[1 pt] Project Description

Our project, **Piggy-Pon**, is a student-friendly budget tracking web application designed to encourage and gamify daily saving habits for students. Through interviews, observations, and an initial beta test, we found that many students struggle with managing allowances and setting realistic savings goals due to lack of digital tools tailored to their needs. Piggy-Pon addresses this gap by offering a visually appealing, intuitive platform accessible through any web browser. Core features include digital coin logging, goal setting with progress tracking, and weekly/monthly summaries. The system is designed with accessibility in mind—ensuring usability for students across various year levels and technical experience, from digital natives to less tech-savvy users.

[4 pts] Requirements Summary

Piggy-Pon is optimized for cross-browser functionality and is lightweight enough to run even on low-end devices, making it accessible to students using outdated school-issued laptops or budget smartphones via mobile browsers.

Table 1. System Requirements for Piggy-Pon (Android and iOS)

Category	**Minimum Requirements**	**Recommended Requirements**
OS	Windows 7+, macOS 10.12+, Android 8+	Windows 10+, macOS 11+, Android 10+
Browser	2 GB	4 GB
Processor	Chrome 70+, Firefox 65+, Edge 80+	Chrome 100+, Firefox 95+, Safari 14+
Connectivity	3G/Wi-Fi	4G LTE/ Wi-Fi
Screen Size	5” or 1024x600	6”+ or 1366x768+
Permissions	Local Storage, Notifications (optional)	Local Storage, Notifications, Cookies

The web app is front-end-heavy, with minimal server-side processing to reduce hosting costs and improve speed. It’s built for simplicity, ensuring quick loading times even under limited internet bandwidth, perfect for campus Wi-Fi or prepaid data users.

[75 pts] Prototype Description

Overview of the Prototype

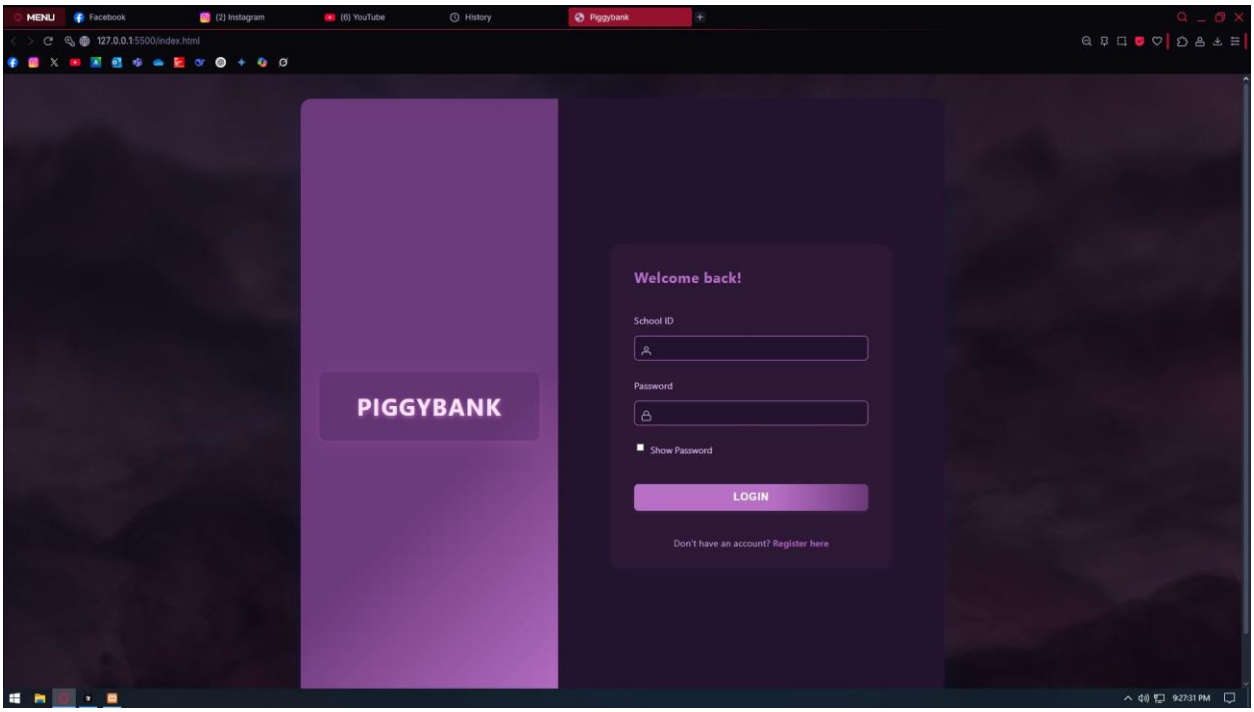
The Piggy-Pon prototype was developed using **Figma** and **HTML/CSS mockups** to simulate core saving and budgeting interactions. The design mimics the final interface experience, allowing testers to preview and engage with essential features such as:

- Logging daily savings in digital coins
- Setting personal financial goals (e.g., ₱500 for a new headset)
- Visualizing progress with percentage bars and charts
- Viewing weekly/monthly breakdowns of savings
- Reviewing transaction logs
- Resetting goals and starting new savings challenges

While the full backend system (e.g., MySQL data persistence and user login sessions) is still under development, the prototype offers realistic interaction flows and UI logic to gather feedback and validate usability among student testers.

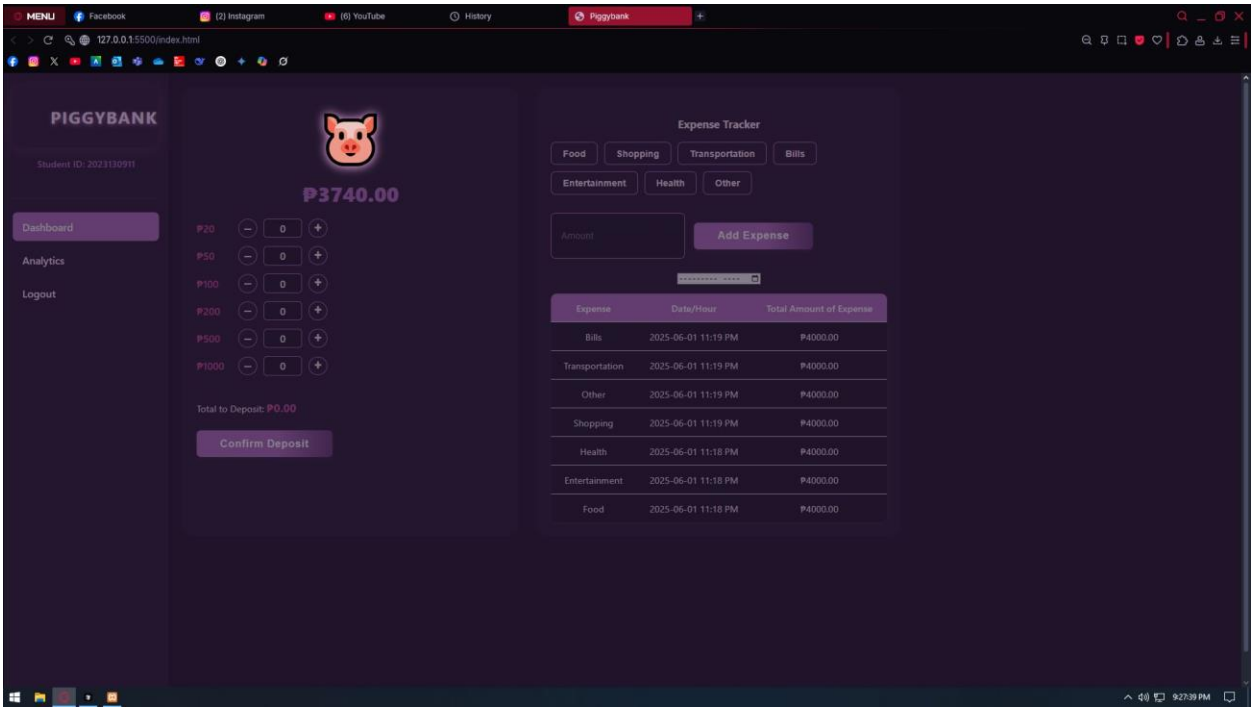
Description of Each Screen

