

MOBILE APPLICATION DEVELOPMENT

Lecture 6: Mobile Application Development Recap

**Course Lecturer: Menard Phiri,
Dept of CSIT, MUST**

LESSON OUTLINE

- Concept development
- Outlining the app
- Design considerations and platforms
- Budget development
- Programming
- Graphic design
- Submission, approval and updates
- Common pitfalls

CONCEPT DEVELOPMENT

Identify the purpose of the application (“app”).

- What will the app do?
 - Specify the minimum functionality. Will the app perform a specific task and/or deliver information?
 - Most apps do one thing and do it well.
 - Avoid overly complex functionality.
- Example: The “The Plant Doctor” app provides interactive diagnosis for plant health problems. Users submit text and photographs to assist the diagnosis.

CONCEPT DEVELOPMENT

Identify the intended users of the app.

- Who will use the app?
 - What is the target audience?
 - Is the app free or for payment?
- Example: Users of “The Plant Doctor” app include: Farmers, gardeners, homeowners, landscape professionals, County Agents, plant quarantine workers, professionals, and scientists.

CONCEPT DEVELOPMENT

Different types of audiences

- Children
- Adults
- Various levels of education
- Various levels of expertise
- Types of devices: Android versus iPhone
- Nationalities: Language, translations
- Your app and its design should *appeal to* and *serve* your target audiences: appearance, colors, functions, tasks

CONCEPT DEVELOPMENT

Specify the device(s) to which the app will be delivered

- Web app vs Native app
- Phones vs Tablets
- iPhone vs Android



- The choices determine the programming language and coding

CONCEPT DEVELOPMENT

Memory and space considerations

- App size is limited only by the space available on a device.
- There is no recommended size for apps.

CONCEPT DEVELOPMENT

Name the app

- The app name should accurately reflect the function of the app
- The app name should be unique, if possible, and not replicate the names of other apps.

OUTLINING A DRAFT OF THE APP

- Create a draft step chart: Identify each thing the app will do, in a stepwise fashion
- Create a draft page chart: Sketch and compose the appearance of various pages
- Specify the graphics and navigation buttons needed for the app functions
- Specify the outputs and calls

DESIGN CONSIDERATIONS

- Design the app for fingers (not for a mouse)
 - 44 x 44 pixels for buttons
 - 6 buttons maximum per width of screen (smart phone)
- TEXT INPUT: People do not like to type, so keep typing at a minimum
- TEXT OUTPUT: People generally do not like to read, so keep text at a minimum unless your app is designed primarily to deliver information in text format.

BUDGET DEVELOPMENT

- **Self-programmed versus Contracted programming**
- Apple (iTunes): programming language = iOS
- Google (Android): programming language = Java
- **Before budget development:** Prepare detailed requirement documents (what the app does). Prepare a detailed use case document (describe each functionality of the app)

PROGRAMMING

○ How much will it cost to develop the app?

1. Personal vs. contract development

2. By hour or by project development. Costs per hour vary among countries (\$15-20 for India: \$50-60 for USA; \$100 for USA technical)

Example: “The Plant Doctor” app development and deployment cost 80 hours (at the US rate) for each platform (iOS and Android), for a total cost of \$9,600.00 (\$4,800.00 for each platform).

