

Let's Play! - Iro

CS177 Long Project

[Link to Live Game](#)

Group Members & Roles

Member List

Name	CCSF username	Github username
Ilmira Andronico	iandroni	iandroni
Christopher Black	cblack1	ChrisBlack1989
Jasmine Farrell	jfarrel6	jasmineiris
Evan Morrison	emorris7	evmo
Tatyana Polyakova	tpolyako	tatyana12
Robert Mitchell	rmitchel	squeakyheatr
David Wayman	dwayman	r3dcrosse
Zack Pierce		
Ariel Salazar		
Greg Stephens		

Functional Teams

- **Coding:** Jasmine Farrell, David Wayman, Chris Black, Robert Mitchell, Tatyana Polyakova
- **Specs & Documentation:** Ilmira Andronico, Evan Morrison, Robert Mitchell

Individual Credits

(Each group member was responsible for writing their own list.)

Jasmine Farrell

- Iro Front page: Created all the images, cluster code and index.html in cluster file.
- Animated “Count the ... yellow circles” pages in Instruction folder: Fixed the animated page transition to 2 sec in lieu of the original 8 secs , changed RGB colors to match game colors, removed extra files, added source path to html file, updated content.

- Final Game page: created test.js and index.html in bubbles folder which contains the: animated circles, colors, bounce effect, fading effect, sound, and mouse click to remove bubbles.

David Wayman

- Essential for the game to be a game.
- Final game page: worked on test.js and index.html in bubbles folder; created score tracker, optional sound buttons, response to win or lose, replay features.
- Animated “Count the ... yellow circles” pages in Instruction folder: Updated content.

Robert Mitchell

- Modified bubbles file and changed name to bubbles2.
- Animated “Count the ... yellow circles” pages in Instruction folder: created fade effect for the pages to transition.
- Originally change content to instruct the child on how to play the game, later updated by Jasmine and David to reflect new content.
- Technical specification.

Tatyana Polyakova

- Creation the form “Tell your friend about IRO.” This form would be helpful to let a lot of other people to know about IRO game.

Ilmira Andronico

- requirements
- function specs
- results of rounds of user testing

Evan Morrison

- functional spec
- misc writing & copy-editing in other sections
- git troubleshooting & assistance with more advanced git commands
- markdown formatting

(in progress)

Functional Specification

Overview

Iro is browser-based, interactive, educational game for children to learn how to distinguish colors and count objects within a time-limited period.

User Characteristics

Iro is designed to be played by anyone who can:

- See.
- Read basic English.
- Count to 21.
- Use a web browser and basic input devices (mouse and keyboard).

Game Objectives

On the main game screen, circular objects (balls) of different sizes and colors are floating around in random directions. The objective is to count how many **yellow** balls are shown, and submit the answer in a text field. There is an effective time limit of approximately 20 seconds, at which point the balls are no longer visible.

Requirements

- Platform-independent: can be played in any modern web browser at [this link](#).
- Coded in Javascript, HTML, and CSS.
- Colorful splash/intro screen that provides entry (via clicking) to game instructions.
- Colorful, dynamic game instructions, which transitions automatically after a few seconds to the game.
- Game screen with balls of different colors and sizes moving in random directions.
- Balls can be removed by clicking them.
- Balls begin fading from visibility after 5 seconds, completely disappearing after approximately 20 seconds.
- Background music (Jackson 5).
 - Buttons to switch music on or off.
- Text field and submit button to enter a guess.
- Pop-up box that provides feedback on guess (correct or incorrect).

- Buttons offering links to play again, or return to game home or game instructions.

Non-Goals

“Might-be-nice” game options that we are not implementing at this time:

- Change target color (the color that is counted).
- Change object shapes (circles only).
- Progression of difficulty levels.
- Concept of “score” across games (no credit for getting 2, 3, 4 correct answers in a row).
- Rewarding players for *speed* in choosing correct answer.
- Saving player results across sessions.

Screen-by-Screen Specification

Iro consists of five screens:

1. Home / splash screen.
2. Instruction #1.
3. Instruction #2.
4. Main game-playing screen.
5. Result / feedback screen.

Home / splash screen

The splash screen (located at the base URL) features the name of the game (**‘Iro’**) at top, the word **‘Enter’** at bottom, and a colorful, animated arrangement of colored circles occupying the space in the middle.

