

## Langkah-Langkah Membuat Dokumentasi Perangkat Lunak

Dokumentasi perangkat lunak yang baik, apakah dokumen spesifikasi untuk programmer dan penguji, dokumen teknis untuk pengguna internal, atau manual perangkat lunak dan file bantuan bagi pengguna akhir, *membantu orang yang bekerja dengan perangkat lunak memahami fitur dan fungsi*. Dokumentasi perangkat lunak yang baik adalah spesifik, singkat, dan relevan, menyediakan semua informasi penting kepada orang yang menggunakan perangkat lunak. Berikut ini adalah petunjuk tentang cara untuk menulis dokumentasi perangkat lunak untuk pengguna teknis dan pengguna akhir.

### Menulis Dokumentasi Software untuk Pengguna Teknis (*Technical Users*)

1. Menentukan informasi apa yang perlu dimasukkan. Dokumen spesifikasi perangkat lunak berfungsi sebagai referensi manual untuk desainer *user interface*, *programmer* yang menulis kode, dan penguji yang memverifikasi bahwa perangkat lunak tersebut bekerja sebagaimana dimaksud. Informasi yang tepat tergantung pada program tersebut tetapi dapat mencakup hal-hal berikut:
  1. File kunci dalam aplikasi. Ini mungkin termasuk file yang dibuat oleh tim pengembangan, *database* diakses selama operasi program, dan program utilitas pihak ketiga.
  2. Fungsi dan subrutin. Ini termasuk penjelasan tentang apa yang setiap fungsi atau *subroutine*, termasuk jangkauan dari nilai *input* dan nilai *output*.
  3. Variabel program dan konstanta, dan bagaimana mereka digunakan dalam aplikasi.
  4. Struktur program secara keseluruhan. Untuk aplikasi berbasis *disc*, ini berarti menggambarkan modul individu program dan perpustakaan, sedangkan untuk aplikasi *Web*, ini berarti menggambarkan halaman yang menggunakan file mana.
2. Putuskan berapa banyak dokumentasi harus berada dalam kode program dan berapa banyak harus terpisah. Dokumentasi yang lebih teknis dikembangkan dalam memulai *source code* program. Minimal, dokumentasi dalam *source code* perlu menjelaskan tujuan fungsi, subrutin, variabel, dan konstanta.
  1. Jika *source code* sangat panjang, dapat didokumentasikan dalam bentuk file bantuan, yang dapat diindeks atau dicari dengan kata kunci. Ini adalah keuntungan tertentu untuk aplikasi di mana logika program terfragmentasi lebih banyak halaman dan mencakup sejumlah file tambahan, seperti dengan aplikasi *Web* tertentu.
  2. Beberapa bahasa pemrograman, seperti *Java* dan *.NET Framework (Visual Basic.NET, C#)*, memiliki standar sendiri untuk mendokumentasikan kode. Dalam kasus ini, mengikuti standar untuk berapa banyak dokumentasi harus disertakan dengan *source code*.

3. Pilih alat dokumentasi yang sesuai. Untuk beberapa hal, ini ditentukan oleh bahasa kode ditulis dalam, baik itu *C++*, *C#*, *Visual Basic*, *Java*, atau *PHP*, sebagai alat khusus ada untuk bahasa ini dan lainnya. Dalam kasus lain, penggunaan *tools* ditentukan oleh jenis dokumentasi yang diperlukan.
- Program pengolah kata untuk *Microsoft Word* yang memadai untuk membuat file teks terpisah dokumentasi, asalkan dokumentasi yang cukup singkat dan sederhana. Untuk lama, file teks yang kompleks, banyak penulis teknis lebih memilih alat dokumentasi seperti *Adobe FrameMaker*.
- File bantuan untuk mendokumentasikan *source code* dapat diproduksi dengan alat bantuan *authoring*, seperti *RoboHelp*, *Help and Manual*, *Doc-To-Help*, *Madcap Flare*, atau *HelpLogix*.

