



# **PERTEMUAN 5**

## **FLUTTER DAN API**



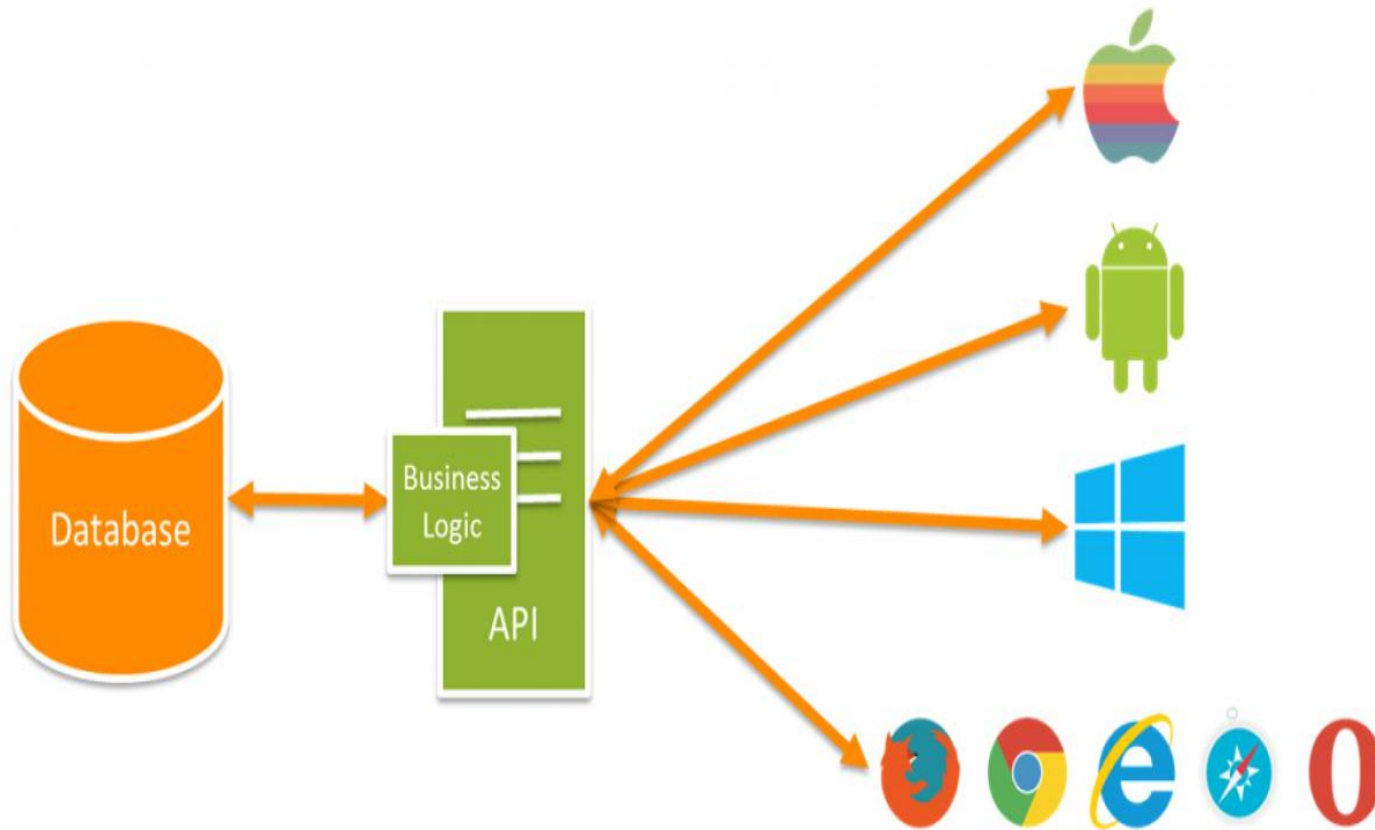
# 1. Membuat proyek flutter yang terhubung dengan API

## a. Apa itu API

API atau Application Programming Interface adalah sebuah interface yang dapat menghubungkan aplikasi satu dengan aplikasi lainnya. Jadi, API berperan sebagai perantara antar berbagai aplikasi berbeda, baik dalam satu platform yang sama atau lintas platform.

Perumpamaan yang bisa digunakan untuk menjelaskan API adalah seorang pelayan di restoran. Tugas pelayan tersebut adalah menghubungkan tamu restoran dengan juru masak. Tamu cukup memesan makanan sesuai daftar menu yang ada dan pelayan memberitahukannya ke juru masak. Nantinya, pelayan akan kembali ke tamu tadi dengan masakan yang sudah siap sesuai pesanan.

Itulah gambaran tugas dari API dalam pengembangan aplikasi.



Sumber : [codepolitan.com](http://codepolitan.com)

## **b. Arsitektur API**

Ada tiga arsitektur API yang sering digunakan oleh developer dalam pembangunan aplikasi. Arsitektur ini berkaitan pada bentuk data yang dikirim. Adapun Arsitektur API yang sering digunakan adalah

# 1. RPC

RPC merupakan teknologi untuk membuat komunikasi antara client side dan server side bisa dilakukan dengan konsep sederhana.

RPC memiliki dua jenis, yaitu XML-RPC dan JSON-RPC. Sesuai namanya, XML-RPC menggunakan format XML sebagai media perpindahan data, sedangkan JSON-RPC menggunakan JSON untuk perpindahan data.



## 2. SOAP

Arsitektur API lainnya adalah SOAP (Simple Object Access Protocol). Arsitektur ini menggunakan XML (Extensible Markup Language) yang memungkinkan semua data disimpan dalam dokumen.

### 3. REST

REST atau Representational State Transfer adalah arsitektur API yang cukup populer karena kemudahan penggunaannya. Tak perlu coding yang panjang untuk menggunakannya.

REST menggunakan JSON sebagai bentuk datanya sehingga lebih ringan. Performa aplikasi pun menjadi lebih baik.



