Faculty of Engineering & Information Technology

Computer Engineering Department.

Codey

Prepared by:
Hala Kawni
Nasser Aker

Supervised by:
Dr. Manar Qamhieh

Presented in partial fulfillment of the requirements for Bachelor Degree in Computer Engineering

2026

Acknowledgment

We express our sincere gratitude to God, whose guidance and blessings gave us the strength and perseverance to see this project through to completion.

We offer our deepest thanks to the professors and teaching assistants of the Computer Engineering Department at An-Najah National University. Their dedication to education and the knowledge they imparted throughout our studies laid the foundation upon which this work was built.

We are especially grateful to our supervisor Dr.manar Qamhieh, for their continuous guidance, constructive feedback, and invaluable direction throughout the development of this project.

Finally, we dedicate this work to our families and friends, whose unwavering support, patience, and encouragement have been a constant source of motivation. And to our brothers and sisters across Palestine your resilience and spirit inspire everything we do.

Disclaimer Statement

This report was written by students at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the students. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

Table of Contents

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

List of Figures

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

List of Abbreviations

Abbreviation	Full Term
AI	Artificial Intelligence
API	Application Programming Interface
AR	Arabic
bcrypt	Blowfish Crypt (password hashing algorithm)
CRUD	Create, Read, Update, Delete
CSS	Cascading Style Sheets
EN	English
GET	HTTP GET Method
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
JSON	JavaScript Object Notation
JWT	JSON Web Token
LLM	Large Language Model
MVC	Model-View-Controller

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

MongoDB	MongoDB Document Database
PNG	Portable Network Graphics
POST	HTTP POST Method
PUT	HTTP PUT Method
DELETE	HTTP DELETE Method
REST	Representational State Transfer
RTL	Right-to-Left
LTR	Left-to-Right
SDK	Software Development Kit
UI	User Interface
URL	Uniform Resource Locator
UX	User Experience
WWW	World Wide Web
XP	Experience Points

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Abstract

Children today live in a digital world, yet few platforms make learning to code genuinely engaging, accessible, and creative. Codey is a full-stack interactive learning platform designed to teach programming concepts, digital literacy, and computational thinking to children through structured courses, gamified challenges, AI-powered tools, and a suite of creative game builders.

The platform was built using Flutter for the cross-platform mobile client and Node.js with Express and MongoDB for the backend. Codey provides four built-in courses: Codey Jr., a 15-level block-based sequencing game; Digital Literacy, covering internet safety; Data is Everywhere, introducing data concepts; and Coding Chatbots, a 9-exercise course in which children learn the fundamentals of artificial intelligence by recording their own body-pose samples using a webcam, training a gesture classification model, and then using that model to control a game character through physical movement. Beyond these, educators can create and publish their own interactive courses using a visual course builder that supports slide lessons, quizzes, word searches, word matching, fill-in-the-blank exercises, swipe classification, and sorting activities all of which can be generated instantly via an AI assistant powered by the Groq API.

A defining feature of Codey is its suite of four game creation environments: a front-view side-scroller builder, a top-view builder, a Scratch-like visual block programming environment with categories including Events, Motion, Control, Operators, and Variables, and a text-code builder with a custom programming language and a live game preview.

Students can design, build, and publish their own games, and browse creations from the community through a dedicated Discover tab.

The system supports three roles: children (learners), parents (who monitor their child's activity by linking to their account using the child's unique code), and administrators (who oversee platform content and users). Classrooms are a peer-based feature where children join using a shared code and compete through weekly and head-to-head challenges with their friends, with activity feeds visible only among classroom members. The platform is fully bilingual in Arabic and English with right-to-left layout support, includes text-to-speech accessibility, and offers Google Sign-In. The result is a comprehensive, creative, and child-friendly coding education ecosystem.

Chapter 1: Introduction

1.1 Statement of the Problem

The demand for digital literacy and coding skills among children has grown substantially over the past decade, driven by the increasing integration of technology into every aspect of modern life. Despite this, most children still lack access to structured, engaging, and age-appropriate programming education, particularly in Arabic-speaking communities where localized educational software remains scarce.

Existing platforms such as Scratch or Code.org provide introductory coding experiences but fall short in several areas: they typically lack adaptive or AI-generated content, do not support real-time classroom collaboration, and offer limited personalization of the learning journey. Parents also have minimal visibility into their child's progress, and educators have few tools to build and publish their own custom curricula.

Furthermore, children are inherently motivated by game-like experiences earning rewards, competing on leaderboards, and progressing through levels. Traditional educational platforms fail to sustain this engagement over time. There is a clear need for a comprehensive, bilingual, AI-assisted, and gamified learning platform tailored specifically to the needs of young learners and their educators.

1.2 Objectives

The primary objectives of the Codey platform are as follows:

- To provide children with an engaging, gamified environment for learning programming concepts, data literacy, and digital skills.
- To enable educators to build custom interactive courses using a visual drag-and-drop course builder without requiring technical expertise.
- To leverage AI (Groq API / Llama 3.1) for automated generation of educational content including quiz questions, word searches, fill-in-the-blank exercises, and sorting activities.
- To support peer-based classroom features including leaderboards, weekly challenges, head-to-head competitions, and activity feeds shared among classmates.
- To give parents a dedicated interface for linking to their child's account using a unique code and monitoring their learning progress.
- To offer a fully bilingual platform (Arabic and English) with text-to-speech accessibility support.
- To provide administrators with robust tools for user management, content verification, and platform analytics.

1.3 Significance of the Work

Codey addresses a genuine gap in the educational technology landscape, particularly for Arabic-speaking young learners. By combining gamification, AI-powered content generation, peer-based classroom features, and creative game building tools in a single platform, the project offers several meaningful contributions:

- It lowers the barrier for educators to create rich, interactive learning content through its AI-assisted drag-and-drop course builder.
- It increases child engagement and learning retention through game mechanics including stars, scores, leaderboards, and peer challenges across all courses and game builders.
- It empowers parents with visibility over their child's learning journey through a dedicated interface linked via a unique child code.
- It demonstrates the feasibility of deploying large language models (LLMs) in an educationally-focused context, with AI-generated content passing through educator review before reaching children.
- It introduces children to real-world AI concepts hands-on through the Coding Chatbots course, where learners train and apply their own gesture recognition model.
- It serves as a reusable, scalable foundation for future educational platforms targeting similar audiences.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Chapter 2: Constraints, Standards/Codes, and Earlier Coursework

2.1 Constraints

Several technical, practical, and environmental constraints shaped the development of this platform:

1. Mobile Development Learning Curve: The Flutter framework was new to the development team. Significant time was invested in learning Flutter's widget system, state management patterns, and the Dart language before productive development could begin.

2. AI API Rate Limits and Latency: The Groq API (used for LLM-based content generation) is subject to rate limits. Under high-concurrency usage or repeated rapid requests, content generation may be delayed or throttled, which impacted the user experience during testing.

3. MongoDB Schema Flexibility vs. Consistency: While MongoDB's schema-less nature accelerates early development, it requires careful discipline to maintain data consistency across models, particularly for complex nested structures such as course levels, builder blocks, and game progress records.

4. Device Performance Variability: Since the application targets children's devices, which range from low-end Android devices to modern tablets, performance

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

optimization was a constant consideration, especially for animated game screens and the rich course builder interface.

5. Localization Complexity: Supporting both Arabic (right-to-left) and English (left-to-right) layouts required careful attention to UI mirroring and text rendering throughout the application.

6. Budget Constraints: The project was developed without budget for paid cloud infrastructure. All backend services were run on local or free-tier hosting, which imposed limitations on scalability testing.

2.2 Standards and Codes

The project adhered to the following standards and best practices:

- **REST Architectural Style:** All server-client communication follows RESTful principles, with consistent HTTP methods (GET, POST, PUT, DELETE) and JSON response structures.
- **MVC Pattern (Backend):** The Node.js backend is organized using a feature-based MVC pattern. Each functional domain (users, courses, games, AI, classrooms, admin) is encapsulated in its own set of models, controllers, services, and routers, keeping the code modular and easier to scale and maintain.
- **JWT Authentication:** All protected routes use JSON Web Token (JWT)-based stateless authentication, with tokens validated by a shared auth middleware.
- **bcrypt Password Hashing:** User passwords are hashed using bcrypt (salt rounds = 10) before storage, in accordance with standard security practice.
- **Flutter/Dart Best Practices:** The client code follows Flutter's recommended widget composition patterns, uses named routes for navigation, and separates UI logic from service/API logic.
- **Git and GitHub:** Used throughout development for source code management and version control.

- **Postman:** Used for testing and verifying API endpoint functionality during development.
- **Android Studio:** Served as the main development environment for the Flutter mobile application.

2.3 Earlier Coursework

The following courses from the Computer Engineering program directly supported the development of this project:

- Data Structures and Algorithms: Fundamental to designing efficient game scoring, leaderboard sorting, and course progress tracking logic.
- Database Systems: Provided the foundation for designing MongoDB schemas, understanding query optimization, and structuring relational-like data in a document-based system.
- Web Programming : Covered JavaScript, Node.js, Express, REST API design, and frontend development all core to the server and admin panel.
- Software Engineering: Introduced MVC architecture, design patterns, and project management principles applied throughout the system's design.
- Mobile Application Development: Provided initial exposure to mobile UI development concepts, supplemented by self-study of Flutter.
- Artificial Intelligence: Gave theoretical grounding for understanding LLMs and prompt engineering, applied in the AI content generation features.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Chapter 3: Literature Review

3.1 Gamification in Education

Gamification, the application of game-design elements to non-game contexts has been widely studied as a motivational tool in educational settings. Deterding et al. (2011) defined gamification as the use of game elements such as points, badges, and leaderboards in non-game contexts to drive engagement. In educational software, these mechanisms have consistently been shown to increase time-on-task, intrinsic motivation, and learning retention among younger learners.

Hamari et al. (2014) conducted a systematic review of empirical studies on gamification and found that while the effectiveness is context-dependent, motivational outcomes are generally positive when the gamification elements align with the learners' goals and preferences. Codey incorporates stars (as rewards), leaderboards (competitive motivation), and progressive level unlocking (challenge-based engagement) all well-supported by the literature.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

3.2 AI-Assisted Content Generation in Education

The emergence of large language models (LLMs) has created new possibilities for automated generation of educational content. Kasneci et al. (2023) provided a comprehensive overview of ChatGPT and similar models' implications for education, noting their potential for generating exercises, explanations, and assessments at scale. The authors also cautioned about risks including hallucination and bias, recommending human review of AI-generated content.

Codey addresses this concern by using AI generation as a teacher-assistance tool rather than a replacement for human authorship. Educators initiate content generation from curated lesson texts, and all generated activities can be reviewed and edited before publication.

3.3 Child-Centered Design and Coding Education

Research on programming education for children consistently emphasizes the importance of visual, tangible, and scaffolded experiences. Resnick et al. (2009) described the design philosophy behind Scratch, highlighting the importance of low floors (easy to start), high ceilings (room to grow), and wide walls (supporting diverse projects). While Codey offers structured courses, it also embraces open-ended creation through its suite of game builders allowing children to design and publish their own games using block-based or text-based programming environments.

Studies on bilingual educational software (Blom et al., 2017) have demonstrated that children learn more effectively when instructional content is available in their native language. This directly motivated Codey's full Arabic-English bilingual support with right-to-left layout adaptation.

3.4 Related Platforms

Several existing platforms were reviewed as part of the background research:

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

- **Scratch (MIT):** An open-ended block-based programming environment for children. Highly creative but lacks structured progression, peer-based classroom features, and AI assistance.
- **Code.org:** A structured curriculum-based platform with strong teacher tools. Limited to web, no bilingual support, and does not offer custom course authoring.
- **Duolingo (as a gamification model):** While not a coding platform, Duolingo's streak system, XP rewards, and leaderboard mechanics are well-studied and served as a design reference for Codey's engagement systems.
- **Khan Academy Kids:** Strong child-friendly UX but focused on general education rather than computing, and lacks an AI content generation layer.

Codey differentiates itself through the combination of: a custom AI-assisted course builder, creative game builders, peer-based classroom features, bilingual support, hands-on AI education through the Coding ai course, and a unified platform for children, parents, and administrators.

Chapter 4: Methodology

4.1 Web Landing Page

The Codey platform includes a public-facing website that serves as the main entry point for new users. The website introduces the platform's features, targets both parents and educators, and provides access to sign-up and login options. It is fully bilingual, supporting both English and Arabic, with a language toggle visible in the top navigation bar.

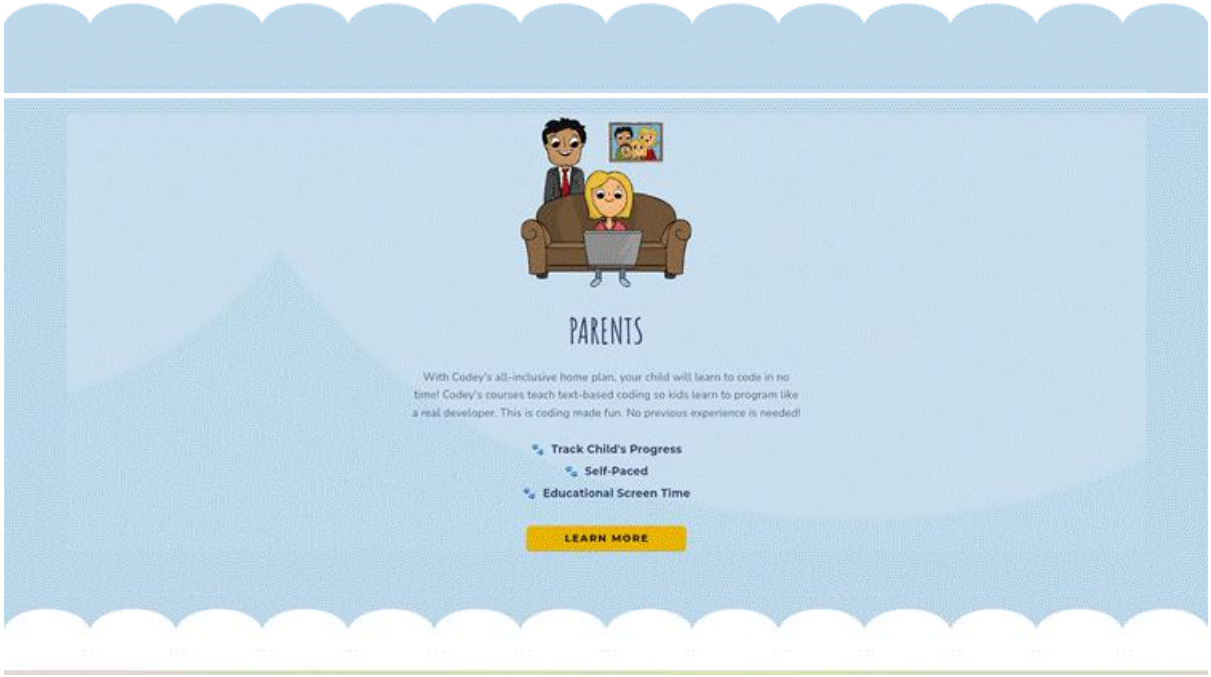
University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Write Code.

Codey is an AWARD-WINNING online platform that teaches kids real coding languages like Python and JavaScript. Children and teenagers learn block-based and text-based coding through an engaging game-like environment.

Millions of Codey's students are now excited about coding! Codey does not require prior coding experience and is designed for home and family use.

Do you want to start coding now? Kids from 5–14 years old can learn block-coding, text-coding, and Python all while playing! Kids as young as 5 can start programming and build their own games. Try it today!



University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

TEACH KIDS COURSES THAT ARE EXCITING

A fun coding course for kids

CURRICULUM

Codey offers educational resources for students of different grades and experience levels. From Codey Jr. to advanced courses, students learn coding basics such as block-based and text-based coding. Kids also learn how to code in real programming languages like Python and JavaScript. Through Codey, kids will develop the necessary skills for the future while having fun!

SIGN UP NOW

free trial

A fun coding course for kids

GET YOUR STUDENTS CODING IN NO TIME!

Codey is a fun and educational game-based environment where kids learn to code without any prior experience. After completing Codey's award-winning coding courses, kids will be able to navigate through the programming world with a sense of confidence and accomplishment.

EDUCATORS PICK
BEST OF STEM
WINNER 2024
AI

KIDS WILL LOVE LEARNING TO CODE WITH CODEY

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

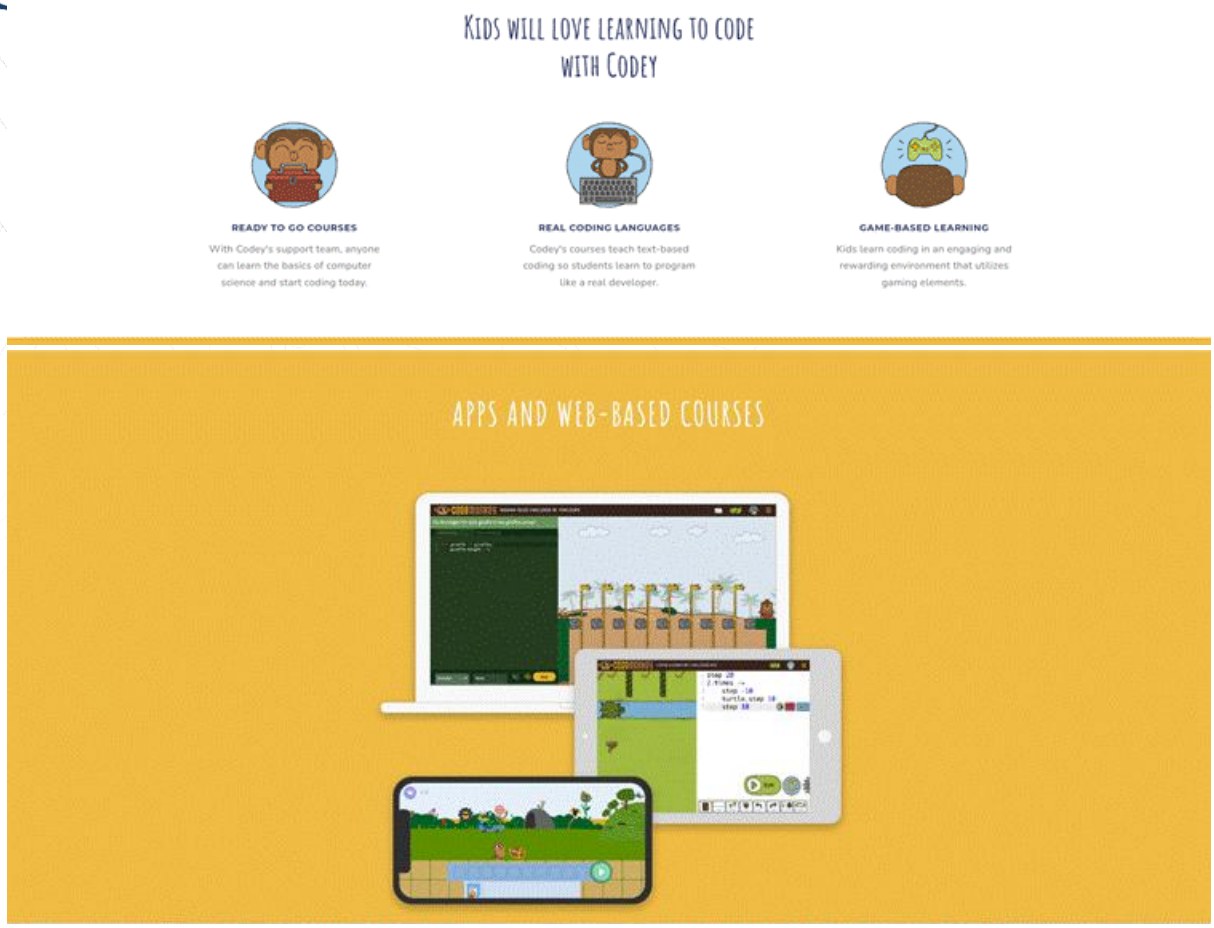


Figure 4.2: Web Landing Page

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

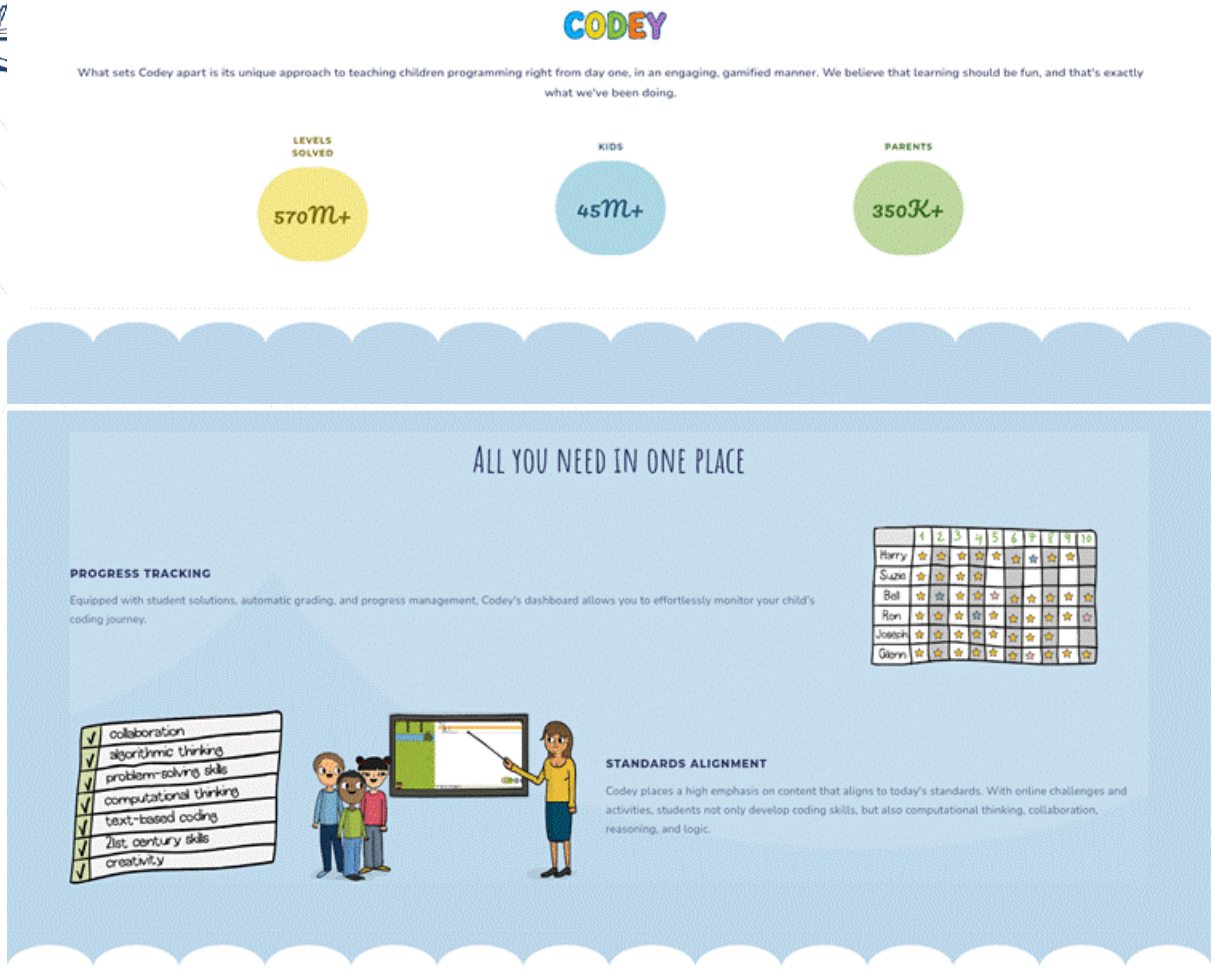


Figure 4.3: Web Landing Page - view 2

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.2 Navigation Bar

The top navigation bar contains the Codey logo on the left, followed by links to Courses, Plans, and Resources. On the right side, three action buttons are available: Kids Sign Up, Sign Up, and Log In. A language selector allows users to switch between English and Arabic.

Figure 4.4: Navigation Bar

Figure 4.5: Navigation Bar - view 2

4.3 Registration Screens

4.3.1 Role Selection Screen

When a new user clicks Sign Up, they are presented with a "Who Are You?" screen asking them to select their role. Two options are displayed as illustrated cards: Student ("Join your classmates in fun coding games") and Parent ("Introduce your child to Computer Science and track their progress"). An "Already a member? Log in to your account" link is available at the bottom. All registration screens maintain the platform's sky-and-clouds background with theCodey elephant mascot.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.6: Role Selection Screen

4.3.2 Classroom Code Screen

After selecting Student, the system asks "Do you have a classroom code?" explained as a code given by a teacher for creating a user account. Two cards are shown: Yes (I have a code) and No (I didn't receive any code). Selecting Yes reveals a text input field at the top where the student enters their classroom code, with a Next button to proceed.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

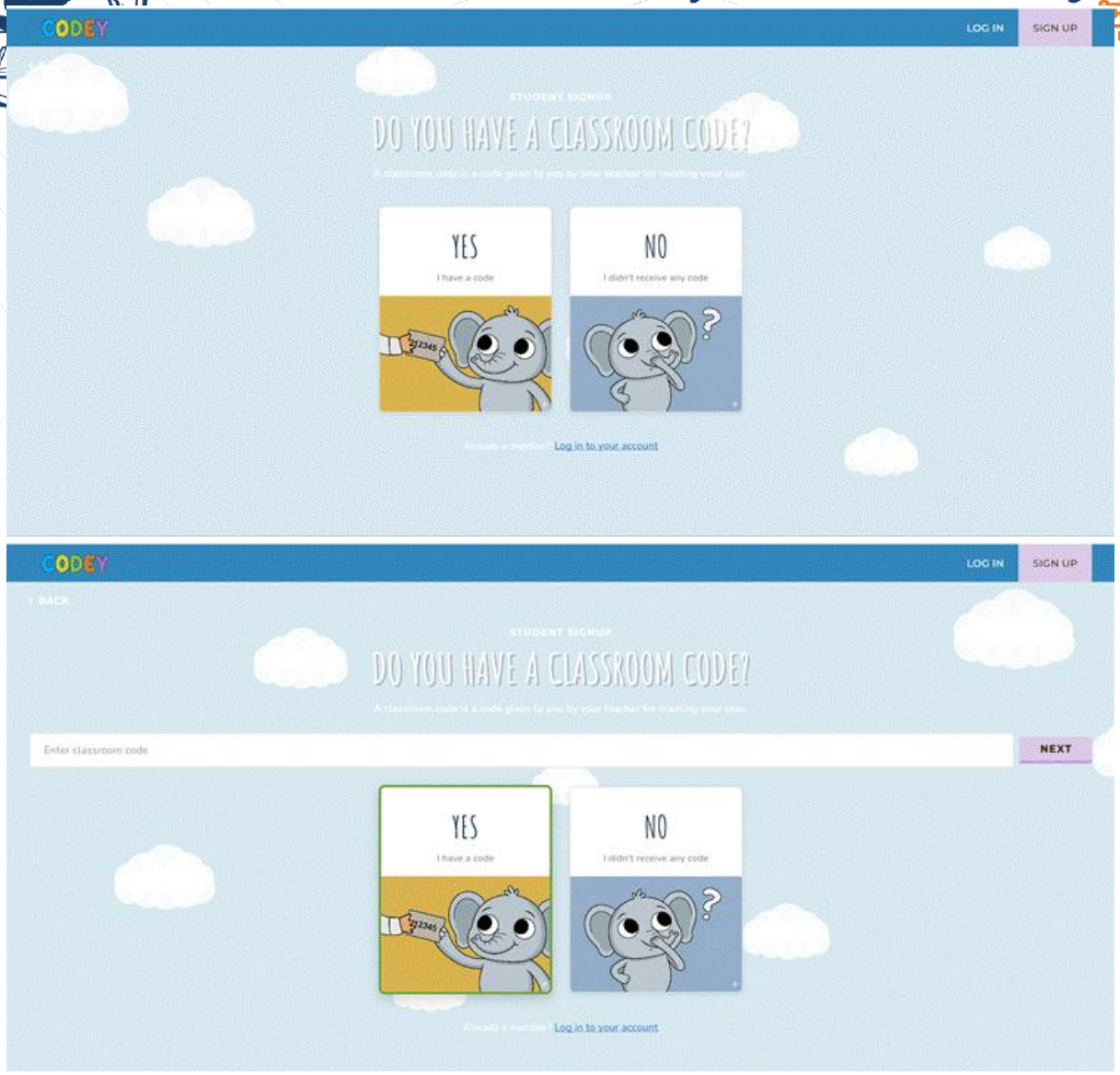


Figure 4.7: Classroom Code Screen

4.3.3 Playing Environment Screen

The next step asks "Where will you be playing?" with two options: Home ("I'm playing on my own") and Classroom ("I belong to a Codey classroom with a friend").

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

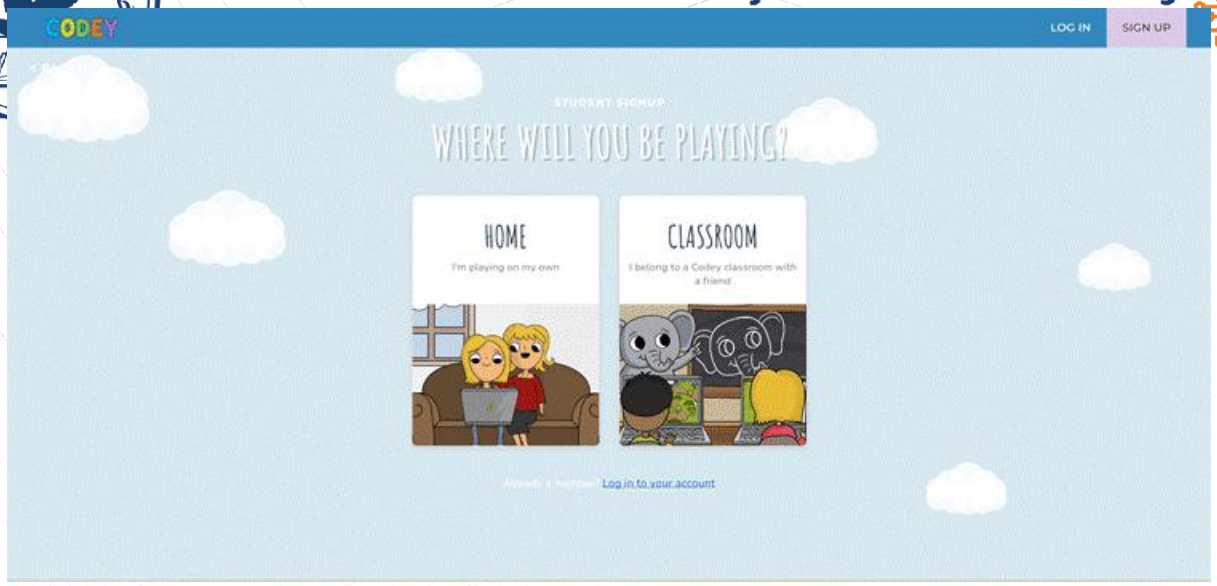


Figure 4.8: Playing Environment Screen

4.3.4 Age Selection Screen

The student is then asked "How old are you?" with a numeric spinner input and a Next button. A playful animation shows the elephant mascot growing in size as the age value increases, providing visual feedback that makes the experience engaging for children.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.9: Age Selection Screen

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.3.5 Gender Selection Screen

The student selects their gender (Male or Female) presented as two illustrated cards with a Next button to continue.

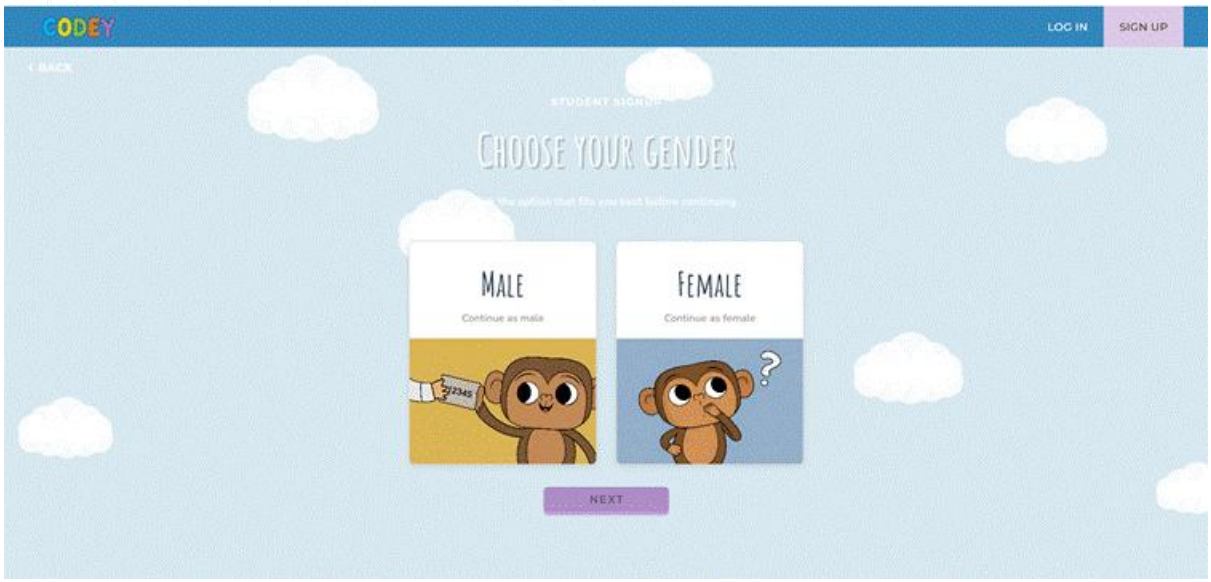


Figure 4.10: Gender Selection Screen

4.3.6 Account Details Screen

The final registration step presents a form with fields for Email, Display Name (with a privacy note advising not to use a full name), Password, and Re-enter Password, each with a show/hide toggle. Real-time validation is shown inline for example "Passwords do not match" or "This field is required" in red below the relevant field. A Sign Up button submits the form. On the right side, alternative sign-up options are offered: Google, Clever, Office 365, and ClassLink.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

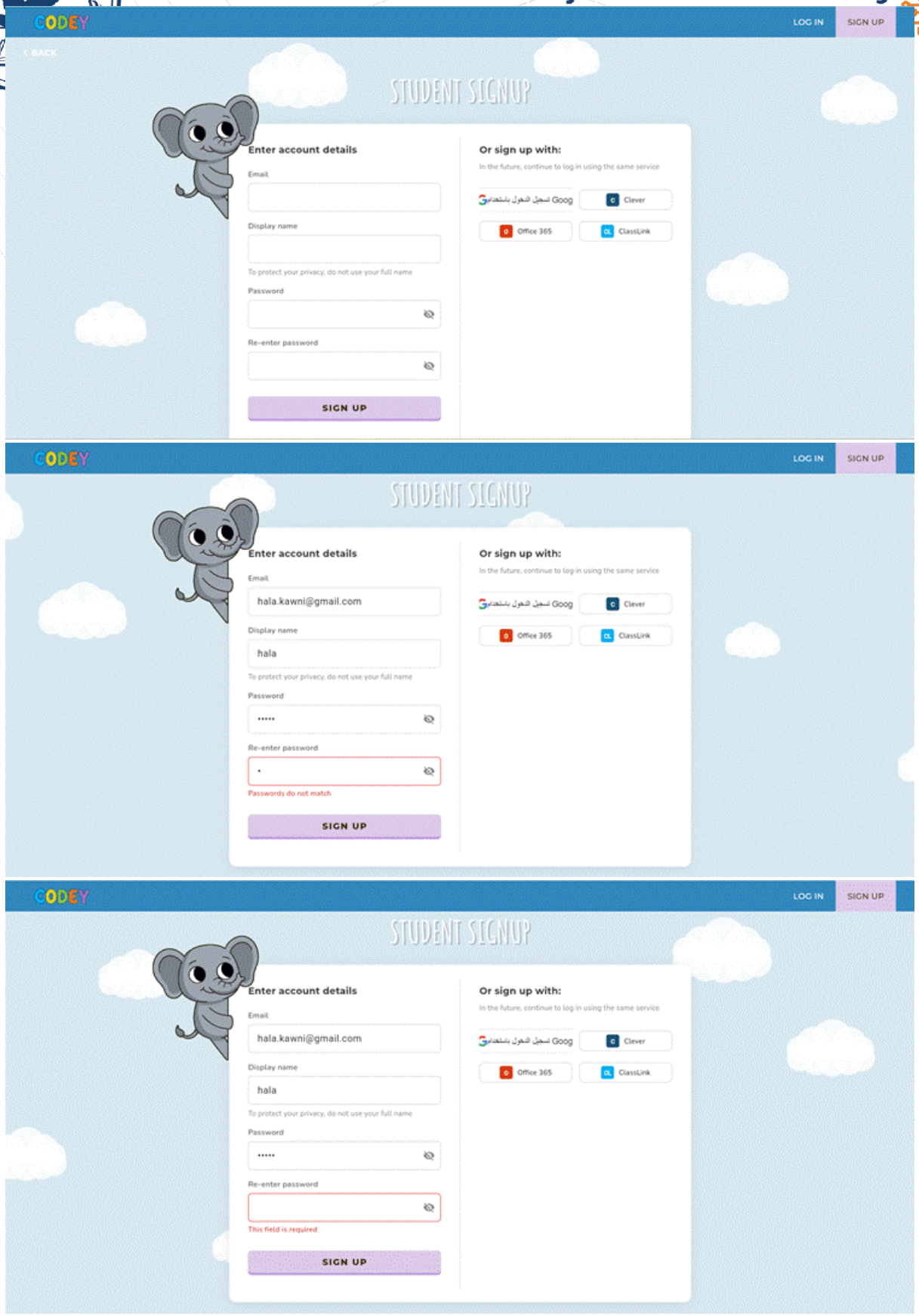


Figure 4.11: Account Details Screen

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.3.7 Email Verification Screen

Upon successful registration, a "Check your email" confirmation screen is displayed, prompting the user to verify their account via the email address provided. Two buttons are available: Resend Email and Back to Login.

Figure 4.12: Email Verification Screen

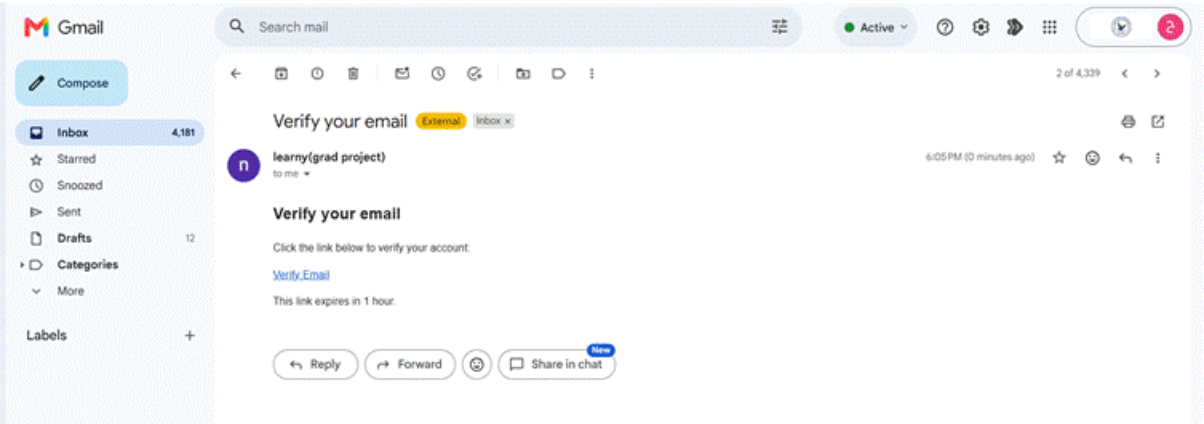


Figure 4.13: Email Verification Screen - view 2

4.3.8 Classroom Setup Screen

When a student selects "Classroom" as their playing environment, they are taken to the "Set Up Your Classroom" screen. Two options are presented: **Join** ("I have a code from a friend") and **Create** ("Start a new classroom").

- Selecting **Join** highlights the Join card with a blue border and reveals a text input field below where the student enters their friend's classroom code, followed by a "Continue to Sign Up" button.
- Selecting **Create** highlights the Create card with a blue border and automatically generates a unique classroom code (e.g. JBX68Y) displayed in large bold text below. The subtitle reads "Share this code with your friends so they can join your classroom." Two action buttons are provided: **Copy Code** (to copy the code to clipboard) and **New Code** (to regenerate a different code), along with a "Continue to Sign Up" button to proceed with registration.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

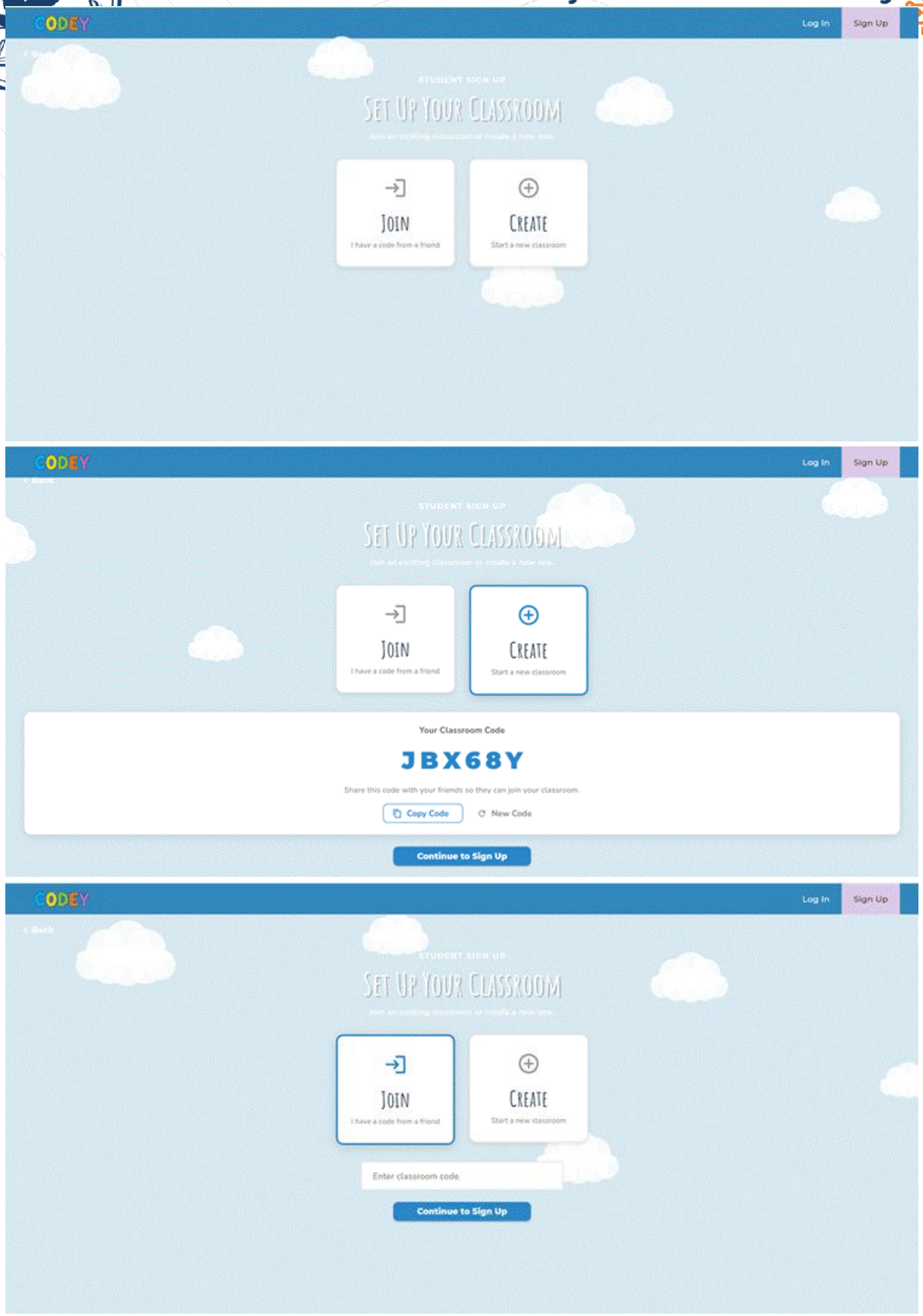


Figure 4.14: Classroom Setup Screen

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.3.9 Parent Registration Screen

When a user selects Parent on the role selection screen, they are taken directly to the Parent Signup page. Unlike the student flow, the parent registration skips the age, gender, and classroom steps and goes straight to the account details form. The form contains the same fields as the student form: Email, Display Name (with the same privacy note), Password, and Re-enter Password with show/hide toggles and inline validation. The same four third-party sign-up options are available: Google. A Sign Up button submits the form.

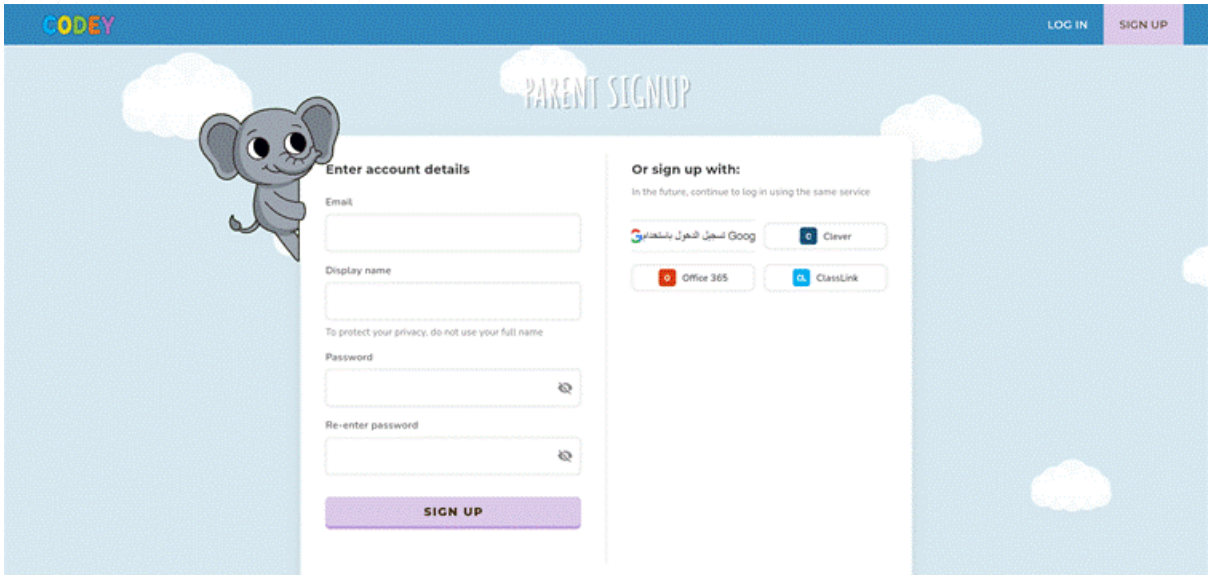
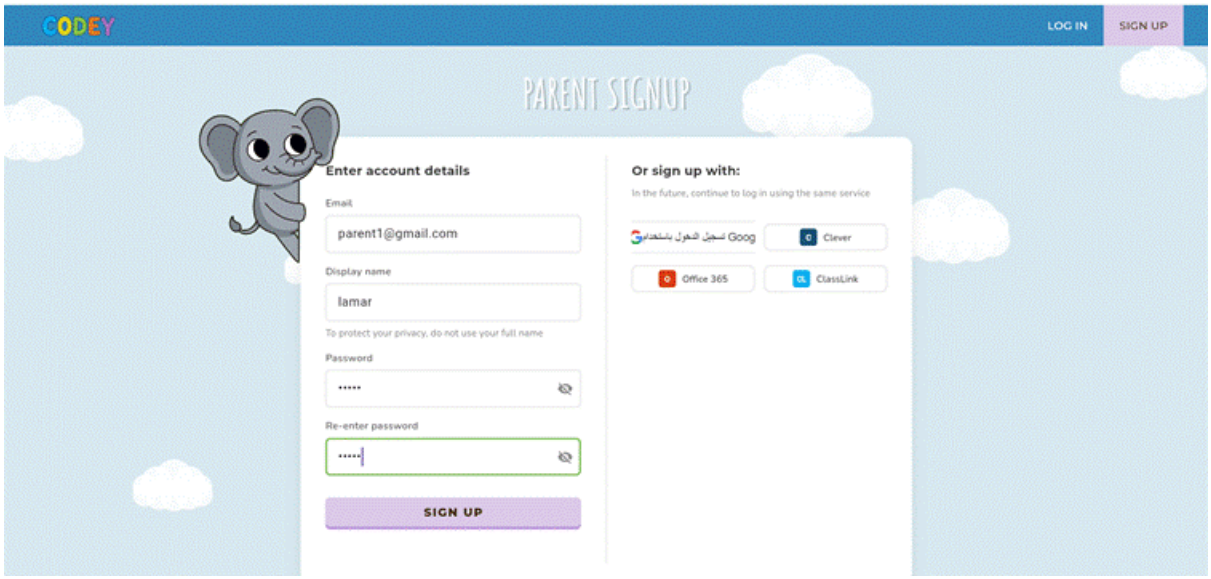


Figure 4.15: Parent Registration Screen



University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.16: Parent Registration Screen– view 2

4.4 Login Screen

The Login screen is accessible from the Log In button in the navigation bar or via the "Already a member?" link on the registration screens. It contains a two-column layout: on the left, an "Account Details" form with Email/Username and Password fields, a "Remember me" checkbox, a "Forgot password?" link, and a Log In button. On the right, alternative login options like: Google. A "Don't have an account? Sign up now!" link is provided at the bottom of the page for new users.

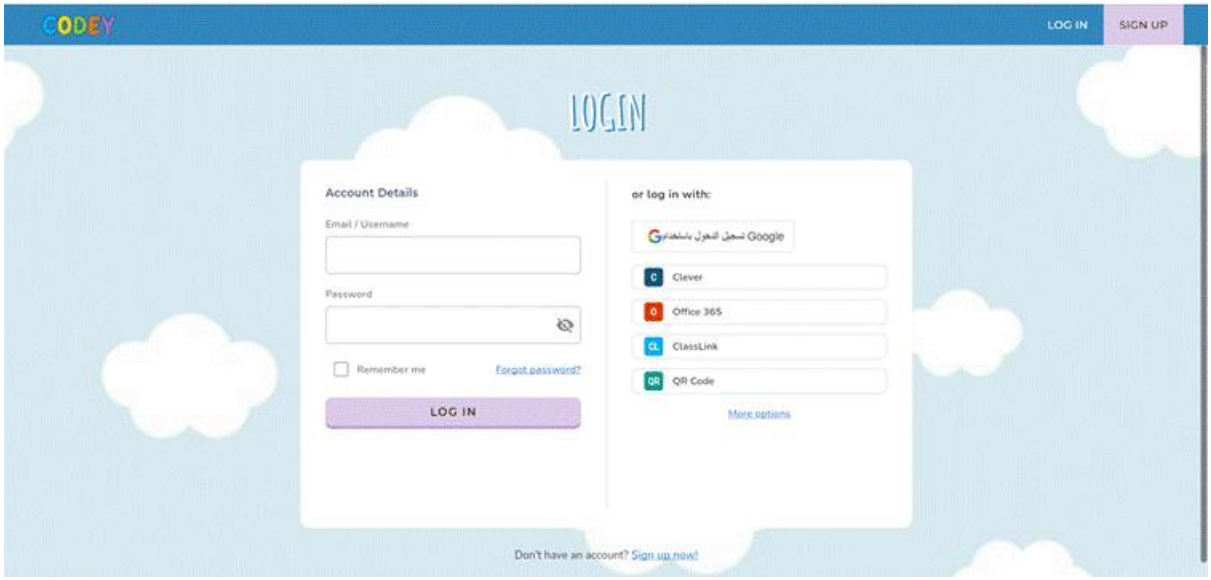


Figure 4.17: Login Screen

4.5 Parent Dashboard

After logging in, a parent is taken to the Parent Dashboard, which uses a distinct pink-accented header to differentiate it visually from the student interface. The left sidebar contains three navigation items: Parent Dashboard, Parent Resources, and Discover. A profile icon and hamburger menu are available in the top right corner.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

The main area greets the parent by display name (e.g. "Hello, lamar") with the subtitle "Welcome to your parent dashboard! Here you can view your children's progress and achievements." A prominent **Link a child +** button allows the parent to connect a child account using the child's unique code.

The dashboard is organized into three tabs: **Summary**, **Courses**, and **Creations**. The Summary tab displays four key statistics: Solutions Submitted, Courses Started, Total Stars, and Courses Created. Below the statistics, three information panels are shown:

- **Parent Dashboard panel**: Shows linked children. When no child is linked yet, it displays a placeholder message: "No children linked yet. Use 'Link a child +' to get started" along with the note "To start playing, log in to your child's account."
- **Current Course**: Shows the child's active course. Displays "No activity yet" when no child is linked.
- **Time Spent on Site**: A bar chart showing the child's daily time spent on the platform over the past 7 days (Monday through Sunday, in minutes).
- **Game Progress**: Shows the child's game progress. Displays "No activity yet" when no child is linked.

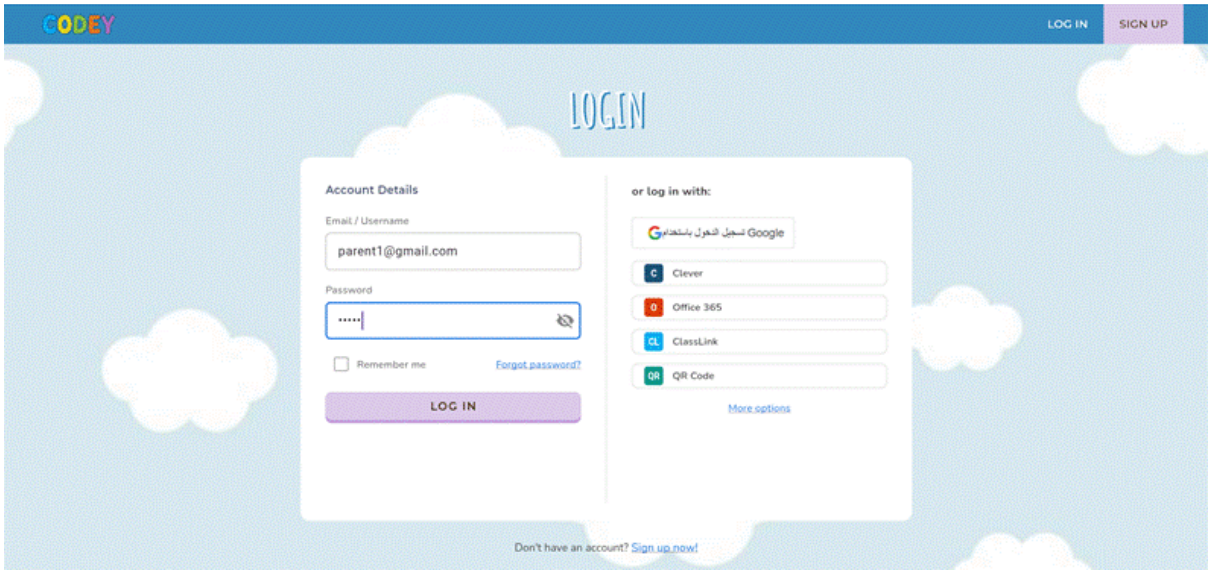


Figure 4.18: Game Progress: Shows the child's game progress

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

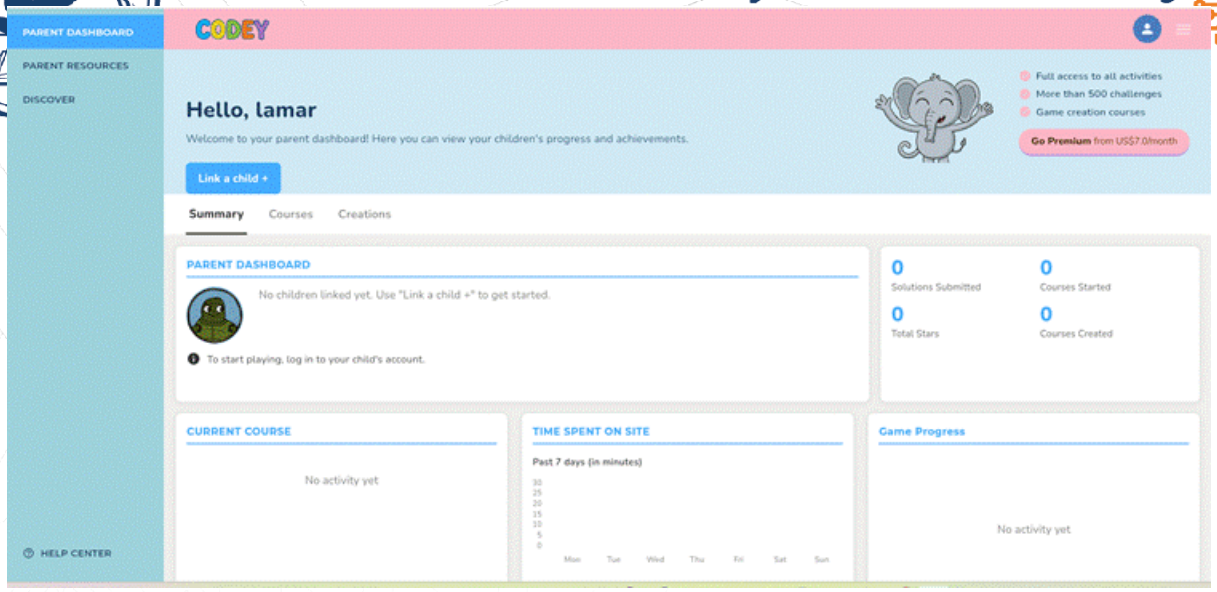


Figure 4.19: Game Progress: Shows the child's game progress - view 2

4.5.1 Child's Link Code Generation

To link a parent account to a child's account, the child must first generate a link code from their own dashboard. At the bottom of the student dashboard, a **"Share your progress with a parent"** option is available with a **Generate Code** button on the right. Clicking it generates a unique 6-character code that the child shares with their parent.

Figure 4.20: Child's Link Code Generation

4.5.2 Link a Child Screen

When the parent clicks the **Link a child +** button on their dashboard, they are taken to the "Link a Child" screen. It contains a single input field labeled "Child's Link Code" with a

placeholder "6-character code" and a **Link Child** button to submit. A "How it works" panel on the right side explains the process in four steps:

1. Your child logs into their account
2. They click "Share with Parent" in their dashboard
3. They tap "Generate Code" to get a 6-character code
4. Enter that code here to link their account

Upon entering a valid code and clicking Link Child, a success confirmation is shown with a green checkmark and the message "[Child's name] has been linked to your account!" along with a Back button to return to the dashboard.

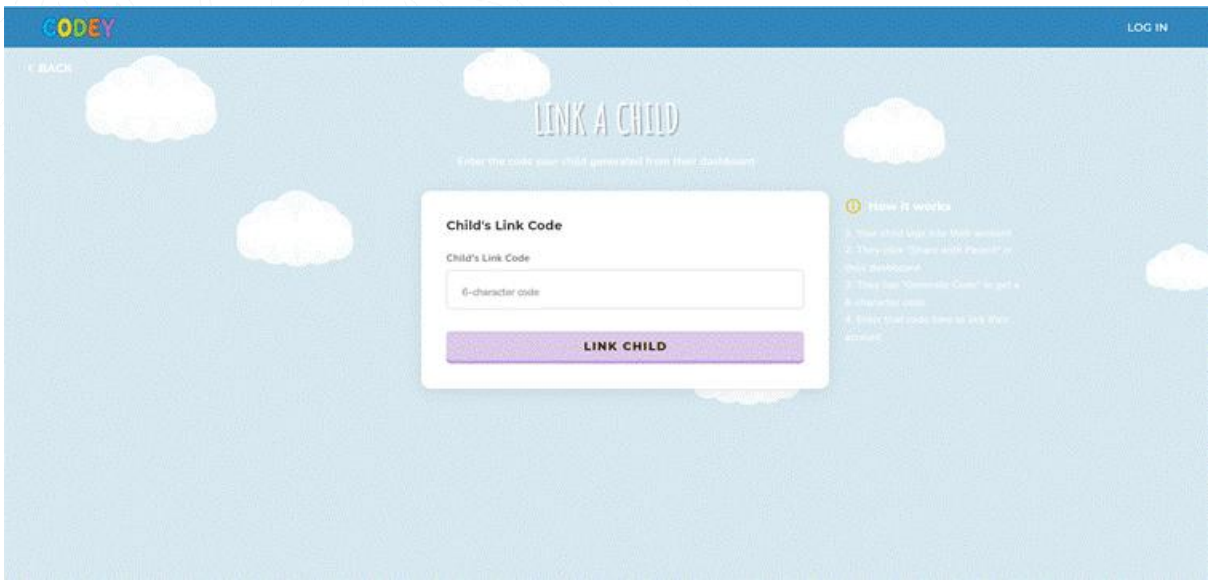


Figure 4.21: Enter that code here to link their account

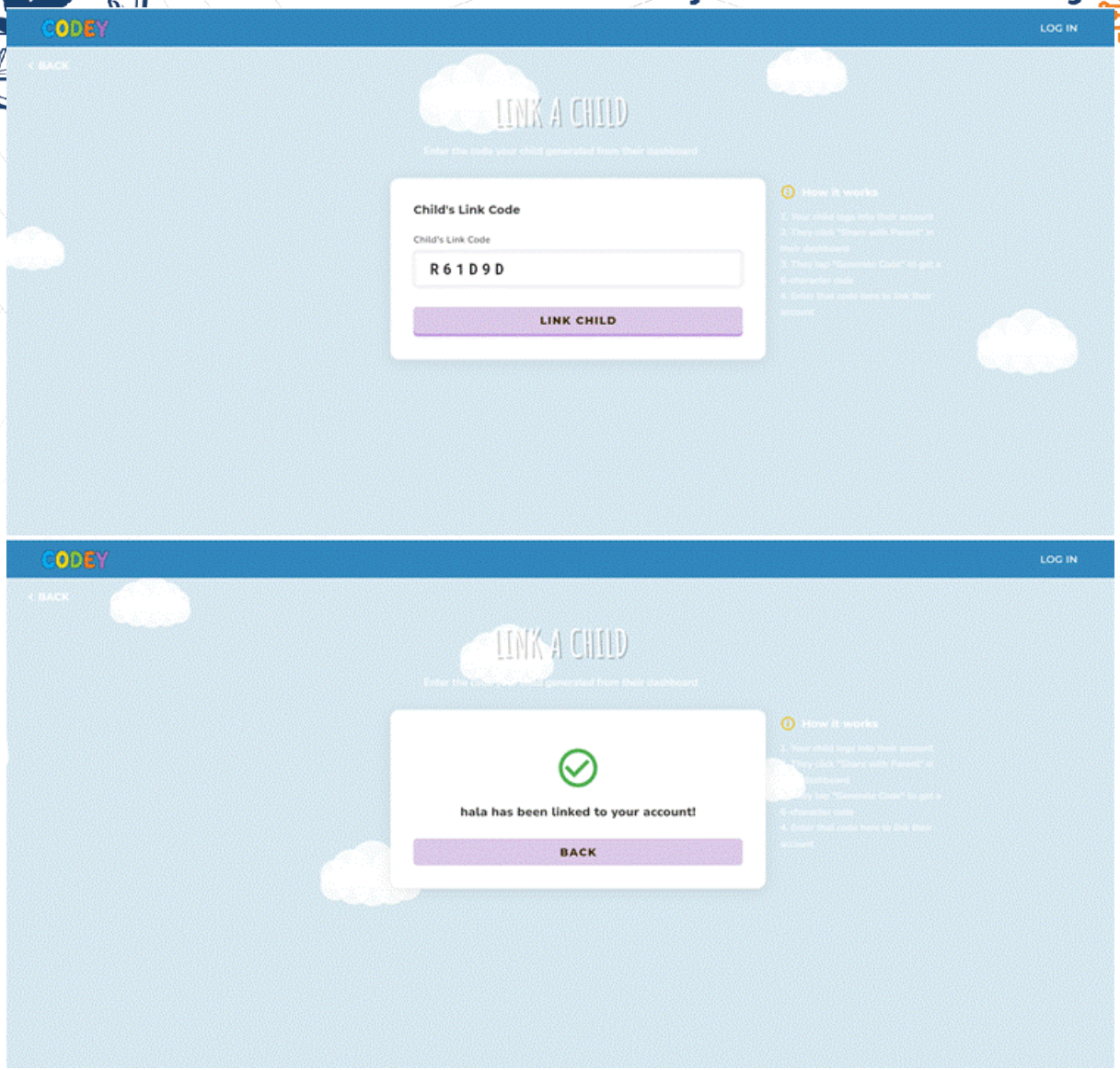


Figure 4.22: Enter that code here to link their account - view 2

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.5.3 Parent Dashboard After Linking

Once a child is successfully linked, the Parent Dashboard updates significantly. The child's display name appears as a tab next to the **Link a child +** button, allowing the parent to switch between multiple linked children. The dashboard now shows real populated data:

- **Parent Dashboard panel**: Displays the child's avatar, name, and the note "Child manages their own account."
- **Statistics panel**: Shows live counters in this example, 24 Solutions Submitted, 4 Courses Started, 64 Total Stars, and 2 Courses Created.
- **Current Course**: Displays the child's currently active course with its cover image, name, and star count (e.g. Digital Literacy 39 stars).
- **Time Spent on Site**: The bar chart now shows actual daily usage data over the past 7 days.
- **Game Progress**: Lists each course the child has engaged with, showing stars earned and number of activities completed per course (e.g. Digital Literacy: 39 stars, 15 activities, 15 levels completed; Coding Chatbots: 9 stars, 3 activities, 3 levels completed; Data is Everywhere: 13 stars, 5 activities, 5 levels completed).

