



CONSTRUCCIÓN DE GUÍAS DE DESARROLLO PARA ANGULAR

PROPONENTE(S)

Andrés Felipe Cárdenas Alarcón

C.C 1.002.366.495

2273661

DIRECTOR

Sergio Arley Puerto Moreno

Tunja
Marzo 2024

INGENIERÍA DE SISTEMAS	PÁGINA 2 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

CONTENIDO

<u>1. FICHA TÉCNICA DEL PROYECTO.....</u>	<u>7</u>
<u>2. PLANTEAMIENTO DEL PROBLEMA</u>	<u>8</u>
<u>3. JUSTIFICACIÓN.....</u>	<u>14</u>
<u>4. OBJETIVOS.....</u>	<u>16</u>
4.1. OBJETIVO GENERAL:	16
4.2. OBJETIVOS ESPECÍFICOS.....	16
<u>5. METODOLOGÍA</u>	<u>17</u>
5.1. DESARROLLO TECNOLÓGICO	21
<u>6. DESARROLLO DEL PROYECTO</u>	<u>22</u>
6.1. REQUISITOS PREVIOS DEL LECTOR.....	22
6.2. ¿POR DÓNDE EMPEZAMOS?	23
6.3. INSTALACIÓN DE HERRAMIENTAS	23
6.4. PRESENTACIÓN.....	44
6.4.1. CARACTERÍSTICAS DE LA MÁQUINA A UTILIZAR EN DESARROLLO	45
<u>7. REPOSITORIOS GIT CON EL “SCAFFOLDING” DE LAS GUÍAS 01, 02 Y 03.....</u>	<u>46</u>
7.1. REPOSITORIO CON EL BACKEND PARA LA GUÍA03	47
7.1.1. PUESTA EN MARCHA DEL PROYECTO BACKEND EN LOCAL	48
7.2. GUIA01	51
7.3. GUIA02	51
7.4. GUIA03	51
7.5. ENCUESTA DE PERCEPCIÓN.....	52
<u>8. HERRAMIENTA SOFTWARE.....</u>	<u>60</u>
<u>9. CONCLUSIONES</u>	<u>64</u>

INGENIERÍA DE SISTEMAS	PÁGINA 3 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

<u>10. REFERENCIAS.....</u>	66
<u>11. ANEXO 1: ETAPA 1 VALIDACIÓN DE LAS GUÍAS PROYECTO MINTIC - 2022..</u>	68
11.1. INTRODUCCIÓN	68
11.2. OBJETIVOS.....	69
11.3. GUÍAS DE DESARROLLO MINTIC.....	70
11.4. PROCESO DE VALIDACIÓN.....	71
11.5. CAMBIOS Y MEJORAS IDENTIFICADAS.....	72
11.5.1. MEJORAS POR INCLUIR.....	73
11.6. CONCLUSIONES	76
<u>12. ANEXO 2: ETAPA 2 Y 3 DOCUMENTACIÓN Y ESTRUCTURA DE LOS PROTOCOLOS PARA LAS NUEVAS GUÍAS</u>	77
12.1. CONSTRUCCIÓN DEL TALLER FRONTEND [GUIA01].....	78
12.2. CONSTRUCCIÓN DEL TALLERES FRONTEND [GUIA02].....	79
12.3. CONSTRUCCIÓN DEL TALLER FRONTEND [GUÍA 03].....	80

INGENIERÍA DE SISTEMAS	PÁGINA 4 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

LISTA DE FIGURAS

FIGURA 5 CURVA DE APRENDIZAJE.....	10
FIGURA 6 ENCUESTA POPULARIDAD EN LENGUAJES.....	11
FIGURA 7 GRAFICO DE BÚSQUEDAS EN GOOGLE.....	11
FIGURA 8 ÁRBOL DE PROBLEMAS.....	13
FIGURA 9 HERRAMIENTA AGREGAR O QUITAR PROGRAMAS.....	24
FIGURA 10 DESINSTALACIÓN DE NODE.....	24
FIGURA 11 CONSOLA DE COMANDOS VERIFICACIÓN DESINSTALACIÓN DE NODEJS.....	25
FIGURA 12 PASOS PARA MOSTRAR CARPETAS OCULTAS.....	26
FIGURA 13 EXPLORADOR DE ARCHIVOS PARA ELIMINAR CARPETAS NPM.....	26
FIGURA 14 SITIO WEB OFICIAL DE NODE JS.....	27
FIGURA 15 PROGRESO DE INSTALACIÓN DE NODE JS.....	28
FIGURA 16 VERIFICACIÓN DE LA INSTALACIÓN NODEJS Y NPM.....	29
FIGURA 17 UTILIZANDO NPM PARA INSTALAR ANGULAR.....	30
FIGURA 18 VERSIÓN INSTALADA DE ANGULAR, NODEJS Y NPM.....	30
FIGURA 19 INSTALACIÓN DE TYPESCRIPT Y SU VERSIÓN.....	32
FIGURA 20 SITIO WEB DE VISUAL STUDIO CODE.....	33
FIGURA 21 INSTALACIÓN CORRECTA DE VISUAL STUDIO CODE.....	34
FIGURA 22 ACTUALIZACIÓN DE VISUAL STUDIO CODE.....	35
FIGURA 23 EXTENSIONES PARA ANGULAR EN VCODE.....	36
FIGURA 24. EXTENSIÓN PARA IMPORTACIÓN DE FILES.....	36
FIGURA 25 RECOMENDACIÓN DEL AUTO GUARDADO ACTIVO PARA VISUAL STUDIO CODE.....	37
FIGURA 26 SISTEMAS OPERATIVOS DISPONIBLES PARA DESCARGAR POSTGRESQL.....	38
FIGURA 27 VERSIÓN DE POSTGRESQL A UTILIZAR.....	38
FIGURA 28 HERRAMIENTA PARA LA ADMINISTRACIÓN - PGADMIN.....	39
FIGURA 29 VENTANA PARA CREAR USUARIO EN EL PGADMIN.....	40
FIGURA 30 VENTANA PARA LA CREACIÓN DE USUARIOS - PRIVILEGIOS.....	41
FIGURA 31 VENTANA PATA LA CREACIÓN DE USUARIOS - PRIVILEGIOS.....	42
FIGURA 32 ESTRUCTURA DE CARPETAS.....	42
FIGURA 33 POWERSHELL EN MODO ADMINISTRADOR.....	43
FIGURA 34 CREACIÓN DE CARPETAS POR COMANDOS.....	43
FIGURA 35 REPOSITORIO GIT DE LAS GUÍAS.....	46
FIGURA 36 INSTALACIÓN DE NODE_MODULES.....	47
FIGURA 37 REPOSITORIO BACKEND API.....	47
FIGURA 38 BACKEND CARPETA DESCOMPRESIONADA.....	48
FIGURA 39 "CODE ." PARA ABRIR EL BACKEND EN VCODE.....	48
FIGURA 40 INSTALACIÓN DE LIBRERÍAS "NPM I".....	49
FIGURA 41 CREACIÓN DE LA BASE DE DATOS.....	49
FIGURA 42 SCRIPT DE LA BASE DE DATOS.....	50
FIGURA 43 COMANDO PARA INICIAR EL BACKEND.....	50
FIGURA 44 CADENCIALES DE LA BD EN EL ARCHIVO "VARIABLES.ENV".....	51
FIGURA 45 ¿PREGUNTA, ES ÚTIL LA PROBLEMÁTICA EJEMPLO Y LOS TALLERES?.....	52

INGENIERÍA DE SISTEMAS	PÁGINA 5 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

FIGURA 46 ¿PREGUNTA, CUBRE EL CONTENIDO LOS FUNDAMENTOS DE ANGULARJS?	52
FIGURA 47 ¿PREGUNTA, ESTÁ DE ACUERDO CON LA ORGANIZACIÓN?	53
FIGURA 48 ¿PREGUNTA, ¿CÓMO CALIFICA LA UTILIDAD GENERAL DEL MATERIAL?	53
FIGURA 49 ¿PREGUNTA, ESTARÍA DISPUESTO A USAR ESTE MATERIAL EN CLASES?	54
FIGURA 50 ¿PREGUNTA, ALGÚN TEMA EN ESPECÍFICO?	54
FIGURA 51 ¿PREGUNTA, ASPECTOS A MEJORAR?	55
FIGURA 52 RETOS PARA EL LECTOR	56
FIGURA 53 ¿PREGUNTA, TIENE ALGUNA SUGERENCIA?	56
FIGURA 54 ¿PREGUNTA, ALGÚN COMENTARIO ADICIONAL?	57
FIGURA 55 PANTALLA PRINCIPAL	60
FIGURA 56 SECCIÓN DE CIUDADES	60
FIGURA 57 SECCIÓN DE PRODUCTOS	61
FIGURA 58 SECCIÓN DE RESTAURANTES	61
FIGURA 59 INICIO DE SESIÓN USUARIO.....	62
FIGURA 60 REGISTRO DE USUARIO.....	62
FIGURA 61 CREACIÓN DE PRODUCTOS	63
FIGURA 62 ADMINISTRAR PRODUCTOS	63
FIGURA 63 <i>ERROR EN CONSOLA EN LAS ANTERIORES GUÍAS</i>	72
FIGURA 64 <i>ERROR EN LA IMPORTACIÓN DE JAVASCRIPTS</i>	73
FIGURA 65 <i>CORRECCIÓN DE IMPORTACIÓN DE JAVASCRIPTS</i>	73

INGENIERÍA DE SISTEMAS	PÁGINA 6 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

LISTA DE TABLAS

TABLA 1. <i>STAKEHOLDER</i>	14
TABLA 2 CATEGORÍAS DE LA METODOLOGÍA	17
TABLA 3. <i>CARACTERÍSTICAS DE LA MAQUINA USADA</i>	45
TABLA 4 EJEMPLO ¡¡¡BONUS!!! A RECURSOS ADICIONALES	57
TABLA 5 <i>COMPARATIVA ENTRE TENER Y NO TENER UN MATERIAL DE APRENDIZAJE.</i>	77

INGENIERÍA DE SISTEMAS	PÁGINA 8 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

2. PLANTEAMIENTO DEL PROBLEMA

En el artículo titulado "*Angular, React o Vue ¿cuál elegir?*" [1], se determina que aprender lenguajes de programación web como React.js o Vue.js, sugiere en el programador la necesidad de mejorar y codificar en términos de robustez, procesamiento de carga del DOM (Document Object Model) de una página web, proyectos de alto valor, productividad e infraestructura escalable. Al elegir una tecnología, el programador tiene un objetivo muy claro: "sentirse útil", poder documentarse fácilmente y contar con una gran comunidad que le permita encontrar soluciones a problemas de código, agilidad en relación entre tiempo y líneas de código escrito, procesamiento de información y programación precisa.

En este contexto, en el año 2012, "Google y sus equipos de programadores publicaron AngularCLI, un framework de código abierto con la funcionalidad de crear aplicaciones web de una sola página o interfaz (single-page)" [2].

AngularJS con TypeScript es uno de los "framework más populares utilizados para crear sitios web dinámicos, ¿Y qué hace a Angular tan especial? Para comenzar, es importante comprender la interacción de JavaScript en una página web, este actúa a manera de puente entre la interacción del usuario y los estilos definidos en HTML (HyperText Markup Language) y CSS (Cascading Style Sheets)" [3]. Sin embargo, la flexibilidad de JavaScript puede dar lugar a lo que se conoce como "código maloliente" (smell code) [2]. Algunos errores comunes en la comunidad de programadores de JavaScript incluyen:

- **Uso de variables globales:** Esto dificulta el seguimiento, lectura e interpretación del código, ya que las variables pueden cambiar de tipo o valor en cualquier momento [4].
- **Falta de estándares de codificación:** La falta de seguimiento de estándares de codificación establecidos puede resultar en un código inconsistente y difícil de entender para otros programadores. Además, la elección de nombres de variables no descriptivos dificulta la interpretación del código [4].

INGENIERÍA DE SISTEMAS	PÁGINA 9 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

- **Manejo inadecuado de errores y cambios en las variables:** La falta de un adecuado manejo de errores puede conducir a comportamientos inesperados con dificultades en la depuración de código, afectando el rendimiento y la eficiencia del programa [4].
- **Código inalcanzable (unreachable code):** El código inalcanzable que nunca se ejecuta y carece de flujo de control o funcionalidad puede afectar la legibilidad y mantenibilidad del código, así como introducir comportamientos inesperados [4].

“AngularJS se destaca por utilizar el patrón o arquitectura Modelo-Vista-Controlador (MVC) para construir aplicaciones web de una sola página” [2]. En el artículo titulado "AngularJS: ¿Qué es y por qué debería usarlo en mis aplicaciones web?", explica que este framework consta de tres componentes principales:

- “Modelo: Gestiona la estructura de datos y recibe información del controlador.
- Vista: La representación de la información.
- Controlador: Responde a la entrada de datos e interactúa con el modelo” [5].

AngularJS aborda los problemas de "código maloliente" mencionados anteriormente, permitiendo una mejor estructura y organización del código, introduciendo TypeScript (TS) que es una versión mejorada y más rígida de JavaScript, lo que “obliga al programador a escribir un código más claro y limpio en términos de manejo de datos (inmutabilidad)” [6]. Finalmente, el código TypeScript se compila a JavaScript para una ejecución más rápida en el navegador.

“Para hacerlo más simple y visible:

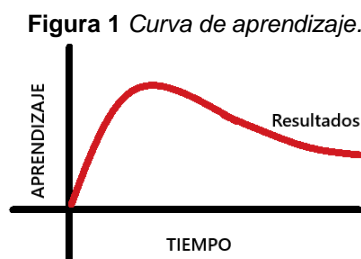
- AngularJS compila a JavaScript y une el HTML.
- JavaScript acepta la entrada del usuario y la envía a AngularJS.
- AngularJS usa la entrada para modificar HTML.” [2]

En resumen, AngularJS se caracteriza por un enfoque estructurado y una arquitectura sólida que mejora la productividad del desarrollo. Sin embargo, “la curva de aprendizaje

INGENIERÍA DE SISTEMAS	PÁGINA 10 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

puede ser más pronunciada en comparación con otros lenguajes y framework, especialmente para aquellos que también deban aprender TypeScript” [1]. AngularJS es una tecnología robusta, pero puede resultar abrumadora para los nuevos programadores debido a su estructura rígida y la cantidad de conceptos que debe dominar.

Como se expone en el artículo “*Angular, React o Vue ¿cuál elegir?*” “Cuando se habla de la curva de aprendizaje nos referimos a la Figura 1 que describe qué tan complicado es aprender algo y qué tan pronto se pueden obtener beneficios y resultados del aprendizaje”.



Fuente
Angular, React o Vue ¿cuál elegir? [1]

“Naturalmente, la curva de aprendizaje de Angular es un poco más alta que la de otros lenguajes en ciertos principios, pero a medida que uno profundiza, la productividad del desarrollo aumentará”[1].

La popularidad de una tecnología también influye un papel importante en la elección de aprenderla. Según una encuesta de StackOverflow en 2021, expuesto en el artículo “*Angular vs React: Una Comparación En Profundidad*”, como se ve en la Figura 2 el 40,14% de 67 mil encuestados eligió React.js, mientras que el 22,96% optó por Angular además indicando que va en constante aumento la popularidad de esta tecnología [7].

Figura 2 Encuesta popularidad en lenguajes.

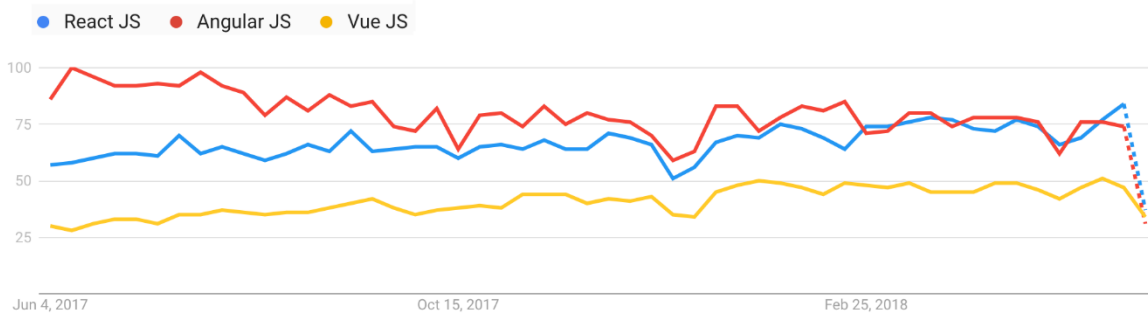


Fuente StackOverflow.

Aunque React.js es más cómodo de aprender, los programadores tienden a migrar a AngularJS cuando se enfrentan a proyectos más grandes y robustos [8].

Con lo anterior dicho en otros análisis más recientes, en un fragmento del artículo “*React vs Angular vs Vue, ¿cuál es la más popular?*”, se observa un crecimiento en la popularidad de Angular en las tendencias de búsqueda de Google Trends, apunta a que cada vez la brecha entre React.js y Angular se vayan reduciendo con relación a la popularidad se ven en la Figura 3.

Figura 3 Grafico de búsquedas en Google.



Fuente Google Trends.

Como se observa en la Figura 3, un primer indicador de tendencias en la comunidad, puede considerarse “suficientemente adecuado como una tecnología emergente que se está haciendo popular y dará de que hablar con el paso de los días” [7].

INGENIERÍA DE SISTEMAS	PÁGINA 12 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

Estos análisis sugieren a la universidad en considerar la adopción de tecnologías emergentes en sus programas académicos al preparar a los estudiantes para las demandas del mercado laboral. Además, el salario inicial promedio de un Angular Developer en Colombia es de \$42'000.000 al año, mientras que un desarrollador en React es de 36'000.00 al año, esto según fuentes como talent.com y co.indeed.com, agrega un incentivo adicional para aprender esta tecnología.

Finalmente, la elección de aprender AngularJS puede ser desafiante debido a su curva de aprendizaje y a la necesidad de dominar TypeScript, pero que ofrece oportunidades en el mercado laboral y proporciona una base sólida para el desarrollo de aplicaciones web modernas.

GitHub es una plataforma de desarrollo que utiliza el sistema operativo Git, herramienta de control de versiones de manera local. GitHub permite subir, alojar y compartir proyectos en la nube. Así mismo los programadores pueden almacenar y administrar el código, registrar cambios o volver a versiones anteriores. Además, actúa como una red social que publica los proyectos de los programadores a usuarios invitados o colaboradores, trabajando todos bajo un mismo archivo. Los desarrolladores pueden alojar proyectos de código abierto de forma gratuita [9].

Teniendo en cuenta la información expuesta y profundizando en el análisis de esta problemática, surge la imperante necesidad de crear un material de estudio destinado a iniciantes en el tema o ilustrados en el uso de AngularJs, que aspiren en adquirir conocimientos sobre las últimas actualizaciones y mejoras incorporadas en este lenguaje de programación. Abordando temas como la instalación, documentación, tutoriales y talleres aplicados, todo con el objetivo principal de facilitar tanto el proceso de enseñanza como el aprendizaje de esta tecnología.

A continuación, en la

Figura 4 se describen las problemáticas que se evidenciaron frente a esta tecnología, todo lo relacionado con la ausencia de información confiable, casos en donde el material de

aprendizaje se encuentra obsoleto, debido a los cambios entre versiones, e información poco intuitiva, dando la percepción de ser un a tecnología compleja con respecto a otras en el mercado.

Figura 4 *Árbol de problemas.*



Fuente Autor.

INGENIERÍA DE SISTEMAS	PÁGINA 14 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

3. JUSTIFICACIÓN

Las tendencias de popularidad, robustez y escalabilidad en el mercado laboral están estrechamente relacionadas con la evolución y el crecimiento de las tecnologías [8]. Estas tendencias surgen tanto en el programador como en su entorno laboral, impulsado por la necesidad de adquirir conocimientos en tecnologías que le permitan un desarrollo tanto intelectual como económico [1]. Esto se debe al crecimiento sostenido de las tecnologías y a las demandas del mercado laboral que influyen en la elección y el aprendizaje de nuevas habilidades, analizadas en los dos artículos anteriores.

El propósito de este trabajo es crear un material de estudio o de docencia a manera de guías de conocimiento PDF, destinado a facilitar los procesos de formación y aprendizaje de esta tecnología, con información clara, vigente y actualizada.

En beneficios sociales, se busca contribuir a la comunidad de ingenieros de sistemas de la Universidad, ayudando y colocando a disposición un material de aprendizaje tecnológico en formato PDF, con código almacenado en GitHub que permitirá al lector descargar y realizar las temáticas propuestos en este material, incluyendo estudiantes e ingenieros, que también busquen adquirir conocimientos de manera autónoma, facilitar la curva de aprendizaje en los procesos de adopción y las destrezas que pide esta tecnología.

