

MANUAL DEL PROGRAMADOR
APLICACIÓN WEB DENOMINADA "KIRA" PARA FACILITAR UNA EVALUACIÓN
OBJETIVA DE LA SATISFACCIÓN DEL CLIENTE POR PARTE DE LOS GERENTES DE
ESTABLECIMIENTOS EN LA CIUDAD DEL ESPINAL

MIGUEL ANGEL RUIZ GUZMAN
ANDRES CAMILO LOZANO CARODOSO
ANDRES CAMILO RODRIGUEZ PORTELA

INSTITUTO TOLIMENSE DE FORMACION TECNICA PROFESIONAL “ITFIP”
FACULTAD DE INGENIERIA Y CIENCIAS AGRO-INDUSTRIALES
TECNOLOGIA EN GESTION DE INFORMACION
ESPINAL-TOLIMA
2025

MANUAL DEL PROGRAMADOR
APLICACIÓN WEB DENOMINADA "KIRA" PARA FACILITAR UNA EVALUACIÓN
OBJETIVA DE LA SATISFACCIÓN DEL CLIENTE POR PARTE DE LOS GERENTES DE
ESTABLECIMIENTOS EN LA CIUDAD DEL ESPINAL

MIGUEL ANGEL RUIZ GUZMAN
ANDRES CAMILO LOZANO CARODOSO
ANDRES CAMILO RODRIGUEZ PORTELA

MONOGRAFIA

ING, JUAN DAVID DIAZ VALENCIA

INSTITUTO TOLIMENSE DE FORMACION TECNICA PROFESIONAL "ITFIP"
FACULTAD DE INGENIERIA Y CIENCIAS AGRO-INDUSTRIALES
TECNOLOGIA EN GESTION DE INFORMACION
ESPINAL-TOLIMA

2025

CONTENIDO

| | |
|--|----|
| Contenido | |
| CONTENIDO | 4 |
| 1. MODELO ENTIDAD RELACION..... | 6 |
| 2. RESUMEN..... | 7 |
| 3. OBJETIVO..... | 8 |
| 4. FINALIDAD DEL MANUAL..... | 8 |
| 5. INTRODUCCION | 9 |
| 6. ASPECTOS TECNICOS | 10 |
| 6.1 Herramientas utilizadas para el desarrollo | 10 |
| 6.1.1 Visual Studio Code | 10 |
| 6.1.2 LARAGON | 11 |
| 6.1.3 GIT BASH | 11 |
| 6.1.4 GITHUB | 12 |
| 7. DIAGRAMAS DE MODELACION | 13 |
| 7.1 Diagramas de clases. | 13 |
| 7.2 Diagrama de casos de uso..... | 14 |
| 7.3 Diccionario de datos | 15 |
| 8. ASPECTO TECNICO DEL DESARROLLO DEL SISTEMA..... | 16 |
| 8.1 MODIFICACION LOCAL | 16 |
| 9. REQUERIMIENTOS DE SOFTWARE..... | 24 |
| 9.1 Requisitos mínimos computador | 24 |
| 10. WEBGRAFIA | 25 |

TABLA DE FIGURAS

| | |
|---|----|
| FIGURA 1 Diagrama de modelado de clases..... | 13 |
| FIGURA 2 DIAGRAMA DE CASOS DE USO..... | 14 |
| FIGURA 3 DICCIONARIO DE DATOS | 15 |
| FIGURA 4 PAGINA WEB PARA DESCARGAR LARAGON..... | 16 |
| FIGURA 5 PAGINA WEB PARA DESCARGA DE VISUAL STUDIO CODE | 16 |
| FIGURA 6 PAGINA WEB DE DESCARGA DE GIT BASH..... | 17 |
| FIGURA 7 PAGINA WEB DE DESCARGA DE GITHUB | 18 |
| FIGURA 8 PAGINA WEB DE DESCARGA DE BOOTSTRAP..... | 18 |
| FIGURA 9 PAGINA WEB DE DESCARGA DE PHP..... | 19 |
| FIGURA 11 PAGINA WEB DE DESCARGA DE COMPOSER | 19 |
| FIGURA 12 PAGINA WEB DE DESCARGA DE NODE.JS | 20 |
| FIGURA 13 PAGINA WEB DE DESCARGA DE ARDUINO..... | 21 |
| FIGURA 14 Ingreso a Terminal de VisualStudioCode..... | 21 |
| FIGURA 15 Activación de React..... | 21 |
| FIGURA 16 Levantamiento local de servicio | 22 |
| FIGURA 17 Activación de Laravel desde Laragon | 22 |

1. MODELO ENTIDAD RELACION

El siguiente manual ha sido elaborado con el propósito de brindar la información necesaria para la instalación, mantenimiento y exploración de la plataforma web KIRA (Kit de Retroalimentación y Apreciación), diseñada como una herramienta híbrida física y digital que permite registrar y gestionar la satisfacción del cliente mediante un sistema de calificación intuitivo.

KIRA está orientada a facilitar la recolección de retroalimentación inmediata en puntos de atención, combinando una interfaz web amigable con una caja física integrada mediante tecnología Arduino, lo cual fortalece los procesos de evaluación y mejora continua en organizaciones de cualquier sector.

Este manual proporciona la información técnica esencial sobre la arquitectura y estructura del software, con el fin de que cualquier desarrollador pueda comprender su funcionamiento interno y realizar modificaciones o mejoras de manera adecuada y estructurada. Se describen los lenguajes, frameworks, y herramientas utilizadas como Laravel, React, Bootstrap y Arduino IDE, así como también se detallan los módulos funcionales y el flujo de navegación del sistema.

Este documento está dirigido tanto a desarrolladores como a personal técnico encargado del mantenimiento y evolución de la plataforma, contribuyendo al aseguramiento de su correcto funcionamiento, escalabilidad y adaptabilidad a distintos contextos organizacionales.

2. RESUMEN

El presente manual detalla los aspectos técnicos e informáticos de la plataforma web KIRA (Kit de Retroalimentación y Apreciación), con el objetivo de explicar su estructura y funcionamiento al personal encargado de administrarla, editarla o configurarla.

Esta guía está organizada en secciones que describen las herramientas utilizadas para el desarrollo de la plataforma, acompañadas de explicaciones paso a paso para su correcta implementación y modificación.

La plataforma combina funcionalidades web con componentes físicos basados en Arduino, lo cual requiere de una integración entre hardware y software. A lo largo del manual se describe el papel de cada uno de estos elementos, brindando recomendaciones para su uso adecuado y garantizando así el óptimo desempeño del sistema de información.

3. OBJETIVO

Dar a conocer el uso adecuado del software KIRA (Kit de Retroalimentación y Apreciación) en aspectos técnicos de manera descriptiva e ilustrada sobre los componentes y funcionalidades que conforman el buen funcionamiento del sistema de información.

4. FINALIDAD DEL MANUAL

La finalidad de este manual técnico es instruir a la persona que quiera administrar, editar o configurar la plataforma KIRA (Kit de Retroalimentación y Apreciación) usando las debidas herramientas.

5. INTRODUCCION

El presente manual se realiza con el fin de detallar el software “KIRA (Kit de Retroalimentación y Apreciación)” en términos técnicos, para que la persona que vaya a administrar, editar o configurar la plataforma web lo haga de una manera apropiada y eficiente. El documento se encuentra estructurado en las siguientes secciones:

-ASPECTOS TEÓRICOS: Se presentan conceptos, definiciones y explicaciones de los componentes del aplicativo desde un punto de vista teórico, facilitando el entendimiento del funcionamiento del sistema de información y las herramientas empleadas.

