

# FROM CONCEPT TO REALITY:

How to Build Impactful Healthcare Apps for Long-Term Success

# **2024 GUIDE**

How to Choose the Right Tech Stack for Healthcare App Development





# **FROM CONCEPT TO REALITY:** How to Build Impactful Healthcare Apps for Long-Term Success



Demand for healthcare apps is rapidly increasing, ushering in a new era in wellness and creating huge opportunities for tech companies and app developers. The challenge, however, is to create apps that not only grab attention but also succeed long-term.

In this guide, we'll explore the key components your healthcare app needs for sustained success.

# Understanding the Healthcare App Market

The term healthcare applications (also known as mobile medical or mHealth applications) represents a range of mobile apps designed to improve health and wellness, manage patients, and provide medical solutions.

The market has grown significantly over the last five years, with a recent survey revealing that over <u>63% of US adults</u> have used an app for health-related purposes in the last 12 months. The global market size for healthcare apps is projected to grow from \$80 billion to more than <u>\$861 billion by 2030</u>. There are a number of factors driving this growth, including:

- An increase in chronic diseases like diabetes
- The popularity of wearable technology such as smart bands and smartwatches
- The rising adoption of mHealth apps by healthcare professionals
- Growing awareness of the importance of maintaining a healthy lifestyle



# Key Elements of Successful Healthcare Apps

Healthcare apps often struggle with long-term adoption and sustained usage, with high postdownload drop-off rates. Healthcare apps that have bucked this trend tend to share certain characteristics. Let's explore the key elements of successful healthcare apps:

# 1. Intuitive, seamless user experience

Creating intuitive, seamless, and engaging user experiences can boost "app stickiness," which is how frequently users return and how long they engage with the app. Design sprints are often leveraged by leading companies for user experience design to quickly pinpoint opportunities, test design concepts, validate features, analyze and integrate feedback.

Medisafe Pill and Med Reminder is a prime example of an app that has leveraged user-centric design to stand out, and it is succeeding, with over 5 million downloads and a 4.6-star rating on Google Play. Medisafe provides:

- A clean, simple user interface
- Visual customization of pill shape and color, simplifying pill recognition and tracking
- Easy-to-manage reminders, allowing users to tap alerts from the unlock screen to confirm their dose was taken



# 2. Credibility, reliability, and user confidence

Building trust is essential for the success of healthcare apps, ensuring credibility, reliability, and user loyalty. By focusing on trust-building strategies, healthcare apps can gain user confidence and expand engagement across various channels, including social media, email, and notifications.

Here are some of the ways healthcare apps are establishing and maintaining trust:

#### Ensuring accessibility:

 Apps that are easily accessible across devices, operating systems, and internet speeds assures patients they can access it when needed.

# Providing credible, up-to-date information:

• Credible, up-to-date medical research, treatment options, and information demonstrates reliability.

### Offering support:

• Many successful healthcare apps provide support, such as 24/7 customer service, online chats, or dedicated support teams.

### Focusing on security and privacy:



• Ensuring app security with features like authentication, data encryption, and transparent policies reinforce users' confidence in you safeguarding their medical information.

#### Encourage ambassadors:

• Positive user reviews and high ratings are crucial for building a reputable image. Encouraging user feedback and promptly addressing concerns fosters a trustworthy relationship with users. Doximity, a networking platform for medical professionals, builds trust with their users through various measures, including:

- A physician-led review process to evaluate comments flagged as potential misinformation
- Articles from reputable medical and science publications, tailored to each user's medical specialty and preferences
- Community guidelines that foster a secure and respectful environment

# 3. Valuable content based on user data

To deliver real value to their users, successful apps leverage user data to tailor their content, fostering deeper engagement and long-term usage. They do this in a number of ways:



# Tapping into behavioral insights:

• Tracking and understanding user behavior, such as content interactions, usage patterns, and preferred features

#### Ensuring content relevance:

• Leveraging data to curate information, treatment options, and health advice

### Adapting in real-time:

• Using real-time user data to adapt and serve timely notifications

The popular app Ada utilizes user input and AI chatbots to deliver tailored health assessments, medical guidance, and advice. This approach is resonating with users. The app is rated 4.8 on iOS and 4.7 on Google Play, with over 13 million users.

