



AI for Children

Artificial Intelligence Curriculum for Elementary and Secondary Schools

Card

Creating Simple Applications Using Vibe Coding

01

Vibe Coding Deck



kurikulum.aidetem.cz/cards

Created by: Eva Nečasová

Expert guarantor: Pavel Kordík

Language proofreading: not yet done

Last update: 04/2026

Version: 02

npi | National Pedagogical Institute
of the Czech Republic

We create teaching materials in cooperation with the National Pedagogical Institute.



[Form for
comments](#)

Teaching Material for the AI Curriculum for Elementary and Secondary Schools
Computer Science at Secondary Schools - Vibe Coding

Creating Simple Applications Using Vibe Coding

What is vibe coding?

Vibe coding is a way of creating applications without writing code. Instead of programming, the student gives instructions to an AI tool using natural language, describing what the application should do and gradually refining the result. In this process, they learn to formulate clear prompts, test outputs, and improve them without needing to know a specific programming language.

Available applications

A wide range of tools for vibe coding is available today, allowing users to create applications using natural language. Well known examples include [Claude Code](#), [OpenAI Codex](#), [Google Antigravity](#), [Google AI Studio](#), [Canva Coding](#), and [Lovable](#). These tools make it possible to generate code, design application structures, or build simple web projects based on text prompts.

In this teaching material, we work with the application [Macaly](#). The main reason is that it was developed in the Czech Republic, where this teaching material originates, and it offers a simple, user friendly interface. At the same time, it currently offers the highest amount of free credits. Another tool that would otherwise be a good fit for teaching, Google AI Studio, currently has age restrictions, as it is only available to users over 18.

As an additional recommended tool, Canva Coding can also be used. If you decide to use a different tool, please make sure to check the age limits of the application.

Vibecoding Deck

Creating Apps and Websites



[Lesson presentation in PDF](#)



[Activity board in Canva](#)

These teaching materials were translated using ChatGPT.
Please note possible imperfections in the expressions or wording.

Note 2: Gender equality is a key value for AI for Children, but to keep our teaching materials concise, we use masculine grammatical forms.

Preparation for teachers

Introduction to the Macaly application

Macaly provides a complete knowledge base, available at: macaly.com/docs/en/welcome/overview

Login

Students register for the Macaly application via the macaly.com website by clicking the “Get started” button. We recommend signing up using school accounts for privacy reasons. Students can register with an email address or sign in using a Google account. After logging in, they are automatically taken to the application environment, where they can begin creating a new project. For classroom use, we recommend checking in advance that students have access to email so they can complete the registration process.

Getting started

After logging in, a chat window appears where students can begin a conversation. They can communicate in English or other languages. Images, documents, audio, and video files can be added using the + button in the bottom left corner of the chat window. Next to the circular A icon, students can choose between different communication modes, which are described below.

- + Auto (default): In this mode, the Macaly agent decides whether to simply chat, create a plan, or immediately start generating application code. This mode is suitable for beginners.
- + Planning: The agent first prepares a plan or asks clarifying questions. This mode is useful for more complex tasks or when students want to discuss the assignment first.
- + Build: The agent directly generates the source code for the application or website.

It is also possible to communicate with the agent using voice. This feature can be activated in the bottom right corner of the chat window. However, we do not recommend using it in class with all students at once, even if they have headphones with a microphone.

In the bottom right corner of the chat window, students can also see how many credits they have remaining for the current month.

How to assist tasks to the agent

Students first describe in plain language what they want to create and then press Enter. A simple rule applies here: the clearer and more specific the prompt, the better the result.

Recommended assignment structure:

- + who the application is for,
- + what its purpose is,
- + what it should look like,
- + how it should work.

This prompt should be preceded by a design process, which is described in this teaching material.

After submitting the first prompt, the interface splits into two parts. On the left is a panel for entering instructions and making edits, while on the right there is a preview of the current version of the application. In the left panel, students add further instructions in natural language or upload files. They can also use the “Visual Edit” feature for quick adjustments such as fonts, colors, or corner rounding. Students can apply changes to the whole project or adjust a specific element in the right hand preview, then confirm the changes by clicking “Save”.

Students continuously review the result, test functionality, and gradually add further refinements. We recommend focusing on one specific feature or visual element at a time to avoid errors.