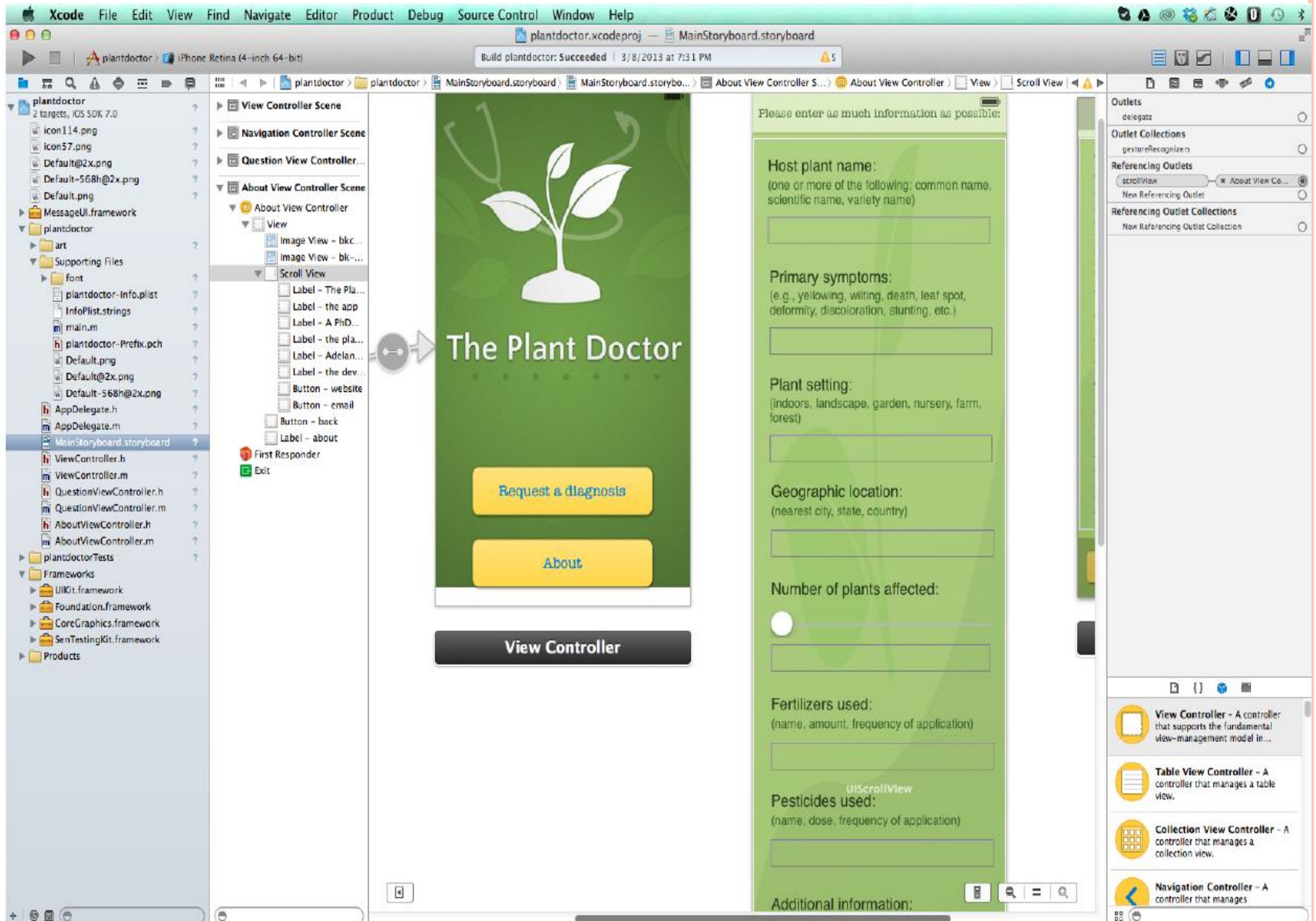
PROGRAMMING

- **They contracted the following companies for the development of “The Plant Doctor” app.**
- **App development company:**
 - Adelante Consulting (Canada)
 - www.adelante.ca
- **Graphics design company:**
 - AQUASoft (Romania), normally only does large jobs
 - <http://aquasoft.ro/>

PROGRAMMING

- Sample pages from the Apple Developers programming software environment appear on the following screens.
- Remember: Developing an app is writing a program using a specific programming language suited for a particular device.
- Therefore, a developer should be familiar with the software and the programming language.

PROGRAMMING



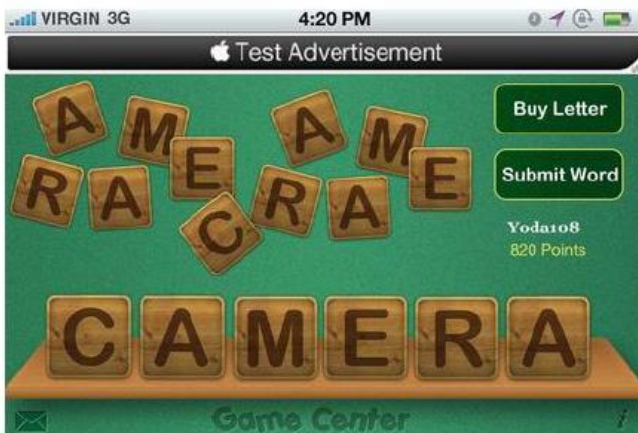
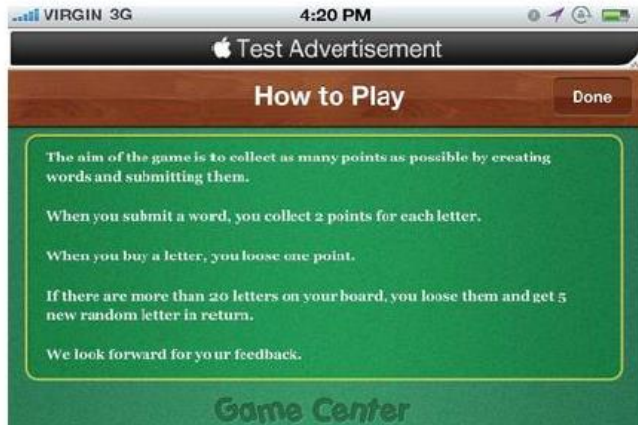
PROGRAMMING

- Example:

The requirements and the use case documents for the “Wordissimo” app appear on the following screens.

PROGRAMMING

Wordissimo



Screens:

- main screen
- feedback screen
- help
- Game Center

Main screen:

- iAds area
- center area
- word area

Rules of the game:

- there is a bag of around 60 letters (not displayed). Letter count in a separate doc
- user starts with 5 letters and with 10 points
- he can buy one new random letter (this costs 1 point)
- he can form a word. In this case the letters go back to the bag and he is awarded with 2 points for each letter
- **if he has more than 20 letters on the board a pop up is displayed and all the letters are lost and 5 new are presented**

The aim of the game is to make words and get points that are uploaded to the Apple Game Center

The words will be validated against a word database

The board is divided into 2 regions:

Center area: contains the letters the user has. The letters should be moved in a natural way by tapping and moving them.

