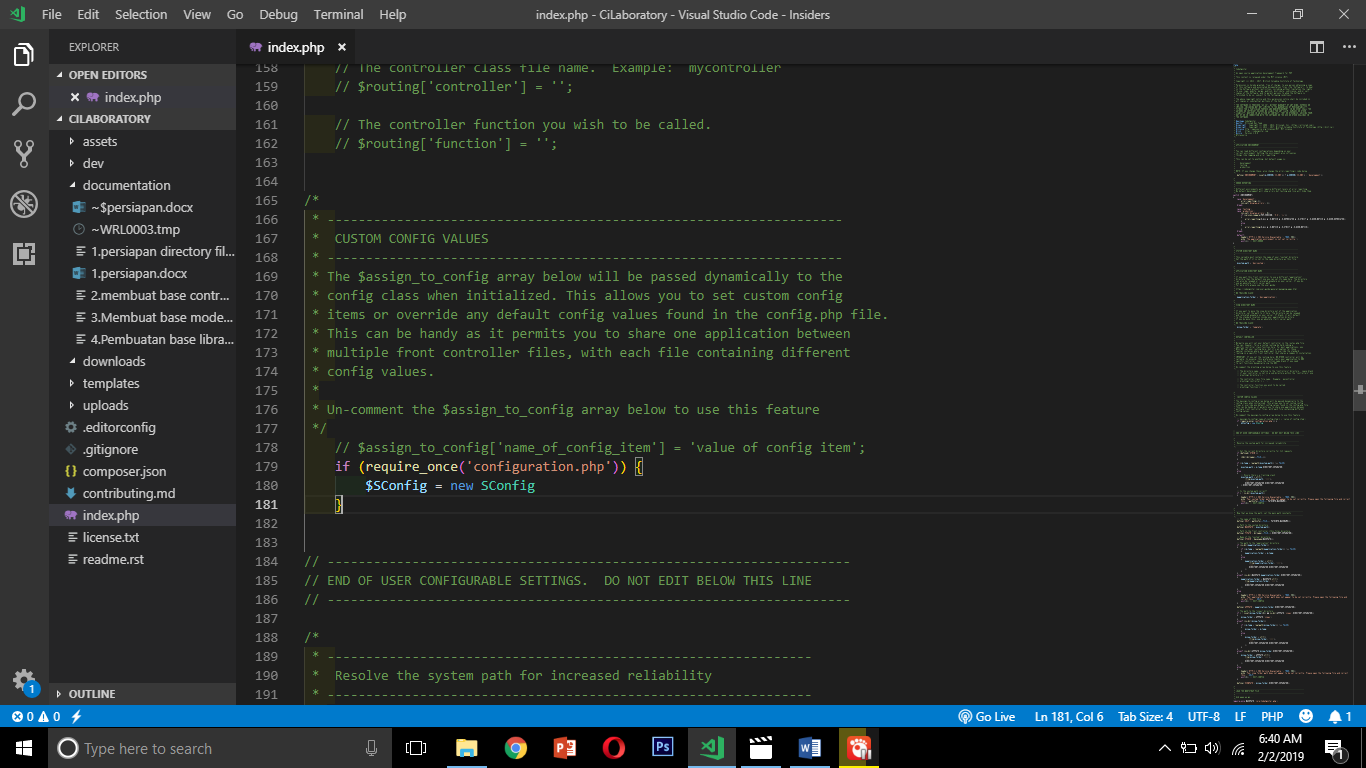
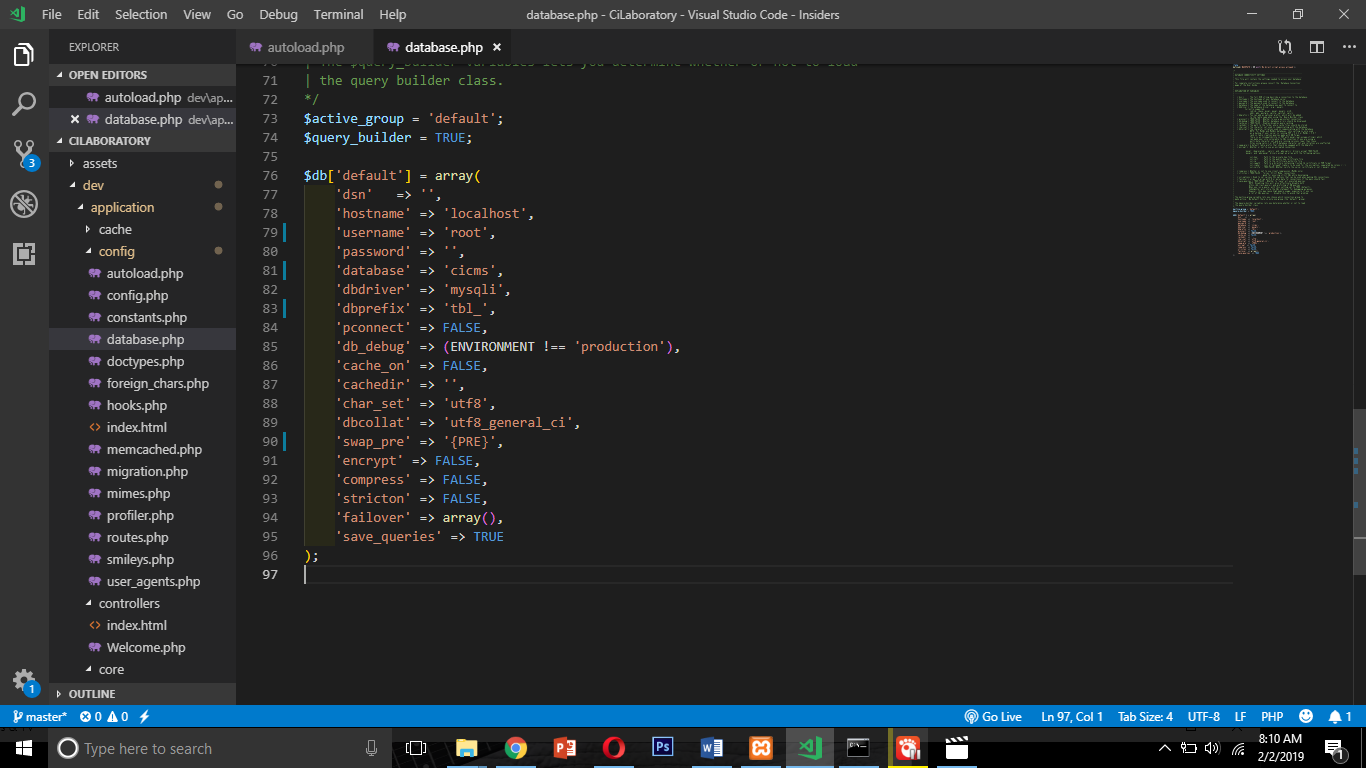
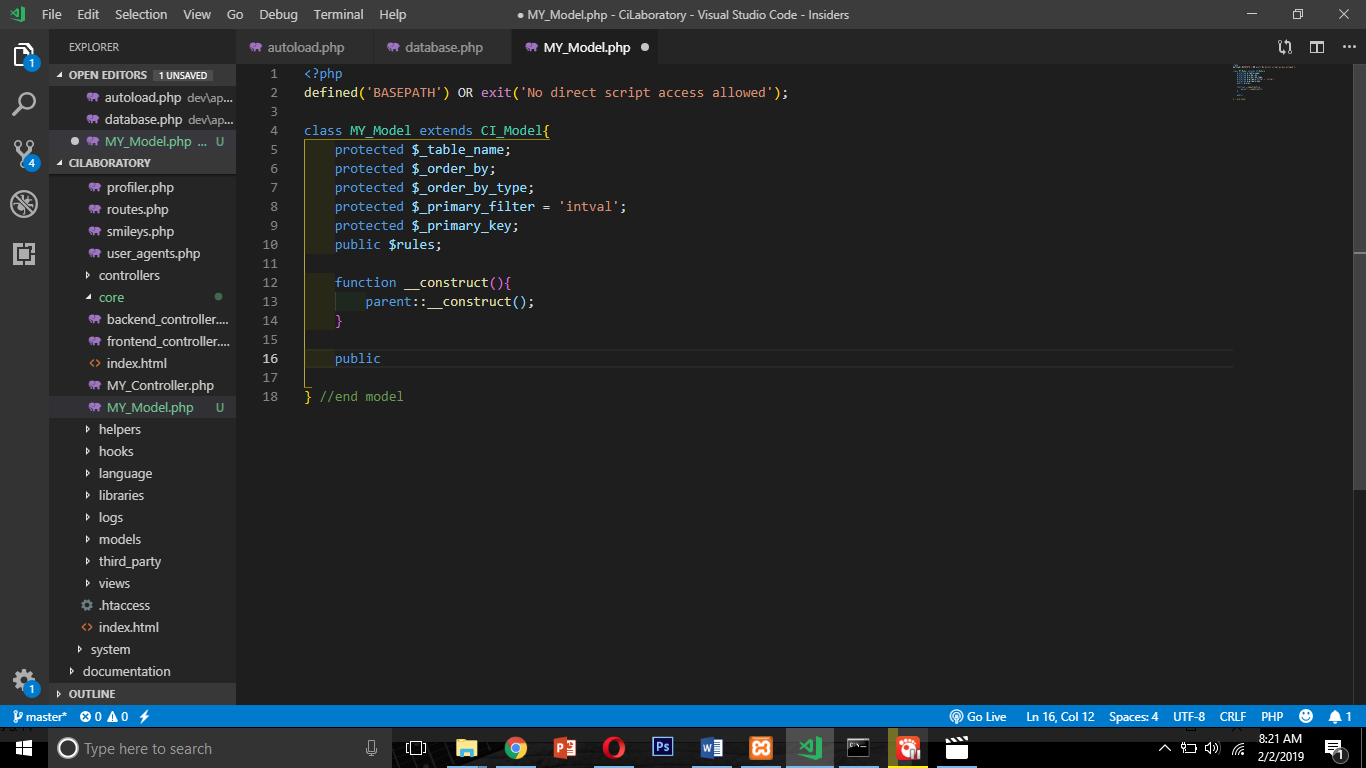
1. **PERSIAPAN DIRECTORY FILE UNTUK CMS.**
   1. Kelola database terlebih dahulu.
