

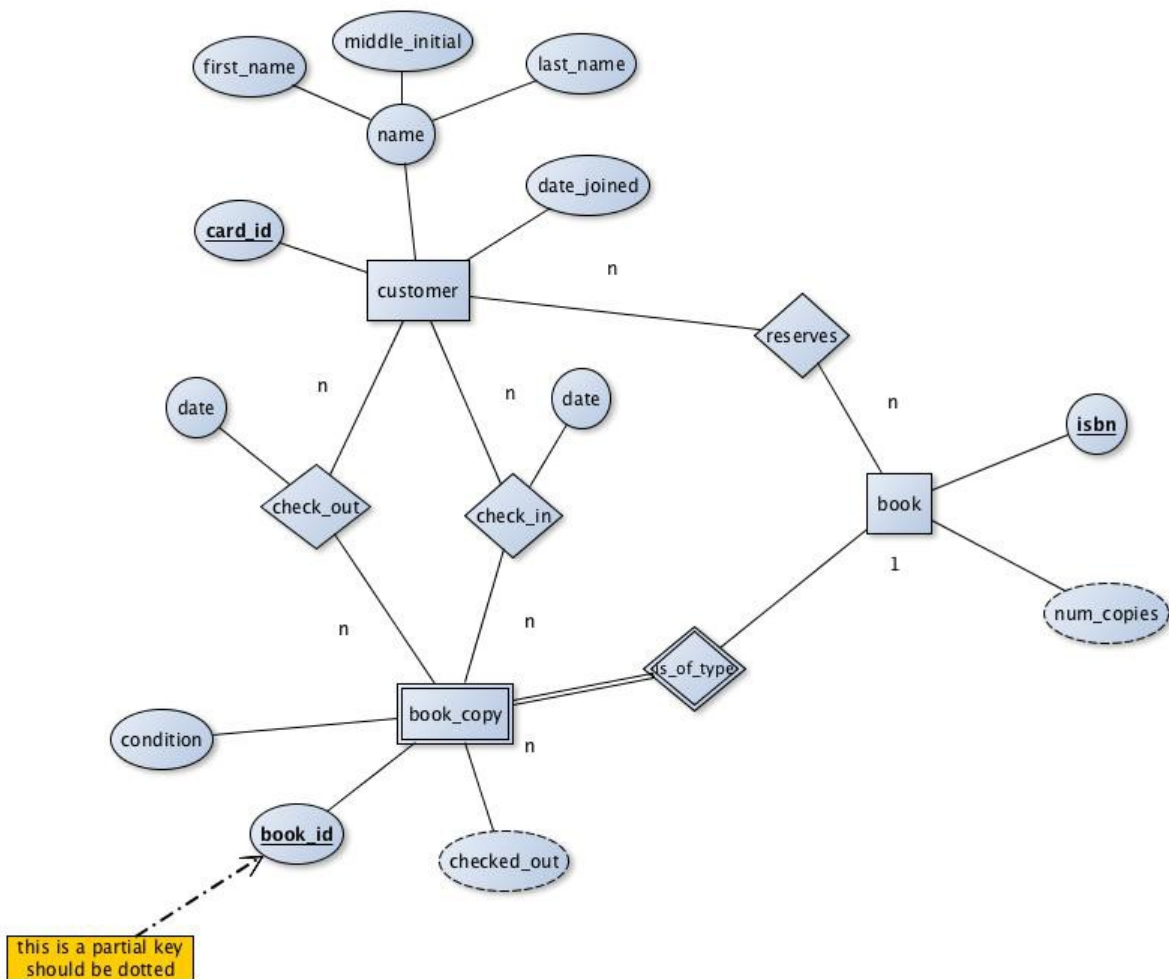
Assignment 3

Detailed Design and Verification & Validation

Total – 50 pts.

