



# SMART SPROUT

---

**Document Title:** Deliverable 6

**Team Name:** Smart Sprout

**Project Name:** Smart Home-Garden System

**Group Number:** Group 9

**Team Member & Student ID:**

- |                          |           |
|--------------------------|-----------|
| • Aditi Patel            | n01525570 |
| • Birava Prajapati       | n01579924 |
| • Darshankumar Prajapati | n01574247 |
| • Zeel Patel             | n01526282 |

**Table of Content**

**Table of Contents..... 2**





**Brief description of the project..... 3**

**Work Completed by each member..... 3**

    Prediction for Release Date:..... 8

### **Brief description of the project**

Our project, **Smart Sprout**, is a smart home gardening app designed to simplify plant care by integrating sensor data. It helps users monitor their plants' needs, such as moisture and temperature. The app automates plant care by offering notifications, weather updates, and a user-friendly interface for managing and tracking plant health, making it easy for users to care for their plants effortlessly.

Name	Student Id	Github Id	Signature	Efforts
Aditi Patel	n01525570	AditiPatel5570		100%
Birava Prajapati	n01579924	BiravaPrajapati9924		100%
Darshankumar Prajapati	n01574247	DarshanPrajapati4247		100%
Zeel Patel	n01526282	ZeelPatel6282		100%

- **GitHub Repo Link:** <https://github.com/DarshanPrajapati4247/SmartHomeGardenSys>

### **Work Completed by each member**

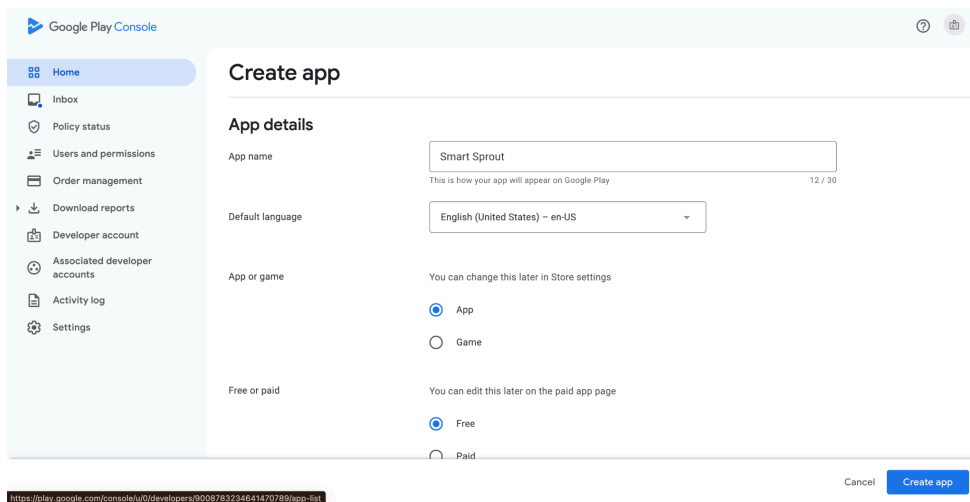
**Birava Prajapati:** I've made several updates to the Smart Sprout app to enhance functionality and maintainability. I updated the application label and notification icon to improve the app's branding and visual consistency. In the DiagnoseFragment, I introduced a "Coming Soon" message to provide users with a clear indication of upcoming features. To streamline the codebase, I removed unused import statements, ensuring cleaner and more efficient code. Additionally, I implemented functionality to delete photos from Firestore and the Room database on a background thread, optimizing performance and ensuring smooth operations. These changes collectively enhance the app's usability, responsiveness, and readiness for future updates.

**Darshan Prajapati:** I updated the alert icon for better visual clarity and made improvements to the plant API, ensuring more reliable and accurate data integration. On the log and registration screens, I added network connectivity checking to enhance user feedback and prevent errors during offline conditions. To further support offline users, I introduced an OfflineActivity with retry and settings functionality, seamlessly integrated with NetworkViewModel to provide real-time network status handling. Additionally, I updated the app's fonts to Poppins for a more modern and cohesive design. These enhancements collectively improve both the functionality and aesthetic appeal of the app.

