

Recommender Assistant

A tool for the general public that enhances user interaction through feedback on bad smells in code.



List of Contents

Introduction	2
Goals	2
How it works	2
Usage Examples	3
Access from Main Page	3
Access from Evaluated Scratch Project (Default/Personal)	5

List of Figures

1	"Recommender Assistant" section on the Dr. Scratch homepage	3
2	Feedback for the "Backdrop Naming" <i>bad smell</i>	3
3	Feedback for the "Duplicate Scripts" <i>bad smell</i> (refactor tab)	3
4	Feedback for the "Duplicate Scripts" <i>bad smell</i> (identification tab)	4
5	Recommender Assistant Section in Dr.Scratch Dashboard Page	5

Introduction

Another problem that has been approached is adapting to more accessible user interfaces according to new trends in *Chat Bots* on the internet. For this reason, a tool has been created that collects the identified *bad smells* (duplicate and dead code, *sprites* and *backdrops* with the default name) already existing in the native analyses of Dr. Scratch, and an interactive system has been developed to guide the user in identifying and solving the *bad smells* in their project.

Goals

The goals are to reach a general audience that, due to advances in artificial intelligence, has developed a clear tendency to interact with this technology through the well-known "Chat bots". These tools simulate a written and bidirectional communication flow similar to a messaging chat. Therefore, this concept has been adapted, with some differences, to fit into an interactive real-time workflow with Scratch projects.

How it works

To use the tool, you need to look for the section corresponding to this mode, *Recommender System*, on the Dr. Scratch *homepage*. To analyze our *bad smells*, you only need to use the *URL* corresponding to that project. It should be noted that for this mode, it is not possible to analyze projects previously downloaded locally, as a continuous connection to the Scratch servers where the project is stored is required to establish a didactic dialogue that guides the user in improving these *bad smells*. Therefore, once the project is analyzed, the following path will be followed:

- **Detection of *bad smells*:** the system detects all existing *bad smells* in the project and provides feedback on one of them, following the hierarchy: duplicate code, dead code, *sprites* with the default name, *backdrops* with the default name. The *feedback* given to the user has the following structure:
 - **Introduction:** to make it more user-friendly, the dialogue begins with a small joke before giving the *feedback*.
 - **Identification of the *bad smell*:** what type it is, along with relevant and concise information about it. This includes information on where the *bad smell* can be found and what steps the user should follow to resolve it.
 - **Additional information (only for duplicate and dead code):** in the case of duplicate and dead code, identification is facilitated through a graphical representation of how it would appear within the Scratch application, also providing the name of the *sprite* where such blocks can be found.

If feedback is given about duplicated scripts, a two-tab interface is shown, where the first tab displays the duplicated code and the *sprite* where it is located, and the second tab shows the refactored code to avoid duplication. Note that if there are multiple duplicated scripts in the project, only one of them will be shown in each *feedback* to avoid overwhelming the user.
 - **General explanation:** in this section, a brief explanation is provided on what the identified *bad smell* consists of as a concept within the global scope of programming and why it is important to solve it, accompanied by a very simple example to reinforce understanding.
 - **Farewell.**
- **Solution:** at this step, the user should try to solve their *bad smells* by following the steps specified in the *feedback* provided by Dr. Scratch, as mentioned above.
- **Verification:** in this step, the user is given the opportunity to verify whether their changes have been effective in the project or if the previously announced *bad smells* have not been resolved. To perform the verification, the user only needs to click the "Fixed it!" button in the bottom left corner of the *feedback* box. Once clicked, without needing to do anything else, the user should wait for a short interval of time for the project to be reanalyzed for *bad smells*. After the analysis is complete, there are two possible scenarios:

- **The previously announced *bad smell* has been resolved:** in this case, the user is congratulated for solving the *bad smell* and informed of the next detected *bad smell*, if any. If not, it will be stated that their project is free of *bad smells*.
- **The previously announced *bad smell* has not been resolved:** in this case, a supportive message is shown to encourage the user, and a similar explanation to the one previously given is provided on how to solve the *bad smell*.

This cycle can be repeated as many times as necessary, or until the project has no *bad smells*, in which case congratulations will be given.

Usage Examples

Access from Main Page

To use this mode, the first step is to locate the "Recommender Assistant" section on the Dr. Scratch main page (Figure 1). Next, enter the URL of the Scratch project and click the *Analyze my Bad Smells* button.



Figure 1: "Recommender Assistant" section on the Dr. Scratch homepage

After submitting your project, you will receive recommendations on how to address any detected bad smells, as illustrated in Figures 2, 3 and 4.