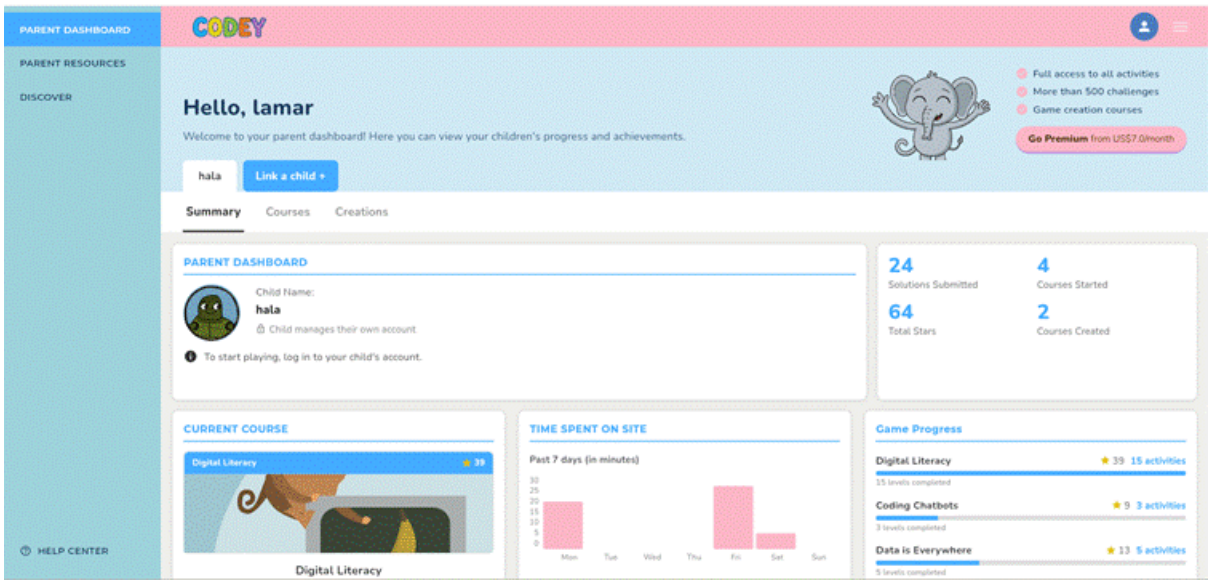


Figure 4.23: Game Progress: Lists each course the child has engaged with, showing stars earned and

4.5.4 Parent Resources Screen

The Parent Resources page is accessible from the left sidebar. It is divided into two sections:

Find Resources by Type offers three cards:

- **All Solutions**: View 3-star solutions for all challenges, with a "Search Solutions" link.
- **Coding Concepts**: Explore coding concepts to enhance understanding, with a "Search coding concept" link.
- **Videos**: Access a video collection to enhance understanding of various concepts, with a "Search videos" link.

More Resources offers three additional cards:

- **Coloring Pages**: Download coloring pages for an enjoyable classroom activity, with a "Download coloring pages" link.
- **Help Center**: Find articles on any aspect of the platform, with a "Go to the help center" link.
- **Media Kit**: Access and download a collection of character images, with an "Open media kit" link.

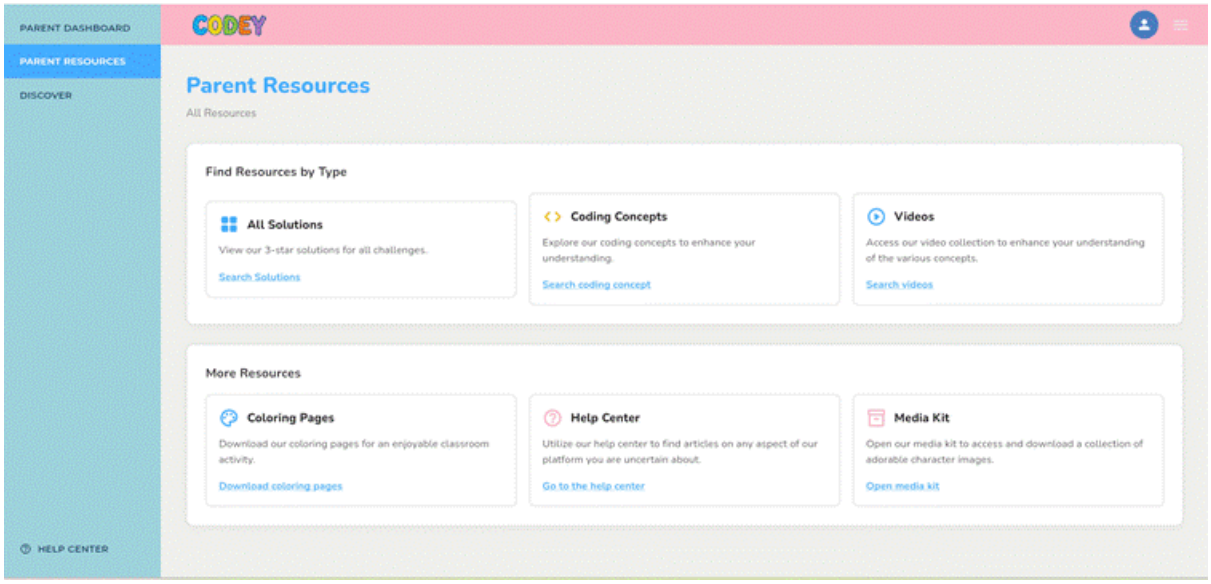


Figure 4.24: Media Kit: Access and download a collection of character images, with an "Open media

4.5.5 Profile Menu

Clicking the profile icon in the top right corner of the parent dashboard opens a small dropdown menu showing the parent's display name, avatar, and two options: **My Account** and **My Profile**.

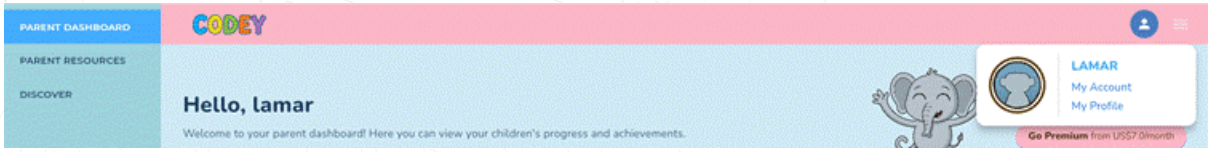


Figure 4.25: Profile Menu

4.5.6 My Account Screen

The My Account screen is organized into four sections:

User Details: Displays the parent's avatar, display name, role (Parent), an Edit button to modify the display name, and a badge indicating the sign-up method (e.g. "Signed Up With Google").

Linked Children: Lists all children linked to the parent account, showing each child's avatar, name, role (Child), and a note "Child manages their own account." An **Unlink** button is available next to each child to remove the link.

Manage Subscriptions: Displays the current subscription in a table showing Subscription Type, Billing Plan, Valid From, Valid Until, and Status. An expired trial shows a red "Expired" status badge and a pink warning banner: "Your trial is over. Please Review our pricing plans." A "Have a redemption code?" link is available on the right.

Settings: Two toggleable checkboxes allow the parent to control the child's experience:

- "Hide Discover and prevent child from publishing games and challenges."
- "Turn off Coding Adventure's Super Hints."

Delete Account: A danger zone section with a clear warning explaining that deleting the account is irreversible, will cancel any auto-renewal, and will automatically unlink all linked children (without deleting their accounts). A **Delete My Account** button is provided to confirm the action.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

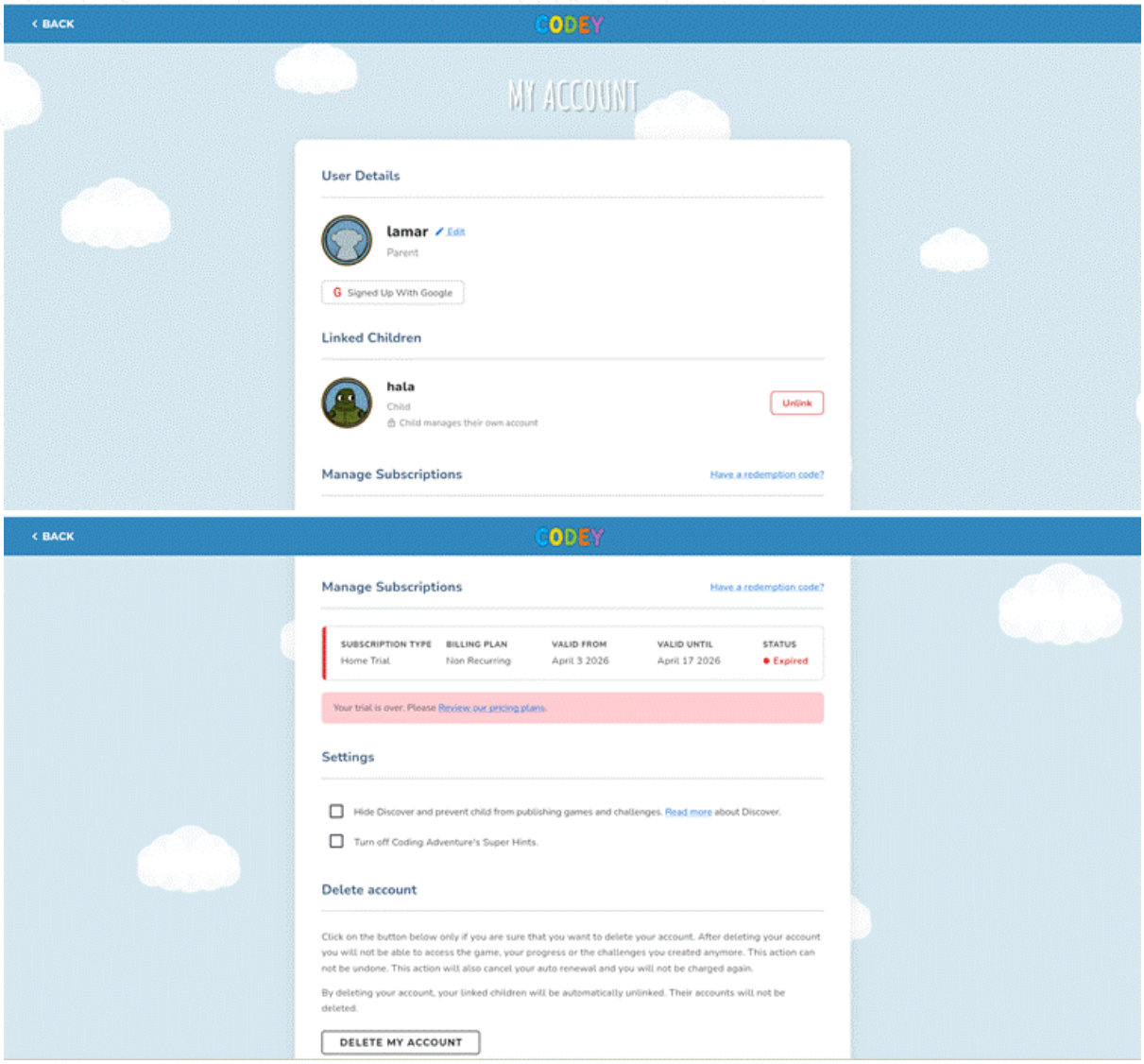


Figure 4.26: "Turn off Coding Adventure's Super Hints."

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.6 Student Home Screen (Courses)

After logging in, the student lands on the main Courses screen. The left sidebar contains four navigation items: Courses (currently active), My Creations, Discover, and My Classroom. A Day Streak counter with a flame icon is displayed at the bottom of the sidebar, tracking consecutive daily activity. A Help Center link is also available at the bottom.

Email Verification Banner: If the student has not yet verified their email, a red banner appears at the top of the page with the message "Please verify your email (email address)" and a Send Email button to resend the verification email, along with an X button to dismiss it.

Welcome Banner: The top of the main content area shows a personalized welcome message ("Welcome, hala!") with the student's avatar on the left. On the right side of the banner, a recommended course is displayed showing the course name, completion progress (e.g. "2 of 2 lessons completed" with a 100% circular progress indicator), and a Start Coding button. Below the banner, a Share your progress with a parent link is shown on the left and a Generate Code button on the right, allowing the student to generate their parent-linking code.

Onboarding Tooltip System: First-time students are guided through the dashboard via a series of contextual tooltip bubbles that appear one at a time, each dismissible with a "Got it ✓" button:

- "New student? Start here!" — advises opening the filter and selecting "Beginner" level to find courses for first-time coders.
- "Use filters to find the right course" : explains filtering by level, topic, or category.
- "All your courses are here" : explains that tapping any course card shows its lessons, and that more courses unlock as the student progresses.

Quick Tips Modal: A question mark button (?) in the bottom right corner opens a "Quick Tips Dashboard" modal that provides a full overview of the platform's features:

- New Student? Start with Beginner Courses : filter by Beginner level for first-time coders.
- Use Filters to Find Your Perfect Course : filter by level, topic, or category.
- Discover Content from Other Students :browse games and challenges published by the community in the Discover tab.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

- Create Your Own Game in My Creations : design and build games using slides, top view, side view, or scratch style.
- Join a Classroom with Your Teacher's Code : enter a classroom code in the sidebar to join.
- Replay Tips :replays the onboarding tooltip bubbles again.

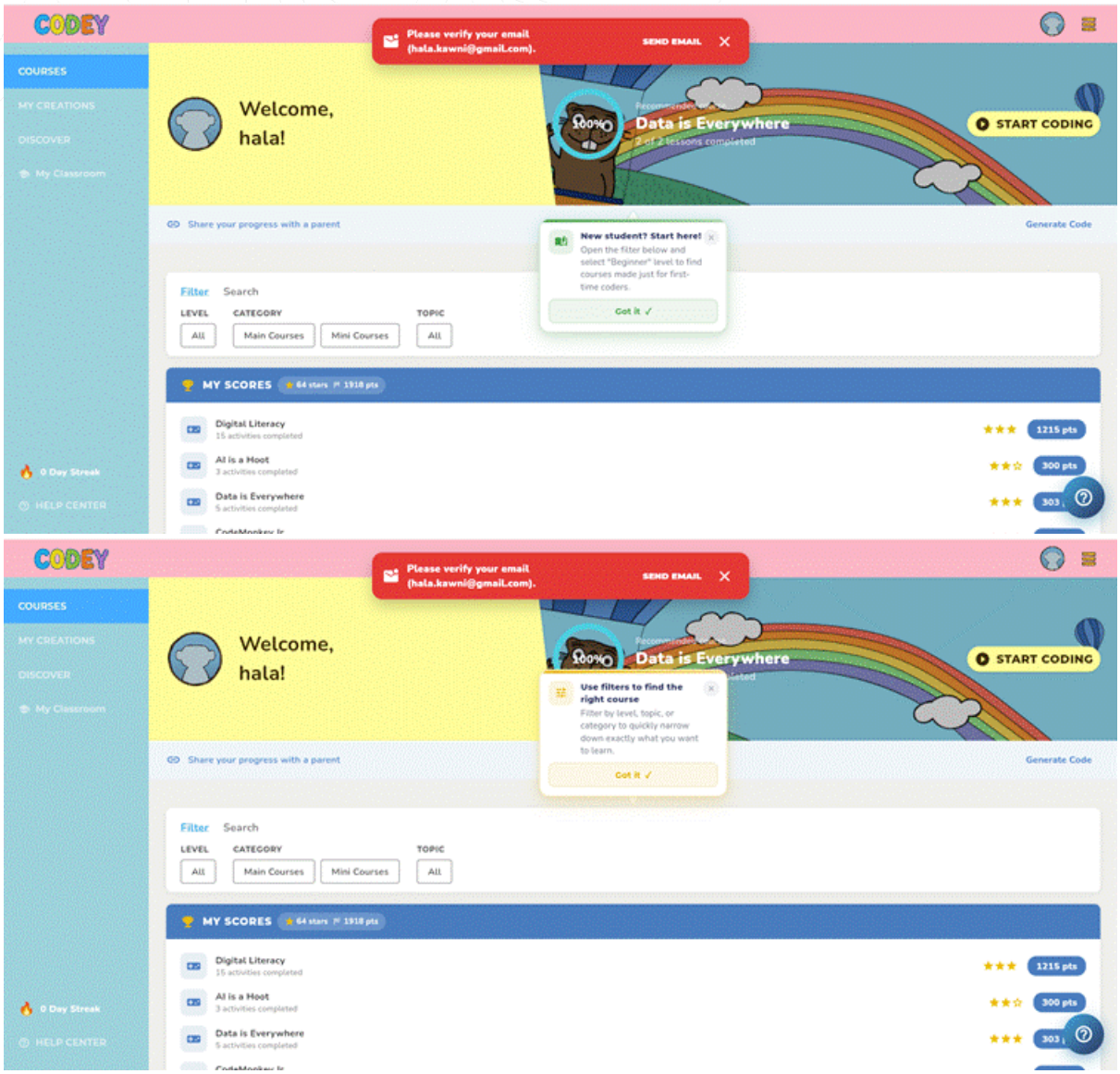


Figure 4.27: Replay Tips :replays the onboarding tooltip bubbles again

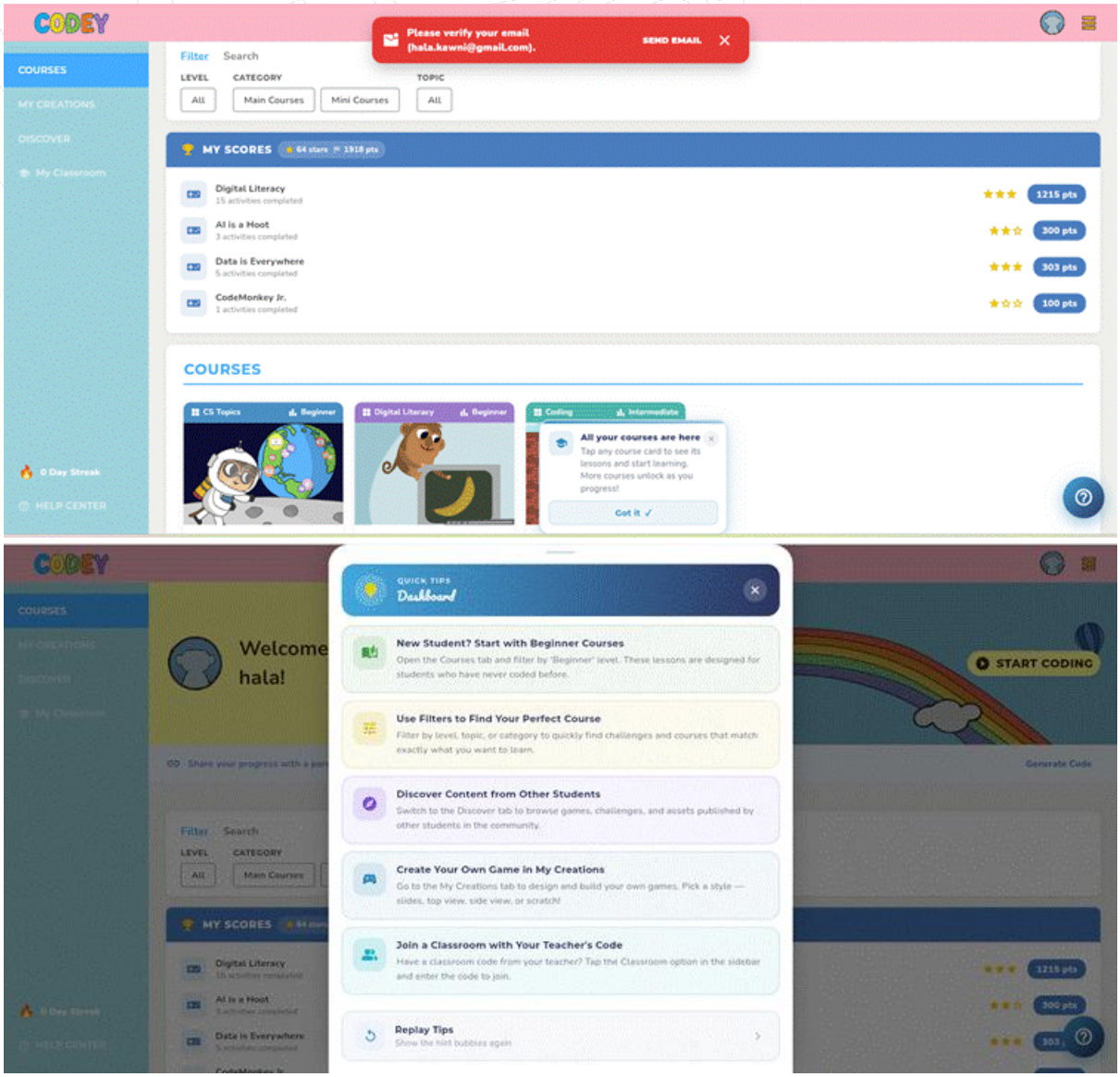


Figure 4.28: Replay Tips :replays the onboarding tooltip bubbles again - view 2

4.6.1 Filter and Search Bar

The Filter bar sits above the course catalog and has two tabs: **Filter** and **Search**. Clicking the Filter tab expands a detailed filtering panel with three columns and an **Apply** button on the right:

- **Level**: Four checkboxes Novice, Beginner, Intermediate, and Advanced with an "Unselect all" link to clear all selections at once.
- **Category**: Three checkboxes Main Courses, Mini Courses, and Seasonal Activities with a "Select all" link.
- **Topic**: Three checkboxes Coding, Digital Literacy, and CS Topics with an "Unselect all" link.

The top of the filter bar also shows the currently active filter selections as quick-toggle buttons (All for Level, Main Courses / Mini Courses for Category, All for Topic), giving students a compact view of their active filters without needing to expand the full panel. Once the desired filters are selected, the student clicks **Apply** to update the course catalog accordingly.

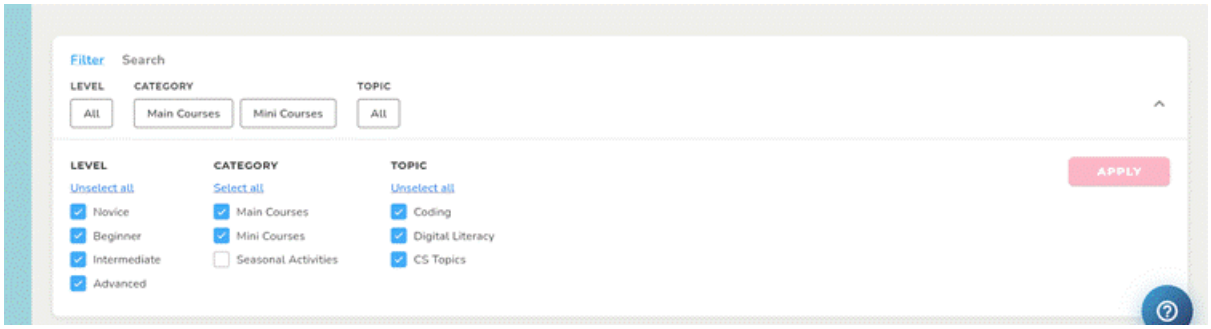


Figure 4.29: The top of the filter bar also shows the currently active filter selections as

4.6.2 Welcome Banner and Course Recommendation

The welcome banner at the top of the Courses screen is split into two halves. The left half displays a yellow background with the student's avatar and a personalized greeting ("Welcome, hala!"). The right half displays a colorful illustrated scene with a rainbow and shows the student's currently active or most recently worked-on course.

The right side of the banner serves as a dynamic course recommendation widget. It displays the label "Recommended course" above the course name, along with a circular progress indicator showing the completion percentage (e.g. 100% for a fully completed course), the number of lessons completed (e.g. "2 of 2 lessons completed"), and a **Start Coding** button to jump directly into the course.

The recommendation is not static once a student completes a course, the system automatically suggests the next most relevant course based on the student's activity history and learning preferences, guiding them along a personalized learning path without requiring them to manually browse the catalog.

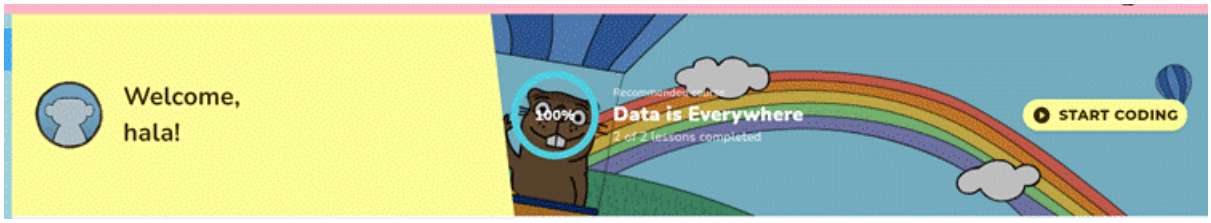


Figure 4.30: Welcome Banner and Course Recommendation

4.6.3 Avatar and Profile Picture

Students can personalize their profile by clicking on their avatar in the welcome banner. This opens a **Change Avatar** modal displaying a grid of pre-designed character avatars to choose from, including human characters of diverse appearances and animal characters. The currently selected avatar is highlighted with a gold border. Two action buttons are available at the bottom: **Upload Photo** (to use a custom image) and **Save** / **Cancel**.

If the student chooses to upload a custom photo, a **Fit Profile Photo** modal opens with a circular preview frame. The student clicks **Choose image** to select a photo from their device. Once an image is selected it appears inside the circular frame with the instruction "Drag or zoom until the photo fits nicely inside the round profile frame."

A zoom slider below the image allows the student to adjust the crop level dragging it right zooms in, allowing precise framing of the photo. Two additional buttons are available: **Replace image** (to select a different photo) and **Remove** (to delete the uploaded photo). Clicking **Apply** saves the cropped photo as the profile avatar, which then appears immediately in the welcome banner and in the top navigation bar, replacing the default illustration.

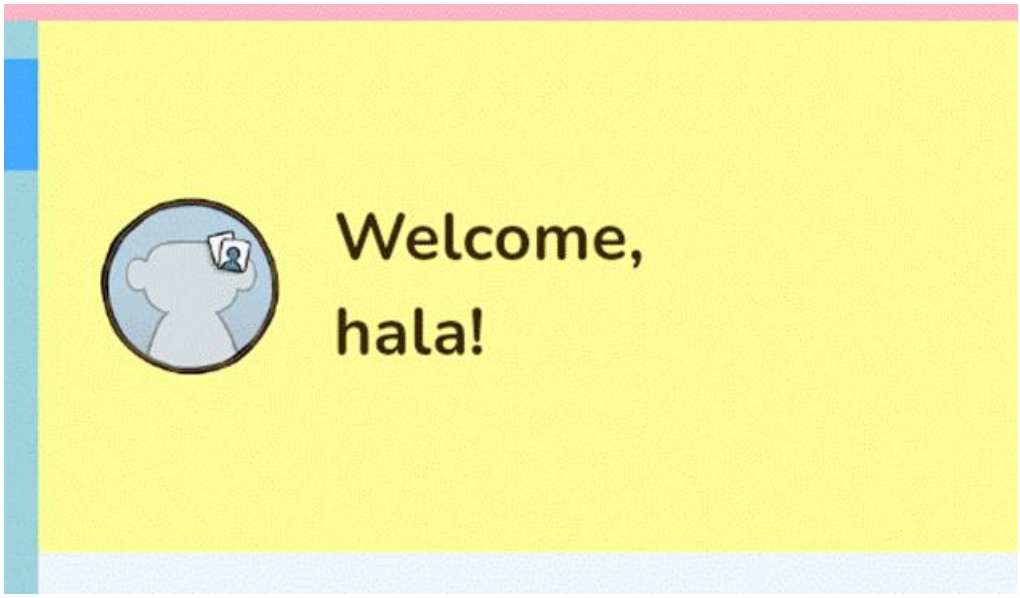


Figure 4.31: Avatar and Profile Picture

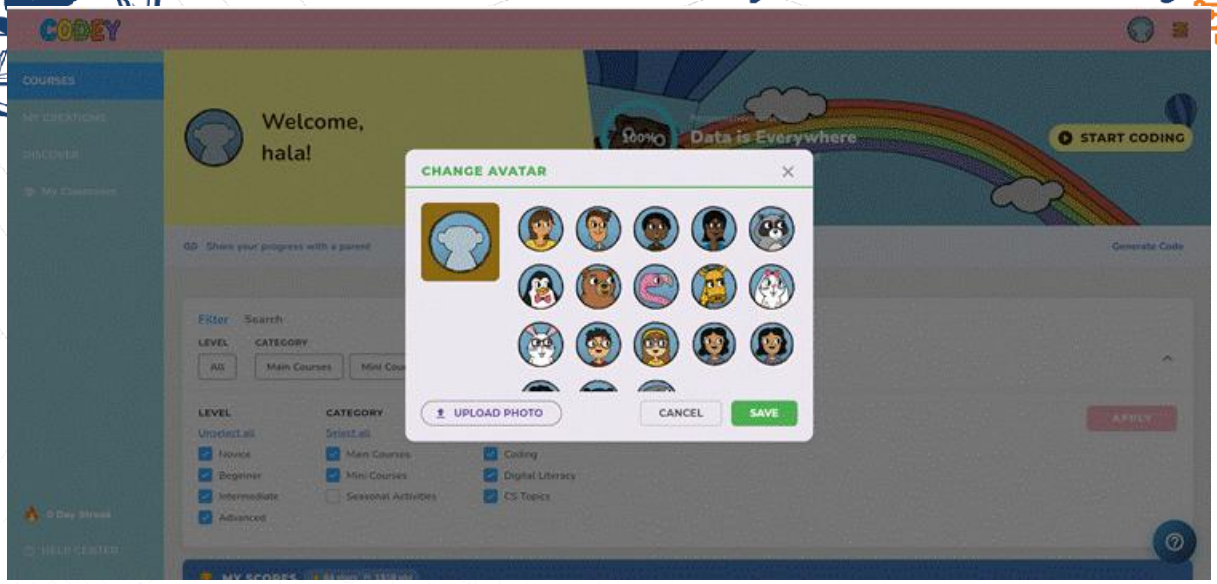


Figure 4.32: Avatar and Profile Picture - view 2

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

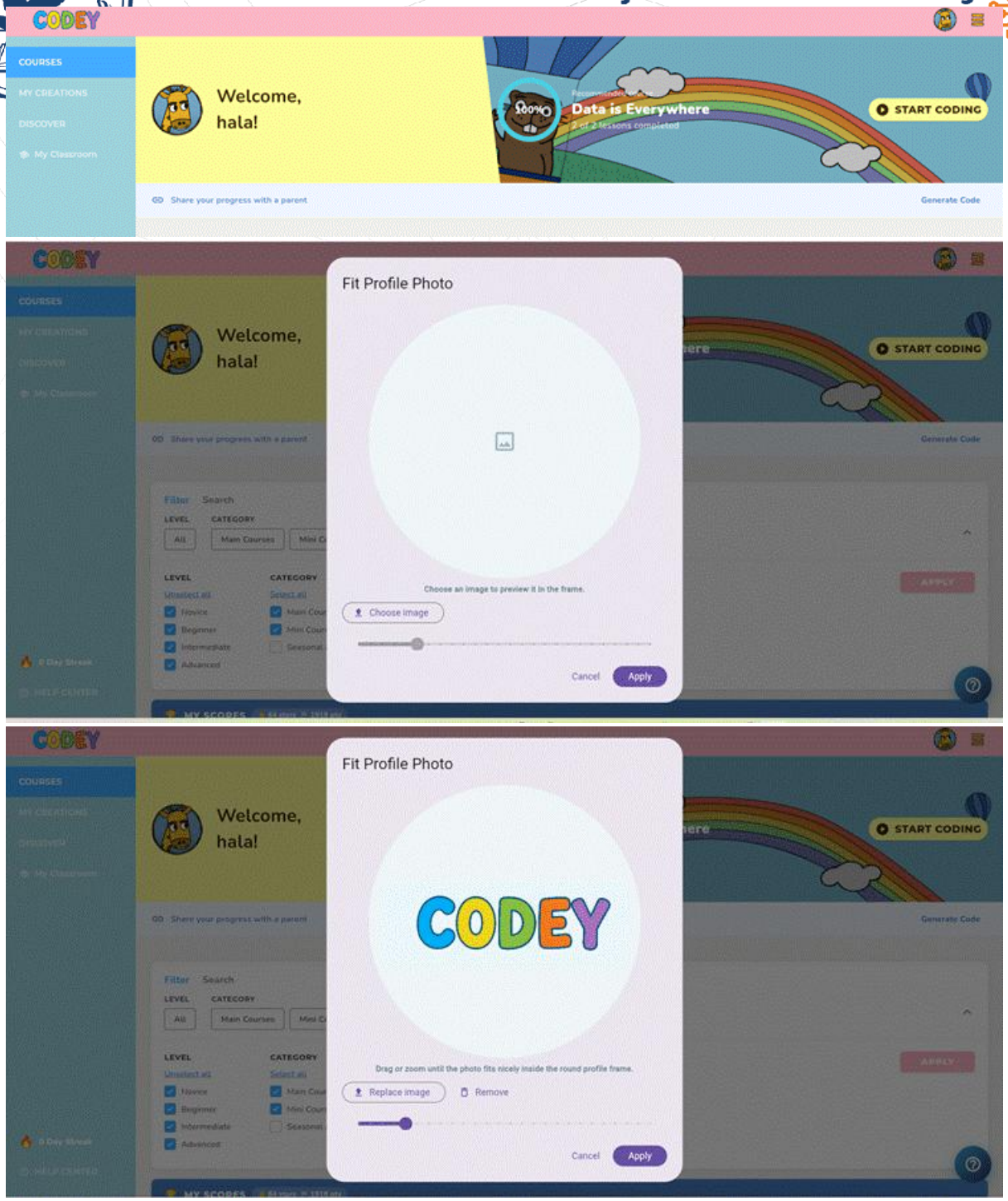


Figure 4.33: Avatar and Profile Picture - view 3

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

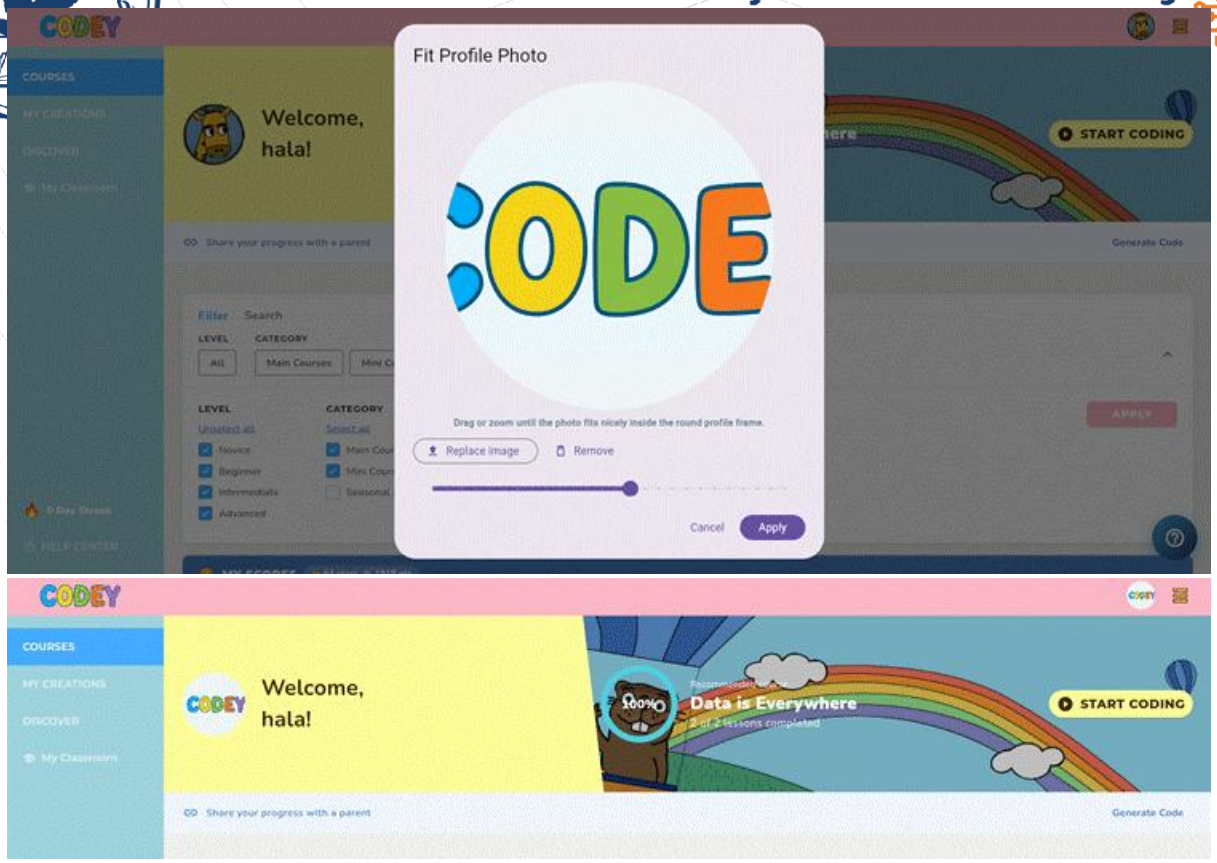


Figure 4.34: Avatar and Profile Picture - view 4

4.6.4 My Scores Section

Below the welcome banner, a blue "MY SCORES" panel displays the student's total accumulated stars and points across all courses (e.g. 64 stars, 1918 pts). Each course the student has engaged with is listed as a row showing the course name, number of activities completed, a 3-star rating indicator reflecting performance, and a points badge:

- Digital Literacy — 15 activities completed — ★★★ — 1215 pts
- AI is a Hoot — 3 activities completed — ★★☆ — 300 pts
- Data is Everywhere — 5 activities completed — ★★★ — 303 pts

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

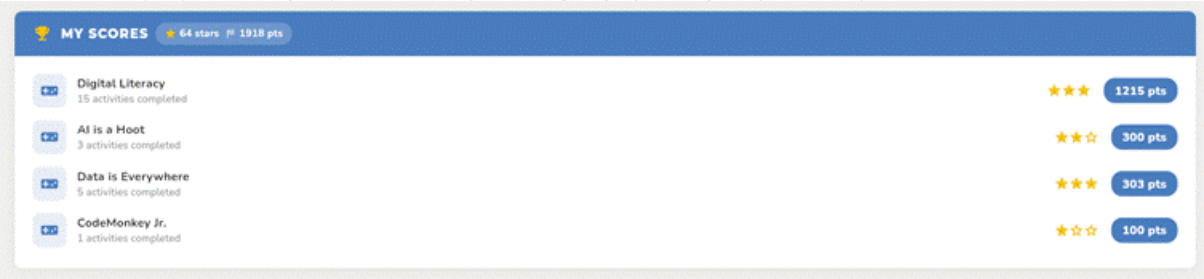


Figure 4.35: Data is Everywhere — 5 activities completed — ★★★ — 303 pts

4.6.5 Courses Catalog

Below the scores section, a "COURSES" section displays the available built-in courses as cards in a horizontally scrollable row. Each card shows:

- A colored category tag at the top left (e.g. CS Topics, Digital Literacy, Coding)
- A difficulty level badge at the top right (Beginner, Intermediate)
- A course illustration in the center
- The course name and subtitle below (e.g. "Data is Everywhere , Functions & Variables", "Digital Literacy , Internet Safety", "AI is a Hoot ,AI & Logic")

The three visible courses are **Data is Everywhere** (CS Topics, Beginner), **Digital Literacy** (Digital Literacy, Beginner), and **AI is a Hoot** (Coding, Intermediate).

4.7 Digital Literacy Course

4.7.1 Course Preview Modal

Clicking on the Digital Literacy course card opens a preview modal with an orange header displaying the course title "Digital Literacy: Internet Safety." The modal shows the course illustration, a status badge ("Not started" in yellow), and a description: "A short introduction to some important topics in the digital world: How to use computers, what are software and hardware, possible threats online and protecting your privacy." Left and right arrow buttons allow browsing between course lessons within the modal. A green **Start Coding** button at the bottom launches the course.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

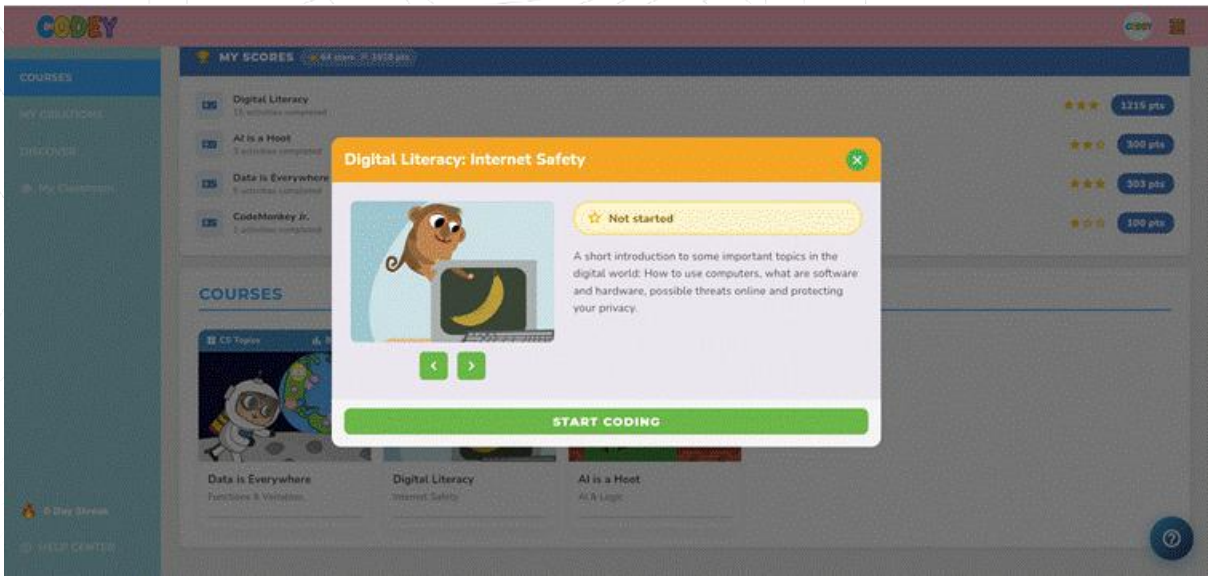


Figure 4.36: Course Preview Modal

4.7.2 Course Overview Screen

After clicking Start Coding, the student enters the Digital Literacy course overview screen, labeled "Digital Literacy: Mini Course" in the pink top bar. The screen is divided into two areas:

The **left panel** displays:

- A "3 Lessons" badge
- The course title "Digital Literacy"
- A short course description: "This course provides an overview of Digital Use and Digital Citizenship."
- A **Track Your Progress** semicircular gauge showing the student's current completion (e.g. 1/3 lessons completed, shown as a teal arc)
- A **Continue Lesson #2** button in yellow to resume from where the student left off

The **right area** displays all three lessons as cards laid out horizontally, each numbered and showing an illustration and lesson title:

- **#1 — Digital Use In A Nutshell**
- **#2 — Digital Citizenship In A Nutshell**

جـــامـــعـة النـجـــاح الـوطـنـيـة
An-Najah National University
Faculty of Engineering | كلية الهندسة
وحدة الجودة والاعتماد - مركز ABET
Quality and Accreditation Unit - ABET Center

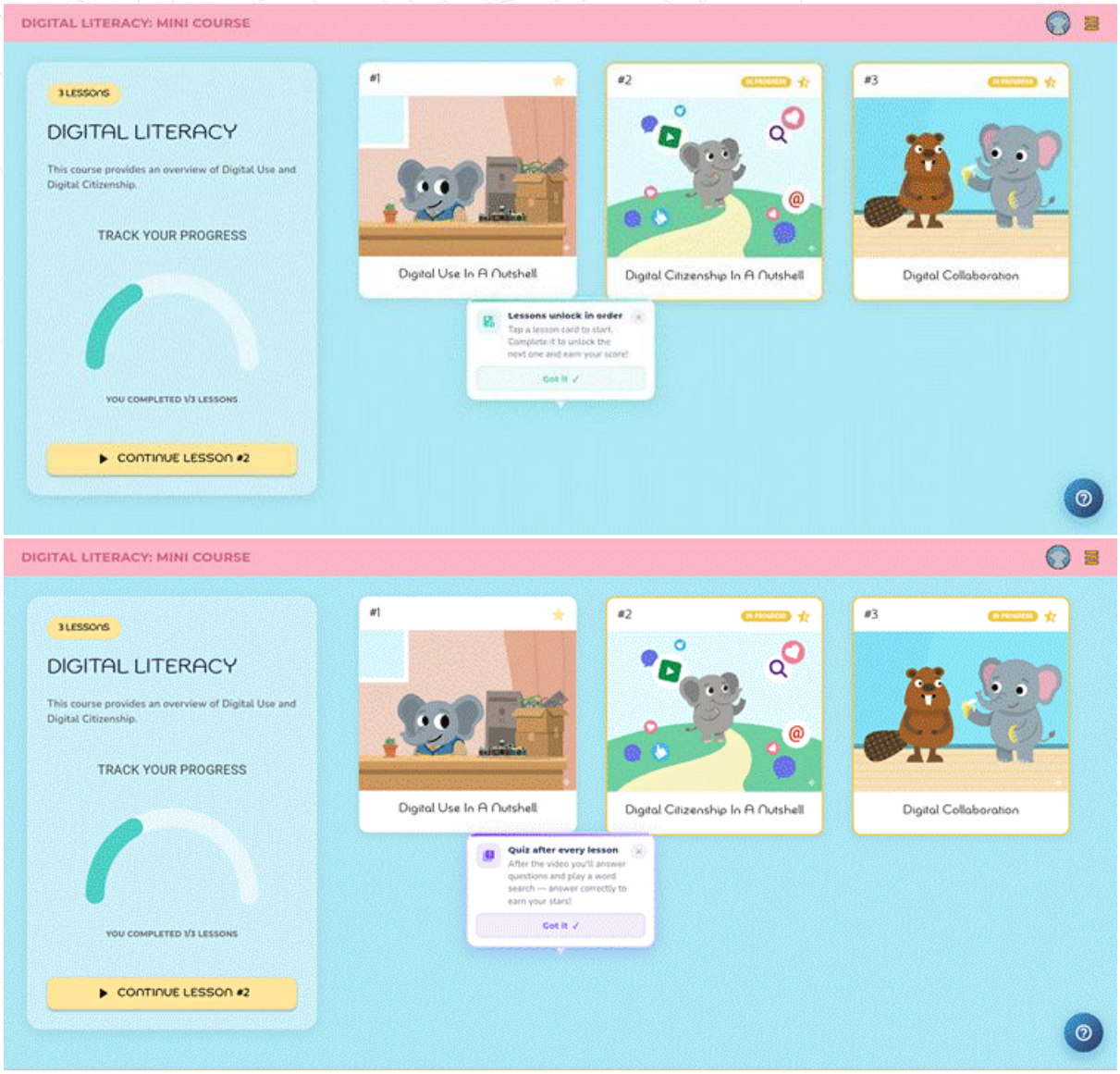


Figure 4.37: #3 — Digital Collaboration

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.7.3 Lesson Screen

Clicking on a lesson card opens the full lesson screen. The top navigation bar shows the lesson number and title (e.g. "#1 Digital Use In A Nutshell"), a Back to Course button on the top left, and a progress bar in the center showing all slides as small circular indicators with the current slide highlighted. On the top right, two locked tabs are shown: Play (0/3) and Review (0/5), which unlock after the student completes the lesson slides. At the bottom, a Listen Mode toggle allows text-to-speech read-aloud of slide content.

The lesson consists of 18 slides navigated using Previous and Next arrow buttons on the left and right sides of the screen. The current slide number is shown at the bottom center (e.g. "1 / 18"). Slides are presented on a purple background with a white or colored card in the center. The lesson covers topics including: what a computer is, hardware vs. software, how the internet and World Wide Web work, email fundamentals, file organization, and useful applications for students.

Slides use a variety of presentation styles some are text-based intro slides listing the lesson's topics, while others use a conversational dialogue format with illustrated animal characters (a monkey asking questions and a giraffe answering) alongside diagrams and visual metaphors to explain concepts in a child-friendly way.

Interspersed throughout the slides are two types of interactive questions:

- Corner Questions : Multiple-choice questions with four answer options displayed as teal buttons. The student selects an answer and the Next button becomes active. A star icon appears next to the correct answer upon selection, rewarding the student. Survey Questions : Opinion or experience-based questions presented in a 2×2 grid of teal answer tiles with a "Submit Your Answer" button. These have no right or wrong answer and are designed to engage the student personally with the topic.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

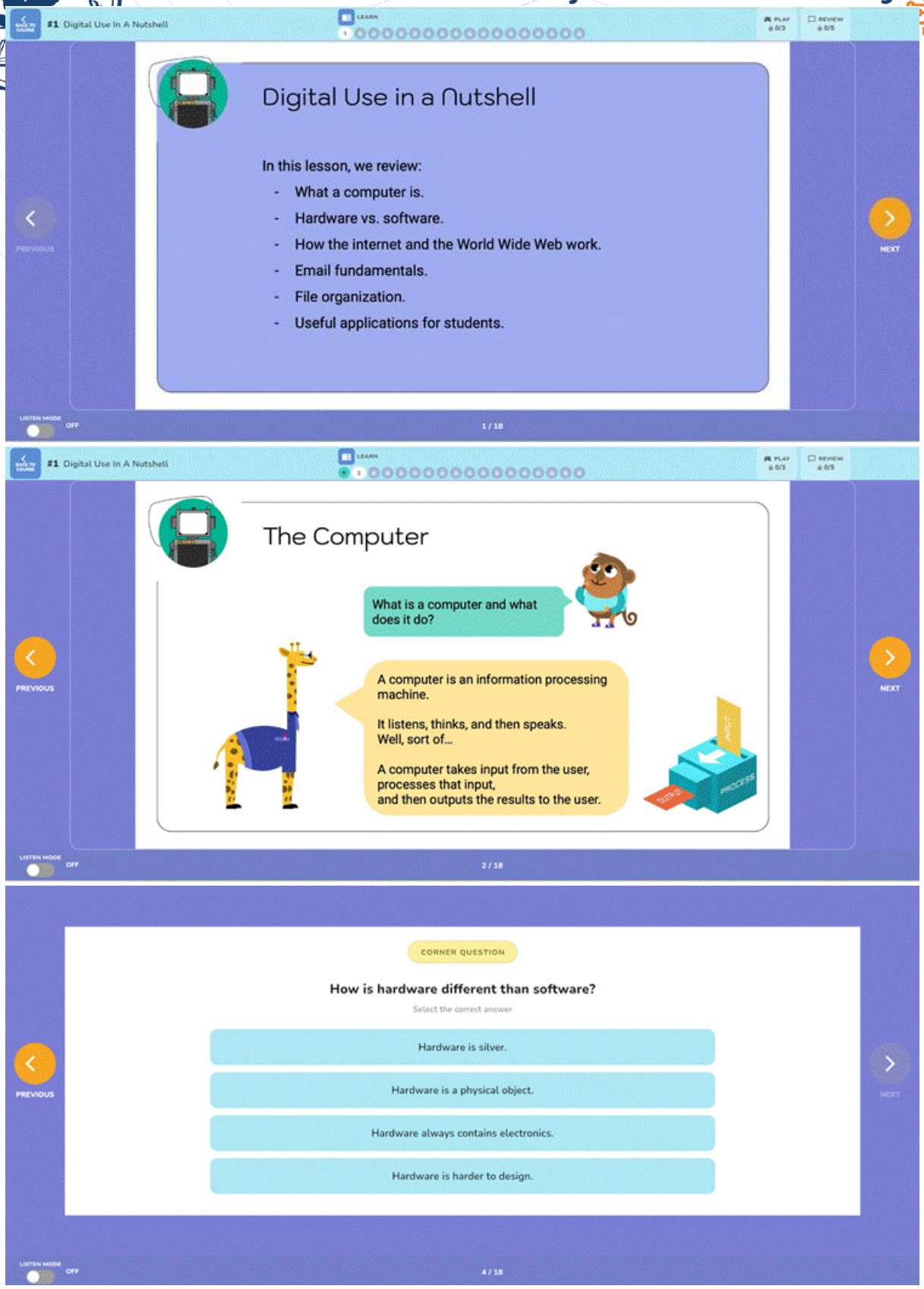


Figure 4.38: Corner Questions : Multiple-choice questions with four answer options displayed as teal

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

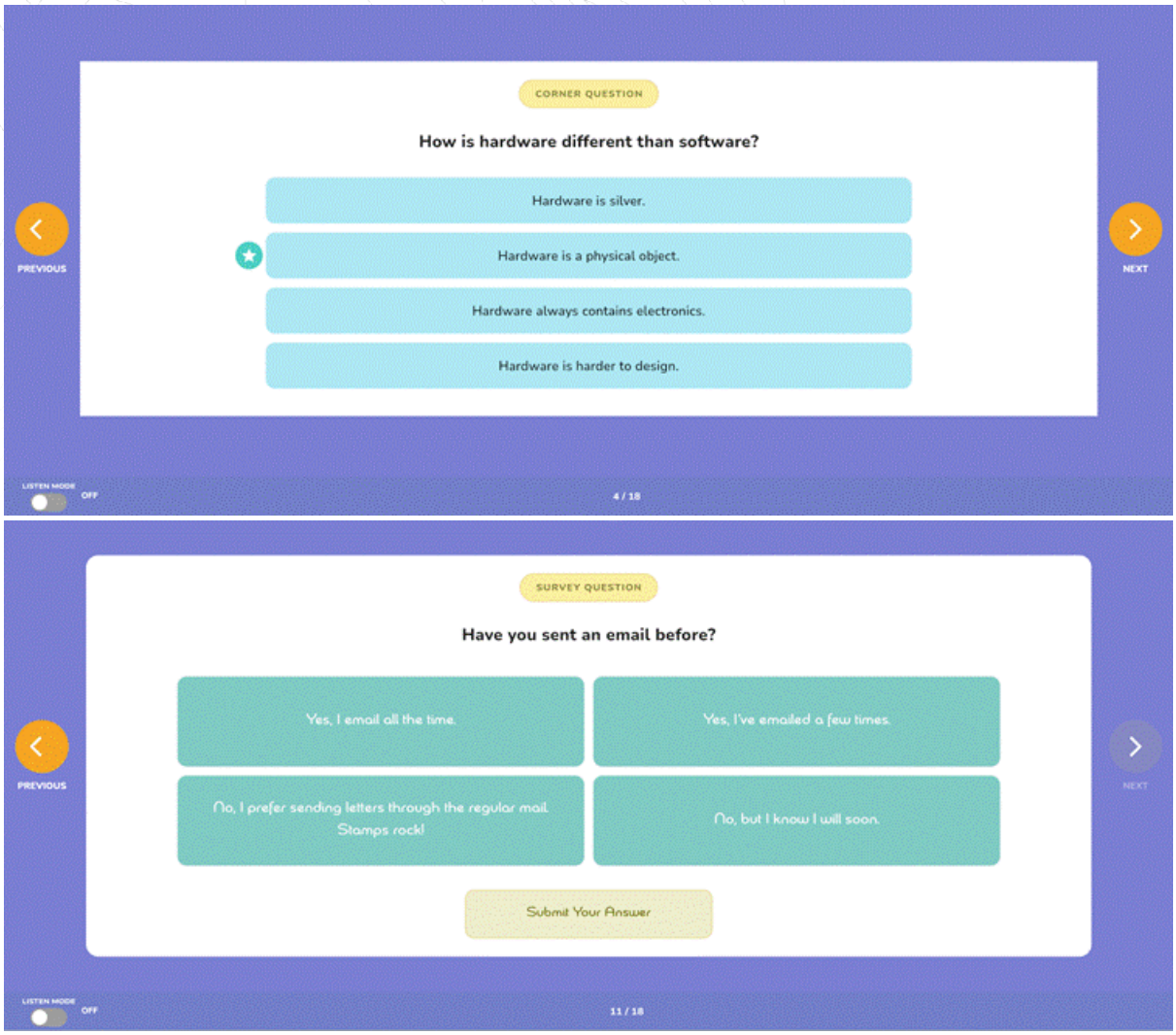
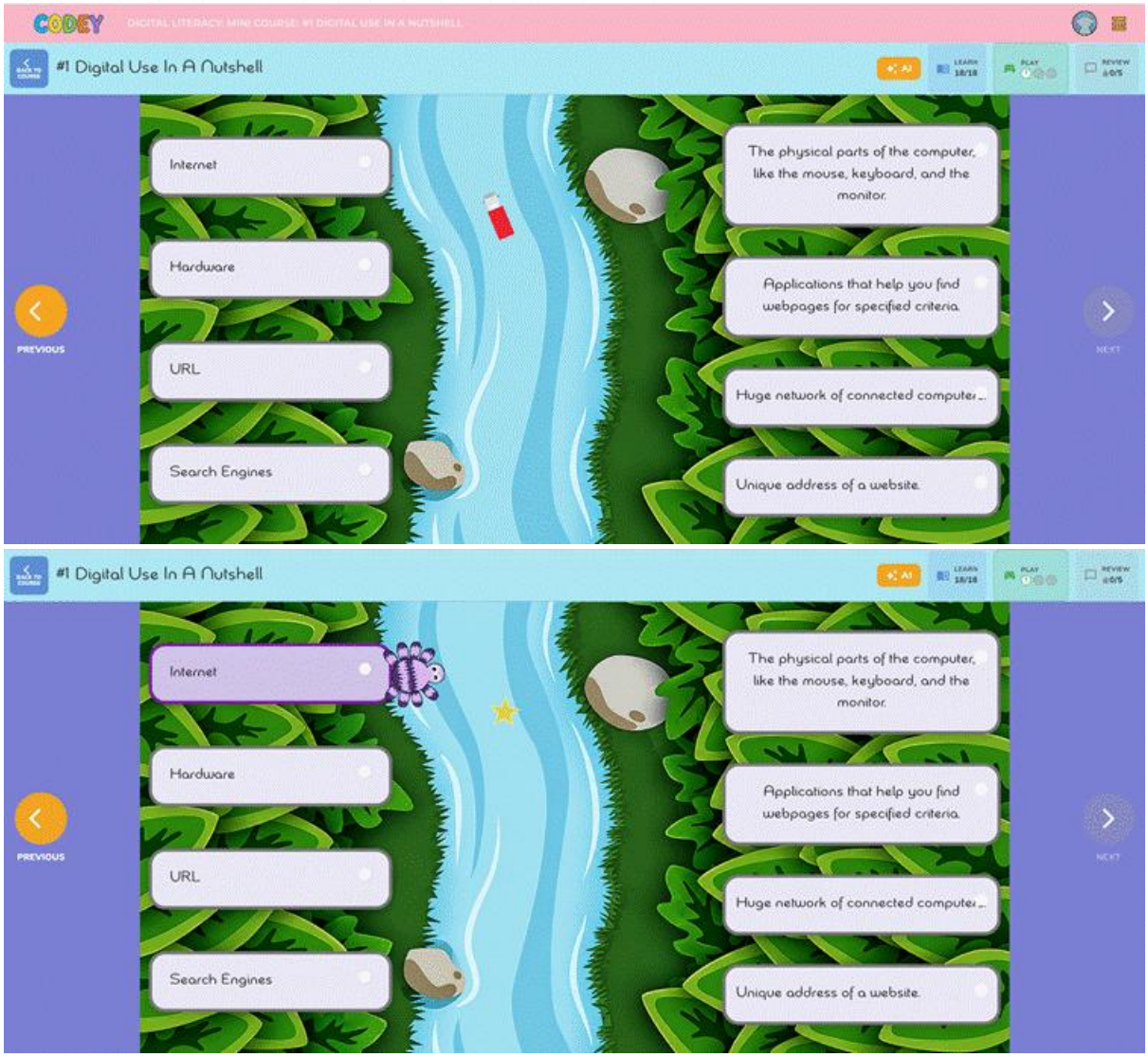


Figure 4.39: Corner Questions : Multiple-choice questions with four answer options displayed as teal - view 2

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.7.4 Play Section — Word Matching Game

After completing all 18 lesson slides, the **Play** tab unlocks. It presents a word matching game set against an illustrated top-down river and jungle scene. Terms appear on the left column (e.g. Internet, Hardware, URL, Search Engines) and their definitions on the right. The student connects each term to its correct definition by clicking both cards. Correct matches turn green with a connecting line; incorrect ones flash red briefly. A coin and star animation plays with each correct match, and the Play counter in the top bar updates in real time. Upon completing all matches, a " Great Job! You matched all the words correctly!" popup appears with a **Continue** button.



