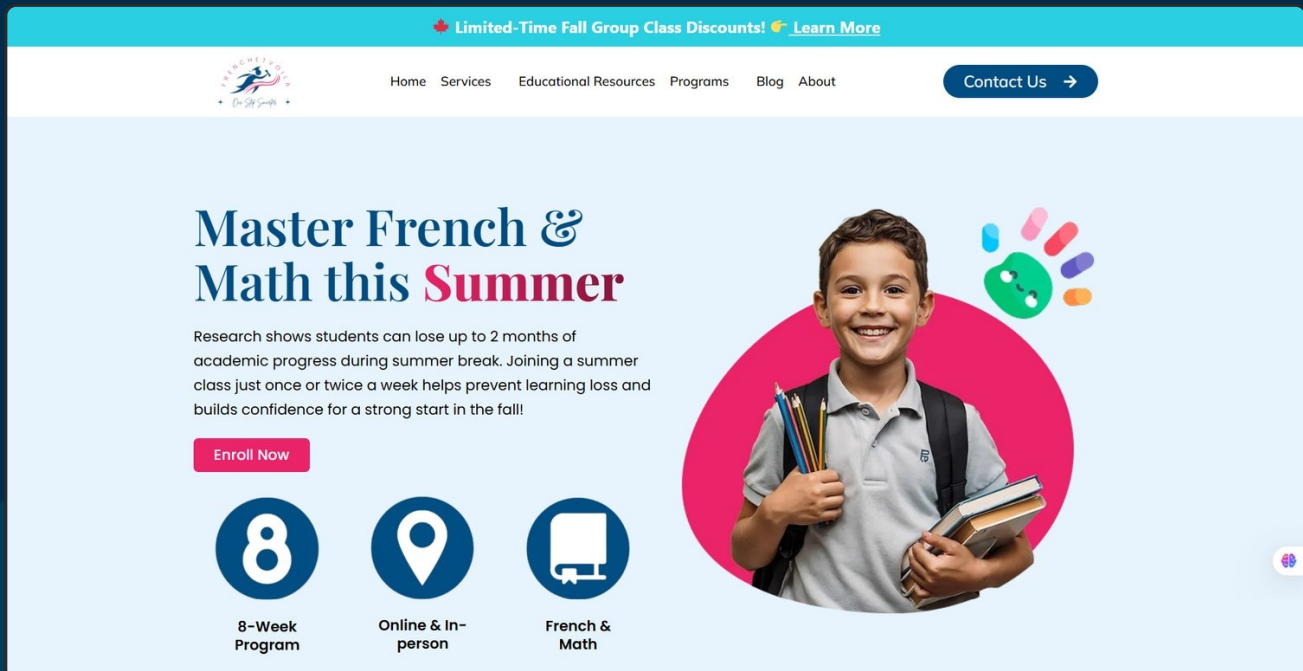


A Course Enrolment Engine for FrenchEtVoila

A custom WordPress plugin and two program experiences — a filterable course listing, dynamic course detail pages, and a nine-step enrolment wizard that prices itself from the course record.



CLIENT
FrenchEtVoila

SECTOR
Education / Tutoring

SCOPE
Plugin + 2 program pages

BUILT BY
Rishfotech Solutions

AT A GLANCE

A booking problem, solved as a reusable product.

FrenchEtVoila is a Canadian tutoring provider running French, Math, Science and Social Studies classes online across Canada and in person in Leduc, Alberta. Their programs run in seasonal intakes — a summer program, a fall program — each with dozens of course variants split by subject, grade band, format, schedule and price.

Every intake meant rebuilding the same thing: a page listing every course, a detail page per course, and some way for a parent to say *this one, for my child, at this level* and pay for it. Done by hand that is weeks of work per season, and it goes stale the moment a price or a schedule changes.

SCOPE OF THIS CASE STUDY

Rishfotech did not build the FrenchEtVoila website. This project covers a defined slice: **two program pages** (8-Week Summer 2026 and Fall 2026), **the course detail pages** behind them, and **the custom enrolment plugin** that powers the course listing, the selection flow and the payment handoff on both. Everything shown in this document is that work.

9

STEP ENROLMENT WIZARD

2

PROGRAM EXPERIENCES

6

LISTING FILTER CATEGORIES

1

PLUGIN DRIVING ALL OF IT

What was built

Course listing	Tab-filtered course grid, rendered from course records rather than hand-placed on the page
Course detail pages	A single template driving every course, with an overview panel, pricing panel and enrolment actions
Enrolment wizard	Nine-step modal covering subject, grade, format, class type, course, duration, language, contact details, review and plan
Pricing engine	Session count × per-session rate plus registration fee, computed live and shown before commitment
Payment handling	Plan selection, payment redirect, and a manual e-transfer confirmation path with its own acknowledgement state
Admin layer	Course fields, enrolment records in a dedicated table, and editable confirmation email templates with merge tags

Platform

CMS

WordPress

Plugin

Custom – PHP, ACF, AJAX

Front-end

Vanilla JavaScript wizard, no framework dependency

Payment

PayPal and Interac e-transfer

LIVE AT

/8-week-summer-program-2026/

/fall-program-2026/

/fall-course/...

