Development Roadmap for "The Simple Rep" App using FlutterFlow

Overview

This roadmap outlines a step-by-step plan to develop the MVP of "The Simple Rep" app using FlutterFlow. The plan is organized in a Gantt-style format, detailing phases, tasks, durations, dependencies, and milestones.

Summary Timeline

- **Phase 1:** Requirements Analysis and Planning (1 week)
- **Phase 2:** Design (2 weeks)
- **Phase 3:** Development (6 weeks)
- **Phase 4:** Testing (2 weeks)
- **Phase 5:** Deployment (1 week)
- **Phase 6:** Monitoring and Feedback (Ongoing)

Total Estimated Duration: 12 weeks

Detailed Roadmap

Phase 1: Requirements Analysis and Planning

Duration: Week 1

Tasks:

1. **Finalize Requirements (2 days)**
    - Review and confirm all functional and non-functional requirements.
    - Define user roles and permissions.
2. **Create User Stories and Use Cases (2 days)**
    - Develop detailed user stories for each user type.
    - Prioritize features for the MVP.
3. **Define Technical Specifications (1 day)**
    - Determine data models and APIs.
    - Plan integrations (contact syncing, calendar, external training resources).

Dependencies: None

Milestones:

- Approval of requirements and user stories.
- MVP scope is clearly defined.

Phase 2: Design

Duration: Weeks 2-3

Tasks:

1. **UI/UX Design (5 days)**
    - Create wireframes for all app screens.
    - Design user interfaces for different user roles.
2. **Prototype Key Screens in FlutterFlow (3 days)**
    - Use FlutterFlow to prototype main navigation and key interfaces.
    - Validate design feasibility within FlutterFlow's capabilities.
3. **User Flow and Navigation Design (2 days)**
    - Map out user journeys for each user type.
    - Define navigation logic and screen transitions.

Dependencies:

- Completion of Phase 1 tasks.

Milestones:

- Approval of UI/UX designs and prototypes.
- Design assets ready for development.

Phase 3: Development

Duration: Weeks 4-9

This phase will be divided into sprints focusing on different modules of the app.

Sprint 1: Project Setup and Onboarding Module

Duration: Week 4

Tasks:

1. **Set Up FlutterFlow Project (1 day)**
    - Configure project settings.
    - Set up Firebase backend (authentication, database).
2. **Implement User Authentication (2 days)**
    - Email/password login.
    - Role-based access control.
3. **Develop Onboarding Checklist Feature (2 days)**
    - Create onboarding tasks and progress tracking.
    - Implement notifications for task deadlines.
4. **Integrate Contact Syncing (1 day)**
    - Use FlutterFlow's integration to access device contacts.
    - Ensure permissions and privacy compliance.

Dependencies:

- Completion of design prototypes for onboarding screens.

Milestones:

- Users can sign up, log in, and begin the onboarding process.
- Onboarding checklist functional.

Sprint 2: Referral Submission and Tracking

Duration: Week 5

Tasks:

1. **Develop Referral Submission Interface (2 days)**
   - Allow users to select contacts and submit referrals.
   - Implement form validations.
2. **Implement Referral Tracking Dashboard (2 days)**
   - Display referral statuses.
   - Integrate notifications for status updates.
3. **Create HR/Admin Referral Management Interface (1 day)**
   - List of submitted referrals.
   - Ability to update statuses and add notes.
4. **Set Up Database Structures (1 day)**
   - Define collections/tables for referrals, candidates, HR activities.

Dependencies:

- Completion of Sprint 1.

Milestones:

- Reps can submit referrals and track their statuses.
- HR/Admin can manage referrals.

Sprint 3: Gamification Elements

Duration: Week 6

Tasks:

1. **Implement Points System (1 day)**
   - Assign points for activities (tasks completed, referrals submitted/hired).
2. **Develop Gamification Tiers and Prizes (2 days)**
   - Configure tiers for referrals submitted and hired.
   - Display progress towards tiers.
3. **Create Leaderboards (1 day)**
   - Display rankings based on points, referrals.
4. **Design Gamification Notifications (1 day)**
   - Notify users when they reach new tiers or earn prizes.

Dependencies:

- Referral submission and tracking features.

Milestones:

- Gamification system functional with tiers and leaderboards.

Sprint 4: Training Modules and First 28 Days Program

Duration: Week 7

Tasks:

1. **Develop Training Module Interface (2 days)**
    - List of training modules.
    - Progress tracking for module completion.
2. **Integrate External Training Content (1 day)**
    - Link to Sam Taggart's training resources.
    - Ensure content is accessible within the app.
3. **Implement First 28 Days Program (2 days)**
    - Define daily and weekly tasks.
    - Track progress and rewards.
4. **Set Up Notifications and Reminders (1 day)**
    - Remind users of pending tasks and deadlines.

Dependencies:

- User authentication and onboarding.

Milestones:

- Users can access training materials and participate in the 28-day program.

