

MASTER AI IN 30 DAYS

Curriculum and Learning Framework

Grades 5 to 12 · Academic Year 2026–27

Learn AI. Create with AI. Get future-ready.

A live, project-first programme in which every student understands how AI works, learns to direct it, tests where it fails, and finishes with one working project they built, published to a link, and can defend.

LIVE SESSIONS	MINUTES EACH	PER BATCH	LEARNING LEVELS	MODE
08	90	10	03	Live online

GRADES 5–6

AI EXPLORER

Explore & Create. Understand the machine, direct it clearly, train a first model and build a helper that someone actually uses. It ends as a link.

GRADES 7–9

AI INNOVATOR

Build & Solve. Ground research in sources, evaluate a model properly and ship a tested prototype against a real problem.

GRADES 10–12

AI ARCHITECT

Design, Analyse & Innovate. Architect, evaluate, automate and document an AI system that does one real job end to end.

Taivas Edutech Services and Consultants Pvt. Ltd. · AI Labs by Taivas · Issued for schools, parents and partners

Version 2.0 · Grade bands restructured 5–6 / 7–9 / 10–12 · 12 September 2026

The programme at a glance

Students already have access to AI. Very few can say what it actually did, why it was confidently wrong, or how to make it better. This programme closes that gap in eight sessions: understand the machine, direct it precisely, check it against evidence, create with it, train and test one of your own, build something useful, publish it, and explain every decision made.

SESSIONS, 90 MIN	PHASES, TWO EACH	LEVELS	PROJECT BUILT	PER BATCH
08	04	5-6 · 7-9 · 10-12	01	10

THE FOUR PHASES

PHASE	SESSIONS	WHAT THE STUDENT IS DOING	THE QUESTION THE PHASE ANSWERS
FOUNDATIONS	01 & 02	Building an accurate mental model of AI, then learning to direct it with structured, repeatable instructions.	What is this machine, and how do I get it to do what I actually meant?
CREATE	03 & 04	Using generative tools for design, image, video, voice and story, and editing the output instead of accepting it.	Can I make something good with it, and can I tell good from generated?
INVESTIGATE	05 & 06	Grounding work in sources, then training a model on their own data and measuring where it fails.	How do I know this is true, and what happens when I change the data?
BUILD & SHOW	07 & 08	Turning everything into one working build, publishing it, testing it on a real person, improving it and defending it live.	Does it work for someone other than me, on their device, and can I explain why?

THE THREE LEVELS

LEVEL	AGES, TYPICAL	PROGRESSION AND FOCUS
5-6 AI EXPLORER	10-12	Explore & Create. Correct vocabulary, curiosity and the habit of checking. Students create real assets, train a first classifier and publish a helper another person can use.
7-9 AI INNOVATOR	12-15	Build & Solve. Evidence, structure and measurement. Students research a problem from sources, evaluate a model properly and ship a prototype tested on real users.
10-12 AI ARCHITECT	15-18	Design, Analyse & Innovate. Systems, metrics and automation. Students architect a grounded system, score it against a written evaluation set and deploy it to a link.

The eight-session spine is identical across all three levels. What changes is conceptual depth, the tool set, the size of the build and the standard of proof demanded.

FIXED IN EVERY LEVEL, EVERY COHORT

01	Every session ends with an artefact that can be opened, shown, tested or assessed.
02	Every level trains or evaluates a model on data the students chose themselves.
03	Every level meets at least one AI failure deliberately, not by accident.
04	Every build is tested by a person who is not the student who built it.
05	Every student demonstrates and defends their own project live in session 08.
06	The tool may be replaced. The learning outcome may not.
07	Every project ends as a link that opens on a phone the student does not own. It is sent to the parent before the demo begins.

THE SENTENCE THIS CURRICULUM HAS TO EARN

"I understand what AI is doing, I know how to direct it, I built something real with it, I published it, I tested it, I know exactly where it fails, and I can explain every choice I made."

The eight sessions, across the three levels

The spine is deliberately constant, so a school running all three levels teaches one coherent programme rather than three unrelated courses. A Grade 5 student and a Grade 12 student cover the same territory in the same week. They do not do the same work.

#	PHASE	GRADES 5-6 · AI EXPLORER	GRADES 7-9 · AI INNOVATOR	GRADES 10-12 · AI ARCHITECT
01	Foundations	*Meet the Machine Rules, learning and generation	*Inside the Model Training, tokens and why models differ	*How Models Behave Parameters, context and reliability
02	Foundations	*Talk to AI Properly The five parts of a good prompt	*Prompting with Precision Structured outputs and templates	*Context Engineering & Reliability System prompts and evaluation sets
03	Create	*Design It With AI Hierarchy, message and editing	*Visual Communication & Campaign Design A visual system, not one poster	*Interface & Product Design Designing for a non-technical user
04	Create	*Pictures, Voice & Story Making media and disclosing it	*Multimodal Storytelling Consistency, consent and provenance	*Generative Media, Rights & Authenticity Licensing, disclosure and risk review
05	Investigate	*Study Smarter, Check Everything Learning with AI without copying	*Source-Grounded Research Evidence tracing and problem framing	*Research, Retrieval & Open Models Grounded assistants and model choice
06	Investigate	*Teach a Machine Data, labels and failure	*Data, Training & Evaluation Per-class accuracy and bias	*Data Analysis, Metrics & Experiments Baselines and controlled change
07	Build	*Build Your First AI Helper Input, AI step, output, user, ship check	*Build, Test & Improve Real users, a measured fix, ship check	*Architect & Automate Triggers, workflows, the human check, ship check
08	Show	*Show & Explain Send it home, three-minute demo	*Pitch, Demo & Defend Send it home, evidence under questioning	*Technical Demo & Viva Send it home, architecture and ethics

How the same topic deepens across the three levels

CAPABILITY	GRADES 5-6	GRADES 7-9	GRADES 10-12
HOW AI WORKS	Rules, learning and generation	Training data, tokens, context window	Embeddings, temperature, cost, latency

CAPABILITY	GRADES 5–6	GRADES 7–9	GRADES 10–12
PROMPTING	Role, task, context, constraints	Output schemas, few-shot, versioning	System prompts, refusals, adversarial cases
VERIFICATION	Check one fact against one source	Trace every claim to a source set	Build a system that refuses out of corpus
CREATIVITY	One asset, edited rather than accepted	A campaign set with one visual system	A product interface and media identity
MACHINE LEARNING	Train a classifier, log what it misses	Balance the data, accuracy per class	Select and evaluate an existing model
DATA LITERACY	Chart a set, check an AI reading of it	Clean, compare, disprove one AI claim	Baseline, one-variable experiment
BUILDING	A helper a classmate can use unaided	A prototype tested with three users	A deployed system with a human check
SHIPPING	A published page a family member can open	A prototype link tested by three strangers	A running system with a share link and a run log
AUTOMATION	A repeatable workflow, written down	Two tools connected so work repeats	A triggered automation, with run logs
RESPONSIBLE AI	Privacy, honesty, asking permission	Bias, consent, disclosure, attribution	Security, misuse, societal cost
EXPLAINING WORK	Three-minute demo with one failure	Five-minute pitch, defended	Technical demo and viva, unseen input

Tools have a job. They are not the course.

Tools are introduced only where they carry a skill the student does not yet have, and they are deliberately not repeated at every level. A Grade 7 student does not need Hugging Face. A Grade 12 student does not need to be taught Teachable Machine. Each level runs a tight, teachable stack, and the workflow is what transfers when the tool is replaced.

TOOL	WHAT IT TEACHES	5–6	7–9	10–12	PUBLISH SURFACE
ChatGPT	General assistance, critique, iteration	Core	Core	Support	—
Claude	Long documents, reasoning, code review	—	Compare	Core	—
Napkin AI	Turning writing into clear diagrams	Core	Core	Diagrams	Image export
Canva AI	Design judgement, editing AI output	Core	Core	Support	Share link / Canva website
AI image tools	Visual prompting, selection, revision	Core	Core	Support	—
AI video & voice	Multimodal story, consent, disclosure	Guided	Core	Support	—
Gamma	Structured presentation and pitch	—	Core	Core	Public page link
Teachable Machine	Data, labels, training, testing, bias	Core	Core	—	Model export link
Google Sheets	Collecting, charting and verifying data	Basic	Core	Core	View-only share
NotebookLM	Source-grounded research and tracing	—	Core	Core	Shared notebook
Perplexity	Cited search and cross-checking a claim	—	Verify	Core	—
Google Sites	Publishing a build others can use	Core	Core	Support	Public URL
Google AI Studio	Guided prototyping, system instructions	—	Core	Core	Trainer-hosted share link
Hugging Face	Model cards, demos, comparing models	—	—	Core	Public model card
Google Colab	Loading, cleaning and charting data	—	—	Core	View-only notebook
AI app builder	Turning a spec into a working front end	—	—	Core	One-click published URL

TOOL	WHAT IT TEACHES	5–6	7–9	10–12	PUBLISH SURFACE
Automation (Apps Script on Sheets)	Triggers, actions, human approval	—	—	Core	Run-log tab, view-only

Core: worked in directly and assessed on the workflow. Guided / Compare / Verify / Basic: met once, under trainer direction, to make a specific point. Dash: not used at this level. Zapier, Make and n8n are demonstrated at 10–12 and never built on, because they sit outside the Taivas tenant.