#### Menulis Dokumentasi Software untuk Pengguna Akhir (*End User*)

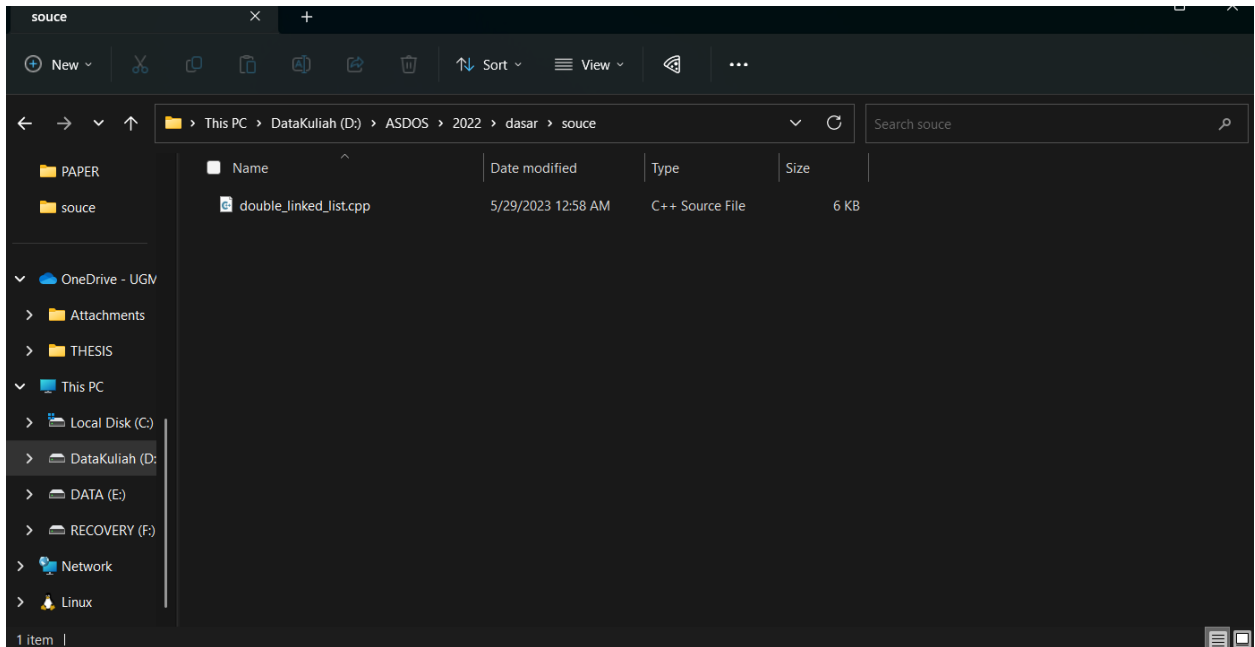
1. Tentukan alasan bisnis untuk dokumentasi Anda. Meskipun alasan fungsional untuk mendokumentasikan perangkat lunak untuk membantu pengguna memahami bagaimana untuk menggunakan aplikasi ini, ada alasan lain juga, seperti membantu dalam pemasaran perangkat lunak, meningkatkan citra perusahaan, dan terutama, mengurangi biaya dukungan teknis. Dalam beberapa kasus, dokumentasi yang diperlukan untuk mematuhi peraturan tertentu atau persyaratan hukum lainnya. Dalam hal ini, bagaimanapun, seharusnya dokumentasi *software* untuk pengganti dari desain antarmuka yang buruk. Jika layar aplikasi membutuhkan bertumpuk-tumpuk dokumentasi untuk menjelaskannya, lebih baik untuk mengubah desain layar untuk sesuatu yang lebih intuitif.
2. Memahami penonton yang merupakan tujuan dokumentasi dibuat. Dalam kebanyakan kasus, pengguna perangkat lunak memiliki sedikit pengetahuan tentang komputer selain tugas-tugas aplikasi yang mereka gunakan. Ada beberapa cara untuk menentukan bagaimana untuk memenuhi kebutuhan mereka dengan dokumentasi. Lihatlah jabatan calon pengguna. Seorang administrator sistem mungkin pakar dengan sejumlah aplikasi perangkat lunak, sementara petugas entri data lebih mungkin untuk mengetahui hanya aplikasinya saat menggunakan untuk memasukkan data. Lihatlah pengguna sendiri. Meskipun jabatan umumnya menunjukkan apa yang dilakukan orang, bisa ada variasi dalam bagaimana judul tertentu digunakan dalam suatu organisasi. Dengan mewawancarai calon pengguna, bisa dirasakan tentang jabatan user untuk menunjukkan akurat atau tidaknya. Lihatlah dokumentasi yang ada. Dokumentasi untuk versi sebelumnya dari perangkat lunak, serta spesifikasi fungsional, memberikan beberapa indikasi seperti apa pengguna akan perlu diketahui untuk menggunakan program. Perlu diingat, bagaimanapun, pengguna akhir tidak begitu tertarik pada bagaimana program itu bekerja ketika mereka berada di apa yang dapat dilakukan untuk mereka. Mengidentifikasi tugas-tugas yang diperlukan untuk melakukan pekerjaan itu, dan apa tugas-tugas yang perlu dilakukan sebelum tugas-tugas dapat dilakukan .

3. Tentukan format yang sesuai untuk dokumentasi. Dokumentasi perangkat lunak dapat disusun dalam satu dari dua format, referensi manual dan buku petunjuk . Kadang-kadang, kombinasi dari format adalah pendekatan yang terbaik. Sebuah format referensi manual dikhususkan untuk menjelaskan setiap fitur dari aplikasi perangkat lunak (*tombol*, *tab*, *field*, dan *dialog box*) dan bagaimana mereka bekerja. Banyak file bantuan yang ditulis dalam format ini, khususnya bantuan konteks-sensitif yang menampilkan topik yang relevan setiap kali pengguna mengklik tombol Bantuan pada layar tertentu. Sebuah format buku petunjuk menjelaskan cara menggunakan perangkat lunak untuk melakukan tugas tertentu. Panduan pengguna sering diformat sebagai panduan atau PDF yang siap dicetak, meskipun beberapa file bantuan mencakup topik-topik tentang bagaimana melakukan tugas-tugas tertentu. (Topik ini biasanya tidak konteks-sensitif, meskipun mereka mungkin hyperlink ke dari topik yang.) Panduan pengguna sering mengambil bentuk tutorial, dengan ringkasan dari tugas-tugas yang akan dilakukan dalam pengenalan dan petunjuk yang diberikan dalam langkah-langkah bernomor .
4. Putuskan apa bentuk dokumentasi. Dokumentasi perangkat lunak untuk pengguna akhir dapat mengambil satu atau beberapa dari banyak bentuk : cetak manual, dokumen PDF, file bantuan, atau bantuan *online*. Setiap bentuk dirancang untuk menunjukkan pengguna bagaimana menggunakan masing-masing fungsi program, baik dalam bentuk langkah-langkah atau tutorial. Dalam kasus file bantuan dan bantuan online, ini mungkin termasuk video demonstrasi serta teks dan masih grafis. File bantuan dan bantuan *online* harus diindeks dan kata kunci dicari untuk memungkinkan pengguna untuk dengan cepat menemukan informasi yang mereka cari. Meskipun alat-alat file bantuan authoring dapat menghasilkan indeks secara otomatis, sering lebih baik untuk membuat indeks secara manual, menggunakan istilah pengguna cenderung untuk mencari.
5. Pilih alat dokumentasi yang sesuai. Dicitak atau manual PDF pengguna dapat ditulis dengan program pengolah kata seperti Word atau editor teks yang canggih seperti FrameMaker, tergantung pada panjang dan kompleksitas. Bantuan file dapat ditulis dengan alat bantuan authoring seperti *RoboHelp*, *Help and Manual*, *Doc-To-Help*, *Flare*, atau *HelpLogix*.

