

Vielen Drache

High-Level & Low-Level Design

By Small-Scale Development

Table of Contents

Architecture.....	2
→ Monolithic Architecture.....	2
→ Component-Based Architecture.....	2
→ Event-Driven Architecture.....	2
Diagrams.....	3
→ Class Diagrams (UML):.....	3
→ Gameplay Data Flow/Activity Diagram:.....	4
Database Design.....	5
→ Database.....	5
Required Specifications.....	5
→ Hardware.....	5
→ Software.....	5
User Interface.....	6
→ Main Menu.....	6
→ Pause Menu.....	10
→ HUD (Head's-Up Display).....	11
→ Hub World (/ Tutorial).....	12
Other Interfaces.....	12
→ Hardware.....	12
→ Video Output.....	13
→ Audio Output.....	13
Security.....	13
→ Login/Access.....	13
→ Logout/Exit.....	13
Reports.....	13
→ In-Game.....	13
Other Outputs.....	13
→ Save Data.....	13

Architecture

→ *Monolithic Architecture*

Vielen Drache is a small, self-contained video game, meaning the architecture doesn't need to connect to an external server. Once the application is downloaded, the user can complete every interaction through a single application.

→ *Component-Based Architecture*

The Unity game engine naturally gravitates towards component-based architectures. In Vielen Drache, objects like the Player, Enemies, Bosses, all

