

English II

MODULE 4

Web Page: TypeScript Features



COMMUNICATIVE CONTEXT

Purpose

At the end of this module, you will be able to:

- explain basic TypeScript features and typing concepts;
- compare TypeScript and JavaScript by identifying advantages and limitations;
- justify technical decisions using simple argumentative structures (A2);
- deliver a short, structured oral presentation about TypeScript.

Table of Contents

01	Reading Comprehension	02
	1.1 TypeScript Basics	03
	1.2 “TypeScript as the Evolution of JavaScript”	05
02	Vocabulary & Grammar	08
	2.1 Languages, Tools and Code	09
	2.2 Future Simple – will	11
03	Listening Comprehension	13
	3.1 Argumentation: Advantages and Limitations	14
	3.2 Oral Presentation Model	15
04	Written Production	21
	4.1 Controlled Production	22
	4.2 Free Production: a technical paragraph	25
05	Speaking	26
	5.1 Oral Presentation – TypeScript vs JavaScript	27
	5.2 Oral Presentation Checklist	20



01

Reading Comprehension

In professional programming environments, developers often need to explain why a specific technology is used. In this module, you will learn how to describe and justify the use of TypeScript in clear, simple English.

READING COMPREHENSION
Text #1



“TypeScript as the Evolution of JavaScript”

1.1 Pre-reading activities

a. **Discuss:**

- Why can flexibility be a problem in large projects?
- What problems can appear when code grows?



b. **Vocabulary: Key Terms (TypeScript & DOM)**

Match the words with their meaning:

Words:

1. static typing
2. transpilation
3. compiler
4. DOM
5. event

Definitions:

- a. the process of converting TypeScript code into JavaScript
- b. a program that checks code and shows errors
- c. a structured representation of a web page
- d. a system where variables have defined types
- f. an action triggered by user interaction

2. Read the text: “TypeScript as the Evolution of JavaScript”

JavaScript is a very flexible language. This flexibility allows developers to write code quickly, but it can also create problems in large projects. When applications grow and there are no clear rules or structures, the code can become difficult to maintain.

TypeScript is considered the natural evolution of JavaScript. It is a superset of JavaScript, which means that all JavaScript code is valid in TypeScript. The main difference is that TypeScript adds static typing and advanced development tools. With static typing, developers can detect errors during development and not in production. For example, if a function expects a number, the compiler will show an error if a string is used. This improves code quality, readability, and maintainability.

TypeScript code is written in its files and then transpiled into JavaScript. This step is necessary because browsers only understand JavaScript. Despite this extra step, TypeScript is widely used because it helps teams work better in large projects. TypeScript also improves the developer experience with intelligent autocompletion and better documentation inside IDEs. For this reason, it is used in many modern frameworks such as Angular, React, and NestJS.

OpenAI. (2026). ChatGPT (versión GPT-5.2) [Large language model]. <https://chat.openai.com>

2.1. While reading activities

Choose the correct option:

TypeScript is:

- a) a replacement for JavaScript
- b) an evolution of JavaScript
- c) unrelated to JavaScript

Answer:

Why is static typing useful in large projects?

.....
.....

2.2. After reading activities

a. Fill in the Blank: Fill in the blank with the correct words from the word bank below.

Word bank: autocompletion, transpiled, flexibility, superset, static

JavaScript's _____ allows developers to write code quickly but can create problems in large projects.

TypeScript is a _____ of JavaScript, meaning all JavaScript code is valid in TypeScript.

The main difference between JavaScript and TypeScript is that TypeScript adds _____ typing and advanced tools.

TypeScript code is _____ into JavaScript because browsers only understand JavaScript.

TypeScript improves the developer experience with intelligent _____ and better documentation.

b. Multiple Choice Questions: Choose the correct answer from the choices for each question.

What is one potential problem with JavaScript in large projects?

- A) It is too slow for modern computers
- B) Its flexibility can make code hard to maintain
- C) It cannot run in web browsers
- D) It has static typing by default

What does it mean that TypeScript is a "superset" of JavaScript?