## 2. Membuat proyek Toko API (Restful API)

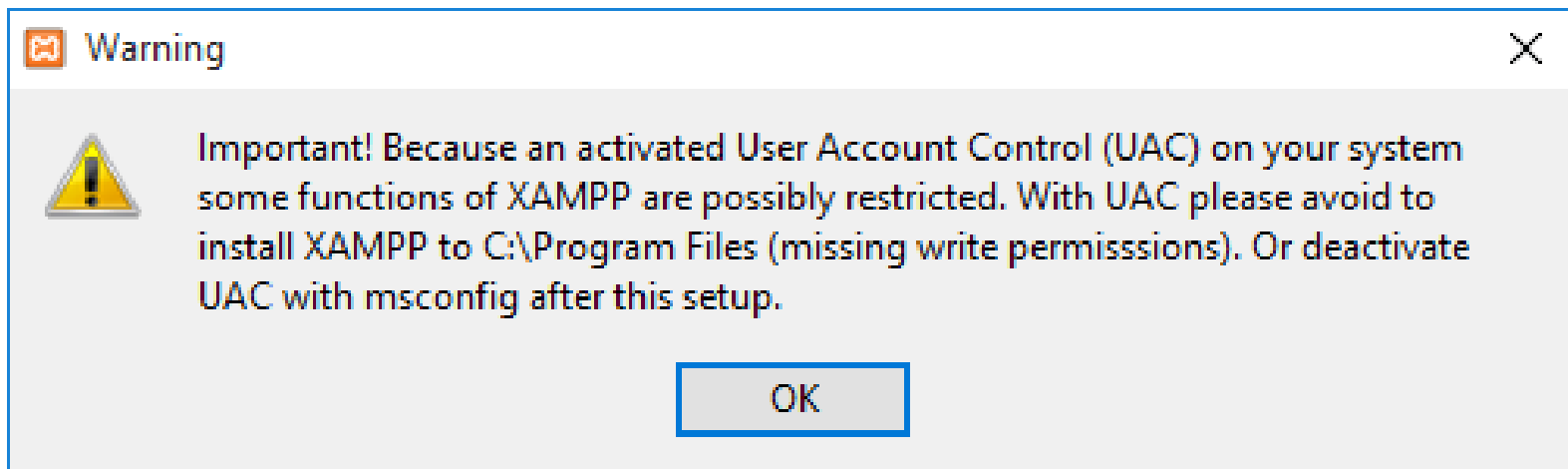
### a. Instalasi Apache, MySql dan PHP (XAMPP)

XAMPP adalah perangkat lunak untuk membuat komputer menjadi web server. XAMPP dapat di download melalui link url:

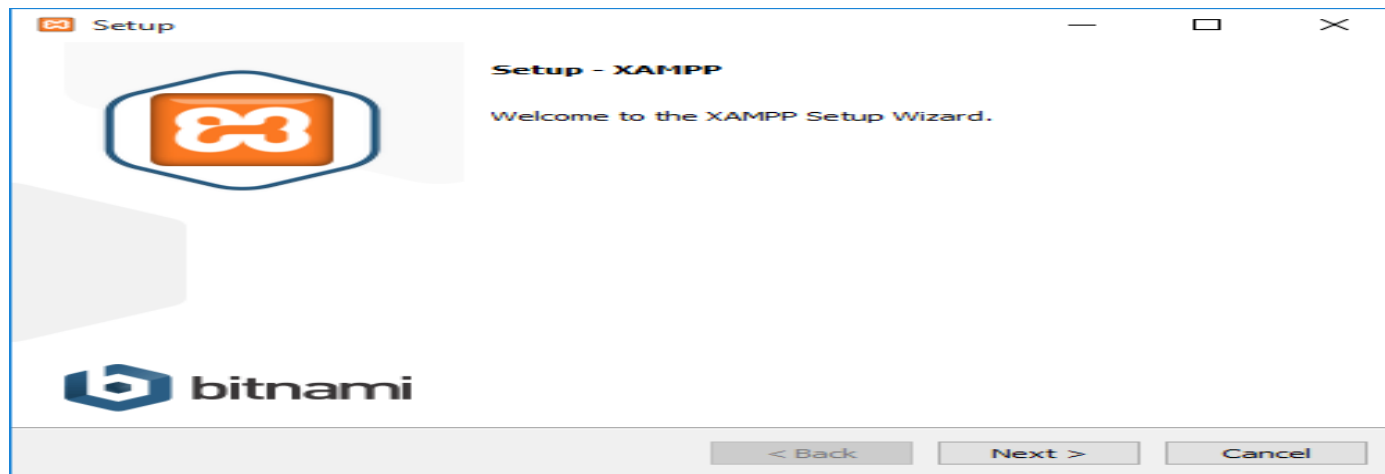
<https://www.apachefriends.org/download.html>

.