INSIDE THIS DOCUMENT

Contents

01	The Brief	04
	Seasonal intakes, dozens of course variants, one enrolment path	
02	The Course Listing System	05
	Tab filtering, card anatomy, content-driven rendering	
03	Course Detail Pages	07
	One template, an overview panel and two entry points	
04	The Enrolment Wizard	08
	Nine steps, and why it is nine	
05	Steps 1–4: Narrowing	09
	Subject, grade, format, class type	
06	Steps 5–7: Choosing	10
	Course, duration and pricing, session language	
07	Steps 8–9: Confirming & Paying	11
	Contact capture, review, plan selection	
08	Context-Aware Entry	12
	The same modal, opened from two places, behaving differently	
09	Pricing & Payment Handling	13
	How the total is derived and how payment is settled	
10	Data Model & Administration	14
	Course fields, enrolment records, email templates	
11	Conversion Design on the Program Pages	15
	Booking path, objection handling	
12	Challenges We Faced — How We Solved Them	16
	Five real problems, worked end to end	
13	Project Outcome	19
	What the plugin now does for the business	

A NOTE ON EVIDENCE

Every screenshot here is captured from the live production site. No performance figures, enrolment numbers or revenue results are claimed, because none have been measured for this project. Where an implementation detail sits behind the administration login, it is marked **Development context**.

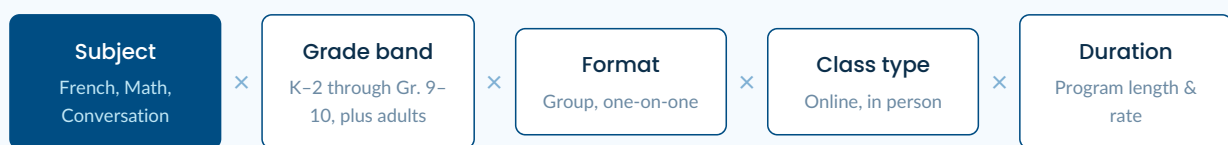
SECTION 01

Dozens of course variants, one enrolment path.

The client sells the same thing many times over, sliced five different ways. The build had to make that legible to a parent in under a minute.

A single seasonal program at FrenchEtVoila fans out into a large matrix. The Fall 2026 intake alone carries courses across two subjects, five grade bands, two delivery formats, several weekday schedules and a per-session price that varies by level.

THE VARIANT MATRIX BEHIND ONE PROGRAM



What a parent actually has to do

A parent arriving on the program page has one question — *which of these is right for my child, and what will it cost?* Answering it means resolving all five dimensions above. Presented as a flat list, that is a wall of near-identical rows. Presented as a form, it is an interrogation.

01

Findable

The right course has to surface in a couple of clicks, without the parent knowing the internal taxonomy.

02

Priced honestly

The total — including the registration fee — must appear before anyone is asked to commit.

03

Editable

Next season's intake has to be a content update, not a rebuild.

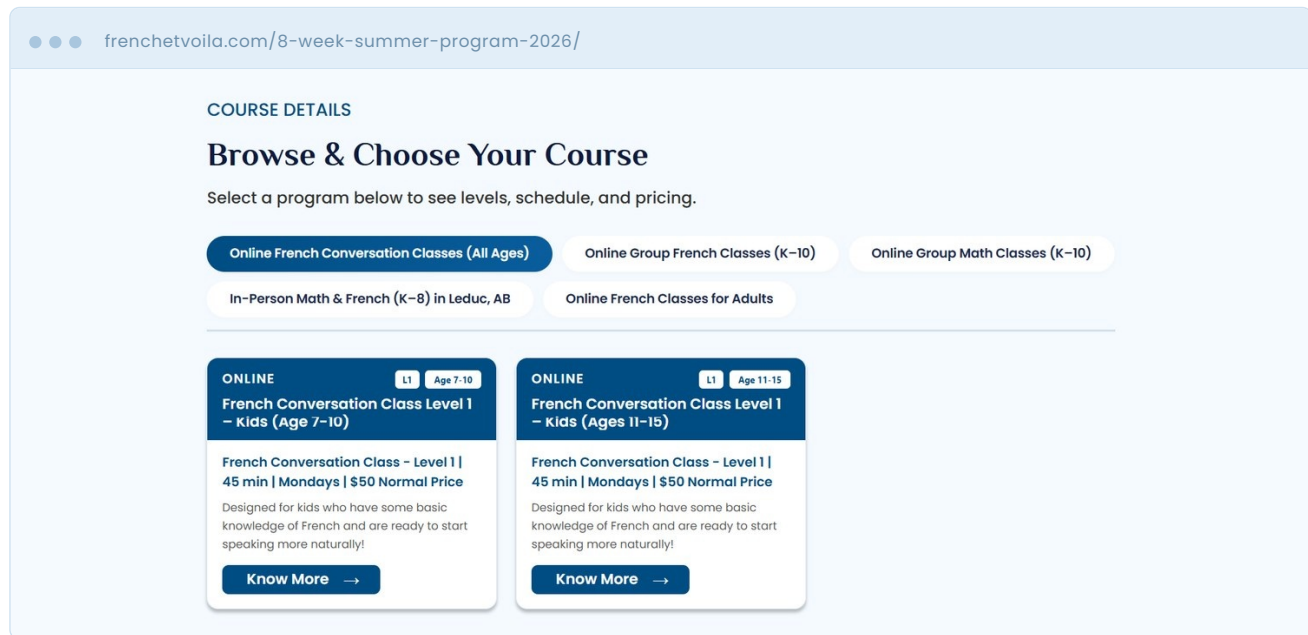
THE DESIGN DECISION

Rather than a long form, the flow was built as a **progressive narrowing wizard**. Each step asks one question and uses the answer to shrink the option set for the next. By step five the parent is choosing between three courses instead of thirty — and they never saw the other twenty-seven.