**Aditi Patel:** I've implemented a notification feature that triggers whenever a user logs into the app. The notifications now appear seamlessly in the Notification Activity. Previously, there were some issues with this functionality, but they have been resolved. Additionally, I've ensured that the image and back button in the Notification Activity are working perfectly, providing a smoother user experience.

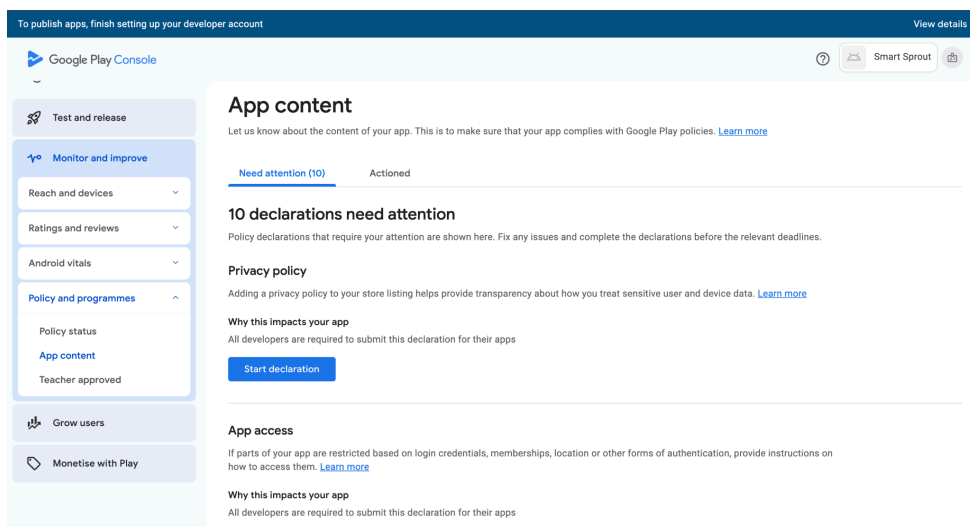
**Zeel Patel:** During this sprint, my focus has been on ensuring the quality and usability of the application. I dedicated time to thoroughly testing the app to identify and resolve any bugs or issues, ensuring a smooth user experience. Additionally, I worked on completing the remaining user interface (UI) elements, ensuring they aligned with the overall design and functionality requirements. To further enhance the development process, I also took the initiative to coordinate with testers, onboarding them to evaluate the app comprehensively. This combination of tasks has contributed to both the refinement and validation of the application.

