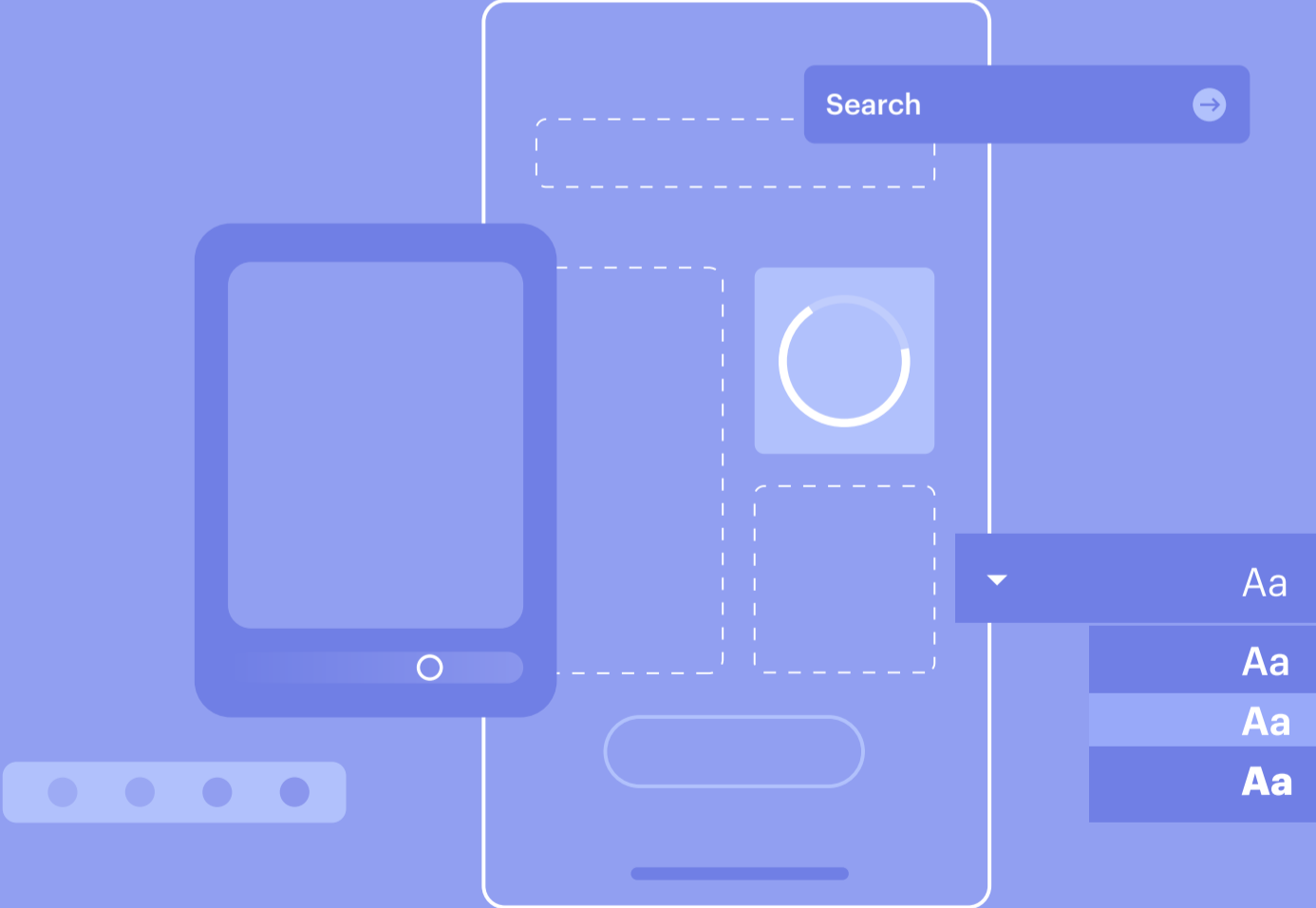


How to generate software with a good design

Whitepaper



Index

Introduction	3
Concepts you should know	4
Basic steps to turn a good design into software	6
Software and design: the most common challenges	6
Unity makes strength!	7
How to generate software with a good design in GeneXus	9
How to create a Design System in GeneXus	11
Recommendations to import external designs in GeneXus	11
Tips for optimization with Atomic Design	12
GeneXus 18 and Unanimo: the key to designing and modeling complex systems	13
What does Unanimo consist of?	14
Craving for more?	16

We live times of swiftness, where changes and technological evolution are permanent.