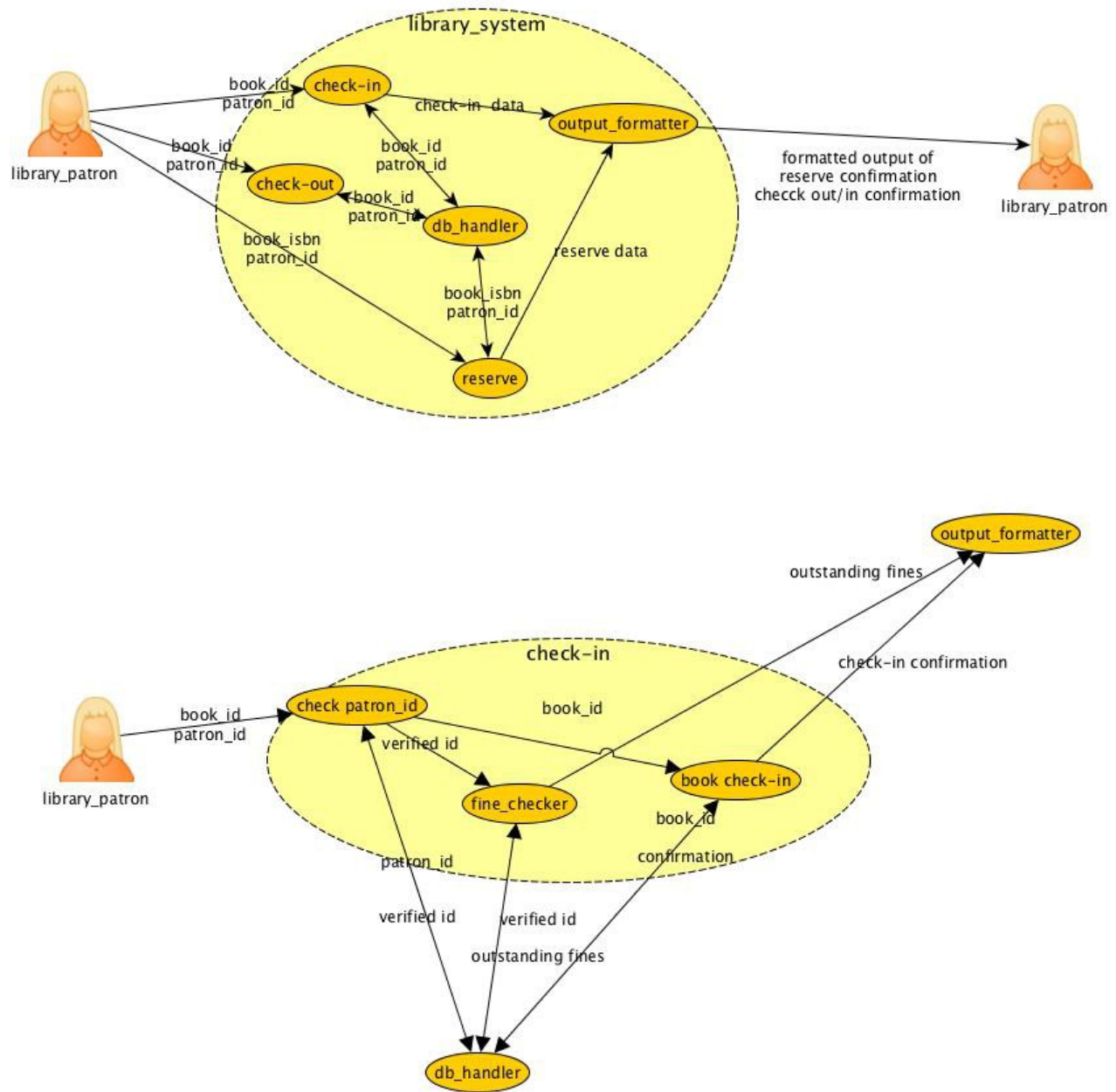
1. (5 pts.) Draw an ER diagram with at least three entities and two relationships for a system that keeps track of check-in/check-out of books in a public library. Provide the following for full credit: a key, at least two other attributes per entity, relationship names, and cardinalities.

Response:



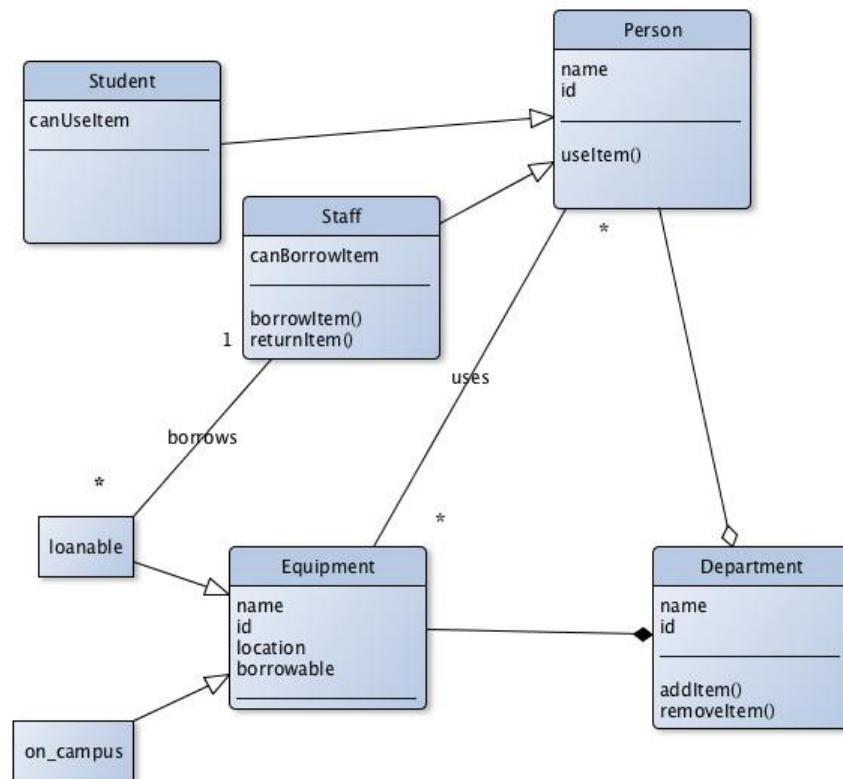
2. (5 pts.) Draw a DFD-1 for the system in question 1 with at least three processes. Assuming that check-in is one of the processes in DFD-1, create a DFD-2 for this process with at least three sub processes. Provide the following for full credit: appropriate process names, labels for data flows, and input/outputs.

Response:



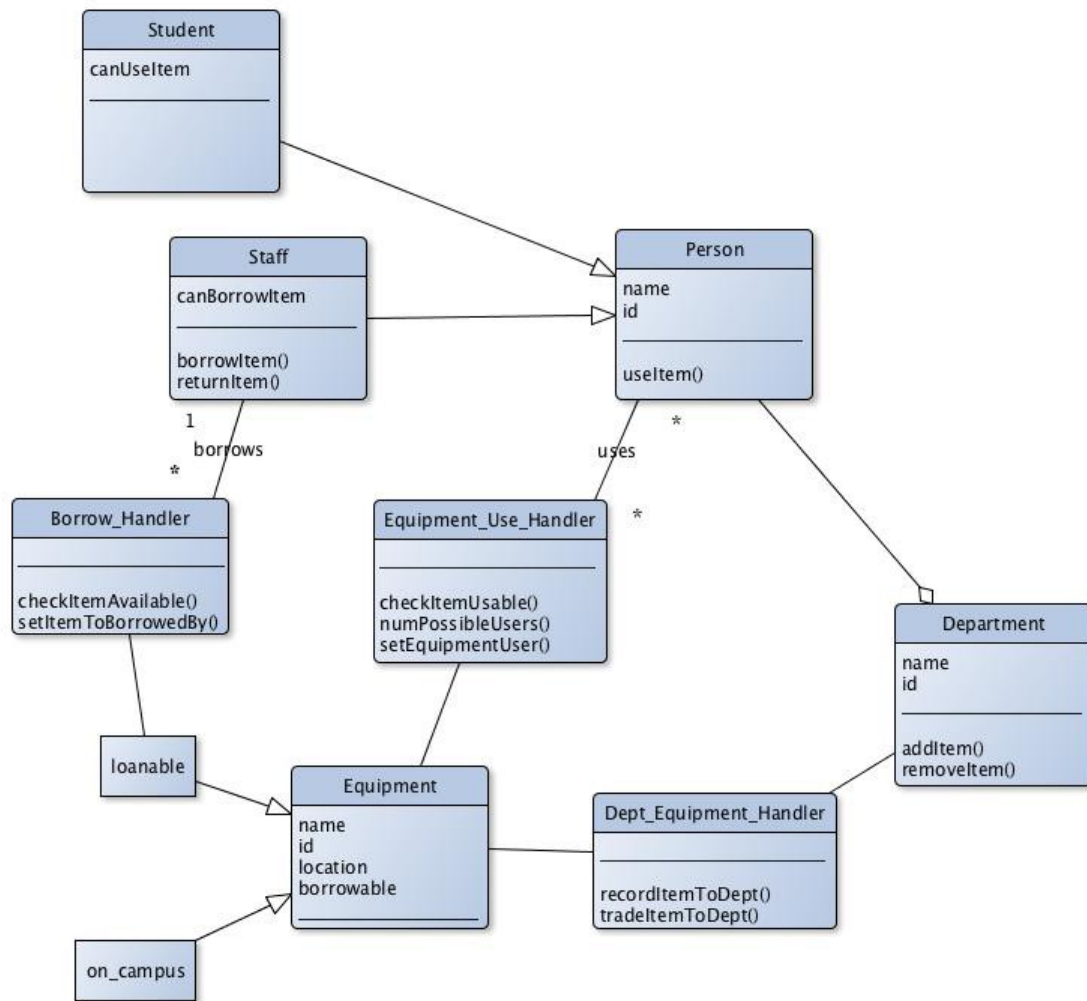
3. (5 pts.) Draw a class diagram with at least three data classes and one each of the three types of relationships (association, inheritance, aggregation) for an inventory system that keeps track of different types of equipments in computer labs. Provide the following for full credit: at least two attributes and one method per class, multiplicities, and association names.

