

ZOVELY

K–12 Coding, AI & Computer Science

MASTER CURRICULUM

A parent-centric, project-first pathway from first instructions to building responsible AI products.

LEARN TODAY	CREATE TOMORROW
Think → Build → Explain → Test → Improve	Every learner leaves with something they built.

THINK Understand the problem before writing code.	BUILD Learn through stories, games, websites, apps and AI.	DEBUG Test, fail safely and improve.	SHOW Explain the work and build a portfolio.
---	--	--	--

Our North Star

“Every learner leaves Zovely with something they built, something they can explain, and a stronger ability to solve problems independently.”

12 progressive pathways • project-first learning • parent-visible outcomes • responsible AI from the beginning

The Zovely K–12 Pathway

The pathway moves from computational thinking to coding, web, Python, AI, ML, generative AI, engineering and innovation.

LEVEL	GRADES	PROGRAM	CORE STACK	SIGNATURE BUILD
01	KG–Grade 1	Code Explorers	ScratchJr + unplugged computing	Interactive story/game
02	Grades 2–3	Scratch Creators	Scratch	Playable game + story
03	Grades 4–5	Game & Web Creators	Scratch Advanced + HTML/CSS + JavaScript preview	Interactive website
04	Grades 5–6	Web Builders	HTML, CSS, JavaScript	Working web app
05	Grades 6–7	Full-Stack & App Builders	JavaScript + Git + APIs + beginner backend concepts	Full application MVP
06	Grades 7–8	Python Problem Solvers	Python	Python problem-solving app
07	Grades 8–9	AI Explorers	Python + Teachable Machine / beginner AI tools	Trained AI prototype
08	Grades 9–10	Machine Learning Builders	Python + NumPy/Pandas + scikit-learn	Evaluated ML model
09	Grades 10–11	Generative AI & LLM Builders	Python + LLM APIs + prompt engineering + embeddings concepts	AI knowledge assistant
10	Grades 11–12	AI App Engineers	Python + APIs + databases + RAG + deployment concepts	Deployable AI app
11	Grades 11–12	Data Science & Applied ML	Python + SQL + Pandas + scikit-learn + BI	Data science case study
12	Grades 11–12 / advanced students	AI Innovator — Grand Capstone	Student-selected stack	End-to-end AI capstone

FOUNDATION KG–5 Logic • creativity • block coding • web	BUILDERS 5–8 Web • apps • Python • data	AI CREATORS 8–12+ AI • ML • LLMs • RAG • agents
--	--	--

*Same learning habit at every age: **Imagine** → **Learn** → **Build** → **Debug** → **Share**.*

What Parents Should See

Zovely should make progress visible. Parents should always know what their child is learning, what they built, what they can now do and what comes next.

BEFORE DAY 1

Diagnostic + interests + goals → recommended level

EVERY CLASS

Concept → guided build → challenge → demo/reflection

EVERY 4 WEEKS

Skill rubric + project evidence + teacher note

MID-COURSE

Mini showcase + growth conversation

END OF COURSE

Capstone + competency rubric + portfolio artifact

NEXT STEP

Clear recommendation for the learner's next pathway

THE ZOVELY CLASS LOOP

1 • RECALL

2 • LEARN

3 • BUILD

4 • CHALLENGE

5 • SHOW

Parent question answered every month: “What can my child do now that they could not do before?”

Recommended delivery: 2 live sessions/week; younger learners use shorter instruction and more guided creation, while older learners get increasing independent build, debugging, documentation and presentation time.

01 | Code Explorers

GRADES KG–Grade 1	CORE TOOLS ScratchJr + unplugged computing	END RESULT Interactive story/game
-----------------------------	--	---

