Creación de proyecto con PHP sin framework.

1. Crear carpeta del proyecto
2. Iniciar composer

$: composer init

* Definir el nombre del proyecto: “sige/nombre-proyecto”
* Descripción: ‘”……”
* Autor: “blsisitemas”
* Mínimum estability: “stable” (valor por defecto)

Al momento de instalar dependencias, composer comprueba la estabilidad del paquete para ver si instalarlo o no según indiques en este paso.

* Tipo de paquete: “project”
* Lincense: “propietary”
* Definir dependencias: “no”
* Mapear src con autoload: si – dar enter
* Confirmamos generación del archivo. “yes”

1. Carpetas básicas a este punto

* Config
* Templates
* Src
* Vendor

1. Crear archivo de setup

Creamos archivo llamado “Bootstrap.php” Para la configuración o para establecer lo necesario para el script.

1. Autolaod

Carga las dependencias automáticamente en nuestros archivos de php

1. Instalar “simfony dotenv”

$: composer require symfony/dotenv

* Sirve para cargar valor de configuración de archivos .ENV

[GitHub - symfony/dotenv: Registers environment variables from a .env file](https://github.com/symfony/dotenv)

1. Estructurar SRC

* Carpeta APPS

Es donde se agregan los controladores de la página o aplicativo

* + Admin
  + Api
  + Blog
* Carpeta Applications

Donde se almacena los casos de uso, las funciones del aplicativo.

* Carpeta Domain

Conceptos del contexto del aplicativo, como las clases y los objetos relacionados a la lógica del negocio.

* Carpeta Infraestructure

Las implementaciones concretas. La conexión de base de datos o implementación de las interfaces, etc.

1. Otras carpetas

* Carpeta Templates

Se estructura como APPS. Se trata de las vistas que se muestran a los usuarios…

* + Admin
  + Api
  + Blog

\*Explicación del procedimiento y organización de las carpetas\*

En la creación de un caso de uso o funcionalidad del aplicativo se crea la función del controlador dentro de “APPLICATION”, esta clase hará uso de interfaces y de modelos que serán creados dentro de la carpeta “DOMAIN”

Para el ejemplo: si queremos crear una funcionalidad que registre la ubicación de un dispositivo al conectarse al aplicativo, crearemos el controlador “RegisterLocationAction” dentro de APPLICATION, este hará uso de una función “\_invoke”, de un validador de los datos que recibirá y hará uso de un repositorio bajo una interfaz.

* El validador puede utilizar una expection creada en DOMAIN que puede llamarse “MissingValueException”.
* La interfaz será creada también en DOMAIN y sera “LocationRepositoryInterface".
* El objeto o clase Location será creado también en DOMAIN.

En si la clase del repositorio, la cual es recibida en el contralador para el manejo del objeto Location se creara en INFRAESTRUCTURE el cual contendrá aquella conexión a base de datos y el procesamiento de la información ya en contacto con la BD