Software that every programmer makes [closed]

Asked 15 years, 9 months ago   Modified 15 years, 8 months ago

Viewed 2k times

5

votes

As it currently stands, this question is not a good fit for our Q&A format. We expect answers to be supported by facts, references, or expertise, but this question will likely solicit debate, arguments, polling, or extended discussion. If you feel that this question can be improved and possibly reopened, visit the help center for guidance.

Closed 12 years ago.

Locked. This question and its answers are locked because the question is off-topic but has historical significance. It is not currently accepting new answers or interactions.

I'm looking for a personal project. One of the sort that "every programmer"* makes. The reasoning is that most developers have a tendency to work on personal projects that have particular traits interesting to all other developers. Software like:

- Bug trackers (made one)

- CMS (made one)

- Blog Engines (Haven't made one; don't want to)

- UI 'libraries' and graphics 'engines' (often for the game development crowd)

- Password management programs

I'm looking to make something

- fun

- interesting

- not particularly challenging (my work is difficult - or obscure enough)

- doesn't have to be original by any means

- preferably an application (ie. not a website)

- not a hardcore learning experience, this is entirely for fun away from work

That being said, I find that the terms 'interesting' and 'challenging' don't necessarily carry the same meaning for developers as they do for other people (and often-times are interchangeable - I'm not looking for something easy or trivial). Ditto with the 'learning' part, I like and want to solve problems, but I'm not going to be getting into design methodologies or learning entirely new APIs and Languages to do this.

I ask because I think that certain projects, or types of projects, appeal to developers for particular reasons. I don't claim to know what those reasons are, nor do I care, but I

do understand that they exist and that being a developer, there is a strong likelihood that the projects that are suggested will appeal to me.

If it helps, my background or preferred medium, without reciting my resume, is c# development for windows/tablet pcs; xna; db design/programming;

*by "every", I mean, "lots of".

`c#`

Share

Of the six criteria you listed, only 2 are not subjective. – Ken Browning Mar 18, 2009 at 19:05

1 You want a list of ideas? This is NOT a discussion forum. Please make your question specific, objective, answerable. VOTING TO CLOSE as "not a real question" – Adam Davis Mar 18, 2009 at 19:22

@Ken - true, but I'm going on the premise that the subjective requirements are fairly consistent among the personality type that is drawn to software development. – Steven Evers Mar 18, 2009 at 21:23

Comments disabled on deleted / locked posts / reviews

20 Answers

Sorted by: Highest score (default) ▾

16 votes

You have XNA experience, and DB experience.

Develop a visualization tool for databases which is visually attractive and makes mining data easier.

This can start as a simple browsing tool and eventually grow to a full-fledged DB management application.

Share

answered Mar 18, 2009 at 19:06

Ben S
69.3k ● 31 ● 173 ● 214

I made one of these in OpenGL for a specific database. They impress the hell out of people. (People like UIs that they've seen on TV and movies) – geofftnz Mar 18, 2009 at 22:08

---

8 votes

If you haven't written one, knocking together an implementation of Conway's Game of Life is always an interesting challenge. It's also the sort of project that once you get it done leads to seemingly endless tinkering ("I'll just add one more feature...")

Share

edited Apr 23, 2009 at 19:45

answered Mar 18, 2009 at 19:51

Electrons_Ahoy
38.5k ● 37 ● 106 ● 127

7 votes

Graphics are always fun to play around with, try writing a program to draw a Sierpiński triangle. It's a fairly simple fractal to generate. I once made a screensaver that drew them in random places and colors on the screen.

If you are looking for something more practical, try writing an email or RSS client. Learning to interact with SMTP/POP3 at low level is pretty interesting. Same thing with RSS, it will expose you to network, GUI, and XML design at a minimum.

Share

answered Mar 18, 2009 at 19:10

DMKing
1,715 ● 1 ● 10 ● 13

+1. Both are good ideas; unfortunately I've done both :) and had a great time doing it. – Steven Evers Mar 18, 2009 at 19:28

1 Sierpinski isn't really interesting: for(x = 0; x < 512; x++) for(y = 0; y < 512; y++) if(x & y) put(x, y, white); else put(x, y, black); – Jasper Bekkers Mar 31, 2009 at 17:18

---

7 votes

Maybe write a compiler? I think every artist/craftsman should, at some point, learn how to build the tools of their art or craft.

Share

answered Mar 18, 2009 at 21:56

Rodrick Chapman
5,543 ● 2 ● 32 ● 34

+1, even though I think that is a ridiculously complicated idea. (Though I admit, I'm working on a pseudo-Scheme interpreter myself) – cwallenpoole Apr 23, 2009 at 13:20

4 votes

Write a character generator for the pen-and-paper rpg of your choice.

It should be able to be used in 2 modes: Build-A-Character (which will let you build a character from scratch) and Generate-An-NPC (which, given some guidelines, will generate a decent NPC).

This is actually a project I've been working on for a while, and I've found it fun. YMMV, especially if you don't game regularly.

Edit - figured I'd give some reasons I find it fun.

1. It's something I and my friends will actually use

2. It helps me understand the game better, knowing what decisions make a good character

3. There's some (limited) AI in the random character generation(mainly a simple decision tree with some random jumps)

4. It's overall fairly simple, a reasonable project for a one-man development team.
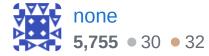
Share

edited Mar 18, 2009 at 19:43

4

votes

> I'm looking for a personal project. One of the sort that "every programmer"* makes.