SECTION 02

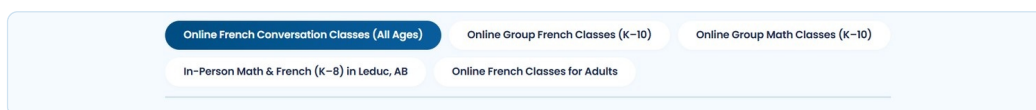
Browse and choose, without the scroll.

The listing is generated from course records, filtered by a tab row that maps to how the client actually sells — by delivery mode and audience, not by internal category.



Browse & Choose Your Course. Six category tabs above a card grid. Switching tabs swaps the visible set without a page load, so a parent can compare categories in seconds.

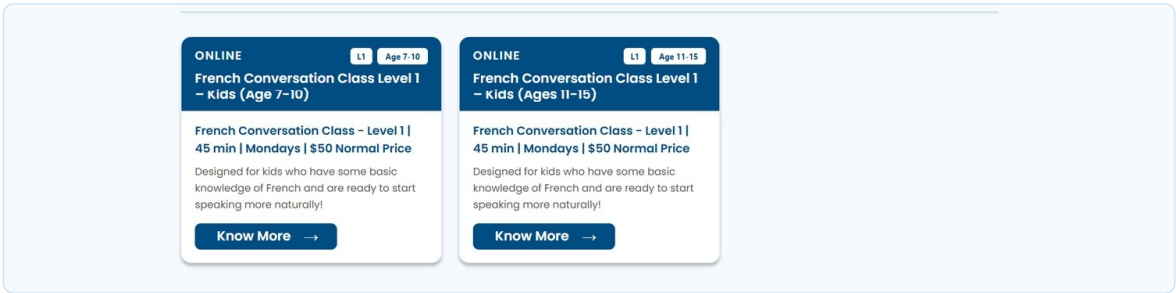
The tab labels are the important detail. They are not subject names — they are **audience and delivery combinations**: Online French Conversation (All Ages), Online Group French (K-10), Online Group Math (K-10), In-Person Math & French (K-8) in Leduc, Online French for Adults. A parent recognises their own situation in one of those phrases immediately. Filtering by "French" or "Math" would have made them do the translation themselves.



The filter row. The active tab is filled; the rest sit as outlined pills. On a set this small, showing all options at once beats a dropdown — the parent can see the full range of what is on offer without interacting at all.

SECTION 02 · CONTINUED

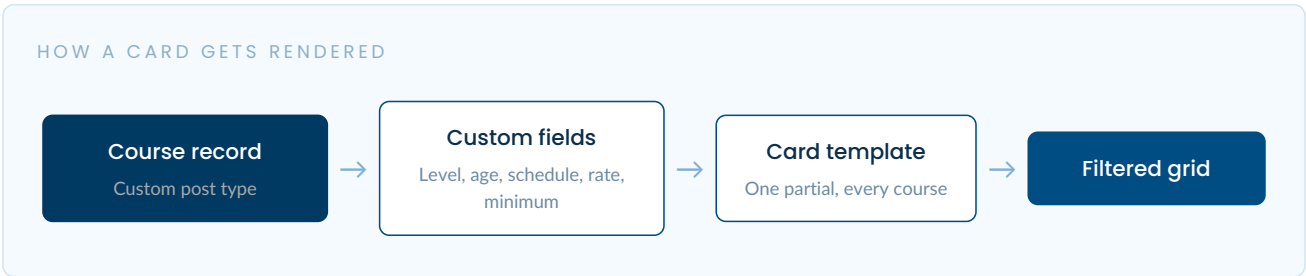
Card anatomy.



Two course cards. Every field on the card is drawn from the course record — nothing is typed into the page.

Each card has to answer five questions at a glance without becoming a paragraph. The solution was a fixed information hierarchy, applied identically to every course.

SLOT	EXAMPLE	WHY IT EARNS ITS PLACE
Mode badge	ONLINE · IN-PERSON	The first filter in a parent's head. Shown as a badge so it reads before the title.
Level & age chips	L1 · Age 7–10	Compact right-aligned chips — scannable down a column when comparing cards.
Course title	French Conversation Class Level 1	The name the client uses in their own materials, so the site and the brochure agree.
Meta line	45 min Mondays \$50 normal price	Duration, schedule and rate on one pipe-separated line — the three facts that decide fit.
Excerpt	Two lines, truncated	Enough to convey tone. Truncation is deliberate: uneven excerpt lengths would break the grid.
Know More	Persistent button	Always visible rather than hover-revealed, because a large share of this audience is on mobile.

