



# FaithConnect – Hackathon Challenge

## Project Overview

FaithConnect is a mobile-first platform where **Worshippers connect directly with Religious Leaders** through content, community, and conversations.

The goal of this hackathon is to build a **functional mobile prototype** of FaithConnect that demonstrates the **core product experience**:

- Discovering religious leaders
- Consuming spiritual content
- Following leaders
- Posting and engaging with content
- Messaging between worshippers and leaders

This is **not about perfection or scale**.

This is about **clarity of product thinking, clean UI/UX, strong backend logic, and smooth execution**.

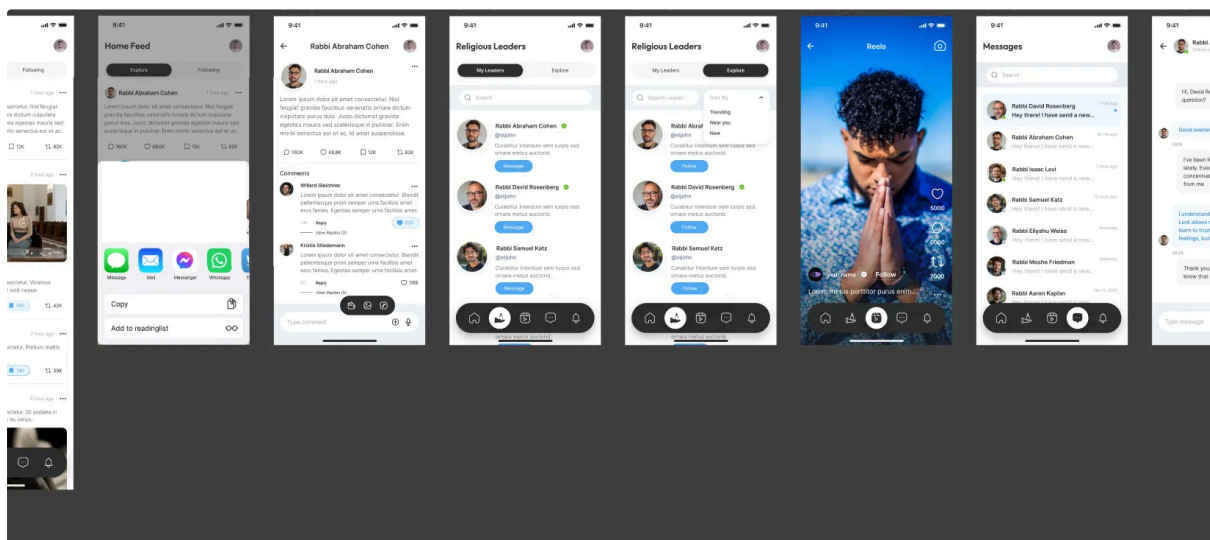
---

## What You Need to Build

Create a **working prototype mobile application** for **FaithConnect** that includes the **core flows** described below.

## Scope Rules (Important)

- No subscription-based features are required.
- The focus should be on core platform functionality, ensuring that all primary user flows are fully functional for both worshippers and religious leaders.
- Both sides of the platform must be handled carefully, with clear, smooth, and intuitive interactions.
- Features may be implemented in a simplified manner, but they must work end-to-end.
- The user experience should remain clean and consistent, aligned with the theme and direction shown in the attached screenshots.
- The objective is to deliver a fully functional prototype based on the described user flow.
- Teams are free to introduce additional features if they believe these add meaningful value to the platform.



## FaithConnect – Application Flow

### 1. Intro / Landing

- App opens with:
  - FaithConnect logo
  - One-line description:  
| “A platform where Worshipers connect with their Religious Leaders.”
- CTA buttons:
  - **Continue as Worshiper**
  - **Continue as Religious Leader**

Keep this screen **minimal, calm, and clean.**

---

## 2. Authentication (Simple but Functional)

- Email + Password login/signup
- Google login (optional bonus)
- No complex validation required

Users choose their role (**Worshiper or Religious Leader**) during onboarding.

---

## Worshiper Flow

### Step 1: Basic Profile Setup

- Name
  - Email
  - Faith selection (Christianity, Islam, Judaism, Other)
  - Profile photo (optional)
- 

### Step 2: Home Tab

The Home tab has **two sections**:

#### Explore

- Shows content from **various religious leaders**

- Each post includes:
  - Leader name & profile photo
  - Caption / text
  - Image or video
  - Actions:
    - Like
    - Comment
    - Save
    - Share

## Following

- Shows posts only from leaders the user has **followed**
- 

## Step 3: Religious Leaders Section

This section helps worshipers discover and manage leaders.

