

The logo consists of the letters 'SK' in white, bold, sans-serif font, centered within a blue rounded square.

Livery Wizard

Turn a picture of a livery into a finished FiveM texture template.

This guide walks through every step of the app: getting an OpenAI key and putting money on it, then each of the five screens, what every option does, what a run costs, and how to check and fix the result on the real 3D vehicle.

Sals Kewl Scripts · October 2026

Contents

1. Before you start: what you need
2. Getting an OpenAI key and adding money
3. Installing it, and keeping it up to date
4. Step 1 — Key and models
5. Step 2 — Template, panels and the 3D model
6. Step 3 — Picture, vehicle and mode
7. Step 4 — Run
8. Step 5 — Result and export
9. Checking it on the 3D model and fixing a spot
10. What a run costs
11. Getting a better result
12. If something goes wrong

1. Before you start: what you need

- **The app.** `SalsKewlLiveryWizard.exe` . Nothing to install: double-click it. Windows 10 or 11.
- **An OpenAI account with a few dollars of credit.** The app uses your own account, so you pay OpenAI directly for what you generate. Section 2 shows exactly how to set that up. Budget about a dollar per livery.
- **The vehicle's livery template PNG.** The texture sheet with the panel outlines, usually pulled out of the vehicle's `.ytd` with OpenIV or CodeWalker. Normally 2048 or 4096 pixels square.
- **A picture of the livery you want.** A promo sheet, a screenshot, a photo, a design someone sent you. It can be any vehicle, at any angle: it does not have to be the vehicle you are making the template for.
- **Optional: the vehicle's model files.** The `.yft` and `.ytd` from the vehicle's `stream` folder. With these the app shows your livery on the real 3D vehicle while it works, and lets you mark problem spots on the body.

Your key stays on your machine. It is saved in your user folder, in a file named `.sals_kewl_livery_wizard.json` , and is only ever sent to OpenAI. It is not inside the app and it is never shared.

2. Getting an OpenAI key and adding money

This is a one-time setup, about five minutes. An OpenAI API account is separate from a ChatGPT subscription: having ChatGPT Plus does **not** give you API credit, and API credit does not give you ChatGPT.

Create the account

1. Go to **platform.openai.com** and sign in, or create an account.
2. If it asks you to create an organization, any name is fine, for example your server name.

Put money on it

1. Open **Settings**, then **Billing**.
2. Click **Add payment details** and enter a card.
3. Click **Add credit** (sometimes "Add to credit balance") and buy an amount. The minimum is usually \$5, and that is a sensible first top-up: roughly five liveries.
4. Optional but recommended: set a **monthly budget limit** on the same page, so nothing can run away from you. \$20 is plenty.
5. Optional: **auto recharge** tops the balance up automatically when it runs low. Leave it off if you would rather top up by hand.

It has to be prepaid credit. If the balance is zero, the app will stop with a message saying you have no credits remaining. Adding credit fixes it immediately; no need to restart the app.

Create the key

1. Still on platform.openai.com, open **API keys** (under Settings, or at the top of the dashboard).
2. Click **Create new secret key**. Give it a name like "Livery Wizard".
3. **Copy it now.** It starts with `sk-` and is only shown once. If you lose it, just delete it and make another.
4. Paste it into the app on Step 1. You will not have to do this again.

Treat the key like a password: anyone who has it can spend your credit. If it ever leaks, delete it on that page and create a new one.

Verification. OpenAI asks some accounts to verify an identity before the image tools can be used. If the app reports that your organization must be verified, do it on the same settings pages and then try again.

3. Installing it, and keeping it up to date

The program is a single file. You can run it straight from your Downloads folder and it will work, but it is better to install it: one click, no administrator rights, nothing written outside your own user folder.

Installing

1. Double-click `SalsKewlLiveryWizard.exe` .
2. A bar across the top offers **Install on this PC**. Click it.
3. It copies itself to your programs folder, adds a **Start Menu** entry and a **Desktop** shortcut, and restarts from its new home. The bar does not come back.

After that, launch it from the Start Menu or the desktop icon like any other program. You can delete the file you originally downloaded.

Prefer not to install? Click **Not now** and keep using it as a single file. Everything works the same; you just will not get updates offered.