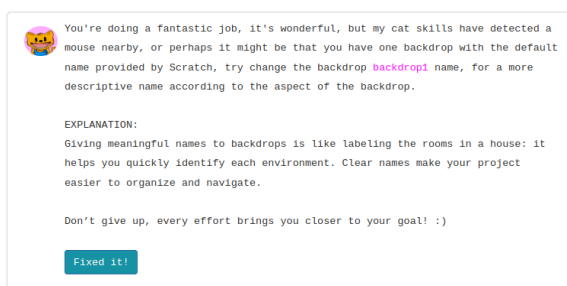


Figure 2: Feedback for the "Backdrop Naming" *bad smell*

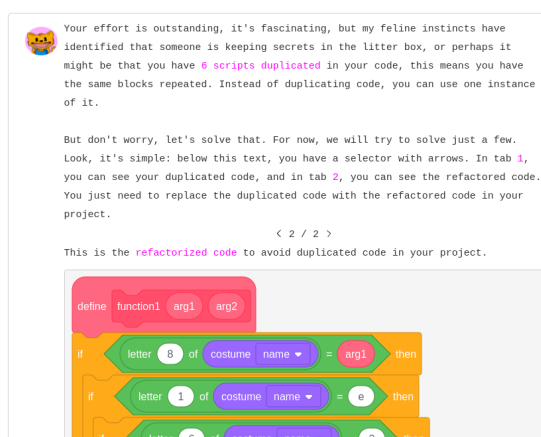
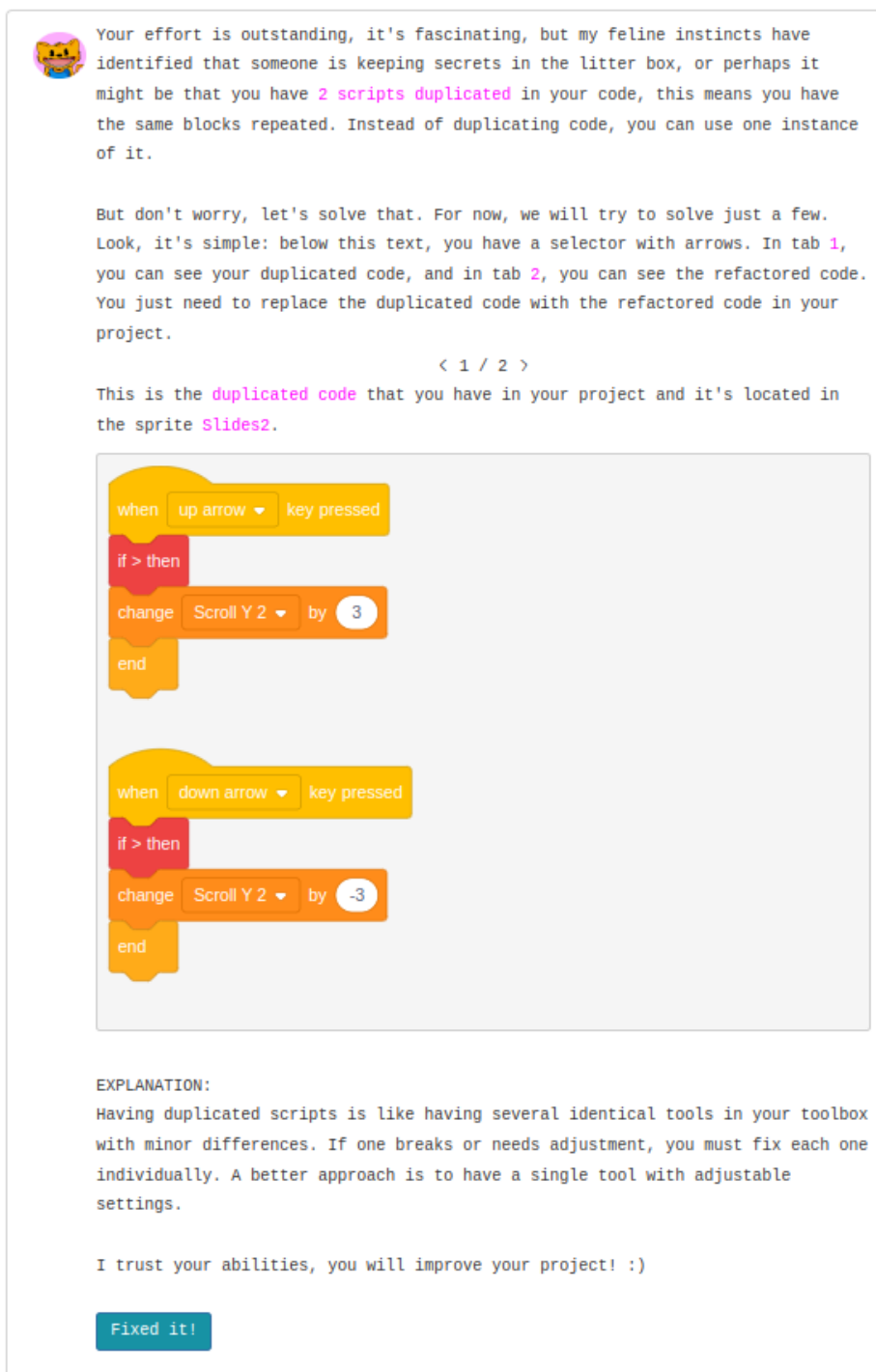