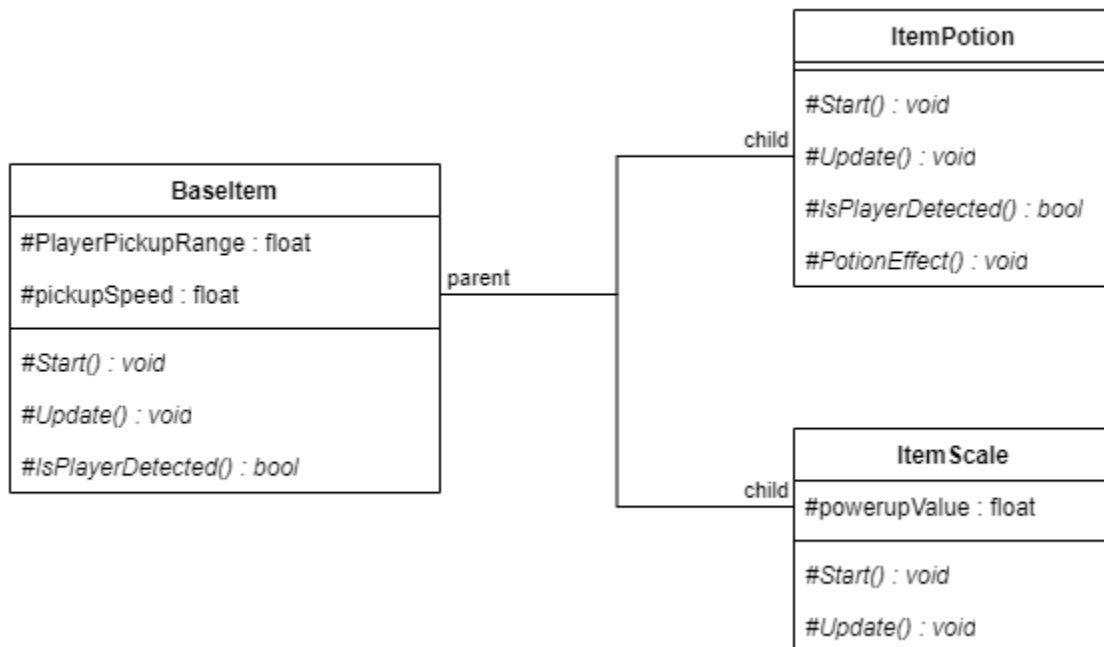
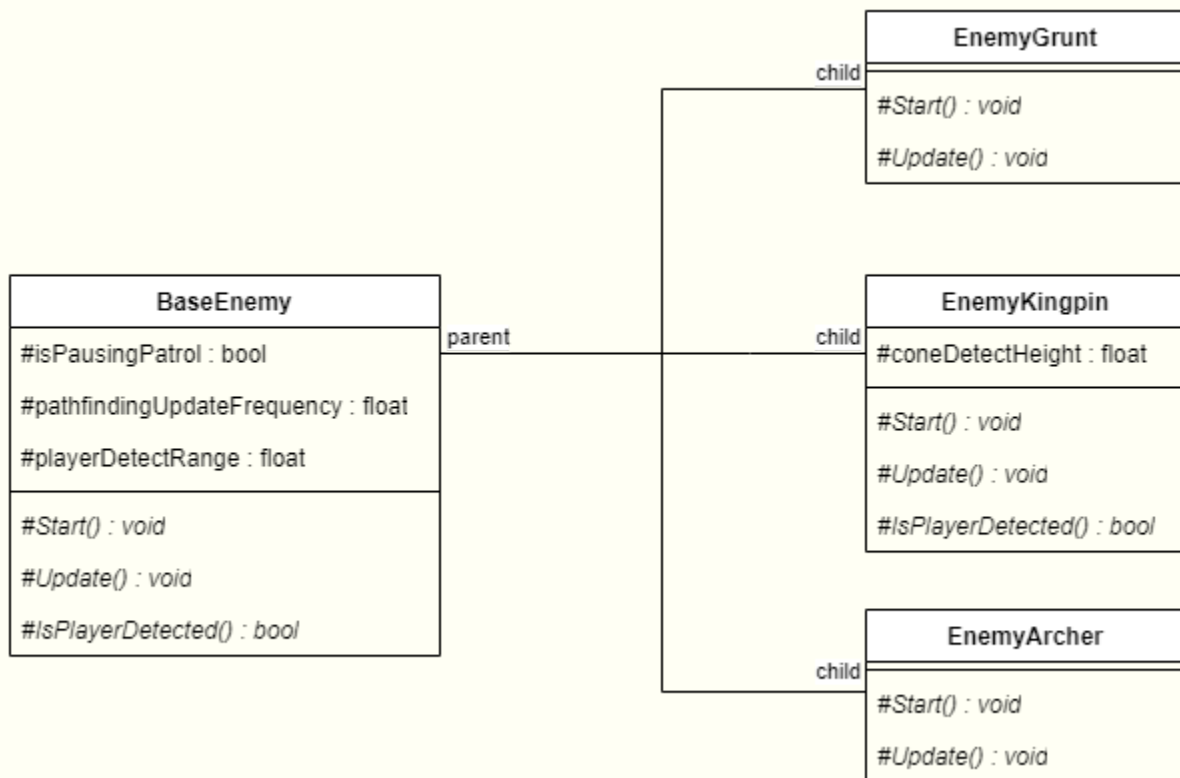
→ *Event-Driven Architecture*

Unity games often use Events and Event Listeners to handle buttons/UI interactions in-game. We may also have need for Events in situations where objects should not have direct access/control over one another, but still need to communicate important information. Examples:

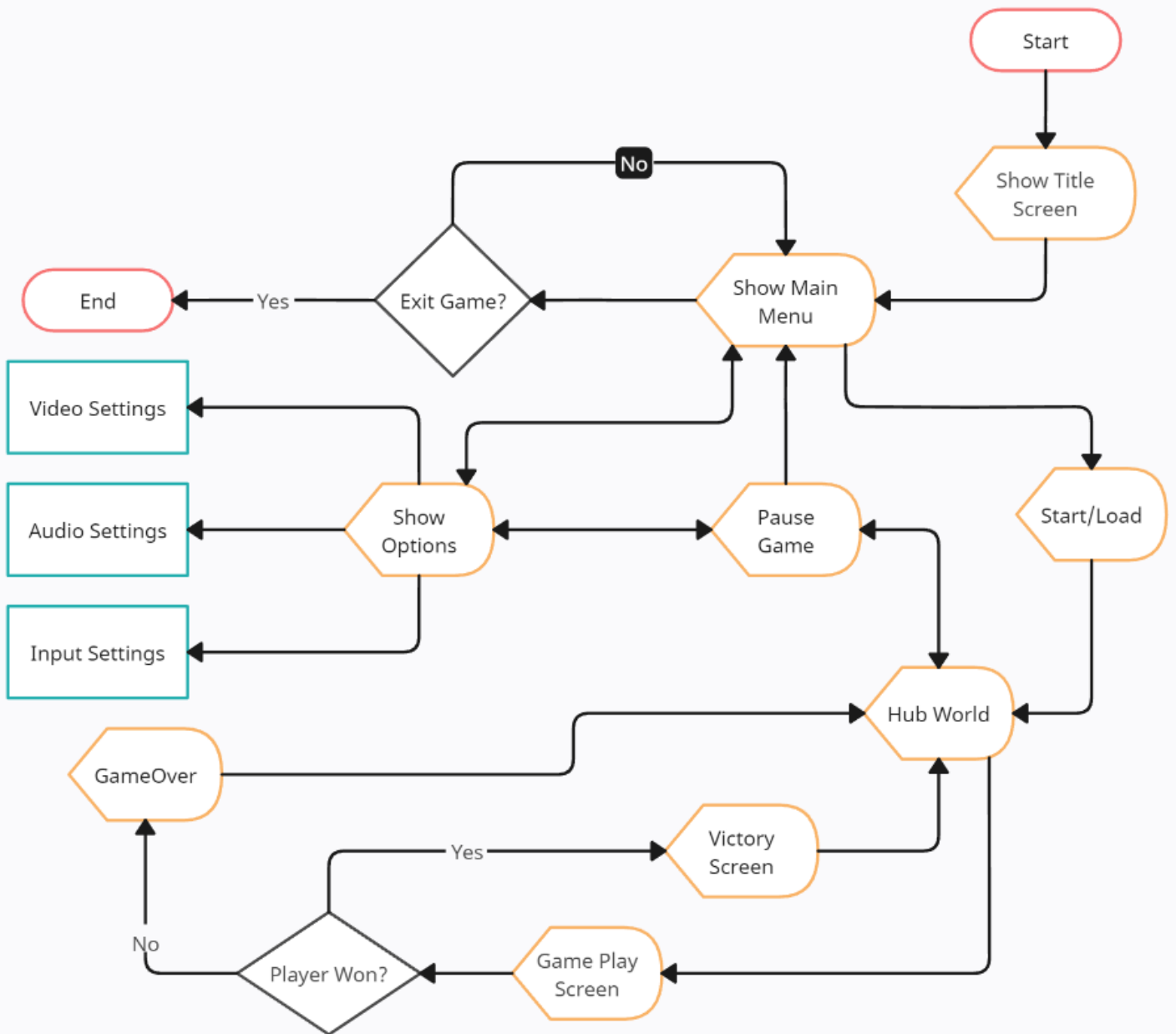
- Players DMGing enemies.
- Enemies DMGing players.
- Updating player Health display when player health changes.
- Updating player Powerup bar when the player picks up a coin.

