, publicado bajo la licencia Berkeley Software Distribution (por sus siglas en inglés BSD) de dos cláusulas, también conocida como FreeBSD.

Este se ejecuta en todos los navegadores modernos que admiten HTML5 y ECMAScript 5; la biblioteca está pensada para usarse tanto en computadoras de escritorio/portátiles como en dispositivos móviles, y admite interacciones táctiles y de puntero (He et al. 2019).

I.4.3 Visual Paradigm

Para el modelado se utiliza la herramienta Visual Paradigm en su versión 8.0. Es una herramienta CASE multiplataforma, que soporta el ciclo completo de desarrollo de software: análisis, diseño, implementación y pruebas. Ideal para ingenieros de software, analistas y arquitectos de sistemas, que están interesados en la construcción de sistemas a gran escala y necesitan confiabilidad y estabilidad en el desarrollo orientado a objetos. Permite la generación de bases de datos, conversión de diagramas entidad-relación a tablas de base de datos, mapeos de objetos y relaciones, ingeniería directa e inversa, la gestión de requisitos de software y la modelación de procesos del negocio (Peña et al. 2016).

I.4.4 Angular

Angular es un framework para aplicaciones web. Desarrollado en TypeScript, de código abierto, mantenido por Google. Angular es un marco de diseño de aplicaciones y una plataforma de desarrollo para crear aplicaciones de una sola página eficientes y sofisticadas. Es una plataforma que puede escalar desde proyectos de un solo desarrollador hasta aplicaciones de nivel empresarial, está diseñada para que la actualización sea lo más sencilla posible, sirve tanto en versiones móviles como escritorio (Farinango Caiza 2020) (Novac et al. 2021).

I.4.4.1 Angular Cli

Herramienta para inicializar, es la forma más rápida y sencilla de desarrollar e incluso para mantener aplicaciones de Angular. También podemos ejecutar tareas de pruebas y realizar despliegue de una aplicación(Farinango Caiza 2020).

I.4.5 Nodejs y express

Nodejs es un programa que provee de información de servicios a otros programas sin ser parte del sistema operativo, es de código abierto para la capa del servidor y es multiplataforma (Mardan 2018). Express es una infraestructura de aplicaciones web Nodejs mínima y flexible que proporciona un conjunto sólido de características para las aplicaciones web y móviles, con una variedad de métodos de programas de utilidad HTTP y middleware a su disposición, la creación de una API sólida es rápida y sencilla (Mardan 2014).

I.4.6 PostgreSQL 14

PostgreSQL es un sistema de GBD (Gestor de Base de Datos) objeto-relacional, de propósito general, multiusuario y de código abierto, que soporta gran parte del estándar SQL y ofrece modernas características como consultas complejas, disparadores, vistas, integridad transaccional y control de concurrencia multiversión. Puede ser extendido por el usuario añadiendo tipos de datos, operadores, funciones agregadas, funciones ventanas y funciones recursivas, métodos de indexado y lenguajes procedurales(Siahaan, Sianipar 2019) (Juba, Volkov 2019).

Conclusiones del capítulo

Con el desarrollo de este capítulo se ha logrado definir de forma clara todos los aspectos teóricos y conceptuales necesarios para la realización de la investigación propuesta, se realizó el análisis y profundización de los conceptos relacionados con el objeto de estudio, lo que posibilitó conocer el entorno del sistema sobre el cual se trabajará. Además, el análisis y estudio de las soluciones existentes, la metodología de desarrollo, herramientas, tecnologías y lenguajes de programación permite especificar el ambiente de desarrollo para la propuesta de solución.

CAPÍTULO II: DISEÑO DE LA SOLUCIÓN PROPUESTA AL PROBLEMA CIENTÍFICO

Para lograr una mayor comprensión de la propuesta de solución, el presente capítulo tiene como objetivo mostrar los resultados que se obtienen una vez concluida la etapa diseño que propone la metodología AUP-UCI en el escenario 4. Entre los elementos a destacar se encuentran el diagrama del modelo de dominio, los requisitos funcionales y no funcionales y las historias de usuario. Como parte del diseño de la aplicación se definió el estilo y los patrones de arquitectura y diseño que se emplearán en el desarrollo de la propuesta de solución.

II.1 Modelo conceptual.

Un modelo conceptual de una organización ofrece la posibilidad de visualizar los diferentes elementos que la conforman, la manera como se pueden organizar y la forma como se integra y articulan para atender las necesidades de los usuarios y de los servicios que ofrecen. El modelo conceptual es una herramienta que facilita la transferencia de la organización representada a otra que desee implementarla (Pressman 2005).

Su importancia radica en que permiten identificar, organizar y realizar razonamientos sobre los componentes y comportamiento del sistema, son la guía para el proceso de diseño del software y puede usarse posteriormente como una referencia para evaluar un diseño particular, razonar sobre la solución realizada y sobre el posible espacio de soluciones. Por su naturaleza deben ser expresivos, fáciles de usar y completos (Mariño, David 2018).

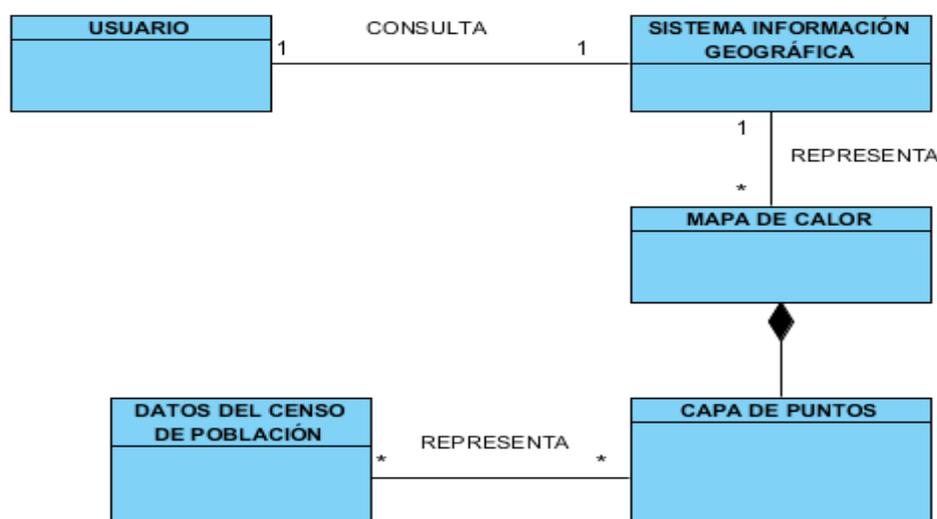


Figura 5: Modelo conceptual

II.1.2 Descripción de la clase del modelo conceptual.

Tabla 2: Descripción de la clase del modelo conceptual.

Concepto	Descripción
Usuario	Persona que interactúa con el SIG para obtener y analizar información mediante el mapa de calor sobre el censo de población.
Sistema de Información Geográfica	Es el encargado de publicar el mapa con su capa de puntos para que el usuario consuma.
Capa de puntos	Es un sistema de coordenadas que son usados para representar la localización de los datos recopilados en el censo de población.
Datos del censo de población	Es donde se obtienen todos los datos que caracterizan a la población.
Mapa de calor	Es donde se visualiza la capa de puntos mediante el mapa de calor.

II.2 Levantamiento de requisitos

La ingeniería de requisitos se define como el uso sistemático de procedimientos, técnicas, lenguajes y herramientas para obtener con un coste reducido el análisis, la documentación, la evolución continua de las necesidades del usuario y la especificación del comportamiento externo de un sistema que satisfaga las necesidades del usuario. La ingeniería de software no se guía por conductas esporádicas, aleatorias o por modas pasajeras, sino que se debe basar en el uso sistemático de aproximaciones contratadas (García-Holgado, Vázquez-Ingelmo, García-Peñalvo 2022).

La ingeniería de requisitos es la primera fase del ciclo de vida del software en la que se produce una especificación a partir de ideas informales. En esta etapa deben obtenerse y fundamentarse los requisitos de información, los requisitos funcionales, los requisitos no funcionales y los criterios para medir el grado de su consecución (Pressman 2005).