The word ‘Enter’ is a hyperlink to the first instruction page (see below).

Instructions

- Page 1: The words **‘Count the’** are formed by an arrangement of different-colored circles. The circles respond to mouse movements via Javascript animation.
- Page 2: The words **‘Yellow circles’** are formed by an arrangement of yellow circles. Similar to the first page, the circles respond to mouse movements via Javascript animation.

Main game screen

The main game screen features a randomly-generated number of circles of different colors and shapes, moving around in randomly-generated directions. At the bottom of the screen is a thin bar containing (left-to-right):

- A near-full-width text-input field which prompts: “How many yellow balls are there?”
- A submit button titled “Go!” – ~50px wide.
- A button-pair for turning the audio on (blue) or off (red) – ~80px wide.

The moving circle animation begins fading after 5 seconds and fades completely after ~20 seconds.

Background music (Jackson 5) starts on page-load by default, but can be switched off by the user.

Result / feedback screen

When the player submits an answer, a popup-style screen is overlaid on the game screen and provides:

- Feedback on the correctness of the player’s answer.
- The actual correct answer.
- Links to play again or return to the Home or Instructions pages.

The result screen is black text on a white background. For increased contrast, the game screen underneath fades to a gray background.

If the answer is incorrect, the feedback says:

Uh Oh! Try again!
Good try! There are actually [number] yellow circles.

If the answer is correct, the feedback says:

CORRECT!
Congratulations! You're correct! There are [number] yellow circles!

The link buttons read:

- Home
- Back To Instructions...
- Play Again

The ‘Play Again’ button is white text on blue background for increased contrast.

Technical Specification

Iro will be developed in Javascript, CSS, and HTML. It will consist of home screen, instructions, game mode and result screen.

- Sources [Codecademy](#) for animation of bubbles in Introduction pages.
- Uses [Bootstrap](#) framework for results pop-up screen, text input, sound buttons (using JQuery),
- Uses [D3.js](#) framework for animation of clusters.

All algorithms used are predefined, except the one that checks for spelled numbers, which was created by David Wayman.

User Documentation

Home / splash screen

Click the **Enter** button to view game instructions.

Instructions

Page 1

Use the mouse to interact with the balls, or wait 2 seconds for the next instruction screen to load automatically.

Page 2

Use the mouse to interact with the balls, or wait 2 seconds for the game screen to load automatically.

Main game screen

- Identify the yellow circles and count them before they fade out (5-20 seconds).
- Click a circle to remove it from the screen.
- When you have formulated an answer, enter it in the text field at the bottom of the screen. You may enter your answer as either a number (5) or a word (five).
- Click “Go!” to submit your answer and view the result.
- To mute the background music, click the red button at far-bottom-right.
- To turn the background music on after you have turned it off, click the blue button next to the red button.

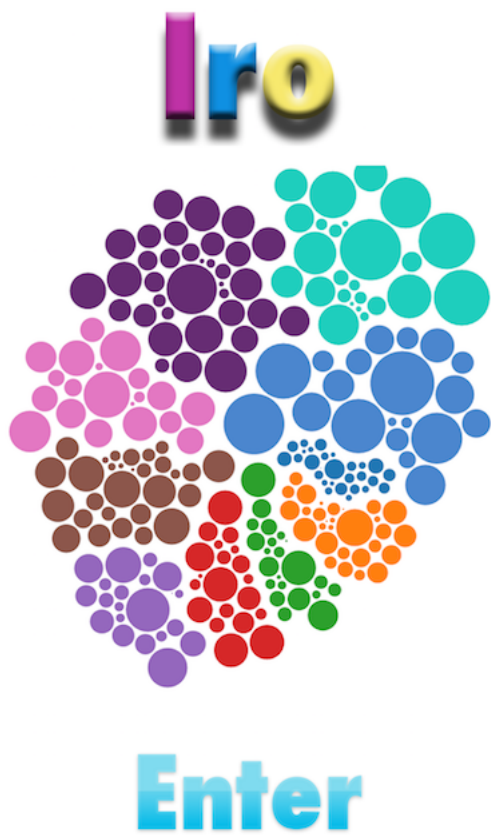


Figure 1:



Figure 2:

Yellow Circles

Figure 3:

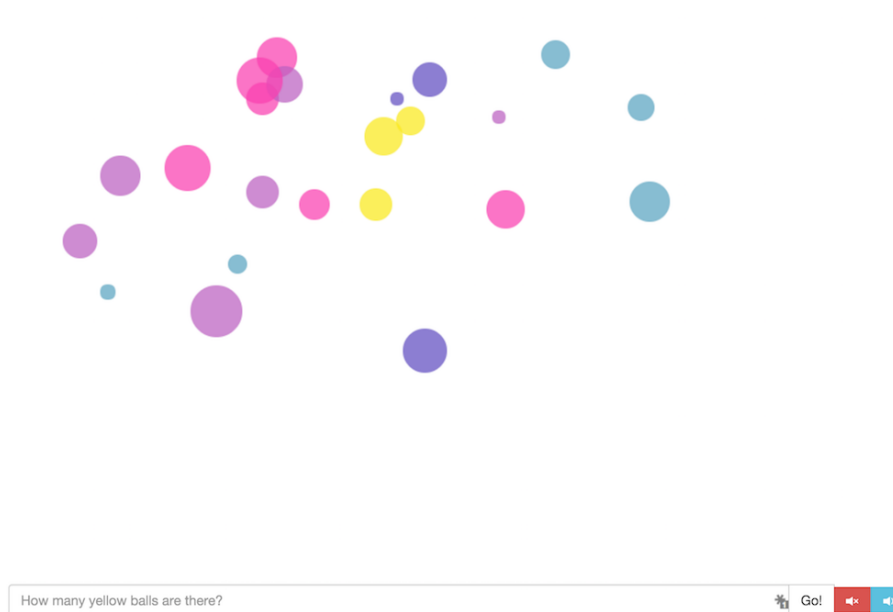


Figure 4:

Results / feedback screen

If answer is incorrect:



Figure 5:

If answer is correct:

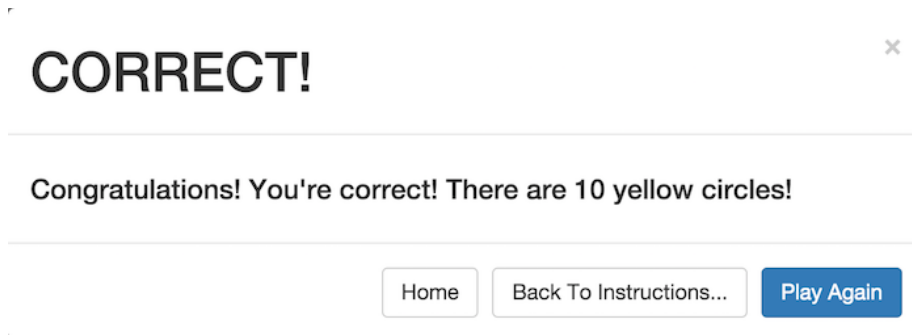


Figure 6:

User Research / Testing

We asked three children to play **Iro** and provide user feedback.

Layden (5 year old boy)

He does not know how to read yet, so he needed help from his parents to play. He liked colored circles and the ability to play with them (move them around and pop them). Also he liked counting them.

Kailyn (7 year old girl)

She was able to play this game on her own. Liked playing it for first 15 min but then she got bored because it seem repetative, there were no other colors or shapes to count.

Kendal (11 year old girl)

She thought this game was too easy to play and needed to be more complex. She would have liked more rounds, different choice of colored circles or shapes to count.

We also got feedback from adult users.

Aaron (computer science instructor)

- Default too many circles
- Create levels:
 - Randomization per circle for each level
- Graphic style should be the same
 - making the font and frame look similar goes a long way
- Fading effect timing should slow down if more circles appear
- Gravity cluster: incorporate better into the game or add more dimension to the circles using the gravity effect

Developer thoughts on user testing

We agree with our test users that **Iro** could be made more interesting by adding complexity, in particular:

- Different target colors and shapes for each round of the game.
- Progression of difficulty levels along with some method of keeping score.
- Vary the fading time.

Unfortunately we were unable to implement these enhancements due to time constraints and limited Javascript programming skills. That being said, it is an intentionally simple game, and a reasonable “minimum viable product” given the constraints.