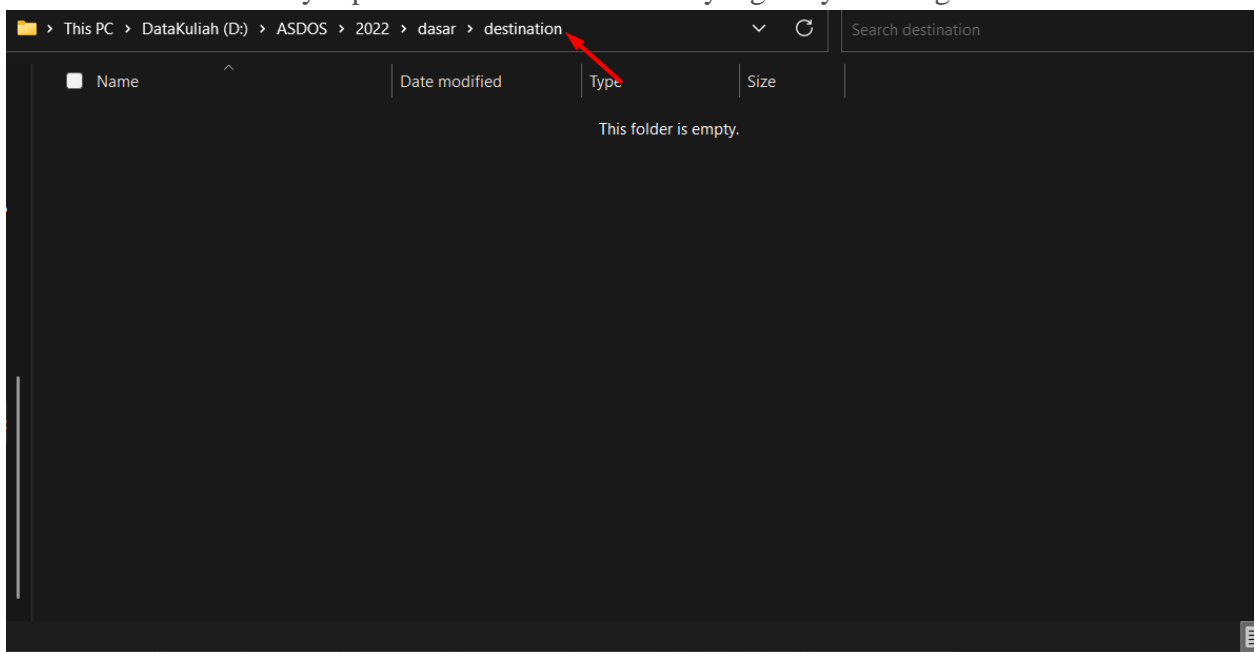
Nah pada sesi ini kita akan belajar membuat dokumentasi perangkat lunak yang ditujukan bagi pengguna teknis dengan alat bantu (tool) yaitu **Doxygen**. Kalian bisa download dan install dulu doxygen ini. Silahkan download pada link <https://www.doxygen.nl/files/doxygen-1.9.7-setup.exe>

Jika sudah diinstall, kita coba melakukan pembuatan dokumentasi teknis dengan menggunakan doxygen ini. Siapkan salah satu program yang telah kalian buat terdahulu. Dan ikuti Langkah-langkah berikut ini.

1. Membuat folder baru yang isinya adalah file project c++ dengan extension .cpp.



2. Buat Folder untuk menyimpan hasil dari dokumentasi yang isinya kosong



3. Buka Doxygen kemudian isilah tentang Project kita seperti yang anda lihat pada gambar .  
 Step 1 berisi folder tempat project yang kita buat, dalam hal ini saya menggunakan contoh project aplikasi switch. Project Name bisa kalian isi dengan yang sesuai proyek anda, Project Sinopsis bisa anda isi dengan penjelasan ringkas tentang proyek anda, project version bisa kalian isi dengan versi proyek anda. Project logo bisa anda isi dengan gambar yang sesuai. Source code directory tidak perlu diisi, centanglah Scan recursively. Destination merupakan folder untuk menyimpan hasil dokumentasi.

