# Instalación yii

Aplicación Base en Git

1. Entrar en GitLab

2. Abre xamp y enciende Apache y Mysql

3. Ir a Htdocs que se encuentra en c:/—>xamp—>htdocs

4. Crear un proyecto en GitLab con el nombre de "My awesome project" … Copiar el html .

5. Dentro de la carpeta htdocs, Hacer "gitbash here". en consola

6. Ir a google Create yii2 project basic.

   Copiar lo que pone en la captura de abajo(en la web vendrá con dos basic al final. (quitar el último)) y en la consola  "pegar" (espacio) y poner punto  y dar intro.

## Installing Yii

With Composer installed, you can install Yii application template by running the followi
folder:

```
composer create-project --prefer-dist yiisoft/yii2-app-basic
```

7. Me preguntará si confio en composer y tecleare la "y" y daré intro.

8. git init (intro)

9. git remote add origin +(el https de gitlab del repo)

10. git add .

11. git commit -m "proyecto instalado"

12. git push origin master

13. Abrir netbeans/carpeta amarilla/php/php aplication with existing sources

14. En browser —> c:/xampp/htdocs/yii
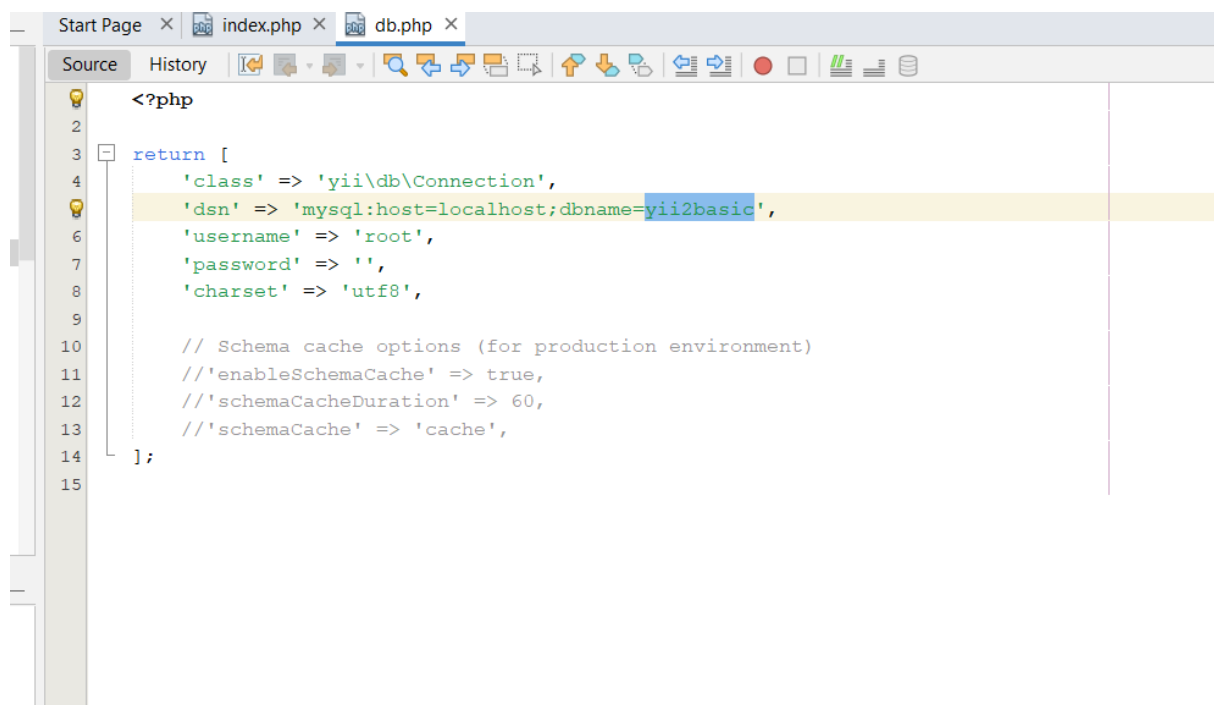
15. Poner lo de las fotos de abajo
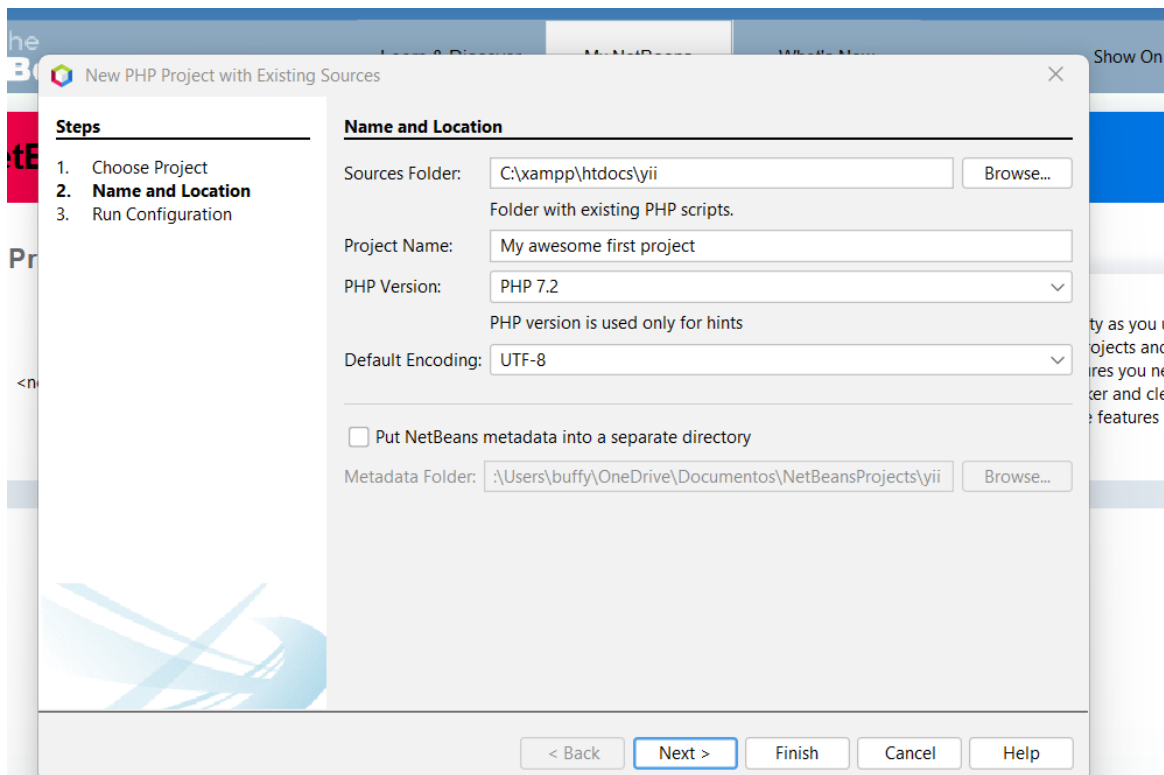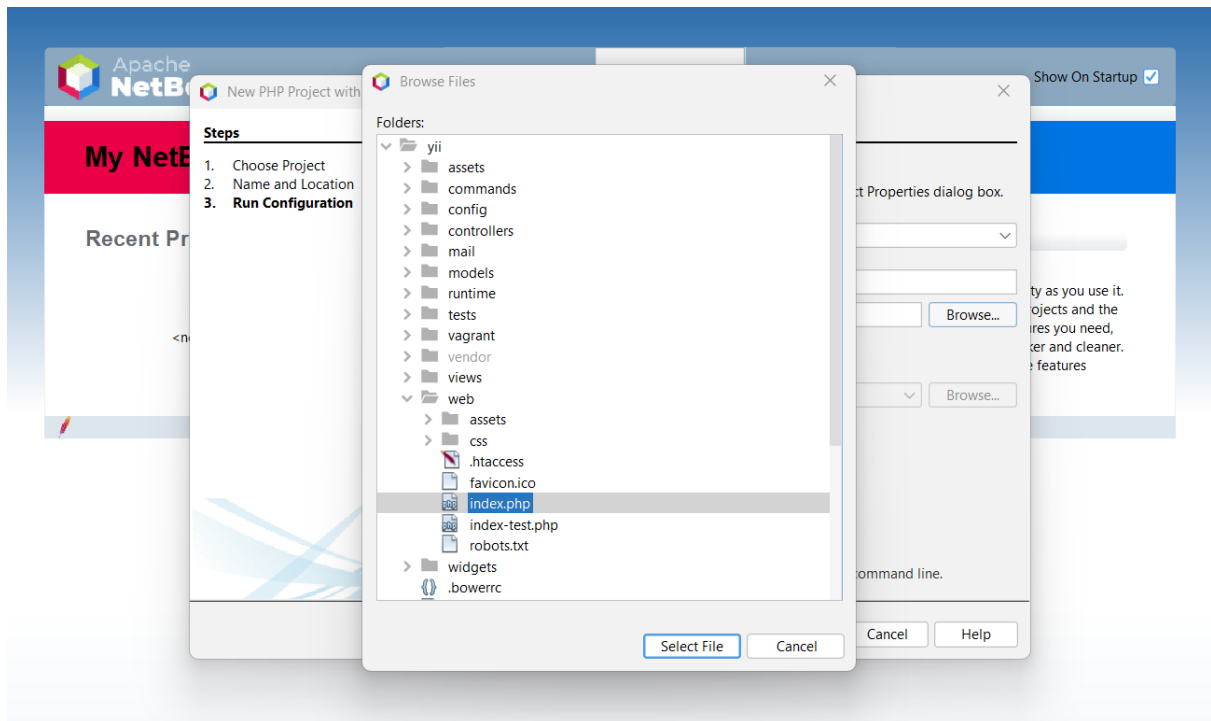
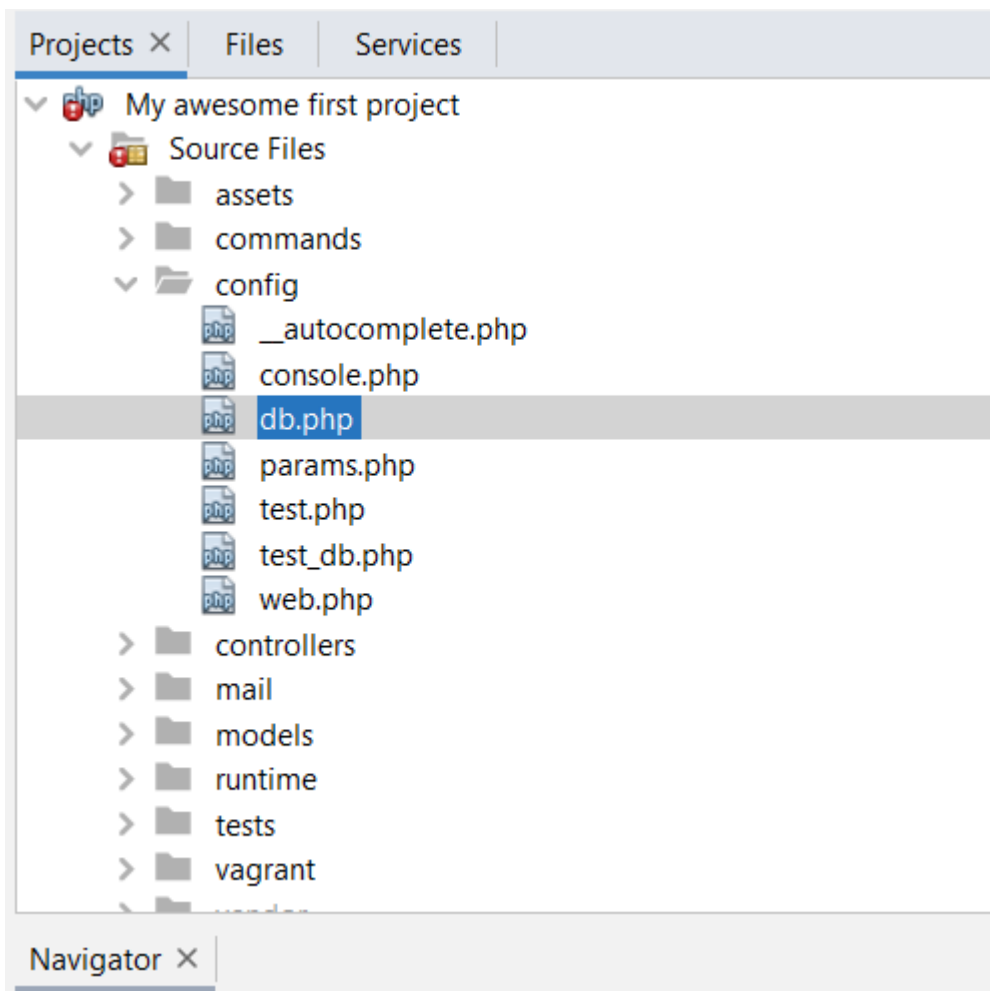16. Dar a finish