Parent promise: Children learn to give instructions, sequence ideas and create their first interactive stories and animations.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	My First Instructions	What is a computer? commands, sequence, cause/effect, unplugged robot game.	
2	Characters & Scenes	ScratchJr interface, sprites, backgrounds, movement blocks	animated greeting card.
	Events & Actions	Start-on-tap, start-on-message, simple triggers	interactive “When I tap...” story.
4	Sequences & Loops	Repeat patterns, rhythm, simple loops	dancing character loop.
	Stories with Logic	Beginning/middle/end, dialogue, timing, scene changes	3-scene digital story.
6	Sounds & Creativity	Voice/sound effects, visual storytelling and design choices	story with narration/sound.
	Problem Solving	Debugging as “find and fix,” testing one change at a time	fix-the-broken-project challenge.
8	Mini Game 1	Goals, scoring concepts, movement rules	Catch-the-Star mini game.
	Mini Game 2	Challenge, feedback, levels, player choices	maze/avoid-the-obstacle game.
10	Design Thinking	Choose a problem, draw a storyboard, build a prototype	My Helpful Invention.
	Polish & Practice	Test with a friend, improve clarity, rehearse explanation	portfolio polish.
12	Demo Day	Present, explain, celebrate	child presents one project and explains the logic.

BY THE END Child can explain a sequence/algorithm, debug with guidance, and confidently show a finished interactive creation.	PARENT CAN SEE Project evidence • learner demo • skill rubric • reflection
---	--

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

02 | Scratch Creators

GRADES

Grades 2–3

CORE TOOLS

Scratch

END RESULT

Playable game + story

Parent promise: Students move from block play to real computational thinking: variables, loops, conditionals, events and game design.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	Coding Foundations	Algorithms, sequences, events, sprites and stages	interactive animation.
2	Motion & Timing	Coordinates, movement, timing and scene control	animated character challenge.
	Loops	Repeat, forever, nested patterns and efficiency	pattern/animation generator.
4	Conditions	If/then, sensing, rules and game decisions	reaction game.
	Variables	Score, lives, timer, counters	score-based game.
6	Broadcast & Messages	Events, multi-scene architecture, modular thinking	choose-your-path story.
	Clones & Advanced Motion	Cloning, randomness, variables and difficulty	endless runner prototype.
8	Game Design	Rules, win/lose states, feedback, player experience	complete mini-game.
	Debugging	Read blocks, isolate bugs, test edge cases	debugging tournament.
10	Creative Computing	Design around a real-world theme	eco/space/health project.
	Project Studio	Iteration, peer feedback, documentation	final project build.
12	Demo Day	Explain algorithm, choices, challenges and improvements	playable portfolio project.

BY THE END

Student can independently break a problem into steps, debug with support and explain how the project works.

PARENT CAN SEE

Project evidence • learner demo • skill rubric • reflection

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

03 | Game & Web Creators

GRADES

Grades 4–5

CORE TOOLS

Scratch Advanced + HTML/CSS + JavaScript preview

END RESULT

Interactive website

Parent promise: Students graduate from block coding into text-based web thinking while keeping projects fun and visual.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	From Blocks to Code	How block concepts map to text code; algorithms and decomposition	pseudocode.
2	Advanced Scratch	Lists, clones, custom blocks, state and game architecture	advanced game prototype.
	Game Mechanics	Physics-like motion, collisions, levels, difficulty curves	multi-level game.
4	Web Thinking	What websites are; HTML structure, tags and links	personal profile page.
	CSS Design	Selectors, colors, spacing, typography, layout	styled portfolio page.
6	Responsive Design	Cards, sections, mobile-first thinking	responsive landing page.
	JavaScript Basics	Variables, conditions, functions and events	interactive web page.
8	DOM & Interaction	Buttons, inputs, changing page content	quiz web app.
	Data & APIs Concept	What data is; JSON; what an API does	API-powered concept demo.
10	Build for a User	User persona, problem statement, wireframe and prototype	useful website.
	Build & Test	Debugging, usability testing, accessibility basics	release candidate.
12	Demo Day	Live demo + 2-minute product explanation	published-style portfolio project.

BY THE END

Student can move from blocks to text concepts and build a portfolio-ready interactive website.

