HOW TO

# CREATE USER STORIES AND EPICS

FOR SOFTWARE DEVELOPMENT

# INDEX

HOW TO
# CREATE USER STORIES AND EPICS
FOR A SOFTWARE DEVELOPMENT PROJECT

# INDEX

HOW TO

# CREATE USER STORIES AND EPICS

FOR A SOFTWARE DEVELOPMENT PROJECT

***Software development user roles, user stories and epics are vital to the success of a development project, whether it entails custom software, a mobile app, an AI platform or another Digital Transformation project. These mission-critical components of a Business Requirements Document (BRD) But how do you create user roles, user stories and epics for a development project?***

---

## What Are User Roles, User Stories and Epics?

Before you set out to create user roles, user stories and epics for a software development project, you must understand the definition for each. Collectively, they will guide the architecture of a platform, which — if it is to be successful — must be purpose-built to accommodate its users.

### What Are User Roles?

User roles — also known as user personas or user types — are created to represent the different individuals who will be interacting with the platform. User roles are based upon the unique needs, behaviors, intentions and access levels within the software, app or web portal.

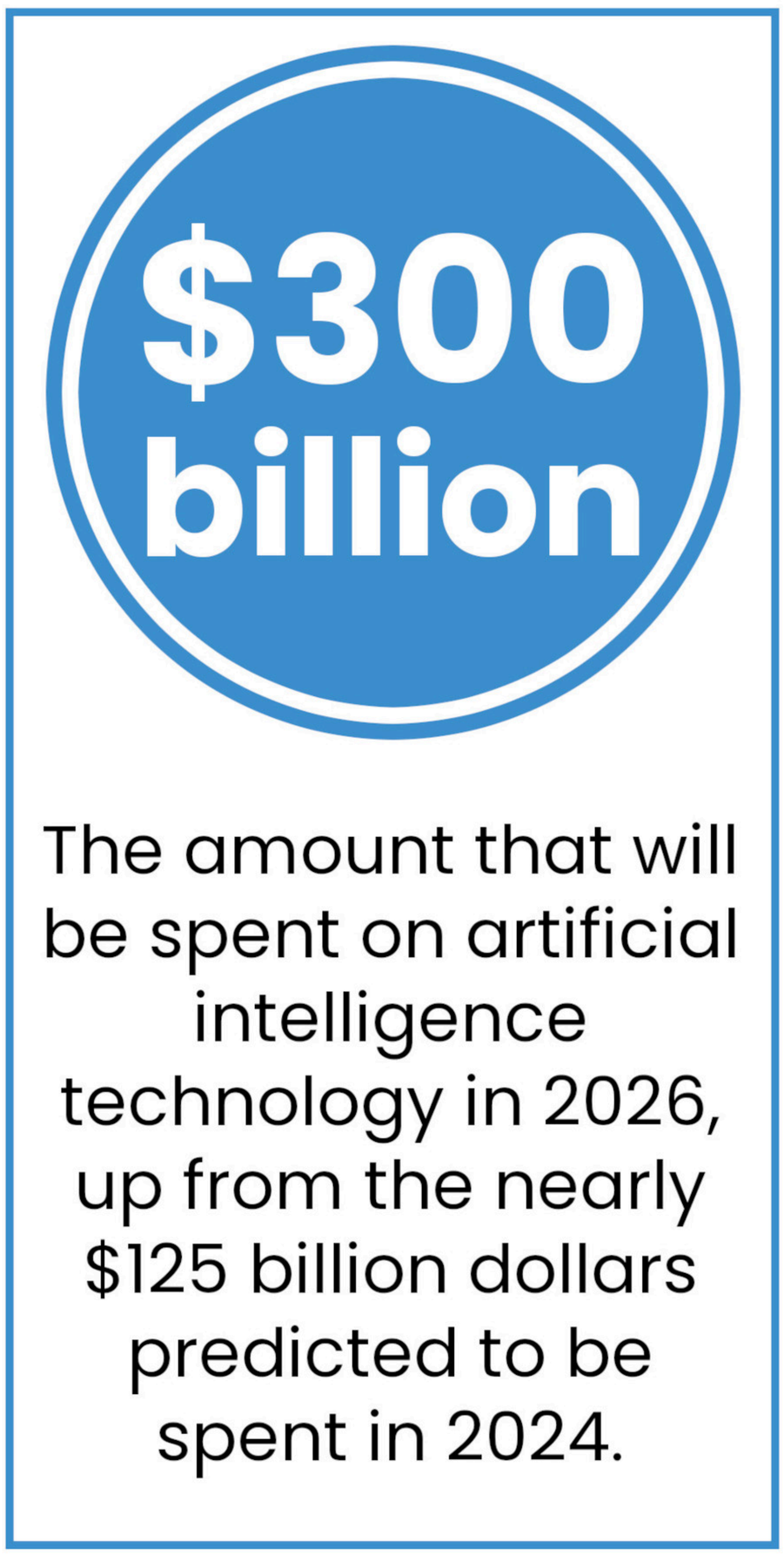
### What Are User Stories?