HOW A TOOL EARNS A PLACE ON THIS LIST

TEST	STANDARD APPLIED
IT CARRIES A SKILL	Included only where it teaches something the student cannot already do. A second tool teaching the same skill is removed, not added.
TEACHABLE IN NINETY MINUTES	Learnable inside one session and usable unaided in the next. Anything needing installation, a card or a long setup is out.
ITS OUTPUT CAN BE MARKED	The student's work in the tool must be inspectable, testable and markable, not merely presentable.
IT SHIPS WITHOUT AN ACCOUNT THE STUDENT MAY NOT HOLD	A tool whose terms exclude the student's age is used only trainer-hosted, and never for between-session work.

The code question, settled

Schools and parents ask whether this requires programming. It does not. This is the whole of the code that exists in the programme, and the answer is the same at every level: AI writes the code, the student reads it, and deletes what they cannot explain.

LEVEL	CODE THE STUDENT MEETS	WHAT THEY DO WITH IT	WHAT THEY ARE NOT ASKED TO DO
5–6	None	—	Write, read or see code
7–9	None	—	Write or read code
10–12	Generated app-builder code (S03) · AI-drafted Colab cells (S06) · AI-drafted Apps Script (S07)	Read it. Explain any line the trainer points at. Delete and regenerate what they cannot explain. Change one value and predict the effect.	Write from a blank file. Learn syntax as a topic. Use Git. Install anything.

THE SENTENCE FOR SCHOOLS, PARENTS AND THE TRAINER

Nobody is taught to code. Grade 10–12 students are taught to read the code AI writes for them, and to own every line they ship.

"Explain This Line" in Session 03 of the senior level is the entire programming assessment. If any project on any menu needs code beyond this, the project is wrong, not the rule.

LEVEL 01 · UPPER PRIMARY AND MIDDLE SCHOOL

GRADES 5-6 AI EXPLORER

Explore & Create

OBJECTIVE

Students at this level build an accurate mental model of what AI is and is not, learn to give it clear instructions, create real work with it, train a small model on examples they collect themselves, and finish by publishing one helper that another person can genuinely use. The emphasis is curiosity, correct vocabulary and the habit of checking.

THE SHIFT THIS LEVEL IS DESIGNED TO PRODUCE

From using AI as an answer machine to noticing what it just did. A student leaves this level able to say what they asked for, what the machine produced, why it may be wrong, what they changed, and where their helper lives.

PRIOR KNOWLEDGE ASSUMED

None. Students need basic typing, file handling and a browser. No coding is required and none is taught, at this level or any other. Typing speed, not comprehension, is the real constraint in Grade 5; sessions are paced for it and every prompt library is reusable rather than retyped.

COGNITIVE FOCUS

Understand, apply and create. Students are asked to explain in their own words, follow and adapt a workflow, and produce original work. They are not asked to evaluate systems formally, which begins at Grades 7–9.

TOOL SET AT THIS LEVEL

ChatGPT · Napkin AI · Canva AI · AI image tool · Teachable Machine · Google Sheets · Google Sites or Canva Sites · AI voice, guided

Deliberately excluded at this level: NotebookLM, Hugging Face, Google AI Studio, Gamma, Colab and automation. These arrive in later levels, once students have something worth grounding, evaluating and automating.

SAFEGUARDING AT THIS LEVEL, STATED PLAINLY

Every student in this band is under 13. That is not a caveat, it is the operating rule. No personal AI account is created on any tool, at any point, for any student.

All work happens on a Taivas-issued Google Workspace for Education account, on a Canva for Education class, or in a trainer-hosted session. No between-session task ever requires a login the student does not already hold from Taivas. No personal data, the student's own or anyone else's, is entered into any model.

Learning outcomes

By the end of this level, a student can:

- 01** Explain the difference between a rule-based program, machine learning and generative AI, with one example of each from their own life.
- 02** Describe in plain language how a generative model produces an answer, and give one reason it can be confidently wrong.

- 03 Write a prompt containing role, task, context, constraints and an example, and improve it across at least three deliberate iterations.
- 04 Spot an invented detail in an AI output and correct it against a supplied source.
- 05 Produce an original visual asset with an AI design tool and explain the design decisions they made rather than accepted.
- 06 Turn a piece of writing into a clear diagram or visual summary that a classmate can follow.
- 07 Create a short illustrated or narrated piece and attach an honest statement of how AI was used in it.
- 08 Collect a labelled dataset of their own, train a classifier and report the cases it gets wrong.
- 09 Explain how the examples they chose changed the model's behaviour, and demonstrate the effect by changing them.
- 10 Organise a small dataset in a spreadsheet, chart it, and check an AI-written interpretation of that chart against the data.
- 11 Build a working AI helper with a defined input, a defined output and a named user.
- 12 Publish that helper to a link and confirm it opens for someone else, on their device, without help.
- 13 Test their build on another person without helping them, record what confused that person, and fix one thing.
- 14 Identify one privacy, honesty or fairness issue in their own project and say what they did about it.
- 15 Demonstrate the project in three minutes and answer questions about what failed.

WHAT A PARENT SEES AT THE END OF THIS LEVEL

A child who can explain AI at the dinner table without using a word they cannot define, a folder of things they actually made, a model they trained themselves along with a written record of everything it got wrong, and a working helper on a link the parent can open on their own phone, any time, without the child present.

01 Meet the Machine

PHASE · FOUNDATIONS

CONCEPTS	What AI is and is not. Rule-based programs versus machine learning versus generative AI. How a model predicts rather than looks up. The five words that matter: model, prompt, output, training data, hallucination. Myths against reality, and where AI already sits in a student's ordinary day.
TOOLS	ChatGPT (trainer-hosted) · Napkin AI (trainer-hosted)
IN SESSION	The Same Question, Five Ways. Students ask one factual question in five phrasings and log how the answer shifts. They then build a one-page visual of how a model answers, in Napkin AI, generated from their own three-sentence explanation rather than a supplied definition.
DELIBERATE FAILURE	Catch the Machine. Ask something the model cannot possibly know, such as a detail about the student's own school, and identify the invented specifics inside a fluent, confident answer.
WALKS OUT WITH	AI vocabulary card · first hallucination log · a Napkin diagram of how a model answers
BETWEEN SESSIONS	Find three places AI is already used at home or in school. Write one line on what each one decides. 30 minutes.

02 Talk to AI Properly

PHASE · FOUNDATIONS

CONCEPTS	The five parts of a strong prompt: role, task, context, constraints, example. Changing one thing at a time. Asking for a format. Making the model ask you questions before it answers. Knowing when to stop prompting and think.
TOOLS	ChatGPT (trainer-hosted) · Google Docs or Sheets on the Taivas account for the library
IN SESSION	The Prompt Ladder. Start from a four-word prompt, add one element per rung, record the output at each rung, and mark the rung at which it became useful. Students then build a library of reusable prompts tagged by the job each does.
DELIBERATE FAILURE	The Blind Test. Students swap prompts with a partner and run them cold. Most prompts do not survive contact with someone who was not in the writer's head, which is the point.
WALKS OUT WITH	Personal Prompt Library of 8 to 10 reusable prompts, each with a note on what it is for
BETWEEN SESSIONS	Use three of your prompts on real schoolwork. Bring the one that failed and be ready to say why. 30 to 40 minutes.

03 Design It With AI

PHASE · CREATE

CONCEPTS	What makes a design work: hierarchy, contrast, one clear message. Briefing a design tool properly. Choosing between options instead of taking the first. Editing AI output rather than shipping it raw. Decoration against communication.
----------	---

TOOLS	Canva AI on a Canva for Education class · Napkin AI
IN SESSION	Design a poster for a real school need: a club drive, a safety notice, a reading week. Generate three concepts, choose one and state why, then edit it properly. Fix the text, fix the hierarchy, remove what does not help.
DELIBERATE FAILURE	Raw versus Edited. The untouched AI version and the student's edited version go side by side for class critique. Students must name three specific things the tool got wrong on its own.
WALKS OUT WITH	A finished poster and a short written note on what was changed and why
BETWEEN SESSIONS	Redesign the worst notice you can find at home or at school. Bring both versions. 30 minutes.