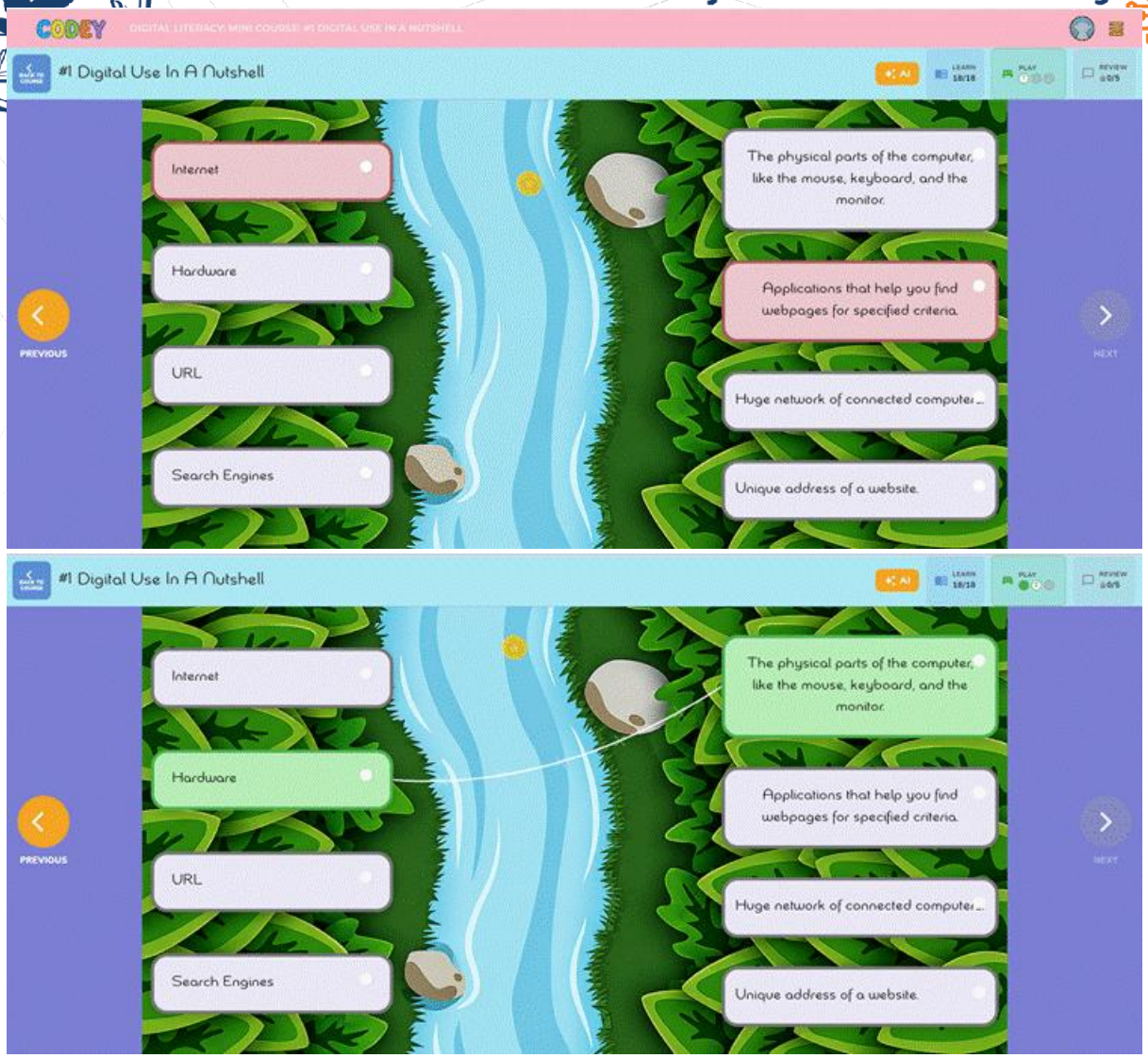


Figure 4.40: Play Section — Word Matching Game

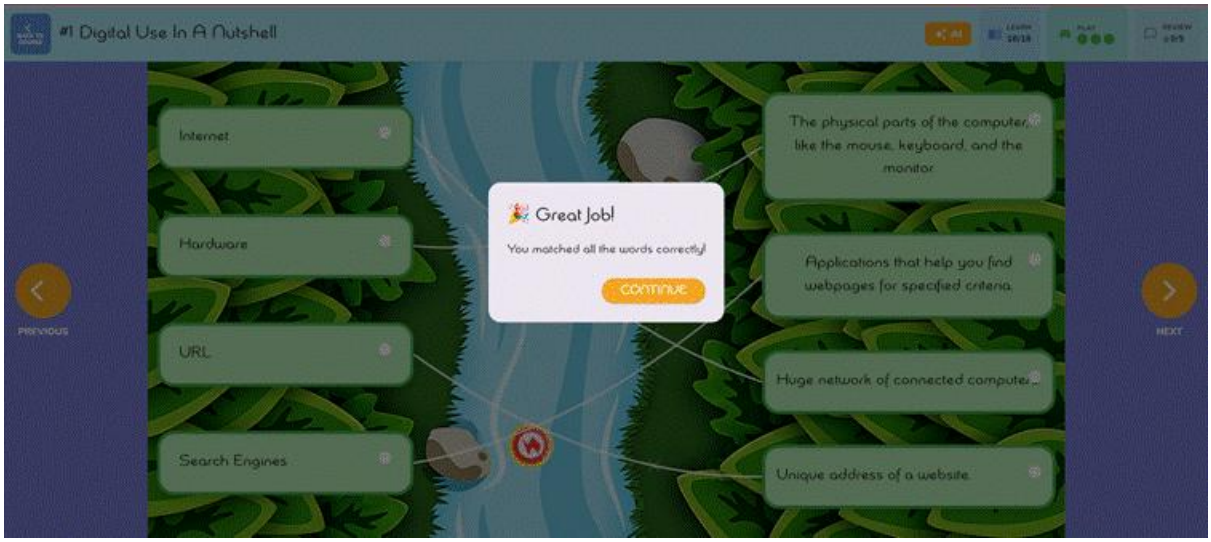


Figure 4.41: Play Section — Word Matching Game - view 2

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

An orange **AI** button in the top navigation bar allows regenerating the word pairs from the lesson content at any time. When new pairs load, a green banner confirms: " New pairs generated from lesson!"

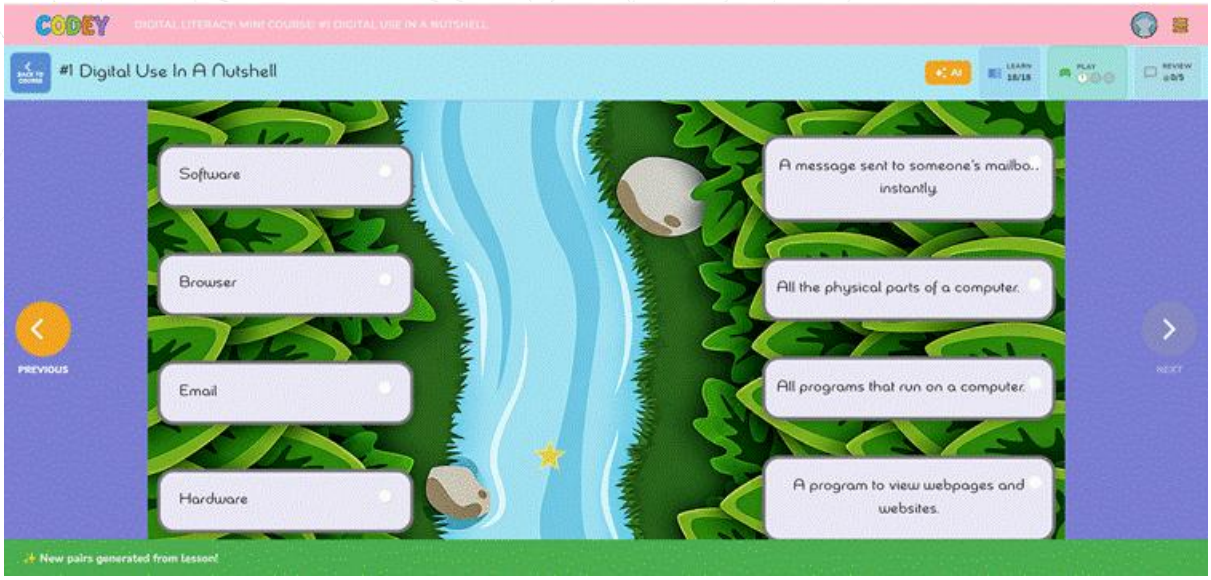


Figure 4.42: Play Section — Word Matching Game - view 3

4.7.5 Review Section — Fill in the Blanks

After the Play section, the **Review** tab unlocks. It presents a fill-in-the-blank activity where sentences derived from the lesson content contain dark blank slots that the student fills by selecting words from a **Word Bank** at the bottom of the screen. The student can either tap a word from the bank then tap a blank, or drag and drop directly. A filled counter in the top right tracks progress.

A small owl mascot at the bottom left provides contextual hints suggesting a word when the student is stuck . When a correct answer is placed, the blank turns green with a checkmark and the mascot responds with encouragement . The used word disappears from the bank and the counter increments.

The activity is split into multiple pages, navigated with Previous and Next arrows on the right side panel.

Four buttons are available on the right sidebar:

- ← / → navigate between pages
- ↺ shuffle/reset the word bank
- AI regenerate new questions based on the lesson slide content
- Hint reveal a hint for the currently selected blank

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

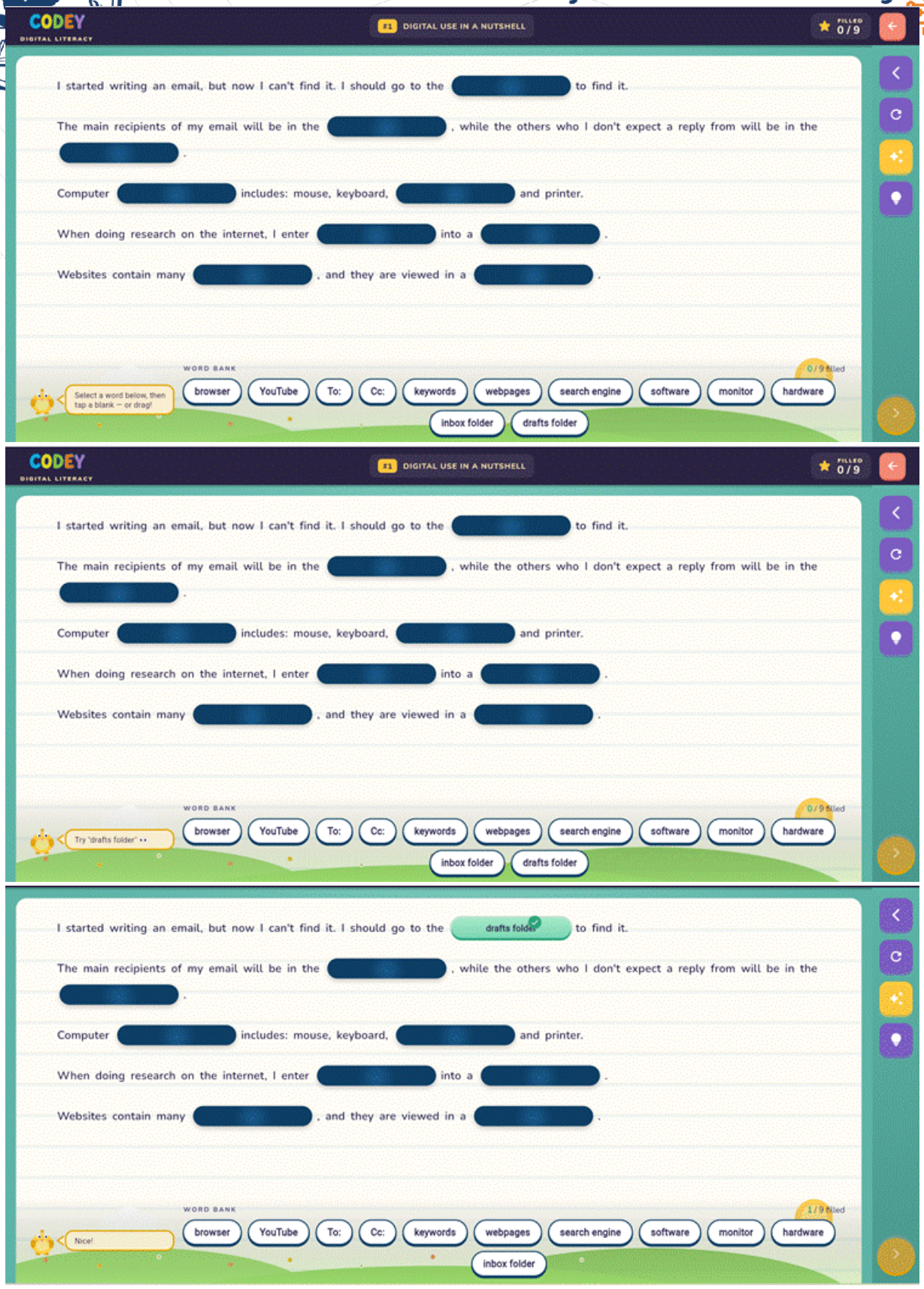


Figure 4.43: Hint reveal a hint for the currently selected blank

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

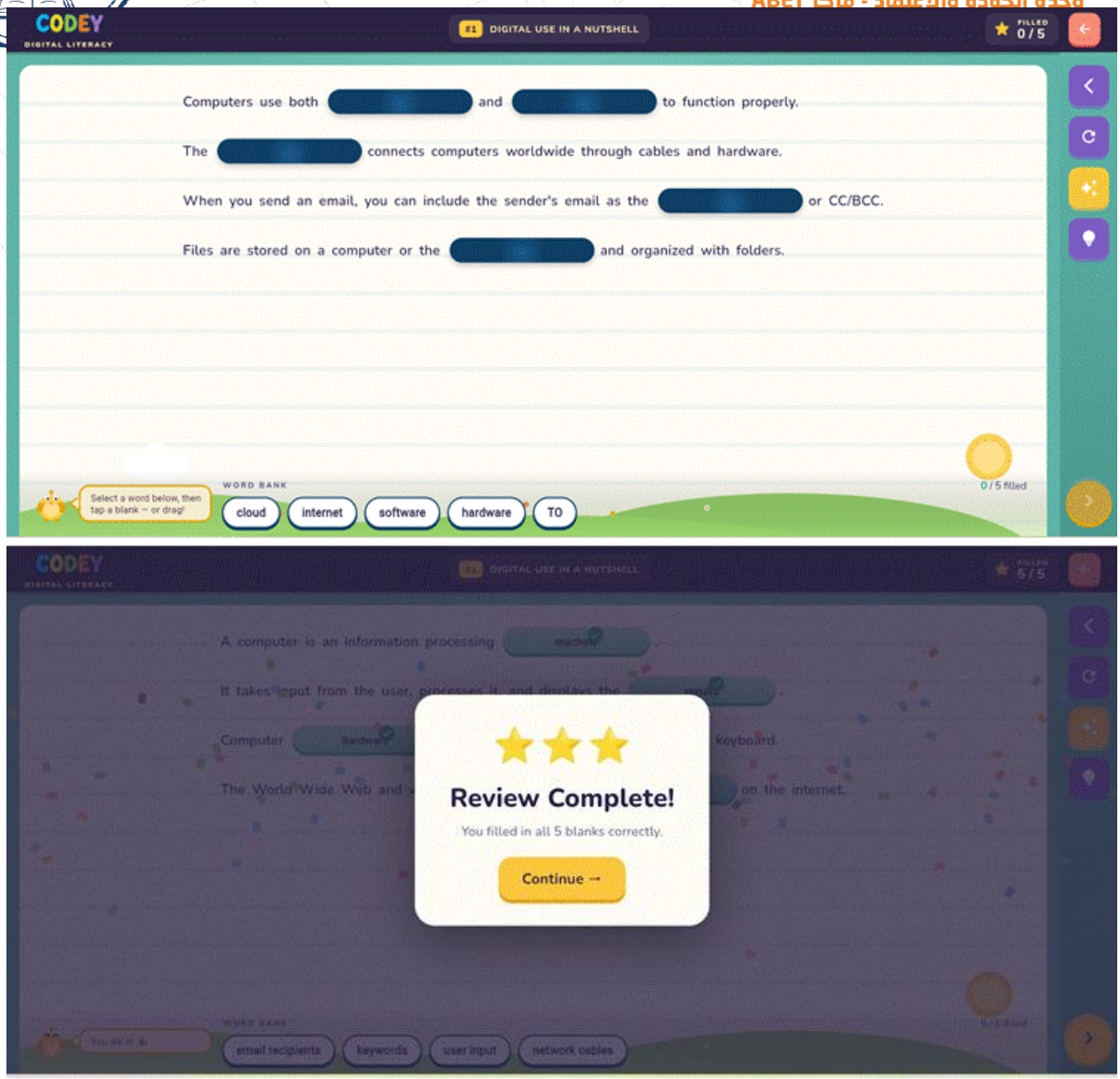


Figure 4.44: Hint reveal a hint for the currently selected blank - view 2

4.7.6 Word Search Game

After the fill-in-the-blank, a Word Search game is presented with a 10×10 circular letter grid. A purple panel on the right lists 6 words to find from the lesson vocabulary. The student drags across letters in any direction to find them found words get struck through and the counter increments. The owl mascot provides encouragement with each find. The AI button regenerates a fresh set of words and grid from the lesson content at any time.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.45: Word Search Game

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

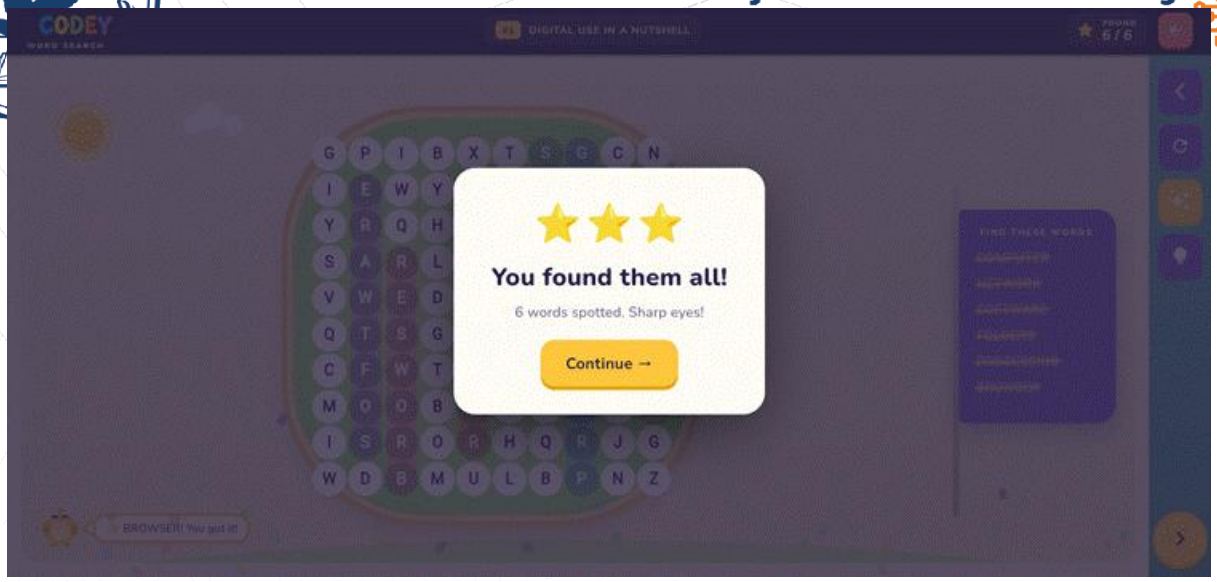


Figure 4.46: Word Search Game - view 2

4.7.7 Review Questions

After the word search, the Review tab presents multiple-choice questions labeled "Review Question #1", "#2", etc. Each question is displayed in a light blue card with four lettered answer options (A–D). The student selects an answer and clicks Submit Your Answer. Upon submission, the chosen answer highlights red with an ✗ if wrong, while the correct answer highlights green with a ✓. The Next button then activates to proceed. The AI button regenerates fresh questions from the lesson content at any time.

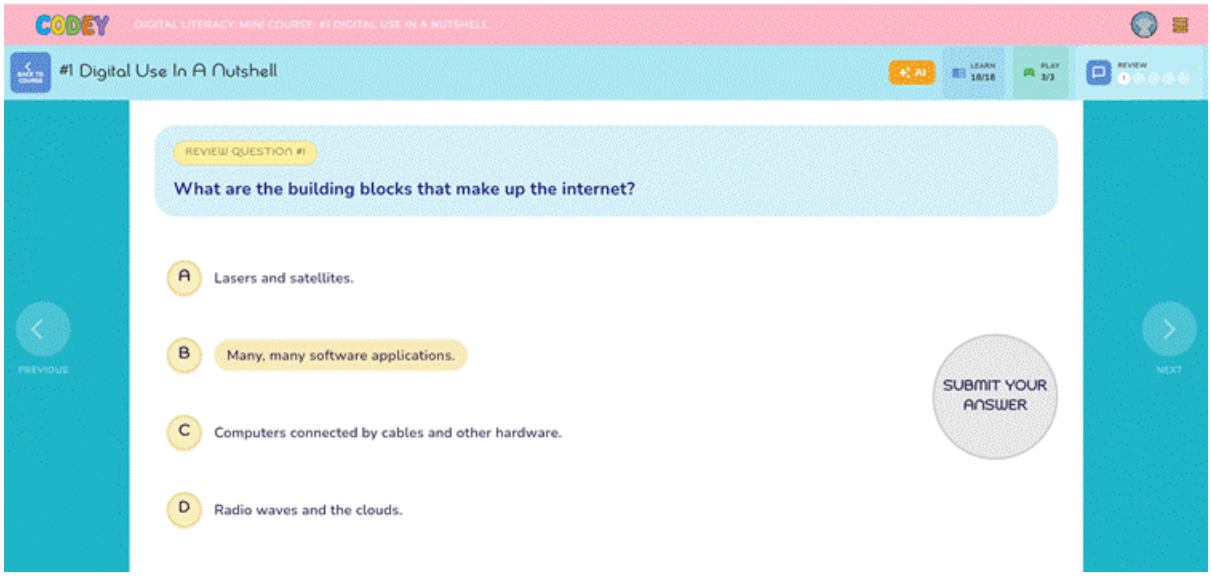


Figure 4.47: Review Questions

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

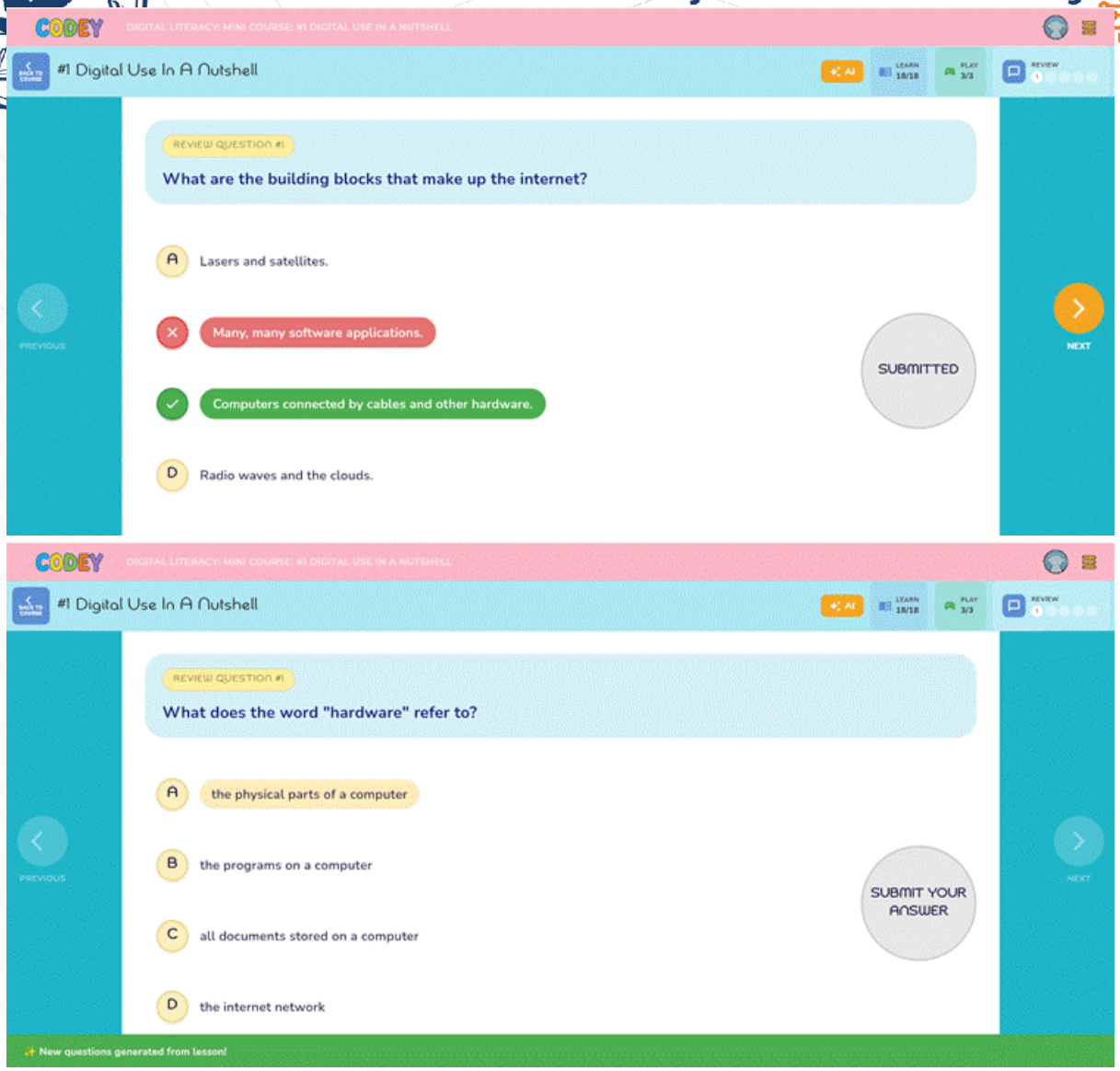


Figure 4.48: Review Questions - view 2

4.7.8 Lesson Completion Screen

After answering all 5 review questions (Review 5/5), a completion screen appears on a teal background with a "GREAT JOB!" message, the lesson title, and the review score . Two buttons are provided: Back to course and Next lesson, allowing the student to either return to the course overview or proceed directly to the next lesson. All three tab counters in the top bar (Learn 18/18, Play 3/3, Review 5/5) are now complete and highlighted green.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

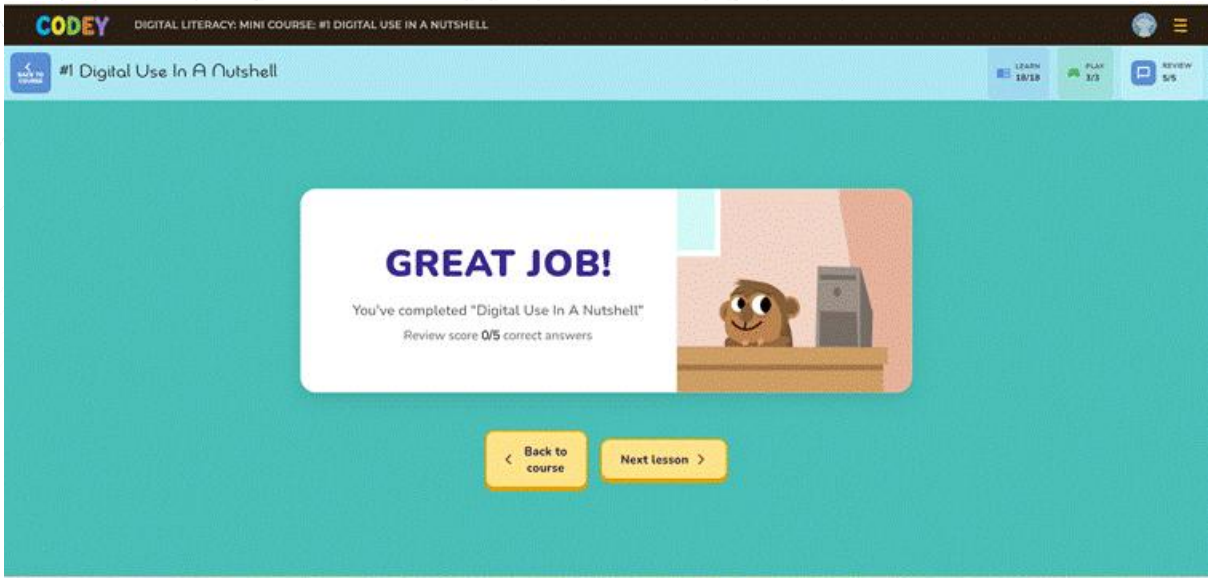


Figure 4.49: Lesson Completion Screen

4.7.9 Lesson 2 — Digital Citizenship In A Nutshell

Lesson 2 follows the same Learn → Play → Review structure as Lesson 1, consisting of 17 slides. The lesson covers three main topics: how to be a responsible, safe, and friendly digital citizen; online threats such as phishing, fake news, and cyberbullying; and choosing strong passwords to protect personal accounts.

Slides use the same illustrated dialogue format with the monkey and giraffe characters. Interspersed throughout are an open question asking students about their offline activities, a survey question about familiarity with phishing, and a corner question testing comprehension of why websites require passwords.

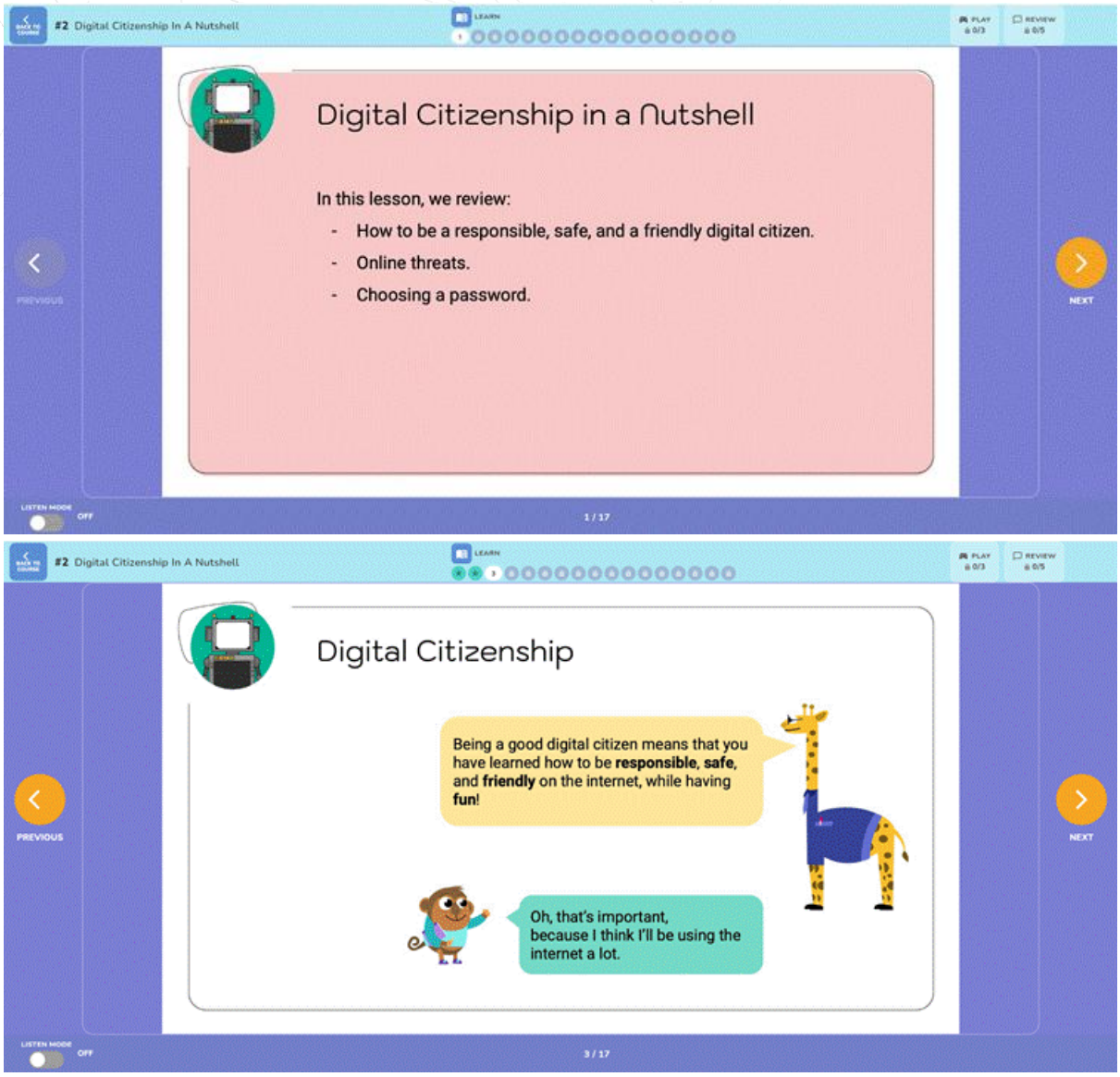


Figure 4.50: Lesson 2 — Digital Citizenship In A Nutshell

4.7.10 Play Section — Pixel Guard: Cyber Inbox

After completing the slides, the Play tab unlocks with a game called Pixel Guard: Cyber Inbox. The student is shown 7 notifications one at a time on a simulated phone screen, each representing a real-world online scenario such as cyberbullying, fake news, or digital balance. The student judges each notification as safe or risky using the green Allow or red Block buttons at the bottom. A combo multiplier rewards consecutive correct answers. Upon finishing, a Mission Clear screen shows the student's grade, accuracy, and total points. An AI Mode button regenerates a fresh set of scenarios from the lesson content at any time.

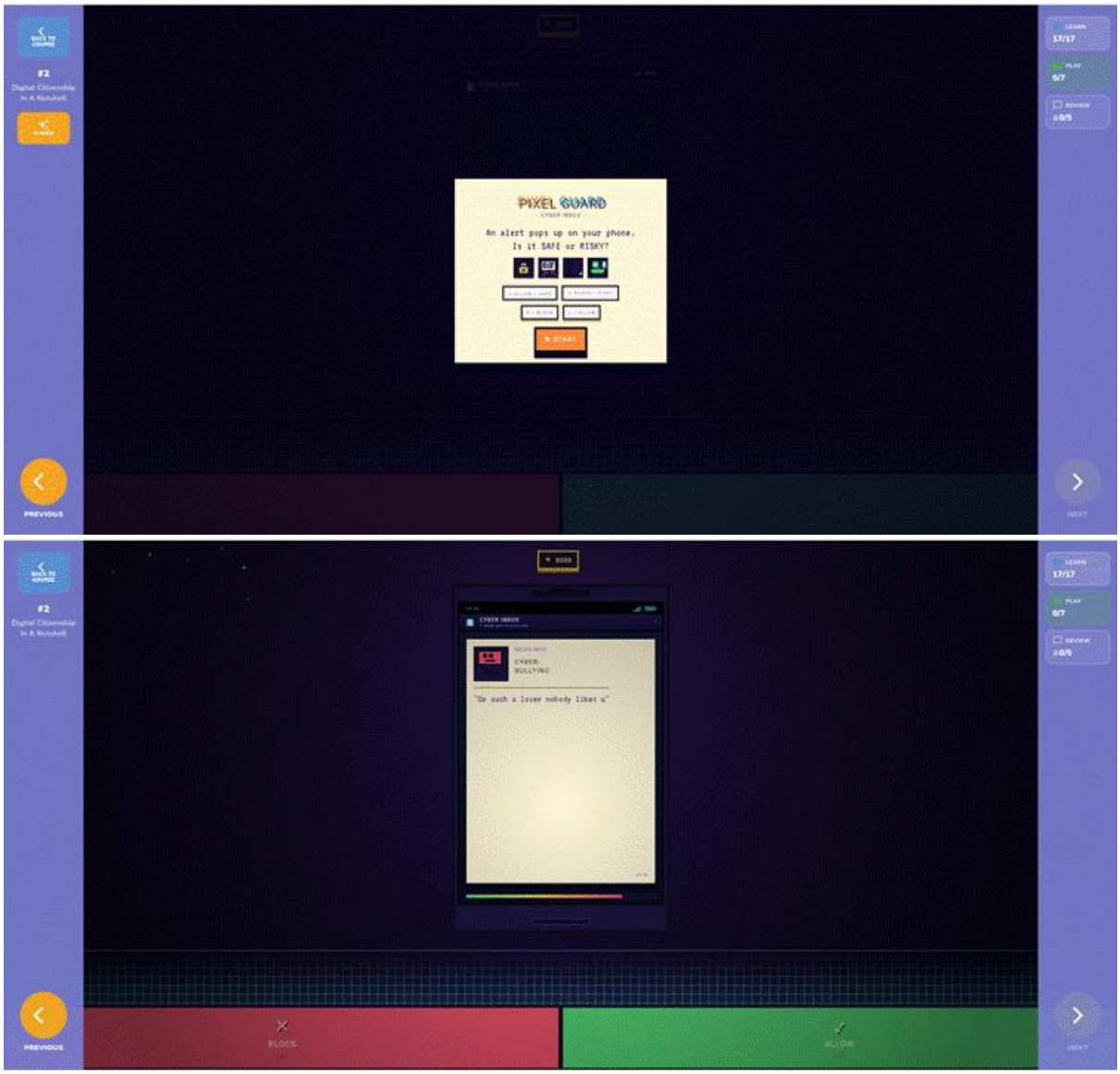


Figure 4.51: Play Section — Pixel Guard: Cyber Inbox

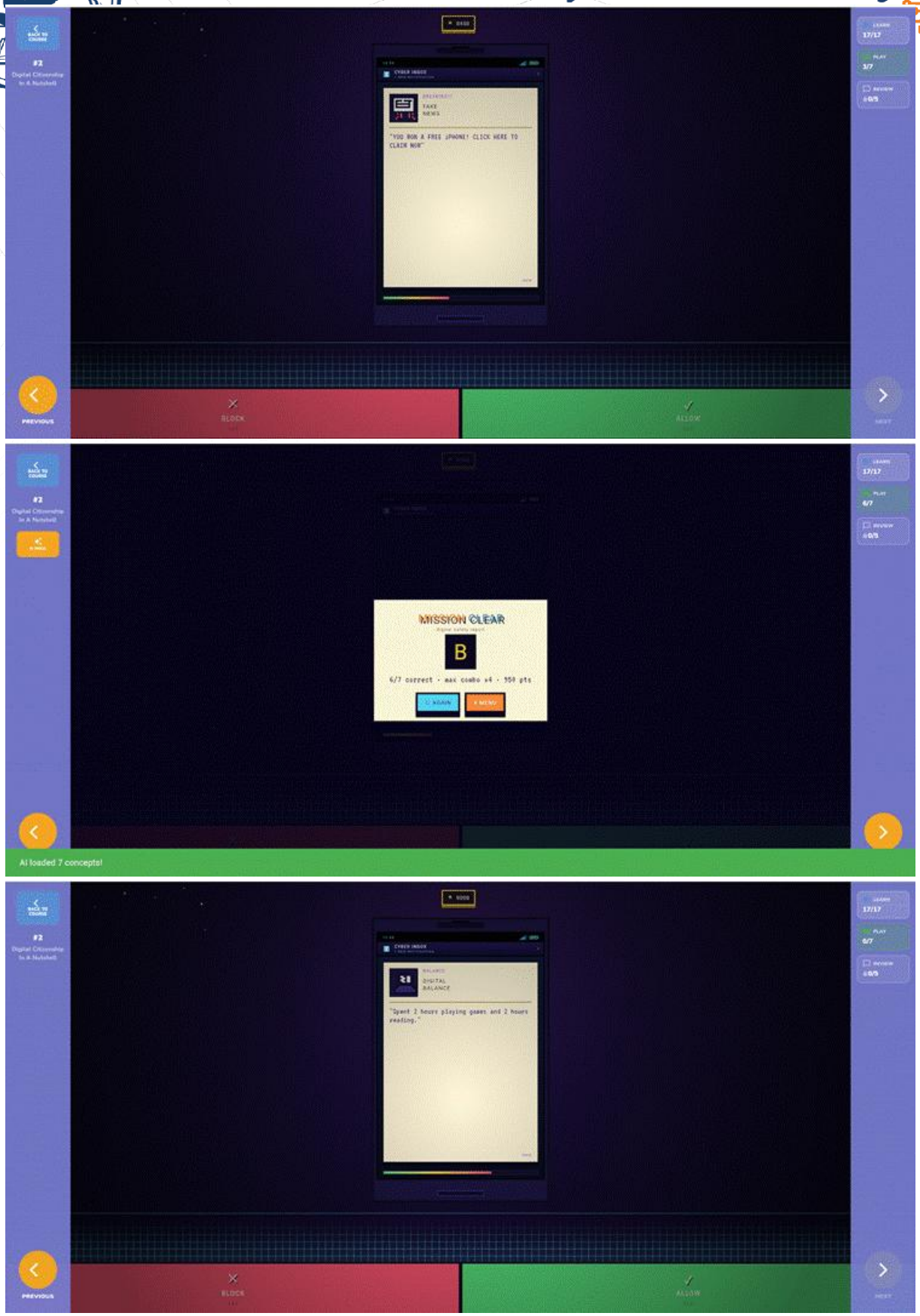


Figure 4.52: Play Section — Pixel Guard: Cyber Inbox - view 2

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.7.11 Play Section — Word Matching Game

The second Play activity is a word matching game identical in mechanics to Lesson 1's, set against the same illustrated top-down river scene. Key terms from the lesson (e.g. Digital balance, Cyberbullying, Phishing, Fake news) appear on the left, and their definitions on the right. The student connects each pair by clicking both cards correct matches turn green with a connecting line, incorrect ones flash red. The AI button regenerates a fresh set of pairs from the lesson content at any time, confirmed by a green "New pairs generated from lesson!" banner.

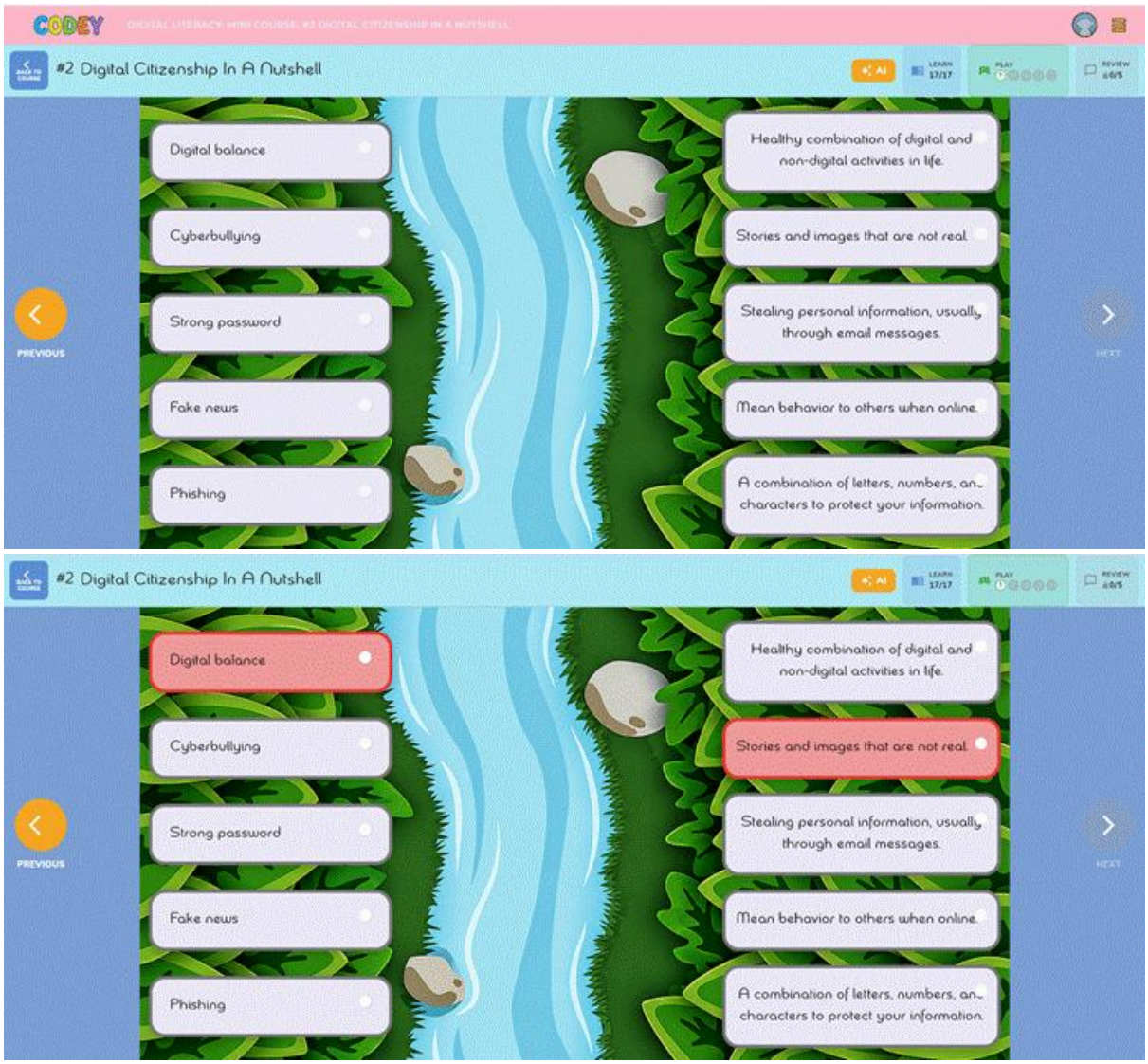


Figure 4.53: Play Section — Word Matching Game - view 4

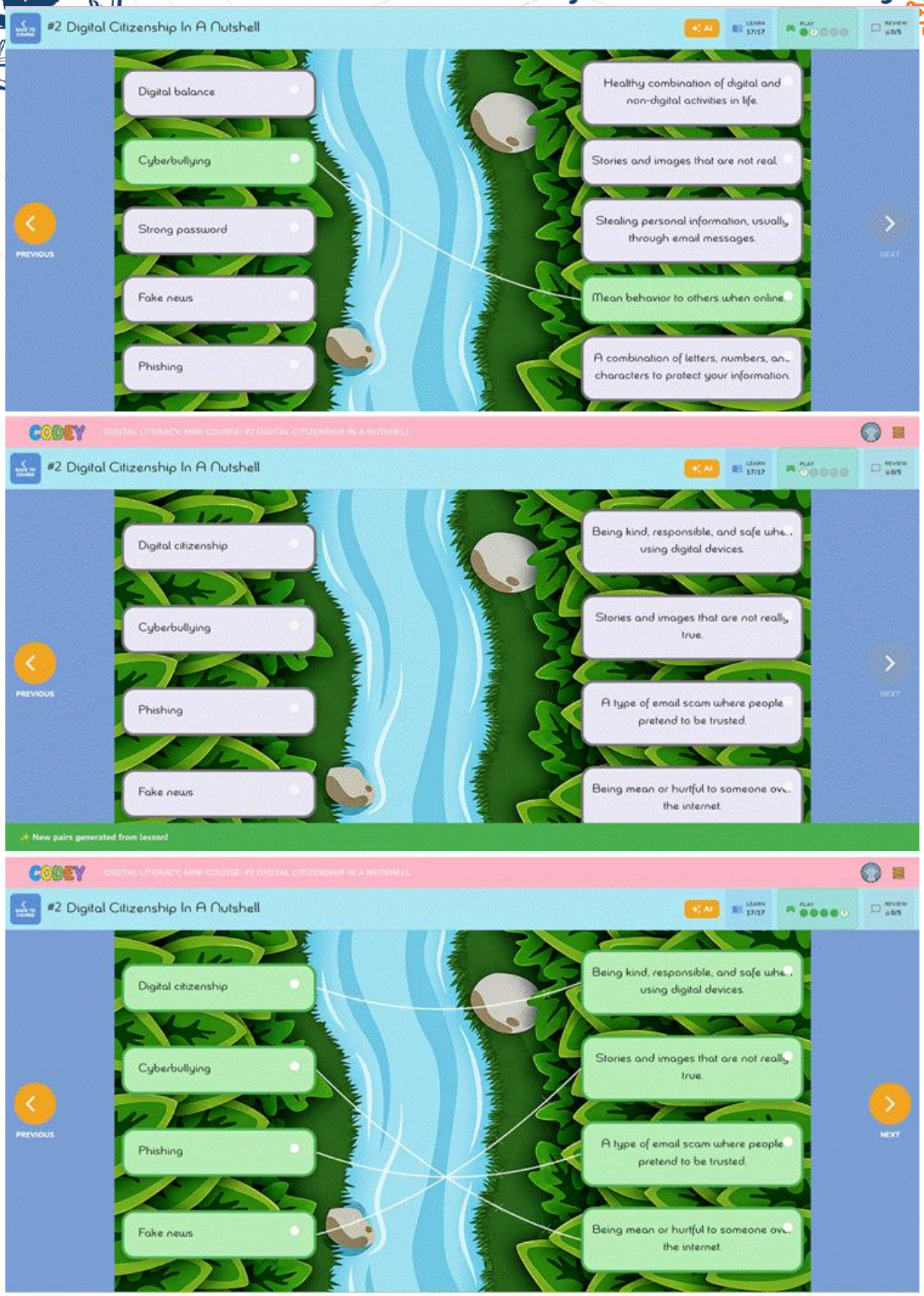


Figure 4.54: Play Section — Word Matching Game - view 5

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.7.12 Play Section — Cyber Match

Cyber Match is a memory flip-card game where 16 face-down cards are laid out in a 4×4 grid. Each card hides either a term or its matching icon from the lesson vocabulary (e.g. Phishing, Cyberbullying, Fake News, Digital Citizen). The student flips two cards at a time to find matching pairs correct matches turn green and stay revealed, while mismatches flip back face-down. A sidebar mascot "Bytey the Star" provides encouragement throughout. The top bar tracks Pairs found, total Moves, and elapsed Time. Upon finding all 8 pairs, a "You did it!" popup appears showing the total moves and time taken, with Play Again and Continue buttons.

Figure 4.55: Play Section — Cyber Match

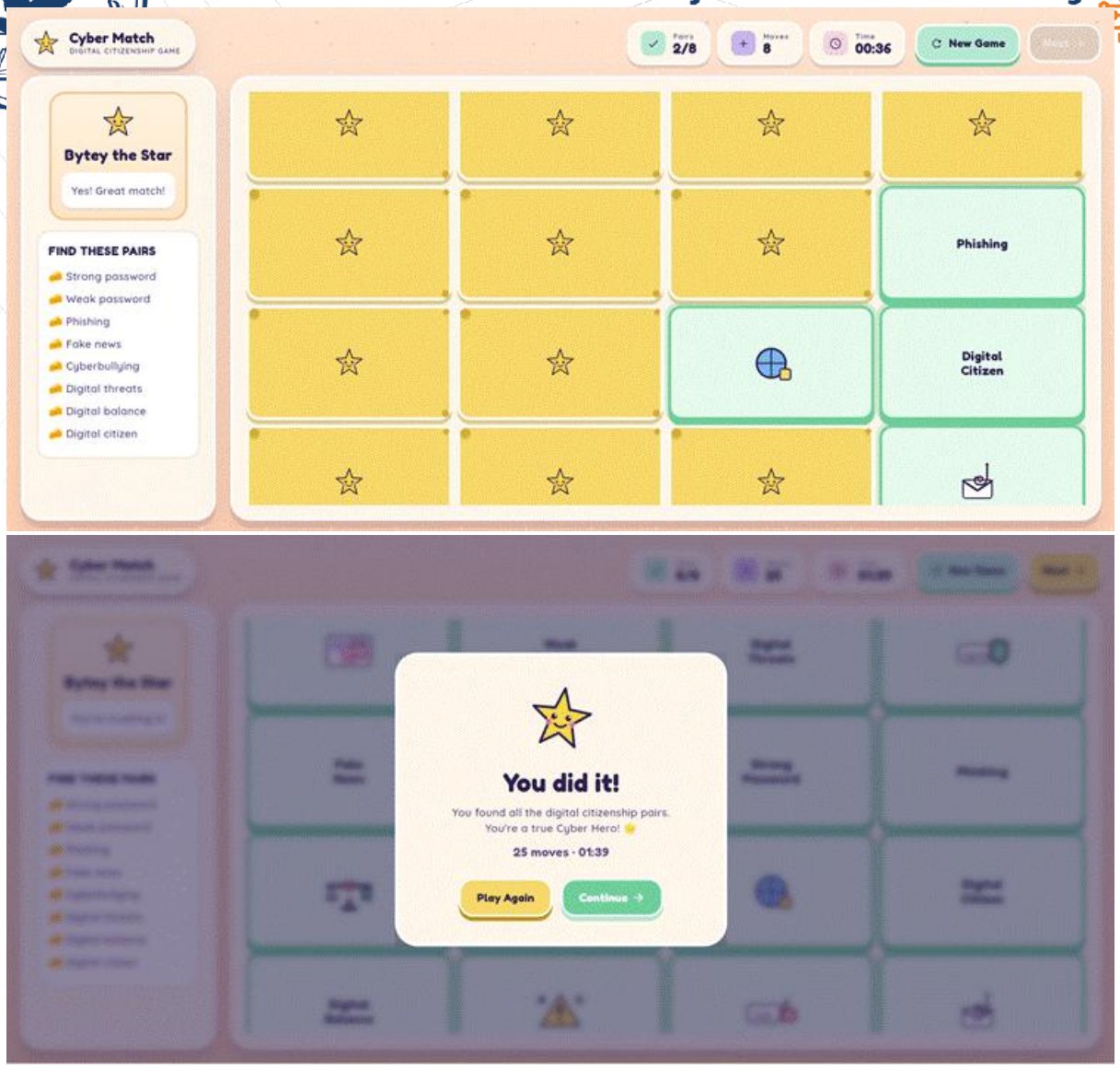


Figure 4.56: Play Section — Cyber Match - view 2

4.7.13 Review Section

The Review section is identical in structure to Lesson 1, presenting 5 multiple-choice questions based on the Digital Citizenship content. The student selects an answer and clicks Submit Your Answer correct answers highlight green with a ✓ and incorrect ones highlight red with an ✗. The AI button regenerates fresh questions from the lesson content at any time.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.57: Review Section

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Upon completing all 5 questions, a "GREAT JOB!" completion screen appears showing the review score, with Back to course and Next lesson buttons.

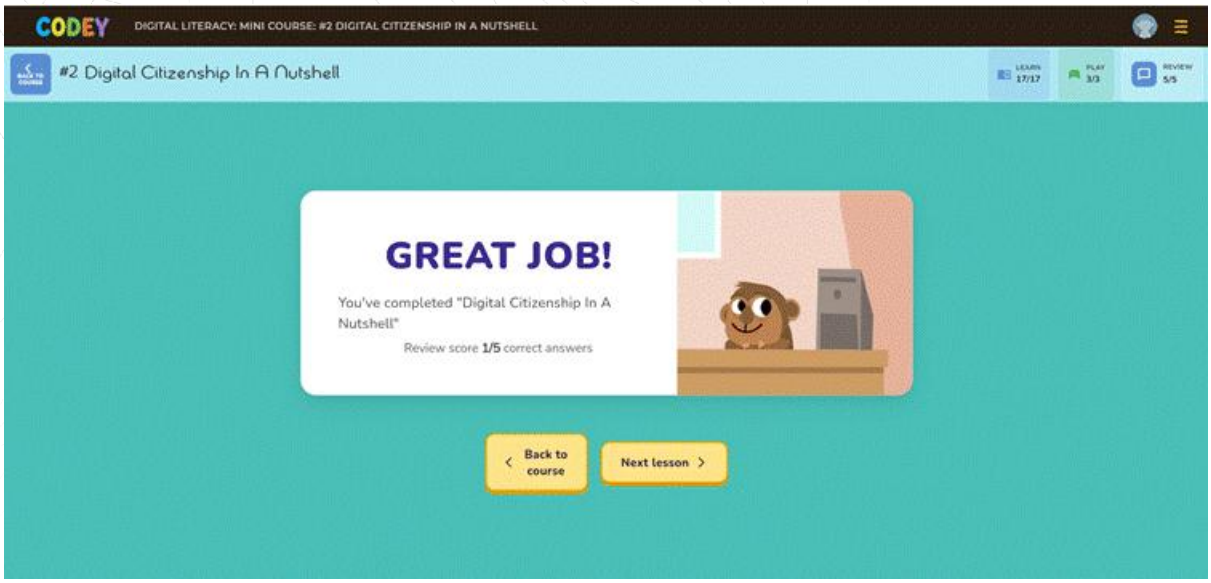


Figure 4.58: Review Section - view 2

4.7.14 Lesson 3 — Digital Collaboration

Lesson Slides

Lesson 3 consists of 13 slides covering the concept of digital collaboration working together with others using digital tools and technology. The lesson explains what collaboration means, introduces tools that enable it such as video conferencing applications and software sharing features like Google Docs and Microsoft Teams, and explores how global trends shape new digital inventions. It also introduces virtual reality as an emerging trend and discusses proper digital collaboration etiquette.

Figure 4.59: Lesson Slides

Play Section Word Matching Game

The Play section uses the same word matching game as the previous lessons, with terms drawn from the Digital Collaboration content (e.g. Video Conference, Virtual Reality, Digital Collaboration, Global Trend) matched to their definitions.

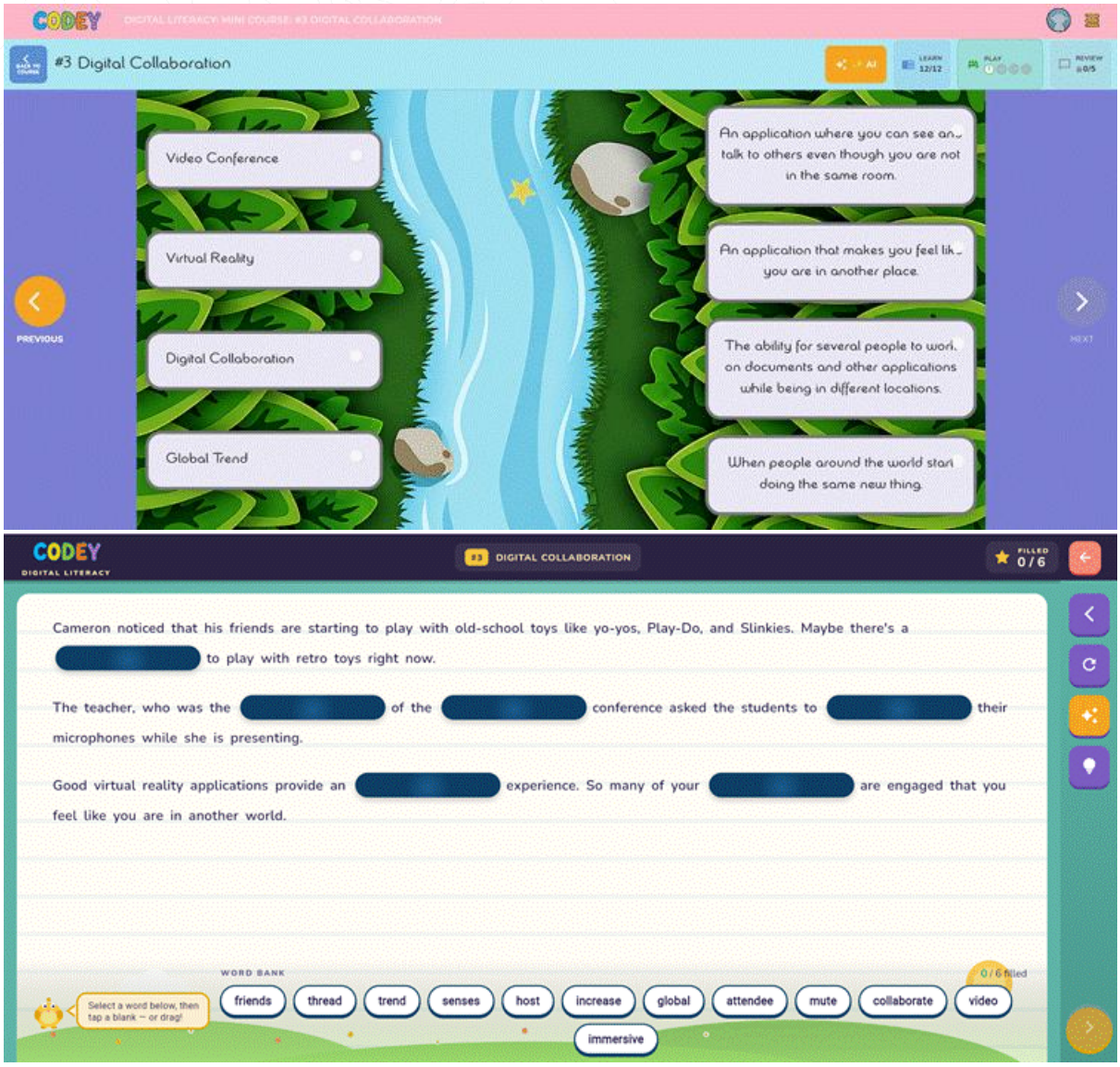


Figure 4.60: Play Section Word Matching Game

Review Section Fill in the Blanks & Multiple Choice

The Review section follows the same structure as Lessons 1 and 2, presenting a fill-in-the-blank activity with a word bank followed by 5 multiple-choice questions based on the lesson content. The AI button regenerates content at any time.

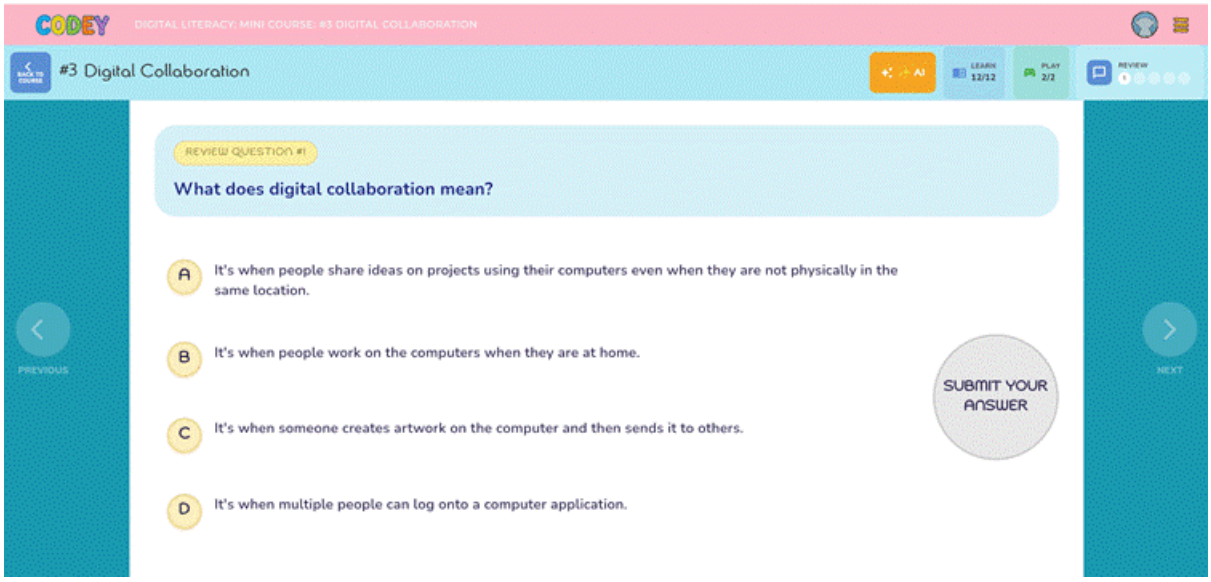


Figure 4.61: Review Section Fill in the Blanks & Multiple Choice

4.8 Data is Everywhere Course

4.8.1 Course Overview

The Data is Everywhere course introduces children to the world of data science through 2 lessons: Collecting Data and Organizing Data. The course overview screen follows the same layout as Digital Literacy, showing both lesson cards with a progress gauge on the left and onboarding tooltips guiding the student to watch each lesson and complete the quiz to unlock the next.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

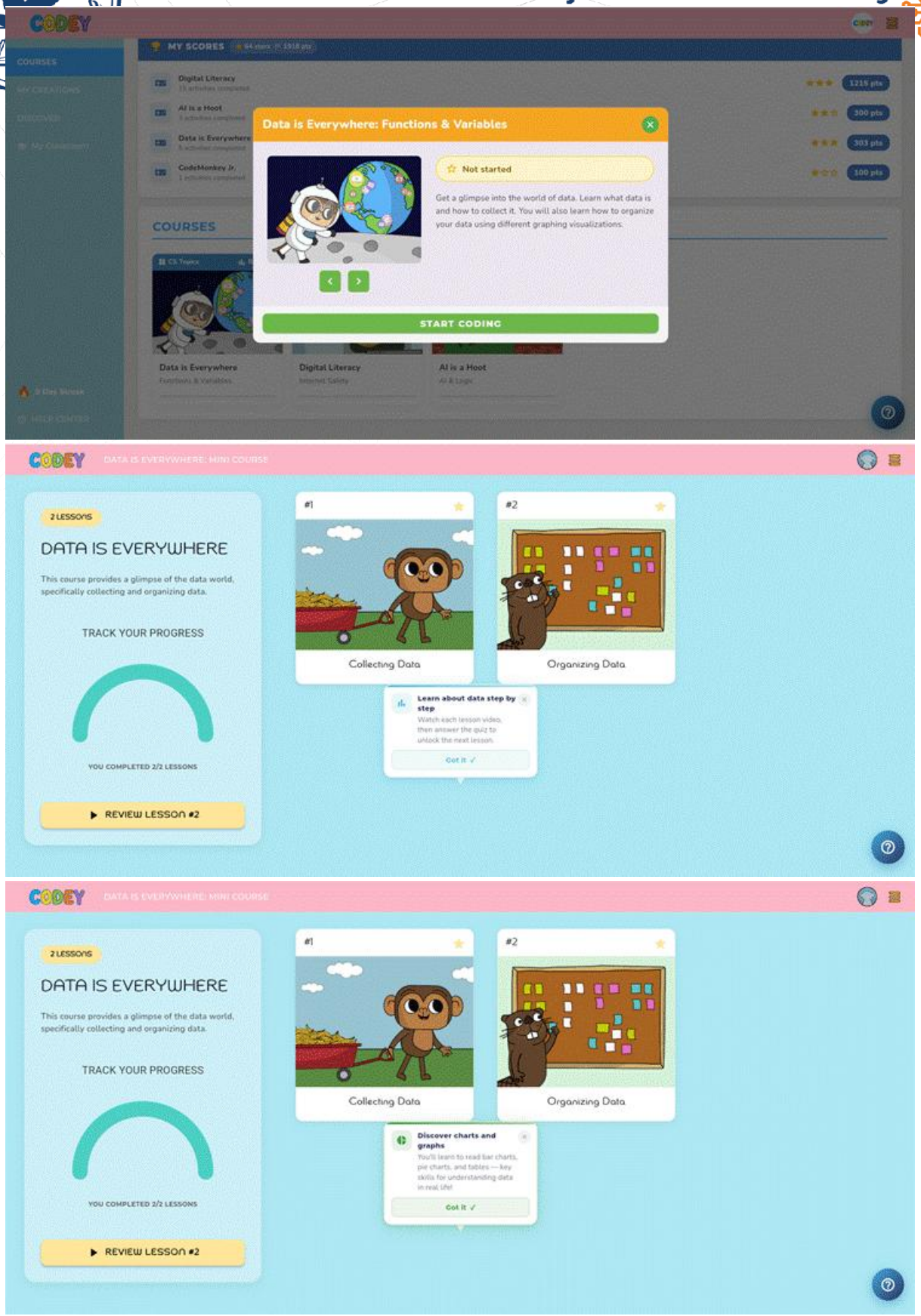


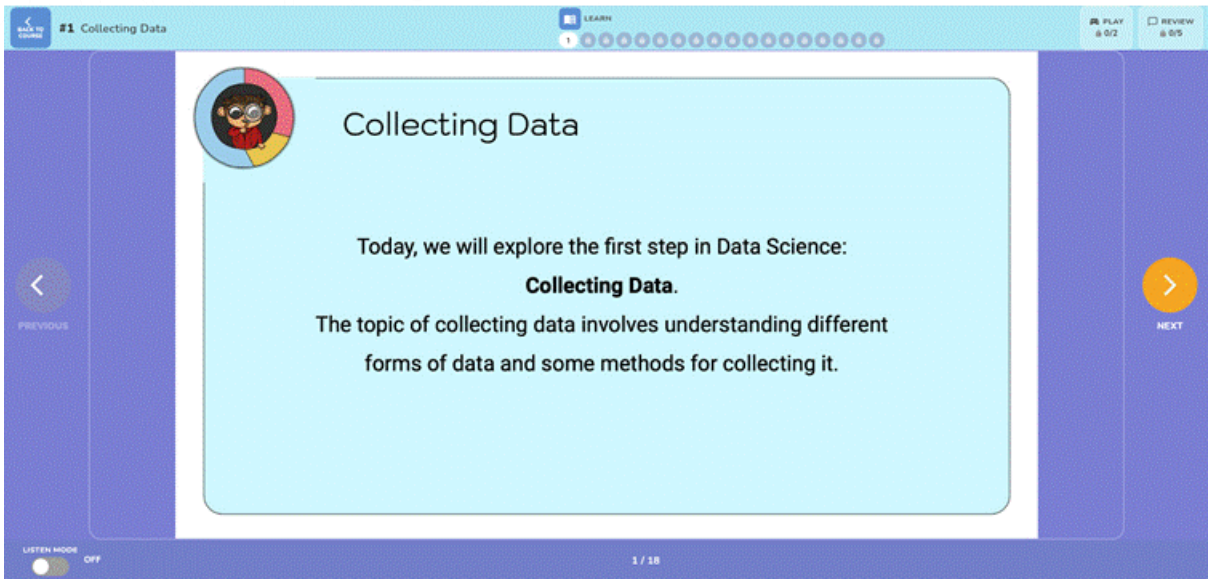
Figure 4.62: Course Overview

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.8.2 Lesson 1 — Collecting Data

The lesson consists of 18 slides covering the first step in data science understanding different forms of data and methods for collecting it, including numerical data, non-numerical data, observations, and sensors.