## Steps for submitting the app to Google Play:



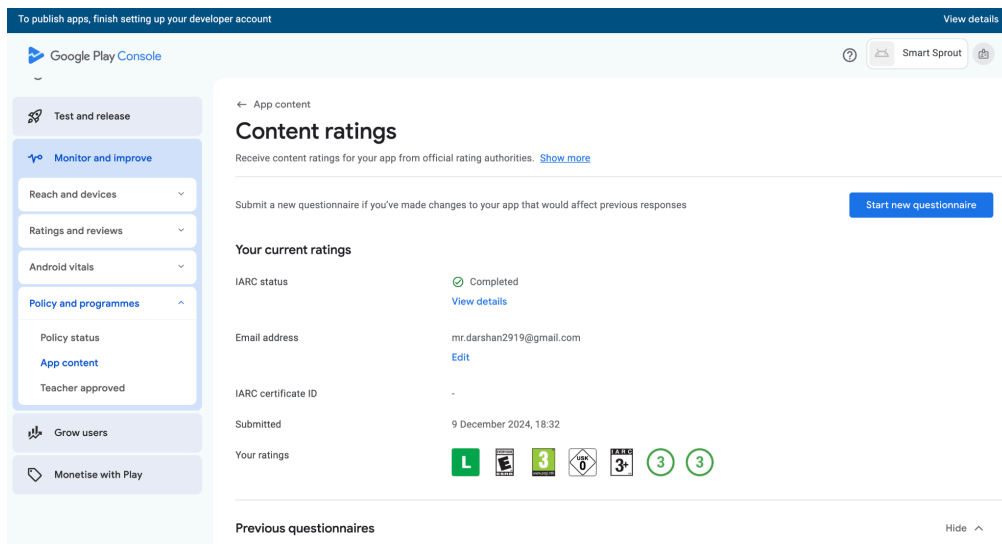
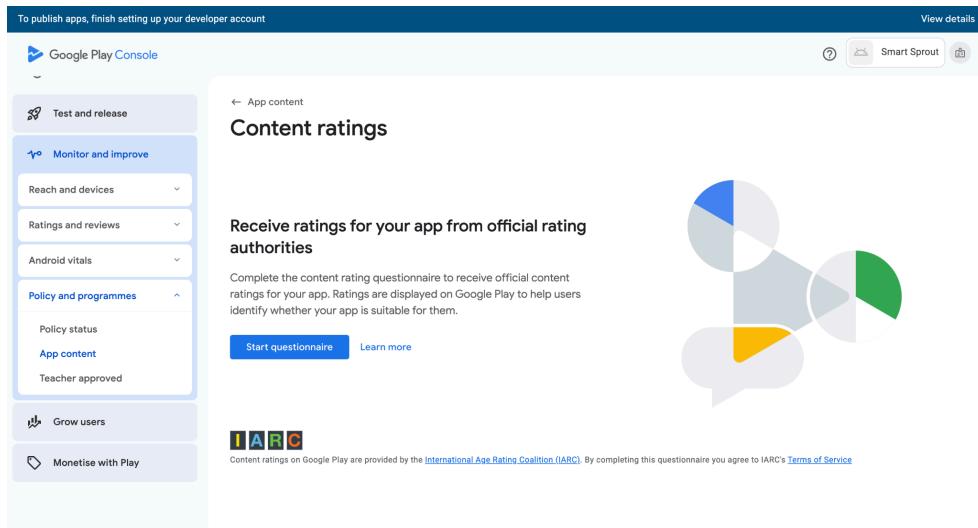
The screenshot shows the 'Create app' page in the Google Play Console. On the left is a sidebar with navigation options: Home, Inbox, Policy status, Users and permissions, Order management, Download reports, Developer account, Associated developer accounts, Activity log, and Settings. The main area is titled 'Create app' and contains 'App details'. The 'App name' field is filled with 'Smart Sprout'. The 'Default language' is set to 'English (United States) - en-US'. Under 'App or game', the 'App' radio button is selected. Under 'Free or paid', the 'Free' radio button is selected. At the bottom right, there are 'Cancel' and 'Create app' buttons. A URL bar at the bottom shows a developer-specific console link.

In this step, the app details, such as the name "Smart Sprout," default language, and type (App or Game), were entered in the **Google Play Console**. The app was marked as **free** before proceeding to the next steps.