PARENT CAN SEE

Project evidence • learner demo • skill rubric • reflection

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

04 | Web Builders

GRADES Grades 5–6	CORE TOOLS HTML, CSS, JavaScript	END RESULT Working web app
---------------------------------	--	--

Parent promise: Students build real websites and interactive browser applications while learning the fundamentals behind the modern web.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	How the Web Works	Browser, URL, client/server concept	multi-section webpage.
2	HTML Foundations	Semantic tags, links, images, lists, forms	personal website.
	CSS Foundations	Selectors, box model, typography, spacing	designed landing page.
4	Layouts	Flexbox, grids, responsive design	responsive product page.
	JavaScript Foundations	Variables, data types, operators, conditions	interactive calculator.
6	Functions & Events	Functions, event listeners, reusable logic	interactive quiz.
	Arrays & Objects	Structured data, loops and common patterns	student directory/data viewer.
8	DOM Programming	Read/write page elements, forms and validation	registration/form app.
	APIs & JSON	Fetch concept, JSON data, API use and error states	live data mini-app.
10	Git & Developer Habits	Version control concept, commits, README, naming and debugging.	
	Product Sprint	User story, wireframe, build, test, iterate	final web application.
12	Launch Day	Demo, code walkthrough, reflection, portfolio packaging	working website + README.

BY THE END Student can independently build and debug a real browser app and explain HTML, CSS, JavaScript, data and APIs.	PARENT CAN SEE Project evidence • learner demo • skill rubric • reflection
---	--

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

05 | Full-Stack & App Builders

GRADES

Grades 6–7

CORE TOOLS

JavaScript + Git + APIs + beginner backend concepts

END RESULT

Full application MVP

Parent promise: Students learn how real applications connect interfaces, logic, data and services.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	Application Architecture	Frontend/backend/database/API mental model	architecture map.
2	Modern JavaScript	Functions, arrays, objects, modules, async thinking	utility app.
	APIs	HTTP, requests, responses, JSON, status codes	API-powered app.
4	Forms & Validation	Input validation, error handling, UX feedback	validated workflow.
	Data Storage	Local storage and database concepts	saved-data application.
6	Backend Basics	Server routes, request/response, simple service architecture	tiny backend/API.
	Authentication Concepts	Accounts, passwords, sessions, privacy, safe handling	mock login flow.
8	Git & Collaboration	Branches concept, pull requests, code review, teamwork	team repository workflow.
	Testing & Debugging	Unit/integration concepts, logs, reproducible bugs	test checklist + fixes.
10	AI Feature Preview	Safe AI API conceptually; prompts, inputs, outputs, limits	AI feature prototype.
	Product Sprint	Build a real user-facing application in teams	MVP release candidate.
12	Demo Day	Product demo, architecture explanation, teamwork reflection	portfolio-ready app.

BY THE END

Student can break down a real application, connect services/data, collaborate and ship a small MVP.

PARENT CAN SEE

Project evidence • learner demo • skill rubric • reflection

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

06 | Python Problem Solvers

GRADES Grades 7–8	CORE TOOLS Python	END RESULT Python problem-solving app
-----------------------------	-----------------------------	---

Parent promise: Students become comfortable with text-based programming, automation and data-driven problem solving.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	Python & Computational Thinking	Variables, input/output, types, algorithms	interactive calculator.
2	Conditions	Boolean logic, if/elif/else, validation	decision-making game.
	Loops	for/while, ranges, patterns, debugging loops	pattern generator.
4	Functions	Parameters, return values, reusable code	math/utility toolkit.
	Strings & Collections	Strings, lists, tuples, dictionaries, sets	text/data organizer.
6	Files & Data	CSV/JSON concepts, reading/writing files	student data analyzer.
	Errors & Testing	Exceptions, assertions, test cases and debugging habits	bug-fix challenge.
8	Object-Oriented Thinking	Objects, classes, attributes and methods	simple simulation.
	Libraries & APIs	Importing libraries, using an API safely, documentation	info app concept.
10	Data Analysis Intro	Pandas concept, tables, cleaning and basic charts	mini data story.
	Project Studio	Real problem, plan, code, test, iterate	Python capstone.
12	Demo Day	Code walkthrough, results, reflection	GitHub-style portfolio project.

BY THE END

Student can write, debug and explain text-based programs while applying coding to real problems.