The Play section features Data Sorter, a game identical in mechanics to Pixel Guard but themed around data classification. A data card appears on a simulated phone screen and the student classifies it as Numerical or Non-Numerical using the two buttons at the bottom. The Review section follows the same multiple-choice format as previous courses



University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

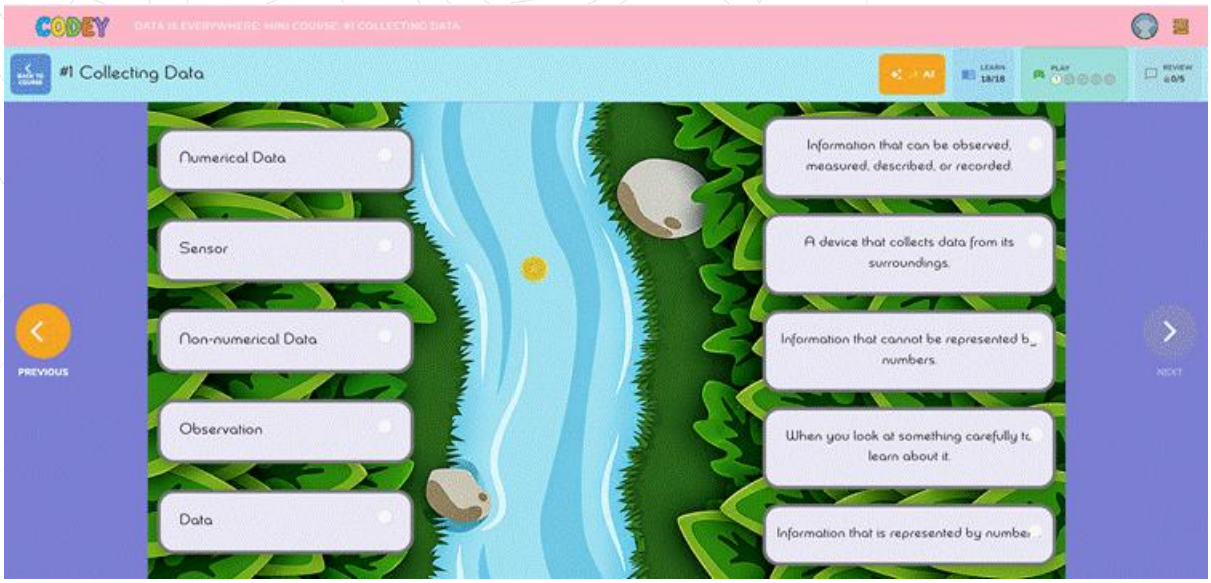


Figure 4.63: Lesson 1 — Collecting Data

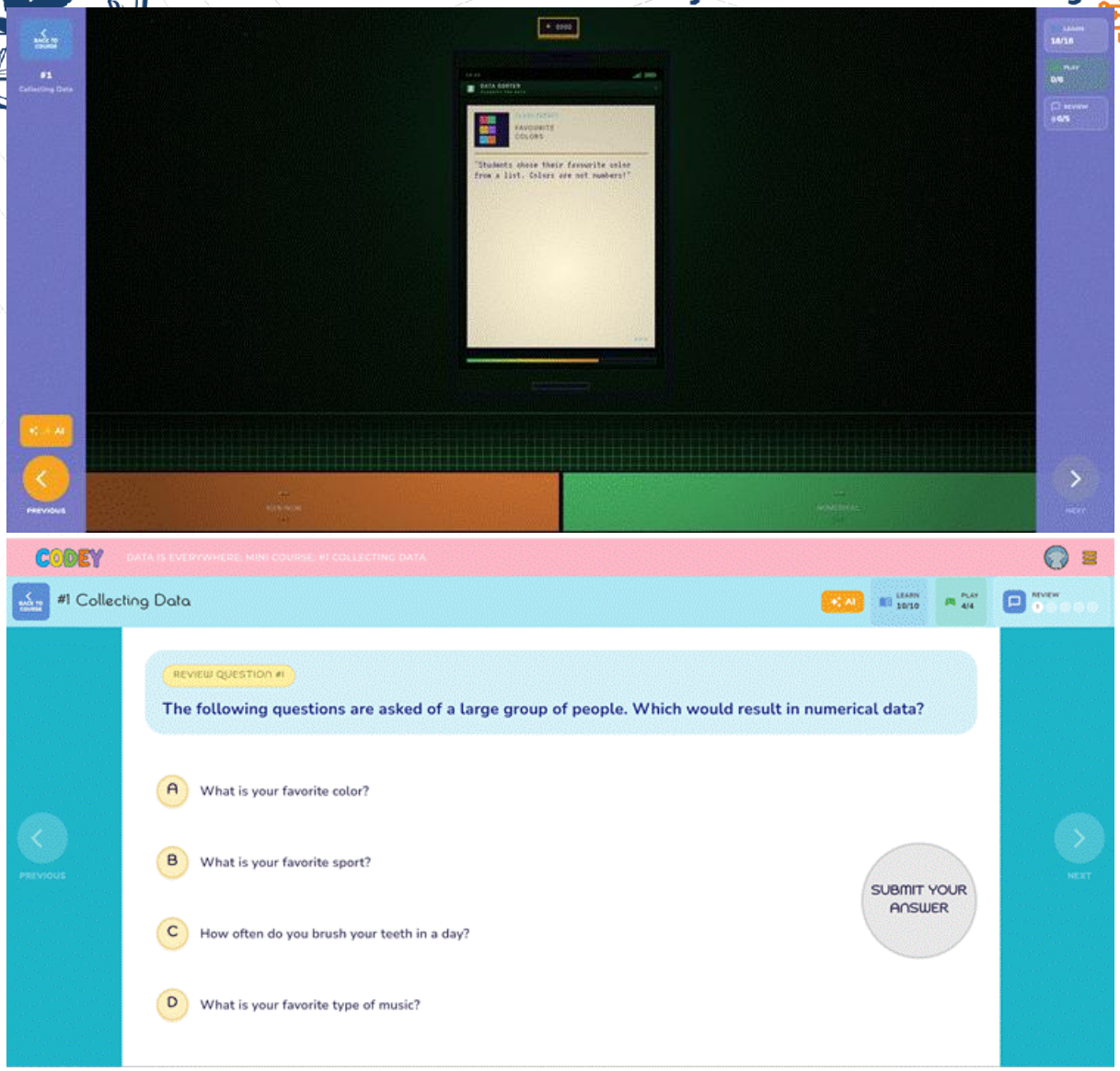


Figure 4.64: Lesson 1 — Collecting Data - view 2

4.8.3 Lesson 2 — Organizing Data

The lesson consists of 15 slides covering different ways to present and organize data so patterns can be identified, including bar graphs, picture graphs, line plots, and tables.

The Play section uses the same word matching game with data visualization terms. The Review section presents a fill-in-the-blank activity followed by 5 multiple-choice questions based on the lesson content.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

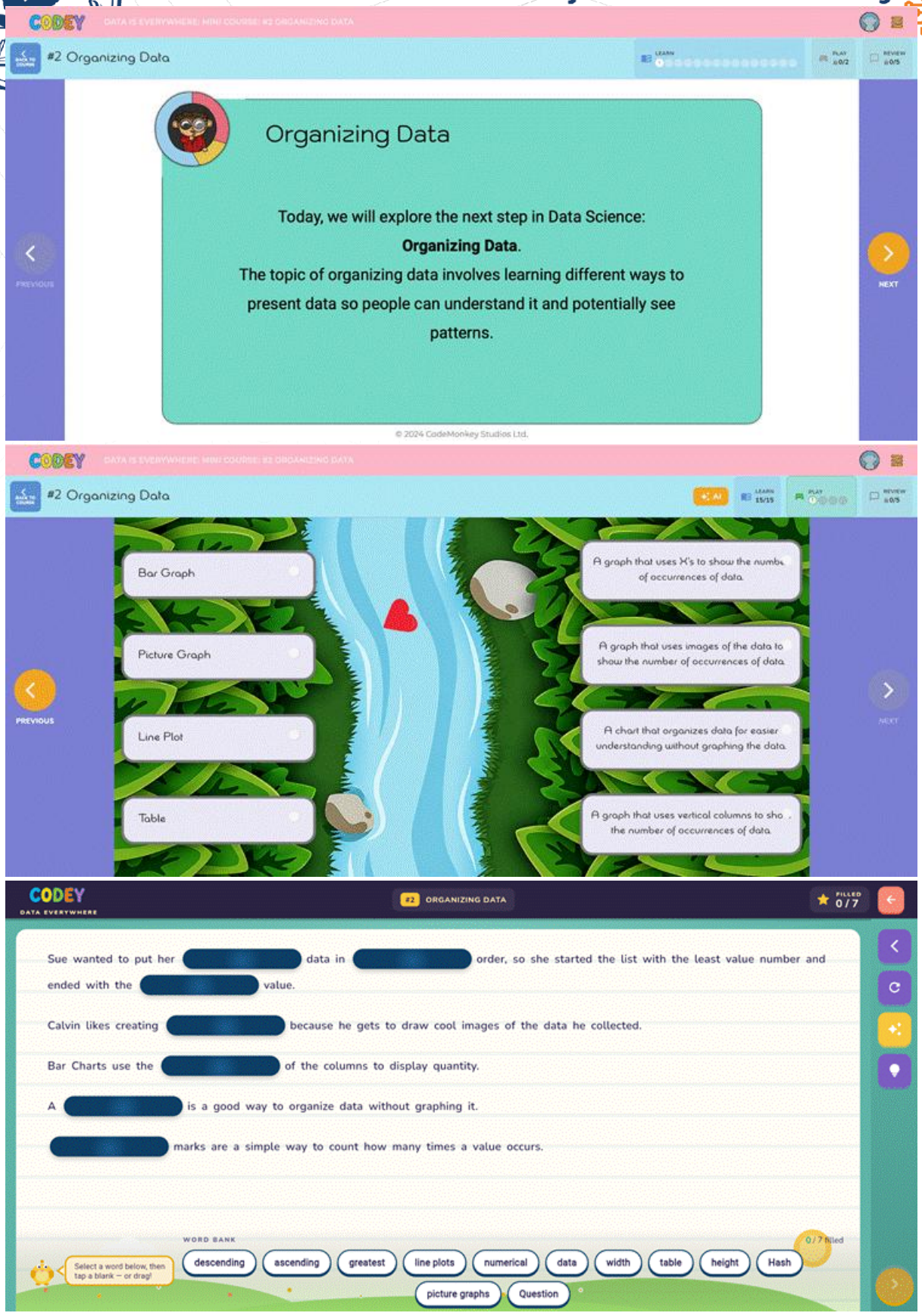


Figure 4.65: Lesson 2 — Organizing Data

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

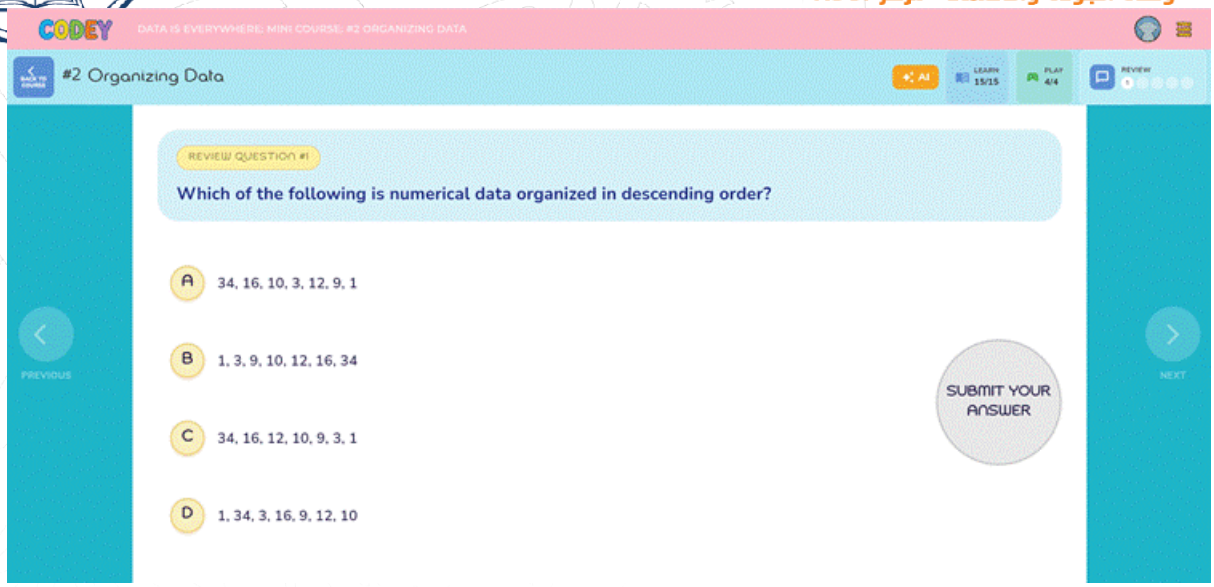


Figure 4.66: Lesson 2 — Organizing Data - view 2

4.9 AI is a Hoot Course

4.9.1 Course Preview Modal

Clicking on the AI is a Hoot course card from the courses catalog opens a preview modal with an orange header displaying the course title "AI is a Hoot: AI & Logic." The modal shows the course illustration featuring Oliver the owl and the monkey characters against a night sky background, a status badge ("Not started" in yellow), and the description: "Start this course to learn exciting coding concepts!" Left and right arrow buttons allow browsing between course lessons within the modal. A green Start Coding button at the bottom launches the course.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

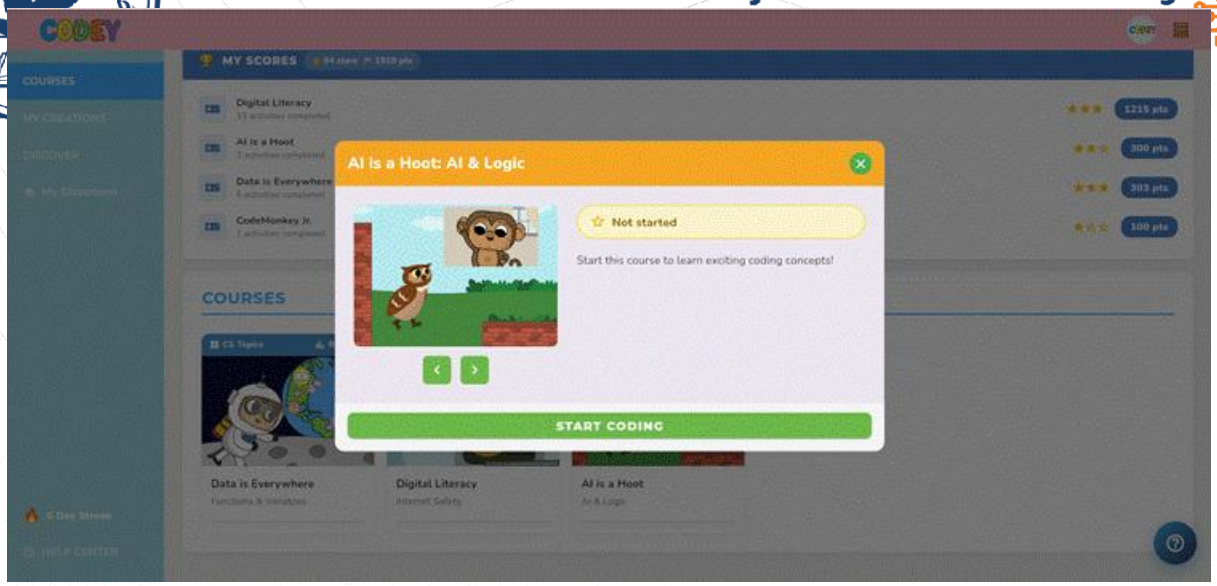


Figure 4.67: Course Preview Modal - view 2

4.9.2 Exercise Screen Layout

The AI is a Hoot course is structured as a series of 9 guided exercises rather than traditional lesson slides. The top navigation bar displays the course title "AI IS A HOOT: EXERCISE #1" in the pink header, along with an exercise navigator showing "Exercise 1 of 9" with left and right arrows to move between exercises, a text-to-speech button, and a reset button.

The screen is divided into three main panels:

- **Left panel Overview:** Displays the exercise title and a written description of the task. For Exercise 1, the overview reads "Nice to Meet You, Oliver." and explains that the student will learn to build an AI-based game to move Oliver the owl between obstacles to reach the end of the route, using different body postures to change the owl's size so it can pass below or over tiles. A preview of the finished game is shown at the bottom of the panel, along with a contextual hint explaining the Loop block. A "Show my previous solution" button at the top allows the student to view their prior submission. A Listen button enables text-to-speech read-aloud of the instructions.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

- **Center panel Block Programming Editor:** The main workspace where the student builds their solution by dragging blocks from a palette. The editor shows the currently selected sprite ("Oliver") at the top with a RUN button to execute the code. The workspace area displays a dotted canvas with the instruction "Drag blocks from the palette here. Double-tap a block to remove it." A trash icon is available for deleting blocks. Below the canvas, the block palette is organized into color-coded categories displayed as buttons: Movement, Events, Display, Widgets, Game and Sounds, Control, Logic and Data, Variables, Object's Functions, Other objects' Functions, and AI. Selecting a category reveals the corresponding blocks in a scrollable tray above the category buttons.
- **Right panel Game Preview:** A live game canvas showing the current state of the game world, with Oliver the owl character visible against a dark starry night background with brick tile obstacles. Four toolbar icons in the top right corner allow switching between selection, move, erase, and draw modes. Below the canvas, four tabs are available: Sprites, Widgets, Sounds, and Game, allowing the student to manage game assets. An "ADD NEW" button and the Oliver sprite card are shown under the Sprites tab, with a gear icon on the sprite card for settings.

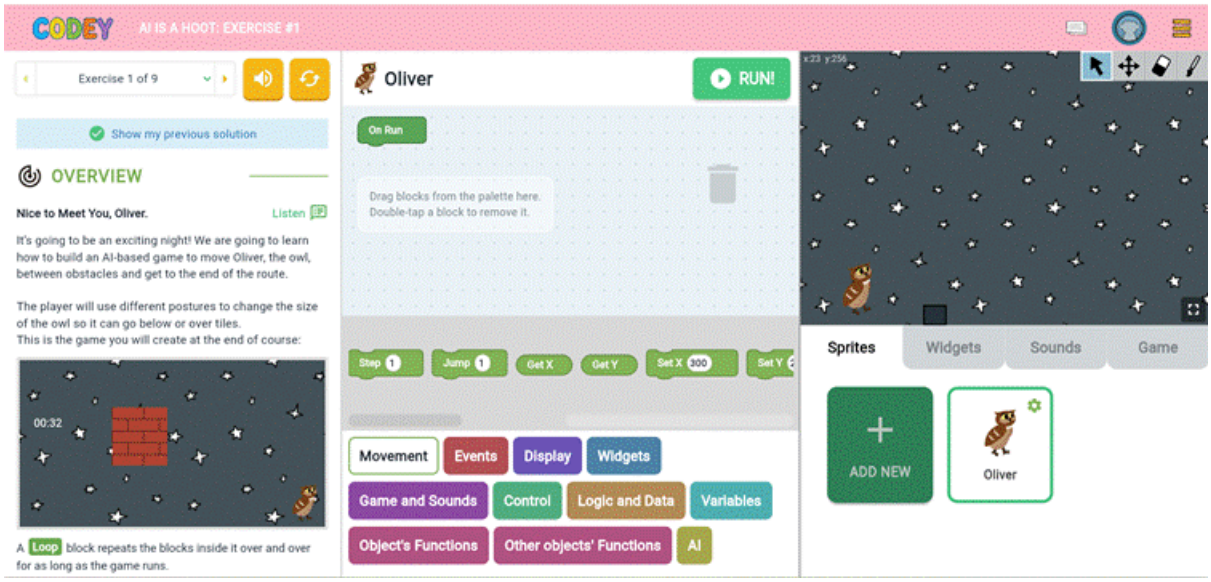


Figure 4.68: Right panel Game Preview: A live game canvas showing the current state of the game

4.9.3 Block Categories

The block palette is the core of the AI is a Hoot programming environment. All block categories were custom-designed for the platform. Each category groups related blocks by function and is color-coded for visual clarity:

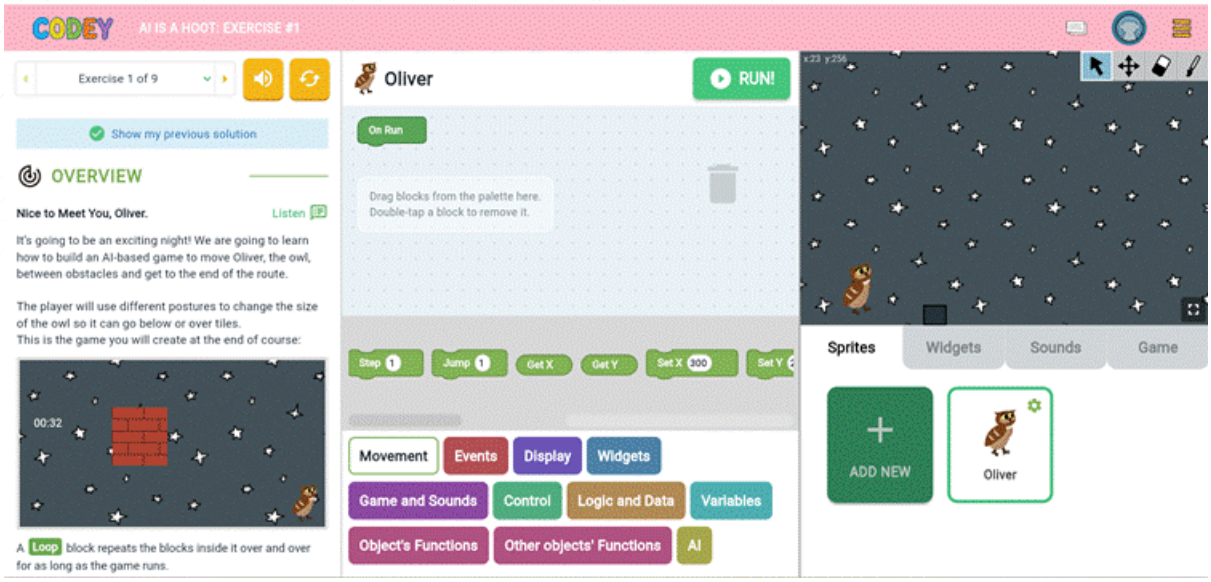


Figure 4.69: Block Categories

Events (dark red): Trigger blocks that fire when specific conditions occur. Available blocks include: On Run, On Key (with a configurable key selector), On Click, On Update, On Collide (with a sprite selector), On Collide With World Bounds (two variants), On Swipe (with direction selector), On Game Tap, and On Drag End.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

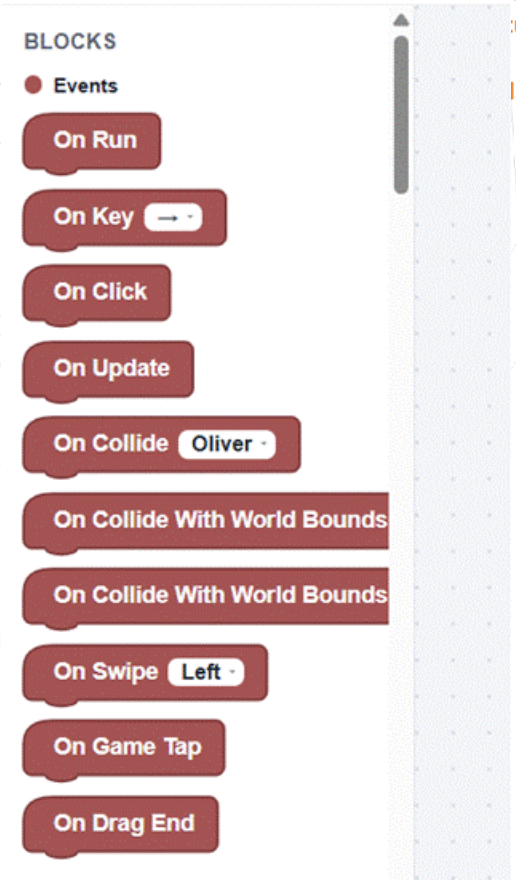


Figure 4.70: Block Categories - view 2

Control (green): Flow control structures including: Loop (infinite repeat), Repeat N times (with a configurable count), if (condition block), and if/else (condition with two branches).

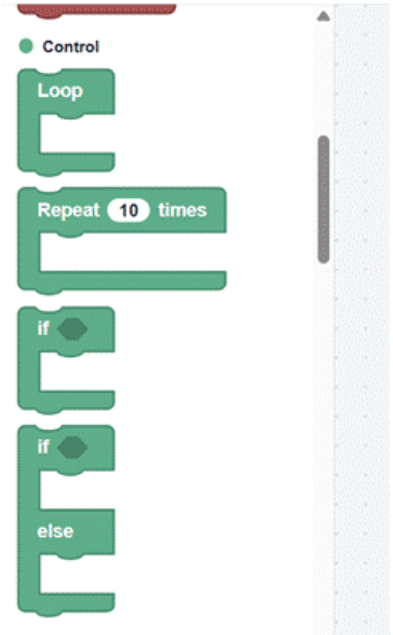


Figure 4.71: Block Categories - view 3

Movement (green): Blocks for controlling sprite position and physics, including: Step, Jump, Set X, Set Y, Change X By, Change Y By, Set Rotation, Change Rotation By, Set Speed, Set Allow Gravity (true/false toggle), Get X, Get Y, and Get Rotation.

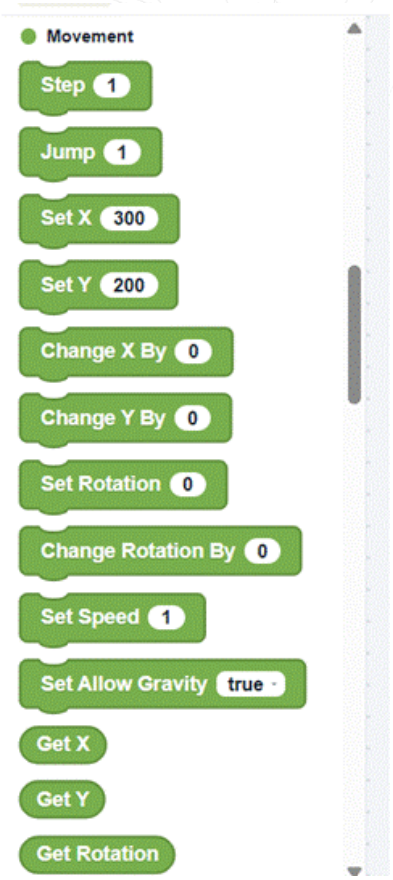


Figure 4.72: Block Categories - view 4

Display (purple): Blocks that control sprite visibility and size: Show, Hide, Destroy, Disable, Enable, Set Scale (with a numeric value), Get Scale, and Get Distance From (with a sprite selector).

Widgets (blue): Blocks for interacting with on-screen UI elements: Set counter To, Change counter By, Set text label To, Set timer To, Start clock, Get counter Value, and Get timer Seconds.

Game and Sounds (purple): High-level game control blocks: Play Sound (with a sound selector), Reset Game, Pause Game, Unpause Game, Get Game Time, and Set Background (with a background selector).

Figure 4.73: Game and Sounds (purple): High-level game control blocks: Play Sound (with a sound

Logic and Data (gold/brown): Boolean and arithmetic blocks: not, and, = (equality), numeric literal (0), string literal (—), arithmetic expression (1 + 1), and Random from N to N.

Variables (teal): Blocks for managing named variables: Set [variable] to [value], Change [variable] by [value], and a variable reference block (my variable).

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

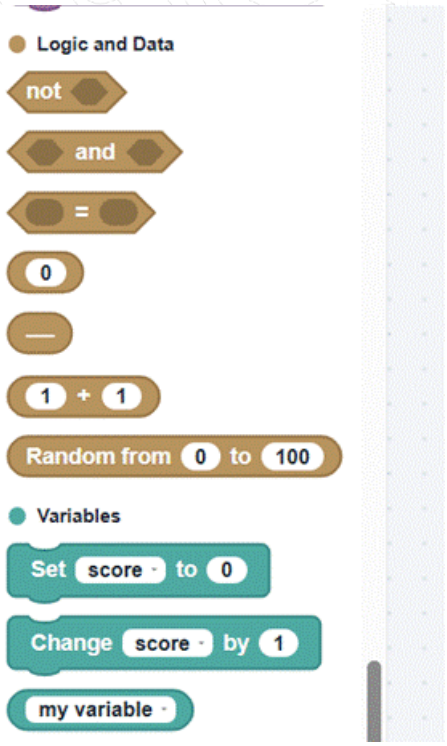


Figure 4.74: Variables (teal): Blocks for managing named variables: Set [variable] to [value], Change

Object's Functions (dark pink/mauve): Blocks for defining reusable functions on the current sprite: a function definition block (? to do something), a function definition with a return value, an if/return conditional, and a standalone return block.

Other objects' Functions (dark pink/mauve): A single block — call [sprite] [function] — allowing one sprite to invoke a named function defined on another sprite.

AI (olive green): Two dedicated AI blocks that connect the exercise to the machine learning model trained in the course: Predict [label], which triggers a prediction using the student's trained gesture model, and On Prediction [label], an event block that fires when the model returns a specific predicted label, allowing the student to wire body-pose predictions directly to game actions.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

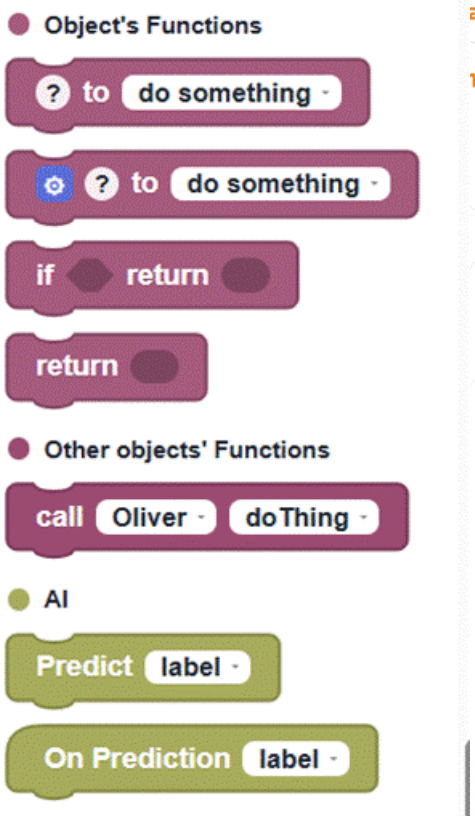


Figure 4.75: AI (olive green): Two dedicated AI blocks that connect the exercise to the machine

4.9.4 Right Panel — Sprites, Widgets, Sounds, and Game Tabs

The right panel of the exercise screen is divided into four tabs :Sprites, Widgets, Sounds, and Game each providing a different layer of game asset management that updates the live game canvas in real time.

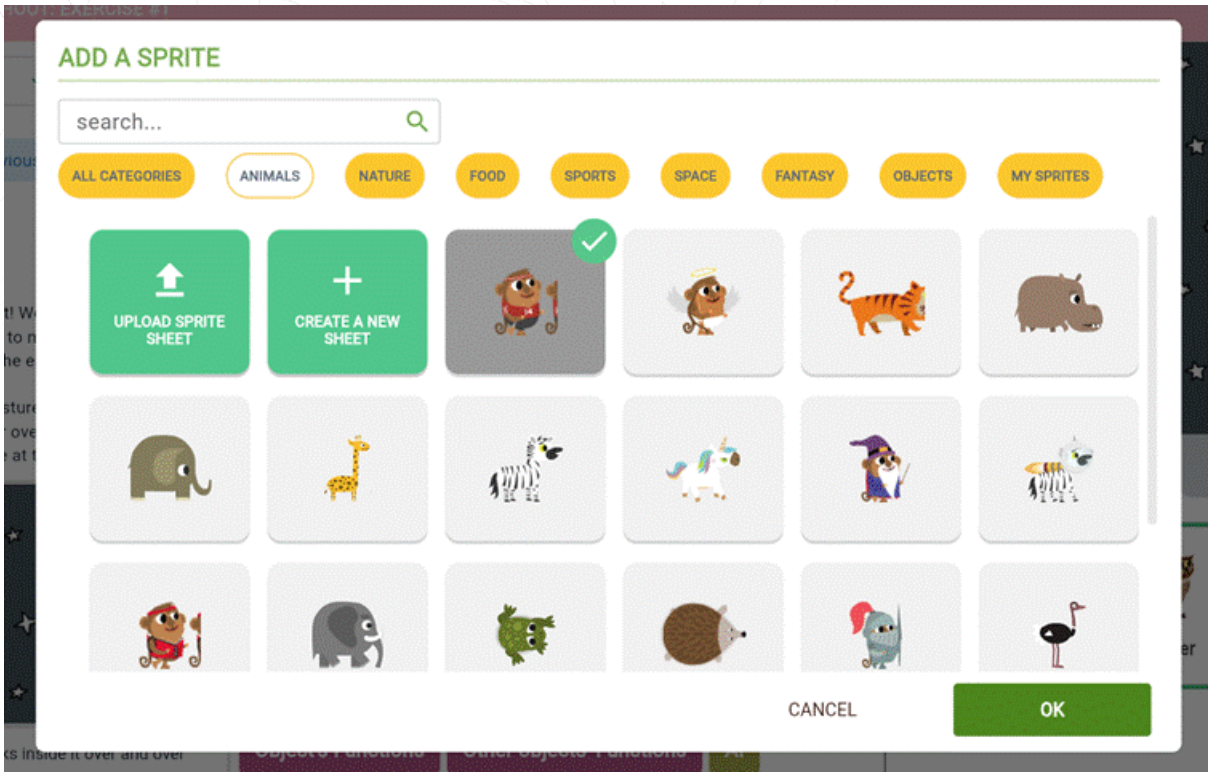
Sprites Tab , Add a Sprite

Clicking the ADD NEW button under the Sprites tab opens an "Add a Sprite" modal. A search bar at the top allows searching by name. Below it, category filter pills let the student browse sprites by: All Categories, Animals, Nature, Food, Sports, Space, Fantasy, Objects, and My Sprites. The currently selected category is highlighted with a white border.

The sprite grid displays the available built-in sprites as illustrated cards. Two action cards appear first in the grid: "Upload Sprite Sheet" (for importing a custom PNG sprite sheet) and "Create a New Sheet" (for drawing one from scratch). The currently selected sprite is marked with a green checkmark. Clicking any sprite card selects it; clicking OK adds it to the game. The Animals category includes characters such as monkeys, a tiger, hippo, elephant, giraffe,

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

zebra, unicorn, wizard, hedgehog, ostrich, frog, and a knight. The Food category shows a banana and a chocolate bar. Other categories populate with their respective themed assets.



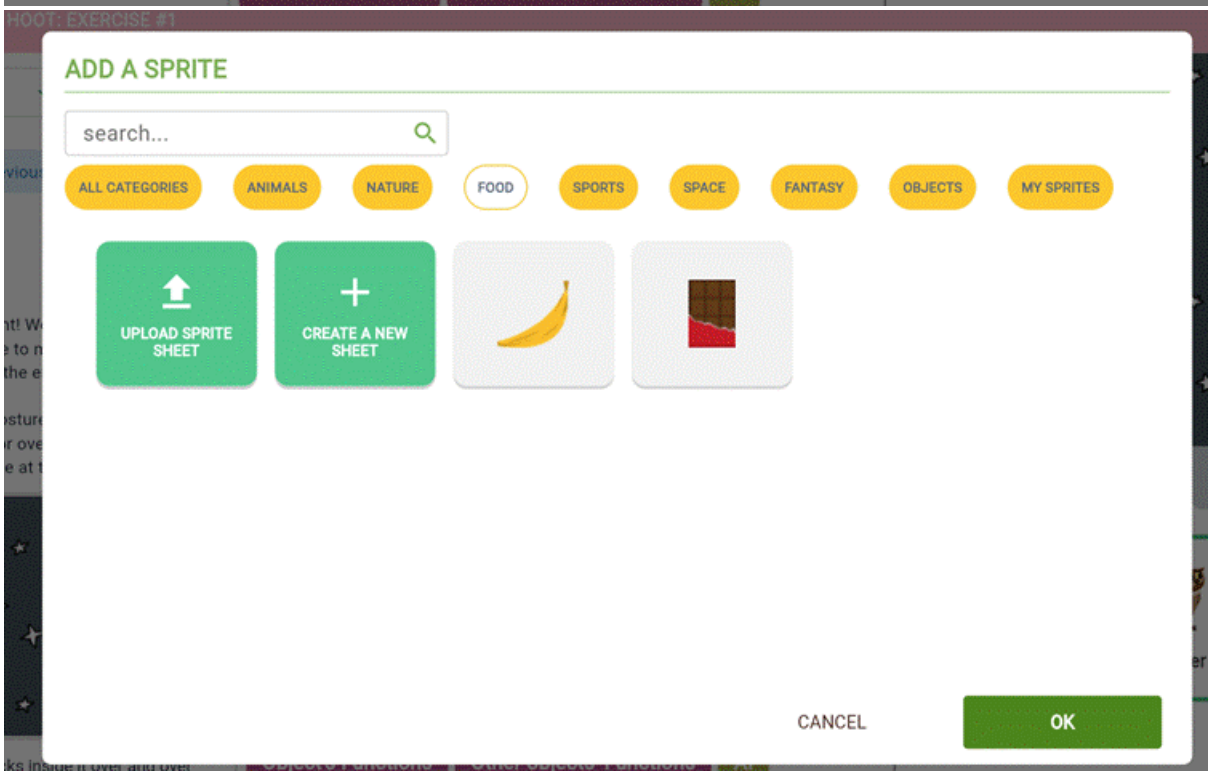
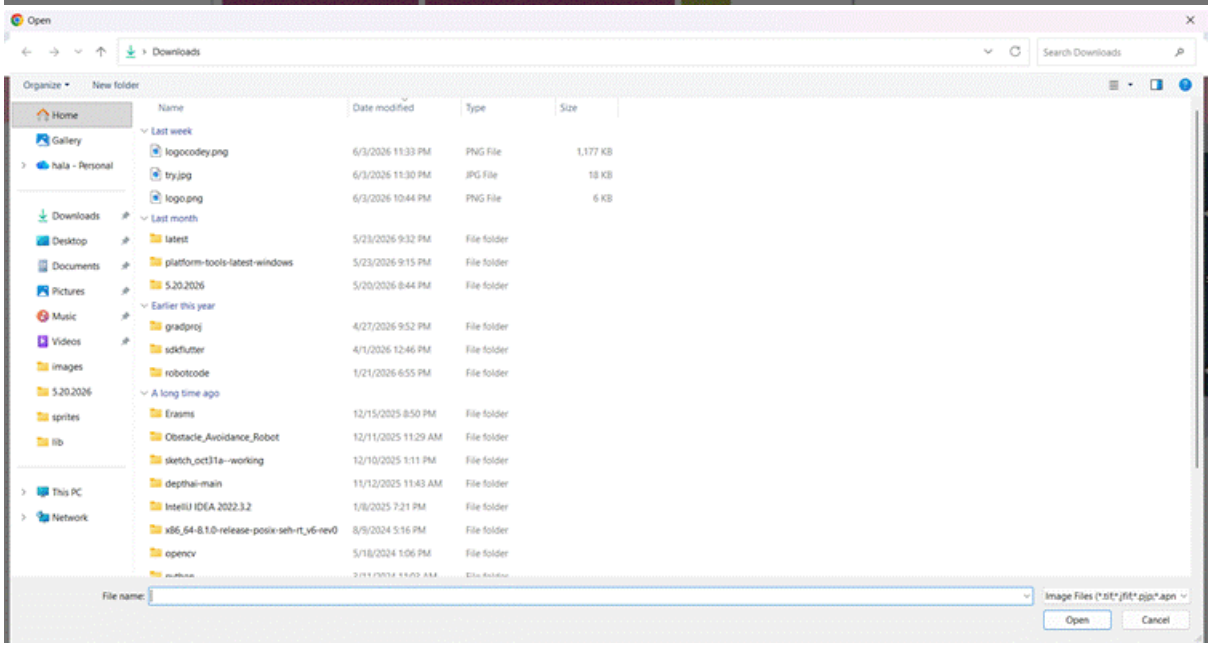


Figure 4.76: Sprites Tab , Add a Sprite

Upload Sprite Sheet

Selecting "Upload Sprite Sheet" opens a dedicated upload screen. An "Important Instructions" panel on the left explains the required format: all animation states must be placed in a single horizontal row, frames must be divided into equal widths, and the background must be transparent using an alpha channel. A visual example of a correctly formatted horizontal sprite sheet is shown. The file must be saved as PNG and cannot exceed 3MB. On the right, a drag-and-drop upload zone with a "Browse Files" button opens the system file picker. A "Sprite Preview" area on the far right shows a preview of the uploaded image once a file is selected, displaying the filename below it. A "Number of frames" spinner allows the student to specify how many animation frames are in the sheet. A "Return to Sprites" button at the top left navigates back. Once uploaded and confirmed with OK, the new sprite appears in the Sprites panel alongside the built-in ones.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu



University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

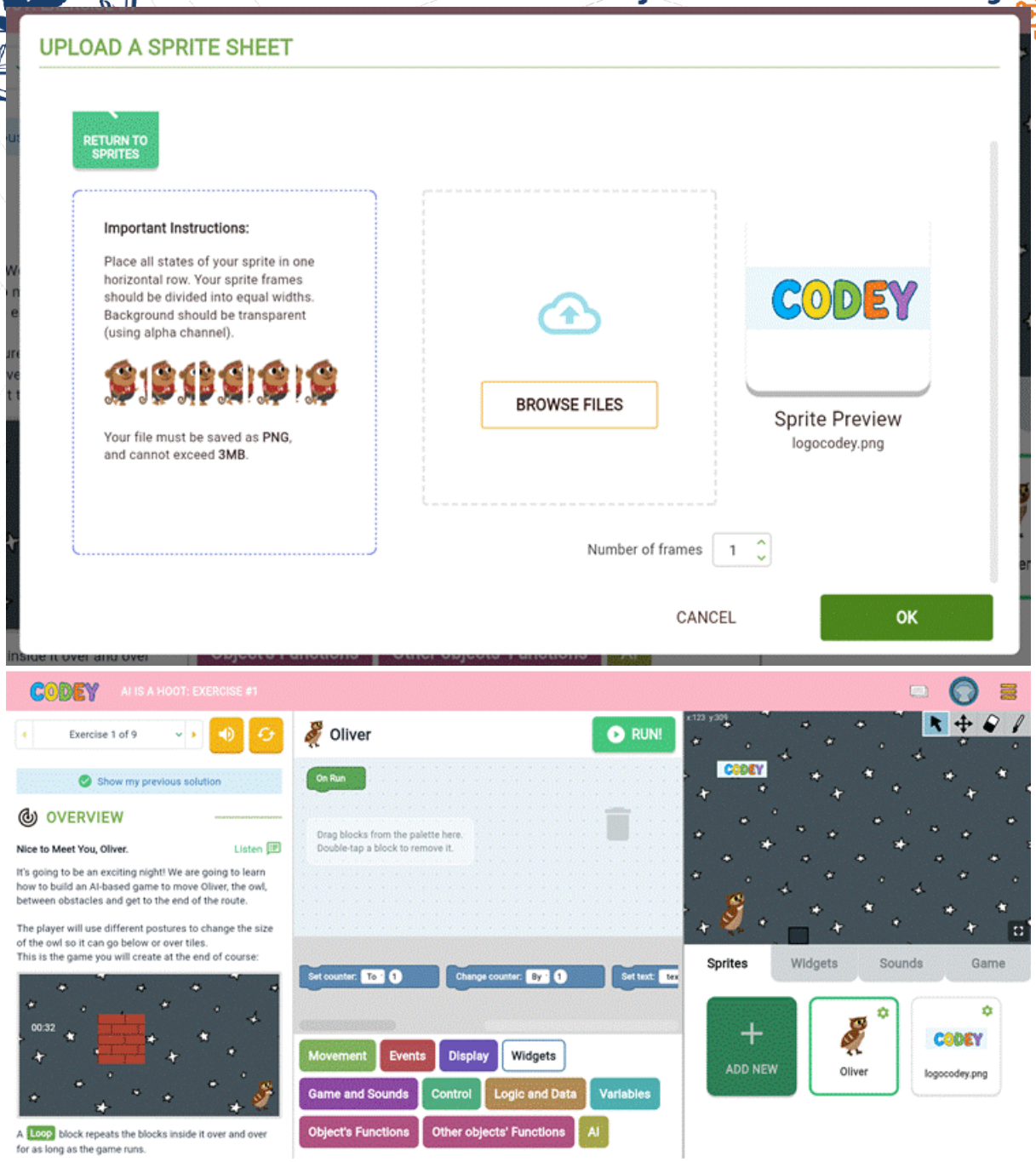


Figure 4.77: Upload Sprite Sheet

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Also the sprite settings

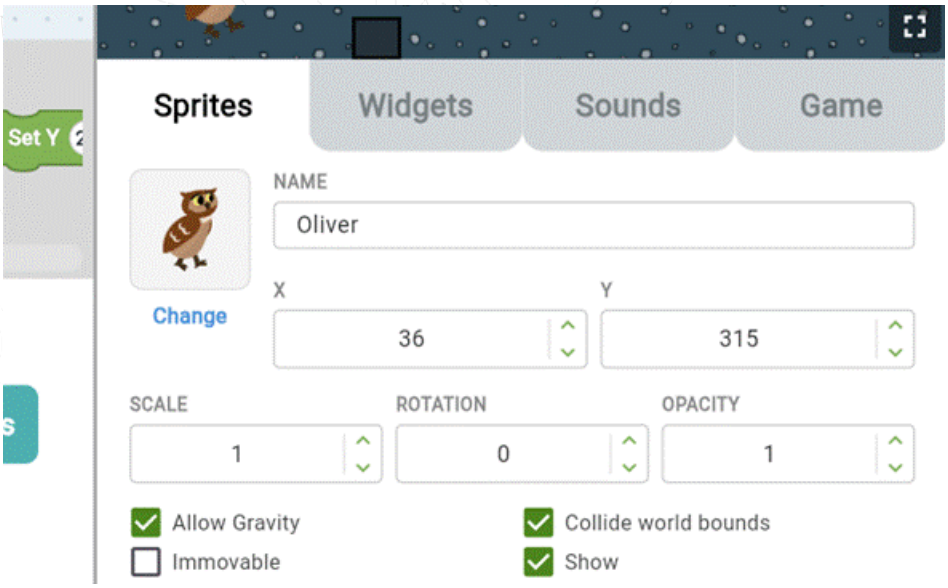


Figure 4.78: Also the sprite settings

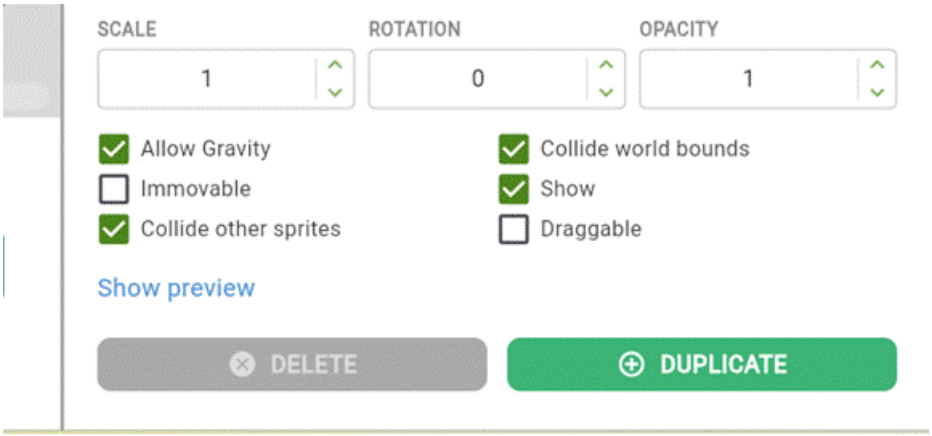


Figure 4.79: Also the sprite settings - view 2

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Create a New Sprite Sheet

Selecting "Create a New Sheet" first asks the student to define the sprite frame size, with numeric spinners for Width and Height (defaulting to 200×200 px) and a live preview canvas showing the frame dimensions to scale. Clicking OK opens the full Sprite Editor, labeled "SPRITE EDITOR 200×200 px." The editor provides a left-side toolbar with drawing tools: pencil, line, rectangle, and circle, along with stroke and fill color pickers, undo/redo buttons, and a Clear button. The top bar offers brush size options shown as dot size selectors. The main canvas is the white drawing area at the specified dimensions. At the bottom, a frame strip shows the current frame as a thumbnail, with a "+" button to add additional animation frames. Clicking OK saves the drawn sprite and returns to the sprite list.

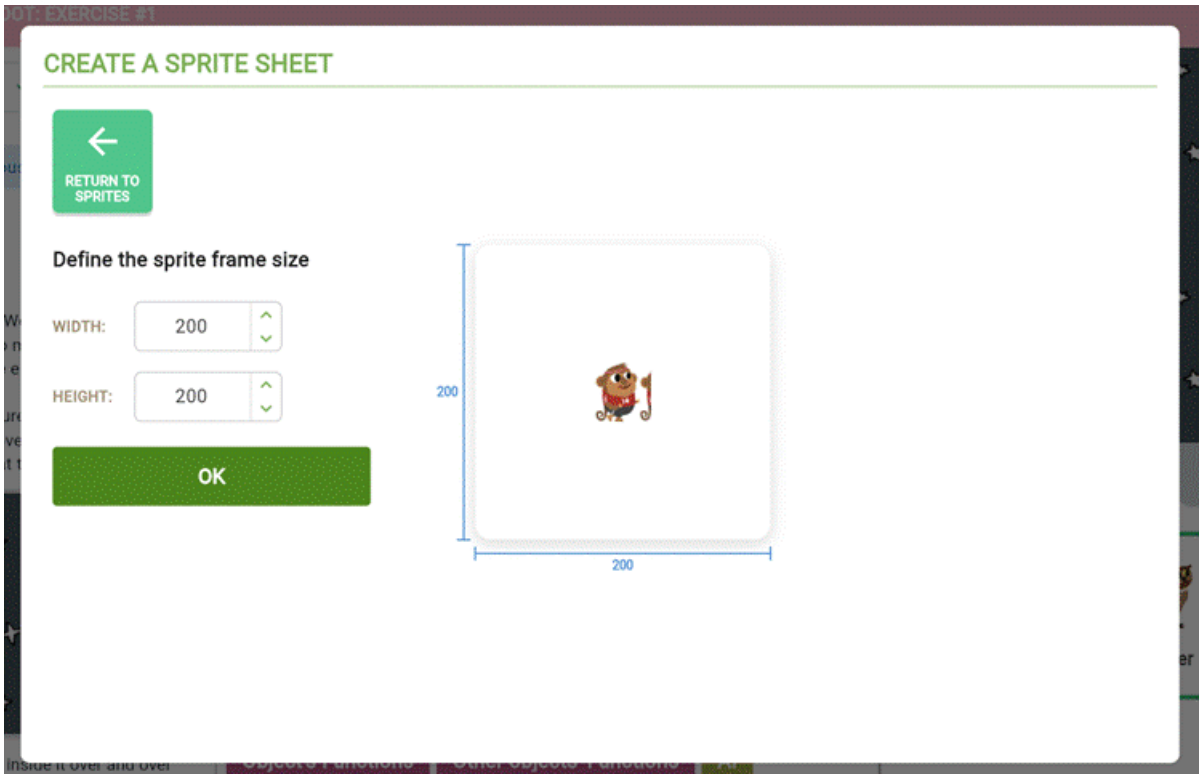
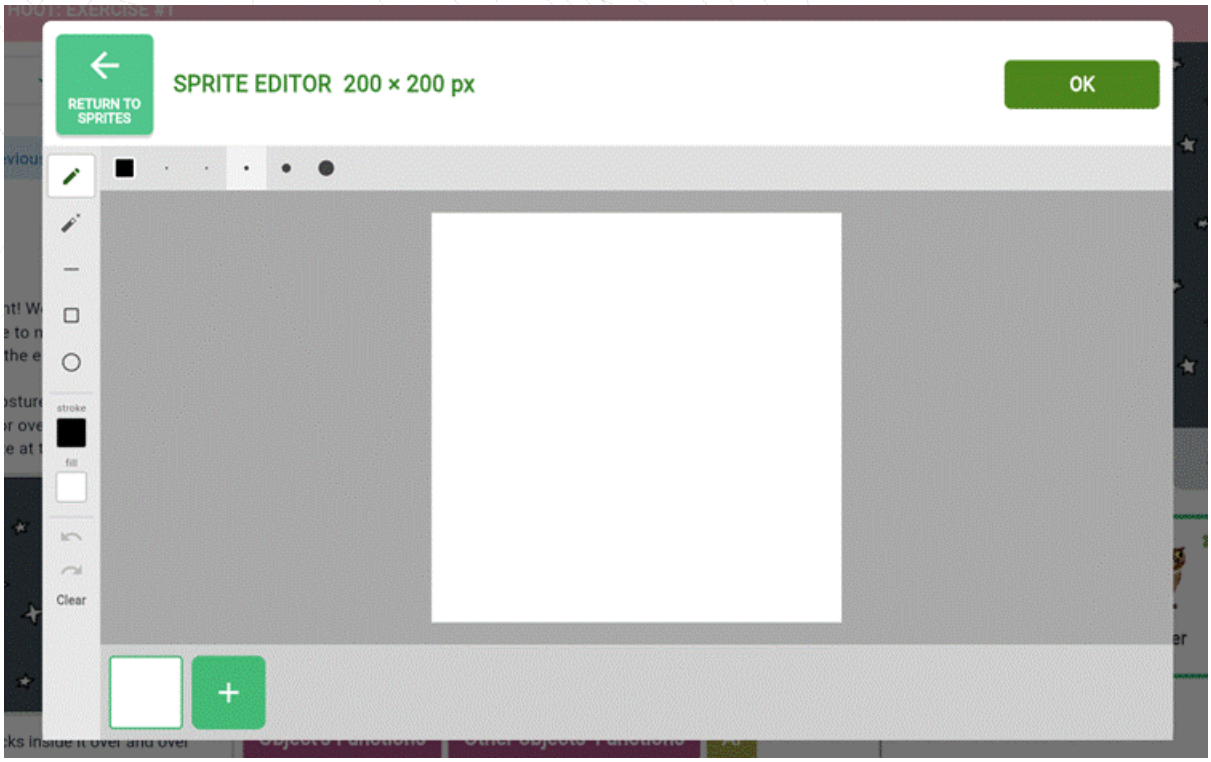


Figure 4.80: Create a New Sprite Sheet



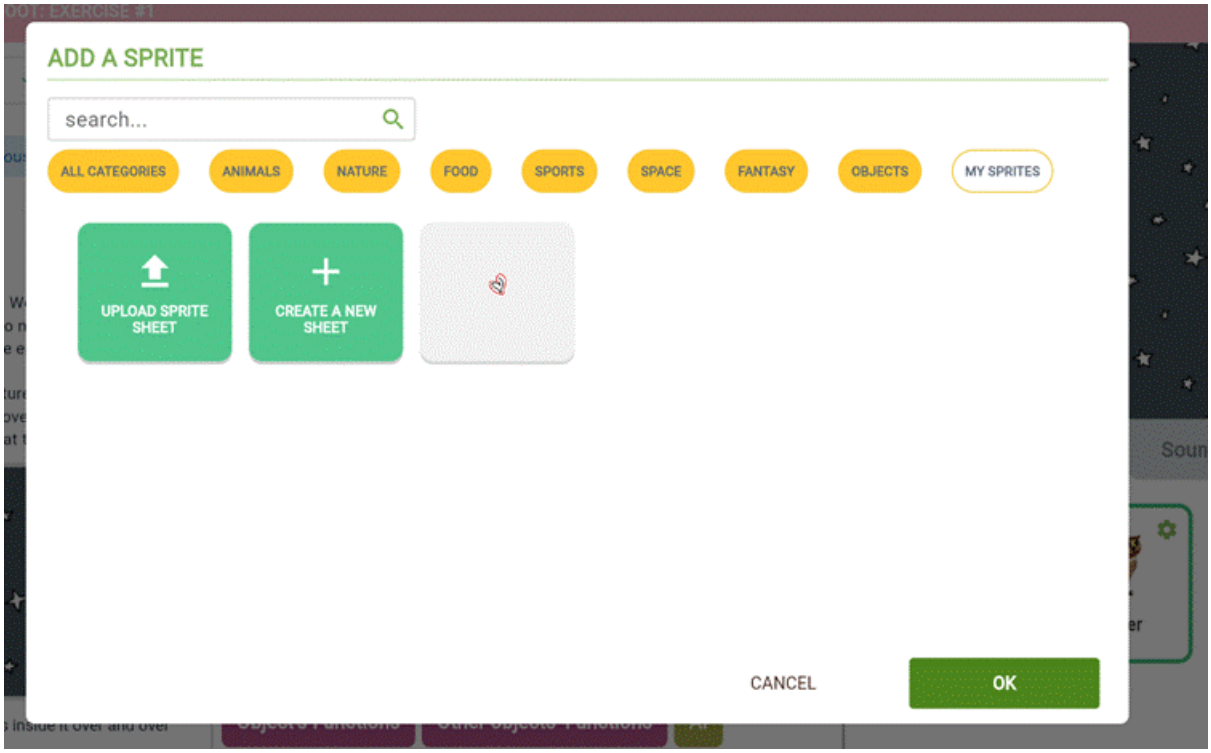


Figure 4.81: Create a New Sprite Sheet - view 2

Widgets Tab

Clicking ADD NEW under the Widgets tab opens an "Add a Widget" modal presenting six widget types as icon cards: Counter, Text, Timer, Clock, Button, and Dialog, plus a Webcam widget. The currently selected widget type is highlighted with a green border and checkmark. Clicking ADD places the widget onto the game canvas.

Each widget type, once added, appears as its own selectable entity in the right panel and in the game canvas, and has its own dedicated block workspace in the center editor.

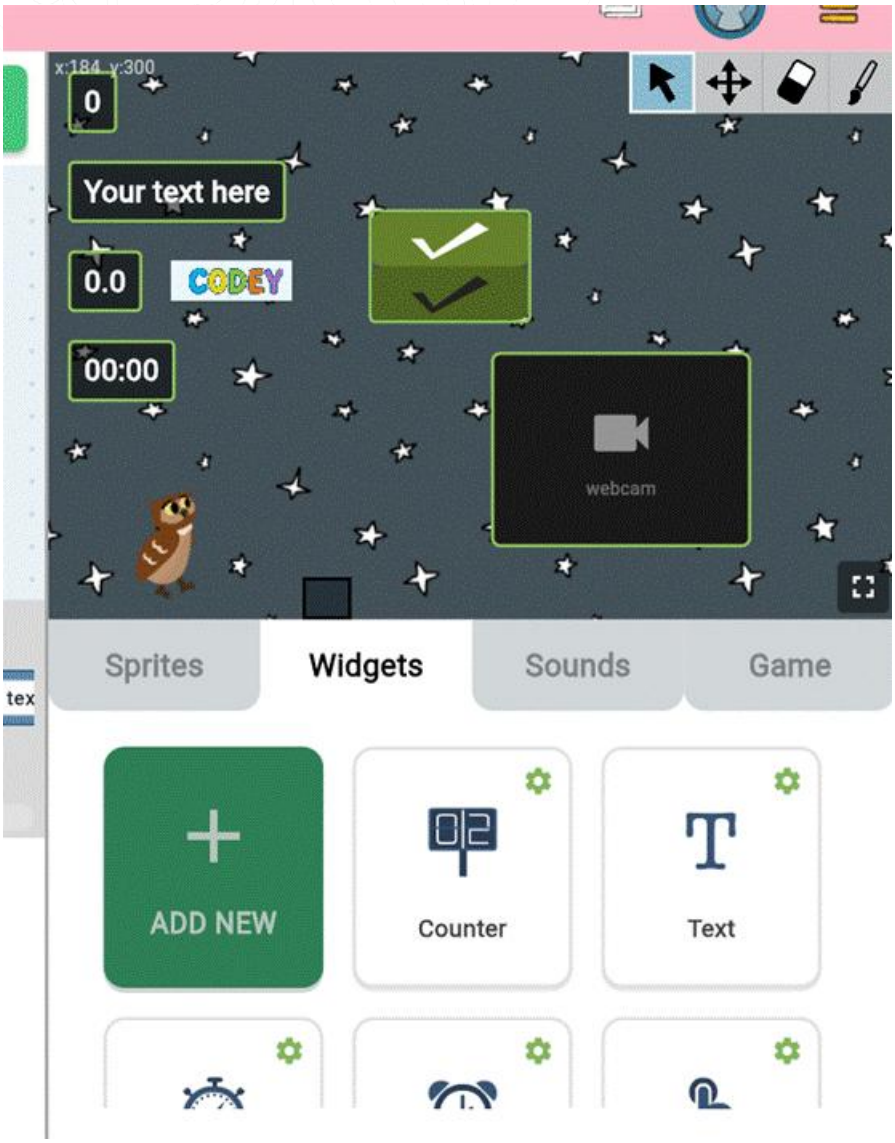


Figure 4.82: Widgets Tab

When a widget is selected, the center panel switches to show that widget's programming context its name and icon appear at the top, and its event/action blocks are loaded:

- **Counter**: Displays a numeric counter on screen. Its workspace shows an "On Run" block with a "Set counter: counter To 0" block pre-attached. The counter appears on the game canvas as a small "0" box.