- A) TypeScript can only run in Node.js
- B) All TypeScript code is valid in JavaScript
- C) All JavaScript code is valid in TypeScript
- D) JavaScript and TypeScript are not related

What feature does TypeScript add to JavaScript?

- A) Dynamic typing
- B) Static typing
- C) Less strict documentation
- D) Fewer development tools

Why must TypeScript code be transpiled?

- A) Browsers only understand TypeScript
- B) TypeScript is too slow for browsers
- C) Browsers only understand JavaScript
- D) TypeScript is not compatible with computers

Which of the following is a benefit of using TypeScript?

- A) Makes code harder to read
- B) Reduces team collaboration
- C) Provides intelligent autocompletion
- D) Removes documentation from IDEs



Text: “Why Developers Choose TypeScript”

1. Pre-reading activities

Discuss

Look at the title and answer:

Why do developers need clear rules when writing code?

What can happen if a team uses different styles in the same project?

2. While reading activities

“Why Developers Choose TypeScript”

In modern software development, teams often work on the same project for a long time.

When many developers write code together, it is important to have clear rules and structures.

Without them, projects can become confusing and difficult to maintain.

TypeScript helps solve this problem. It is based on JavaScript, but it adds tools that make code easier to understand and organize. One of its most important features is static typing.

This allows developers to define what type of data a variable can store. Because of this, many errors are detected early, before the application is in production.

Another advantage of TypeScript is that it improves teamwork. When code is clear and well-structured, new developers can understand it more quickly. This saves time and reduces mistakes. Although TypeScript requires an extra step before running the code, many teams consider this step a small cost for better quality and safety.

Today, TypeScript is widely used in professional environments because it supports long-term projects and helps developers create more reliable applications.

OpenAI. (2026). ChatGPT (versión GPT-5.2) [Large language model]. <https://chat.openai.com>

3. While-reading Activity

Choose the correct answer.

Why is clear structure important in team projects?

- a) Because it makes code run faster
- b) Because it helps developers understand and maintain code
- c) Because it removes the need for documentation

What does static typing help developers do?

- a) Write code more slowly
- b) Detect errors earlier
- c) Avoid using JavaScript

Why do teams accept the extra step in TypeScript?

- a) Because it is required by browsers
- b) Because it improves code quality
- c) Because it removes teamwork

4. After-reading Activity

Read the sentences carefully and complete the blanks with the right word from the box

Word bank:
structure – errors – teamwork – reliable – production

1. Clear _____ helps developers understand code more easily.
2. TypeScript helps detect _____ before the application is in _____.
3. Good _____ is easier when the code is well organized.
4. Many teams choose TypeScript because it helps create more _____ applications.

02

LISTENING COMPREHENSION

Argumentation: Advantages and Limitations



This section will give you insights on the Oral Presentation Genre. In doing so, you'll learn specific structures and expressions to put it together.

2.1. Oral Presentation Model

PRE LISTENING ACTIVITY

1. Listen to a team member deliver an oral presentation about the use of a programming language. As you listen, pay attention to:

- the format of the oral presentation genre: what parts or sections can you identify? What are they?
- What kind of language can you recognize? Formal, informal? How can you tell?
- Which key words and expressions do you know from before? Are there any new words? Make a list and look them up.
- Go over the words in bold and think: what's their function in this text?



AUDIO HERE: 1766616293269620811-348419506040937.mp3

Read the transcript below

Good morning everyone.

Today, I would like to explain why our team decided to use TypeScript for this project instead of using only JavaScript. I will briefly describe the main advantages and also mention some limitations before concluding.

TypeScript is a programming language based on JavaScript, but it adds static typing and development tools that help organize code. Because of these features, it is especially useful in large projects and team environments.

One important advantage is that TypeScript helps detect errors during development. Although this language requires an additional compilation step, it reduces problems before production. This is important when many developers work on the same codebase.

Another benefit is code maintainability. Despite being more strict than JavaScript, TypeScript improves readability and structure. As a result, team members can understand and modify the code more easily. However, there are also some disadvantages. In spite of its benefits, TypeScript can add complexity to small projects or quick prototypes. The learning curve is slightly higher, though developers who already know JavaScript can adapt quickly.

