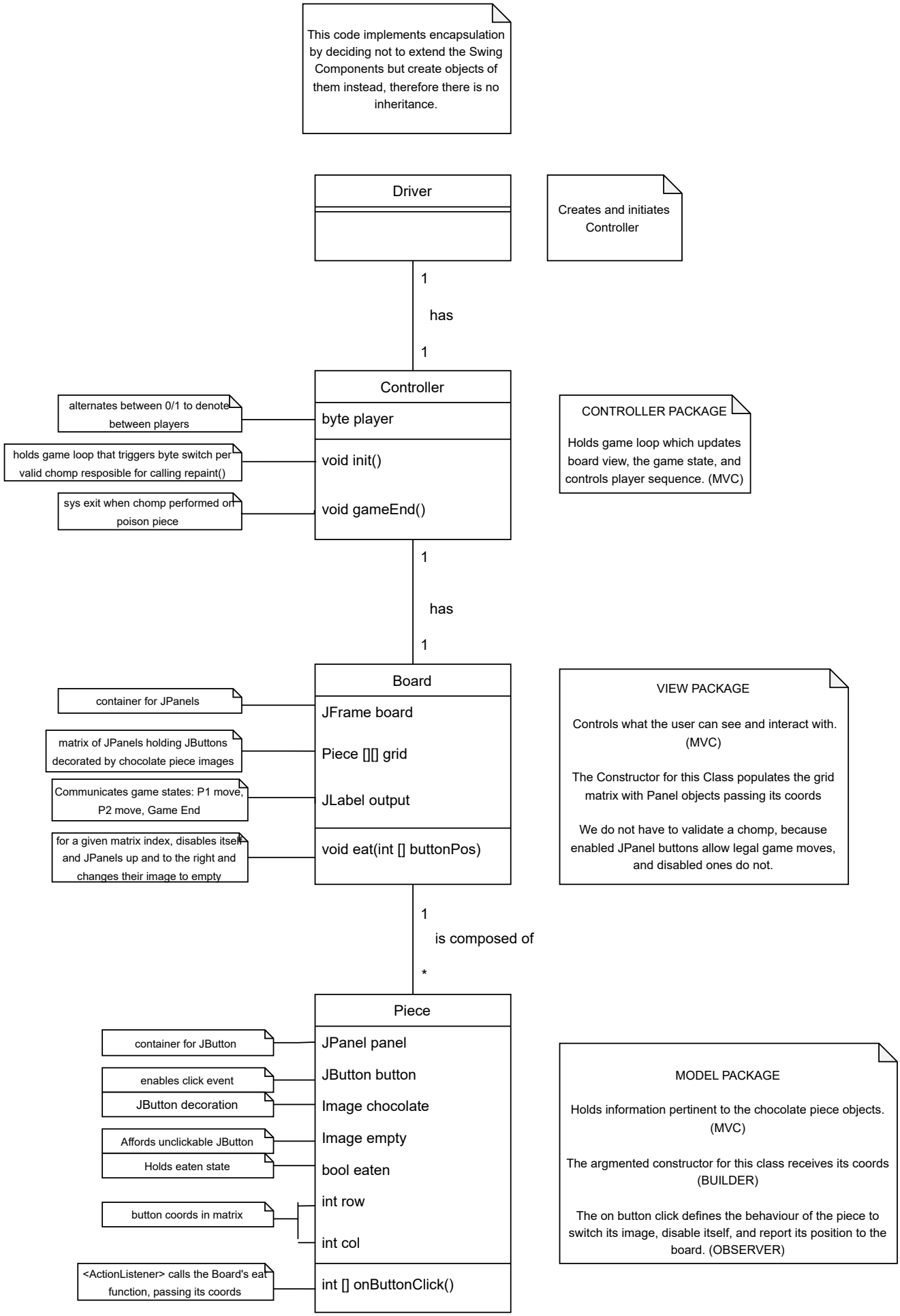
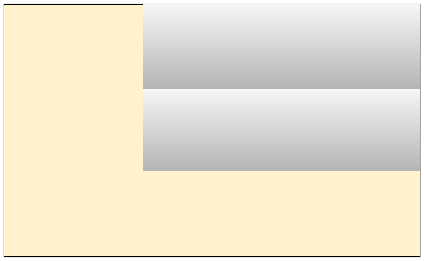

This code implements encapsulation by deciding not to extend the Swing Components but create objects of them instead, therefore there is no inheritance.
Driver
Creates and initiates Controller
1
has
1
Controller
alternates between 0/1 to denote between players
byte player
holds game loop that triggers byte switch per valid chomp resposible for calling repaint()
void init()
sys exit when chomp performed on poison piece
void gameEnd()
CONTROLLER PACKAGE
Holds game loop which updates board view, the game state, and controls player sequence. (MVC)
1
has
1
Board
container for JPanels
JFrame board
matrix of JPanels holding JButtons decorated by chocolate piece images
Piece [][] grid
Communicates game states: P1 move, P2 move, Game End
JLabel output
for a given matrix index, disables itself and JPanels up and to the right and changes their image to empty
void eat(int [] buttonPos)
VIEW PACKAGE
Controls what the user can see and interact with. (MVC)
The Constructor for this Class populates the grid matrix with Panel objects passing its coords
We do not have to validate a chomp, because enabled JPanel buttons allow legal game moves, and disabled ones do not.
1
is composed of
*
Piece
container for JButton
JPanel panel
enables click event
JButton button
JButton decoration
Image chocolate
Affords unclickable JButton
Image empty
Holds eaten state
bool eaten
button coords in matrix
int row
int col
<ActionListener> calls the Board's eat function, passing its coords
int [] onButtonClick()
MODEL PACKAGE
Holds information pertinent to the chocolate piece objects. (MVC)
The argmented constructor for this class receives its coords (BUILDER)
The on button click defines the behaviour of the piece to switch its image, disable itself, and report its position to the board. (OBSERVER)

WIRE FRAME

PLAYER 1 TURN

1. Game starts player 1 move

PLAYER 2 TURN

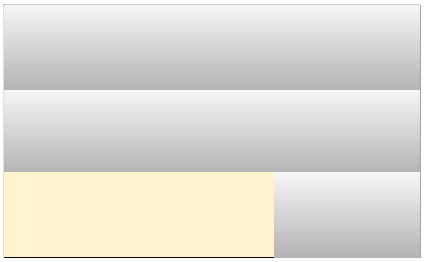


2. Player 1 clicked on (1,1)
Player 2 move

PLAYER 1 TURN

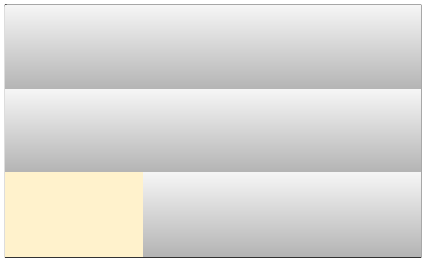
2. Player 2 clicked on (0,1)
Player 1 move

PLAYER 2 TURN



2. Player 1 clicked on (2,2)
Player 2 move

PLAYER 1 TURN



2. Player 2 clicked on (1,2)
Player 1 move

GAME OVER! PLAYER 2 WINS

2. Player 1 clicked on (0,2)
Game Over