-DIAGRAMAS DE MODELAMIENTO: Incluye diagramas e ilustraciones representativos del funcionamiento general del sistema, su arquitectura y procesos internos.

-ASPECTO TÉCNICO DEL DESARROLLO DEL SISTEMA: Proporciona una guía detallada sobre los componentes del software desde una perspectiva técnica, abarcando temas como almacenamiento de datos, estructura del código, tecnologías utilizadas y recomendaciones para el uso adecuado de la plataforma.

-REQUERIMIENTOS DEL SOFTWARE: Enumera los requerimientos mínimos y recomendados, tanto de hardware como de software, necesarios para el correcto funcionamiento de KIRA.

6. ASPECTOS TECNICOS

La plataforma web “KIRA (Kit de Retroalimentación y Apreciación)” tiene como finalidad optimizar los procesos de recolección, análisis y gestión de retroalimentación por medio de las funciones que integra la plataforma, tanto en su entorno digital como físico. Se recomienda que el presente manual sea manipulado exclusivamente por la persona encargada de administrar, editar o configurar la plataforma KIRA, con el objetivo de salvaguardar la integridad y seguridad de los datos almacenados en la base de datos, los cuales podrían ser sensibles y no deben utilizarse con fines distintos a los establecidos por el sistema.

6.1 Herramientas utilizadas para el desarrollo

En esta sección se procede a explicar las herramientas informáticas empleadas para el desarrollo de la plataforma.

6.1.1 Visual Studio Code

Visual Studio Code (VSCode) es un entorno de desarrollo integrado (IDE) de código abierto desarrollado por Microsoft. Es ampliamente utilizado por desarrolladores para escribir, depurar y mantener código en múltiples lenguajes de programación. Para el desarrollo de la plataforma web “KIRA (Kit de Retroalimentación y Apreciación)”, se utilizó esta herramienta gracias a sus múltiples beneficios:

- Multiplataforma: VSCode funciona en sistemas operativos Windows, macOS y Linux, lo que permite a los desarrolladores trabajar desde diferentes entornos sin complicaciones.

- Extensible: Una de las principales ventajas de VSCode es su extensibilidad mediante extensiones. Estas permiten agregar soporte para distintos lenguajes, integraciones con bases de datos, herramientas de desarrollo, temas visuales, y muchas otras funcionalidades útiles para el desarrollo web.

- Editor de texto avanzado: Ofrece características como resaltado de sintaxis, autocompletado inteligente, sugerencias contextuales, refactorización de código y navegación eficiente, lo que mejora la productividad del desarrollador.

- Integración con Git: VSCode incluye integración nativa con Git, permitiendo gestionar versiones del código, realizar commits, revisar cambios y colaborar con otros desarrolladores de manera sencilla.

- Depuración integrada: Proporciona potentes herramientas de depuración para diversos lenguajes, facilitando la identificación y solución de errores directamente desde el entorno de desarrollo.

- Terminal integrada: Cuenta con una terminal que permite ejecutar comandos sin salir del IDE, lo cual resulta útil para instalar paquetes, correr scripts, ejecutar servidores locales, entre otros.

- Snippets y atajos de teclado: Soporta snippets personalizables y una extensa colección de atajos de teclado, lo cual agiliza tareas repetitivas y mejora el flujo de trabajo del desarrollador.

En resumen, Visual Studio Code es una herramienta robusta, flexible y eficiente que se convirtió en la opción ideal para el desarrollo de KIRA, gracias a su excelente rendimiento, facilidad de personalización y compatibilidad con las tecnologías utilizadas.

La versión utilizada para el desarrollo fue la versión gratuita disponible públicamente.

6.1.2 LARAGON

Laragon es un entorno de desarrollo local moderno, rápido y portable, diseñado especialmente para facilitar la creación de aplicaciones web. A diferencia de otras herramientas como XAMPP, Laragon ofrece una arquitectura más ligera, rápida y flexible, ideal para trabajar con múltiples tecnologías modernas. A continuación, se describen sus principales características:

- Portabilidad y ligereza: Laragon es portable, lo que significa que puede ejecutarse desde una memoria USB o cualquier ubicación sin necesidad de instalación compleja. Su rendimiento es notablemente superior gracias a su arquitectura optimizada.
- Incluye Apache y Nginx: Laragon ofrece soporte para servidores web como Apache y Nginx, permitiendo que el desarrollador elija el más conveniente para su proyecto.
- Bases de datos: Viene preconfigurado con MySQL y MariaDB, proporcionando un entorno robusto para la gestión de datos. Además, soporta otras bases de datos como PostgreSQL si el desarrollador así lo requiere.
- Soporte para múltiples lenguajes: Aunque se centra principalmente en PHP, Laragon también permite configurar entornos para trabajar con otros lenguajes como Node.js, Python, Ruby, y Java, gracias a su arquitectura modular.
- Integración con herramientas modernas: Incluye compatibilidad con Composer, Git, Node.js, npm, y otras herramientas de desarrollo modernas que se usan en proyectos actuales como el de KIRA.
- Virtual Hosts automáticos: Laragon facilita la creación de dominios personalizados (ej. kira.test) de forma automática sin necesidad de modificar archivos del sistema manualmente, lo que mejora la organización de proyectos locales.
- Interfaz sencilla: Posee una interfaz amigable que permite arrancar y detener servicios con un solo clic, ver logs, acceder rápidamente a las carpetas de proyecto, base de datos o configuraciones del servidor.

En resumen, Laragon fue seleccionado para el desarrollo de KIRA debido a su facilidad de uso, rendimiento optimizado, portabilidad y compatibilidad con las tecnologías utilizadas en el proyecto, superando a otras alternativas como XAMPP en flexibilidad y velocidad de trabajo.

6.1.3 GIT BASH

Git Bash es una herramienta que proporciona una interfaz de línea de comandos basada en Bash para sistemas Windows, permitiendo a los desarrolladores utilizar comandos de Git de manera similar a como lo harían en sistemas operativos Unix/Linux. Esta aplicación forma parte del paquete de instalación oficial de Git para Windows.

Su función principal en el desarrollo del sistema KIRA es facilitar el uso de comandos de control de versiones, como git clone, git commit, git push o git pull, permitiendo así llevar un seguimiento eficiente del código fuente mediante Git. Además, ofrece una terminal flexible para ejecutar scripts, mover archivos y navegar por directorios del proyecto con comandos avanzados, todo desde una ventana de consola.

Gracias a su compatibilidad con el entorno Bash, Git Bash permite ejecutar instrucciones que no estarían disponibles por defecto en la terminal de Windows, lo que lo convierte en una herramienta esencial para el flujo de trabajo moderno en desarrollo web y colaborativo con plataformas como GitHub.

6.1.4 GITHUB

GitHub es una plataforma de alojamiento de código fuente basada en la nube que permite gestionar proyectos mediante el sistema de control de versiones Git. Esta herramienta es ampliamente utilizada en el desarrollo colaborativo de software, ya que facilita el trabajo en equipo, el seguimiento de cambios, la revisión de código y la integración continua.

En el desarrollo del sistema KIRA, GitHub cumple un papel fundamental como repositorio central del proyecto. Permite a los desarrolladores almacenar y compartir el código de forma remota, mantener un historial de versiones, crear ramas para nuevas funcionalidades, fusionar cambios (pull requests) y llevar un control detallado del progreso del software.