04 Pictures, Voice & Story

PHASE · CREATE

CONCEPTS	How image generation works, simply. Describing a picture properly: subject, style, mood, framing. AI voice and avatars. What a deepfake is and why consent matters. Crediting AI use honestly, as a strength rather than an admission.
TOOLS	AI image tool (trainer-hosted) · Canva AI · AI voice-over (trainer-hosted)
IN SESSION	Build a 60 to 90 second story or explainer on a chosen topic: generate the images, write the script, add narration, and assemble it so it plays end to end.
DELIBERATE FAILURE	The Line You Do Not Cross. Students identify one image that should not be generated, such as a real classmate or a fake event, and write the rule their own class should follow.
WALKS OUT WITH	An illustrated, narrated story or explainer, plus a one-line AI disclosure statement
BETWEEN SESSIONS	Write the disclosure line for your piece and one AI rule you think your class should adopt. 30 minutes.

05 Study Smarter, Check Everything

PHASE · INVESTIGATE

CONCEPTS	Using AI to learn rather than to copy. Summarising against understanding. Making the machine quiz you instead of answer for you. The check-the-source habit. What must never be pasted into a chatbot: your own personal details and, more importantly, other people's.
TOOLS	ChatGPT (trainer-hosted) with a teacher-supplied source pack · Google Docs · Napkin AI
IN SESSION	Build a study system: a repeatable set of prompts that turns any chapter into notes, a visual map and a ten-question quiz. Students run it live on a real chapter and keep the workflow card.
DELIBERATE FAILURE	Fact-Check Face-Off. Five AI-written statements are handed out. Three are true and two are subtly wrong. Every student must mark each one and point to the evidence in the source pack that settles it.
WALKS OUT WITH	My AI Study System workflow card · a verified chapter summary with the sources marked
BETWEEN SESSIONS	Run the study system on a real chapter and bring the quiz it produced. 40 minutes.

06 Teach a Machine

PHASE · INVESTIGATE

CONCEPTS	Data, labels and classes. Training against testing. Why a model is only as good as its examples. What bias looks like inside a dataset of thirty photographs. Accuracy, and the specific cases a model gets wrong.
TOOLS	Teachable Machine (no account) · Google Sheets
IN SESSION	Train a three-class image or sound classifier on examples the students collect themselves in the room. Then change only the training data by adding examples, balancing the classes and varying the background, retrain, and chart accuracy before against after.
DELIBERATE FAILURE	The Break Test. Feed the model ten awkward examples it has never seen: bad light, odd angles, a class it was never taught. Every miss is logged in Sheets with a note on why the student thinks it happened.
WALKS OUT WITH	A trained classifier · a failure log · a before and after accuracy chart
BETWEEN SESSIONS	Write three sentences on what your model learned from you rather than from the internet. 30 minutes.

07 Build Your First AI Helper

PHASE · BUILD

CONCEPTS	Turning a real problem into a simple product. Input, what it does, output. Who it is for. The difference between something that works when you demonstrate it and something that works when a stranger opens it on their own phone. Writing a workflow down so it repeats.
TOOLS	ChatGPT (trainer-hosted) · Google Sites or Canva Sites · Google Sheets · Canva AI

IN SESSION	Choose from the project menu overleaf. Define the input, the AI step and the output on paper before touching a keyboard. Build version one. Publish it so it opens on someone else's screen. SHIP CHECK, last 20 minutes. Every student publishes to Google Sites or Canva Sites on their Taivas account, copies the link into the class Sheet, and opens a classmate's link on a phone. A link that does not open on a phone is not finished. It is fixed in the room, not at home.
DELIBERATE FAILURE	Hands Off. Give the laptop to a classmate and watch them use it in silence. No hints, no explaining. Everything that confuses them is written down and one weakness is fixed on the spot.
WALKS OUT WITH	A working version one, peer test notes, a version two with one fixed weakness, and a live link in the class Sheet tested on a phone
BETWEEN SESSIONS	Ask a parent to forward your link to one family member. Ask that person to use it without you in the room. Note the first thing they got wrong. 30 minutes.

08 Show & Explain

PHASE · SHOW

CONCEPTS	Explaining a build to someone who was not there. The honest slide: what did not work. What I would do next if I had another month.
TOOLS	Canva AI · the live project link
IN SESSION	SEND IT HOME, first 10 minutes. The trainer projects one instruction. Every student's link goes to the parent in the room, from the class Sheet or from the parent's own phone, and the parent opens it live. Under-13 students hold no messaging accounts, so the trainer sends on their behalf where needed. Anything that fails is fixed by the trainer in under two minutes. A three-minute live demonstration follows, run from the phone screen and never from a slide, then five minutes of questions from the trainer and the cohort. Every student must show one failure and the fix that followed it. The session closes with a written reflection and a next-step plan.
ASSESSED ON	Does it work, is it useful to the person it was built for, can the student explain what it uses and where it breaks, and did they improve it on evidence rather than on instinct.
WALKS OUT WITH	The final project link, a presentation, a completed Project Card and the certificate

The final project

One helper, built for one real person, published to a link, that still works when the student is not standing next to it. The project is assembled from the work of sessions 03 to 07, never started from nothing in the last week.

WHAT EVERY PROJECT AT THIS LEVEL MUST CONTAIN

REQUIREMENT	STANDARD
A NAMED USER	A real person: a classmate, a younger sibling, a teacher. Not "students" in general.
A DEFINED INPUT AND OUTPUT	The student states what goes in and what comes out in one sentence each, without opening the project.
A LIVE LINK	Opens on a phone, on a network the student is not on, without the student present.
CHECKED CONTENT	Anything factual inside it is verified against a supplied source.
ONE TESTED IMPROVEMENT	A real person used it, something confused them, and the student changed that specific thing.
ONE STATED LIMIT	The student names a question their helper answers badly, out loud, during the demo.

PROJECT MENU

PROJECT	WHAT IT DOES	TOOLS, PUBLISH SURFACE AND THE SKILL IT PROVES
SCHOOL FAQ HELPER	Answers what new students actually ask, from information the school has approved.	ChatGPT, Sheets, published on Sites. Proves grounding in an approved source rather than inventing.
STUDY BUDDY	Explains a topic simply and generates practice questions from supplied material.	ChatGPT, Napkin AI, published as a Canva Site. Proves prompt design that gives consistent output.
CLASS QUIZ MASTER	Builds and runs revision quizzes from a chapter, with answers and explanations.	ChatGPT, Sheets, published on Sites. Proves structured output and a verified answer key.
READING RECOMMENDER	Asks about interests and suggests titles from an approved list in a spreadsheet.	Sheets, ChatGPT, published on Sites. Proves a helper can be bounded to a list it may not leave.
AWARENESS PAGE	An illustrated page on a school or environment issue, with sources shown.	Canva AI, image tool, published on Sites. Proves design judgement and honest sourcing.
SORTING ASSISTANT	A trained classifier put to real use: waste, lost property, shelving, accuracy published.	Teachable Machine export link wrapped in a Site, Sheets. Proves training, testing and honest reporting.

ASSESSMENT AT THIS LEVEL

COMPONENT	MARKS	WHAT FULL MARKS LOOK LIKE
CONCEPT UNDERSTANDING	20	Explains models, prompts, training data and hallucination in their own words.
TOOL FLUENCY AND WORKFLOW	20	Completes the core task in each tool with decreasing trainer support.
PROJECT QUALITY	25	The helper works for its intended person from the published link, is clearly the student's own, and is useful.
TESTING AND IMPROVEMENT	15	Tested on a real person; the fix is traceable to what was observed.
RESPONSIBLE AI	10	Names a privacy, honesty or fairness issue in their own work, and the response.
DEMONSTRATION	10	Three clear minutes: what it does, what failed, what changed, what is next.

THE CHECK THE TRAINER RUNS ON EVERY STUDENT

What did you build? Who is it for? What did you feed it? How did you know that fact was true? Where does it live, and did it open on someone else's phone? What did you change, and how did you know it worked? Where does it still fail?

LEVEL 02 · MIDDLE AND SECONDARY SCHOOL

GRADES 7-9 AI INNOVATOR

Build & Solve

OBJECTIVE

Students move from using AI to applying it against a real, evidenced problem. They compare models rather than trusting one, engineer prompts that return a reliable structure, ground research in a source set, evaluate a classifier properly, analyse a dataset, and publish a prototype tested on real users from its own link.

THE SHIFT THIS LEVEL IS DESIGNED TO PRODUCE

From it looks right to here is the evidence. A student at this level does not claim their build is better. They show the number that moved and the test that produced it.

THE SPREAD INSIDE THIS BAND

Grade 7 and Grade 9 are three school years apart, and this is the widest band in the programme. The spine, the tools and the standard of proof are identical across all three grades. What scales with age is the size of the source set, the ambition of the problem chosen in session 05, and how much of the prototype the student scopes without help. Three user tests is the floor at every grade, not a target that rises.

Batches are formed from adjacent grades wherever enrolment allows, so a batch runs 7–8 or 8–9 rather than 7 with 9. A batch is never mixed across levels.

DESIGNED TO FIT SCHOOL ASSESSMENT

The final project is sized and documented to sit inside a Class 9 project file, an internal assessment or a school innovation fair, without rework.

COGNITIVE FOCUS

Apply, analyse and evaluate. Students compare, measure, justify a choice against criteria they wrote themselves, and identify what their own evidence does not prove.