Creation is therefore not a one step action, but an iterative process of gradual improvement. Students learn to respond to AI outputs, fix issues, and systematically enhance their application.

Publishing output

When students are satisfied with the result, they can publish it. In the top right corner of the interface, there is a “Publish” button. In the free version, Macaly only allows public publishing and automatically assigns a public URL where the application or website is displayed.

Troubleshooting

There may be times when the agent does not carry out the task exactly as instructed. This is a natural part of vibe coding. Let students know that this can sometimes be frustrating, but it is worth approaching the problem creatively and exploring alternative solutions, such as rephrasing the prompt, using the “Visual Edit” feature, or providing an example of how the output should look. Unfortunately, making manual edits or downloading the source code is not possible in the free version.

If Macaly becomes unresponsive or takes too long to react, the page can be reloaded or the application can be restarted. This option can be found in the right hand preview panel above the output display (refresh icon).

Lesson Overview

Recommended Age, Lesson Length

Children aged 13–16, 90–160 minutes.

Building Blocks

Vibe coding, applications, user experience (UX).

What Are the Students Learning?

They design and create a simple application using vibe coding.

Why Are They Learning This?

Students develop the skill of designing and creating applications based on user needs.

How Do We Know They Have Learned It?

Students design and create a meaningful and functional application using vibe coding.

Tools

Teacher: Projector, presentation for display.

Students: Writing materials, sticky notes, a digital device for each student or pair, registration in the Macaly or Google AI Studio application.

Digital Competence

Facilitating Learners' Digital Competence, Digital Content Creation.

Bloom's Taxonomy

Creating: Students design and build a simple application using vibe coding.

Evaluating: They evaluate and test the functionality and visual design of the application.

Five Big Ideas

5-B-I AI & Culture (AI in Daily Life).

5-C-II AI & The Economy (Effects on Employment).

Application Design Process

45–90
min

[Activity board
in Canva for
Students](#)

Introduction to the activity

In the initial vibe coding lessons, we recommend starting very simply. The goal is not to build a technically complex application, but to understand the process: formulating a prompt, observing the output, testing functionality, and gradually refining the details. That is why it is best to begin with static web pages or a simple game for one or two players.

The most important part is the application design process itself, which we describe below. You can guide students through this process using a [Canva board](#) that we have prepared for you.

Who is the application for?

The vibe coding process can be understood as the gradual refinement of an idea, from the initial concept to its implementation and sharing. The first step is to clarify who the application is for and what problem it is meant to solve. Students are encouraged to think about a specific user, such as a classmate, teacher, or parent, and imagine a situation in which they would use the application. This helps ground the entire project in a real world context.

What will the application do?

The next step is to consider what the application should do. Instead of focusing on programming, the emphasis is on logic: which features will be useful for the user, what should happen when different elements are clicked, and what problem the application actually solves. It is important to guide students toward simplicity and clarity so they do not get lost in overly complex ideas and can distinguish between what is essential and what is optional.

What will the application look like?

The following step is designing the application itself. This can be as simple as a sketch on paper. Students think about what the application will look like, what screens it will include, and how the user will move through it. The focus is not on visual design, but on basic structure and orientation. This step helps turn an abstract idea into a more concrete form that can be further developed.

Creating a prompt for AI

Once it is clear who the application is for, what it should do, and what it should roughly look like, the key phase begins: creating a prompt for the AI. Students learn to express their ideas in a way that artificial intelligence can understand. The prompt should be as specific as possible. This is where students develop the ability to clearly formulate and structure information, which is one of the core skills needed for successful vibe coding.

Building, testing, and improving the application

Based on this prompt, the AI generates an initial version of the application. After submitting the prompt, it usually takes about 5–10 minutes to generate. If it takes longer, we recommend refreshing the page. However, the process typically does not end with the first version. It is followed by working with the result, evaluating it, and gradually improving it. Students compare their expectations with the actual outcome, identify gaps, and adjust the prompt to refine the result. This cycle of iteration and refinement is essential. High quality solutions emerge over time, not on the first attempt.

Publishing and sharing the application

In the final stage, the application is published. This means sharing it with others, such as classmates or a wider group of users. They can try out the application and provide feedback, which can then serve as a basis for further improvements. In this sense, the process is brought to a close, while at the same time remaining open to continued development.