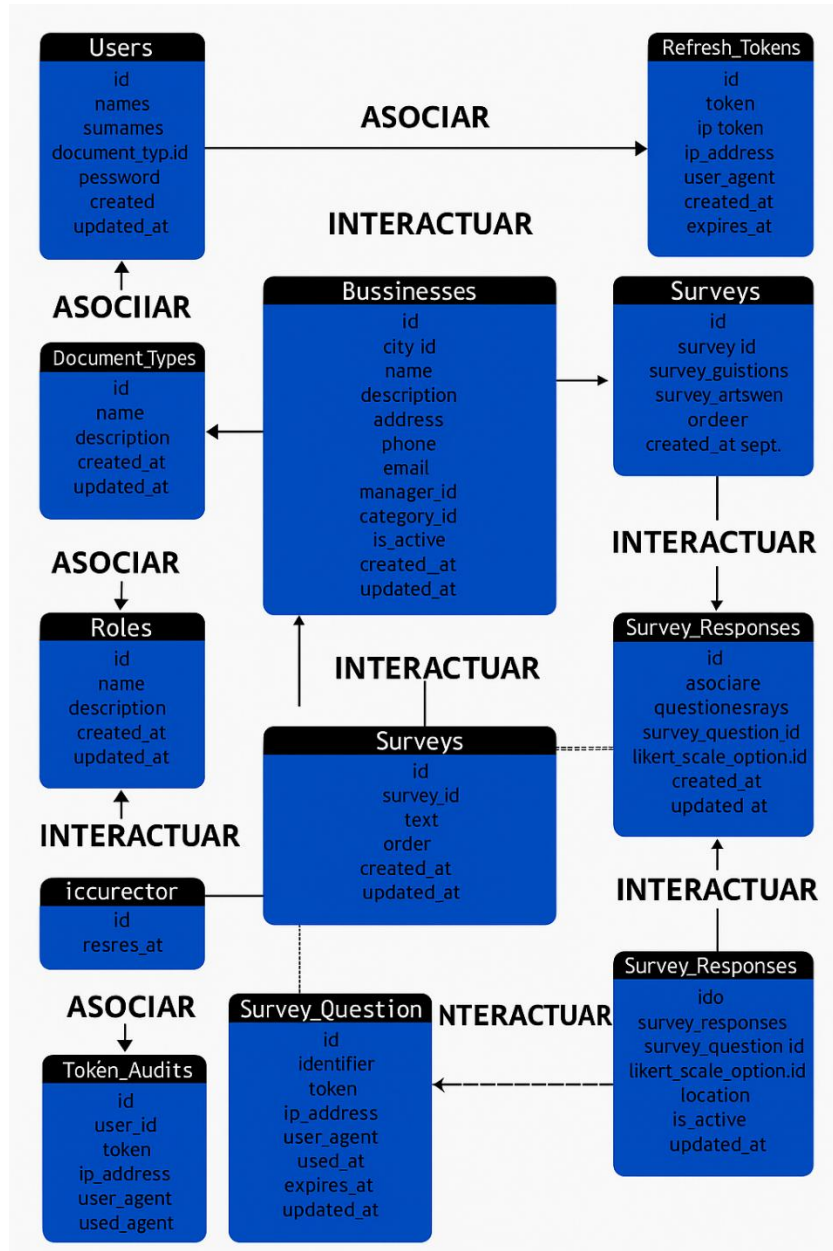
Además, GitHub brinda herramientas adicionales como tableros de proyectos, issues, documentación (README, Wikis), y protección de ramas, lo que fortalece la organización del equipo y mejora la calidad del código entregado. También sirve como medio de integración con otras herramientas de desarrollo como Git Bash, Visual Studio Code y plataformas de despliegue continuo.

7. DIAGRAMAS DE MODELACION

7.1 Diagramas de clases.

El diagrama de clases está compuesto de las entidades y atributos que se crearon para el almacenamiento de datos del software.

FIGURA 1 Diagrama de modelado de clases.



FUENTE: Por los autores.

En la Figura 1, se representa el modelado de clases del sistema KIRA, en el cual se detallan las principales entidades (clases) utilizadas para la estructuración, gestión y almacenamiento de datos dentro del aplicativo. Cada clase cumple una función específica dentro del flujo de evaluación de la satisfacción del cliente, integrando tanto componentes del entorno web como físico (IoT).

-USERS: Representa a los usuarios del sistema, incluyendo administradores, gerentes y empleados. Esta clase almacena información personal, de autenticación y de relación con el negocio al que pertenecen, así como su rol dentro de la plataforma.

-BUSINESSES: Corresponde a los establecimientos registrados en la plataforma. Almacena información como nombre, dirección, ciudad, gerente encargado y categoría comercial. Esta clase está directamente relacionada con encuestas, dispositivos físicos y usuarios.

-SURVEYS: Define las encuestas que serán aplicadas en los negocios. Cada encuesta puede estar vinculada a preguntas, dispositivos físicos y accesos mediante códigos QR o enlaces.

-SURVEY_QUESTIONS: Almacena las preguntas que componen cada encuesta. Estas preguntas pueden estar asociadas a dispositivos físicos (botoneras) y se ordenan según su prioridad de visualización.

-SURVEY_RESPONSES: Registra las respuestas proporcionadas por los clientes, ya sea mediante dispositivos físicos o desde el entorno web. Esta clase conecta directamente con preguntas, encuestas, negocios y empleados evaluados.

-SURVEY_RESPONSE_ANSWERS: Contiene las respuestas individuales de cada pregunta dentro de una encuesta, con su correspondiente opción en la escala Likert seleccionada.

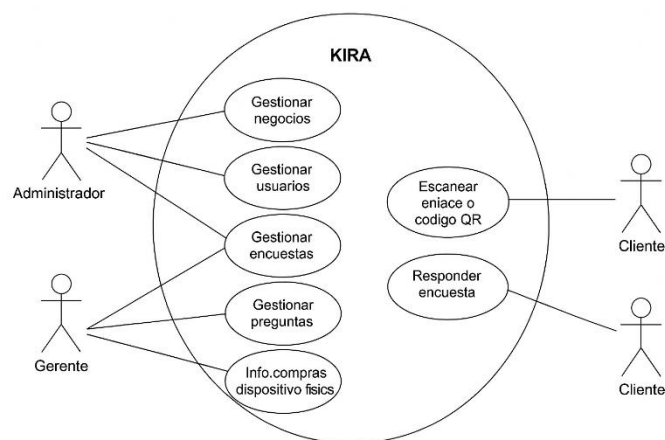
-IOT_DEVICES: Representa los dispositivos físicos integrados (como botoneras de reacción) que permiten al cliente registrar su experiencia en tiempo real. Cada dispositivo está vinculado a una encuesta, una pregunta específica y un negocio determinado.

Cada clase en este modelo mantiene relaciones de tipo ASOCIAR (cuando hace referencia a otra clase como dependencia) o INTERACTUAR (cuando intercambia datos activamente con otra clase). Esta estructura permite una gestión ordenada y escalable de los procesos de evaluación, asegurando trazabilidad, segmentación de la información y conexión directa entre el entorno físico y digital.

7.2 Diagrama de casos de uso

En el diagrama de caso de uso se detalla el papel a desempeñar en relación con el aplicativo por parte de las personas relacionadas, en este caso el actor principal es el administrador, el gerente y el cliente.