User stories — also known as user journeys — are short, simple descriptions of feature functionality from the perspective of a user, as defined for one of the various user roles. In software development, user stories focus on the "who," "what" and "why" of the user's experience, capturing the essence and intentions of each user role's interaction within the platform.

User stories are designed to be easily understood by members of the business team and the technical team.

### What Are Epics?

Epics are large, overarching compilations that include multiple user stories for a wide variety of user roles. Epics commonly represent large piece of functionality with




The amount that will be spent on artificial intelligence technology in 2026, up from the nearly $125 billion dollars predicted to be spent in 2024.

multiple features which are often associated with a high level of complexity. In fact, these features and functionalities are often too complex to be completed in a single development sprint. You can think of epics as a “container” for related user stories, serving as an effective management mechanism for large segments of a development project.

# How to Create and Define User Roles

User roles can be defined and created using three simple steps.

## Identify All Potential Users

Start by identifying all of the different types of users who will interact with the system. This may include end-users, administrators, managers, and even third-party partners or clients.

## Create Personas

For each user role, you must develop a persona. Personas include demographic details, user goals, pain points and behaviors. This helps in understanding the context in which these users will be interacting with the platform.

## Prioritize Roles

Not all user roles are created equal. Prioritize them based upon how critical they are to the system’s success, what type of information they will have access to and the frequency with which this user type will interact with the system.

# How to Create User Stories for Software Development

User stories are an essential mechanism for conveying a user type’s intentions and needs from a platform, making them critical for informing the development and quality assurance teams of a platform’s various functionalities and features. But how do you write software development user stories? Consider the following simple outline.

## Use the Standard Format

The most common format for a user story is as follows.

*As a [user role], I want to [objective], so that [reason].*

# Key components of a user story include:

- **User Role –** Describe the user who wil l benefit from this feature.
- **Goal –** What is the user's objective? What does the user wish to achieve as they interact with the platform?
- **Reason –** Why does the user want to achieve this goal? What is their motivation?

# A User Story Example

A user story for software development should provide guidance on what features and functionalities that the associated user role would need to access. The following is a user story example that meets this criteria.