Login / Sign-Up Page



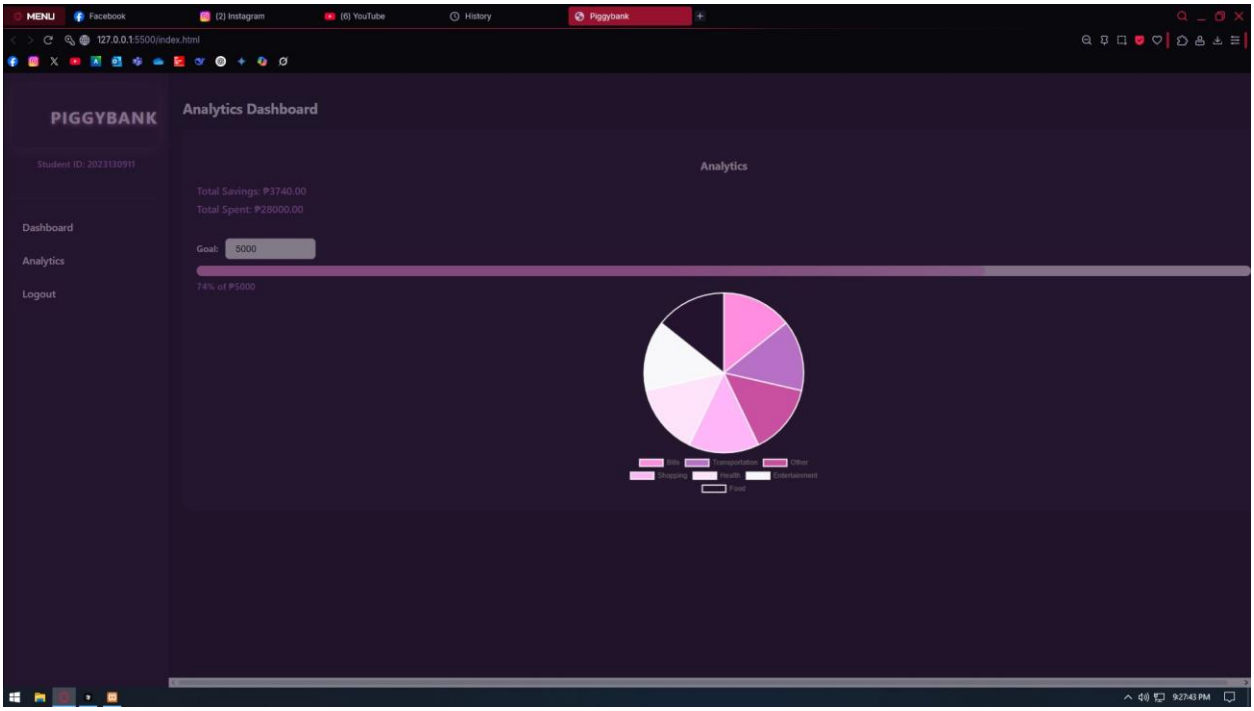
The login interface uses a card-style layout with a modern floating container. At the top sits the Piggy-Pon icon, reinforcing brand identity. Input fields for email and password are large and center-aligned, with intuitive coin and lock icons to guide entry. A “Forgot Password?” link is available below the form, and a toggle below it lets users switch to **Sign Up** mode seamlessly. Soft hover effects and shadowing provide a clean, interactive feel that builds trust and ease of use. The login serves as the welcome screen as well.