Seeking to surf through such challenges, companies are increasingly demanding more and more software. However, they don't ask for any software but for that software that will enable them to securely transform obsolete systems or to rapidly and easily generate and maintain digital ecosystems that must be future-proof. In this crusade, design plays a central role.

In the world of software, design relates not only to aesthetics but also to the flow of interactions between users and various screens and platforms. Everything must be clear, coherent and consistent for an application to have a uniform appearance.

Today's applications may no longer be thought of as products created by software developers but rather by multidisciplinary teams that also include visual and interaction designers.

According to **Gartner** Consultancy, [it is important to create and maintain efficient practices of DesignOps](#) (design operations) to ensure operational alignment, in-depth cooperation, and the ongoing innovation of products.

"Companies that work with [DesignOps](#) have competitive advantages over those that don't resort to these practices. Overcoming the gap between design and development should not remain a pending issue. GeneXus offers users great tools and practices to tackle the challenge in the best manner possible," affirmed [Gabriel Simonet](#), **Marketing Director** (CMO) at [GeneXus](#).

This whitepaper is a practical guide for generating quality software quickly and with a good design.

Concepts you should know

User Experience

This expression, usually referred to as **UX**, dates back to 1993, when Apple created the User Experience Architects' Office, under the direction of psychologist and cognitive designer [Don Norman](#) (watch [this video](#) to find out his opinion about UX).

According to the **Nielsen Norman Group**, the consulting firm founded by Norman and Interface Engineer [Jakob Nielsen](#), the main [requirement for an exemplary UX](#) is the satisfaction of the client's exact needs, free from all problems and complications. Then come the simplicity and elegance that produce products that are a pleasure to own and to use. "For a company's offer to achieve a high-quality user experience, there must be a perfect fusion between the services of several disciplines, which should include engineering, marketing, graphic and industrial design, and interface design."

In sum, the UX covers all aspects of the end user's interaction with the company, and with the company's services and products.

Interaction design

Interaction Design, known by the acronym IxD, is the practice of designing interactive digital products, environments, systems and services. It focuses on the design of human behavior by synthesizing and imagining how a user would interact with digital products.

The purpose of interaction design is to satisfy needs and optimize user experience within the relevant technical or commercial limitations.

Common topics in Interaction Design include design, human-computer interaction, and software development.

Usability

The [Nielsen Norman Group](#) defines **usability** as a "quality attribute to assess the easy use of user interfaces. Usability also includes the methods meant to improve easy use during the design process. In the web, usability is a condition necessary for survival. When a website is not easy to use, people will leave it behind."

Design Systems

Grouped principles (clarity, efficiency, consistency and aesthetics), patterns (typography, colors, spaces and components) and practices (processes and tools for developing and designing) oriented at building a consistent, uniform and robust digital product.

[Nathan Curtis](#), an expert in the design of digital products, also adds that "a **Design System** comprises a library, visual style, and components -among other elements- that have been documented and released as code and design tools by an individual, a team or a community, so that the products

adopting it will be more efficient and cohesive.”

It is important to understand that Design Systems are not universal. Even when they share conformities, they are defined for each product or family of digital products. They also depend on the software’s tools.

DesignOps

The [Nielsen Norman Group](#) defined **DesignOps** as “the orchestration and optimization of people, processes, and craft in order to amplify design’s value and impact at scale”.

This term is used to address challenges such as:

- > Recruitment and hiring individuals with skills related to the field.
- > Creation of efficient workflows to improve the quality and impact of design results.

The objective is to establish processes and measures to back up scalable solutions for these challenges, allowing designers to also focus on designing and researching.

“DesignOps also applies to anyone who uses design thinking processes centered on the user in order to solve problems. The term ‘designer’ includes UX designers, user researchers, visual designers, content strategists, service designers, communication designers and any other individual contributing to the experience of end users.”

The three main areas that DesignOps focuses on are:

- > How team members interact with one another.
- > How the team works.
- > The impact that the work generates.

Basic steps to turn a good design into software



Software and design: the most common challenges

Some challenges encountered by companies seeking to keep their applications up-to-date are:

1. Research

Studying and devising what the solution required should be like is a task that may imply high costs that not all enterprises can afford.

2. Team

The optimal completion of a software project calls for the interaction of various profiles and roles, including designers, developers, experts in the deployment of applications, writers and testers.

3. Implementation

After the design for the solution is ready it must be implemented, and this also implies significant costs.

4. Changes in design

Companies are often reluctant to changes when a solution works perfectly, and design evolution is usually detained by fears of the risks implied in changes.

5. Achieve a coherent and consistent design

Es difícil tener un diseño perfectamente coherente en una única plataforma; y esa dificultad aumenta exponencialmente con cada plataforma que se agrega a la solución.

