|  |
| --- |
|  |
| AnimalFarm |
| OOP-Projekt des m320 Unterrichts |

|  |
| --- |
|  |



Inhalt

Seitenzahl

[Vorwort 2](#_Toc155873056)

[Klassendiagram (vor Projektbeginn) 3](#_Toc155873057)

[Klassendiagram (UML-Exported) 4](#_Toc155873058)

[Sequenzdiagramm 5](#_Toc155873059)

[Projekt Arbeit 6](#_Toc155873060)

[Zeitaufwand 6](#_Toc155873061)

# Vorwort

Ich habe versucht eine möglichst Frei verwendbare Animal(Tier) Farm zu machen. Im Grunde wollte ich eine einfach OOP-Projektlösung die mind. ein Interface beinhaltet und mind. eine Baseklasse hat. Diese zwei Kriterien konnte ich mit dieser Lösung erfüllen.

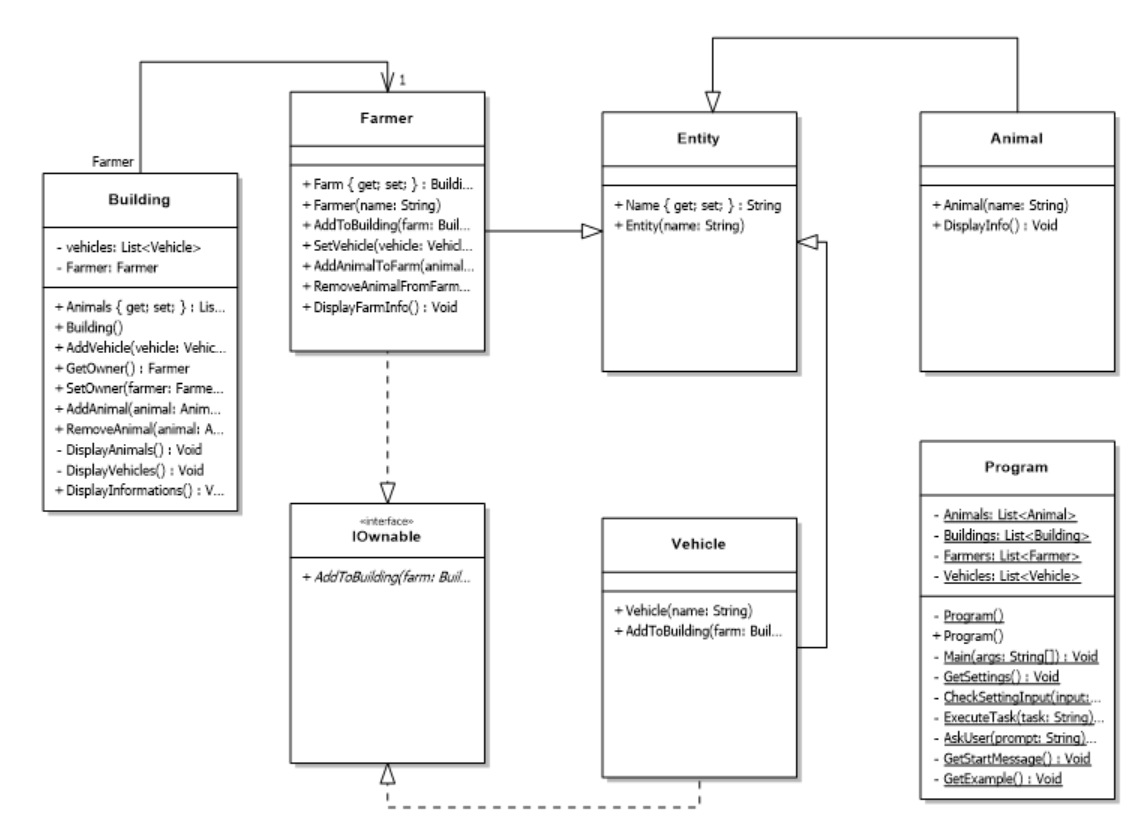
In meiner Applikation geht es darum das man entweder eine schon fertige Lösung (example) öffnen kann und dann sieht wie das ganze aussehen sollte wenn man es von hand macht (natürlich in der Konsole) oder eine ganz eigene Lösung erstellen die variabel sein kann. In der eigenen Lösung wird man durch die Konsole geleitet wie man einen Farmer erstellen kann und Tiere in den Bauernhof leiten. Durch diese einfach Lösung wurde das OOP-Prinzip sehr klar und einfach dargestellt.

# Klassendiagram (vor Projektbeginn)

Ein Bild, das Text, Diagramm, Plan, Reihe enthält.

Automatisch generierte Beschreibung

# Klassendiagram (UML-Exported)



# Sequenzdiagramm

…

# Projekt Arbeit

## Zeitaufwand

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task | Geschätzter Zeitaufwand | Realer Zeitaufwand | Differenz | Erledigt am |
| Initial Commit   * Alle Klassen erstellen und Grundoperationen in Klassen einbauen | 1h | 1.25h  ~1h 15min | 1.25h – 1h = 0.25h  0.25h = ~15min | 1. Januar 2024 |
| Add Driveable Cars   * Klasse Vehicle erstellen mit Grundoperationen und in bestehende Klassen implementieren | 0.75h | 1.17h  ~1h 10min | 1.17h – 0.75h = 0.42h  0.42h = ~25min | 2. Januar 2024 |
| Added console commands and prompts to code   * Applikation von Ablauf in Main zu Konsolen gesteuert migrieren | 3h | 5.5h  ~5h 30min | 5.5h – 3h = 2.5h  2.5h = ~2h 30min | 5. Januar – 6. Januar 2024 |
| Add Unittest   * Unittest implementieren | 1h | 1.45h  ~1h 27min | 1.45h – 1h = 0.45h  0.45 = ~27min | 8. Januar 2024 |
| Create Docs   * Erstellen von diesem File | 0.5h | 0.6h  ~36min | 0.6h – 0.5h = 0.1h  0.1h = ~6min | 11. Januar 2024 |