In conclusion, TypeScript is not a replacement for JavaScript, but an evolution. Although it adds extra steps to the development process, it improves code quality, collaboration, and long-term maintenance. For these reasons, our team believes that TypeScript is the best option for this project.

Thank you for your attention. That's all from my side. Does anyone have any questions or comments?

WHILE LISTENING ACTIVITY



COMPREHENSION CHECK

1. Read the presentation again and answer the questions.

- What is the main purpose of the presentation?
- Why does the team choose TypeScript for the project? Mention two advantages.
- What limitation of TypeScript is mentioned?
- According to the speaker, in which type of projects is TypeScript more useful?
- Is TypeScript presented as a replacement for JavaScript? Why or why not?

2. Decide if the sentences are True (T) or False (F). Correct the false ones.

- TypeScript detects errors only after the code is in production.
- TypeScript improves readability and maintainability of code.
- The speaker mentions only advantages, not disadvantages.
- TypeScript is recommended for large projects and team work.
- TypeScript completely replaces JavaScript.



GENRE FOCUS – DESCRIPTIVE & ARGUMENTATIVE ORAL TEXT

What type of text is this?

This presentation is a descriptive and argumentative oral text used in professional and academic contexts.

Descriptive, because it explains what TypeScript is and how it works.

Argumentative, because it presents reasons, advantages, and limitations to justify a technical decision.

Section	Purpose	Content in the presentation
<i>Introduction</i>	<i>Present the topic and purpose</i>	<i>Topic, comparison, structure</i>
<i>Development</i>	<i>Explain and justify</i>	<i>Advantages, disadvantages, examples</i>
<i>Conclusion</i>	<i>Close and reinforce</i>	<i>Final opinion and decision</i>

CONTENT ELEMENTS

- technical description (what it is)
- justification (why it is used)
- contrast (advantages vs limitations)
- professional register and clear organization

AFTER LISTENING ACTIVITY

4. LANGUAGE FOCUS - Connectors for Argumentation

Explanation: Why do we use connectors?

In technical presentations, connectors help to:

- organize ideas clearly,
- show contrast,
- explain causes and reasons,
- sound professional and coherent.

Connector	Use	Example
<i>although / though</i>	<i>contrast</i>	<i>Although TypeScript is strict, it is safer.</i>
<i>despite / in spite of</i>	<i>contrast (noun / -ing)</i>	<i>Despite its complexity, it is useful.</i>
<i>because of</i>	<i>cause / reason</i>	<i>Because of static typing, errors are detected early.</i>

a. Choose the Correct Connector

1. Complete the sentences with:

although – despite – because of – though

1. _____ its flexibility, JavaScript can be difficult to maintain.
2. TypeScript improves code quality _____ static typing.
3. _____ TypeScript adds complexity, it helps large teams.
4. It requires compilation, _____ developers adapt quickly.

b. Controlled Production. Complete the sentences orally.

- Although TypeScript _____, it _____.
- Despite _____, this language _____.
- Because of _____, developers can _____.

Work with a partner and say two sentences each.



c. Communicative Speaking Task

Situation: You are in a team meeting. A colleague asks:



“What are the advantages and disadvantages of using TypeScript in this project?”

Speak for 1–2 minutes.

Use:

- ☒ at least 2 connectors,
- ☒ technical vocabulary from the module,
- ☒ a clear structure (introduction – development – conclusion).



You may begin with:
In my opinion...

- Although...
- Because of...
- In conclusion...

SPEAKING PRODUCTION CHECKLIST: ORAL TECHNICAL PRESENTATION – ENGLISH II

1. Format of the Genre

- ☐ I clearly introduce the topic and purpose of my presentation.
- ☐ My presentation has a clear development (ideas are explained, not listed).
- ☐ I include a conclusion that summarizes my opinion or decision.