Word area: a lower region where the words can be formed. Here the letters should be added, removed or moved in a different position of the word. If one letter is added in a position, all the letters to the right move one position to the right. If one is removed, all the letters to the right move one position to the left

When a button is pressed, the word is validated and if correct, **the word is animated outside of the screen** and the points are awarded and uploaded to Game Center.

Letter frequency and dictionary

[here](#)

This should be easy changeable. Either keep them in a file or database

PROGRAMMING

Use cases

1. Program start

- five random letters are presented on the screen
- **the points are loaded from Game Center**
- if the user is not registered with GC, a popup is displayed with a message and the user is invited to register

2. Letter moves in the center area of the screen

- A letter is taped
- the letter is moved a little to simulate the 3D elevation. A fade shadow is added underneath the letter
- when the finger is moved, the letter moves accordingly. **When dropped the letter pushes the others around so that the letters are not covering each other**

3. letter moves in the lower part of the screen

3.1 letter add

- A letter is picked in the center area and dropped into the word area
- The letter is inserted at the closest place in the word
- if there are more letters that fit on the screen, a scroll bar is displayed

3.2 letter move

- a letter in the word area is tapped and moved
- the letters in the word change position to allow the moved letter to be inserted

3.3 letter remove

- a letter in the word area is tapped and moved to another position outside of the word area
- the letter is removed from the word area
- the other letters in the word change position to fill the empty space

4. Buy letter

- **Buy letter** button is pressed
- A new letter is animated inside the screen
- 1 point is deducted from the total points

5. word validation

- the word in the word area is changed (a letter is added, removed, moved)
- the word is validated against the dictionary
 - *if the word is valid, the button in the right part is changed to green (not used)*
 - *otherwise the button is red (not used)*
- if the word is valid, the button is enabled: **Submit the word**

The validation is done on the fly, without user intervention

6. submit the word

- *the button in the right of the word area is pressed (not used)*
- *if the button is red a message is displayed: "Word not valid" (not used)*
- *if the button is green, meaning the word is valid, all the letters are animated outside of the screen (not used)*
- Submit button is pressed
- All the letters are animated outside of the screen
- For each letter, the total points is increased with 2 points

7. Game center

- Game center button is pressed
- the Game center interface is **displayed**

8. iAds

If iAds are server, they are displayed in the top part of the screen.
If iAds are not available, the iAd view is moved outside the page

9. Sounds

A sound will be played for the following:

- a letter is bought
- a letter is picked
- **a letter is dropped**
- a word is submitted

New Developement

1. Bonus

Offer a bonus for the following cases:

- One point for each letter extra in a word longer than 3 characters
- For example
- if the word is 3 chars long: award $3 \times 2 = 6$ point
 - if the word is 7 chars long: award $7 \times 2 + (7-3) = 18$ points

- Offer 10 points bonus when, after a word is submitted, the board remains empty

Bonus Display

When the bonus is offered, display for 2 seconds, unded the words "nn Points" the following message:

"nn Bonus"

PROGRAMMING

For example for the 7 chars long, display:

34 Points

4 Bonus

After 2 seconds, remove the words "4 Bonus" by slowly fading it out

2. When some points are added, add them 1 by one, let's say 1 every 0.25 sec

For example, the user has 10 points and submits a word and wins 8 more points. Then the Point area will change:

10 Points

11 Points

12 Points

....

18 Points

About screen

The aim of the game is to collect as many points as possible by creating words and submitting them.

Form words in the lower part of the screen and "Sell" them for 2 points a letter.

For words longer than 3 letters, you get 1 point bonus for each letter over the 3rd one.

Clear the table and get 10 points bonus.

You can buy a letter for one point.

But be mindful, you cannot have more than 20 letters, or you're gonna loose them and get 5 new ones in return.

Looking forward for your feedback.

Chages in the code:

- create images and sound folders
- added **HTML_BODY**
- remove the iad from the about page. Enlarge it
- change information.png
- [submitword_off@2x.png](#) and [submitword_on@2x.png](#)

