**Língua inglesa – Software Development**

|  |
| --- |
| **Objectives:**  To encourage students to use English-English Internet dictionaries. To reinforce the habit of using a monolingual dictionary. To help learners be more autonomous in their learning.  To encourage students to search the internet as part of their work. To read intensively to get a more detailed understanding of a topic. To clearly communicate ideas and produce well-structured arguments. |

EXERCÍCIOS

**Look at the dictionaries (links below) to help you answer to the questions:**

<https://www.collinsdictionary.com/dictionary/english/>

<https://www.dictionary.com/>

<https://www.merriam-webster.com/>

**Group work**

Read the text and do the activities.

**What is Software Development?**

Software development is the process of conceiving, specifying, designing, programming, documenting, testing, and fixing bugs involved in creating and maintaining applications, frameworks, or other software components. In other words, it's the process of making a new software product, from its initial concept to its final release, and beyond.

This software could be a small tool to help with day-to-day tasks, a mobile game, a business process system for large corporations, or anything in between. Software is embedded in countless aspects of our modern lives and plays a key role in numerous industries such as healthcare, finance, entertainment, and more.

**What is the Role of Software Development?**

The role of software development is crucial in today's digital age, where almost everything runs on software. At its core, software development enables people and businesses to solve problems, achieve specific goals, and generally function more effectively. Here are some of the key roles of software development:

**Problem-Solving**: Software can automate repetitive tasks, analyse large amounts of data quickly, improve efficiency, and solve numerous other problems.

**Innovation**: New software can disrupt industries, changing how we work, communicate, and live our lives.

**Enabling Business Processes**: Software is often key to providing goods and services, whether it's a website for an online retailer, an app for a food delivery service, or a complex system managing logistics for a large corporation.

**What is the Software Development Process?**

The software development process is the series of steps that developers follow to create a new software product. While the exact steps can vary, most software development includes the following stages:

**Requirements Analysis**: This initial stage involves understanding what the software needs to do. Developers work with clients or end-users to determine the features and functionality required.

**Design**: Once the requirements are clear, developers create a plan for how the software will work. This can include deciding on the architecture, interfaces, and data models that will be used.

**Coding**: This is the actual writing of the software. Developers create code in a suitable programming language to bring the design to life.

**Testing**: Developers and testers work together to find and fix any issues with the software. They aim to ensure the software works correctly, is easy to use, and doesn't have any bugs.

**Deployment**: Once the software is tested and refined, it's released to the users. This could involve deploying it to a server, publishing it in an app store, or distributing it in another way.

**Maintenance**: After release, developers continue to work on the software. They might add new features, fix any problems that come up, or update the software to keep up with changing technology.

This process isn't always linear - it can be iterative, with developers going back to previous steps as needed. This is especially common in Agile development, where the process is divided into small, manageable parts called "sprints", allowing for continuous improvement and flexibility.

In a nutshell, software development is a creative and technical process that plays a vital role in shaping the way we live and work. It's a process that involves collaboration, problem-solving, and continual learning, and is an exciting field with endless possibilities.

***Activity 1*** – **Developing vocabulary skills**

The function of a word in a sentence is determined by its position and how it relates to other words in the sentence. To enhance your understanding, it's essential to examine and think about the role or 'job' of the underlined words in the text. How are these words contributing to the overall meaning of the sentence? What purpose do they serve?

Choose your favourite online dictionary. Identify the part of speech, meaning, and pronunciation of these vocabulary items. Write an example sentence showcasing the use of these words. Remember, your focus should not just be on the words themselves, but also on what they are doing within those sentences - their function or 'job'.

In the sentence, "Software development is the process of ***conceiving***, specifying, designing, programming, documenting, testing, and fixing bugs…", 'conceiving', 'specifying', 'designing', 'programming', 'documenting', 'testing', and 'fixing' are all gerunds. A gerund is a verb form that ends in "-ing" and works like a noun in the structure of a sentence. We use gerunds when we want to focus on ***the general idea of an activity***, not the specific action. So, when we say 'conceiving' or 'designing', these words are acting like nouns, even though they are based on verbs.