TOOL SET AT THIS LEVEL

ChatGPT · Claude, trainer-hosted · NotebookLM · Perplexity · Teachable Machine · Google Sheets · Google AI Studio, trainer-hosted · Canva AI · Napkin AI · Gamma · AI image, video & voice · Google Sites or Canva Sites

New at this level: Claude as a second model to compare against, NotebookLM for grounded research, Perplexity for verification, Google AI Studio for prototyping, and Gamma for pitch assembly. Still excluded: Hugging Face, Colab, Python, app builders and automation platforms. There is no code at this level, in any session.

PRIOR KNOWLEDGE ASSUMED

None formally. Students who completed Grades 5–6 move faster in sessions 02 and 06. Students joining fresh are brought up in session 01, which teaches the foundations at this level's depth rather than repeating the junior version.

ACCOUNTS AT THIS LEVEL, STATED PLAINLY

This band straddles the thirteenth birthday. Grade 7 students are typically under 13; Grade 9 students are typically over it. The programme does not run on that distinction.

Every student works on a Taivas-issued Google Workspace for Education account and a Canva for Education class. Claude and Google AI Studio state a minimum age of 18 and are therefore trainer-hosted in every session at this level, never set as homework. Where a tool permits accounts at 13, a student of 13 or over may use one with recorded parental consent, but no between-session task is ever written to require it. A Grade 7 student is never disadvantaged and never asked to hold an account they are not old enough for.

Learning outcomes

By the end of this level, a student can:

- 01** Explain training data, tokens and the context window, and say why two models answer the same prompt differently.
- 02** Compare two models against success criteria they wrote first, and justify a choice.
- 03** Write prompts that return a specified structure reliably across inputs, and store them as versioned templates.
- 04** Detect and document a hallucination, then redesign the prompt so it happens less.
- 05** Run grounded research, trace every key claim to a source, and state what the sources cannot answer.
- 06** Convert research into a problem statement with a measurable success criterion, and a campaign asset set held together by one visual system.
- 07** Produce a short multimodal piece and state the authenticity, consent and rights issues in it.
- 08** Build a balanced training set, train a classifier and report accuracy per class, not overall.
- 09** Identify which class a model fails on, form a hypothesis, and test it by changing the data.
- 10** Clean, chart and interpret a dataset, and disprove at least one AI-generated claim about it.
- 11** Build a prototype for a real user, test it with three people and evidence one measurable improvement.
- 12** Publish the prototype to a link and run every user test from that link, on the user's own device.
- 13** Connect two tools into a workflow that repeats, and complete a Project Card recording sources, intended use, tested performance and limits.
- 14** Pitch the project in five minutes and defend the evidence under challenge.

WHAT SEPARATES THIS LEVEL FROM GRADES 5 TO 6

At Grades 5–6 a student checks a fact; here they trace every claim to a source set. There they log what their model missed; here they compute accuracy per class and test a hypothesis. There they fix what confused a classmate; here they test three users from a published link and report the number that moved.

01 Inside the Model

PHASE · FOUNDATIONS

CONCEPTS	How a large language model is trained and what it stores. Tokens and the context window. Why two models differ on one prompt. Hallucination as a predictable property rather than a bug. Where machine learning sits inside AI.
TOOLS	ChatGPT and Claude (trainer-hosted), side by side
IN SESSION	Model Bake-Off. Students write the scoring criteria first, then run the same three prompts through two models and score both. They finish by naming, for each model, one job it should be given and one job it should not.
DELIBERATE FAILURE	The Confidence Trap. Find a question where both models are fluent, agree with each other, and are wrong. Write up why the agreement proved nothing.
WALKS OUT WITH	A scored model comparison table · a written hallucination case study
BETWEEN SESSIONS	Find one task where the weaker model in your test would still be the better choice, and say why. 40 minutes.

02 Prompting with Precision

PHASE · FOUNDATIONS

CONCEPTS	Structured prompting: role, task, context, constraints, examples, output schema. Few-shot examples. Asking a model to show its working. Self-critique prompts and their limits. Templates, version numbers, and why yesterday's prompt stops working.
TOOLS	ChatGPT · Claude (trainer-hosted) · Google Sheets as a versioned prompt library
IN SESSION	Build one prompt that returns the same structure across five different inputs. The test is consistency, not cleverness. Then write a rubric, have the model grade its own output against it, and check whether that grade was honest.
DELIBERATE FAILURE	The Fifth Input. Most prompts hold for four inputs and collapse on the fifth. Students find the input that breaks their own template and fix the prompt rather than the input.
WALKS OUT WITH	A versioned prompt template library with success criteria and pass or fail notes per input
BETWEEN SESSIONS	Run your template on five inputs you have not seen. Record every failure. 40 minutes.

03 Visual Communication & Campaign Design

PHASE · CREATE

CONCEPTS	Message architecture: one audience, one claim, one action. A visual system rather than three unrelated posters. Information design against decoration. AI for concepts, judgement kept human. Charts that carry real data honestly.
TOOLS	Canva AI · Napkin AI · AI image tool · Gamma

IN SESSION	Produce a three-piece campaign set on the problem being investigated: a social asset, an infographic carrying real numbers, and a one-slide summary. All three share type, colour and grid.
DELIBERATE FAILURE	The Misleading Chart. Students build a chart that is technically true and genuinely misleading, then fix it, and write down the rule they broke.
WALKS OUT WITH	A campaign asset set with a stated audience, claim and call to action
BETWEEN SESSIONS	Show the campaign to two people outside the cohort. Ask what the action is. Record their answers. 40 minutes.

04 Multimodal Storytelling

PHASE · CREATE

CONCEPTS	Image, video, voice and avatar generation. Prompting for consistency, so a character survives across shots. Synthetic media, deepfakes and consent. Provenance, watermarking, disclosure. Copyright and the training-data question, from both sides.
TOOLS	AI image tool · AI video · AI voice (all trainer-hosted) · Gamma
IN SESSION	Produce a 60 to 90 second explainer for the campaign, with consistent characters and narration that matches the evidence rather than exceeding it.
DELIBERATE FAILURE	The Provenance Lab. Students see how easily a plausible fake is made, stop, and draft the synthetic-media rules they believe their own school should adopt.
WALKS OUT WITH	A short film or explainer · a disclosure statement · a one-page school policy proposal
BETWEEN SESSIONS	Find one real synthetic media incident. Write four lines on what it cost someone. 40 minutes.

05 Source-Grounded Research

PHASE · INVESTIGATE

CONCEPTS	Grounded answering against open generation. Building a source set worth trusting. Evidence tracing. Primary against secondary sources. What the sources do not say. Writing the success criterion before the solution exists.
TOOLS	NotebookLM on the Taivas tenant · Perplexity · ChatGPT or Claude · Google Docs
IN SESSION	Load six to ten approved sources into NotebookLM and interrogate them. Every claim in the brief carries a source line, and it closes with a measurable success criterion for whatever gets built in session 07.
DELIBERATE FAILURE	The Gap Hunt. List three questions the source set cannot answer and say how you would find out. A strong brief names its own holes.
WALKS OUT WITH	An evidence brief with a traced claim table, a problem statement and a success criterion
BETWEEN SESSIONS	Cross-check two of your traced claims in Perplexity. Report any that do not hold. 40 minutes.

06 Data, Training & Evaluation

PHASE · INVESTIGATE

CONCEPTS	Dataset design and class balance. Splitting training from testing, and why testing on training data proves nothing. Why an overall accuracy figure hides the failure. Error per class. How bias enters through data. Verifying an AI reading of a dataset.
TOOLS	Teachable Machine · Google Sheets · Claude or ChatGPT for analysis
IN SESSION	Train a classifier for the chosen problem, build a twenty-case unseen test set, record every prediction in Sheets and compute accuracy per class. Then clean and chart a real dataset and ask AI for three observations.
DELIBERATE FAILURE	Unbalance It. Skew the training data and show the effect on the weakest class. Separately, disprove at least one AI observation against the actual numbers.
WALKS OUT WITH	An evaluation table with accuracy per class · a data story containing one AI claim disproved
BETWEEN SESSIONS	Collect ten more examples for your weakest class. Retrain and record what moved. 45 minutes.

07 Build, Test & Improve

PHASE · BUILD

CONCEPTS	Prototype scoping. Input, AI step, output. The human-in-the-loop check and where to place it. Usability testing with people who were not involved. Measuring an improvement instead of asserting one. Why a test run on the builder's laptop proves nothing.
TOOLS	Google AI Studio (trainer-hosted) · Google Sheets · Google Sites or Canva Sites · Gamma
IN SESSION	Build version one against the written problem statement. Test with three users on a fixed

	task, using a fixed observation sheet, in silence. Fix only the highest-impact weakness, then re-test. SHIP CHECK, last 20 minutes. Version two is wrapped in a Sites or Gamma page that states the problem, the success criterion and the before and after number, and links to the prototype. The link goes into the class Sheet. Every user test from this point is run from that link, on the user's device, not from the builder's laptop.
DELIBERATE FAILURE	No Helping. The builder may not speak during the test. Everything the user gets wrong is the build's fault, and is recorded as such.
WALKS OUT WITH	A working version two · three completed user test sheets · a before and after measure · a public project page carrying the prototype link
BETWEEN SESSIONS	Run one more test with someone outside your age group, from the link, on their device. Record where it broke. 45 minutes.