Diagrams

→ *Class Diagrams (UML):*

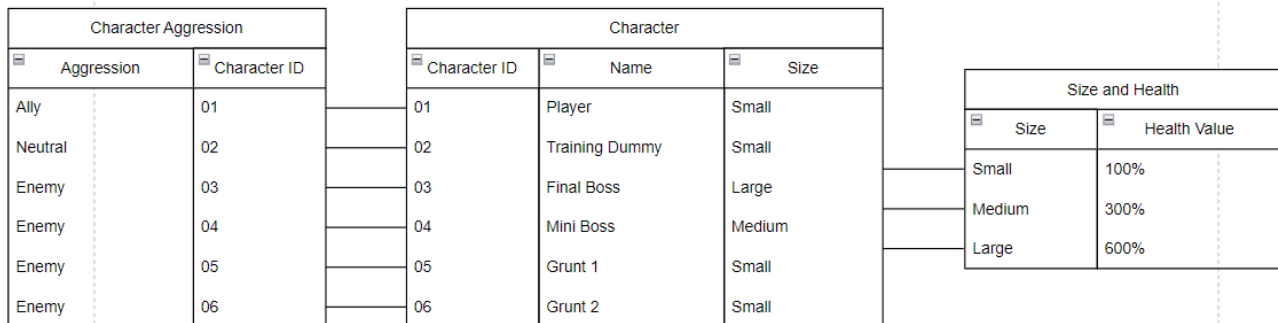


→ **Gameplay Data Flow/Activity Diagram:**



Database Design

→ Database



Required Specifications

→ Hardware

Minimum Hardware Requirements:

- CPU:
 - 2GHz dual-core processor or better
- RAM:
 - 4GB DDR3 system memory or better
- Disk Space:
 - 8GB available storage
- Screen Resolution:
 - (HD) 1440 x 900 or more

→ Software

Minimum Software Requirements:

- Windows:
 - Operating System:
 - ◆ Windows 10 version 21H1 (build 19043) or newer
 - Graphics API:
 - ◆ DirectX10-capable GPU
 - ◆ DirectX11-capable GPU
 - ◆ DirectX12-capable GPU
 - Other:

- ◆ Hardware vendor officially supported drivers
- ◆ For development: IL2CPP scripting backend requires:
 - Visual Studio 2019 with C++ Tools component or later
 - Windows SDK version 10.0.19041.0 or newer
- macOS:
 - Operating System:
 - ◆ macOS Big Sur 11.0+
 - Graphics API:
 - ◆ Metal-capable Intel GPU
 - ◆ Metal-capable AMD GPU
 - Other:
 - ◆ Apple officially supported drivers
 - ◆ For development: IL2CPP scripting backend requires Xcode.
 - ◆ Targeting Apple Silicon with IL2CPP scripting backend requires:
 - macOS Catalina 10.15.4 or newer
 - Xcode 12.2 or newer
- Linux:
 - Operating System:
 - ◆ Ubuntu 20.04, Ubuntu 22.04, and CentOS 7
 - Graphics API:
 - ◆ OpenGL 3.2+ capable GPU
 - ◆ Vulkan capable GPU
 - Other:
 - ◆ Gnome desktop environment running on top of X11 windowing system
 - ◆ Other configurations and user environments as provided stock with the supported distribution (such as Kernel or Compositor)
 - ◆ Nvidia GPU using Nvidia's official proprietary graphics driver
 - ◆ AMD GPU using AMD Mesa graphics driver

User Interface

→ Main Menu

New Game

- This is how you can create and begin playing on a new save game.
- Always available/never greyed out.

Continue

- This is a bypass of the following load option to continue in the most recently played save game.
- Greyed out if there are no available saved games to continue in.

Load

- This enables the user to view all saved games and allows them to select the save that they wish to play. This is integral to allowing the user to have multiple different saves on a single computer.
- Greyed out if there are no available saved games to load into.

Options

- These are options that the user can change to help tweak their experience to perfection and maximize enjoyment of the game.
- Always available/never greyed out.
- **Video Settings** (The look of the application)
 - **Windowed**
 - ◆ Allows the application to run as a window, which helps with ease of switching between applications.
 - **Borderless**
 - ◆ A hybrid between true fullscreen and windowed, borderless allows the application to run as a fullscreen view but as a window, removing minimize/maximize, fit, and close buttons for a cleaner view.
 - ◆ When matched to the native resolution of the display, borderless will appear to be just like fullscreen, with the added benefit of a more stable/easier time tabbing in and out of the application.
 - **Fullscreen**
 - ◆ Allows the application to run in fullscreen, which fits and fills the display following application settings. This isolates the application from the desktop/command-line interfaces, allowing a user to much easier provide full attention to the application.
 - **Back**
 - ◆ Return to the previous menu screen.
- **Audio Settings** (The sound of the application)
 - **Main**
 - ◆ A slider to adjust the overall percentage of audio output from the application.
 - ◆ 0% - 100%
 - **Ambience**