The preposition 'of' usually indicates possession, origin, or composition. So, 'of' in our sentence is used to detail what the process consists of, or is composed of, and its objects are all the steps involved in software development, represented by our gerunds. In simpler terms, if we say that software development is a process, the 'of' tells us more about this process. The process includes all those steps: conceiving an idea, specifying requirements, designing the software, programming it, documenting the work, testing it for errors, and fixing those bugs. Each one of these activities is an integral part of the process 'of' software development.

| **vocabulary item** | | **Part of speech** | **meaning** | **pronunciation** | **example sentence** |
| --- | --- | --- | --- | --- | --- |
| ***1*** | ***Conceiving*** | gerund, verbal noun | Forming or devising a plan or idea in the mind | /kənˈsiːvɪŋ/  /Cãmciivin/ | Gerund: "***Conceiving*** a unique software solution requires a deep understanding of the problem space."  **Subject**: "***Conceiving*** *innovative features* can significantly enhance the user experience of the software."  **Object**: "The initial phase of software development involves thoroughly ***conceiving*** the concept to ensure its viability and usefulness." |
| ***2*** | maintaining | gerund, verbal noun | Keeping something in good condition or continuing it effectively | /meɪnˈteɪnɪŋ/ | Subject: Maintaining a secure and reliable database is critical for ensuring data integrity in software development. |
| ***3*** | frameworks | plural noun | A basic structure underlying a system, concept, or text, especially in software | /ˈfreɪmˌwɜːrks/ | Object: Developers use frameworks to build software faster and more efficiently. |
| ***4*** | Release | noun | The act of making a product or update available for use | /rɪˈliːs/ | Subject: Small releases help development teams deliver new features quickly and reduce the risk of bugs. |
| ***5*** | Day-to-day | adjective | Happening as a regular part of daily activity; routine | /ˌdeɪtəˈdeɪ/ | Subject: Managing day-to-day tasks effectively is crucial for keeping a software project on track. |
| ***o*** | Embedded | adjective | Built into something else and working as part of it | /ɪmˈbɛdɪd/ | Object: Embedded software is commonly used in devices like smart TVs and industrial machines. |
| ***7*** | Core | Noun/ adjective | The most important or central part of something | /kɔːr/ | Subject: The core of software development is creating innovative solutions to complex problems. |
| ***8*** | Effectively | adverb | In a way that works well and gets results | /ɪˈfɛktɪvli/ | Object: Developers must communicate effectively to ensure smooth collaboration in a project. |
| ***9*** | Automate | verb | Transform something that usually was done manually into something | /ˈɔːtəmeɪt/ | **Object** Teams often automate testing to save time and avoid manual errors. |
| ***10*** | Efficiency | noun | Doing something well without wasting time or effort | /ɪˈfɪʃənsi/ | Subject: Code reviews help improve the efficiency of the development process |
| ***11*** | Disrupt | verb | To interrupt or change something in a big or unexpected way | /dɪsˈrʌpt/ | **Object:** A single bug can disrupt the entire deployment process if not caught early. |
| ***12*** | enabling | verb | Making something possible, practical, or easier to do | **ɪˈneɪblɪŋ/** | Object: The platform focuses on enabling real-time communication between users. |
| ***13*** | providing | gerund,  Verbal noun | If you provide something that someone needs or wants, or if you provide them with it, you give it to them or make it available to them. | /prəˈvaɪdɪŋ/ | **Object**: The developers discussed **providing better customer service** |
| ***14*** | End-users | noun | The end user of a product or service is the person that it has been designed for, rather than the person who installs or maintains it. | /ˈɛnd/ /ˈjuːzər/ | **Subject**: **End-users** like when the software runs smoothly |
| ***15*** | Interfaces | noun | An interface of a particular piece of computing software is its presentation on screen and how easy it is to operate. | /ˈɪn.tə.feɪ.sɪz/ | **Subject**: **Good interfaces** are key to the UX. |
| ***16*** | Refined | verb | Improve something by having small changes made to it. | /rɪˈfaɪnd/ | **Subject**: **Refining a software** is mandatory before launching it. |
| ***17*** | deploying | gerund, verbal noun | To put into action; to move into position for use. | /dɪˈplɔɪɪŋ/ (UK) or /dɪˈplɔɪɪŋ/ (US) | **Object:** Their primary responsibility is overseeing the deploying of new security patches across the network. |
| ***18*** | features | Noun | Distinctive attributes or aspects. | /ˈfiːtʃərz/ | **Object:** The product team prioritized the most requested features for the next release. |
| ***19*** | Come up | Verb (phrasal verb) | To think of; to approach. | /kʌm ʌp/ | **Object:** We need to address any issues that come up during the testing phase. |
| ***20*** | iterative | adjective | Involving repetition of a process for improvement. | /ˈɪtərətɪv/ (UK) or /ˈɪt̬ərət̬ɪv/ (US) | **Object:** The design team adopted an iterative approach to prototyping. |
| ***21*** | manageable | adjective | Able to be controlled or handled. | /ˈmænɪdʒəbəl/ | **Object:** We divided the complex problem into several manageable sections. |
| ***22*** | In a nutshell | idiom | In a very brief or concise way | /ɪn ə ˈnʌtʃɛl/ | Object: The executive asked for the proposal in a nutshell to save time during the meeting. |
| ***23*** | shaping | Gerund, verbal noun | Giving form or influence to something's development | /ˈʃeɪpɪŋ/ | Gerund: Shaping a product based on user feedback leads to higher satisfaction rates. |
| ***24*** | Agile development | Noun phrase | A flexible, iterative approach to software development focused on collaboration and customer feedback | /ˈædʒaɪl dɪˈvɛləpmənt/ | Subject: Agile development allows for quick adjustments based on stakeholder input. |
| ***25*** | sprints | Noun (plural) | Short, time-boxed work periods in Agile development used to complete specific tasks | /sprɪnts/ | Subject: Sprints keep the development process organized and focused on short-term goals. |