# 4. A personalized user experience

Personalizing the user experience is more than offering relevant content; it means empowering users with choices and the ability to tailor the app to their preferences and requirements. This can include:

#### Adaptive interfaces:

• Allowing users to personalize the user interfaces allows easier access to pertinent information.

### Dynamic content suggestions:

• Al-powered content suggestions based on user behavior can enhance user engagement.

#### Tailored health insights:

• Delivering tips or recommendations based on users' health conditions and activity levels ensures relevant, actionable content.

The WebMD app, for example, gives users options to tailor the dashboard to their requirements. It then provides individualized suggestions based on users' conditions, preferences, and activity levels.



# 5. An engaged, supportive community

Establishing a vibrant community can significantly amplify user engagement. Successful apps have accomplished this in a number of ways, including:

- Enabling users to connect with each other to foster a sense of camaraderie
- Providing a platform for users to share progress and cultivate a supportive environment
- Introducing challenges, groups, or forums to stimulate engagement and encourage interaction



Strava, for example, is a popular fitness app that allows users to connect, share progress, and participate in challenges. The aforementioned WebMD has also done well by providing a platform for users to engage in health discussions, share personal experiences, and access expert advice.

# 6. An effective rewards system

A thoughtfully designed rewards strategy can serve as a powerful incentive for users to remain engaged, fostering continued usage and loyalty. Successful healthcare apps have achieved this in a number of ways, including:

- Gamification to incentivize behavior, such as medication adherence or exercise routines
- Offering rewards, badges, or points for health-related goals or medication adherence
- Introducing rewards for consistent behaviors to promote healthy routines and practices

Mango Health is an example of an app that has incentivized medication adherence through gamification techniques, fostering user engagement by rewarding users for maintaining healthy habits.

# 7. Technology that enables scalability

App creators and developers are increasingly embracing cloud-native, software as a service (SaaS), and infrastructure as a service (IaaS) technologies. A cloud-native approach can reduce complexities, minimize coding and quality assurance work, and generally expedite development.

Cloud technologies also offer unparalleled scalability, enabling healthcare apps to expand beyond the minimum viable product stage, accommodate increased user loads and seamlessly integrate additional features. Thrillworks has long championed the use of design sprints and cloud-native approaches, streamlining development processes for numerous clients, and ensuring efficiency and scalability.

# 8. The right focus to streamline regulatory complexities

Choosing the right focus for your app can help simplify long-term regulatory compliance. Focusing on preventative health and wellness, for example, can avoid many of the complexities associated with FDA approval and prescriptions.

Strava, for example, is a non-clinical app that has proven to be very successful with a focus on fitness and overall wellness, and this avoids the regulatory landscape.



# Unlock your App's Potential with Thrillworks

For over two decades, Thrillworks has partnered with leading organizations across North America, creating technology-driven, user-centric apps.

Our approach emphasizes collaboration and obsessive, value-based prioritization to identify the most valuable digital opportunities. We then design an accelerated path to bring your app to life, covering the life cycle of all things digital, from visioning and prototyping to realization and optimization.

In short, we turn ideas and experiences into marketable, successful products.

Contact us to fast track your app's development and unlock its potential.



711 Oval Court Burlington, Ontario L7L 6A9 Canada <u>thrillworks.com</u>





# **2024 GUIDE**

# How to Choose the Right Tech Stack for Healthcare App Development





Demand for healthcare apps (aka mobile medical or mHealth applications) is growing rapidly, creating huge opportunities for tech companies, app developers, and medical practitioners.