Updating

Each time it starts, the app quietly checks whether a newer version has been published. If there is one, a bar appears at the top saying so, with an **Update and restart** button. Click it and the app downloads the new version, checks it is intact, swaps itself out and reopens. It takes about a minute on a normal connection. Click **Later** to be reminded next time.

If you are offline, or there is nothing new, you will never see the bar.

Removing it

It appears in Windows **Settings, Apps, Installed apps** as "Sals Kewl Livery Wizard". Uninstall it there, or run the program once with `--uninstall` . Your OpenAI key, your settings and your past runs are left where they are, in your user folder.

Nothing is installed system-wide. The program lives in `%LOCALAPPDATA%\Programs\SalsKewlLiveryWizard` , which belongs to your Windows account. That is why no administrator prompt appears, and why it cannot affect anyone else using the PC.

Windows SmartScreen. The first time you run a newly downloaded copy, Windows may show a blue "Windows protected your PC" box, because the program is new and not yet widely seen. Click **More info**, then **Run anyway**. This is normal for small-publisher software.

4. Step 1 — Key and models

SK

Livery Wizard

picture in, FiveM livery out

Start over

1. KEY & MODELS

2. TEMPLATE

3. PICTURE & VEHICLE

4. RUN

5. RESULT

OpenAI key and models

Everything runs through OpenAI. The key is stored in your home folder and used only for these requests.

OpenAI key

key ready

list this key's models

Model for views

gpt-6.1-sol

Model for sheet & panels

same as above

Image quality

medium

gpt-6.1 is the default because it lays the livery onto the template most accurately. A chat model drives the image tool and keeps all pictures in one conversation; an image model (gpt-image...) draws directly. The image tool bills the same either way, so a cheaper chat model saves little.

PICTURES

3D MODEL

appear here as they are made

The template, your picture, every view the model makes, the painted sheet, the sharpened panels and the finished livery show up here.