   2. Download dan extract library codeigniter pada local server(c:/xampp/htdocs).
   3. Buat folder dev, templates, uploads, downloads dan assets.
   4. Pindahkan folder application dan system kedalam folder dev.
   5. Ubah index.php pada $sistem\_path : dev/system dan $application\_folder : dev/application.
   6. Pindahkan isi file pada folder (dev/application/view/) kedalam folder templates.
   7. Ubah index.php pada $view\_folder : templates.
   8. Buath file configuration.php dan buat class SConfig
   9. Pada index.php buat fungsi seperti berikut pada bagian custom.



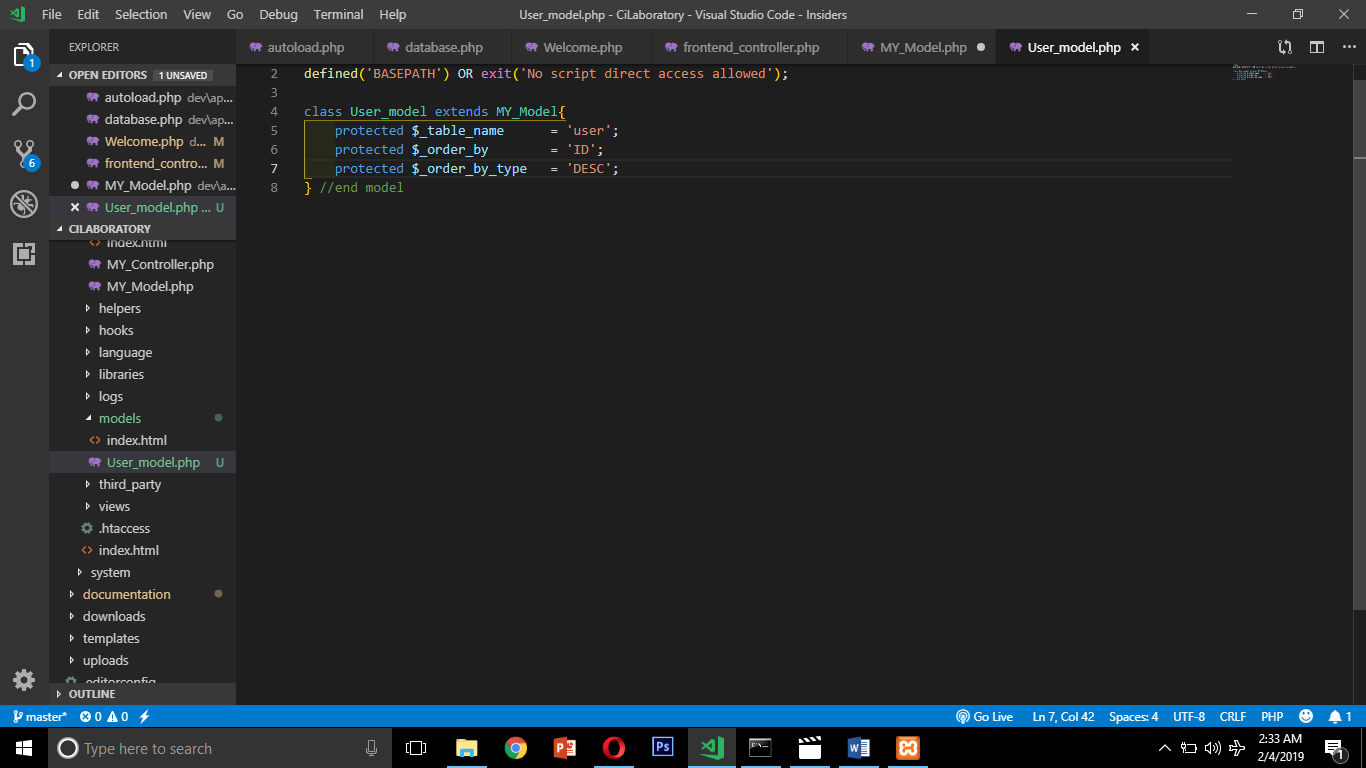
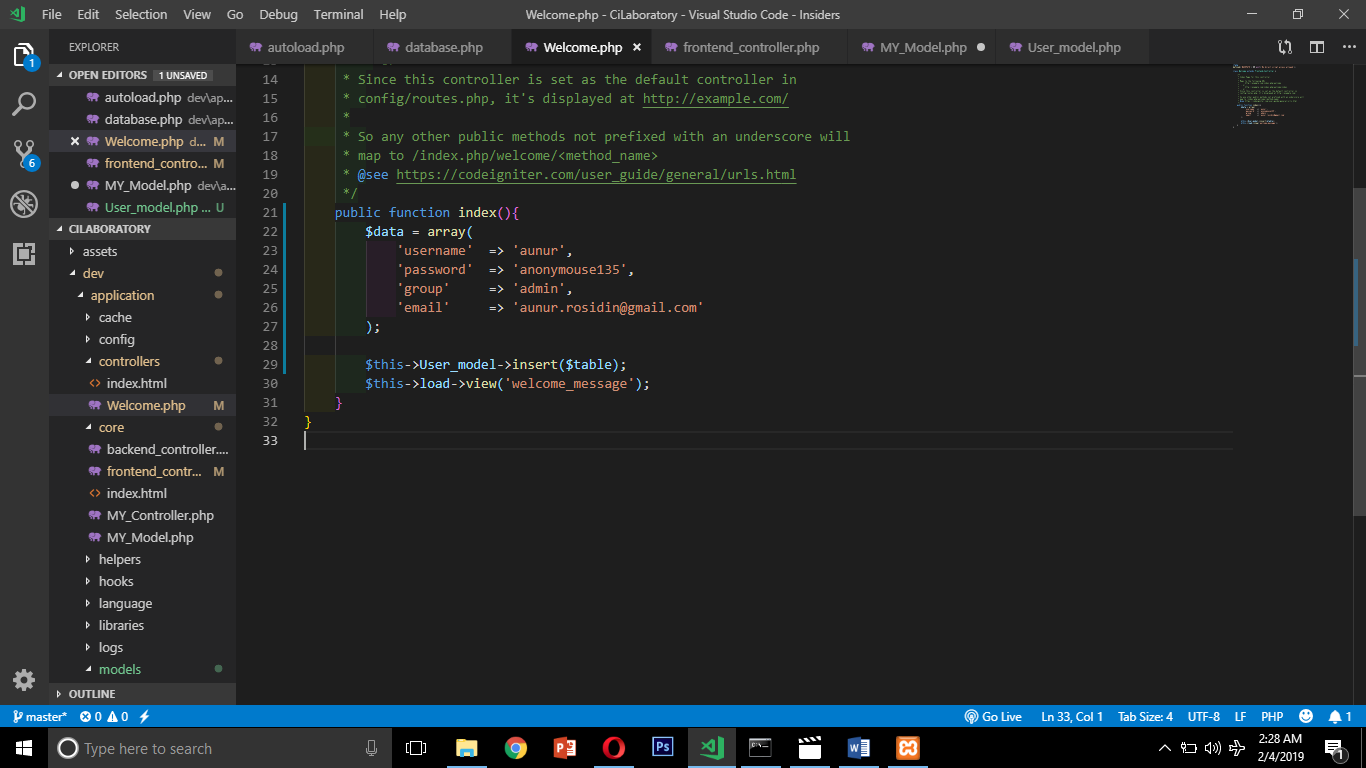
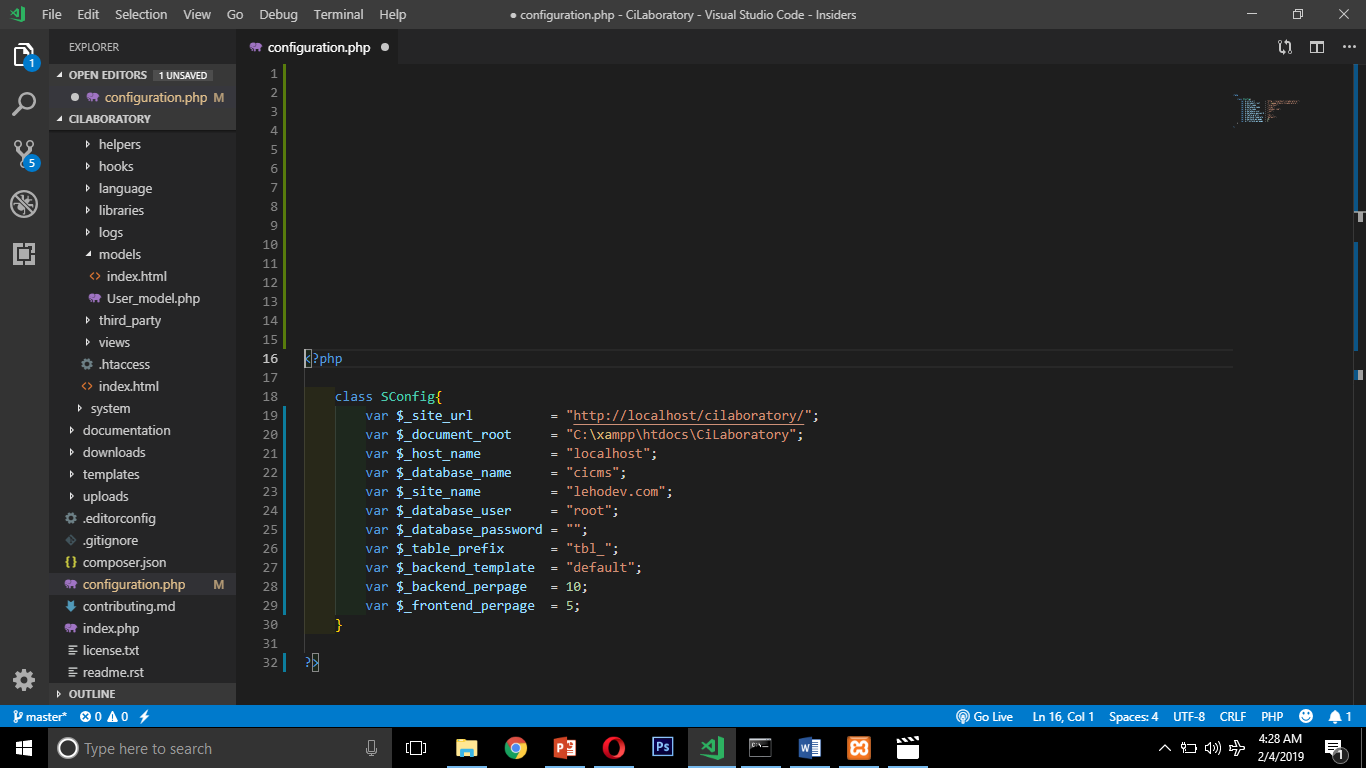
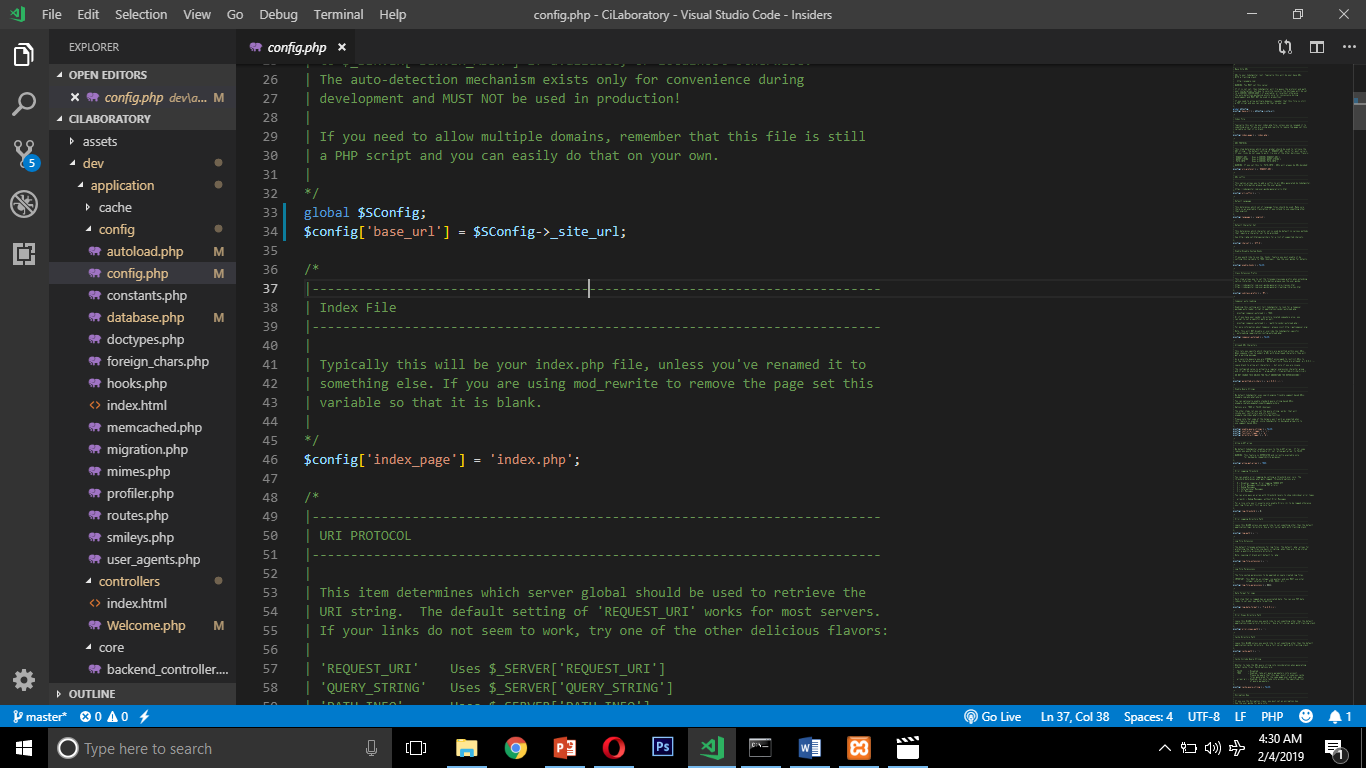
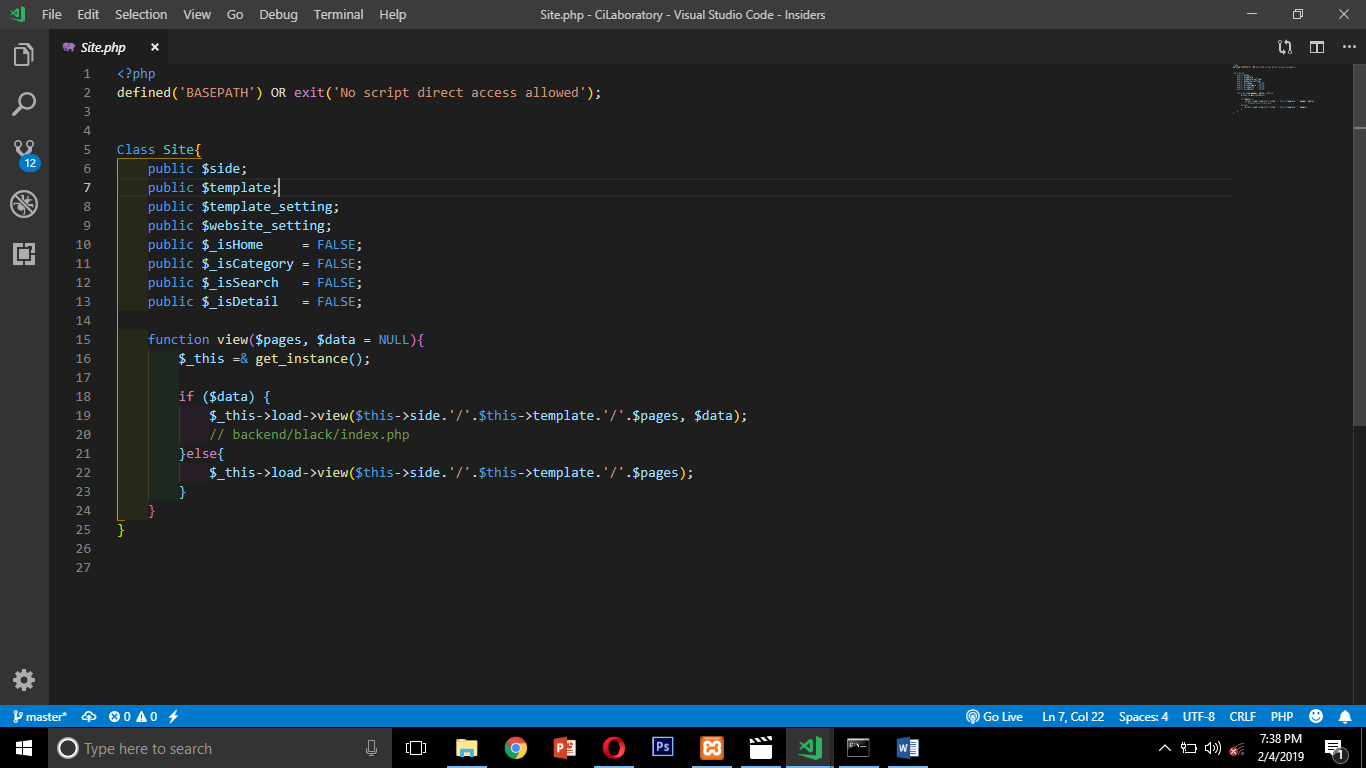
