Homework Assignment #2

5.1 What's the difference between a component-based architecture and a service-oriented architecture?

A component-based architecture aims to maximize the separation of each system element, enabling different developer teams to work on them independently. It also decouples the pieces of code while having them all contained within the same executable program, so they communicate directly. A service-oriented architecture is like a component-based architecture, but instead of components, it utilizes self-contained programs that run independently and communicate across a network.

5.2 Suppose you're building a phone application that lets you play tic-tac-toe against a simple computer opponent. It will display high scores stored on the phone, not in an external database. Which architectures would be most appropriate and why?

A single-player tic-tac-toe is a simple, independent application, and since it is not multiplayer there is no need for an online database or server which is why a Monolithic Architecture would be most appropriate. A Data-Centric Architecture would also work well for this game, since it uses tables instead of hard-wired code to control the application. The user interface would need to respond to the moves that the player makes, and in that sense it would be appropriate for it to be Event-Driven.

5.4 Repeat question 3 [after thinking about it; it repeats question 2 for a chess game] assuming the chess program lets two users play against each other over an Internet connection.

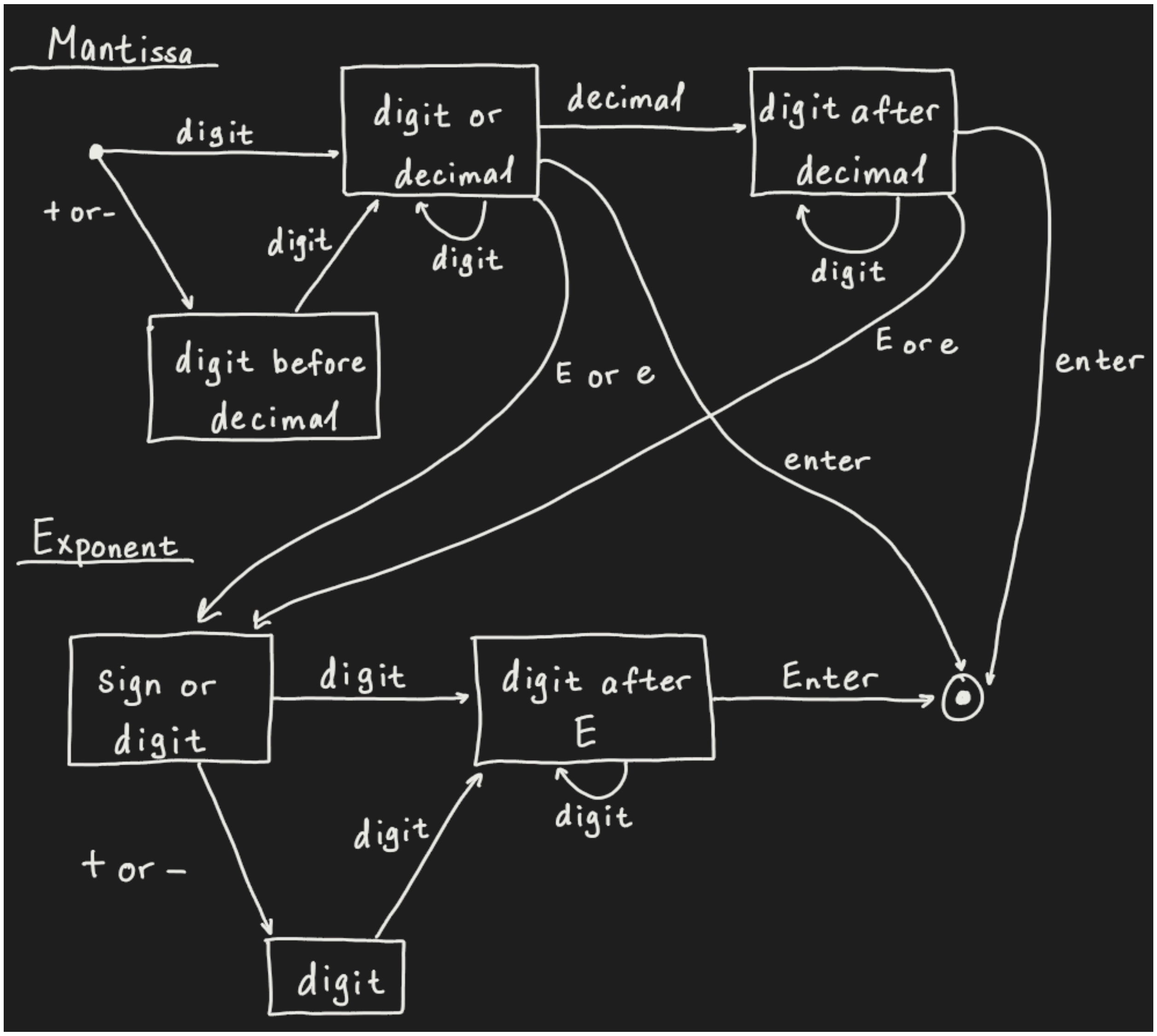
Considering the requirement for online multiplayer gameplay using the Internet, the Client/Server Architecture would be the most appropriate since it involves a server managing the game state and facilitating communication between the two clients. It would provide efficient communication between players, centralized game state management, and scalability to handle multiple concurrent game sessions. It would also be a good idea to combine this with a Service-Oriented Architecture because it can enhance the overall flexibility and maintainability of the

system. This would allow each player's client to interact with backend services for tasks such as matchmaking, game state management, and authentication.