A survey by Insider Intelligence found that 63% of American adults have used an app in the last 12 months for health-related purposes, including for nutrition, weight loss, and sleep. This represents a 6% increase in health app users since 2018, and overall, the global market size for healthcare apps is projected to grow from \$80 billion to more than \$861 billion by 2030.

To take advantage of this exciting market, app creators and developers have to create apps that stand out, with exceptional user experiences, and the capacity to adapt and scale in response to evolving market trends. Choosing the right tech stack is a foundational step that ensures your project launches on solid footing and succeeds in the long-term.

In this guide, we'll take a closer look at the concept of a tech stack, why it's important for software and app development, and how you can choose the right tech stack for your healthcare app.



# What is a tech stack?

A tech stack, short for technology stack, is an amalgamation of technologies that enables companies to build and operate websites, applications, or projects. This amalgamation is also often referred to as a solutions stack, technology infrastructure, or data ecosystem. The term "stack" is widely used, however, because it helps describe the layering or "stacking" of multiple technological solutions to create a fully functional system.

A tech stack is generally composed of two main parts:

- The front-end, which involves the client-facing set of tools that developers use to create and enhance the user experience. Common frontend tools include HTML, CSS, and JavaScript.
- The backend: Server-facing tools that support the inner operations of your site, including programming languages, frameworks, databases, and servers.

# Why is a tech stack important for healthcare apps?

The importance of a tech stack in healthcare app development cannot be overstated; it serves as the foundation for your app's functionality, scalability, and future growth. A well-chosen tech stack will play a pivotal role in shaping the capabilities, integrations, and expertise required to manage your healthcare project effectively.

# The evolution of modern tech stacks for healthcare apps

Modern technology stacks tailored for healthcare apps must prioritize scalability to accommodate diverse devices, fluctuating traffic, growing data volumes, and dynamic user interactions. Without this scalability, your healthcare app may struggle to adapt to the evolving needs of patients, healthcare providers, and regulatory requirements.

To address these scalability challenges, modern tech stacks for healthcare apps have evolved to include a comprehensive range of components, including:

- Performance monitoring tools to track, review, and optimize each layer.
- Business intelligence to analyze data from various healthcare sources for decisionmaking and resource allocation.
- API services for easy integration and data exchange between different components and tools every tool in your stack.
- Containerization to efficiently manage and segment data while ensuring compliance with data privacy regulations.
- Behavioral and product analytics tools for insights into patient behavior and engagement patterns for personalized care delivery.
- Cloud services to provide flexible, scalable infrastructure for hosting and managing healthcare apps securely.



# Tailoring your tech stack to your project needs

To avoid potential pitfalls, it's crucial to thoroughly understand the unique requirements of your healthcare project. Here are some tailored questions to consider when selecting your tech stack:

- What is the scope and scale of the project in terms of functionality and user base, considering factors like patient volume and data complexity?
- How complex are the features, integrations with medical systems, and technical requirements?
- What are the anticipated growth and scalability needs?
- What are the existing skills and expertise of the development team? Do they have experience in healthcare-specific technologies and regulatory requirements?
- What is the expected timeline for launching the product or specific features?
- What are the security and compliance needs specific to healthcare data, such as HIPAA, GDPR, or other regulatory standards?
- What level of ongoing maintenance and support will be required post-launch?
- What is the budget and what are the cost considerations for tech choices?

It's also important to recognize that the objectives of your project may require a different focus or need. As an example, here are some of the important considerations for various healthcare project types:

- Static websites: Relying on fixed content, static websites require swift loading times and robust security measures.
- Dynamic sites: Require interactivity, efficient data processing, and scalability, ensuring they can adapt to evolving content and user interactions seamlessly.
- Web apps: Depend on great user experiences, including real-time updates and scalability to handle intricate functionalities.
- Portals: Projects revolving around portals need user authentication, convenient content management, and the ability to customize the experience effectively.
- Mobile apps: Successful healthcare apps are increasingly compatible with different platforms, can support offline capabilities, and have the capacity to optimize efficiency, responsiveness, and overall speed.

