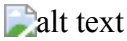
Software Architecture

Diagram



Description of Architecture

The architecture of PolyChord is a classic example of the model-view-controller architectural pattern; as a web application, it is written using HTML, Javascript, and CSS. There are no class-based structures in the implementation. Rather, the HTML code, representing the “model” component of the architecture, establishes the major data elements of the application (the keyboard with all of its keys). It includes the Javascript code (corresponding to the “controller” component), which primarily defines various event listeners that capture input to the keyboard/UI and appropriately modify the state of the keys or the view (the CSS defining the visual presentation of the HTML keyboard model), and that also trigger other functionality such as chord recognition.

The model-view-controller architecture was selected as the most appropriate for this app for a number of reasons. Overall it was simply the most intuitive given the nature of the application: we knew that the app would be fundamentally event-based, with a digital keyboard UI receiving input and processing it accordingly; and we knew that we wanted PolyChord to be platform independent; therefore, we ruled out the possibility of a native application, as such would involve bulky classes and more arduously crafted organization to represent the various aspects of the user interface and system functionality (whereas this is trivial with HTML/JS/CSS), and perhaps issues with achieving platform independence. So, a web application presented itself as the most viable strategy: this way, we could easily attain platform independence, and the concerns of the keyboard interface (model), app functionality, and visual presentation of the keyboard could be separated very cleanly and naturally with HTML/Javascript/CSS (intuitively in model-view-controller fashion), allowing for the greatest degree of parallel development.