

3.7 AI-based Story Writer

Duration: 50-55 minutes | Slide deck aligned: Slides 1-20

Students will be able to:

- Generate a short creative story with ChatGPT based on user-defined parameters, recognizing text generation as a use case beyond conversation.
- Build a user-friendly application interface using label, textbox, button, and layout row widgets instead of a chat window.
- Compose dynamic ChatGPT requests by combining fixed instructions with user input and displaying the response in an output textbox.
- Improve the Story Writer through prompt design, UI polish, peer review, and creative adaptation.

Learning Activity Summary

- Warm-up/project launch: introduce the AI Story Writer goal and discuss what makes a story feel original using Slides 1-4 (5 minutes).
- Guided walkthrough: create the project, add core widgets, arrange layout rows, add the output textbox, and plan the ChatGPT request flow using Slides 5-13 (30 minutes).
- Prompt lab and complete build: revise a boring prompt, build the final button-to-ChatGPT flow, and test the app using Slides 14-15 (10 minutes).
- Creative adaptation, peer review, submission checklist, discussion, and exit ticket using Slides 16-20 (10 minutes).

Student Materials

- Computer with Internet access
- CreatiCode account and Playground access
- Lesson 3.7 slide deck
- The "An AI-based Story Writer" tutorial
- Optional prompt-planning or peer-review sheet

Teacher Preparation

- Project slides projected and CreatiCode Playground open
- Completed reference project ready for demonstration
- Student accounts/logins checked before class
- Widget wiki pages ready: label, button, textbox, layout rows
- Sample story topics and prompt constraints prepared for testing
- Submission checklist/rubric visible for the final self-check

Teacher Notes / Differentiation

Support: Provide a block-by-block checklist, prompt sentence starters, and partner support for students who need help with widget names, layout rows, or request composition.

Extension: Ask advanced students to add genre/tone/audience controls, a Clear button, a Generating... status message, streaming output, or a new creative-writing app.

Pacing: Build the interface first, test each layout row, and save deeper prompt/design choices until the required Go button flow works.

AI Use / Safety: Remind students not to enter private information, to keep topics school-appropriate, and to review AI-generated stories before sharing.

Teacher tip: Keep vocabulary consistent: label1, textbox1, button1, textbox2, layout row, request variable, story/output variable, waiting mode, and temperature.

Key for this guide

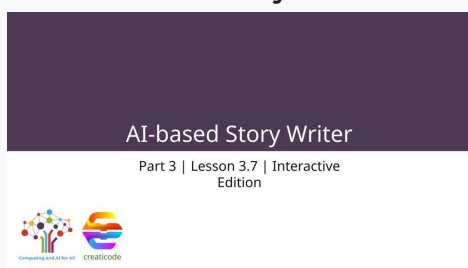
🗒 = explicitly explain

✚ = student action

! = teacher tip

WARM-UP / PROJECT LAUNCH (5 MINUTES)

Slide 1 AI-based Story Writer



Lesson Launch

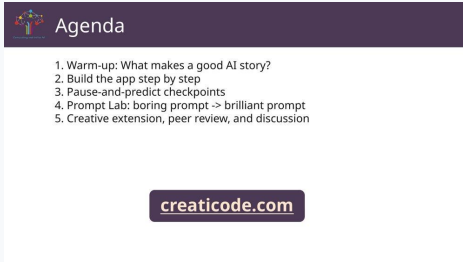
Slide shows: Title slide with the lesson name, Part 3 / Lesson 3.7, and CreatiCode branding.

🗒 Today students will build an AI-based Story Writer app that takes a topic and returns a short AI-generated story.

✚ Students open CreatiCode, confirm they can access the Playground, and get ready to create a new project.

! Connect this lesson to prior AI app lessons: the focus is text generation through a custom interface, not a chat window.

Slide 2 Agenda



Agenda

1. Warm-up: What makes a good AI story?
2. Build the app step by step
3. Pause-and-predict checkpoints
4. Prompt Lab: boring prompt -> brilliant prompt
5. Creative extension, peer review, and discussion

creaticode.com

Agenda and Lesson Flow

Slide shows: The lesson agenda: warm-up, step-by-step build, pause-and-predict checkpoints, Prompt Lab, creative extension, peer review, and discussion.

- 👤 The class will move from story ideas to interface building, then to prompt improvement and peer testing.
- + Students preview the phases and listen for where the work shifts from coding to prompt/design decisions.
- ! Set expectations that the final app should be tested by a peer, not only by its creator.

Slide 3

From MBTI App to AI Story Writer



From MBTI App to AI Story Writer

The AI-based Story Writer

- User enters a topic
- ChatGPT generates a short story

Learning Goals

- Use AI to generate text rather than chat
- Use "layout row" to position widgets

Think: What makes an AI story feel original instead of generic?