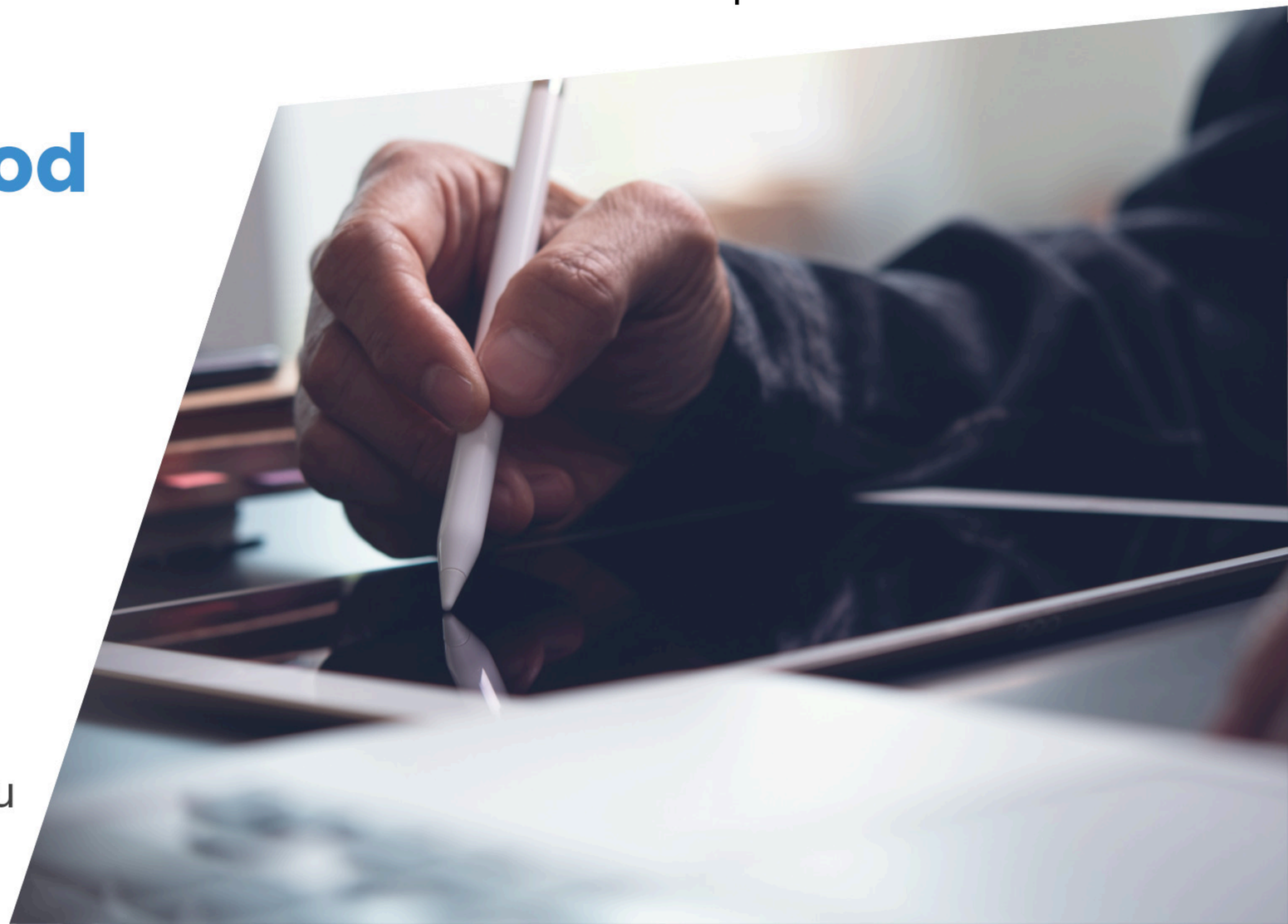
*"As a Marketing Manager, I want to generate a report of daily website traffic and individual page performance data, so that I can assess the performance of our marketing campaigns."*

# Tips for Writing Good User Stories for Dev Projects

To write suitable user stories for your development project, you must be concise and specific, while clearly depicting the user's motivations and objectives while using the platform. Here are a few tips to consider as you begin developing user stories.

- **Keep It Concise –** User stories should be short and to-the-point. They are not meant to address every potential need or functionality. They exist to provide sufficient information to cover a specific need or feature.
- **Focus on Value –** Each user story should clearly convey the platform's value to the user and the business. Ask yourself, "How does this feature help the user to achieve their goals?"
- **Include Acceptance Criteria –** While the user story itself is brief, it's essential to include acceptance criteria that define what "finished" looks like. These are conditions that the system must meet in order for the user story to be considered complete.

After you've written your user stories, re-review these tips to ensure that they're accurately and fully reflected in your work. With these characteristics, your user stories will be well-positioned to drive the development of your new software platform, mobile app or web portal. latform and a companion mobile app, you may need to develop a separate set of user stories for each platform.