Summary

Vibe coding is not based on programming knowledge, but on the ability to think about user needs, break a problem down into smaller parts, and clearly formulate a prompt. For educators, this represents a shift from teaching technical skills to developing thinking, creativity, and the ability to communicate ideas effectively.

A few ideas for student projects

If students are unsure what kind of applications to create, we have prepared a few ideas for you, listed below. Through testing this teaching material, we are also gradually collecting examples of applications created not only by students, but also by educators.

You can find an overview of those currently available in English in this continuously updated [table](#).

Student personal website

A simple one page presentation that students can use, for example, when applying to a school or to introduce themselves. Develops: content structuring, working with layout, colors, and sections.

Content:

- + short introduction,
- + my interests,
- + favorite apps, games, or books,
- + "Contact" section (without a functional form).

School project website

For example: "A Day Without a Phone," "School Trip," or "Eco Challenge."

Develops: structuring content and working with visual hierarchy.

Content:

- + introductory banner,
- + project explanation,
- + schedule or timeline,
- + gallery (illustrative images).

Thematic informational website

For example: "How Artificial Intelligence Works," "Which Drinks Contain the Most Sugar," "How to Avoid Social Media Overload."

Develops: the ability to present content clearly and consider the target audience.

Content:

- + topic explanation,
- + key points overview,
- + tips and recommendations,
- + simple FAQ section.

Simple Quiz (without saving results)

For example: an AI quiz, an internet safety quiz, or a class themed quiz.

Features::

- + question,
- + answer selection,
- + instant feedback.

More ideas:

- + Excuse Generator for Being Late
- + Random School Challenge Generator
- + “What Meme Do You Need Today?” App
- + Dice Roll Simulator
- + Random Name Picker for the Class
- + “Yes/No/Maybe” Decision Button
- + Two player mini game “Click 10 Times as Fast as You Can” with time tracking
- + Coin Flip Simulator (heads or tails)
- + Digital “Rock Paper Scissors” against the computer
- + Stopwatch with a large start stop button
- + Traffic Light Simulator (reaction game for green light)
- + “Guess the Number” activity (without score saving)
- + Mini game “Find the Right Button” (randomly changing position)

Engage



Presentation slide 02

Share a simple game with your students that we created using vibe coding in Macaly.

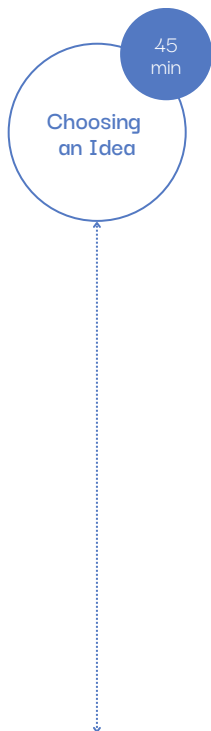
You can find the QR code in the presentation on slide 02, or share this URL with your students: macaly-kr52mxrbv12lltcypewv9a61.macaly.app/ or alternatively: bit.ly/quiz-vibecoding

Then discuss the concepts of application, vibe coding, and user experience (UX).

- + Application (program, software)
Any digital tool that performs a task based on predefined rules. This can be a mobile app, a website, a game, a calculator, or a phone alarm. The key aspect is that it is a program that responds to user input.
- + Vibecoding
A way of creating applications without traditional coding. Instead of programming, the user gives instructions to a generative AI tool in natural language. They learn to clearly formulate prompts, test outputs, and gradually improve the application.
- + User Experience (UX)
UX, or User Experience, refers to how a person feels when using an application and how easy it is to interact with it. Good UX means the application is clear, intuitive, and works as the user expects.

Explain to students that the quiz game (application) they just played was created using vibe coding. In other words, its creator described how the game should look and work, and the application designed for this purpose then generated it (programmed it). Students will now try to create their own application in the same way, either a simple game or a basic website. The goal is for their application to be user friendly and easy to navigate.

Understand



Presentation slide 03

Students think about the application they are going to create.