Back

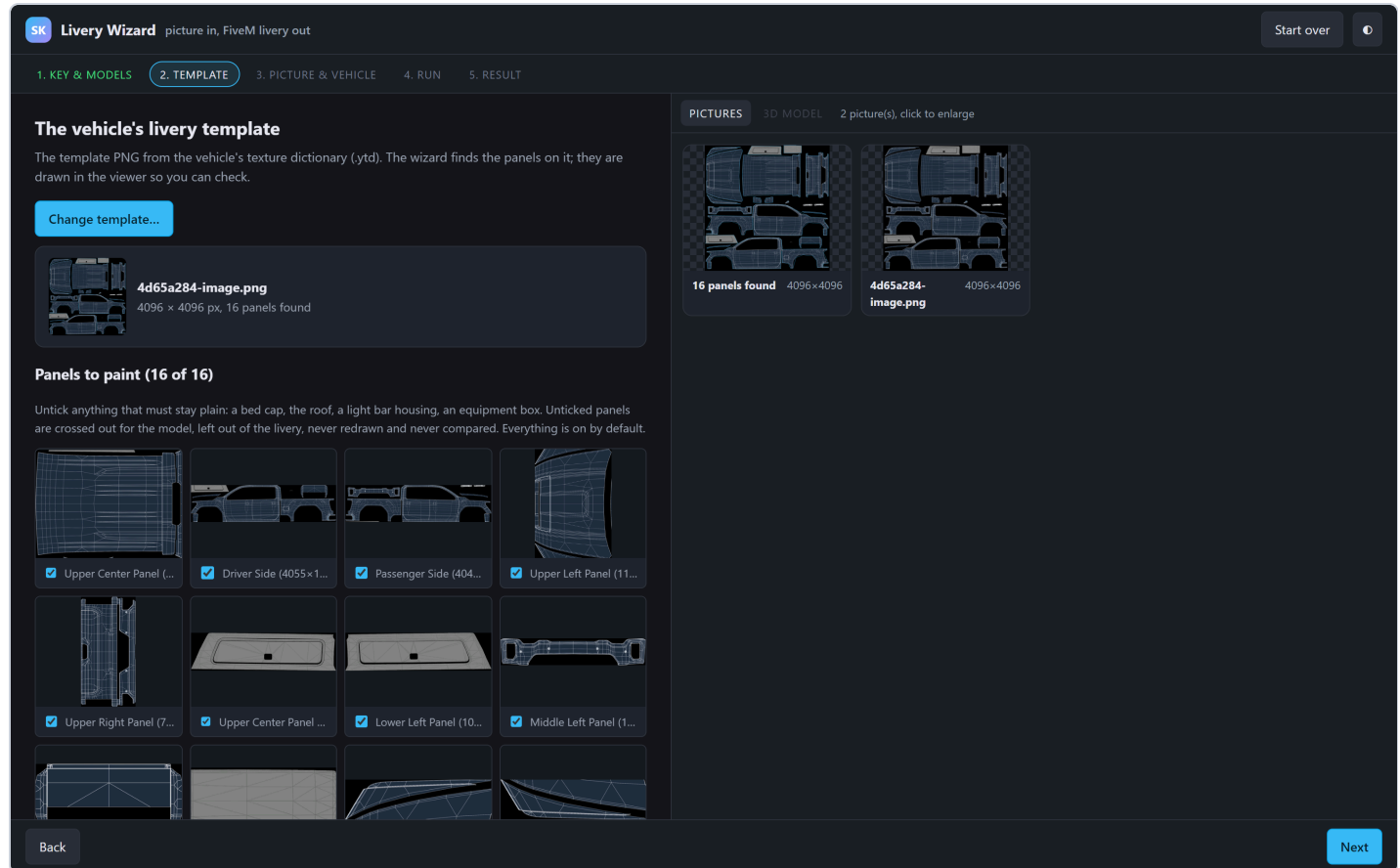
Next

The first screen. Paste the key once and it is remembered.

Option	What it is
OpenAI key	Paste the key you made in section 2. The badge turns to "key ready". It is saved immediately, and the field is left blank on later runs because it is already stored.
List this key's models	Optional. Fills the model boxes with the models your account can actually use. Handy if a model name is rejected.
Model for views	Leave it at <code>gpt-6.1-sol</code> . This is the model that looks at your picture and draws the vehicle wearing the livery. It gives the most accurate results of anything tested.
Model for sheet & panels	Leave blank to use the same model. Only set this if you want the template painting done by a different model.
Image quality	Leave at medium . Low is cheaper and softer, high costs about four times as much for very little gain on a texture that gets wrapped onto a vehicle.

Click **Next**.

5. Step 2 — Template, panels and the 3D model



Choose the template PNG. The app finds the panels and outlines them in the picture list on the right.

Choose template PNG

Pick the vehicle's livery template. The app detects every panel on the sheet (doors, hood, roof, tailgate, bumpers, mirrors) and shows them outlined so you can confirm it read the sheet correctly.

Panels to paint

Below the template you get a thumbnail and a tick box for every panel. Everything is ticked by default. **Untick anything that must stay plain**, for example a bed cap, the roof, a light bar housing, or an equipment box on a fire truck or ambulance. Unticked panels are marked off-limits for the model, left out of the livery, never redrawn and never checked. The same list also appears on Step 3, so you can change your mind right before running.

Vehicle model for the 3D preview (optional)