Tabla 1.
Stakeholder

Stakeholder	Intereses
Ingenieros y/o Docentes	Recursos educativos de calidad, material actualizado, de información pedagógica.
Comunidad Académica e Investigadores (Semilleros)	Acceso a información relevante sobre la evolución de AngularJS, tendencias educativas, oportunidades de investigación.
Estudiantes de Ingeniería de Sistemas	Acceso a recursos educativos actualizados, comprensión clara de conceptos de AngularJS, facilidad para aprender.

Fuente: Autor

INGENIERÍA DE SISTEMAS	PÁGINA 15 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

Por lo anteriormente expuesto, este trabajo plantea los siguientes objetivos para abordar con la validación y actualización del anterior material además de cubrir las problemáticas previamente expuestas.

INGENIERÍA DE SISTEMAS	PÁGINA 16 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

4. OBJETIVOS

4.1. Objetivo General:

Construir guías de conocimiento tecnológico para la implementación Frontend en TypeScript mediante el desarrollo de protocolos en la presentación de código y repositorios de descarga, para brindar instrumentos que permitan apoyar el proceso de formación y actualización de ingenieros de software.

4.2. Objetivos específicos

1. Validar las guías de desarrollo realizadas por la Facultad de Ingeniería de Sistemas en el proyecto MISION TIC 2022, para determinar la calidad y actualidad del material, implementando los protocolos definidos en las guías, utilizando las nuevas versiones de la tecnología a la fecha.
2. Definir la estructura de los protocolos para establecer los módulos y componentes que deben ser incluidos en el material, utilizando las fuentes bibliográficas propuestas para el marco de trabajo de Google en el desarrollo de aplicaciones Frontend.
3. Implementar los módulos y componentes de código que harán parte del cuerpo de los protocolos y el establecimiento de los repositorios Git para la distribución de código.
4. Verificar el funcionamiento de los protocolos desarrollando una aplicación Frontend de conformidad con los pasos definidos, tomando como referencia el código almacenado en los repositorios Git.

5. METODOLOGÍA

La investigación realizada para la construcción de guías de desarrollo para Angular se caracteriza por su enfoque descriptivo y su naturaleza cualitativa. Esta investigación tiene como objetivo identificar y facilitar un contenido en forma de guías de aprendizaje PDF, implementando Frontend en TypeScript para la tecnología AngularJS. En el proceso, se empleó la metodología Agile Scrum para el desarrollo y elaboración de las guías, lo que permite abarcar cada uno de los objetivos específico a través de iteraciones, asegurando entregas rápidas y ajustes continuos sobre el material.

La investigación descriptiva se enfocó en proporcionar guías de aprendizaje a manera de componentes, explicando y solucionando prácticas a problemáticas comunes en el desarrollo de aplicaciones. Además, al no buscar establecer relaciones causales, se clasifica como una investigación cualitativa que se centra en describir detalladamente el fenómeno del estudio, respondiendo al contexto de la creación de las guías de AngularJS.

Tabla 2 Categorías de la metodología

Categorías	Ítems	Descripción
Componentes de AngularJS	<ul style="list-style-type: none"> Fundamentos principales. Uso de directivas. Implementación de servicios. Integración entre módulos. 	Desde los conceptos básicos hasta los fundamentos propios de la tecnología como directivas, servicios, módulos. Componentes esenciales para el desarrollo de aplicaciones en AngularJS.
Soluciones prácticas	<ul style="list-style-type: none"> Implementación de patrones de diseño. Manejo de errores y banners de información. Estrategias de seguridad en aplicaciones. Integración de APIs externas. 	Se centra en la resolución de problemas, implementación de patrones de diseño, manejo de errores, seguridad y la integración de APIs externas. Proporciona enfoques prácticos para abordar desafíos específicos en el desarrollo.
Metodología Agile Scrum	<ul style="list-style-type: none"> Planificación de iteraciones sobre el material PDF. 	Metodología ágil utilizada en el proceso de desarrollo de las guías. Incluye la planificación de iteraciones y tareas, con fechas de entrega

INGENIERÍA DE SISTEMAS	PÁGINA 18 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

	<ul style="list-style-type: none"> • Seguimiento de tareas y avances en el proyecto. • Retroalimentación y mejora continua en el proceso de desarrollo • Ajustes ágiles a cambios y requerimientos del proyecto. 	para posteriormente recibir retroalimentación ante nuevos cambios.
Enfoque en componentes	<ul style="list-style-type: none"> • Diseño y estructura de componentes (HTML y CSS). • Reutilización de código entre componentes. • Modularidad y escalabilidad de componentes. • Buenas prácticas para el desarrollo de componentes. 	Se centra en el diseño, reutilización, modularidad, escalabilidad e interacción de componentes en proporcionando métricas para codificar y estandarizar los componentes de forma eficientes, entendibles y estructurados.
Impacto en el aprendizaje autónomo	<ul style="list-style-type: none"> • Fomentar el autoaprendizaje y la experimentación práctica. • Evaluación de la comprensión y aplicación de conceptos a través de talleres prácticos. • Apoyo a la formación autodidacta en el desarrollo con AngularJS 	Apoya la formación autodidacta en el desarrollo facilitando el autoaprendizaje, fomentando la experimentación, y evaluando la comprensión de conceptos y la replicación de las temáticas a la hora de resolver problemáticas.

Estas categorías de la **Tabla 2** se utilizan para organizar y abordar diferentes aspectos relevantes en el proceso de investigación y desarrollo de guías de aprendizaje para AngularJS, brindando un enfoque integral que cubre desde los componentes técnicos hasta el impacto en el aprendizaje autónomo del lector.

Debido a la naturaleza de esta metodología, la creación de los componentes de las guías se llevó a cabo de manera iterativa. Esto nos proporcionará la capacidad de validar y ajustar las guías de forma constante, asegurándonos de su calidad y vigencia. Así mismo, permitiendo satisfacer de manera precisa y específica las temáticas que un ingeniero necesita para la formación.

El alcance de esta metodología consiste en identificar y establecer las características del material de aprendizaje que resultarán pertinentes para satisfacer las necesidades de los

INGENIERÍA DE SISTEMAS	PÁGINA 19 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

lectores en relación con los temas que serán abordados para esta tecnología. Esto incluye aspectos como la estructura del material y el contenido del mismo [10].

Las etapas en consideración para esta metodología son las siguientes:

Etapas 1: Revisión de la literatura

Se lleva a cabo una revisión exhaustiva de la literatura existente sobre el tema de las guías de aprendizaje, específicamente aquellas previamente desarrolladas para el proyecto MISION TIC 2022. Esto facilitará la comprensión del estado actual de las temáticas presentadas, así como de los conocimientos y metodologías ya establecidas, para luego proponer posibles nuevas mejoras y soluciones tecnológicas innovadoras. En él **ANEXO 1: ETAPA 1 VALIDACIÓN DE LAS GUÍAS PROYECTO MINTIC - 2022** se encuentra a detalle todo lo relacionado con esta etapa.

Etapas 2: Formulación de hipótesis

Se formulan hipótesis acerca de las características de los usuarios que requieran el material de aprendizaje. Además, de actualizar y agregar nuevas funcionalidades de la tecnología, consideradas relevantes a ser impartidas, con el objetivo de satisfacer las necesidades de los aprendices al acceder, aprender e implementar los módulos y componentes que contendrá este nuevo material de código en formato PDF.

Etapas 3: Construcciones de las Guías (Talleres)

Construcción de los talleres para la parte de Frontend en TypeScript se tiene propuestos una serie de talleres que serán acumulativos, estos contendrán los contenidos temáticos para el área de Frontend en Angular.

Etapas 3.1: Recopilación de datos

La funcionalidad del desarrollo de una aplicación Frontend se tomará como indicador de verificación, siguiendo los pasos acumulativos previamente definidos en los módulos de enseñanza de este material.

INGENIERÍA DE SISTEMAS	PÁGINA 20 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

En el anexo se encuentra a detalle la información, formulación y construcción a detalle de los contenidos que estarán en el nuevo material **ANEXO 2: ETAPA 2 Y 3 DOCUMENTACIÓN Y ESTRUCTURA DE LOS PROTOCOLOS PARA LAS NUEVAS GUÍAS.**

INGENIERÍA DE SISTEMAS	PÁGINA 21 DE 82
PROPUESTA DE DESARROLLO TECNOLÓGICO	VERSIÓN: 21

5.1. Desarrollo Tecnológico

El desarrollo tecnológico de la investigación de este documento se centra en la construcción de guías de desarrollo para AngularJS, con un enfoque en componentes y soluciones prácticas ante problemáticas comunes en el desarrollo de aplicaciones. Para llevar a cabo este proceso, se empleó la metodología Agile Scrum, que permitió abordar cada objetivo específico a través de iteraciones, asegurando entregas rápidas y ajustes continuos sobre el material.

El desarrollo tecnológico para esta implementación de Frontend en TypeScript en la tecnología AngularJS, se basa con el objetivo de proporcionar una serie de temas paso a paso en forma de guías de aprendizaje en formato PDF. Cubriendo temáticas como el uso de directivas, servicios, integración de módulos, implementación de patrones de diseño, manejo de errores, estrategias de seguridad, integración de APIs externas, entre otros.

También, se recalca la importancia de facilitar el aprendizaje autónomo para esta tecnología, fomentando la experimentación, evaluación de la comprensión y aplicación de conceptos, estimulando la creatividad y resolución de problemas, con el fin de apoyar la formación autodidacta en el desarrollo con AngularJS.

Por último, el desarrollo tecnológico de esta investigación se orienta a la creación de guías de aprendizaje, con apartados teórico-prácticos y cierto material de apoyo almacenado en repositorio GitHub para ayudar a la experiencia del lector a la hora de replicar las temáticas y mediante una metodología ágil se garantizar la calidad, relevancia y actualización constante de este del material proporcionado.

6. DESARROLLO DEL PROYECTO

6.1. Requisitos previos del lector

Para empezar a aprender de Angular, es importante disponer de algunos conocimientos y manejo de algunas de las herramientas básicas que se trabajarán y se usarán a lo largo de este material. En la siguiente lista se presentan los requisitos previos con los que debe estar familiarizado.

- **Conocimientos básicos de JavaScript:** Angular se basa bajo el lenguaje de programación interpretado JavaScript, por lo hay que tener un cierto conocimiento del lenguaje [11].
- **Familiaridad con la terminal CMD:** A lo largo del proceso de instalación y cuando se utiliza Angular, la creación de componentes y otros archivos, se trabajan bajo la línea de comandos (en Windows) o terminal (en macOS y Linux). La familiaridad con los comandos básicos y la navegación es esencial para empezar a utilizar Angular [11].
- **Familiaridad con TypeScript (opcional):** Aunque no es estrictamente necesario, tener un conocimiento básico de TypeScript puede ser útil cuando se trabaja con Angular, ya que el framework está construido sobre TypeScript. Para tener más información se puede remitir a la [documentación oficial](#) [11].
- **Node.js y npm instalados:** Angular requiere que tanto Node.js como el Gestor de Paquetes de Node (npm) estén instalados en su sistema. Si aún no los tiene instalados, a continuación, se presentan los pasos para esta y todas las herramientas que se usan a lo largo de este material [11].
- **Conocimientos de HTML/CSS:** A lo largo de este material, se trabajará con plantillas HTML y los estilos que provee [Bootstrap 5](#). La familiaridad que el lector tiene con respecto a los conocimientos básicos de las etiquetas HTML y su significado, así como algunos de los conceptos de CSS [11].

6.2. ¿Por dónde empezamos?

A nivel de Frontend está Angular, el cual es un Framework para desarrollar en JavaScript, a nivel de servidor está NodeJS, el cual es una plataforma que permite ejecutar JavaScript en el Servidor. ¿Cómo se puede comunicar Angular con Node?, existen varias respuestas, aunque la más utilizada hoy en día es Express [12].

Una estrategia muy acertada para desarrollar en Angular y Node, es utilizar TypeScript; TypeScript es un lenguaje de código abierto basado en JavaScript. La potencia de TypeScript es su sencillez, el tipado de datos, la orientación a objetos y el cumplimiento de estándares como EcmaScript [12].

Resumiendo, vamos a utilizar en estas guías de conocimiento: Angular, NodeJS, TypeScript, PostgreSQL y algunos otros Frameworks que harán el proceso de desarrollo mucho más interesante [12].

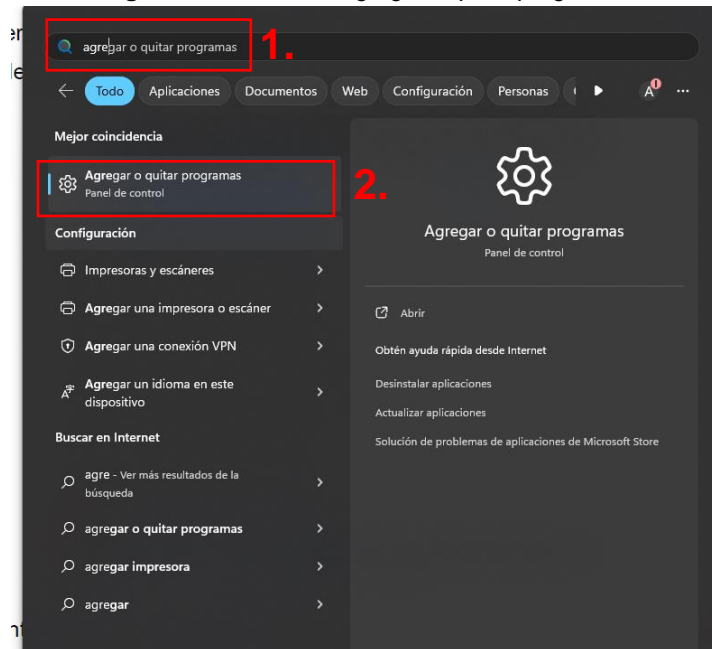
6.3. Instalación de herramientas

Si ha instalado versiones inferiores a la 16.17.5 LTS de Node JS, el siguiente apartado puede ser muy útil pues le permitirá remover las versiones antiguas e instalar una versión más actual de Node JS.

6.3.1.1 *Remover versiones anteriores Node JS*

Para remover instalaciones previas de Node JS se utilizará la herramienta agregar o quitar programas de Windows. La **Figura 5** presenta los pasos a seguir para lanzar la aplicación.

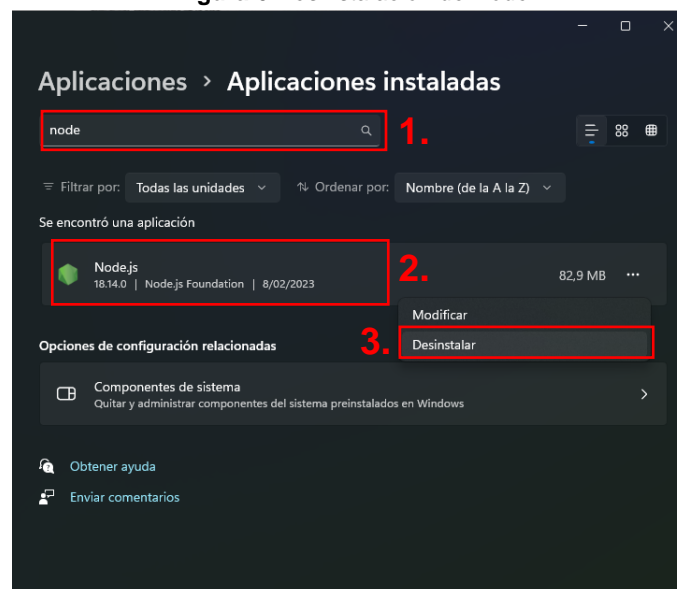
Figura 5 Herramienta agregar o quitar programas



Fuente Autor.

La herramienta para eliminar programas de Windows presenta una caja de texto en donde se puede escribir el nombre del programa que se desea eliminar, con esto se filtran las demás aplicaciones y se previenen errores. La **Figura 6** presenta la interfaz de la herramienta para eliminar programas.

Figura 6 Desinstalación de Node

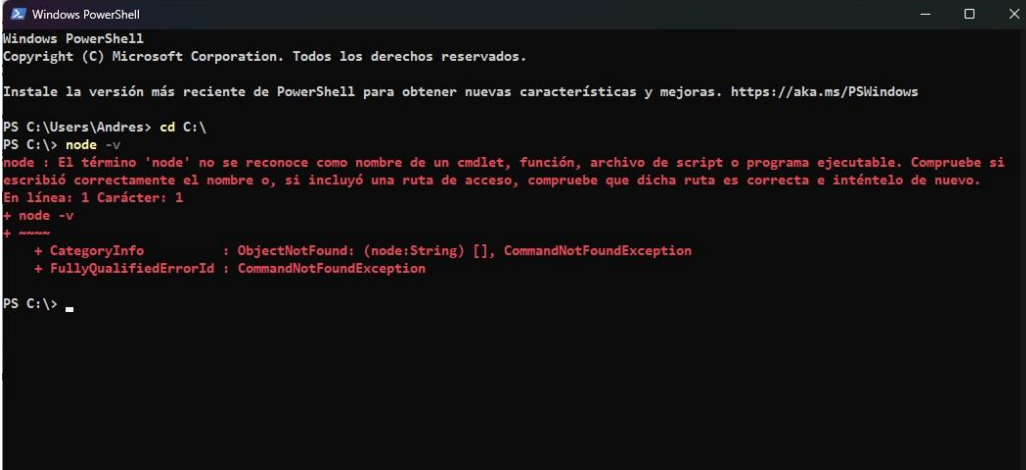


Fuente Autor.

Finalizado el proceso de desinstalación del Node JS, es posible verificar que efectivamente ha sido desinstalado, para esto, se debe abrir una consola de comandos de Windows (CMD). La **Figura 7** presenta la verificación de la desinstalación de Node JS. El comando para verificar la existencia de Node JS en el equipo es:

```
node -v
```

Figura 7 Consola de comandos verificación desinstalación de NodeJS



```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\Users\Andres> cd C:\
PS C:\> node -v
node : El término 'node' no se reconoce como nombre de un cmdlet, función, archivo de script o programa ejecutable. Compruebe si
escribió correctamente el nombre o, si incluyó una ruta de acceso, compruebe que dicha ruta es correcta e inténtelo de nuevo.
En línea: 1 Carácter: 1
+ node -v
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (node:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

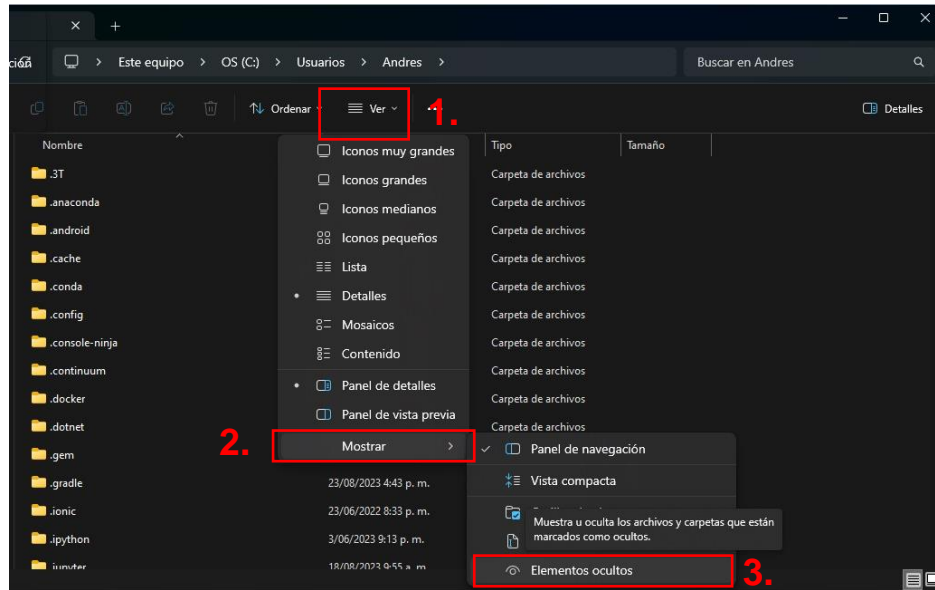
PS C:\> █
```

Fuente Autor.

Uno de los problemas frecuentes al desinstalar aplicaciones es la persistencia de archivos adicionales. La información adicional que no es eliminada generalmente corresponde a archivos de configuración o caché, los cuales son almacenados para mejorar la experiencia del usuario ante una posible nueva instalación. Sin embargo, es recomendable eliminar todo rastro de la versión anterior de Node JS y de posibles Frameworks adicionales que hayan sido instalados. La **Figura 9** presenta las carpetas que deben ser eliminadas manualmente para finalizar el proceso de desinstalación de Node JS.