PARENT CAN SEE

Project evidence • learner demo • skill rubric • reflection

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

07 | AI Explorers

GRADES

Grades 8–9

CORE TOOLS

Python + Teachable Machine / beginner AI tools

END RESULT

Trained AI prototype

Parent promise: Students understand what AI actually is, train simple models and learn to use generative AI critically and safely.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	What Is AI?	AI vs automation vs ML; examples and limitations	AI myth-vs-fact.
2	AI Perception	Images, sound, sensors and classification	image/sound classifier experiment.
	Data Makes AI	Features, labels, training/test data, bias and quality	dataset design challenge.
4	Machine Learning Loop	Train → validate → test → improve	simple model experiment.
	Classification	Classes, probabilities, confusion matrix intuition	classifier.
6	Regression	Predicting numbers, relationships and error intuition	prediction experiment.
	Natural Language	Text data, tokens concept, sentiment/classification	text classifier demo.
8	Generative AI	LLMs, prompts, context, hallucinations, verification	prompt lab.
	Responsible AI	Privacy, bias, copyright, deepfakes, human oversight	AI ethics case debate.
10	AI Product Design	Problem framing, users, data, model, interface, success metric	blueprint.
	Build & Iterate	Create a small AI project and test with peers	working prototype.
12	AI Demo Day	Explain data, model, limitations, ethics and impact	AI portfolio project.

BY THE END

Student can explain the AI workflow, build a simple AI prototype and identify model limitations and responsible-use issues.

PARENT CAN SEE

Project evidence • learner demo • skill rubric • reflection

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

08 | Machine Learning Builders

GRADES Grades 9–10	CORE TOOLS Python + NumPy/Pandas + scikit-learn	END RESULT Evaluated ML model
----------------------------------	---	---

Parent promise: Students move from “using AI” to understanding and building classical machine-learning systems.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	ML Foundations	Supervised/unsupervised learning, features, labels, workflow	end-to-end ML map.
2	Data Collection & Cleaning	Missing values, duplicates, types, outliers, leakage	clean dataset.
	Exploratory Data Analysis	Statistics intuition, distributions, correlations, visualization	EDA notebook.
4	Regression	Linear regression, train/test split, MAE/MSE/R² intuition	predictor.
	Classification	Logistic regression/tree-based classification	binary classifier.
6	Model Evaluation	Confusion matrix, precision, recall, F1, ROC-AUC concept	compare models.
	Decision Trees & Ensembles	Trees, overfitting, random forests, feature importance	improvement lab.
8	Unsupervised Learning	Clustering, similarity, dimensionality reduction concept	grouping project.
	Feature Engineering	Encoding, scaling, feature selection, pipelines concept	improve model.
10	Bias, Fairness & Explainability	Dataset bias, model limitations	model audit report.
	ML Capstone	Problem framing → data → model → evaluation → presentation.	
12	Model Review Day	Technical defense, business impact and limitations	portfolio notebook + report.

BY THE END Student can build, evaluate and critique an ML model rather than treating predictions as automatically correct.	PARENT CAN SEE Project evidence • learner demo • skill rubric • reflection
--	--

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

09 | Generative AI & LLM Builders

GRADES Grades 10–11	CORE TOOLS Python + LLM APIs + prompt engineering + embeddings concepts	END RESULT AI knowledge assistant
-----------------------------------	---	---

Parent promise: Students learn how modern generative AI works at a practical level and build useful AI experiences—not just prompt chatbots.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	Inside Generative AI	Tokens, context, neural networks at a high level, strengths/limits	visual.
2	Prompt Engineering	Roles, context, constraints, examples, structured outputs	prompt benchmark.
	AI Reliability	Hallucinations, verification, citations, uncertainty, evaluation	fact-checking pipeline.
4	Multimodal AI	Text, image, audio, multimodal workflows	multimodal prototype.
	Embeddings	Semantic similarity, vectors, retrieval intuition	similarity search demo.
6	RAG	Documents → chunks → embeddings → retrieval → answer	knowledge-base assistant.
	Tool Use	Function/tool calling, schemas, safe boundaries, permissions	tool-using assistant.
8	Agents	Planning, tools, memory, loops, failure modes, guardrails	agent workflow.
	AI UX	Human-in-the-loop design, feedback, transparency, error recovery	AI interface.
10	Evaluation	Task metrics, test sets, rubric-based evaluation, red teaming	evaluation suite.
	Capstone Build	Design, build, test and document an AI application.	
12	AI Product Demo Day	Demo + architecture + evaluation + ethics + business/user value.	

