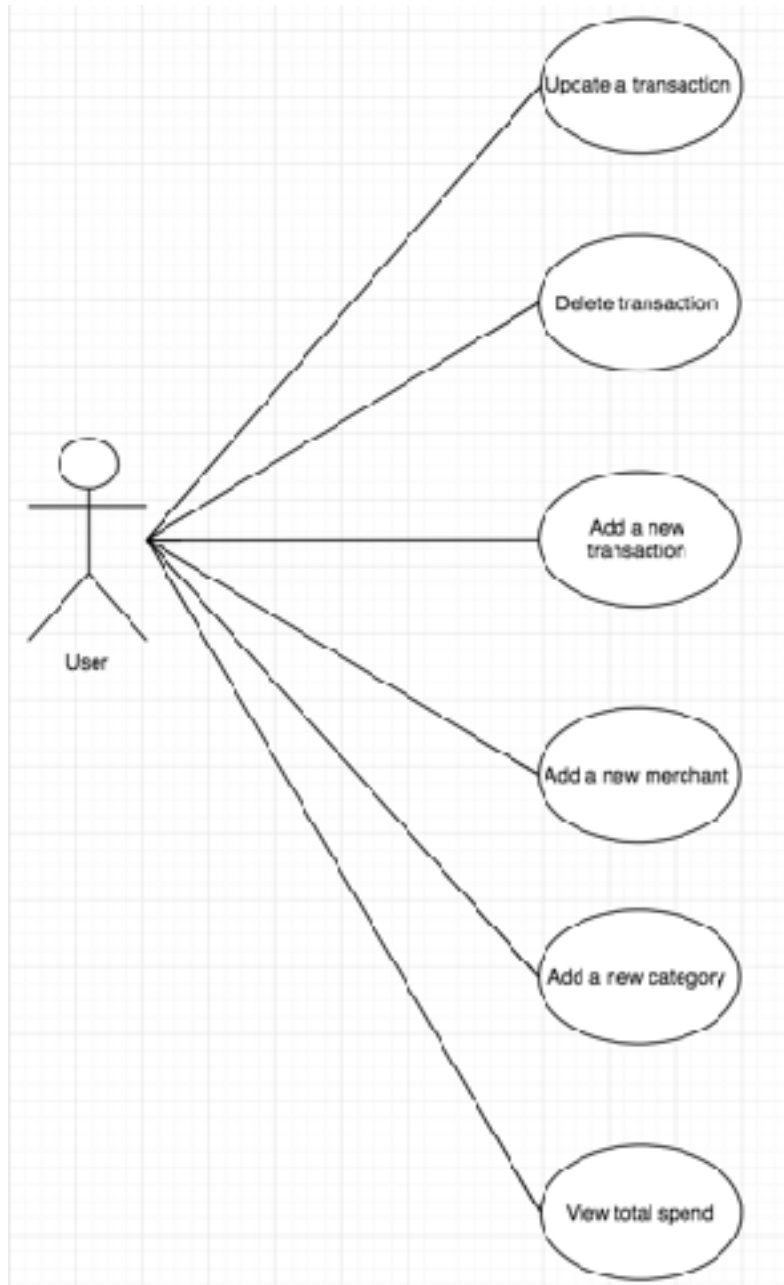