II.2.1 Requisitos funcionales

Los requerimientos funcionales de un sistema, son aquellos que describen cualquier actividad que este deba realizar, en otras palabras, el comportamiento o función particular de un sistema o software cuando se cumplen ciertas condiciones.

Requerimiento Funcional (RF) es una descripción del servicio que debe ofrecer el software. Describe un sistema de software o su componente. Solo se implementan entradas en el sistema de software, su transporte y salidas

(Sommerwille 2011).

A continuación, se muestran los RF identificados:

RF 1. Visualizar mapa.

RF 2. Acercar área del mapa (zoom in).

RF 3. Alejar área del mapa (zoom out).

RF 4. Visualizar mapa de calor a partir de un criterio determinado. Los criterios pueden ser, densidad poblacional, sexo, edad.

RF 5. Exportar imagen del mapa de calor.

II.2.3 Requisitos No Funcionales

Los requisitos no funcionales (RNF) son propiedades o cualidades que el sistema debe tener. Se refieren a las cualidades, restricciones y características del software. A diferencia de los funcionales, no determinan una funcionalidad del sistema a desarrollar. Los RNF se caracterizan por ser específicos, cuantificables y verificables (SOMMERVILLE 2011).

A continuación, se muestran lo RNF identificados:

Eficiencia

RNF 1. El tiempo de respuesta dependerá de la cantidad de información a procesar.

Interfaz y diseño

RNF 2. La solución que se propone constituye una aplicación Web que permitirá el análisis y representación sobre un mapa de los datos del censo de población.

RNF 3. Los dispositivos de los clientes deberán cumplir con los siguientes requisitos:

Hardware

- Procesador 512 MHz como mínimo.
- Memoria RAM: 512 Mb como mínimo.
- Disco Duro: 40 Gb como mínimo.
- Tarjeta de red.

Software

- Un navegador web que cumpla con los estándares W3C y cuente con soporte para CCS3, HTML5 y JS.

RNF 4. El lenguaje de programación usado para la implementación es JavaScript con la librería Open Layers.

RNF 5. Para la modelación se utilizará como lenguaje de modelado UML 2.1 y como herramienta el Visual Paradigm for UML 8.0.

Usabilidad

RNF 6. La aplicación estará diseñada para su correcta visualización en distintos tipos de dispositivos (Móvil, Tablet, PC).

Interfaz de usuario y diseño

RNF 7. La interfaz de usuario deberá tener apariencia profesional y diseño gráfico sencillo.

RNF 8. Las funcionalidades a realizar deben representarse con íconos intuitivos.

II.3 Historia de Usuario

La historia de usuario es una explicación general e informal de una función de software escrita desde la perspectiva del usuario final. Su propósito es articular como proporcionará una función de software valor al cliente (Sánchez 2015).

Para especificar los requisitos del sistema identificado se emplean las historias de usuario con el objetivo de describir las salidas necesarias, características y funcionalidades del sistema a desarrollar. Cada una de estas historias es descrita por el cliente, colocada en una tarjeta y en cualquier momento es posible escribir nuevas historias de usuario.

A continuación, se muestran las tablas de historia de usuario:

Tabla 3: HU Visualizar mapa.

Historia de usuario	
Número: HU_1	Nombre Historia de Usuario: Visualizar mapa
Prioridad en negocio: Alta	
Descripción: Muestra el mapa de Cuba con sus provincias y municipios.	
Observaciones: Al cargarse el sistema debe cargar el mapa.	
Prototipo:	

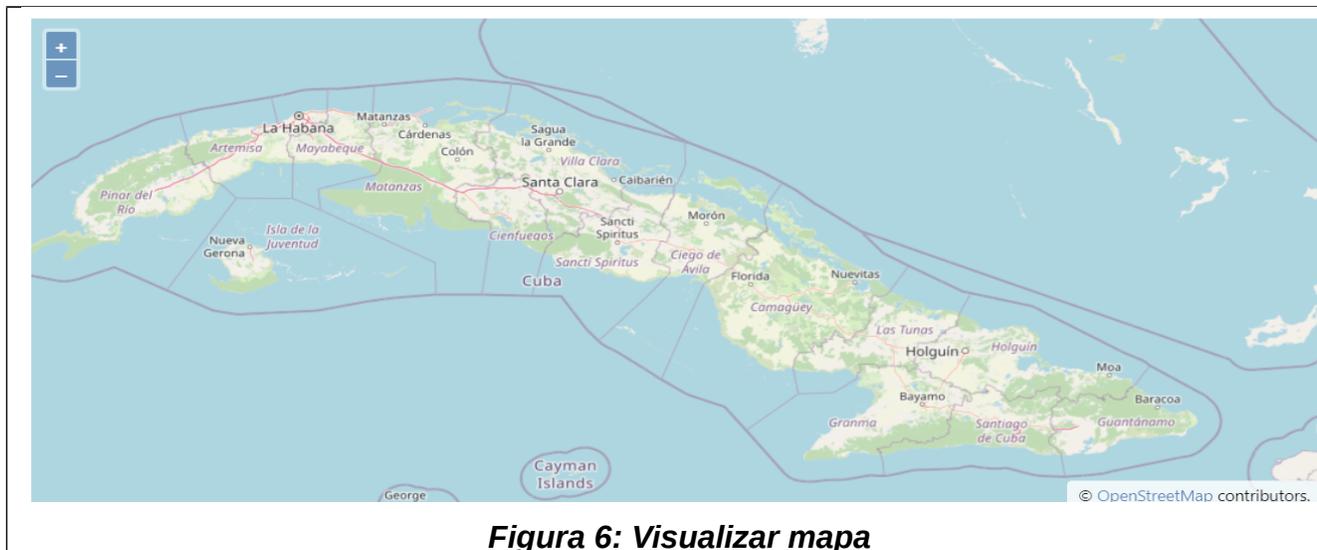


Figura 6: Visualizar mapa

Tabla 4: HU Acercar área del mapa (zoom in).

Historia de usuario	
Número: HU_2	Nombre Historia de Usuario: Acercar área del mapa (zoom in)
Prioridad en negocio: Media	
Descripción: Comienza cuando el usuario da clic sobre el botón + y el sistema debe acercar el área del mapa.	
Observaciones: el botón + tiene un límite de acercamiento.	
Prototipo:	
	

Figura 7: Acercar área del mapa (zoom in)

Tabla 5: HU Alejar área del mapa (zoom out).

Historia de usuario	
Número: HU_3	Nombre Historia de Usuario: Alejar área del mapa (zoom out)
Prioridad en negocio: Media	
Descripción: Comienza cuando el usuario da clic sobre el botón - y el sistema debe alejar el área del mapa.	
Observaciones: el botón - tiene un límite de alejamiento.	
Prototipo: Figura 7: Acercar área del mapa (zoom in)	

Tabla 6: HU Visualizar mapa de calor a partir de un criterio determinado.

instrucciones de cambiar el estado (habitualmente desde el controlador).

- La Vista es la que maneja la visualización de la información.
- El Controlador es el que interpreta las acciones del ratón y teclado, informando al modelo y/o a la vista para que se cambien según resulte apropiado.