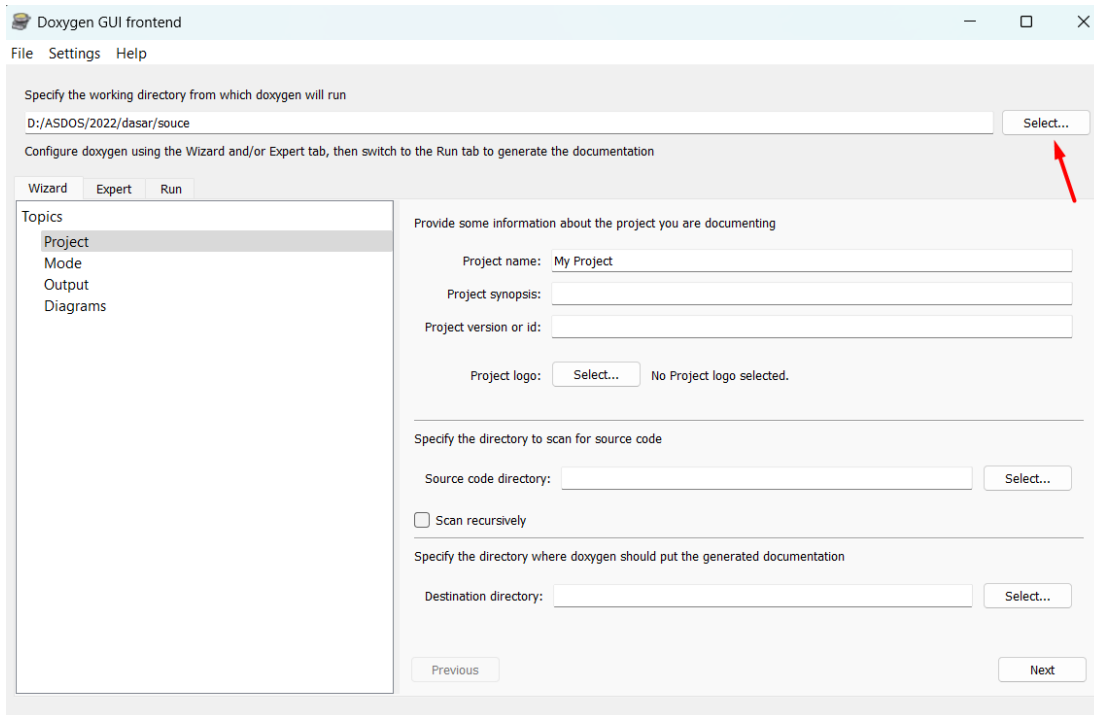


Figure 1 Folder Source

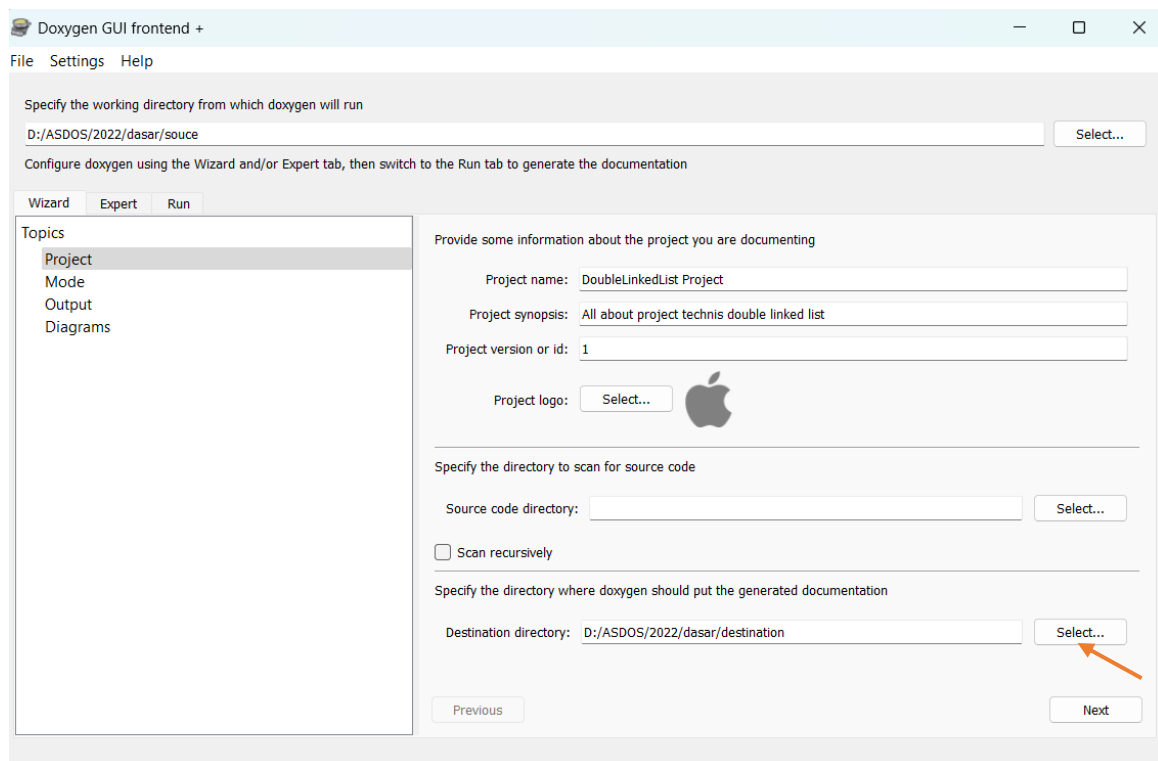
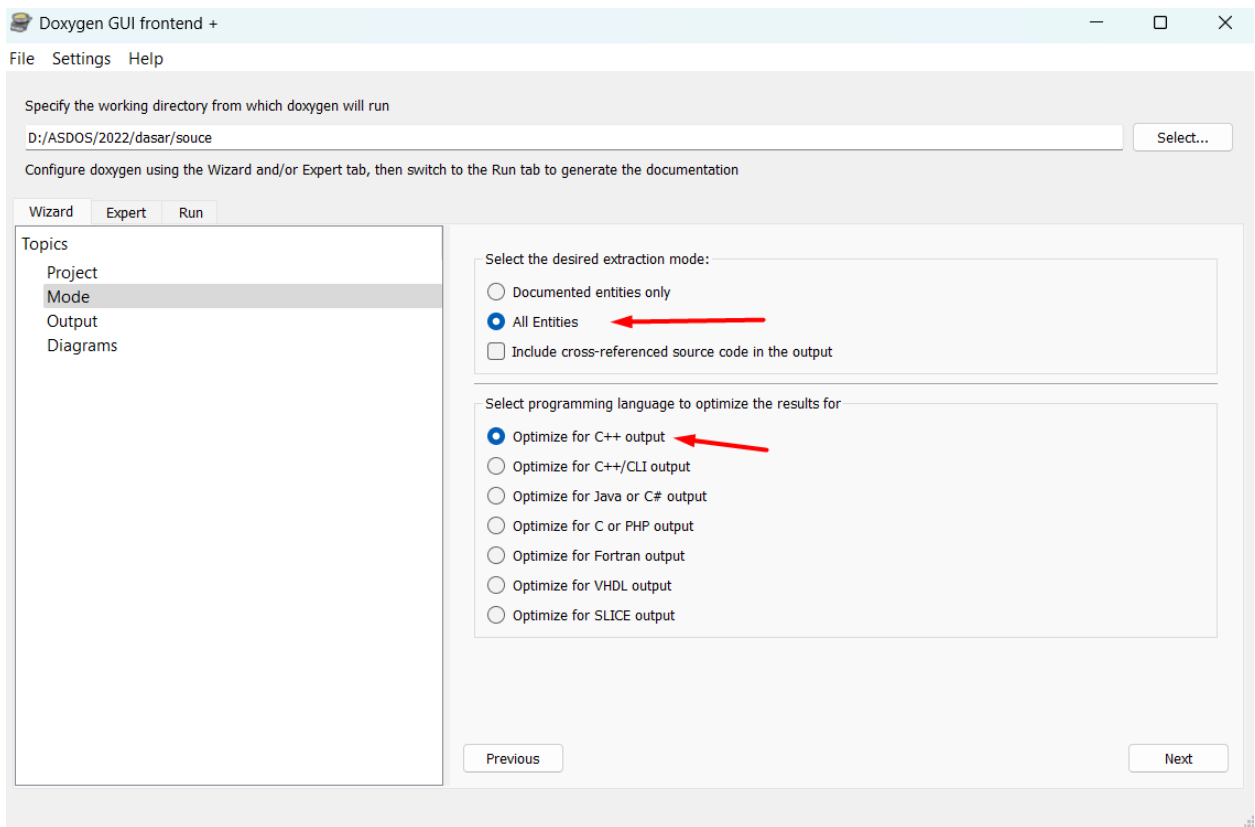


