

**VIDHYADEEP INSTITUTE OF COMPUTER & INFORMATION TECHNOLOGY**

# CERTIFICATE

This is to certify that the seminar , submitted for the Seminar entitled “**Low/No Code Development**” has been carried out by “DG-Estate” at BCA Department of Vidhyadeep Institute of Computer & Information Technology, Kim for partial fulfilment of BCA degree to be awarded by Veer Narmad South Gujarat University.

Place:- KIM

Date:-

Head of Department

\_\_\_\_\_\_\_\_\_\_\_\_

Internal Guide

\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_

JURY

# Index

[CERTIFICATE I](#_Toc99732480)

[Index II](#_Toc99732481)

[1 Abstract 1](#_Toc99732482)

[1.1 Background 1](#_Toc99732483)

[1.2 Aim 1](#_Toc99732484)

[1.3 Method 1](#_Toc99732485)

[1.4 Result 1](#_Toc99732486)

[1.5 Conclusion 1](#_Toc99732487)

[2 What is Low/No Code Development 2](#_Toc99732488)

[3 Why Low Code Development 3](#_Toc99732489)

[4 History of Low Code Development 4](#_Toc99732490)

[5 Types of Low Code Development 5](#_Toc99732491)

[5.1 Low code 5](#_Toc99732492)

[6 8](#_Toc99732493)

# Abstract

## **Background**

In recent years, Low-code development (LCD) is growing rapidly, and Gartner and Forrester have predicted that the use of LCD is very promising. Giant companies, such as Microsoft, Mendix, and Outsystems have also launched their LCD platforms.

## Aim

In this work, I explored two popular online developer communities, Stack Overflow (SO) and Reddit, to provide insights on the characteristics and challenges of LCD from a developers and Industry perspective.

## **Method**

I used two LCD related terms to search the relevant posts in SO and extracted many posts. Meanwhile, I explored three LCD related subreddits from Reddit and collected many posts from there too.

## **Result**

Findings show that LCD may provide a graphical user interface for users to drag and drop with little or even no code, the equipment of out-of-the-box units (e.g., APIs and components) in LCD platforms makes them easy to learn and use as well as speeds up the development, LCD is particularly favoured in the domains that have the need for automated processes and workflows; and developers have conflicting views on the advantages and disadvantages of LCD.

## Conclusion

Findings suggest that researchers should clearly define the terms when they refer to LCD, and developers should consider whether the characteristics of LCD are appropriate for their projects.

# What is Low/No Code Development

No-code implies that no hand-coding is necessary, while Low-code relies on no-code elements such as visual drag-and-drop components combined with some hand-coding.

Technically These two can be two different topics in their own, but because No code development isn’t that vast enough so here, we will talk about Low Code Development with little bit of No Code Development.

Firms prefer to choose low-code alternatives for fast, continuous, and test-and learn delivery. The survey performed by Forrester also shows that LCD platforms can accelerate development by 5 to 10 times. Moreover, these platforms also offer enterprises a more economical way to fulfil the market and/or enterprises internal requirements.

Although LCD is booming in industry, there is no clear understanding of LCD as well as its practices. To this end, we plan to explore the characteristics and challenges of LCD from the perspective of developers.

Research found that most developers tend to use low-code (e.g., “The coding effort is low”) to describe LCD. In other words, they think that the coding effort is low in LCD. The term drag and drop comes second, followed closely by visual programming.

Some developers also use pre-designed templates, non-professional programmers friendly, what you see is what you get (WYSIWYG), and business process to demonstrate their understanding and perception of LCD. A few others consider that LCD utilizes a graphical user interface to develop programs, and one use case is built automation to “automate unattended operations with minimal human involvement ”.

# Why Low Code Development

With the growth of the Internet and the wave of digitalization, there is a growing need for enterprises to make quick and resilient responses to changing market requirements.

According to the research company Gartner, the demand for information systems will increase five times faster than the ability to provide them by IT departments, because number of employees is not growing at a sufficient pace. Furthermore, recruiting software engineers has become increasingly difficult as demand is high and supply is low.