2. Use of Specific Vocabulary

- ☐ I use vocabulary related to programming languages, tools, and code.
- ☐ I use technical words appropriately (e.g. language, feature, project, code, function).
- ☐ My vocabulary helps explain technical ideas clearly.

3. Content and Adequacy

- ☐ My ideas are related to the topic (TypeScript vs JavaScript).
- ☐ I mention at least one advantage and one limitation.
- ☐ My explanation is clear for a technical audience.

4. Quality of Ideas

- ☐ My ideas are logically connected.
- ☐ I justify my points with simple reasons or examples.
- ☐ I use connectors (although, because of, despite, though) to organize my ideas.

5. Pronunciation and Intonation

- ☐ My pronunciation is clear and understandable.
- ☐ I use appropriate intonation to highlight important ideas.
- ☐ I speak at a clear and appropriate pace.

Overall Reflection

- ☐ I feel confident explaining a technical decision in English.
- ☐ I can improve my explanation with more practice.

03

Vocabulary & Grammar



Vocabulary in Context: Languages, Tools and Code

These activities help you work with key vocabulary related to programming languages and code, before focusing on argumentation.

Motivation & Goals

1. Read the following passage related to a development project.

Do you recognize the words in bold?

In a web development **project**, developers choose a programming **language** and the right **framework** to build an application. Each **tool** they use adds a new **feature** to improve the quality of the **code**.

The **browser** and the **server** work together to run the **application** correctly. Inside the code, every function has a clear purpose, and each **variable** stores important data. An **object** helps organize this information and makes the system easier to understand and maintain.



2. Classify the words. Put the words in the correct category:

language – framework – feature – tool – browser –
server – code – project – application – function –
variable – object

Categories:

- Programming language / tools:
- Code elements:
- Environment / context:



3. Match each word with the correct definition:

1. programming language
2. tool
3. feature
4. function
5. variable

- a set of instructions written to perform tasks
- a characteristic or capability of a language or software
- something used to help create or develop software
- a block of code that performs an action
- a value that can change

3. Complete the sentences with a word from the box:

language – code – browser – project –
function – variable

1. JavaScript is a programming _____.
2. A _____ stores information that can change.
3. A _____ performs a specific action.
4. Developers write _____ to create applications.
5. Large _____ need good structure.
6. The _____ executes JavaScript.

4. Identify the context. Read the sentences and choose the best option:

“This tool helps developers write better code.”

- a) tool
- b) variable
- c) browser

“The application runs in the browser.”

- a) language
- b) environment
- c) function

“TypeScript adds new features to JavaScript.”

- a) code element
- b) tool
- c) language feature

Key Terms (TypeScript & DOM)

5. Match the words with their meaning:

- | | |
|-----------------|--|
| • static typing | 1. the process of converting TypeScript code into JavaScript |
| • transpilation | 2. a program that checks code and shows errors |
| • compiler | 3. a structured representation of a web page |
| • DOM | 4. a system where variables have defined types |
| • event | 5. an action triggered by user interaction |



GRAMMAR



Future Simple – will

When do we use will in this module?

In this module, will is used in professional and academic communication to:

- talk about **future results** of using TypeScript;
- describe **expected consequences** of technical decisions;
- explain **what a tool or feature will do** in a project;
- make **simple predictions** related to code quality, development process, or teamwork.

Typical situations include:

- explaining why a team chooses TypeScript;
- describing what will happen if a feature is implemented;
- presenting expected benefits in a short technical report or oral presentation.

Form and Use of will

Use	Form	Example (Context: Programming)
Future result	will + base verb	TypeScript will reduce errors in large projects.
Prediction	will + base verb	This feature will improve code readability.
Expected consequence	will + base verb	Static typing will help developers work faster.
Professional explanation	subject + will + verb	The compiler will detect problems early.

Note: Will is used with **all subjects** (I, you, we, they).

Examples

- Using TypeScript **will make** the code easier to maintain.
- This tool **will help** teams avoid common mistakes.
- The project **will need** an extra compilation step.



Let's practice!

1. Complete the sentences using will and the verb in brackets.

- TypeScript _____ (help) developers find errors early.
- Static typing _____ (improve) code quality.
- This feature _____ (make) large projects easier to manage.
- The compiler _____ (show) errors before production.