Figure 2 Destination Folder

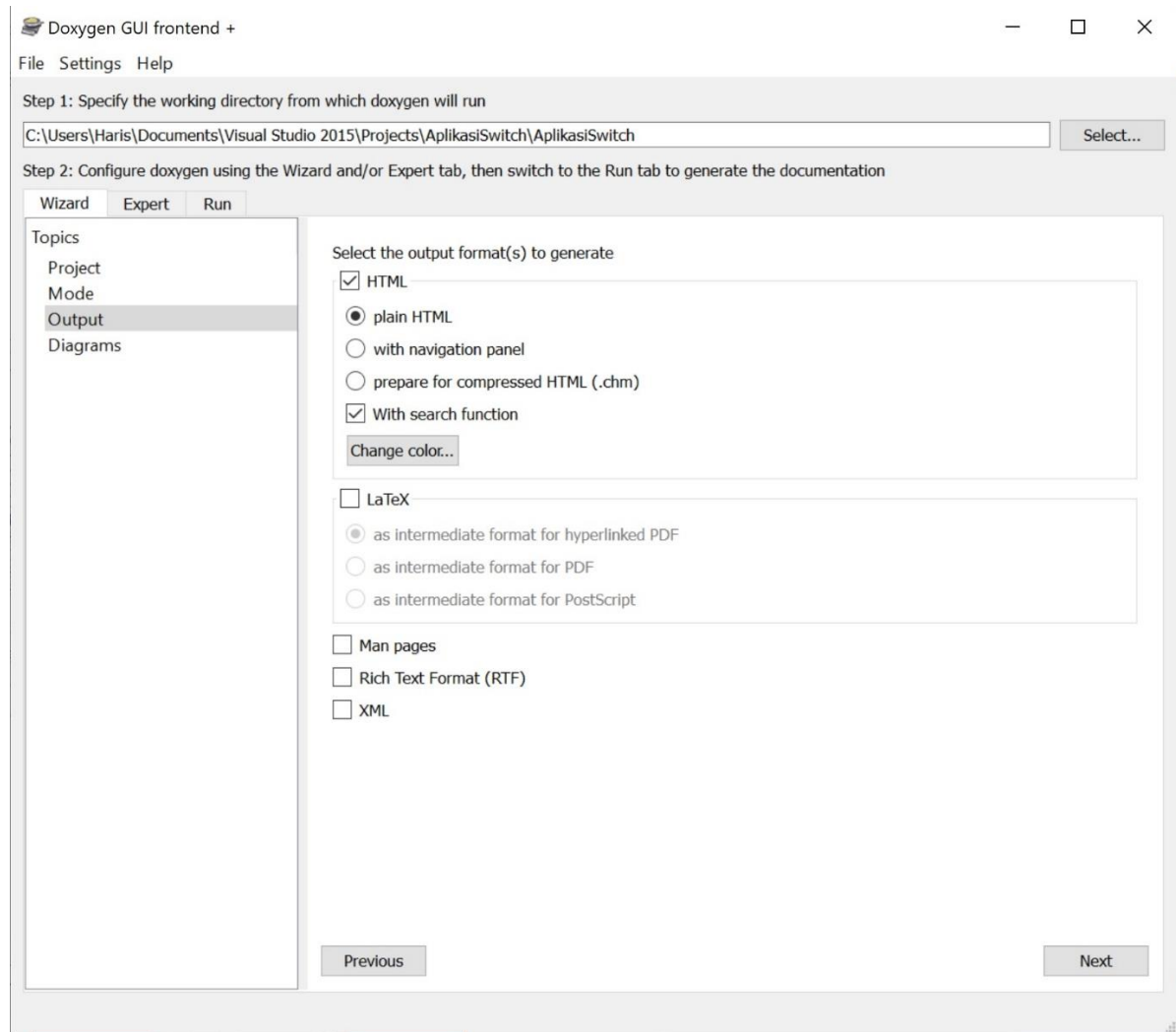
4. Klik “next” dan isilah seperti pada gambar berikut

Pada mode pilihlah “All entities” kemudian Bahasa pemrograman pilihlah “optimize to java or C++ output”

