We talk a lot about app and website development as well as the developers. A front-end developer deals with the interface and a back-end developer takes care of the functionality. We know the tools they use and what the components of the software business are. However, we need to remember that ‘in the beginning was the Word’. In IT, the word means the coding language. That is why, in today’s article, I would like to focus on the main activity the developers perform – coding.

What is coding?

To properly answer the question of what coding is, let’s imagine a real-life situation. Two people speaking different languages and wanting to communicate have two options. They can either learn how to speak a third common language, or one of them learns how to speak the other one’s language. Now imagine the same situation between a human and a computer. To communicate with the computer effectively, humans needed to create a special language. Coding is the use of that special language to write a set of instructions. These instructions result in the creation of a piece of software, a website, or an application.

Frontend vs backend coding – what is the difference?

Understandably, what coding is in front-end development will differ from what it is in back-end development. A front-end developer uses programming languages, which enhance the visual part of a website. A back-end developer codes in programming languages that ensure the functionality of a given piece of software. However, these languages are not the same. Some elements or commands may be similar, but the construction differs as much as the use.

Examples of front-end languages:

- HyperText Markup Language (HTML)
- Cascading Style Sheets (CSS)
- JavaScript
- React
- Angular
- Vue
- jQuery

- Swift

Examples of back-end languages:

- C#
- Golang (Go)
- Java
- JavaScript
- PHP
- Python
- Ruby
- SQL

What are coding methodologies?

As in many other areas of life, there are also different ways of approaching coding. Such approaches are called programming methodologies. A multitude of different coding methodologies formed over the years. They all have a common goal of finding the best way to improve code quality, team productivity and collaboration. Let us have a look at some of the most important methodologies in software development.

Waterfall development

The waterfall model is the primary and oldest product development methodology. It is based on the belief that project development consists of sequential phases. The next one starts when the previous one is completed. It is considered one of the less flexible methodologies and the opposite of agile development.

Agile development

Agile is one of the most famous software development methodologies of the 21st century. It was created in 2001 by several developers signing the Agile Manifesto. The Manifesto consists of twelve principles regarding the conduct of development. The core rules are based on simplicity and continuous product development through collaboration and requirements evolution. Within Agile methodology, there are a few methods focusing on different factors, like practices or the flow of work.

Dynamic systems development method (DSDM)

DSDM places focus on the improvement in collaboration rather than technology. According to this method, more projects fail due to people's errors in collaboration.

Kanban

A very popular method of visualization of tasks, work capacity and progress. Many may associate the name Kanban with Kanban boards. They are a simplified version of the idea behind Kanban development. Kanban boards work well whenever an improvement in project management is needed.

Scrum

Scrum framework is commonly used not only in software development but also in sales, or marketing. Scrum relies on sprints, which are the timeframes within which the team achieves small tasks or goals. Such sprints last between 2-4 weeks.

Lean software development

The lean methodology follows the belief that anything that is not a value to the Client is a waste that should be eliminated. This method also follows a few more rules:

- Amplifying learning
- Delaying the commitment
- Delivering as fast as possible
- Empowering the team
- Building integrity in
- Optimizing the whole

Continuous integration

In this practice, developers frequently merge the code changes with a central repository and run tests.

Incremental development

This method allows for building systems in increments. The end requirement is known from the start of the project.

Rapid application development

This approach places emphasis on adaptation rather than project planning. This method often uses multiple prototypes in the process of software development.

Spiral development

The spiral model combines the waterfall approach and the iterative approach. It involves repeating the set phases of the project and adding functionalities until the project is ready to be released. Spiral development works exceptionally well with large projects.

Shape Up

This method consists of the timely delivery of predefined projects within product development. It is regarded as an effective method to manage risk and uncertainty.

Advanced methodologies

There are plenty of high-level approaches that focus on various factors of software development. Have a look at a few of the most important ones:

Chaos model

This method prioritizes resolving crucial tasks and issues.

Slow programming

The focus is placed on the precise and gradual completion of tasks without emphasizing the time pressure.

Extreme programming (XP)

This methodology focuses on improving software quality and responsiveness according to your client’s changing requirements. It is characterized by frequent releases and short development cycles.

What if a developer makes a mistake?

You will probably be surprised by the revelation I am about to spring on you, but all developers make mistakes. We are all humans. Coding and other IT processes are very complex. No matter if you are a front-end developer or a back-end developer, you are not infallible. The beginner makes mistakes, but an expert does too. The mistakes are just of a different nature due to different levels of advancement. How come all our software still works? Because we implement the processes that help us recognize the erroneous code and fix it. A part of this process is called a code review.

How does code review prevent mistakes from happening?

Code review is the process of checking the source code created by the developer. This is sometimes called a peer review because the code is checked not by the creator but by other people. The main aim of this activity is to identify possible mistakes in coding. It also allows us to improve code, find alternative solutions and serve as a learning tool. There are several ways of checking other developers' coding:

- **pair programming and over-the-shoulder reviews** – real-time reviews of coding and collaborating.
- **tool-assisted reviews** – done through dedicated tools that check the accuracy of the code. It is the most effective when combined with pair programming or over-the-shoulder reviews.
- **email code review** – code is sent by email and checked by a reviewer at their own convenience.

How does code review help with business stability?

Catching the possible errors assures the functionality of the system. However, code review has another benefit, which is very important to your business. It is stability. Knowing the mistakes in advance helps to avoid a snowball effect. It means moving along as if nothing happened and writing the code on top of a mistake. As you can imagine, the later you catch the mistake, the more time (and money!) you will need to spend fixing it. Catching mistakes as early as possible ensures you can stay within projected project timeframes and budgets. And that ensures your business

stability. Want to know more about how using code review tools helps your business? Check out our post about Sentry and its benefits.

Coding errors and bugs in your product? Never with Sailing Byte!

I hope I managed to sufficiently explain what coding is and how code review works. In Sailing Byte, code review is standard practice. It ensures all our applications and websites are fully functional and error-free before the release. Book a call to get an insight into coding methodology and how we eliminate errors while developing a product perfectly tailored to your needs.