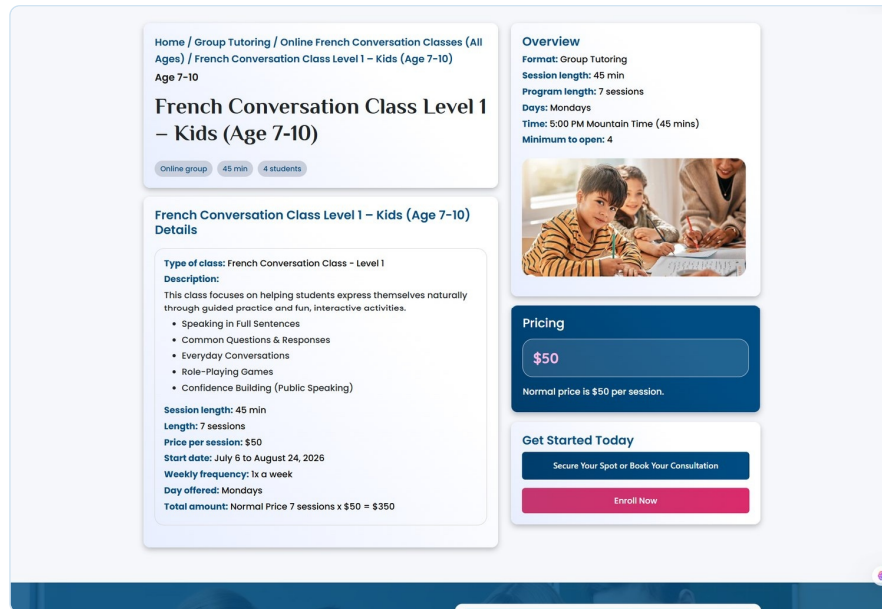


Because the grid is generated, adding a course to next season's intake means creating one record. It appears in the listing, in the correct tab, in the detail template and in the wizard's option set — with no template work at any point.

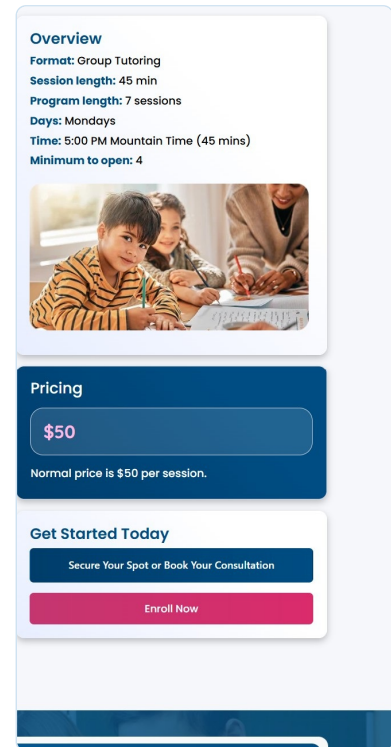
SECTION 03

One template, every course.

The detail page is where a parent decides. It carries the full curriculum, the exact schedule, the price maths, and two different ways to act on it.



Course detail page. Breadcrumb, title, attribute chips, a full details panel, and a right rail carrying overview, pricing and actions.



The right rail. Overview, pricing, then the two actions.

Why two buttons, not one

The rail ends with **Secure Your Spot or Book Your Consultation** in navy, and **Enroll Now** in the brand pink. That is not decoration — it is two genuinely different intents. A parent who already knows the level goes straight to enrolment. A parent who is unsure which level their child belongs in needs a consultation first, and forcing that person into a payment flow would lose them. Both paths are offered at the same moment, with the higher-intent one weighted by colour.

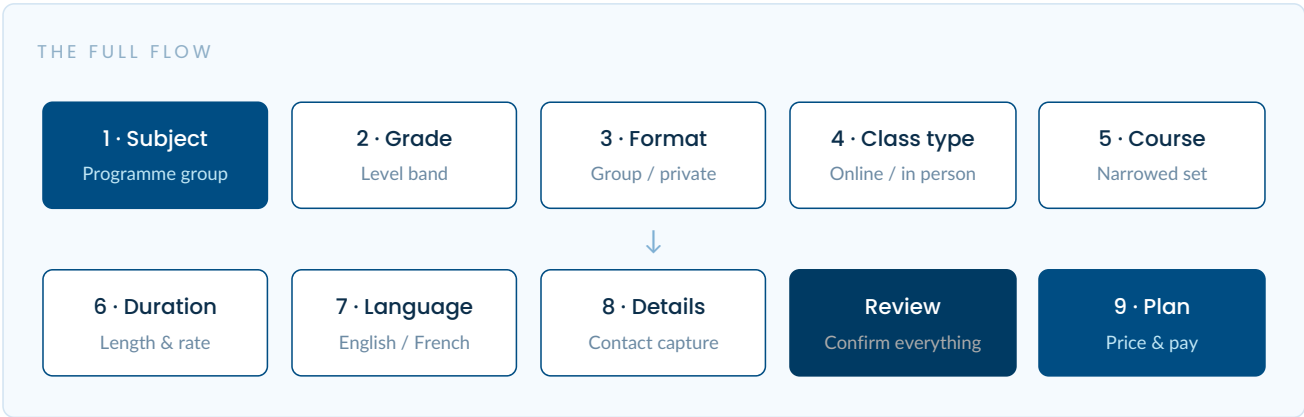
THE DETAIL THAT MAKES THE PAGE TRUSTWORTHY

The details panel spells out the arithmetic in plain language — session length, number of sessions, price per session, and the total written out as $7 \text{ sessions} \times \$50 = \$350$. Nothing is hidden until checkout. In a market where parents are comparing tutoring providers on price, showing the full sum on the detail page removes the single biggest reason to abandon.