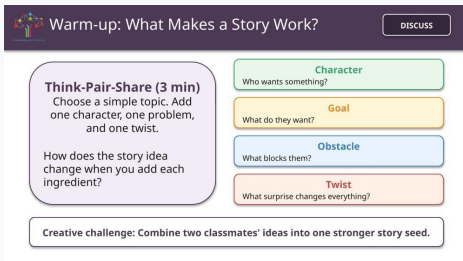
Project Overview and Learning Goals

Slide shows: The AI Story Writer goal: the user enters a topic, ChatGPT generates a short story, and students use layout rows to position widgets.

- 👤 Unlike a chat app, this app composes one request from the user topic and displays the returned story in an output box.
- + Students name the input, the AI processing step, and the output they will build.
- ! Use the question about originality to introduce prompt constraints such as genre, tone, character, conflict, and twist.

Slide 4

Warm-up: What Makes a Story Work?



Warm-up: What Makes a Story Work?

DISCUSS

Think-Pair-Share (3 min)
Choose a simple topic. Add one character, one problem, and one twist.
How does the story idea change when you add each ingredient?

Character
Who wants something?

Goal
What do they want?

Obstacle
What blocks them?

Twist
What surprise changes everything?

Creative challenge: Combine two classmates' ideas into one stronger story seed.

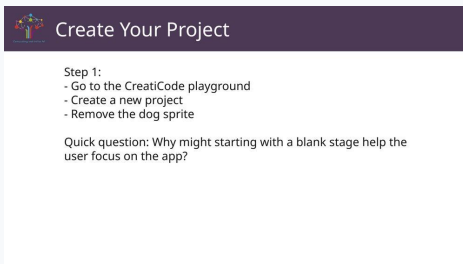
Warm-Up Think-Pair-Share

Slide shows: A 3-minute story seed activity using character, goal, obstacle, and twist, plus a creative challenge to combine ideas.

- 👤 AI stories improve when the prompt gives story ingredients instead of only a topic.
- + Students choose a topic, add one character/problem/twist, and share or combine ideas with a partner.
- ! Capture two or three strong story seeds to reuse later when students test their apps.

GUIDED WALKTHROUGH (30 MINUTES)

Slide 5 Create Your Project



Create Your Project

Step 1:

- Go to the CreatiCode playground
- Create a new project
- Remove the dog sprite

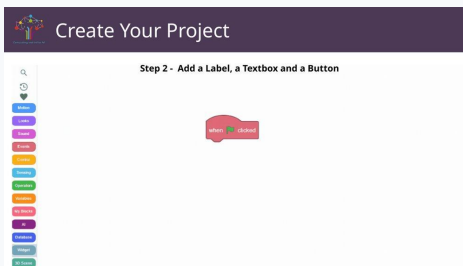
Quick question: Why might starting with a blank stage help the user focus on the app?

Step 1 - Start with a Blank Project

Slide shows: Project setup instructions: go to the CreatiCode playground, create a new project, and remove the dog sprite.

- 👤 A blank stage helps the user focus on the app interface instead of sprites on the stage.
- + Students create a new project, remove the default dog sprite, and confirm the stage is ready for widgets.
- ! Check logins and project setup early so students do not fall behind before widget work begins.

Slide 6 Create Your Project



Create Your Project

Step 2 - Add a Label, a Textbox and a Button

Step 2 - Add the Core Widgets

Slide shows: Instructions to add a label, a textbox, and a button.

- 👤 The top row needs a label that tells the user what to type, a textbox for the topic, and a Go button to trigger the app.
- + Students add label1, textbox1, and button1, keeping widget names consistent for later blocks.
- ! Avoid deep customization at this point; first make sure every student has the same core widgets.

Slide 7 Create Your Project

Create Your Project

Step 3 - Place Widgets using "Layout Row"

- Imagine dividing the stage into invisible rows
- Each row is further divided into cells
- Place widgets into those cells
- learn more about layout rows >>

Creative analogy: Is the stage more like a grid, a table, or a map? Defend your analogy.

Example: 2 rows, with 2 and 3 cells

Step 3A - Understand Layout Rows

Slide shows: Layout rows divide the stage into invisible rows and cells, with an example of 2 rows containing 2 and 3 cells.

👤 A layout row is a UI organizer: it places widgets by row/cell percentages rather than exact x/y coordinates.

+ Students sketch a row-and-cell plan and decide whether the stage feels more like a grid, table, or map.

! Pause on percentage thinking before students copy blocks; many layout bugs come from unclear row/cell proportions.

Slide 8 Checkpoint: Layout Row Puzzle

Checkpoint: Layout Row Puzzle

Before the answer slide...

Sketch the stage as rows and cells.

1. How tall should the input row be?
2. How many cells should it have?
3. Which widget needs the most space?

Vote, then defend your layout.