FIGURA 2 DIAGRAMA DE CASOS DE USO



FUENTE: Por los autores.

7.3 Diccionario de datos

Para el almacenamiento de datos del software, se define los campos necesarios para cada una de las entidades relacionadas con el aplicativo.

FIGURA 3 DICCIONARIO DE DATOS

| personal acces tokens | | | |
|-----------------------|---------------|--------|--|
| Nombre de campo | Tipo de campo | Tamaño | Descripción |
| id | bigint | 20 | Identificador único del token de acceso personal. |
| tokenable_type | varchar | 255 | Tipo de modelo asociado al token (por ejemplo, usuario, administrador, etc.). |
| tokenable_id | bigint | 20 | ID del modelo relacionado con el token (referencia al usuario u otra entidad). |
| name | varchar | 255 | Nombre descriptivo del token, útil para identificarlo fácilmente. |
| token | varchar | 64 | Valor encriptado del token que se utiliza para autenticación. |
| abilities | text | | Permisos o capacidades asignadas al token (puede incluir acciones permitidas). |
| last_used_at | timestamp | | Fecha y hora de la última vez que se utilizó el token. |
| expires_at | timestamp | | Fecha y hora en que el token expirará (si aplica). |
| created_at | timestamp | | Fecha y hora de creación del registro. |
| updated_at | timestamp | | Fecha y hora de la última actualización del registro. |

1

¹ [1] A. C. Rodríguez Portela, Diccionario de datos tablas KIRA [archivo Excel], en Estructura de datos de la plataforma KIRA, 2025. [En línea]. Consultado: 01-jun-2025. Disponible en: <https://github.com/CamiloDax/KIRA-PROYECTO-DE-GRADO/blob/main/DICCIONARIO%20DE%20DATOS%20TABLAS%20KIRA.xlsx>

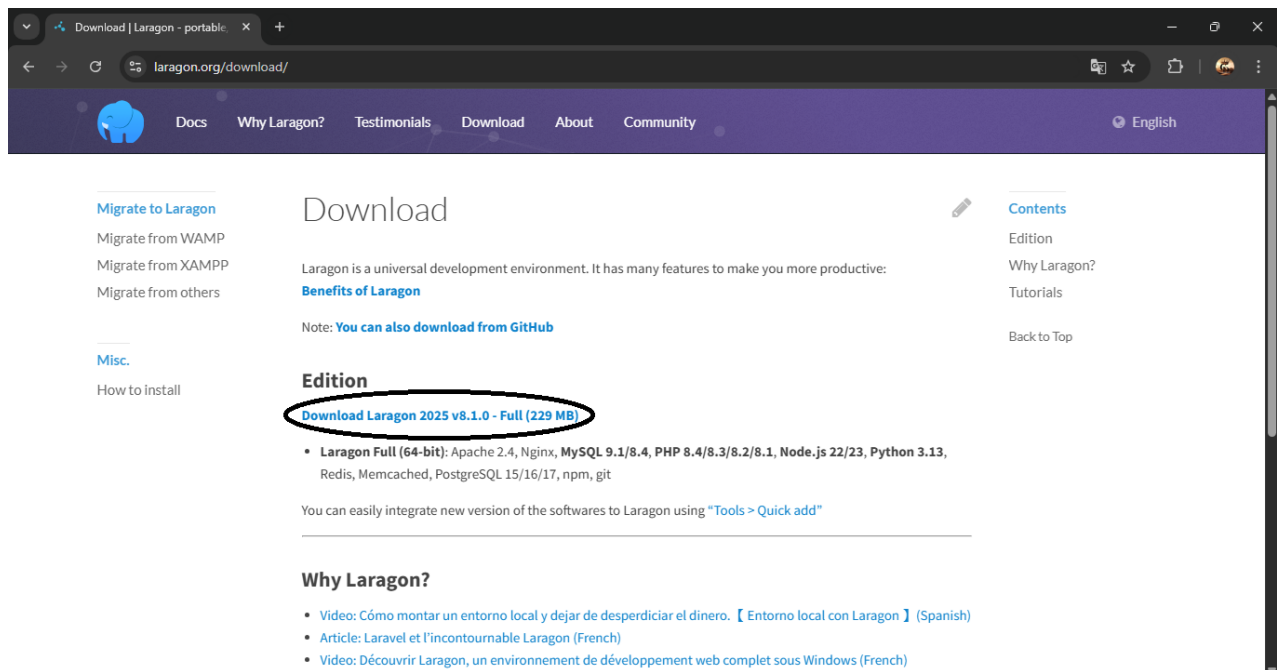
8. ASPECTO TECNICO DEL DESARROLLO DEL SISTEMA

En la siguiente sección se procede a realizar una descripción detallada sobre los aspectos técnicos del aplicativo, relacionado con la instalación de las herramientas necesarias para realizar modificaciones requeridas de manera ordenada.

8.1 MODIFICACION LOCAL

Si el desarrollador quiere realizar modificaciones del Software de manera local, tendrá que realizar la instalación de componentes adicionales, para empezar, se debe instalar Laragon versión 8.1.0, la cual se consigue de manera gratuita en la página: <https://laragon.org/download/>.

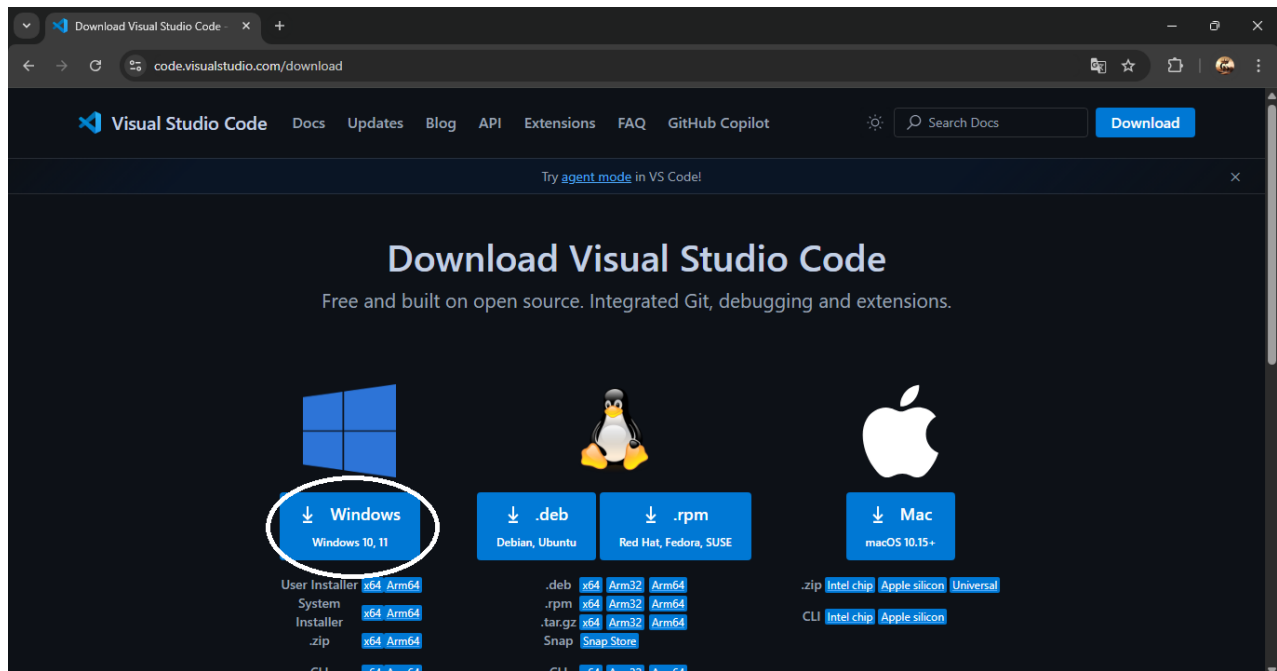
FIGURA 4 PAGINA WEB PARA DESCARGAR LARAGON



FUENTE: (LARAGON)

Al instalar Laragon podemos proceder a instalar el editor de código Visual Studio Code, descargamos la última versión en la página oficial: <https://code.visualstudio.com/download> esta descarga es completamente gratuita.