SECTION 04

Nine steps — and why it is nine.

Nine screens sounds like a lot until you count the alternative: a single form with nine required fields, most of which a parent cannot answer without reading the whole catalogue first.



The rules the wizard follows

One question per screen	Never two. A parent on a phone answers a single question far more reliably than they scan a grouped form.
Answers narrow what follows	Every choice filters the next step's option set. Steps 2, 3 and 4 frequently render a single option, because the earlier answers already determined it.
Contact details come late	Name, email and phone are asked at step 8 — after the parent has invested seven decisions. Asking first would cost completions from people still browsing.
Money comes last	Price is shown at step 6 per session and totalled at step 9. Nothing is charged before the parent has seen the full sum broken down.
Always reversible	A Back control on every screen and a numbered progress bar that shows exactly how far there is left to go.

THE SINGLE-OPTION CASE

When a parent's earlier answers leave only one valid choice, the step still renders — showing that one option. It looks redundant, but it is doing real work: it confirms to the parent what has been decided on their behalf, and it keeps the step numbering stable so the progress bar never lies.

SECTION 05

Narrowing: subject to class type.

Step 1 — Select Subject. The same five groupings as the listing tabs, so a parent who browsed first sees vocabulary they already recognise.

Step 2 — Select Grade. Here the earlier answer has narrowed it to a single band. The step still renders, confirming the inference.

Step 3 — Learning Format. Group or private tutoring.

Step 4 — Class Type. Online or in person, which for this client also determines whether the Leduc location is relevant.

Four screens, four decisions, and the option set has already collapsed from the full catalogue to a handful. The progress bar at the top of every screen carries all nine numbers with the current one highlighted — a small thing that materially reduces abandonment, because the parent can see the flow is finite.

SECTION 06

Choosing: course, duration, language.

Course Enrollment

1 2 3 4 5 6 7 8 9

Select Course
Available courses matching your selection.

French Conversation Class Level 2 – Adults

French Conversation Class Level 1 – Adults

French Beginner Class Level 1 – Adults

← Back Continue →

Step 5 – Select Course. Three options, not thirty. This is the payoff from the four narrowing steps: the parent chooses between courses that are all genuinely valid for their child.

Course Enrollment

1 2 3 4 5 6 7 8 9

Program Duration
How long would you like to study?

8 Weeks
\$50/session

← Back Continue →

Step 6 – Program Duration. Length and per-session rate together.

Course Enrollment

1 2 3 4 5 6 7 8 9

Session Language
Would you like sessions in French or English?

English

French

← Back Continue →

Step 7 – Session Language. English or French delivery.

Step 6 is where price first appears, and the pairing is deliberate. Showing *8 Weeks* and *\$50/session* on the same card means the parent is never choosing a duration blind and then being surprised by what it costs. The available durations come from the course record, so a course offered over seven weeks and one offered over twenty-four both render correctly from the same step.

Step 7 exists because this is a bilingual provider operating in Canada. Whether the class is *taught in* French or English is a distinct question from whether the subject is French — an adult beginner may want French classes delivered in English. Collapsing the two would have produced mis-placed students and refund requests.

SECTION 07

Confirming, then paying.

Step 8 — Your Information. Three fields only.

Review. Every answer, with an explicit confirmation checkbox.

Step 9 — Choose Your Plan. The full sum, itemised.

The review screen is the most important part of the whole flow, and it was the last thing added. It lists all nine answers back — mobile number, course, subject, grade, format, class type, duration, session count, language — in a scrollable table, gated behind a checkbox reading *I confirm that the above details are correct*.

Why the review exists	Nine sequential decisions are easy to lose track of. Without a summary, a parent reaching payment has to trust their own memory of what they picked seven screens ago.
Why it is a checkbox	An explicit affirmation, not a passive Continue. It converts the summary from something skimmable into something acknowledged — which matters when the next action takes money.
Why it names the button differently	The control reads <i>Continue to Pricing</i> , not Continue. The parent knows exactly what the next screen contains before they reach it.

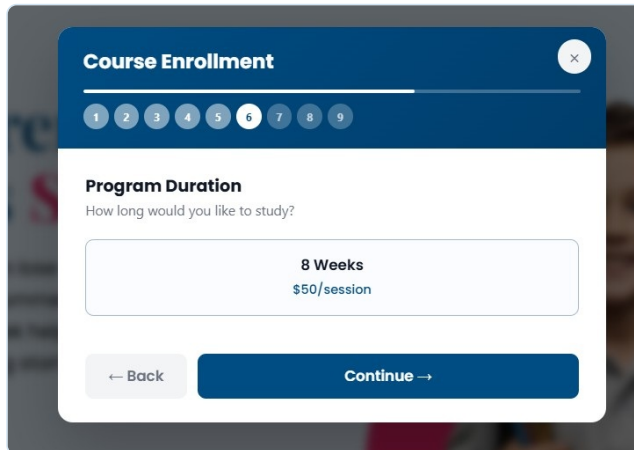
STEP 9, IN FULL

The plan screen restates the course, duration, session count and language; flags the registration fee separately in an amber notice; then shows the total with the arithmetic printed underneath — *rate × sessions + registration fee = total*. A parent can verify the number themselves before clicking. Nothing about the price is inferred, rounded or hidden.