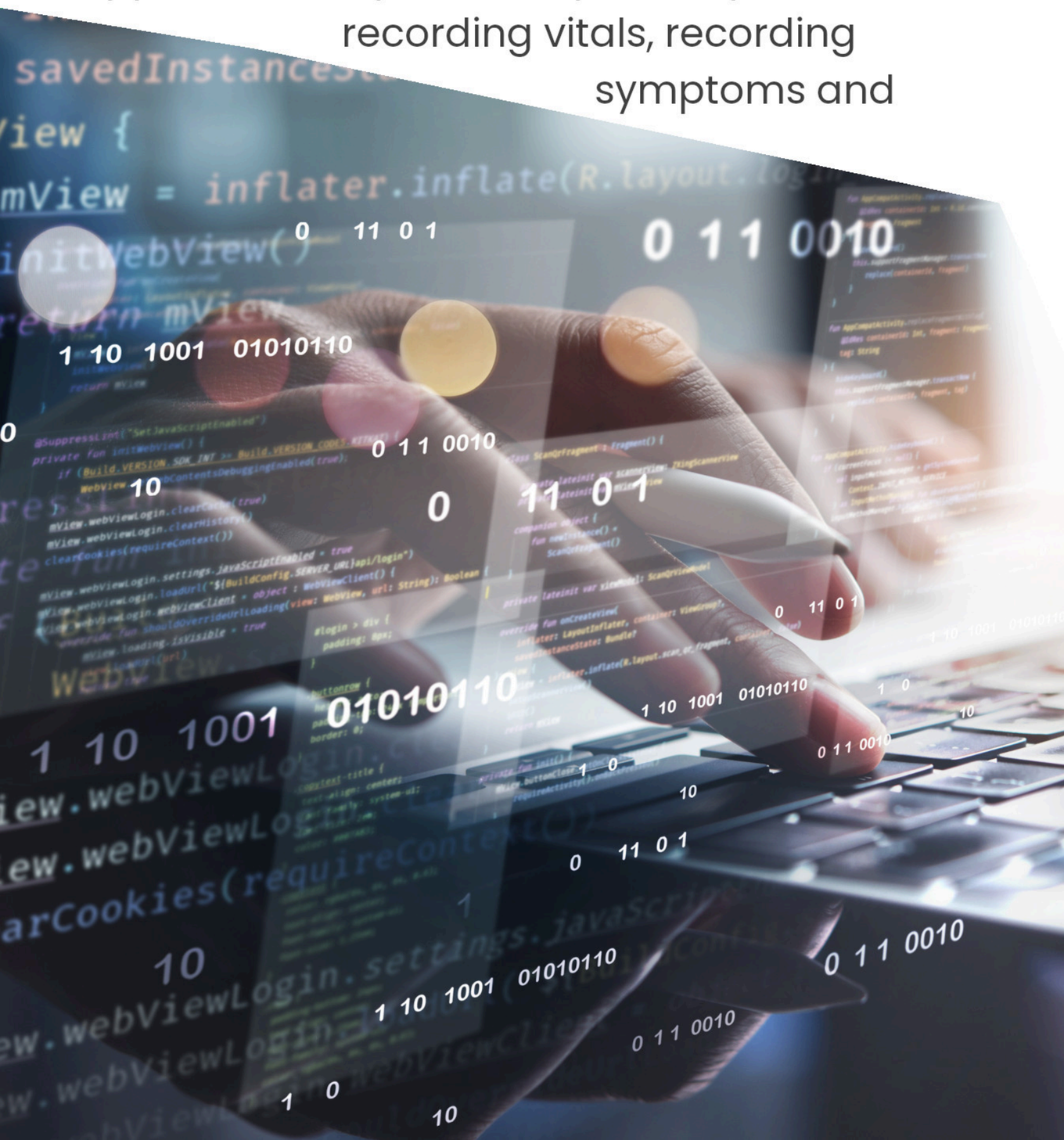


# Adapting User Stories to Multiple Platforms

If you're developing multiple platforms — such as a desktop software platform and a companion mobile app, you may need to develop a separate set of user stories for each platform.

Often, your user stories can be cloned and adapted to the secondary or tertiary platform. Keep in mind that not every user will be using every platform and their motivations and goals may differ somewhat according to the platform that they're using.

For instance, if you have a platform for a medical clinic, a nurse user may primarily perform charting and messaging on the desktop platform. That same user would not need full charting capabilities on a mobile app, which may be used primarily for recording vitals, recording symptoms and



and messaging. Your user stories would need to reflect this differential.

# How to Create Epics

Epics are essential for a successful development project, as they serve as a compilation of multiple user stories, illustrating the capabilities, functionalities and features that are available to the various user roles within a platform.

As previously mentioned, epics serve as a "container" of sorts of your user stories, making it easier to manage large segments of a Digital Transformation project. A single project will have multiple epics, based upon logical "clusters" of features and functionalities.

But how do you create epics? Consider the following facets of developing an epic.

- **Identify Major Features to Plan Your Epics –** Begin by identifying the major features or components of the system that are too large to be tackled in a single sprint. These features will be explored in your epics. A given project may have 5, 10 or even 50 epics for larger enterprise level projects, with each depicting a logical segment or grouping of features and tools.

- **Compile User Stories into Epics –** Consider each feature or functionality that's mentioned in your epic. Then, pull together all of the associated user

stories from all user roles that are associated with your epic's features and functionalities.

- **Prioritize and Organize User Stories Within the Epic -** Organize user stories within each epic, based upon the priority and logical sequence. This ensures that the most critical aspects of the epic are addressed and developed first.

