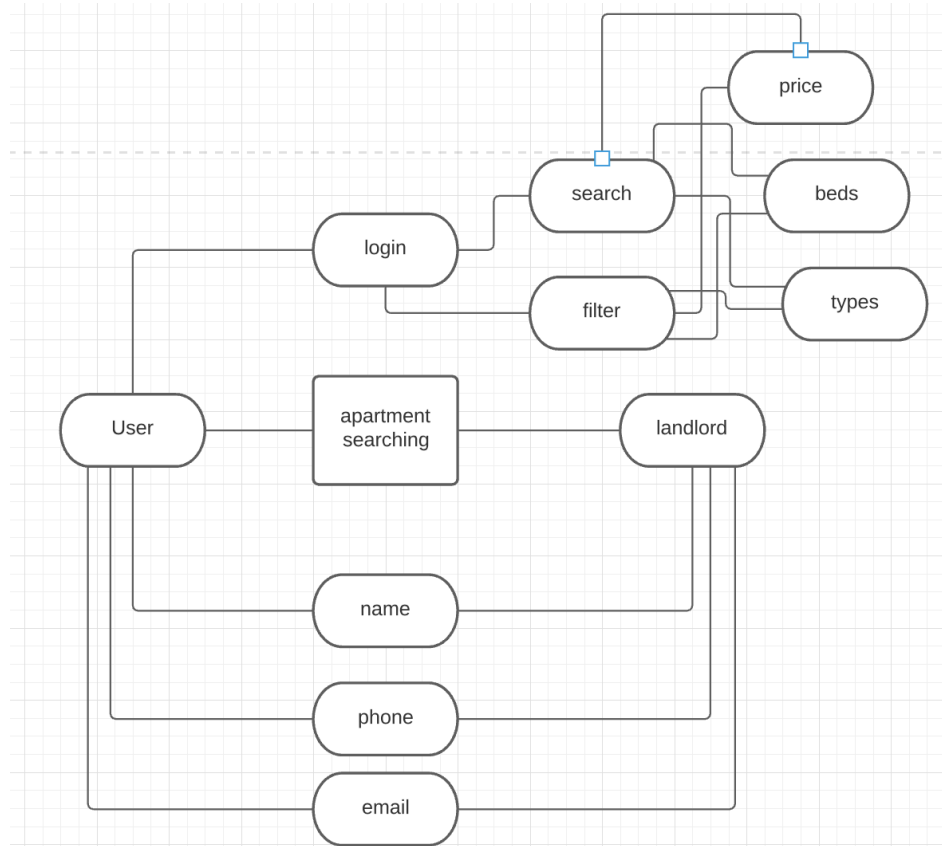**Question 1:**
i) Based on the dotted line in system A (one component is circled), I would apply unit testing to the system because unit testing is the testing of an individual unit or module in isolation to validate that each unit of the software performs as designed. In unit testing, we can have individual units or components of the software tested.

ii) Based on the dotted line, the entire of system B is circled by the line. I would apply system testing to this system because System testing involves testing the complete system as a whole. The purpose of this test is to evaluate the system's compliance with the specified requirements.

iii) In system C, multiple components are selected. I would apply integration testing to this system because as the name suggests, it is mainly done to test the modules or components work as expected when they are integrated. It ensures that individual modules that are working fine do not have issues when brought together.

Question 2:
Deliverable:

The platform I drew was about apartment searching where users can login, search and filter any available apartments that they wish to rent.

i) *Verifications:*
- Check to see if the user is able to login into their account.
- Check to see if the filter button does something when it is clicked.

Validations:
- Check to see if the user has logged into the right account.
- Check to see if the filter works as it intended to.

ii) *Non-functional:*
- After login, the dashboard loads in less than 3 seconds.
- When the user clicks search, the user is redirected to a new screen with all the listed available apartments.

Functional:
- When the user types in the right username and password, the login function works.
- When filter is clicked, the page loads with filtered apartments.

Question 3:

https://github.com/truongh1/leapYear

Question 4:

i) My project would be a website where users can buy, sell, or exchange legal items without the involvement of the third party. The website contains the items that are being sold or exchanged, and the contact information of the owners. It allows the users to search for any items they wish to buy or exchange.

ii)

Testing in Agile	Testing in Waterfall
- Integration testing could be used because agile relies less on formalized processes, with an emphasis on interaction, collaboration, and aims at continuous integration and release. - Making sure that individual modules that are working fine do not have issues when brought together - Faults are exposed in the interaction between integrated units.	- Unit testing could be applied to validate that a particular unit of the software system performs as expected or designed. - Having individual units or components of the software, tested - Waterfall requires a lot of planning and mistakes are problematic, so unit testing could be most appropriate.