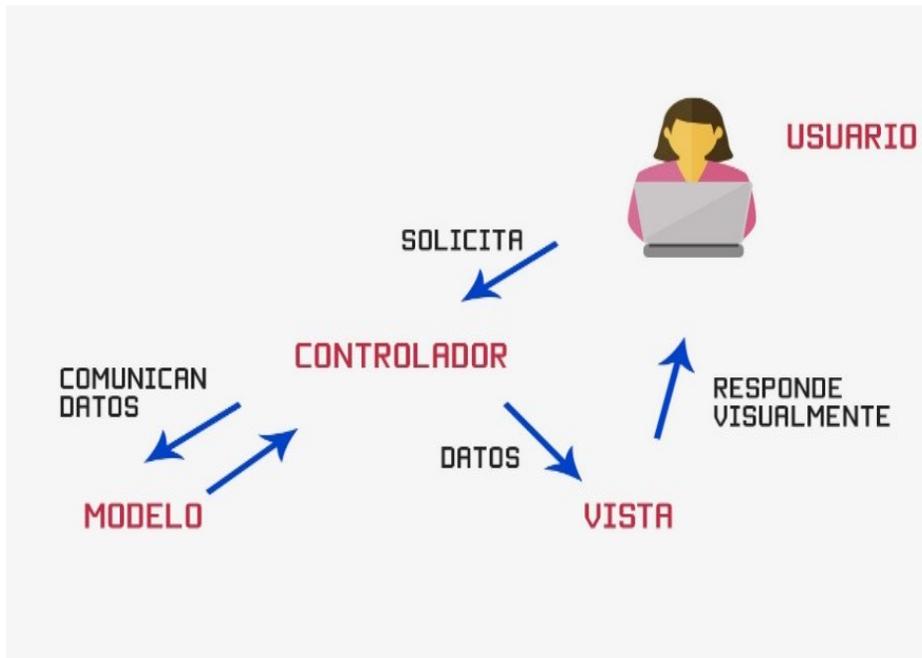


Figura 10: Arquitectura MVC.

II.5. Patrones de diseño

Los patrones arquitectónicos imponen una transformación en el diseño de una arquitectura. Un patrón impone una regla sobre la arquitectura, pues describe la manera en que el software maneja algún aspecto de su funcionalidad al nivel de la infraestructura. Los patrones arquitectónicos tienden a abarcar aspectos específicos del comportamiento dentro del contexto de la arquitectura (Sommerville 2005).

En el desarrollo de la propuesta de solución se tuvo en cuenta la utilización de los Patrones Generales de Software para Asignación de Responsabilidades (GRASP, por sus siglas en inglés) y los patrones Gang of Four (GoF, por sus siglas en inglés).

A continuación, se describen los utilizados en el diseño de la solución.

II.5.1. Patrones GRAPS

Los patrones GRAPS son patrones generales del software para asignación de responsabilidades, es el acrónimo de General Responsibility Assignment Software Patterns. Aunque se considera que más que patrones propiamente dichos, son una serie de buenas prácticas de aplicación recomendable en el diseño de software (Ortega 2021).

Experto: Este patrón plantea que se debe asignar una responsabilidad al experto en información, en otras palabras, a la clase que cuenta con los datos necesarios para cumplir la responsabilidad. Evidenciándose su uso en la clase Control Mapa.

Bajo acoplamiento: Es el principio que se debe tener en cuenta durante las decisiones de diseño. Es utilizado por el sistema en la mayoría de clases posibles, con el objetivo de eliminar la dependencia de una clase con el resto, garantizando que esta no se afecte por cambios en otros componentes.

Controlador: Permite que a través de un archivo se procesen todas las peticiones de manipulación por analizar, a través de un objeto de controlador único. En el componente mapa de calor según datos del censo de población, los eventos generados por el usuario son redirigidos a la clase controladora Control Mapa.

II.5.2. Patrones GOF

Los patrones GOF se definen como combinaciones de componentes, casi siempre clases y objetos que por experiencia se sabe que resuelven ciertos problemas de diseño comunes. En términos generales, es posible decir que un patrón de diseño es una solución a un problema recurrente en el diseño de software (Guerrero, Suárez, Gutiérrez 2013).

Observer (Observador): define una relación de uno a muchos entre objetos, de manera que cuando un objeto cambie de estado se notifique y actualicen automáticamente todos los objetos que dependen de él. Se evidencia en las actualizaciones de las variables en AngularJS.

II.6. Diagrama de clase del diseño

Un diagrama de clase de diseño muestra la especificación para las clases software de una aplicación, incluye la siguiente información: clases, asociaciones, atributos, interfaces con sus operaciones y constantes, métodos, navegación y dependencia. Es una representación más concreta que el diagrama de clases del análisis, representa la parte estática del sistema y representa las clases y sus relaciones.

A continuación, se muestra el diagrama de clases del diseño:

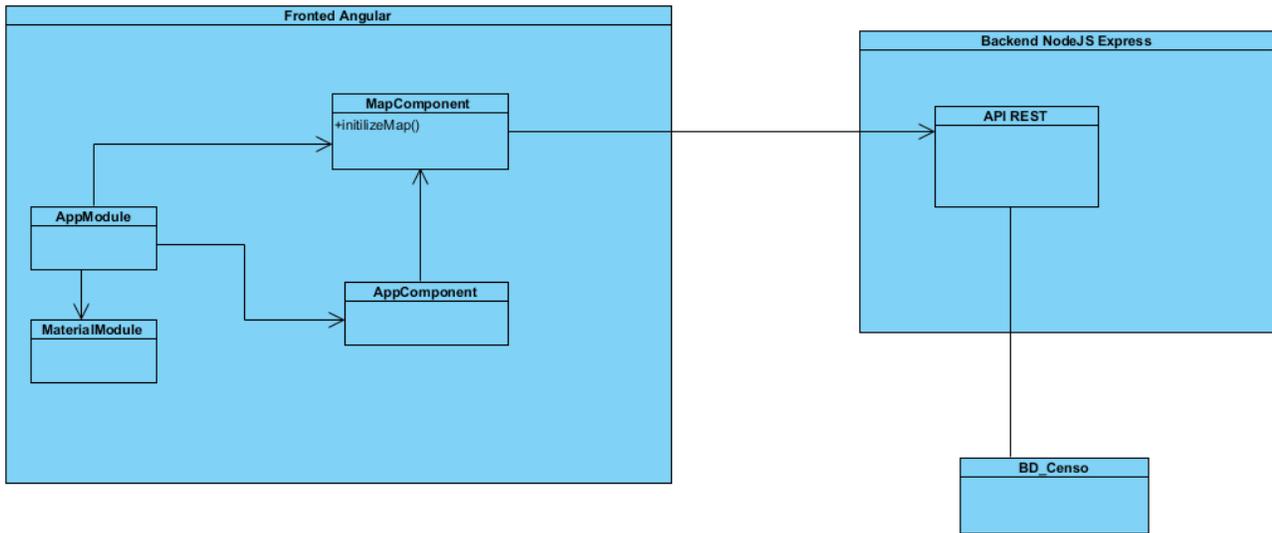


Figura 11: Diagrama de clases del diseño.

II.7. Modelo de datos

Los modelos de datos son mecanismos que permiten la abstracción y representación de un dominio, mediante un conjunto de reglas y símbolos pertenecientes a un lenguaje de modelado que es conforme al modelo, estos consideran un conjunto de elementos claves los cuales permitirán describir algo físico, abstracto o una realidad hipotética.

Para el desarrollo del componente de mapa de calor según datos del censo de población se definió el siguiente modelo de datos apoyado en el diagrama Entidad-Relación. Este representa un ejemplo, considerando que antes deba existir una base de datos del censo de población de Cuba, que brinde la información necesaria a este modelo que se muestra a continuación:

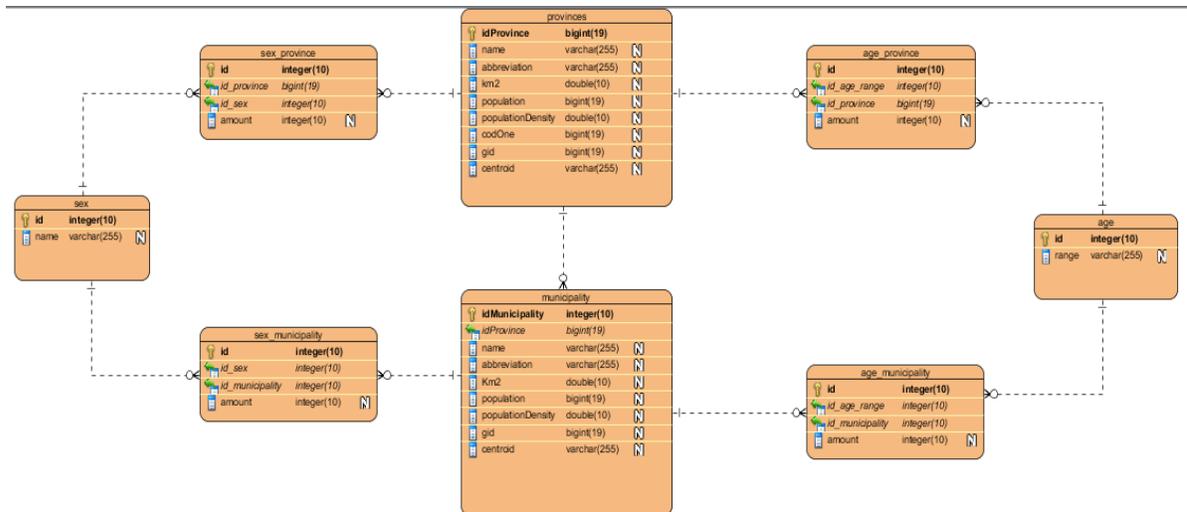


Figura 12: Modelo de datos.