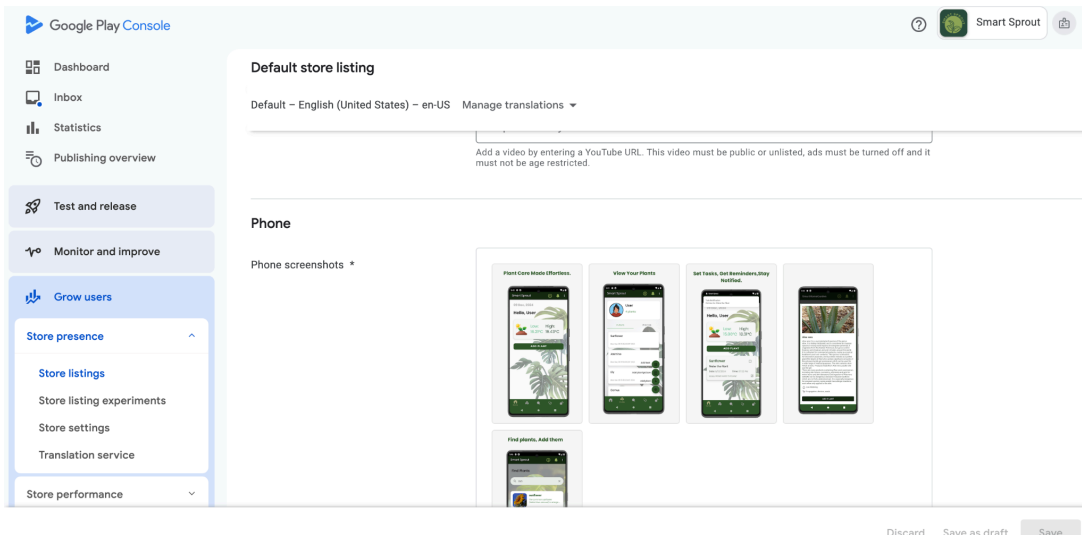


The screenshot shows the 'App content' page in the Google Play Console. The top bar indicates 'To publish apps, finish setting up your developer account'. The left sidebar has a 'Monitor and improve' section with options like 'Reach and devices', 'Ratings and reviews', 'Android vitals', 'Policy and programmes', 'Policy status', 'App content', and 'Teacher approved'. The main area is titled 'App content' and includes a 'Need attention (10)' section with a link to 'Learn more'. Below this, there are sections for '10 declarations need attention', 'Privacy policy', and 'App access'. A 'Start declaration' button is visible under the 'Privacy policy' section.

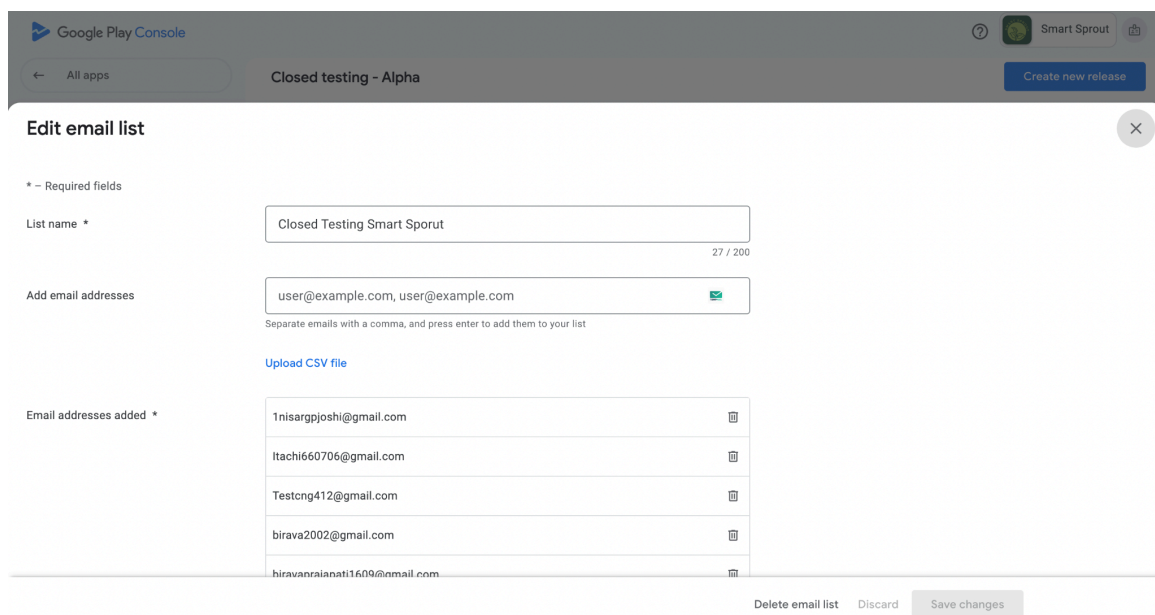
In this step, Google Play Console requires developers to address policy declarations to ensure the app complies with Play Store guidelines. If applicable, key actions include submitting a privacy policy and providing details on app access restrictions. Developers must complete these declarations to proceed with app publication.



In this step, complete the Content Rating Questionnaire the International Age Rating Coalition (IARC) provided to determine your app's age suitability. Ratings help users understand whether your app is appropriate for their demographic and are displayed on its Play Store page.



The **Store Listing** section lets us showcase SmartSprout's features with engaging screenshots and descriptions. It helps users understand the app's functionality and encourages downloads on the Google Play Store.