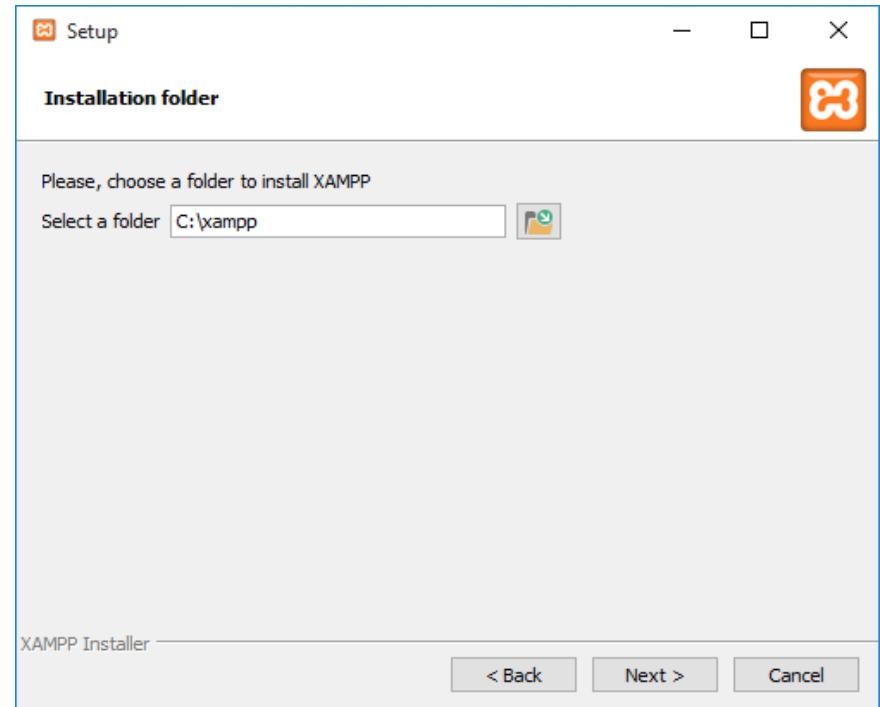
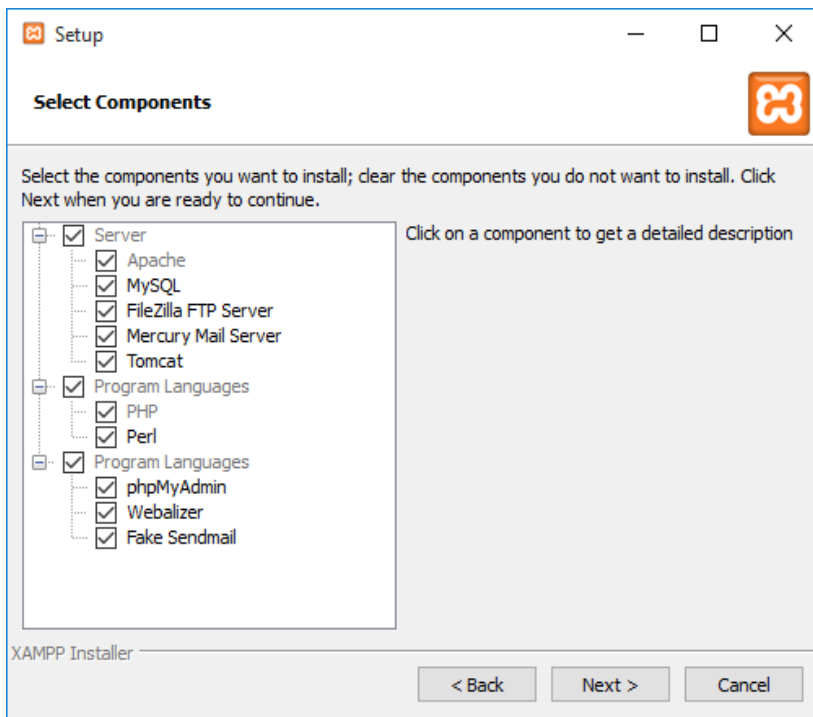
*Setelah itu, double klik file xampp yang baru di download. Jika muncul pesan error seperti gambar dibawah, abaikan saja dan lanjutkan klik tombol OK.*



Berikutnya akan muncul jendela **Setup-XAMPP**. Klik tombol **Next** untuk melanjutkan proses berikutnya.

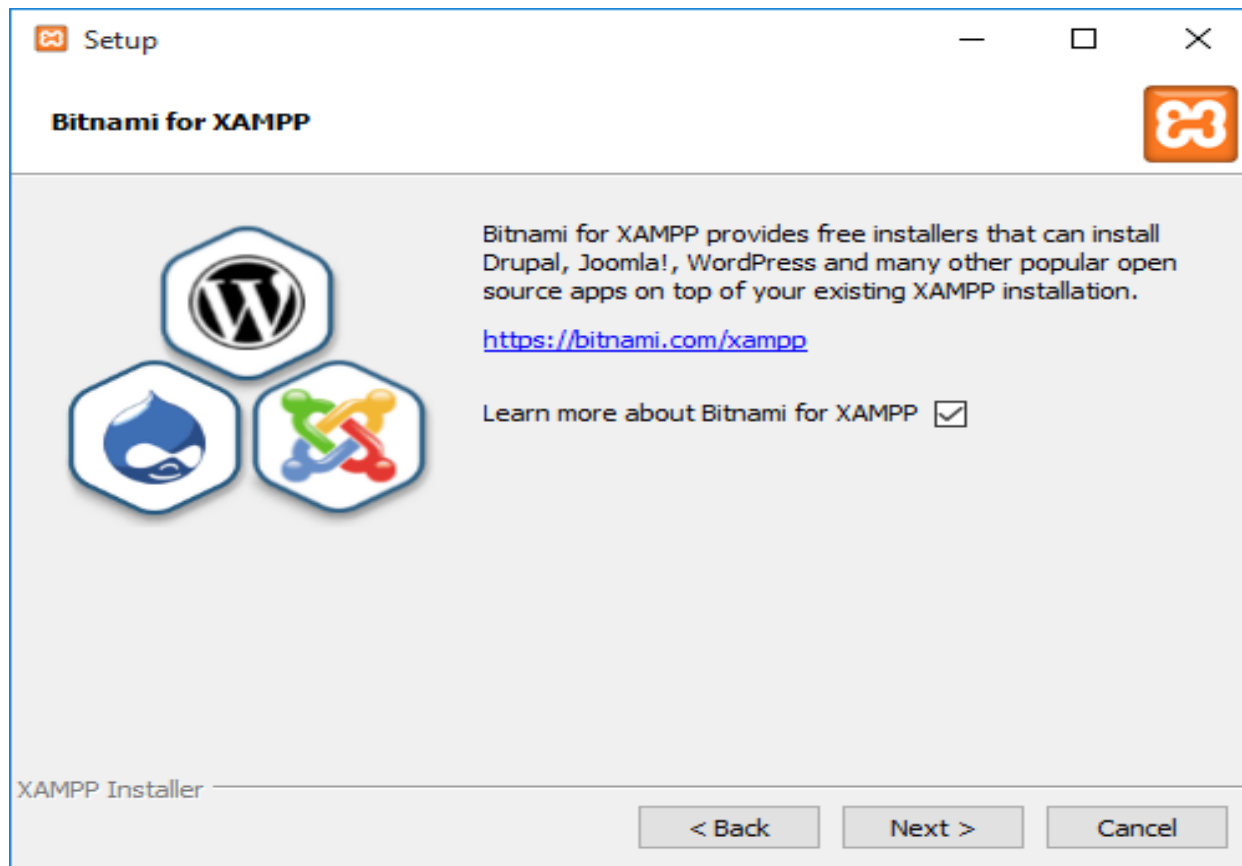


Selanjutnya akan muncul jendela **Select Components**, yang meminta untuk memilih aplikasi yang akan diinstall. Centang saja semua kemudian klik tombol **Next**.