GRAPHIC DESIGN

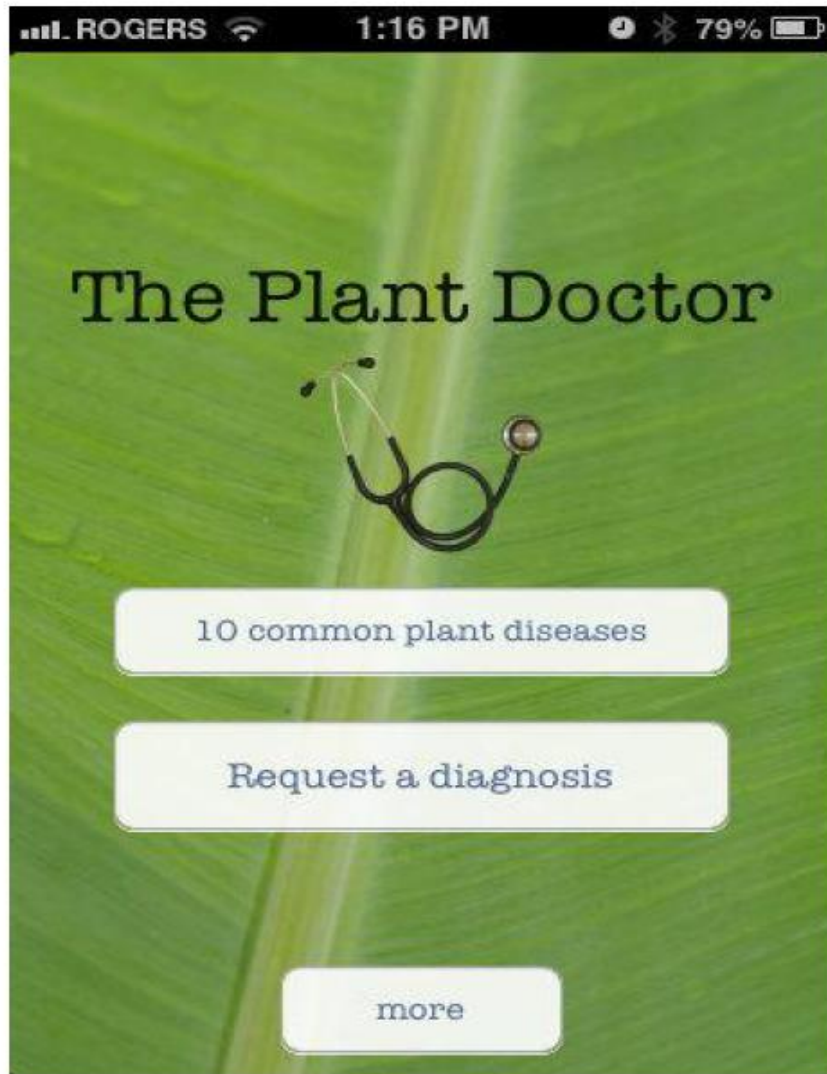
- Appearance of the app is very important and should be a high priority, with sufficient budget provided for graphics
- Graphic design is best done professionally
- Costs:
 - Romania = \$200 to \$300 per screen
 - USA = \$500 per screen
- One can expect much higher costs for the unique design of multiple screens. The cost for “The Plant Doctor” graphics design was \$300

GRAPHIC DESIGN

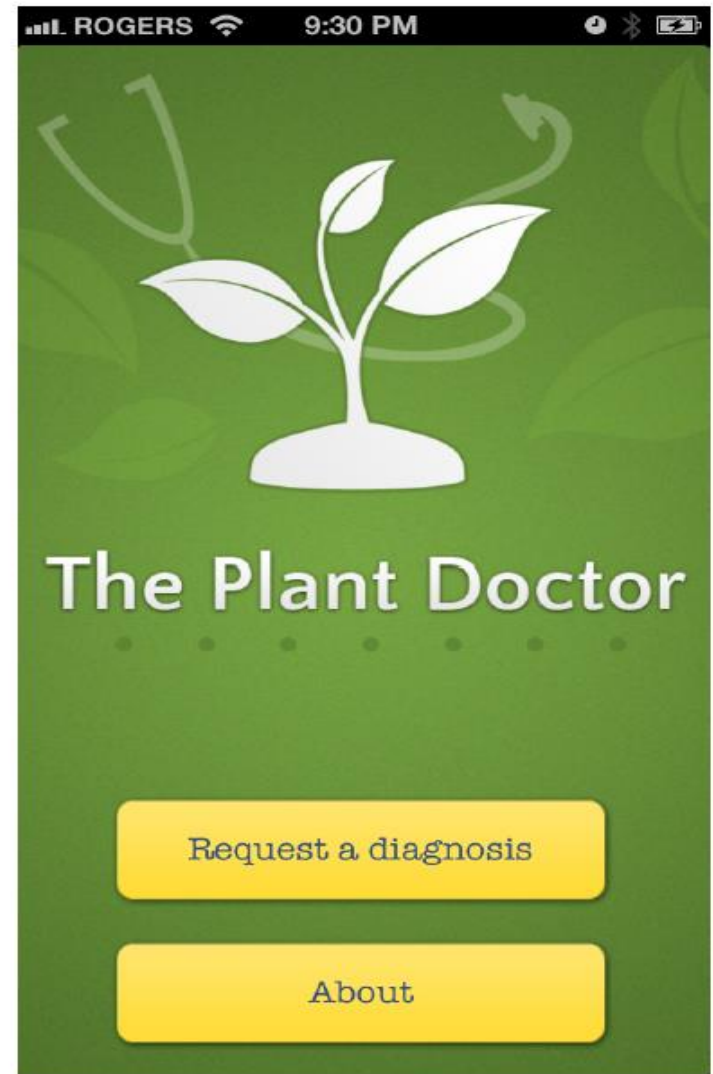
Considerations in graphics design:

- Colors and color theme (background, buttons)
- Art
- Text (size, color, font)
- Layout
- Balance
- Layering

GRAPHIC DESIGN



Before professional design



After professional design

GRAPHIC DESIGN



ROGERS 1:17 PM 79%

Please enter as much information as possible in the spaces below

Host plant name:
(one or more of the following: common name, scientific name, variety name)