Unity makes strength!

The era of user experience and digital transformation has not only increased the complexity of system development; it has also generated the need for experts in interaction, visual design and information architecture, for all of them to focus together on the end user.

All roles are fundamental in the creation process of digital products, but the success of the interface and the UX lies in the interaction between designer and developer.

“That is where the risk of quality loss is the highest. Designers and developers use different working tools, and the lack of integrated tools usually implies repetition, by the developer, of tasks that the designer considers already solved and done,” explained [Fabián Bonilla](#), an expert in consulting, technical support and training for [GeneXus](#).

Design creation and implementation may be even more complex in [Mission Critical Systems](#).

«As it happens with code, the quality of a product often depends on the quality of the developers involved, and the same goes for design. A product’s design quality is closely related to the quality of its designers. As well, it is common for quality to also depend on the relationship between designers and developers. Having a “design-speaking” counterparty, a frontend developer, in the development team is essential for translating design into code»

Said GeneXus Design Chief [Mauro Canziani](#).

Experience tells us that the development process of the user interface in an application represents the greater part of the development cycle. So, when the design of the user interface is in the hands of an expert, there are two advantages:

1. The application will have a professional appearance and style.

2. The developer will have a set of previously created panels that might require minimal changes, but with the possibility of immediately focusing fully on the business logic.

This is possible with [DesignOps](#), and the process implies not only a change in tools but also a change in mindset.

«A central part of such changes is to know the roles required to start the path towards DesignOps. After deciding the change in mindset and placing design where it belongs, we must start using the proper tools and encouraging all others to go for the change. This process may start with a single individual willing to acquire new skills; and something that will surely follow is an actual diversification of the team into various profiles. Choosing GeneXus as development platform will simplify several steps of [DevOps](#) and DesignOps»

Explained [Damián Salvia](#), Computer Science Engineer at GeneXus.

How to generate software with a good design in GeneXus

Aiming at team productivity and integration, and seeking to simplify processes, [GeneXus](#) offers tools for importing and exporting designs -including all assets required (like images and font files)- from [Figma](#) and [Sketch](#). This enables developers to generate their applications with all the interface and user experience specifications defined by the designer.

GeneXus combines the best from those two worlds, while increasing the quality of applications developed with its low-code platform.

“At GeneXus, we strongly believe that Design Systems are the best strategy for maintaining and evolving application design meant for complex solutions,” said [Mauro Canziani](#), Design Chief at GeneXus.

Users of GeneXus have the following strategies available to work with Design Systems:

1. Using the Design Systems included in GeneXus

In order to improve visual languages and generate applications quickly, GeneXus offers:

- [SAP Fiori Design System](#)

Allows GeneXus users the possibility to create applications related to the SAP world, without the need for implementing any design. It is available in two main versions: SAP Fiori 2 and 3.

- [Unanimio Design System](#)

This new base Design System is meant for business applications and well as for mission critical and multiexperience apps, and it is part of every GeneXus product. This open source system, also customizable and extensible, was developed on the basis of the Design System object that allows the creation, modeling and evolution of Design Systems in the way that GeneXus does it traditionally, that is: independently from the technology.

2. Using the Design Systems of GeneXus' Partners

A second option for accelerating software development and implementing a high quality design is using Design Systems created by partners from the GeneXus community. They each have specific capacities that are suitable for different types of projects:

- [Ignia Framework](#) | [Simplifica](#)

- [Orion](#) | [K2BTools](#)

- [WorkWithPlus](#) | [WorkWithPlus](#)

3. Modifying or creating a proprietary design system with GeneXus tools to create customized Design Systems

The third strategy is recommended for large and complex multiexperience systems that require customized solutions.

To navigate complexity, GeneXus offers an array of tools and processes for creating, automating, maintaining and evolving full Design Systems:

- [Design System Object](#)

The Design System Object is the evolution of what was known in GeneXus as Themes. It is the source for creating, customizing and evolving Design Systems independently from technology.

It may be modularized, and it is also possible to define style features for screen interfaces in order to achieve more abstract designs towards easier reuse and assembly.

Additionally, it uses a structure with a format similar to CSS, allowing for a smoother interaction with the design team.

Designers and developers may work with the Design System Object using a common language, and this increases the cooperation between them.

- [Patterns](#)

The most usual business interactions may be solved with just one click.

- [User Controls](#)

Customized controls for creating apps with rich interfaces that are easy to use, adding

resources like menus, maps, graphs, calendars and diagrams, among others.