- ◆ A slider to adjust the percentage of audio output categorized as ambient noise from the application.
 - ◆ 0% - 100%
- **SFX**
 - ◆ A slider to adjust the percentage of audio output categorized as SFX from the application.
 - ◆ 0% - 100%
- **Music**
 - ◆ A slider to adjust the percentage of audio output categorized as music from the application.
 - ◆ 0% - 100%
- **Back**
 - ◆ Return to the previous menu screen.
- **Input Settings** (Will allow the user to tweak/set input key binds or alter the controller control scheme.)
 - **Keybindings**
 - ◆ **Move Left**
 - | Primary Keybind | ← (*Default binding preset*)
 - | Secondary Keybind | ← (*Default binding preset*)
 - | Tertiary Keybind|
 - (Rebind Option)
 - ◆ **Move Right**
 - | Primary Keybind | ← (*Default binding preset*)
 - | Secondary Keybind | ← (*Default binding preset*)
 - | Tertiary Keybind|
 - (Rebind Option)
 - ◆ **Jump**
 - | Primary Keybind | ← (*Default binding preset*)
 - | Secondary Keybind | ← (*Default binding preset*)
 - | Tertiary Keybind| ← (*Default binding preset*)
 - (Rebind Option)
 - ◆ **Interact**
 - | Primary Keybind | ← (*Default binding preset*)
 - | Secondary Keybind |
 - | Tertiary Keybind|
 - (Rebind Option)
 - ◆ **Attack**

- | Primary Keybind | ← *(Default binding preset)*
- | Secondary Keybind |
- | Tertiary Keybind|
- (Rebind Option)

◆ Pause

- | Primary Keybind | ← *(Default binding preset)*
- | Secondary Keybind |
- | Tertiary Keybind|
- (Rebind Option)

○ **Controller Button Mapping (Xbox Controller)**

◆ Move Left

- | Primary Mapping | ← *(Default binding preset)*
- | Secondary Mapping |
- | Tertiary Mapping|
- (Remap Option)

◆ Move Right

- | Primary Mapping | ← *(Default binding preset)*
- | Secondary Mapping |
- | Tertiary Mapping|
- (Remap Option)

◆ Jump

- | Primary Mapping | ← *(Default binding preset)*
- | Secondary Mapping |
- | Tertiary Mapping|
- (Remap Option)

◆ Interact

- | Primary Mapping | ← *(Default binding preset)*
- | Secondary Mapping |
- | Tertiary Mapping|
- (Remap Option)

◆ Attack

- | Primary Mapping | ← *(Default binding preset)*
- | Secondary Mapping |
- | Tertiary Mapping|
- (Remap Option)

◆ Pause

- | Primary Mapping | ← (*Default binding preset*)
- | Secondary Mapping |
- | Tertiary Mapping|
- (Remap Option)

- **Back**

- ◆ Return to the previous menu screen.

Exit

- This allows the user to exit the application from inside the application because there is a good chance that the application will be in fullscreen mode, in which exiting without a menu option is not a user-friendly process.
- Always available/never greyed out.

→ Pause Menu

Resume

- This is how you can unpause the game once it has been paused.
- Always available/never greyed out.

Save

- This enables the user to save their game (start of the current level) so that they can continue at a later time.