Dashboard / Home



Upon logging in, users land on the dashboard—a summary hub that displays their **current savings** and **active goal** while charts show recent weekly activity. Users can also quickly tap **"Add Savings"**, **"Set Goal"**, or **"View History."** The dashboard lets users log savings easily. A list menu lets them organize savings entries (e.g., "Allowance," "Gift," "Extra Income," etc.). The "Add Expense" button provides real-time confirmation. The layout uses warm green and yellow hues to associate positivity with saving behavior. In this screen you can also define a specific goal with fields for **goal name**, **target amount**, and **optional deadline**. A visual meter preview updates dynamically as inputs change. The dashboard uses cards, icons, and soft animations to enhance clarity and engagement. On the bottom right, there's a passbook-style card where users can review all past transactions. Each entry includes the **amount**, **date**, **note**, and **category**. Users can filter by day, week, month, or custom date range.

Analytics Screen



This screen gives users data-driven insights into their saving patterns. A **bar chart** shows daily/weekly savings. A **donut chart** visualizes current goal progress. Below, a section displays stats like **total saved**, **average per week**, and **completion rate**. It's minimal yet functional, built for both data lovers and casual users.

Scenario from User's Perspective

At 8:00 PM, Zuriel just got home after a long day of classes. He pulls out his phone to check how much allowance he has left. Opening the **Piggy-Pon** web app, he sees a clean dashboard showing he has saved ₱180 out of his ₱500 goal. He remembers he skipped milk tea that day, so he quickly logs the ₱50 he didn't spend. The progress bar jumps to ₱230. Curious about how he's been doing this week, he taps on the **Analytics** tab and views a bar graph of his daily savings. Impressed with himself, he sets a new micro-goal called "Weekend Movie Fund" for ₱150 and returns to the dashboard with a satisfied smile. All of this happens in under three minutes—with zero stress, and maximum feel-good energy.

Rationale: Why This Prototype?

For the Concept:

We focused our project on building **Piggy-Pon**, a web-based budgeting app tailored specifically for students. Many students struggle with managing their daily or weekly allowance, often due to a lack of accessible tools that make saving feel intuitive, engaging, and rewarding. Traditional budget apps tend to be complex, overwhelming, or designed for working adults with credit cards and investments—not helpful when you're just trying to save ₱20 from your baon.

Piggy-Pon addresses these gaps with a **gamified, goal-oriented, and student-friendly interface**. Our goal was to make saving money feel like leveling up in a game, while still retaining serious functionality like transaction history, analytics, and streak tracking. The app focuses on micro-savings, progress visualization, and ease-of-use—key elements we identified as essential through user interviews and HCI design principles.

By tackling a universal yet often ignored student problem—daily financial discipline—Piggy-Pon empowers users to save smarter with zero overwhelm.

For the Design:

This prototype was selected from several concept iterations because it achieved the best balance between **usability, engagement, and practicality**. It includes a focused dashboard, responsive design, minimal user input steps, and optional advanced features like savings analytics.

Feedback from testers emphasized that they wanted something "simple," "motivating," and "non-boring." These insights guided our decision to:

- Use date/time in transaction logging for practicality
- Display savings progress visually
- Limit steps to key actions like "Add Savings" or "Set Goal" in 1–2 clicks

It was also crucial that our interface worked just as well on a mobile browser as on a desktop or tablet, making it flexible for student users across devices.

Advantages:

- Offers an all-in-one, centralized view of everything a user needs to know or do.
- Focuses on day-to-day budget management with relatable UX language.
- Incorporates HCI best practices such as recognition over recall, consistency, feedback, and error prevention.
- Mobile-friendly, browser-based access: No need to download an app; just open in a browser and go.
- Minimal steps, strong feedback: Every major action provides instant confirmation, supporting user trust.

Disadvantages:

- Backend dependency for analytics and login validation: Some features require a reliable server and database setup, which might be limited in offline or slow-connection scenarios.
- Users new to mobile applications may initially require onboarding or visual aids to navigate all available tools.
- New users may require a quick onboarding guide: First-time users might not immediately understand streaks or goal setting without a walkthrough.

Despite these limitations, this prototype is ideal for usability evaluation, showcasing clear and testable flows for core features like adding savings, setting goals, and reviewing personal analytics.