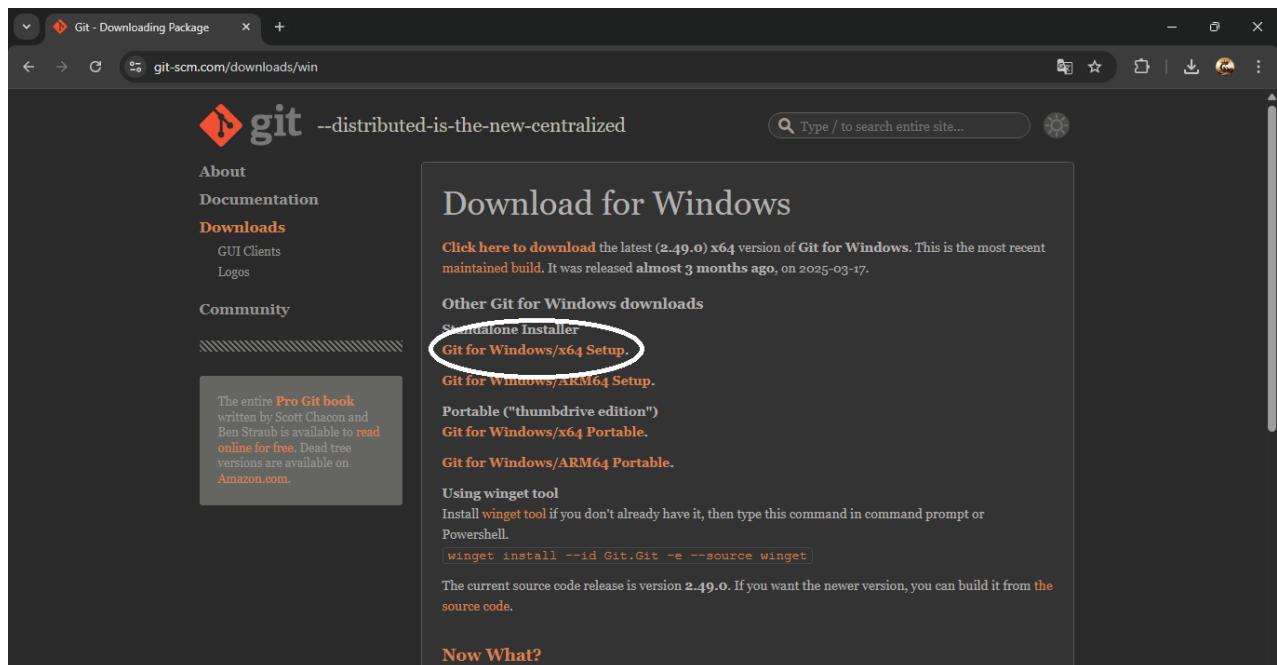
FIGURA 5 PAGINA WEB PARA DESCARGA DE VISUAL STUDIO CODE



FUENTE: (Visual Studio Code)

Al instalar Visual Studio Code, podemos continuar con la instalación de las demás herramientas, para la proxima herramienta que es GitBash, debemos dirigirnos directamente a la pagina oficial y descargarla de forma gratuita: <https://git-scm.com/downloads/win>.

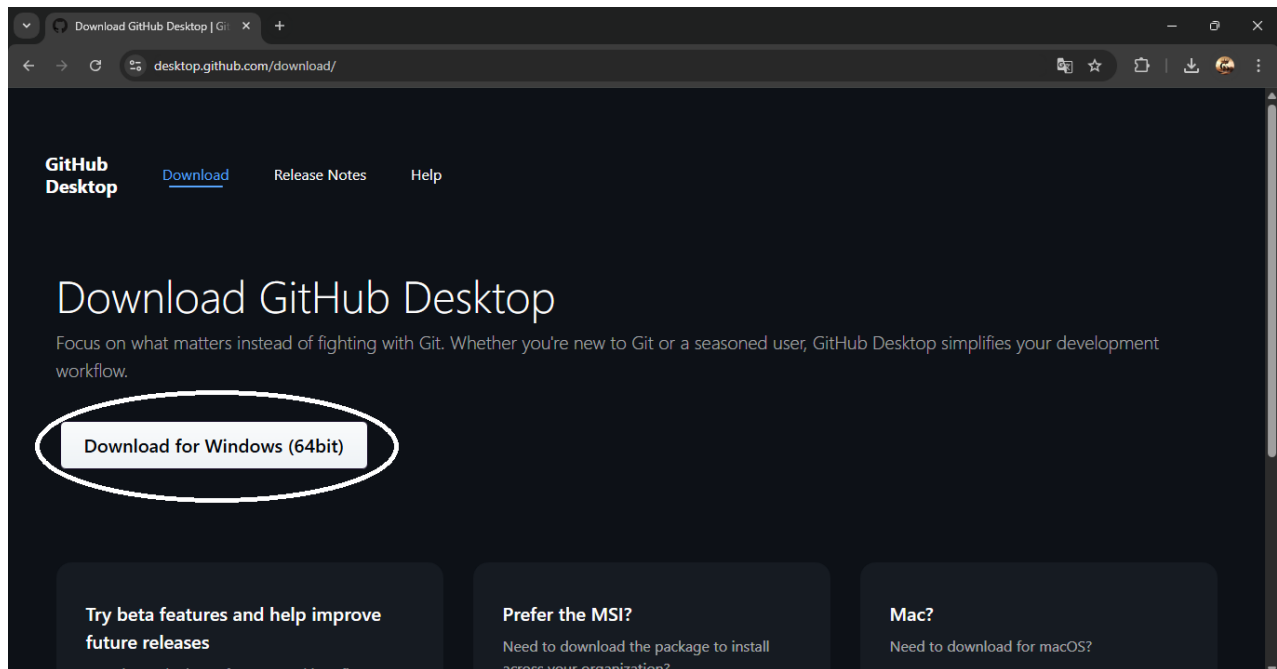
FIGURA 6 PAGINA WEB DE DESCARGA DE GIT BASH



FUENTE: (Git Bash)

Ya luego de terminar la instalación de GitBash, podemos continuar instalando GitHub desde su pagina oficial completamente gratis: <https://desktop.github.com/download/>.