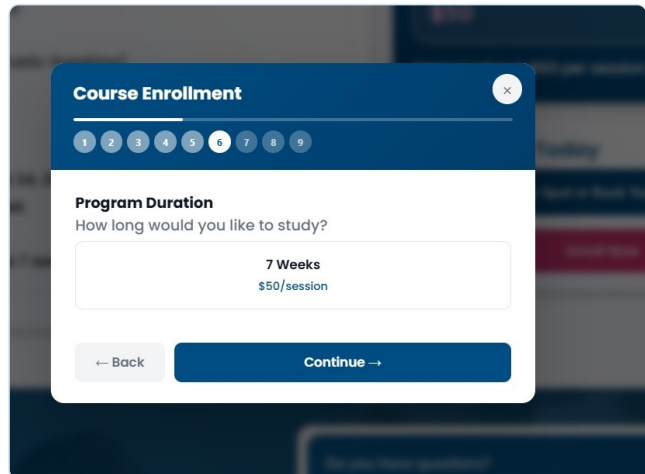
SECTION 08

One modal, two entry points, two behaviours.

The same wizard is launched from the program page and from a course detail page. Opened from a detail page it already knows which course you are looking at — and its options change accordingly.

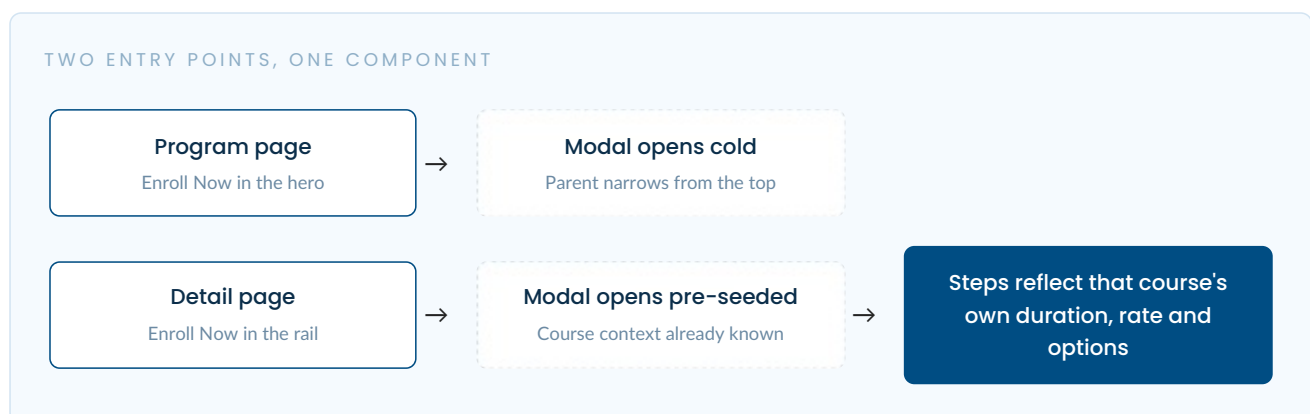


Step 6, opened from the program page. Offers **8 Weeks** at \$50 per session — the summer programme's length.



Step 6, opened from a course detail page. Offers **7 Weeks** at \$50 per session — that specific course's length.

Those two screenshots are the same step in the same modal. The difference is where it was opened from, and it is the clearest demonstration of what the plugin actually does.



Building this as two separate flows would have been the obvious route and a long-term mistake — two code paths, two sets of validation, two places for a pricing bug to hide. Instead the wizard reads its starting context on launch and seeds its state from it. One component, one set of rules, two experiences.

DEVELOPMENT CONTEXT

The modal is driven by a single JavaScript controller that maintains a step state object and requests the valid option set for each step over AJAX. Course data is fetched from the course records rather than duplicated into the script, which is why a content edit in the admin changes the wizard immediately.

SECTION 09

How the total is derived, and how it gets settled.

Prices are never typed into a page. Every figure the parent sees is computed from the course record at the moment it is shown.

THE PRICING CHAIN

Course record
Per-session rate

×

Session count
From the chosen
duration

+

Registration fee
Flagged separately

=

Total
Shown with its own
arithmetic

The registration fee is deliberately not folded into the headline number. It appears in its own amber notice above the total, and again inside the breakdown line. Burying a mandatory fee inside a total is the fastest way to generate a support email, and in a parent-facing market it costs trust that is hard to recover.

Settling the payment

This client takes payment two ways, and the flow supports both without pretending they are the same thing.

Card & PayPal

The plan action hands off to the payment provider, and the wizard shows a processing state while the redirect happens — so the parent is never looking at a frozen screen wondering whether it worked.

Interac e-transfer

A distinct path with its own control — *I have made the payment via e-transfer*. The enrolment is recorded immediately as pending, and the parent gets an acknowledgement explaining that confirmation follows once the transfer clears.

WHY THE E-TRANSFER PATH NEEDED ITS OWN DESIGN

E-transfer is manual and asynchronous — the money arrives minutes or hours later, out of band. If the flow had treated it as a normal payment, the parent would have seen a success screen for money that had not moved. Instead the confirmation state says plainly that the enrolment is recorded and will be finalised once the transfer is confirmed, and it tells the parent to check their spam folder for the confirmation email. Honest state beats optimistic state.

