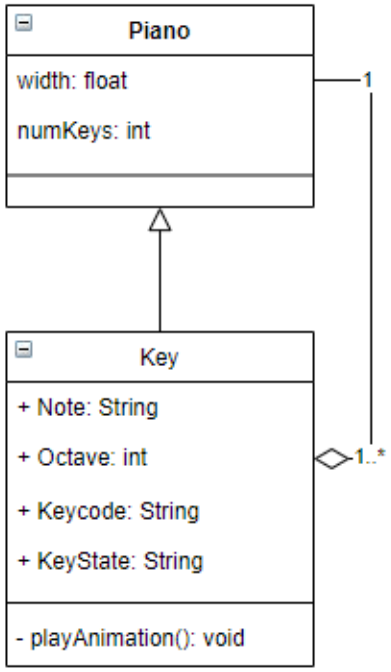
Detailed Design Models and Design Patterns

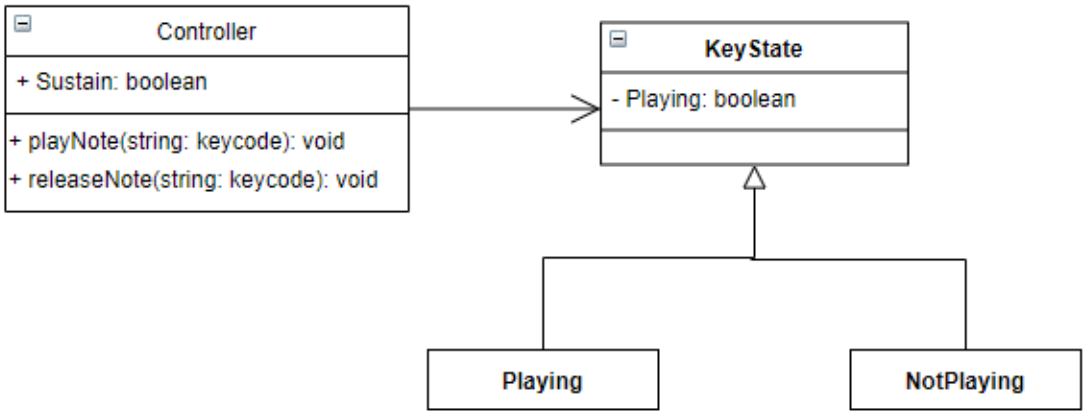
Design Patterns

Structural Pattern: Composite



The composite pattern was chosen because we have several different objects that need to be treated identically on our webpage. We have 2 different types of keys, white and black keys. The white keys are normal notes, while the black keys are sharps/flats. The keys need to be treated the exact same way when they are played, but they need to have a different value for their octave and note name, which the sound to be played are looked up from. By treating all these keys the same way, the coding will be much simpler and more readable.