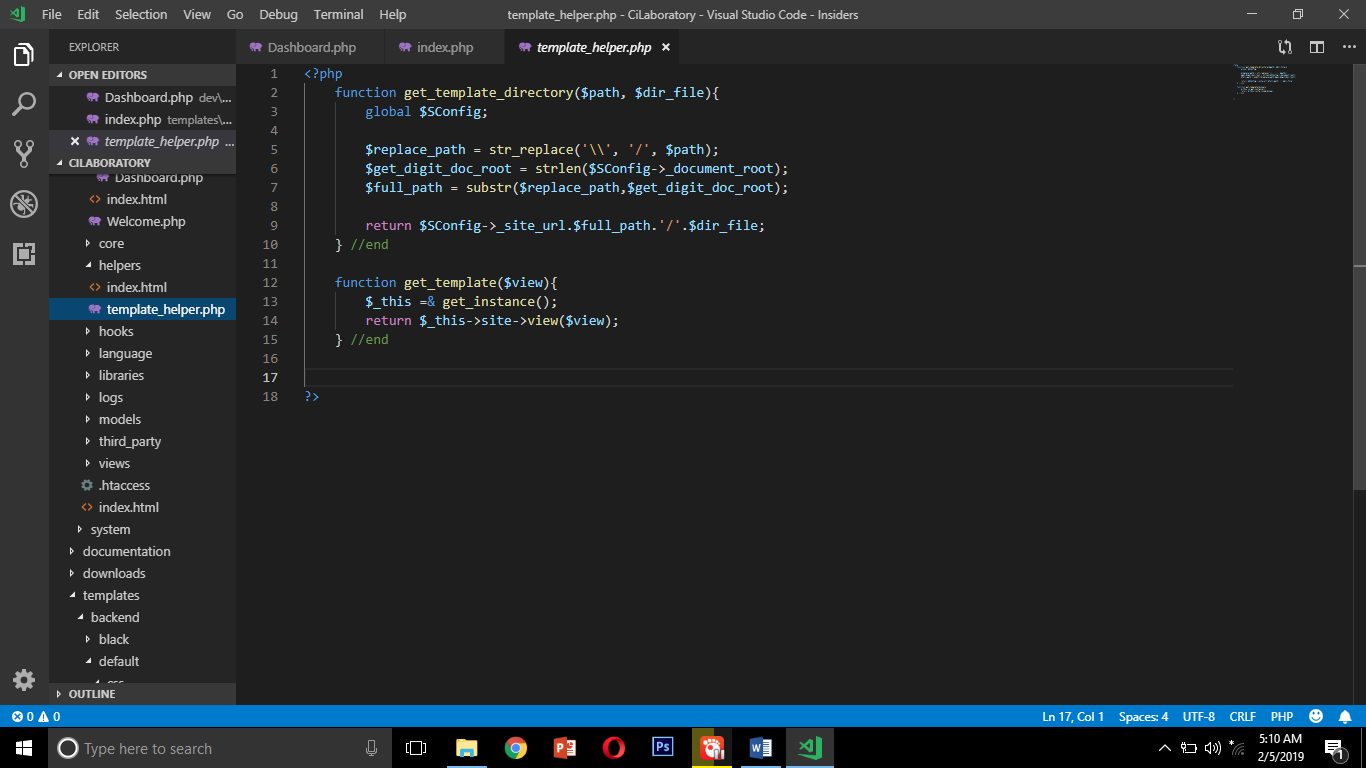
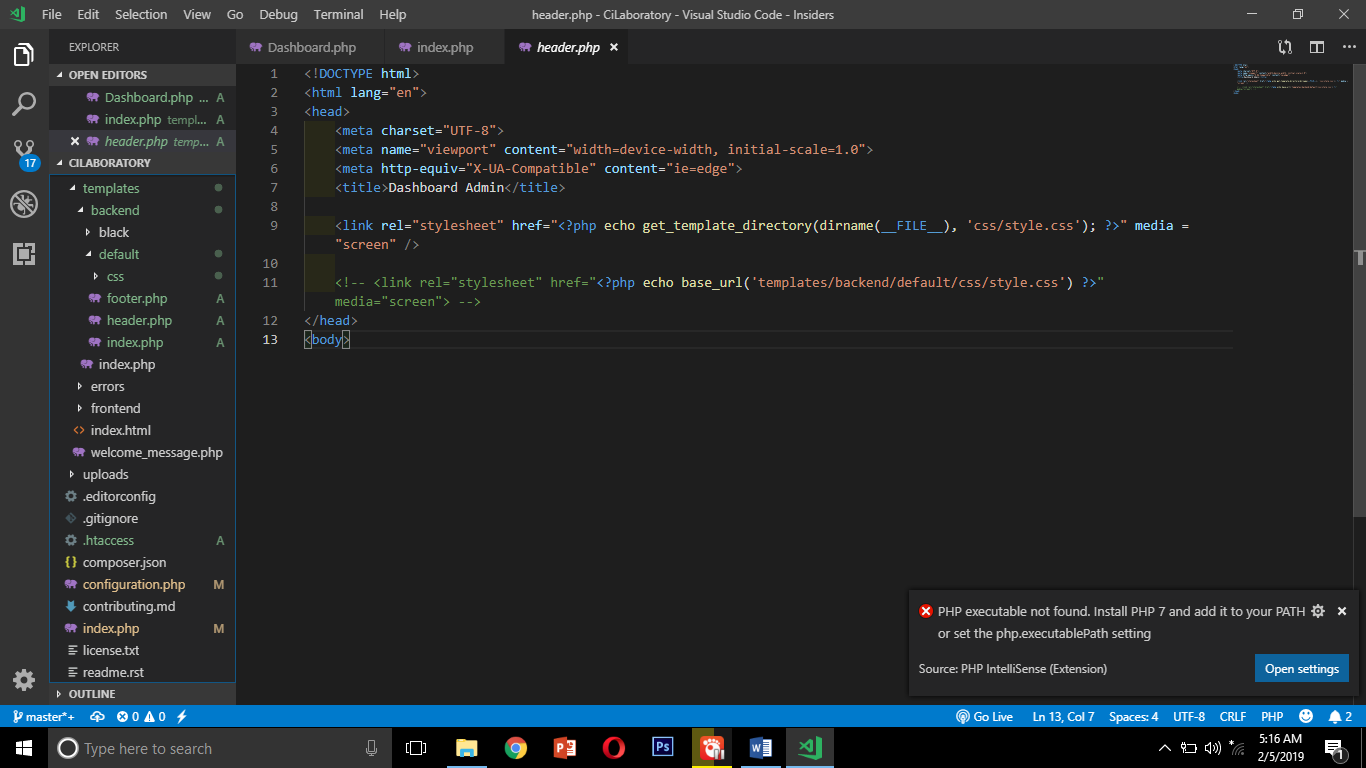
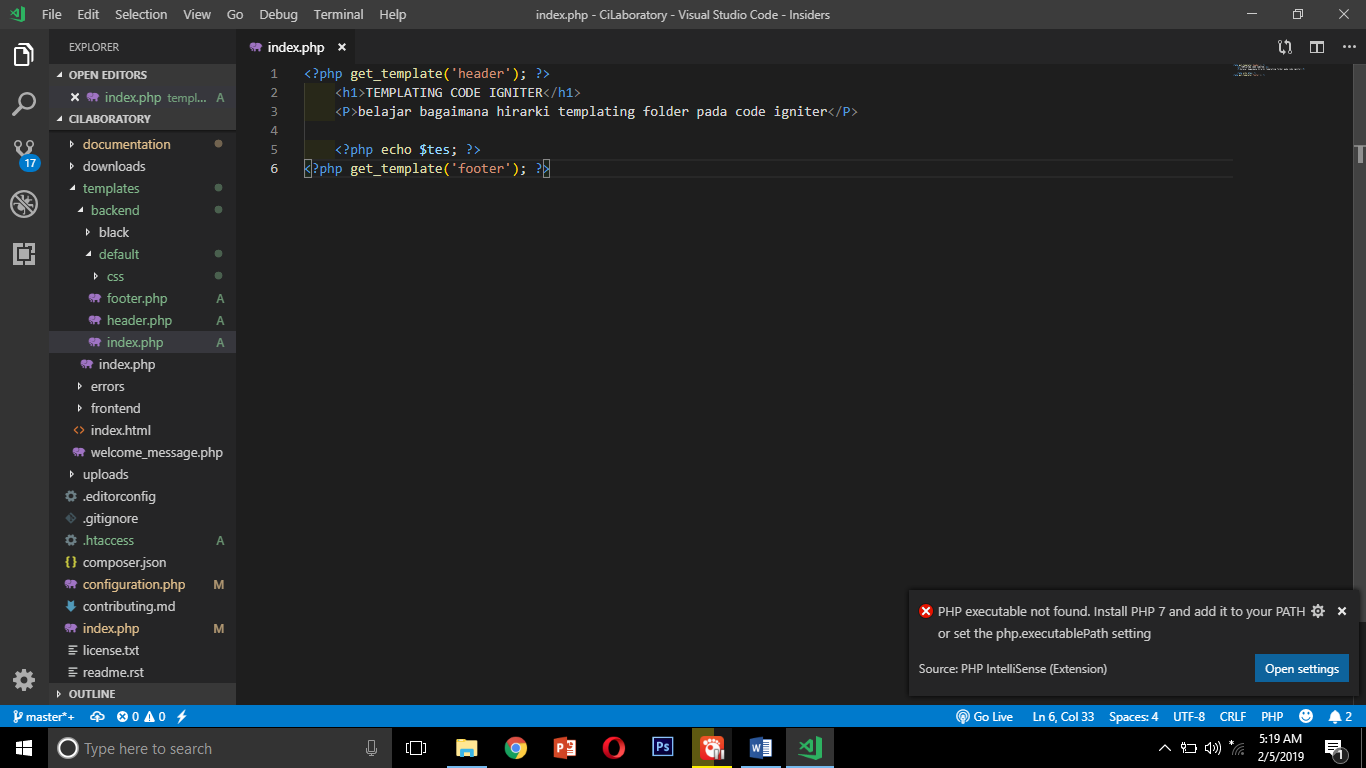
1. **MEMBUAT BASE CONTROLLER, FRONTEND CONTROLLER DAN BACKAND CONTROLLER**
   1. Buat file MY\_Controller.php pada folder (dev/application/core)
   2. Definisikan class MY\_Controller extends CI\_Controller, buat property $data dengan value array kosong dan buat function \_\_construct.