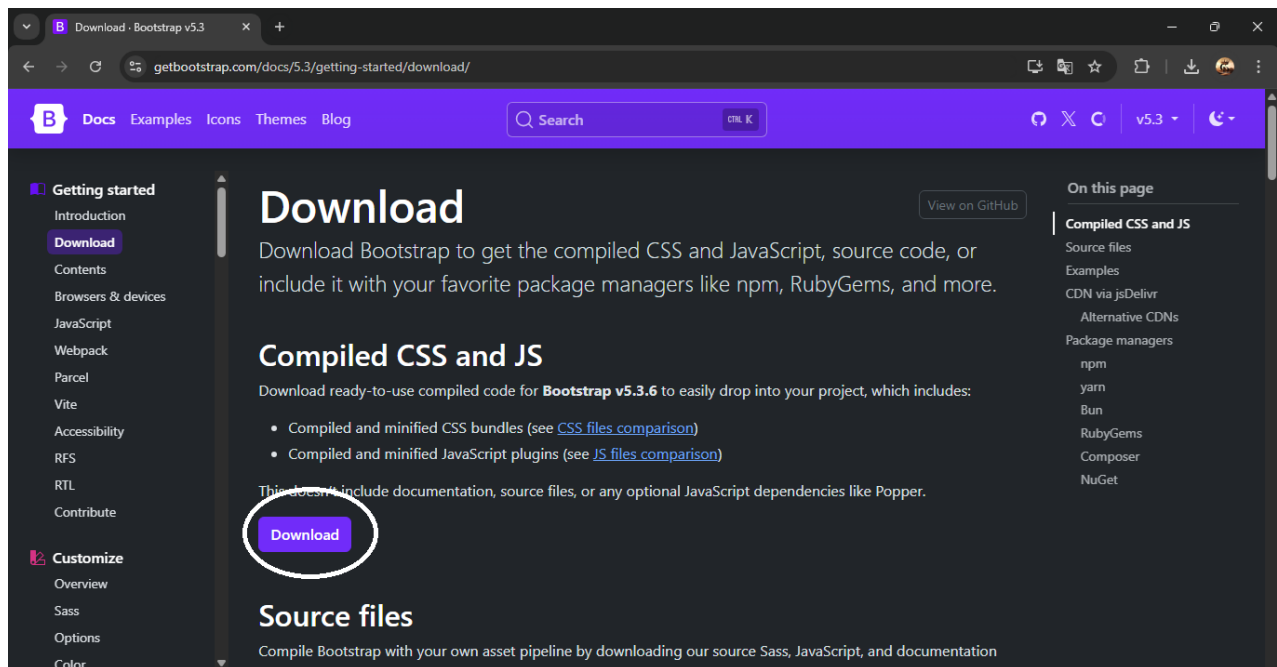
FIGURA 7 PAGINA WEB DE DESCARGA DE GITHUB



FUENTE: (GITHUB)

Instalado ya GitHub podemos comenzar la instalación de librerías, y tecnologías, la primera librería instalada es la de Bootstrap, para ello vamos a la pagina oficial que dejo a continuación y descargamos como en la imagen: <https://getbootstrap.com/docs/5.3/getting-started/download/>.

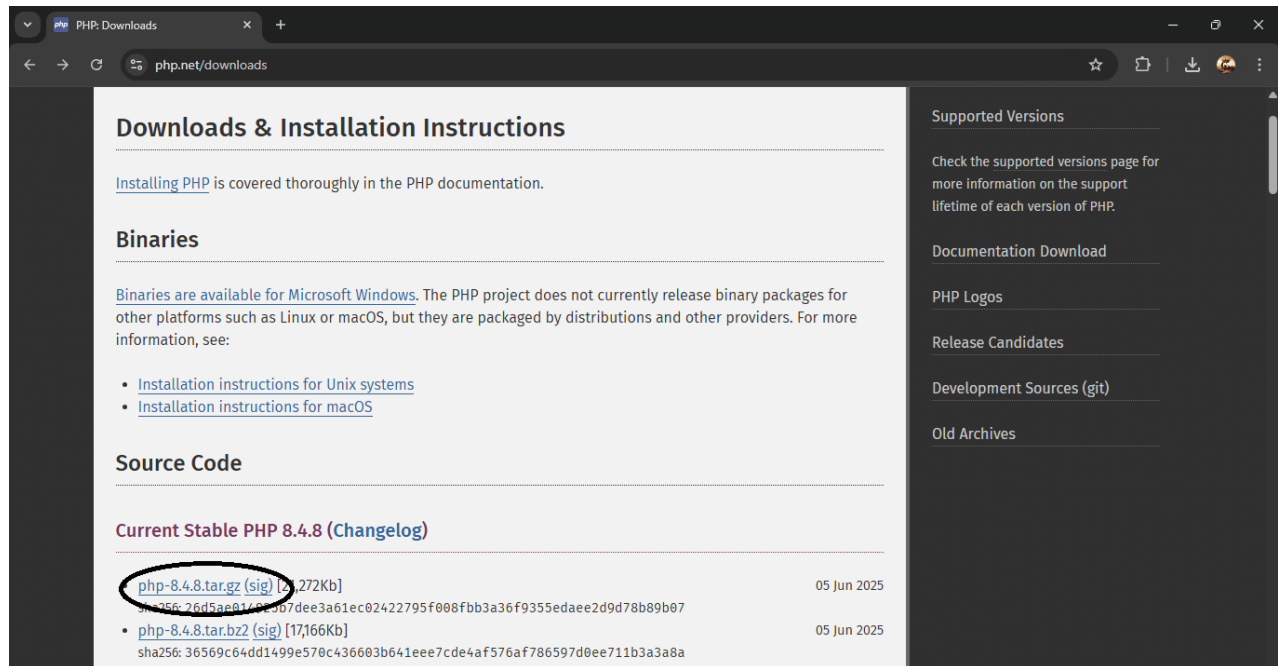
FIGURA 8 PAGINA WEB DE DESCARGA DE BOOTSTRAP



FUENTE: (Bootstrap)

Terminada la instalación de Bootstrap, podemos continuar instalando PHP para que quede habilitado en nuestro computador si es la primera vez que trabajamos con PHP, desde la siguiente pagina que es el sitio oficial: <https://www.php.net/downloads>.