Options

- These are options that the user can change to help tweak their experience to perfection and maximize enjoyment of the game.
- Always available/never greyed out.
- Video Settings (The look of the application)
 - Windowed
 - ◆ Allows the application to run as a window, which helps with ease of switching between applications.
 - Borderless
 - ◆ A hybrid between true fullscreen and windowed, borderless allows the application to run as a fullscreen view but as a window, removing minimize/maximize, fit, and close buttons for a cleaner view.
 - ◆ When matched to the native resolution of the display, borderless will appear to be just like fullscreen, with the added benefit of a more stable/easier time tabbing in and out of the application.
 - Fullscreen

- ◆ Allows the application to run in fullscreen, which fits and fills the display following application settings. This isolates the application from the desktop/command-line interfaces, allowing a user to much easier provide full attention to the application.
- Audio Settings
 - Main
 - ◆ A slider to adjust the overall percentage of audio output from the application.
 - ◆ 0% - 100%
 - Ambience
 - ◆ A slider to adjust the percentage of audio output categorized as ambient noise from the application.
 - ◆ 0% - 100%
 - SFX
 - ◆ A slider to adjust the percentage of audio output categorized as SFX from the application.
 - ◆ 0% - 100%
 - Music
 - ◆ A slider to adjust the percentage of audio output categorized as music from the application.
 - ◆ 0% - 100%
- Input Settings
 - Will allow the user to tweak/set input key binds or alter the controller control scheme.

Save and Exit to Main Menu

- This enables the user to save their game (start of the current level) so that they can continue at a later time as well as allows the user to return to the main menu from the pause menu, saving game progress (start of levels)
- Always available/never greyed out.

→ HUD (Head's-Up Display)

Health Displays

- A visual representation of remaining player health will be present at all times in the top left hand corner.
- If applicable, enemies (bosses and mini-bosses) will have a healthbar shown statically at the top of the screen.

Point Total Display

- A visual representation of the player's score will be displayed in the top middle of the screen. This number, while calculated based on how many

point-generating actions the player performs, is really just arbitrary and will not halt or progress the game in any way.

Time/Stopwatch Display

- A visual representation of elapsed playtime will be present at all times in the top right hand corner while in levels. Time elapsed is chosen over a timer enforcing a limit so as to be a rather relaxing platformer.

→ Hub World (/ Tutorial)

Hub World

- The player will immediately begin in the tutorial area of the dragon's cave, and have the option to either complete the tutorial or enter a level.
- The player can play in the current level or a previous one. Other levels are locked until the one before it is complete.

Other Interfaces

→ Hardware

Keyboard Controls (Defaults as Example)

- Move Left
 - | A | Left Arrow Key |
- Move Right
 - | D | Right Arrow Key |
- Jump
 - | W | Up Arrow Key | Space Bar |
- Interact
 - | F |
- Attack
 - | E |
- Pause
 - | Escape |

Controller Layout (Xbox) (Defaults as Example)

- Move Left
 - | Left Joystick Left |
- Move Right
 - | Left Joystick Right |
- Jump
 - | A |
- Interact

- | B |
- Attack
 - | X |
- Pause
 - | Start |

→ **Video Output**

Video output to the user's monitor depending on where the application window was located.

→ **Audio Output**

Audio output to the user's default audio output device.

Security

→ **Login/Access**

Open the executable, load the save you want to play, or make a new one.

→ **Logout/Exit**

Close the executable by the exit option of the main menu, or by force via the operating system (**not** advised).

Reports

→ **In-Game**

Visual feedback on if a player has taken damage or defeated an enemy. The application should signify that actions were completed by variation or spontaneous signal in general video output, such as animation or an in-game popup dialogue.

Audio feedback on if a player has taken damage or defeated an enemy. The application should signify that actions were completed by variation or spontaneous signal in audio output, such as a character-specific sound effect.

Other Outputs

→ **Save Data**

The application will store localized save data on a user's computer, with only simple terms to signify where the player/user had last left off. Basically, an integer variable for the current level that a player/user is on, and maybe a boolean variable to signify if they have reached a checkpoint in the level.