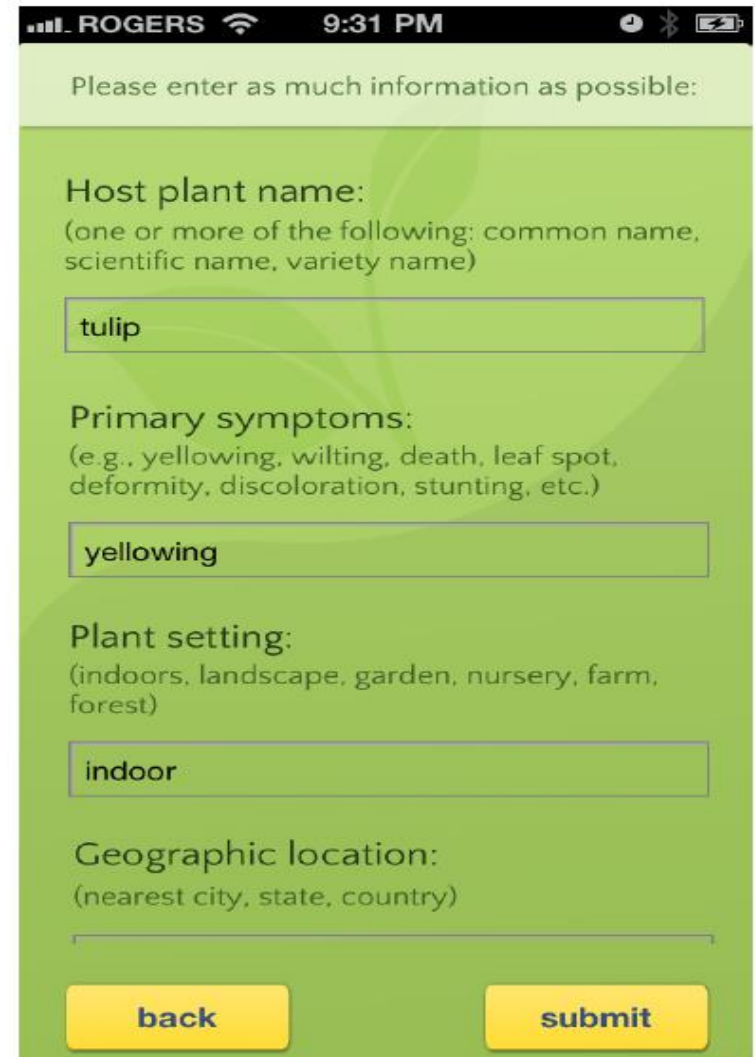
Primary symptoms:
(e.g., yellowing, wilting, death, leaf spot, deformity, discoloration, stunting, etc)

Plant setting:
(indoors, landscape, garden, nursery, farm, forest)

back submit

The interface is a simple mobile app form with a green background. It contains three text input fields for 'Host plant name', 'Primary symptoms', and 'Plant setting'. The text is in a basic sans-serif font. At the bottom, there are two buttons labeled 'back' and 'submit'.

Before professional design



ROGERS 9:31 PM

Please enter as much information as possible:

Host plant name:
(one or more of the following: common name, scientific name, variety name)

Primary symptoms:
(e.g., yellowing, wilting, death, leaf spot, deformity, discoloration, stunting, etc.)

Plant setting:
(indoors, landscape, garden, nursery, farm, forest)

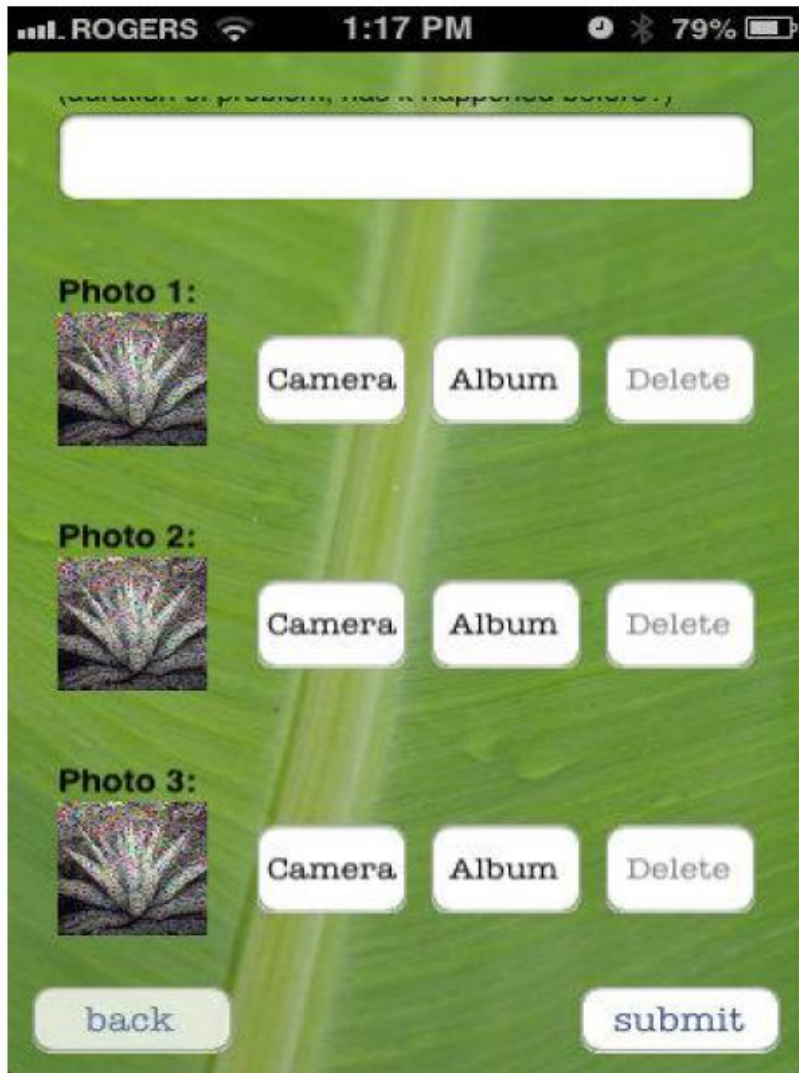
Geographic location:
(nearest city, state, country)