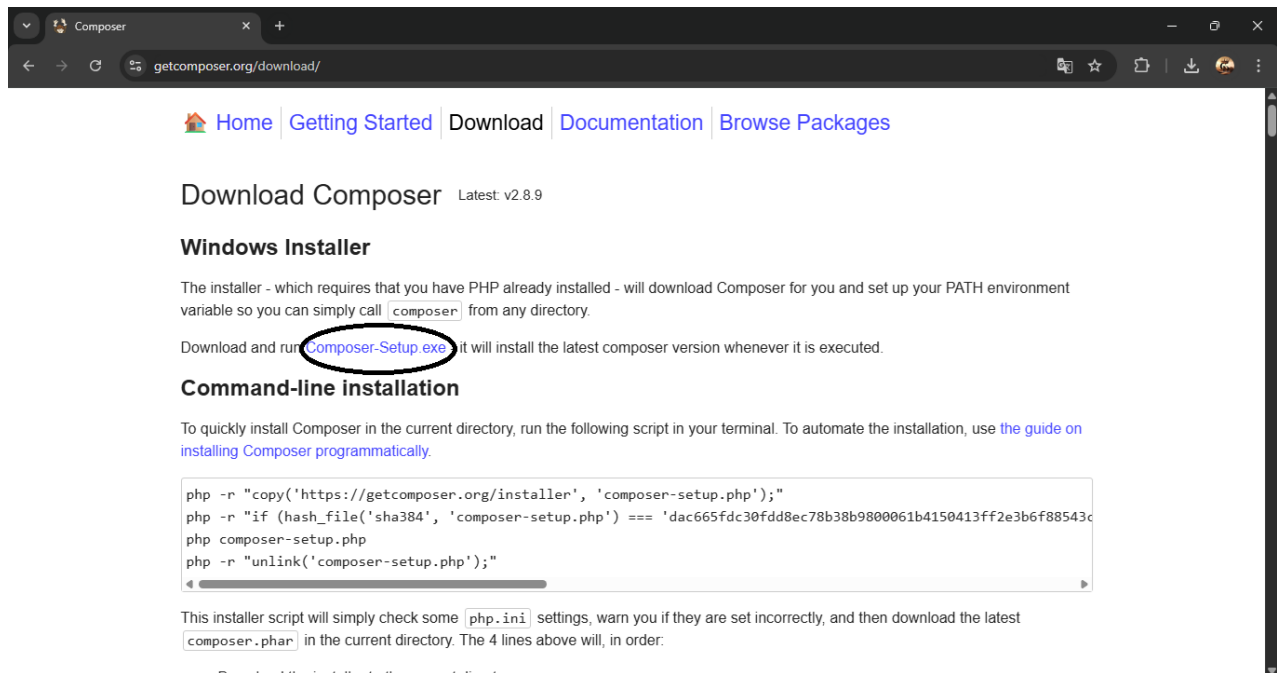
FIGURA 9 PAGINA WEB DE DESCARGA DE PHP



FUENTE: (PHP)

Ya luego de terminar la instalación de PHP, podemos continuar instalando Composer que es el gestor de paquetes de PHP en su última versión desde el siguiente enlace que es completamente gratis y desde el sitio oficial: <https://getcomposer.org/download/>.

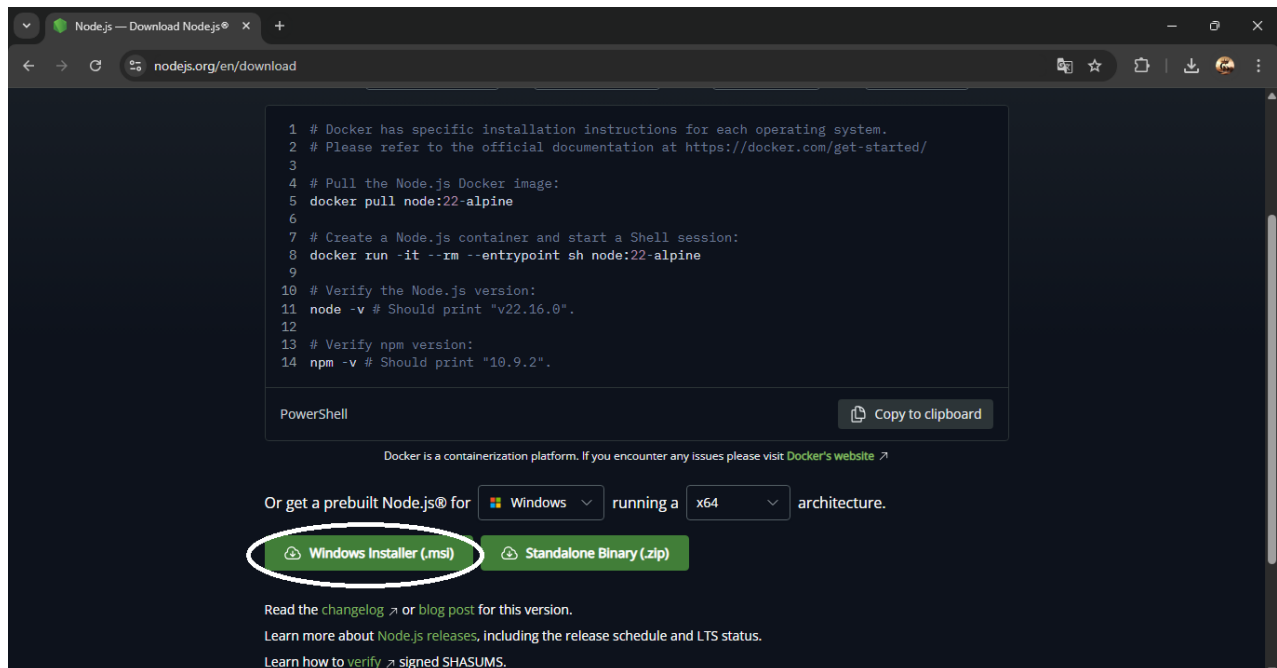
FIGURA 11 PAGINA WEB DE DESCARGA DE COMPOSER



FUENTE: (COMPOSER)

Ya que tenemos instalado COMPOSER podemos continuar instalando NODE.js que vendría siendo lo ultimo necesario para entorno web, desde el sitio web: <https://nodejs.org/en/download> que es completamente gratis.

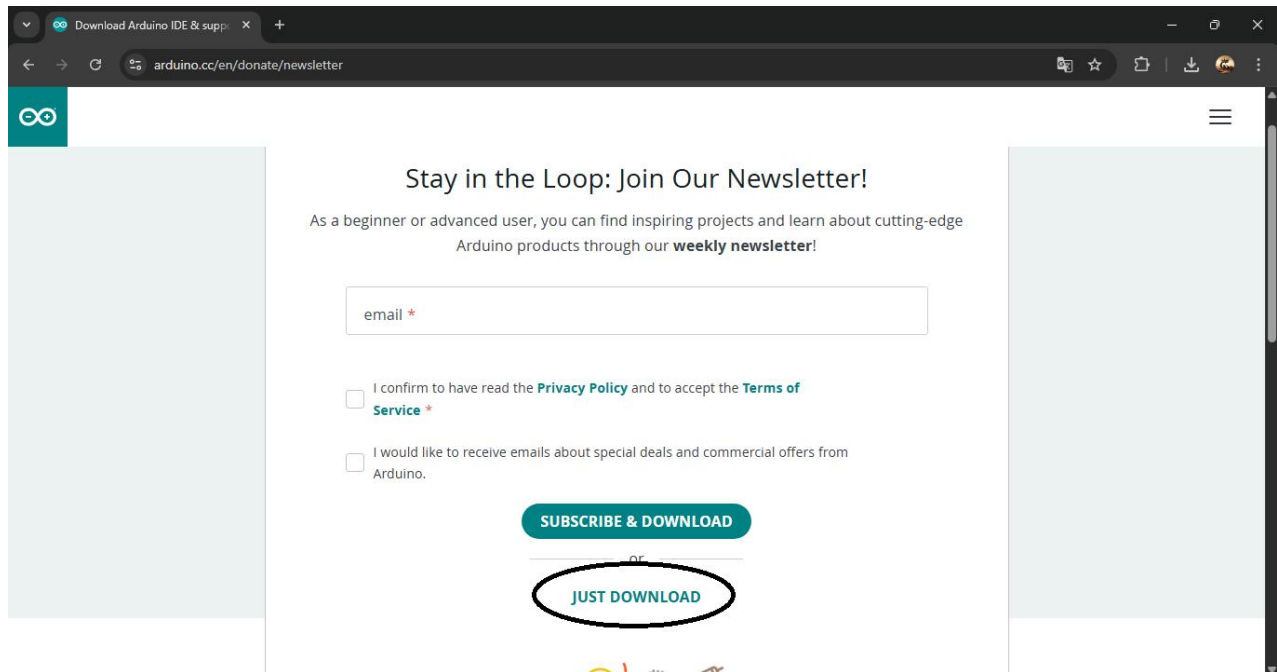
FIGURA 12 PAGINA WEB DE DESCARGA DE NODE.JS



FUENTE: (NODE.JS)

Cuando tengamos instalado Node.js lo último que nos faltaría instalar sería Arduino, que lo encontraríamos completamente gratis desde su sitio oficial: <https://www.arduino.cc/en/donate/newsletter>.