17. Bola del Mundo—>Seleccionar el google Chrome de la parte de abajo.

18. Flecha verde,  Hacer Click

19. Config—>db.php (fotos de abajo)

Projects × | Files | Services

- My awesome first project
  - Source Files
    - assets
    - commands
    - config
      - __autocomplete.php
      - console.php
      - db.php
      - params.php
      - test.php
      - test_db.php
      - web.php
    - controllers
    - mail
    - models
    - runtime
    - tests
    - vagrant

Navigator ×

Start Page × | index.php × | db.php ×

Source | History

```php
<?php

return [
    'class' => 'yii\db\Connection',
    'dsn' => 'mysql:host=localhost;dbname=yii2basic',
    'username' => 'root',
    'password' => '',
    'charset' => 'utf8',

    // Schema cache options (for production environment)
    //'enableSchemaCache' => true,
    //'schemaCacheDuration' => 60,
    //'schemaCache' => 'cache',
];
```

```php
<?php

return [
    'class' => 'yii\db\Connection',
    'dsn' => 'mysql:host=localhost;dbname=ciclistas',
    'username' => 'root',
    'password' => '',
    'charset' => 'utf8',

    // Schema cache options (for production environment)
    //'enableSchemaCache' => true,
    //'schemaCacheDuration' => 60,
    //'schemaCache' => 'cache',
];
```

20. Config—>web.php (fotos de abajo)

```php
<?php

$params = require __DIR__ . '/params.php';
$db = require __DIR__ . '/db.php';

$config = [
    'id' => 'ciclistas',
    'name' => 'Aplicación para la gestión de ciclistas',
    'language' => 'es-ES',
    'basePath' => dirname(__DIR__),
    'bootstrap' => ['log'],
    'aliases' => [
        '@bower' => '@vendor/bower-asset',
        '@npm'   => '@vendor/npm-asset',
    ],
    'components' => [
        'request' => [
            // !!! insert a secret key in the following (if it is empty) - this
            'cookieValidationKey' => 'E4bT-TxwkSXOvzAOYQJCpWY0xYdsyWR2',
        ],
        'cache' => [
            'class' => 'yii\caching\FileCache',
        ],
```

```
45                  ],
46              ],
47              'db' => $db,
48      /*
49              'urlManager' => [
50                  'enablePrettyUrl' => true,
51                  'showScriptName' => false,
52                  'rules' => [
53                  ],
54              ],
55      */
56          ],
57          'params' => $params,
58      ];
59
60  if (YII_ENV_DEV) {
61      // configuration adjustments for 'dev' environment
62      $config['bootstrap'][] = 'debug';
63      $config['modules']['debug'] = [
```

Source    History

```
37              ],
38              'log' => [
39                  'traceLevel' => YII_DEBUG ? 3 : 0,
40                  'targets' => [
41                      [
42                          'class' => 'yii\log\FileTarget',
43                          'levels' => ['error', 'warning'],
44                      ],
45                  ],
46              ],
47              'db' => $db,
48
49              'urlManager' => [
50                  'enablePrettyUrl' => true,
51                  'showScriptName' => true,
52                  'rules' => [
53                  ],
54              ],
55
56          ],
57          'params' => $params,
58      ];
59
60  if (YII_ENV_DEV) {
61      // configuration adjustments for 'dev' environment
62      $config['bootstrap'][] = 'debug';
63      $config['modules']['debug'] = [
64          'class' => 'yii\debug\Module',
65          // uncomment the following to add your IP if you are not connecting fro
```

21. Abrir consola —> git add .

22. git commit -m "aplicacion configurada"

23. git push origin master

24. En el navegador poner lo de la foto de abajo:
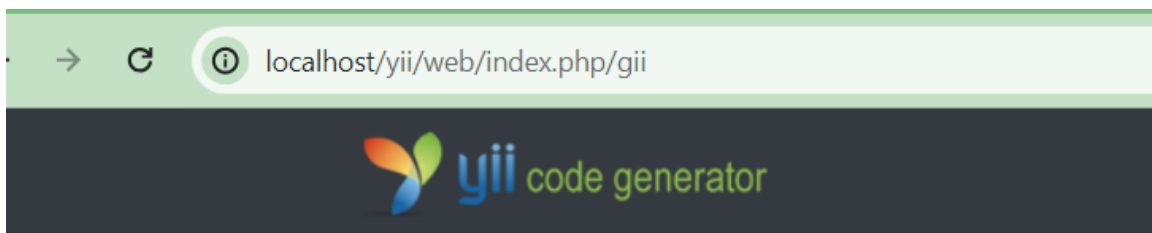


25. Start model generator/table



26. Preview—>generate

27. Crud generator

# CRUD Generator

This generator generates a controller and views tha
model.