II.8 Modelo de despliegue

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria (Pressman 2005).

Los diagramas de despliegue son los complementos de los diagramas de componentes que, unidos, proveen la vista de implementación del sistema. Describen la topología del sistema, la estructura de los elementos de hardware y el software que ejecuta cada uno de ellos. Los diagramas de despliegue representan a los nodos y sus relaciones (Vidal et al. 2012).

Los nodos son conectados por asociaciones de comunicación tales como enlaces de red y conexiones TCP/IP. Para cada componente de un diagrama es necesario que se deba documentar las características técnicas requeridas, el tráfico de la red y el tiempo de respuesta.

A continuación, se muestra el diagrama de despliegue:

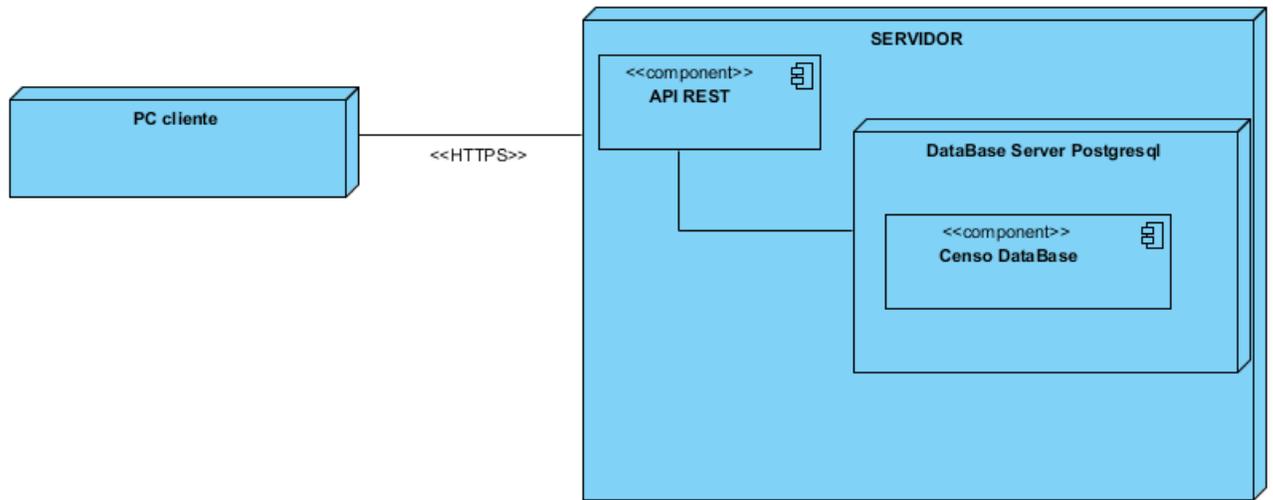


Figura 13: Diagrama de despliegue.

Conclusiones del capítulo

Con el desarrollo del presente capítulo se arribó a las siguientes conclusiones:

1. Con la obtención de los requisitos funcionales y no funcionales se garantiza una propuesta de solución que satisface las necesidades del cliente, además se identificaron las funcionalidades del sistema y los atributos que debe mostrar el mismo.
2. Los artefactos generados según la metodología AUP-UCI, la arquitectura y los patrones de diseño crearon las bases necesarias para la construcción de la propuesta de solución. También ayudó al programador a entender las funcionalidades que desea el cliente, implementando un sistema conforme a las necesidades del mismo.

CAPÍTULO III: VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

En el presente capítulo se aborda lo referente a la implementación y prueba del sistema. Una de las disciplinas críticas en el desarrollo de software es la implementación, pues aquí es donde se materializa el análisis y el diseño. Se muestra la organización del mismo mediante el modelo de implementación utilizado, el diagrama de componentes, así como las pruebas efectuadas para verificar el buen funcionamiento del sistema.

III.1 Modelo Implementación

El modelo de implementación es comprendido por un conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Entre los componentes podemos encontrar datos, archivos, ejecutables, código fuente y los directorios. Fundamentalmente, se describe la relación que existe desde los paquetes y clases del modelo de diseño a subsistemas y componentes físicos. Como parte del correcto desarrollo del modelo de implementación se obtiene el diagrama de componentes que a continuación se presenta.

III.2 Diagrama de componente

Un diagrama de componentes proporciona una visión general del sistema y documenta la organización de los componentes del sistema, sus relaciones y dependencias mutuas. Los diagramas de componentes proporcionan una visión orientada a la ejecución, es decir, dan al desarrollador información sobre el sistema, si funciona de forma coherente y cumple sus tareas y objetivos (SOMMERVILLE 2011).

Los diagramas de componentes UML representan las relaciones entre los componentes individuales del sistema mediante una vista de diseño estática. Pueden ilustrar aspectos de modelado lógico y físico.

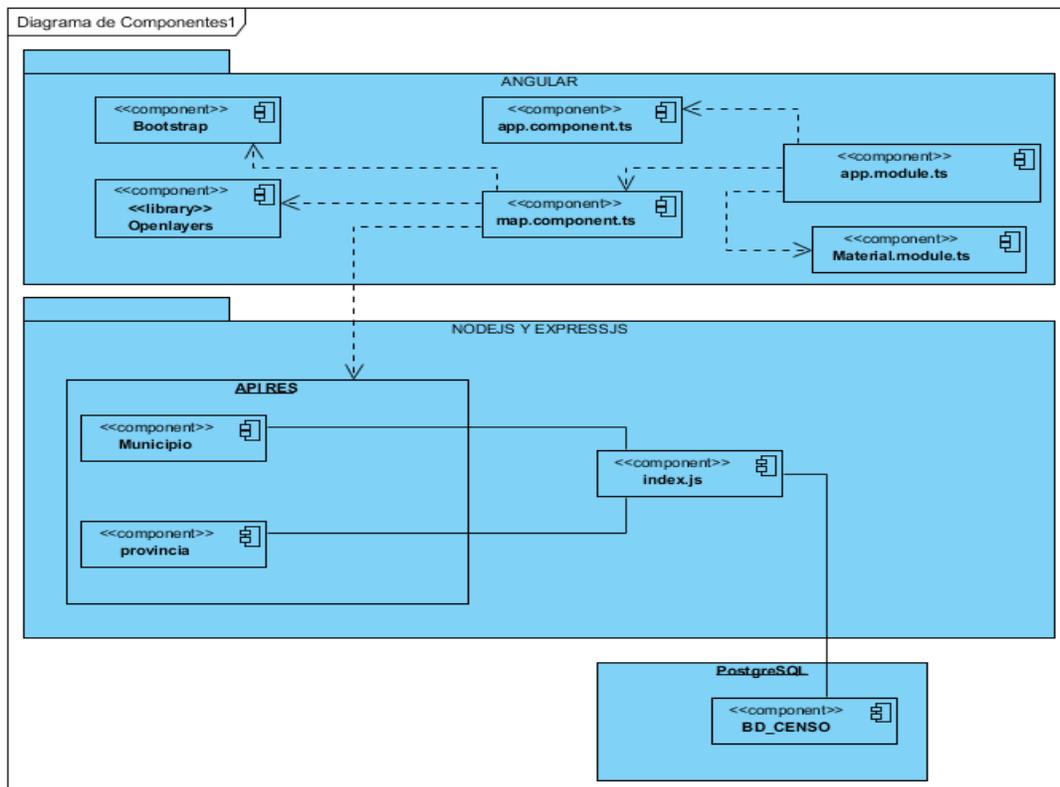


Figura 14: Modelo de componente

III.3 Estándares de codificación

Un estándar de codificación comprende todos los aspectos de la generación del código. Usar técnicas de codificación sólida y realizar buenas prácticas de programación con vista a generar un código de alta gama es de gran importancia para la calidad del software y para obtener un buen rendimiento (Luna 2019). En la actualidad existen diferentes estándares de codificación cuya utilización favorece a la comunicación fluida entre los desarrolladores, permiten la reutilización y el mantenimiento de los sistemas.