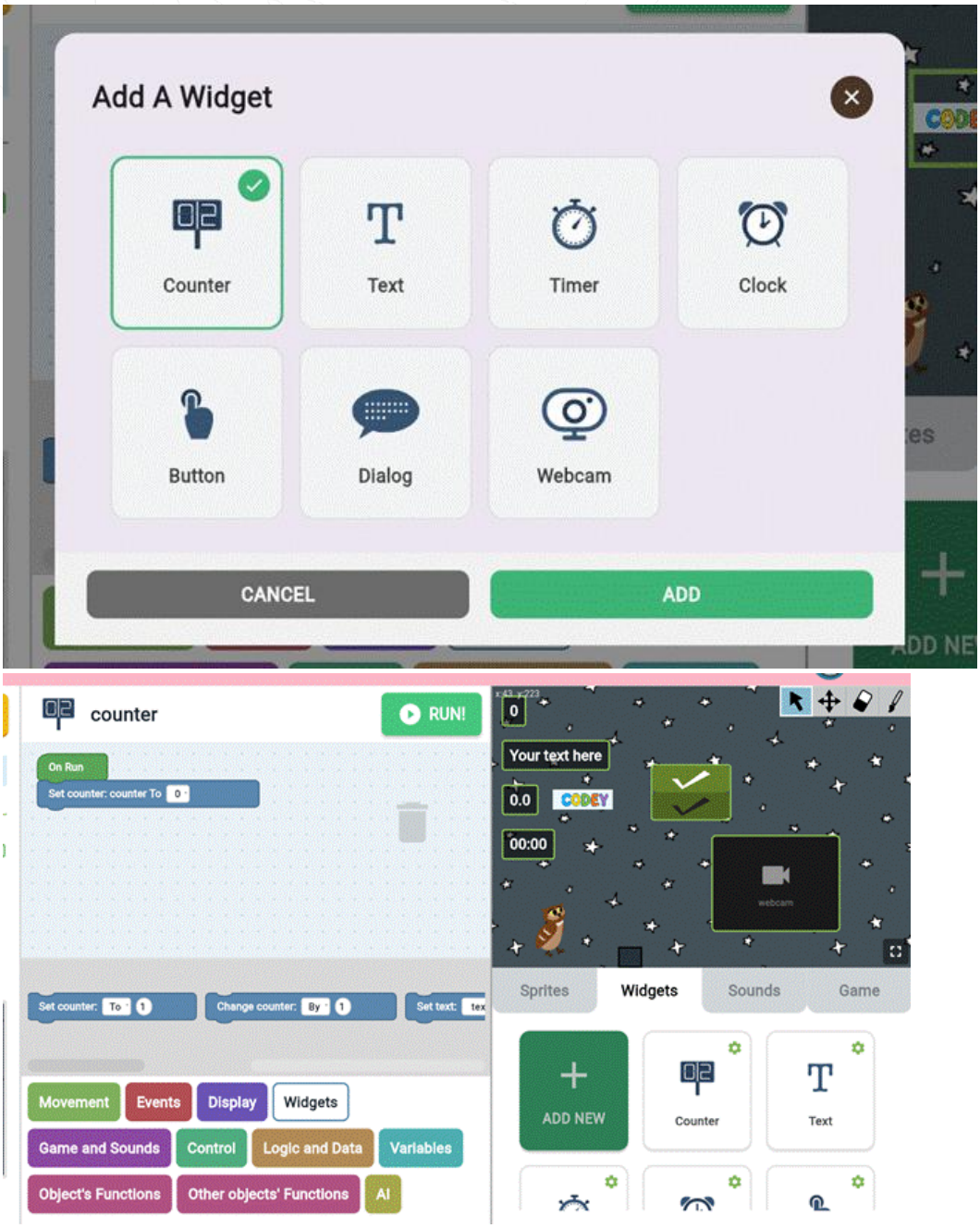


Figure 4.83: Widgets Tab - view 2

- **Text**: Displays a text label. Its workspace shows "On Run" with "Set text: text To 0." The text widget appears on the canvas as a "Your text here" label.

Figure 4.84: Text: Displays a text label

- **Timer**: A countdown or count-up timer. Its workspace shows "On Run", "Set timer: timer To 0", and an "On End" event block that fires when the timer reaches zero. Appears on canvas as "0.0."

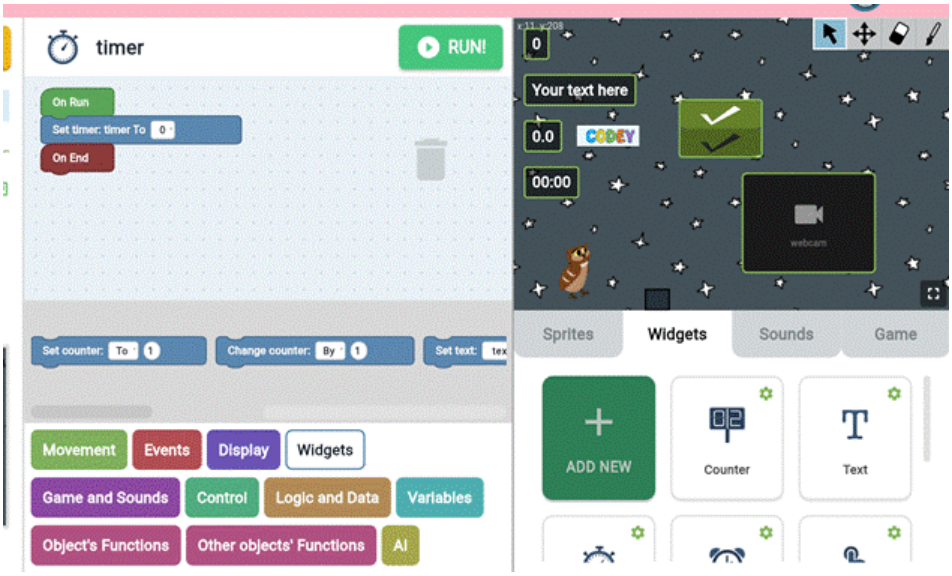


Figure 4.85: Timer: A countdown or count-up timer

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

- **Button**: An interactive clickable button. Its workspace shows "On Run" along with "On Click" and "On Down" event blocks, allowing the student to trigger actions when the button is pressed. Appears as a checkmark button on canvas.

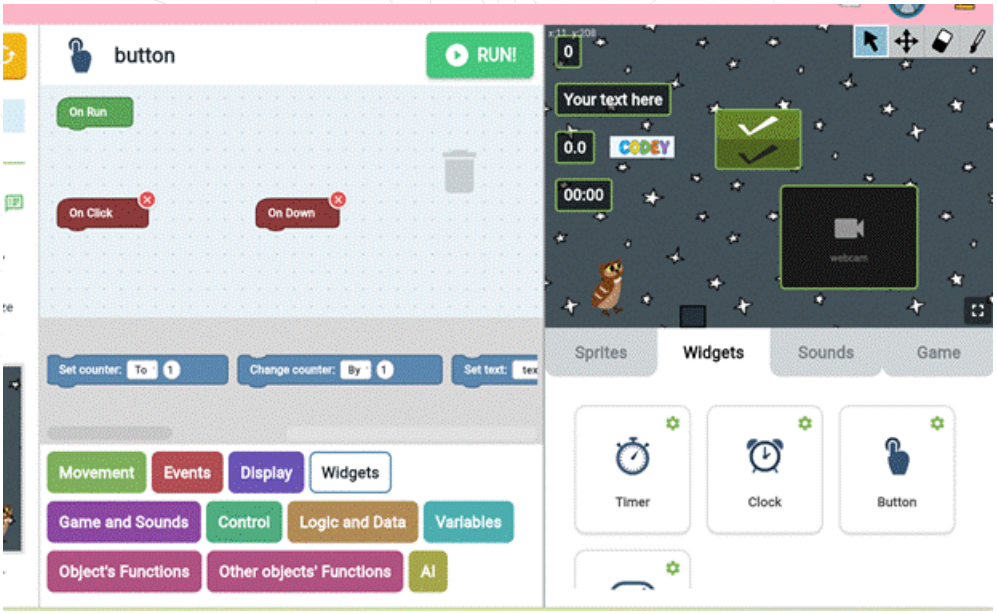


Figure 4.86: Button: An interactive clickable button

- **Webcam**: Places a live webcam feed window onto the game canvas. Its workspace shows "On Run" with an empty block area. The webcam widget appears on canvas as a dark rectangle with a camera icon labeled "webcam." This widget is the key component of the AI exercises, as it captures the student's body-pose input for the gesture recognition model.

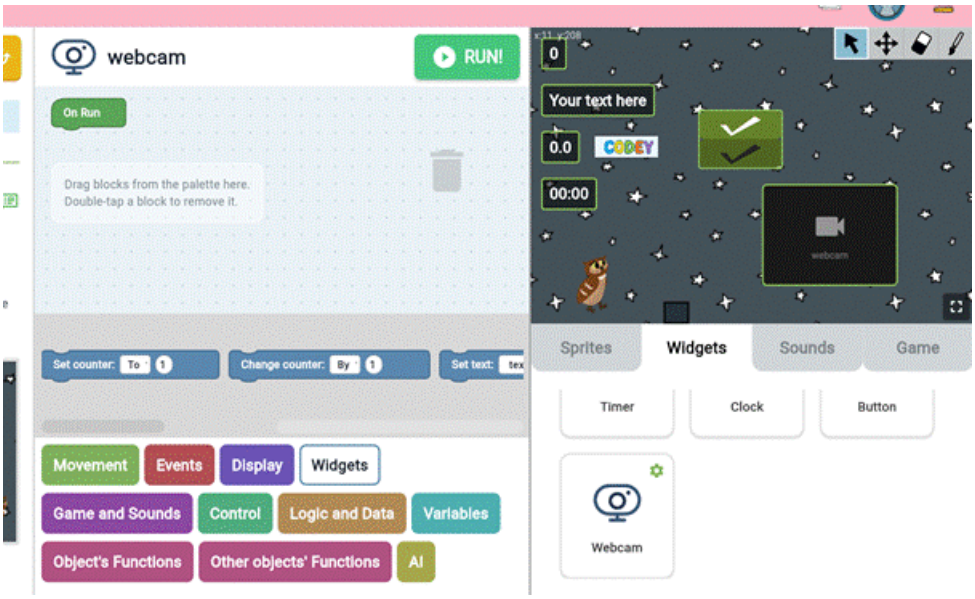


Figure 4.87: Webcam: Places a live webcam feed window onto the game canvas

All added widgets are visible simultaneously on the game canvas and listed under the Widgets tab with gear icons for individual settings.

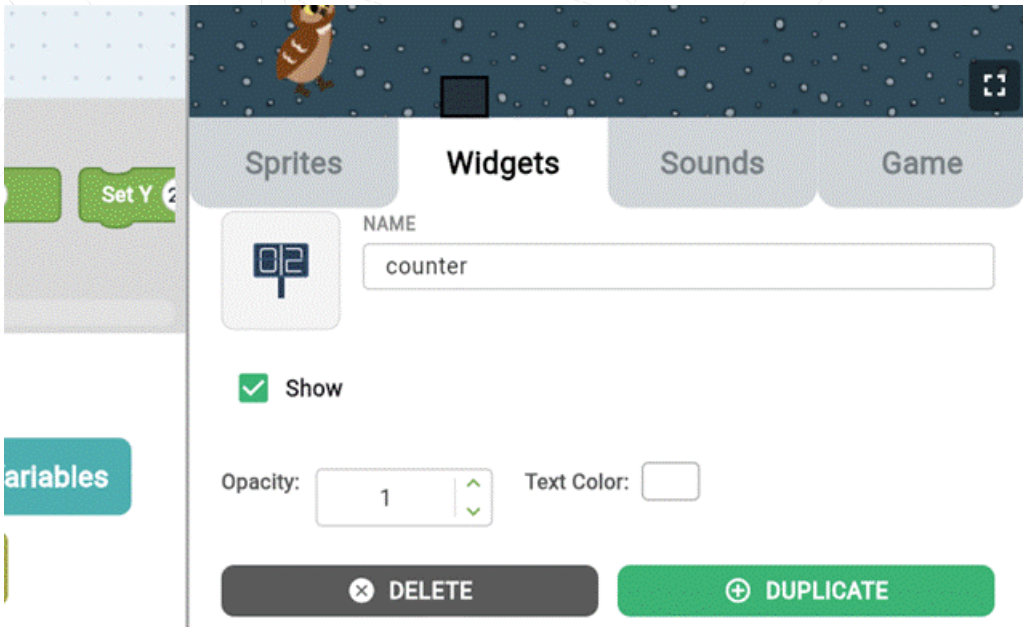


Figure 4.88: All added widgets are visible simultaneously on the game canvas and listed under the

Game Tab — Game Settings

The Game tab at the far right of the panel exposes global game configuration settings under a "Game Settings" header:

- **Background**: Shows the currently selected background name (e.g. "starry_night") with a "Change" link that opens a "Choose a Background" modal. The modal displays a grid of built-in background options including: starry_night, jungle, jungle_night, ocean_night, bricks, bricks_2, grass, grass_2, road, clouds_bright, bubbles, winter, underground, and sunny. An "Upload Background" card at the top left allows the student to supply a custom image. The currently selected background

is marked with a green checkmark. Selecting a new background and clicking OK immediately updates the game canvas preview.

- **Camera Target**: A dropdown for selecting which sprite the camera should follow during gameplay (defaults to None).
- **World Width / World Height**: Dropdowns for setting the dimensions of the game world (both default to 600×400).
- **Gravity**: A numeric spinner controlling the gravity strength applied to sprites (defaults to 1800).
- **Physics**: A dropdown for selecting the physics mode (defaults to ARCADE).

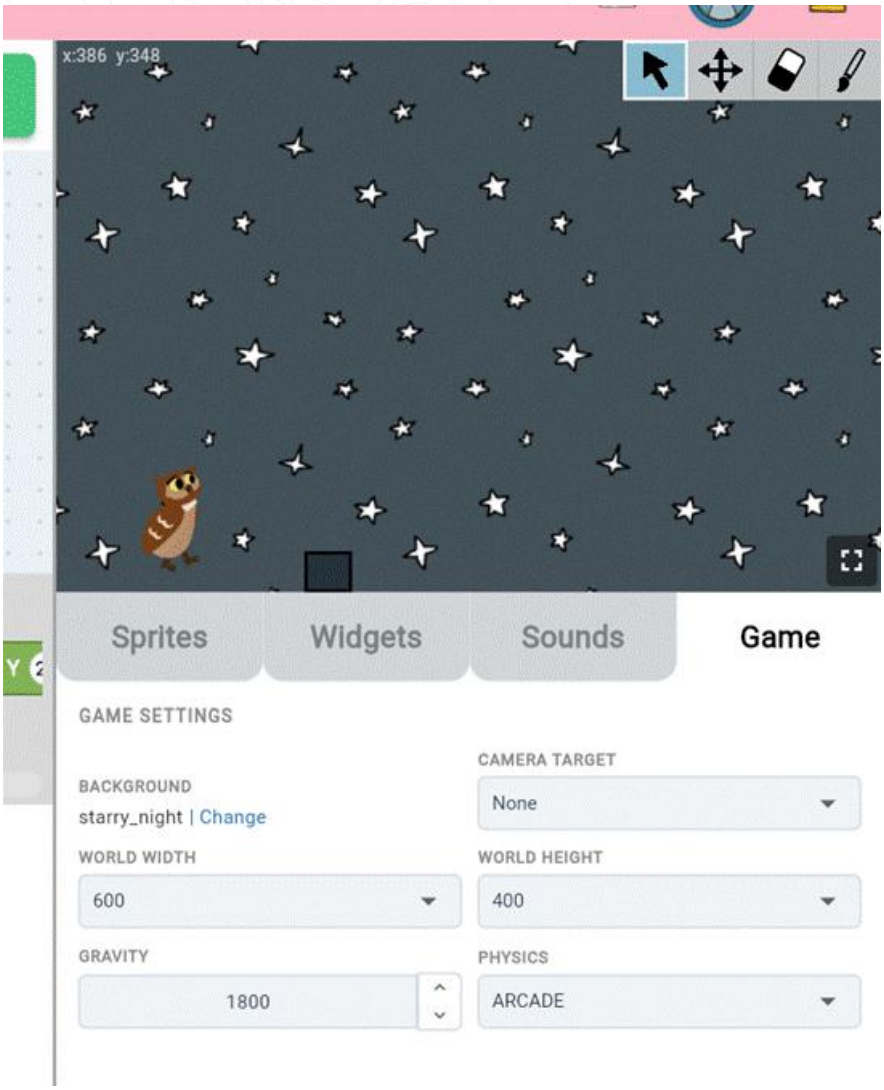
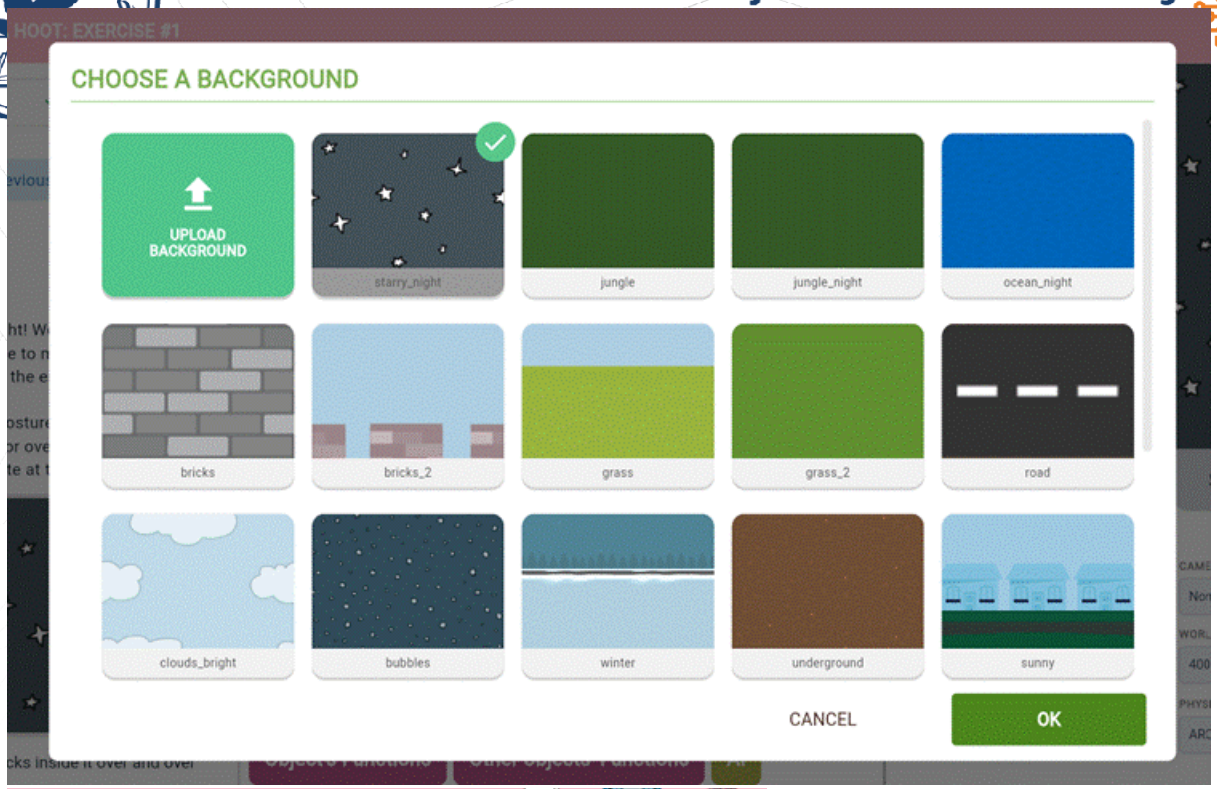


Figure 4.89: Physics: A dropdown for selecting the physics mode (defaults to ARCADE)

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu



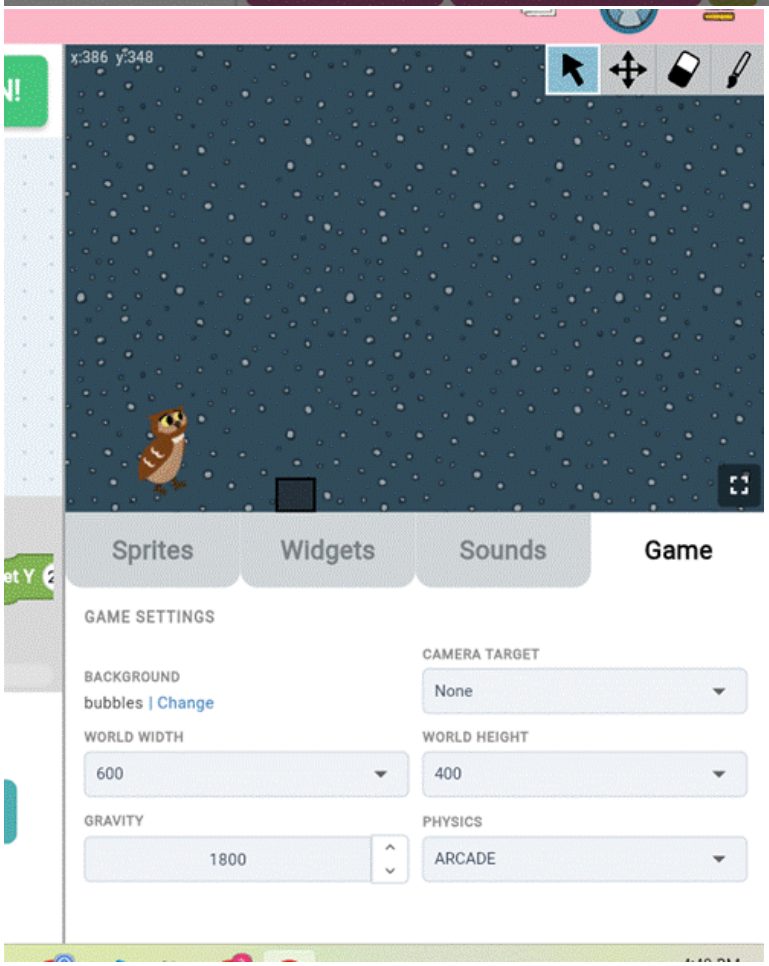


Figure 4.90: Physics: A dropdown for selecting the physics mode (defaults to ARCADE) - view 2

4.9.5 Game Canvas Toolbar

The game canvas in the right panel includes a four-button toolbar in the top-right corner that controls how the student interacts with the canvas in edit mode. Each tool changes the cursor behavior and what clicking or dragging on the canvas does:

Select Tool (arrow cursor icon): The default mode. Clicking on any sprite or widget on the canvas selects it, highlighting it with a green bounding box. The selected object's programming context loads in the center editor panel, allowing the student to write or edit its blocks. The current x/y coordinates of the cursor are shown in the top-left corner of the canvas in real time.

Figure 4.91: Game Canvas Toolbar

Move Tool (four-direction arrow icon): Activates drag-to-reposition mode. Clicking and dragging any sprite or widget repositions it freely anywhere on the canvas. The object's new coordinates update live and can be used as reference for Set X / Set Y blocks in the code.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.92: Game Canvas Toolbar - view 2

Erase Tool (eraser icon): Activates tile erasing mode. In this mode, clicking or dragging across the canvas removes tiles that were placed using the draw tool. This is used to clear individual cells from tile-based layouts painted onto the game world.

Draw Tool (pencil icon): Activates tile painting mode. When selected, a color palette appears along the right edge of the canvas offering a set of tile colors (green, brown, dark brown, red/brick, gray, and yellow). The student clicks or drags across the canvas to paint colored tile blocks directly onto the game world, building platforms and obstacles cell by cell. The painted tiles appear as solid colored rectangles on the canvas for example, a row of red brick tiles forming a platform. Using the eraser tool, individual tile cells can be removed to create gaps, such as splitting a continuous platform into two separate segments.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.93: Draw Tool (pencil icon): Activates tile painting mode

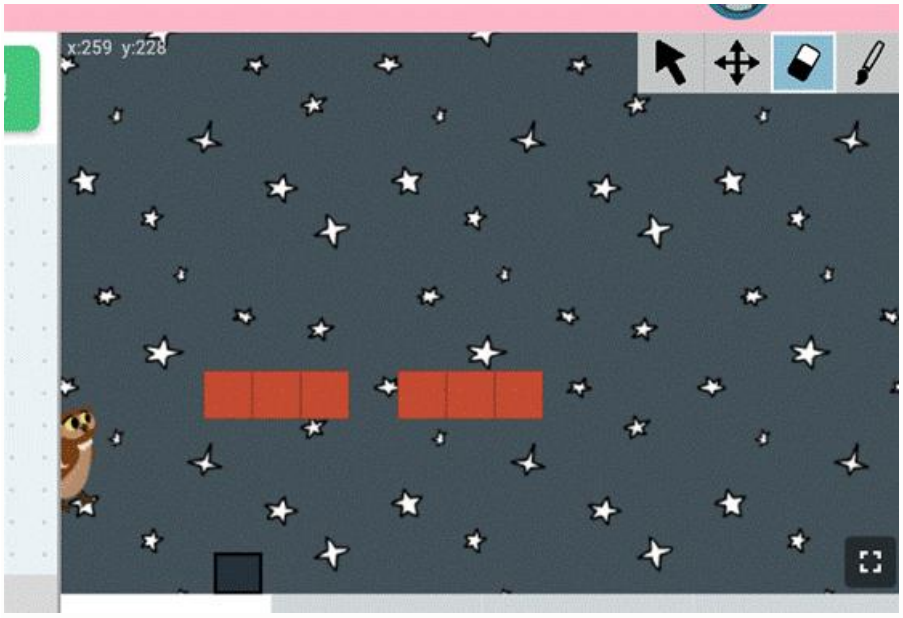


Figure 4.94: Draw Tool (pencil icon): Activates tile painting mode - view 2

Fullscreen Button: A resize icon in the bottom-right corner of the canvas expands the game preview to fill the full browser window, allowing the student to see the complete game world at full scale.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

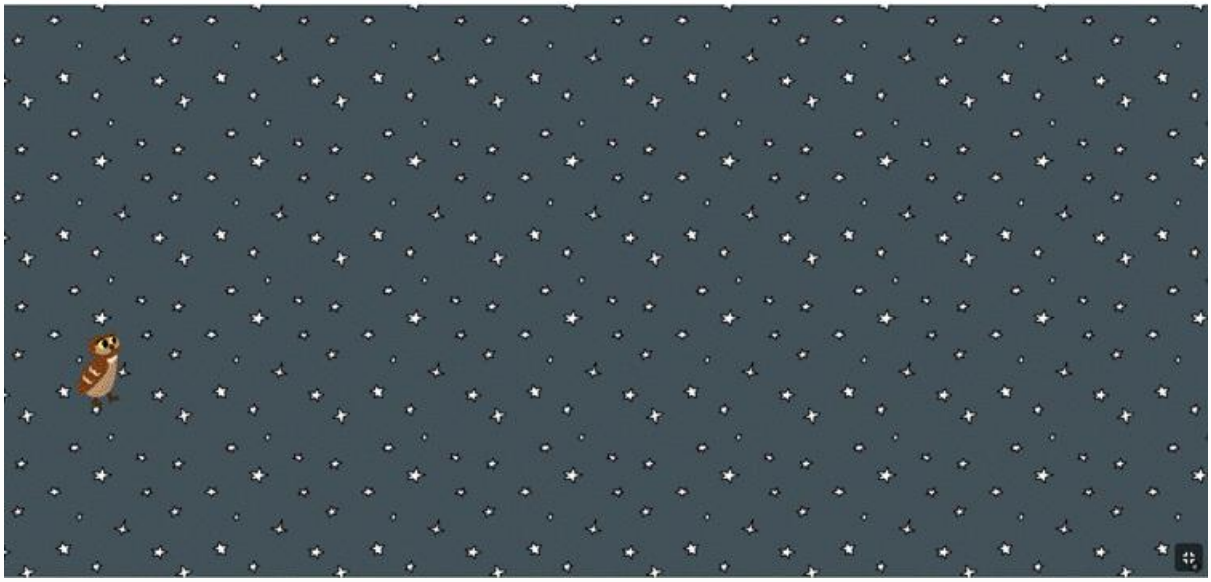


Figure 4.95: Fullscreen Button: A resize icon in the bottom-right corner of the canvas expands the

4.9.6 Left Panel — Overview and Instructions

The left panel is divided into two sections.

Overview introduces the exercise with a short narrative describing what the student will build, an embedded preview of the finished game, a relevant block hint, and a Listen button for text-to-speech read-aloud.

Instructions lists the exercise steps as sequential cards, each with a bold task title and sub-steps shown as radio-button circles. The system auto-detects the student's block arrangement in real time and checks off each sub-step automatically as it is completed — no manual confirmation needed. Steps marked with a gold star require the student to click RUN and observe the result. Once RUN is clicked, the button turns red and changes to STOP, and the game executes live on the canvas. All completed steps are marked with a green checkmark.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

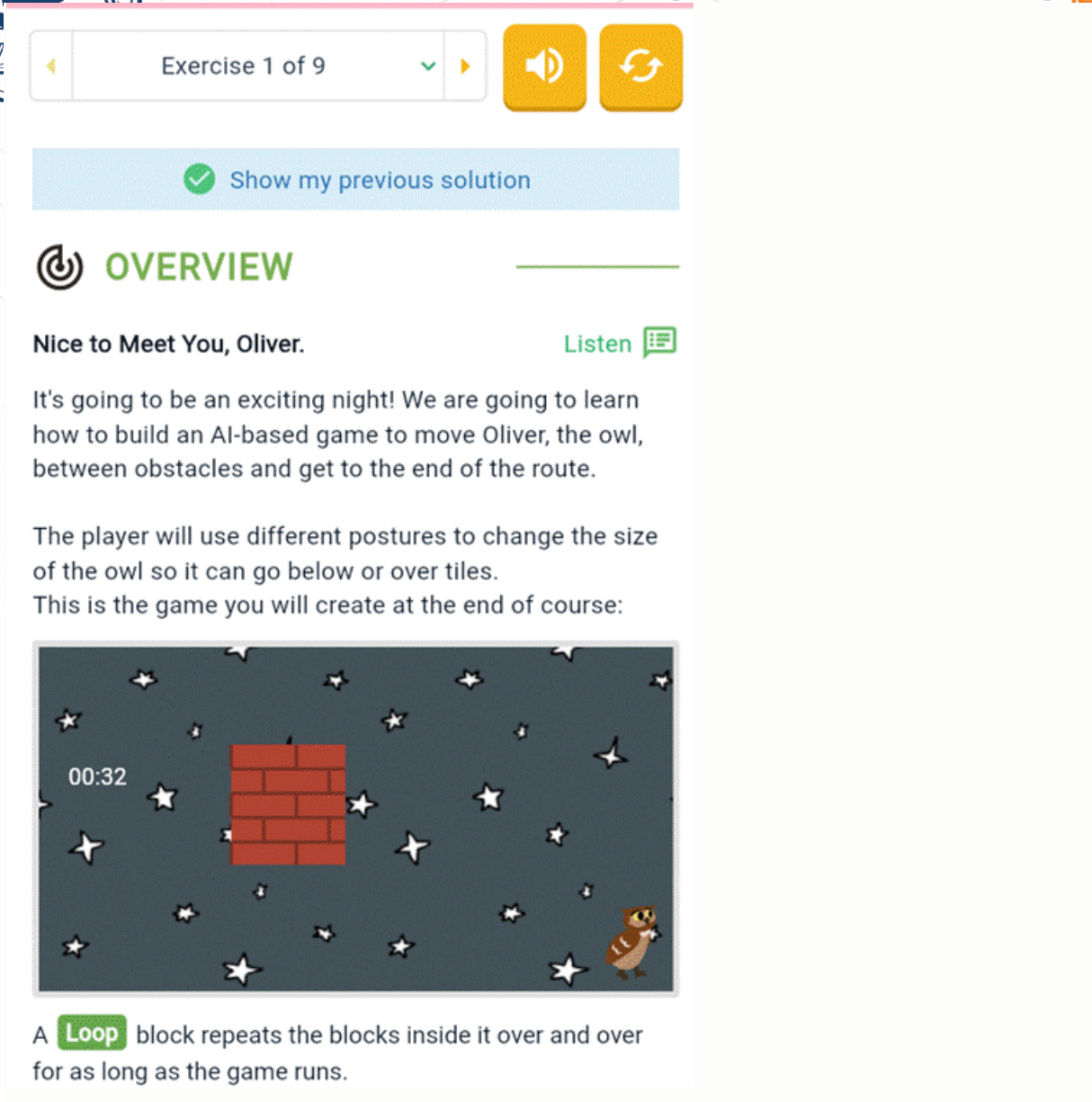


Figure 4.96: Left Panel — Overview and Instructions

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

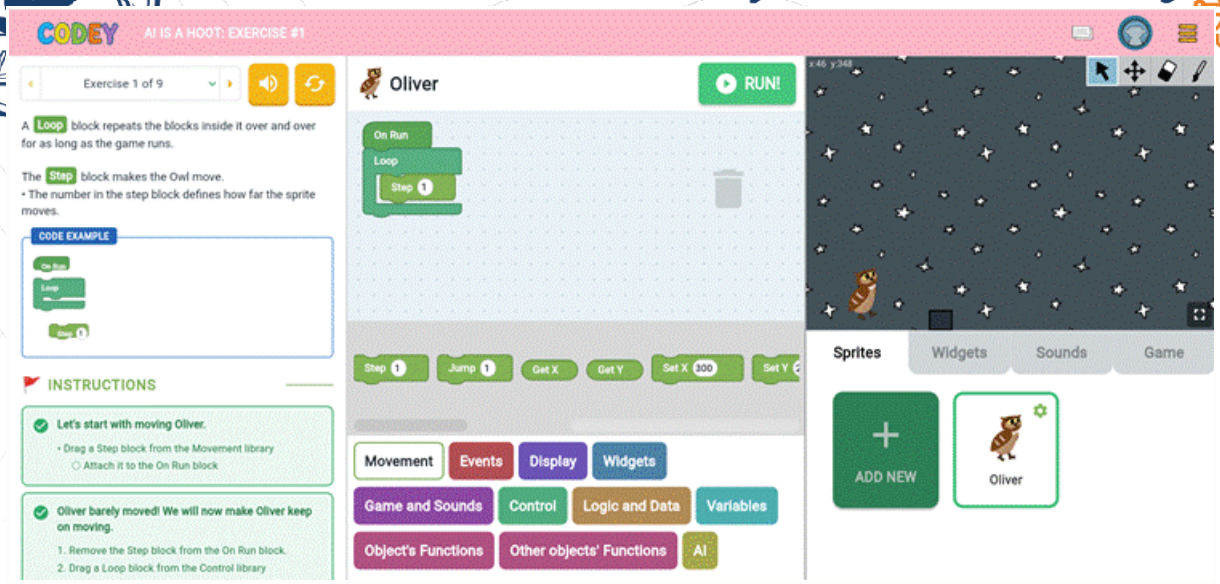


Figure 4.97: Left Panel — Overview and Instructions - view 2

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.98: Left Panel — Overview and Instructions - view 3

4.9.7 Exercise 2 — Training the AI Model

Exercise 2, titled "AI to the Rescue," introduces the AI model training workflow. The overview explains that the student will create a pose classification model to differentiate between two gestures: raising hands and standing straight. A "Deep Dive" expandable panel explains how AI models learn from datasets collections of labeled samples and why recording varied poses improves recognition accuracy.

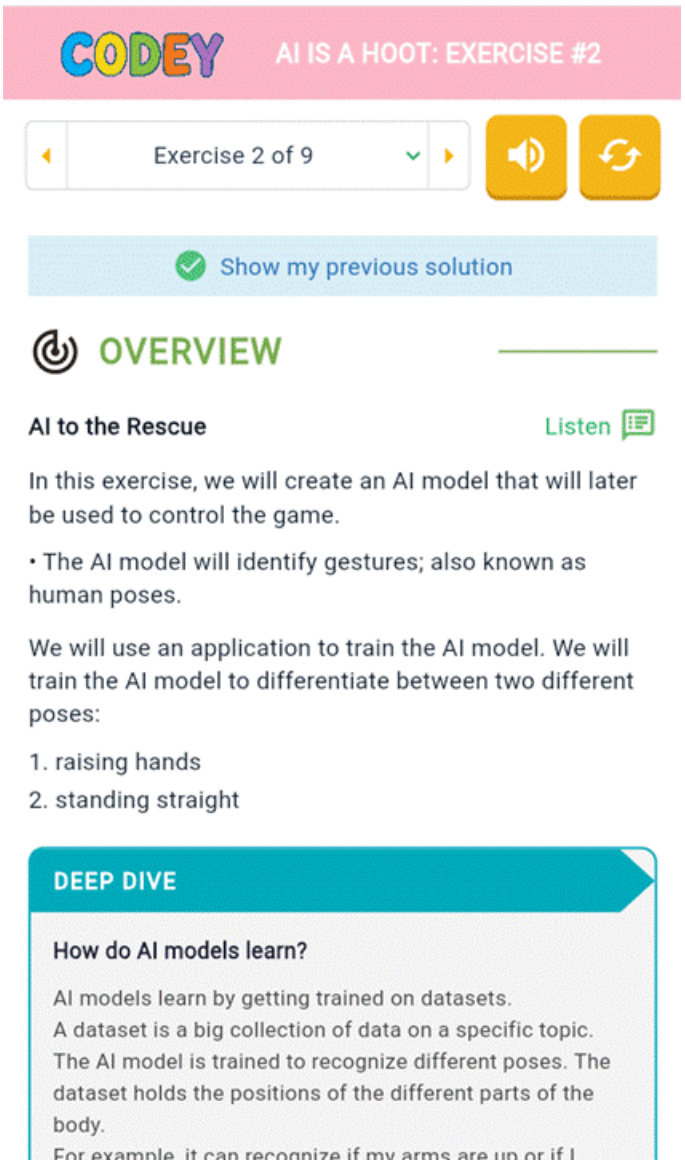


Figure 4.99: Exercise 2 — Training the AI Model

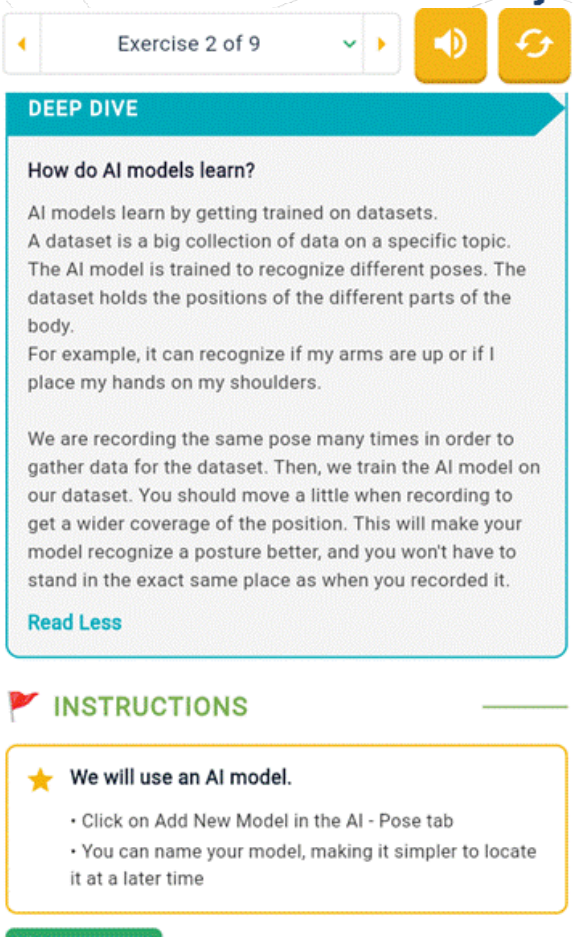


Figure 4.100: Exercise 2 — Training the AI Model - view 2

AI - Pose Tab

Exercise 2 introduces a fifth tab in the right panel: **AI - Pose**. It contains an "ADD NEW MODEL" button and a list of saved models. Clicking it opens the **New AI Model - Pose** modal, which guides the student through three stages shown as a left-to-right flow with arrows:

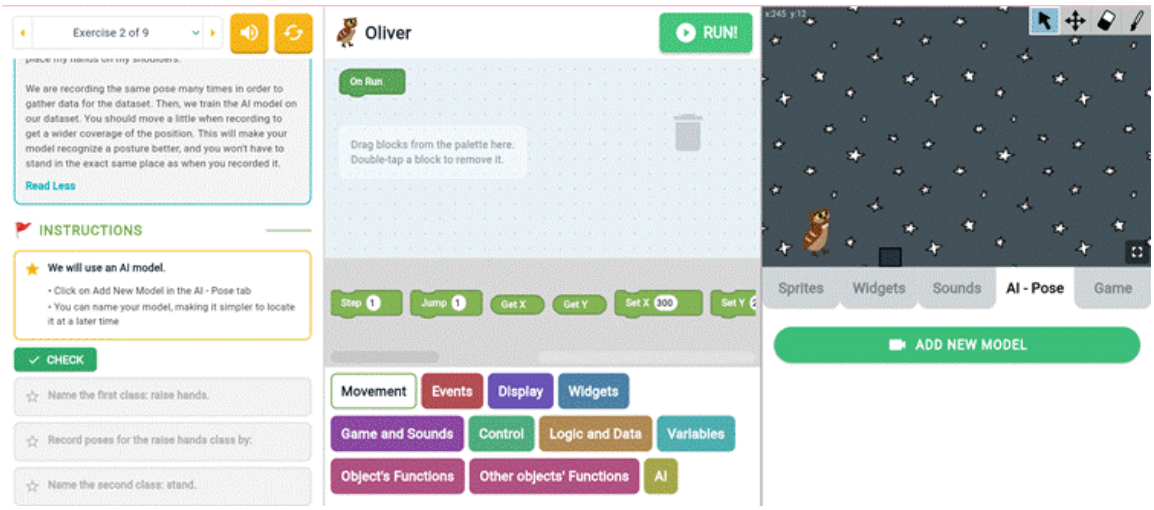


Figure 4.101: AI - Pose Tab

1. Recording Samples

The modal shows a named class (e.g. "raise hands") with an "Add Samples" section containing a RECORD button and a timer icon. When recording starts, the webcam activates and displays a live feed with a green skeletal pose overlay drawn over the student's body — tracking key joints in real time. A yellow "HOLD TO RECORD" button captures frames continuously. The student records multiple samples per class by moving slightly between captures to improve coverage. Each recorded frame appears as a thumbnail in a grid beside the live feed. The sample count is shown above the grid (e.g. "30 Samples"). A "Remove All Samples" link clears the set. The student repeats the same process for the second class ("stand"), resulting in two named classes each with their own sample sets.

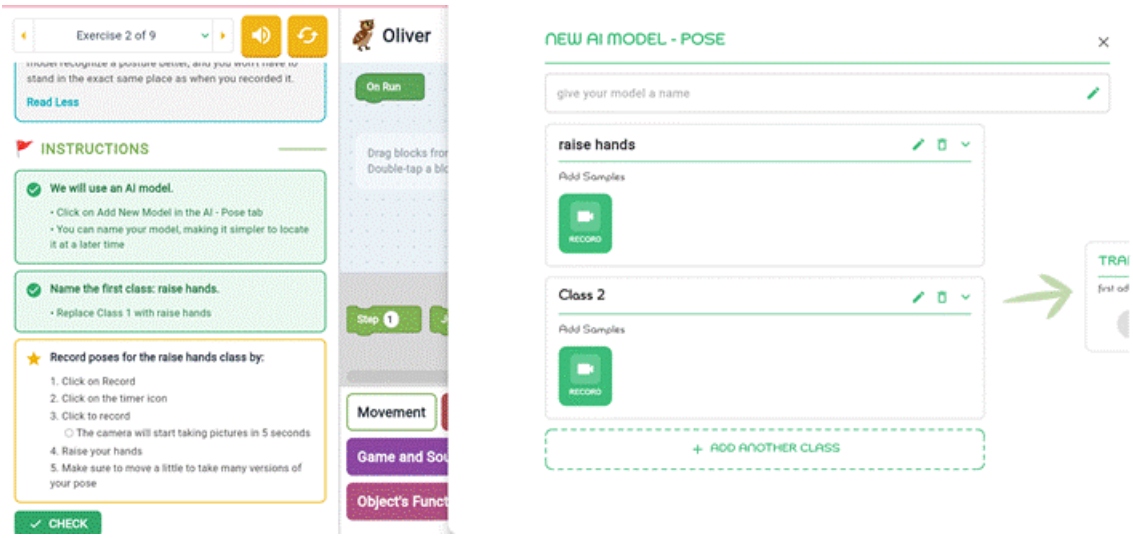


Figure 4.102: Recording Samples

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

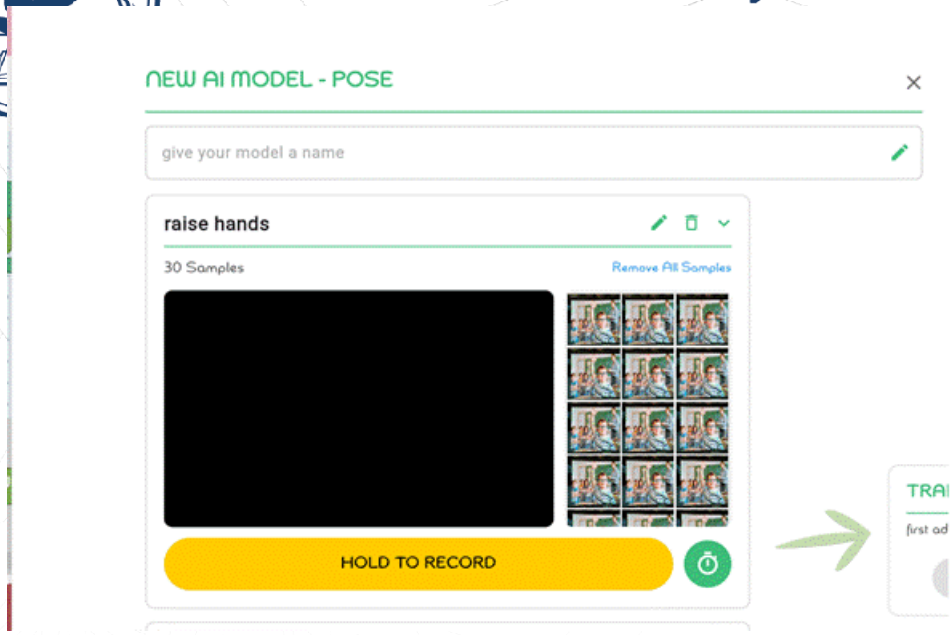


Figure 4.103: Recording Samples - view 2

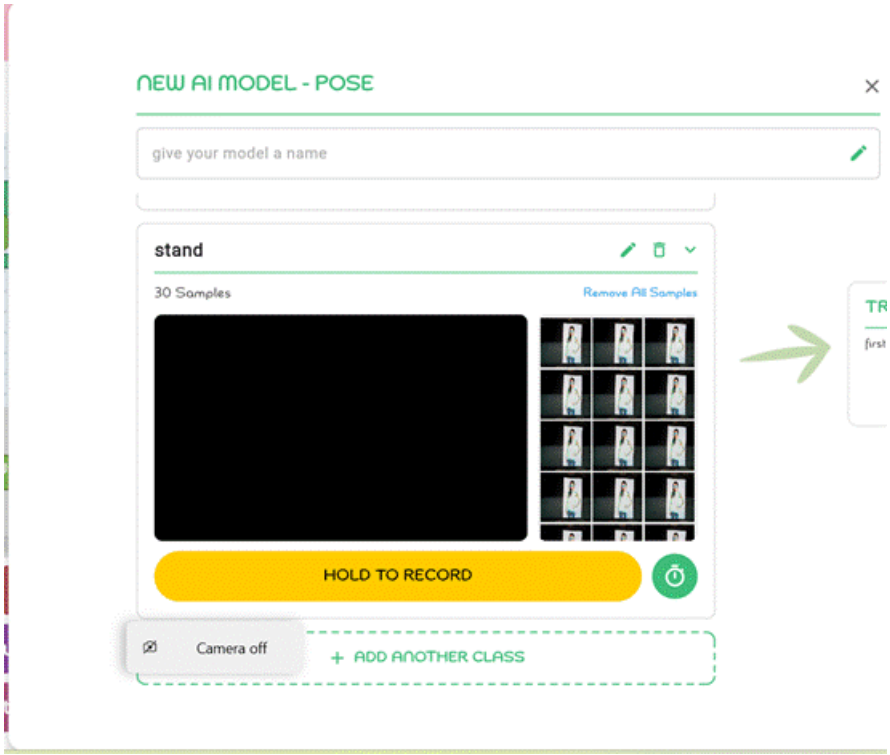


Figure 4.104: Recording Samples - view 3

2. Training

Clicking "Train Model" starts the training process, shown as a progress bar with a percentage indicator (e.g. "22%... Training..."). Once complete, the Training card displays "Training complete!" and the button changes to "TRAINED!".

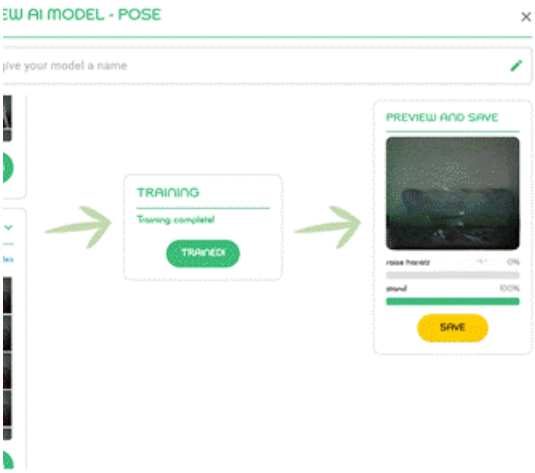


Figure 4.105: Training

3. Preview and Save

After training, the Preview and Save panel activates. The live webcam feed resumes with the pose skeleton overlay, and two confidence bars show the model's real-time prediction one bar per class (e.g. "raise hands 60%" / "stand 40%"), updating live as the student changes pose. When confidence reaches a satisfactory level (the instructions suggest above 95% per pose), the student clicks Save. The saved model then appears in the AI - Pose tab under the exercise, labeled with its class tags (e.g. "raise hands" / "stand") and a "Give It A Try" button.

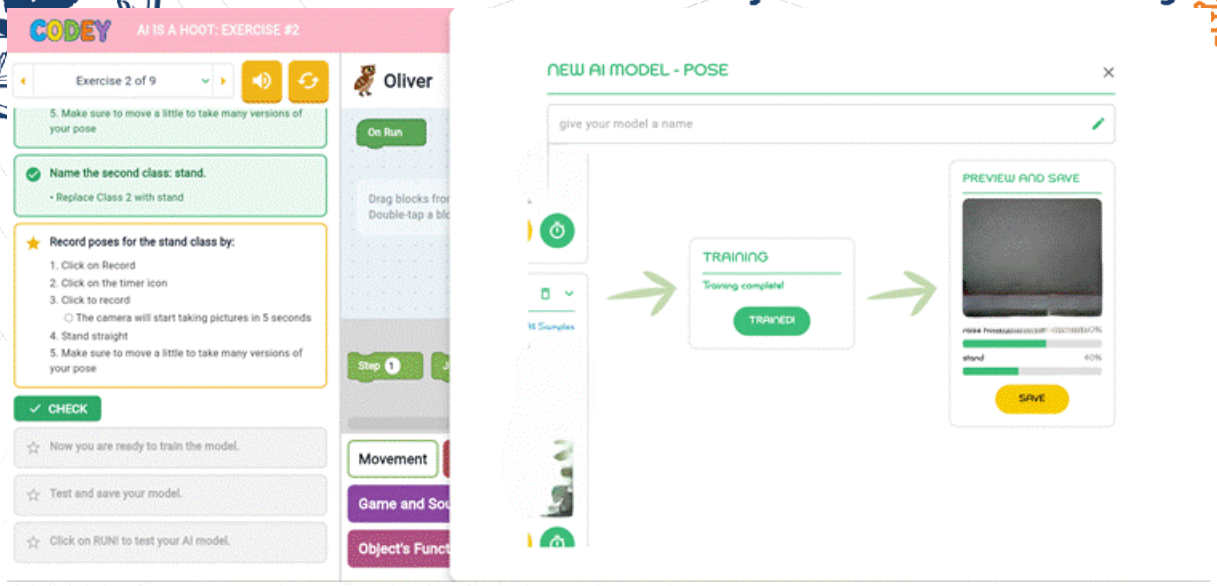


Figure 4.106: Preview and Save

Exercise Completion

Once all steps are checked off adding the model, naming both classes, recording samples for each, training, testing, and saving a "Great Job! You completed Exercise 2!" card appears at the bottom of the instructions panel with a NEXT button to proceed to Exercise 3.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

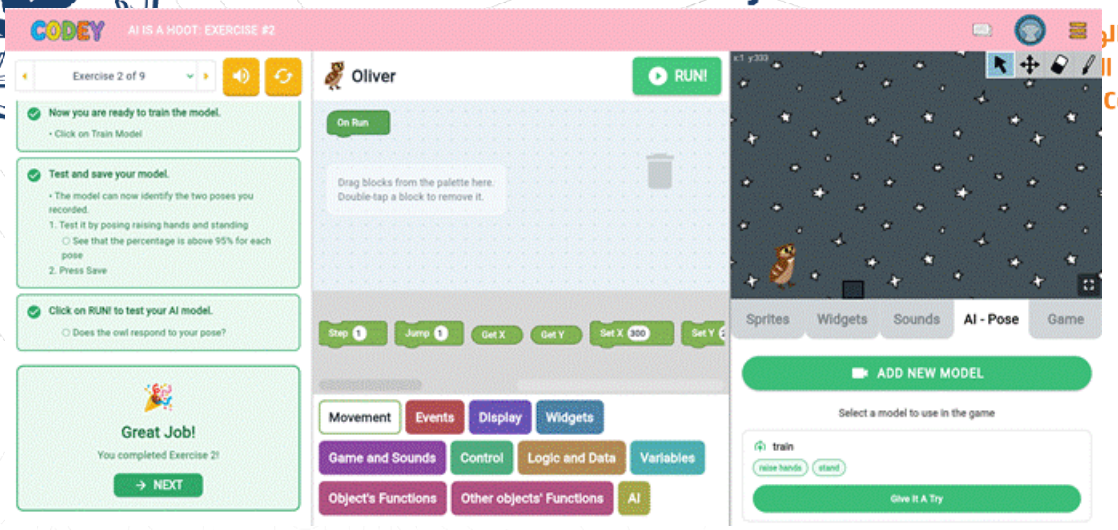


Figure 4.107: Exercise Completion

4.9.8 Exercise 3 Connecting the AI Model to the Game

Exercise 3, titled "Add the Model," connects the trained pose model to Oliver's behavior. The overview explains that the **On Prediction** block from the AI category is the key link it fires as an event whenever the model detects a specific pose, allowing the student to trigger game actions in response to real-time body position.

The instructions guide the student through four steps, all auto-checked:

1. **Choose the AI model** from the previous exercise by clicking on it in the AI - Pose tab.
2. **Add the Webcam widget** from the Widgets tab and drag it to the upper area of the game canvas. The live webcam feed then appears in the top-left corner of the game preview, showing the student's face and upper body with the green pose skeleton overlay active during gameplay.
3. **Click the Oliver sprite** to open its code, then drag an **On Prediction** block from the AI category and set its dropdown to "raise hands." Inside the On Prediction block, attach a **Jump** block from the Movement library.
4. **Click RUN** and raise hands to make Oliver jump over the tiles.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.108: Click RUN and raise hands to make Oliver jump over the tiles

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.9.9 Exercise 4 — Displaying the Prediction Confidence

Exercise 4, titled "What's the Prediction?", introduces the **Predict** block. The overview explains that Predict returns the model's confidence percentage for a given pose class, and that placing it inside an **On Update** block which runs every game frame (30–60 times per second) causes a text widget to display the live confidence value in real time as the student moves.

The instructions guide the student through two steps:

1. **Display the percentage** Click the Oliver sprite, drag an **On Update** block from the Events library, drag a **Set text** block from the Widgets library inside it, then drag a **Predict** block from the AI library and attach it to the Set text block with the dropdown set to "raise hands."
2. **Click RUN** Strike different poses and observe that the text widget updates live with the confidence percentage. Oliver jumps only when the prediction reaches 95% or higher. The student then moves Oliver all the way to the right to complete the exercise.

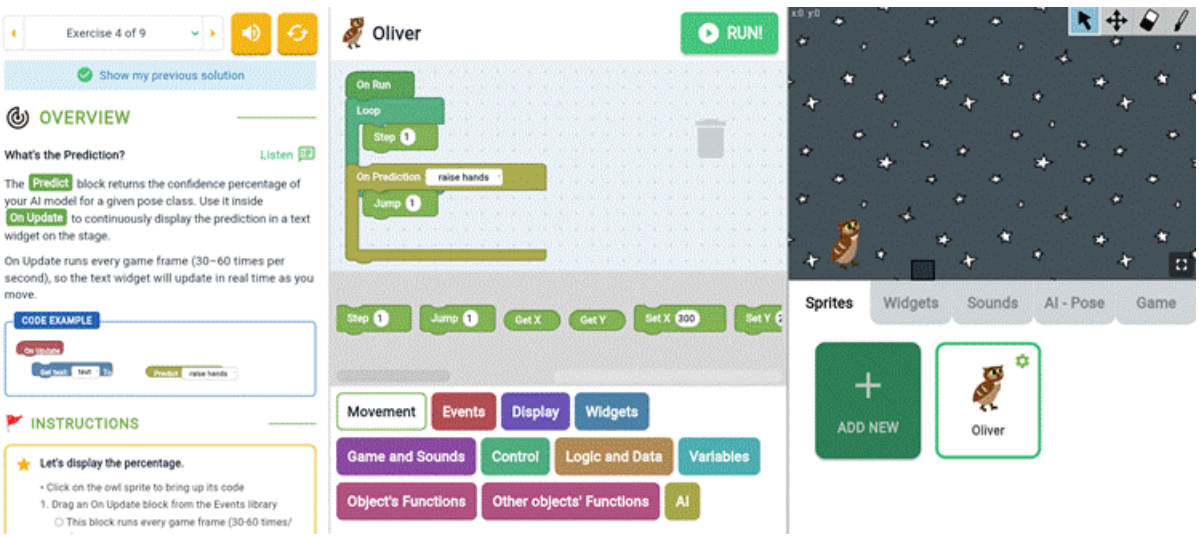


Figure 4.109: Exercise 4 — Displaying the Prediction Confidence

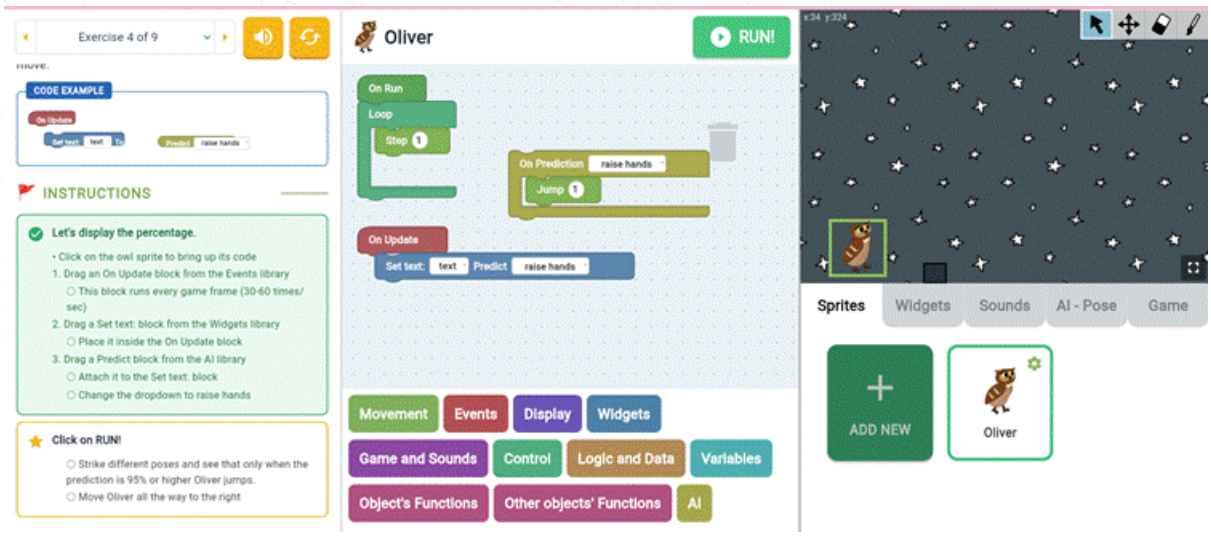


Figure 4.110: Exercise 4 — Displaying the Prediction Confidence - view 2

And other exercise to make it more engaging

4.10 My Classroom

The My Classroom feature allows students to join or create a peer classroom and compete with friends. The classroom interface uses a distinct purple sidebar and light lavender background, separate from the main courses UI. The sidebar shows the classroom name and code (e.g. "Room 7 · Coders / BIT7QX") and four navigation items: Overview, Members, Ranks, and Activity. A "Share class code" button at the bottom opens a modal displaying the classroom code in large bold text with a Copy Code button, allowing the student to invite friends.

Overview

The Overview page shows four summary stats at the top: Total Stars, Levels Done, Members count, and activity This Week. Below them, two panels are displayed side by side:

- **Weekly Challenge**: Set by the classroom creator, showing a shared goal for all members (e.g. "Beat 40 levels together!") with a progress bar tracking the class total (e.g. 28/40 levels completed). A Change button opens the "Set Weekly Challenge" modal where the creator sets a challenge title, selects an optional course from a dropdown (All courses, CodeMonkey Jr, Digital Literacy, Data Course, AI Course), and sets a target level count using +/− buttons. Clicking "Set Challenge" publishes it to all members.
- **Head-to-Head**: Lists active one-on-one challenges between classroom members. Each entry shows the two students' names, the course, a score to beat, and a status badge (Done or Pending).
- **Streak Wall**: A ranked list of all members showing each student's name, avatar, flame icons representing consecutive daily activity days, and their streak count.

Members

The Members page displays all classroom members as cards in a grid, ranked by total points. Each card shows the student's rank number, avatar, name, star count, points total, and day streak. A red "Challenge" button on each card opens a "Challenge [Name]" modal where the student picks a course (CodeMonkey Jr, Digital Literacy, Data Course, AI Course) and clicks "Send Challenge" setting a head-to-head where the challenger's best score becomes the target to beat.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Ranks

The Ranks page shows a per-course leaderboard. A Course dropdown filters by CodeMonkey Jr, Digital Literacy, Data Course, or AI Course. The left panel displays a podium visualization with the top 3 students on gold, silver, and bronze platforms. The right panel lists all ranked members with their points, stars, and level reached.

Activity

The Activity page shows a live feed titled "Latest wins from the class." Each card shows a member's name, the course and level they completed, a star rating, a timestamp (e.g. "just now", "14m ago"), and reaction emoji buttons (fire, clap, star) that classmates can tap to react. The reaction counts update live on each card.