# Exploring frontend frameworks for healthcare apps

Each framework has its unique strengths, and the choice depends on your project's specific requirements and your development team's expertise. When selecting a frontend framework, one crucial consideration is its integration with a headless content management system (CMS), like Contentful, Storyblok, or Contentstack. Given the dynamic nature of content in healthcare apps, which often require updating information or goals quickly, robust CMS integration is becoming increasingly essential.

Here is an overview of some prominent frameworks and their characteristics:

#### Gatsby

Gatsby is a web development framework that is known for its proficiency in static site generation (SSG), which eliminates the need for server-side processing. It is especially well-suited for healthcare projects where a large number of pages remain static, such portfolio and documentation sites.

Choosing Gatsby can simplify resourcing and the process of assembling a development team, as it leverages React (which many developers are familiar with), and provides a number of tools and features that enable efficient debugging and deployment.

In addition, Gatsby uses GraphQL to ensure precise data fetching and prevent overfetching from different sources. It is also noteworthy for its robust Search Engine Optimization (SEO) features, including streamlined metadata setup for efficient search engine crawling and indexing, and the fact that it seamlessly integrates with popular Content Management Systems (CMS), such as Contentful.

While Gatsby Cloud has been integrated into Netlify, we at Thrillworks would recommend using your own cloud solution (AWS, Azure, etc.), particularly for healthcare projects requiring incremental changes.



### Next.js

Next.js offers a unique blend of power, flexibility, and developer friendliness, making it a commonly used framework. Next.js has great support for static site generation (SSG) and incremental static regeneration (ISR), allowing you to update static pages without necessitating a full site rebuild. It is, however, also an excellent choice for dynamic websites employing server-side rendering (SSR) and client-side rendering (CSR), and web applications.

Like Gatsby, Next.js is built on React and provides a number of tools and features to enable a efficiency for developers, including:

- A dynamic community to support developers
- Comprehensive error-handling mechanisms
- Smooth integration with different tools and services

Next.js offers a way to handle backend tasks using serverless functionality, simplifying server management by hosting the app on the cloud and automatically scaling to handle uncertain traffic. The framework's feature for creating API routes within the Next.js app simplifies backend integration and offers flexibility.

Next.js recommends utilizing your preferred cloud solution (AWS, Azure, etc.) for the backend, enabling flexibility in selecting databases based on data types and specific considerations such as consistency, availability, and partition tolerance.

#### Angular

Led by Google, Angular is a comprehensive, TypeScript-based framework recognized for its robust features and scalability, making it suitable for large-scale healthcare applications. The framework provides a comprehensive platform that supports teams and applications as they grow, including a variety of features right out of the box.

With a more opinionated structure and a built-in separation of logic, Angular offers a complete solution for building robust web applications in the healthcare sector.

#### React

While React is a popular and lightweight JavaScript library, it is often compared with frameworks due to its robust capabilities and the fact that it acts as the foundation for many frameworks.

With a large community and strong developer support, React is an excellent choice for healthcare projects seeking flexibility and reliability. It also provides a declarative approach to building user interfaces (UIs), making the code more predictable and easier to debug, and supports encapsulated components that manage their own state, allowing you to build more complex UIs.

# Exploring mobile frameworks



There are two prominent frameworks that excel in mobile development:

#### React Native

React Native is a framework that allows developers to build mobile applications using React. It enables the development of cross-platform mobile apps, meaning developers can write code once and deploy it on both iOS and Android platforms.

React Native boasts a large community of developers already familiar with React, making the transition to mobile development smoother compared to other frameworks. This familiarity ensures a robust ecosystem of resources and support. Key features include:

- Native feel: React Native provides a native feel for applications by leveraging native APIs by default. This enables developers to create apps that easily integrate with the look and feel of the user's device.
- Flexibility: Increased flexibility for customization, allowing developers to tailor the user experience according to their specific project requirements.