BY THE END Student can build with generative AI beyond prompting, ground outputs in data and evaluate reliability and safety.	PARENT CAN SEE Project evidence • learner demo • skill rubric • reflection
---	--

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

10 | AI App Engineers

GRADES

Grades 11–12

CORE TOOLS

Python + APIs + databases + RAG + deployment concepts

END RESULT

Deployable AI app

Parent promise: Students learn the engineering workflow required to turn an AI idea into a usable application.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	AI Engineering Foundations	System architecture, model/API choices, latency, cost, reliability.	
2	Backend with Python	API concepts, endpoints, validation, error handling	AI API service.
	Databases	Tables, schemas, SQL basics, CRUD, data modeling	AI app database.
4	RAG Engineering	Ingestion, chunking, retrieval, grounding, source display	document Q&A.;
	Tool-Calling Systems	Tools, schemas, permissions, failure handling	assistant with tools.
6	Agent Workflows	Planner/executor patterns, state, memory, stopping conditions	task agent.
	Security & Privacy	Secrets, prompt injection, data minimization, permissions, safe outputs.	
8	Evaluation & Observability	Logs, traces, test cases, regression tests, quality metrics.	
	Deployment Concepts	Environment variables, hosting, APIs, monitoring, cost controls.	
10	Product & Growth	User journey, activation, feedback, analytics	AI product metrics plan.
	Capstone Sprint	Build, test, deploy, document, collect user feedback	beta.
12	Engineering Defense	Architecture review, live demo, trade-offs, roadmap.	

BY THE END

Student can design an AI system, connect backend/data components, test it and defend engineering trade-offs.

PARENT CAN SEE

Project evidence • learner demo • skill rubric • reflection

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

11 | Data Science & Applied ML

GRADES Grades 11–12	CORE TOOLS Python + SQL + Pandas + scikit-learn + BI	END RESULT Data science case study
-----------------------------------	--	--

Parent promise: Students learn to turn messy data into decisions, predictions and clear business stories.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	Data Science Workflow	Question → data → analysis → model → decision	problem statement.
2	SQL Foundations	SELECT, filters, joins, aggregation, subqueries	business SQL challenge.
	Advanced SQL	CTEs, window functions, analytical queries	cohort/retention query.
4	Data Cleaning	Quality checks, missingness, duplicates, reproducible cleaning	pipeline.
	EDA & Statistics	Distributions, sampling, correlation vs causation, hypothesis intuition.	
6	Visualization & Storytelling	Chart selection, dashboards, narrative, executive summaries.	
	Predictive Modeling	Regression/classification workflow and baseline models	prediction model.
8	Model Evaluation	Metrics, cross-validation, overfitting, model comparison	evaluation report.
	Segmentation & Retention	Clustering, customer behavior, churn, interventions	segmentation.
10	Responsible Data Science	Privacy, bias, leakage, fairness, explainability	audit.
	Capstone	Real-world dataset, analysis, model/dashboard, recommendations.	
12	Data Science Showcase	Present insights to a non-technical audience	portfolio case study.

BY THE END Student can query, clean, analyze and visualize data, build a predictive model and turn results into recommendations.	PARENT CAN SEE Project evidence • learner demo • skill rubric • reflection
--	--

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

12 | AI Innovator — Grand Capstone

GRADES Grades 11–12 / advanced students	CORE TOOLS Student-selected stack	END RESULT End-to-end AI capstone
---	---	---

Parent promise: A startup-style experience where students identify a real problem, build an AI solution, validate it with users and present it professionally.

12-WEEK LEARNING MAP • Every week ends with visible evidence.

