Lesson 4.3: Independent Projects

PROBLEM **4.3.1**

Make an App

INTRODUCTION

The Make an App project brings together all of the skills you have learned this year. Using the software development cycle you have used for other projects, you will begin with Planning and Design, and end with Test and Delivery. In deciding what your app should do, choose from three categories:

- An app that solves a problem in the community.
- An app that tells a story.
- A game app.

You will brainstorm, identify a stakeholder group, gather input, and create backlogs and task lists. You will design your app, choose its features, and develop the app. Finally, you will deliver a presentation, first to your class and then to your stakeholder group.

Materials

- Computer with Android™ Studio
- Android™ tablet and USB cable, or device emulator
- *Optional* resources such as Backendless, Google Play services, LibGDX
- Tools with which to create visual representations of your code
- Tools for creating and using a sprint task list and product backlog
- Presentation software

RESOURCES

Problem 4.3.1 Requirements
Resources available online

Problem 4.3.1 Rubric - Common
Resources available online

Problem 4.3.1 Rubric - Android
Resources available online

Problem 4.3.1 Rubric - LibGDX
Resources available online

Problem 4.3.1 Rubric - Android One Page
Resources available online

Problem 4.3.1 Rubric - LibGDX One Page
Resources available online

Procedure

Part I: Plan

1. Read through **Problem 4.1.3 Requirements** then read this entire problem definition. Know all the requirements before you begin any work.

2. Decide whether you will create a new app or enhance an app from another lesson. Choose one of the following categories:

   - Identify a problem that exists for your school, you or your family, your friends, or your community.

     Consider data sources you may need to help define or treat the problem. You may use existing data you find on the Web, or you may simulate reasonable data.

   - Tell a story that has meaning to you and your team.

     Your story must be interactive in some way and have a logical and sensible plot. It should progress in different ways, depending on user interaction, and demonstrate variable behavior. For example, a series of fixed comic strips is not sufficient.

   - Develop a game.

     You can create a game using Android features, such as animation, or you can extend the Unit 3 game Emu on the Loose. Different from a story, the game should have a winnable goal, increasing difficulty, and user progress that is maintained and displayed.

3. Create a problem statement for your project.

4. Define a minimal set of requirements—what you want the app to do and how it will satisfy the problem statement.

One part of your project plan needs detail:

5. Decide on the set of features you want for your app.

   a. Include at least one new feature. In other words, a feature you have not implemented before. New features are discussed in Step 6.

   b. Also, include known features, meaning features you already know how to implement. Known features are discussed in Step 8.

6 Research **Android**, **Backendless**, **Google Services**, and **LibGDX** to discover new features or major sub-features you would like to have in your app. (If you completed the AP Unit 4 in place of the Android/LibGDX Unit 4, do not choose the LibGDX option.)

- Examples of new features you have not used yet include:
  - managing accessibility (for people with disabilities
  - accessing resources on Google Drive
- Examples of major sub-features include:
  - managing Contact groups
  - accessing Twitter using Backendless.

7 Choose the new feature(s) for your app:

- If you are creating a *new* app, you need to implement one new feature/sub-feature.
- If you are enhancing an *existing* app, you must implement two new features/sub-features.

8 Choose a subset of features you already know from one of the following lists. Choose from *either* the Android list or LibGDX list. (If you completed the AP Unit 4 in place of the Android/LibGDX Unit 4, do not choose the LibGDX option.) Your teacher will tell you how many features are required.

- **Android Features**

Feature	Lesson	Details
Sound	Activity 2.1.3	Play sounds that a user requests, or respond to events with sound.
Camera	Activity 2.4.1	Use the camera to take and display pictures; provide storage and retrieval for the pictures.
Backendless	Activities 3.1.2–3.1.5	Implement front-end and back-end services to access data; tables must be unique to this app.
Google Play	Activities 3.2.1–3.2.2	Implement Location Awareness and Maps that are integrated into the app.
Contacts	Activities 3.3.2–3.3.3	Use implicit intents to access local Contacts data in a new way.

- **LibGDX Features**

Feature	Lesson	Details
Backendless	Lesson 3.1	Implement front-end and back-end services to save game state data; tables must be unique to this app.
Menu Screen	Lesson 2.1	Create a menu screen, reachable through in-game events, to provide players with options for game play.
Interactivity	Lesson 4.2	Create interactive elements to a game that receive input from the user and keep score.