In order to solve the problems above and adapt to this rapidly evolving world, companies are looking for quicker and cheaper ways to meet their software needs. In response, low-code development (LCD) platforms have emerged with the promise that organizations can hire business professionals with no coding experience to build applications.

The main goal of LCD as it suggests is to make development require less coding or less coding experience.

The main goal of LCD as it suggests is to make development require less coding or less coding experience.

# History of Low Code Development

While the term “Low code development” is relatively new, the actual method it works is not that new.

Low-code development platforms trace their roots back to Fourth-Generation Programming Language (4GL) a concept that was developed from the 1970s through the 1990s, overlapping most of the development of 3GL.

* 3GL examples: C, C++, Java, Python, PHP, Perl, C#
* 4GL examples: Unix Shell, SQL

4GL made programming relatively easy and more human readable which made it more developer friendly and required less developer effort than 3GL languages.

So, any technology which made a tiresome task easy can be considered a step forward in Low Code Development.

Reference: <https://betterprogramming.pub/low-code-history-b756c095494f>

# Types of Low Code Development

For now, there is no standard definition for LCD in industry according to my research, and there are no predefined types of LCD so here I have divided them by the methods it uses to explain its type accordingly.

|  |  |
| --- | --- |
| Term | Description |
| Low code | You need less programming skills and you are able to realize your processes without the need of coding. |
| Drag and drop | Can probably do everything through drag-and-drop. |
| Visual programming | low-code is a visual approach to software development. |
| Pre-made packages | Pre made packages or templates to ease tasks. |
| WYSIWYG | Meant for What You See Is What You Get app maker. |
| Wizards | They provide you with a graphical wizard. |

## Low code

Low-code is a visual approach to software development that optimizes the entire development process to accelerate delivery. With low-code, you can abstract and automate every step of the application lifecycle to streamline the deployment of a variety of solutions.

The term “Low code” is often used for solution that provide everything for developing an or S/W from start to the end like Creatio and Mendix. These two examples are commercial and do not provide open-source availability so developers who want control don’t prefer them while it is great choice for non-developers with little to no knowledge of programming.

## Drag and Drop

The term “Drag and drop” as it suggests is used for solutions that offer drag and drop feature to make applications. Some solutions also provide drag and drop plus code for more flexibility.

This solution makes it extremely user friendly cause user can just drop components and arrange them according to their liking, without coding a single line. Many Website builders fall under this category like Wix.com, WordPress, and AppyPie etc.

While drag and drop is easy for beginners and non-experienced not all of them provide full fledged solutions and there comes in drag and drop plus code where custom code can be implemented with drag and drop components. Like Visual Studio, WordPress etc.

## Visual Programming

The term “Visual Programming” is used for visualized code where instead of blocks of code it is showed in parts of code is showed in form of building blocks or nodes and you can change or add values inside these blocks manually. this method makes it beginner friendly and different from commonly used languages. like Scratch, Blockly etc.

While this method is mostly used in teaching kids or newbies programming but it is widely used in 3d modelling and games studios like Unreal, blender, Unity. Nodes based visual programming is difficult in its own and needs understanding to use but many people in community have made fabulous things out of it.

## Pre-made Packages

This term “Pre-made Packages” also self-describe itself ; it means using already made packages provided by community or solution provider.

These packages can be pre-installed or can be installed according to our need, there are many open source package managers in community like npm, composer yarn, etc.

Most of the time one package only provides few functionalities so we have to install multiple of them to full-fill our needs, while most of the package managers provide easy management one still needs to have fair enough programming knowledge, but if one has the necessary knowledge packages makes so many things easy.

## WYSIWYG

The term “WYSIWYG” stands for “What You See Is What You Get” another one which explains itself , WYSIWYG is type of system where user makes page like a word document and solution provides a web page based on what you entered, text editors used inside web browsers like CKEditor, summernote are very known example of this.

WYSIWYG is relatively old now because in most cases it only provides tag formatted code, so it doesn’t complete whole job but it makes it easier, WYSIWYG can still be seen in web solutions where user have to commonly change content there we can use WYSIWYG Editor to make things easy.

## Wizards

# Test