- [Stencils](#)

They allow the possibility to create and expand UI components to make them reusable in various parts of the application. They have no associated code, and their benefits imply agile design and development.

- [Import from External Design](#)

This tool enables the integration, the GeneXus knowledge base, of the design and interactions designed in **Sketch** and **Figma**, transforming all that visual language into objects such as Design Systems, Stencils, Panels, and much more. This reduces the time of UI implementation and facilitates the natural integration of the design team with less conflict.

- [Live Editing](#)

This tool is a fundamental aspect for developing quality user interfaces. It allows working live on the design of applications, from the final device, and the possibility to view, in real time, any modifications introduced in the panels, in order to achieve quality finished results in considerably less time.

How to create a Design System in GeneXus

There are three ways to go about this:

- 1.** Creating a design from scratch, from the [New Object](#) dialog inside the GeneXus [Integrated Development Environment](#) (IDE).
- 2.** Importing a Design System in the knowledge base in order to extend it later.
- 3.** Using the Design Import option, either with Sketch or with Figma, in order to modify and extend it later.

Recommendations to import external designs in GeneXus

- 1.** Have an in-depth knowledge about the design tools and the platform where the application will be executed (web or mobile environment, screen resolution, and orientation of the app, among others). This is because some may have a predefined Design System, such as [Apple's Human Interface Guidelines \(HIG\)](#) or [Material Design for Android](#).
- 2.** Design with the system in mind, that is, structuring the design, based on Design Systems or on components.
- 3.** Preserve a smooth dialog between designers, frontend specialists and developers, in order to reach an agreement relative to the aspects that will have an effect on the system's development and maintenance. In GeneXus, all design decisions are taken into account at the time of adjusting design files and importing, even cases of files where the tools assign names by default, with no semantics for a developer.

Tips for optimization with Atomic Design

According to [Mauro Canziani](#), **complex designs should be based on Design Systems**. Thus his recommendation to use the Atomic Design methodology, devised by [Brad Frost](#), to deal with inconsistencies and optimize design and digital product development to their maximum level.

“The idea of Atomic Design is taking standardized basic elements to define more complex elements. In the interface, the most basic elements, such as a button, a text field or a label, are called atoms. Atoms are combined into groups of two or three -like, for example, a text field with a button and a label, and an icon to define a searcher- to build molecules. By grouping molecules, it is possible to define larger elements called organisms.”

1. Organisms = Objects

“These objects include **Stencils** for defining reusable and multiplatform interface elements.”

2. Atoms and Molecules = GeneXus Components

“The components may be buttons and controls. To create your own, there are **User Controls** that allow the creation of complex controls.”

3. Sub-atoms = Design System Object

“With the Design System Object we can model tokens, styles and elements; and in doing so we are overriding GeneXus’ native controls.”

Complex designs
should be based
on Design Systems
to deal with
inconsistencies and
optimize design
and digital product
development to their
maximum level.

GeneXus 18 and Unanimomo: the key to designing and modeling complex systems

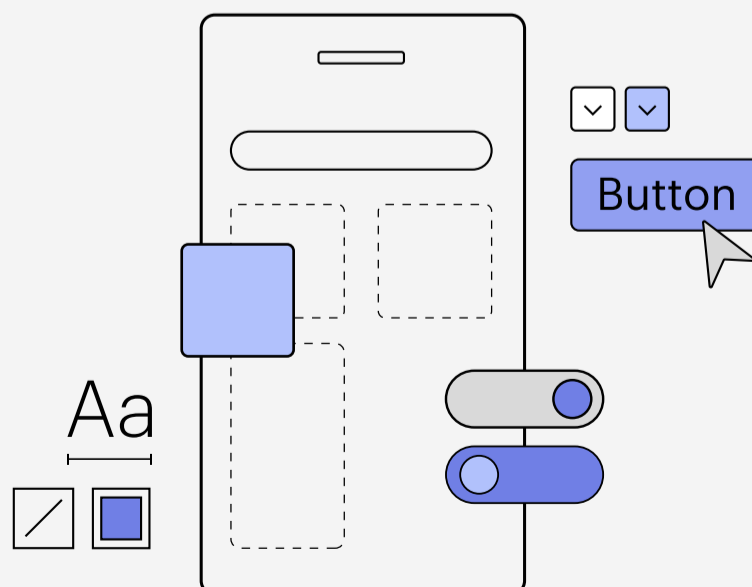
GeneXus 18 has all the components necessary for creating user experiences in accordance with modern design and usability practices.

Thanks to [Unanimomo](#) -the Design System included by default since GeneXus version 18-, the interfaces of applications are more unified and consistent.

