A lot of the bugs are associated with collision detection. These are largely legacy bugs from Iteration1, as fixing them turned out to be too large a task. However, since lots of the functionality in priority 3 of Iteration2 depends on collisions, they actually introduced new bugs. First I start with bugs relating to “Player” (formerly Robot), as these bugs are largely the same as before.

1. Since the heading angle of the colliding entity is not being used in calculations, collision calculation is inherently flawed (not angle of incidence). This is due to the way arguments are set up in Arena::CheckForEntityCollision.
2. For the Player and HomeBase, collisions off obstacles (not walls) look reasonable about 30% of the time for the robot. It seems to depend on the angle of approach (and definitely on the speed; higher speeds work better), but I can't seem to determine a rule for when it will work.
3. Since Player slows down after collisions, it tends to get stuck in objects when colliding at low speeds (reasonable collisions can sometimes be seen at higher speeds).
4. HomeBase can not be set to change angle often (randomly), because it overrides the angle set by the collision. While this happens rarely (HomeBase does not change angle often currently), it still occurs on occasion.
5. While wall collisions tend to work well for Player and HomeBase, coming in at angles close to parallel can cause mobile entities to get stuck in the wall.
6. Sometimes when stuck the Player can escape with repeated speed up commands, but other times it gets permanently trapped.

Other bugs include:

1. The heading angle doesn't initialize to the same value as restart sets it to.
2. The angle of the text on the Player does not correspond to the actual heading angle. I'd prefer if it didn't rotate at all but that hasn't been tried either as of right now.

There are additional bugs associated with Robots and SuperBots:

1. Most of the time when a Robot collides with the HomeBase, it turns into a SuperBot. However, sometimes this doesn’t happen. Furthermore, sometimes when a Robot turns into a SuperBot, another Robot does as well, even though it is not touching the HomeBase.
2. Robots and SuperBots only collide properly with certain walls. Other walls seem to cause strange behaviors.

3. Because of the erratic nature of the Robot and SuperBot movement, it is not clear if frozen Robots unfreeze when touched by other Robots (I simply can't test this). I believe they should, but I can't know for sure.
4. While Robots do seem to freeze when touched by a Player, the game is almost instantly lost afterwards because Player gets stuck inside of the Robot.
5. When Robots collide with each other or with HomeBase, they tend to glide together because they're stuck.

Finally, these are the requirements of Iteration2 that could not be satisfied, either because implementing them caused tricky bugs, or because I simply didn't have time.

1. If the Player touches the HomeBase the game is won, even though the winning conditions have changed in this iteration.
2. SuperBots do not freeze the player even though code is in place to facilitate this. I can not figure out why it doesn't work.
3. Code for avoiding obstacles using proximity sensors had to be commented out because it caused Robots to accelerate towards infinity.
4. Since avoiding obstacles doesn't work, SuperBots technically satisfy the requirement that they should not avoid the Player… but only technically. In reality, doing this the correct way would entail quite a bit more work.
5. The game is not lost when all Robots are SuperBots and the game is not won when all Robots are frozen.