- **Maintain Flexibility Within Your Epic Structure -** As you develop your epics, be specific enough to provide a solid direction to developers. At the same time, epics should be broad enough to allow for flexibility in how the various features and functionalities are implemented within the platform. Stated differently, offer actionable insights, without micromanaging the development of the platform's architecture, amongst other aspects..

## The Relationship Between User Roles, User Stories and Epics

To develop successful user stories, user roles and epics, it's important to understand how they are interrelated and how they are intertwined with the greater development process.

- **User Roles Are Foundational:** User roles serve as the foundation and user stories are built upon your user roles.. Before your development team writes a single line of code, you must have a solid understanding who your users are and what they need to accomplish as they use the platform. User roles are foundational and they must be well-defined before you can delve into a discussion of system functionalities and features.

- **User Stories Are Developed Based Upon User Roles -** Each user story is written from the perspective of a user role, so the latter drives the development of the former. By basing your user stories upon the already-developed user roles, you ensure that the functionality you're describing is relevant and valuable to a specific group of users.

- **Epics as Containers for User Stories -** Epics serve as "containers" that allow you to pull together related user stories into a single grouping. This makes it easier to manage large projects by keeping related functionalities organized under a single umbrella in the form of an epic.

# The Importance of Iterative Refinement

Iterative refinement is essential for success. This process entails creating multiple iterations of each user role, user stories and epics as you gain new insights and feedback throughout the development process. In fact, it's commonplace for a client to develop multiple iterations of their user roles, user stories and epics before development actually begins.

The act of putting everything down on paper can lead to new insights and realizations that may warrant revision and refinement to ensure that everything aligns with user needs and the project's goals.

Iterative refinement is a natural part of the development process. It is prudent to keep prior iterations of your user roles, user stories and epics on-file in the event that you need to reference them at some point down the road.

# Developing Acceptance Criteria for User Stories

Acceptance criteria must be included for each and every user story. Acceptance criteria refers to specific conditions that must be met in order for the story to be considered complete. For example, in the case of a healthcare clinic platform, your acceptance criteria may be as follows for a physician's user story..

- The user can add new patient chart notes, including text, images and test results.
- When viewed, the chart notes update dynamically without the need for a page reload.
- Patient chart notes are searchable by date and by keyword.
- The user can view historical chart notes and add updates.

Acceptance criteria are important for providing you with a checklist of sorts, allowing your team to determine when development has reached a point of completion for that particular portion of the project.

# Examples of User Roles for a Development Project

Each user needs slightly different functions and capabilities within a platform. A well-developed user role must provide a user with access to the tools and data that they need to succeed and without clutter from tools, data and features that fall outside of their scope of work.

User role is used to determine what data a user can access and, conversely, what data is off-limits to individuals who are logged in under a particular user role.

For example, in the case of a medical clinic's customized ERP platform, patient users would have a "read only" view of clinical notes, test results, diagnostic imaging and upcoming appointments. This user role's level of access would be very limited.

The clinic's clerical staff would have a far more comprehensive view, but users in this role may have a limited view of HIPAA-protected patient data and they would not be permitted to edit or create patient medical records or clinical notes.

Similarly, a CNA cannot have access to a digital prescription pad, nor can they edit others' clinical notes.

Meanwhile, a physician needs access to virtually all of a patient's historical medical data, since an individual's past medical history can inform today's diagnosis. But a physician generally doesn't need their screen cluttered up with insurance and billing information, so that data would be omitted from view for an individual who's logged in under a physician user role.

Here's a look at a few different user roles that may be developed for a healthcare software platform, with information that can be developed into one or more user stories and ultimately, epics.

# User Role Development into User Roles

| User Role | User Role Description |
| --- | --- |
| **The Patient User Role** | Patients will log onto a web-based patient portal or mobile app to view upcoming appointments, notes from clinicians, x rays and other diagnostic imaging, test results and prescriptions. This user role will typically have limited, primarily "read only" access within the portal, with the exception of a messaging feature that the patient can use to communicate with their healthcare providers. Patients may also have the ability to input vitals information as part of a remote patient monitoring program, for example. |
| **The Physician User Role** | Physicians represent another user role and they would need a highly-granular view that includes access to the mobile app and web portal. They will need full access to a patient's medical records, in addition to the ability to add notations on the patient and their condition(s), order and view tests and diagnostic imaging, write new prescriptions and view past prescriptions, add referral information and beyond. |
| **Ancillary User Roles** | In a medical clinic setting, there are countless ancillary users, so several user roles may be developed to accommodate the various subtypes. Ancillary medical staff may include medical assistants and certified nursing assistants (CNAs). You also have ancillary clerical and office staff who handle scheduling, billing and insurance-related tasks. These team members represent additional ancillary user roles. |
| **Admin User Roles** | Admins are essential for maintaining the ongoing integrity of an enterprise platform. A healthcare software system may include multiple admin user roles; one admin may have access to the clinical regions of the platform, including HIPAA-protected data, whereas another type of admin may focus on non-medical areas of the interface, such as those regions that are related to clerical tasks, insurance and accounting. "Super admins" may have global access. |