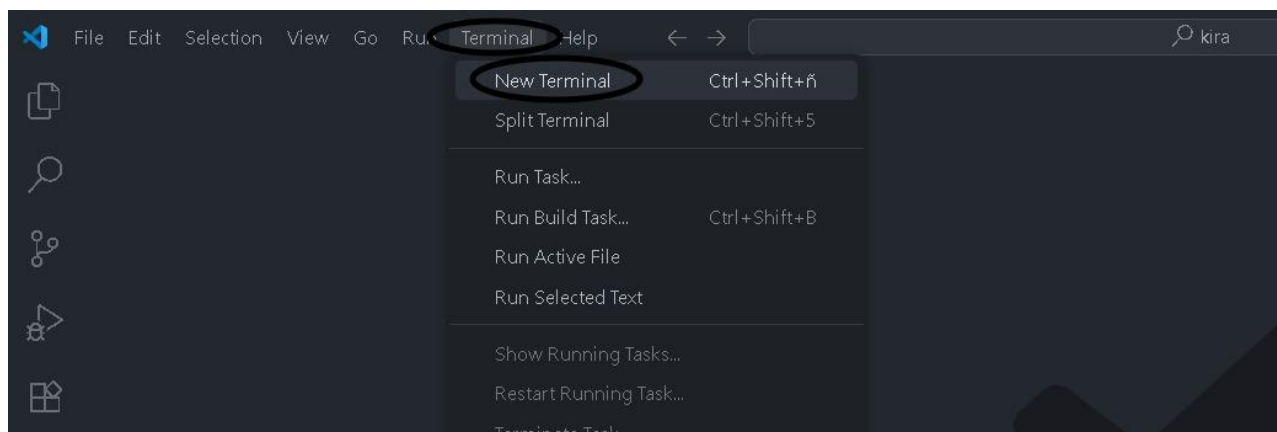
FIGURA 13 PAGINA WEB DE DESCARGA DE ARDUINO



FUENTE: (ARDUINO)

Ya que contamos con todo instalado debemos para poder encender los servicios, hacerlo de la siguiente manera, primero vamos a VisualStudioCode, vamos a Terminal, luego a New Terminal, y para encender React, debemos colocar el comando: `npm run dev` y nos da el test, luego para encender la APP ingresamos el comando `npx expo start`, para que levante, de forma local, y luego en Laragon, debemos dar click y cuando se despliegue el menú vamos a Apache, luego a SSL y le damos click en Enabled, ya con esto podrías comenzar a manipular el software.

FIGURA 14 Ingreso a Terminal de VisualStudioCode



FUENTE: Por los autores

FIGURA 15 Activación de React

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS
/c/laragon/www/kira (main)
npm run dev
> kira@0.0.0 dev
> vite

VITE v6.0.3 ready in 754 ms
→ Network: https://kira.test:5173/
→ press h to show help
```

FUENTE: Por los autores

FIGURA 16 Levantamiento local de servicio

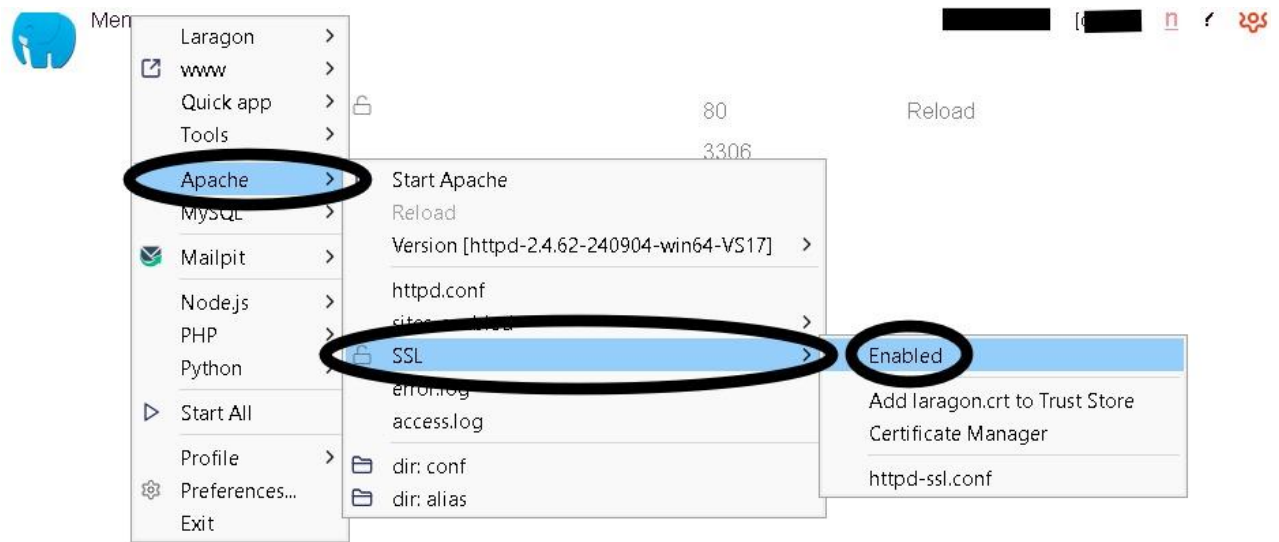
```
/c/laragon/www/kiraApp (master)
$ npx expo start
Starting project at C:\laragon\www\kiraApp
Starting Metro Bundler

[QR Code with a large black X over it]

> Metro waiting on
> Scan the QR code above to open the project in a development build.
Learn more: https://expo.fyi/start
> Web is waiting on http://localhost:8081
> Using development build
> Press s | switch to Expo Go
> Press a | open Android
> Press w | open web
```

FUENTE: Por los autores

FIGURA 17 Activación de Laravel desde Laragon



Laragon - The Dev Environment for Entrepreneurs

FUENTE: Por los autores

9. REQUERIMIENTOS DE SOFTWARE

En esta sección se detallarán los requisitos mínimos del sistema para poder ejecutar los aplicativos usados para modificar el software KIRA.

9.1 Requisitos mínimos computador

- Sistema operativo: Windows 10/macOS 10.15+/Ubuntu 20.04+
- Procesador: Intel Core i5 (o equivalente AMD)
- Memoria RAM: 8GB
- Memoria en disco: 10GB libres
- Conexión a internet: Requerida para instalación de paquetes

10. WEBGRAFIA

FUENTE (LARAGON): <https://laragon.org/download/>

FUENTE (VISUAL STUDIO CODE): <https://code.visualstudio.com/download>

FUENTE (GITBASH): <https://git-scm.com/downloads/win>

FUENTE (GITHUB): <https://desktop.github.com/download/>

FUENTE (BOOTSTRAP): <https://getbootstrap.com/docs/5.3/getting-started/download/>

FUENTE (PHP): <https://www.php.net/downloads>

FUENTE (COMPOSER): <https://getcomposer.org/download/>

FUENTE (NODE.JS): <https://nodejs.org/en/download>

FUENTE (ARDUINO): <https://www.arduino.cc/en/donate/newsletter>