

# Quick Quacker: Sprint 2

## System Design

### Group 1:

- ☒ Phoebe Gamit
- ☒ Kristin Villena
- ☒ Michael David
- ☒ Amnol Misra
- ☒ Urgyen Kalsang

## Table of Contents

<b>Title Page</b>	<b>1</b>
<b>Table of Contents</b>	<b>2</b>
<b>CRC Cards</b>	
Duck	
Game Manager	
UI Manager	3
Storage Manager	
Task	
Task Manager	4
XP Manager	
Achievement Manager	
Calendar	
Customization Manager	5
<b>Software Architecture Diagram</b>	<b>6</b>

## CRC Cards

Duck	
<p>Maintain its name, gender, level, XP, hunger, thirst, and mood</p> <p>Respond to user actions like feeding, giving water, and playing</p> <p>Display animations and visual changes based on interactions.</p> <p>Level up when XP reaches a threshold</p> <p>Duck grows and evolves gaining a certain level.</p>	<p>GameManager (Handles interactions)</p> <p>UIManager* Updates the display)</p> <p>StorageManager ( Saves and loads duck data)</p> <p>CustomizationManger( changes appearance and look of duck)</p>

Game Manager	
<p>Control game logic( e.g. hunger and thirst decrease over time)</p> <p>Process user inputs (e.g., feeding, playing)</p> <p>Handle XP gain and level-up logic</p> <p>Unlock achievements</p>	<p>Duck (Updates the pet's stats)</p> <p>UIManger (Updates the UI)</p> <p>AchievementManger (Tracks Milestones)</p>

UI Manager	
<p>Render duck stats and animations</p> <p>Update XP, hunger, thirst, and level displays</p> <p>Manage settings and modal interactions</p> <p>Display achievements when unlocked</p> <p>Render task list</p> <p>Display filter options (All, Active, Completed)</p> <p>Provide an interface for adding/editing tasks.</p>	<p>GameManager (Receives updates)</p> <p>Duck (Displays status)</p> <p>TaskManager (Fetches and updates tasks)</p> <p>Task (Displays individual tasks)</p> <p>XPManager (Updates progress bar based on task completion)</p>

Update task stats and XP level.	
---------------------------------	--

Storage Manager	
Save and load duck stats using LocalStorage	GameManager( Handles save/load logic)
Retrieve stored progress when the game starts	UIMaanger (Loads settings on startup)
Store achievements and preferences	TaskManager (Manages tasks)
Saves and retrieve tasks from Local Storage	XPManger (Stores XP progress)
Persist user preferences and XP progress	
Load saved tasks when the app starts	
Store coins used to pay for cosmetics	
Stores cosmetics that user have already bought	

Task	
Store task details (title, priority, completion status, due date and time)	TaskManager (Manages task operations)
Toggle completion status	UIManger( Displays tasks)
Provide information for rendering in UI	Calendar(Updates the due date to calendar )

Task Manager	
Add, remove, and edit tasks	TaskManager (Manages task operations)
Toggle task completion	UIManager (Displays tasks)
Save and load tasks from LocalStorage	Task (Handles individual task data)
	UIManager( Updates UI based on task operations)
	StorageManager (Persists tasks)
	Calendar (Obtains dates as options)

XPManger	
Track XP level based on completed tasks Update and display XP progress bar. Unlock achievements.	TaskManager (Earns XP from task completion)  UIManager (Displays XP progress)  AchievementManager (Unlocks rewards)

Achievement Manager	
Track and unlock task-based achievements Display notifications for new achievements Store unlocked achievements.	XPManger (Triggers achievements based on XP milestones)  UIManager (Displays achievements)

Calendar	
Track due dates of tasks  Display a full calendar of the month of the year  Display indication of current day  Display tasks on days where their due date is placed on.	UIManager (Displays calendar)  TaskManager ( Gets a date to input)  Task ( adds due date for specified task to calendar)  Task Manager ( inputs a date for when task should be completed before)

Customization Manager	
Contains a shop feature where duck can have cosmetics ( hats, colour, clothes, etc.)  Coins are used to pay for cosmetics  Display wide range of cosmetics  Option to change appearance of Duck	UIManager ( Displays shop and cosmetics)  Duck ( Changes appearance of Duck)  Storage Manager ( stores bought cosmetics)

## Software Architecture Design