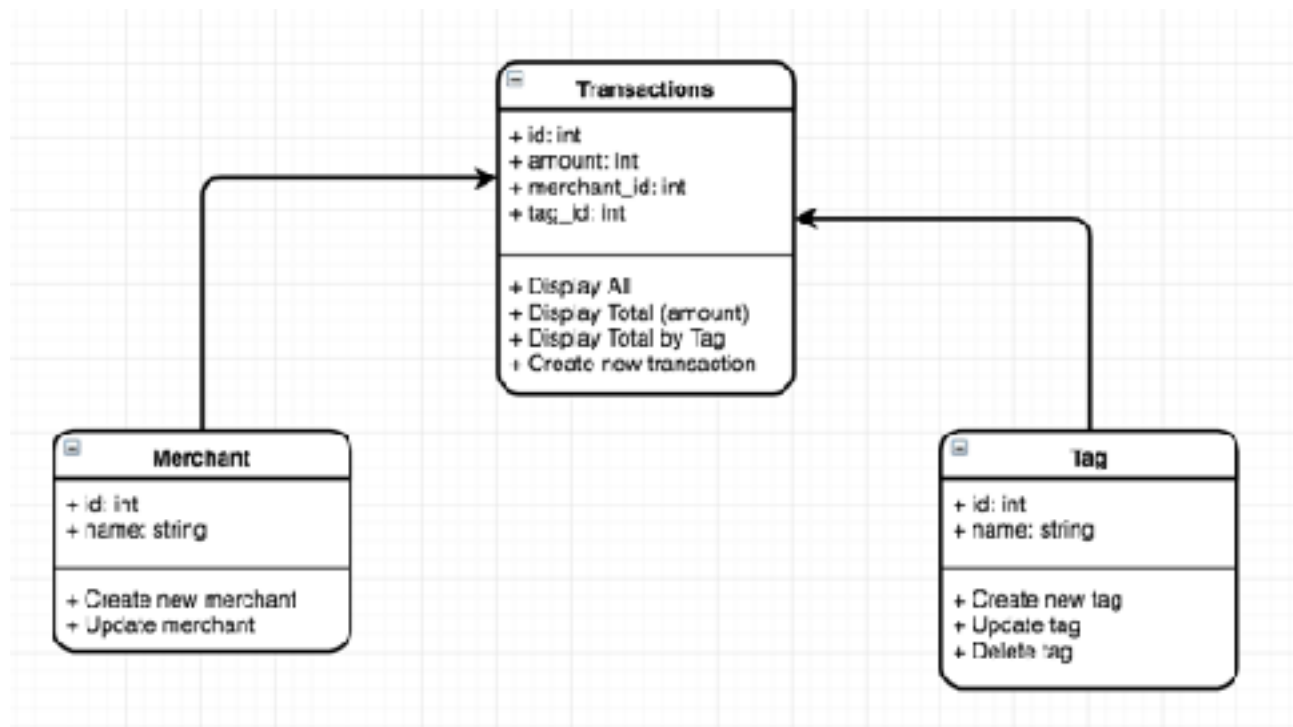
## Evidence for Analysis and Design Unit