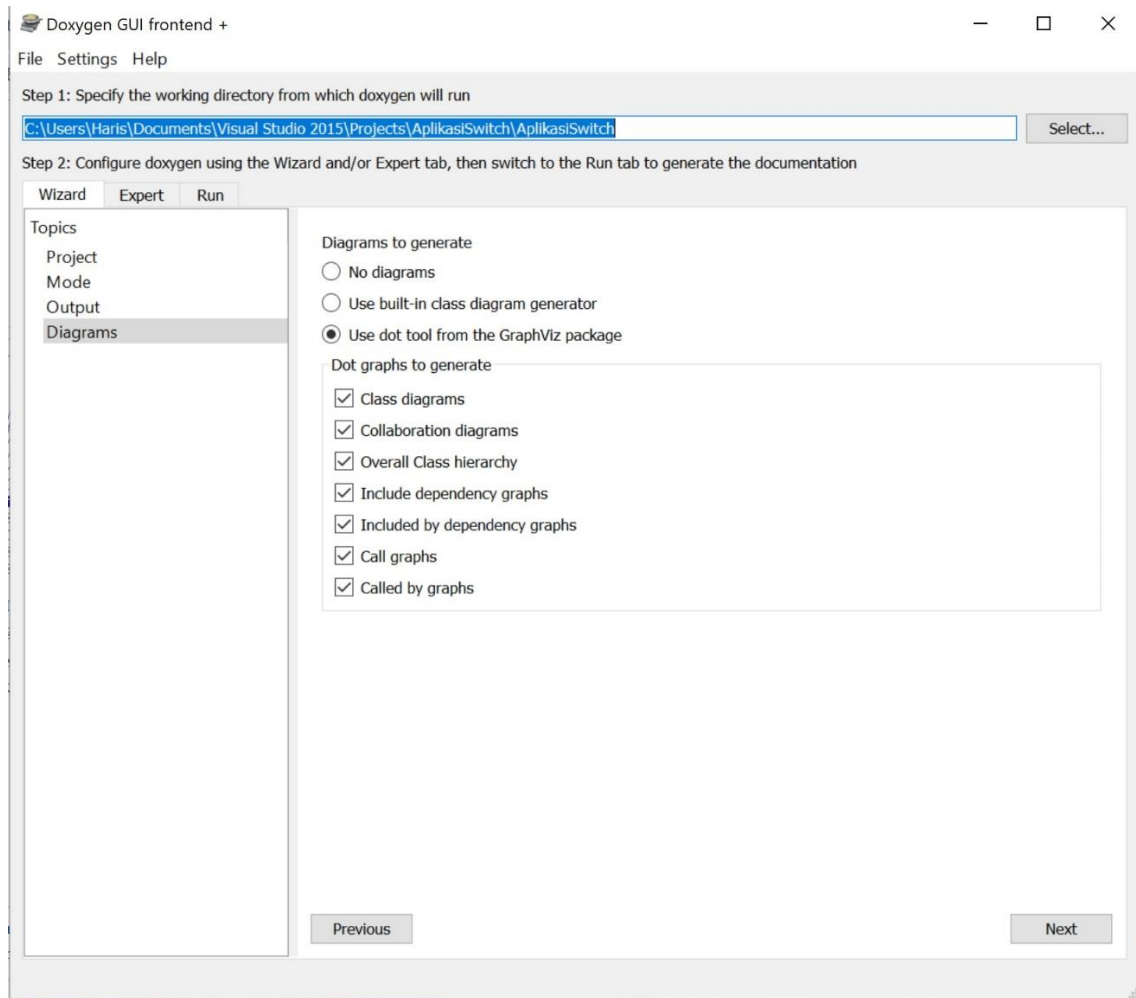


5. Klik “next” dan isilah seperti gambar berikut

Pada setting Output, pilihlah format outputnya adalah HTML agar nanti bisa dibuka dan dibaca di browser saja.