Sprint 5: Communication Tools

Duration: Week 8

Tasks:

1. **Implement In-App Messaging (2 days)**
    - Enable messaging between reps, mentors, managers, and HR.
2. **Develop Notifications Center (1 day)**
    - Central place for users to view all notifications.
3. **Set Up Group Channels (1 day)**
    - Create team channels for announcements and updates.
4. **Enable Media Sharing (1 day)**
    - Allow users to upload and share photos (e.g., "turfing" pictures).

Dependencies:

- User roles and permissions in place.

Milestones:

- Communication features operational.

Sprint 6: HR/Admin Features and Analytics

Duration: Week 9

Tasks:

1. **Enhance HR/Admin Dashboard (2 days)**
    - Advanced filters and sorting for referrals.
    - Candidate pipeline management.
2. **Implement Analytics Dashboards (2 days)**
    - KPIs for referrals, hiring rates, per-rep analytics.
3. **Develop Reporting Tools (1 day)**
    - Generate reports on recruitment metrics.
4. **Set Up Permissions and Access Control (1 day)**
    - Ensure HR/Admin functionalities are secure.

Dependencies:

- Basic HR/Admin features from Sprint 2.

Milestones:

- HR/Admin can fully manage referrals and access analytics.

Phase 4: Testing

Duration: Weeks 10-11

Tasks:

1. **Unit Testing (1 week)**
    - Test individual components for functionality.
2. **Integration Testing (3 days)**
    - Ensure different modules work seamlessly together.
3. **User Acceptance Testing (4 days)**
    - Gather feedback from a small group of users.
    - Identify and fix bugs.
4. **Performance Testing (2 days)**
    - Test app performance under load.

Dependencies:

- Completion of development phase.

Milestones:

- App is stable and ready for deployment.

Phase 5: Deployment

Duration: Week 12

Tasks:

1. **Prepare App for Deployment (2 days)**
    - Configure app settings for production.
    - Optimize assets and code.
2. **App Store Submission (2 days)**
    - Prepare app listing for Google Play Store and Apple App Store.
    - Submit app for review.
3. **Deployment of Backend Services (1 day)**
    - Ensure Firebase services are configured for production.

Dependencies:

- Successful completion of testing phase.

Milestones:

- App submitted to app stores and awaiting approval.

Phase 6: Monitoring and Feedback

Duration: Ongoing after Deployment

Tasks:

1. **Monitor App Performance**
    - Use analytics tools to monitor user engagement and app stability.
2. **Collect User Feedback**
    - Implement feedback forms within the app.
    - Monitor app store reviews.
3. **Plan Updates and Improvements**
    - Prioritize bugs and feature requests for future releases.

Dependencies:

- App is live on app stores.

Milestones:

- Continuous improvement based on user feedback.

Dependencies Summary

- **Design Phase depends on:** Completion of Requirements Analysis.
- **Development Sprints depend on:** Completion of Design and previous sprints.
- **Testing Phase depends on:** Completion of all Development Sprints.
- **Deployment depends on:** Successful Testing.
- **Monitoring depends on:** Deployment of the app.

Resource Allocation

- **Project Manager:** Oversee the project timeline and coordinate between teams.
- **UI/UX Designer:** Responsible for app design during Phase 2.
- **FlutterFlow Developer(s):** Build the app using FlutterFlow during Phase 3.
- **QA Tester(s):** Conduct testing during Phase 4.
- **HR/Admin Representative:** Assist in defining HR functionalities and testing.

Considerations for FlutterFlow

- **Platform Limitations:**
  - Ensure that FlutterFlow supports all required functionalities, especially for complex features like in-app messaging and advanced analytics.
  - Custom code may be needed for certain integrations (e.g., contact syncing, calendar integration).
- **Backend Services:**
  - Utilize Firebase for authentication, database, and storage.
  - Ensure data structures are optimized for real-time updates.
- **Third-Party Integrations:**
  - Verify that necessary APIs (e.g., for contact access) can be integrated within FlutterFlow.

Risk Management

- **Potential Delays:**
  - Integration challenges with FlutterFlow limitations.
  - App store approval process may take longer than expected.
- **Mitigation Strategies:**
  - Early prototyping to identify limitations.
  - Maintain close communication with FlutterFlow support.
  - Prepare app store submissions in advance.

Milestones Recap

- **Week 1:** Requirements and planning completed.
- **Week 3:** Designs approved and ready for development.
- **Week 9:** Development completed.
- **Week 11:** Testing completed; app is stable.

- **Week 12:** App submitted to app stores.
- **Post Week 12:** App is live; monitoring and feedback collection begins.

Post-MVP Considerations

After the MVP is launched, consider planning for:

- **Feature Enhancements:**
  - Additional gamification elements.
  - Advanced analytics and reporting.
  - Offline functionalities.
- **User Feedback Implementation:**
  - Address common user requests.
  - Improve user experience based on feedback.
- **Scalability Improvements:**
  - Optimize for increased user load.
  - Enhance performance for data-heavy operations.

Conclusion

This roadmap provides a structured plan to develop "The Simple Rep" app using FlutterFlow, covering all essential features for the MVP within an estimated 12-week timeline. By following this plan, the development team can ensure that the app is delivered on time, meets the requirements, and provides value to its users.