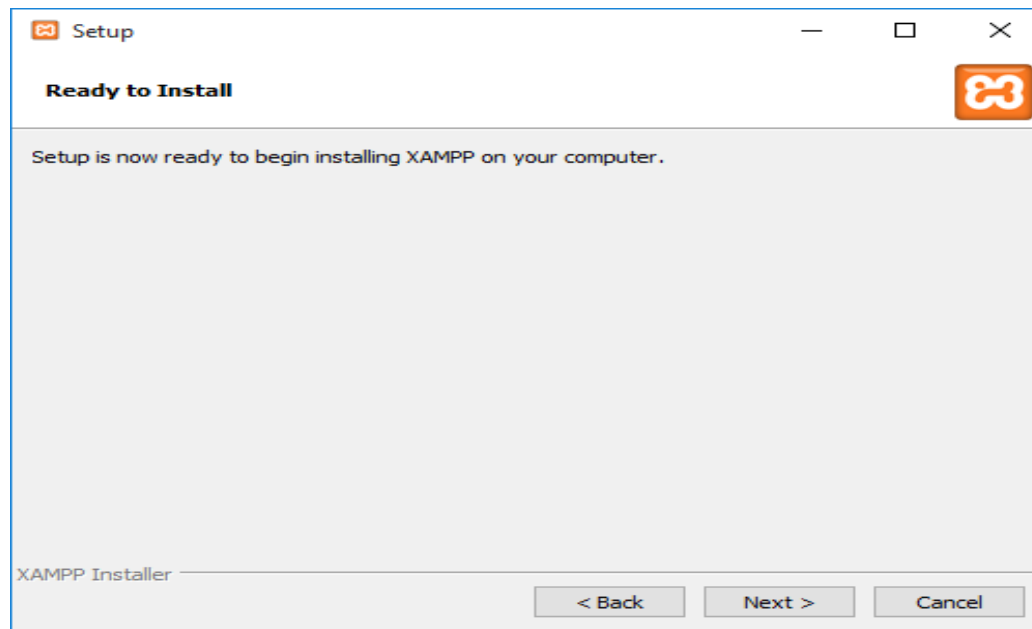
Kemudian akan muncul jendela **Installation Folder**, dimana anda diminta untuk menentukan lokasi penyimpanan folder xampp, secara bawaan akan diarahkan ke lokasi **c:\xampp**. Jika anda ingin menyimpannya di folder lain, anda dapat menekan tombol bergambar folder (**Browse**), kemudian klik tombol **Next**.

Kita akan menjumpai jendela tawaran untuk mempelajari lebih lanjut tentang Bitnami. Silahkan checklist jika ingin mempelajari lebih lanjut. Bitnami adalah pustaka dari aplikasi client server yang populer seperti misalnya CMS WordPress atau Drupal, dengan penawaran kemudahan dalam installasi hanya dengan satu klik. Kemudian klik tombol **Next**.

