Republic of the Philippines
BATANGAS STATE UNIVERSITY
The National Engineering University
Alangilan Campus
Golden Country Homes, Alangilan Batangas City, Batangas, Philippines 4200
Tel Nos.: (+63 43) 425-0139 local 2847

BeeHive: A Programming Learning Mobile Application for Inclusive Learning

Submitted by:

Sandara B. Bajio

Genesis Jim C. Cuasay

Karylle L. De Los Reyes

Godwin Miles G. Ricafort

John Benedict A. Tolentino

Submitted to:

Mr. Sean Mark M. Segui

Course Instructor

IT 331 Application Development and Emerging Technologies

Rationale

Learning to code is a fundamental and increasingly essential skill in today's digital age, which is the core motivation for this application. For many aspiring programmers, particularly students and those in resource-limited settings, acquiring coding skills is not a straightforward educational pursuit but a frustrating challenge riddled with high costs and pedagogical barriers. Even with a wealth of online information, knowing how to start, what to learn next, and how to practice effectively is often unclear, resulting in abandoned learning journeys or reliance on prohibitively expensive bootcamps and platforms.

At the same time, learners want to build proficiency but struggle with maintaining motivation, overcoming abstract concepts, and tracking their progress in a meaningful way. This gap adds a significant layer of difficulty to the task of self-education, preventing many from aligning their learning with their career goals. This project BeeHive accomplishes and ties together both problems directly: by offering a built-in, gamified learning environment, a structured curriculum with hands-on coding exercises, and a progress-tracking system that fosters community and competition.

BeeHive is not just another online course but a comprehensive tool that empowers users to build programming skills through consistent, engaging, and affordable practice. It elevates the learning approach from passive video consumption to an active, integrated experience that combines theory, immediate practice, and motivational rewards, thereby reducing the need to juggle multiple disconnected apps and platforms.

BeeHive fulfills educational challenges by not only optimizing knowledge retention but also increasing accessibility, and by creating a supportive learning community, all in one service. It is also a contributor to broader educational initiatives aimed at democratizing digital skills, reducing the financial barriers to tech education, and fostering a more inclusive and skilled future workforce.

Introduction

In today's fast-paced world, programming has become an essential skill for many industries and careers. It is now vital for business, healthcare, engineering, and education as it is the foundation of innovation in providing better services that help sustain the growing needs of the future.

As the demand for coding proficiency continues to grow, young learners turn to online platforms in order to acquire sufficient skills because of its accessibility and wide range of resources. While many platforms provide interactive lessons, built-in coding environments, and gamified learning, access to their full features often comes at a significant cost. This makes it difficult for students and institutions in resource-limited settings to gain equal access to quality coding education. In effect, while opportunities exist, affordability and scalability remain real barriers.

Moreover, many proprietary platforms further restrict access by locking advanced features behind high subscription fees or licensing costs, limiting their reach to only those who can afford them. Coding bootcamps and structured training programs often charge tuition fees ranging from ₱80,000 to ₱100,000 for three (3) to six (6) months of study, making them inaccessible to many aspiring programmers and institutions with limited budgets [1]. Even the more affordable options, priced around ₱20,000 to ₱50,000, typically offer fewer features or limited mentorship support [2]. Such costs present a substantial barrier, preventing widespread access to quality programming education in the country and highlighting the need for a free, scalable alternative.

This app, BeeHive, seeks to respond to this need by providing a free coding education platform that institutions and organizations can adopt to deliver programming instruction at scale. The platform integrates hands-on coding practice, real-time feedback, structured mentorship, and progress tracking—all without the financial barriers imposed by most existing platforms. For learners, this ensures equal access to high-quality programming education. For mentors, it provides

centralized tools to guide, monitor, and support students efficiently. For institutions, it offers a scalable, cost-free, and standardized training solution, making coding education more inclusive and sustainable. For its prototype testing, the project specifically targets Batangas State University - The National Engineering University, Alangilan Campus as the initial institution for implementation.

Project Objectives

To address the challenges of costly platforms and limited access, this initiative introduces a scalable mobile learning solution. By leveraging Flutter's cross-platform capability and research-backed gamification strategies, the project seeks to create an engaging, equitable environment that supports hands-on programming practice and sustained learner motivation across diverse educational settings.

This project aims to:

1. Develop a cross-platform mobile application using Dart and Flutter to teach programming concepts to beginner and intermediate learners;
2. Implement gamification features—including streaks, badges, points, and leaderboards—to enhance learner motivation and engagement;
3. Provide real-time progress tracking to enable adaptive learning paths and personalized feedback;
4. Support offline access to lessons and exercises to ensure uninterrupted learning;
5. Integrate mentor/instructor support for guidance and troubleshooting;
6. Evaluate the effectiveness of the gamified interactive approach in improving programming knowledge retention and learner satisfaction.
7. Offer a completely free platform as an alternative to premium programming learning apps and websites, ensuring that learners, instructors, and institutions—especially those in resource-limited settings—can access quality education without financial challenges.