08 Pitch, Demo & Defend

PHASE · SHOW

CONCEPTS	Pitch structure: problem, evidence, solution, test, limits, next step. Defending a claim rather than repeating it. The limitations slide as a mark of seriousness, not an apology.
TOOLS	Gamma or Canva AI · the live prototype link
IN SESSION	SEND IT HOME, first 10 minutes. Every student sends their project page link to the parent who is with them, and the parent opens it live on their own phone. The trainer projects the class Sheet: ten names, ten links, ten green ticks. Anything that fails is fixed by the trainer in under two minutes. A five-minute pitch follows, opening from the link on the parent's phone rather than from a slide, then hostile questioning: show me the evidence, what happens if I do this, what does it get wrong, who does it not work for.
ASSESSED ON	Whether the evidence supports the claim, whether the improvement is measured rather than asserted, and whether the student knows their own limits.
WALKS OUT WITH	The final prototype link, a pitch deck, a completed Project Card and the certificate

The final project

A researched problem, a working prototype on a public link, and a number that moved because of something the student changed. The build grows out of the evidence brief written in session 05, never chosen for convenience in session 07.

WHAT EVERY PROJECT AT THIS LEVEL MUST CONTAIN

REQUIREMENT	STANDARD
A RESEARCHED PROBLEM	Every key claim in the brief traces to a source in the set.
A SUCCESS CRITERION	Written before the build, measurable, and specific enough to fail.
A WORKING PROTOTYPE	Opens and works for someone who was not in the room and gets no instructions.
A LIVE LINK	The project page and prototype open on a stranger's device with no instructions.
A STRUCTURED USER TEST	Three users, one fixed task, one observation sheet, run from the link, no help from the builder.
ONE EVIDENCED IMPROVEMENT	A stated before figure, a change, a stated after figure. Not "it feels better".
A PROJECT CARD	Purpose, data sources, intended user, tested performance, limits, next step.

PROJECT MENU

PROJECT	WHAT IT DOES	TOOLS, PUBLISH SURFACE AND THE SKILL IT PROVES
GREEN SCHOOL DASHBOARD	Turns school water, energy or waste data into verified trends and an action.	Sheets, AI Studio, NotebookLM, published on Sites with an embedded Sheet chart. Proves data literacy and an evidenced recommendation.
SMART SORTER	A trained classifier for waste, recycling or lost property, accuracy per class.	Teachable Machine export wrapped in a Site, Sheets. Proves honest evaluation, including where it fails.
COMMUNITY INFORMATION HELPER	Answers a real local need from approved sources: exams, health awareness, transport.	NotebookLM, AI Studio share link, published on Sites. Proves grounding and refusing out-of-scope questions.
JUNIOR STUDY COMPANION	An AI study system for a Grade 5 or 6 student, tested on a real younger student.	Claude, Sheets, published on Sites. Proves designing for a user unlike yourself.
ACCESSIBILITY ASSISTANT	Simplifies notices, writes alt text or reads content aloud for a stated need.	AI Studio link, Canva AI, voice, published on Sites. Proves a build defined by a constraint, not a feature list.
ISSUE CAMPAIGN & DATA STORY	An evidence brief, a dataset, campaign assets and a call to action, tested on viewers.	NotebookLM, Sheets, Canva AI, published as a Gamma page. Proves research, data and communication as one piece.

ASSESSMENT AT THIS LEVEL

COMPONENT	MARKS	WHAT FULL MARKS LOOK LIKE
CONCEPT UNDERSTANDING	15	Explains model behaviour, evaluation and bias, and applies it to their build.
TOOL FLUENCY AND WORKFLOW	15	Moves between research, data, prototype and pitch tools unprompted.
PROJECT QUALITY	25	Solves the stated problem for the stated user from a public link, and is clearly the student's own.
EVIDENCE, TESTING AND IMPROVEMENT	25	Claims are traced, the test is structured and run from the link, the improvement is a number that moved.
RESPONSIBLE AI AND DOCUMENTATION	10	The Project Card is complete and the limitations are stated unprompted.
PITCH AND DEFENCE	10	Holds up under challenge, and concedes where the evidence does not reach.

THE CHECK THE TRAINER RUNS ON EVERY STUDENT

What is the problem, and who says so? What is your success criterion? Where does it live, and did a stranger open it? Who tested it, and what did they get wrong? What number moved? Which class does your model fail on? What are you claiming that your evidence does not support?

LEVEL 03 • SENIOR SCHOOL

GRADES 10-12 AI ARCHITECT

Design, Analyse & Innovate

OBJECTIVE

Students design, build, evaluate, automate and document an AI system that does one real job end to end. The work is grounded in a corpus or a dataset, scored against a written evaluation set rather than an impression, published to a link, and shipped with a human check and a statement of what it cannot do.

THE SHIFT THIS LEVEL IS DESIGNED TO PRODUCE

From I built something that works to here is what it scores, here is where it breaks, and here is why I chose this design over the alternative.

WHAT THIS LEVEL IS NOT

It is not machine learning engineering. Nobody trains a foundation model, and nobody is taught to program. Students learn to select, ground, evaluate, orchestrate and document AI systems, which is what applied AI work consists of. The only code skill assessed here is reading: the student reads the code AI writes for them, explains any line on request, and deletes what they cannot explain.

COGNITIVE FOCUS

Analyse, evaluate and create. Students design under constraint, defend a trade-off, quantify performance, and report a result honestly, including a result that shows their change did nothing.

TOOL SET AT THIS LEVEL

Claude, trainer-hosted • ChatGPT • Google AI Studio, trainer-hosted • NotebookLM • Perplexity • Hugging Face • Google Colab • Google Sheets with Apps Script • AI app builder • Gamma • Canva AI • Napkin AI

New here: Hugging Face for model cards and inference demos, Colab as a place to read and run AI-drafted analysis, an AI app builder for a real front end, and Apps Script on Sheets as the automation layer. Apps Script is chosen deliberately: it lives inside the Taivas Workspace tenant, needs no card, and has triggers and execution logs built in. Zapier and Make are demonstrated, never built on. Teachable Machine is retired.

PRIOR KNOWLEDGE ASSUMED

None formally. Session 01 rebuilds the foundations at senior depth. Students arriving from Grades 7–9 already hold the evaluation habit, the single biggest predictor of how far a student gets by session 07.

GRADE 10 AND THE BOARD YEAR

Grade 10 sits at the bottom of this band and is a board year in most Indian systems. The programme is scoped for that and does not pretend otherwise.

Between-session work stays at 45 minutes and never rises. The final project is documented to sit inside a Class 10 internal assessment, project file or innovation fair without rework, so the work counts twice rather than competing with the syllabus. Schools running this in Grade 10 are advised to place the cohort in Term 1 or an activity block rather than in the run-up to the board examinations. A Grade 10 student who needs the senior content at a lighter load is placed in the Grades 7–9 cohort, which is the correct answer rather than a demotion.

ACCOUNTS AT THIS LEVEL, STATED PLAINLY

Claude.ai and Google AI Studio state a minimum age of 18. Every student in this band is below it. Both are therefore trainer-hosted in every session and never set as homework; the share link the trainer produces is what the student owns and builds on.

Everything else runs inside the Taivas Workspace for Education tenant: Sheets, Apps Script, Colab, NotebookLM, Sites. The AI app builder is selected before each cohort on one test, that it publishes without a card and without Git, and is used trainer-hosted or on a consented account. GitHub, Vercel and Netlify are not in the programme.

Learning outcomes

By the end of this level, a student can:

- 01 Draw a system diagram showing inputs, retrieval, the model step, the human check, outputs and failure paths.
- 02 Explain tokens, embeddings, context windows, temperature and cost against latency, and set them deliberately.
- 03 Explain retrieval-augmented generation conceptually and build an assistant that answers only from a supplied corpus.
- 04 Engineer a system prompt with role, constraints, refusal behaviour, an output schema and edge cases.
- 05 Recognise prompt injection, explain why untrusted text is not an instruction, and defend against it.
- 06 Write an evaluation set, score a system against it, and report a pass rate rather than an impression.
- 07 Read a model card, run inference demos and write a defensible model selection memo.
- 08 Read, correct and explain AI-drafted analysis code that loads, cleans, summarises and charts a dataset.
- 09 Define a baseline, change exactly one variable, re-score, and report the result including a null result.
- 10 Design and deploy an automation that runs on a trigger and does a real job without being rebuilt each time.
- 11 Use AI to write, review and debug code; read every line they ship and delete any line they cannot explain.
- 12 Design an interface that shows its sources, fails gracefully and states what it refuses to do.
- 13 Publish the interface, the run log and the Model Card so that each opens from its own shared URL on a phone.
- 14 Produce a Model Card covering purpose, data, method, evaluation, limitations, risk and appropriate use, and assess the system for bias, privacy, security, environmental cost and misuse.
- 15 Defend architecture and evaluation choices in a viva, on a live input they have not seen, entered through the published interface.