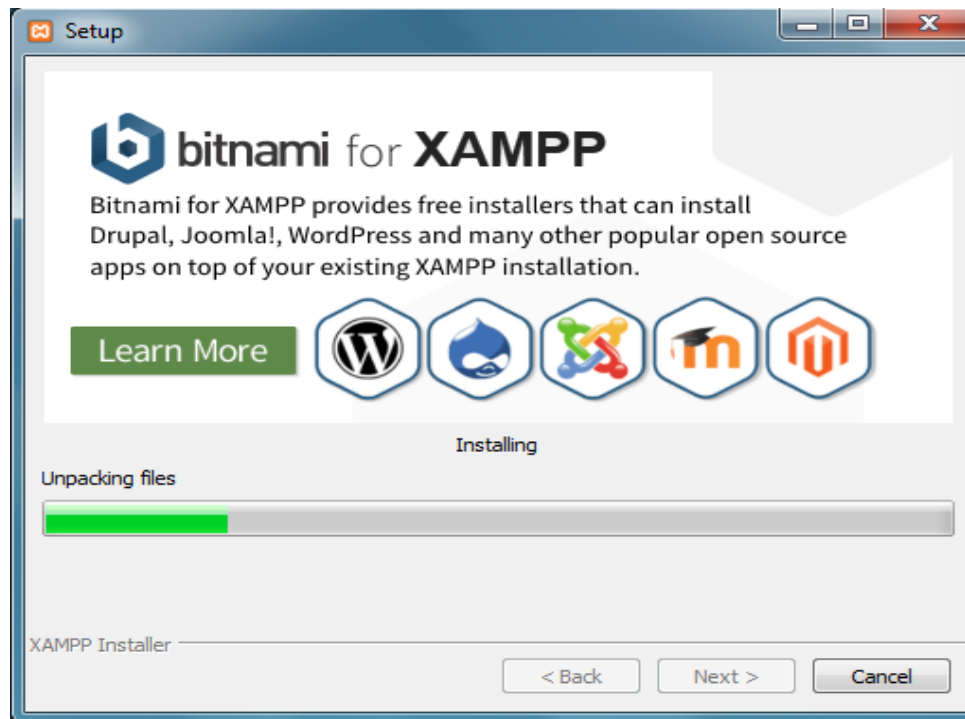




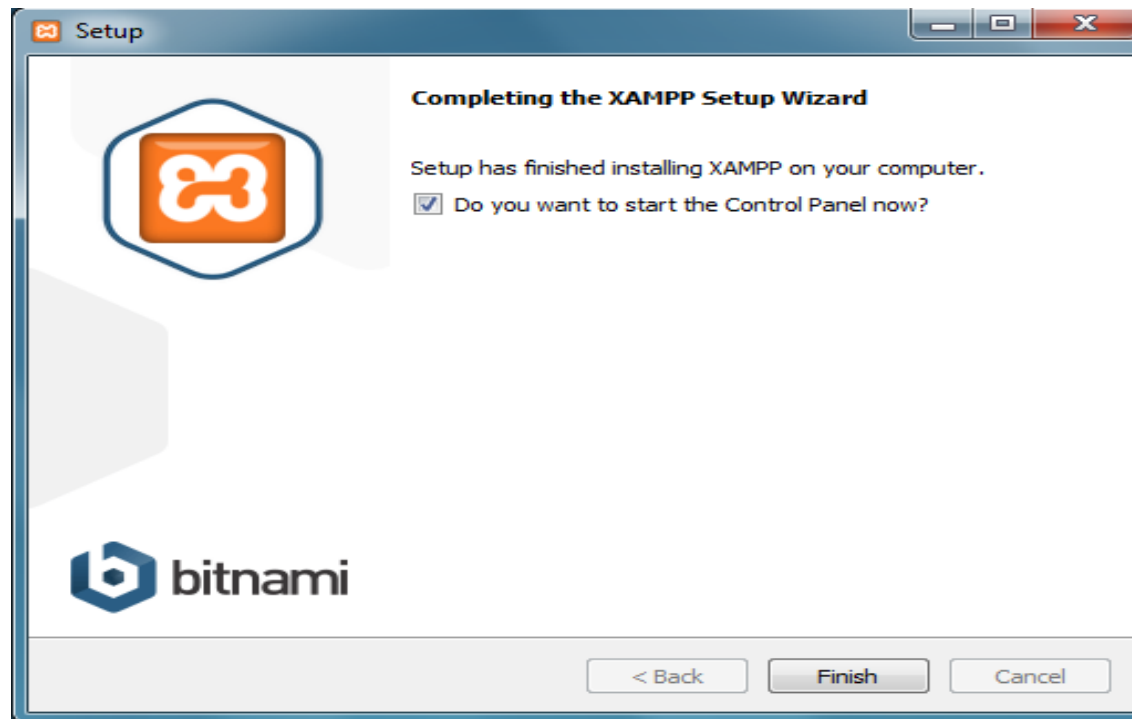
Kemudian muncul jendela **Ready To Install** yang menunjukkan xampp sudah siap di install. Kemudian klik tombol **Next**.



Dan proses instalasi pun berjalan.



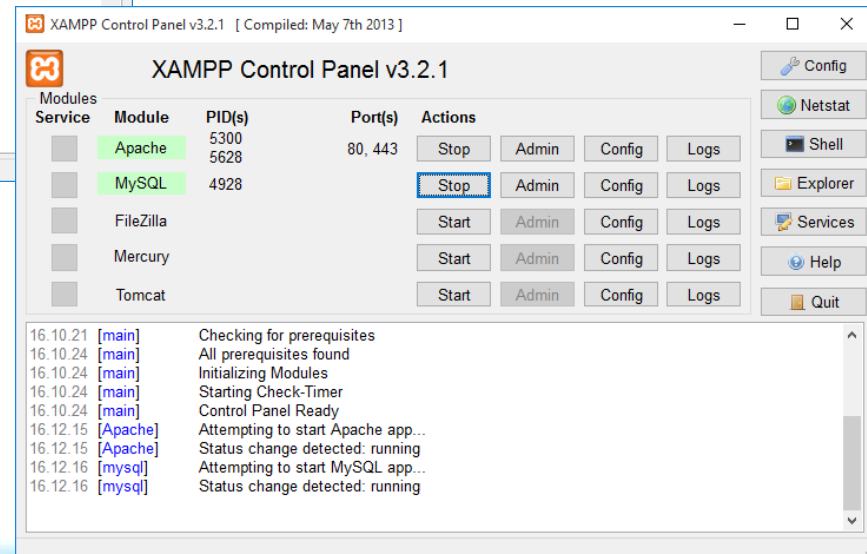
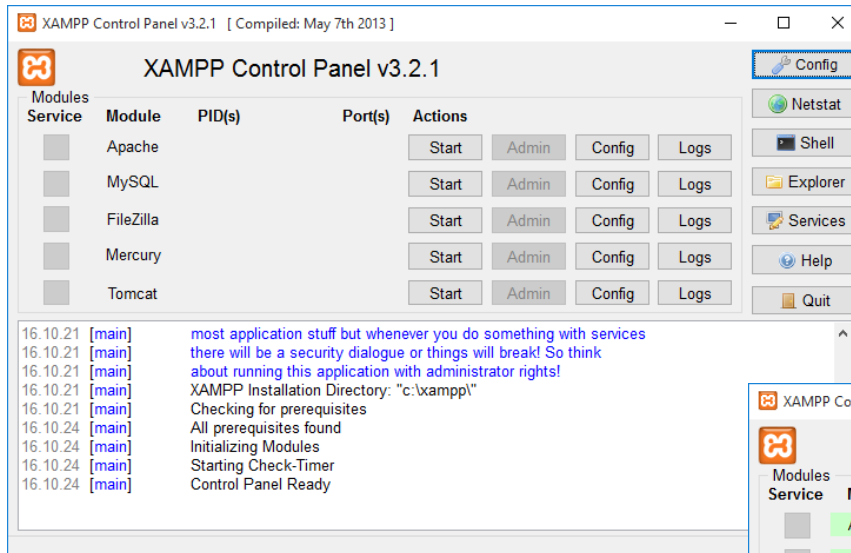
Tunggu hingga proses install selesai dan muncul jendela sebagai berikut.