***Activity 2*** - Choose at least six of the previous vocabulary items and create a 5–8-line text.

***Activity 3*** – **Group Research and Presentation**

**Objective**

You will work in small teams to research, explain, and present a topic related to **Software Development**.

**The aim is to:**

* Practice your **English communication skills**
* Deepen your **understanding of essential software development concepts**
* Connect these concepts to **real-world applications** in your future professional roles

**Group Work Instructions**

* Work in groups of **3–4 students**
* Choose **one topic** related to **Software Development** (see list below)
* Prepare a **10–12 minute presentation** in English
* After each presentation, the class will have a short **Q&A session** (3–5 minutes)
* Focus on:
  + What the topic is
  + Why it’s important for software developers
  + How it is used in real-world projects and companies
* Use **clear and simple English** — **Teach us something useful!**

**Presentation Goals**

Each group should:

1. **Explain the topic** – What is it? How does it work?
2. **Explain its importance** – Why does this matter for a software developer?
3. **Show real-world applications** – Where and how is it used today?
4. **Mention tools, languages, or methodologies** – What technologies or skills are required?
5. **Discuss challenges and solutions** – What problems can happen and how are they solved?
6. **Connect it to your future** – How will you use this in your future job?

**Suggested Topics (you may also propose your own)**

1. The Software Development Life Cycle (SDLC)
2. Agile vs. Waterfall Methodologies
3. Requirements Analysis and User Stories
4. UI/UX Design in Software Development
5. Version Control Systems (e.g. Git, GitHub)
6. Programming Languages (e.g. JavaScript, Python, Java)
7. Front-End vs. Back-End Development
8. Software Testing and Debugging
9. APIs and System Integration
10. Software Maintenance and Updates
11. DevOps and CI/CD Pipelines
12. Mobile App Development
13. Web Application Architecture
14. Documentation and Code Comments
15. Security in Software Development

**Language Support Table for Presentations**

|  |  |  |
| --- | --- | --- |
| Function | Useful Phrases | Example |
| 1. Introducing the Topic | - We’re going to talk about… - Our topic is… - This presentation is about… | We’re going to talk about *Virtual Machines* and how they are used in cloud computing. |
| 2. Giving a Definition | - This refers to… - It means… - It’s a system/process/tool that… | Cloud computing refers to delivering services over the internet, such as storage or processing power. |
| 3. Introducing a Process | - This process starts with… - “There are several steps… - First, we need to… | This process starts with setting up the network infrastructure. |
| 4. Showing Sequence | - First… / Next… / Then… / After that… / Finally… | First, we configure the DNS. Then we check the IP settings. Finally, we test connectivity. |
| 5. Explaining a Step | - In this step, we… - This part is responsible for… - Here, the system/user/admin… | In this step, the system verifies the credentials of the user. |
| 6. Describing Tools/Platforms | - This is done using… - We use tools like… - The platform provides… - An example of this tool is… | We use Ansible to automate server configurations. |
| 7. Explaining Importance | - This is important because… - Without this, it would be difficult to… - It ensures/improves/protects… | Firewall configuration is important because it protects the network from attacks. |
| 8. Connecting to the Job | - As future administrators, we will need to… - This is a key responsibility for… - We’ll use this to… | As network administrators, we’ll need to manage IP address allocation using DHCP. |
| 9. Giving Real-World Examples | - A real example of this is… - This is commonly used by… - For instance, in companies like… | A real example is Google Cloud using containers to run scalable apps. |
| 10. Mentioning Protocols/Standards | - We follow the [protocol name] to ensure… - This is based on industry standards such as… | We use the TCP/IP protocol to enable reliable data transmission. |
| 11. Identifying Challenges | - One common issue is… - Sometimes, we face problems with… - The main difficulty is… | One common issue with VPNs is a drop in connection speed. |
| 12. Suggesting Solutions | - To solve this, we can… - A common solution is… - The best practice is to… | To solve this, we can upgrade the network hardware to support more bandwidth. |
| 13. Showing Impact on Customers | - This helps users by… - The customer benefits from… - It improves the user experience by… | It improves the user experience by offering fast, secure remote access to data. |
| 14. Showing Results/Benefits | - As a result… - This leads to… - The benefit is… - It reduces/increases/improves… | As a result, downtime is reduced and productivity increases. |
| 15. Wrapping Up | - In conclusion… - To summarise…  - So, we’ve seen how… | To summarise, containers help organisations deploy applications faster and with greater flexibility. |
| 16. Inviting Questions | - Do you have any questions? - Would you like us to clarify anything? - What do you think about…? | Do you have any questions about network segmentation? |