A continuación, se muestran los estándares de codificación que fueron utilizados en el desarrollo:

Indentación:

- ❖ Para la indentación de contenidos se utilizará siempre tabuladores.

Nombres de las Variables:

- ❖ Los nombres de las variables se deberán redactar evitando ambigüedades y se deben evitar los

nombres que no sean de fácil comprensión o que no tengan ningún sentido.

- ❖ Todas las variables de un bloque de código se declararán al inicio del método en cuestión.

Nombres de las Clases:

- ❖ Los nombres de las clases comenzarán con mayúscula, deben ser significativos y explicar en lo posible el uso del elemento.

General:

- ❖ Se exceptúan el uso de las tildes y letras ñ.
- ❖ Añadir comentario descriptivo a cada declaración de variables, métodos y clases, si es necesario.
- ❖ La mayoría de los elementos se deben nombrar usando sustantivos.

III.3 Pruebas

Las pruebas de software (en inglés software *testing*) son las investigaciones empíricas y técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto a la parte interesada. Es una actividad más en el proceso de control de calidad, donde se detectan los defectos que pueda tener el software; también se verifica que todos los requisitos se hayan implementado correctamente y se verifica que haya una integración adecuada de los componentes. Además, identifica y asegura que los defectos encontrados se han corregido antes de dar por finalizado el software (Pressman 2005).

III.3.1 Pruebas Unitarias

Se encargan de probar una clase en concreto, testeando cada uno de sus métodos y viendo si dados unos parámetros de entrada, la salida es la esperada. La realización de estas pruebas debe consumir el menor tiempo posible, por lo que se hace necesario el uso de herramientas para automatizar este proceso.

Las pruebas unitarias están específicamente encaminadas al enfoque de caja blanca, ya que centran el proceso de verificación en la menor unidad del diseño: el módulo, la clase, el método, entre otros. Estas utilizan la descripción del diseño detallado como guía, para probar los caminos de control importantes, con el fin de descubrir errores en la unidad.

Angular nos proporciona 2 herramientas fuertes para realizar las pruebas, Karma y Jasmine.

Jasmine, es un framework de Desarrollo Dirigido por Comportamientos (Behavior Driven Development) para efectuar pruebas unitarias (Unit Testing) de código JavaScript. Cada día es más necesario hacer pruebas a nuestro código, para asegurar su estabilidad y hacerlo escalable. Actualmente, existen muchas librerías y frameworks en diferentes lenguajes de programación que nos ayudan a lograrlo. Con Jasmine es posible hacer

fácilmente nuestras pruebas, con una sintaxis idiomática inspirada en ScrewUnit, JSSpec, JSpec y RSpec (Ruby).

Karma, es donde se ejecutan las pruebas de Jasmine, este nos permite ejecutar las pruebas sin necesidad de tener que refrescar el navegador, si surge algún cambio en los archivos de pruebas automáticamente karma ejecutará nuevamente nuestras pruebas.

A continuación, se muestra las pruebas efectuadas al sistema con uso de las herramientas antes mencionada:

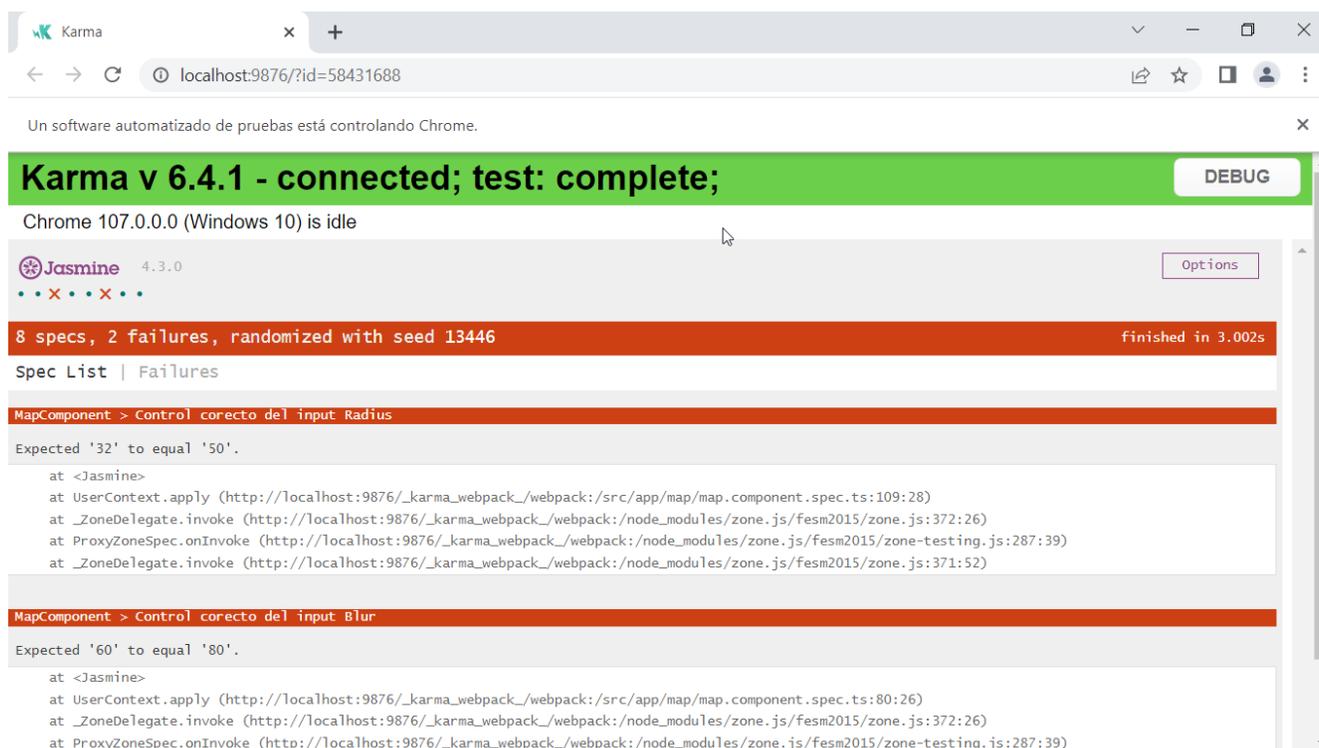


Figura 15: Prueba unitaria con Jasmine y Karma.

En la figura 15 se muestra el resultado de la primera prueba arrojando un total de dos errores.

