# Peer-graded Assignment: Capstone Assignment 4.1 – Determine the app's information needs

Submit by December 30, 11:59 PM PST

**Important Information**
It is especially important to submit this assignment before the deadline, December 30, 11:59 PM PST, because it must be graded by others. If you submit late, there may not be enough classmates around to review your work. This makes it difficult - and in some cases, impossible - to produce a grade. Submit on time to avoid these risks.

ⓘ It looks like this is your first peer-graded assignment. Learn more ✕

## Instructions

## My submission

This assignment will challenge you to think about the app's information needs when storing data remotely.

Discussions

**Review criteria** less ^

You will be graded on the completeness, clarity, and readability of your diagram.

**Step-By-Step Assignment Instructions** less ^

**Setup Instructions**

Here are instructions on how to setup Android Studio for the course:

Mac_AndroidStudioTutorial.pdf

Windows_AndroidStudioTutorial.pdf

To determine the app's information needs you will need to download and examine the following code:

SharingApp_c3_local_storage.zip

For your reference here an Android Studio Tips guide:

Android Studio Tips.pdf

You will be expected to upload a PDF of your diagram. A free online tool you may use to make your diagram is Lucidchart.

**Recap**

For those of you who are new to this specialization, the code base you will be working with involves an Android application. Capstone assignments in the Service-Oriented Architecture course build off the Android app from the earlier Software Architecture course.

In the application's current state:

A user of the app is able to create and edit a profile with a unique username and an email address.

A user of the app is able to login and logout.

An owner is able to record the items they own and wish to share.

A bidder is able to place bids on items they wish to borrow from other owners, which changes the status of the item to "Bidded".

An owner is able to accept a bid and lend the item to the bidder, which changes the status of the item to "Borrowed".

An owner may change the status of an item they own from "Borrowed" to "Available" when the item is no longer being borrowed.

An owner may view: all of their items; their "Available" items; their "Bidded" items; or, their "Borrowed" items.

A borrower may view a list of items they are borrowing.

A user of the app is able to search a list of all "Available" and "Bidded" items by keywords.

An owner is able to view the profile of a borrower who is borrowing one of their items.

## How to create your assignment

We want to have all the application's saved data stored remotely, so that users can share information across their devices. In order to do this, we will have to decide what data we want stored and how we want to store it for the app to continue functioning properly.

Review the application provided. While you review the code, determine which classes need their variables to be stored remotely.

Once you have identified what needs to be stored, create a small UML Class Diagram to show which data should be stored. You only have to include the classes and corresponding attributes that will have data stored remotely. You shouldn't have to include every class in the app. It is okay to leave methods out of the diagram for the purpose of this assignment.

## Style guidelines for UML class diagram

- whole things should be drawn to the left of the part
- there should be few crossing edges
- boxes should not overlap other boxes or edges
- diagram should flow from top to bottom and left to right