# Technical Design Document (TDD)

## Project Name:

TestWhiz - Employee Management System

## Document Version:

1.0

## Author:

[Your Name]

## Date:

[Insert Date]

---

### 1. Overview

This document provides the technical details for automating the Employee Management System using Playwright. The tests are designed to validate the functionality of adding employees and verifying the employee list.

---

### 2. Test Environment

- \*\*Framework:\*\* Playwright

- \*\*Programming Language:\*\* TypeScript

- \*\*Test Runner:\*\* Playwright Test

- \*\*URL:\*\* `http://localhost:4200/employees`

- \*\*Browser:\*\* Chromium, WebKit, Firefox

---

### 3. Test Cases

#### Test Case 1: Add Employee and Verify Employee List

- \*\*File Name:\*\* `new.spec.tsqx`

- \*\*Test Name:\*\* `test1`

- \*\*Description:\*\* Automates the process of adding a new employee and verifying the employee list.

- \*\*Implementation:\*\*

```typescript

test('test1', async ({ page }) => {

await page.goto('http://localhost:4200/employees');

await page.getByRole('link', { name: 'Employee List' }).click();

await page.getByRole('link', { name: 'Add Employee' }).click();

await page.locator('#firstName').fill('sajid');

await page.locator('#lastName').fill('khan');

await page.locator('#emailId').fill('sajid@gmail.com');

await page.getByRole('button', { name: 'Submit' }).click();

await page.getByRole('link', { name: 'Employee List' }).click();

await page.getByRole('cell', { name: 'First Name' }).click();

await page.getByRole('cell', { name: 'Last Name' }).click();

await page.getByRole('cell', { name: 'Email Id' }).click();

});