Response:



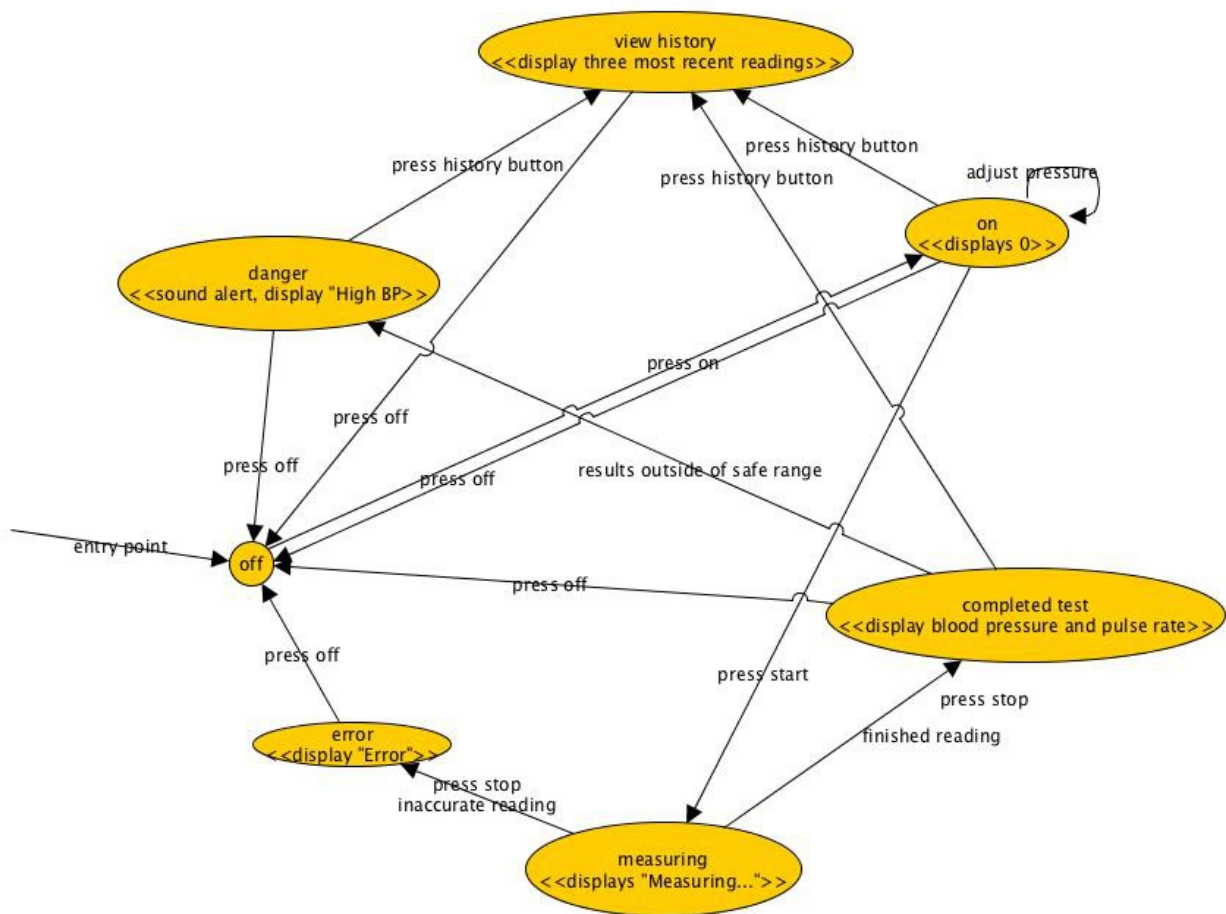
4. (5 pts.) Assuming an OO design, provide a class diagram that shows program control and behavior for the system in question 3 with at least three new classes. Provide at least two methods per class, multiplicities, and association names for full credit.