Nota: Pero antes de todo debemos mostrar los elementos y carpetas ocultas como se muestra en la **Figura 8**.

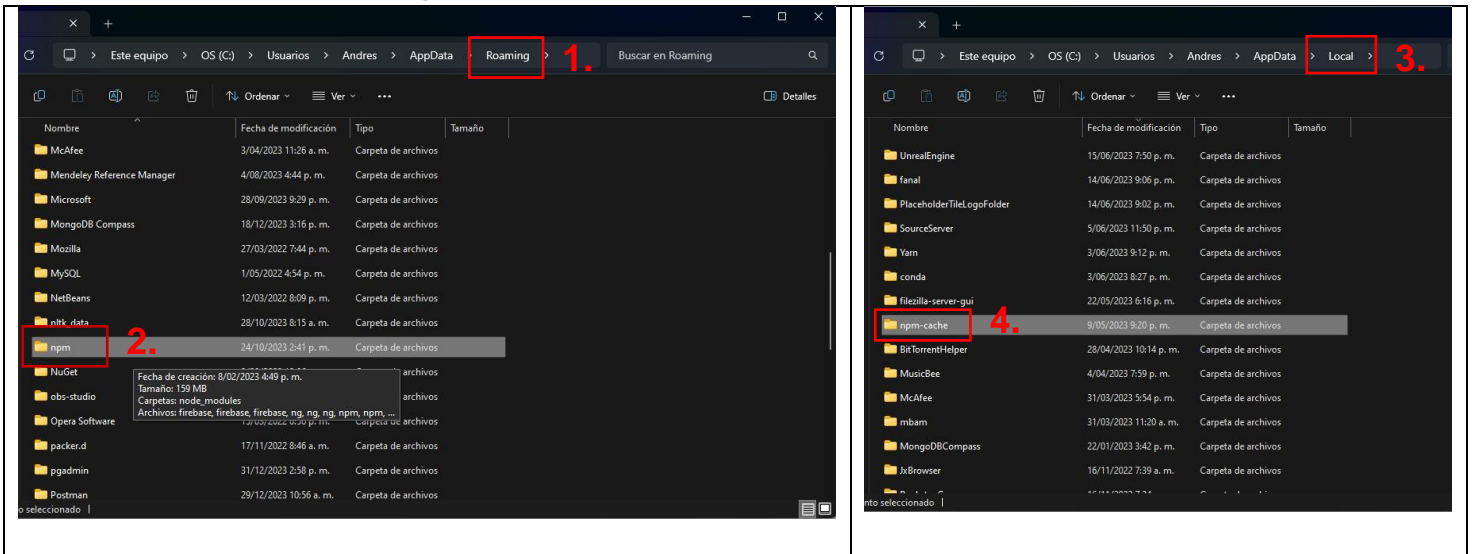
Figura 8 Pasos para mostrar carpetas ocultas



Fuente Autor.

Esta son las carpetas que debemos eliminar de manera manual.

Figura 9 Explorador de archivos para eliminar carpetas NPM



Fuente Autor.

6.3.1.2 Instalación de Node JS

Requisitos para NodeJs: “Node.js no requiere una configuración de hardware sofisticada para funcionar; la mayoría de los ordenadores de esta época deberían manejar Node.js de forma eficiente. Incluso los ordenadores más miniaturizados como el BeagleBone o el Arduino YÚN pueden ejecutar Node.js”[13].

Node es ambiente para ejecutar código en JavaScript. Si una aplicación necesita o utiliza código en JavaScript es susceptible de utilizar Node para ejecutar código en un servidor a través de Ajax. Básicamente funciona con una arquitectura basada en eventos[12].

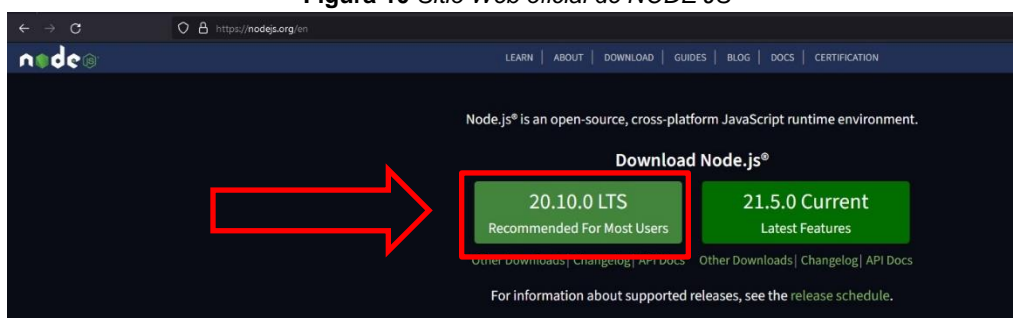
La potencia de NodeJS se basa en la capacidad para gestionar la memoria, un sistema tradicional incrementa el uso de memoria a medida que un usuario se conecta, esto implica que la cantidad de usuarios conectados va a depender de la cantidad de RAM del equipo, por ejemplo, si un servidor tiene 32 gigas en RAM, tiene una alta probabilidad de tener capacidad para administrar máximo 16.000 usuarios, lo cual es un número aceptable. No obstante, ¿Qué pasa si los usuarios superan por mucho ese número? Acá es donde entra Node JS pues no utiliza el concepto de hilos, por el contrario, ejecuta en su motor un nuevo evento cuando se ejecuta un nuevo usuario, lo cual implica una optimización extrema del uso de la memoria, hasta el punto de permitir hasta millones de usuarios. Si es desarrollador de sistemas Web tradicionales, es posible que le interese este artículo:

<https://kinsta.com/es/blog/nginx-vs-apache/> [12]

Como se presenta en la **Figura 10**, se debe descargar y de forma gratuita del sitio oficial:

<https://nodejs.org/es/>.

Figura 10 Sitio Web oficial de NODE JS



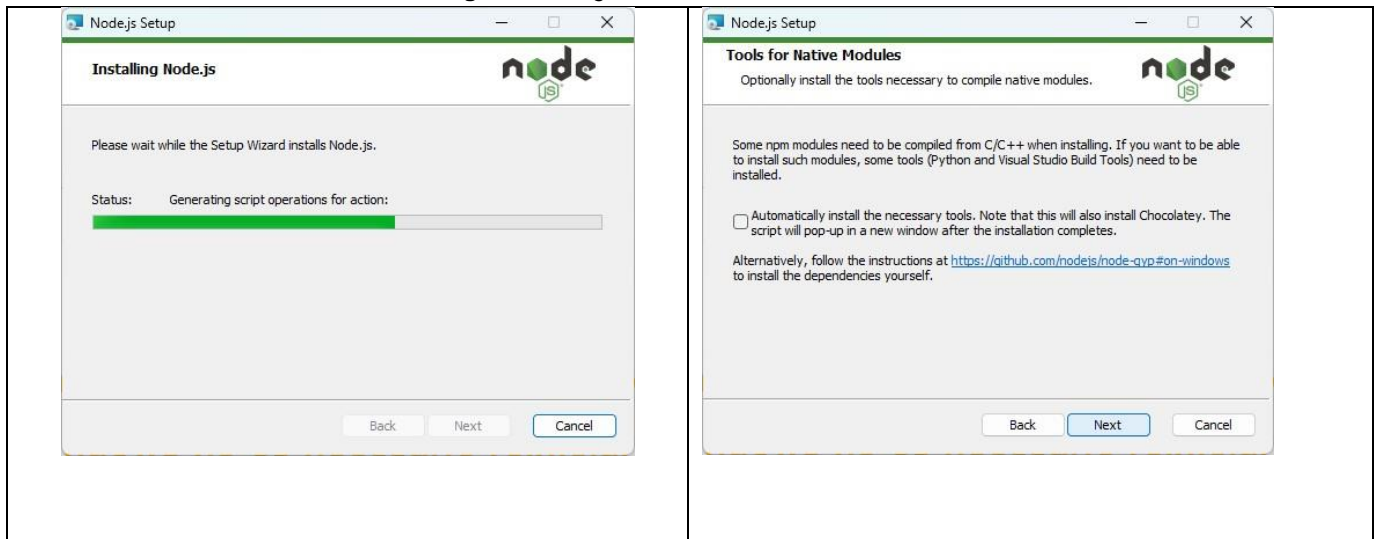
Fuente Autor.

En esta guía se utilizará la versión 20.10.0 LTS para Windows de 64 Bits. Si aún desea utilizar una versión previa de Node JS, asegúrese que sea una versión superior a Node JS 16.0, las versiones anteriores probablemente no funcionarán con este contenido.

¿Por qué usar una versión recomendada en vez de la versión actual? como se puede visualizar en la **Figura 10** la versión 20.10.0 LTS termina con las siglas LTS, esto significa que la versión tiene soporte a largo plazo, garantiza que los errores críticos se solucionaran en un lapso corto, lo que no sucede con la versión actual que no cuenta con el soporte a largo plazo.

Después de descargar Node JS, se debe realizar la instalación. La instalación no requiere conocimientos técnicos, es sencilla y basta con dar clic en siguiente varias veces. Se sugiere no cambiar las opciones por defecto de la instalación. La **Figura 11** presenta el progreso de la instalación de Node JS.

Figura 11 Progreso de instalación de Node JS



Fuente Autor.



**Se recomienda reiniciar el equipo
Ya que Windows después de borrar e
Instalar NODEJS no actualiza los nuevos
cambios.**

Para verificar la instalación de Node JS y su gestor de paquetes npm (Node Package Manager) se debe abrir una consola de comando de Windows (CMD) como se aprecia en la **Figura 12** para ejecutar los siguientes comandos:

```
node -v
```

```
npm -v
```

Figura 12 Verificación de la instalación NodeJS y npm

```
Windows PowerShell
PS C:\> node -v
v20.10.0
PS C:\> npm -v
10.2.3
PS C:\>
```

Fuente Autor.

6.3.1.3 Instalación de Angular

Requisitos para AngularJs: Para AngularJs es esencial contar con los siguientes requisitos de nuestra maquina:

- Sistema operativo: Windows 10, macOS 10.10 (Yosemite) o posterior, o una distribución reciente de Linux (como Ubuntu 18.04 o posterior)
- Memoria: Al menos 4 GB de RAM
- Espacio de almacenamiento: Al menos 10 GB de espacio libre en disco

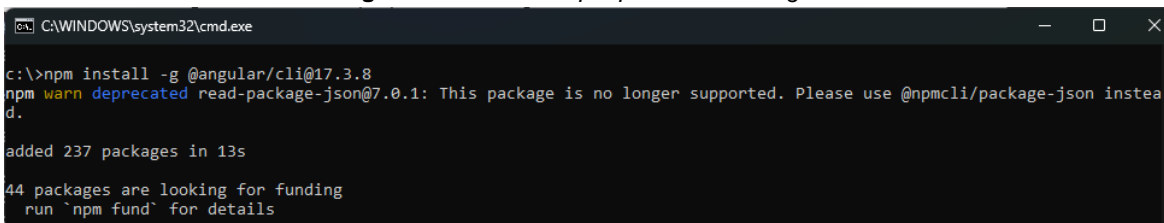
Estos requisitos garantizan que Angular se ejecute sin problemas en nuestro sistema para desarrollar y probar las aplicaciones con eficacia [11].

Para instalar Angular es necesario tener instalado NodeJS y por ende el gestor de paquetes npm. Angular se puede instalar de forma **global**, esto quiere se podrá utilizar desde cualquier carpeta del disco e instalará la última versión disponible. EL comando para instalar Angular es:

```
npm install -g @angular/cli@17.3.8
```

La **Figura 13** presenta el resultado de la instalación global de angular en el equipo.

Figura 13 Utilizando npm para instalar Angular



```
C:\WINDOWS\system32\cmd.exe
c:\>npm install -g @angular/cli@17.3.8
npm warn deprecated read-package-json@7.0.1: This package is no longer supported. Please use @npmcli/package-json instead.
added 237 packages in 13s
44 packages are looking for funding
  run `npm fund` for details
```

Fuente Autor.

Para verificar la versión de Angular instalada se debe ejecutar el siguiente comando en una consola de Windows:

```
ng version
```

La **Figura 14** presenta el resultado del comando ng version.

Figura 14 Versión instalada de angular, nodeJs y npm

```
C:\WINDOWS\system32\cmd.exe
c:\> ng version

Angular CLI
Angular CLI: 17.3.8
Node: 20.14.0
Package Manager: npm 10.8.0
OS: win32 x64

Angular:
...

Package          Version
-----
@angular-devkit/architect    0.1703.8 (cli-only)
@angular-devkit/core        17.3.8 (cli-only)
@angular-devkit/schematics  17.3.8 (cli-only)
@schematics/angular         17.3.8 (cli-only)
```

Fuente Autor.



Si ya tenía instalada una versión de Angular y se desea conservar la versión de Node JS, es posible desinstalar solamente angular de la siguiente manera:

```
npm uninstall -g @angular/cli
npm cache clean
npm cache verify
npm cache clean --force
```

En caso de tener problemas en la instalación de estas dos herramientas en este [Link](#) podrá encontrar una guía de instalación más detallada.

6.3.1.4 Instalación de TypeScript

Este lenguaje para apoyar el desarrollo sobre JavaScript se puede utilizar de dos formas, por un lado, realizando la instalación de manera global, así cualquier proyecto tanto en angular como en NodeJS lo podrá utilizar. Por otro lado, instalándolo como un paquete adicional en un proyecto particular. En esta guía vamos a instalar TypeScript de manera global a través de la ejecución del siguiente comando.

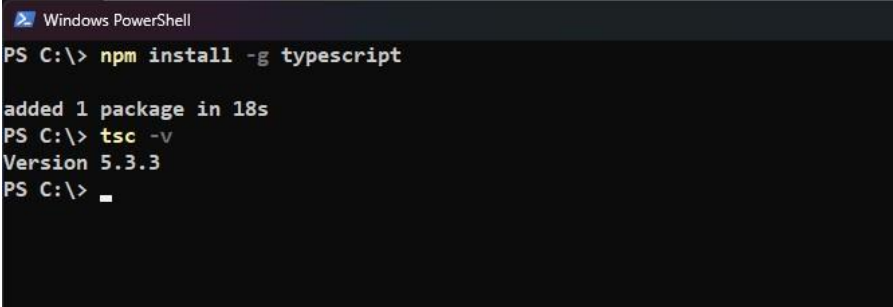
```
npm install -g typescript
```

Para verificar la instalación de TypeScript se utiliza el siguiente comando:

```
tsc -v
```

La **Figura 15** presenta el resultado de la instalación global de TypeScript y la versión Instalada

Figura 15 *Instalación de typescript y su versión*



```
Windows PowerShell
PS C:\> npm install -g typescript

added 1 package in 18s
PS C:\> tsc -v
Version 5.3.3
PS C:\> _
```

Fuente Autor.

6.3.1.5 *Instalación de Visual Studio Code*

Requisitos para Visual Studio Code: “VS Code es compatible con las siguientes plataformas:

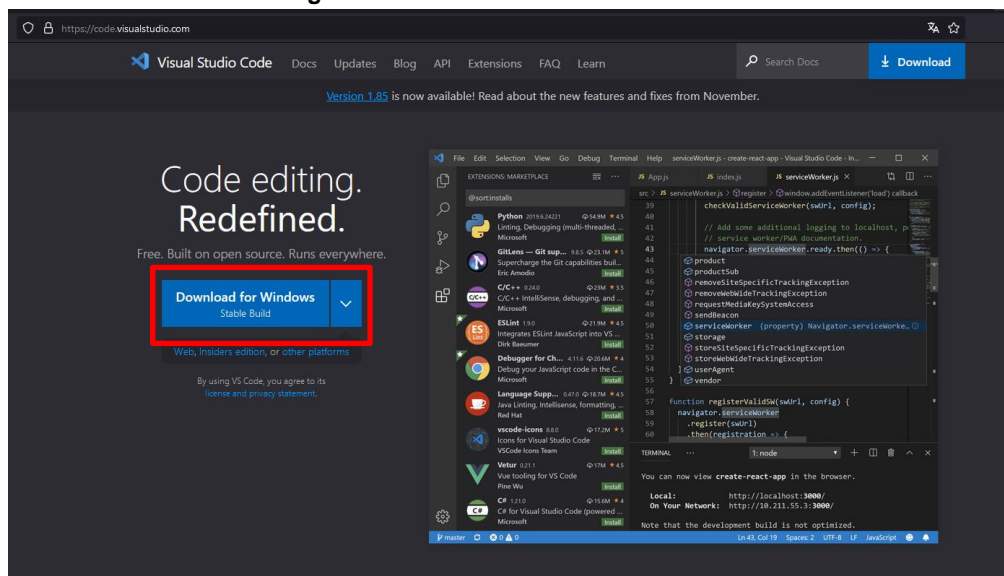
- Windows 10 y 11 (64 bits)
- Versiones de macOS compatibles con actualizaciones de seguridad de Apple. Esta suele ser la última versión y las dos versiones anteriores.
- Linux (Debian): Escritorio Ubuntu 20.04, Debian 10
- Linux (Red Hat): Red Hat Enterprise Linux 8, Fedora 36

Hardware: Visual Studio Code es una descarga pequeña (< 200 MB) y ocupa un espacio en disco menos de 500 MB. VS Code es liviano y debería ejecutarse fácilmente en el hardware actual. Se recomienda:

- Procesador de 1,6 GHz o más rápido
- 1 GB de RAM”[14]

La descarga del editor seleccionado se realiza del sitio web <https://code.visualstudio.com> en la opción “**Download**” en la parte superior derecha. En la **Figura 16** se puede apreciar el sitio web del editor de código seleccionado para el Frontend.

Figura 16 Sitio web de Visual Studio Code



Fuente Autor.

El proceso de instalación es simple, como todo instalador de Windows con varios clics en siguiente, siguiente y finalizar es suficiente. La **Figura 17** presenta la instalación exitosa del editor de código.

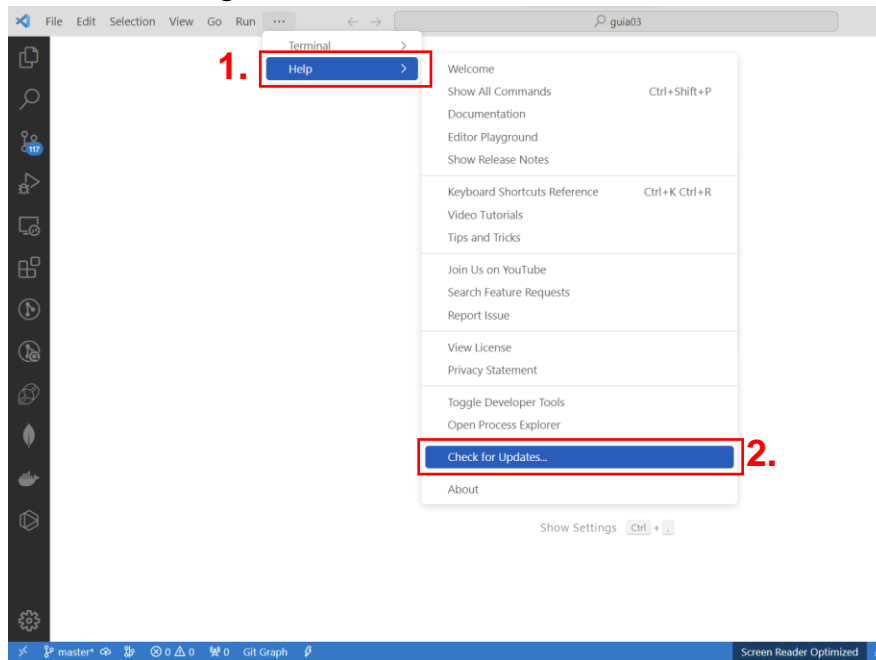
Figura 17 Instalación correcta de Visual Studio Code



Fuente Autor.

Si ya tiene instalado el editor Visual Studio Code, no es necesario desinstalarlo, es suficiente con hacer clic en el menú ayuda (Help) y seleccionar la opción buscar actualizaciones (Check for Updates). La **Figura 18** presenta la forma de realizar la actualización en Visual Studio Code y las extensiones más útiles al momento de desarrollar en Angular.