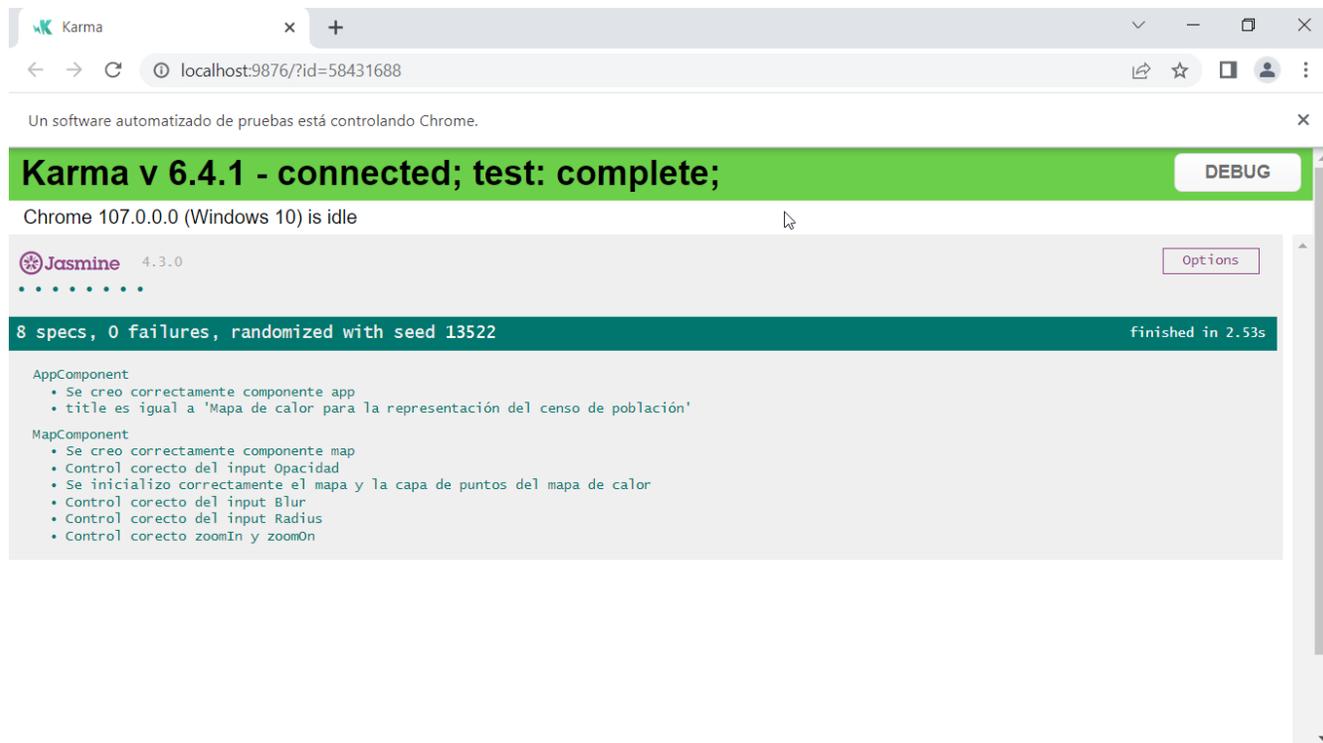


Figura 16. Prueba unitaria con Jasmine y Karma.

En la figura 16 se muestra los resultados satisfactorios arrojados por la aplicación de la herramienta de prueba.

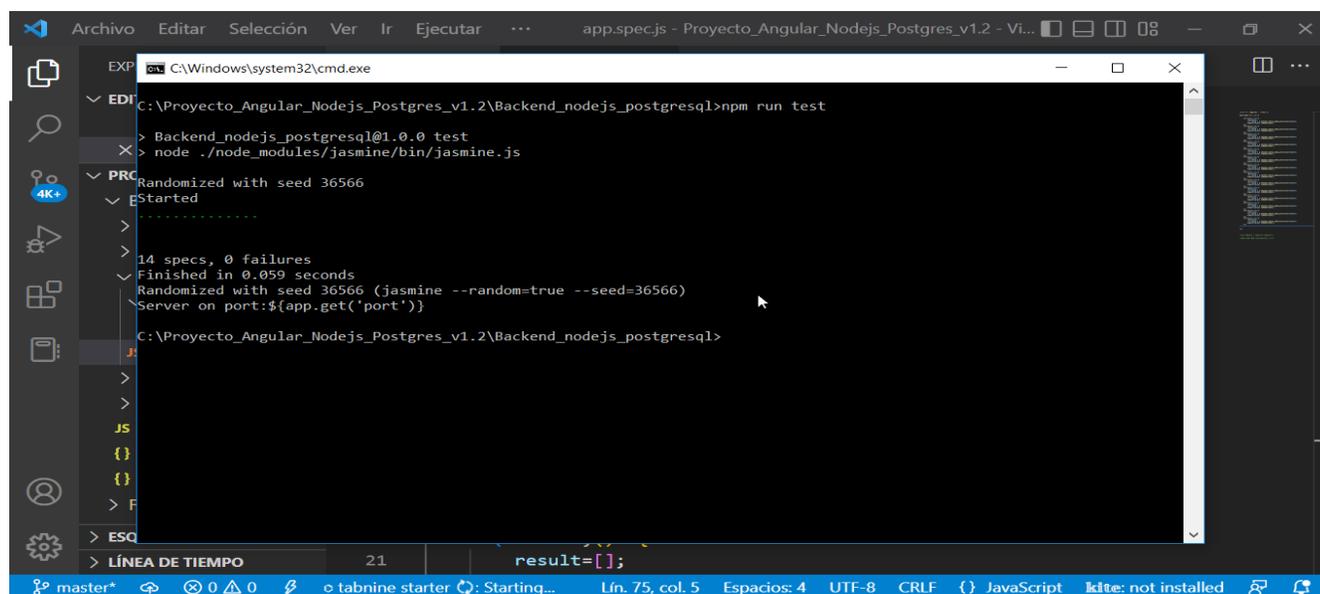
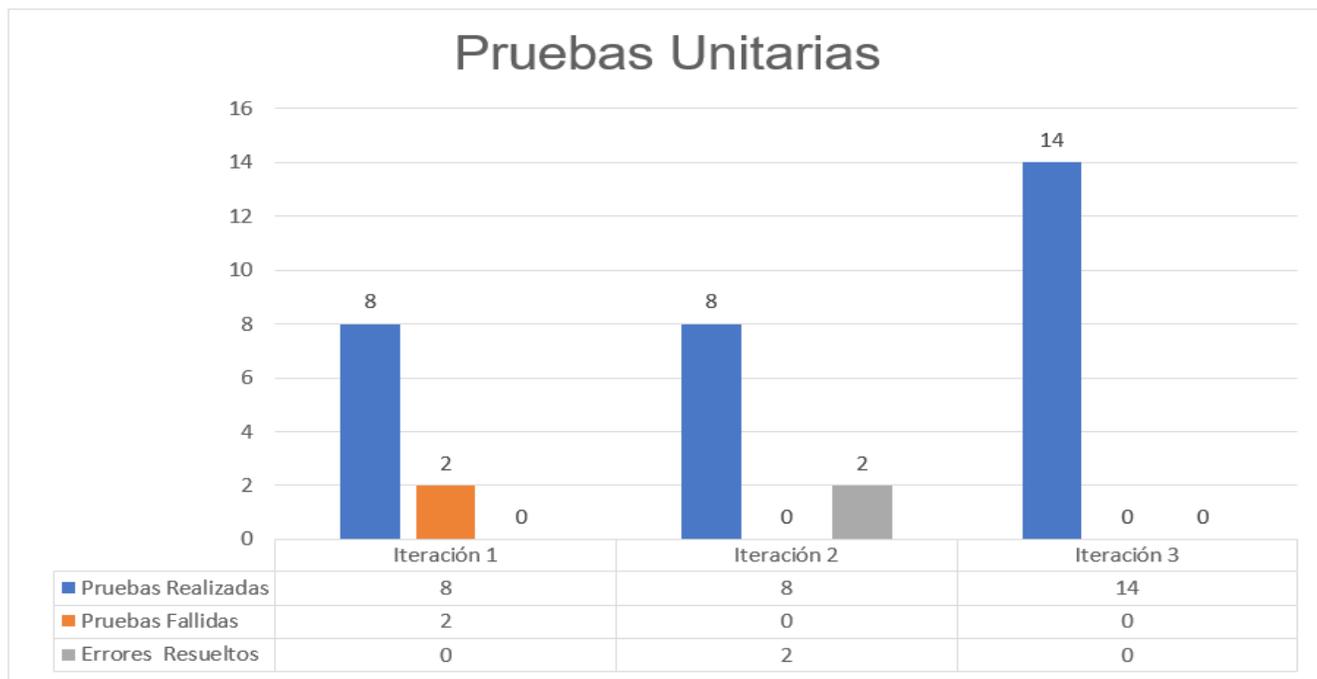


Figura 17. Prueba unitaria con Jasmine.