This Design System, created and maintained by GeneXus, also simplifies the inte-

raction between designers and frontend specialists, while speeding up the software development cycle.

“Unanimomo brought along tools for implementing design in a simpler and easier manner, making interfaces behave and look alike, with all tasks and business processes along the same line in relation to design,” pointed out [Eugenio García](#), Product Manager of the GeneXus Suite.



Facts you should know about Unanimimo:

- **It was released with Apache 2.0 license**, and created as the core of Mission Critical Systems due to its visual and code solutions that may be implemented fast and with little effort.

- **Distributed in GeneXus, “out of the box”**, meaning that new knowledge bases (KBs) created and generated from the GeneXus 18 version will have the Unanimimo look and feel by default.

- **It is multi-experience**, and meant for web, native mobile y conversational.

- **It is open code, flexible and customizable**, allowing users to apply their own look and feel, or the look and feel of their company or business.

- **It is distributed as a module**. It includes all benefits of the use of modules in GeneXus, such as portability and easy installation and update, allowing users to do updates and handle versions independently from the GeneXus version.

- **It is extensible**. Its capacity may be extended by building new reusable components as part of the design system. Because it allows the addition of new classes and tokens, it is easier to adapt it to the specific needs of each project.

- **It is incremental**. It may be gradually included in the KB, without significant impact. This facilitates its adoption by projects and knowledge bases created with versions prior to GeneXus 18, or with Design Systems created through different Themes. In any case, the migration may be done by means of modules or objects.

- **There is one for every preference**.

It comes in two modes: Unanimimo Light and Unanimimo Dark.

Abiding by the principles of GeneXus, Unanimimo reduces the distance between design work and software development implementation, thus allowing for more modeling and less programming.

What does Unanimimo consist of?

In addition to the Design System Object, Unanimimo has also the following:

1. Objects and User Controls

Objects and User Controls solve the common needs of web and mobile apps.

2. Sidebar or side menu

It is loaded dynamically in GeneXus by means of an SDT, and it's used to navigate the application's options.

3. User dropdown

It is a control used to view the user logged on the application and to administer or modify that user's data, or to close the session.

4. Interaction controls

The most common ones are the switch control, a progress indicator, and an alert.

5. External library

For controls requiring a more complex logic, there is an external library, also developed by GeneXus, for encapsulating all that logic; it offers users a user control modeled as a markup, with the possibility of a composition between them.

6. Fonts

Fonts are a file inside the knowledge base that may be modified through the Tokens of the Design System. When a different font is required, it just needs to be imported to the knowledge base and its reference modified by means of a token.

7. Images and icons

They come in SVG format and allow the customization of certain elements in the Design System. Unanimimo provides a set of icons by default: profile, configuration, filter, and magnifiers, among others.

8. Colors

There are two alternatives or schemas for color representation: the dark mode and the light mode, which may be modified through the object's tokens.

Craving for more?

Find out more about these topics reading the following contents:

- > [Total Experience with GeneXus 18](#)
- > [Unanimo Design System: Vision, Capabilities & Customization](#)
- > [The benefits of modeling Multiexperience Apps to increase the productivity of Software Development](#)
- > [How to integrate a Design Team with GeneXus 18](#)
- > [K2BTools with GeneXus 18](#)
- > [WorkWithPlus with GeneXus 18](#)
- > [DesignOps Strategies: From Design to Code the GeneXus Way](#)
- > [Amplify Design Value and Impact at Scale with DesignOps](#)
- > [DesignOps and GeneXus](#)
- > [DesignOps – Designer Guide](#)
- > [DesignOps – Best practices for designers](#)
- > [Design Systems Introduction](#)
- > [GeneXus default Design System](#)
- > [Design Import](#)
- > [Frontend Development: Visual construction and interaction](#)
- > [Automated DevOps Course](#)

It's now time to import and model your apps' designs with cutting-edge technology.

info@genexus.com



MONTEVIDEO - URUGUAY

Av. Italia 6201- Edif. Los Pinos, P1

(598) 2601 2082

CIUDAD DE MÉXICO - MÉXICO

Hegel N° 221, Piso 2, Polanco V Secc.

(52) 55 5255 4733

MIAMI - USA

8950 SW 74th Ct, Suite 1406

(1) 201 603 2022

SÃO PAULO - BRASIL

Rua Samuel Morse 120 Conj. 141

(55) 11 4858 0300

TOKYO - JAPAN

2-27-3, Nishi-Gotanda

(81) 3 6303 9381

Shinagawa-ku, Tokyo, 141-0031

(81) 3 6303 9980