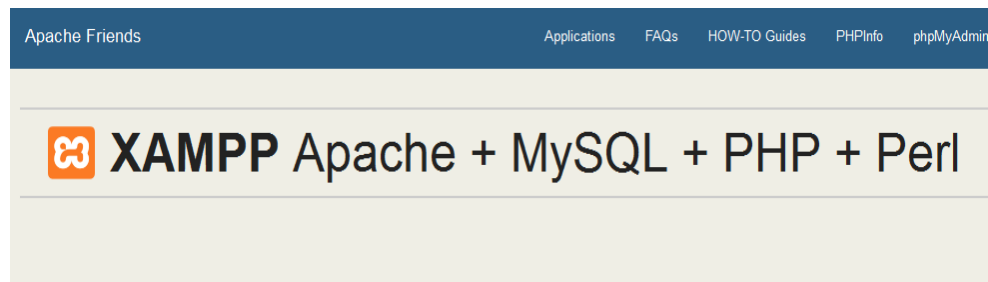


Klik tombol **Finish** setelah itu akan muncul jendela **Xampp Control Panel** yang berguna untuk menjalankan server.

Klik tombol **Start** pada kolom **Actions** untuk module **Apache** dan **MySQL**.



Untuk mengetahui apakah instalasi telah berhasil atau tidak, ketikkan 'localhost/' pada browser, jika berhasil akan muncul halaman seperti gambar dibawah.



## Welcome to XAMPP for Windows 5.6.12

You have successfully installed XAMPP on this system! Now you can start using Apache, MySQL, PHP and other components. You can find more info in the [FAQs](#) section or check the [HOW-TO Guides](#) for getting started with PHP applications.

Start the XAMPP Control Panel to check the server status.





## b. Install Composer

Composer adalah Dependency Management Tools untuk PHP. Sederhananya Composer akan membantu kita mencari dan mendownload file-file yang diperlukan oleh sebuah aplikasi yang akan install yang akan tersimpan pada folder **vendor**. Composer untuk windows dapat didownload di

<https://getcomposer.org/Composer-Setup.exe>.

Install composer sangat mudah, cukup double klik file composer yang telah di download.



Dependency Manager for PHP

[Getting Started](#)

[Download](#)

[Documentation](#)

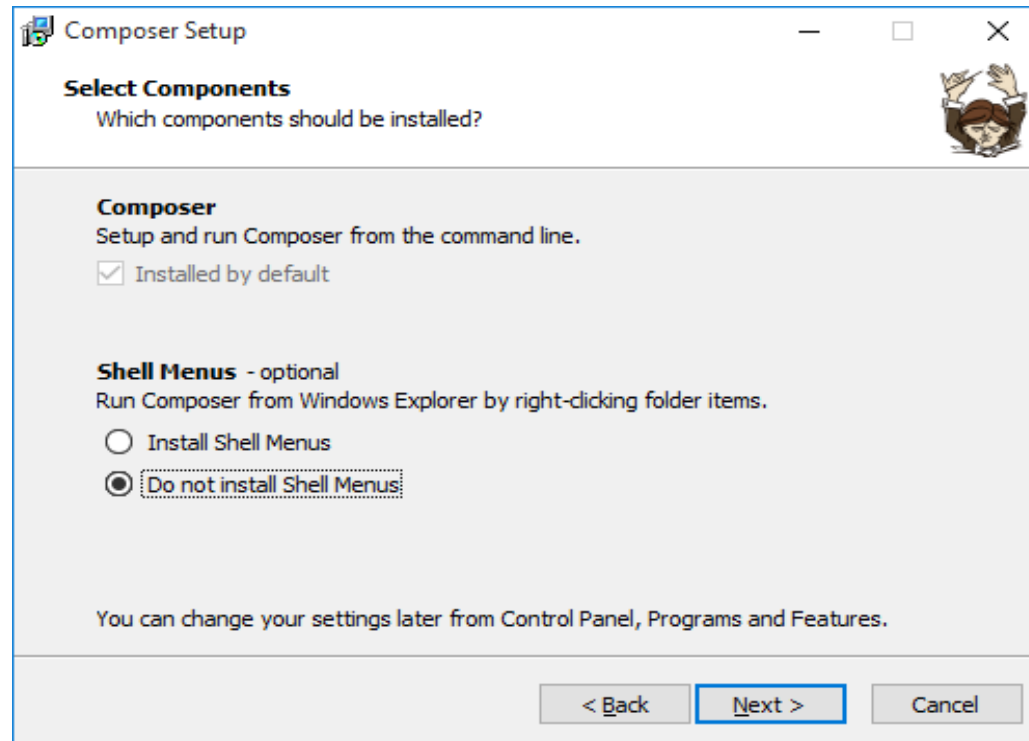
[Browse Packages](#)

[Issues](#)

[GitHub](#)