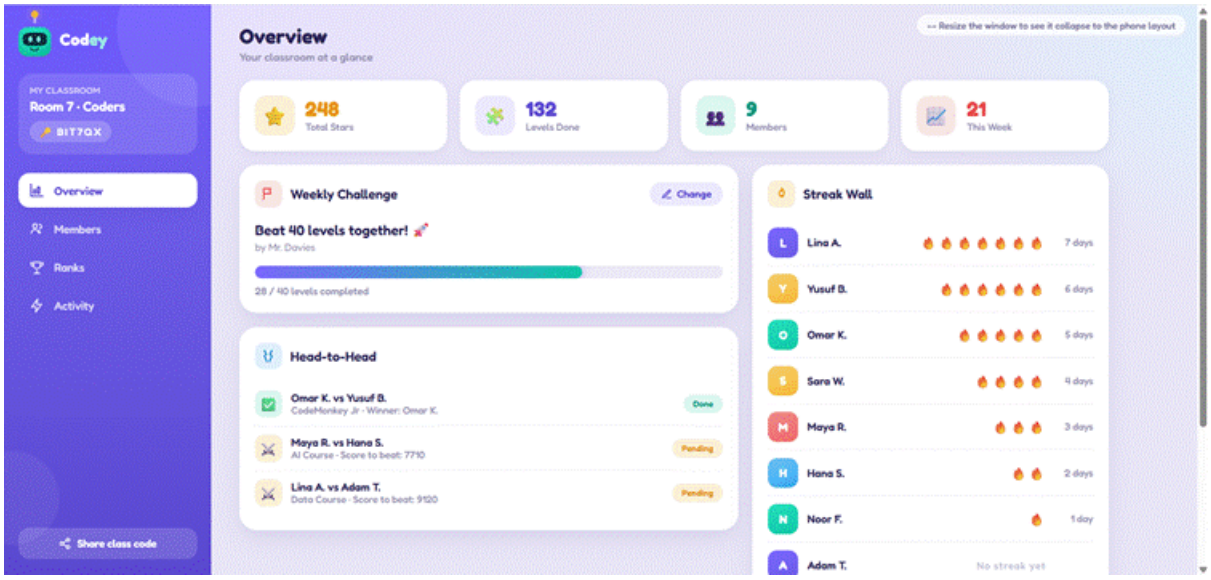
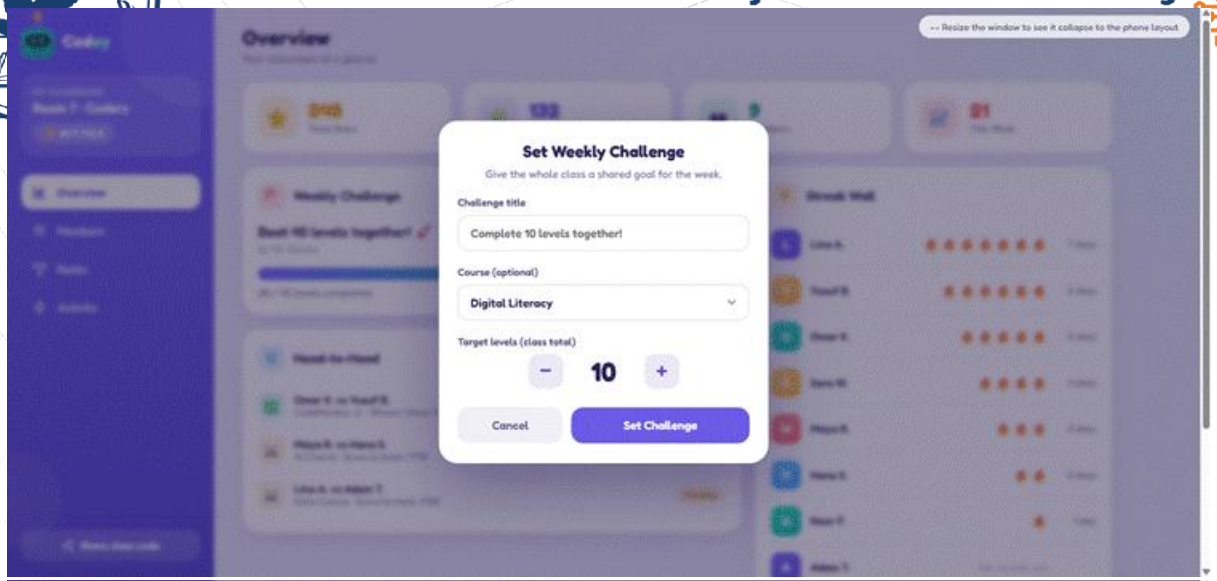
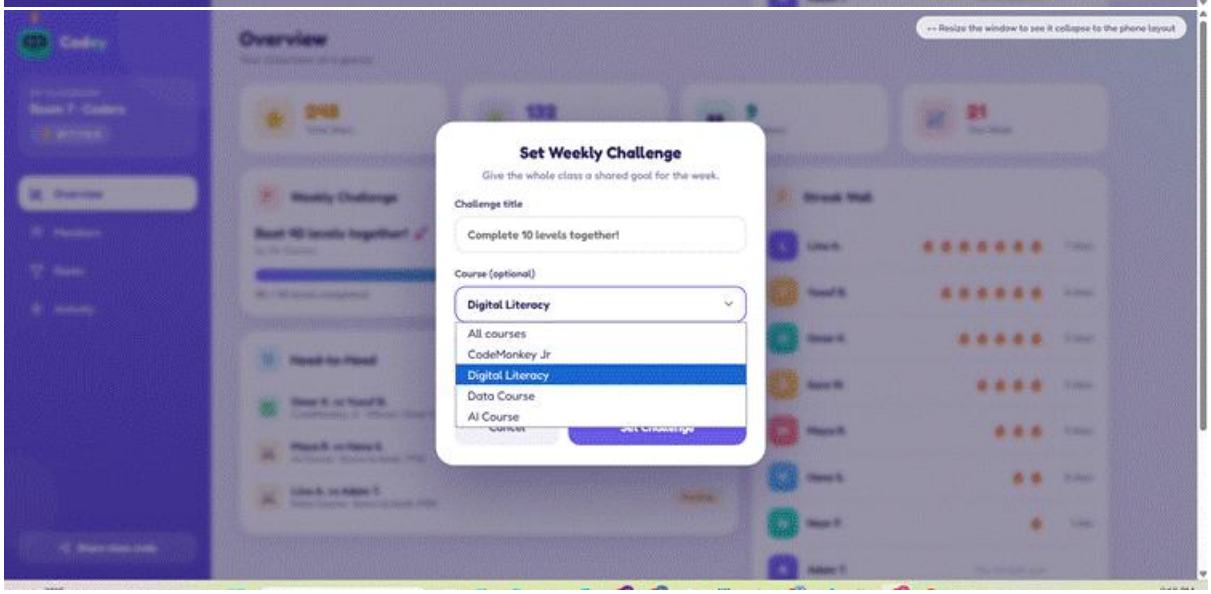
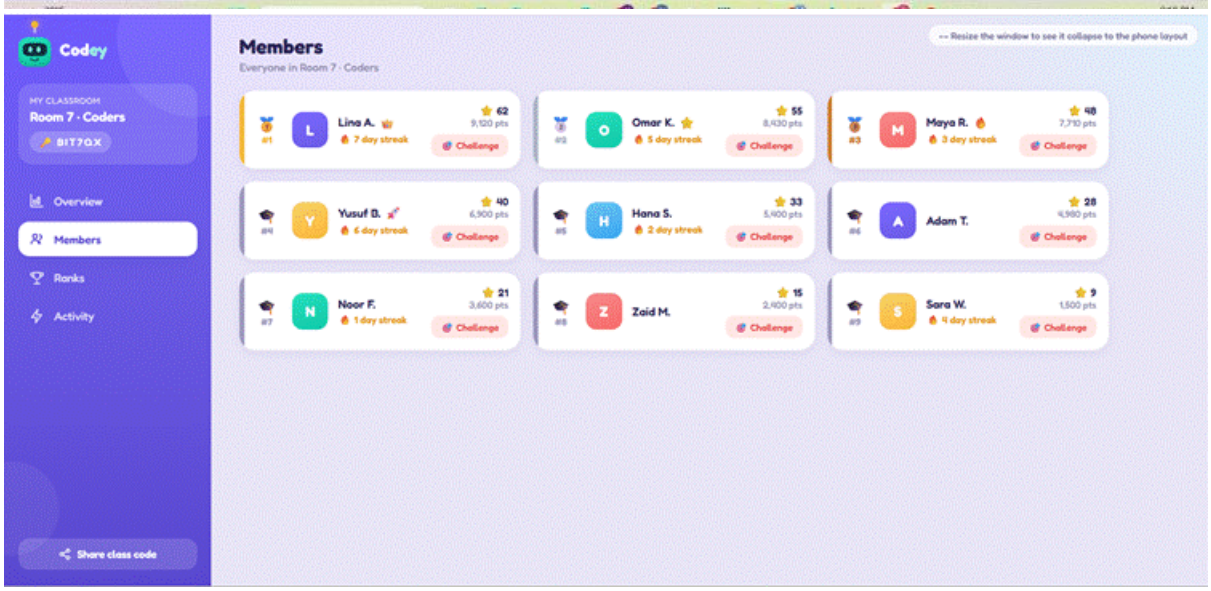


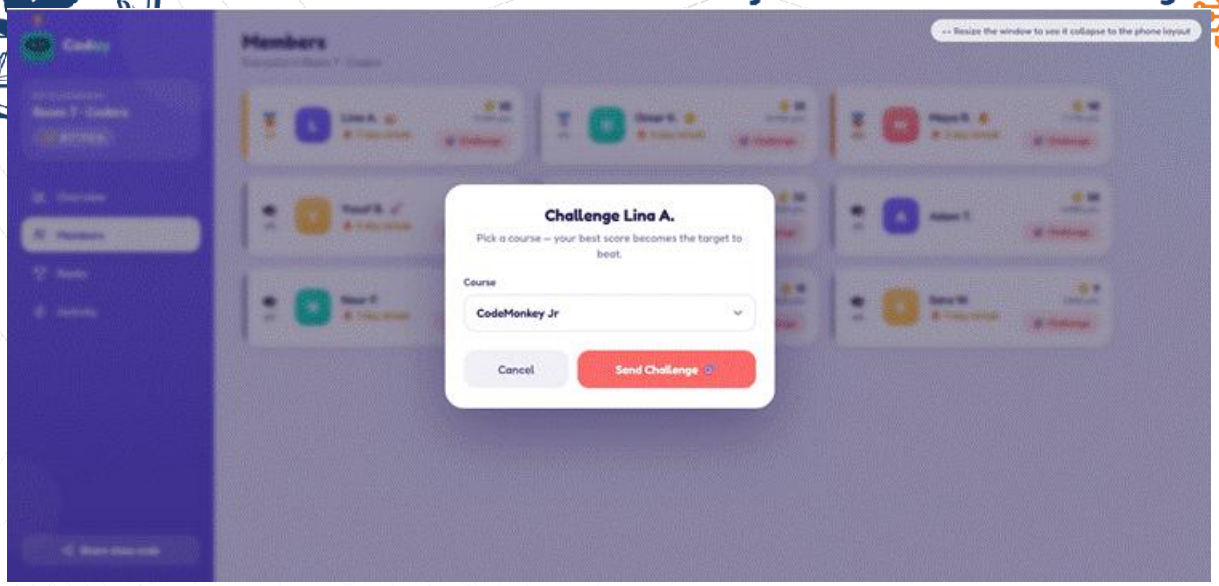
Figure 4.111: Activity

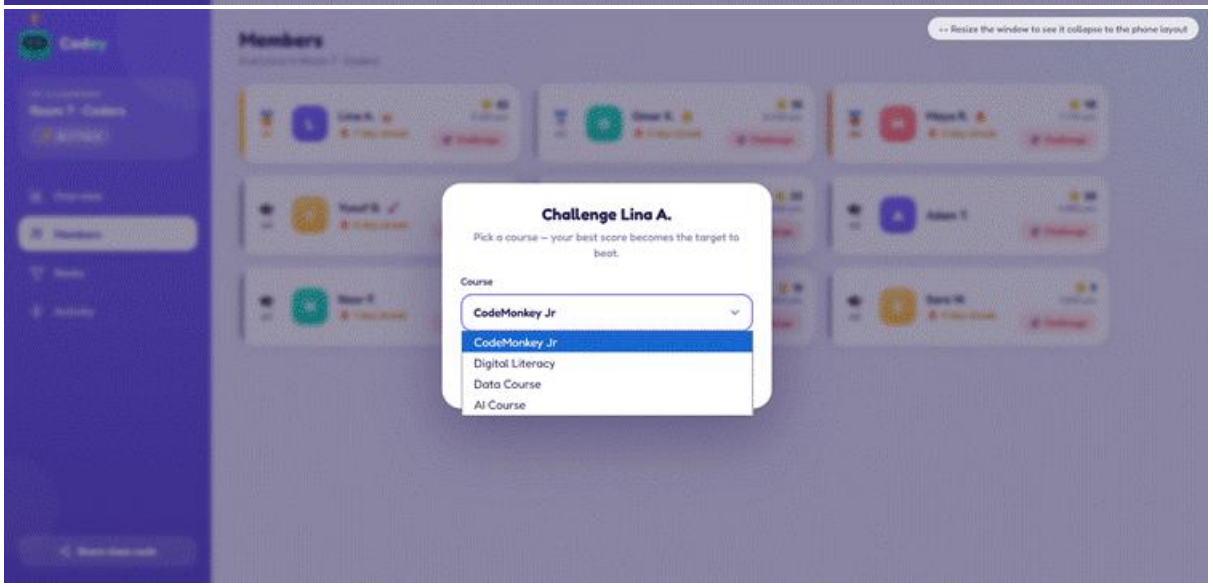


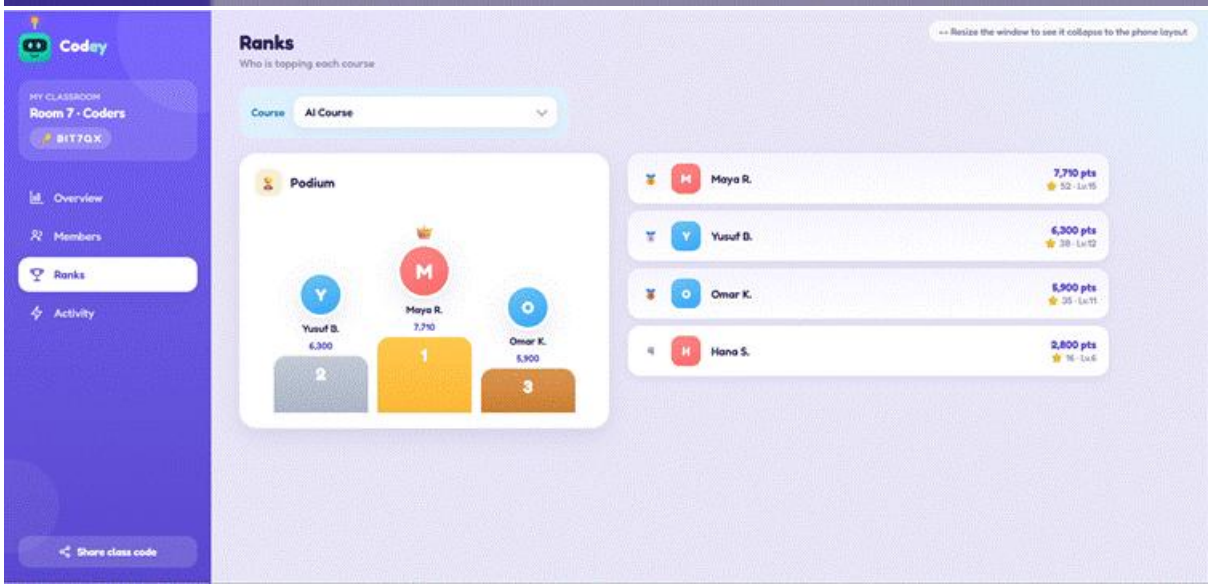




University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu







University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

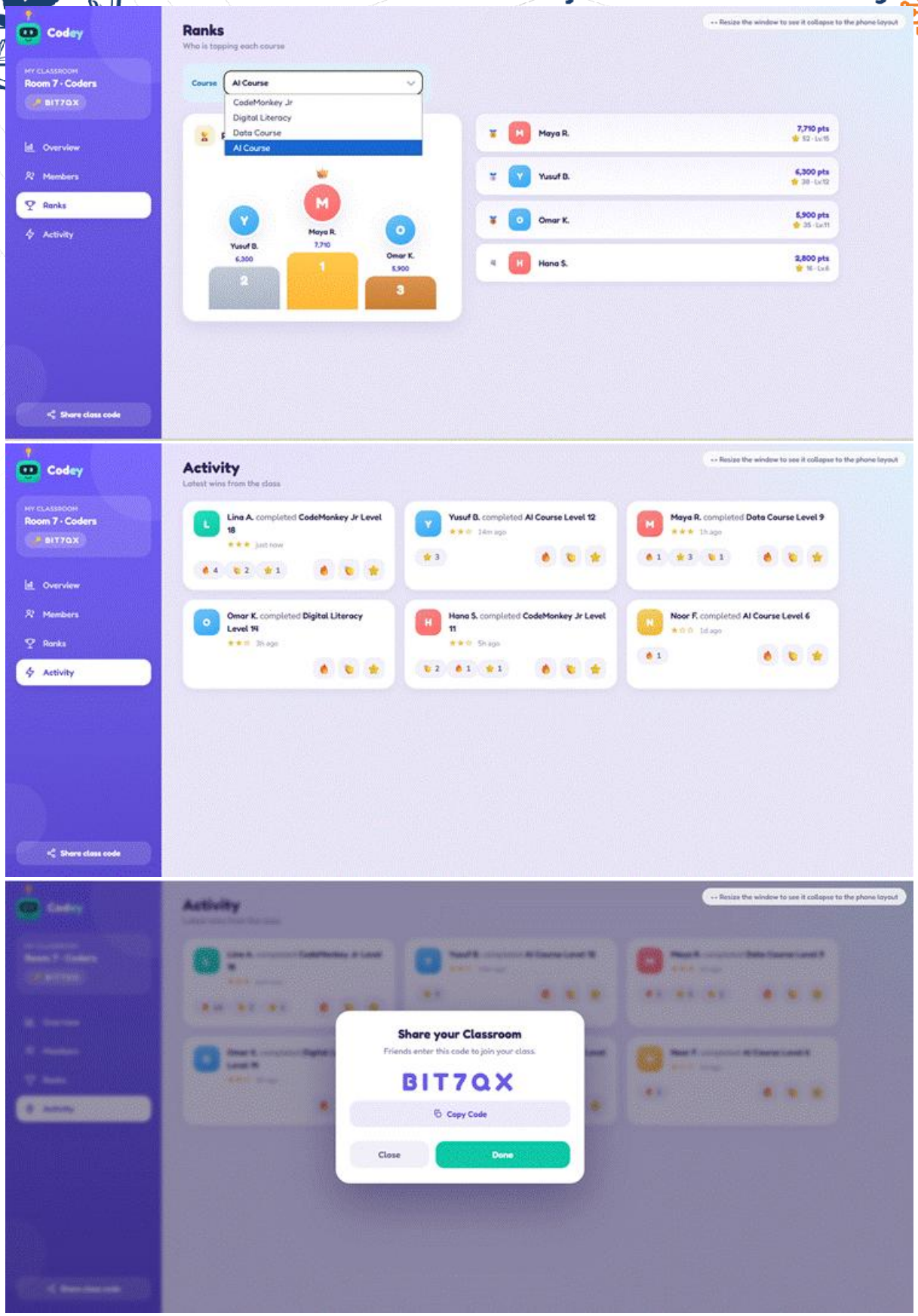


Figure 4.112: Activity - view 2

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.11 Dashboard Dropdown and My Profile Page

This section explains two account-related areas in the Codey student interface: the dashboard dropdown menu and the My Profile page. These features give the student quick access to account navigation, language settings, profile details, created projects, security actions, and logout controls. The design keeps account actions separate from learning content while still making them easy to reach from the dashboard.

4.11.1 Dashboard Dropdown Menu

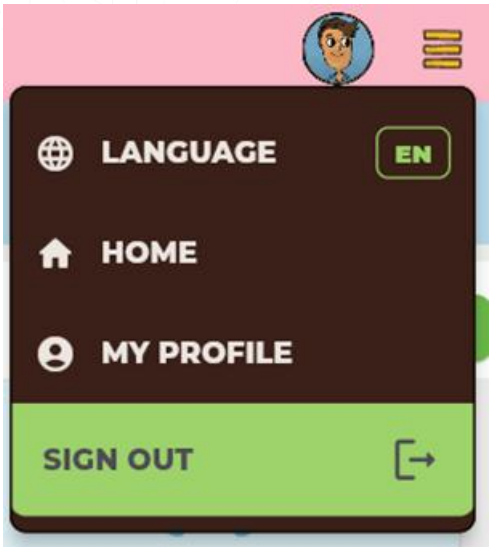


Figure 4.11.1: Dashboard dropdown menu with language, home, profile, and sign-out actions.

The dashboard dropdown appears from the top-right area of the student dashboard, near the avatar and menu icon. It works as a compact navigation panel for actions that are related to the current account rather than to a specific course or game. The menu uses large icons, clear labels, and a strong visual separation between normal navigation actions and the sign-out action.

The dropdown includes the following options:

- **Language:** shows the current application language using a small language badge, such as EN. Selecting this option opens the language selection dialog.
- **Home:** returns the student to the main dashboard or home screen.
- **My Profile:** opens the profile page where the student can view account information, project statistics, and quick actions.
- **Sign Out:** ends the current session and returns the user to the public entry or login flow. The sign-out area is highlighted separately to make it easy to identify.

This dropdown improves usability because students do not need to search through several pages to reach basic account controls. It also keeps the dashboard cleaner by moving secondary account actions into a small menu.

4.11.2 Language Selection Dialog

Figure 4.11.2: Language selection dialog showing English and Arabic options.

When the student chooses the Language option from the dropdown, the system displays a language selection dialog. The dialog contains two available language choices: English and Arabic. The selected language is highlighted with a colored border and a checkmark indicator, making the current language easy to recognize.

The language feature is important because Codey is designed as a bilingual platform. Switching the language affects labels and interface text across the application. Arabic support also requires right-to-left layout behavior in the relevant screens, so the language setting is not only a translation feature but also part of the interface direction and accessibility design.

The dialog is simple and child-friendly:

- Each language is displayed as a large selectable card.
- The language abbreviation, such as EN or AR, is shown beside the language name.
- A checkmark indicates the currently selected option.
- The close button allows the student to exit the dialog without changing the current selection.

4.11.3 My Profile Page Overview

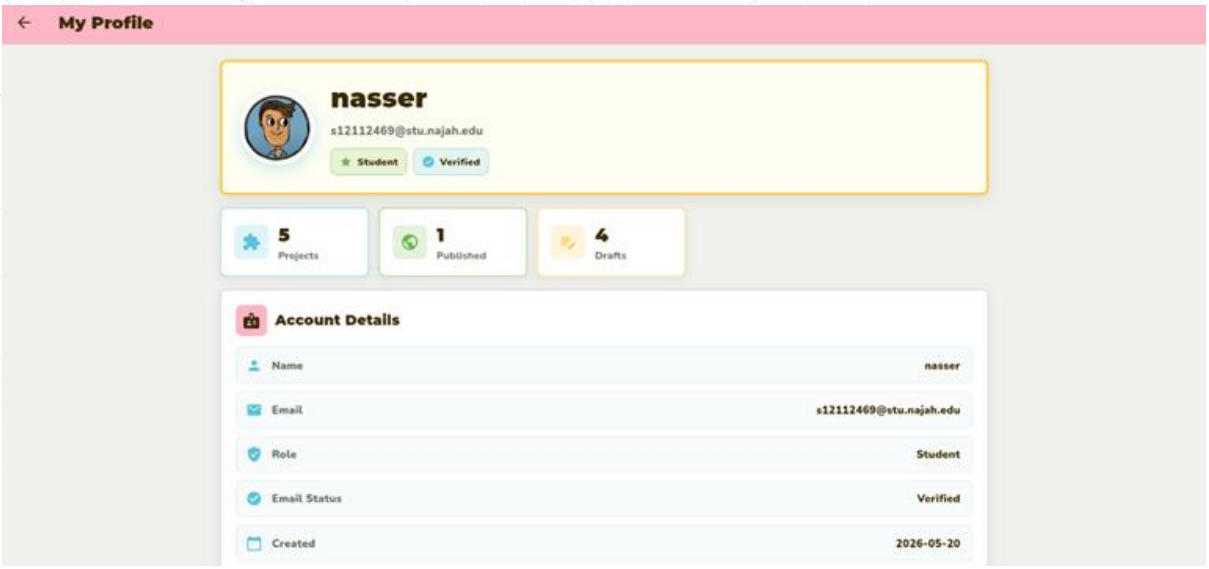


Figure 4.11.3: My Profile page header, user summary, project statistics, and account details.

The My Profile page is opened from the dashboard dropdown or from profile-related quick actions. It provides a central place for the student to review account information and manage personal settings. The page uses the same friendly visual style as the rest of the platform, with a pink top bar, a back button, rounded cards, colorful icons, and a clear separation between profile information, statistics, account details, and actions.

At the top of the page, the profile header displays the student avatar, display name, email address, role badge, and verification badge. In the shown example, the user is marked as Student and Verified. This helps the user confirm that they are logged in to the correct account.

Below the profile header, the page displays three statistic cards:

- **Projects:** the total number of saved builder projects connected to the user.
- **Published:** the number of projects that have been published and shared.
- **Drafts:** the number of saved projects that are still private or unfinished.

Technically, these counts are produced by loading the user profile and the user builder projects. The project list is counted, then filtered by project status to calculate published projects and draft projects. This gives the profile page a useful summary of the learner's creative activity.

4.11.4 Account Details Section

The Account Details panel organizes the main account fields into readable rows. Each row contains a small icon, a label, and the stored value. This makes the page easy to scan and avoids placing account information in long paragraphs.

The Account Details section includes:

- **Name:** the student display name used inside the platform.
- **Email:** the email address connected to the account.
- **Role:** the account role, such as Student.
- **Email Status:** whether the account email is verified or unverified.
- **Created:** the date when the account was created.
- **Last Updated:** the date when the account was last changed.
- **Last Login:** the date of the most recent login, when available.

This section is useful for both usability and security. The student can confirm their role and email status, while the system can show important account dates that help identify whether the account information is up to date.

4.11.5 Quick Actions Section

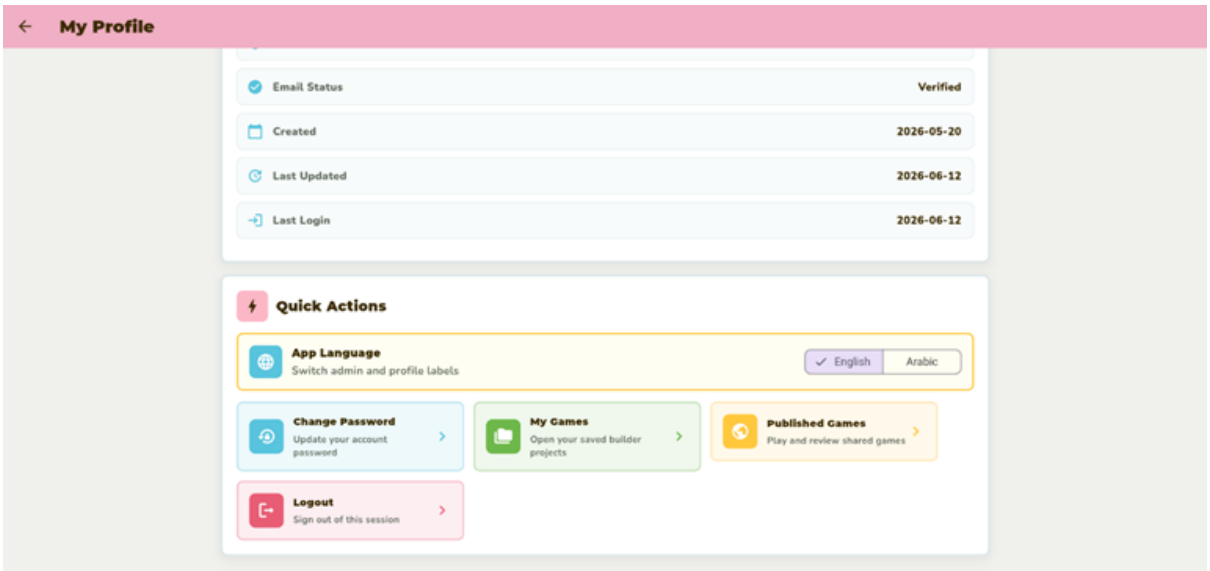


Figure 4.11.4: Quick Actions section with app language, password change, games, published games, and logout.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

The Quick Actions section appears below the account details. It contains the most important account actions in large clickable tiles. This design is suitable for children because each action has an icon, title, subtitle, and color. The student can understand the purpose of each option without reading a long settings list.

The Quick Actions section includes:

- **App Language:** allows the student to switch between English and Arabic directly from the profile page. The language selector appears as a segmented control, showing the currently selected language.
- **Change Password:** opens the password update dialog and allows the student to change their account password.
- **My Games:** opens the user's saved builder projects so they can continue editing or reviewing their creations.
- **Published Games:** opens the games that the user has shared publicly.
- **Logout:** signs the user out of the current session. This action is visually separated with a warning-style color because it ends the current account session.

The profile implementation also handles cases where the email is not verified. If the account is unverified, an additional action can appear to resend the verification email. This keeps verification-related actions available only when they are needed.

4.11.6 Change Password Dialog

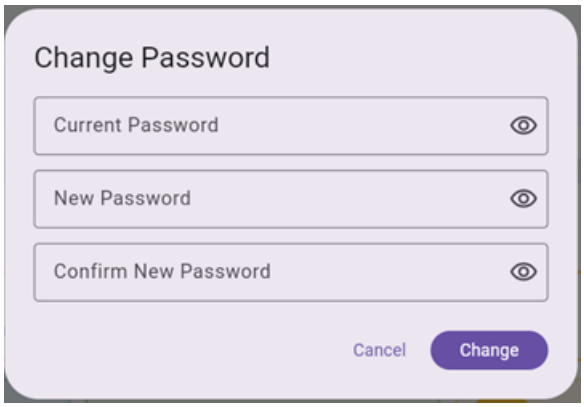


Figure 4.11.5: Change Password dialog with current password, new password, and confirmation fields.

The Change Password dialog is opened from the Quick Actions section. It contains three password fields: Current Password, New Password, and Confirm New Password. Each field includes an eye icon so the user can show or hide the entered password. This improves usability because the student can check the typed value when needed while still keeping the password hidden by default.

The dialog includes two actions:

- **Cancel:** closes the dialog without changing the password.
- **Change:** submits the password update request.

Before sending the request, the interface validates the entered information. For example, the new password and confirmation must match. After validation, the request is sent through the API using the authenticated session token. If the backend confirms that the password was changed successfully, the dialog closes and a success message is shown. If the request fails, the dialog displays an error message so the user can correct the problem.

This feature supports account security by allowing students to update their password from inside the profile page without needing to leave the platform.

4.11.7 Technical Implementation

The dashboard dropdown and profile page are connected to the same authenticated user session used throughout the student dashboard. The session contains the current user information and authentication token. This token is used when loading profile data, fetching builder projects, updating avatar or profile information, changing the password, and performing other protected actions.

The My Profile page loads two main sets of data:

- **Profile data:** loaded from the profile API and converted into the user model used by the Flutter client.
- **Builder project data:** loaded from the builder projects API and used to calculate the total number of projects, published projects, and drafts.

The page also supports loading and error states. While the profile data is being requested, the interface can show a loading state. If the request fails, an error message is stored and displayed instead of leaving the page empty. This improves reliability and makes problems easier to understand during use.

The profile header supports both asset avatars and uploaded profile photos. Uploaded photos can be stored with frame settings such as scale and position so that the image

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

appears correctly inside the circular avatar frame. This matches the personalization approach used in other parts of Codey.

The language controls are connected to the application language system. When the student switches between English and Arabic, the visible labels in the interface update through localized text keys. This keeps the profile page consistent with Codey's bilingual design.

4.12 My Creations and Course Creation System

The My Creations section is the main authoring workspace in Codey. It allows learners to move from using ready-made lessons into creating their own challenges, assets, and courses. The section brings together the creative tools of the platform in one place, so a student can create new content, reopen saved work, manage uploaded assets, and prepare content for publishing or verification.

This part of the system is important because it supports one of the main goals of Codey: allowing children to learn programming and digital skills by building. Instead of only consuming course material, users can design course content, build game levels, upload visual assets, and organize their work into reusable projects.

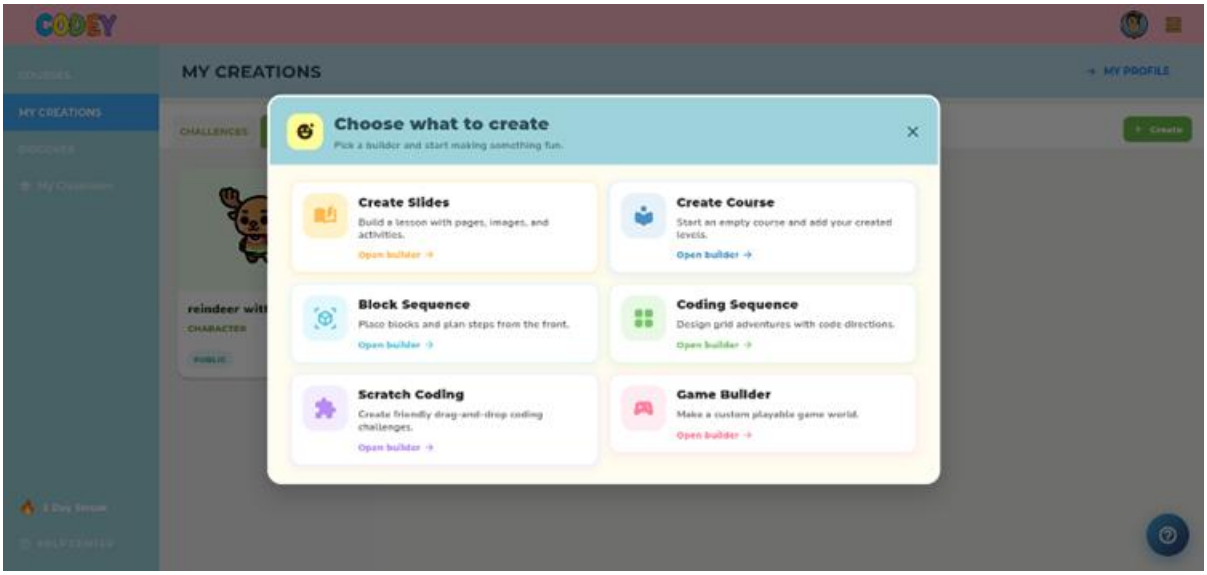


Figure 4.12.1: Create popup in the My Creations tab, showing the available creation types.

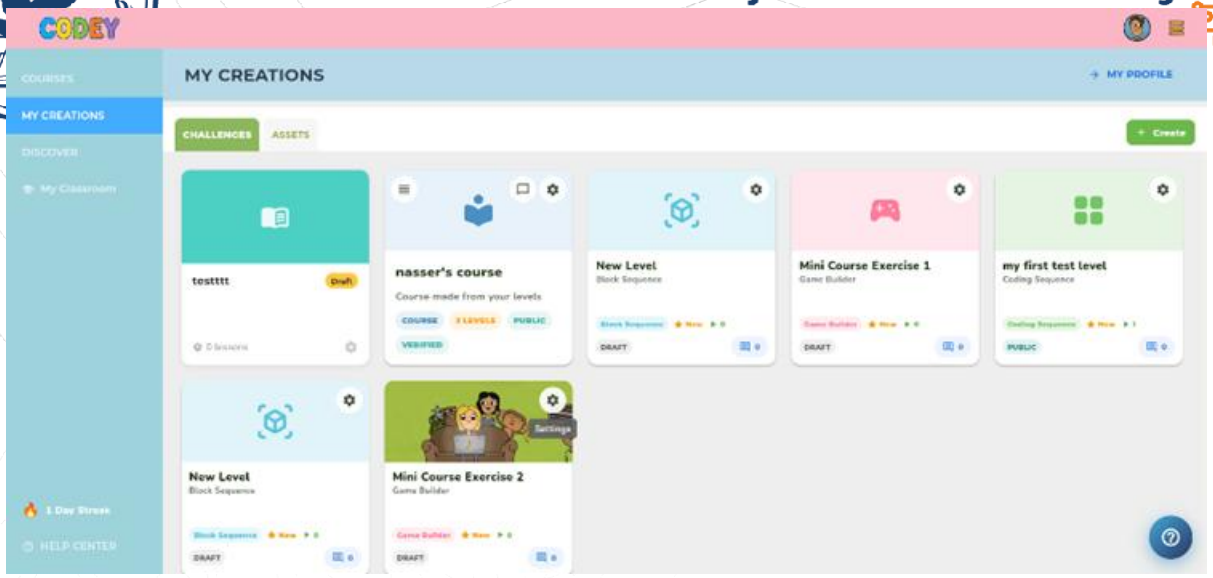


Figure 4.12.2: Challenges tab showing created projects as cards.

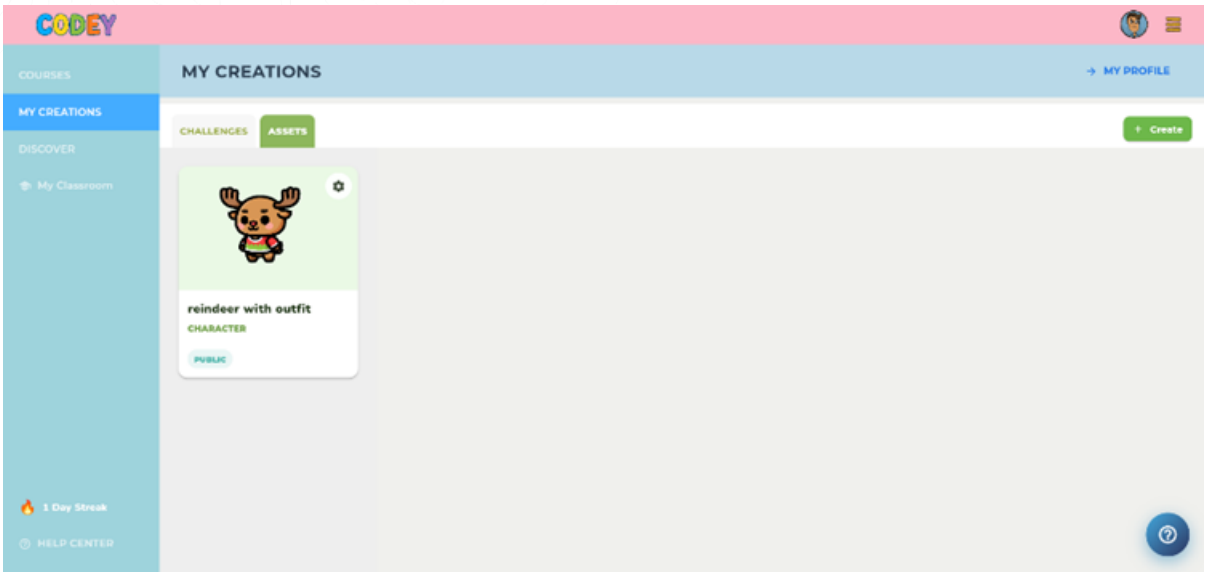


Figure 4.12.3: Assets tab showing a created or collected asset.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.12.1 My Creations Layout

The My Creations page uses a simple dashboard layout. The left sidebar remains consistent with the student area, while the main content area is dedicated to the learner's created content. The page is divided into tabs to separate different types of content and avoid mixing game projects with reusable assets.

- **Challenges tab:** displays the games, challenges, and builder projects created by the user. Each item appears as a card with a visual preview and basic metadata.
- **Assets tab:** contains custom or collected assets that can later be reused inside the builders. This allows the user to build a small personal asset library.
- **Create button:** opens the creation selection popup, where the user chooses the type of project to start.
- **Project cards:** represent saved work and allow the user to reopen or manage existing creations.

The card-based design makes the page easy for children to use because each creation has a visual identity. It also supports long-term project work: a student can start a project, save it, leave the platform, and continue editing it later.

4.12.2 Create Button and Builder Routing

The Create button is the entry point for starting new content. When it is pressed, the platform displays a dialog that asks the user to choose the type of creator to open. The GitHub implementation separates the builder choices using the `_GameCreatorOption` enum, which includes scratch, frontView, topView, and fourthDemo options. This keeps the creation process organized and makes it possible to route each choice to its own builder page.

- **Scratch Builder:** opens the block-programming style creator.
- **Front View:** opens the side-view sequence/platform builder.
- **Top View:** opens the overhead code-sequence builder.
- **Fourth Demo/Game Builder:** opens the advanced custom-code game builder.

Technically, when an existing project card is opened, the frontend checks the project type. If the project is marked as top view, scratch, or fourth demo, the system routes to the corresponding builder route. Otherwise, it opens the front-view builder. This allows all builder projects to be shown in one My Creations page while still preserving separate editing environments for each project type.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Relevant GitHub structure: client/lib/features/builder/pages/my_games_page.dart
Project loading: ApiService.getAllBuilderProjects(authToken)
Routing examples: AppRoutes.topViewBuilder, AppRoutes.scratchBuilder, AppRoutes.fourthDemoBuilder, AppRoutes.builder

4.12.3 Project Loading, Status, and Deletion

The My Creations page loads builder projects from the backend through the API service. The page keeps loading and error states so the interface can show a loading indicator, an error message, or the project grid depending on the response. This prevents the screen from appearing empty without explanation when the network request is still running or fails.

- **Loading state:** shown while the projects are being fetched from the backend.
- **Error state:** shown when the request fails, with an option to try again.
- **Empty state:** shown when the user has no saved projects yet.
- **Refresh behavior:** the project list can be refreshed after returning from a builder or after deleting a project.
- **Deletion confirmation:** before deleting a project, the user is asked to confirm the action, especially if the project is already published.

This workflow protects the user from accidental data loss and keeps the displayed creations synchronized with the backend after edits or deletions.

4.12.4 Course Creation Screen

The course creation screen allows the user to create a structured educational course rather than a single game challenge. The left panel contains the main course information, while the rest of the screen is used for adding lessons. This separation makes the course-building workflow easier to understand: first the user defines the course, then they add lessons, and finally they create slide content inside each lesson.

- **Course cover image:** a visual image that represents the course in cards and previews.
- **Course title:** the main name of the course.
- **Description:** a short explanation of the course topic and purpose.
- **Progress indicator:** shows how many lessons have been completed or added.
- **Save Course button:** stores the course information so it can be reopened and edited later.

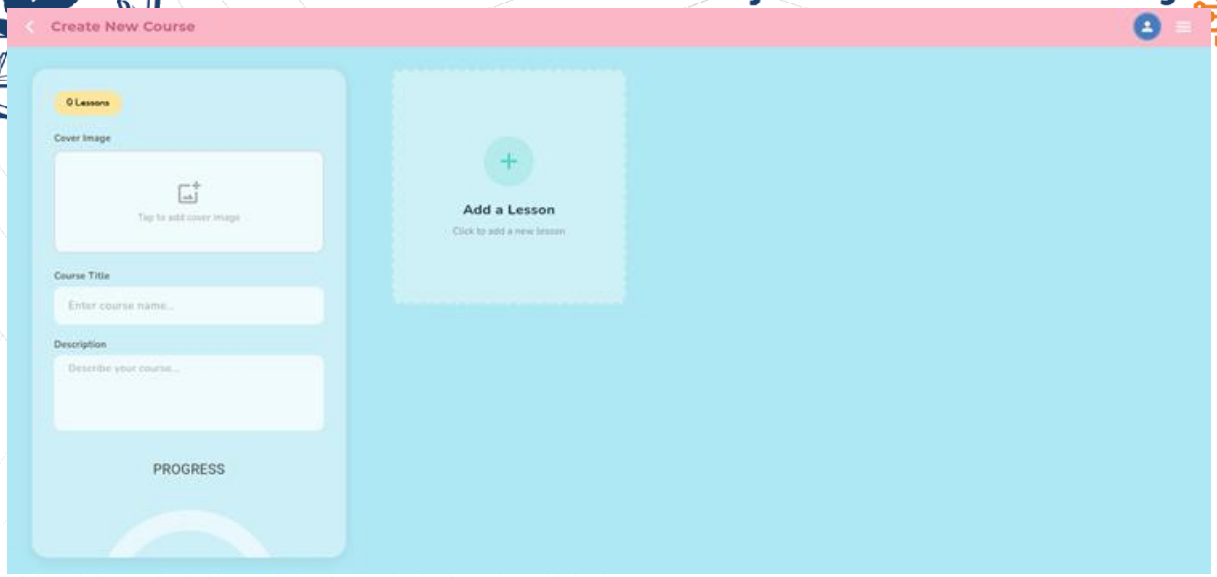


Figure 4.12.4: Initial course creation screen with course details and an Add a Lesson card.

4.12.5 Course Cover Upload and Frame Adjustment

The cover upload feature allows the creator to choose an image from the device and use it as the visual identity of the course. After uploading, the image is displayed in the course cover area. The system also supports image replacement, removal, and frame adjustment so the creator can control how the image fits inside the visible cover frame.

- **Upload:** opens the device file picker and stores the selected image.
- **Preview:** displays the selected cover inside the course form.
- **Replace:** allows the creator to choose a different image.
- **Remove:** clears the selected cover image.
- **Zoom and positioning:** allow the image to be fitted properly inside the cover frame.

On the backend side, the course model supports storing the image as Base64 together with frame settings such as scale and offset. This means that the same visual crop can be restored later when the course is loaded again.

Course model fields include: courseImageBase64, coverFrameScale, coverFrameOffsetX, coverFrameOffsetY

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

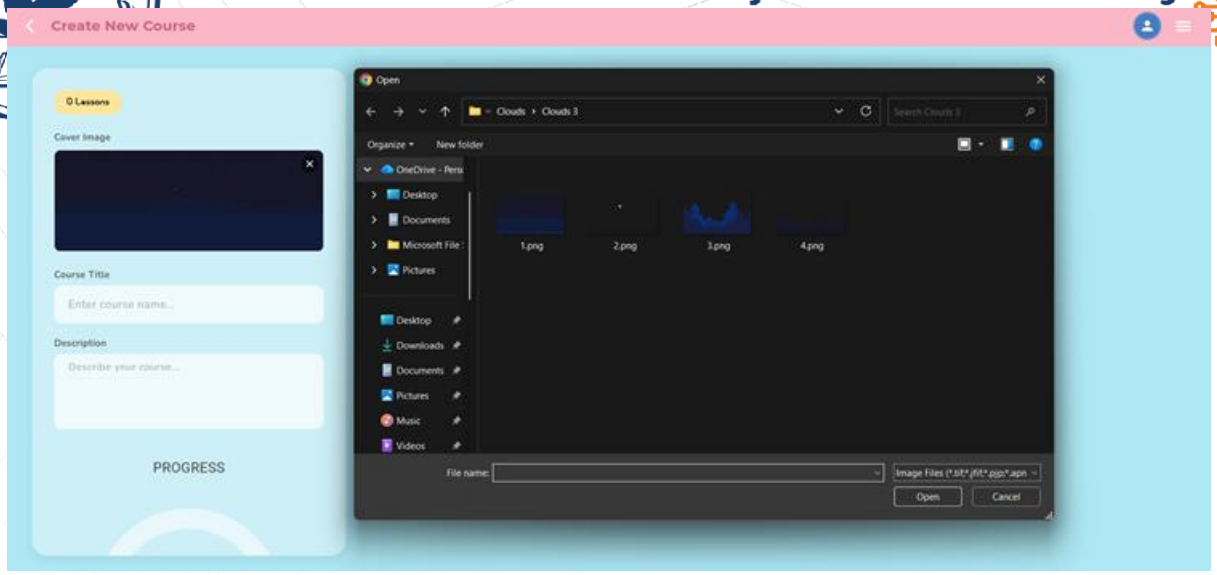


Figure 4.12.5: Selecting a course cover image from the device.

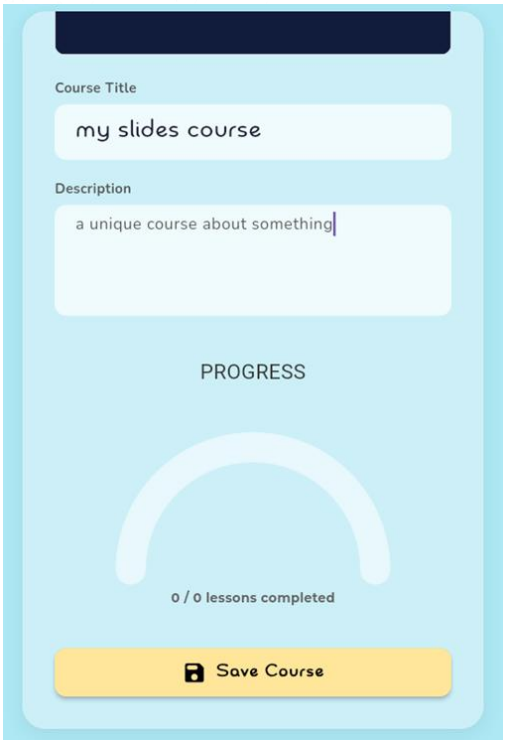


Figure 4.12.6: Course information panel after entering the title, description, and cover image.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.12.6 Adding Lessons

After creating the basic course information, the user can add lessons. A lesson acts as a smaller learning unit inside the course. This makes the course easier to organize because each lesson can focus on a specific topic, explanation, or exercise.

- **Add a Lesson card:** shown with a large plus icon to start a new lesson.
- **Lesson number:** identifies the order of the lesson inside the course.
- **Lesson title:** allows the creator to name the lesson.
- **Lesson image:** can be added to give the lesson a visual preview.
- **Create Slides button:** opens the slide editor for that lesson.

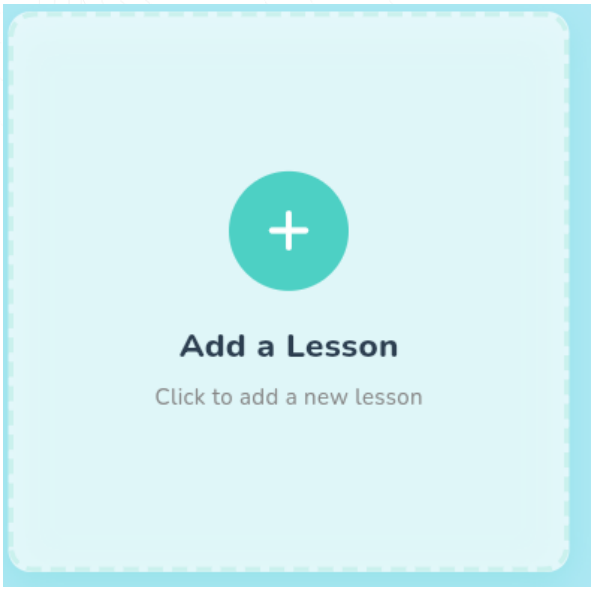


Figure 4.12.7: Add a Lesson card used to create a new lesson.

Figure 4.12.8: Lesson card after naming the lesson and adding a photo placeholder.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.12.7 Slide Template Selection

When the creator chooses to create slides, the system first shows a template selection screen. Templates provide ready-made slide layouts so that the user does not need to start every slide from a completely empty canvas. This improves usability and helps keep lesson slides visually consistent.

The template screen contains different layout types, including blank slides, image-and-text layouts, split layouts, dialogue-style slides, and multi-section layouts. The creator selects the template that best matches the lesson content and then edits it in the slide editor.

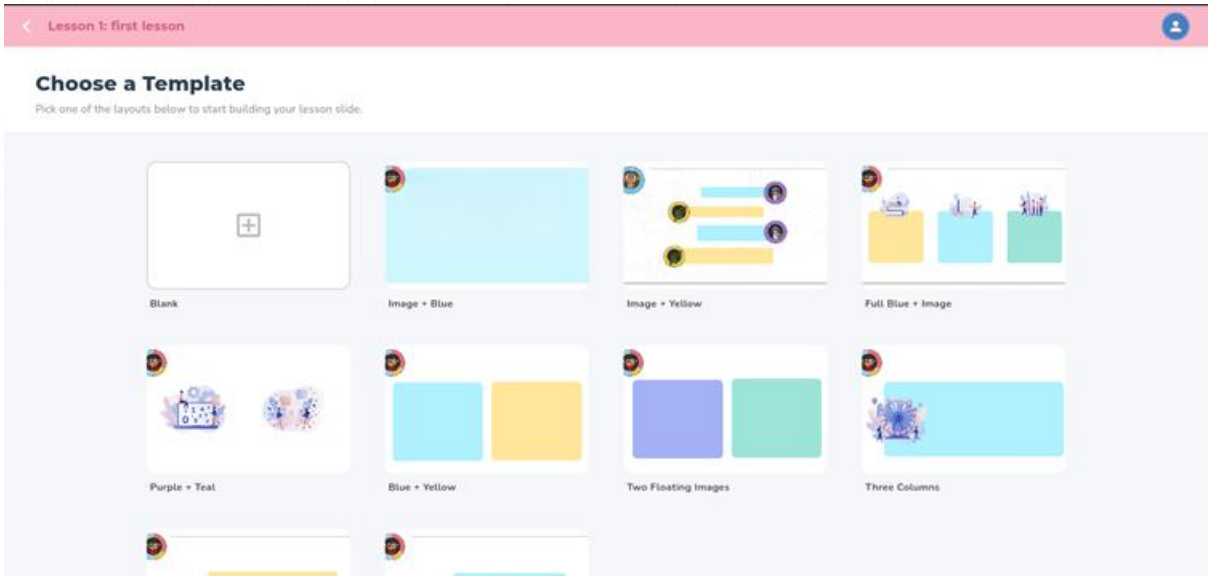


Figure 4.12.9: Template selection screen used before entering the slide editor.

4.12.8 Slide Editor Tools

The slide editor is the main workspace for building lesson content. It contains the slide canvas, slide thumbnails, editing toolbar, and save controls. The creator can design the slide visually by adding text, images, colors, shapes, and notes. This turns the course builder into a flexible authoring tool rather than a simple form-based course creator.

- **Text tool:** allows the creator to add titles, explanations, labels, and instructions.
- **Image tool:** opens the file picker so images can be inserted into the slide.
- **Color tool:** changes background or object colors to improve visual clarity and style.
- **Shape tool:** adds visual elements such as rectangles, circles, and other shapes.
- **Notes tool:** stores speaker notes or additional explanations related to the slide.
- **Slide thumbnails:** allow the creator to move between slides and manage the lesson sequence.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

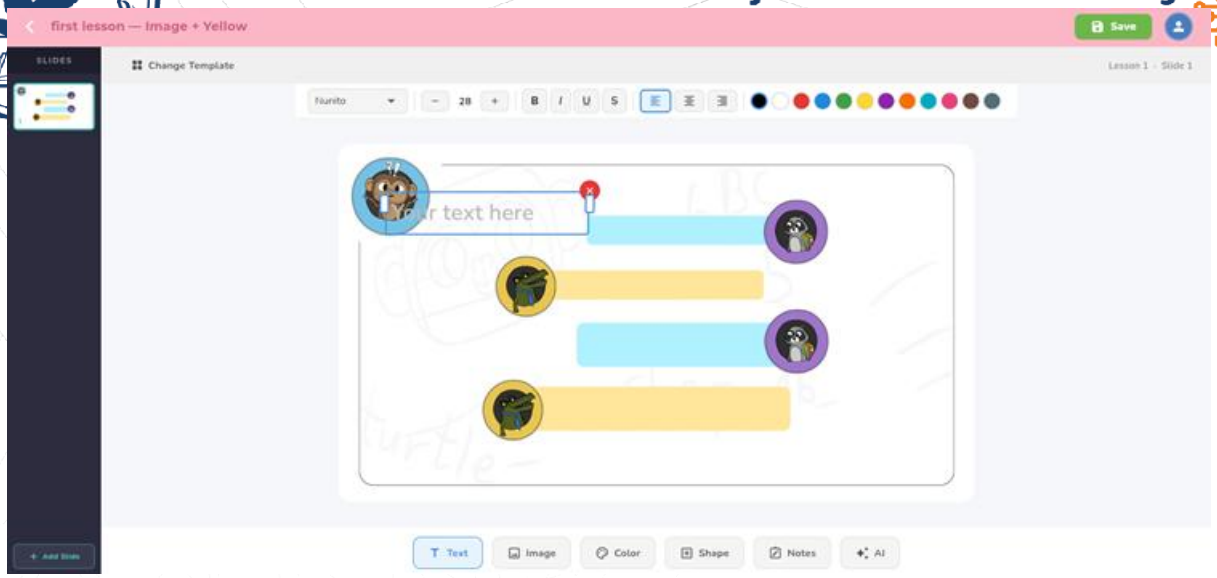


Figure 4.12.10: Slide editor with text and visual layout editing.

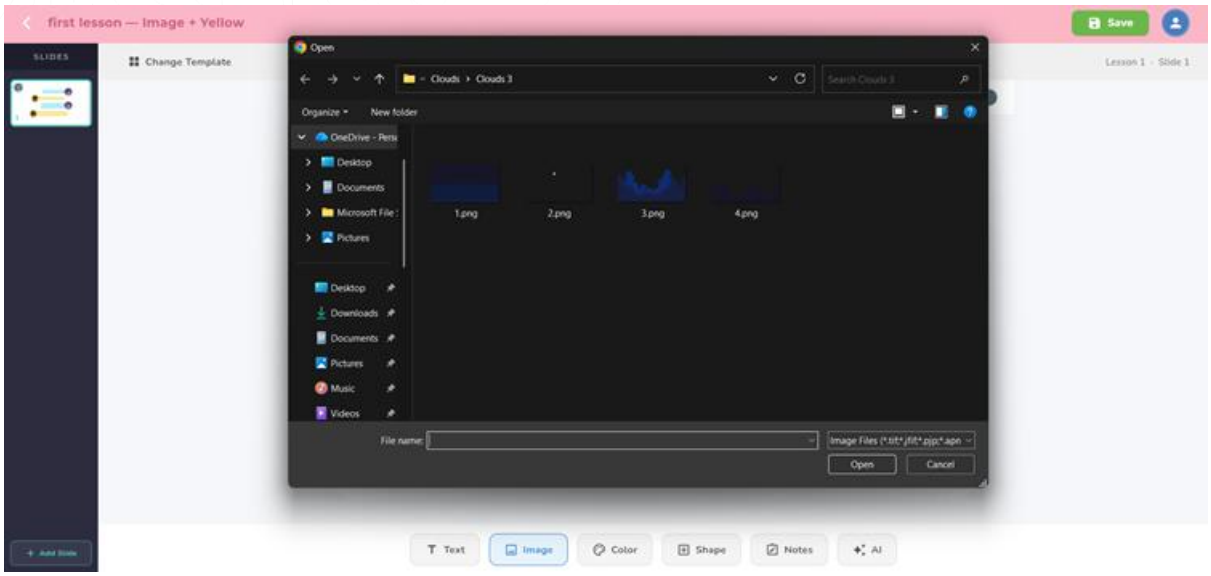


Figure 4.12.11: Image insertion in the slide editor.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

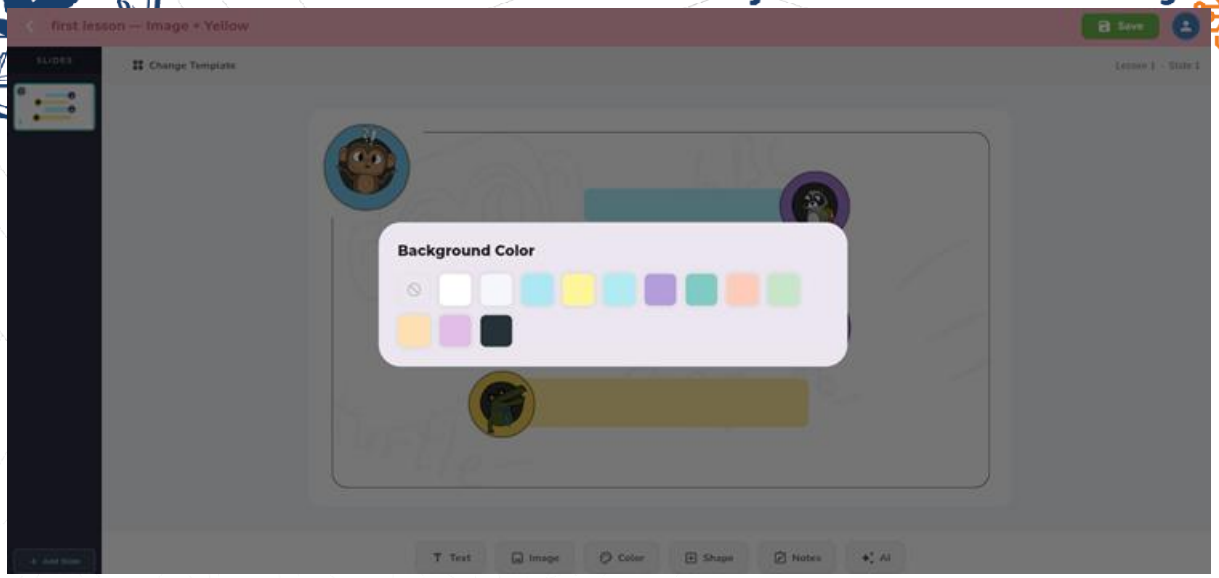


Figure 4.12.12: Background color selection in the slide editor.

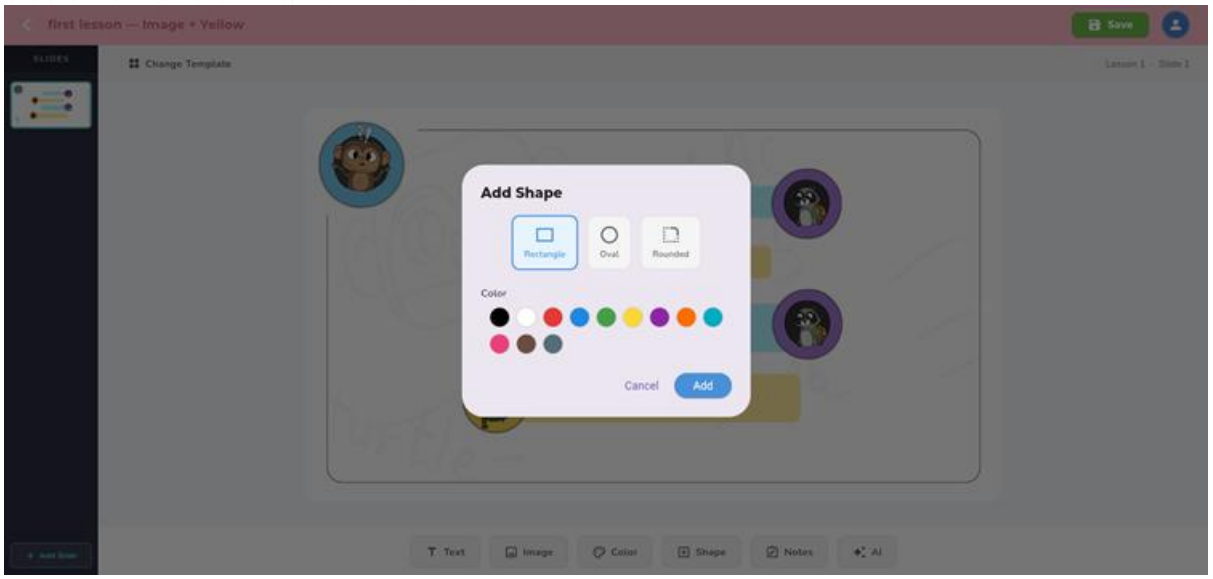


Figure 4.12.13: Shape insertion tool with style and color options.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

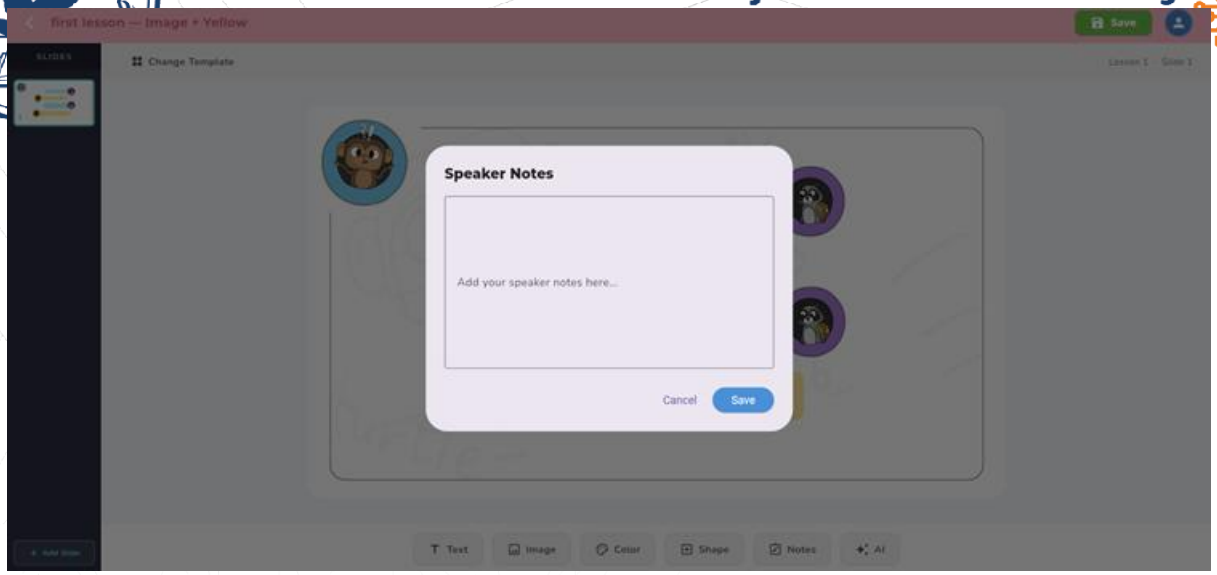


Figure 4.12.14: Speaker notes dialog used to add extra explanation to a slide.

