Architectural Styles

Please use the reference materials to fill out the following table:

Architectural Style	Strengths	Weaknesses	Examples
Layered	Santiago Bermudez There is improved cohesion which would allow for many physical deployment options. There is also decreased coupling that promotes re-use.	Santiago Bermudez It can be very inefficient to try and communicate through multiple layers. This often results in so-called "cheating".	Santiago Bermudez An example of layered architecture could be comparing the user interface that users see with the domain model that users cannot easily see.
Pipe-Filter	Enoch Hsu They interact with the environment in limited ways and simplify systems maintenance and enhance its reuse •Simple •Easy to extend and/or modify	Chris Long 1. Not very flexible because filters can't communicate with each other, so each filter really can only do one thing. 2. An error in the first filter could get piped to the rest of the filters.	Matthew Petruescu When you would use this - when the project can be broken down its parts. - When it would help the system from distributing it to multiple servers.
Client-Server	Tyler Ito - Inherits Layered architecture advantages Improved cohesion which allows for separate physical locations 2. Decreased coupling promotes re-use 3. Centralized control	Tyler Ito 1. Because there is centralized control, there is a single point of failure in the server 2. Difficult	Ramo Tucakovic An example of Client - Server could be the World Wide Web, Email, and network printing.
Event-Driven	Eric Truong -Entities are grouped into event generators/producers and event receivers/handlers. -They communicate through an intermediary called and event queue. There is usually a predefined hierarchy of objects to handle events. -Due to this it makes it easier to re-use certain pieces of code. It is also easy to	Eric Truong -It makes synchronization difficult due to the amount of events that can happen.	Eric Truong An example of an Event-Driven architecture style would be swiping a credit card because one event leads to more events happening. Another one would be pressing a button that leads to more events happening.

	modify a code when needed.		
MVC	Dane Coleman 1. Reusable without modification. 2. The model, view, and controller are loosely coupled. This means they act independently of each other and allow for easy updates on any of the three. 3. Allows for separation of concerns (dividing the logic into 3 buckets so that they can act independently).	Dane Coleman 1.Views and controllers are often hard to separate. 2.Frequent updates may slow data display and degrade user interface performance. 3.User interface components highly depend	Ramo Tucakovic A car for example. ● Every car consists of three main parts. ● View= User interface : (Gear lever, panels, steering wheel, brake, etc.) ● Controller- Mechanism (Engine) ● Model- Storage (gas or Diesel tank)