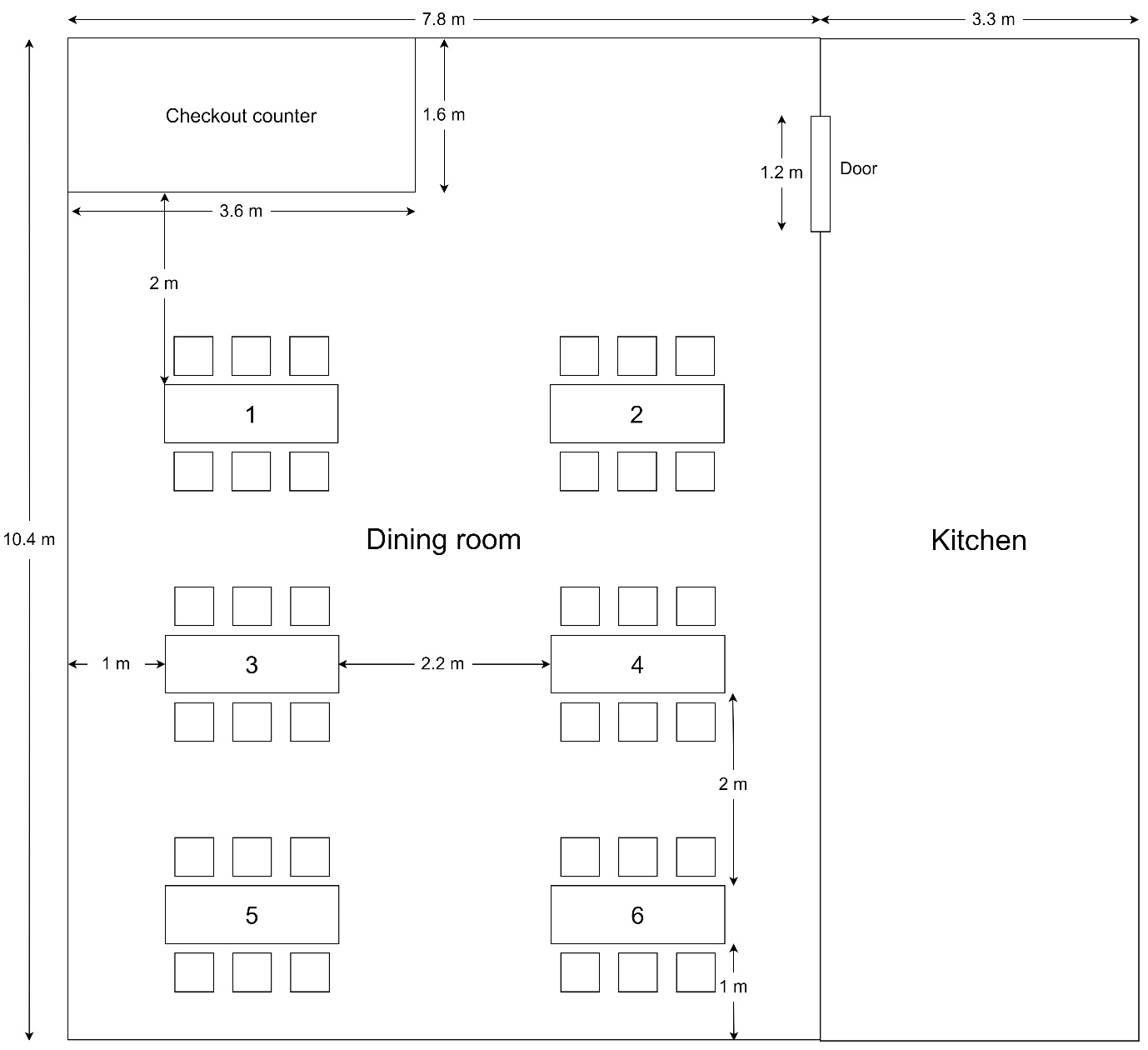
**Use case of optimization of energy and serving time for Sevibot:**



|  |  |  |  |
| --- | --- | --- | --- |
| 1. The kitchen staff puts the first dish on the tray and determines the corresponding table (same goes for the next dishes) | | | |
| 2. Robot initiates timer (2 minutes) | | | |
| 3.1. Not enough 3 dishes and timer finishes counting | 3.2. There are 3 dishes to put in 3 trays | | 3.3 The kitchen staff presses “Start” button |
| 4. Robot comes from the kitchen to the dining room | | | |
| 5. Robot stops at the nearest table in the order list | | | |
| 6. Robot initiates a timer to wait for guests to take the dishes (30s) | | | |
| 7.1. The dishes are taken | | 7.2. The dishes are not taken and timer finishes counting | |
| 8.1.1. There still exists table(s) need serving | | 8.2.1. There are not any tables need serving | |
| 8.1.2. Robot moves and serves the nest nearest table in the order list | |  | |
| 9. Robot goes back to the kitchen | | | |

Additional function:

- There is a button to command the robot to go back to the kitchen immediately

- Warn the kitchen staff when the robot is not able to move (switch to manually control mode)