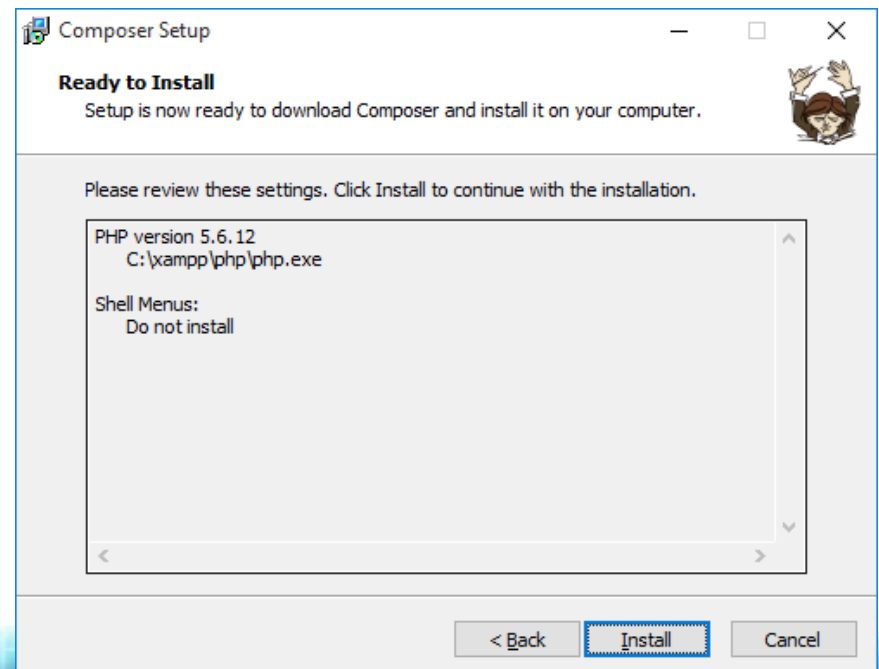
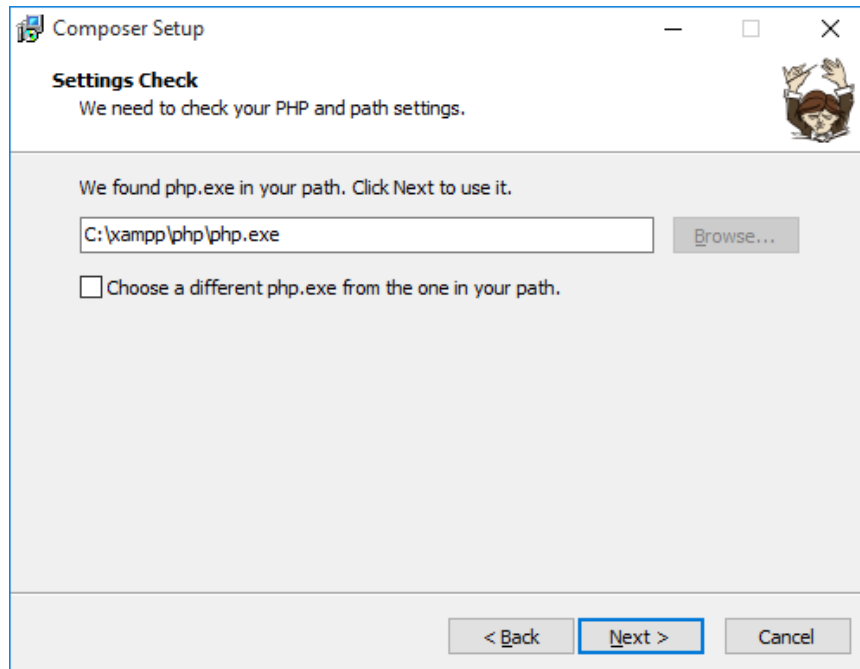


Kemudian klik tombol **Next**.



Pada jendela **Selects Components** kita dapat lanjutkan dengan menekan tombol **Next**.

Pada umumnya secara otomatis composer akan mendapatkan file php.exe yang telah kita install sebelumnya (Installasi XAMPP). Lanjutkan dengan menekan tombol **Next** dan **Install**.





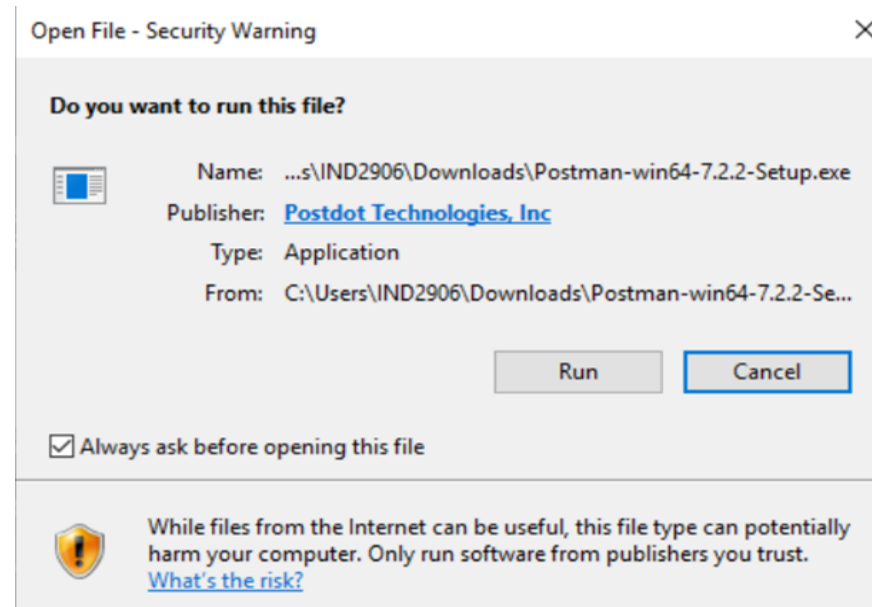
## c. Install Postman

Postman adalah sebuah aplikasi fungsinya adalah sebagai REST Client atau istilahnya adalah aplikasi yang digunakan untuk melakukan uji coba REST API yang telah kita buat. Postman ini merupakan tools wajib bagi para developer yang bergerak pada pembuatan API, fungsi utama postman ini adalah sebagai GUI API Caller Pemanggil. namun sekarang postman juga menyediakan fitur lain yaitu Sharing Collection API for Documentation (free), Testing API (free), Realtime Collaboration Team (paid), Monitoring API (paid), Integration (paid).

