

Clinton wrote the code for the UI interface, so Joanna will be doing code review for the UI interface. Clinton will be doing some code review for the enemy pathfinding and some collision logic.

Finding	Detail explanation	Potential solution
bad/confusing variable names	<p>In UI.java, there is a method called “public void draw(Graphics2D g2)” which is drawing a different user interface according to the user’s choice. This can be confusing to viewers because we have the method called “draw()” for other game characters.</p> <p>Method search() in Pathfinder.java is somewhat vague. Maybe something like searchForPath() or getBestPath()?</p>	<p>It would be better if the naming of the method can be more specific. Modification:</p> <pre>public void draw(Graphics2D g2)</pre> <p>-></p> <pre>public void drawGameStatus(Graphics2D g2)</pre> <p>Renamed to searchForPath() for less ambiguity</p>
Code duplication	I noticed there are a lot of duplications for the set up of the text message. In every single screen print method, x and y are being initialized again and again.	Initialize x and y as public variables, and create a method for setting up the parameter for displaying the text messages. See <pre>public void setFontAndColor(Font font, Color color)</pre>
Lack of documentation	<p>In the method <pre>public UI(GamePanel gp)</pre> is unclear what is actually getting displayed. I thought it’s supposed to display the main menu but it’s actually the in-game interface.</p> <p>The Pathfinder and Node classes are a bit unclear about what the purpose of the node initialization and searching is. Maybe add some comments to better describe its functionality?</p>	<p>Added comments to better explain the method.</p> <p>Brief summary of Pathfinder and Node classes added, and a short explanation of what both classes’ primary purposes are in the overall functionality of the program.</p>
unused or useless variables	2 methods were never used <pre>public void drawSubWindow(int x, int y, int width, int height)</pre> <pre>public void helpScreen()</pre>	These 2 methods can be removed. Game runs as expected without these 2 methods.
Unclear/unused variable	<p>In the method <pre>public void pauseScreen()</pre>, we don’t need the object g2 for anything. It’s just there and defined for no reason.</p> <pre>if (openList.size() == 0)</pre> <p>Should be using isEmpty() instead as openList is an ArrayList</p>	<p>Simply remove the line <pre>this.g2 = g2;</pre> The method can just use the public defined object g2.</p> <p>While isEmpty() effectively performs the same logic as size() == 0, it is more easily understandable.</p>

	object.	
Classes/methods that are too large and/or try to do too much	<pre>public void drawTitle()</pre> <p>This method is a bit too long and can be better if we divide this into smaller segment of codes</p> <p>PathFinder's searchForPath() function could be modularized a bit further.</p>	<p>Divided into 3 section:</p> <ol style="list-style-type: none"> 1. Displaying game title with the desired font and art effect 2. <pre>public void drawBunnyOnTitle()</pre> 3. <pre>public void drawMenuOptions()</pre> <p>searchForPath() has been divided further into scanListForBestNode()</p>
Unsafe code	<pre>public void instantiateNodes()</pre> <p>The above method could be private instead as it is only used in the PathFinder class.</p>	<p>Most of the methods used in PathFinder may be more fit to be private instead of public.</p>
Unnecessary if/else or switch/case statements	<p>Some conditions in searchShortestPath() in the Object.java class could be combined. It is somewhat difficult to follow the logic currently.</p>	<p>Combined some else-if statements in searchShortestPath().</p>