Custom Software Development Checklist

The following is a ***checklist of best practices*** for choosing professional developers and should be included in the requirements for any web-based or mobile application.

☐ Is There An Application Architecture Phase?

It isn't wise to build a house without blueprints. Likewise, it's not a good idea to start a custom programming job without first going through application architecture. This phase collects all the concepts and functionality the client hopes to achieve. Database and object models are defined in *UML documents* which outline for developers how data should be stored, validated and associated. *Process Work-flow documents* define the desired functionality in the system. With detailed *Application Architecture*, a client can take a development project to any reputable developer and get a competitive quote. Without architecture, the cost will be nebulous, the project will be difficult to manage and the developer can not be held accountable for poor implementation.

☐ Does The Developer Use Database Schema Standards?

It's critical that the software developer have experience with designing *database schema*.

☐ ***Do They Understand Data-types?***

The use of correct the data format for each property impacts how that data is stored and how effiicently it can be queried from the database.

☐ ***Do They Design For Normalization?***

Normalization is the practice of representing data in the database without repetition and with attention to conservation of space - increasing data accuracy and reducing app logic.

☐ ***Do They Design For Relational Flexibility?***

It's important that relationship scenarios are properly defined between tables, such as "has many" and "has and belongs to many", so schema doesn't restrict app functionality.

☐ ***Do They Provide Optimization?***

Optimization is the practice of ensuring the database is configured and data queries are designed for retrieving data quickly with the lowest possible resource usage.

☐ Does The Developer Use Version Control Software?

GIT, developed by Linux creator Linus Torvalds, is the *version control system* of choice for professional developers. *Version control* allows for the tracking of changes to every line of code in a project including the date that line of code was written, by who, and if it is in potential conflict with changes from another developer. Equally important, GIT provides frictionless and easily disposable project *branching*. *Branching* allows multiple developers to easily work around one another and fearlessly try new things without worrying about unknowingly breaking the entire system. If an idea doesn't work, the developer can simply delete that branch.

Ryan Williams, President
websuasion
ryan@websuasion.com
(404) 418-8909
https://websuasion.com

websuasion

Custom Software Development Checklist (con't)

☐ Does The Developer Use Test-First Development Procedures?

Test-First Development is an approach to software development where programmers **first** write tests for the expected functionality *before* they write the actual code that make the tests pass. These tests are very specific, atomic characteristics that were defined in the architecture phase. The process is: run the test suite, a test fails, write code to make that test work, run the test suite again, that test passes but the next test fails. Repeat. This serves several purposes. It keeps the developer on task. It acts as a specification checklist. It promotes the writing of lean code. It documents the code as it is written. It makes the application easier to maintain. And it reduces the cost of change in the future.

☐ Does The Developer Use Asynchronous API Design?

The modern approach to application development for both mobile and web apps is to use an approach called *asynchronous design*. This means having many, small data transmissions between the device and cloud server instead of waiting on one very large transmission. This helps websites to load quickly, and keeps the user from having to wait for a process to complete before they can move on to something else. It also allows for very dynamic apps that can do a lot of work from a single page.

☐ Does The Developer Use Three Server Environments?

Professional developers generally have, at minimum, **three** *server environments* to keep errors in the development process from ever reaching the end-user.

☐ *Do They Have A Production Environment?*

The *Production Environment* is what the end-users see when they go to the website or open their mobile app. Only fully tested and client-approved code resides here, and care is taken back-up and properly migrate data when any changes are released.

☐ *Do They Have A Quality Assurance Environment?*

When the developers have completed work on new functionality for the client to see, the code is pushed up to the *Quality Assurance* (or *QA*) *Environment*. This resides on a server in the cloud, but is only for the eyes of QA-testers and the client. It uses a separate database from Production and usually uses fake data that can be deleted and regenerated for testing without concern of loss.

☐ *Do They Have A Development Environment?*

Developers work locally on the code in their own branches of *Development Environment*. Any code they write will not affect any other developers, the client or the end-users. Only the developer will see changes until sent to a project manager. The project manager will review the changes, merge them into the stable development branch, and ultimately release them to QA and Production environments.

Ryan Williams, President
websuasion
ryan@websuasion.com
(404) 418-8909
https://websuasion.com

websuasion