**Special Needs Conveyance (SNC)**

**Source Code Documentation and Analysis**

***team:***

|  |  |
| --- | --- |
| **Basem Damir (**  **Team leader )** | **201910739** |
| **Elham Ali** | **201910641** |
| **Samar Abu Dayyeh** | **202010433** |
| **Fatin El-bitar** | **202020640** |

4/8/2023

link GitHub:[**https://github.com/elhamaltaie22**](https://github.com/elhamaltaie22)

Amman arab Universitay

**Source Code Documentation:**

1. Comments and Documentation:

- Add comments to explain complex logic, algorithms, and important methods.

- Document external libraries used: [List libraries and their purposes].

- High-level comments in each file: Briefly describe the file's purpose.

2. Method Documentation:

csharp

/// <summary>

/// Registers a new user with provided details.

/// </summary>

/// <param name="name">User's full name.</param>

/// <param name="email">User's email address.</param>

/// <param name="password">User's password.</param>

/// <returns>True if registration is successful, False otherwise.</returns>

public bool RegisterUser(string name, string email, string password)

{

// Implementation...

}

3. Variable and Constant Documentation:

csharp

/// <summary>

/// The maximum allowed login attempts.

/// </summary>

private const int MaxLoginAttempts = 3;

/// <summary>

/// The current number of login attempts.

/// </summary>

private int loginAttempts = 0;

4. Class Documentation:

csharp

/// <summary>

/// Represents a service for handling user-related functionalities.

/// </summary>

public class UserService

{

/// <summary>

/// Creates a new user account.

/// </summary>

/// <param name="name">User's full name.</param>

/// <param name="email">User's email address.</param>

/// <param name="password">User's password.</param>

public void CreateUser(string name, string email, string password)

{

// Implementation...

}

/// <summary>

/// Updates user information.

/// </summary>

/// <param name="userId">ID of the user to update.</param>

/// <param name="newName">New name for the user.</param>

/// <param name="newEmail">New email for the user.</param>

public void UpdateUser(int userId, string newName, string newEmail)

{

// Implementation...

}

}

5. Use Case Examples:

csharp

// Example: Registering a new user

bool registrationSuccess = RegisterUser("Elham Ali", "elhamaltaie001@gmail.com", "securepassword");

Console.WriteLine(registrationSuccess ? "Registration successful" : "Registration failed");

// Example: Updating user information

UserService userService = new UserService();

userService.UpdateUser(123, "New Name", "newemail@example.com");

**Code Analysis:**

1. Static Code Analysis:

- Utilize tools like Roslyn Analyzers to analyze code quality and adherence to coding standards.

2. Code Review:

- Conduct regular peer code reviews to ensure code correctness and quality.

3. Performance Analysis:

- Profile code using profiling tools to identify performance bottlenecks.

- Optimize critical methods for improved performance.

4. Accessibility Analysis:

- Ensure accessibility features for users with mobility impairments by testing keyboard navigation and focus.

5. Documentation Review:

- Review source code documentation for clarity and correctness.

6. Testing Integration:

- Integrate code analysis tools and documentation reviews into your CI/CD pipeline.