En la figura 17 muestra las pruebas con la herramienta Jasmine realizadas al backend.

A continuación, se muestra un gráfico con las pruebas unitarias realizadas por cada una de las iteraciones, las fallidas y los errores arreglados en cada una.



Conclusiones del capítulo

Durante el presente capítulo se diseñó el Diagrama de Componente como sustento de la implementación de la solución al problema de investigación. La definición y utilización de los estándares de codificación en la implementación del sistema propuesto permitió el desarrollo de los componentes con un alto grado de legibilidad; lo que provee una guía para el mantenimiento y actualización del sistema, con código claro y bien documentado. Las pruebas realizadas ayudaron a identificar los errores existentes en el componente para darle solución, se logró de esta forma que el producto cumpla sus expectativas deseadas.

CONCLUSIONES FINALES

Una vez culminada la investigación es posible afirmar que se cumplió el objetivo general trazado al inicio, por lo que se concluye que:

1. Se diseñó la propuesta de solución del componente y se definieron los requisitos con los que debe cumplir, a partir de los artefactos correspondientes a la metodología de desarrollo AUP-UCI en su escenario 4.
2. La identificación de los patrones de diseño y el estilo arquitectónico evidencia que la solución propuesta tiene un alto grado de adaptabilidad ante posibles modificaciones.
3. Las pruebas permitieron detectar y corregir los errores durante la implementación, posibilitando cumplir con las especificaciones requeridas y la validación de la aplicación implementada.

RECOMENDACIONES

Se recomienda:

- Ampliar los servicios implementados para el análisis de otros datos recopilados en el censo de población y viviendas.

REFERENCIAS BIBLIOGRÁFICAS

- ANDRADE, María Isabel y DE SABALAIN, Cristina Klimsza, 2001. Aplicación de un sistema de información geográfica para una Dirección de Estadística municipal. *Actas del VIII Encuentro de Geógrafos de América Latina*. 2001.
- BRITO MORALES, BRAVO y JIMÉNEZ, 2019. Aplicación Móvil para el análisis de la información captada en SIGEv3.0. . 2019. Vol. V12.N6, pp. p5.
- BUZAI, GUSTAVO D., BAXENDALE, CLAUDIA A., HUMACATA, Luis y PRINCIPI, N., 2016. Sistemas de información geográfica. *Cartografía Temática y Análisis Espacial*. Buenos Aires: Lugar Editorial. 2016.
- CAMPOS, Jéssica Menéndez y ZAMBRANO, María Zambrano, 2020. Calidad de los censos tradicionales de población y vivienda: Evaluación de sus etapas. *Matemática*. 2020. Vol. 18, no. 2.
- CEPAL, NU, 2019. Aspectos conceptuales de los censos de población y vivienda: desafíos para la definición de contenidos incluyentes en la ronda 2020. . 2019.
- CEPAL, NU, 2022. Observatorio Demográfico América Latina y el Caribe 2021. Los censos de población y vivienda de la ronda de 2020 en América Latina y el Caribe en el contexto de la pandemia: panorama regional y desafíos urgentes. . 2022.
- DE EMPRESARIOS DE ANDALUCIA, Confederación, 2014. Sistemas de Información Geográfica, tipos y aplicaciones empresariales. *Recuperado el*. 2014. Vol. 8.
- DELGADO, Dixon Fabián Flórez y GARCÍA, Deisy Katherine Fernández, 2017. Los Sistemas De Información Geográfica. Una Revisión. *Revista Facultad de Ciencias Agropecuarias-FAGROPEC*. 2017. Vol. 9, no. 1, pp. 11-16.
- FARINANGO CAIZA, Edison Patricio, 2020. *Estudio del framework angular para desarrollar aplicaciones single-page. Desarrollo del sistema de información, seguimiento y control para la hacienda ganadera La Vega*.
- FERNÁNDEZ NÚÑEZ, Héctor Manuel, 2006. SIG-ESAC: Sistema de Información Geográfica para la gestión de la estadística de salud de Cuba. *Revista cubana de higiene y epidemiología*. 2006. Vol. 44, no. 3, pp. 0-0.
- GARCÍA-HOLGADO, A., VÁZQUEZ-INGELMO, A. y GARCÍA-PEÑALVO, F. J., 2022. Ingeniería de requisitos. en línea. 5 febrero 2022. [Accedido 28 octubre 2022]. Recuperado a partir de: <https://repositorio.grial.eu/handle/grial/2515>Accepted: 2022-02-05T18:20:52Z
- GARCIA-ISLAS, 2021. Herramienta de apoyo para la enseñanza de desarrollo Web-SPA-MVC p. 79-84. . 2021. Vol. 9, pp. 79- 84.

GOMEZ, Nestor Javier, 2021. Mapa de calor del Covid-19 en Santa Fe: análisis provincial y metropolitano. . 2021.

GONZÁLEZ, Zucel Peña y CASAÑOLA, Yaimí Trujillo, 2019. Mapas de calor en sitios web. *Serie Científica de la Universidad de las Ciencias Informáticas*. 2019. Vol. 12, no. 9, pp. 97-111.

GU, Zuguang, EILS, Roland y SCHLESNER, Matthias, 2016. Complex heatmaps reveal patterns and correlations in multidimensional genomic data. *Bioinformatics*. 2016. Vol. 32, no. 18, pp. 2847-2849.

GUERRERO, Carlos A., SUÁREZ, Johanna M. y GUTIÉRREZ, Luz E., 2013. Patrones de Diseño GOF (The Gang of Four) en el contexto de Procesos de Desarrollo de Aplicaciones Orientadas a la Web. *Información tecnológica*. 2013. Vol. 24, no. 3, pp. 103-114. DOI 10.4067/S0718-07642013000300012.

HASHEMI, Negar, TAHIR, Amjed y RASHEED, Shawn, 2022. An empirical study of flaky tests in javascript. *arXiv preprint arXiv:2207.01047*. 2022.

HE, Yuqin, BI, Zhenbo, TIAN, Haobin, DUAN, Kai, WU, Jiashuai y WANG, Hongwei, 2019. Application of OpenLayers in marine information monitoring. *E3S Web of Conferences*. 2019. Vol. 118, pp. 03006. DOI 10.1051/e3sconf/201911803006.

INEC, 2022. 2. Mapas temáticos del Instituto Nacional de Estadísticas y Censos de Costa Rica. en línea. 2022. Recuperado a partir de: <https://www.inec.cr/cartografia/mapas-tematicos> Instituto Nacional de Estadísticas y Censos de Costa Rica

INSMET, 2022. en línea. [Accedido 3 noviembre 2022]. Recuperado a partir de: <http://www.insmet.cu/asp/genesis.asp?TB0=PLANTILLAS&TB1=RADAR&TB2=../Radar/04Camaguey/cmwMAXw01a.gif>

JIMÉNEZ, Antonio Argüeso, 2019. Los censos de población y viviendas de 2021 en España se basarán en registros administrativos. *Indice: Revista de Estadística y Sociedad*. 2019. No. 74, pp. 10-12.

JUBA, Salahaldin y VOLKOV, Andrey, 2019. *Learning PostgreSQL 11: A beginner's guide to building high-performance PostgreSQL database solutions*. Packt Publishing Ltd. ISBN 1-78953-521-2.

LUNA, Fernando, 2019. *JavaScript-Aprende a programar en el lenguaje de la web*. RedUsers. ISBN 987-49580-8-1.

MARDAN, Azat, 2014. *Express.js Guide: The Comprehensive Book on Express.js*. Azat Mardan.

MARDAN, Azat, 2018. Using Express.js to create Node.js web apps. En: *Practical Node.js*. Springer. pp. 51-87.