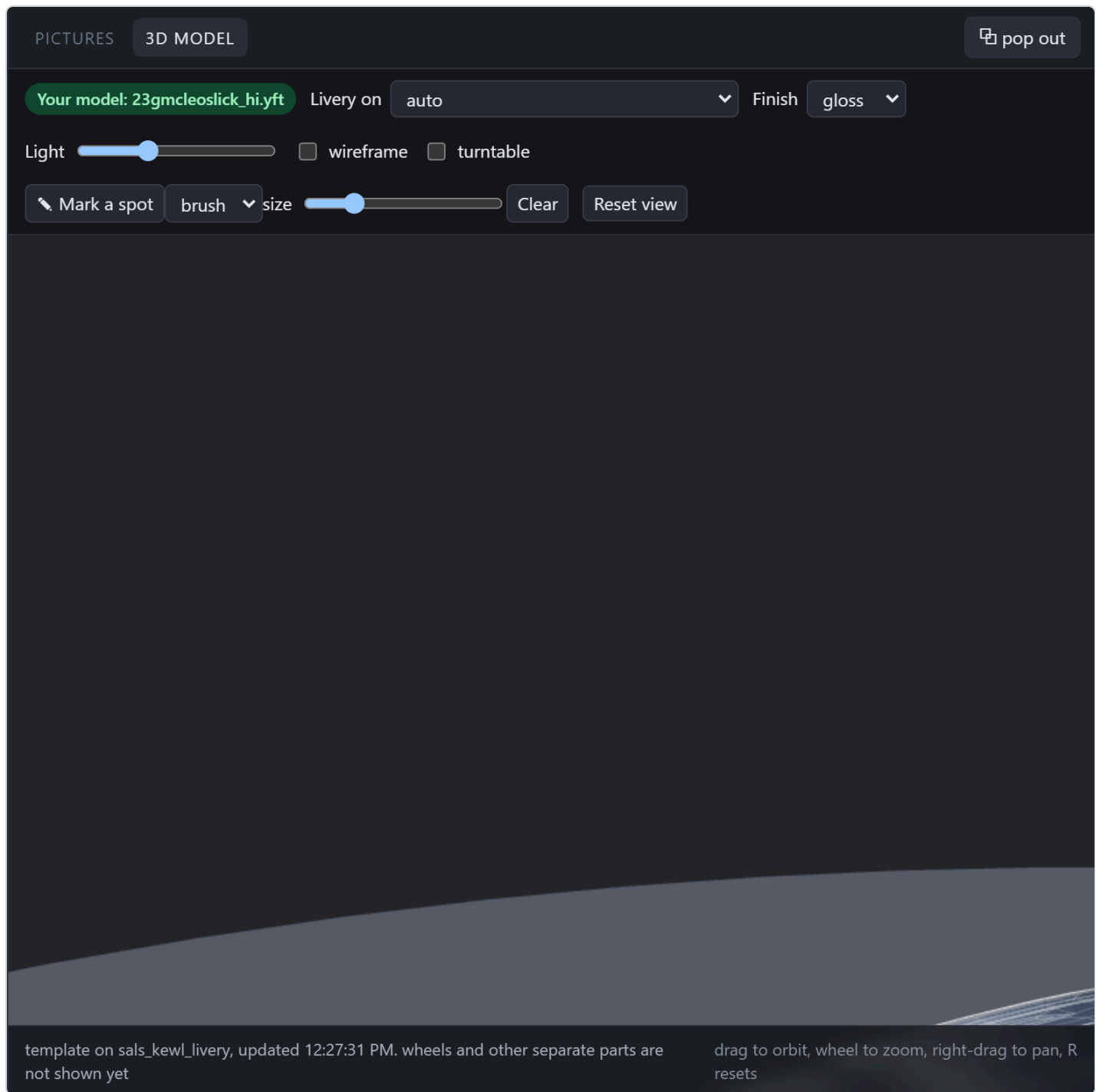
The image shows a selection interface for vehicle body panels. It consists of a 4x4 grid of 16 panels, each with a wireframe preview and a checkbox. The panels are labeled as follows:

- Row 1: Upper Center Panel (...), Driver Side (4055×1...), Passenger Side (404...), Upper Left Panel (11...
- Row 2: Upper Right Panel (7...), Upper Center Panel ..., Lower Left Panel (10...), Middle Left Panel (1...
- Row 3: Lower Right Panel (7...), Upper Right Panel 2 ..., Upper Left Panel 2 (...), Middle Left Panel 2 (...)
- Row 4: Middle Center Panel..., Middle Center Panel ..., Trim Strip (788×113), Trim Strip 2 (786×1...

Below the grid, the section is titled "Vehicle model for the 3D preview (optional)". It contains a paragraph explaining the process: "Pick the vehicle's .yft from its stream folder (the _hi one is used when present, and the .ytd next to it for the other textures). The finished livery is shown on the real body in a 3D window. Wheels and other separate parts are not shown yet." At the bottom of this section are three buttons: "Change model...", "Show 3D preview", and "remove".

Point it at the vehicle's .yft and it reads the real body, including its own textures.

Click **Choose .yft** and pick the vehicle's model file out of its `stream` folder. If there is a `_hi` version the app uses that, and it reads the `.ytd` next to it for the wheels, light bars and interior textures. A **3D model** tab then appears beside the pictures, showing your livery on the real vehicle as it is produced.



The 3D tab, right after loading the model: the bare template is on the body.

Separate parts such as wheels are not drawn yet. Vehicles protected with FiveM asset escrow cannot be read by this or any other tool; you would need the unprotected files.