## User Roles and Industry-Specific Regulations Such as HIPAA

In the context of a development project for a medical clinic, there are added considerations related to HIPAA, which is intended to protect patient data privacy. As such, each user role within a platform must be architected so as to grant access to HIPAA-protected data only when it's absolutely necessary.

The same is also true for financials and accounting data. Virtually any business type will want to restrict access to financial information and other sensitive data. This is best achieved by taking the time to carefully architect user roles and user stories, which are compiled into epics within

the context of your new platform.

The general rule of thumb is that a user should only have access to the data that is required to perform their job duties; nothing more and nothing less.  It's a good example of how industry-specific regulations such as HIPAA can have a significant impact on the development of a platform's user roles.

# Examples of User Stories by Industry

Here are a few examples of user stories for a variety of different industries. Remember, each project will have multiple user stories, which are then consolidated into epics, which help guide the development process from start to finish.

## E-commerce Website User Story

"As a customer, I want to filter products by price so that I can find products that are priced within my budget range."
Explanation: This story focuses on a specific feature (price filtering) that helps the user achieve their goal of finding affordable products.

## Banking App User Story

"As an account holder, I want to receive notifications of transactions so that I can monitor my account activity in real time."
Explanation: This story addresses the need for real-time monitoring of account activities, which helps users stay informed about their financial status.

## Educational Platform User Story

"As a student, I want to track my progress on courses so that I can view my grades and determine how close I am to achieving my educational goals."
Explanation: The story is about providing a feature that allows students to track their progress, which is essential for providing motivation and helping students to manage their studies and schedules with greater efficacy.

## Healthcare User Story

"As a physician,  I need the ability to quickly view patient history so that I can make rapid, informed decisions during consultations."
Explanation: This story focuses on enabling physicians to access patient history quickly and efficiently, which is crucial for effective medical care.

## Social Media  User Story

"As a user, I want to upload photos and tag friends so that I can share my experiences with the people who are within my network."

Explanation: This story is centered on a common social media feature, enhancing the user experience by allowing them to share and connect with fellow social platform users.

## An Example of an Epic

As mentioned above, user stories are collected into larger, broader stories, called epics. By combining user stories into an epic, you have easy-to-manage “bites” of information that are easy to read and, if needed, modify. Here’s an example.

**Epic**: “As an online shopper, I want to manage my shopping cart so that I can add, remove, and purchase items easily.”

**Related User Stories:**

As a shopper, I want to add items to my cart so that I can buy multiple products at once.”

“As a shopper, I want to remove items from my cart so that I can adjust my purchase before checking out.”

“As a shopper, I want to apply discount codes to my cart so that I can save money on my purchase.”

## User Stories: Helping You Maintain a User-Centric Development Process

User stories allow you to maintain a user-centric development process, ensuring that the team builds features that provide real value to the end users. User stories are concise, easy to understand and they can be adapted as needed, making them a cornerstone of agile project management.

User roles, user stories and epics are critical components of a project’s Business Requirements Document or BRD. This document serves as an essential guide to the actual architecture and development of a platform. Informing virtually every aspect of these processes.

At 7T, we take a problem → solution approach to development and a BRD is the first step in the process of creating a user-friendly platform, with all of the features and functionalities that your organization needs to succeed. Your completed BRD will guide the process of developing a platform that solves problems, resolves inefficiencies and helps you to overcome challenges — all while delivering an exceptional ROI both now and in the future.

https://7T.co

## Houston, TX
## Regional Office

**1334 Brittmoore Road, Suite D**
**Houston, Texas 77043**
**+1 (832) 632-4869**

## Dallas
## Headquarters

**16803 Dallas Parkway**
**Suite 300**
**Addison, Texas 75001**
**+1 (214) 299-5100**

## Charlotte, NC
## Regional Office

**2115 Rexford Road**
**Suite 570**
**Charlotte, NC 28211**
**+1 (980) 350-5100**