SECTION 10

Built so the client runs it.

The whole point of building this as a plugin rather than a set of pages was that next season's intake should not need a developer.

WHAT THE PLUGIN OWNS

Course records
Custom post type, one per course

Course fields
Level, age band, schedule, rate, sessions, minimum to open

Enrolment table
Dedicated database table, not post meta

Email templates
Admin and parent, with merge tags

Three decisions worth explaining

A dedicated table, not post meta	Enrolments are transactional records that get queried, filtered and exported. Storing them as WordPress posts would have made the admin list slow and the data awkward to work with as it grows season over season.
Schema migrations built in	The plugin runs a database upgrade routine that adds new columns to existing installations, alongside the create-table path used on fresh ones. This is why fields could be added to the enrolment record later without breaking the live site.
Editable email templates	Both the admin notification and the parent confirmation are editable in the admin with a published list of merge tags — course, schedule, session count, amounts. The client can reword their own confirmation email without touching code.

What lands in the confirmation email

The parent's confirmation restates the booking in full: format, session length, program length, days, time, minimum to open, the registration fee, and the total. That last one is computed rather than stored as a literal, so the email can never disagree with what the wizard showed.

DEVELOPMENT CONTEXT

Enrolment persistence, sanitisation and email dispatch are server-side and sit behind the administration login, so they are described here rather than shown. Transactional email delivery was later moved onto a dedicated sending service, under an account owned by the client rather than by Rishfotech.

SECTION 11

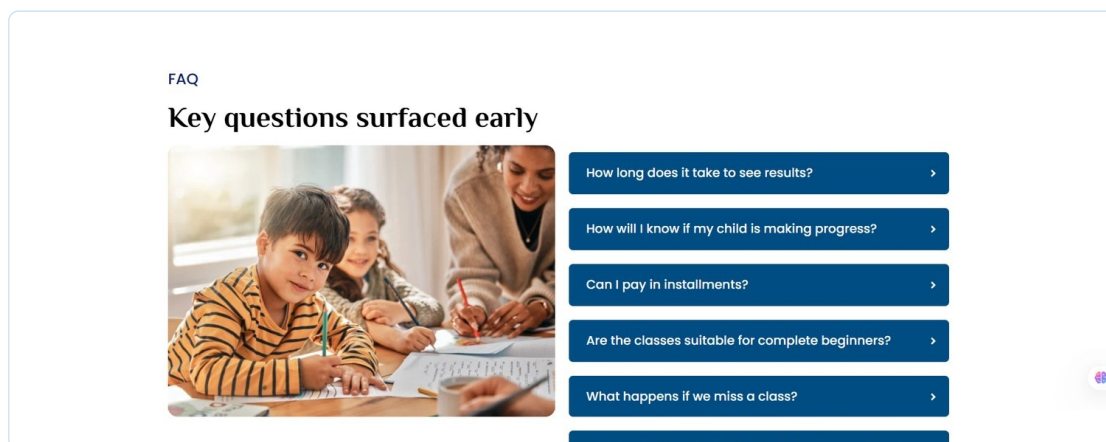
The page around the wizard.

A booking flow only converts if the parent trusts the provider before they open it. Both program pages are sequenced to do that work first.



How Parents Book The Right Course Path. A four-step explainer placed directly below the listing – choose a program, match the level card, enrol and pay, start learning.

This section exists because of a specific observation about the audience. A parent who is unsure whether they are doing it right will hesitate rather than guess. Spelling out the four steps in advance removes that hesitation, and it does so *after* the listing, when the parent has already seen enough to be interested but may not yet know how to proceed.



Key questions surfaced early. An accordion answering the six objections that would otherwise become emails – results timeline, progress visibility, instalments, beginner suitability, missed classes, and enrolling in more than one subject.

OBJECTION HANDLING AS CONVERSION WORK

Two of those six questions – can I pay in instalments, and are the classes suitable for complete beginners – are the ones most likely to stop a parent mid-enrolment. Answering them on the page, before the wizard opens, means the parent reaches the payment step with the objection already resolved rather than abandoning to go and ask.

SECTION 12

Challenges we faced — how we solved them.

Five real problems from this build, each taken from the symptom through to the outcome.

CHALLENGE 01

A catalogue too wide to present as a list

PROBLEM	One seasonal intake fans out across subject, grade band, format, class type and duration. Presented flat, that is dozens of near-identical rows a parent has to read in full before they can rule anything out.
ANALYSIS	We looked at how a parent actually describes what they want, and it is never by internal category. It is “online French, group, for my Grade 4” — a combination. Any interface organised by a single dimension forces them to translate.
IMPLEMENTATION	Two complementary surfaces built from the same course records: a tab-filtered listing labelled by audience-and-delivery combinations, and a progressive wizard that narrows the set one question at a time.
SOLUTION	Browsers get the grid; decided parents get the wizard. By step five the wizard is offering three courses instead of thirty, and the parent never saw the rest.
OUTCOME	The catalogue can grow without the interface degrading. A wider intake next season means more records, not a harder page to read.