6. Step 3 — Picture, vehicle and mode

The picture and the vehicle

One picture of the livery on any vehicle: a promo sheet, a screenshot, a photo. Then the vehicle this template belongs to, the way you would describe it to a person.

Add more pictures...

1 picture(s). One is enough; several views of the same livery help.



c8e56a1d-image.png

remove

Vehicle

2023 GMC Sierra Crew Cab Pickup

Changes & stipulations

no graphics on the bed cap

Anything here is applied at every stage: the rendered views, the painted sheet, the panel redraws and spot fixes. Describe changes to the design (colors, where the graphics go, what to leave off) or rules it must follow.

Lettering says

SAN ANDREAS, STATE POLICE, DUBBZ CITY, DIAL 911

Listing the exact words keeps the spelling right when panels are redrawn.

Views to render



driver side



passenger side



rear



front



hood



roof

Mode

Final



medium quality everywhere, 8 panel redraws, up to 4 corrections. About 14 image calls, around \$1.20. Panels you untick below and panels the sheet leaves plain are never redrawn.

Panels to paint (16 of 16)

Untick anything that must stay plain: a bed cap, the roof, a light bar housing, an equipment box. Unticked panels

Everything about the design you want goes on this screen.

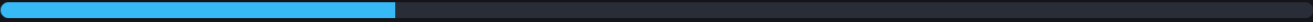
Option	What it is
--------	------------

Choose picture(s)	The picture of the livery. One is enough. Add more if you have other angles of the same design: they help the model get the back and the roof right.
Vehicle	The vehicle the <i>template</i> belongs to, described the way you would say it out loud: "2023 GMC Sierra Crew Cab Pickup". This is how the model knows what shape to draw.
Changes & stipulations	Free text, applied at every stage. Use it for changes to the design and for rules it must follow. Examples: "make the stripe green instead of yellow", "carry the mountains onto the hood", "no lettering on the roof", "use only the car in the middle row of the picture", "leave the bed cap plain black".
Lettering says	Type the exact words on the livery, separated by commas. This is what keeps the spelling right when panels are redrawn sharp.
Views to render	Which flat views of the vehicle to draw first. Driver side, passenger side and rear cover most liveries. Add front, hood or roof if the design wraps onto them. Each view is one image call.
Mode	Final, Economy or Preview. See section 10 for what each costs.
Panels to paint	The same tick list as Step 2, repeated here.
Redraw the eight biggest panels	Leave ticked. After the whole sheet is painted, the biggest panels are redrawn at full resolution so lettering and badges come out crisp.
Symmetric livery	Leave ticked for almost every police, fire and EMS livery. The passenger side is built as an exact mirror of the finished driver side and only the lettering is redrawn, so the two sides cannot disagree. Untick it only if the design is genuinely different on each side.
Check the output	Leave ticked. The app measures its own result against the template, fits every panel to its exact outline, compares the sides, and asks OpenAI to correct what does not match.

7. Step 4 — Run

Working...

Render 3 view(s) of "2023 GMC Sierra Crew Cab Pickup" with gpt-6.1-sol, paint the livery onto 4d65a284-image.png, redraw the big panels sharp. About 10 minutes, final mode.



```
Views of a 2023 GMC Sierra Crew Cab Pickup with gpt-6.1-sol; sheet and panels with gpt-6.1-sol;
medium quality. OpenAI only.
gpt-6.1-sol: rendering the driver side (1/3)...
  wrote sals_kewl_view_driver_side.png (1536x1024)
gpt-6.1-sol: rendering the passenger side (2/3)...
  deriving it from the mirrored driver side; lettering re-rendered to read normally
  wrote sals_kewl_view_passenger_side.png (1536x1024)
gpt-6.1-sol: rendering the rear (3/3)...
  wrote sals_kewl_view_rear.png (1024x1024)
gpt-6.1-sol: painting the whole sheet from 3 view(s)...
```

Stop

Progress, with every step written out as it happens. Stop is always available.

Press **Start**. A full run takes about ten minutes; Preview mode takes about three. In order, the app:

1. draws the flat views of your vehicle wearing the livery;
2. paints the livery onto the template sheet;
3. measures where each panel actually landed and fits it onto its real outline;
4. redraws the biggest panels at full resolution;
5. builds the mirrored panels from their finished partners and checks the rest.

Every picture appears in the list on the right the moment it is made, so you can watch it come together. Nothing is charged after you press **Stop**.