WHAT SEPARATES THIS LEVEL FROM GRADES 7 TO 9

There a student tests a build on three users; here they score it against a written evaluation set and report the pass rate. There they improve it; here they change one variable, re-run the same evaluation, and accept the answer even when nothing moved. There the link is a page; here it is a running system with a log.

01 How Models Behave

PHASE · FOUNDATIONS

CONCEPTS	Architecture at a working level: tokens, embeddings, context window, temperature and sampling. Why one prompt gives different answers. Capability against reliability. Cost and latency as design constraints. Open against closed models.
TOOLS	Claude (trainer-hosted) · ChatGPT · Google AI Studio with parameter controls (trainer-hosted)
IN SESSION	Parameter Lab. Run one prompt across several temperature settings and document the spread. Then pad the context with irrelevant material and measure where quality starts falling. Finally, map a real AI product they use into a system diagram.
DELIBERATE FAILURE	Lost in the Middle. Bury the key fact halfway through a long context and watch a capable model walk past it. Students record the length at which their model stopped being reliable.
WALKS OUT WITH	A behaviour log with parameter evidence · a system map of an existing AI product
BETWEEN SESSIONS	Pick one AI product and write what you think it retrieves, what it generates and where the human sits. 45 minutes.

02 Context Engineering & Reliability

PHASE · FOUNDATIONS

CONCEPTS	System prompts against user prompts. Output schemas. Few-shot design. Refusal and fallback behaviour. Edge cases and adversarial inputs. Prompt injection, and why text retrieved from anywhere is data rather than an instruction. Versioning and regression testing.
TOOLS	Google AI Studio (trainer-hosted) · Claude (trainer-hosted) · Google Sheets as the evaluation set
IN SESSION	Write a system prompt for a defined job, then a fifteen-case evaluation set including three adversarial cases, and score version one against it. Iterate the prompt and re-run the identical set. The score has to move, or the change was cosmetic.
DELIBERATE FAILURE	Injection Exercise. Students attempt to hijack a classmate's system with instructions hidden inside ordinary-looking input, then write the defence into their own system prompt.
WALKS OUT WITH	A versioned system prompt · an evaluation set with pass rates for v1 and v2 · an injection findings note
BETWEEN SESSIONS	Add five cases to your evaluation set that you expect your system to fail. Score it. 45 minutes.

03 Interface & Product Design

PHASE · CREATE

CONCEPTS	Designing for a user who does not know how it works. Making uncertainty visible instead of hiding it. Affordances, empty states and error states. When to show sources. Building a design system rather than assorted screens.
----------	--

	Reading code: variables, functions, calls, conditions. Enough to explain any line you shipped. Twelve minutes, once, here, and nowhere else in the programme.
TOOLS	One AI app builder with a one-click publish, selected before the cohort · Claude (trainer-hosted) for code review · Napkin AI for architecture diagrams · Canva AI
IN SESSION	Design and build the front end for the final system, including how it displays sources, how it fails gracefully, and what it declines to answer. Every generated line must be explainable by the student who shipped it. The interface is published at the end of the session and the URL goes into the class Sheet. Unpublished is unfinished.
DELIBERATE FAILURE	Explain This Line. The trainer selects a random line of AI-generated code and asks what it does. Anything the student cannot explain is deleted and regenerated. This exercise is the entire programming assessment in the programme.
WALKS OUT WITH	A published interface URL · an architecture diagram · a written design rationale
BETWEEN SESSIONS	Show the interface to someone non-technical, from the link, on their device. Note every moment they hesitate. 45 minutes.

04 Generative Media, Rights & Authenticity

PHASE · CREATE

CONCEPTS	Diffusion at a conceptual level. Controlling consistency across generated assets. Provenance standards and synthetic-media detection. Copyright, training data and the live legal argument on both sides. Disclosure norms. Brand and voice consistency in a product.
TOOLS	AI image, video and voice tools (trainer-hosted) · Gamma · Canva AI
IN SESSION	Produce the media layer for the final product under one consistent identity, then complete a Rights and Provenance Review of the student's own assets rather than of someone else's.
DELIBERATE FAILURE	Which One Is Real. A blind identification exercise on generated and authentic media. Students record their own accuracy, which is usually humbling, and draw the conclusion themselves.
WALKS OUT WITH	A product media set · a Rights and Provenance Review listing tool, licence, risk and disclosure per asset
BETWEEN SESSIONS	Write the disclosure statement your product will carry, in the words a user would read. 40 minutes.

05 Research, Retrieval & Open Models

PHASE · INVESTIGATE

CONCEPTS	Grounding and retrieval-augmented generation, conceptually. Building and curating a corpus. Why long documents break, and what chunking is for. Citation and traceability. The open model ecosystem. Reading a model card. Benchmark literacy, and what a benchmark score does not tell you.
TOOLS	NotebookLM on the Taivas tenant · Hugging Face (trainer-hosted demos) · Perplexity · Claude (trainer-hosted)
IN SESSION	Build a grounded assistant that answers only from a curated corpus. On Hugging Face, shortlist three models for a narrow task, read their model cards, run the inference demos and write a selection memo that recommends one.
DELIBERATE FAILURE	The Refusal Test. Ask five questions the corpus cannot answer. A grounded system that answers all five confidently has failed, however good the answers sound.
WALKS OUT WITH	A grounded assistant with a refusal test log · a model selection memo with a recommendation
BETWEEN SESSIONS	Add three documents to your corpus and re-run the refusal test. Report any regression. 45 minutes.

06 Data Analysis, Metrics & Experiments

PHASE · INVESTIGATE

CONCEPTS	Cleaning and missing values. Descriptive statistics. Correlation against causation. Baselines. Controlled experiments and changing one variable. Choosing the metric before the result. Reporting a null result. Visualising honestly.
TOOLS	Google Colab on the Taivas account · Google Sheets · Claude (trainer-hosted) for code drafting and review
IN SESSION	Load a real dataset in Colab, clean it, compute summary statistics and produce two charts, with AI drafting every cell and the student reading, correcting and explaining it. No cell survives that the student cannot explain. Then run one controlled experiment on their own system.
DELIBERATE FAILURE	Accept the Null. Many changes will not move the score. Students report that honestly and are marked up for it, not down.
WALKS OUT WITH	A view-only Colab notebook link · an experiment report stating baseline, variable, result and interpretation
BETWEEN SESSIONS	Run a second experiment changing a different single variable. Report both results. 45 minutes.

07 Architect & Automate

PHASE · BUILD

CONCEPTS	Workflow and agent design. Triggers, actions, conditions and error handling. When a task
----------	--

	should not be automated. Human-in-the-loop checkpoints. Logging and monitoring. Key hygiene and basic security. Controlling cost.
TOOLS	Google Sheets with Apps Script (primary) · Google AI Studio (trainer-hosted) · Claude (trainer-hosted) for code review · Zapier or Make, demonstration only
IN SESSION	Build an automation that does one real job on a trigger: an incoming item is classified, summarised, logged and a draft response is prepared for human approval. Run it over ten real cases and log every failure. SHIP CHECK, last 20 minutes. The system has three public artefacts in the class Sheet: the published interface URL from session 03, the Apps Script run-log tab shared view-only, and the Model Card as a Gamma or Sites page. All three open on a phone.
DELIBERATE FAILURE	Ten Real Cases. Demonstrations pass. Ten real inputs do not. Students add error handling for the two most common failures and re-run, then write the security note: what breaks this, what data it touches, where the human sits.
WALKS OUT WITH	A deployed automation with a public run log · a run log over ten cases · a failure and security note
BETWEEN SESSIONS	Run it on ten further cases without intervening. Report the failure rate. 45 minutes.

08 Technical Demo & Viva

PHASE · SHOW

CONCEPTS	Presenting one system to a technical and a non-technical audience in the same seven minutes. Evidence over assertion. Limitations as a mark of rigour. The responsible next step.
TOOLS	The live system link · Gamma or Canva AI
IN SESSION	SEND IT HOME, first 10 minutes. Every student sends the interface link to the parent who is with them, and the parent opens it live on their own phone. The trainer projects the class Sheet: ten names, three links each, all green. A seven-minute technical demonstration follows on live inputs. The unseen input is entered by the trainer through the published interface, not through the student's laptop. A viva follows on architecture, evaluation, data provenance, failure modes and ethics.
ASSESSED ON	Whether the architecture is justified, the evaluation is real, the limitations are known before they are pointed out, and the student can answer for a design choice they made in session 03.
WALKS OUT WITH	The deployed system link, a view-only run log, a Model Card, evaluation results, the technical demo and the certificate

The final project

One system, one real job, one real user, running on a link, with the performance written down rather than asserted.