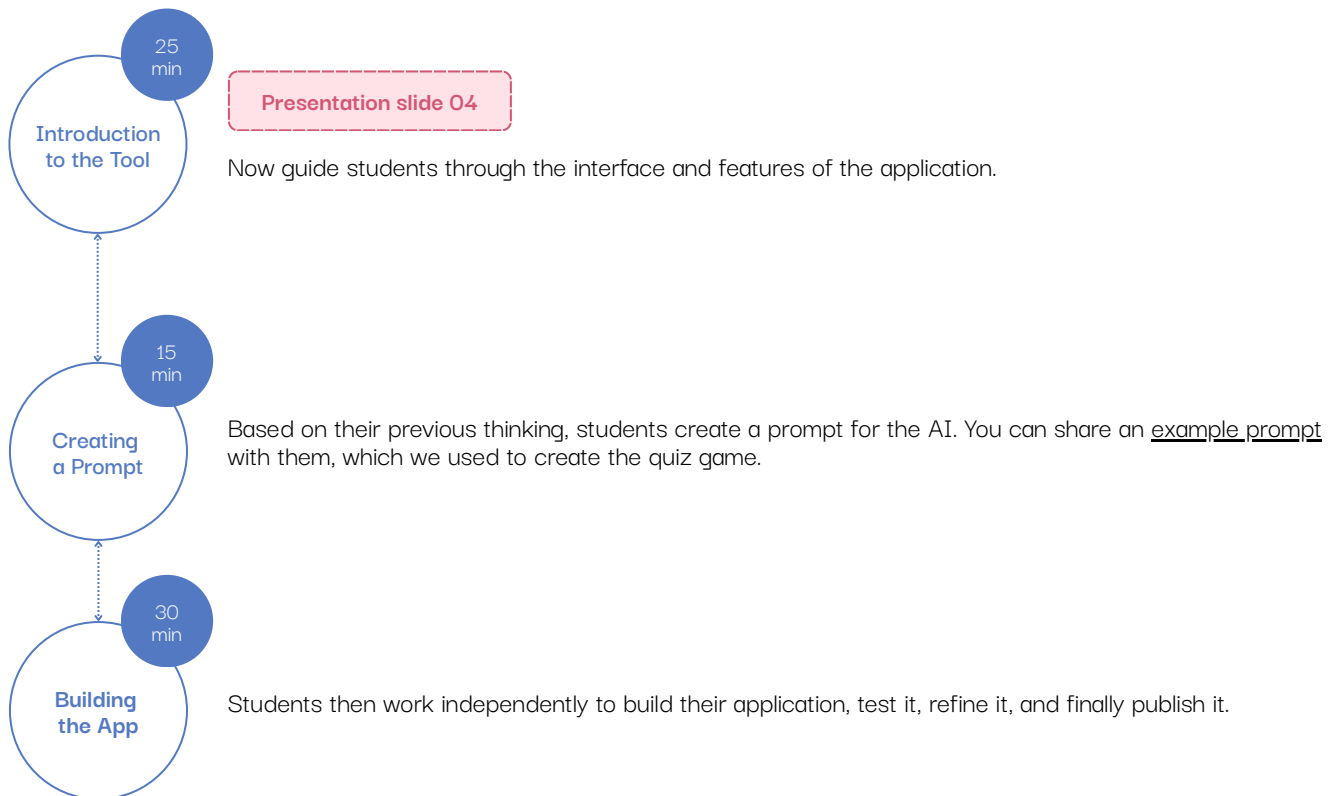
Using design thinking principles, students reflect on the application they will develop. They first consider who the application is for and what problem it should solve. Then they design what the application should do, what features it will include, and how the user will navigate through it, including a simple outline of its appearance.

We recommend using a [Canva board](#) for the design process, which we have prepared for this purpose. During the lesson, you can invite students to the board and guide them through individual stages, where they complete time limited tasks together.