Proposed Features and Functionalities

The application is engineered with a comprehensive suite of features designed to provide learners and instructors with an effective, engaging, and scalable programming education environment. These features aim to simplify the learning process, promote consistent practice, and support measurable skill development through structured and interactive content. The following proposed features and their corresponding functionalities are organized by user role and described below:

Learner Features

- In-App Integrated Development Environment (IDE) - This feature provides a fully functional coding workspace embedded within the application, enabling learners to write, execute, and debug code directly without needing to switch to an external program. It offers syntax highlighting and real-time feedback, facilitating a hands-on, immersive learning experience that reinforces theoretical concepts through immediate practice.
- Comprehensive Progress Tracking System - This system comprises multiple elements designed to visualize achievement and maintain learner motivation:
    - Achievement Badges: Digital rewards are awarded upon the completion of specific milestones or the mastery of particular concepts, providing tangible recognition of skill advancement.
    - Lesson Progress Bar: A visual indicator displays the completion status of modules and lessons, allowing learners to track their overall journey through the curriculum.
    - Daily Streak Counter: This mechanism encourages consistent study habits by tracking and incentivizing daily engagement with the platform.
- Interactive Lesson Formats - Lessons are delivered through a variety of interactive activity types to accommodate diverse learning styles and reinforce comprehension. These include matching exercises, fill-in-the-blanks tasks for

vocabulary and syntax acquisition, and direct programming challenges that require applying concepts to solve problems.

Instructor Features

- Lesson Management Tools - This suite of functionalities empowers instructors to create and maintain a relevant and up-to-date curriculum:
    - Lesson Creation: Enables the authoring and integration of new lessons and modules into the learning pathway.
    - Content Updates: Allows for the modification and improvement of existing lesson materials to correct errors or incorporate new information.
    - Content Deletion: Provides the ability to remove outdated, redundant, or incorrect content to ensure curricular integrity.
- Learner Management Tools - These tools facilitate administrative control over classroom composition and access:
    - Learner Enrollment: Permits instructors to manually add students to a specific class or cohort within the platform.
    - Learner Removal: Allows for the management of class rolls by removing students as necessary.
    - Progress Oversight Dashboard: Provides instructors with a centralized view of aggregate and individual learner progress, including metrics on lesson completion, assessment scores, and activity streaks. This enables data-driven instruction and timely intervention for struggling students.

These features are proposed in direct response to identified challenges in programming education. For learners, the integrated IDE, gamified progress tracking, and diverse interactive lessons are designed to lower accessibility barriers, sustain motivation, and deepen engagement. For instructors, the comprehensive management and oversight tools ensure that educational content remains current,

instruction can be adapted based on real-time analytics, and learner needs can be efficiently identified and addressed.

Project Timeline

Table 1. Project Timeline of BeeHive

Date	Activity	Output
Sept 15 - Sept 25, 2025	**UI/UX Planning & Design**	Create app mockups (Learner + Instructor dashboards).
		Decide branding (bee/hive theme, colors, flow).
Sept. 26– Nov 10, 2025	**Development Phase**	Build main features: • Learner side (IDE, progress tracking, lessons). • Instructor side (lesson + learner management).
		Add gamification (badges, streaks, progress bar).
Nov 11 – Nov 24, 2025	**Testing & Refinement**	Try the app with classmates/peers.
		Debug and polish user experience.
Nov 25 – Dec 6, 2025	**Finalization**	Lock in features, finalize design.
		Prepare documentation
		Ensure the app is ready by **Dec 1**.
Dec 7 – Dec	**Presentation &**	Present BeeHive app.

12, 2025	**Evaluation**	Showcase learner + instructor features.

The timeline presents the step-by-step development of the BeeHive Programming Learning App, starting from UI/UX design, followed by development, testing, and finalization. The project is targeted for completion before December 1, allowing time for preparation. The anticipated presentation or defense is estimated to fall within December 7–12, hence the final phase accounts for preparation toward that period.

References

[1] Eskwelabs, “Short and Sweet Bootcamps as a Faster and More Meaningful Educational Alternative for Career Growth,” Eskwelabs, Oct. 2023. [Online]. Available: https://www.eskwelabs.com/blog/short-and-sweet-bootcamps-as-a-faster-and-more-meaningful-educational-alternative-for-career-growth

[2] Nucamp, “Top 5 Most Affordable Coding Bootcamps in Philippines in 2025,” Nucamp, Jan. 2025. [Online]. Available: https://www.nucamp.co/blog/coding-bootcamp-philippines-phl-top-5-most-affordable-coding-bootcamps-in-philippines-in-2025