# Course Subscription App – README

## 1. Introduction

This application is a prototype Course Subscription system built using **ServiceNow’s App Engine Studio and UI Builder**. It enables users to **view available courses, subscribe/unsubscribe to courses**, and manage their subscriptions. This solution was developed demonstrate understanding of table design, business logic, and user interface development in the ServiceNow platform.

**2. Feature Description and Mapping to Requirements**

| **Feature** | **Description** | **Assignment Requirement** |
| --- | --- | --- |
| **Course Listing** | Implemented using custom table course. Displayed using repeater component within UI Builder. (e.g: Ui.png) | Course listing view |
| **Course Detail View (optional)** | Default view for admin to add course as show in screenshot (e.g: course.png) | Course detail (if required) |
| **Course Subscription** | Subscription handled via subscription table. Toggle button on each card performs subscribe/unsubscribe via Data Resource actions. (e.g : builder.png) | Subscription interaction |
| **Duplicate Prevention** | A Business Rule (Prevent Duplicate Subscriptions) prevents the same user from subscribing to the same course more than once.(e,.g: buisness.png) | Business rule logic |
| **UI Experience** | Created through Experience: “CourseHubExperience”, with a main page using repeater and dynamic Data Resource for subscriptions. | UI Builder experience |
| **Subscribed / Un Subscribed logic** | Created a scripted for toggle to create and delete subscription from table (e.g toggle.png) | Script logic |
| **Subscribed Courses** | Filtered view of courses user is subscribed to using user reference in subscription table. (e.g script.png) | User-specific view |

## 

## 3. Step-by-Step User Interaction

1. **User sees subscription status**

Each course displays a **Subscribe/Unsubscribe** toggle based on whether the current user is already subscribed (checked via Data Resource condition).

1. **User subscribes to a course**
   * On clicking "Subscribe", a record is created in subscription with current user's sys\_id and course reference.
2. **Duplicate prevention**
   * If user tries to subscribe again, the **business rule** blocks duplicate entries and optionally shows an alert message .
3. **User unsubscribes**
   * Clicking "Unsubscribe" deletes the user's subscription record.

## 4. Test Suite Summary

A high-level test strategy was applied:

* Manual UI tests performed for:
  + Subscribing and unsubscribing behavior.
  + Validation of business rule on duplicates.
  + Correct display of subscription toggle per user.
* Backend tests included:
  + Creating dummy records in the Subscription table.
  + Checking behavior of business rule on duplicate inserts.
* Automate testing using robot framework  
   This Robot Framework test automates these steps:
  + - Opens Chrome browser and goes to the ServiceNow login page.
    - Logs in using provided credentials.
    - Navigates to a specific courses page.
    - Finds and clicks the toggle button for the first course subscription.
    - Waits, reloads the page, then checks if the toggle button’s state changed correctly (either "Subscribe" or "Unsubscribe").
    - Logs the toggle state and fails the test if the state is unexpected.
    - Finally, it closes the browser.
    - **Purpose:**  
      To verify that the user can successfully log in and toggle course subscription status in the ServiceNow app, ensuring the toggle functionality works as intended.

1. **Screenshots**

The following screenshots demonstrate the implemented features:

* Ui.png - Course listing view with repeater component
* course.png - Course detail view for admin
* builder.png - UI Builder setup showing subscription toggle
* business.png - Business rule for duplicate prevention
* toggle.png - Toggle script logic implementation
* script.png - Subscribed courses filtered view

1. **Parts of Assignment Not Implemented**

All core requirements from the Dev Hire assignment have been successfully implemented:

* PRD01: Course List (fetch, display, subscribe button)
* PRD02: Course Subscription (basic and optional features)
* QA01: Robot Framework test suite

No significant parts of the assignment were omitted from the implementation.