**Group Presentation Checklist**

* Introduce your team and your topic clearly
* Explain your topic in simple English
* Show why it is important for software development
* Include real-world examples, tools, or platforms
* Discuss challenges and suggest solutions
* Connect the topic to your future job or role
* Use visuals (slides, diagrams, images)
* Practice and speak clearly
* Be prepared to answer questions

**Evaluation Criteria**

You will be assessed both on your group's performance leading up to the presentation and on different aspects of the actual presentation.

**Group's Performance Leading Up to the Presentation**

**Communication:** How well the group shares information and communicates within the team.

**Collaboration:** How effectively the group works together and supports each other’s contributions.

**Task Delegation:** How tasks are assigned within the group to ensure everyone has a role.

**Research:** The thoroughness and depth of the group's research on the topic.

**Information Gathering:** How effectively the group collects necessary data and resources.

**Critical Thinking:** The ability to analyse information and form reasoned conclusions.

**Problem Solving:** How the group addresses and resolves any challenges that arise.

**Time Management:** The group's ability to use time wisely throughout the project preparation.

**Meeting Deadlines:** How well the group adheres to deadlines set for tasks and milestones.

**Presentation Planning:** The process of organising and structuring the presentation content.

**Preparation:** The readiness of the group for the presentation, including rehearsal and material preparation.

**Group Dynamics:** The interaction and relationship dynamics within the group.

**Aspects of the Presentation**

**Introducing the Topic:**

**Clarity:** The introduction should be clear and easy to understand.

**Engagement:** The opening should capture the audience's attention effectively.

**Conciseness:** The opening statement should be brief and to the point.

**Overview of Presentation:**

**Organisation:** The presentation should have a clear structure.

**Explanation:** Clearly explain what will be covered in the presentation.

**Transitions:** Move smoothly from one section to another.

**Beginning a New Section:**

**Introduction:** Each new section should start clearly and engagingly.

**Guidance:** Use signposts to help the audience follow along.

**Finishing a Section:**

**Wrap-up:** Each section should end with a clear summary of key points.

**Transition:** Shift smoothly to the next section or topic.

**Analysing Points and Offering Recommendations:**

**Depth:** Show a deep understanding of the topic in your analysis.

**Recommendations:** Provide thoughtful and actionable suggestions.

**Communication:** Explain complex ideas in a clear and straightforward way.

**Giving Examples:**

**Relevance:** Use examples that clearly relate to and illustrate your points.

**Clarity:** Explain how each example supports the topic.

**Engagement:** Choose interesting and engaging examples.

**Paraphrasing and Clarifying:**

**Rephrasing:** Effectively simplify and clarify complex ideas.

**Conciseness:** Keep explanations brief and to the point.

**Summarising and Drawing Conclusions:**

**Summarising:** Clearly recap the main points at the end of the presentation.

**Conclusions:** Present final thoughts clearly and concisely.

**Relevance:** Ensure the summary and conclusions are directly related to the topic.

**Inviting Questions/Discussion:**

**Openness:** Be welcoming to questions and discussions from the audience.

**Encouragement:** Actively encourage audience interaction.

**Responsiveness:** Address audience questions and feedback effectively and clearly.