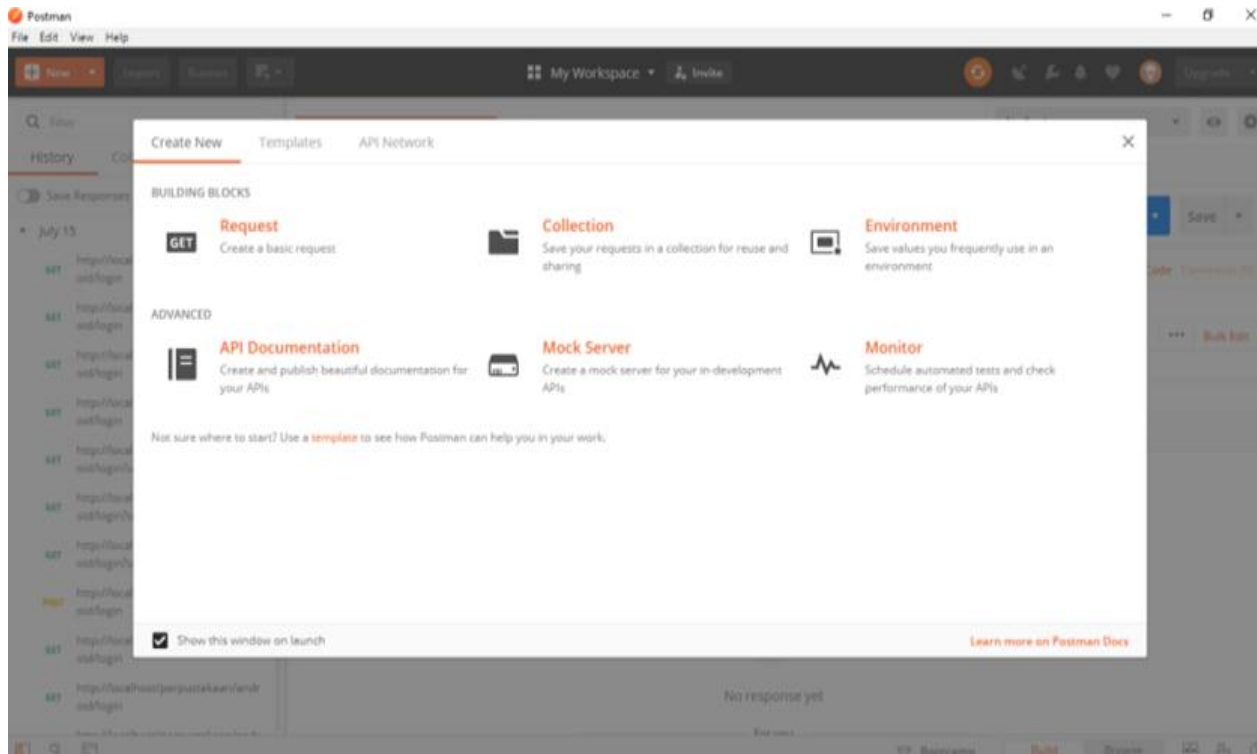


Postman tersedia sebagai aplikasi asli untuk sistem operasi macOS, Windows (32-bit dan 64-bit), dan Linux (32-bit dan 64-bit). Untuk mendapatkan aplikasi Postman, dapat diunduh pada website resminya yaitu [getpostman.com](https://www.getpostman.com) atau dapat diunduh pada halaman <https://www.postman.com/downloads/>

Setelah berhasil mengunduh paket instalasi postman, kemudian jalankan dengan cara klik dua kali. Pilih run jika muncul pop up seperti berikut :



Kemudian tunggu hingga proses instalasi selesai dan muncul seperti gambar berikut





## 3. API SPEC

API SPEC ini bermaksud untuk membuat standar API sebagai dokumentasi kepada pengembang baik itu frontend maupun backend

## a. Registrasi

EndPoint	/registrasi
Method	POST
Header	· Content-Type: application/json
Body	<pre>{   "nama"    : "string",   "email"   : "string, unique",   "password" : "string" }</pre>
Response	<pre>{    "code" : "integer",   "status" : "boolean",   "data" : "string" }</pre>

## b. Login

EndPoint	/login
Method	POST
Header	· Content-Type: application/json
Body	<pre>{   "email" : "string"   "password" : "string" }</pre>
Response	<pre>{   "code" : "integer",   "status" : "boolean",   "data" : {     "token" : "string",     "user" : {       "id" : "integer",       "email" : "string",     }   } }</pre>



## c. Produk

### 1. List Produk

EndPoint	/produk
Method	GET
Header	· Content-Type: application/json
Response	<pre>{   "code" : "integer",   "status" : "boolean",   "data" : [     {       "id" : "integer",       "kode_produk" : "string",       "nama_produk" : "string",       "harga" : "integer",     },     {       "id" : "integer",       "kode_produk" : "string",       "nama_produk" : "string",       "harga" : "integer",     }   ] }</pre>

## 2. Create Produk

EndPoint	/produk
Method	POST
Header	· Content-Type: application/json
Body	<pre>{   "kode_produk" : "string",   "nama_produk" : "string",   "harga"       : "integer" }</pre>
Response	<pre>{    "code" : "integer",   "status" : "boolean",   "data" : {     "id" : "integer",     "kode_produk" : "string",     "nama_produk" : "string",     "harga" : "integer",   } }</pre>

### 3. Update Produk

<b>EndPoint</b>	<code>/produk/{id}/update</code>
<b>Method</b>	POST
<b>Header</b>	<code>Content-Type: application/json</code>
<b>Body</b>	<pre>{   "kode_produk" : "string",   "nama_produk" : "string",   "harga"       : "integer" }</pre>
<b>Response</b>	<pre>{   "code" : "integer",   "status" : "boolean",   "data" : "boolean" }</pre>

## 4. Show Produk

EndPoint	/produk/{id}
Method	GET
Header	Content-Type: application/json
Response	<pre>{   "code" : "integer",   "status" : "boolean",   "data" : {     "id" : "integer",     "kode_produk" : "string",     "nama_produk" : "string",     "harga" : "integer",   } }</pre>

## 5. Delete Produk

<b>EndPoint</b>	/produk/{id}
<b>Method</b>	DELETE
<b>Header</b>	Content-Type: application/json
<b>Response</b>	<pre>{   "code" : "integer",   "status" : "boolean",   "data" : "boolean" }</pre>