Response:



5. (5 pts.) Draw the state diagram for a software system that controls a personal blood pressure monitor. When the monitor is turned *on*, it displays a reading of zero. At this point, the monitor allows a user to change the maximum pressure applied to the arm, as an option. Once the *start* button is pressed, the monitor starts working and displays “Measuring...”. The user can press the *stop* button at any time during the measurement. When the measurement is done, it displays the blood pressure and the pulse rate. The user can optionally select the *history* button to view the last three readings for comparison purposes. If a successful measurement was not taken, the monitor displays “Error”. If the blood pressure or pulse reading is outside the normal range, the monitor beeps and displays an appropriate message, such as, “High BP”. The monitor is turned off by using the *off* button.

Response:



6. (5 pts.) Assuming you have objects of the Student and Course classes in a course registration system, draw a sequence diagram that shows the process of registration of a student into a course. Make use of at least one other object from a different class. Show at least five messages that are passed between these objects for full credit.

Response:

7. (5 pts.) Explain the differences between the types of relationships between classes in an object-oriented design – association, aggregation, composition, and inheritance. Explain how each of these relationships would get implemented in the code, i.e., attribute, method calls, etc..

Response:

Association: defines a relationship between classes of objects that allows one object instance to cause another to perform an action on its behalf.

Aggregation: a relationship between objects wherein one object is included as a part of another.

Composition: similar to aggregation where one object is included within another, however, in composition, the included object couldn't otherwise exist.

Inheritance: a relationship between objects where an object inherits all of the

qualities of another object.

8. (5 pts.) Check the following web page for an understanding of the different UML diagrams that are available to model a software application. http://upload.wikimedia.org/wikipedia/commons/d/d6/Uml_diagram2.png

Choose one diagram that wasn't discussed in the class and explain clearly the purpose of the diagram, its elements, and an example.

Response:

9. (3 pts.) Explain the difference between verification and validation of software. Using your group project provide one example each of a verification and a validation task.

Response: Verification is the process of assuring that the requirements are met, while validation is the processes of assuring that the software meets all of the client's needs are met.

Put simply:

Verification is answering the question: "Are we building the product right?"

Validation is answering the question: "Are we building the right product?"

10. (2 pts.) What kind of useful data can you get from user testing? Differentiate between alpha and beta testing.

Response: User testing is a way to validate and verify functionality; user testing can generate errors and defects that might otherwise not show up in alternative testing. Alpha testing is what happens to prepare the product for user readiness, done to emulate ~80% of user interactions, while beta testing is a more general testing allowing a larger variety of people to interact hopefully discovering the corner cases of the software.

11. (5 pts.) Using the template given (during lecture), create a test suite with at least five test cases to test a software module that computes the shipping and handling fee for an online retail store. The fee is added to each purchase which may include multiple items. The type, weight, and quantity of the items shipped determine the fee and some items are shipped for free. Make sure to include some valid and some invalid test cases in your suite.

Response:

Test Case	Input	Expected	Actual	Valid
1	Items = [{book, 2lbs}]	\$4.00	\$4.00	Valid
2	Items = [{car, 50lbs}, {tv, 16lbs}]	\$16.00	\$16.00	Valid
3	Items = []	\$0.00	Error	Invalid

Test Case	Input	Expected	Actual	Valid
4	Items = [{car, -1lbs}]	Error	Error	Invalid
5	Items = [{semi, MAX_INTEGER}]	\$0.00	\$56.00	Valid