Rob Gathergood  
E19

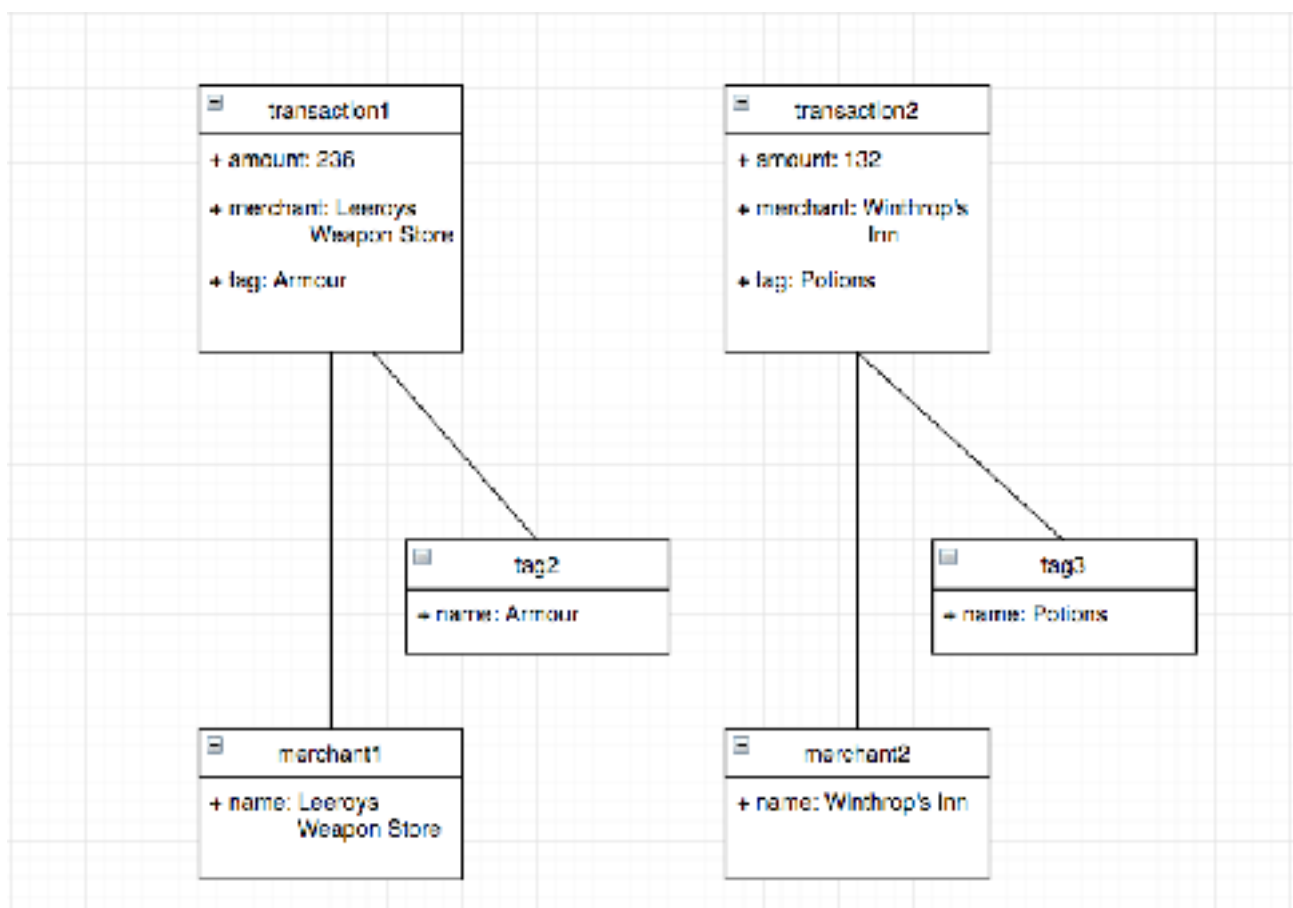
### A.D.1 Use Case Diagram



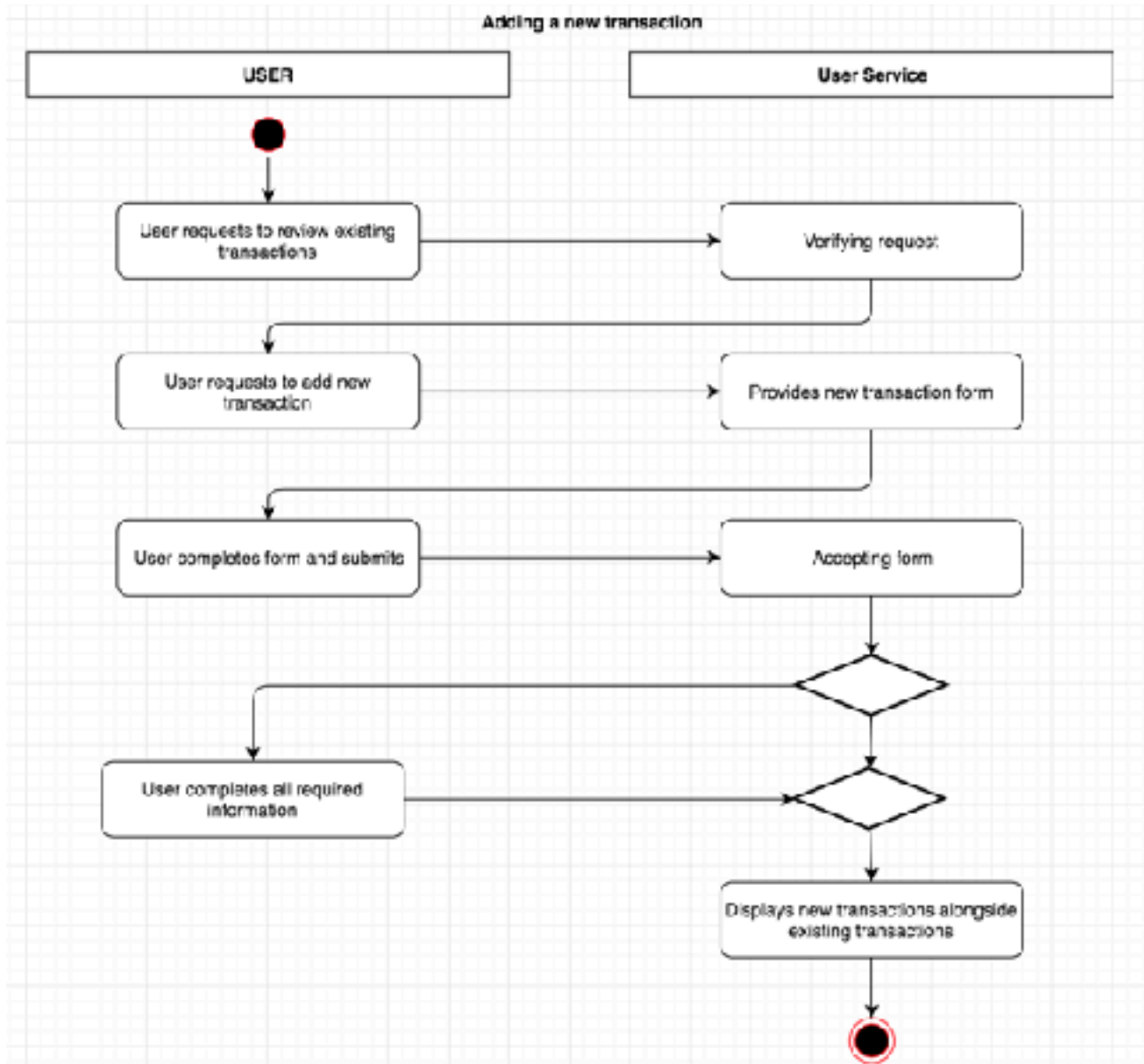
### A.D.2 Class Diagram



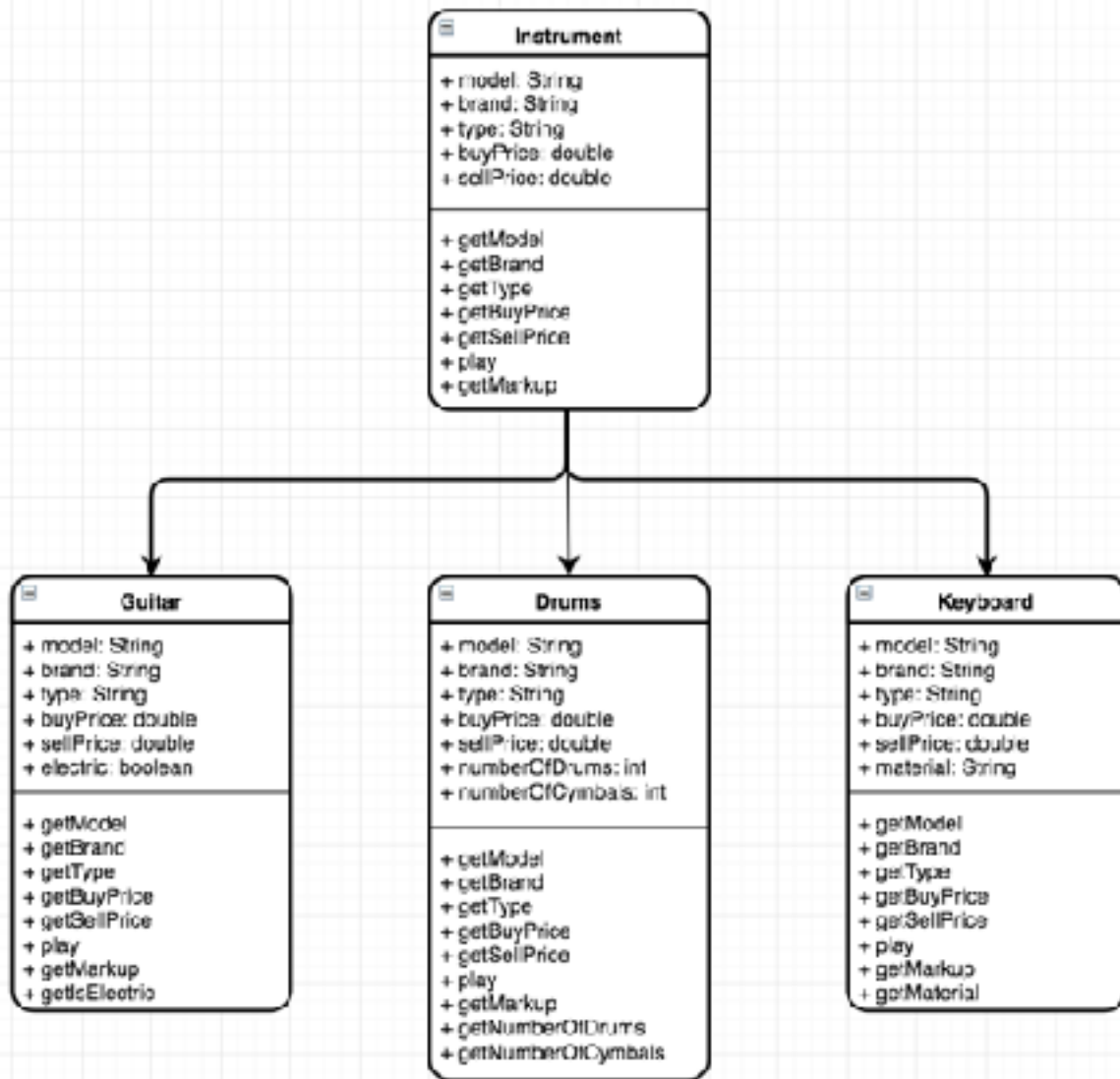
### A.D.3 Object Diagram



#### A.D.4 Activity Diagram



## A.D.5 Inheritance Diagram



## A.D.6 Implementations Constraints

	Constraint and possible effect	Solution
<b>Hardware and software platforms</b>	Software not up to date, cannot run plugin to load all data. Database information may fail to load. Users cannot access all features.	Ensure all software used is up to date and that the user is informed as to what is required for the product to run.
<b>Performance Requirements</b>	Hardware specifications not adequate, not enough RAM on device. Responsiveness is slow and data takes long to load. Users struggles to use the product as intended.	Use software compatible with current hardware specifications, ensure the user is informed and has access to the correct version.
<b>Persistent storage and transactions</b>	Cloud backup fails and data doesn't save. Information is not saved persistently. User loses valuable information.	Ensure the product has reliable storage and suitable alternatives. Encourage the user to regularly backup up work to be able to revert to previous versions.
<b>Usability</b>	Accessibility not taken into account, users unable to clearly read text on a webpage. Not all information is clear. Final product difficult/impossible to use for some users.	Take accessibility into consideration when planning, test UI before implementation.
<b>Budgets</b>	No allocated budget for product. Limits project scope. No access to alternative software/hardware or the ability to host the product without funds.	Set requirements accordingly, plan features which require no additional spend. Ensure the product spec doesn't need any funds.
<b>Time</b>	Product schedule of 1 week, no possibility of extension. Product is limited to functionality that can be implemented within a week. The product may not be completely finished by the end of the week.	Ensure all functionality is achievable within one week. Schedule work sensibly to allow contingency for errors. Don't add last minute features which could compromise usability.