5.6 What kind of database structure and maintenance should the ClassyDraw application use?

The ClassyDraw is a more complex MS Paint but it stores each shape as an object, which means we can store each drawing in its own file. In that sense, we wouldn't necessarily need a database because we can use the file storage system that comes with the computer's Operating System. If the file is actively open and being edited, we can use a temporary file like the .part file that appears when you're in the middle of a download.

5.8 Draw a state machine diagram to let a program read floating point numbers in scientific notation as in +37 or -12.3e+17 (which means -12.3 x 10^{17}). Allow both E and e for the exponent symbol. [Jeez, is this like Dr. Dorin's DFAs, or *what*???]

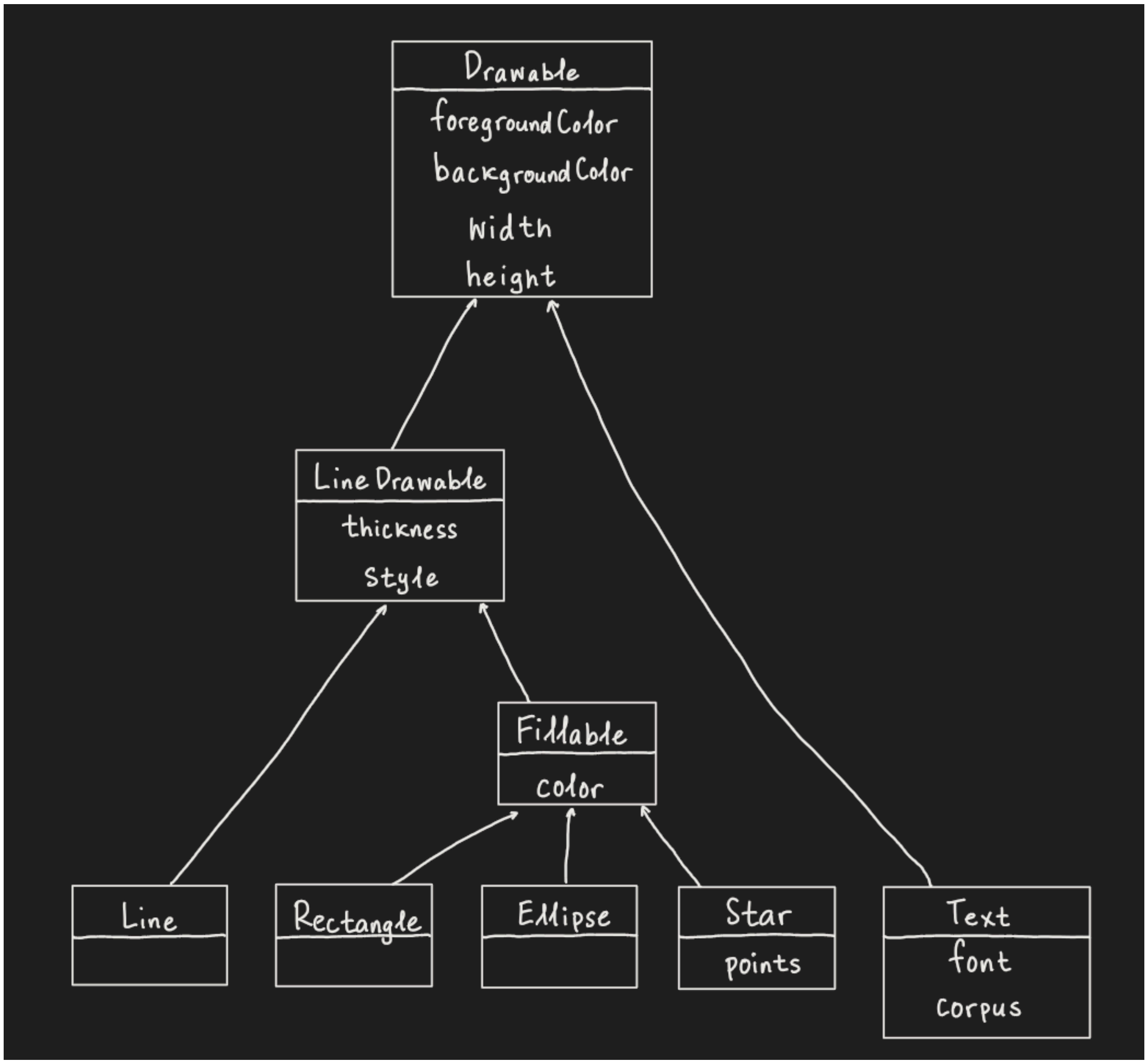


6.1 Consider the ClassyDraw classes Line, Rectangle, Ellipse, Star, and Text. What properties do these classes all share? What properties do they not share? Are there any properties shared by some classes and not others? Where should the shared and nonshared properties be implemented?

The classes Line, Rectangle, Ellipse, Star, and Text represent what is being drawn which means they share the similar drawing properties such as background color. These classes can use the height and the width to define the position of the drawing. However, since these classes have different properties they each will have some unique data that relates to their unique shape. For example, all classes besides

Text, would need to store data such as line thickness for their respective shapes. In addition, Star, Rectangle and Ellipse are shapes that we can fill with color, so they would need to store this color data.

6.2 Draw an inheritance diagram showing the properties you identified for Exercise 1. (Create parent classes as needed, and don't forget the `Drawable` class at the top.)


Drawable
foregroundColor
backgroundColor
width
height
Line Drawable
thickness
style
Fillable
color
Line
Rectangle
Ellipse
Star
points
Text
font
corpus