4.12.9 AI Chatbot Assistance

The slide editor includes an AI assistant panel that helps the creator generate or improve educational content. This feature is useful when the creator needs help writing explanations, creating examples, or expanding a short idea into a complete lesson section.

- **Content generation:** the AI can generate slide text based on the requested topic.
- **Content improvement:** the AI can help rewrite or clarify educational text.
- **Lesson support:** the assistant can suggest examples, explanations, or step-by-step content.
- **Human review:** the creator remains responsible for checking the generated content before adding it to the lesson.

The Add to Slide button allows generated content to be inserted into the slide. This makes the workflow faster, but the generated content is not automatically published. The creator still controls what is inserted, edited, saved, and later made visible to learners.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

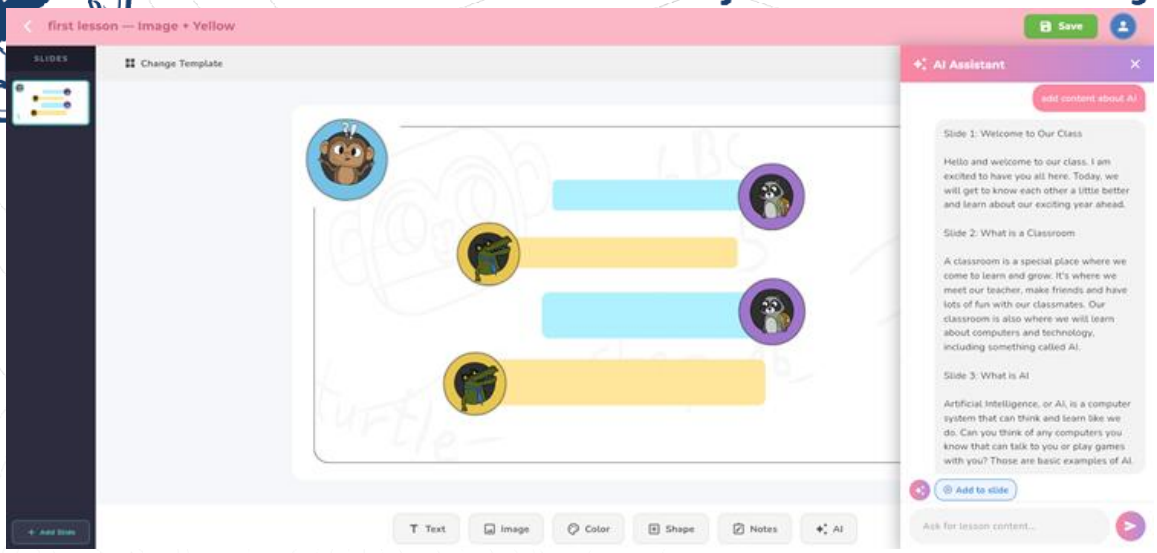


Figure 4.12.15: AI chatbot panel inside the slide editor.

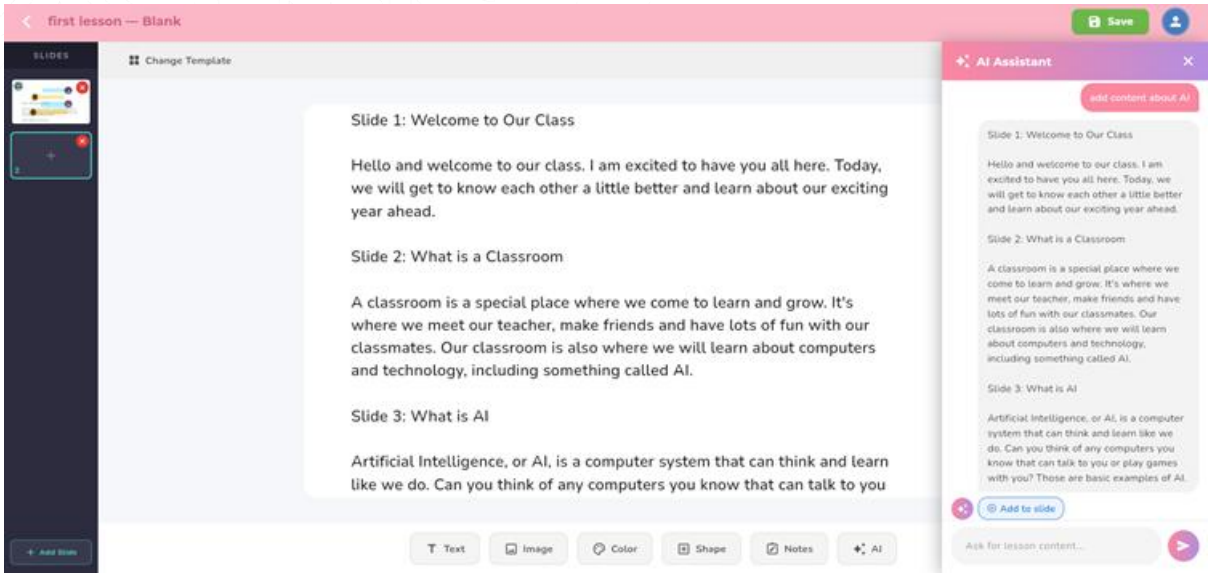


Figure 4.12.16: AI-generated lesson text with the Add to Slide option.

4.12.10 Quick Course Creation from My Creations

A course can also be created directly from the Create button in the My Creations page. In this workflow, a Create Course dialog appears over the page. The dialog collects the course name, category, description, and cover image. After the user creates the course, the platform opens the course-level management screen so the creator can begin adding lessons or levels.

- **Course name:** defines the title used throughout the platform.
- **Category:** classifies the course, such as Coding.
- **Description:** summarizes the course content.
- **Upload cover:** adds or replaces the course image.
- **Create button:** submits the course data to the backend.

The API service connects this workflow to the backend through authenticated course endpoints. The frontend sends course data to `/api/courses/mine`, and the backend stores it as a user-owned course.

API methods: createMyLevelCourse(), updateMyLevelCourse(), deleteMyLevelCourse(), requestMyLevelCourseVerification()

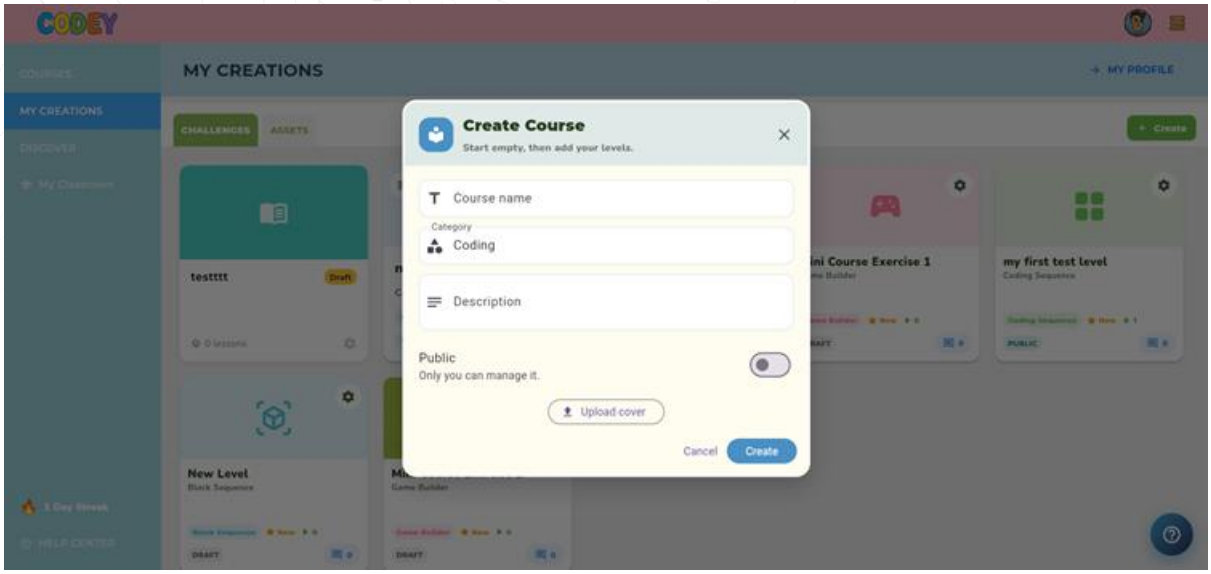


Figure 4.12.17: Create Course dialog opened from the My Creations page.

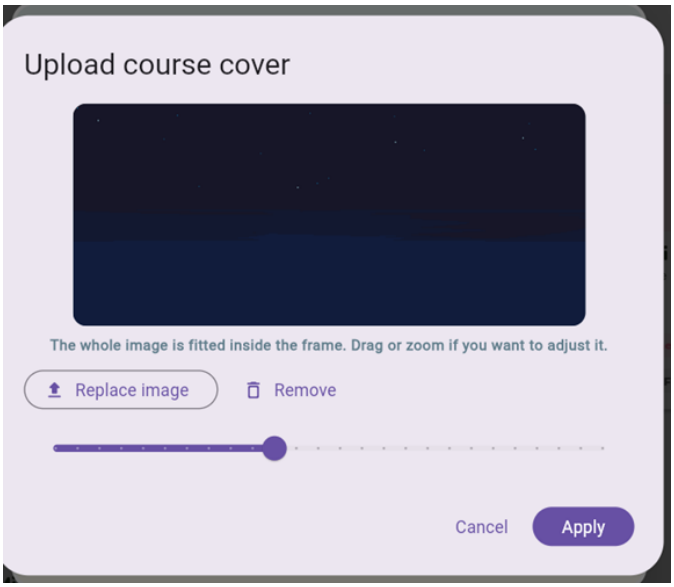


Figure 4.12.18: Course cover upload and adjustment window.

4.12.11 Course Levels Management

After a course is created, the user can manage its connected levels. The levels page displays the current course at the top and provides an Add button for creating or connecting a new level. When the course does not contain any levels, the page displays an empty-state message to clearly inform the user that no levels have been added yet.

- **Course selector:** allows the creator to choose the course or level group being managed.
- **Empty-state message:** appears when the course has no levels.
- **Add button:** allows the creator to create a new level or connect a level to the course.
- **Level dropdown:** shows existing exercises such as Mini Course Exercise 1 and Mini Course Exercise 2.

This structure keeps course metadata separate from course content. The creator first defines the course, then gradually adds the levels or exercises that learners will complete.

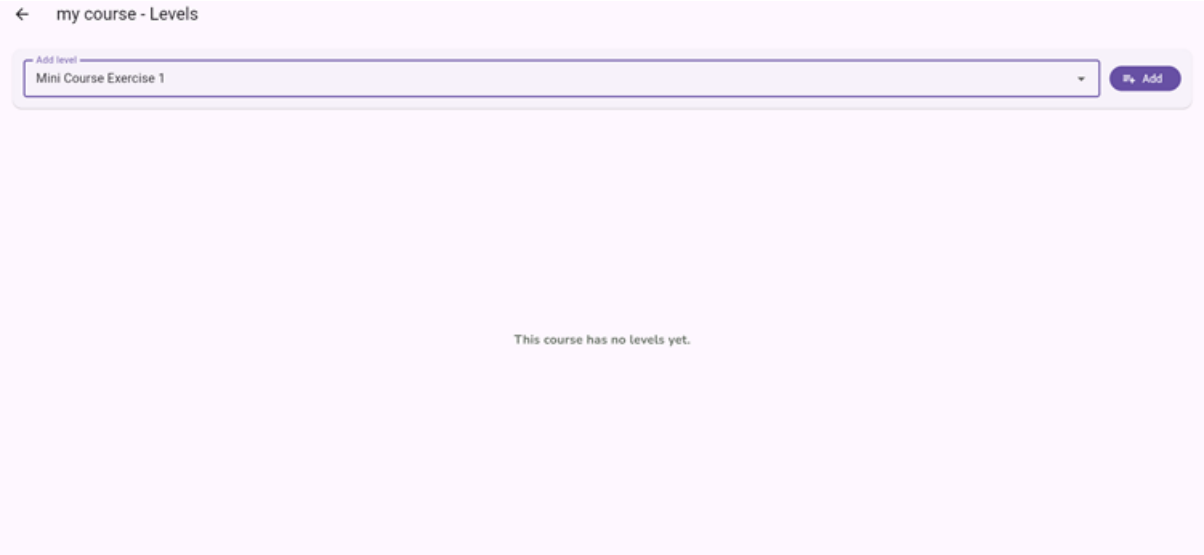


Figure 4.12.19: Course levels screen showing an empty course state.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

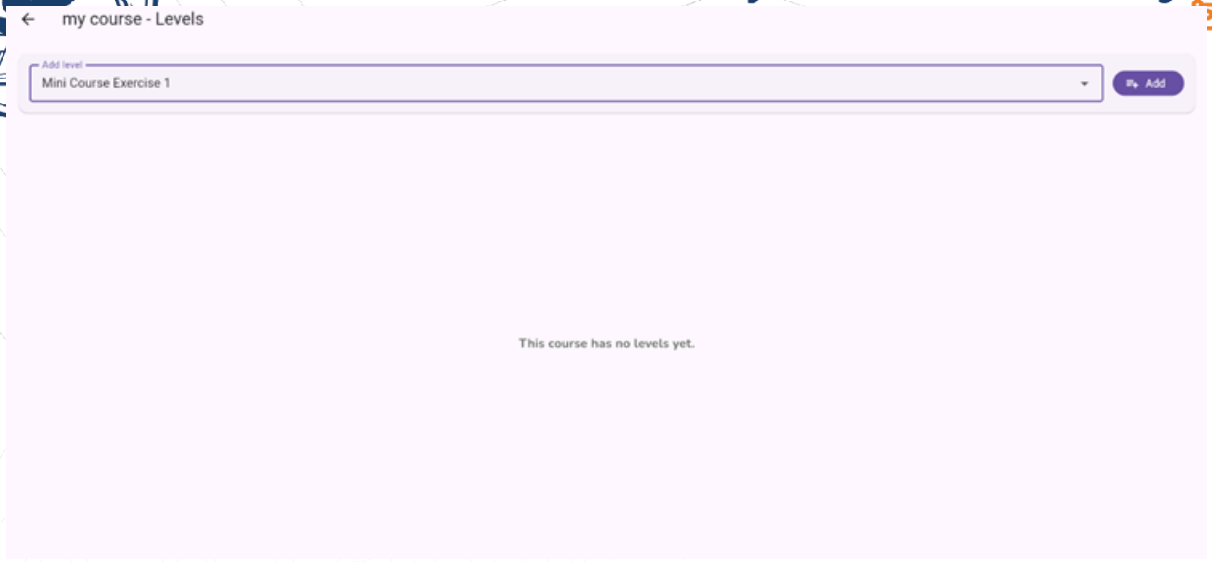


Figure 4.12.20: Course levels screen before adding levels.

Figure 4.12.21: Course level dropdown showing existing exercises and a new level option.

4.12.12 Course Settings and Verification Request

The Course Settings window allows the creator to update the course after it has been created. It includes fields for the course name, category, description, public visibility, cover image, level management, deletion, saving, and verification request. This window is important because it centralizes course administration actions for the course owner.

- **Course name:** updates the course title.
- **Category:** changes the course classification.
- **Description:** updates the course explanation.
- **Public toggle:** marks whether the course should be visible to learners.
- **Upload cover:** updates the course image.
- **Manage levels:** opens the level management screen.
- **Request verification:** sends the course to admin review.
- **Delete:** removes the course from the creator's content.
- **Save:** stores settings changes.

The verification workflow protects the quality and safety of public content. When a course is not verified, the settings screen shows a message explaining that verification can be requested. After the user submits the request, the status changes to pending admin review. The course can then be reviewed by an administrator before being treated as verified public content.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Technically, the course router supports a dedicated endpoint for this workflow: `POST /api/courses/mine/:id/verification-request`. The course model also stores verification fields such as `verificationStatus`, `verificationRequestedAt`, `verificationReviewedAt`, `verifiedAt`, `verifiedBy`, and `verificationRejectedReason`. These fields allow the backend to track whether the course is unverified, pending, approved, or rejected.

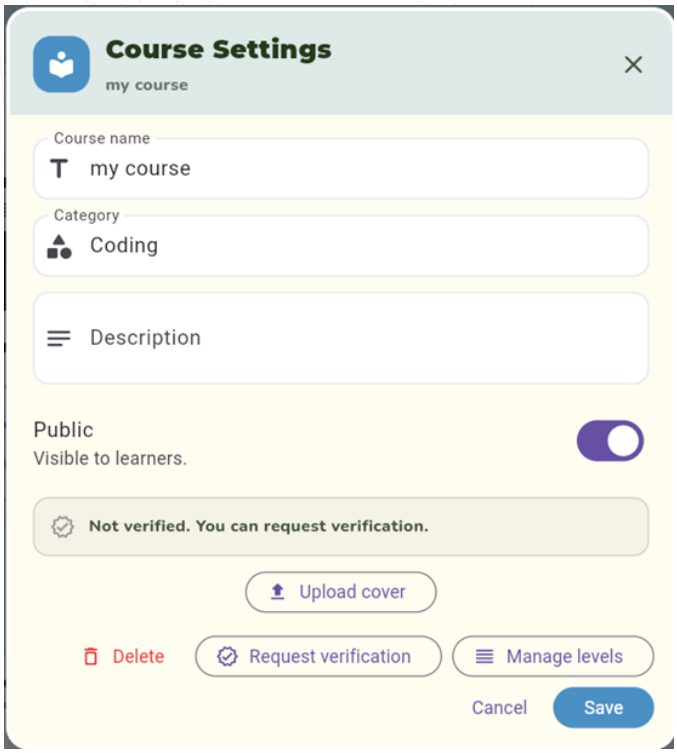


Figure 4.12.22: Course Settings window before requesting verification.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

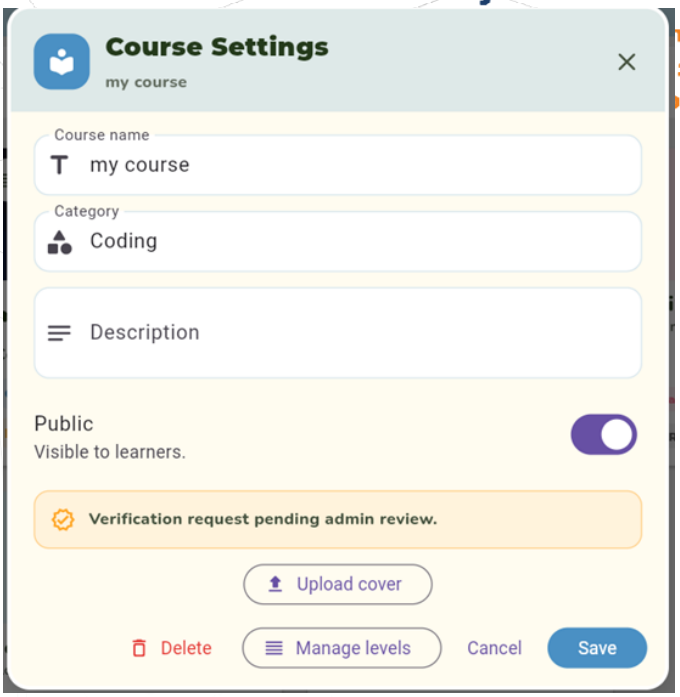


Figure 4.12.23: Course Settings window after the verification request becomes pending.

4.12.13 Backend Data and API Support

The My Creations and course creation system is supported by a full client-server workflow. The Flutter frontend handles the visual editing experience, while the Node.js backend stores courses, projects, assets, visibility settings, verification state, and ownership data.

- **Builder projects:** created, updated, loaded, deleted, published, played, commented on, and rated through `/api/builder/projects` endpoints.
- **Builder assets:** uploaded and retrieved through `/api/builder/assets` endpoints. Assets include metadata such as name, type, MIME type, image data, and public visibility.
- **User-created courses:** managed through `/api/courses/mine` endpoints for create, read, update, and delete operations.
- **Course verification:** handled through a separate verification request endpoint so that admin review is part of the publishing workflow.
- **Localization:** API requests can include the current language code so course content can be returned according to the selected language.

This technical separation keeps the system modular. The frontend focuses on editing, previewing, and user interaction, while the backend handles persistence, ownership, validation, publishing state, and verification status.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Backend endpoints used by this section:
GET/POST/PUT/DELETE /api/courses/mine
POST /api/courses/mine/:id/verification-request
GET/POST/PUT/DELETE /api/builder/projects
GET/POST/PUT/DELETE /api/builder/assets

4.13 Discover Page and Community Interaction

The Discover page is the community-facing area of Codey. It allows students to browse published challenges and shared assets created by other users, open playable content, view engagement information, and interact through ratings, plays, favorites, and comments. This section supports the creative side of the platform because students are not limited to their own projects; they can also explore what classmates and other learners have created.

The page works as a bridge between the My Creations system and the learner experience. Projects that are published by users can appear in Discover, where other students can play them and provide feedback. This creates a simple community loop: users create content, publish it, others discover it, and the original creator receives engagement through plays, ratings, comments, and favorites.

4.13.1 Discover Page Layout

The Discover page uses the same student dashboard structure as the rest of the learner interface. The left sidebar keeps the main navigation visible, including Courses, My Creations, Discover, and My Classroom. This allows the student to move between learning, creating, and exploring without leaving the main dashboard environment.

The top area includes a visual banner and tab buttons. The main content area below the banner changes depending on the selected tab. The screenshots show tabs for Challenges, Assets, and Favorites. These tabs separate playable user-created content from reusable visual assets and saved favorite items.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

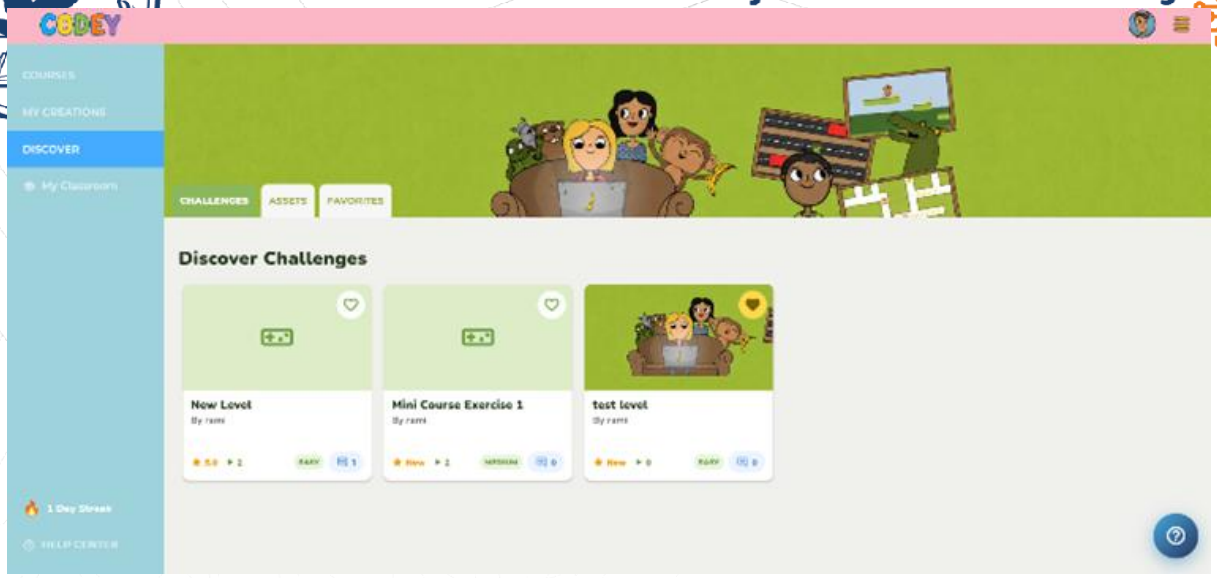


Figure 4.13.1: Discover page showing the main navigation, favourites, and assets.

4.13.2 Discover Tabs

The Discover page organizes community content into clear tabs so that students can quickly find the type of item they want to explore.

- The Challenges tab displays published game challenges that students can open and play.
- The Assets tab displays shared visual assets such as characters and backgrounds that can be reused in builders.
- The Favorites tab displays the items that the student marked using the heart icon, making it easier to return to useful or interesting content later.

This tab-based design avoids mixing different content types in one long list. It also makes the page easier for children because every tab has a clear purpose and the content is presented as visual cards.

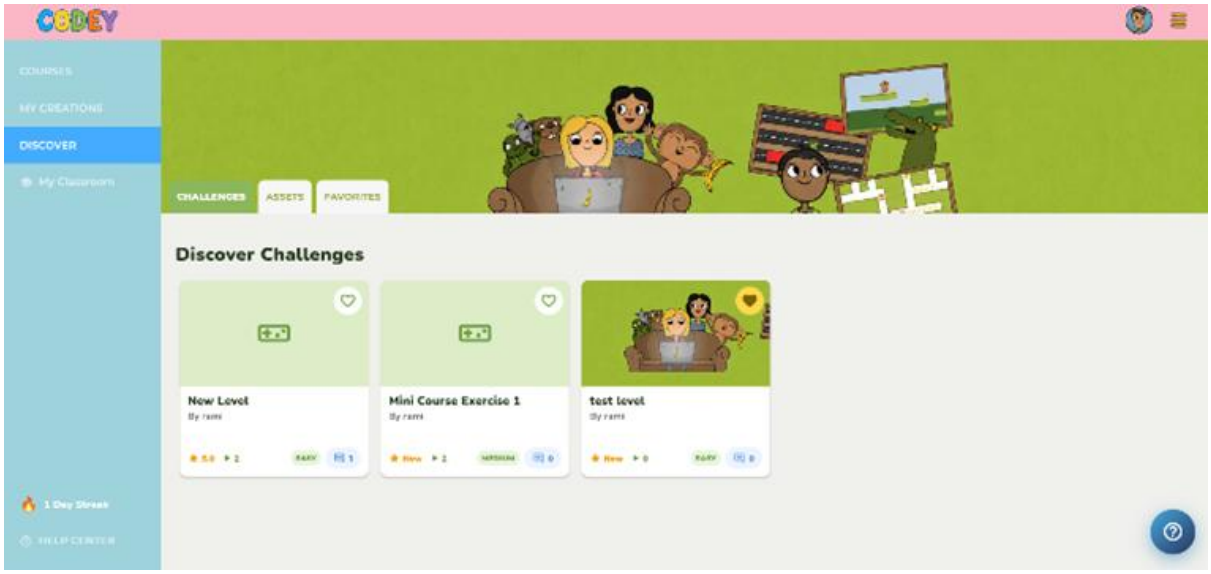


Figure 4.12.2: Challenges tab showing published community challenges.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

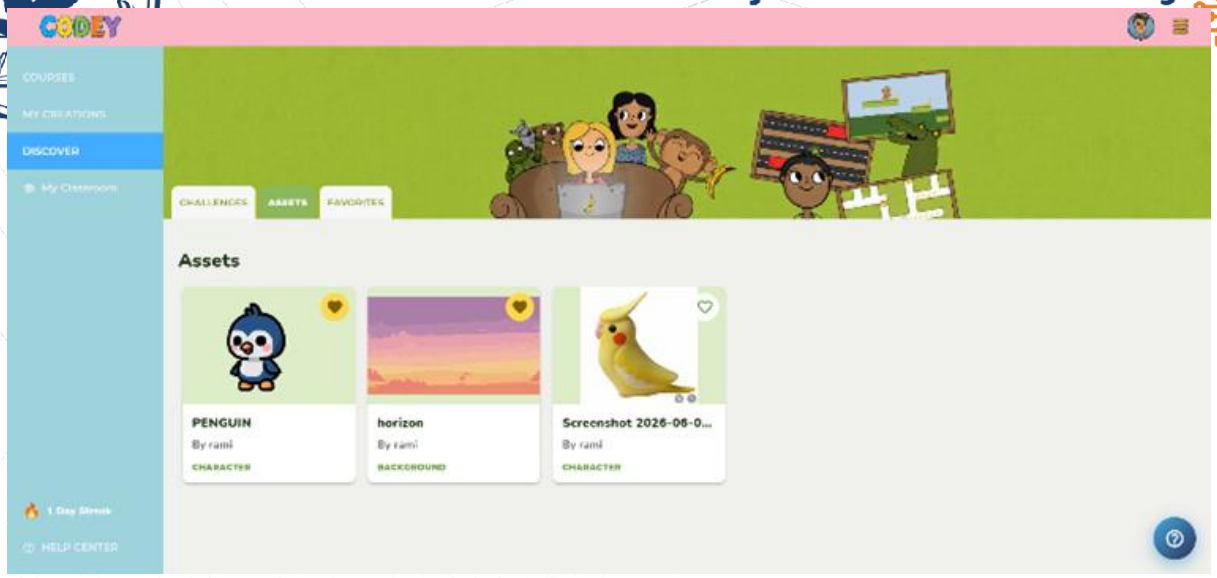


Figure 4.13.3: Assets tab showing reusable community assets.

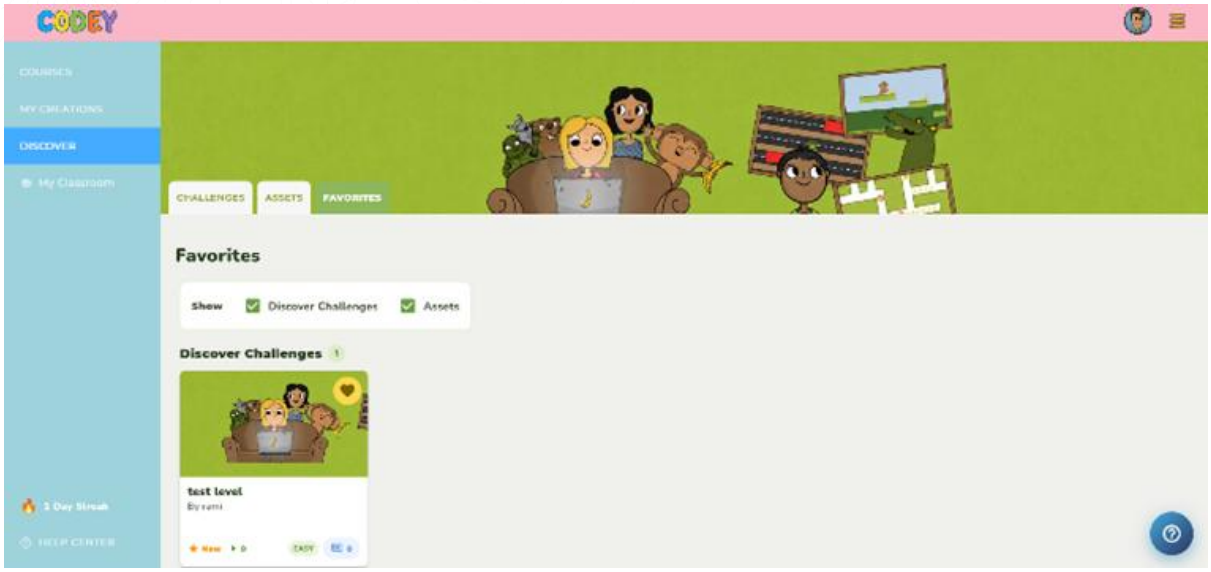


Figure 4.13.4: Favorites tab showing saved challenges and assets.

4.13.3 Challenge Cards

Each published challenge is displayed as a card. The card gives the student enough information to decide whether they want to open the challenge before playing it. The card design is visual and compact, which makes it suitable for younger users.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

A challenge card includes the following information and controls:

- A preview area or cover image that represents the challenge visually.
- The challenge title, such as “New Level”.
- The creator name, displayed using the “By” label.
- A star rating average, such as 5.0, to show how other players evaluated the challenge.
- A play count showing how many times the challenge has been opened or played.
- A difficulty badge, such as Easy, Medium, or Hard.
- A comment counter showing how many discussion messages are attached to the challenge.
- A heart icon that allows the student to mark the challenge as a favorite.

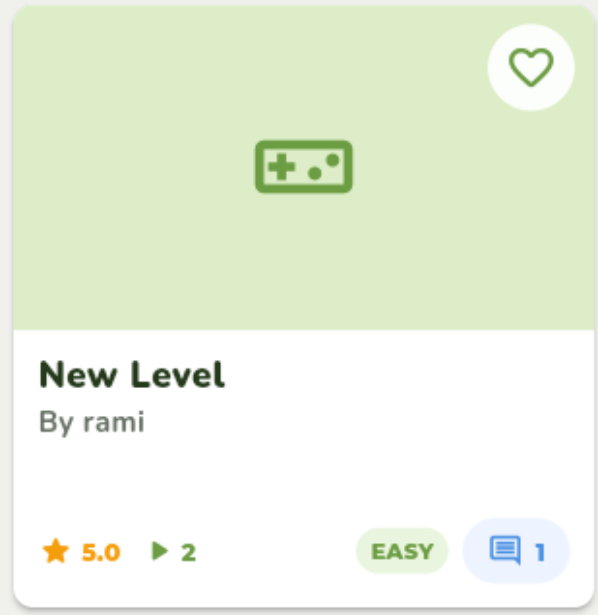


Figure 4.13.5: Challenge card showing rating, play count, difficulty, comments, and favorite button.

4.13.4 Opening Published Challenges

When a student selects a challenge card, the platform opens the correct play view depending on the builder type used to create the challenge. Technically, the frontend maps each published project into a SavedBuilderProject object. This object includes a builderType field, which can identify the project as a front-view builder, top-view builder, Scratch builder, or fourth game builder project.

The Discover page uses the builder type to route the student to the correct game view. For example, top-view projects are opened in the top-view builder play mode, Scratch projects are opened in the Scratch builder route, fourth-builder projects are opened in the fourth game builder, and front-view projects are opened in the normal builder play route. The route data passes the session, project ID, project title, play mode flag, and published-access flag so the challenge can be played without giving the viewer edit ownership.

Technical behavior used when opening a challenge:

- The project ID is passed to the selected builder route so the correct saved project can be loaded.
- allowPublishedAccess is used so a published project can be opened by users other than the owner.
- playMode is used so the challenge opens as a playable level rather than an editing workspace.
- The rating prompt is enabled when the current user has not already rated the project.

4.13.5 Rating, Plays, and Comments

The Discover page uses engagement features to make community content more meaningful. These features help students identify interesting challenges and give creators feedback about how their work is being received.

The play count is increased when a published challenge is played. This indicates how many times the challenge has been opened by learners. The backend stores this value in the builder project record as playCount.

The rating system allows students to rate a challenge using a value from one to five stars. The backend stores ratings as user-specific entries, then calculates the rating average and rating count when project data is returned to the frontend. This prevents the card from needing to store only one fixed rating value and allows every user to submit their own rating.

The comment system allows students to leave short feedback messages on a challenge. Comments are stored with the user ID, user name, message, and timestamp. To keep comments manageable, the backend validates that a comment is not empty and limits the message length to 500 characters.

4.13.6 Challenge Discussion Modal

The Challenge Discussion popup opens when the student clicks the comment area of a challenge. This modal displays the name of the challenge, the number of comments, the current average rating, and the number of plays. Below that, the existing comments are listed with the commenter name, message, and date.

At the bottom of the modal, the student can type a new comment in the feedback field and press the Post button. This sends the comment to the backend through the builder project comments endpoint. After the request succeeds, the project data is updated and the comment count shown on the card can also change.

- The modal supports social feedback without leaving the Discover page.
- The comment count on the card gives a quick preview of how active the discussion is.
- The design keeps the discussion focused on one challenge at a time.

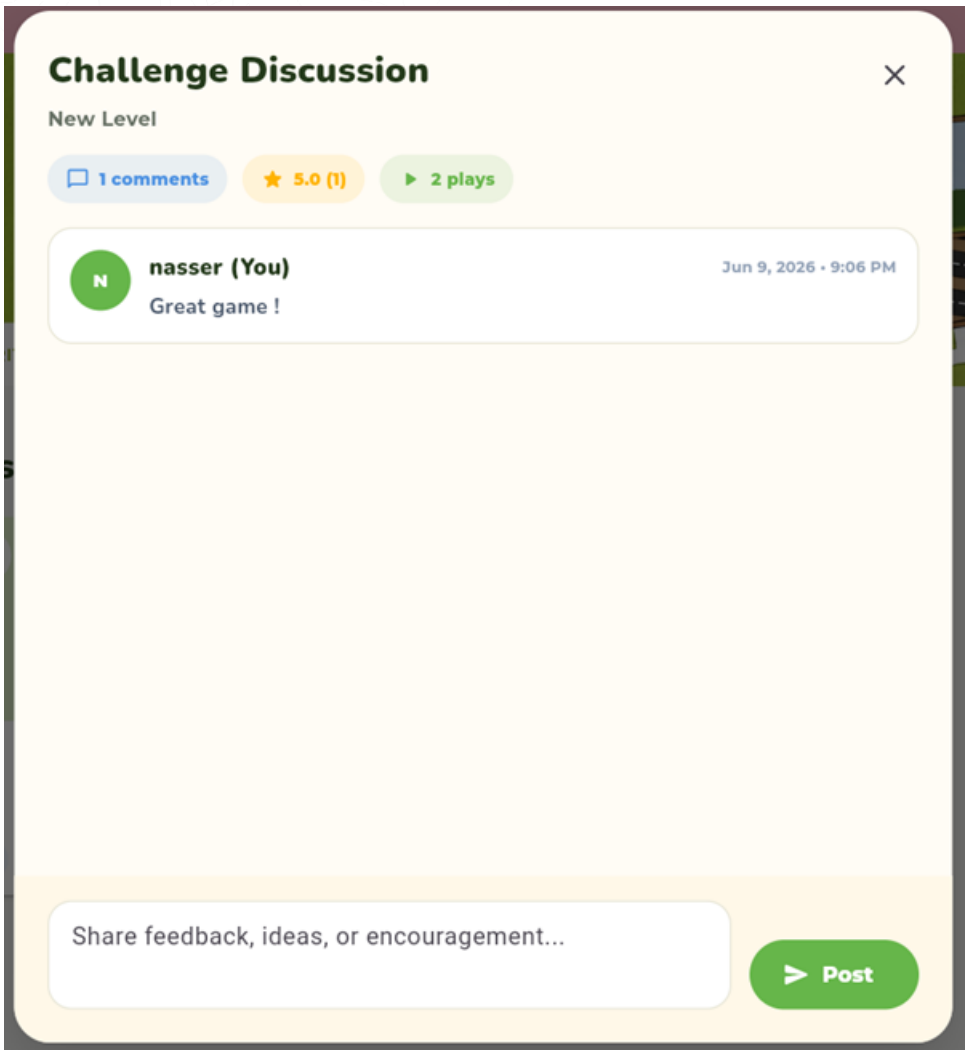


Figure 4.13.6: Challenge Discussion modal with comments, rating, play count, and posting field.

4.13.7 Rating a Challenge After Playing

After a student finishes or leaves a play challenge, the system can show a rating dialog. The purpose of this dialog is to collect feedback while the experience is still fresh. The dialog asks the user to rate the challenge before leaving, with five stars displayed as selectable options.

The dialog provides three actions. Keep Playing returns the student to the challenge. Leave exits without submitting a rating. Rate & Leave saves the selected rating and then exits the challenge. This makes the rating process optional while still encouraging students to support the community by reviewing the challenge.

Technically, the frontend checks whether the current user has already rated the project. If the currentUserRating value is empty, the challenge can show the rating prompt. If the user already rated the project, the system can avoid asking again, preventing repeated rating prompts for the same challenge.

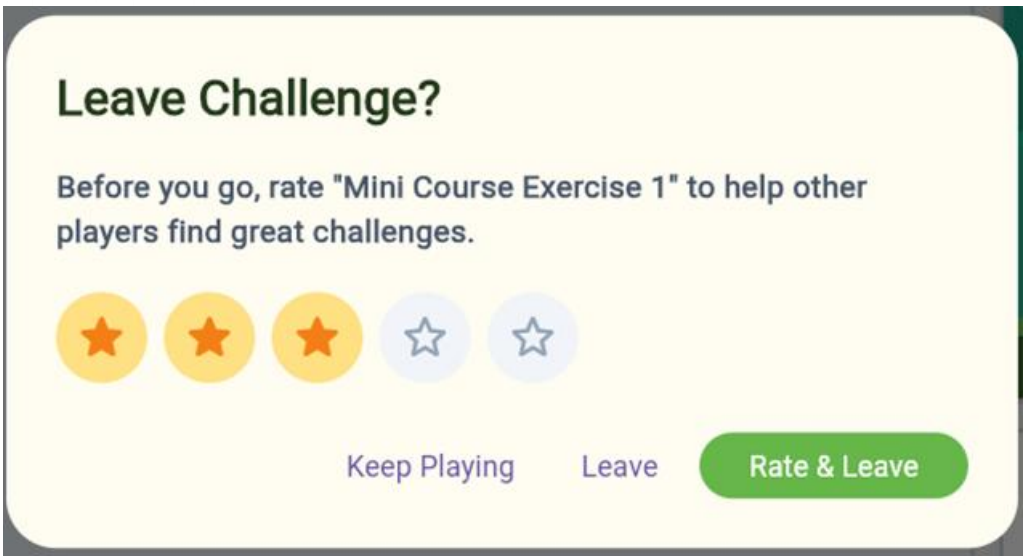


Figure 4.13.7: Rating popup displayed when leaving a played challenge.

4.13.8 Favorites Feature

The heart icon shown on challenge and asset cards is used to mark items as favorites. This feature helps students save interesting content and return to it later through the Favorites tab. It is especially useful when the Discover page contains many community items, because the student can keep a personal collection of preferred challenges and assets.

From a usability perspective, the favorite button is simple and visual. Children can recognize the heart icon quickly, and the Favorites tab gives them a dedicated place to find the items they selected. This supports repeated play, asset reuse, and easier navigation inside the community area.

4.13.9 Backend and Data Structure Support

The Discover page is supported by the builder project backend. The frontend requests published projects from the backend using the published projects API. The backend returns only projects with published status, and the returned data includes the project title, description, builder type, difficulty, owner information, cover image data, play count, comments, and ratings.

The builder project model stores both project data and community engagement data. Important stored fields include:

- status, which determines whether a project is draft or published.
- builderType, which identifies whether the project belongs to the front-view, top-view, Scratch, or fourth game builder.
- difficulty, which describes the challenge level as easy, medium, or hard.
- playCount, which records how many times the published challenge has been played.
- comments, which stores discussion messages with user information and timestamps.
- ratings, which stores one-to-five star ratings connected to users.
- coverImageBase64 and cover frame values, which allow the challenge card to display a saved visual preview.

When the backend serializes a project for the Discover page, it calculates the rating average, rating count, comment count, and current user rating. This makes the frontend card simpler because it receives ready-to-display engagement values instead of manually calculating them every time.

4.13.10 Educational and Community Value

The Discover page adds an important community layer to Codey. Students are not only solving lessons or building private projects; they can publish their work, receive feedback, and explore ideas from others. This encourages creativity, motivation, and peer learning.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

The page also supports the educational goals of the platform in several ways:

- Students can learn from other students' game designs and level structures.
- Ratings and comments provide feedback that can encourage creators to improve their challenges.
- Play counts give creators a sense of how often their work is being used.
- Shared assets allow students to reuse creative materials and build richer projects.
- Favorites make it easier for students to collect useful examples and return to them later.

Overall, the Discover page extends Codey from an individual learning platform into a collaborative learning environment. It connects creation, publishing, playing, rating, commenting, and asset sharing in one place, giving students a stronger sense of ownership and participation inside the platform.

4.14 Game Builders

The platform includes several game builders that allow students to create their own interactive games and challenges. These builders are accessed from the My Creations section in the student dashboard. Instead of only completing ready-made lessons, students can design levels, choose game objects, test the logic, save projects, and publish their work.

Purpose of the builders

- Allow students to apply programming concepts in practical game creation.
- Support different learning levels, from beginner visual sequencing to advanced code-based game logic.
- Encourage creativity by allowing students to customize characters, assets, backgrounds, rules, and interactions.
- Teach testing and debugging through immediate visual feedback.

Builder types

- **Block Sequence Builder:** A front-view builder focused on visual command sequencing.
- **Code Sequence Builder:** A top-view builder focused on typed or arranged code commands.
- **Game Builder:** An advanced builder that combines a custom programming language, live preview, sprites, widgets, sounds, and instruction writing.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.14.1 Block Sequence Builder

The Block Sequence Builder is designed for younger students who are still learning the basics of programming. It uses a front-view game scene where the character is viewed from the side, similar to a side-scrolling or platform-style game.

Figure 4.14.1: Block Sequence Builder main interface.

Main idea

- **Visual sequencing:** The student solves the level by arranging movement blocks in the correct order.
- **No typing required:** Commands are represented as blocks, so the learner does not need to write code syntax.
- **Step-by-step execution:** When the sequence is run, the character follows the blocks one after another.
- **Immediate feedback:** The student watches the character move and can identify whether the sequence is correct.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Level creation workflow

1. Choose or customize the level background.
2. Place the player character in the starting position.
3. Add obstacles, collectible items, and the finish/goal object.
4. Assign score values to collectible items.
5. Build a block sequence that moves the character from start to finish.
6. Run the level to test whether the solution works.
7. Edit the blocks or objects if the result is incorrect.
8. Save or publish the level after testing.

Figure 4.14.2: Block sequence execution area.

Customization panel

The builder includes a side panel that gives the creator control over the main level elements. This panel makes the builder more flexible and allows each student to design a level with a unique style.

Area	Function
Tools	Select the object type to place in the scene, such as ground, obstacle, player, collectible, or goal.
Character	Choose the player character used in the level.
Direction	Control the character direction or starting orientation.
Collectible	Choose the collectible item that appears in the level.
Score	Set the number of points awarded for each collectible item.
Background	Select the background style used in the game scene.
Grid size	Choose the level size, such as a small or large grid, to control the amount of available space.

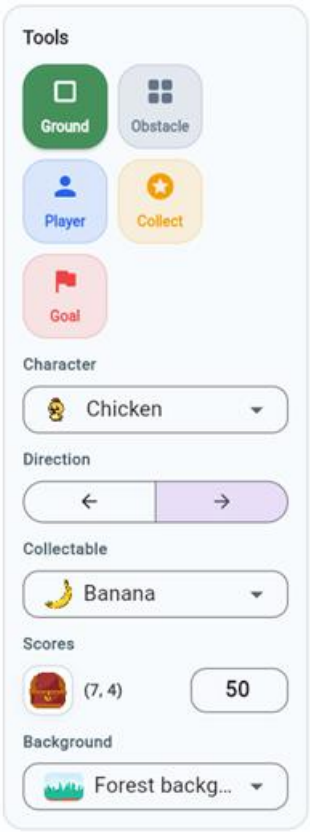


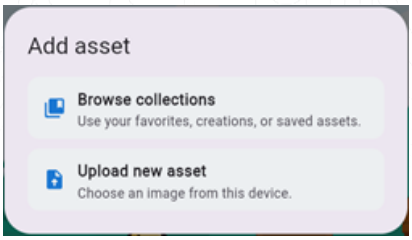
Figure 4.14.3: Object and asset customization tools.

Figure 4.14.4: Grid size and level actions.

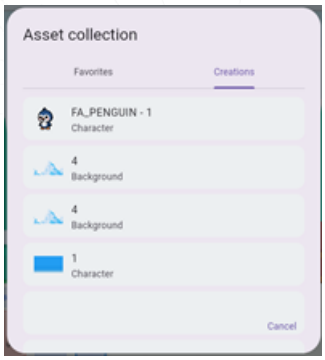
Custom assets

To make game creation more dynamic, the Block Sequence Builder supports custom asset creation. Users can upload an image from their device to create a unique asset, or they can browse and reuse assets collected or created by other users.

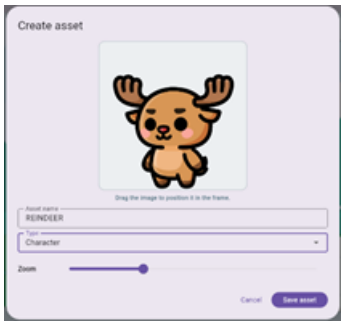
- Upload a new asset from the user device.
- Browse existing asset collections.
- Reuse saved or collected assets.
- Create a more personalized visual style for each level.



Add asset options



Asset collection



Create asset screen

Print Solution feature

The Print Solution button simplifies solution creation and testing. It automatically generates a valid sequence for the created level by walking the character from the starting position to the finish while collecting all collectible items along the way.

- Helps the creator check that the level is solvable.
- Provides a ready solution path for testing or demonstration.
- Reduces the time needed to manually create a correct solution.
- Supports debugging by showing the expected path through the level.

Difficulty recommendation before publishing

Before publishing a level, the creator selects a difficulty. The system recommends a difficulty level based on the complexity of the created level. The recommendation is calculated using the number and type of building blocks and solution blocks, where different block types contribute different point values. After the recommendation is displayed, the creator can still choose the final difficulty manually.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.14.5: Difficulty selection dialog.

Educational value

- Introduces sequencing in a simple visual way.
- Reduces syntax errors by using blocks instead of typed code.
- Teaches cause and effect between commands and character movement.
- Encourages trial-and-error learning and basic debugging.
- Prepares students for more advanced code-based builders.

4.14.2 Code Sequence Builder

The Code Sequence Builder introduces students to text-based command writing while keeping the game environment simple and visual. It uses a top-view scene, making it suitable for grid movement, path planning, maze-style levels, collecting objects, and avoiding obstacles.

Figure 4.14.6: Code Sequence Builder main interface.

Main idea

- **Top-view movement:** The character moves across the scene from above using directions such as up, down, left, and right.
- **Command writing:** The student writes or arranges commands that control the character.
- **Path planning:** The student analyzes the map, avoids obstacles, collects items, and reaches the goal.
- **Bridge to coding:** The builder helps students move from visual blocks toward written programming commands.

Code editor and solution blocks

The right side of the interface contains the code area, run controls, and solution blocks. Students can type commands directly or use the available blocks as a guide for constructing the solution.

- Run button executes the command sequence.
- Reset button returns the character to the starting state.
- Clear button removes the written solution.
- Solution blocks provide available commands such as step, turn, up, and right.
- The Hint button can guide students when they need help.

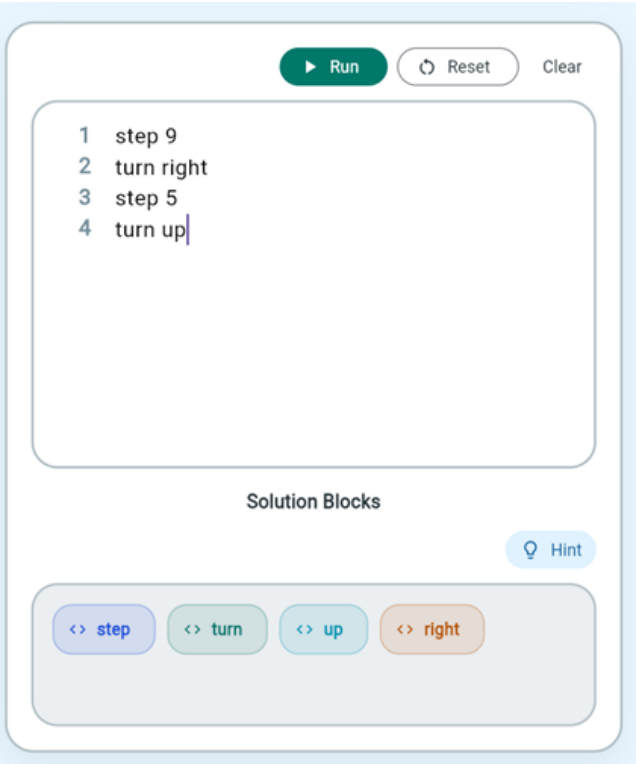


Figure 4.14.7: Code editor and solution blocks.

Level logic and execution

When the student runs the code sequence, the system reads the commands in order and moves the character inside the top-view scene. If the commands are correct, the character reaches the goal or completes the required task. If the commands are incorrect, the character may move in the wrong direction, miss a collectible, collide with an obstacle, or fail to finish the level.

- The builder teaches students to plan before writing commands.
- The output is visual, making mistakes easier to understand.
- Students can edit the commands and run the level again.
- This repeated process introduces debugging as a normal part of programming.

Editing and customization features

The Code Sequence Builder includes the same general editing options as the Block Sequence Builder, but it applies them to a top-view board. The creator can customize the scene and level elements before writing the solution.

- Upload or reuse custom assets.
- Change the player character.
- Change collectible items and obstacle types.
- Control the score value of each collectible item.
- Choose the board style and visual theme.
- Place objects in a way that creates a path-planning challenge.

Print Solution, turn angles, and difficulty

The Code Sequence Builder also supports the Print Solution feature. In this builder, the solution can include turn angles as part of the generated path. This is useful because the top-view character may need to rotate or change direction while moving through the level.

- Use Turn Angles can be enabled when the generated solution should include turning direction.
- Print Solution generates a path from the start position to the finish while collecting required items.
- Clear Level removes the current board content and logic.
- The difficulty recommendation is based on the complexity of the level and generated solution.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Figure 4.14.8: Level actions with turn angles.

Figure 4.14.9: Difficulty recommendation.

Educational value

- Teaches algorithmic thinking and path planning.
- Introduces written command sequences in a controlled environment.
- Encourages students to analyze the level before coding.
- Provides immediate visual feedback for debugging.
- Acts as an intermediate step between block sequencing and advanced game logic.

4.14.3 Game Builder

The Game Builder is the most advanced builder in Codey. It provides a more complete game creation environment where students can design a game scene and control it using a custom programming language with a live preview. This builder is intended for students who are ready to move beyond simple sequencing and begin working with more flexible game logic.

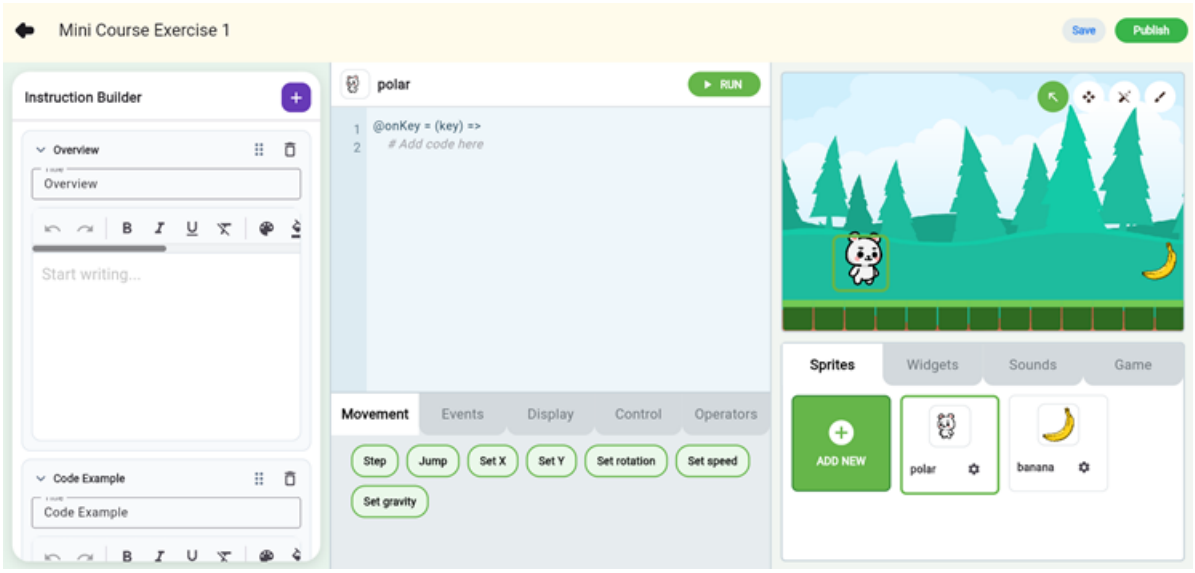


Figure 4.14.10: Game Builder main interface.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Main interface areas

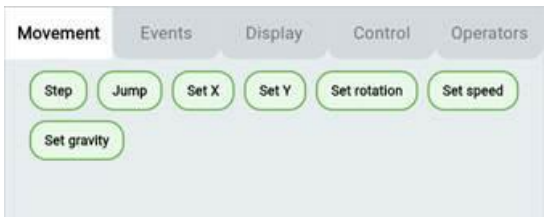
Area	Function
Instruction Builder	Allows the creator to write lesson instructions, steps, and explanations for the player. It supports text styling so instructions can be clear and organized.
Code Editor	Allows the student to write code using the custom programming language. The code controls sprites, widgets, sounds, events, and game behavior.
Live Game Preview	Shows the game scene and lets the creator run the code to immediately test the result.
Asset Tabs	Organize game components into Sprites, Widgets, Sounds, and Game settings.

Custom programming language

The Game Builder uses a custom programming language designed for this project. The language is simpler than professional programming languages but still introduces important concepts such as events, conditions, loops, movement, display control, operators, and object interaction.

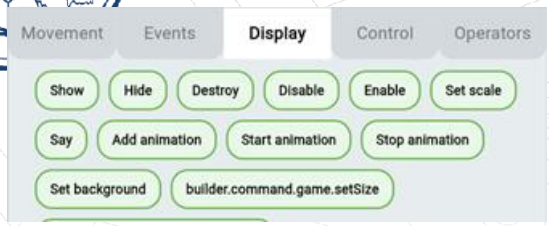
Command categories

- **Movement:** Commands such as step, jump, set X, set Y, set rotation, set speed, and set gravity.
- **Events:** Triggers such as on start, on key, on collide, on click, on world bounds, and on animation end.
- **Display:** Commands for showing, hiding, destroying, disabling, enabling, scaling, starting animations, stopping animations, setting background, and changing the game size.
- **Control:** Commands for loops, repeated actions, conditions, functions, returns, timers, and timing operations.
- **Operators:** Logical and comparison operators such as not, and, or, equals, less than, greater than, true, and false.



Movement commands

Events commands



Display commands

Control commands

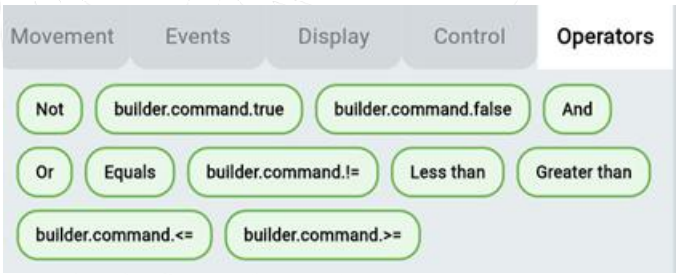


Figure 4.14.11: Operators commands.

Live preview, validation, and debugging

A live preview is included so the student can run the code and immediately see the result. If the code works correctly, the game behaves as expected. If there is an error, the builder displays a validation message that helps the student identify the problem.

- Validation messages identify unknown or invalid commands.
- The interface can guide the student to the error location.
- Students can edit the code and test again immediately.
- Debugging becomes part of the learning process instead of a separate technical step.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

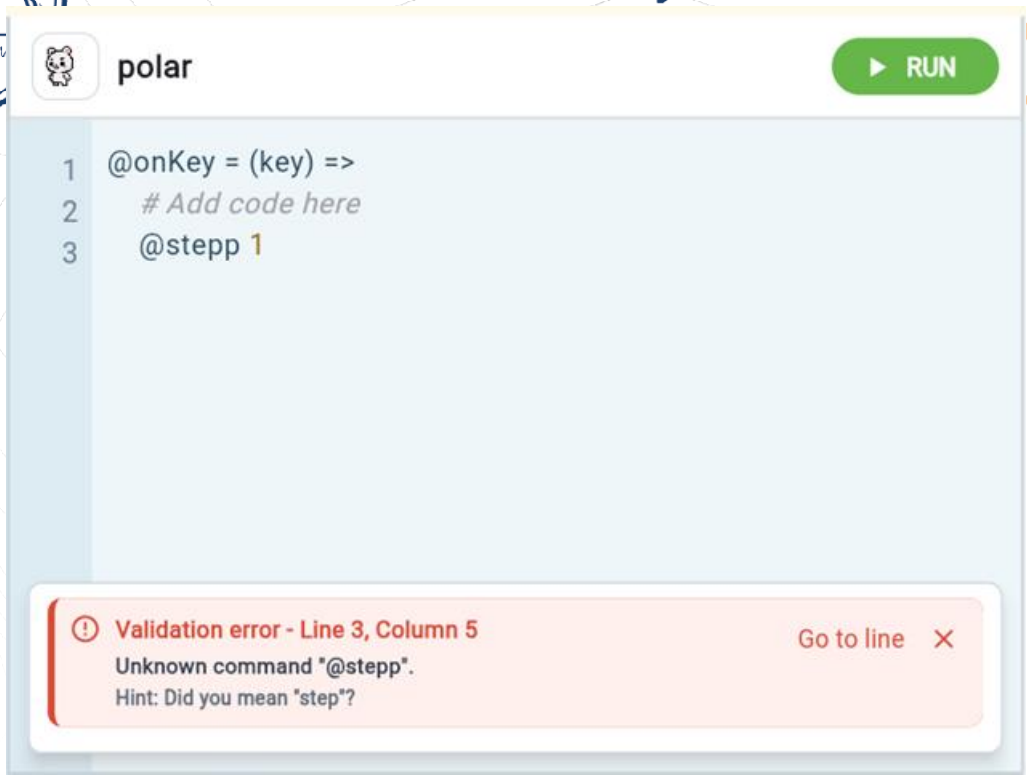


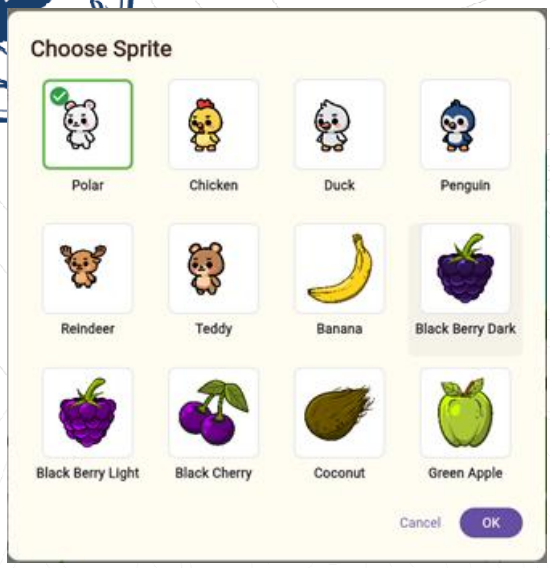
Figure 4.14.12: Code validation and error feedback.