9 Verify with your teacher that the feature set you chose is reasonable and manageable and can be completed in the time allowed for this project.

10 Based on the problem statement and the requirements of your app, create a user persona. Write specific ideas and include motivations and personalities.

11 Identify a group of people that fits your user persona. They will become your stakeholder group. Plan for a meeting with them.

12 Document how you prepare for the meeting, specifically how you will introduce and discuss your project and app. Your teacher may provide sample prompts for talking with your stakeholder group.

13 Meet with your stakeholder group. Together you will review the problem statement.

a. Define the “must-have” functions; what the app *needs* to do. Prioritize these functions. Be prepared to modify your requirements and feature set based on this input.

b. Describe the major characteristics, such as theme (playful, educational, professional), color schemes, important media, and so on. Take care not to delve too deeply into details; this should be a high-level talk, not how buttons are shaped or comparing different shades of favorite colors.

c. Create a “wish list” of functions that go beyond this first phase of development.

d. At the end of the meeting, thank them for their time!

14 With the input from your meeting, adjust your user persona, if necessary.

Part II: Design

With the planning phase complete and with stakeholder feedback in mind, design your app.

15 Brainstorm a variety of designs, create sketches and prototypes, and decide on one design to bring back to your stakeholder group. You may use scrum poker to reach a group decision.

16 Meet again with your stakeholder group to validate your prototype.

a. Verify that all major functions are represented in the prototype and that the prototype meets with their general approval.

b. Document their feedback and anything else they share with you.

c. Again, thank them for their time.

17 If necessary, modify your design based on the feedback from your second meeting with your stakeholder group.

18 Create the user stories for your app and create a backlog.

19 Design classes, methods, and/or instance fields you need for your app. UML diagrams will help with this task.

Part III: Develop and Test

You are now ready to begin implementation. All features should be seamlessly and logically arranged. Do not simply duplicate ones from other apps. Integrate them so they fit naturally and authentically into your design.

20 Create the major components of the user interface and other necessary XML files. In no particular order:

- Create the new classes and methods you designed for your app.
- Implement the new feature(s).
- Incorporate existing features from other apps.

21 Cite the sources where you discovered new feature(s) and how to implement them.

22 Provide Javadocs and other in-line documentation.

23 Test using a physical device and an emulator.

24 Test landscape and portrait orientations. If your app relies on a fairly large number of images, you may choose to lock the orientation to reduce the number of images you have to modify.

25 If directed by your teacher, share your app with another team during development and test another team's app. Provide constructive feedback and respond to feedback in a positive way.

Part IV: Deliver and Present

Prepare and deliver two presentations—one to your stakeholder group and one to the class. The stakeholder group presentation will be a subset of the classroom presentation.

26 Prepare and deliver a presentation to the class:

a. Explain the team's decision to create this particular app and share your problem statement.

b. Describe the user persona and how your app meets the needs stated in your problem statement.

c. Describe the current state of the product with respect to the problem statement, product backlog, and sprint task list.

d. Demonstrate the features of the app.

e. Explain the team's contributions of original code to the product solution, as well the integration of code from previous projects.

a. Explain the new feature and what you learned about it.

b. Reflect on how the app represents what you have learned in this course.

27 Prepare an executive summary presentation to deliver to your stakeholder group:

a. Describe the current state of the product with respect to the problem statement, product backlog, and sprint task list.

b. Demonstrate the features of the app that the team created.

c. Collect feedback from the stakeholders as they use the app themselves.

Problem 4.3.1 Make an App - Requirements

Problem 4.3.1 Make an App requires documentation, specific functionality, and a team presentation.

Documentation Requirements

1. Problem statement
2. Brainstorm list and sketches, prototypes
3. User persona including meeting preparation, stakeholder needs, ideas, wish list, and feedback
4. Product backlog and sprint task list
5. Citations for sources of new features
6. Product source code

Functionality Requirements

1. Address a problem that can be solved, tell a story, or play a game.
2. Choose from three options:
    - Create a new, original app using Android™ features.
    - Enhance an existing app from a previous lesson.
    - Modify the Emu on the Loose game from Unit 4 with new LibGDX functionality.
3. Implement the specific requirements described in **4.3.1B Make an App**.
4. Use the **4.3.1 Problem Rubric**, which describes the common rubric, and either **4.3.1 ProblemRubric Android** or **4.3.1 ProblemRubric LibGDX**, which describe the rubrics specific to each feature set.