8. Step 5 — Result and export

Export

8 layers in 72 s, 15 image calls, about \$1.50 at list prices (final mode). The PNG is the game texture at the template's exact size; the PSD carries every layer for Photoshop.

To fix a spot: open the 3D model tab, press "Mark a spot", paint over the problem on the vehicle, describe what should change, and send it. Only the marked area is redrawn.

► Checks (25)

Save as

sals_kewl_livery.png

...

☒ PNG texture

☒ Layered PSD

Export

Open folder

View on 3D model

Run again

In OpenIV, replace the livery texture in both <model>.ytd and <model>_hi.ytd, keeping the texture name, then restart the resource and spawn a fresh vehicle.

The finished livery, with what it cost and every check the app ran.

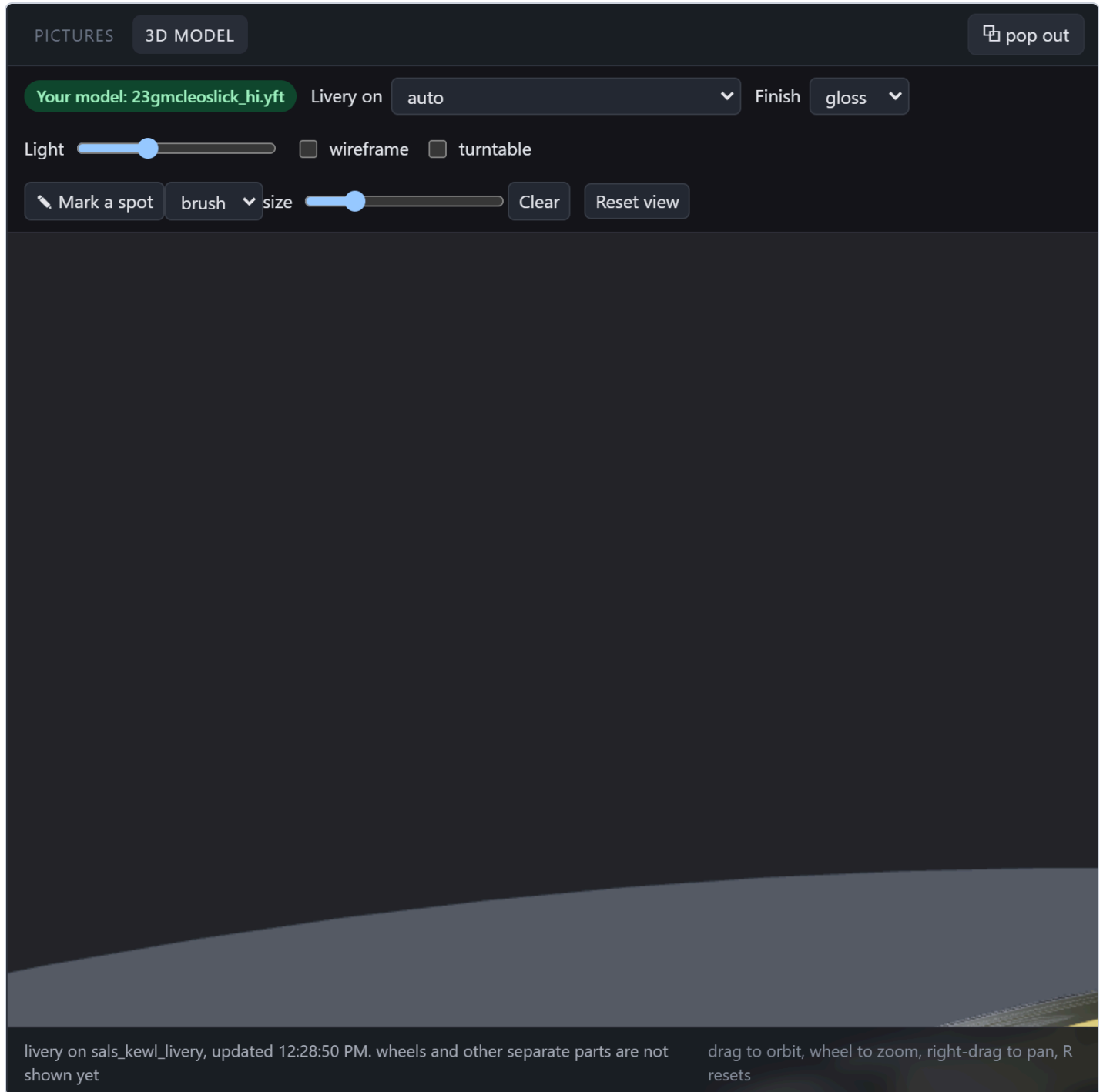
- **Save as** — where to write the files. The PNG name you give is also used for the PSD.