Sprites

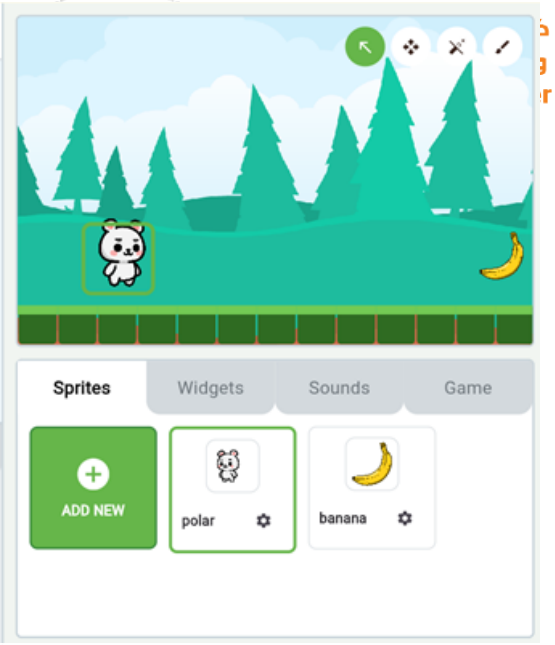
Sprites are the visual characters and objects used inside the game. The creator can add different sprites, select from available characters and objects, and place them in the game preview area.

- Choose characters such as Polar, Chicken, Duck, Penguin, Reindeer, or Teddy.
- Add object sprites such as bananas or fruits.
- Use sprites as player characters, collectibles, or interactive objects.
- Control sprite behavior through code.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu



Choose Sprite dialog

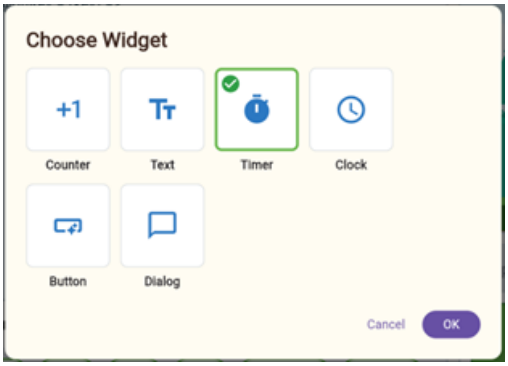


Sprites tab after adding objects

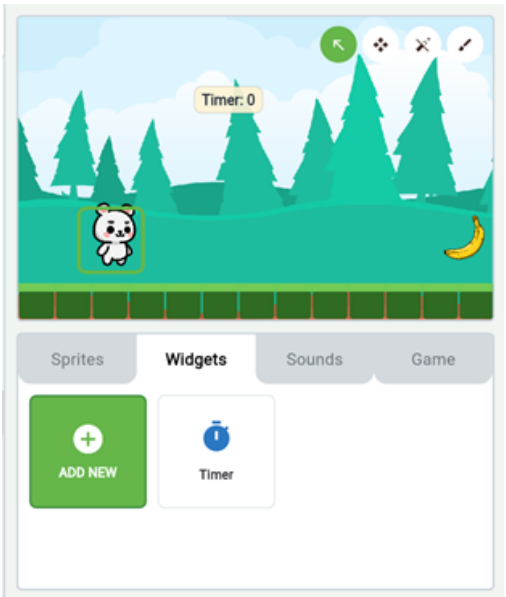
Widgets

Widgets are interface elements that can be added to the game. They allow the creator to display information or create interactive elements inside the game scene.

- Available widgets include counter, text, timer, clock, button, and dialog.
- Widgets can be displayed in the game preview area.
- Code can control widget behavior, such as starting or stopping a timer.
- Widgets make the game more interactive and informative.



Choose Widget dialog



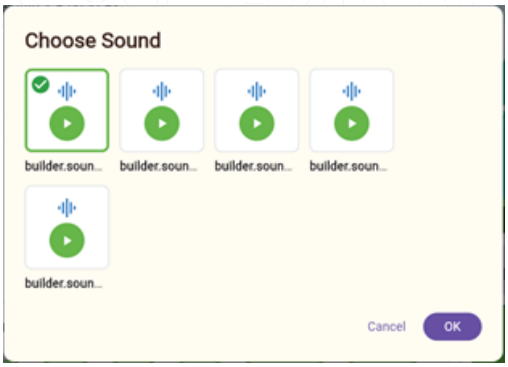
Widgets tab with timer widget

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

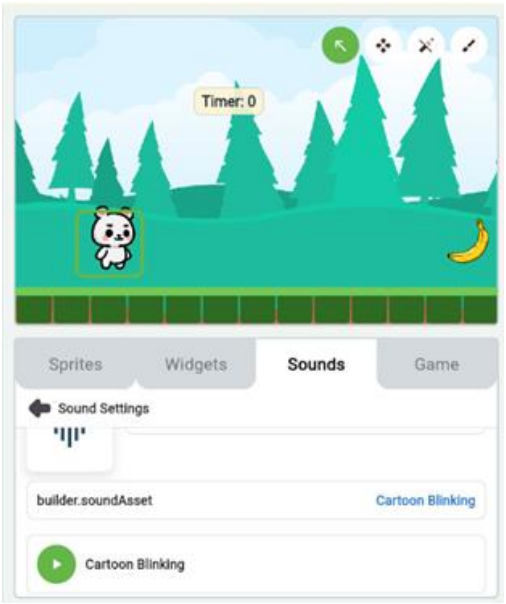
Sounds

The Game Builder supports adding sounds to make the game more engaging. Sounds can be selected from available sound assets and controlled through code.

- Choose different sound effects from the sound dialog.
- Preview sounds before using them.
- Add sounds to the game asset list.
- Trigger sounds from code during gameplay events.



Choose Sound dialog



Sounds tab after adding a sound

Instruction writing and text styling

The instruction writing area allows the creator to write instructions and steps for the player. It includes formatting tools so the instructions can be styled clearly and presented in an organized way.

- Write an overview or step-by-step instructions.
- Use text formatting tools such as bold, italic, underline, alignment, and lists.
- Separate explanations from the actual game code.
- Make the game easier for other students to understand.

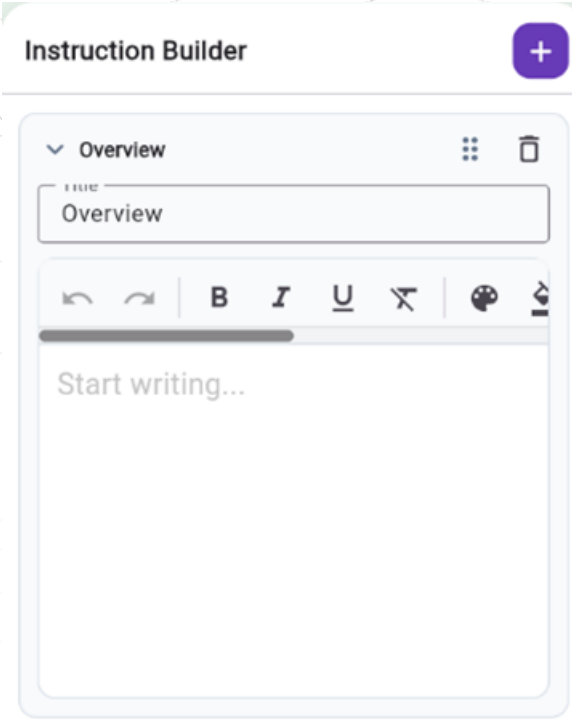
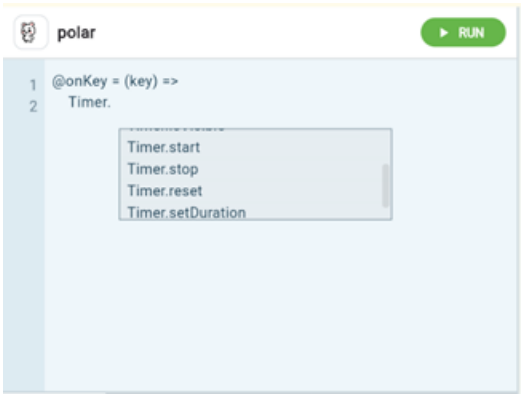


Figure 4.14.13: Instruction Builder and text styling tools.

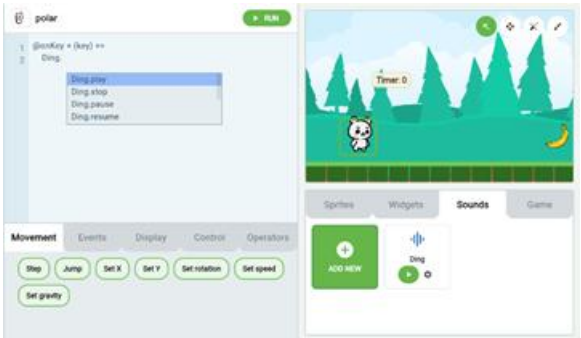
Controlling assets with code

A key feature of the Game Builder is that sprites, widgets, and sounds can be controlled by code. This allows the student to create interactive behavior instead of only placing objects visually.

- Use events to trigger behavior when the game starts, when a key is pressed, or when objects collide.
- Move and animate sprites using movement commands.
- Start, stop, or reset timers and other widgets through code.
- Play sound effects in response to game events.
- Use conditions and loops to create more advanced game behavior.



Widget code suggestions



Code controlling game elements

Educational value

- Teaches a deeper understanding of how code controls a game system.
- Introduces events, conditions, loops, operators, and debugging.
- Gives students freedom to build games based on their own ideas.
- Connects programming logic with sprites, widgets, sounds, and visual feedback.

4.14.4 Builder Learning Progression

The three builders are connected as a gradual learning path. Codey does not force children to move directly into complex text-based programming. Instead, it introduces programming in stages.

Area	Function
Block Sequence Builder	Beginner level. Students use visual blocks and focus on command order, cause and effect, and simple movement.
Code Sequence Builder	Intermediate level. Students write or arrange text-based commands while still receiving clear top-view visual feedback.
Game Builder	Advanced level. Students use a custom programming language to control sprites, widgets, sounds, events, and interactive behavior.

This progression helps students build confidence gradually. First they learn that commands must be ordered correctly, then they learn how to express commands using code, and finally they learn how code can control a complete game environment.

4.14.5 Testing and Feedback in the Builders

- Each builder includes a run or preview feature that allows the student to test the project.
- The feedback is mostly visual: the character moves, reacts, stops, collides, collects items, or reaches the goal based on the student's instructions.
- If the result is incorrect, the student can edit the blocks, commands, or code and run the project again.
- This repeated process teaches debugging naturally and shows that mistakes are part of programming.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.14.6 Saving and Continuing Builder Projects

The builders support saving projects so students can return to them later. When a game is saved, the system stores the builder type, level structure, placed objects, commands, custom assets, and related game data. This allows the project to reopen in the correct builder with the previous work restored.

- Supports long-term creative work.
- Allows students to improve games over multiple sessions.
- Keeps all student creations organized in My Creations.
- Prevents loss of progress during testing and editing.

4.9.7 Publishing Builder Projects

After completing a game, the student can publish it so that it appears in the Discover section. Publishing gives students a sense of ownership and achievement because their work becomes visible to others.

- Encourages creativity and peer learning.
- Allows students to explore games made by classmates or other users.
- Turns programming practice into a shareable final product.

4.15 Admin Dashboard

After logging in with an administrator account, the user is directed to the Admin Dashboard. This area is separated from the student and parent interfaces because it is intended for platform supervision, content control, and system management.

Main layout and access control

- **Dashboard structure:** The admin interface uses a left sidebar for navigation and a main content area for the selected management page.
- **Navigation sections:** The sidebar contains Dashboard, Courses, Levels, Notifications, Users, Statistics, and Profile.
- **Centralized management:** All admin tools are grouped in one interface, making it easier to move between platform overview, content management, user management, and statistics.
- **Protected access:** Only users with an administrator role can open this dashboard. Authentication and role-based access control protect sensitive actions such as deleting content, approving courses, suspending users, and viewing platform data.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.15.1 Dashboard Overview Screen

The Dashboard Overview screen is the first screen displayed inside the admin interface. It gives the administrator a quick summary of the platform before moving to more detailed management sections.

Displayed information

- Total courses available on the platform.
- Number of total levels and their current status.
- Number of registered users.
- Number of published and draft levels.
- Recently created content and overall system activity.

The dashboard values are loaded from protected backend admin endpoints. This screen acts as a starting point for monitoring the platform and identifying areas that may need attention.

Figure 4.15.1: Admin dashboard overview.

4.15.2 Courses Management Screen

The Courses Management screen allows the administrator to manage the educational content available on Codey. Since the platform supports both built-in courses and user-created courses, this page is essential for keeping the course catalog organized, safe, and suitable for children.

Main functions

- View all published, draft, and user-created courses.
- Display course information such as title, category, visibility, status, and level count.
- Create new courses from the admin interface.
- Edit existing course information and content.
- Delete courses when they are no longer needed or are unsuitable.

Manage the levels inside each course.

Review user-created courses before they become publicly available.

This screen supports content organization and moderation. It gives the administrator control over what appears in the student course catalog and helps ensure that the learning content remains appropriate.

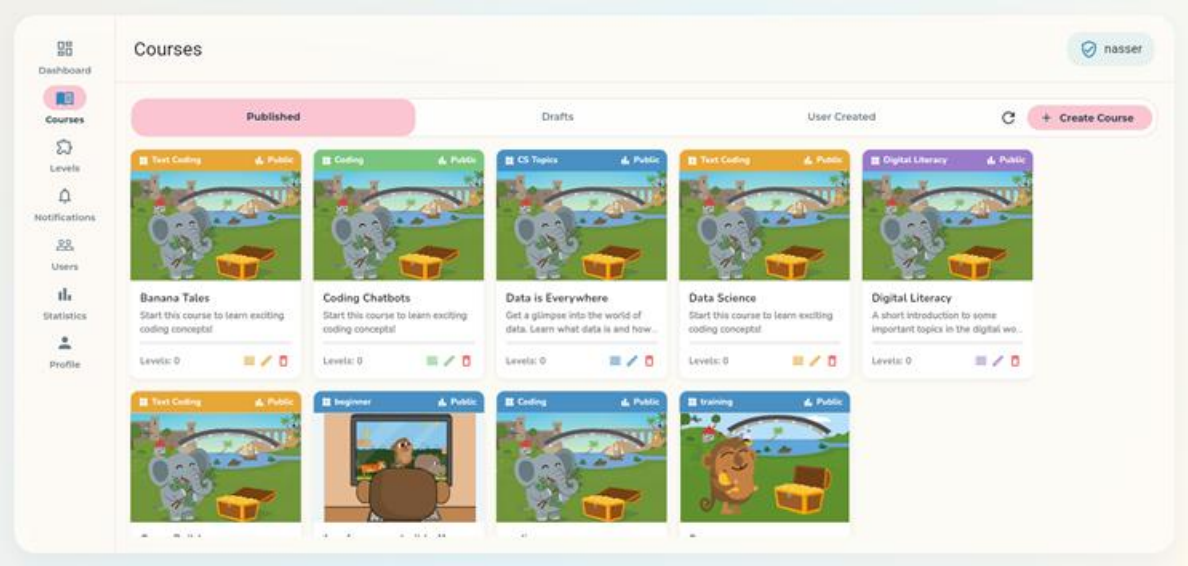


Figure 4.15.2: Courses management screen.

4.15.3 Levels Management Screen

The Levels Management screen is used to manage the levels connected to courses and game-based activities. Levels are a core part of Codey's learning experience because they provide progression, challenge, and practice.

Main functions

- View available levels across the platform.
- Separate levels according to their status, such as published, draft, or user-created.
- Open level details for review.
- Update level information when changes are required.
- Delete unsuitable, incorrect, or broken levels.
- Maintain the correct structure of course progression.

This section helps ensure that learners follow a clear and correct learning path. If a level contains incorrect information, broken logic, or unsuitable content, the administrator can modify or remove it.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

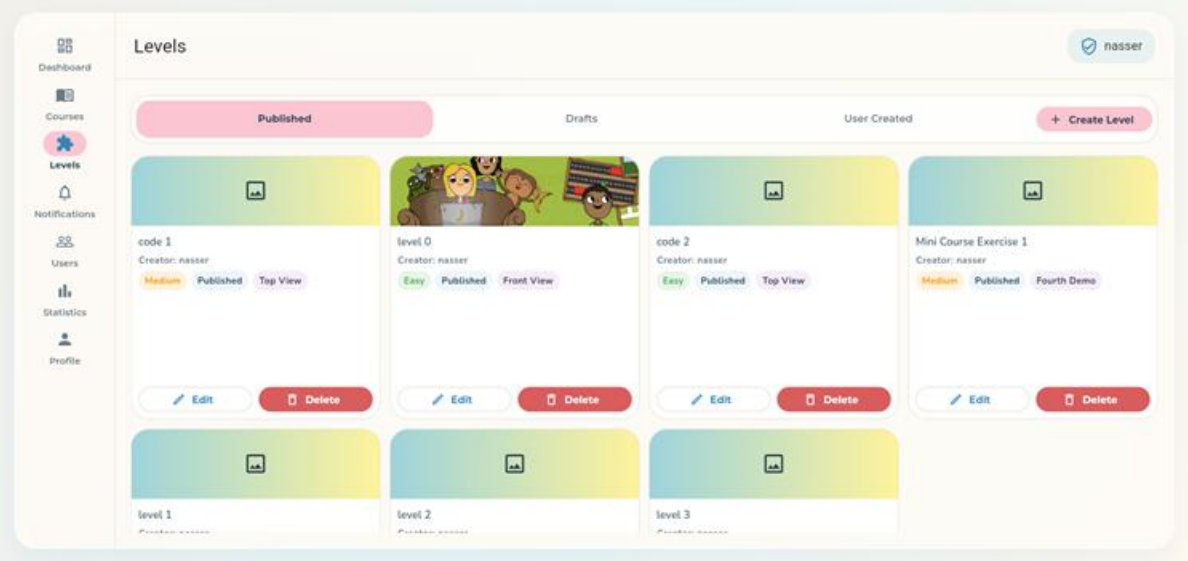


Figure 4.15.3: Levels management screen.

4.15.4 Notifications Screen

The Notifications screen helps administrators track important updates and pending review actions. It is especially important for content moderation because Codey allows users to create and submit courses.

Main functions

- View notifications related to course verification requests.
- Review courses submitted for public use.
- Accept suitable courses after checking their content.
- Reject incomplete or inappropriate courses.
- Receive updates about verified user-created courses.

Course verification creates a moderation layer between course creation and public publishing. This ensures that the content shown to children is complete, safe, and educationally suitable.

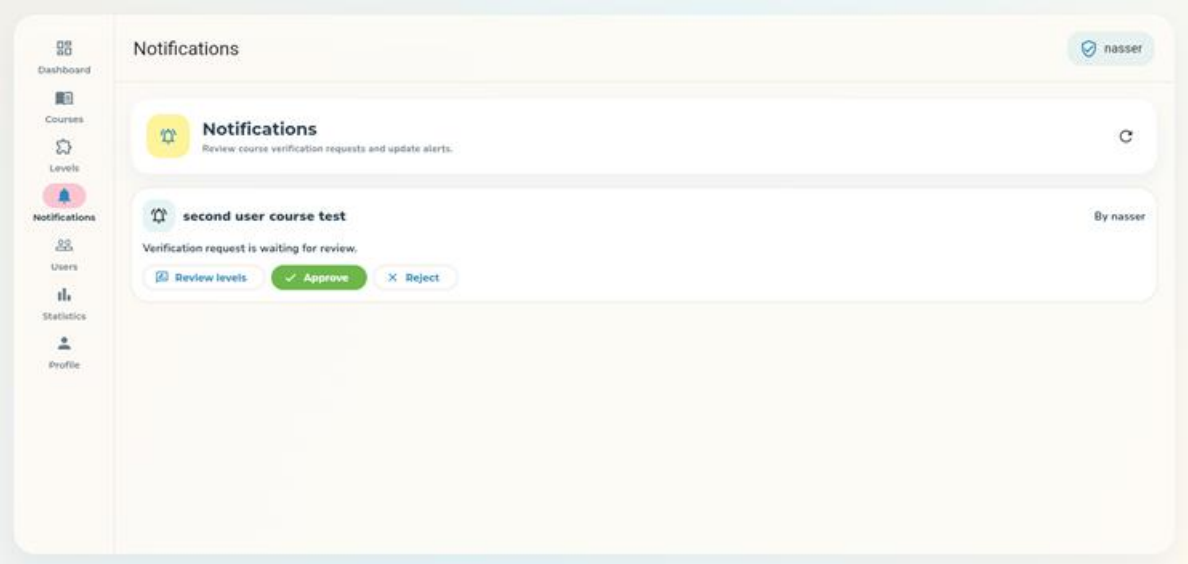


Figure 4.15.4: Notifications and verification requests.

4.15.5 Users Management Screen

The Users Management screen allows the administrator to manage registered accounts. Since Codey supports children, parents, and administrators, user management is necessary for platform safety and account organization.

Displayed user data

- User display name.
- Email address.
- User role, such as child, parent, or admin.
- Account status.
- Join date or account creation information.

Available actions

- Search for users by name or email.
- View user account information.
- Promote a user to administrator when needed.
- Suspend an account temporarily without deleting its data.
- Cancel admin access if required.
- Delete accounts when necessary.

Suspension is useful when an account should be blocked temporarily, while deletion is treated as a sensitive action. This section is especially important because the platform targets children and must maintain a safe learning environment.

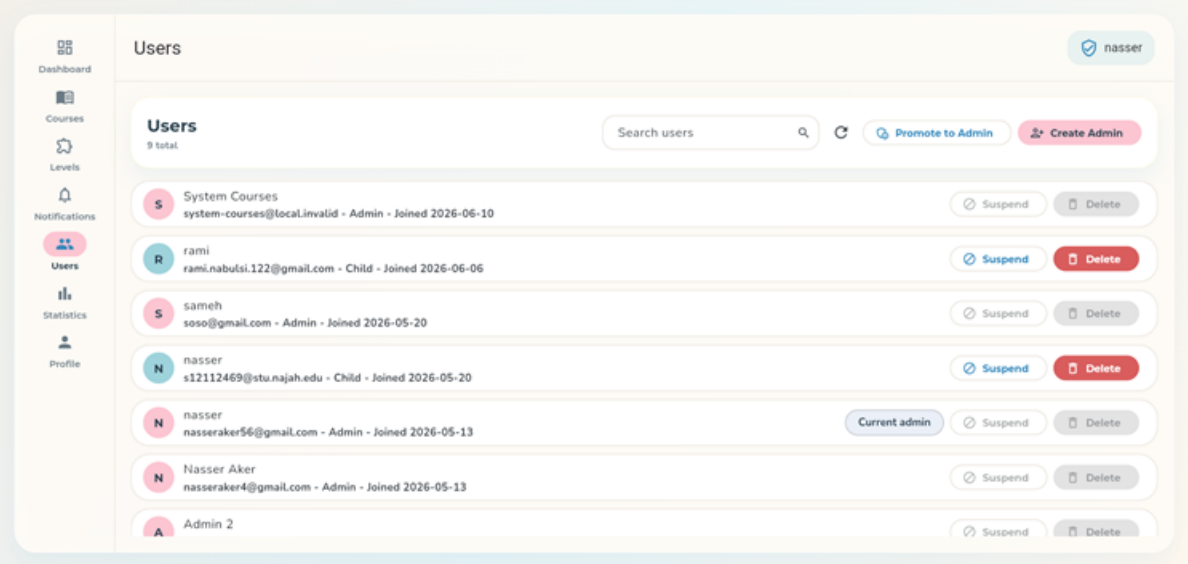


Figure 4.15.5: Users management screen.

4.15.6 Statistics Screen

The Statistics screen provides analytical information about the platform. It helps administrators understand how users interact with Codey and how the educational content is performing.

Statistics displayed

- Total number of courses.
- Users grouped by role, including children, parents, and admins.
- Levels grouped by status, such as published and draft.
- Course performance and course usage data.
- Platform activity indicators that can support future improvements.

These statistics help administrators evaluate the effectiveness of the platform, identify active areas, and decide which courses or features may need improvement.

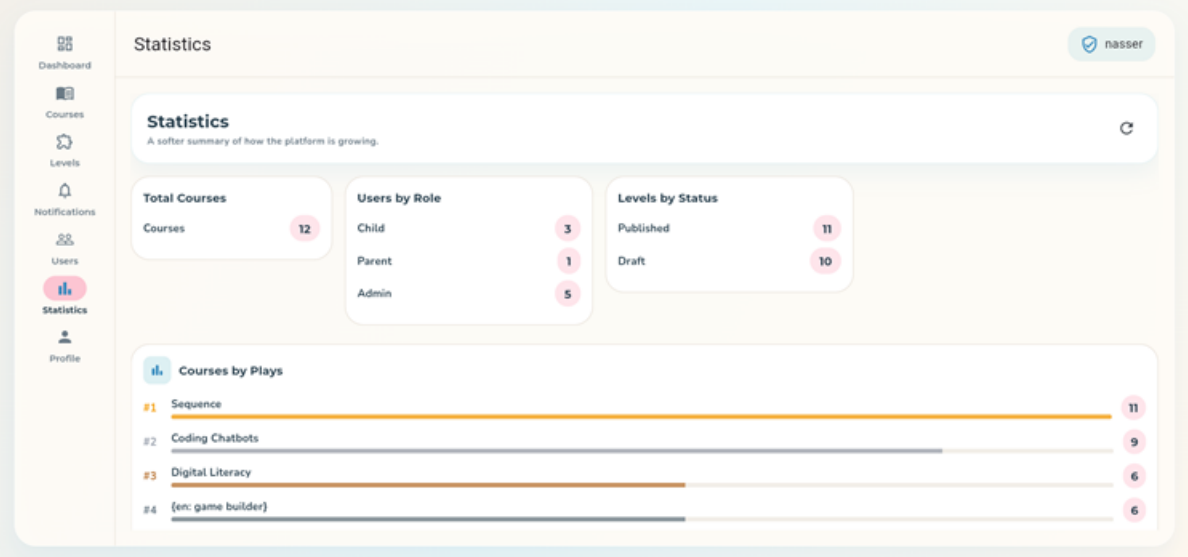


Figure 4.15.6: Statistics screen.

4.15.7 Admin Profile Screen

The Admin Profile screen allows the administrator to view and manage basic account information while keeping the admin account separate from student and parent accounts.

Profile information and actions

- Display the administrator name, email, role, and verification status.
- Show account activity such as last upload and last login information.
- Display quick statistics such as projects, published games, and drafts.
- Allow quick actions such as changing password, viewing saved items, changing language, and logging out.

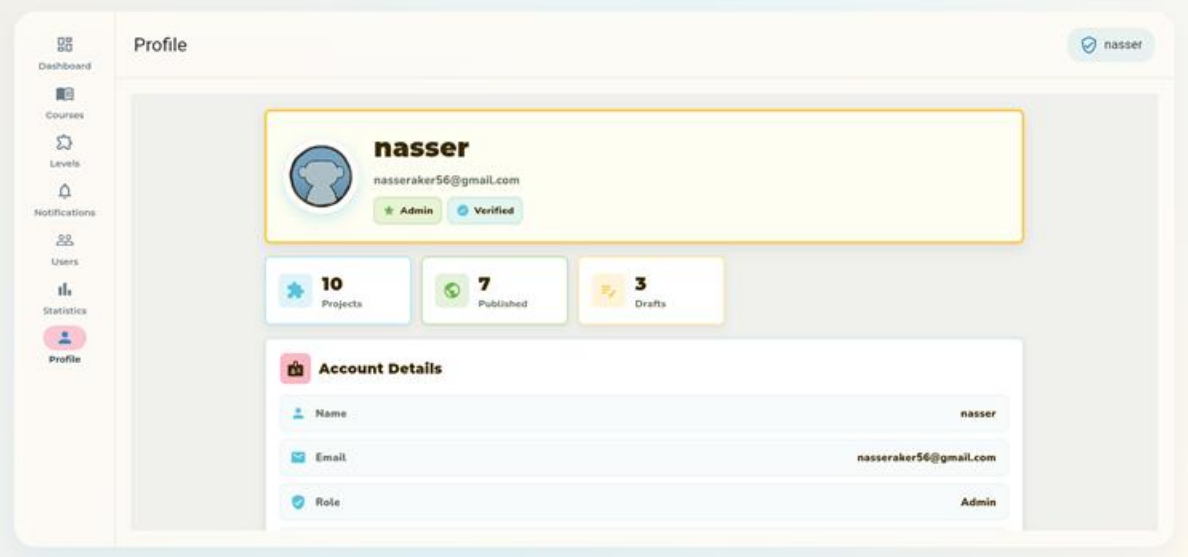


Figure 4.15.7: Admin profile overview.

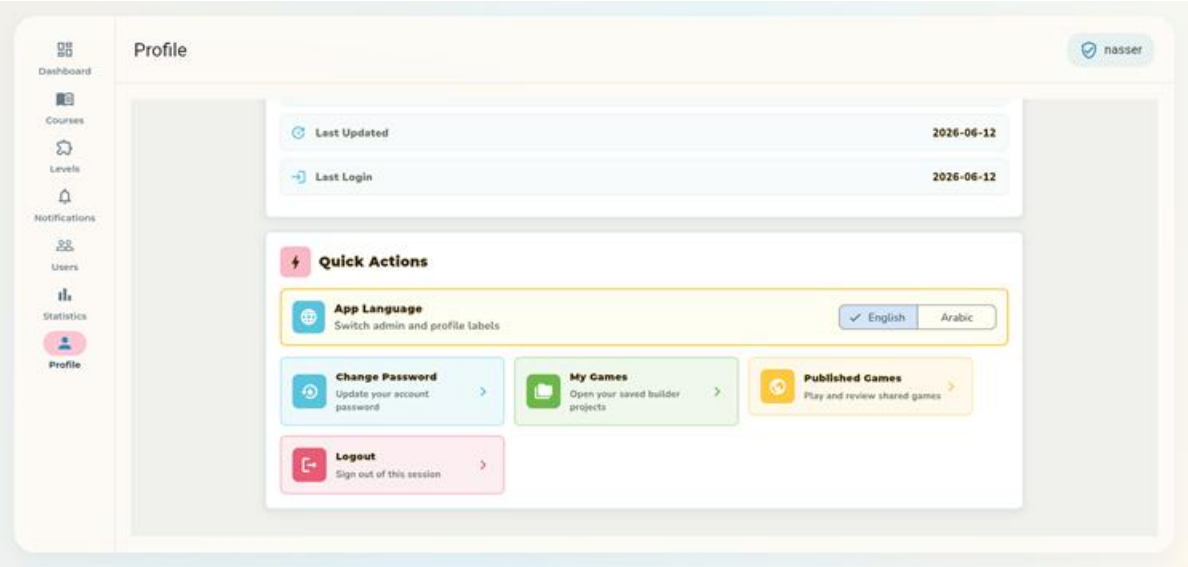


Figure 4.15.8: Admin profile quick actions.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.16 Shared Game-View Structure

Although the three game views use different levels of programming difficulty, they share a similar learning flow. Each level introduces the task, gives the student an area to build or write a solution, allows the student to run the result, and then provides feedback based on the outcome.

- **Level objective:** Each level begins with a clear goal, such as reaching a chest, moving to a target, collecting an item, or completing a required action.
- **How to Play popup:** At the start of the level, a short popup explains what the student needs to do before solving the challenge.
- **Solution area:** The student either arranges blocks, writes commands, or writes custom game code depending on the course type.
- **Run action:** The Run button executes the current solution and shows the result inside the game scene.
- **Reset action:** The Reset button returns the level to its initial state after the student tests a solution.
- **Clear action:** The Clear button removes the current solution so the student can start again.
- **Visual feedback:** The character movement, collisions, success state, and failure state are shown directly on the screen.

The hinting system is used across the level-based views to support students without immediately replacing their thinking. The system compares the student's current solution with the saved correct solution for the level. Based on this comparison, it can identify whether the student is missing a command, using an incorrect command, or placing a command in the wrong order. The hint then guides the student toward the next useful step instead of simply explaining the whole answer at once.

The score system is mainly used in the Block Sequence and Code Sequence game views. After the student completes a level successfully, a completion popup appears with a positive message, a star rating, and a numerical score such as 50 / 50. The popup also provides actions such as Play Again and Next, allowing the student either to retry the level or continue to the following challenge.

4.16.1 Block Sequence Game View

The Block Sequence Game View is the beginner-level game view. It is designed for young learners who are still being introduced to programming and may not be ready to write text commands. The level is displayed in a front-view scene, where the character is shown from the side in a colorful environment with a background, ground area, obstacles, collectible items, and a finish goal.

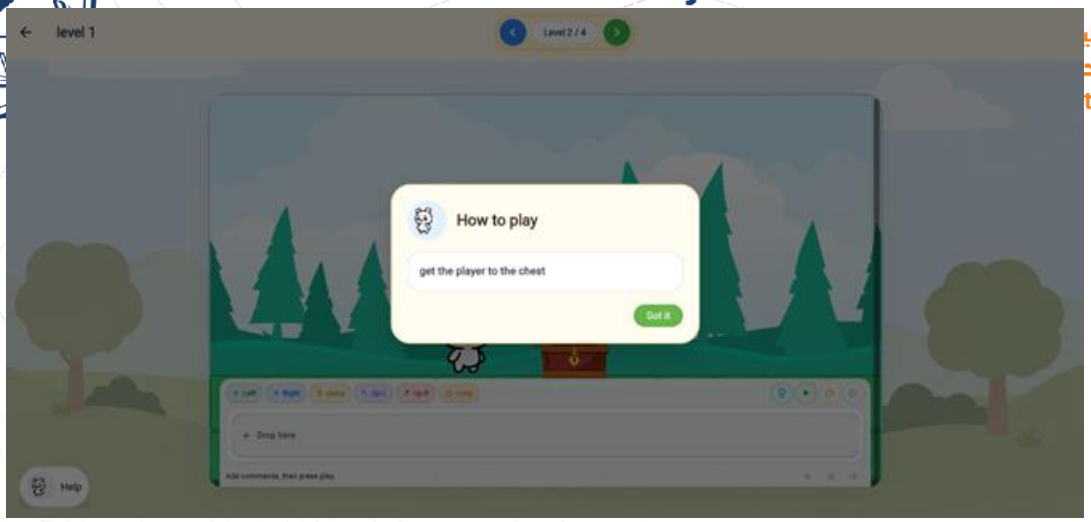


Figure 4.16.1: The Block Sequence level opens with a How to Play popup that explains the goal before the student starts solving.

The main idea of this view is command sequencing. The student is given visual blocks that represent actions the character can perform. The student places these blocks in the solution area in the correct order, then presses Run to execute them. The character follows the sequence exactly as arranged by the student.

- **Game scene:** Shows the character, background, obstacles, collectibles, and final target in a front-view layout.
- **Command blocks:** Provide simple actions such as moving, jumping, or direction-based movement depending on the level.
- **Solution strip:** The area where the student places blocks in the order they should be executed.
- **Run and playback controls:** Allow the student to test the solution and watch the character perform the commands.
- **Level instruction:** A short instruction is displayed to remind the student of the level objective.

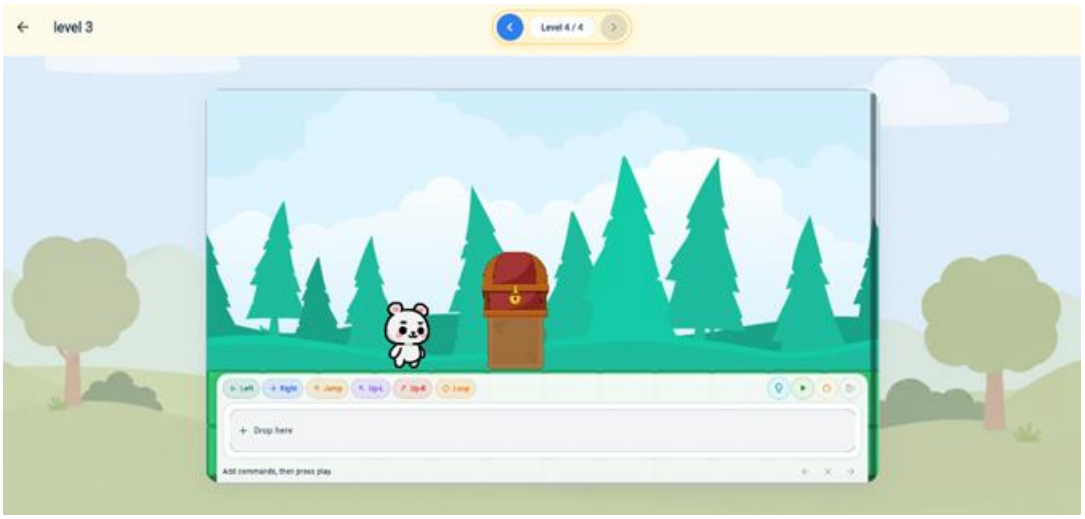


Figure 4.16.2: The Block Sequence Game View shows the front-view scene, the character, the goal, and the solution area.

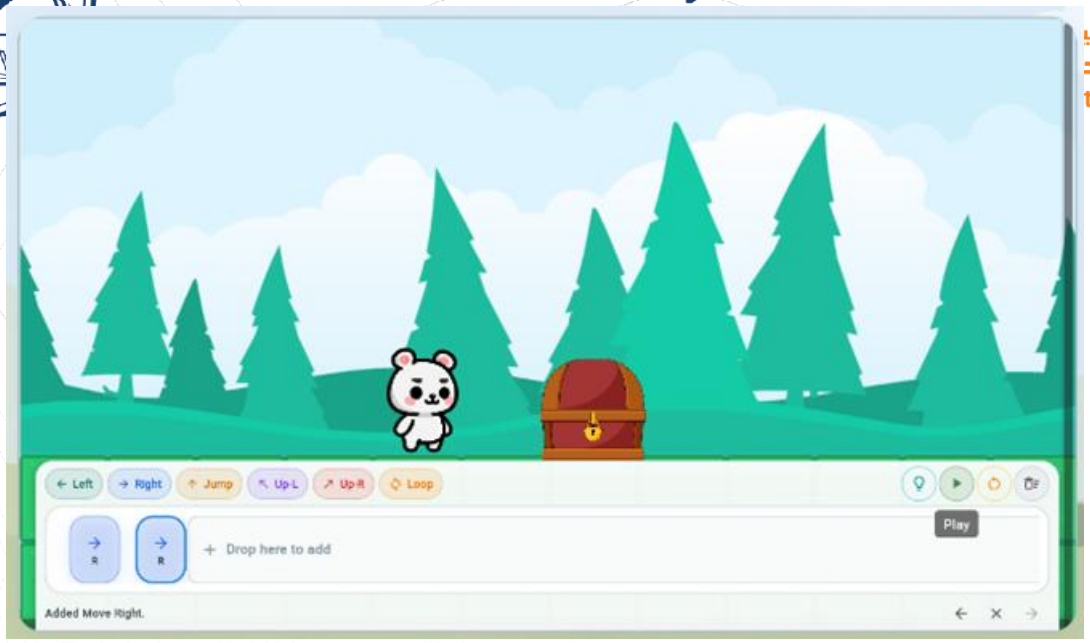


Figure 4.16.3: The solution area contains movement blocks that the student arranges to guide the character.

This view teaches the basic programming concept of ordering instructions. A program is not only a group of commands; it is a sequence where the order matters. If the student places a jump too early, forgets a move command, or arranges commands incorrectly, the character may fail to reach the goal. The student can then adjust the sequence and run the level again.

The Block Sequence Game View is useful because it removes syntax difficulties. The student does not need to worry about spelling, punctuation, or code structure. Instead, the focus is on solving the problem logically and understanding how each block affects the character.

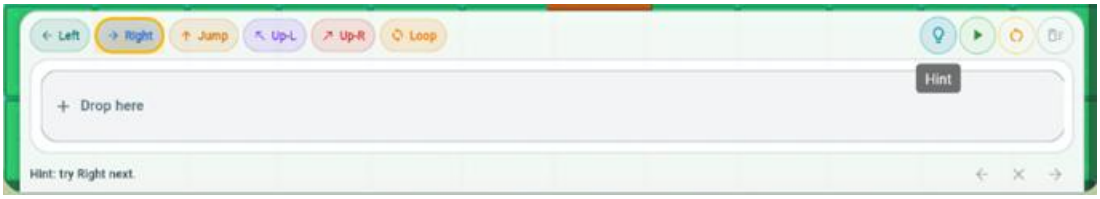


Figure 4.16.4: The bottom solution strip includes block options, the active sequence area, and the hint button.

When the level is solved correctly, the system displays a completion popup. The popup confirms that the level was finished, shows the number of earned stars, and displays the score. This gives the student a clear reward and encourages progression to the next level.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

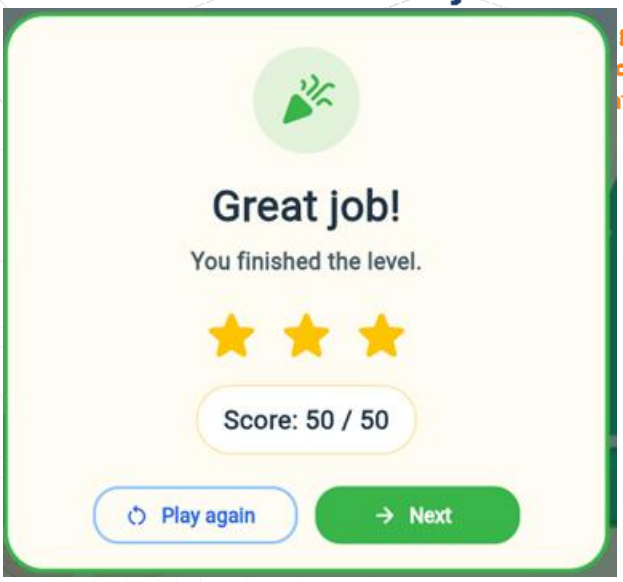


Figure 4.10.5: The completion popup shows the success message, stars, score, Play Again, and Next actions.

4.16.2 Code Sequence Game View

The Code Sequence Game View is the intermediate-level view. It introduces students to writing commands while keeping the environment visual and easy to understand. The level uses a top-view layout where the character moves on a map toward a target point or collectible item.

Figure 4.16.6: The Code Sequence level also begins with a How to Play popup that explains the goal.

The top-view layout is suitable for path planning. The student can analyze the starting position, the target location, the distance between them, and the direction the character must follow. This makes the level behave like a programming puzzle where the learner must plan the correct sequence before running the code.

- **Top-view game area:** Displays the character, target, and map from above.
- **Code editor:** Allows the student to write the solution as text commands, one command per line.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

- **Solution blocks:** Show the supported command options, such as step, turn, up, and right.
- **Hint button:** Uses the comparison between the current written solution and the stored correct solution to guide the next step.
- **Run, Reset, and Clear controls:** Allow the student to test, restart, or rewrite the solution.

The **Ruler feature** is used as a visual measuring tool inside the game area. Since this builder depends on writing movement commands, the ruler helps the student understand the distance and direction between the character and the target before writing the solution. It draws a line from the character's position toward the goal or selected point and displays the measured distance in tiles, such as "10 tiles." This makes it easier for the learner to translate the visual path into code commands, for example deciding how many step commands are needed and when the character should turn or change direction. The feature supports path planning and algorithmic thinking, because the student must analyze the map before writing the command sequence. It also works well with the builder's solution-generation and turn-angle options, since the measured distance and direction help produce or understand a more accurate movement path.

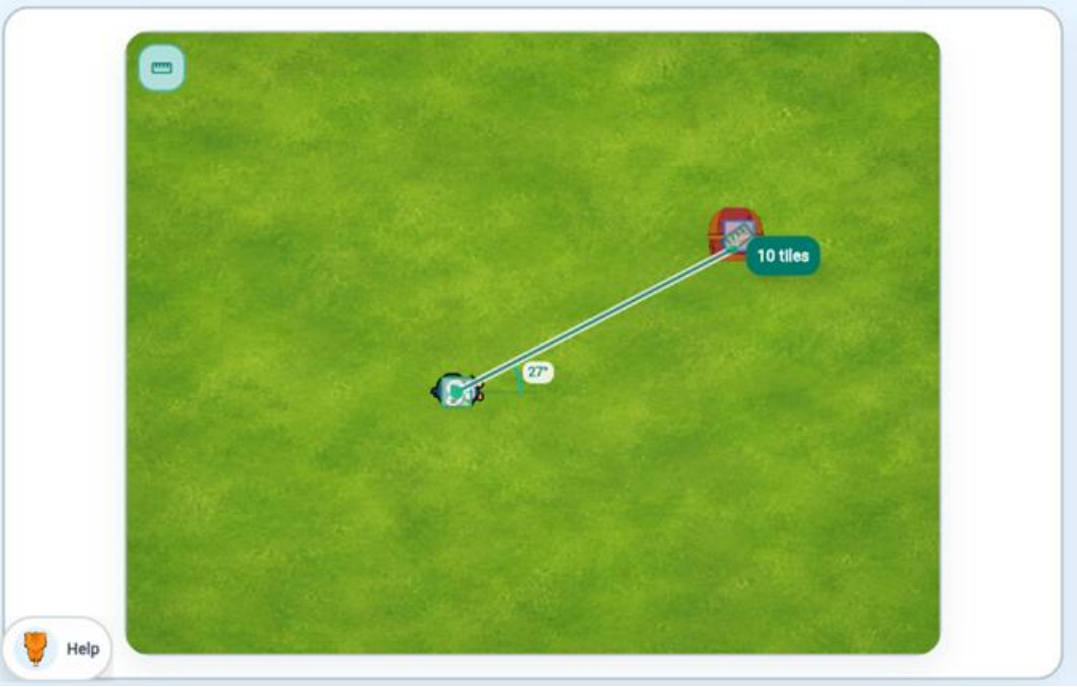


Figure 4.16.7: Path visualization helps the student estimate direction and distance, such as the displayed 10-tile path.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

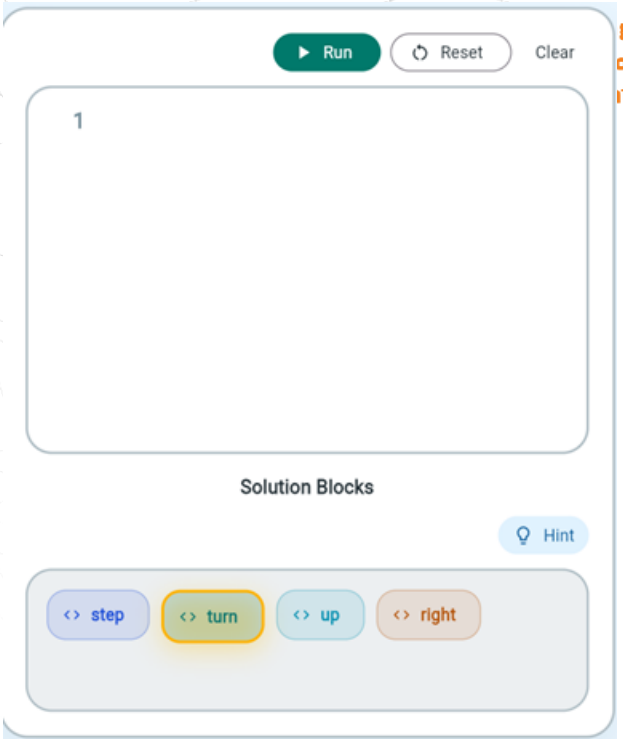


Figure 4.16.8: The Code Sequence editor includes Run, Reset, Clear, line numbers, solution blocks, and a hint button.

In this view, the student begins moving from block-based logic to written programming. For example, a solution may include commands such as step 9, turn right, step 5, or turn up. The system executes the commands in order and updates the character inside the top-view scene.

This view develops algorithmic thinking because the student must decide the path before writing the solution. They must determine how many steps are required, which turns are needed, and whether the character must collect an item or avoid an obstacle. If the written commands are wrong, the character moves incorrectly, and the student can edit the code and try again.

When the solution is correct, the completion popup appears with the success message, stars, and score. This confirms that the written commands produced the expected result and allows the student to either replay the level or continue.

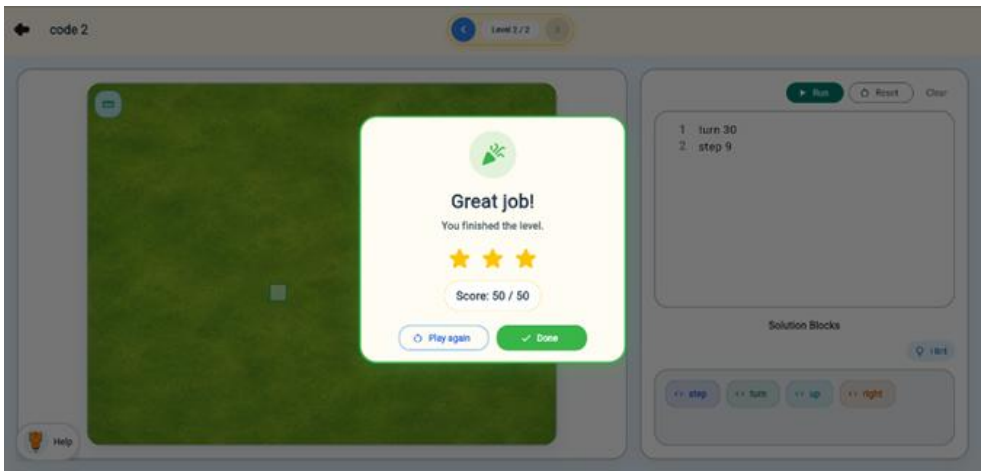


Figure 4.16.9: The Code Sequence completion popup confirms success and displays the score and stars.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.16.3 Game Builder View

The Game Builder View is the most advanced of the three views. It gives students a more complete programming environment where they can write custom code, run it, and see the result in a live game preview. This view is not limited to arranging a short sequence; it introduces students to broader game logic and event-based behavior.

The screen is divided into three main working areas:

- **Instruction panel:** The left side contains the exercise overview, code example, and detailed instructions.
- **Code editor and command area:** The middle area allows the student to write code and use command categories such as Movement, Events, Display, Control, and Operators.
- **Live game preview and assets:** The right side shows the game scene and includes tabs for Sprites, Widgets, Sounds, and Game settings.

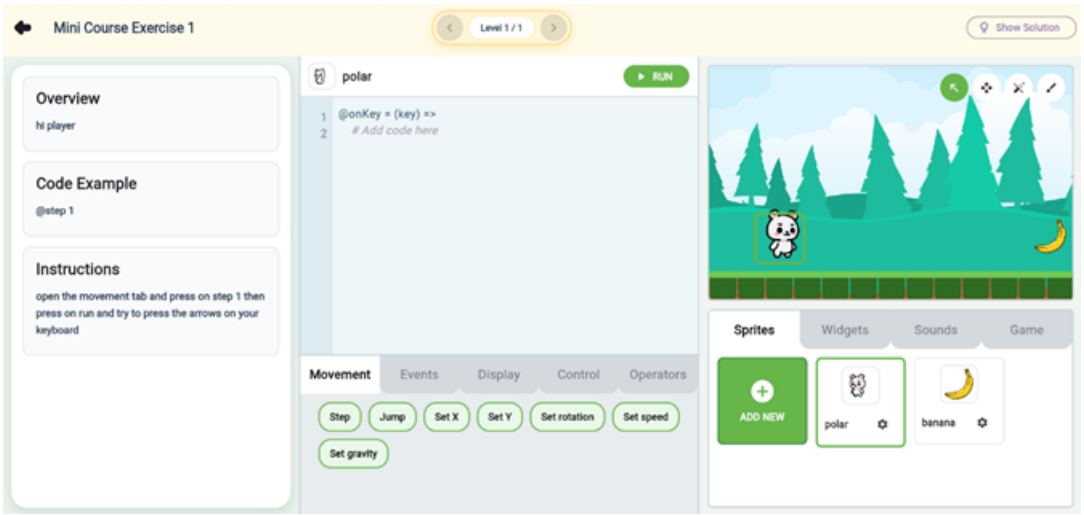


Figure 4.16.10: The Game Builder View includes the instruction panel, code editor, command categories, live preview, and asset tabs.

The instruction panel helps the learner understand the task before coding. It contains:

- **Overview:** A short description of the exercise or level context.
- **Code Example:** A small example that shows the command style expected in the exercise.
- **Instructions:** A direct explanation of what the student should do, such as adding a movement command and testing it using the Run button or keyboard input.

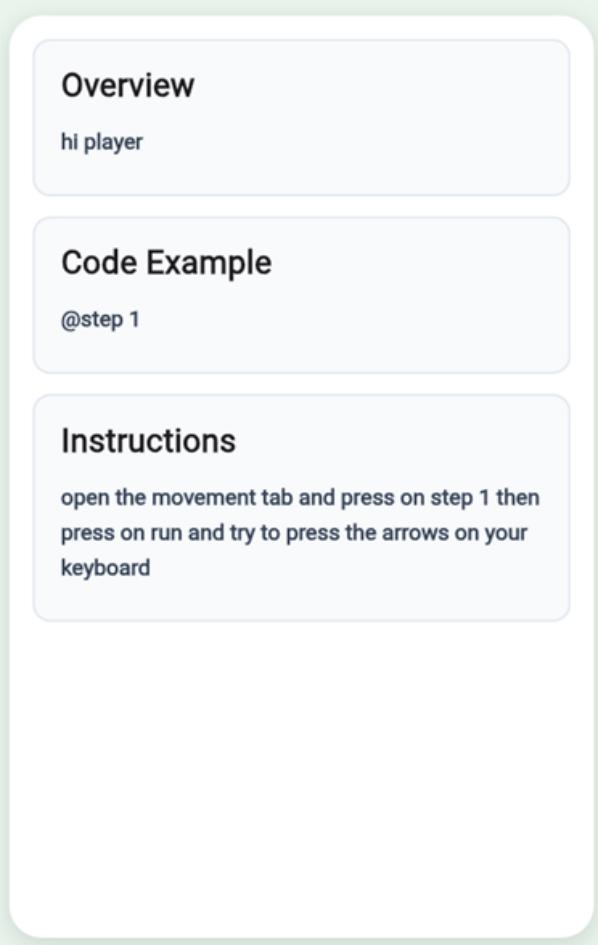


Figure 4.16.11: The instruction panel provides the overview, code example, and task instructions.

The command categories make the editor easier to use because they organize available commands instead of forcing the student to memorize everything. The Movement category includes commands for actions such as stepping, jumping, setting position, setting rotation, setting speed, and setting gravity. Other categories support events, display behavior, control flow, and operators.

The Game Builder also includes a solution support feature. When the student opens the solution dialog, the expected code is displayed. This is useful when the student is stuck, wants to compare their answer with the correct one, or needs to understand the correct syntax structure.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

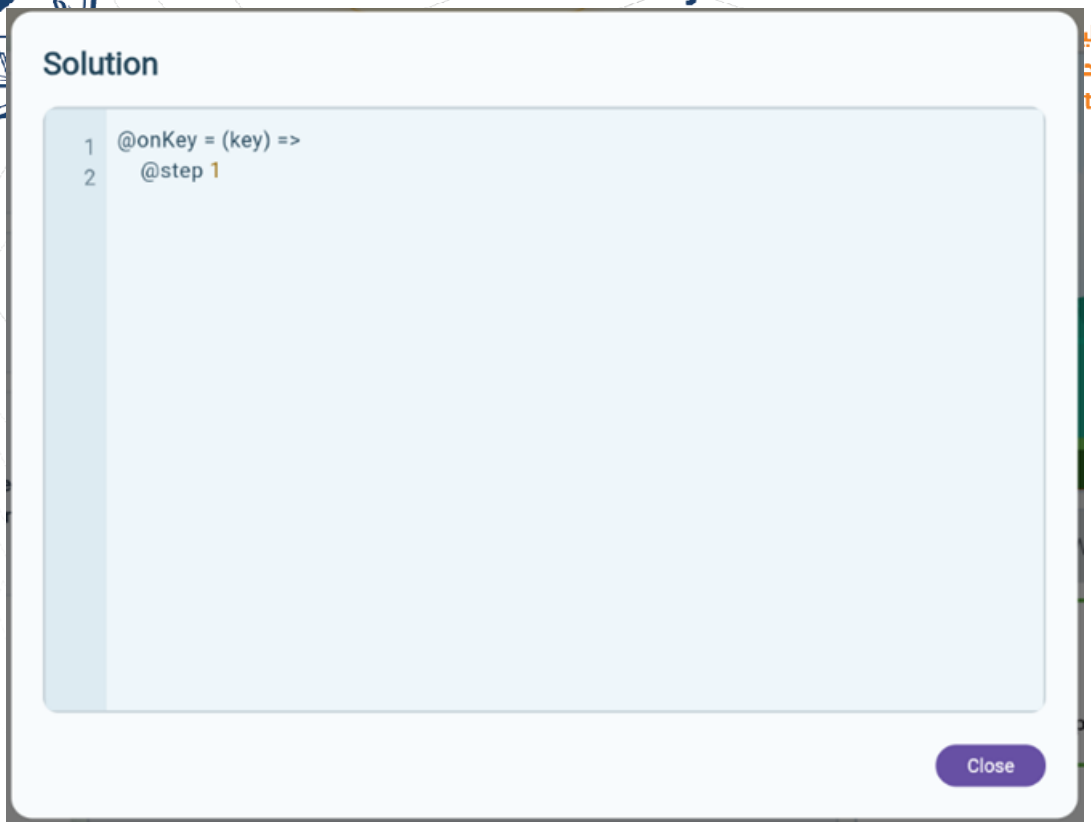


Figure 4.16.12: The solution dialog shows the expected code structure for the exercise.

The live preview is an important part of this view. When the student presses Run, the code is executed and the character or objects respond inside the preview area. This allows the learner to test an idea immediately and then edit the code if the result is not correct.

The asset tabs expand the builder from a coding exercise into a creative game environment. Students can manage sprites, widgets, sounds, and general game settings. For example, a sprite can represent a character or object, a widget can display information such as a timer or counter, and a sound can be triggered by game logic.

Unlike the sequence games, this view focuses more on code behavior, live testing, and solution support than on star scoring. It is used as an advanced learning stage where students understand how code controls objects, responds to events, and creates interaction inside a game.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

4.16.4 Testing, Feedback, and Learning Progression

Testing is a central part of the three game views. The student builds a solution, runs it, observes the result, and then improves it if necessary. This repeated process teaches debugging naturally because mistakes are treated as part of the learning cycle.

The progression between the views is intentional:

1. **Block Sequence Game View:** Introduces programming through visual sequencing in a front-view scene.
2. **Code Sequence Game View:** Introduces written commands and path planning in a top-view scene.
3. **Game Builder View:** Introduces custom code, event handling, command categories, assets, and live game behavior.

Together, these views allow Codey to teach programming gradually. The student first learns that commands must be ordered correctly, then learns to express those commands as text, and finally learns how code can control a full interactive game environment. This makes the learning process more accessible, practical, and engaging for children.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Chapter 5: Results and Discussion

The Codey platform was successfully designed, implemented, and tested as a full-stack educational system. This chapter discusses the outcomes achieved relative to the objectives stated in Chapter 1.

Platform Completeness

All major features described in the methodology were implemented and functional. The platform delivers four built-in courses (Digital Literacy, Data is Everywhere, AI is a Hoot, and Codey Jr.), a visual course builder with AI-assisted content generation, four game creation environments, a peer classroom system, a parent monitoring dashboard, and a full admin panel. The system operates across the Flutter mobile client and the Node.js/MongoDB backend with no critical missing components.

Gamification and Engagement

The gamification layer stars, points, streaks, leaderboards, head-to-head challenges, and weekly classroom goals was integrated across all courses and builders. The My Scores panel, classroom Ranks page, and activity feed provide students with continuous feedback on their progress and standing relative to peers, directly addressing the engagement gap identified in the problem statement.

AI-Assisted Content Generation

The Groq API integration (Llama 3.1) was successfully used to generate quiz questions, word search grids, fill-in-the-blank sentences, word matching pairs, and sorting activities from educator-supplied lesson text. All generated content is presented for educator review before publication, addressing the hallucination and bias concerns noted in the literature review. In practice, the AI generation feature significantly reduced the time required to author a complete interactive lesson.

Hands-On AI Education

The AI is a Hoot course delivers a genuine hands-on machine learning experience. Students record their own body-pose samples using a webcam, train a gesture classification model in the browser, test it with real-time confidence feedback, and wire the trained model to game actions using the On Prediction block. This is a meaningful departure from theoretical AI education and was fully functional in testing.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Bilingual Support

Full Arabic and English bilingual support was implemented across all screens, including right-to-left layout mirroring for Arabic. The language can be switched at any time from the profile page or the dashboard dropdown without requiring a restart.

Constraints Encountered

Several constraints identified in Chapter 2 had a visible impact on development. The Flutter learning curve slowed early progress but was overcome through self-study. Groq API rate limits were encountered during load testing and required request spacing on the frontend. MongoDB schema flexibility required careful manual discipline to maintain consistency across the course, builder, and game progress models. Free-tier hosting limited scalability testing to small concurrent user loads.

Comparison with Related Platforms

Relative to the platforms reviewed in Chapter 3, Codey provides a more complete feature set for its target audience. Unlike Scratch, it adds structured courses, AI-generated content, and a classroom system. Unlike Code.org, it supports bilingual content, custom course authoring, and creative game builders. The result is a platform that addresses the specific gaps identified in the literature for Arabic-speaking young learners.

University
Nablus, Palestine - Tel: +972(9)-2345113 - EX. 88-2223 - Fax: +972(9)-2345982 - E: info@najah.edu
department
equ@najah.edu
ANajahUni
https:// www.najah.edu

Chapter 6: Conclusions and Recommendations

Conclusions

Codey demonstrates that a comprehensive, bilingual, AI-assisted, and gamified coding education platform can be built by a small team using modern open-source tools. The platform successfully meets all seven objectives stated in Chapter 1: it provides a gamified learning environment for children, enables educators to build custom interactive courses, leverages AI for automated content generation, supports peer-based classroom features, gives parents a monitoring interface, offers full bilingual support, and provides administrators with management and analytics tools.

The AI is a Hoot course in particular stands out as a novel contribution: it allows children to train and apply their own gesture recognition model as part of a structured exercise sequence, making machine learning concepts tangible and accessible to young learners without requiring any prior technical background.

The game builder suite from the beginner Block Sequence Builder to the advanced Game Builder with its custom programming language provides a meaningful creative outlet that reinforces the programming concepts taught in the structured courses. The ability to publish games to the Discover page and receive ratings, comments, and play counts from peers adds a community dimension that motivates continued engagement.

Recommendations for Future Work

Several directions could extend and improve the platform:

- **Scalability**: Migrating the backend to a managed cloud service and adding caching and load balancing would allow the platform to support larger concurrent user bases, which free-tier hosting currently prevents.
- **Additional AI Model Types**: The current AI course supports pose classification only. Future versions could extend the AI - Pose system to support sound classification or image classification, broadening the range of learnable AI concepts.
- **Adaptive Learning Paths**: The current course recommendation is based on activity history. A more sophisticated recommendation engine that adapts to a student's demonstrated strengths and weaknesses could further personalize the learning journey.
- **Expanded Course Catalog**: The four built-in courses cover digital literacy, data, AI, and sequencing. Future courses could address topics such as web development basics, cybersecurity for children, or introductory robotics concepts.

References

Blom, N., Boerma, T., Bosma, E., Cornips, L., & Everaert, E. (2017). Cross-linguistic influence affects bilingual children's syntactic choices in both languages. Linguistic Approaches to Bilingualism, 7(1), 63–84.

Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: Defining gamification. In Proceedings of the 15th International Academic MindTrek Conference (pp. 9–15). ACM.

Hamari, J., Koivisto, J., & Sarsa, H. (2014). Does gamification work? A literature review of empirical studies on gamification. In Proceedings of the 47th Hawaii International Conference on System Sciences (pp. 3025–3034). IEEE.

Kasneci, E., Seßler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., & Kasneci, G. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. Learning and Individual Differences, 103, 102274.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., & Kafai, Y. (2009). Scratch: Programming for all. Communications of the ACM, 52(11), 60–67.

MongoDB, Inc. (2024). Mongoose ODM documentation. Retrieved from https://mongoosejs.com/docs/

Flutter Team. (2024). Flutter documentation. Retrieved from https://docs.flutter.dev/

Groq Inc. (2024). Groq API documentation. Retrieved from https://console.groq.com/docs/