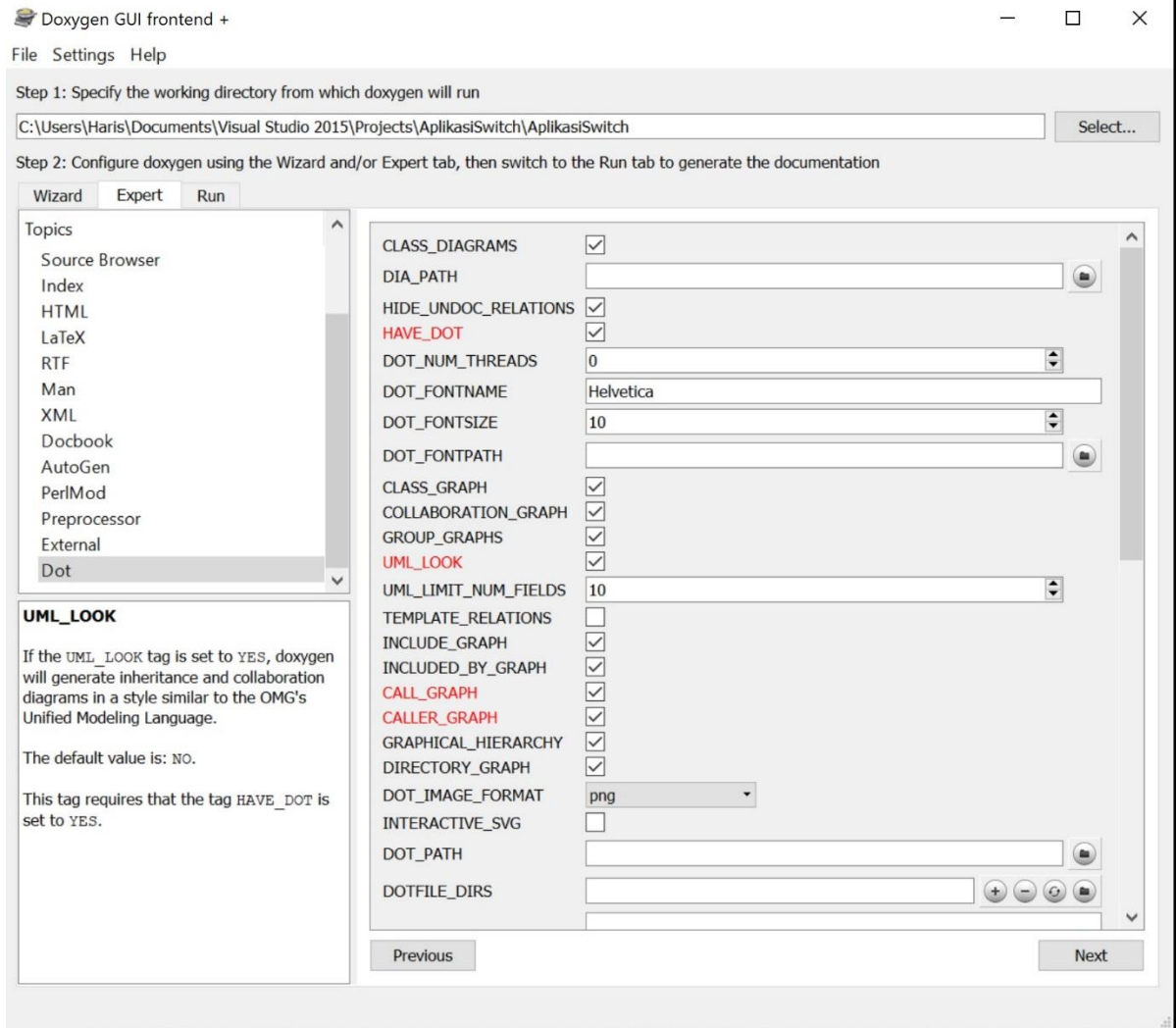


6. Klik next dan isilah seperti gambar berikut  
Pada setting 'Diagram', pilihlah "use dot tool from Graphviz package". Kemudian klik tab expert.

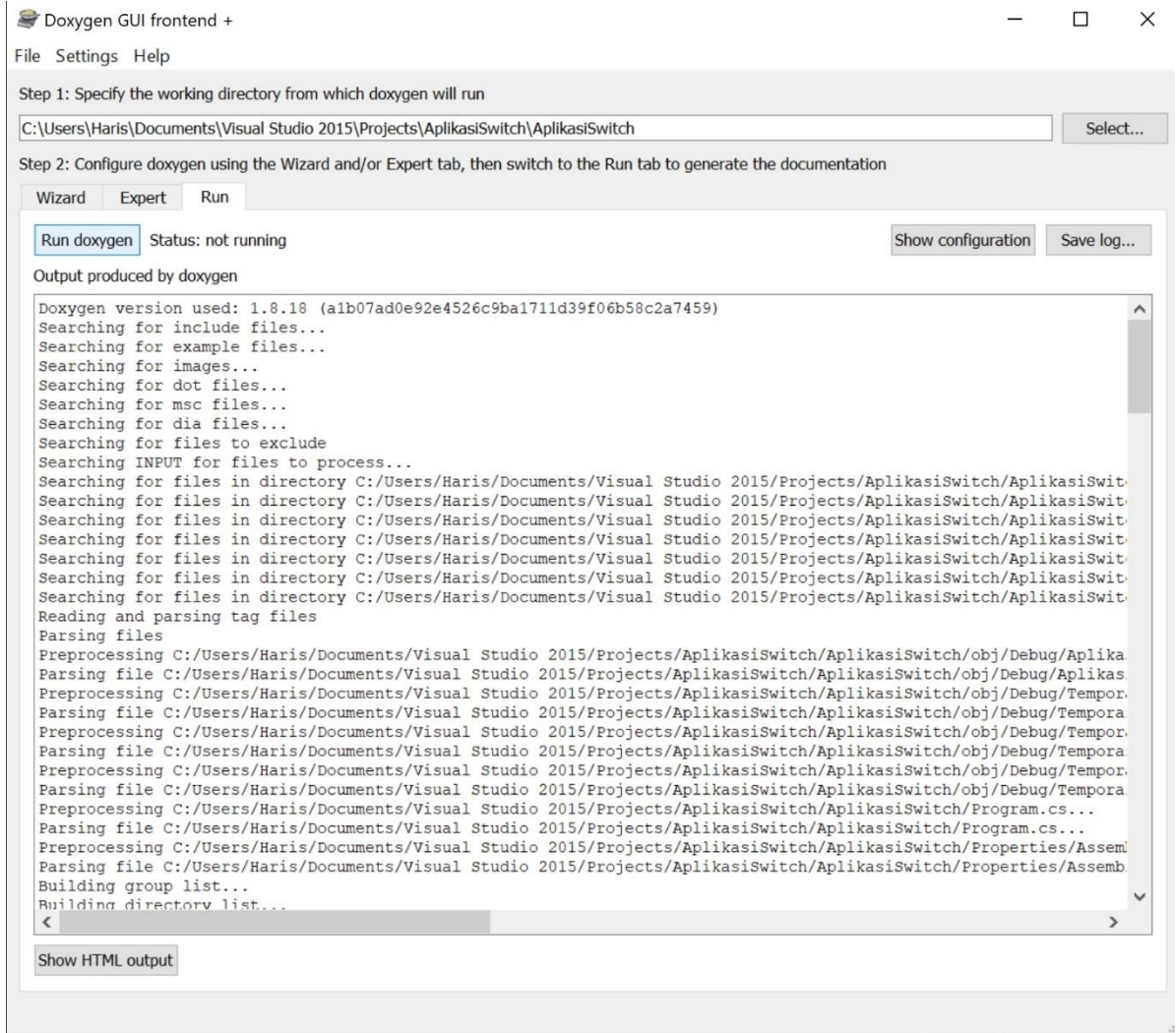


7. Klik tab “expert” dan scroll kebawah dan pilihlah “dot”, disebelah kana nada pilihan-pilihan yang harus anda centang, sementara anda ikuti itu dulu, selebihnya nanti anda bisa mengeksplorasi fitur-fitur yang dimiliki oleh doxygen.