back submit

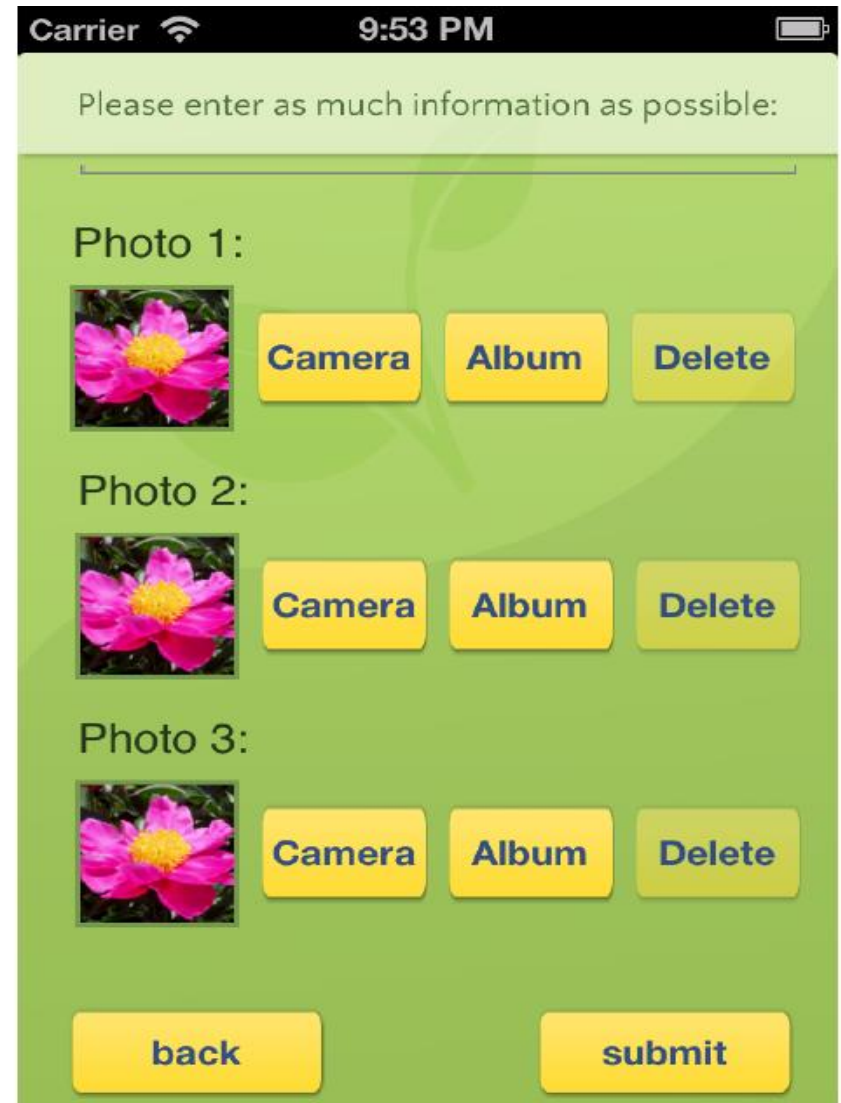
The interface is a more polished mobile app form with a light green background. It contains four text input fields for 'Host plant name', 'Primary symptoms', 'Plant setting', and 'Geographic location'. The text is in a clean sans-serif font. At the bottom, there are two buttons labeled 'back' and 'submit'. The overall design is more modern and user-friendly.

After professional design

GRAPHIC DESIGN



Before professional design



After professional design

SUBMISSION AND APPROVAL

The app submission process varies by platform

- Apple: checks for no crashes; the app does what it says it will do; and minimum information content may be needed
- Developer accounts:
 - Apple: Developer's license = \$100/year
 - Google: Developer's license = \$20/year
- Note: For the for-sale apps, payment to Apple or Google is 30% of your profit. Free apps do not incur this charge.

SUBMISSION AND APPROVAL

The screenshot displays the Google Play Developer Console interface for the app 'THE PLANT DOCTOR' by 'ca.adelante.thepantdoctor'. The left sidebar shows navigation options: All applications, Game services, Settings, and Announcements. The 'Store Listing' section is active, showing a list of tabs: Statistics, Ratings & Reviews, Crashes & ANRs, Optimization Tips, APK, Store Listing (selected), Pricing & Distribution, In-app Products, and Services & APIs. The 'Store Listing' tab is expanded, showing the 'STORE LISTING' section with a 'Saved' button. Below this is the 'PRODUCT DETAILS' section, which includes a language selector set to 'English (United States) - en-US' and an 'Add translations' button. The 'Title' field is filled with 'The Plant Doctor' (16 of 30 characters). The 'Description' field is filled with a detailed text about the app's interactive diagnosis features, user-supplied information, and the professional pathologist's expertise. The 'Promo text' field is empty (0 of 80 characters). The 'Recent changes' field is empty (0 of 500 characters). The 'GRAPHIC ASSETS' section is at the bottom, indicating that no localized graphics have been added yet.