PRACTICE SECTION

SPEAKING TASK: Supporting a decision



● Situation 1: You are part of a development team meeting.

A teammate asks:

“Why should we use TypeScript in this project?”

Write 3 sentences explaining the expected results using will.

Example:

Using TypeScript will improve code structure.

1. Explaining a Technical Choice

Explain a technical decision using basic programming vocabulary and the future simple (will).

● Situation 2:

You are a junior developer working in a small development team. The team is starting a new web project.

The project manager asks:

“Why should we use TypeScript instead of only JavaScript?”

Work in pairs.

Student A: explains the technical choice.

Student B: listens and asks one follow-up question.

Then, change roles.



Instructions for Student A

Prepare a short explanation (1–2 minutes). Your explanation must include:

at least 3 vocabulary items related to languages, tools or code;

at least 2 sentences using will;

reference to future results or expected benefits for the project or the team.

Use the prompts below to organize your ideas. You do not need to use all of them.

Guiding Prompts

- TypeScript is a _____ (language / tool / extension).
- It is used for _____ (projects / applications / teams).
- Using TypeScript will _____ (reduce / improve / help).
- This feature will help developers _____.
- In this project, TypeScript will _____.

Word Bank – Technical Vocabulary and Actions

Languages & Tools

programming language – tool – feature – framework – application – project – code

Code Elements

function – variable – object – property

Actions & Results

help – improve – reduce – make – detect – organize – maintain

Common Expressions

work better – avoid errors – large projects – development process – development team

Instructions for Student B

Listen and ask one question, for example:

- What will this tool do for the project?
- How will it help the team?
- Model Example

*TypeScript is a programming language based on JavaScript. Using TypeScript will reduce the production time and avoid errors.

04

Written Production



In this section, you will practice writing a short technical text about TypeScript. You will use the ideas from the Reading section as a model to organize your writing and express your opinions clearly. The activities will help you move from guided writing to independent production, using technical vocabulary, will to talk about future results, and connectors to show contrast.

STEP 1: Read the Model



- 1. Before writing, read again the text in Section 1 (Reading) about TypeScript as the evolution of JavaScript. Use it as a model of content and organization.**

JavaScript is a very flexible language. This flexibility allows developers to write code quickly, but it can also create problems in large projects. When applications grow and there are no clear rules or structures, the code can become difficult to maintain.

TypeScript is considered the natural evolution of JavaScript. It is a superset of JavaScript, which means that all JavaScript code is valid in TypeScript. The main difference is that TypeScript adds static typing and advanced development tools. With static typing, developers can detect errors during development and not in production. For example, if a function expects a number, the compiler will show an error if a string is used. This improves code quality, readability, and maintainability.

TypeScript code is written in its files and then transpiled into JavaScript. This step is necessary because browsers only understand JavaScript. Despite this extra step, TypeScript is widely used because it helps teams work better in large projects. TypeScript also improves the developer experience with intelligent autocompletion and better documentation inside IDEs. For this reason, it is used in many modern frameworks such as Angular, React, and NestJS.

Step 2: Controlled Production

1. Guided Writing

a. Complete the paragraph using the words from the box.

Word bank

extension – static typing – errors – although – project – code

TypeScript is an _____ of JavaScript. It adds _____, which helps developers detect _____ during development. This is useful in a _____ environment. _____ TypeScript requires more _____, it improves maintainability and reliability.

2. Answer the question:

What advantage of TypeScript is described in this text?

.....

3. Sentence Building: Write one sentence for each prompt.

Describe what TypeScript is.

→ TypeScript is a _____.

Describe one advantage using will.

→ Using TypeScript will _____.

Describe one limitation using a contrast connector.

→ Although _____, TypeScript _____.



STEP 3: Free Production – Technical Paragraph

d. Write a short paragraph (80–100 words) explaining the use of TypeScript in a professional project.

Use the Reading text (Section 1) as a reference for:

- content,
- organization,
- and level of formality.