- **PNG texture** — the game texture, at the template's exact size. This is the file you put into the vehicle.
- **Layered PSD** — the same livery with every piece on its own layer for Photoshop. At 4096 this file is large and takes about a minute to write.
- **Checks** — expand it to see everything the app measured and corrected: which panels were off their outline and by how much, which were left plain, which side was mirrored.
- **Open folder, View on 3D model, Run again.**

Putting it in the game

In OpenIV, replace the livery texture inside both `<model>.ytd` and `<model>_hi.ytd`, keeping the original texture name. Then restart the resource and spawn a fresh vehicle.

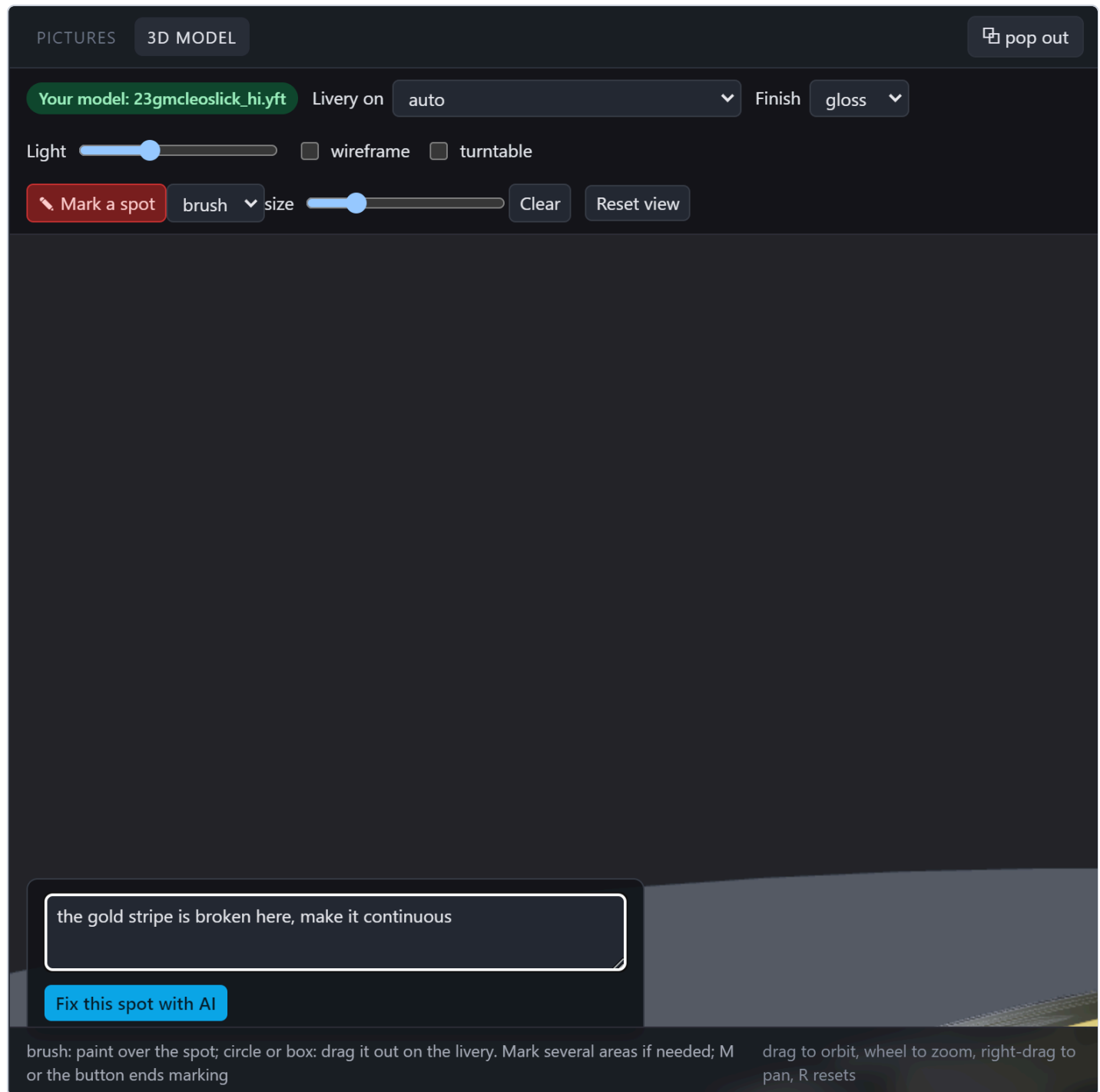
9. Checking it on the 3D model and fixing a spot