In this step, the **email list** for closed testing was edited in the **Google Play Console**. Testers' email addresses were added, and the list was saved to enable them to access the app for testing.

Google Play Console

**Darshan Prajapati**  
Personal account · Account ID: 9008783234641470789

**Pinned apps** ⓘ  
Pin apps here to access them quickly and view key metrics

**1 app** [Create app](#)

Filter by: **All**

App	Installed audience	App status	Update status	Last updated
Smart Sprout ca.smartsprout.it.smart.smarthome...	0	Closed testing		10 Dec 2024

© 2024 Google · [Mobile app](#) · [Status Dashboard](#) · [Terms of Service](#) · [Privacy](#) · [Developer Distribution Agreement](#)

Google Play Console

**Dashboard** [Last 30 days](#)

**Production**

**Apply for access to production**

Production is where you make your app available to billions of users on Google Play. Before you can apply for production access, you need to run a closed test that meets our criteria. When you apply, you'll also need to answer some questions about your closed test. [Preview questions](#)

- ✓ Publish a closed testing release
- Have at least 12 testers opted-in to your closed test  
*3 testers currently opted in*
- Run your closed test with at least 12 testers, for at least 14 days

[Apply for production](#) [Learn more](#)

**Your KPIs**

Based on the screenshot, the app currently has **3 testers opted in**, but **at least 12 testers** need to opt in, and the app must be tested for a **minimum of 14 days**.

## Prediction for Release Date:

Assuming the following:

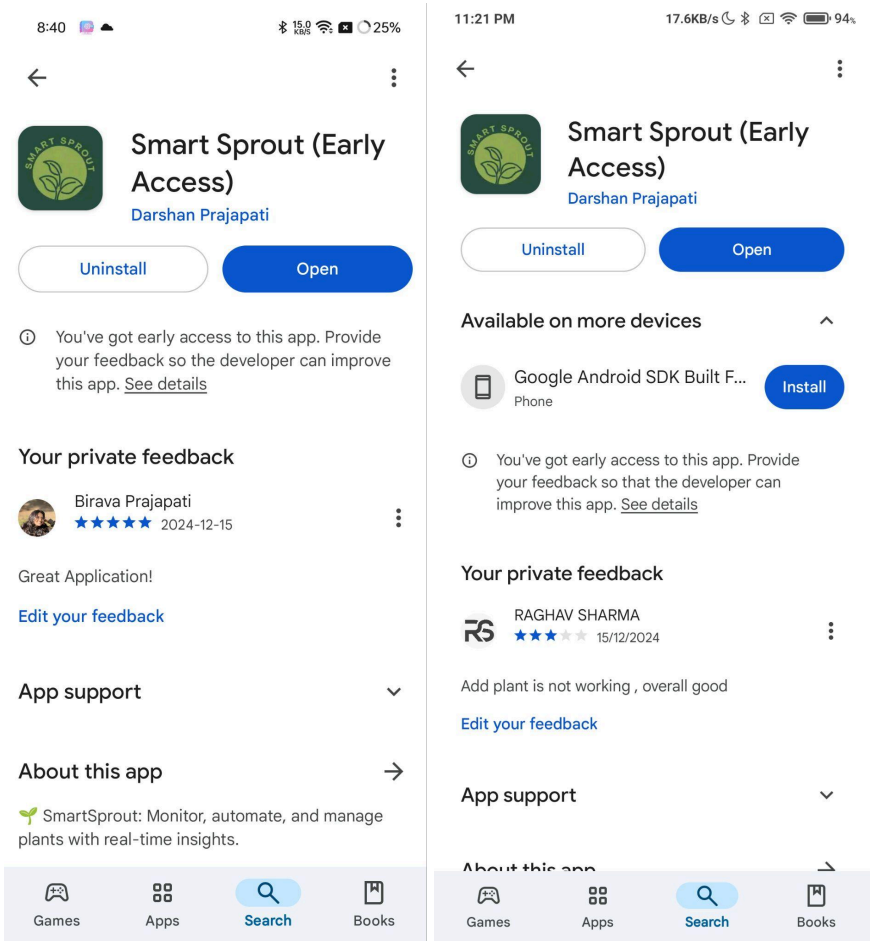
1. **12 testers** are gathered by **December 16, 2024**.
2. The app undergoes **14 days of closed testing** starting on **December 16, 2024**.

