EECS 448

Team 16

Software Architecture

When creating the software architecture for our prototype, we developed the structure and organization of the software as well as identifying numerous elements. This includes the interactions with components such as the start page, paddle, ball movement, bricks, etc. We discuss each function in detail to facilitate its construction.

Considering that we have utilized the Event Driven Design model as one of our design paradigms, one software architecture present in our project is the Peer-to-Peer architecture. This type of architecture is when a distributed application will allocate tasks or workloads between peers. Event Driven Design encompasses a significant tool that can speed event notifications alongside applications and the Peer-to-Peer architecture. This architecture supports the system by ensuring that there is not a crash given one failure. In comparison, the Event Driven Design holds the ability to have multiple sources which processes the event if and when one element failed.

All users have the same privilege accessing this application and can participate on it too. Here, users will make a portion of the resources such as storage and processing power. They can download Github resources, and the application would still function given that it is not controlled by servers or a stable host. If there is another avenue being used by the system, we can conclude that there are both supplies and consumer for the resources since users can change the code without violating any rules.

The advantage of our architecture it that it has great availability. Users can upload and download our original code to their Github account. The loss of one user or if someone’s file breaks down will not corrupt the entire system. It has a fast recovery and can restore on another peer. The prototype we have chosen is a rapid prototyping life cycle model because we have a basic idea and understanding of what we are going to implement. Our group members act as users by interacting and experimenting with the project application. When disadvantages or a bug is brought up after running, we can fix the application again.