Class vote: 15/85, 25/75, or your own ratio?

Pause-and-Predict - Plan the Layout

Slide shows: A prediction puzzle asking how tall the input row should be, how many cells it needs, and which widget needs the most space.

👤 Before showing the answer, students should reason about how much space each part of the interface needs.

+ Students sketch the input row and story output row, vote on a layout ratio, and defend their choice.

! Wrong guesses are useful here; use them to surface why the story output needs most of the screen.

Slide 9 Create Your Project

Create Your Project

Step 3 - Place Widgets using "Layout Row"

- Top row height: 15%
- Cell widths: 35%, 50%, and 15%
- Widget names: label1, textbox1, button1

Predict: What might happen if the cell widths do not add up to 100%?

apply layout row named r1 height 15 % cell width 35 % 50 % 15 %

widgets label1 textbox1 button1 margin 2 padding 5

Step 3B - Build the Top Layout Row

Slide shows: The answer layout: top row height 15%, cell widths 35%, 50%, and 15%, and widget names label1, textbox1, button1.

👤 The top 15% of the stage is divided among the label, topic input, and Go button.

+ Students add or configure the layout row for label1, textbox1, and button1, then run the project to check placement.

! If widgets overlap or disappear, inspect row height, cell widths, and exact widget names first.

Slide 10 Create Your Project

Create Your Project

Step 4 - Add another textbox to show the story

- Multiple rows
- Name is "textbox2" (used for layout)

Discuss: Why should the input box and output box be separate?

add textbox at x width 200 height 50 padding 5

label multiple scroll scroll mode read only

textbox2

Step 4 - Add the Story Output Textbox

Slide shows: Instructions to add another textbox, use multiple rows, and name the output textbox textbox2.

👤 The input textbox collects the topic, while textbox2 displays the generated story and needs multiline output space.

+ Students add textbox2 and keep it separate from the topic input box.

! Ask why separating input and output makes the app easier for a user to understand.

Slide 11 Create Your Project

Create Your Project

Step 5 - Place textbox2

- Use the second layout row
- Try it yourself first

3-minute pair challenge: place textbox2 before checking the answer.

Step 5A - Place textbox2 Yourself

Slide shows: A 3-minute pair challenge to place textbox2 using the second layout row before checking the answer.

👤 Students now apply the layout-row idea without immediately copying the final configuration.

+ Pairs place textbox2 in the second row, test the result, and compare strategies.

! Use a short timer so the challenge stays productive and does not take over the guided build.

Slide 12 Create Your Project

Step 5B - Confirm the Output Row

Slide shows: The answer layout for textbox2: row height 85% and one cell at 100%, plus a debugging question.

- 👤 The story output should take most of the stage because the generated story is longer than the topic input.
- + Students set the second row to 85% height and 100% width, then run the project to check the output area.
- ! If textbox2 is tiny, inspect the 15/85 split and the widget name before changing other code.

Slide 13 Create Your Project

Step 6A - Plan the ChatGPT Request Flow

Slide shows: The Go button flow: compose a request from the topic input, send the request to ChatGPT, and show the response in textbox2.

- 👤 The button event connects the UI to AI: read the topic, build a prompt, wait for ChatGPT, then write the response.
- + Students identify the blocks that read textbox1, send the request, and set textbox2.
- ! Discuss empty input as an extension: the app could ask the user to enter a topic before sending a request.

PROMPT LAB / COMPLETE BUILD (10 MINUTES)

Slide 14 Prompt Lab: Boring -> Brilliant

Prompt Lab - Make the Request Specific

Slide shows: A basic prompt compared with a richer 100-word mystery prompt, followed by a discussion and rewrite challenge.

- 👤 Prompt design changes story quality; constraints like genre, tone, vivid details, audience, and twist give the AI creative direction.
- + Students rewrite the basic prompt with one added constraint and predict how the story will change.
- ! Keep one strong student prompt visible so students can adapt it in the final code.

Slide 15 Create Your Project

Step 6B - Build and Test the Full Request

Slide shows: The completed button-click script that composes the request, sends it to ChatGPT, and shows the story in textbox2.

- 👤 The final script should dynamically combine the fixed instruction with value of textbox1, store the response, and set textbox2.
- + Students implement the button-click flow and test the topic "a robot chef afraid of soup."
- ! Use consistent model settings during the build: waiting mode, about 200 tokens, and temperature about 0.8 for creative but controlled output.

CREATIVE ADAPTATION / PEER REVIEW (10 MINUTES)

Slide 16

Design Debate: How Should the App Feel?

Design Debate: How Should the App Feel? CHOOSE

Pick ONE design priority. What feature would you build first?