## Model Class

app\models\Ciclista

---

# CRUD Generator

This generator generates a controller and views that implement CRUD (Create, Read, Update, Delete)
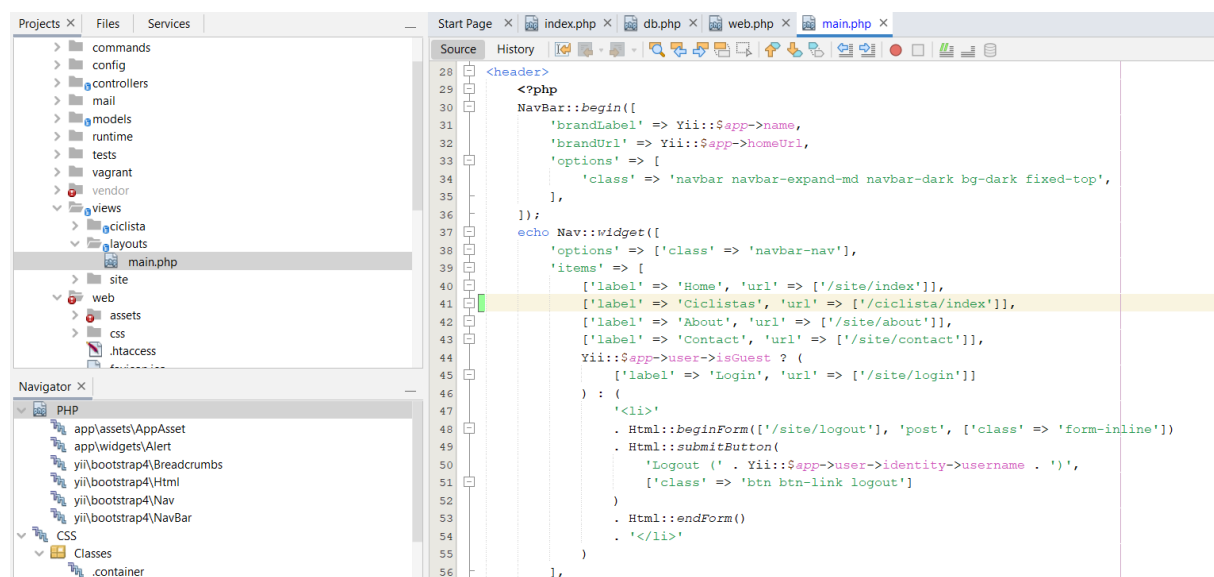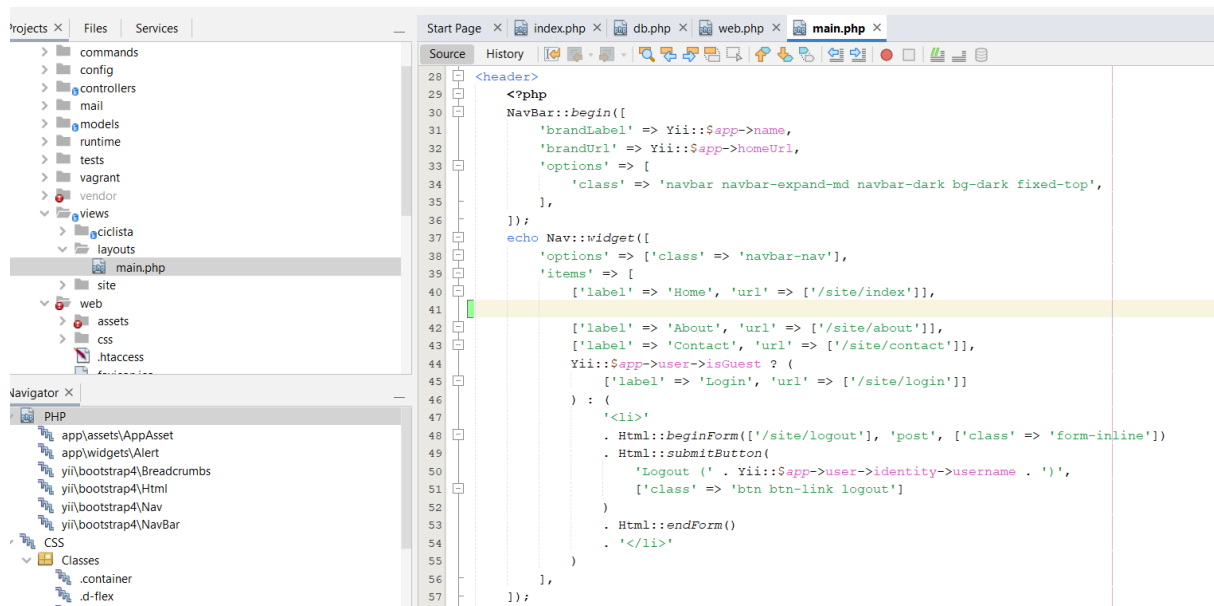model.

## Model Class

app\models\Ciclista

## Search Model Class

## Controller Class

app\controllers\CiclistaController

28. Modificar main de la vista

29. `git add .`

30. `git commit -m "modelo y crud de ciclista creado + menu añadido"`

31. `git push origin master`