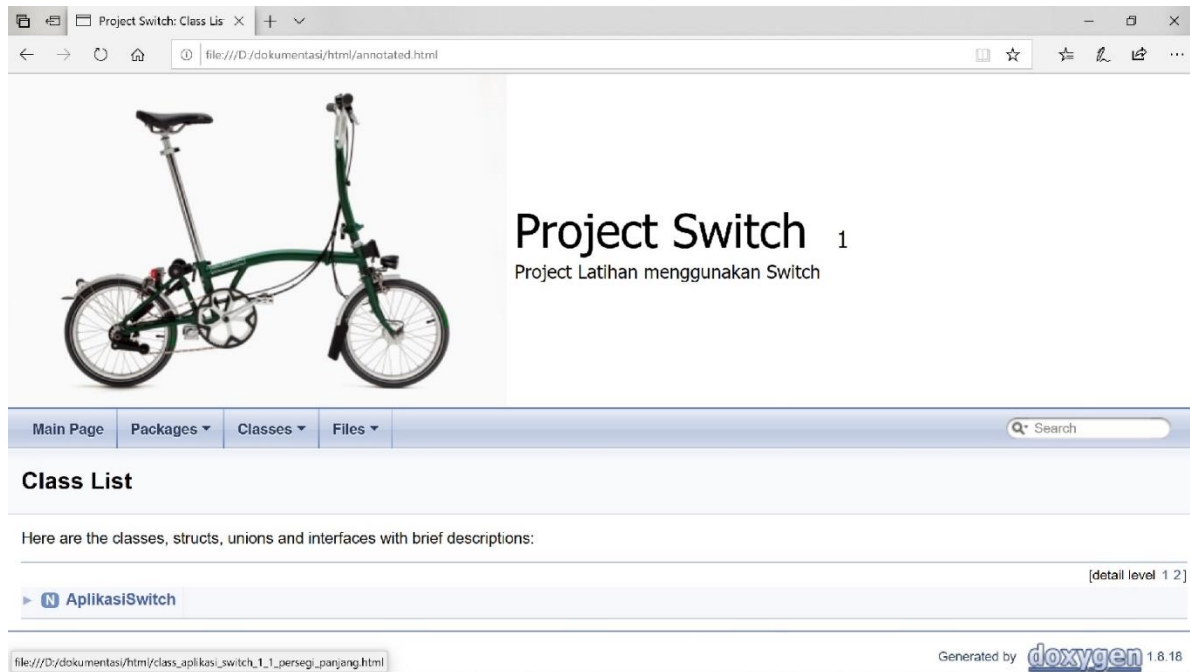




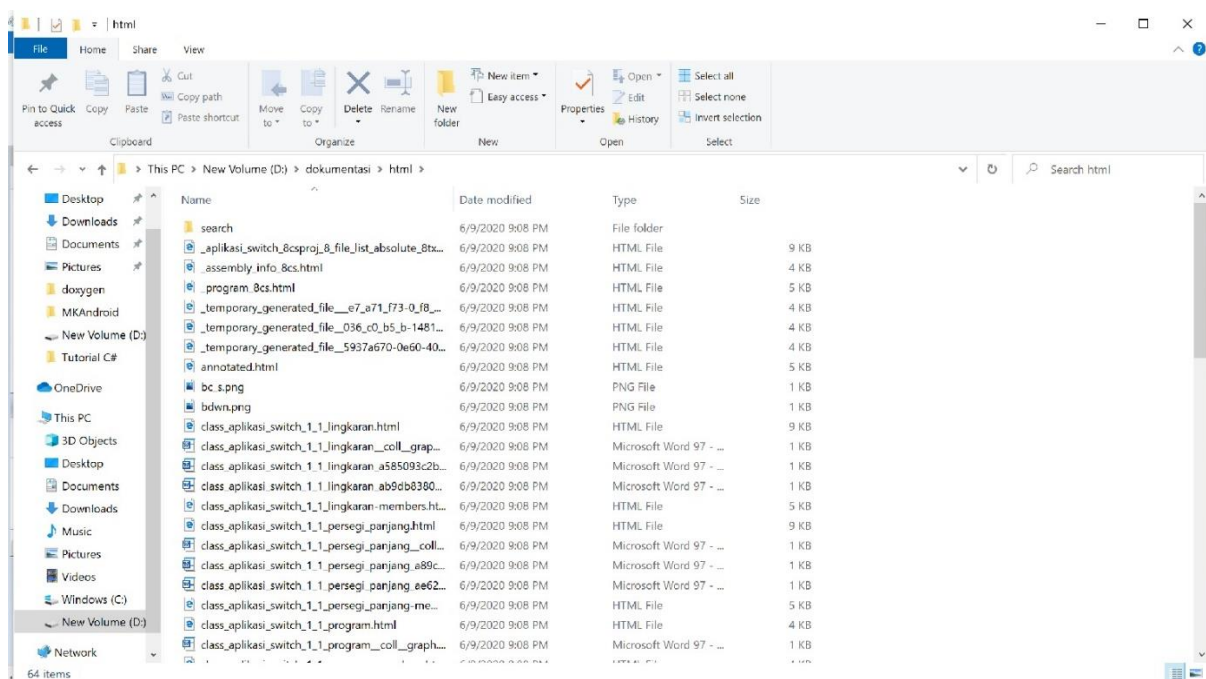
8. Klik "next" atau anda masuk ke tab 'Run'
- Klik "Run Doxygen" dan tunggu beberapa saat hingga muncul hasil Running seperti pada gambar.



Setelah itu klik “show HTML output” akan keluar hasilnya :



Atau hasilnya bisa anda lihat di folder yang kalian definisikan di setting project awal. Anda bisa klik file dengan extension 'HTML file'.



**Selamat mencoba membuat dokumentasi teknis.**