WHAT EVERY PROJECT AT THIS LEVEL MUST CONTAIN

REQUIREMENT	STANDARD
A DEFINED JOB AND USER	One job, stated in a sentence, for a named user or a named recurring task.
A SYSTEM DIAGRAM	Inputs, retrieval or data layer, model step, outputs, failure paths, and the named point where a person approves or overrides.
A GROUNDED INTELLIGENCE LAYER	A curated corpus or a real dataset, not a bare prompt with confident output.
A WRITTEN EVALUATION SET	At least fifteen cases, including adversarial ones, with a reported pass rate.
ONE CONTROLLED EXPERIMENT	A baseline, one changed variable, a re-score, the result reported either way.
A LIVE LINK	Interface, run log and Model Card each open from a shared URL, and the viva is run through the interface link.
A MODEL CARD	Purpose, data, method, evaluation, limits, risks, appropriate and inappropriate use.

PROJECT MENU

PROJECT	WHAT IT DOES	TOOLS, PUBLISH SURFACE AND THE SKILL IT PROVES
GROUNDED RESEARCH ASSISTANT	Answers only from a curated corpus, cites sources, refuses out-of-scope questions.	NotebookLM, AI Studio, Claude, published at an app-builder URL with a shared notebook. Proves retrieval, citation and refusal.
OPERATIONS AUTOMATION	Classifies, routes, summarises and drafts replies for a recurring task, with approval.	Sheets with Apps Script, AI Studio, run log shared view-only. Proves workflow design and error handling.
AI PRODUCTIVITY AGENT	Takes an ambiguous request, decomposes and prioritises it, returns a reviewable plan.	Claude, Sheets, AI Studio, published at an app-builder URL. Proves decomposition and a checkable output.
SUSTAINABILITY INTELLIGENCE	A real dataset, an analysis, a recommendation, with the uncertainty stated.	Colab view-only notebook, Sheets, published as a Gamma page. Proves analysis and honest interpretation.
APPLIED MODEL STUDY	Selects, evaluates and deploys an open model for a narrow task, benchmarked.	Hugging Face, Colab, Sheets. Proves model selection defended by measurement.
ACCESS OR LANGUAGE TOOL	Serves one access or language need, evaluated with the intended user	AI Studio, app builder URL, voice. Proves designing under a real constraint.

PROJECT	WHAT IT DOES	TOOLS, PUBLISH SURFACE AND THE SKILL IT PROVES
	group.	

ASSESSMENT AT THIS LEVEL

COMPONENT	MARKS	WHAT FULL MARKS LOOK LIKE
CONCEPT UNDERSTANDING	15	Explains retrieval, evaluation and model behaviour, applied to design choices.
TOOL FLUENCY AND WORKFLOW	10	Moves between corpus, notebook, prototype and automation unprompted.
SYSTEM QUALITY	25	Does its stated job reliably for a real user, from a live link, with the human check in place.
EVALUATION AND EXPERIMENT	25	A real evaluation set, a reported pass rate, an experiment reported either way.
RESPONSIBLE AI AND MODEL CARD	15	Bias, privacy, security and misuse assessed; limits stated before anyone asks.
TECHNICAL DEMO AND VIVA	10	Defends architecture and evaluation on a live, unseen input entered through the published interface.

THE CHECK THE TRAINER RUNS ON EVERY STUDENT

What job does it do, for whom? What does it retrieve? What is in your evaluation set, and what does it score? What did you change, and did the number move? Where does a human sit? Show me the run log on your phone. What line of this code can you not explain? What must it never do?

Responsible AI, taught where it actually bites

Responsible AI is not a closing lecture. It is taught at the moment a student is about to do the thing: when they are about to paste a classmate's name into a chatbot, generate a face, publish an unchecked fact, or ship a model that works badly for one group. Each level meets the same eight themes at the depth it can act on.

THEME	GRADES 5–6	GRADES 7–9	GRADES 10–12
PRIVACY AND PERSONAL DATA	Never paste your own or anyone else's details into a model.	What a service retains, and what a screenshot gives away.	Data minimisation by design, and what the system logs.
ACCURACY AND VERIFICATION	Check one fact against one source before using it.	Trace every claim; know what the source set cannot answer.	Ground the system, then test whether it refuses out of scope.
ATTRIBUTION AND INTEGRITY	Say what AI did. Honesty is the rule, not the exception.	School and board policy on AI in submitted work.	Disclosure norms in academic and professional contexts.
BIAS AND FAIRNESS	Notice your model works worse on examples unlike yours.	Measure error per class and name who it fails.	Assess representational harm and evaluate across groups.
CONSENT AND SYNTHETIC MEDIA	Do not generate a real person. Ask before using a likeness.	Deepfakes, provenance and a school policy proposal.	Rights, licensing and a provenance review of their own assets.
SECURITY	Do not share logins. Do not run instructions from strangers.	Why text from the internet is not an instruction.	Prompt injection, key hygiene and untrusted input handling.
PUBLISHING AND EXPOSURE	What a public link means. Nothing on the page identifies a child.	Who can reach the link, and what the page should never collect.	Access control, logging and what a run log must not store.
COST AND IMPACT	AI uses real electricity and water somewhere.	Compute and energy behind everyday generation.	Cost, latency and environmental trade-offs as design choices.
HUMAN JUDGEMENT	You decide. The machine suggests.	Where a person must sign off before anything happens.	Designing and defending the human-in-the-loop checkpoint.

SAFEGUARDING AND DATA POLICY

01	No student enters personal data, their own or anyone else's, into any AI tool during the course.
02	Every student in Grades 5–6 is under 13. No personal AI account is created at that level, on any tool, at any point.
03	Grades 7–9 straddle the thirteenth birthday. No between-session task is written to require an account a student may be too young to hold.
04	Claude and Google AI Studio state a minimum age of 18 and are trainer-hosted at every level that uses them. They are never set as homework.

05	Source material for research sessions is supplied or approved by the trainer.
06	Training data for classifiers uses objects and sounds, never identifiable images of other children.
07	Published project pages carry no student surname, no school name and no photograph of a child without written parental permission.
08	Taivas-issued accounts are provisioned per batch and revoked 30 days after the showcase.
09	Sessions are conducted in a monitored virtual classroom. Recording policy follows the school's own standard.

ACADEMIC INTEGRITY

Schools reasonably worry that an AI course teaches students to hand in work they did not do. This curriculum is built to produce the opposite habit.

PRACTICE	HOW IT IS ENFORCED
DISCLOSURE BY DEFAULT	Every artefact carries a line stating what AI was used for.
PROCESS OVER PRODUCT	Failure logs, test sheets and version history are assessed, not just the final file.
LIVE DEFENCE	Marks are carried by a viva on the student's own work. A project the student cannot explain scores badly.
VERIFICATION HABIT	Unchecked AI content is treated as an error at every level, from session 01 onward.

Delivery

Designed to be realistic to teach and realistic to finish inside thirty days, with a batch small enough that every student is seen building every week.

FORMAT

ITEM	DETAIL
STRUCTURE	8 live sessions of 90 minutes, 2 per week, across one month.
BATCH	Capped at 10 students, grouped by level, never mixed across levels. Within Grades 7–9, batches are formed from adjacent grades wherever enrolment allows.
COHORTS	Weekday and weekend options. Levels run in parallel so siblings can enrol together.
MODE	Live online with a trainer present throughout. Sessions are not pre-recorded.
BETWEEN SESSIONS	One short task at a time: collect examples, finish an asset, test the build from its link, prepare a question. 30 to 45 minutes.
SUPPORT	A cohort channel for questions between sessions, monitored by the trainer.
FINAL EXPERIENCE	Demo Day: the link goes to the parent first, then a live demonstration, questions and reflection. Parents attend.

SETUP REQUIRED

ITEM	REQUIREMENT
DEVICE	Laptop or desktop. Tablets and phones are not sufficient for sessions 06 and 07.
CONNECTION	Stable broadband. Camera and microphone on during build and demo sessions.
SOFTWARE	Browser based throughout. Nothing is installed. No card is required for any tool.
ACCOUNTS	One Taivas-issued Google Workspace for Education account per student, provisioned before session 01.
STORAGE	One shared cohort folder per student, holding every artefact for the duration.
FOR THE DEMO	A parent with a phone, present at session 08.

THE 90 MINUTES, EVERY TIME

CLOCK	PHASE	WHAT HAPPENS
0–10	Recap	Last session's artefact is shown and connected to today's idea.
10–30	Teach & demo	One concept explained, then the mechanism demonstrated live.
30–70	Build	Students work. The trainer circulates and asks diagnostic questions.
70–80	Test	Students deliberately try to break what they just made, and record it.
80–88	Show	Selected students show outputs, failures included and named as such.
88–90	Close	The artefact is confirmed and the next task is set.