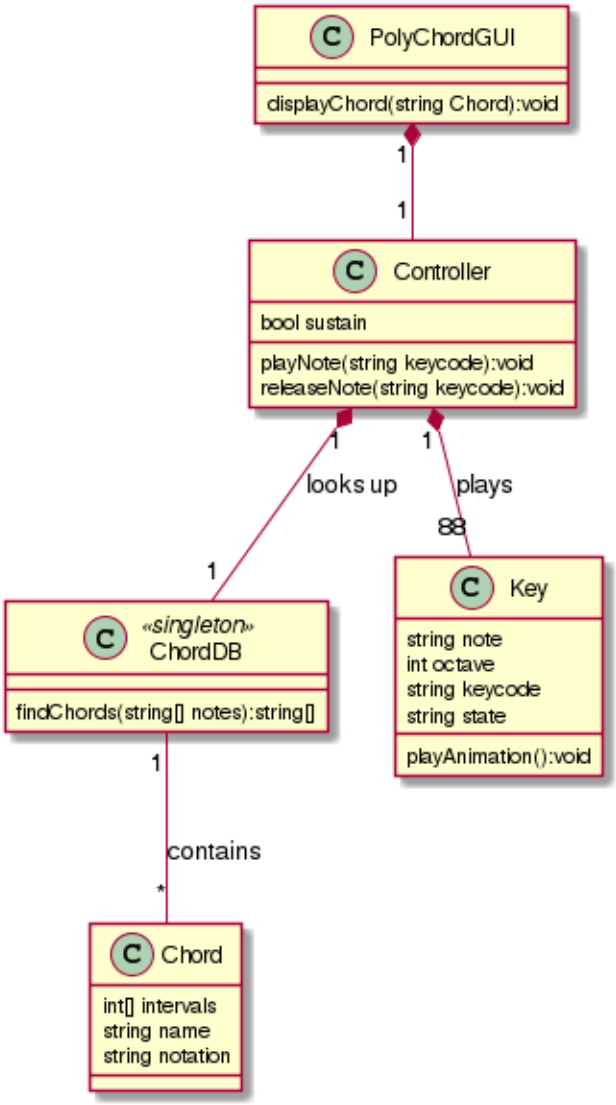
Behavioral Pattern: State



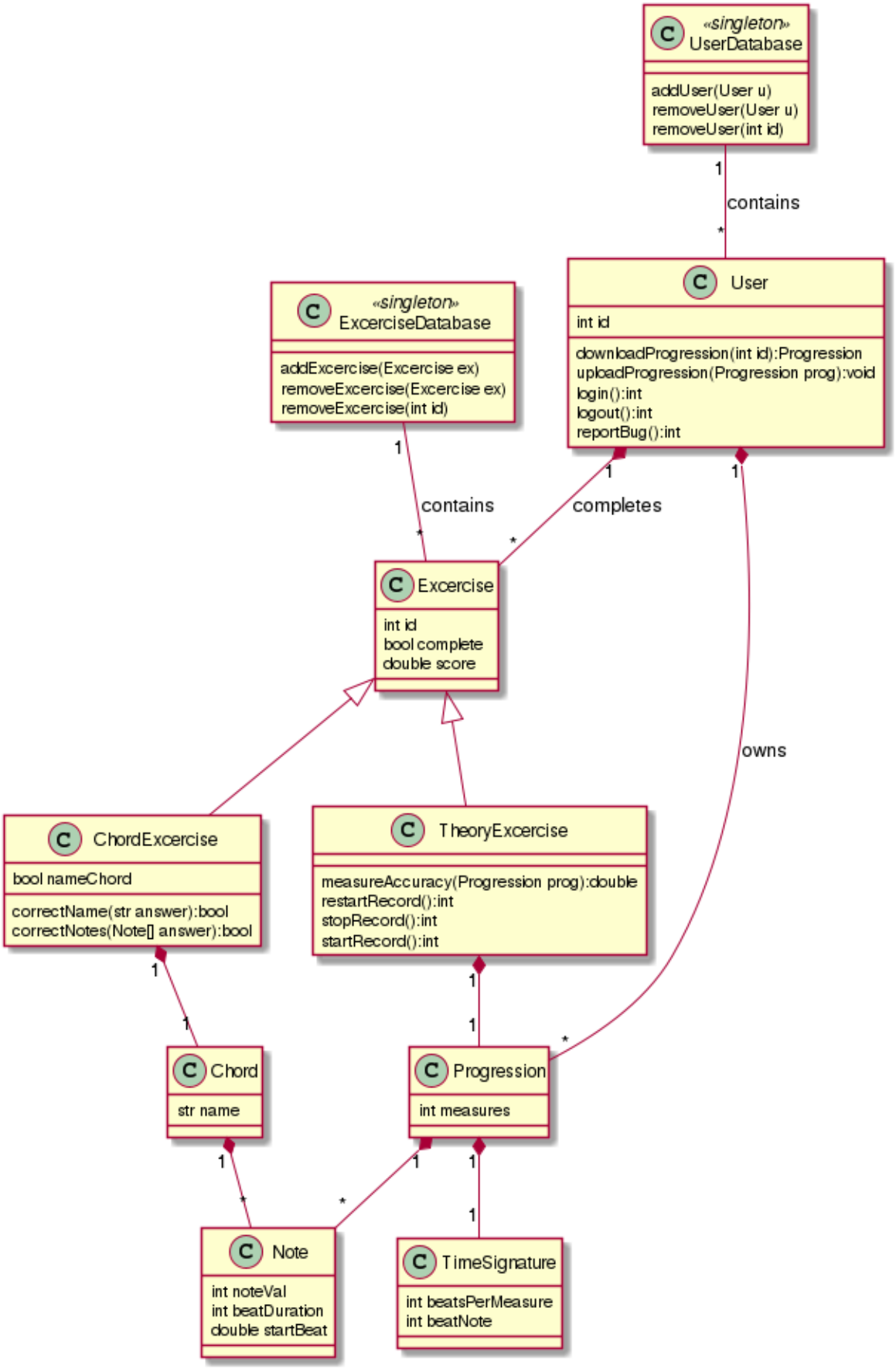
The State behavioral pattern is useful for our application becuase each key has two states, playing and notplaying. The controller will receive the command to play a certain note, and it needs to know if it is already playing or not so it can be played if it isn't, or not start playing it again it if it already playing. If a note is already playing and is played again, it would layer on top, potentially causing discomfort to the user and it might not be able to be stopped.

Design Class Diagrams

Detailed Design: Playing the Piano



Detailed Design: Users and Progression


«singleton»
UserDatabase
addUser(User u)
removeUser(User u)
removeUser(int id)
1
contains
*
User
int id
downloadProgression(int id):Progression
uploadProgression(Progression prog):void
login():int
logout():int
reportBug():int
«singleton»
ExcerciseDatabase
addExcercise(Excercise ex)
removeExcercise(Excercise ex)
removeExcercise(int id)
1
contains
*
1
completes
*
1
owns
Excercise
int id
bool complete
double score
ChordExcercise
bool nameChord
correctName(str answer):bool
correctNotes(Note[] answer):bool
TheoryExcercise
measureAccuracy(Progression prog):double
restartRecord():int
stopRecord():int
startRecord():int
1
1
1
1
*
Chord
str name
Progression
int measures
1
*
1
*
1
1
1
Note
int noteVal
int beatDuration
double startBeat
TimeSignature
int beatsPerMeasure
int beatNote