   3. Buat file backend\_controller.php “huruf kecil” pada folder(dev/application/config) dan definisikan class Backend\_Controller extends MY\_Controller serta buat function \_\_construct.
   4. Pada file config.php yang berada pada folder(dev/application/config) buat baris kode seperti berikut.
   5. gunakan file welcome.php yang berada pada folder(dev/application/controller) dan ganti extends kepada Backend\_Controller.
   6. Echo sebuah string pada backend\_controller pada function \_\_construct. Jika muncul pada browser maka perindukan selesai.
   7. Buat file frontend\_controller.php “huruf kecil” pada foler(dev/application/core) dan lakukan prosedur yang sama seperti pada backend\_controller.
   8. Untuk pengecekan perindukan selanjutnya, isikan property yang sebelumnya sudah disiapkan pada file MY\_Controller.php dengan mengisinya pada backend\_controller.php : $this->data[‘coba] = “coba” dan tampilkan pada file welcome.php : print\_r($this->data).
   9. Tahap selesai berishkan sisa-sisa script uji coba.
2. **MEMBUAT BASE MODEL**
   1. konfigurasi $autoload[‘libraries’]= ‘database’ pada autoload.php
   2. konfigurasi database pada database.php seperti berikut.



* 1. buat file MY\_Model.php pada folder core, buat class MY\_Model extends CI\_Model,
  2. Buat property sebagai berikut:



* 1. Buat method yang akan digunakan.

1. **IMPLEMENTASI BASE MODEL.**
   1. Buat file User\_model.php pada (dev/application/models) definisikan classnya dan extends kan pada MY\_Model.
   2. Definisikan property. 
   3. Load User\_model pada construct frontend\_controller.php
   4. Setelah itu implementasi method yang telah dibuat pada MY\_Model.php kepada welcome.php (dev/application/controllers). 
   5. Jalankan pada browser dan lihat pada localhost/phpMyAdmin pada table user.
   6. Coba beberapa fungsi seperti update dan delete.
2. **PEMBUATAN BASE LIBRARY**
   1. buat property pada configuration.php 
   2. panggil property yang telah didefinisikan pada configuration.php pada config.php pada (dev/application/config) 
   3. setelah itu lakukan $autoload[‘helper ‘] = ‘url ‘ pada autoload.php pada(dev/application/config).
   4. Lakukan hal yang sama pada konfigurasi database.php (dev/application/config)
   5. cek konfigurasi dengan cara echo base\_url() pada controller welcome.php
3. **MEMBUAT BASE LIBRARY SITE-VIEW**
   1. Buat folder frontend berisi file index.php pada folder template, lalu tulis frontend.
   2. Buat folder backend bersii file index.php pada folder template, lalu tulis backend.
   3. Buat file Site.php pada folder libraries lalu buat class, definisikan property dan buat methodnya.
   4. Pada MY\_Controller.php load library Site
   5. Pada frontend\_controller.php isi property yang sudah dibuat pada library site.php : $this->site->side = “frontend”;
   6. Tulis juga pada backend\_controller.
   7. Panggil method view pada controller welcome.php : $this->site->view(‘index’);
   8. Jalankan pada browser jika muncul kata frontend maka pembuatan site view sukses.
   9. Pada welcome controller ganti extends yang semula menginduk kepada frontend\_controller menjadi backend\_controller.php untuk mengecek fungsi sekalilagi.
   10. Buat file black berisi index.php pada folder (template/backend) dan tulis black template.
   11. Pada backend\_controller definisikan property template yang sudah dibuat pada library : $this->site->template = “black”. Dan jalankan ada browser sehingga muncul text sesuai template yang didefinsikan pada backend\_controller.php.
4. **PEMBUATAN BASE HELPER**
   1. Buat controller Dashboard.php defisinikan class dan extends ke Backend Controller. Lalu buat function index()
   2. Hilangkan index.php pada url pada config.php
   3. Copy file .htaccess dari projek yang telah dibuat sebelumnya.
   4. jalankan link “localhost/CiLaboratory/admin/dasboard”
   5. Buat folder default dan file index.php didalamya lalu Definisikan html.
   6. Buat helper template\_helper.html 
   7. Buat folder css pada folder (templates/backend) dan buat file style.css lalu isikan style untuk percobaan kepada index.php tadi.
   8. Buat file header.php pada folder (templates/default) dan panggil file style.css yang telah dibuat tadi. 
   9. Buat file footer.php pada folder (templates/default). Dan ambil bagian sisa yang berada pada index.php
   10. Setelah pada bagian index.php panggil header dan footer. Isikan konten di antara keduanya 
   11. Jalankan pada browser