

Here’s how you can structure your project into multiple Markdown files based on the README content you provided. Each file corresponds to a specific aspect of your project:

Project Structure Overview

```
SUPER-MARIO-BROS-OOP/
├── docs/
│   ├── README.md           # Main entry point with project introduction
│   ├── structure.md        # Details about project structure
│   ├── features.md         # Highlights project features
│   ├── dependencies.md     # Information on dependencies
│   ├── how_to_run.md       # Steps to run the project
│   ├── oop_concepts.md     # Explanation of OOP concepts applied
│   ├── gameplay.md        # Information about gameplay and scenes
│   ├── assets.md           # Details about game assets
│   ├── contributing.md     # Guidelines for contributing
│   ├── license.md          # Project license
│   └── about.md            # About the project
```

File Templates

README.md


```
# SUPER MARIO BROS - OOP

Welcome to the **Super Mario Bros - OOP** project! This is a class project for
demonstrating object-oriented programming (OOP) principles in the context of game
development. The project implements a simplified version of the Mario game using a
structured and modular design.

Check out the documentation files in the `./docs/` directory for detailed
information.

---
- 📁 [Project Structure](docs/structure.md)
- 🚀 [Features](docs/features.md)
- ⚙️ [Dependencies](docs/dependencies.md)
- 🔑 [How to Run](docs/how_to_run.md)
- 📖 [OOP Concepts in Action](docs/ooop_concepts.md)
- 🎮 [Gameplay](docs/gameplay.md)
- 📁 [Assets](docs/assets.md)
- 💻 [Contributing](docs/contributing.md)
- 📄 [License](docs/license.md)
- 🧑‍💻 [About](docs/about.md)
---
```


structure.md

 Project Structure

Here is an overview of the directory structure and its purpose:


SUPER-MARIO-BROS-OOP/ | — .git/ # Git version control files | — dependencies/ # External dependencies and libraries | — Project1/ | — Assets/ # Game assets including sounds and textures | | — Sounds/ # Audio files for the game | | — Textures/ # Images and textures used in the game | | — Fonts/ # Fonts used in the game | — Game/ # Core game logic and components | | — Components/ # Reusable game components | | — Control/ # Input handling and game controls | | — Entities/ # Entity classes like Mario, enemies, etc. | | — Scenes/ # Game scenes | | — Utils/ # Utility functions and helper classes | — UI/ # Utility functions and helper classes

features.md

 Features

- **OOP Principles**: Encapsulation, inheritance, polymorphism, and abstraction.
- **Modular Design**: Organized components for scalability and maintenance.
- **Custom Assets**: Includes sound effects and textures.
- **Scene Management**: Multiple scenes for gameplay, menus, and settings.

dependencies.md


 Dependencies

Make sure you have the following dependencies installed:

- **Game Development Framework**: [Framework, e.g., SFML, Pygame]
- **Programming Language**: [Language, e.g., Python, C++]

Instructions for installing dependencies are in the `/dependencies/`` directory.

how_to_run.md

 How to Run

1. Clone this repository:

```
```bash
git clone https://github.com/your-repo/super-mario-bros-oop.git
```

### 2. Navigate to the project folder:

```
cd SUPER-MARIO-BROS-OOP/Project1
```

### 3. Build or run the project:

- [Provide specific build instructions, e.g., using `make` or an IDE]

```
`oop_concepts.md`
```markdown
# 📖 OOP Concepts in Action

- Encapsulation: Entities, utilities, and helpers are organized into specific folders.
- Inheritance: Shared functionality is abstracted for reuse.
- Polymorphism: Entities like Mario and enemies override shared methods.
- Abstraction: Interfaces or abstract classes enforce consistent design.
```

gameplay.md

🎮 Gameplay

The game includes the following scenes:

- **Home Scene**: Main menu.
- **Game Scene**: Core gameplay.
- **Settings Scene**: Adjust settings.
- **Exit Scene**: Exit the game.
- **Login Scene**: User login.
- **Register Scene**: User registration.
- **Game Over Scene**: Displays game over.

assets.md

📁 Assets

Assets are stored in the ``Assets/`` directory:

- **Sounds**: Audio files like background music and sound effects.

- **Fonts**: Fonts used in the game.
- **Textures**: Sprite sheets and game textures.

Adding New Assets

Sounds: Use ``sound_interface.txt`` to add new sounds.

sound_name path/to/sound.wav

Textures: Use ``texture_interface.txt`` to add new textures.

texture_name path/to/texture.png

contributing.md


Contributing

1. Fork the repository.
2. Create a new branch:
````bash  
git checkout -b feature-name`

3. Commit changes and push:

```
git commit -m "Description"
git push origin feature-name
```

4. Open a pull request.

```
`license.md`
```markdown  
#  License
```

This project is licensed under the [MIT License](../LICENSE).

about.md

About

This project was created as part of an Object-Oriented Programming class assignment at [Your University Name]. It demonstrates game development concepts in an academic setting.

You can now organize your project files into these Markdown documents for easier management and maintainability.