How about writing your own scripting language interpreter? That's a very common thing to do, which is usually not only a highly educational experience, but also fun and interesting.

Once you are finished with something working, you'll start thinking completely differently about programming, it will change your way of thinking!

3

votes

A Sudoku game (or Connect Four, Clue, etc.). Something that relies mostly on logic, and not artificial intelligence (like chess).

Sudoku, Connect Four and Clue could all be solved with simple AI. AI and logic are not disjoint. – Ben S Mar 18, 2009 at 19:39

3

votes

How about a ray tracer? It can be very simple if you do everything by brute force. They are very fun to write and can lead to many interesting problems. Writing a ray tracer meets all your criteria =)

Share

answered Mar 18, 2009 at 20:20

bigmonachus
993 ● 1 ● 8 ● 15

1

vote

Each and every developer is different. About three years ago I asked exactly your question on some other site. It brought nothing because we all have different interests.

Maybe to make a website with focus on what you like? Like a small community within a university or your other fellows?

Share

answered Mar 18, 2009 at 19:06

User
30.9k ● 22 ● 81 ● 108

1

vote

In order to learn Ruby, I just recently made a mini DSL/XML generator for a custom XML based language I use in a program of mine. I simply turn some real text into its XML representation. It was just a fun little side project that didn't

provide much benefit in the long run but helped me to learn the language.

So... I suggest trying to create a DSL. The experience could actually prove invaluable as DSL's are pretty damn useful.

Share

answered Mar 18, 2009 at 19:07

kgrad
4,772 ● 7 ● 38 ● 57

1

vote

Write a 3D environment populated by artificially intelligent virtual beings, implemented on a public server, that we can all access using a WPF-based client.

Share

answered Mar 18, 2009 at 19:20

Daniel Earwicker
117k ● 38 ● 208 ● 286

1

vote

Personally 'interesting' strongly overlaps 'difficult', so I'm kind of at a loss ...

... but perhaps you should **take a non-programming interest you may have and try to solve a simple problem in that space**.

A friend of mine once built a model railway database so people could track the trains they had. He was trying to sell it, that didn't go so well, but it was simple and fun.

Share

answered Mar 18, 2009 at 19:21

1

vote

Why not rolling your own

1. Web MVC framework or

2. Web templating system or

3. Persistence framework (+1 if you make an OR-mapper) or

4. Interceptor-based aspect-oriented framework

But oh, those were the sins of early 2000's.

Something cloudish then? :)

Share

answered Mar 18, 2009 at 20:13

Magnus
129 🟤 4

OMG I COULD MAKE A CLOUD TO STORE ALL MI FILZ!
– Steven Evers Mar 18, 2009 at 21:21

1

vote

I had fun programming an audio player (like winamp / WMP) to learn .NET before I started my current job. There are free libraries to read MP3 files (and other audio formats), so that gets the complicated part out of the way ;)

An audio player is not too complicated and is good for learning threads, UI, and working with the file system.

answered Mar 18, 2009 at 21:42

Meta-Knight
17.8k ● 1 ● 50 ● 59

I like this idea. I've always made my own alarm clocks, but they always invoked media player or winamp in the past and I've always thought that it'd be nice to have it play the music itself. The mp3 libraries would also be a nice reference point to coding the playing part myself. – Steven Evers Mar 19, 2009 at 14:59

1
vote

I've always had fun trying to play with compilers/interpreters. I started out making pathetic attempts at my own Javascript interactive shell (I had no prior experience or knowledge of compilers/interpreters and just tried to wing it). And now I'm about to start working on writing my own lexer/parser from scratch.

Personally, I find compilers and interpreters interesting. You can use them, or their components as a start of other project ideas too if you wanted. Also, you can play with them in any language of your choice, so it might not be a bad side project if you want to learn or use a new language and aren't sure what to focus on to do it.

answered Mar 18, 2009 at 21:58

Pete
18.1k ● 4 ● 33 ● 30

1 I've always enjoyed writing a little physics simulator or the
vote canonical mandlebrot / julia set generator (parallel
extensions anyone :)

Share

answered Mar 20, 2009 at 14:54

PaulB
24.3k ● 14 ● 59 ● 76

---

0 Some kind of GPS tracker? I never made one but I believe
votes that of lot of devs do.

Share

answered Mar 18, 2009 at 19:33

tuinstoel
7,306 ● 29 ● 28

---

0 Write an touch screen game for a Windows mobile device
votes that competes with the iPhone. Maybe like,'Squash the
Bug' or something. Chicks dig that kind of stuff.

Share

answered Mar 18, 2009 at 21:00

Bill Martin
4,943 ● 10 ● 57 ● 87

---

0 Search for a good flash puzzle game and implement a
votes solver for it :)

Share

answered Mar 18, 2009 at 21:10

Tetha
4,904 ● 1 ● 18 ● 17

0

votes

Personally, it based on your own preferences. For me, it was a grade book program. Since I always had teacher friends, they always wanted me to update it. So I first wrote it in Pascal, then wanted to learn C, so I wrote a C version, then a C++ version and so on.

My daughter was playing basketball, so I wanted to keep score and I had a PocketPC at the time, so I wrote a little app for it.

My $.02

Share

answered Mar 18, 2009 at 21:34

dvallejo
1,053 ● 11 ● 26