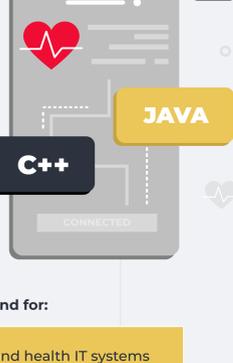


Programming Language Choice For Healthcare Solutions

The lack of physical freedom during the COVID-19 pandemic has caused an impressive app-usage boom in a wide variety of sectors, helping us fill the human connection void we were all craving for. Our social and working lives – but more prominently our health – became software-dependant.



The healthcare app market is currently registering a huge demand for:

- mHealth apps
- Telehealth and health IT systems
- Wearables and fitness trackers
- Software for medical devices

The programming language is the backbone of any type of software.

There are a plethora of available programming languages. However, there are many crucial aspects that healthcare app developers need to take into account before making a choice. Here are some examples:

There are many crucial aspects that healthcare app developers need to take into account before making a choice. Here are some examples:

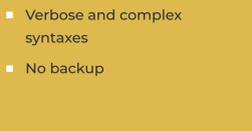


- App type (e.g. mobile, wearable, website, etc.)
- Supported device or platform (e.g. iOS, Android, Windows, etc.)
- Security features to ensure HIPAA compliance
- Maintainability to ensure long-term sustainability
- Complexity of the app
- The desired app features

When it comes to developing healthcare apps, there are no particular best programming languages. The developer's choice depends – among many other factors – on the above-mentioned aspects too.

Here are the most used programming languages for healthcare apps:

JAVA



This popular and multi-purpose programming language has been extensively used in healthcare IT. A few examples would include Elinext's Medical Practice And Billing Software or Healthcare Data Anonymization Platform.

PROS

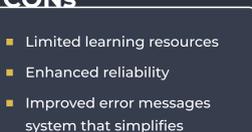
- Stable and economical to maintain
- Platform Independent (i.e. runs on any system that supports JVM)
- Cross-platform development capabilities
- Easy to learn, code, and debug
- Built-in security features (e.g. security manager, sandboxing, etc.)
- Broad user community
- Wide range of APIs

CONS

- Slow performance
- Poor GUI
- Verbose and complex syntaxes
- No backup



Kotlin



Kotlin is a relatively new – yet promising – Android programming language that comes with a series of advantages and disadvantages.

PROS

- Open-source
- Can work either natively or with JAVA
- Fully Java-interoperable
- Intuitive and succinct syntax
- Clear and compact codebase that ensures stability and consistency
- Easy to maintain
- Enhanced reliability
- Improved error messages system that simplifies finding and fixing bugs

CONS

- Limited learning resources
- Enhanced reliability
- Improved error messages system that simplifies finding and fixing bugs

SQL



Designed to interact with databases, SQL allows healthcare professionals to build clinical databases, patient tracking, create charts to analyse data, create custom reports for patients, etc.

PROS

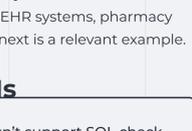
- Fast query processing
- Easy to learn
- Requires minimum coding and writing skills
- Portable
- Highly interactive

CONS

- Some versions are expensive
- Complex interface
- Users don't have full control over databases
- Constantly under threat due to the sensitive data it holds



MySQL



Thought out to manage information in relational databases, MySQL allows healthcare organizations to build complex systems such as patient portals, EHR systems, pharmacy databases, etc. The Senior Caregiving Platform developed by Elinext is a relevant example.

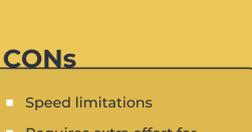
PROS

- Enhanced security and reliability
- Open-source
- Portability
- Compatible with most operating systems
- Easy to manage
- Robust transactional support

CONS

- Doesn't support SQL check constraints
- Poor debugging tool
- Development is not community-driven
- Certain stability issues

PYTHON



A fast-growing and versatile programming language, Python is currently widely used in healthcare to create ML algorithms used in drug discovery, genomic studies, medical image diagnostics (e.g. the Pneumonia Diagnosis Tool developed by Elinext), predictive analytics, etc.

PROS

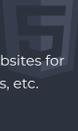
- Enhanced security (HIPAA compliance)
- Open-source
- Automation
- Readable and easy syntax
- Scalable when dealing with huge amounts of data
- Extensive community support
- Plethora of available libraries and frameworks
- Code reusability
- Interoperability and versatility
- Ideal for prototypes

CONS

- Speed limitations
- Requires extra effort for Android and iOS usage
- High memory usage



HTML



HTML is specifically used for building websites. Healthcare organizations use websites for online appointments, promote their medical teams and services, inform patients, etc.

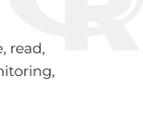
PROS

- Simple to learn
- Browser-friendly
- Open-source
- Straightforward structure
- Lightweight and fast to load
- Platform-independent
- Huge community support

CONS

- Requires extensive code
- Restricted security
- Static language
- Limited security features
- Errors tend to be costly
- Each page has to be programmed separately

R



Designed for statistical computing, R provides new ways to manipulate, read, and visualize data. For instance, it is the ideal choice for designing, monitoring, and analysing clinical trials.

PROS

- Multiple built-in statistical functions
- Straightforward syntax
- Good graphing facilities
- Massive community support
- Cross-platform support
- It can perform web scraping, data wrangling, and data cleansing
- Can be used in ML
- Compatible with various programming languages

CONS

- Slow speed
- Poor memory management
- Poor security measures



C



Ideal for healthcare organizations that use old operating systems such as LINUX or UNIX, C is a general-purpose programming language. As a practical example, we can highlight Elinext's Day Nurseries And Care Homes apps.

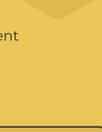
PROS

- Serves as a building base for other languages such as C++, Java, etc.
- Easy to learn
- Numerous built-in functions
- Portable
- Dynamic memory allocation
- Uses procedural programming
- Able to extend itself
- Low level of abstraction

CONS

- Complex code checking
- Lack of exception handling
- No code-reuse
- Prone to memory corruption that affects data security
- No OOP concepts
- Requires manual high-level constructs

C++



C++ is found in multiple medical apps that can range from imaging systems (e.g. Elinext's Body Tan Scan) to EHR systems, lab testing equipment, or patient information systems.

PROS

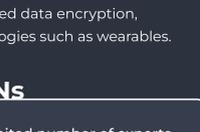
- Scalable
- Can be treated both as a low-level language and a high-level language
- Compatible with C
- Full control over memory management
- Portable
- Multi-paradigm
- Large community support
- Useful for a wide array of apps

CONS

- Uses memory-consuming pointers
- Security issues due to features such as pointers or friend functions
- No garbage collector
- Very complex to learn
- Does not support built-in threads
- Very strict syntax



SWIFT



Another young general-purpose programming language, Swift is revolutionizing the healthcare sector by delivering multiple security layers, improved data encryption, simplifying coding, and acquiring a central role in new technologies such as wearables.

PROS

- Open-source
- Improves readability and prevents errors
- Interoperability with Objective-C
- Cross-device support
- Easy to maintain
- Enhanced safety and performance
- Dynamic community support

CONS

- Limited number of experts to hire
- Poor interoperability with IDEs and third-party tools
- Compatibility issues

Taking into consideration all the above-mentioned crucial aspects and comparing the PROS and CONS of various programming languages will help you make a sensible decision.