MARIÑO, Ledesma y DAVID, Luis, 2018. Modelo conceptual para el aseguramiento de la calidad de los requerimientos en proyectos de software. en línea. 21 junio 2018. [Accedido 28 octubre 2022]. Recuperado a partir de: <http://repositorio.ucp.edu.co/handle/10785/5017> Accepted: 2018-10-17T22:08:20Z

- NOVAC, Ovidiu Constantin, MADAR, Damaris Emilia, NOVAC, Cornelia Mihaela, BUJDOSÓ, Gyöngyi, OPROESCU, Mihai y GAL, Teofil, 2021. Comparative study of some applications made in the Angular and Vue.js frameworks. En: *2021 16th International Conference on Engineering of Modern Electric Systems (EMES)*. junio 2021. pp. 1-4. DOI 10.1109/EMES52337.2021.9484150.
- ONEI, 2014. *Informe nacional: censo de población y viviendas. Cuba 2012*. ONEI La Habana.
- OPENLAYERS, 2022. OpenLayers - Welcome. en línea. 2022. [Accedido 11 junio 2022]. Recuperado a partir de: <https://openlayers.org/>
- ORTEGA, Gilberto Andrés Vargas, 2021. Lineamientos para el diseño de aplicaciones web soportados en patrones GRASP. *Ciencia e Ingeniería*. 20 noviembre 2021. Vol. 8, no. 2, pp. e5716304-e5716304.
- PEÑA, Dayana, BAQUERO HERNÁNDEZ, Lionel, MAR CORNELIO, Omar y RODRIGUEZ VALDÉS, Osviel, 2016. *Extensión de la herramienta Visual Paradigm for UML para la evaluación y corrección de Diagramas de Casos de Uso*.
- PERALES MIRANDA, Víctor Hugo, PÁRRAGA GUACHALLA, Marcelo Gonzalo y USNAYO SIRPA, Jhosep Luis, 2021. Censo en Bolivia: apuntes para la construcción de un indicador de multilocalidad. *Temas Sociales*. 2021. No. 49, pp. 138-165.
- PRESSMAN, Roger S., 2005. *Software engineering: a practitioner's approach*. Palgrave macmillan. ISBN 0-07-301933-X.
- RAMÍREZ MARTÍN y RODRÍGUEZ DONATIEN, 2009. Sistema para la Identificación de Aguas en Pozos Petroleros (SIAPP). . 2009. La Habana
- RODRÍGUEZ CÁRDENAS, Alejandro y DUARTE JIMÉNEZ, Karen Tatiana, 2019. Desarrollo de una aplicación web para generar mapas de calor a partir de los incidentes viales en la ciudad de Bogotá. . 2019.
- SÁNCHEZ, Tamara Rodríguez, 2015. Metodología de desarrollo para la Actividad productiva de la UCI. *La Habana: sn*. 2015.
- SCHMITTGEN, Thomas D. y LIVAK, Kenneth J., 2008. Analyzing real-time PCR data by the comparative CT method. *Nature protocols*. 2008. Vol. 3, no. 6, pp. 1101-1108.
- SIAHAAN, Vivian y SIANIPAR, Rismon Hasiholan, 2019. *POSTGRESQL FOR PYTHON GUI: A Progressive Tutorial to Develop Database Project*. SPARTA PUBLISHING.
- SOMMERVILLE, 2011. *Ingeniería del Software*.
- SOMMERVILLE, Ian, 2005. *Ingeniería del software*. Pearson Educación. ISBN 978-84-7829-074-1. Google-Books-ID: gQWd49zSut4C
- VANEGAS, Claudia Elena Durango, 2014. Asociación de datos espacio-temporales en bases de datos Oracle. *Ingenierías USBmed*. 2014. Vol. 5, no. 2, pp. 100-108.

VIDAL, Cristian L., SCHMAL, Rodolfo F., RIVERO, Sabino y VILLARROEL, Rodolfo H., 2012. Extensión del Diagrama de Secuencias UML (Lenguaje de Modelado Unificado) para el Modelado Orientado a Aspectos. *Información tecnológica*. 2012. Vol. 23, no. 6, pp. 51-62. DOI 10.4067/S0718-07642012000600007.

WILKINSON, Leland y FRIENDLY, Michael, 2009. The history of the cluster heat map. *The American Statistician*. 2009. Vol. 63, no. 2, pp. 179-184.

ANEXOS

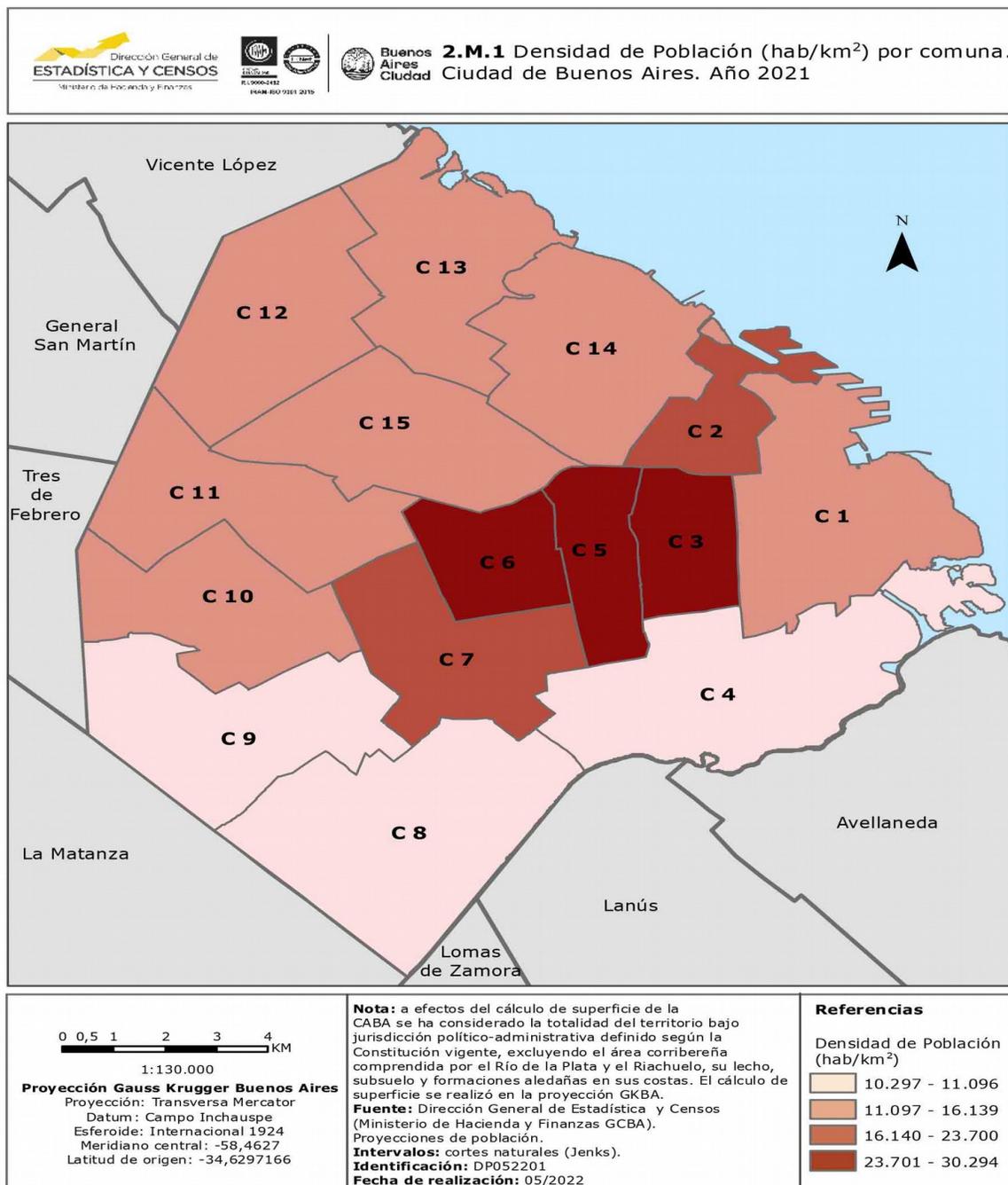


Figura 18: Mapa temático Densidad Poblacional por comuna ciudad de buenos aire