Figure 3: Feedback for the "Duplicate Scripts" *bad smell* (refactor tab)



The screenshot shows a feedback message from the Scratch team. It starts with a cat icon and a message explaining that duplicated scripts are inefficient. It then shows two examples of code blocks: one for the up arrow key that increases 'Scroll Y 2' by 3, and another for the down arrow key that decreases it by 3. Below the code, there is an 'EXPLANATION' section and a 'Fixed it!' button.

Your effort is outstanding, it's fascinating, but my feline instincts have identified that someone is keeping secrets in the litter box, or perhaps it might be that you have 2 scripts duplicated in your code, this means you have the same blocks repeated. Instead of duplicating code, you can use one instance of it.

But don't worry, let's solve that. For now, we will try to solve just a few. Look, it's simple: below this text, you have a selector with arrows. In tab 1, you can see your duplicated code, and in tab 2, you can see the refactored code. You just need to replace the duplicated code with the refactored code in your project.

< 1 / 2 >

This is the duplicated code that you have in your project and it's located in the sprite `slides2`.

```
when up arrow key pressed
  if > then
    change Scroll Y 2 by 3
  end

when down arrow key pressed
  if > then
    change Scroll Y 2 by -3
  end
```

EXPLANATION:
Having duplicated scripts is like having several identical tools in your toolbox with minor differences. If one breaks or needs adjustment, you must fix each one individually. A better approach is to have a single tool with adjustable settings.

I trust your abilities, you will improve your project! :)

[Fixed it!](#)

Figure 4: Feedback for the "Duplicate Scripts" *bad smell* (identification tab)

Once you have addressed these issues, you can recheck the project by clicking the *Fixed it!* button. This will analyze the project again to confirm if the issues have been resolved and to detect any remaining or new bad smells.

Access from Evaluated Scratch Project (Default/Personal)

Another way of using this tool is accessing from the Dashboard of the **Default Mode** or the **Personal Mode**. The only thing you will need to do is to point the cursor on the image of the Dashboard, in this case the Monkey (Figure 5).

The screenshot displays the Dr. Scratch dashboard interface. At the top, there is a logo for 'Dr. Scratch' with the tagline 'Analyze your Scratch projects here!'. Below the logo are two buttons: 'Vanilla Mode' and 'Extended Mode'. The main content area is divided into several sections:

- Monkey Tip:** A section featuring a cartoon monkey and a tip box that says 'Want a Banana-Tip? Click me and I will show you a way to make your project Monkey-rrific!!'. Below this, it states 'The level of your project is... **DEVELOPING!** You're doing a great job. Keep it up!!!' and includes a link to 'Come back to your Scratch project.'.
- Bad habits:** A section listing project issues: '0 duplicated scripts.', '7 sprite naming.', '4 backdrop naming.', and '4 dead code.'.
- Project certificate:** A section showing a URL 'https://scratch.mit.edu/projects/957967074/' and a 'Download' button.
- Compare project:** A section with a text input field containing 'http://scratch.mit.edu/projects/your_number', a question 'Is the new project an update from the original?' with 'Yes' and 'No' radio buttons, and a 'COMPARE PROJECTS' button.
- Level up / Level:** A large section on the right showing a progress bar for various coding concepts:

Concept	Progress
Abstraction	1 / 3
Parallelism	1 / 3
Logic	0 / 3
Synchronization	3 / 3
Flow control	3 / 3
User interactivity	1 / 3
Data representation	3 / 3

Figure 5: Recommender Assistant Section in Dr.Scratch Dashboard Page

Once you click that image it will open a new tab, where it will show the same information as the one if you access from the Main Page.