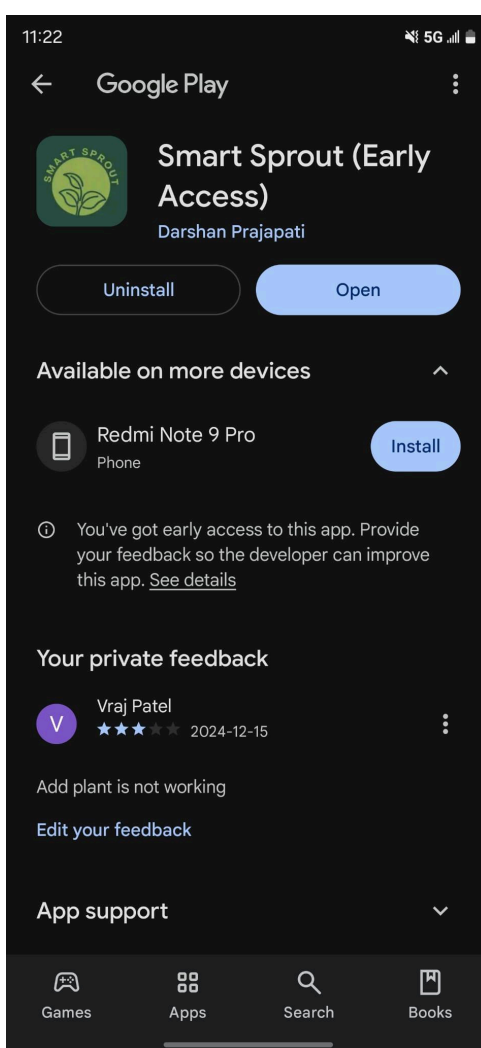
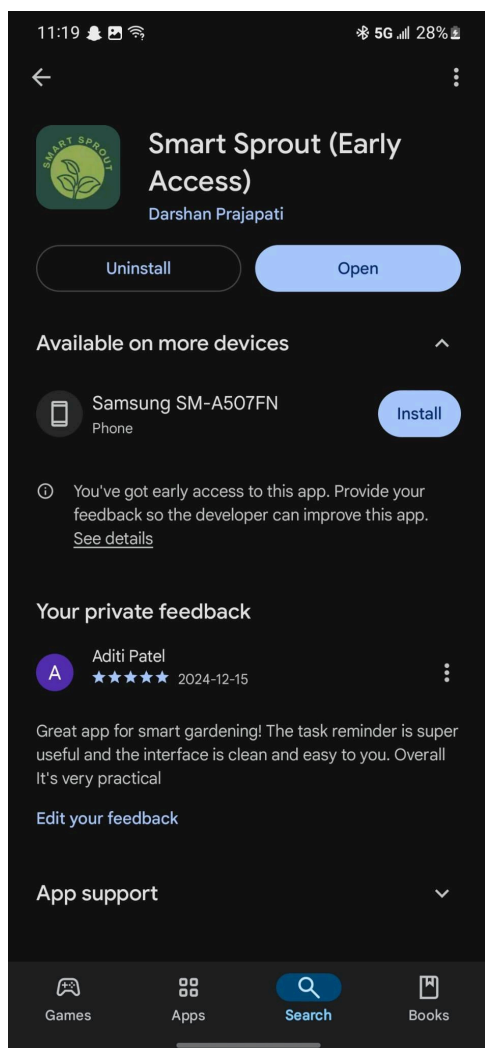
The earliest date the app could be submitted for **production access** would be:

**December 30, 2024.**



Once submitted, Google Play's review process may take an additional **2–7 days**, meaning the app could be publicly available around **January 1 to January 5, 2025**, depending on the review timeline.





Google Play Link:

<https://play.google.com/store/apps/details?id=ca.smartsprout.it.smart.smarthomegarden>