Presentation Requirements

Develop one presentation and prepare two versions of it, one short version and one long version. For presentation details, see **4.3.1B MakeAnApp**.

Make an App Rubric - Common Elements

RESOURCES

4.3.1 Problem Rubric Common
Resources available online

Presentation Criteria	4	3	2	1	Total
Product Demonstration	The tools selected to communicate about the project allow for an exceptionally clear view of the team's work.	The tools selected for communicating about the project allow for a mostly clear view of the team's work.	The tools selected for communicating about the project allow for a somewhat clear view of the team's work.	The tools selected for communicating about the project give an unclear view of the team's work.	□
	Demonstration of the product highlights all features of the product.	Demonstration of the product highlights most of the features of the app.	Demonstration of the product highlights many of the features of the app.	Demonstration of the product highlights a small subset of the features of the app.	□

Presentation Criteria	4	3	2	1	Total
	The team discusses: • The new feature(s) • How a new feature compares to at least one known feature • How the team learned the new feature(s)	The team discusses two of the following: • The new feature(s) • How a new feature compares to at least one known feature • How the team learned the new feature(s)	The team discusses two of the following: • The new feature(s) • How a new feature compares to at least one known feature • How the team learned the new feature(s)		□
	All team members contributed equally to the presentation.	All team members contributed to the presentation, but some more than others.	All team members contributed to the presentation, but one member dominated.	One or more team members contributed nothing to the presentation.	□
Development Process Comments	The team thoroughly and clearly explains how they chose their app, including: • Justification • Brainstorming ideas • Early sketches or prototypes	The team explains two of the following: • Justification • Brainstorming ideas • Early sketches or prototypes	The team explains one of the following: • Justification • Brainstorming ideas • Early sketches or prototypes	The team briefly describes the app they chose without justification, brainstorming, or prototypes	□

Presentation Criteria	4	3	2	1	Total
	The team describes the user persona in detail, including: • Needs • Ideas • Wish list	The team describes the user persona, including two of the following: • Needs • Ideas • Wish list	The team describes the user persona, including one of the following: • Needs • Ideas • Wish list	The team minimally describes the user persona.	□
	The team describes: • How the app meets the needs of their stakeholders • The user experience • User feedback	The team describes two of the following: • How the app meets the needs of their stakeholders • The user experience • User feedback	The team describes one of the following: • How the app meets the needs of their stakeholders • The user experience • User feedback	The team briefly mentions their stakeholders.	□
	The team thoroughly and clearly explains the next steps for development of the product.	The team mostly explains the next steps for development of the product.	The team somewhat explains the next steps for development of the product.	The team minimally explains the next steps for development of the product.	□

Presentation Criteria	4	3	2	1	Total
User Interface	User interface is exceptionally logically arranged.	User interface is mostly logically arranged.	User interface is somewhat logically arranged.	User interface is not logically arranged.	□
	Navigation between fragments, scenes, or levels is exceptionally easy.	Navigation between fragments, scenes, or levels is mostly easy.	Navigation between fragments, scenes, or levels is clumsy or difficult.	Navigation between fragments, scenes, or levels is non-existent.	□
New Feature (Android™ or LibGDX)	A new feature is well integrated into the app, working smoothly and logically with other features of the app.	A new feature is implemented, but it is logically out of place or isolated from other features.	A new feature is implemented, but it appears both logically out of place and isolated from other features.		□

Presentation Criteria	4	3	2	1	Total
	The new feature represents significant learning, methods are used properly, and there are no problems/bugs in the app.	The new feature represents moderate learning, methods are used properly, and there are few or no problems/bugs in the app.	The new feature represents some learning, methods are used, and there are few problems/bugs in the app.	The new feature represents little learning, methods may be used improperly, and/or there are bugs in the app.	□
Design	The relationships among the team's original classes, fields, and methods are exceptionally clear.	The relationships between the team's original classes, fields, and methods are mostly clear.	The relationships between the team's original classes, fields, and methods are somewhat clear.	The relationships between the team's original classes, fields, and methods are not clear.	□
	The reason for the creation of each of the team's original classes, fields, and methods is exceptionally clear.	The reason for the creation of each of the team's original classes, fields, and methods is mostly clear.	The reason for the creation of each of the team's original classes, fields, and methods is somewhat clear.	The reason for the creation of each of the team's original classes, fields, and methods is not clear.	□