For the prototyping tool:

We chose to use **pen and paper** as our primary design and prototyping tool due to its simplicity, speed, and accessibility. Creating the **Piggy-Pon** prototype by hand allowed our team to quickly sketch and iterate on key interface components such as:

- The home dashboard
- Add Savings interface
- Goal-setting modal
- Analytics screen with charts
- History and transaction logs

Paper prototyping enabled real-time collaboration without needing devices or internet access. Team members could freely contribute ideas and adjust screens instantly during brainstorming and planning sessions. This approach also made it easier for participants to focus on the structure and usability of the design, rather than being influenced by animations or visual polish.

Limitation: Paper prototypes are static, so user interactions (like tapping or scrolling) had to be simulated manually. However, for testing layout, flow, and navigation logic, pen-and-paper prototyping provided an effective and low-cost solution. It allowed us to gather unbiased feedback and validate our core ideas before moving to digital development.

Changes to Requirements

During the design and prototyping phase, we discovered areas where our initial web-based app requirements needed refinement. These changes reflect feedback from early testers and align with best practices for web application usability.

Table 2. Revised Requirements and Justifications

Original Requirement	Revised Requirement	Reason for Change
Manual Savings Entry Only	Manual Entry + Auto-Saving Suggestions	This adds convenience and engagement.
Goal Tracking (Monthly Only)	Flexible Goal Duration (Daily, Weekly, Monthly)	Users preferred short-term goals (e.g., weekly movie budget) in addition to long-term savings. Improves motivation.
Static Progress Bar	Animated Charts with Streak Indicators	A static bar felt uninformative. We added dynamic charts with savings streaks and celebration animations to encourage consistency.
Add/Withdraw Buttons Only	Add/Withdraw with Optional Notes and Icons	Users found traditional long forms overwhelming. A multi-step form that asks one question per screen was preferred, reducing Notes like "Saved tricycle fare" made the app feel more personal and emotionally rewarding. and helping users feel less intimidated.

[20 pts] Initial Evaluation Plan

This phase focuses on evaluating the **usability, effectiveness, and user experience** of Piggy-Pon's web-based prototype. Testing will occur through online platforms, simulating everyday interactions with a personal finance tool accessed via browser.

Usability Specifications

Our evaluation will be anchored on these 7 usability goals:

Table 3. Usability Criteria and Evaluation Methods

Criteria	How We Will Evaluate It

Accessibility	Evaluated by how easily users across different demographics (including low-tech users) could access and navigate the platform. We observed interaction patterns and collected user ratings on perceived accessibility.
Error Handling	Users were tasked to simulate common mistakes (e.g., invalid input, duplicate entries). We evaluated whether the system provided helpful feedback and prevented errors from escalating.
Functionality	Assessed by users' ability to complete core tasks such as adding savings, setting goals, and viewing progress. High scores reflected success in executing these actions without issues.
Installation	Since Piggy-Pon is purely web-based, we measured installation through ease of access — i.e., no app downloads required. User ratings reflected the simplicity of opening the app via browser.
Support	We evaluated user perception of available help, such as instructions, tooltips, or guidance. The lack of interactive or real-time support affected this score.
Usability	Assessed overall ease-of-use, navigation, and task flow. We checked if users could perform actions with minimal instruction and little confusion.
Other	Captured miscellaneous comments and ratings on visual design, dark mode preference, or additional feature suggestions. These helped us understand unanticipated user expectations.

Test Participants (Target Population)

We will invite **10–15 participants** that reflect our app's target users:

- College students learning to manage daily allowance
- Young adults practicing basic budgeting
- Young adults practicing basic budgeting

These participants will interact with our clickable prototype, either in person or virtually, and will be guided to complete benchmark tasks.

Prototype Tasks (Benchmark Tasks)

Participants will be asked to complete the following tasks during the test session. Each task is designed to test specific parts of the interface and ensure real-life relevance.

Table 4. Benchmark Tasks and Corresponding Evaluation Goals

Task	Goal
1. Open the app in a browser and view their dashboard	Test quick access to core info
2. Log any amount for savings	Test manual input flow and customization
3. Set a "Jollibee Weekend" savings goal of ₱300	Assess intuitiveness of goal creation
View progress chart	Evaluate data visualization and user motivation

Each task will be timed and observed. If tasks take too long or cause confusion, it will indicate a usability issue.