Store Listing - The Plant Doctor - Google Play Developer Console

https://play.google.com/apps/publish/?dev_acc=05827107549459919231#MarketListingPlace=ca.adelante.thepantdoctor

Saved Tabs • YouTube • Yahoo! • 7tab=3m5m1mbox • Google Maps • Wikipedia • News • Popular • Sites • Plant Disease Diagnosis • PEPS 430 - Management • PEPS 405

Inbox - sctomelson@gmail.com - Gmail

Store Listing - The Plant Doctor - Google Play Developer Console

Google play | Developer Console

Adelante Consulting Inc. • sctomelson@gmail.com • Sign out • Help & Feedback

THE PLANT DOCTOR - ca.adelante.thepantdoctor

✓ Published

STORE LISTING Saved

PRODUCT DETAILS Fields marked with * need to be filled before publishing.

English (United States) - en-US Add translations

Title *
English (United States) - en-US The Plant Doctor
16 of 30 characters

Description *
English (United States) - en-US
The Plant Doctor provides interactive diagnosis for plant diseases in gardens, landscapes, nurseries and farms.
The app collects user-supplied information (text, photographs) about the problem and sends it to a professional plant pathologist with a Ph.D. and more than 20 years of experience in the science.
The user promptly receives five communications from the pathologist, including the probable name of the plant disease and causal pathogen, tactics for managing the disease, and contact information for local university experts who may be able to provide more information or examine samples in person.
Every year, plant diseases cause tremendous losses to plants and untold human suffering worldwide.
One must accurately and quickly identify the causes of disease in order to manage or control it properly.
1181 of 4000 characters
Please check out these [tips on how to create policy compliant app descriptions](#) to avoid some common reasons for app suspension.

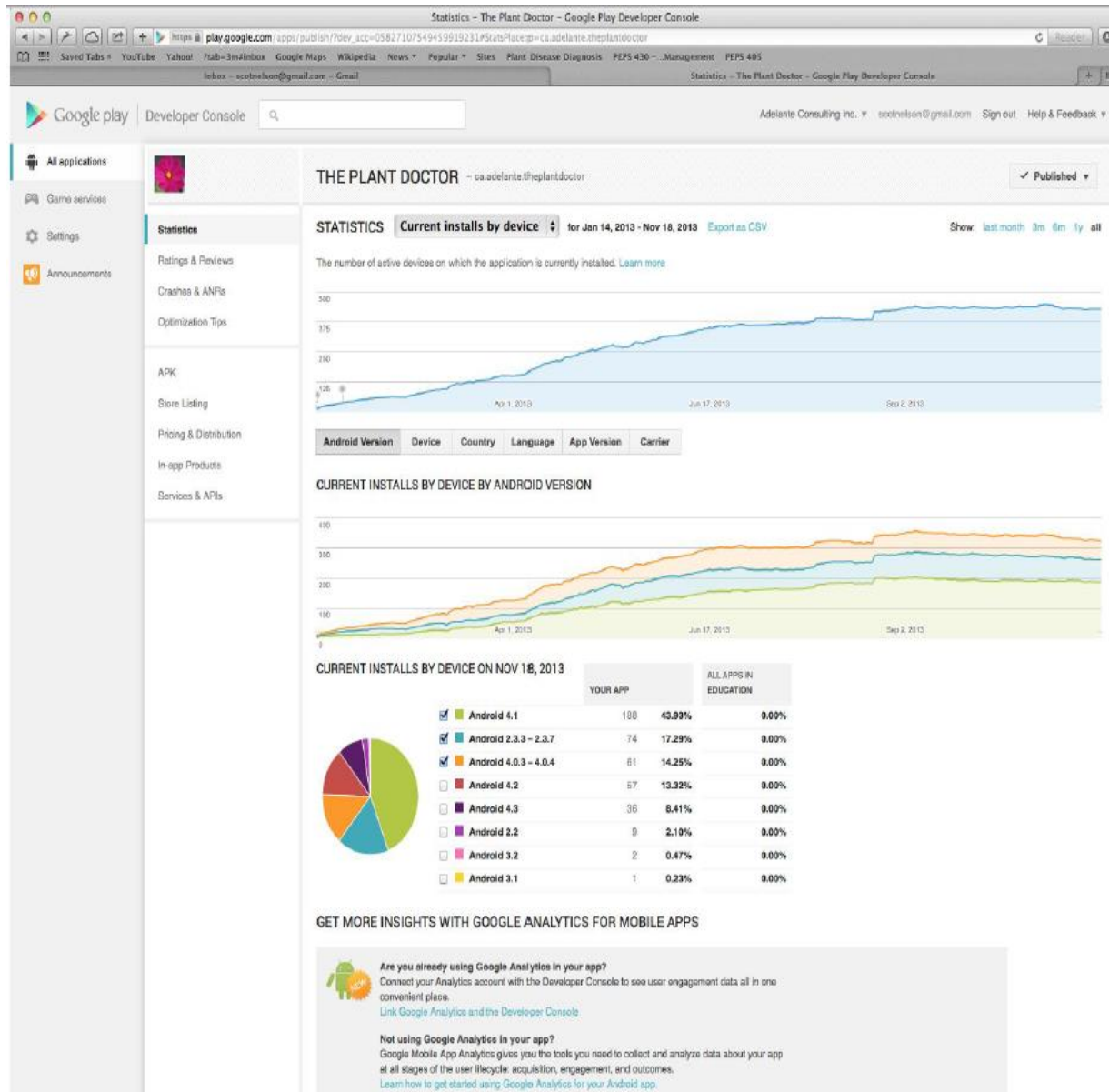
Promo text
English (United States) - en-US
0 of 80 characters