Figura 18 Actualización de Visual Studio Code

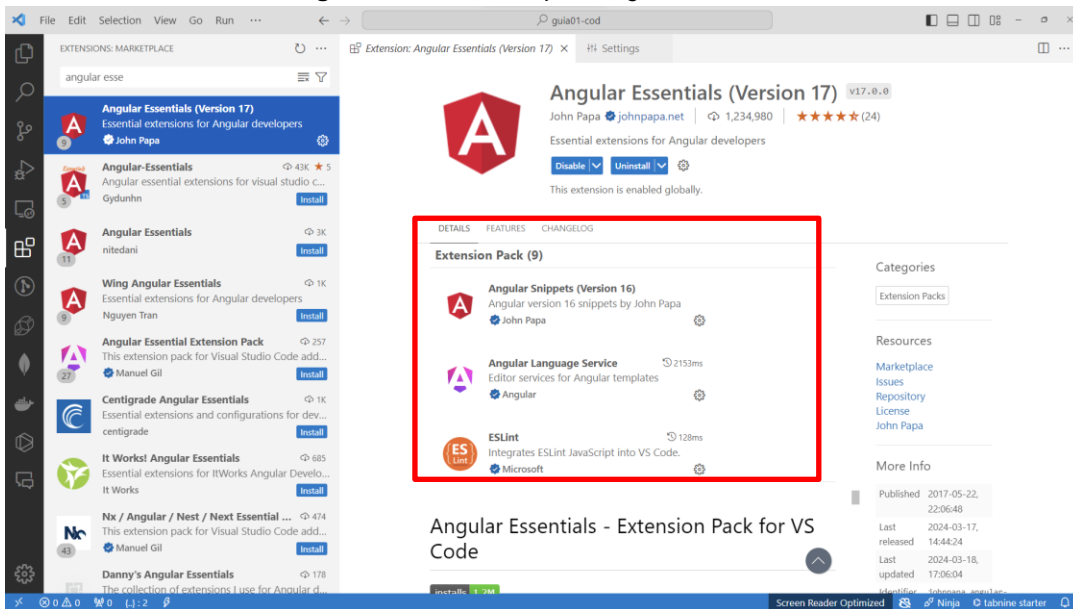


Fuente Autor.

6.3.1.0 Extensiones Angular VCode

Como se puede observar en la **Figura 19** Se recomienda instalar el paquete de extensiones llamado "Angular Essentials (Version 17)". Este conjunto de extensiones ha sido diseñado específicamente para potenciar el entorno de desarrollo AngularJS.

Figura 19 Extensiones para angular en VCode

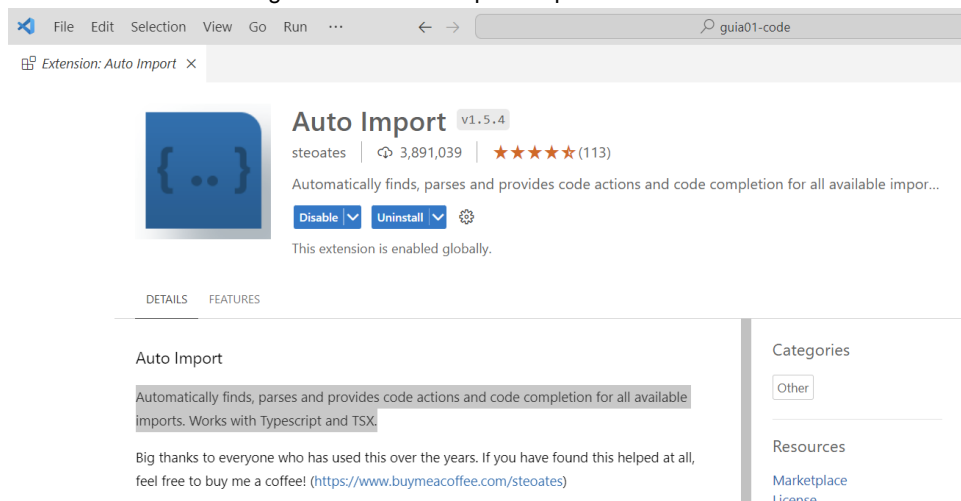


Fuente: Autor

6.3.1.1 Extensión Auto Import

Encuentra, analiza y proporciona automáticamente acciones de código y finalización de código para todas las importaciones disponibles. Funciona con Typescript y TSX.

Figura 20. Extensión para importación de files



Fuente: Autor

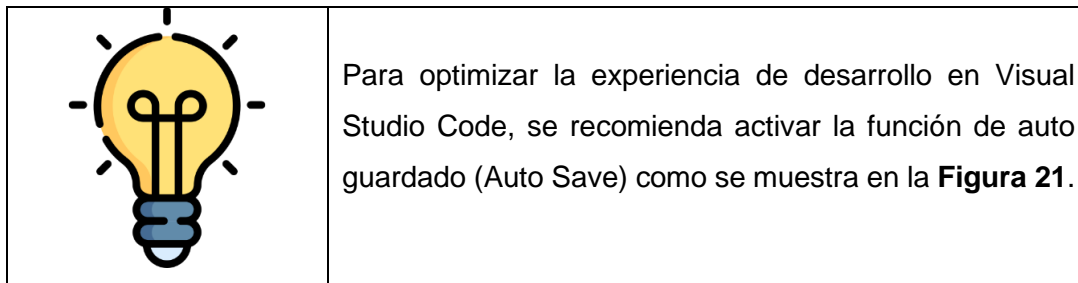
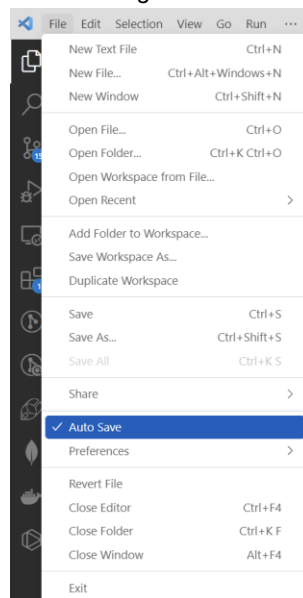


Figura 21 Recomendación del Auto guardado activo para Visual Studio Code



Fuente: Autor

6.3.1.2 Instalación de PostgreSQL

PostgreSQL es el motor de base de datos relacional de código abierto más potente en la actualidad. Se puede descargar del sitio Web <https://www.postgresql.org> seleccionando el sistema operativo sobre el cual se desee trabajar y asegurándose de descargar los instaladores binarios. La **Figura 22** presenta el sitio Web para descargar PostgreSQL.

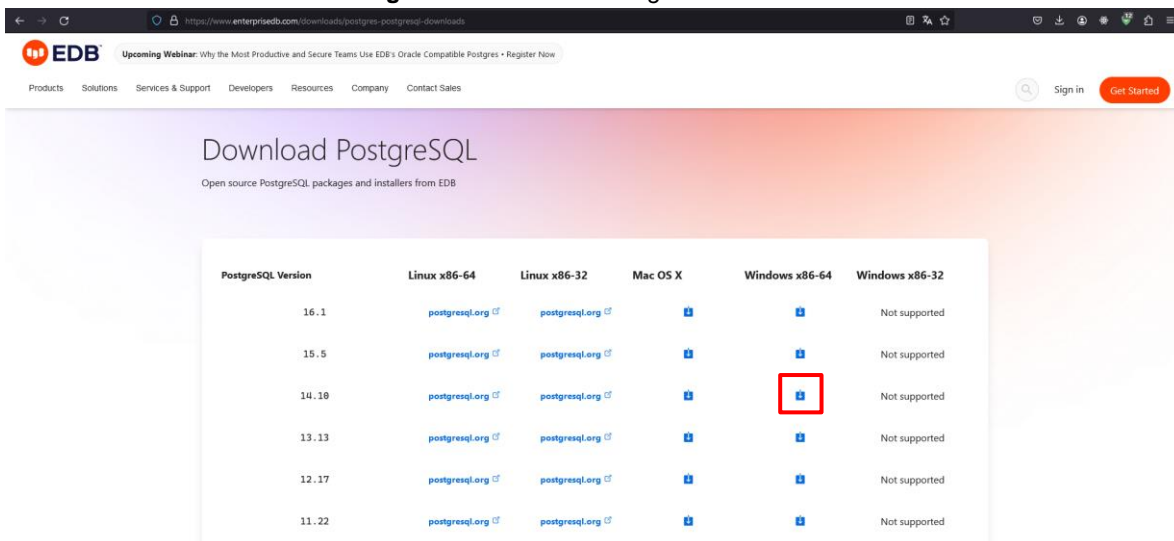
Figura 22 Sistemas operativos disponibles para descargar PostgreSQL



Fuente: Autor

El número de la versión y el sistema operativo depende del desarrollador, en esta guía se utilizará la versión **14.10** para Windows de 64 bits. La **Figura 23** presenta la descarga seleccionada en esta guía.

Figura 23 Versión de PostgreSQL a utilizar

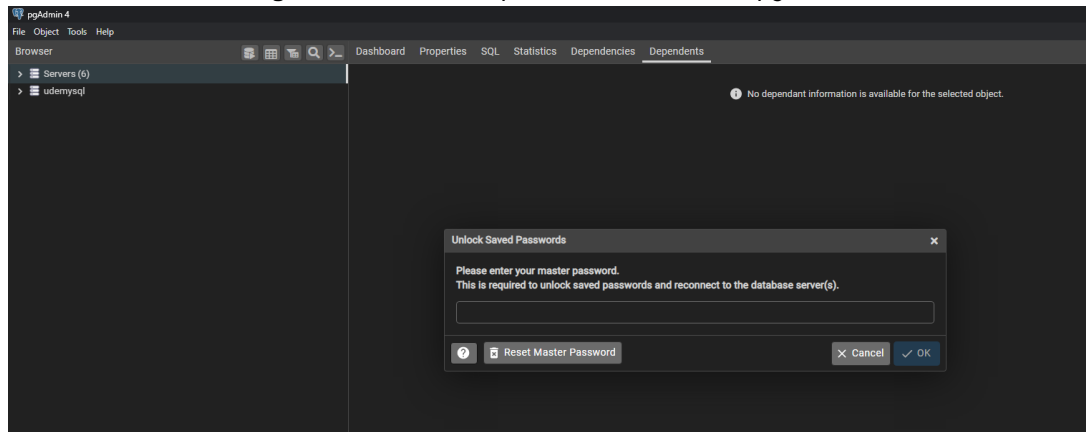


Fuente: Autor

La instalación de PostgreSQL se puede realizar con todos los valores por defecto, no es necesario cambiar nada. Es importante que el desarrollador recuerde la contraseña que asignará a la base de datos.

La herramienta gráfica para la gestión de bases de datos a utilizar es pgAdmin. Se puede encontrar en el menú de Windows una vez finaliza la instalación. Al ejecutar pgAdmin la herramienta solicitará la contraseña asignada en el proceso de instalación. La **Figura 24** presenta la interfaz gráfica que se utilizará para la gestión de las bases de datos.

Figura 24 Herramienta para la administración - pgAdmin

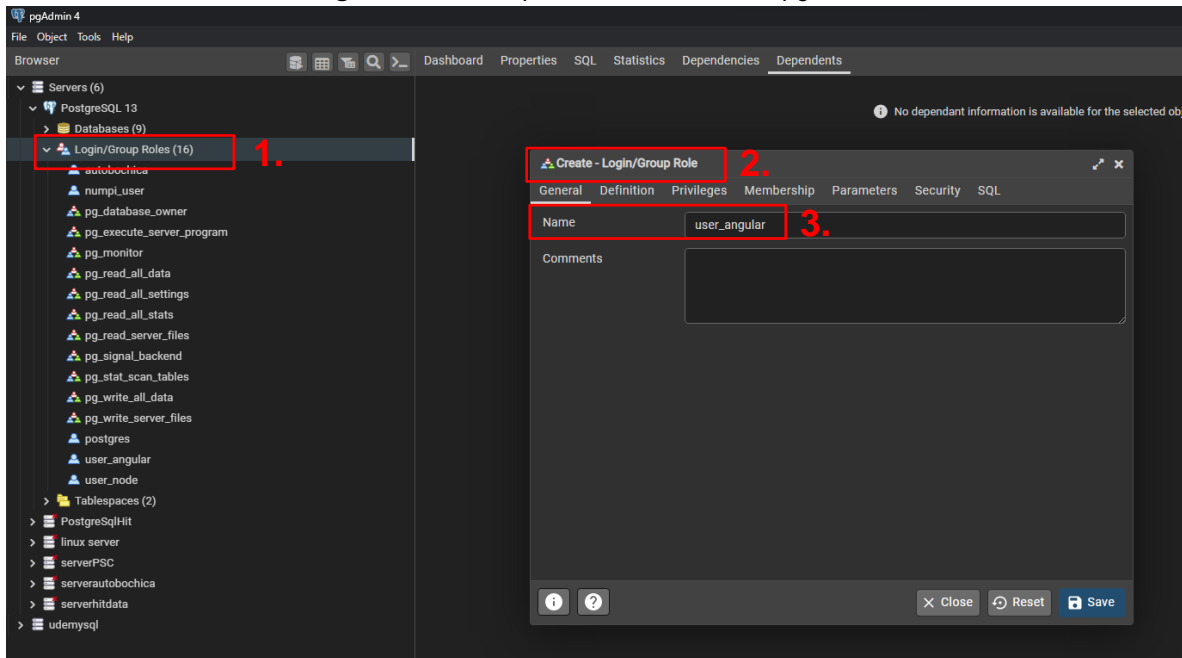


Fuente: Autor

6.3.1.2.1 Creación de un usuario “user_angular” para las guías.

Es de vital importancia la creación de un usuario que permita gestionar las bases de datos que se crearán. La **Figura 25** presenta el árbol de opciones que agrupa y contiene la opción Login/group Roles. En la opción Login/group Roles se debe hacer clic derecho para crear un nuevo usuario. La **Figura 25** presenta la interfaz para la creación del nuevo usuario.

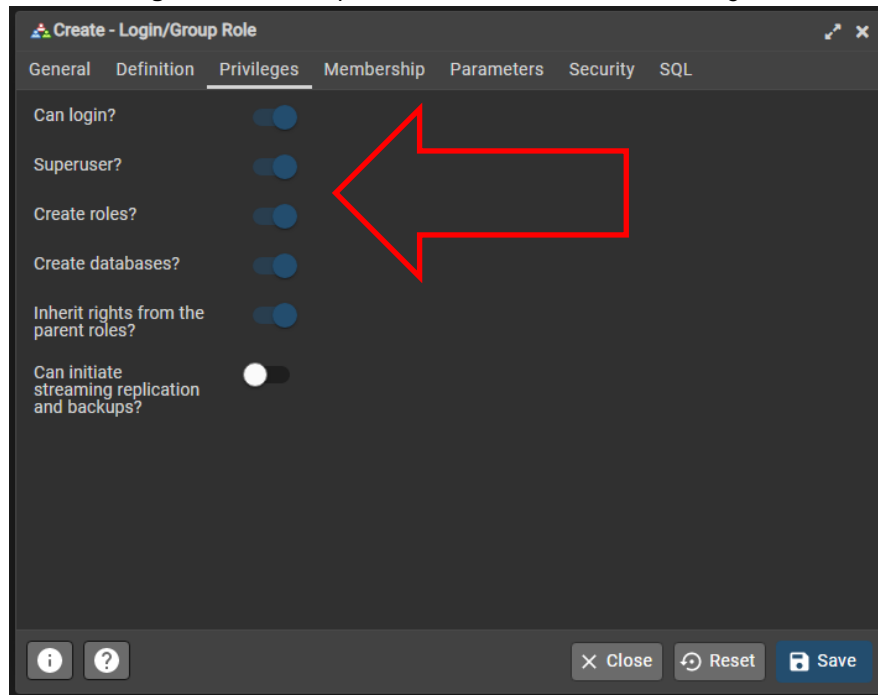
Figura 25 Ventana para crear usuario en el pgAdmin



Fuente: Autor

En la pestaña: de “Definition” (podría asignar la contraseña: 123456) con todos los privilegios diligenciados de manera completa, de lo contrario el usuario no estará en capacidad de permitir el acceso a las bases de datos que se le asignen. La **Figura 26** presenta los privilegios que se deben asignar para completar la creación del usuario.

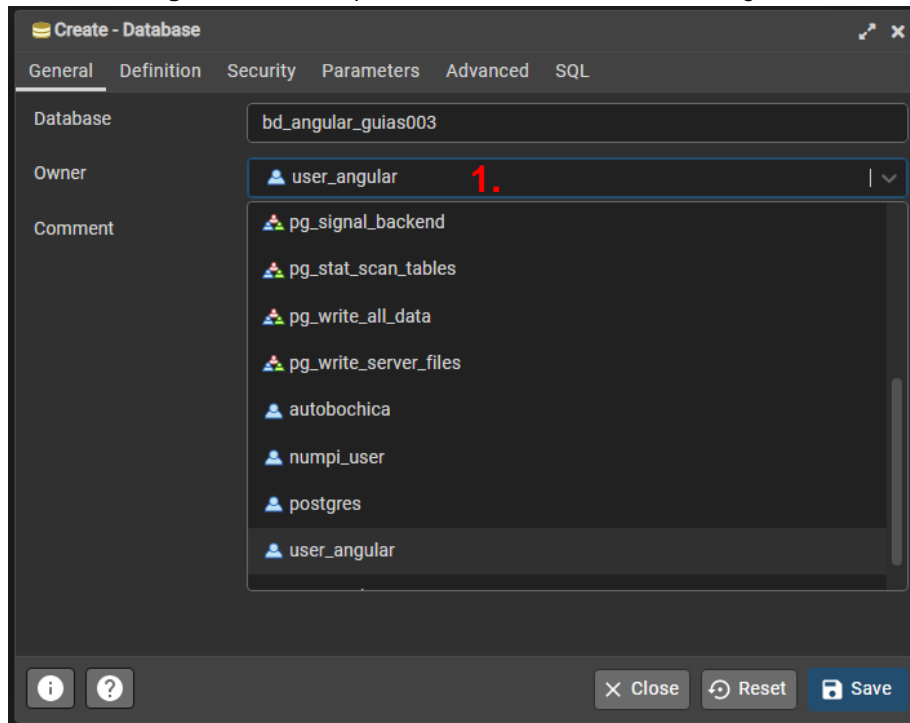
Figura 26 Ventana para la creación de usuarios - Privilegios



Fuente: Autor

Para finalizar, la creación de una base de datos permitirá verificar todo el proceso de instalación de PostgreSQL y de la creación del usuario para permitir la conexión posterior desde Node JS. En el árbol de la izquierda en la opción bases de datos (Databases) se debe hacer clic derecho y seleccionar la opción nueva base de datos. La **Figura 27** presenta la ventana modal para la creación de una base de datos.

Figura 27 Ventana para la creación de usuarios - Privilegios

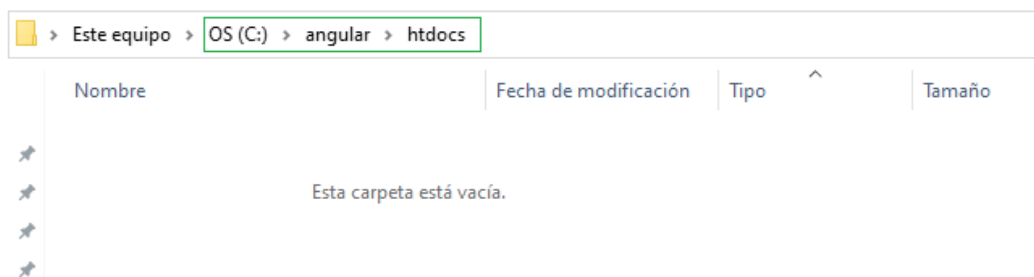


Fuente: Autor

6.3.1.3 Estructura de carpetas sugerida

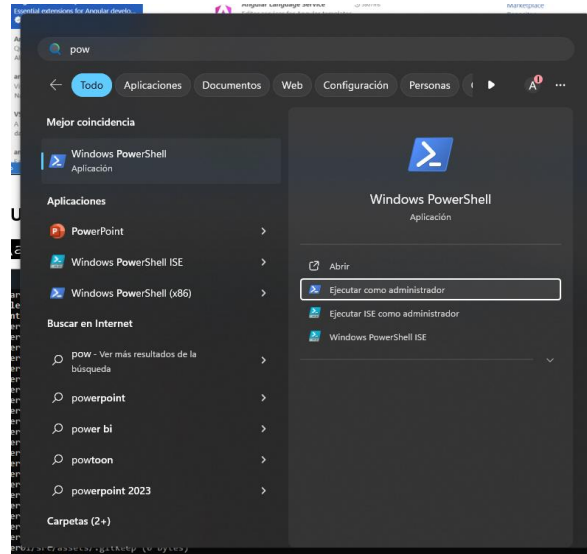
La organización de las carpetas para administrar los proyectos de Angular queda a criterio del desarrollador, sin embargo, a continuación, se propone una estructura de carpetas que permita organizar los proyectos de forma similar a como lo hacen otros lenguajes de programación Web. La **Figura 28** presenta la estructura de carpetas sugerida. La creación de estas carpetas se realiza de manera manual en el explorador de Windows o por la línea de comandos en una terminal.