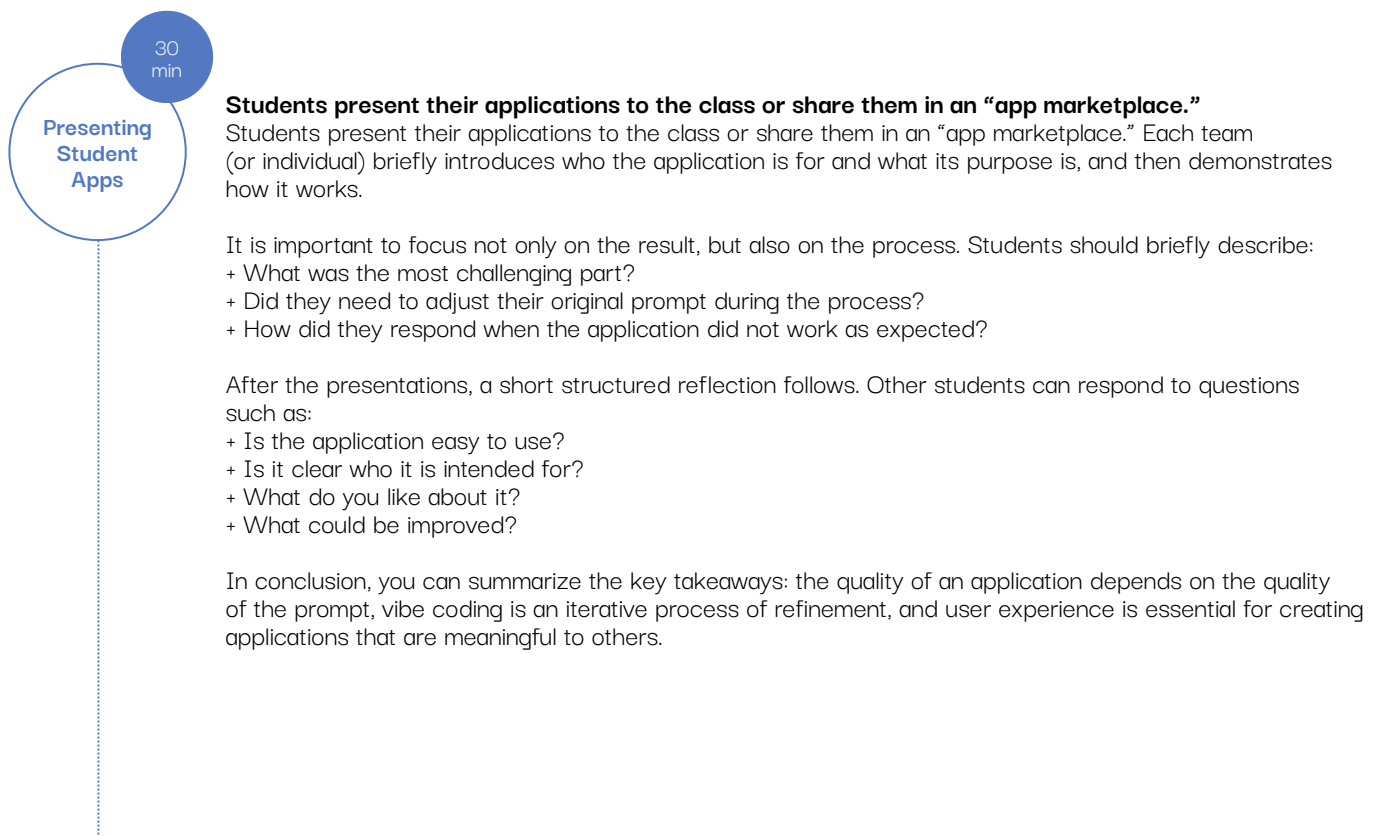
If you choose not to use the Canva board, it is helpful to follow this structure:

Students should define:

- + who the application is for,
- + what its purpose is,
- + what it should look like,
- + how it should work.



Reflect





How to set up an app marketplace

In the following lesson, students can create a short presentation of their application, for example using Canva:

- + app name,
- + target audience,
- + how it works,
- + screenshots,
- + what worked well and what they would improve,
- + link to the application.

Students then share their presentation with classmates (via a link or QR code). Each student or group selects at least one application to test.

Testing and feedback

Students try out each other's applications and provide short structured feedback, for example:

- + what works well,
- + what is unclear,
- + what could be improved,
- + how it felt to use the application.

It is important to emphasize respectful and constructive communication: we evaluate the application, not the creator.

Goal of the Marketplace

The app marketplace helps develop the ability to present one's work, receive feedback, and think about user experience.

Students realize that an application only becomes meaningful when someone else actually uses it.