Presentation Criteria	4	3	2	1	Total
	All features of the design are encapsulated as much as possible; all methods and classes are organized according to function.	Most features of the design are encapsulated; most methods and classes are organized according to function.	Some features are encapsulated; some methods and classes are organized according to function.	Encapsulation is minimally present.	□

For each feature set, also refer to the appropriate rubric:

- **4.2.1 ProblemRubric_Android**
- **4.2.1 ProblemRubric_LibGDX**

Make an App Rubric - Android™ Features (not all may be required)

RESOURCES

4.3.1 Problem Rubric Android
Resources available online

Android Criteria	4	3	2	1
Sound	MediaPlayers: • Create an integrated, natural use of sound in the app • Are stored in a list • Have varying delays • Allow optional "quiet mode"	MediaPlayers accomplish three of the following: • Create an integrated, natural use of sound in the app • Are stored in a list • Have varying delays • Allow optional "quiet mode"	MediaPlayers accomplish two of the following: • Create an integrated, natural use of sound in the app • Are stored in a list • Have varying delays • Allow optional "quiet mode"	MediaPlayers accomplish one of the following: • Create an integrated, natural use of sound in the app • Are stored in a list • Have varying delays • Allow optional "quiet mode"
Camera	The camera is accessed and images are: • Captured • Stored • Retrieved • Displayed	Three of the following are accomplished regarding camera images: • Captured • Stored • Retrieved • Displayed	Two of the following are accomplished regarding camera images: • Captured • Stored • Retrieved • Displayed	One of the following is accomplished regarding camera images: • Captured • Stored • Retrieved • Displayed

Android Criteria	4	3	2	1
Backendless	Tables are: • Newly created • Uploaded • Downloaded • Used to demonstrate data persistence	Three of the following are accomplished regarding tables: • Newly created • Uploaded • Downloaded • Used to demonstrate data persistence	Two of the following are accomplished regarding tables: • Newly created • Uploaded • Downloaded • Used to demonstrate data persistence	One of the following is accomplished regarding tables: • Newly created • Uploaded • Downloaded • Used to demonstrate data persistence
Google Play	The service is implemented and is an integrated, functional part of the app that works well with other features of the app.	The service is implemented but is not an integrated part of the app and seems to stand apart from the rest of the app.	The service exists but is missing some of its implementation to be fully functional.	The service exists but is missing most of its implementation to be fully functional.
Contacts	Contact information is implemented in a new and unique way, and is well integrated within the app.	Contact information is implemented in a new and unique way, but has an ill fit in the app.	Contact information is implemented within the app, but is not used in a new or unique way.	Contact information is retrieved but without originality and integration.

Android Criteria	4	3	2	1
Animation	Four or more of the following are implemented: • Location animation • Color animation • Scale animation • Transparency animation • Touch response	Three of the following are implemented: • Location animation • Color animation • Scale animation • Transparency animation • Touch response	Two of the following are implemented: • Location animation • Color animation • Scale animation • Transparency animation • Touch response	One of the following is implemented: • Location animation • Color animation • Scale animation • Transparency animation • Touch response

Make an App Rubric - LibGDX Features (not all may be required)

RESOURCES

4.3.1 Problem Rubric LibGDX
Resources available online

LibGDX Criteria	4	3	2	1
Persistent Data	All game state data can be saved and loaded by the players.	Most game state data can be saved and loaded by the players.	Some game state data can be saved and loaded by the players.	Minimal game state data can be saved and loaded by the players.
Multiplayer Matchmaking	Multiplayer game matchmaking options are successfully implemented.	Multiplayer game matchmaking options are mostly implemented.	Multiplayer game matchmaking options are somewhat implemented.	Multiplayer game matchmaking options are minimally implemented.
Multiplayer Hosting	Options for hosting and joining a game are successfully implemented.	Options for hosting and joining a game are mostly implemented.	Options for hosting and joining a game are somewhat implemented.	Options for hosting and joining a game are minimally implemented.
Physics	2D or 3D libraries from LibGDX or other sources simulate many physical phenomena.		2D or 3D libraries from LibGDX or other sources simulate some physical phenomena.	

LibGDX Criteria	4	3	2	1
Menu Screen	A menu screen is reachable through in-game events; it provides players with options for gameplay.		A menu screen is reachable through in-game events but does not affect gameplay.	
Game Engine	The game world is implemented using an original system.	The game world is implemented using a mostly original system.	The game world is implemented using a somewhat original system.	The game world is implemented using a minimally original system.

