CSE3310-003

Fall 2023

Midterm Exam

25 Points

1. Closed book.
2. No notes.
3. Individual work.
4. No calculator.
5. No phone.
6. Grading is based upon the UTA Standard, A is “Excellent”, B is “Good”, C is “Satisfactory”, and D is “Poor”.
7. What is Software Engineering? (2)
8. What is Computer Science? (1)
9. Describe the four “Fundamental Activities”. For each give an example of the inputs required and the output produced
   1. (1)
   2. (1)
   3. (1)
   4. (1)
10. Fill in the table below (‘yes', 'no’) :(1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SDLC | Adapts to Change? | Provides customer visibility into progress? | Allows for early delivery? | Meets the customers' original requirements? \* |
| Waterfall |  |  |  |  |
| Iterative |  |  |  |  |
| Integration and Configuration |  |  |  |  |

\* Refers to the situation in one SDLC that forces a customer to change requirements due to the in availability of a feature in the marketplace or the high cost of modification to support that feature

5. Describe the steps needed to improve a software process. (1)

6. What is a prototype? (include what a prototype is used for and what it should not be used for) (1)

7. What is the Capability Maturity Model? (1)

8. What is a functional requirement? (2)

9. What is a non-functional requirement? (2)

10. Review the following requirements. State if they are functional or non-functional. If they have a problem, describe the problem. (2)

1. Room temperature shall be maintained at the entered target temperature with +- 1 degree.
2. The system will be scalable and performant.
3. The product will be written in Java.
4. The daily report shall contain a listing of all the products shipped.
5. The system will create an abbreviated financial report, when warranted, that summarizes the complete financial report.

11. Describe a Unique Identifier (UID) in the context of requirements management. (include a discussion of deletion and re-use of the UID). (1)

12. Describe a technique used to validate Requirements? (1)

# Applied Modelling

AuthorPay!, a small startup in the Arlington area, is developing a software system to disrupt how authors and editors are paid for distributing creative written works. AuthorPay! Has experience with MySQL, so this database must be used for this software system. They also have experience with Perl, so that is the language that is going to be used.

“Creative written works” are also known as short stories and novels. They will be managed and delivered as PDF files.

The system connects readers to content producers, charging a 'per use' fee to the readers and passing on the proceeds (minus a small processing fee) to those that created it.

All users must log in and provide a credit card number. This number is then used to interact with the Credit Card Processing System ( a system not owned or controlled by AuthorPay! ) every time a file is requested and served to a user. The money is deposited by the Credit Card Processing System into the AuthorPay! Account.

Every PDF has at least one author, it may have many. Every PDF has a single editor. A customer can only request one PDF at a time. The only limit to the number of sequential requests is the credit limit of the customer.

Every month, the system provides a report on the payments earned by authors and editors. The management of AuthorPay! can then determine if they want or how they want to disburse payments (the sending of checks is not the responsibility of this software program).

For each user transaction:

· User logs onto website

· User selects PDF

· Transaction is sent to credit card processor

· If the transaction is OK, the PDF is allowed to be downloaded

The cost to the user of each file is determined when the file is uploaded. The percentages for each author, the editor, and AuthorPay! are also determined at this time. This application must be responsive to user requests because they bore easily. The entire transaction from selection to available for reading can take no more than 2 seconds.

Authors and editors have a name and an address. They have no credit card information.

13. Draw the context diagram for this software system. (1)

14. Draw a static model (class diagram) of this system as it is described. (2)

15. Identify 3 functional requirements from the problem statement and express them accurately and precisely. (Clear, concise, specific, accurate, ...) (2)

16. Identify 1 nonfunctional requirement from the problem statement and write it as a requirement. (1)

You are tasked with building a static model representing the products sold at a coffee shop. The model should show the relationships between the products, attributes, and methods. (4 points)

Beverages are either hot or cold. A beverage is either made from drip coffee, espresso, or tea.

There are various kinds of tea (Earl Grey, Tasmanian, Pekoe, etc.) and various kinds of coffee (Arabica, Robusto, or others).

An espresso, a mocha, a cappuccino, and a latte are all made with espresso. The cappuccino and latte are made with milk and are ‘frothed’ for contrasting times. A mocha has espresso, milk, and chocolate. And is frothed.

All beverages have a price. All cold beverages are served with ice. The policy of the coffee shop is to serve a latte with a complimentary cookie. Drip coffee can only be kept for 1 hour after being made; after that time, it needs to be discarded. Espresso and tea beverages are made on demand for each customer. (put answer on next page, please)

A coffeemaker has the following events:

Carafe Full. This event is triggered when the carafe is full of coffee. Carafe Empty. This event is trigged when the carafe is empty

Tank Full. This event occurs when the cold-water tank is full.

Start. This event occurs when the operator presses the button. An internal timer.

The coffeemaker has the following controls or outputs:

Dispense Ground Coffee. This is a one-time output that instructs the hardware to release enough ground coffee and a filter for 1 pot of coffee. The action takes place on the low to high transition. (It is a promising idea, but not required, to set the signal true when entering a state, and then clear it when transitioning out of a state) Water On/Off controls the input water supply. (Boolean). When true the water flows into the cold-water tank. When off, the water does not flow. Heater On/Off controls the heater that boils the water from the cold-water tank.(Boolean) If an error condition is encountered, the error light comes on and the device enters an error state. (Error light is Boolean). To get out of the error state, power must be cycled to the system. At power on, the carafe is empty, and the cold-water tank is empty. The system must diagnose a leaky cold-water tank; if it takes more than 30 seconds to fill the tank, then it is assumed there is a leak or a problem with the water source. Draw the state diagram of the coffee maker in operation, constrained by the above list of events and controls. (draw on next page, please) (4 points) In the event some behavior is not described in the above assignment, describe any assumptions you have made about the system.