Figura 28 Estructura de carpetas



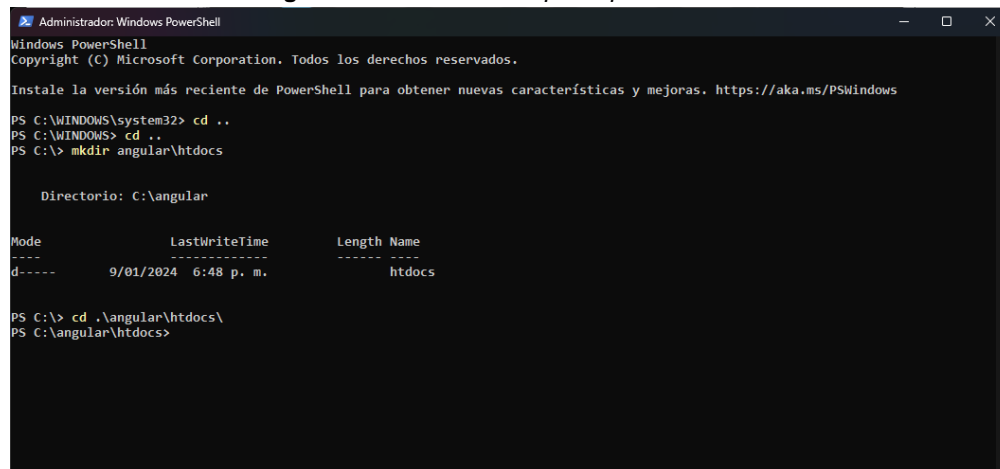
Fuente: Autor

Figura 29 PowerShell en modo administrador



Fuente: Autor

Figura 30 Creación de carpetas por comandos



Fuente: Autor

6.4. PRESENTACIÓN

Angular es un marco de trabajo basado en JavaScript y soportado por Google. Su función es crear aplicaciones del tipo SPA (*Single Page Aplicación*) eficientes y sofisticadas. La estrategia de Angular es administrar el software desarrollado a partir de módulos y componentes, por lo tanto, comprender la estructura de una página en forma de componentes es la clave para entender este marco de trabajo [15].

Las razones por la cual Angular eleva la productividad de los desarrolladores están soportadas en:

1. AngularCLI, es un conjunto de herramientas en modo consola que permite crear cualquier elemento necesario en Angular.
2. El uso de plantillas, en Angular reutilizar plantillas html5 es una tarea que simplifica el desarrollo Web.
3. Escalabilidad, Angular está diseñado para que sus aplicaciones sean escalables de un proyecto pequeño a uno de nivel empresarial.
4. Desarrollar por componentes independientes hacen del proceso una tarea más eficiente.

Para complementar el proceso de desarrollo, el Frontend requiere de su contra parte de un Backend, por lo tanto, es importante introducir el entorno de desarrollo de JavaScript más popular: NodeJS.

NodeJS es un entorno de ejecución de JavaScript con una estructura basada en eventos para aplicaciones escalables. NodeJS se destaca por la comunicación con el protocolo HTTP con una baja latencia entre sus comunicaciones, lo cual lo convierte en un recurso atractivo para librerías o Frameworks web [16].

Como motor de persistencia, en estas guías de conocimiento se utilizará la base de datos

PostgreSQL. Es el más grande sistema de bases de datos relacionales de tipo open source en la actualidad. Desde su lanzamiento, PostgreSQL ha demostrado ser confiable, asegurando la integridad en los datos, la extensibilidad y su adaptabilidad a volúmenes de información altos con tolerancia a fallos.

6.4.1. Características de la máquina a utilizar en desarrollo

Tabla 3.

Características de la máquina usada

Ítem	Descripción
Procesador	11th Gen Intel(R) Core (TM) i5-1135G7 @ 2.40GHz
Memoria física total	12 gigas
Espacio disco disponible	480 gigas SSD
Adaptador de pantalla	Intel(R) Iris(R) Xe Graphics
Sistema operativo	Windows 11 version 22H2

Fuente: Autor

7. REPOSITORIOS GIT CON EL “SCAFFOLDING” DE LAS GUÍAS 01, 02 Y 03

En los siguientes repositorios encontrará la estructura de carpetas, archivos e imágenes que se necesitaran para desarrollar el proyecto.

<https://github.com/andrescardenasalarcon/guia01/blob/main/guia01.zip>

<https://github.com/andrescardenasalarcon/guia02/blob/main/guia02.zip>

<https://github.com/andrescardenasalarcon/guia03/blob/main/guia03.zip>

El siguiente proceso aplica para cualquier repositorio, accediendo al enlace de la guía correspondiente.

Pasos:

1. Descargar el proyecto a trabajar como se muestra en la **Figura 31**.

Figura 31 Repositorio git de las guías



Fuente: Autor

2. Descomprimir el archivo zip de la guía, accedemos a la carpeta que se abrirá en el editor de código VCode.

3. En la terminal de nuestro proyecto en VCode se ejecuta el comando “*npm i*”, esto para descargar las librerías *node_modules* como se muestra en la **Figura 32**.

Figura 32 Instalación de *node_modules*



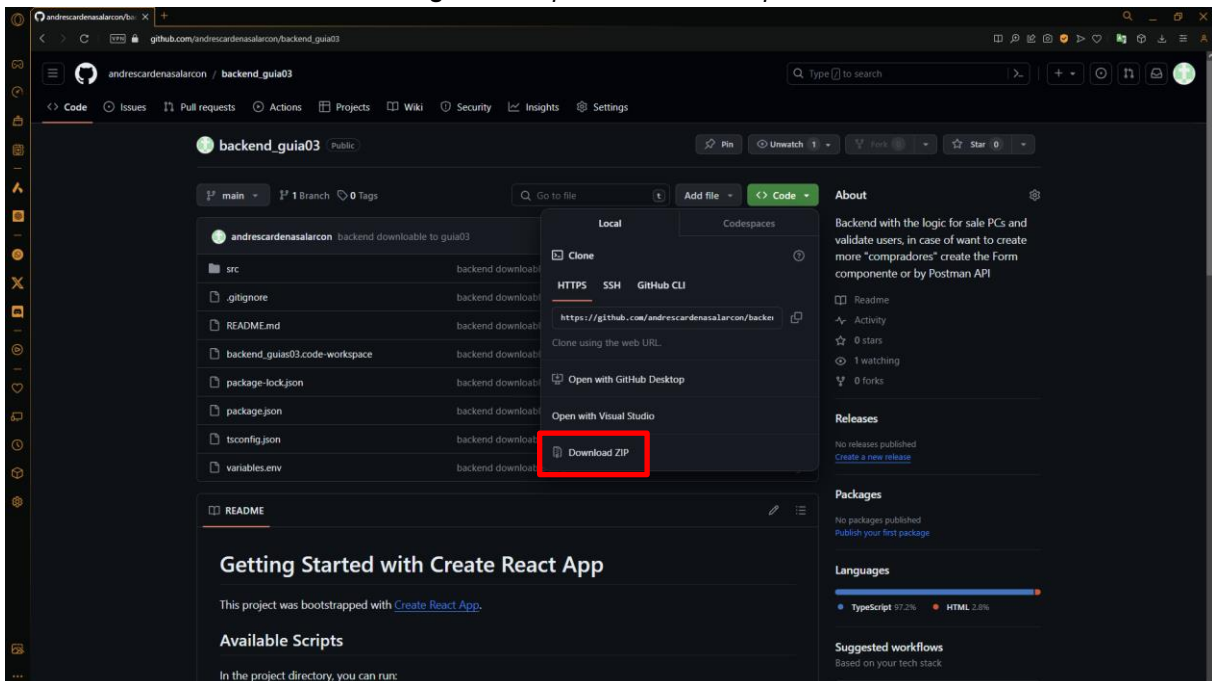
Fuente: Autor

4. Empezar a aprender de AngularJs con la [Guia01] [Guia02], [Guia03] 🙌 .
5. NOTA: Aquí encontrará el código del proceso de las guías: <https://github.com/andrescardenasalarcon/GuiasAprendizajeResultas>.

7.1. Repositorio con el Backend para la guía03

El siguiente enlace es el repositorio donde está alojado el backend para la guia03 https://github.com/andrescardenasalarcon/backend_guia03.git

Figura 33 Repositorio Backend api

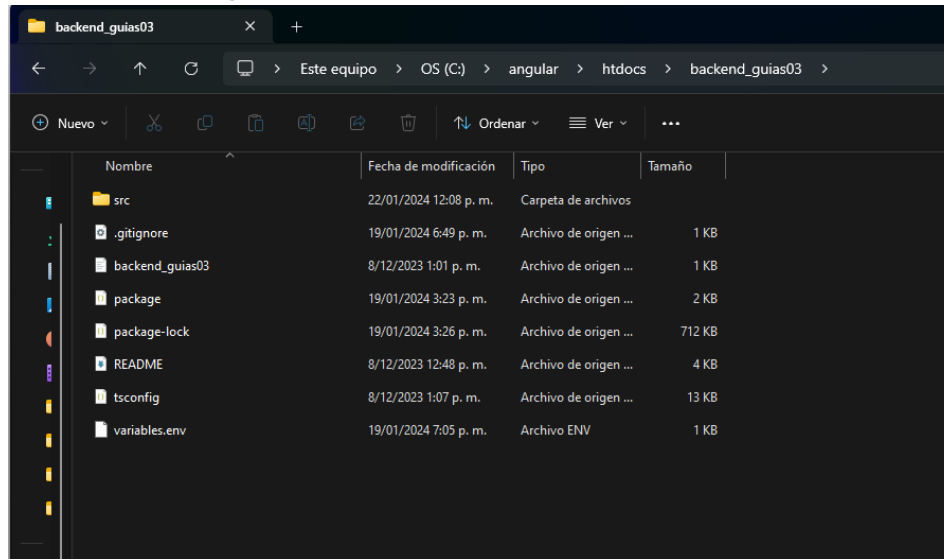


Fuente: Autor

7.1.1. Puesta en marcha del proyecto Backend en local

1. Se descomprime el .zip del backend, para después acceder dentro de la carpeta como se muestra en la **Figura 34**.

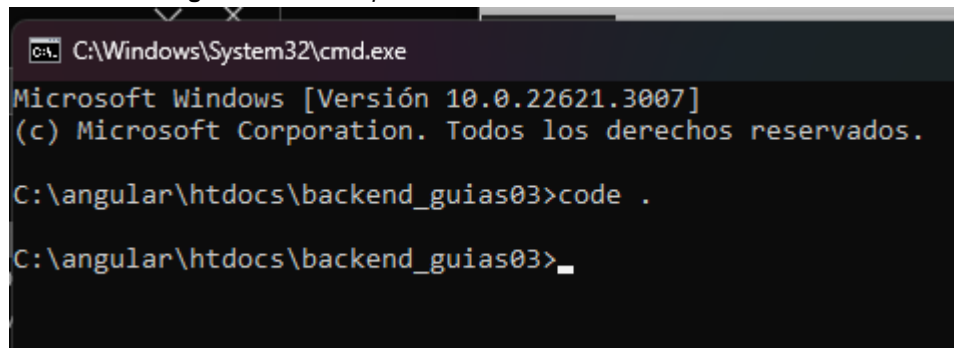
Figura 34 Backend carpeta descomprimida



Fuente: Autor

2. Accede a la ruta donde está la carpeta del backend desde una terminal cmd, para posteriormente abrir el proyecto con el comando "`code .`" como se muestra en la **Figura 35**.

Figura 35 "Code ." para abrir el backend en VCode.



Fuente: Autor

3. Abierto el proyecto en VisualCode, en una terminal se descarga de internet todas las librerías necesarias “node_modules” con el siguiente comando “`npm i`” como se muestra en la **Figura 36**.

Figura 36 Instalación de librerías “npm i”

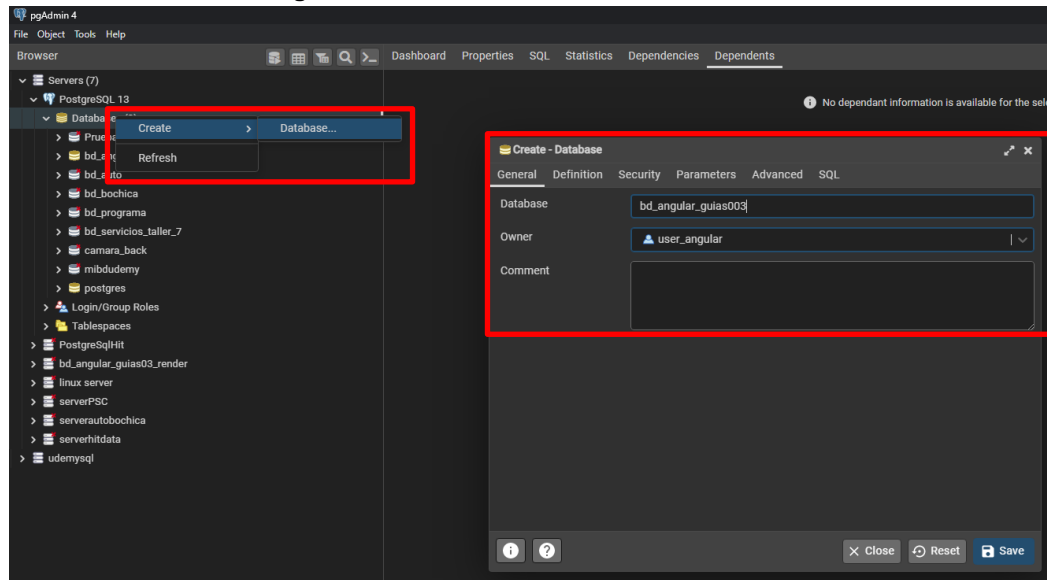


```
PROBLEMS OUTPUT TERMINAL PORTS COMMENTS DEBUG CONSOLE
PS C:\angular\htdocs\backend_guias03> npm i
[#####.....] / reify:fsvents: sill reify mark deleted [ 'C:\\
```

Fuente: Autor

4. Mientras se descargan las librerías en la carpeta “`src\scriptbd\scriptBd.sql`” esta todo el script de la base de datos PostgreSQL.
5. Como se mostraba en la “**Creación de un usuario “user angular” para las guías.**”, este es el usuario con el que crearemos nuestra base de datos.
6. Creamos una nueva base de datos con el nombre de “`bd_angular_guias003`” y usando el usuario creado en el paso anterior “`user_angular`” lo asignaremos a esta BD como se muestra en la **Figura 37**.

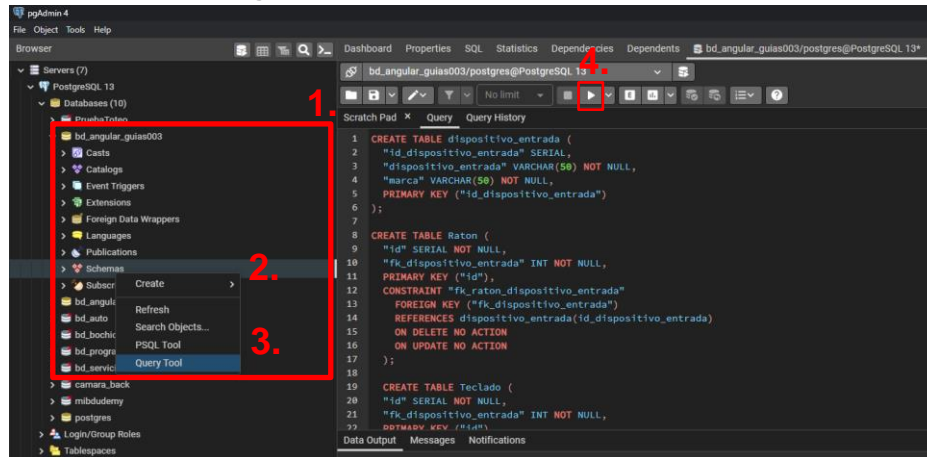
Figura 37 Creación de la base de datos



Fuente: Autor

7. En la query tools de pgAdmin copiamos todo el archivo del paso 4 y se ejecuta como se muestra en la **Figura 38**.

Figura 38 Script de la base de datos



Fuente: Autor

8. Creadas y añadidas todas las tablas, en nuestra terminal de VCode se ejecuta el siguiente comando `"npm run build"`
9. Generamos otra terminal en el + y allí ejecutaremos el comando `"npm run dev"` como se muestra en la **Figura 39**.

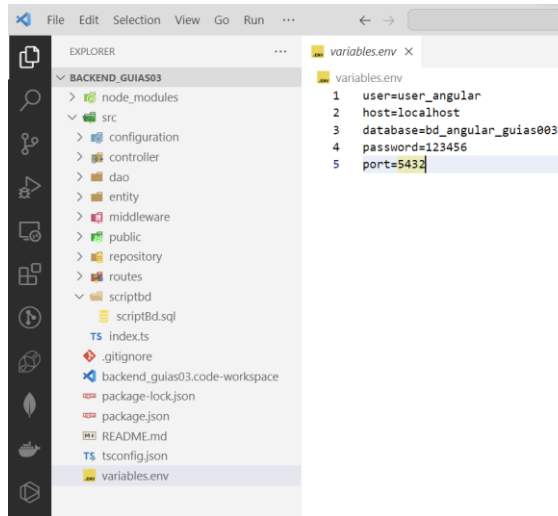
Figura 39 Comando para iniciar el backend



Fuente: Autor

10. En caso de haber usado otros nombres o credenciales para la base de datos se deben modificar en el archivo `"variables.env"` como se muestra en la **Figura 40**.

Figura 40 Cadenciales de la BD en el archivo “variables.env”



```
1 user=user_angular
2 host=localhost
3 database=bd_angular_guias003
4 password=123456
5 port=5432
```

Fuente: Autor

7.2. Guia01

Se dispone del material con sus correspondientes pasos a seguir en el siguiente anexo para la guía 01:



ANEXO_GUIA01.pdf

7.3. Guia02

Se dispone del material con sus correspondientes pasos a seguir en el siguiente anexo para la guía 02:



ANEXO_GUIA02.pdf

7.4. Guia03

Se dispone del material con sus correspondientes pasos a seguir en el siguiente anexo para la guía 03:



ANEXO_GUIA03.pdf

7.5. Encuesta de percepción

Para dar cumplimiento la desarrollo y objetivo de este material PDF que va dirigido en su parte a la docencia de la Facultad de ingeniería de sistemas, se realiza una encuesta [Enlace](#) a una muestra de docentes de la facultad afines en el área de desarrollo tecnológico para el semestre, esto con el fin de obtener sus opiniones y oportunidades de mejora que pudiese tener el material, antes de la entrega final.

Figura 41 ¿Pregunta, Es útil la problemática ejemplo y los talleres(títulos) proporcionados para comprender los



Fuente Google Forms.

El gráfico circular de la Figura 41 muestra que el 100% de los encuestados encontró útil la problemática ejemplo y los talleres (títulos) proporcionados para comprender los conceptos de AngularJS. Esto significa que todos los participantes en la encuesta consideraron que estos recursos son valiosos para el aprendizaje.

Figura 42 ¿Pregunta, Cubre el contenido los fundamentos de AngularJs?



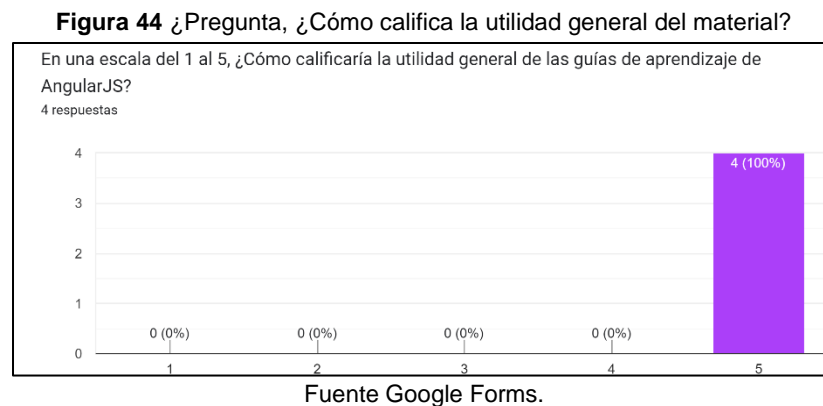
Fuente Google Forms.

El gráfico circular de la **Figura 42** muestra que el 100% de los encuestados considera que el contenido presentado en las guías de aprendizaje cubre de manera adecuada los

fundamentos de AngularJS. Esto significa que todos los participantes en la encuesta consideraron que estos recursos fueron completos y se proporciona suficiente información para comprender los conceptos básicos de AngularJS.