CHALLENGE 02

The same modal, opened from two very different places

PROBLEM	Enrolment starts either from the program page, where the parent has chosen nothing, or from a course detail page, where they have effectively chosen everything. The same nine steps cannot serve both naively.
ANALYSIS	Building two flows was the obvious route and the wrong one — two code paths, two sets of validation, and two places for a pricing bug to hide. The real difference is not the flow, it is the starting state.
IMPLEMENTATION	A single modal controller that reads its launch context and seeds its state from it, then requests the valid option set for each step from the course records rather than from anything hard-coded in the script.
SOLUTION	Opened from the program page the wizard starts cold. Opened from a detail page it already knows the course — and the duration step offers that course's own length and rate rather than the programme default.
OUTCOME	One component, one set of rules, two experiences. A change to validation or pricing lands on both entry points at once, because there is only one of them.

SECTION 12 • CONTINUED

Challenges 03 and 04.

CHALLENGE 03

Two payment rails that behave nothing alike

PROBLEM	The client takes payment by card and PayPal, and by Interac e-transfer. One completes in the browser in seconds; the other happens out of band, minutes or hours later, with no callback to the site.
ANALYSIS	Treating both as a single “payment” would have shown a success screen for money that had not moved. That is not a cosmetic problem — it produces parents who believe they are enrolled when the client has no record of payment.
IMPLEMENTATION	The two rails were given separate paths and separate end states. The redirect path shows a processing state while the handoff happens. The e-transfer path has its own explicit control and records the enrolment as pending.
SOLUTION	The e-transfer confirmation states plainly that the enrolment is recorded and will be finalised once the transfer is confirmed — and tells the parent to check their spam folder for the confirmation email.
OUTCOME	The parent's understanding matches the client's records at every point. Honest state instead of optimistic state, at the cost of one extra screen.

CHALLENGE 04

Nine decisions, and no way to check them

PROBLEM	A parent reaching the payment step had made nine sequential choices with nothing to verify them against. The first thing they would see after committing was an invoice for a course they could not remember selecting.
ANALYSIS	The risk is not that people choose wrongly — it is that they cannot tell whether they chose wrongly, so they stop rather than risk it. Uncertainty immediately before a payment is where flows lose people.
IMPLEMENTATION	A review screen inserted before pricing, listing all nine answers in a scrollable table and gated behind an explicit “I confirm that the above details are correct” checkbox. The forward control was renamed <i>Continue to Pricing</i> .
SOLUTION	The parent verifies everything in one place, affirms it deliberately, and knows exactly what the next screen contains before reaching it.
OUTCOME	The step before payment became a confidence step rather than a leap. It also gives the client a clean audit trail: every enrolment was explicitly confirmed by the parent.

SECTION 12 • CONTINUED

Challenge 05.

CHALLENGE 05

A record structure that had to change after launch

PROBLEM	After the plugin was live and taking enrolments, the client needed the confirmation emails to carry the full schedule — format, session length, program length, days, time and minimum to open — plus the registration fee and the combined total. None of that was in the enrolment record.
ANALYSIS	Adding fields to a table that is already holding live enrolments is where plugins usually break. A fresh install would get the new schema; existing installations would not, and the mismatch surfaces as a fatal error on the next submission.
IMPLEMENTATION	A database upgrade routine that runs alongside the create-table path, adding the new columns to existing installations. The new fields were threaded end to end — admin field definitions, the course data array, the submission payload, sanitisation, persistence, and the template tag resolver.
SOLUTION	The total is computed at send time from the course amount plus the registration fee rather than stored as a literal, so the email can never disagree with what the parent saw in the wizard.
OUTCOME	A live schema change shipped without downtime or data loss, and the plugin gained a migration path it can reuse for every future field.

THE PATTERN BEHIND ALL FIVE

Every one of these came down to the same instinct: put the complexity in the data model, not in the interface. A wide catalogue, two entry points, two payment rails and a changing record structure are all manageable if the pages are generated from records — and all become rebuild-every-season problems if they are not.

SECTION 13

What the plugin now does for the business.

FrenchEtVoila can open a new seasonal intake by adding course records — and the listing, the detail pages, the wizard, the pricing and the confirmation emails all follow.

BEFORE	AFTER
Each intake meant building a new program page and a set of course pages by hand.	Course records drive everything — listing, detail template, wizard options and emails all read from the same source.
Prices written into page copy, drifting out of sync between page, brochure and invoice.	Every figure is computed from the course record at the moment it is displayed.
Enquiries arriving as free-text messages needing manual back-and-forth to pin down.	Structured enrolments with subject, level, format, schedule, language and contact details already resolved.
No distinction between an online payment and a pending e-transfer.	Two payment paths with separate states, so records and parent expectations agree.
Confirmation emails needing developer edits.	Editable templates with published merge tags, maintained by the client.

The durable result

The plugin is not a page — it is a mechanism. The two program experiences shipped on it are the first two of many: each new season is a set of content entries against a system that already knows how to list, filter, describe, price and sell them.

IN SHORT

FrenchEtVoila sells the same thing many times over, sliced five different ways. The build turned that catalogue into something a parent can navigate in nine clicks — and something the client can extend without calling a developer.

Web development that holds up **after** launch.

We build custom WordPress, Shopify, Laravel and Next.js platforms for clients across Europe, Australia, the US and India — with a bias toward systems a client's own team can run without us.

PROJECT

frenchetvoila.com

STUDIO

www.rishfotech.com

EMAIL

sales@rishfotech.com

PHONE

+91 98747 00540

STUDIO ADDRESS

Ecospace Business Park, Newtown, Kolkata, India