WEEK	MODULE	WHAT WE COVER	WHAT YOUR CHILD BUILDS / SHOWS
	Problem Discovery	Interview-style discovery, pain points, user personas, validation	problem brief.
2	AI Feasibility	Data availability, risks, alternatives, success metrics	feasibility memo.
	System Design	Architecture, model/tool choices, data flow, privacy boundaries	system design.
4	Prototype	Rapid prototype, UI flow and first working model/API.	
	Data & Evaluation Plan	Dataset, labels, test set, metrics, acceptance criteria.	
6	Build Sprint 1	Core feature development with version control and documentation	alpha build.
	Build Sprint 2	RAG/ML/agent integration as appropriate; improve reliability	beta build.
8	Safety & Ethics Review	Bias, privacy, misuse, security, human oversight, failure modes.	
	User Testing	Structured user tests, feedback capture, iteration	testing report.
10	Productization	UX polish, onboarding, analytics, documentation, demo script	release candidate.
	Pitch & Portfolio	Problem, solution, evidence, architecture, impact, business model, roadmap.	
12	Zovely AI Demo Day	Live demo, technical defense, Q&A;, reflection	capstone shipped.

BY THE END Student can take a real problem from discovery to responsible AI prototype, user testing, portfolio and professional presentation.	PARENT CAN SEE Project evidence • learner demo • skill rubric • reflection
---	--

ZOVELY LEARNING CYCLE • LEARN TODAY → CREATE TOMORROW

Zovely Assessment Standard

The goal is not memorization. Assessment shows whether a learner can transfer knowledge into a new problem.

DIMENSION	KG–5	6–8	9–12
Computational thinking	Sequencing, patterns, decomposition	Algorithms, abstraction, debugging	System design, trade-offs, optimization
Programming	Blocks → simple text	Python/web/app coding	Production-style code, APIs, data and AI
Projects	Stories, games, simple apps	Useful applications and AI prototypes	End-to-end systems and capstones
Data / AI	AI awareness and simple classifiers	Data, models, bias and evaluation	ML/LLMs, RAG, agents, evaluation, safety
Communication	Explain what I made	Explain how it works	Defend technical and product decisions
Responsible computing	Digital citizenship	Privacy, bias, safe AI use	Ethics-by-design, security, governance

100-POINT COMPETENCY RUBRIC

30 Technical understanding • **25** Project execution • **15** Problem solving/debugging • **10** Testing/evaluation • **10** Communication • **10** Responsible practice

Progress bands: **Emerging** → **Developing** → **Independent** → **Creator**. Certificates should represent demonstrated competency, not attendance.

AI SAFETY & RESPONSIBLE USE

Privacy • verification of AI outputs • bias & fairness • copyright • deepfakes • human oversight • age-appropriate tools • safe sharing • data minimization

Zovely measures both what students know and what students can make with what they know.

What Makes Zovely Different

The curriculum becomes powerful when the delivery system makes learning personal, visible and consistent.

01 • ZOVELY DIAGNOSTIC

Place students by demonstrated readiness, not only by grade.

02 • SKILL PASSPORT

Track coding, AI, data, communication and responsible computing.

03 • PROJECT PORTFOLIO

Every course adds a meaningful artifact a student can show.

04 • DEMO DAYS

Students explain their work to parents and peers.

05 • TEACHER QUALITY

Standard plans, mock classes, observations and rubrics.

06 • REAL-WORLD BRIEFS

Climate, education, accessibility, games, community and product problems.

07 • AI EVALUATION

Older learners assess reliability, safety, grounding and usefulness.

08 • CAREER EXPOSURE

Show pathways across software, data, ML, AI product and research.

09 • NO CERTIFICATE INFLATION

Credentials follow demonstrated skills and portfolio evidence.

ZOVELY NORTH STAR

“Every learner leaves Zovely with something they built, something they can explain, and a stronger ability to solve problems independently.”

Implementation Notes for Zovely

The strongest launch is not the one with the most courses. It is the one where a small number of courses are delivered exceptionally well and produce measurable student outcomes.

PHASE 1

Launch 3–4 flagship tracks; perfect teaching playbooks, diagnostics, parent reports and project rubrics.

PHASE 2

Add the middle-school progression, Python and ML once educator quality and operations are stable.

PHASE 3

Scale advanced AI engineering, data science and Grand Capstone with specialist mentors.

THE ZOVELY PRODUCT IS THE OUTCOME

Do not sell only “coding hours.” Sell **visible progress**: a child who can think computationally, build projects, explain their work, debug independently and eventually create responsible AI systems.

LEARN → BUILD → DEBUG → EXPLAIN → CREATE

Zovely • Learn Today. Create Tomorrow.