El gráfico circular de la **Figura 43** muestra que el 100% de los encuestados está de acuerdo con la organización y estructura del contenido en las guías de aprendizaje. Esto significa que todos los participantes en la encuesta consideraron que estos recursos están bien organizados, son fáciles de seguir y navegar.



El gráfico de barras de la **Figura 44** muestra la distribución de las respuestas a la pregunta de la encuesta sobre la utilidad general de las guías de aprendizaje de AngularJS. La mayoría de los encuestados (100%) calificaron las guías con una puntuación de 5, lo que indica que las consideran muy útiles. No hubo respuestas para las calificaciones 1, 2, 3 o 4.

Figura 45 ¿Pregunta, Estaría dispuesto a usar este material en clases?



Fuente Google Forms.

Como pregunta de índice de aprobación para este material de aprendizaje, la **Figura 45** muestra en el gráfico circular que el 100% de los encuestados encuentran muy útil y en caso de ser necesario, dispuestos a usar estos conceptos como material de docencia para una posible electiva de Angular Frontend.

Figura 46 ¿Pregunta, algún tema en específico?



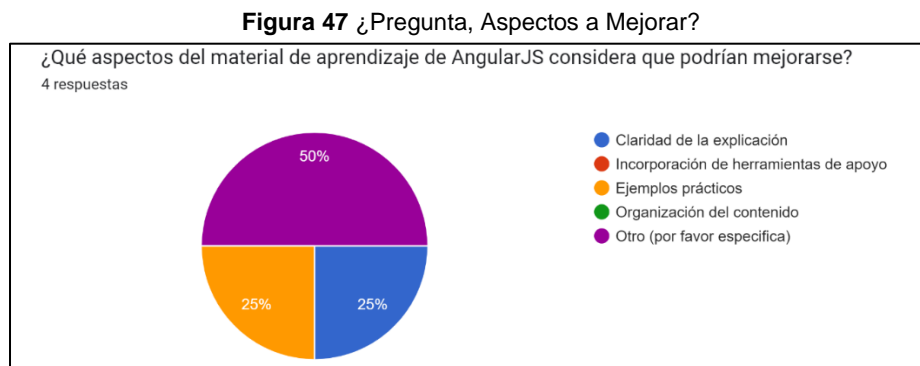
Fuente Google Forms.

Se destaca en rasgos generales dos respuestas para la pregunta de la **Figura 46**:

- *Una introducción específica con respecto al nivel de conocimiento previo que debe contar el usuario.*
- *Un apartado con las especificaciones técnicas del equipo al momento de instalar las herramientas que se usaran a lo largo de la enseñanza.*

Para dar cumplimiento a las respuestas de esta pregunta se agrega un apartado en el material de estudio una pequeña lista que enumera los requisitos necesarios que debe contar el lector para poder replicar este material [**6.1 Requisitos previos del lector**].

En el segundo apartado, las especificaciones técnicas del equipo, en todos los ítems de instalación de algo se agrega un título en donde se presentan los requerimientos y la arquitectura que debe contar nuestro equipo antes de instalarlo [**Instalación de Node JS**].



Fuente Google Forms.

Como se muestra en la **Figura 47**, el apartado de Otro (por favor especifica), estas fueron las respuestas de los encuestados:

- *Poner numeración en las capturas de pantalla que contenga más de 3 pasos para facilitar la lectura.*
- *Casos de estudio ligados a buenas prácticas y ejemplos.*
- *Incluir ejercicios propuestos no desarrollados (para que el lector pueda practicar más)*

Se toma en cuenta la apreciación el primer ítem, y se realiza la respectiva numeración a las figuras que contuvieran más de 3 pasos, se presenta un ejemplo de un como muestran las imágenes bajo una numeración de pasos a la hora de eliminar Npm "**Figura 9**".

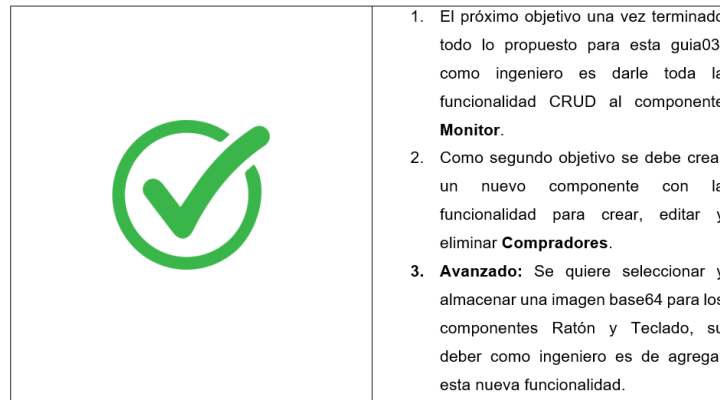
En el apartado de casos de estudio, con respecto a buenas prácticas y ejemplos se deja propuesto una serie de contenidos o enlaces que se encontrarán a medida que se avance

en la creación de componentes dentro de la guía además de dar soporte con relación a la temática abordada.

Se incluye un apartado al final de cada documento una serie de ejercicios como se muestra en la **Figura 48** esto con el fin de que el lector decida el camino y la forma en cómo piensa abordar la solución, con base a todos los conceptos que se presentaron durante cada componente.

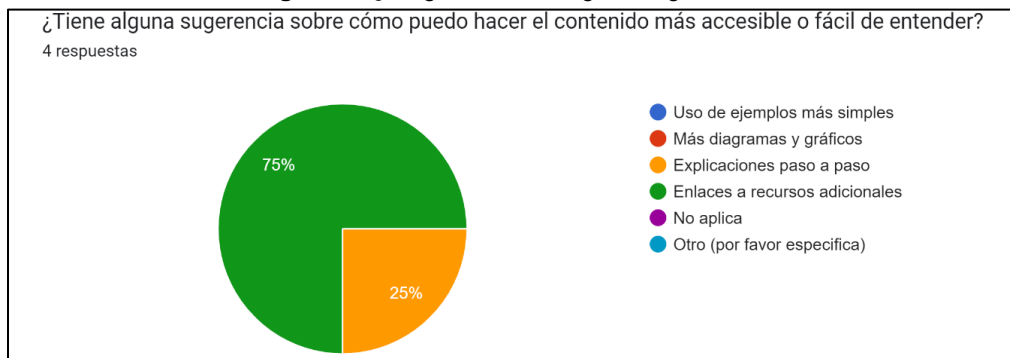
Figura 48 Retos para el lector

6. Reto para el lector



Fuente Autor.

Figura 49 ¿Pregunta, Tiene alguna sugerencia?



Fuente Google Forms.

Seguidamente en relación con los casos de usos, previamente estipulado en la anterior respuesta, y del diagrama circular de la **Figura 49**, se orienta esta recomendación al porcentaje que está en verde (75%), a medida que se presenta un nuevo tema, se agrega

un apartado con título de **!!!Bonus!!!** esto para motivar al lector a leer y entender los conceptos presentados mediante unos artículos o blogs que presentan la información con más variedad de ejemplos y las ventajas de usar o implementar cierta tecnología. La Tabla 4 presenta un ejemplo de cómo aparece los enlaces a recursos adicionales dentro del paso a paso del material.

Tabla 4 Ejemplo **!!!Bonus!!!** a recursos adicionales

!!!Bonus!!! En el siguiente blog enseñan acerca de qué es una interfaz y cómo usarlas mediante ejemplos "[*Interfaces en Typescript*](#)".

Fuente Autor.

Figura 50 ¿Pregunta, Algún Comentario adicional?



Fuente Google Forms.

En el gráfico de circular muestra de la **Figura 50** las respuestas a la pregunta abierta de la encuesta sobre comentarios adicionales o recomendaciones. La categoría más común es "Sí, estoy satisfecho/a con el material proporcionado" (50%), lo que indica que la mayoría de los encuestados están contentos con los recursos proporcionados. El otro (50%), indica que están satisfechos con la organización, conceptos, temáticas y el material en cuestión.

Con todas las preguntas anteriormente expuestas, seguidamente se debe tomar o seleccionar una muestra que más se ajusta a nuestros objetivos, análisis y público. "Una muestra es un subconjunto seleccionado de individuos de una población estadística. Se utiliza para representar o proporcionar información sobre la población total. Las muestras son fundamentales en la investigación y el análisis estadístico."

De los distintos tipos de muestras se selecciona la muestra probabilística; detalla que una muestra es probabilística cuando cada elemento de la población tiene la misma probabilidad y valor de ser seleccionado para la muestra.

Plan de análisis

“La investigación cualitativa es aquella donde se estudia la calidad de las actividades, relaciones, asuntos, medios, materiales o instrumentos (material de aprendizaje a la docencia) en una determinada situación o problema. Esta procura por lograr una descripción holística, esto es, que intenta analizar exhaustivamente, con sumo detalle, un asunto o actividad en particular”[17].

El plan de análisis que se tiene para trabajar con los distintos tipos de datos recolectado en las encuestas se basa en las medidas de tendencia central; son medidas que tratan de representar la tendencia de los valores de las variables. Entre estas se hace alusión a la medida ponderada. Este tipo de medida se selecciona por qué se debe a un promedio de todos los valores, pero cada una tiene diferente importancia o peso.

$$\bar{X}_w = \frac{\sum x_i P_i}{\sum P_i}$$

Pregunta	Nota	Importancia/Peso
¿ Encuentra útil la problemática ejemplo y los talleres del material?	5.0	2
¿Considera que el contenido cubre de manera adecuada los fundamentos de AngularJS?	5.0	2
¿Está de acuerdo con la organización y estructura del contenido?	5.0	2
¿Cómo calificaría la utilidad general de las guías de aprendizaje?	5.0	5
¿Estaría dispuesto/a a utilizar este material en sus clases para enseñar AngularJs?	5.0	3

¿Hay algún tema específico que debería aparecer en las guías de aprendizaje?	2.5	1
¿Qué aspectos del material de aprendizaje de AngularJS considera que podrían mejorarse?	3.1	2
¿Tiene alguna sugerencia sobre cómo puedo hacer el contenido más accesible?	3.0	4
¿Tiene algún comentario adicional o alguna otra recomendación?	5.0	5

Fuente Autor.

$$X_w = \frac{50*2+5.0*2+5.0*2+5.0*5+50*3+2.0*1+3.1*2+3.0*4+5.0*5}{2+2+2+5+3+1+2+4+5} = 4.5$$

El resultado de esta encuesta se toma y se maneja como muestra probabilística, ya que independientemente del grupo de docentes encuestados, el resultado final con las métricas y objetivos para la docencia serán iguales, además de alcanzables y mejorables en pro de mejorar e incrementar tanto las temáticas como la calidad de este material. Finalmente, el resultado obtenido y la forma en cómo se determina el impacto de esta encuesta, se interesa más es en saber cómo se da la dinámica o cómo ocurre el proceso en que se dan las problemáticas y los aportes por parte de los encuestados en pro de ayudar a un material garante, vigente y completo.

8. HERRAMIENTA SOFTWARE

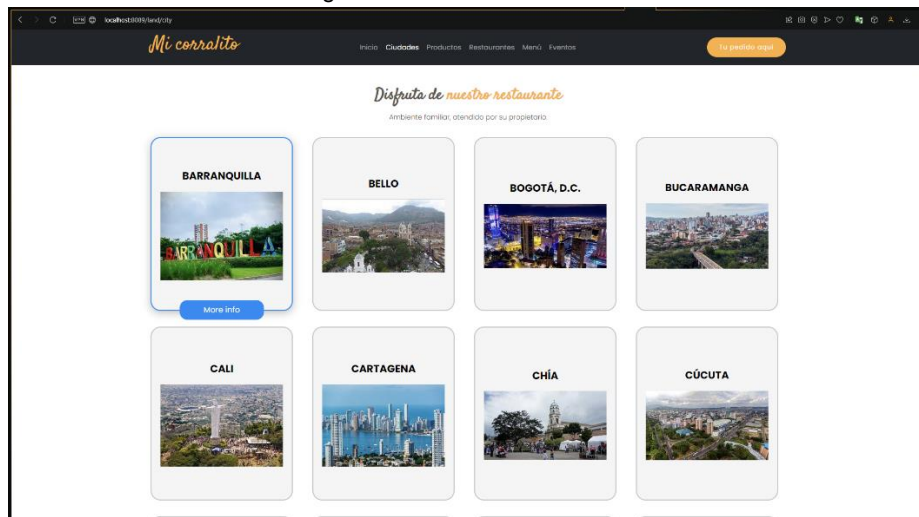
A continuación, se muestran capturas de pantalla de la aplicación final desarrollada a partir del material de las guías de conocimiento. Estas capturas no pertenecen a los ejercicios de cada guía individual, sino que son el resultado de utilizar las guías de conocimiento en un proyecto completo y se puede encontrar almacenado en el siguiente repositorio [Frontend](#) y [Backend](#).

Figura 51 Pantalla principal.



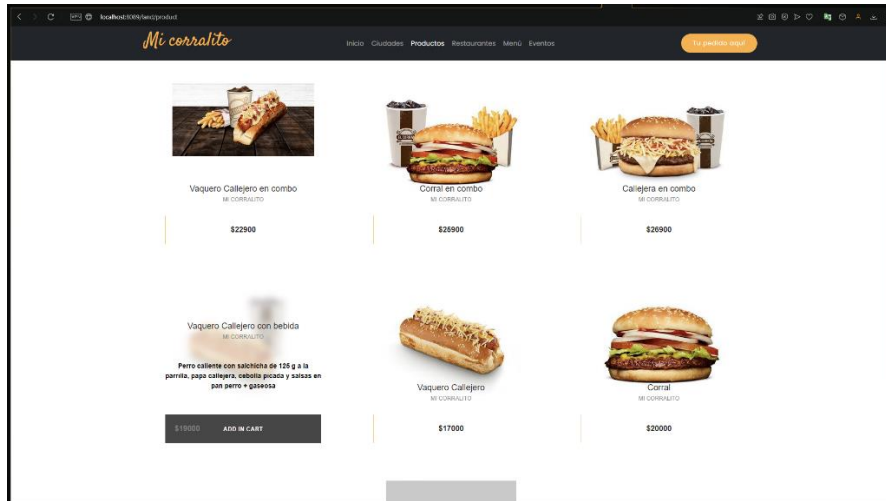
Fuente Autor.

Figura 52 Sección de ciudades.



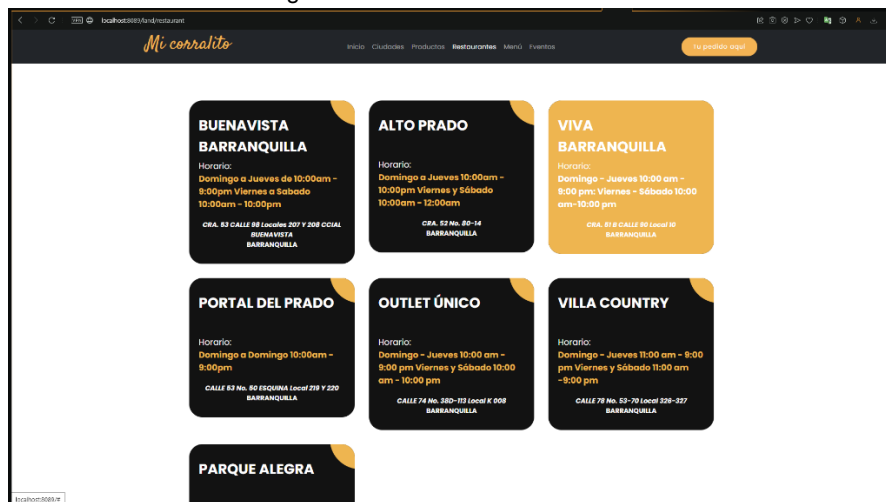
Fuente Autor.

Figura 53 Sección de productos.



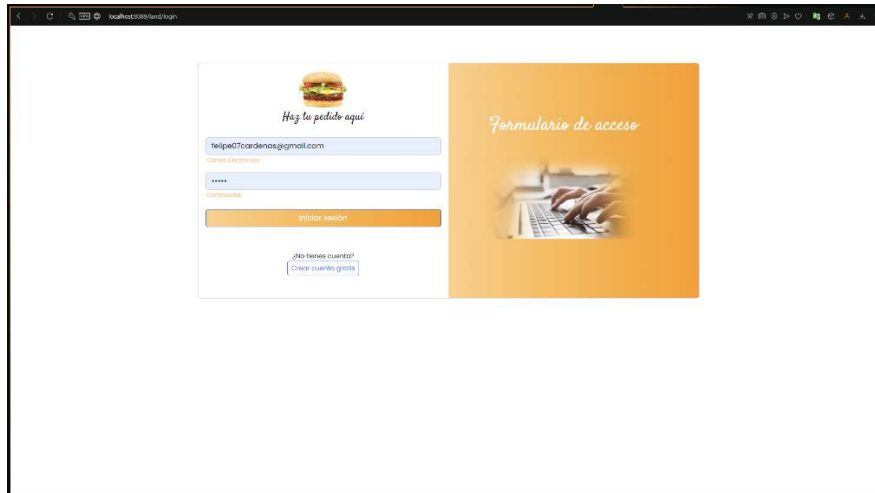
Fuente Autor.

Figura 54 Sección de restaurantes.



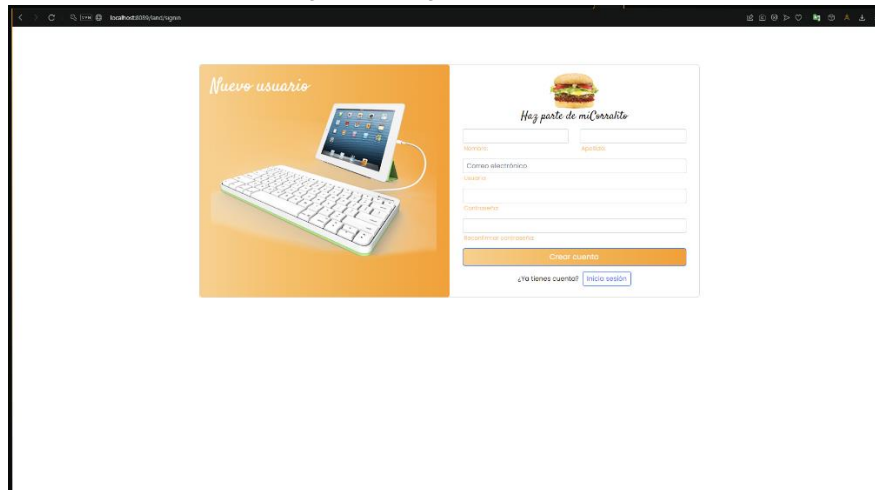
Fuente Autor.

Figura 55 Inicio de sesión usuario.



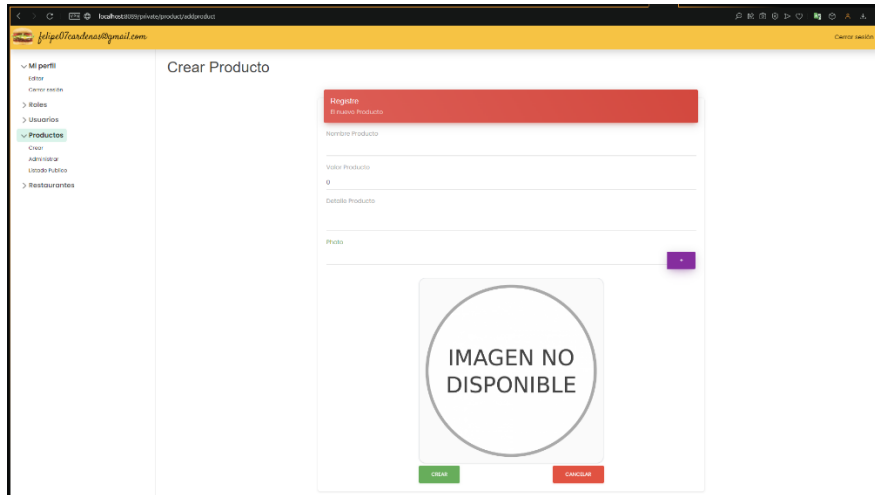
Fuente Autor.

Figura 56 Registro de usuario.