Roles of Team Members

Table 5. Team Member Roles During Testing

Team Member	Role
Zuriel Magtibay	Facilitator – guides participant, explains task
Harvey Quijada	Notetaker – observes participant behavior, logs errors
Kendrick Flores	Survey & Debrief – assists with feedback forms and final comments

Success Time Benchmarks (Time-Based Interpretation)

Table 6. Success Time Benchmarks (Task Time Thresholds)

Task Type	Target Time	Interpretation

Navigation tasks (e.g., View balance/goals)	≤ 30 seconds	Successful
Interactive tasks (e.g., Add savings, Edit goals)	≤ 1 minute	Acceptable
Complex tasks (e.g., Save + Track + Check history)	≤ 2 minutes	Acceptable
If > target time or errors occur	—	Review for redesign

Heuristic Evaluation (Nielsen's 10 Principles)

To further improve the Piggy-Pon prototype, we conducted a heuristic evaluation based on **Jakob Nielsen's 10 Usability Heuristics**. Below is how the web app aligns with each principle:

- **Visibility of System Status**

  Users receive instant confirmation after adding savings or reaching goals, including real-time updates on total balance and visual goal progress.

- **Match Between System and Real World**

  The app uses familiar metaphors like "piggybank," "goals," and "balance." Currency formats and labels match everyday usage (e.g., "₱50 added to 'New Shoes' goal").

- **User Control and Freedom**

  Users can easily undo or cancel actions such as editing a goal or deleting a savings entry, preventing accidental changes.

- **Consistency and Standards**

  Icons, typography, and color themes are used uniformly across all screens. Common elements like the "Add" and "Back" buttons maintain placement consistency.

- **Error Prevention**

  Input fields have safeguards to prevent adding invalid amounts (e.g., non-numeric input, zero or negative values).

- **Help Users Recognize and Recover from Errors**

  Clear messages like “Invalid amount” or “Goal name already exists” guide users to fix problems without frustration.

- **Recognition Rather Than Recall**
  Dashboard cards, labeled icons, and persistent nav elements reduce memory load. Frequent tasks (e.g., adding coins) are one tap away.

- **Flexibility and Efficiency of Use**
  The system supports both casual users and power users. Quick-add shortcuts and recently used goals streamline repeated actions.

- **Aesthetic and Minimalist Design**
  The layout emphasizes clarity. Progress rings, savings charts, and minimal copy keep focus on tasks without overwhelming visuals.

- **Help and Documentation**

  Users can access tooltips on hover/tap and view a basic FAQ section. An integrated support request form is planned for future versions.

Participant Survey and Feedback

After interacting with the Piggy-Pon prototype, participants will be asked to complete a short post-test survey designed to capture both measurable insights and open-ended reflections. This will guide the next phase of iteration by highlighting usability strengths and identifying problem areas.

Table 7. Participant Survey Methods

Method	Description
Survey (Quantitative)	5-point Likert scale to evaluate ease of use, satisfaction, clarity, and design appeal.
Open Feedback (Qualitative)	Short written responses to highlight what participants liked, struggled with, or wanted improved.

Survey Sample Questions

- On a scale of 1–5, how easy was it to navigate the Piggy-Pon interface?
- How would you rate the clarity of the savings goals and balance information?
- Was it straightforward to add savings or create a goal?
- Did the dashboard give you a helpful overview of your financial progress?
- Were there any features that were confusing or difficult to use?
- What is one feature you would like to see added or improved?

Evaluation Rubric (Success Thresholds)

Table 8. 5-Point Likert Scale Interpretation Guide

Scale	Interpretation
5	Highly Acceptable (Effortless experience)
4	Acceptable (Few minor issues)
3	Neutral (Some difficulties)
2	Needs Improvement
1	Not Acceptable (Confusing, error-prone)

If a feature scores below **3.5**, we will flag it for redesign or enhancement in part 3.2.