Something to consider is that React Native requires a JavaScript Virtual Machine (JVM) as an additional layer, leading to increased memory requirements and app startup time. However, the framework compensates for this with its extensive community support.

#### Flutter

Flutter takes a different approach to mobile development, using an extensive set of Widgets that cover visual, structural, platform, and interactive elements. This reduces reliance on third-party libraries and packages, providing a more comprehensive development environment.

Flutter also compiles apps to native code, which ensures that Flutter applications can match the speed and responsiveness of native apps.

There are two key considerations to keep in mind with Flutter:

- There may be a learning curve for developers unfamiliar with Dart, the programming language used in Flutter.
- Unlike React Native, Flutter does not leverage native APIs for UI elements by default, which makes achieving a truly native look more challenging.



# Backend and database considerations for healthcare projects

The choices you make regarding backend and database infrastructure can profoundly affect your project's performance and scalability. Choosing the right combination ensures smooth functionality, scalability, and cost-effectiveness, which are crucial when attempting to build an app that grabs attention and succeeds long-term.

Here are some considerations you should keep in mind:

- Cloud solutions to boost flexibility: Utilizing cloud solutions such as AWS and Azure is essential due to their scalability and cost-effectiveness. These platforms offer a flexible infrastructure that can easily adapt to evolving needs, ensuring reliable performance and data security.
- Backend-database agnosticism: Healthcare app backends and databases should be agnostic to frontend technologies, ensuring smooth integration and interoperability. Modern architectures, often based on REST APIs communicating with JSON, facilitate transitions and interoperability, although challenges may arise with legacy systems using outdated technologies like SOAP APIs communicating with XML.
- Consistent languages to improve efficiency: If your developers are full-stack instead of separated into frontend and backend roles, having the same programming language for both frontend and backend can reduce learning curves and streamline development processes.
- Robust data encryption: Implement strong encryption for in-flight and at-rest data, especially for sensitive patient information. This will limit unauthorized access, and ensure security and compliance with regulations like HIPAA and GDPR.



# Database types:

The choice of a database can be impacted by a number of considerations, including the nature of the data, consistency, availability, and partition tolerance. There are two primary categories:

- NoSQL databases: NoSQL databases are well-suited for healthcare apps dealing with diverse and unstructured data. They offer excellent scalability and performance, making them ideal for handling vast amounts of patient data and medical records.
- SQL databases: SQL databases are preferred for healthcare apps requiring strong data integrity and relational structures. They ensure consistency and reliability, crucial for managing complex patient relationships and medical data.

# Examples of databases

# AWS databases

- DynamoDB: A NoSQL database offered by AWS, DynamoDB is designed for scalable highperformance applications. It excels in scenarios where quick and predictable responses are crucial.
- Relational database service (RDS): RDS is a managed relational database service by AWS, supporting multiple database engines like MySQL, PostgreSQL, and SQL Server. This flexibility caters to various project requirements, ensuring a reliable and scalable relational database solution.

#### Other popular databases:

- MongoDB: A widely used NoSQL database, MongoDB is known for its flexibility in handling unstructured data and scalability. It is suitable for projects with evolving and dynamic data structures.
- PostgreSQL: PostgreSQL is a powerful open-source relational database with a focus on reliability and robust features. It is an excellent choice for projects that demand strong data integrity and consistency.

# Unlock your App's Potential with Thrillworks

For over 20 years, Thrillworks has built lasting relationships with some of the world's most respected organizations — accelerating the development of human-centric, technology-driven healthcare products and services.

We turn ideas into experiences and experiences into marketable products and services that are valuable to customers and businesses alike.

Contact us to fast track your app's development and unlock its potential.



711 Oval Court Burlington, Ontario L7L 6A9 Canada <u>thrillworks.com</u>



