# **📘 Capstone Project Development Guide**

### **🏁 Project: QR Code Attendance Check-In App**

This guide outlines the steps to help you design and implement your capstone project - a QR code-based attendance system built with the Frappe Framework. It is structured to help you think through functionality, user flow, and implementation logic without prescribing exact technical steps.

## **🧭 Project Summary**

You're building an app that allows users to check in by scanning a dynamic QR code. After scanning, users verify their identity by entering their Member ID and confirming it with a one-time password (OTP) sent via email, SMS, WhatsApp - or all of them. The process should be secure, time-bound, and intuitive.

## **🎓 What You'll Learn by Building This Project**

By completing this project, you will gain practical experience in:

* Creating public-facing web pages in Frappe
* Generating dynamic QR codes and encoding session-specific data
* Handling time-bound tokens for secure access
* Designing a secure multi-step check-in process
* Building and consuming REST APIs in Frappe
* Sending OTPs via SMS, email, or WhatsApp
* Managing and validating user input and real-time feedback
* Logging attendance records reliably
* Thinking through real-world use cases and edge cases
* Delivering a complete product from front-end to backend

This project brings together everything you've learned during the bootcamp into one practical, usable, and demonstrable solution.

## **✅ Step-by-Step Development Flow**

### **🔹 Step 1: Display a Public Check-In QR Code**

Design a public-facing web page on your site that **anyone can access without logging in**.

* This page should display a **QR code** that updates every 5 minutes.
* Each QR code encodes a unique reference.
* When scanned, the QR code directs users to a check-in link
* The system uses the reference to determine if the QR scan is still valid.

### **🔹 Step 2: Handle QR Code Scans**

When a QR code is scanned:

* The user is taken to the check-in verification page (e.g. /checkin/member) with the session token in the URL.
* The system first checks if the token is valid and has not expired.
* If the token is invalid, show an appropriate error message and stop the process.

### **🔹 Step 3: Prompt for Member ID**

On the check-in page:

* Ask the user to enter their Member ID.
* After submitting, validate that the Member ID exists.
* If it’s valid, send a one-time password (OTP) to the member.

🔔 **You can send the OTP via Email, SMS, WhatsApp, or a combination — the method is up to you.**

### **🔹 Step 4: Display OTP Field (With Delay)**

* After submitting a valid Member ID, wait **3 seconds**, then show the OTP input field.
* This improves user flow and mimics real-world verification delays.

### **🔹 Step 5: Verify OTP and Complete Check-In**

* When the user enters the OTP and submits:
  + Validate that the OTP is correct, has not expired, and has not been used.
  + If valid, log the check-in and show a success message.
  + If invalid, show an error and allow one or two retry attempts.
* Ensure each member can only check in once per token.

### **🔹 Step 6: Log the Check-In**

After a successful OTP verification:

* Log the check-in with member details, timestamp, and reference to the scanned QR token.
* Prevent repeated or fraudulent check-ins using the same token or expired sessions.

## **💡 Tips for a Great Project**

* Keep the user interface clean and mobile-friendly.
* Use countdown timers or labels to show QR code refresh times if needed.
* Handle expired or reused tokens gracefully.
* Choose OTP delivery channels based on what you’re most comfortable implementing.
* Focus on **security**, **usability**, and **reliability**.

## **🗓 Deadline**

Your project must be submitted within **30 days** from the final day of the bootcamp.

## **🧪 Submission Instructions**

* Push your complete project to a public GitHub repository.
* Include a clear README.md with setup instructions and an overview of how your app works.
* Submit the repository link via email or the provided submission form.