### Explore Leaders

- List of leaders available on the platform
- Each leader card shows:
  - Profile photo
  - Name
  - Faith tag
  - Short description
  - **Follow button**

### My Leaders

- Shows leaders the user has followed
  - From here, users can:
    - Open leader profile
    - Message the leader
- 

## Step 4: Leader Profile Page

When a worshiper opens a leader profile:

- Profile header:
    - Photo
    - Name
    - Faith
    - Bio
  - Tabs:
    - **Posts**
    - **Reels**
  - Actions:
    - Follow / Unfollow
    - Message leader
- 

## Step 5: Reels Tab

- Vertical, swipe-based video feed
  - Short inspirational content
  - Actions:
    - Like
    - Comment
    - Save
    - Share
- 

## Step 6: Chats

- Worshipers can message leaders they follow
  - Simple 1-on-1 chat
  - Messages must be **stored and fetched via backend APIs**
- 

## Step 7: Notifications

- Activity-based notifications:
  - New post from a followed leader
  - New reel
  - New message
  - Likes / comments on posts

(Basic notification logic is sufficient, but it must work.)

---

## Religious Leader Flow

### Step 1: Leader Profile Setup

- Name
- Faith / denomination
- Short bio
- Profile photo

(No verification logic required for this hackathon)

---

### Step 2: Leader Dashboard

Leaders can:

- View their posts
  - View followers list (simple list)
  - Access messages
- 

### Step 3: Create Content

Leaders should be able to:

- Create a **Post**
  - Text
  - Image or video
- Create a **Reel**
  - Short vertical video

Posts should appear:

- On Home → Explore

- On followers' feeds
  - On leader profile
- 

## Step 4: Messages

- Leaders can:
    - Receive messages from worshipers
    - Reply to messages
  - Keep chat functionality simple but **fully functional**
- 



## Design & UX Guidelines

- Mobile-first design
- Clean, calm, respectful UI
- No clutter
- Smooth navigation
- Focus on **readability and usability**

You will be provided **reference screens / wireframes** to guide visual direction.

You are free to:

- Improve UX
  - Simplify flows
  - Make thoughtful product decisions
- 



## Tech Expectations

Participants are free to choose their stack, for example:

- **Mobile App Required**
  - Flutter
- Backend:
  - Can be Firebase / Supabase / simple API
- **Backend:**
  - Node.js
  - Python

- **Database:**
    - PostgreSQL / MySQL
    - MongoDB
    - Can be Firebase / Supabase / simple API
  - **Auth:**
    - JWT / Firebase Auth (only if backed by APIs)
  - Dummy data is acceptable where needed
- 



## Points to Note

1. **UI/UX:** Clean, intuitive, and user-friendly design. Keep the flow smooth and simple.
  2. **Code Quality:** Follow best practices and maintain well-structured code.
  3. **Backend must be functional and API-driven**
  4. Core flows (follow, post, message) and functionalities must work end-to-end and smoothly.
  5. Doing **one part extremely well** is better than doing everything poorly
  6. **App Submission:** Submit the APK and a demo video. In the final stage, you will be required to upload the app to TestFlight; the team will cooperate and might assist with this process.
  7. **No Code Submission:** You are **Not Required to submit your code**. Winners will continue development as part of the founding team under expert guidance. This challenge focuses on Skills, **AI leverage and creativity**.
  8. **Use of AI Tools:** Set up Cursor or similar tools to improve productivity and speed act smartly.
  9. **Discipline:** We are strict about **discipline** and behavior, so make sure to consider this. Those who stay committed, consistent, and respectful until the end will receive a certificate in recognition of their efforts.
- 



## Submission Instructions



## 1. Demo Video (3–5 minutes)

- Walk through:
  - Worshiper flow
  - Leader flow
  - Content creation
  - Messaging

## 2. App Build

- APK (mandatory)
- TestFlight (Last Step)

## 3. Join the Telegram group for submissions: <https://t.me/+rajq1VIUd0g3MDdl>

---

## Prizes and Opportunities

- 🏆 **Hackathon Winner:**
    - The most functional and user-friendly platform creator will receive a **full-time job offer** as a **founding member with a high CTC and base pay**.
  - 🌟 **Certificates:**
    - Participants with functional MVPs, even if not winners, will receive **certificates of appreciation** if their submissions are liked by the founders.
  - 🌟 **Prizes:**
    - Depending on the submissions, prizes may be distributed to the top 3 winners. The prize amounts will be shared later.
- 

## Closing Note

This hackathon is more than a challenge it's a chance to **shape your future** while working on one of the most **advanced projects** out there.

It's a test for your **dedication and obsession** towards your role not just how much you know, but how much you're willing to figure out.

It's a **test of skills over resumes** in the era of AI, you're **completely free to use any AI tool** to do this assignment. What matters is how smartly you use it and how much ownership you show.

We want to work with **people who enjoy challenges**, the ones who keep pushing even when things break, and **don't stop until it finally works**.

Once you're onboard, you'll be working with a **team of geniuses** and on some seriously exciting stuff. But before that, you've got to **prove you're the right fit** for this role.