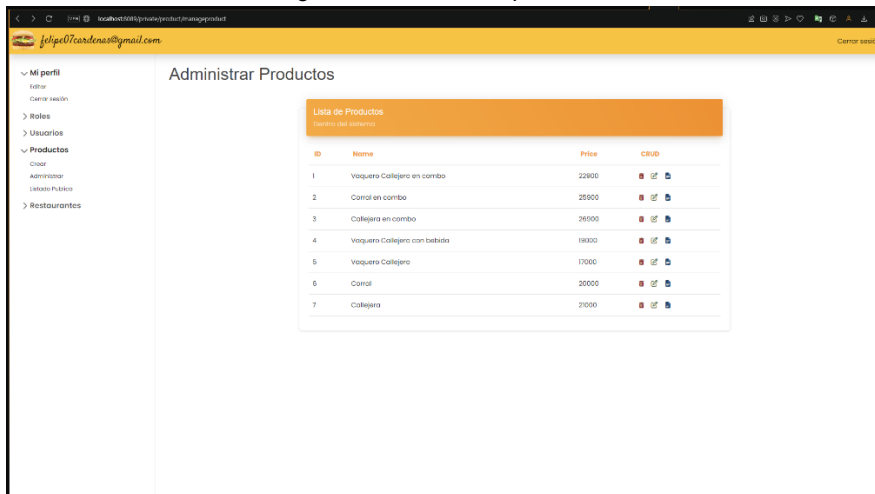
Fuente Autor.

Figura 57 Creación de productos.



Fuente Autor.

Figura 58 Administrar productos



Fuente Autor.

9. CONCLUSIONES

1. Evaluando de manera detallada las guías de desarrollo del proyecto MISION TIC 2022 y aplicando protocolos específicos para validar su calidad y actualidad. Los errores de librerías fueron documentados, y los resultados se presentan de manera exhaustiva en el archivo "ANEXO 1: ETAPA 1 VALIDACIÓN DE LAS GUÍAS PROYECTO MINTIC - 2022", que no solo registra las observaciones, sino que también destaca oportunidades de mejora para optimizar el material según las versiones más recientes de esta tecnología.
2. Mediante el proceso de la documentación se establecieron los nuevos módulos y protocolos conforme a las fuentes bibliográficas de Google para el desarrollo de aplicaciones Frontend. Este proceso resultó en la creación de un documento estructurado en el archivo "ANEXO 2: ETAPA 2 Y 3 DOCUMENTACIÓN Y ESTRUCTURA DE LOS PROTOCOLOS PARA LAS NUEVAS GUÍAS" que detalla los contenidos de los componentes, incluyendo temáticas, talleres a abordar atreves de una problemática ejemplo además de enriquecer el contenido del material ya existente.
3. En la implementación de los módulos y componentes necesarios para desarrollar cada una de las temáticas propuestas dentro del material de aprendizaje, se establecieron repositorios en GitHub esto con el fin de gestionar y distribuir un mismo código.
4. En las temáticas presentadas en el archivo Anexo/Guía 03, este se presenta como en un ejemplo concreto de una aplicación Frontend FullStack que sigue los pasos y abarca los temas definidos en a lo largo de los contenidos. Además, emplea un backend prácticamente similar a situaciones del mundo real. Con estos temas vistos se demuestra que es completamente viable desarrollar una aplicación profesional utilizando estas guías de conocimiento.
5. Para validar y conocer la percepción y opiniones de uno de los principales Stakeholder de este desarrollo tecnológico y mediante una encuesta de percepción

de conceptos y temáticas a una muestra de docentes de la facultad, como resultado se obtuvieron en consideración ciertos ítems que se tuvieron en cuenta para ser un documento completo y garante para la enseñanza, finalmente se evidencia una clara aprobación por parte de los docentes frente al documento y las temáticas presentadas para esta tecnología.

10. REFERENCIAS

- [1] Uriel Hernández, “Angular, React o Vue ¿cuál elegir?” <https://codigofacilito.com/articulos/angular-react-vue>
- [2] D. A, “¿Qué es Angular y cuáles son sus ventajas?,” 2023. <https://www.hostinger.es/tutoriales/que-es-angular>
- [3] A. M. Fard and A. Mesbah, “JSNOSE: Detecting javascript code smells,” *IEEE 13th Int. Work. Conf. Source Code Anal. Manip. SCAM 2013*, pp. 116–125, 2013, doi: 10.1109/SCAM.2013.6648192.
- [4] N. Almasfi and L. Lu, “Code Smell Detection Tool for Java Script Programs,” in *2020 5th International Conference on Computer and Communication Systems (ICCCS)*, IEEE, May 2020, pp. 172–176. doi: 10.1109/ICCCS49078.2020.9118465.
- [5] Alejandra Mujica, “AngularJS: ¿Qué es y por qué debería usarlo en mis aplicaciones web?,” 2021. <https://www.crehana.com/blog/transformacion-digital/angular-js/>
- [6] D. Galán, “¿Qué es AngularJS y por qué deberías usarlo?,” 2020. <https://ifgeekthen.nttdata.com/es/que-es-angularjs-y-por-que-deberias-usarlo>
- [7] TrabajoFrontend, “React vs Angular vs Vue, ¿cuál es la más popular?,” [Online]. Available: <https://www.trabajofrontend.com/blog/react-angular-vue-mas-popular>
- [8] Z. Powell, “Angular vs React: Una Comparación En Profundidad,” 2023. <https://kinsta.com/es/blog/angular-vs-react/>
- [9] Jose Angel Saavedra, “Qué es Github y para qué sirve: una guía para principiantes,” *04 Jun 2023*, 2023. <https://ebac.mx/blog/que-es-github> (accessed Apr. 04, 2024).
- [10] P. B. L. Roberto Hernández Sampieri, Carlos Fernández Collado, *Metodología de la investigación*, 6ª. España: 2014, 2008. [Online]. Available: <https://academia.utp.edu.co/grupobasicoclinicayaplicadas/files/2013/06/Metodología-de-la-Investigación.pdf>
- [11] Kinsta, “Cómo Instalar Angular en Windows, macOS y Linux - Kinsta®. Kinsta®,” 2023. <https://kinsta.com/es/base-de-conocimiento/instalar-angular/>
- [12] J. Sebastián, U. Parra, F. Técnica, and D. E. L. Proyecto, “Docente facultad de ingeniería de sistemas Tunja Diciembre 2021,” 2021, [Online]. Available: <https://repository.usta.edu.co/handle/11634/42975>
- [13] Zadhid Powell, “Cómo Instalar Node.js y npm en Windows, macOS y Linux,” 2023.

- <https://kinsta.com/es/blog/como-instalar-node-js/#cmo-instalar-nodejs-en-macos>
(accessed May 08, 2024).
- [14] “Requirements for Visual Studio Code.,” 2021.
<https://code.visualstudio.com/docs/supporting/requirements>
- [15] Google, “Introduction to the Angular docs.” <https://angular.io/docs>
- [16] Kinsta, “Qué es Node.js y por qué deberías usarlo.,” 2023.
<https://kinsta.com/es/base-de-conocimiento/que-es-node-js/>
- [17] Dr. Lamberto Vera Vélez, “LA INVESTIGACION CUALITATIVA,” *UIPR*, p. 1, [Online].
Available:
https://www.trabajosocial.unlp.edu.ar/uploads/docs/velez_vera__investigacion_cualitativa_pdf.pdf
- [18] Google, “CLI Overview and Command Reference.” <https://angular.io/cli>
- [19] I. G. C. MONDRAGÓN, “¿Por qué Bootstrap 5 dejo de usar jQuery? y ¿Qué es lo que viene?,” 2021, 2021. <https://platzi.com/tutoriales/1331-bootstrap/9730-por-que-bootstrap-5-dejo-de-usar-jquery-y-que-es-lo-que-viene/> (accessed Jan. 08, 2024).
- [20] Microsoft Logo, “TypeScript Documentation.”
- [21] T. L. Ben Lesh, David Driscoll, Jan-Niklas Wortmann, Nicholas Jamieson, OJ Kwon, Paul Taylor, “Reactive Extensions Library for JavaScript.” <https://rxjs.dev/guide/>

11. ANEXO 1: ETAPA 1 VALIDACIÓN DE LAS GUÍAS PROYECTO MINTIC - 2022

11.1. INTRODUCCIÓN

El presente documento lleva a cabo una revisión exhaustiva de los contenidos y talleres abordados en las anteriores guías, con el objetivo de evaluar su persistencia, calidad y la posible obsolescencia de los protocolos vinculados, en especial aquellos que pudieran haber quedado desactualizados o (*deprecated*) debido a cambios en las librerías empleadas.

Las anteriores guías tenían como objetivo de “crear material orientado a docentes de la Universidad Santo Tomas, dentro de este material se encuentran guías de conocimiento enfocadas en la parte Frontend con el framework Angular en la versión 12, talleres para la interacción de la temática y una herramienta software que es la abstracción y aplicación del material”[12].

11.2. OBJETIVOS

Validar las guías de desarrollo realizadas por la Facultad de Ingeniería de Sistemas en el proyecto MISION TIC 2022, para determinar la calidad y actualidad del material, implementando los protocolos definidos en las guías, utilizando las nuevas versiones de la tecnología a la fecha.

1. Replicar las guías, esto con el fin de determinar, los contenidos, y los temas que abarcados.
2. Identificar las tecnologías, librerías y practicas usadas, para luego comparar la información con las nuevas versiones enmarcadas en la documentación de las tecnologías.
3. Registrar desactualización, posibles errores o aspectos que se deban mejorar e incluir.
4. Evaluar los protocolos definidos en las guías, para posteriormente desarrollar el nuevo material con las correspondientes actualizaciones, instrucciones y procedimientos.

11.3. GUÍAS DE DESARROLLO MINTIC

Como parte integral de la concepción de este material de apoyo y respaldo para el aprendizaje, se establece un objetivo claro “la constante evolución de las tecnologías implica que un recurso de calidad puede volverse obsoleto en el transcurso de dos años. Sin embargo, se concibió la posibilidad de convertir este material en una guía de conocimiento, permitiendo su actualización periódica por parte de los docentes y sus equipos de trabajo, con una periodicidad semestral. Esta estrategia asegura que las guías de conocimiento mantengan su vigencia a lo largo del tiempo, una tarea desafiante para los libros en un lapso tan breve” [12].

Las anteriores guías abarcan una amplia gama de temas, comenzando con la instalación del entorno de ejecución, Node.js que esta creado sobre el motor multiplataforma V8 de Google en la versión en su momento 14.17.5 LTS para Windows de 64 Bits. Posteriormente, se procedió a la descarga de los paquetes necesarios mediante el gestor de paquetes NPM, requeridos por Angular CLI. Esta última es una “herramienta de interfaz de línea de comandos utilizada para inicializar, desarrollar y mantener aplicaciones Angular desde la terminal”[18]. Con estos fundamentos establecidos, se llevaron a cabo talleres específicos, cada uno centrado en temáticas particulares y librerías pertinentes, que se detallan a continuación:

- Instalación de programas.
- Instalación de librerías.
- Talleres para el manejo de componentes.
- Talleres para la gestión de rutas.
- Talleres para el manejo de imágenes.
- Inclusión de modelos y mocks, propiedades e inicialización.
- Talleres para la gestión de CRUDS.
- Talleres para el manejo de módulos.

El repositorio que contiene las guías anteriores, junto con sus respectivos objetivos y temas abordados, se encuentra disponible en el siguiente enlace: [Enlace al Repositorio](#).

11.4. PROCESO DE VALIDACIÓN

La validación de las guías se ejecutó de manera sistemática mediante la reproducción paso a paso de cada tema abordado en el material. Durante este proceso, se identificaron hallazgos significativos relacionados con dependencias desactualizadas, así como la necesidad de modificar módulos o componentes debido a cambios en la documentación de las tecnologías. Además, se observaron situaciones en las cuales ciertas librerías, como JQuery, habían quedado en desuso.

Este enfoque de validación implicó una revisión exhaustiva de cada etapa del material, asegurándose de abordar incongruencias o posibilidades de mejora derivada de la evolución de las tecnologías y prácticas de desarrollo.

A continuación, se presenta un resumen de algunos de los hallazgos clave junto con pasos específicos que indican cómo se manifiestan estos problemas en las guías:

1. Desactualización de Versiones:

- **Descripción:** La versión de Node.js, AngularJS, TypeScript y Bootstrap evidencian una desactualización significativa con respecto a las más recientes.
- **Pasos de las Guías:** Se observa que las guías hacen referencia a versiones antiguas de las tecnologías mencionadas, lo que puede afectar la compatibilidad y la ejecución de los procedimientos propuestos.

2. Desuso de jQuery en Bootstrap 5:

- **Descripción:** Se identifica que la librería jQuery ya no es dependencia para Bootstrap 5, decisión tomada por Bootstrap para mejorar la carga y la compatibilidad con navegadores [19].
- **Pasos de las Guías:** Las guías hacen referencia a la dependencia de jQuery en Bootstrap, resultando en incompatibilidades y problemas de replicación.

3. Inclusión de Bootstrap Icons:

- **Descripción:** Se propone la inclusión de las librerías Bootstrap Icons junto con FontAwesome para ampliar la variedad de iconos disponibles.
- **Pasos de las Guías:** No se proporciona información sobre la inclusión de Bootstrap Icons en los pasos de las guías.

4. Cambios en la Sintaxis de Bootstrap 5.2:

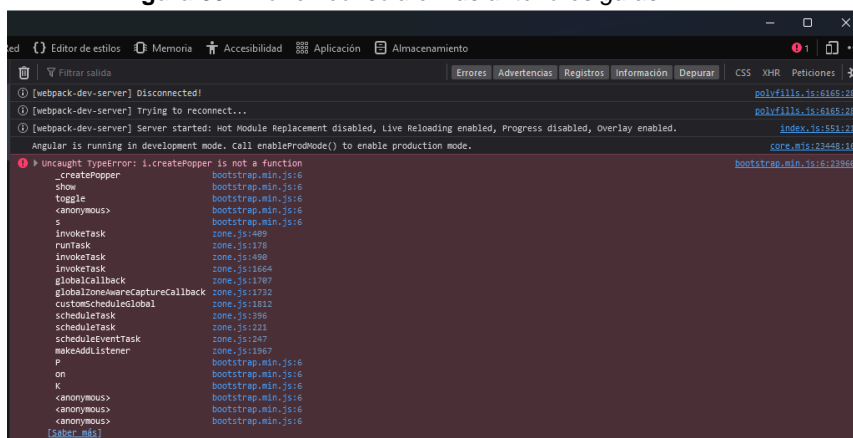
- **Descripción:** En versiones más recientes de Bootstrap 5.3, se han producido cambios en la sintaxis de muchos estilos, lo que afecta la presentación de componentes.

- **Pasos de las Guías:** Las guías reflejan sintaxis en los estilos de Bootstrap para una versión anterior, lo que llevar a inconsistencias con respecto a los cambios en la última versión.

5. Error en Consola del Navegador:

- **Descripción:** Se evidencia un error en la consola del navegador al interactuar con los estilos Dropdown de Bootstrap.
- **Pasos de las Guías:** No se aborda el error en la consola ni se proporcionan soluciones en las guías.

Figura 59 Error en consola en las anteriores guías



Fuente: Autor

11.5. CAMBIOS Y MEJORAS IDENTIFICADAS

A continuación, se presentan las recomendaciones específicas para abordar los hallazgos identificados en el apartado anterior de las guías de desarrollo Proyecto MINTIC. Además, se introduce una propuesta de ampliación temática, considerando que la formación de un Ingeniero de Sistemas debe ser integral y abarcar diversos conocimientos y recursos esenciales para su desarrollo profesional.

1. Actualización de versiones:

- Se realiza una revisión completa de las referencias de versiones en las guías, actualizando a las versiones más recientes y compatibles con las mejores prácticas actuales, parches de seguridad y la performance en la creación de soluciones en cada una de estas tecnologías.

2. Eliminación de dependencia jquery en bootstrap 5:

- Modificar las guías para reflejar la eliminación de la dependencia de jQuery en el contexto de Bootstrap 5, siguiendo las directrices oficiales de Bootstrap.

3. Inclusión de Bootstrap Icons:

- Ampliar las guías para proporcionar instrucciones detalladas sobre la instalación y uso de Bootstrap Icons junto con FontAwesome, brindando a los desarrolladores una variedad más extensa de iconos.

4. Incorporación de cambios de sintaxis en Bootstrap 5.2:

- Actualización en los componentes de cada una de las nuevas guías esto para reflejar los cambios de sintaxis introducidos en la nueva documentación de Bootstrap 5.2.

5. Resolución del error en la consola del Navegador:

- Se identifica un error que provenía en la forma de cómo se importaba los scripts del angular.json, con el respectivo ajuste.
- **Nota:** También se ajusta la manera en cómo se importan los estilos de Bootstrap.

Antes:

Figura 60 Error en la importación de JavaScripts

```
"scripts": [  
  "node_modules/jquery/dist/jquery.min.js",  
  "node_modules/@popperjs/core/dist/umd/popper.min.js",  
  "node_modules/bootstrap/dist/js/bootstrap.min.js"  
]
```

Fuente Autor.

Después:

Figura 61 Corrección de importación de JavaScripts

```
"scripts": [  
  "node_modules/@popperjs/core/dist/umd/popper-base.min.js",  
  "node_modules/bootstrap/dist/js/bootstrap.bundle.min.js"  
]
```

Fuente Autor.

11.5.1. MEJORAS POR INCLUIR

En consulta con el director de Tesis, se toma la decisión de enriquecer el contenido de las guías de desarrollo AngularJS mediante la inclusión de temas adicionales a los presentados en el material anterior, la decisión de agregar más temas se fundamenta en la necesidad de proporcionar al público en formación un recurso educativo integral y actualizado.

Áreas y temas para la ampliación de conocimientos:

- **Modelos**
 - **Interfaces:** Exploración detallada de la implementación y beneficios de las interfaces en TypeScript [20].
 - **Herencias:** Profundización en el concepto de herencia y su aplicación en el desarrollo de software [20].
 - **Composición:** Abordaje de la composición como alternativa a la herencia y su relevancia en el diseño de clases [20].
 - **Métodos entre Clases:** Estudio de los métodos y técnicas de interacción entre clases en TypeScript [20].

- **Operadores**
 - **Instanceof:** Análisis de la utilización del operador `instanceof` para la comprobación de tipos en tiempo de ejecución [20].
 - **Unshift en Arreglos:** Comprensión y aplicación del método `unshift` en operaciones de manipulación de arreglos [20].
- **Tipos**
 - **boolean:** Detalles sobre el tipo de dato booleano y su aplicación en lógica condicional [20].
 - **Uniones “|”:** Exploración de las uniones de tipos mediante el operador `|` en TypeScript [20]
- **Promesas**
 - **Funciones Asíncronas/Await:** Estudio exhaustivo de las funciones asíncronas y el uso de la palabra clave `await` para gestionar promesas[21].
 - **Subscripciones:** Entendimiento y aplicación de subscripciones en el contexto de programación asíncrona[21].
 - **Observadores:** Análisis de la implementación y utilidad de observadores en la programación reactiva[21].
- **Componentes**
 - **Componentes Globales de Inicialización:** Consideraciones y buenas prácticas al trabajar con componentes de inicialización global en AngularJS.
 - **Formularios Compartidos:** Desarrollo y gestión de formularios compartidos entre componentes.
- **Servicios HttpClient**
 - **http.get:** Detalles sobre el uso del método `get` del servicio HttpClient para realizar solicitudes GET [15].
 - **http.post:** Aplicación y consideraciones al utilizar el método `post` para enviar datos a un servidor [15].
 - **http.put:** Implementación y buenas prácticas al emplear el método `put` para actualizar recursos [15].
 - **http.delete:** Manejo de solicitudes de eliminación con el método `delete` de HttpClient [15].
 - **pipe() y map:** Utilización de los operadores `pipe` y `map` en combinación con HttpClient [15].
 - **Token Interceptor:** Implementación y configuración del interceptor de token para manejar autorización en las solicitudes HTTP [15].
- **JSON**

- **fromJson:** Aplicación y contexto del método `fromJson` para convertir datos JSON a objetos en TypeScript [4].
- **Rutas**
 - **ActivatedRoute:** Exploración de `ActivatedRoute` para acceder a los parámetros de una ruta [15].
 - **Router:** Manejo avanzado y estrategias con el servicio Router en AngularJS [15].
 - **Guards (Vigilantes):** Concepto y aplicación de guards para controlar el acceso a rutas basándose en ciertas condiciones [15].

11.6. CONCLUSIONES

1. Durante el proceso de replicación de las guías, estas presentaban desactualizaciones en las versiones de Node.js, AngularJS, TypeScript y Bootstrap. Estas diferencias afectan la vigencia y aplicabilidad del material, destacando la necesidad de una actualización para garantizar su relevancia y eficacia en entornos tecnológicos actuales.
2. Ciertas librerías, en particular la librería jQuery, se encuentra en desuso en versiones más recientes de (Bootstrap5.3), está ya no es dependiente de JQuery. Este hallazgo indica la importancia de revisar y ajustar las dependencias para reflejar las prácticas actuales y evitar el uso innecesario de tecnologías obsoletas.
3. Se abordaron problemas específicos encontrados durante la replicación de las guías, especialmente en relación con errores presentados en la consola. Se proporcionaron soluciones detalladas y se presentó la forma correcta de importar las librerías, asegurando la resolución efectiva de inconvenientes técnicos.
4. Como respuesta a la evolución del panorama tecnológico, se propusieron nuevos conceptos que enriquecerán las guías existentes. Estos incluyen áreas como modelos, operadores, tipos, promesas, componentes, servicios HttpClient, rutas, manipulación de JSON, interceptores de token y guards.

