

1. Skapa nya interfaces, en som heter ClickListener som ska hantera knapptryck.
2. Byt namn på interfacet LoadableObject till VehicleLoader för att ge en bättre representation av vad klassen gör.
3. Skapa ett nytt interface, LoadableObject som ska implementeras i Car och LongTruck. Detta är för att få bort kompositionen som fanns mellan Car och Longtruck.
4. Dela upp VehicleController i flera klasser, **VehicleManager**, **TimerListener**.
5. VehicleManager ska implementera interfacet ClickListener, som behandlar knapptryck. Den ska även innehålla en instans av TimerListener, samt även hålla en instans av Vehicle.
6. Dela upp VehicleView till två klasser, Button Logic och Button Visuals.
7. Button Visuals ska innehålla den visuella delen från VehicleView.
8. ButtonLogic ska implementera ClickListener, samt ha en komposition av ButtonVisuals för att kunna prata med VehicleManager. Denne lösning använder Dependency Inversion Principle, vilket undviker den tidigare direkta kopplingen mellan VehicleController och Vehicle.
9. Även DrawPanel ska delas upp i två klasser, en klass som skapar ram, Frame, och en som uppdaterar ramen, UpdateFrame.
10. Frame ska hämta bilderna, och skapa själva ramen som kommer användas.
11. UpdateFrame, kommer först skapa ramen och sedan ha en funktion som uppdaterar bildernas position och uppdaterar det visuella.