Recent changes
English (United States) - en-US
0 of 500 characters
[Learn more about recent changes.](#)

GRAPHIC ASSETS
If you haven't added localized graphics for each language, graphics for your default language will be used.
[Learn more about graphic assets.](#)

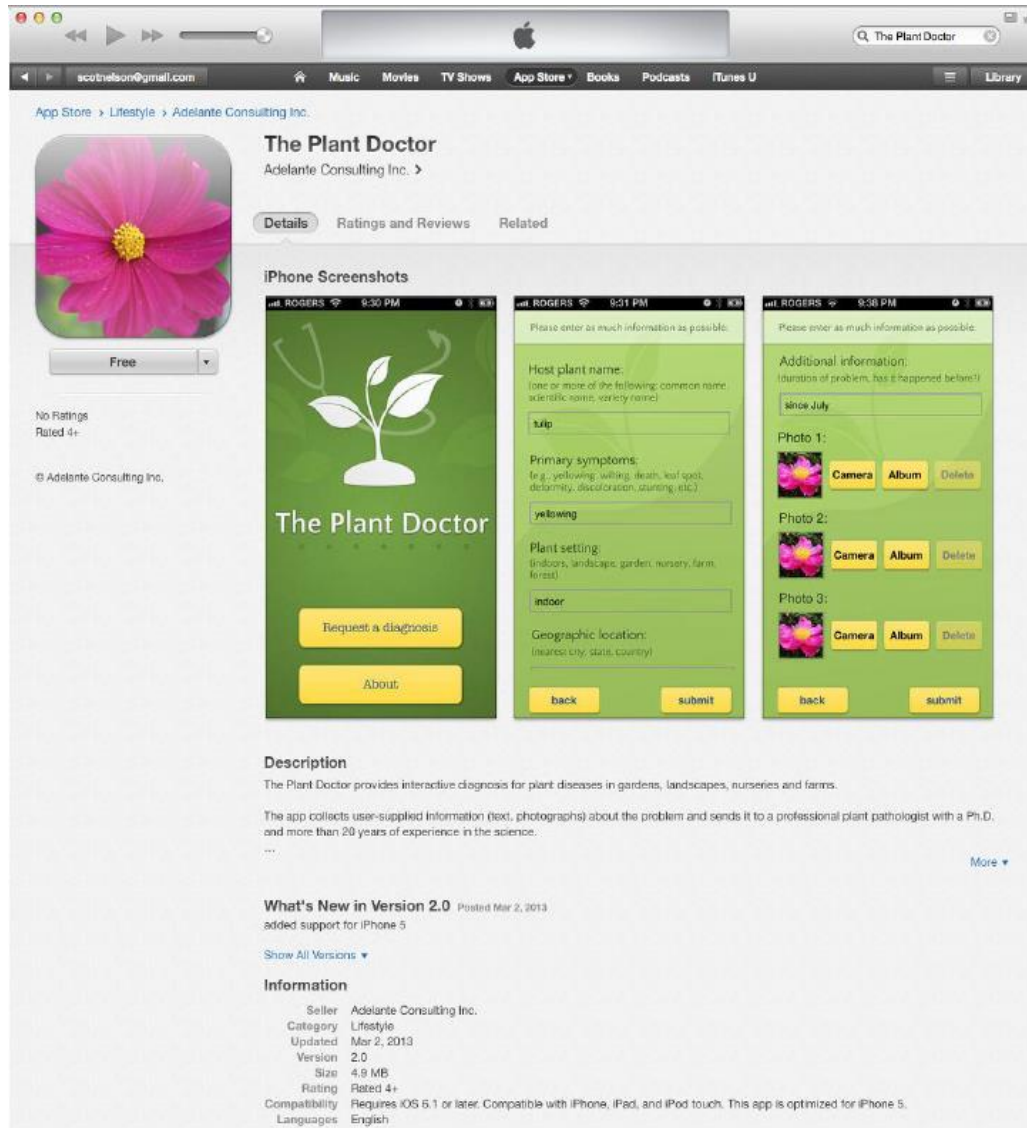
Google
developer console
(Android)

SUBMISSION AND APPROVAL



Google
developer console
(Android)

SUBMISSION AND APPROVAL



App description and download page at the iTunes store for “The Plant Doctor” app

SUBMISSION AND APPROVAL

- Updates for apps may be required as new versions of phones emerge onto the market.
- The new phones may have updated or newer versions of software which require app updates.
- App developers should prepare for this eventuality.