# Image Slider Project Report

## 1. Project Overview

The Image Slider project is a front-end web application built using React.js. It allows users to navigate through a series of game-related images with descriptive content, providing features such as autoplay, manual navigation, and thumbnail previews. The application demonstrates interactive UI design, state management, and custom hooks.

## 2. Technology Stack

- React.js (Functional Components, Hooks)
- JavaScript (ES6+)
- CSS3 for styling and animations
- HTML5

## 3. System Architecture

- Component Layer: Main ImageSlider component with navigation bar and slider UI.
- Custom Hook Layer: useImageSlider manages refs, autoplay logic, and event handlers.
- Presentation Layer: CSS-based styling and animations for transitions and hover effects.
- Integration Layer: External links for 'See More' and 'Buy Now' buttons.

## 4. Module Descriptions

| Module | Description |
|---|---|
| ImageSlider Component | Handles UI rendering of slider, navigation bar, thumbnails, and arrows. |
| useImageSlider Hook | Contains logic for image transitions, autoplay, and navigation. |
| Nav Bar | Provides sections (Home, Contents, Info) with hover and click interactions. |
| CSS Styles | Defines layout, animations, and responsive design. |

## 5. Workflow

1. User hovers or clicks on navigation bar links to view content.
2. User can manually click arrows to move to the next or previous image.
3. Autoplay feature automatically advances the slider every 3 seconds.
4. Thumbnails dynamically update to reflect current image order.
5. Buttons allow redirection to external resources such as Wikipedia or PlayStation Store.

## 6. Error Handling & Logging

As this is a front-end project, error handling primarily involves defensive coding in React. The showSlider function validates DOM nodes before performing transitions. Additionally, event listeners are cleaned up on component unmount to prevent memory leaks.

## 7. Testing

The project can be tested using React Testing Library or Jest. Tests should cover navigation bar interactions, autoplay behavior, and rendering of slider items. At present, testing is minimal.

## 8. Conclusion & Future Enhancements

The Image Slider project demonstrates effective use of React.js and custom hooks to create an interactive image slider. Future improvements may include: - Adding pause/play controls for autoplay. - Enhancing accessibility with ARIA attributes. - Supporting dynamic image lists from an API instead of static assets. - Improving test coverage with integration and snapshot tests.