In session 07 at every level the Test and Show blocks are replaced by a fixed 20-minute SHIP CHECK. In session 08 the Recap block is replaced by a fixed 10-minute SEND IT HOME.

Accounts, age and access

The grade bands in this document map onto ages that sit either side of the thirteenth birthday, and every major AI tool sets its own minimum. The table below is the operating answer. Stated ages are as known at the date of issue and are verified against live terms before every cohort.

TOOL	STATED MINIMUM AGE	5–6 (10–12)	7–9 (12–15)	10–12 (15–18)	HOW TAIVAS RUNS IT
Google Workspace: Sites, Sheets, Docs, Colab	Any, under school admin	Yes	Yes	Yes	Taivas Workspace for Education tenant. One account per student, revoked 30 days after showcase.
Teachable Machine	No account	Yes	Yes	—	Direct, no sign-in.
Canva	13+ personal; under-13 via Canva for Education	Teacher class	Teacher class	Teacher class	Canva for Education, trainer as teacher, for all three bands.
ChatGPT	13+ with parental consent	Trainer-hosted	Trainer-hosted; own at 13+ with consent	Own, with consent	Consent line carried on the enrolment form.
Claude.ai	18+	—	Trainer-hosted only	Trainer-hosted only	Never homework. In-session use only.
Google AI Studio	18+	—	Trainer-hosted only	Trainer-hosted only	Trainer runs a shared workspace; the share link is what students own.
NotebookLM	18+ personal; 13+ on Workspace for Education	—	Workspace account	Workspace account	Inside the Taivas tenant only.
Perplexity	13+	—	Trainer-hosted at Grade 7; own at 8–9	Own	Verified per cohort.
Gamma	13+	—	Trainer-hosted at Grade 7; own at 8–9	Own	Verified per cohort.
Napkin AI	Verify per cohort	Trainer-hosted	Trainer-hosted	Own	
Hugging Face	13+	—	—	Trainer-hosted demos	No student accounts. Model cards are public pages.
AI image, video	Mostly 18+	Trainer-	Trainer-hosted	Trainer-hosted	Trainer-hosted at every

TOOL	STATED MINIMUM AGE	5–6 (10– 12)	7–9 (12–15)	10–12 (15– 18)	HOW TAIVAS RUNS IT
and voice tools		hosted			level, without exception.
AI app builder	Verify per product	—	—	Trainer-hosted, or own with consent	Product selected before each cohort on one test: publishes without a card and without Git.
Apps Script	Workspace	—	—	Yes	Inside the Taivas tenant.
GitHub, Vercel, Netlify	13+	—	—	—	Not in the programme, at any level.

PROVISIONING BEFORE EVERY COHORT

01	Workspace for Education tenant with a student organisational unit per batch; Canva for Education class per batch.
02	One trainer ChatGPT seat, one trainer Claude login and one trainer AI Studio login per batch.
03	The enrolment form carries two consent lines: consent to the child using ChatGPT under trainer supervision, and confirmation that no student personal data is entered into any model.
04	Every stated minimum age in the table above re-checked against live terms, and the table reissued if any has moved.

RUNNING THIS INSIDE A SCHOOL

MODEL	HOW IT RUNS
TIMETABLED CLUB	One 90-minute slot twice a week for four weeks, or one slot a week across eight weeks where the timetable is tighter.
INTENSIVE BLOCK	Four consecutive days, two sessions a day, inside an activity week or a term break.
PARALLEL LEVELS	Grades 5–6, 7–9 and 10–12 run at the same time in separate batches, sharing one Demo Day.
GRADE 10 PLACEMENT	Scheduled in Term 1 or an activity block rather than the run-up to boards. The final project is documented to count as an internal assessment or project file entry.
THE SCHOOL PROVIDES	A room with laptops or a bring-your-own arrangement, broadband, and a staff point of contact.
TAIVAS PROVIDES	The trainer, the curriculum, all source packs and datasets, tool access, accounts, assessment and certification.

Assessment

The same six components are assessed at every level. The weighting shifts as the standard of proof rises: understanding and tool fluency matter most in Grades 5–6, evidence and evaluation dominate by Grades 10–12.

COMPONENT	5–6	7–9	10–12	WHAT IS BEING JUDGED
CONCEPT UNDERSTANDING	20	15	15	Explains the idea in their own words and applies it to their build.
TOOL FLUENCY AND WORKFLOW	20	15	10	Trainer support is measurably decreasing session by session.
PROJECT QUALITY	25	25	25	It works, from a live link, for a named person, and is clearly the student's own.
TESTING, EVIDENCE AND IMPROVEMENT	15	25	25	The improvement is evidenced rather than asserted.
RESPONSIBLE AI AND DOCUMENTATION	10	10	15	Limits and risks are stated before anyone points them out.
DEMONSTRATION AND DEFENCE	10	10	10	The student answers for their own decisions, live.
TOTAL	100	100	100	Assessed continuously across eight sessions, not in a final test.

PERFORMANCE BANDS

BAND	DESCRIPTOR
DISTINCTION • 80+	Works for a real user from a live link, the evidence is real, and limits are volunteered.
MERIT • 65–79	Works, published, testing happened, explained accurately when asked.
PASS • 50–64	Works in demonstration, some testing, explained with prompting.
NOT YET • BELOW 50	Incomplete, or the student cannot account for it. A resit demo is offered.

CERTIFICATION

A Taivas certificate is issued on four conditions together: a working final project, a live link that opens on a device the student does not own, a completed Project or Model Card, and a live demonstration with questions answered. Attendance alone does not certify. Distinction is recorded on the certificate.

THE PROJECT CARD AND MODEL CARD

A one-page Project Card at Grades 5–6, becoming a Model Card at Grades 10–12, written the way practitioners write them.

FIELD	WHAT IT RECORDS
PURPOSE	The one job this does, and for whom.
DATA AND SOURCES	What it was built on, and where that came from.

FIELD	WHAT IT RECORDS
METHOD	How it works, in the student's own words.
EVALUATION	How it was tested and what it scored.
WHERE IT LIVES	The public link, and what opens from it.
LIMITATIONS	What it gets wrong, and for whom it works worst.
APPROPRIATE USE	What it should and should not be used for.

WHAT THE PARENT RECEIVES AT THE END OF THIRTY DAYS

A child who can explain AI in plain language, a folder of artefacts from every session, a model they trained and honestly evaluated, a working project built around a real problem, a link on their own phone that opens it, written evidence of testing and improvement, and a live demonstration they can give on request.

Trainer brief

The curriculum works when the trainer protects the balance between explaining, making and testing. It fails in exactly one way: the trainer builds the project, the demo looks excellent, and the student cannot answer a question about it.

THE TWELVE NON-NEGOTIABLES

01	Every session ends with an artefact that can be opened, shown, tested or assessed.
02	Teach the idea before the tool. If the tool changes, keep the outcome and replace the tool.
03	Show at least one failure or limitation every session. No student should leave believing AI is reliable by default.
04	The student makes the decisions. The trainer demonstrates, scaffolds and asks questions.
05	Every factual claim gets checked. At 5–6 against one source; at 7–12 against a source set.
06	No personal data goes into any model, at any level, for any reason.
07	No account is created outside the age and consent policy in section 08. If a student is too young for a tool, the trainer hosts it or the tool is not used.
08	Every build is tested by someone who is not the builder, in silence, without help, from the published link.
09	Session 07 does not end until every link in the class Sheet opens on a phone. A link that does not open is fixed in the room.
10	Nobody is taught to code. At 10–12, anything the student cannot explain is deleted and regenerated.
11	Depth must serve the project. Do not add advanced concepts because they sound impressive.
12	Close every session on the same four questions: what did I build, what failed, what changed, what next.

DIAGNOSTIC QUESTIONS WHILE CIRCULATING

Asked during the build block, not at the end. The purpose is to find out whether the student owns the work while there is still time to fix it.

- What is this supposed to do, and for whom?
- What did you feed it?
- Where did that fact come from?
- How would you know if it were wrong?
- What did you change, and what happened?
- Has anyone else tried it yet, on their own device?
- What would break this?

WHAT TRANSFERS WHEN THE TOOL CHANGES

AI tools change faster than any syllabus can be reprinted. The programme is written against workflows, so a tool substitution costs a slide, not a session.

WORKFLOW	WHAT THE STUDENT KEEPS
DIRECT	Specifying a task precisely enough that any competent system could execute it.

WORKFLOW	WHAT THE STUDENT KEEPS
GROUND	Insisting an answer is traceable to a source before it is used.
TEST	Building the cases in advance and scoring against them.
ITERATE	Changing one variable at a time and recording what actually moved.
SHIP	Putting the work somewhere another person can reach it, and checking that they can.
EXPLAIN	Owning the output: what it does, what it used, where it breaks, what happens next.

FINAL CURRICULUM PRINCIPLE

Tools will change. The journey will not. This course therefore prioritises understanding, experimentation, creativity, data literacy, responsible use, project building, publishing the work, and the ability to explain your own.