Problem 4.3.1 Rubric - Android One Page

RESOURCES

4.3.1 Problem Rubric One Page Android
Resources available online

Presentation Criteria	
Product Demonstration	**Development Process**
The tools selected to communicate about the project structure allow for an exceptionally clear view of the team's work.	The team thoroughly and clearly explains how they chose their app, including: • Justification • Brainstorming ideas • Early sketches
Demonstration of the product highlights all features of the app.	The team describes the user persona in detail, including: • Needs • Ideas • Wish list

Presentation Criteria	
Product Demonstration	**Development Process**
The team: • Discusses the new feature. • Compares it to at least one known feature. • Explains how the team learned the new feature. □	The team explains: • How the app meets the needs of their stakeholders • User experience • User feedback □
All team members contribute equally to the presentation. □	The team thoroughly and clearly explains the next steps for development of the product. □

Product Criteria			
User Interface and New Feature	**Design**	**Android™ Functionality**	
User Interface User interface is exceptionally logically arranged. □	The relationships among the team's original classes, fields, and methods are exceptionally clear. □	MediaPlayers accomplish all of the following: • Create an integrated, natural use of sound in the app • Are stored in a list • Have varying delays • Allow optional "quiet mode" □	Google Play service is implemented and is an integrated, functional part of the app that works well with other features of the app. □

Product Criteria			
User Interface and New Feature	**Design**	**Android™ Functionality**	
Navigation between fragments, scenes, or levels is exceptionally easy. □	The reason for creation of each of the team's original classes, fields, and methods is exceptionally clear. □	The camera is accessed; images are: • Captured • Stored • Retrieved • Displayed □	Contact information is: • Retrieved • Parsed for a new and unique use • Well integrated with the app □
New Feature A new feature is well integrated into the app, working smoothly and logically with other features of the app. □	All features of the design are encapsulated as much as possible; all methods and classes are organized according to function. □	Backendless tables are: • Newly created • Uploaded • Downloaded • Used to demonstrate data persistence □	The following effects are implemented: • Location animation • Color animation • Scale animation • Transparency animation • Touch response □
The new feature represents significant learning, methods are used properly, and there are no problems/bugs in the app. □			*(Optional student-chosen feature)* □

Problem 4.3.1 Rubric - LibGDX One Page

RESOURCES

4.3.1 Problem Rubric One Page LibGDX
Resources available online

Presentation Criteria		Product Criteria			
Product Demonstration	**Development Process**	**User Interface and New Feature**	**Design**	**LibGDX Functionality**	
The tools selected to communicate about the project structure allow for an exceptionally clear view of the team's work. □	The team thoroughly and clearly explains how they chose their app, including: • Justification • Brainstorming ideas • Early sketches □	**User Interface** User interface is exceptionally logically arranged. □	The relationships among the team's original classes, fields, and methods are exceptionally clear. □	All game state data can be saved and loaded by the players. □	A menu screen is reachable through in-game events; it provides players with options for gameplay. □

Presentation Criteria		**Product Criteria**			
Product Demonstration	**Development Process**	**User Interface and New Feature**	**Design**	**LibGDX Functionality**	
Demonstration of the product highlights all features of the app. □	The team describes the user persona in detail, including: • Needs • Ideas • Wish list □	Navigation between fragments, scenes, or levels is exceptionally easy. □	The reason for creation of all of the team's original classes, fields, and methods is exceptionally clear. □	Multiplayer game matchmaking options are successfully implemented. □	The game world system is implemented using an original game engine system. □
The team: • Discusses the new feature • Compares it to at least one known feature • Explains how the team learned the new feature □	The team explains: • How the app meets the needs of their stakeholders • User experience • User feedback □	**New Feature** A new feature is well integrated into the app, working smoothly and logically with other features of the app. □	All features of the design are encapsulated as much as possible; all methods and classes are organized according to function. □	Options for hosting and joining a game are successfully implemented. □	*(Optional student-chosen feature)* □

Presentation Criteria		Product Criteria			
Product Demonstration	Development Process	User Interface and New Feature	Design	LibGDX Functionality	
All team members contribute equally to the presentation. □	The team thoroughly and clearly explains the next steps for development of the product. □	The new feature represents significant learning, methods are used properly, and there are no problems/ bugs in the app. □		2D or 3D libraries from LibGDX or other sources simulate many physical phenomena. □	