The finished livery on the real body. Drag to orbit, scroll to zoom, right-drag to pan.

This is the fastest way to spot problems: a stripe that does not line up across a door, lettering in the wrong place, a panel that got decorated when it should be plain.

Mark a spot



Marked areas in red, with the instruction box underneath.

1. Press **Mark a spot**.
2. Choose **brush**, **circle** or **box** and mark the area on the vehicle. Mark as many areas as you like for one fix.
3. Type what should change, for example "the gold stripe is broken here, make it continuous".
4. Press **Fix this spot with AI**.

Only the marked area is redrawn; everything else is left untouched. The fix arrives as its own layer, and **Undo last fix** on the Result step removes it if you do not like it. One fix is one image call, about 5 to 15 cents.

10. What a run costs

You pay OpenAI directly, per image generated. The app counts its calls and shows an estimate on the Result step.

Mode	What it does	Image calls	Rough cost
Final	Medium quality throughout, eight panels redrawn sharp, up to four corrections. Use it for the livery you are going to ship.	about 14	\$1.00 – \$1.50
Economy	Views drawn at low quality (they are only references), smaller reference images, six panels redrawn, up to two corrections.	about 11	\$0.60 – \$0.90
Preview	Everything at low quality, no sharp redraws, no corrections. A quick look at the design on the 3D model before committing.	about 4	\$0.15 – \$0.25
One spot fix		1	\$0.05 – \$0.15

A good way to work: run **Preview** first, look at it on the 3D model, adjust the wording in Changes & stipulations, preview again if needed, then run **Final** once. Two previews and a final cost about the same as two finals, and you are far more likely to be happy with the result.

Prices are estimates from OpenAI's published rates and can change. Your real spend is always on your OpenAI billing page.

11. Getting a better result

- **Name the vehicle properly.** "2023 GMC Sierra Crew Cab Pickup" works far better than "truck".
- **Type the lettering.** It is the single biggest thing you can do for spelling.
- **Untick panels that should stay plain** before you run, rather than fixing them afterwards.
- **Say what to ignore.** If the picture is a promo sheet with several vehicles, say which one: "use only the pickup in the middle row".
- **Leave Symmetric livery ticked.** It is the difference between two sides that match and two sides that nearly match.
- **More pictures help** when the design wraps onto the back or the hood.
- **Fix small things with Mark a spot** instead of re-running: it is a tenth of the cost.

12. If something goes wrong

What you see	What to do
"You have no credits remaining"	Add credit on platform.openai.com, Settings, Billing. Then press Start again; nothing is lost.
"The model ... does not exist"	Your account cannot use that model name. Press "list this key's models" on Step 1 and pick one from the list.
Your organization must be verified	Complete OpenAI's verification on their settings pages, then try again.
"No panels found on this image"	That PNG is probably not the livery template. It should be the texture sheet with panel outlines, not a picture of a vehicle.
The model will not load	Pick the <code>.yft</code> from the vehicle's <code>stream</code> folder. Escrow-protected vehicles (files that start with FXAP) cannot be read by any tool outside FiveM.
Export seems to hang	The 4096 PSD takes about a minute. The footer shows what it is writing; the files appear when it finishes.
A panel came out wrong	Use Mark a spot on the 3D model. If a whole panel should be plain, untick it and run again.
Something else	The Run screen's log says exactly what happened at each step. Copy it and send it over.

Where things are kept

- Your key and settings: `C:\Users\<you>\.sals_kewl_livery_wizard.json`
- Every run, with all its pictures: `C:\Users\<you>\.sals_kewl_wizard\runs\`
- Exports: wherever you pointed "Save as".