- Fast & simple**
Fewer choices, quick story.
- Personalized**
Ask genre, tone, age, setting.
- Beautiful UI**
Colors, font, spacing, clear labels.
- Surprise reveal**
Stream the story piece by piece.

Debate: Better prompt or better interface? Defend your choice with a user story.

Design Debate - Choose an App Priority

Slide shows: Four design priorities: fast and simple, personalized, beautiful UI, and surprise reveal, plus a debate prompt.

👤 App design includes both prompt quality and user experience choices.

+ Students pick one design priority and explain which feature they would build first.

! Use this slide to transition from required build steps to thoughtful creative enhancement.

Slide 17

Creative Adaptation

Creative Adaptation

Work in pairs: choose ONE extension and explain why it improves the app.

Creative paths:

- Prompt remix: poem, joke, song, game intro, comic scene
- UI polish: colors, fonts, sizes, spacing, button text
- Story reveal: stream the story chunk by chunk
- Story control: genre, tone, setting, audience, characters, twist, theme

Discussion: Which choice changes the user experience the most: prompt design, UI design, or streaming? Why?

Creative Adaptation Challenge

Slide shows: Pair-work extension paths: prompt remix, UI polish, story reveal, and story controls for genre/tone/setting/audience.

👤 Extensions should improve content quality, interface clarity, or how the story is revealed to the user.

+ Pairs choose one extension and explain why it improves the app before implementing or planning it.

! Ask students to describe the user benefit, not only the technical change.

Slide 18

Submission Checklist + Peer Review

Submission Checklist + Peer Review

Category	Criteria	Max	Score
AI Application	On Go: compose a prompt from the user topic (= 100 words limit), send it to ChatGPT, and display the generated story text (not a chat reply) in the output box.	2	—
Project Development	Build the UI with layout rows: label, input textbox, and button in the top row; output textbox in the second row. Wire the button click → request → response flow so results appear in the output textbox.	1	—
Creative Adaptation	Extend the app creatively (e.g., different content types like poem/joke/song, UI styling, optional streaming reveal, or refined instructions for genre, language, tone, audience, setting, characters, plot, themes).	2	—
Total:		5	—

Submission Checklist / Rubric

Slide shows: A rubric table for AI Application, Project Development, and Creative Adaptation, totaling 5 points.

👤 The project is assessed on a working AI story flow, a clean layout-row interface, and meaningful creative adaptation.

+ Students use the checklist to self-score before peer review and submission.

! Make sure project links are shareable and project names are clear before students submit.

Slide 19

Peer Review: Test Like a User

Peer Review: Test Like a User REVIEW

1. Try it
Use another team's app with a surprising topic.

2. Glow + grow
Name one thing that worked and one thing to improve.

3. Ask "What if..."
Suggest one creative feature they could add next.

Reviewer question: What would make this app more fun, more useful, or more surprising?

Peer Review - Test Like a User

Slide shows: Peer review steps: try another team's app, give one glow and one grow, and ask a "What if..." feature question.

👤 A useful reviewer tests the app with a surprising topic and gives specific feedback about output and user experience.

+ Students test a partner app, name one thing that worked, one improvement, and one creative feature idea.

! Keep feedback focused on clarity, story quality, and user experience rather than only colors or preferences.

Slide 20

Discussion: From Builder to Designer

Discussion: From Builder to Designer

Prompt insight
Which prompt change made the story noticeably better? Why?

Surprise moment
What surprising AI output did you get, and how would you steer it?

Human control
Where should the human stay in control when AI writes?

Next app
What other app could use the same click → prompt → response pattern?

Exit ticket: one feature you would build next

Wrap-Up Discussion and Exit Ticket

Slide shows: Discussion prompts about prompt insight, surprising AI output, human control, next app ideas, and an exit ticket.

👤 The click → prompt → response pattern can support many AI apps beyond stories.

+ Students complete the exit ticket: one feature they would build next.

! Reinforce human control: students choose prompts, constraints, review choices, and how AI output is used.

Assessment Notes (0-5 points, cumulative)

Points	Criteria	Total
1 pt	Link opens; project is clearly named; widget UI is present with label, topic textbox, Go button, and output textbox arranged with layout rows.	
1 pt	Correct event and request wiring: button click reads textbox1, builds the prompt, sends it to ChatGPT in waiting mode, and stores the story.	
1 pt	Base Story Writer behavior works: Go generates a story about the entered topic and displays it in textbox2 without errors.	
1 pt	Enhancement of the same app, such as streaming, UX polish, prompt controls, length/format control, or cleaner styling/layout.	
1 pt	New creative writing app using the same skills, such as a Poem Maker, Joke Writer, Song Lyric Generator, or Scene Writer.	/5

Teacher tip: To earn all 5 points, the project should function correctly, include one concrete improvement, and extend the idea into a new creative-writing theme or polished adaptation.