12. ANEXO 2: ETAPA 2 Y 3 DOCUMENTACIÓN Y ESTRUCTURA DE LOS PROTOCOLOS PARA LAS NUEVAS GUÍAS

Antes de iniciar la documentación y estructura de con los nuevos protocolos para esta tecnología, se realiza una serie de hipótesis con el objetivo de comparar y respaldar los beneficios ofrecen estas guías en relación con la problemática general de la falta de material instructivo. La

Tabla 5 muestra una serie de ítems, en donde se comparan las ventajas de contar con este material de aprendizaje.

Tabla 5 Comparativa entre tener y no tener un material de aprendizaje.

Aspecto Beneficiado	Desarrolladores con Guías	Desarrolladores sin Guías
Curva de Aprendizaje	Reducción significativa en la curva de aprendizaje con información detallada y estructurada.	Mayor dificultad para aprender tecnologías nuevas sin orientación específica.
Productividad	Mayor eficacia y productividad al tener recursos disponibles y vigentes.	Menor eficiencia y la posibilidad de cometer errores al carecer de orientación y buenas prácticas.
Resolución de Problemas	Facilidad para resolver problemas gracias a documentación y ejemplos prácticos.	Dificultad para encontrar soluciones eficientes y claras, lo que puede llevar a invertir más tiempo en la resolución.
Mantenimiento del Código	Código más limpio y estructurado en componentes, a forma MVC como lo pide la tecnología	Mayor riesgo en código desorganizado y difícil de interpretar.
Recursos de aprendizaje	Material vigente y disponible para la comunidad de la Facultad de Sistemas.	Falta de recursos estructurados, actualizados y disponibles para la comunidad.

Fuente: Autor

Ahora, con lo nuevo para este framework, Angular ha mejorado significativamente el rendimiento global de la plataforma, lo que se traduce en una experiencia de usuario más fluida y rápida. Otra característica destacada incluye un sistema de enrutamiento mejorado que proporciona una navegación más eficiente y flexible entre las diversas vistas de la aplicación.

En el marco del desarrollo del proyecto, se adopta la metodología de desarrollo prototipado, caracterizada por la construcción iterativa de prototipos que permite una interacción constante con el lector al visualizar y comprender el producto en desarrollo. Esta metodología busca fortalecer los conocimientos y proporcionar una asimilación práctica del funcionamiento de cada componente y protocolo propuesto.

Proceso de instalación y selección de versiones actualizadas.

Análisis y selección de versiones actualizadas	
Entradas	<ul style="list-style-type: none"> • Sitio web nodeJs • Sitio web Angular • Sitio web Bootstrap • Sitio web PostgreSQL
Proceso	Una de las actividades de esta fase es verificar la versión LTS (Long TermSupport) de nodeJs, adicionalmente se comprueban las versiones del framework (Angular), Bootstrap, Express, Type Script y los demás paquetes necesarios en la versión actual. Además de contar con un motor de base de datos PostgreSQL.
Resultados	<ul style="list-style-type: none"> • Imágenes adicionadas a guías de conocimiento

Dicho por (Ulloa, 2021) y lo mencionado en las anteriores guías según (López Herrera, 2016) "PostgreSQL se define como un sistema de gestión de bases de datos objeto-relacional (ORDBMS) fundamentado en PostgreSQL. Por otro lado, PostgreSQL es un sucesor de código abierto del código original de Berkeley, adicionalmente es compatible con una gran parte de la Estándar SQL" [12].

De acuerdo con (Navajas Manuel & Pamplona Sonia 2009). Los Frameworks se define como estructuras de código genérico y reusable, lo anterior con el objetivo de ayudar a los desarrolladores en tareas de programación y tiene como finalidad aumentar la productividad.

(Ulloa, 2021) citando a (Cámara Braña,2020) manifiesta que "angular se define como un marco de trabajo creado, desarrollado y mantenido por Google. Adicionalmente emplea una arquitectura SPA para el desarrollo de aplicaciones Web. Por otro lado, la filosofía de programación es de tipo declarativo, lo cual le favorece más que el tipo imperativo al tener que interactuar con interfaces de usuario. Por otra parte, entre sus características se encuentra el aislamiento del código entre componentes, la inyección de dependencias y su orientación a servicios" [12].

12.1. Construcción del Taller Frontend [Guía01]

Construcción de Talleres Frontend Guía 01	
Problemática Ejemplo	Una empresa fabricante de computadoras ha expresado su interés en desarrollar una aplicación web que le permita crear y configurar computadoras, con distintos periféricos. La aplicación debe permitir la configuración de componentes, incluyendo un monitor con especificaciones de marca y tamaño, así como un ratón y un teclado, ambos con especificaciones idénticas de dispositivo de entrada y marca.
Entradas	En la construcción de los talleres para la parte de Frontend se tiene propuestos una cantidad de talleres que serán acumulativos estos contendrán los contenidos temáticos para el área de Frontend en Angular.
Proceso	<ul style="list-style-type: none">• Taller instalación y prerrequisitos es un proyecto generado por la CLI de Angular el cual solo es ejecutado.• Taller de estructura proyecto generado por la CLI de Angular contendrá el uso del componente principal la creación de estilos y maquetado HTML.• Taller múltiples componentes proyecto generado por la CLI de Angular contendrá la creación de componentes que son cargados en el componente principal.• Taller instalación de librerías proyecto generado por la CLI de angular en este taller se hace uso de Bootstrap y PopperJs creando dos componentes que usaran maquetado de este framework de diseño.• Taller uso de iconos y asignación de puerto proyecto generado por la CLI de angular en este taller se hará uso de la librería de iconos FontAwesome y se hace el cambio de puerto en el despliegue de los proyectos.• Taller de CRUD uso de modelos y pruebas de front, en este proyecto se genera el primer CRUD con su respectivo modelo y se hacen las pruebas de Frontend con un mock.• Taller para la gestión de formularios proporcionada por NgForm y así gestionar la entrada de datos y garantizar la validez de la información ingresada.

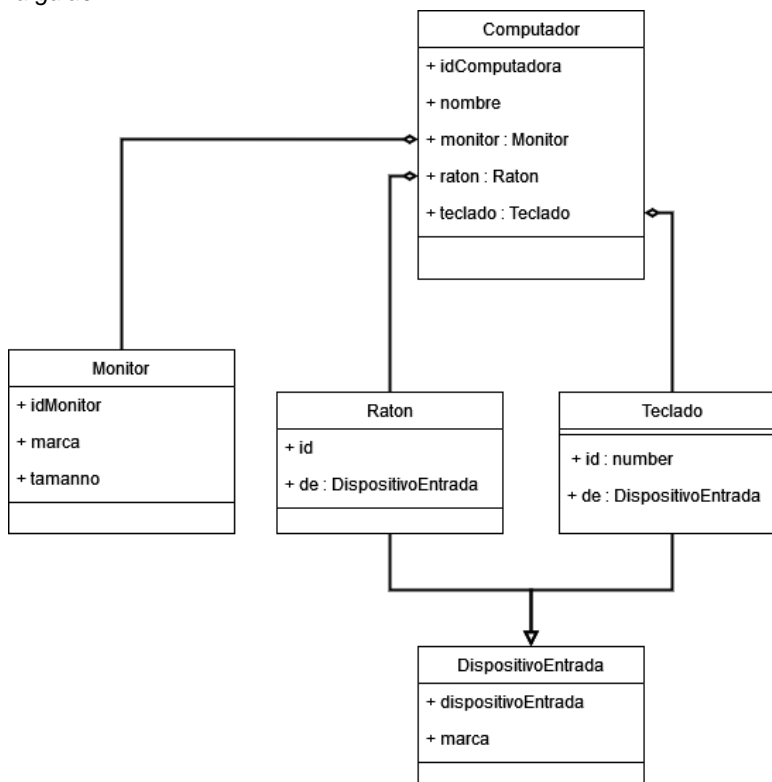
Resultados

- Imágenes adicionales a guías de conocimiento

Diagrama: El diagrama de la **Imagen 1** muestra el contenido temático de clases y métodos a ser trabajados en la guía01.

Imagen 1

Diagrama UML de la guía01



Fuente Autor.

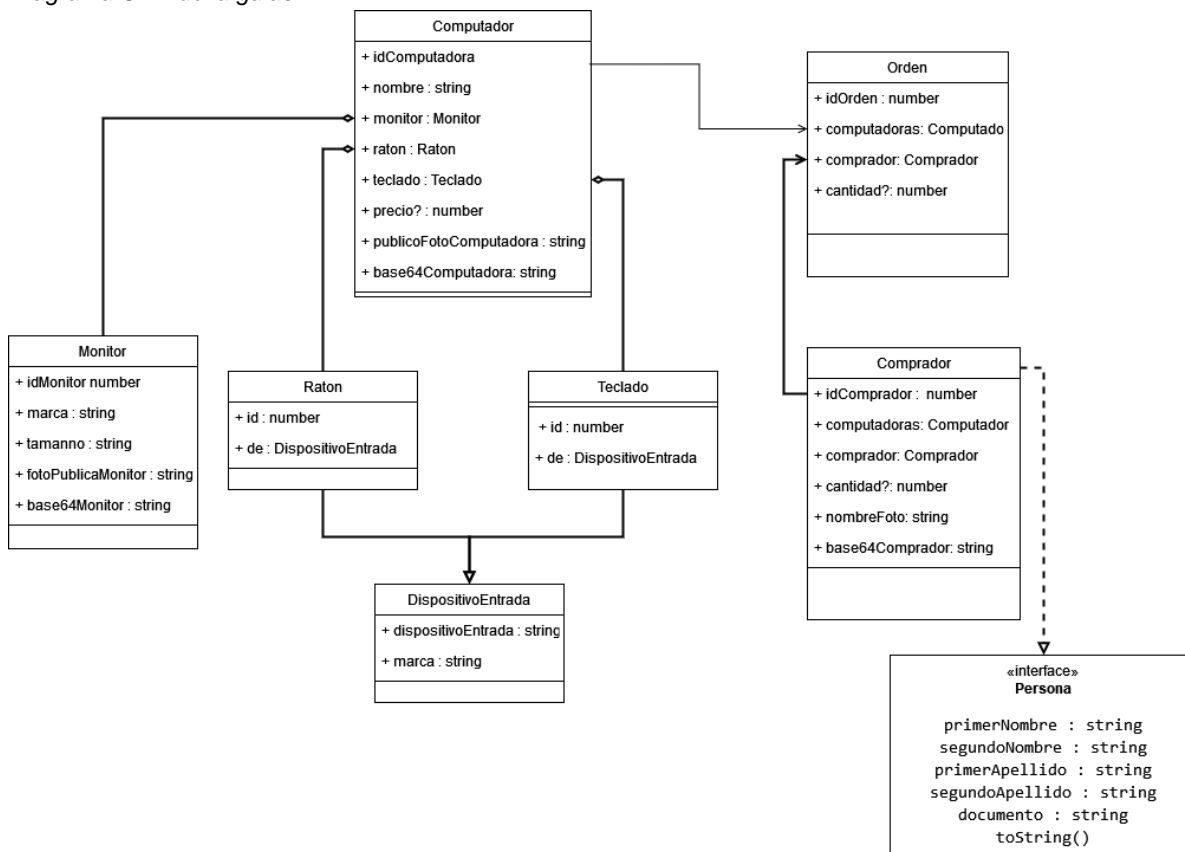
12.2. Construcción del Talleres Frontend [Guia02]

Construcción de Talleres Frontend Guía 02	
Problemática	Una empresa fabricante de computadoras ha expresado su interés en desarrollar una aplicación web que le permita crear y configurar computadoras, con distintos periféricos. La aplicación debe permitir la configuración de componentes, incluyendo un monitor con especificaciones de marca y tamaño, así como un ratón y un teclado, ambos con especificaciones idénticas de dispositivo de entrada y marca.
Entradas	En este nuevo caso, se pide que se agregaren compradores y las facturas de venta (las “órdenes de venta” en donde se detalla las características del computador comprado).
Proceso	En la construcción de los talleres para la parte de Frontend se tiene propuestos una cantidad de talleres que serán acumulativos estos contendrán los contenidos temáticos para el área de Frontend en Angular. • Taller CRUD con imágenes, este proyecto contendrá un CRUD con imágenes donde para manipular las imágenes se hará uso del Base64. • Taller gestión de ruteo es el primer taller que hará uso del ruteo para hacer la navegación de una pequeña aplicación CRUD entre los modelos relacionados. • Taller implementación de ventana de alertas toastr. • Taller implementación de una modal para eliminar.
Resultados	• Imágenes adicionales a guías de conocimiento

Diagrama: El diagrama de la **Imagen 2** muestra el contenido temático de clases y métodos a ser trabajados en la guía02.

Imagen 2

Diagrama UML de la guía02



Fuente Autor.

12.3. Construcción del Taller Frontend [Guía 03]

Detalle del servicio Backend.

Detalle del servicio API Backend.	
Entradas	<ul style="list-style-type: none"> Talleres Frontend Aplicación final Frontend
Proceso	En la construcción de los talleres Frontend se contará con un Backend ya creado, con tecnologías en (express), el apoyo de un marco de trabajo TypeScript recibiendo y permitiendo la creación y la validación de tokens mediante middleware provenientes del backend.
Resultados	<ul style="list-style-type: none"> Backend descargable de un repositorio Git con su respectivo manual de instalación y puesta en marcha del proyecto.

Consumo de un api Backend.

Este apartado tiene como fin que el lector pueda evidenciar el proceso, la experiencia de descargar y conectarse a un api (en local) para hacer uso de un entorno backend, sistema en donde se procesa y se almacenamiento de la información, la seguridad y el acceso a los datos mediante un Token. El objetivo de este material es dar más realismo de cómo se vería una aplicación completa, además de explicar paso a paso (desde el Frontend) como consumir, trabajar y visualizar con los datos que nos entrega la base de datos y el backend.

Consumo de un API Backend y construcción de talleres para la Guía 03

Problemática

Una empresa fabricante de computadoras ha expresado su interés en desarrollar una aplicación web que le permita crear y configurar computadoras, con distintos periféricos. La aplicación debe permitir la configuración de componentes, incluyendo un monitor con especificaciones de marca y tamaño, así como un ratón y un

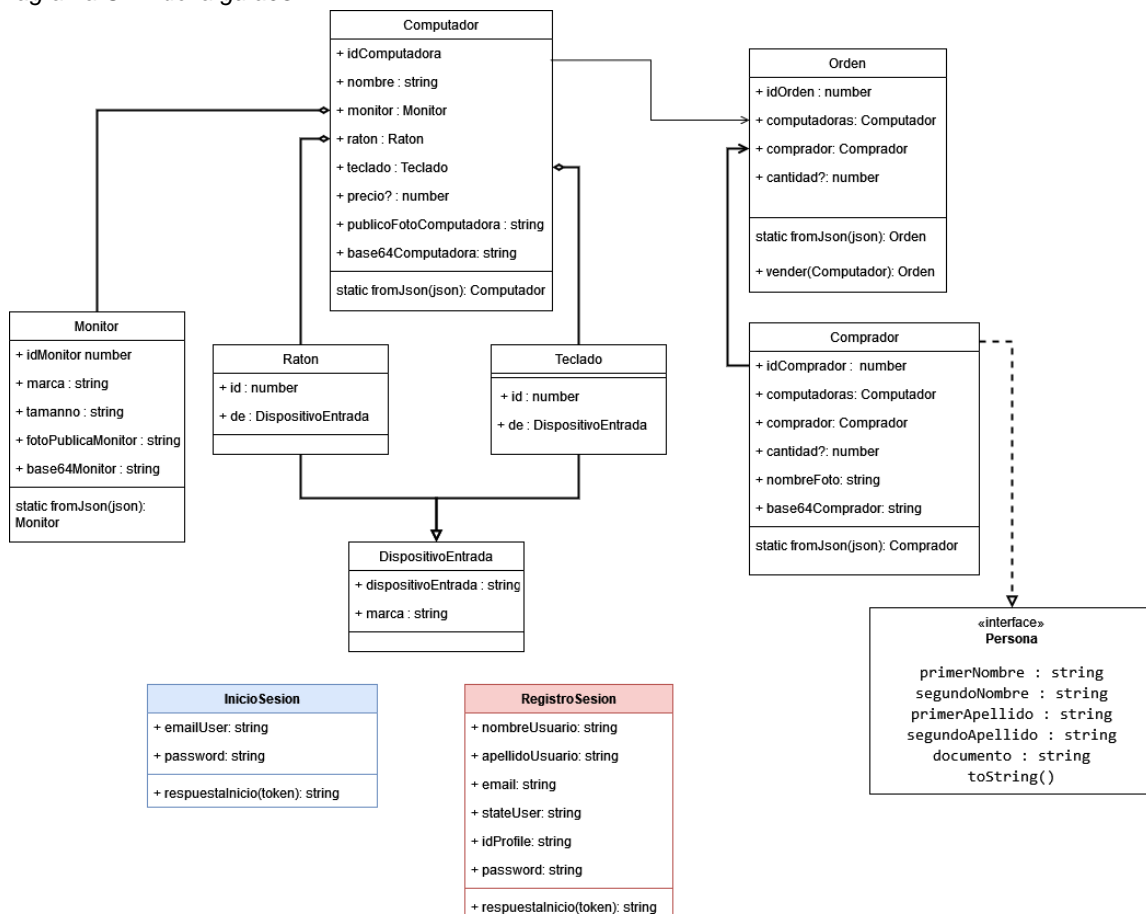
Entradas	<p>teclado, ambos con especificaciones idénticas de dispositivo de entrada y marca.</p> <p>En este nuevo caso, se pide que se agregaren compradores y las facturas de venta (las “órdenes de venta” en donde se detalla las características del computador comprado).</p> <p>Finalmente se pide que el sistema contenga un Login y Signin, esto para la protección y carga de datos.</p> <p>En la construcción de los talleres para la parte de Frontend se tiene propuestos una cantidad de talleres que serán acumulativos estos contendrán los contenidos temáticos para el área de Frontend en Angular.</p>
Proceso	<ul style="list-style-type: none"> • Taller módulos este proyecto contendrá el manejo de múltiples módulos con su ruteo, uso de ventanas modales, pipe, maps, ordenamiento etc. • Taller uso y consumo de web services en este taller se hará el consumo de una API sencilla en express, contendrá un inicio de sesión, ajuste de perfil de usuario y CRUD'S de la problemática anteriormente expuesta. • Taller construcción aplicación final es la recopilación de todos los talleres para la creación de un módulo de autorización y autenticación (token interceptor y un vigilante guard).
Resultados	<ul style="list-style-type: none"> • Imágenes adicionadas a guías de conocimiento

De acuerdo con (Alamilla Hernández, Pérez Romero, et. Al, 2021) citado por (Ulloa, 2021) Los Web Services (WebServices) permiten enlazar, a través del uso de diferentes protocolos, cualquier tipo de tecnología o servicio en la web, obteniendo que aplicaciones se comuniquen con otras aplicaciones de una organización o de cualquier tercero[12].

Diagrama: El diagrama de la **Imagen 3** muestra el contenido temático de clases y métodos a ser trabajados en la guía03.

Imagen 3

Diagrama UML de la guía03



Fuente Autor.

Preparación de recursos: Las actividades para esta fase son la construcción y uso de repositorios.

Preparación de recursos.	
Entradas	<ul style="list-style-type: none">• Talleres Frontend• Aplicación Final
Proceso de actividades	<ul style="list-style-type: none">• Para esta fase son la construcción y uso de repositorios.
Resultados	<ul style="list-style-type: none">• Repositorios en GitHub

Fase 9: Entrega del trabajo. Para la fase de entrega se tienen en cuenta las actividades de creación del documento final como los posibles imprevistos.

Verificación de las guías de conocimientos.	
Entradas	<ul style="list-style-type: none">• Propuesta• Guías de conocimiento• Sugerencias
Proceso de actividades	Para esta fase de entrega, se consideran las actividades para la creación del documento final, anexos, guías, código o repositorio de GitHub, así como la gestión de posibles imprevistos.
Resultados	<ul style="list-style-type: none">• Documento final con formato Word con el contenido y desarrollo del trabajo.