Your paragraph must include:

- a brief description of what TypeScript is,
- at least one advantage and one disadvantage,
- will to express future results,
- one contrast connector (although, though, despite, in spite of, because of).

Writing Prompts : You may use these prompts to organize your ideas:

- TypeScript is a _____ based on JavaScript.
- It is used in _____ projects.
- Using TypeScript will _____.
- Although _____, it _____.
- In conclusion, TypeScript is _____.

Word Bank – Technical Vocabulary

Languages & Tools

programming language – tool – feature – project – application – code

Actions & Results

help – improve – reduce – maintain – detect – organize

Quality & Process

errors – readability – maintainability – development process – team

Writing Checklist – Self-assessment

Before submitting your text, check:

Content & Genre

- ☐ I describe what TypeScript is.
- ☐ I mention at least one advantage and one disadvantage.
- ☐ My ideas are clear and relevant to a technical context.

Language Use

- ☐ I use will to talk about future results.
- ☐ I use at least one contrast connector correctly.
- ☐ I use technical vocabulary from the module.

Organization & Clarity

- ☐ My paragraph has a clear beginning, middle and ending.
- ☐ My sentences are connected and easy to follow.



05

Speaking Production



In this section, you will practice speaking about a technical topic in a clear and organized way. You will prepare a short oral presentation to explain what TypeScript is, compare it with JavaScript, and justify a technical choice. This activity will help you develop confidence when speaking in professional and academic contexts, using simple and effective language.

Oral Presentation – TypeScript vs JavaScript

Work individually or in pairs.

Prepare a short presentation (2–3 minutes):



1. **Introduction:** What is TypeScript?
2. **Development:** Advantages and basic features
3. **Conclusion:** Why developers use it

Use expressions such as:

In my opinion...

One advantage is...

In conclusion...

Self-assessment Checklist

- ☐ I can explain what TypeScript is.
- ☐ I can compare TypeScript and JavaScript.
- ☐ I can justify a technical choice in English.
- ☐ I can give a short oral presentation.



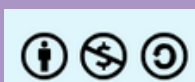
See Assessment Rubric – Oral Presentation



Criteria	4 – Excellent	3 – Good	2 – Developing	1 – Needs Improvement
Content accuracy & relevance	Explains clearly what TypeScript is, compares it accurately with JavaScript, and justifies	Explains TypeScript and compares it with JavaScript with minor inaccuracies. Gives a	Information is limited or partially correct. Comparison is unclear or incomplete.	Information is mostly incorrect or very unclear.
Organization & genre format	Clear structure: strong introduction, logical development, and effective conclusion.	Structure is present but one part could be clearer or more developed.	Some structure, but ideas are not well organized.	No clear structure.
Use of target language (expressions & connectors)	Uses several required expressions accurately and naturally (<i>In my opinion...</i> , <i>One</i>	Uses some required expressions correctly.	Uses few expressions or with errors that affect clarity.	Does not use target expressions.
Technical vocabulary	Uses technical vocabulary accurately and appropriately (<i>framework, feature,</i>	Uses some technical vocabulary with minor errors.	Uses very limited technical vocabulary.	Does not use relevant technical vocabulary.
Fluency & interaction	Speaks fluently with good pacing and natural flow.	Speaks clearly with some hesitation, meaning is always clear.	Frequent pauses and hesitation affect communication.	Very hesitant; difficult to follow.
Pronunciation & intonation	Clear pronunciation and effective intonation; easy to understand.	Mostly clear pronunciation with minor issues.	Several pronunciation problems that affect understanding.	Pronunciation seriously limits comprehension.

Bibliography & Web Resources

- Oxford English for Information Technology – Glendinning & McEwan
- Cambridge Dictionary: <https://dictionary.cambridge.org>
- Olejniczak, M (2011). English for Information Technology.
- Vocational English Coursebook. Pearson Education Limited.



English II, Module 4 booklet. Tecnicatura Universitaria en Programación a Distancia.UTN. © 2026 by M. Antonella Diblasi is licensed under CC BY-NC-SA 4.0