Computer Science 491IP: iOS Programming

CamLib Business Plan

Samuel Silverman

Product Definition Statement

CamLib is for individuals who are frequently creating bibliographies and/or gathering information from books, providing the ability to generate useful information about a book straight from the camera or a saved image.

User Definition

The CamLib platform is not planned to support picture books or children books. However, despite this age restriction, anyone from an infrequent reader to a book enthusiast can benefit from the simple to use features of CamLib. The ideal user for CamLib is an individual relying on bibliographies and/or general information about a book.

Market Analysis

Competing Apps and Platforms

The two main competitors for CamLib are EasyBib.com and generic search platforms. Search platforms allow users to quickly look up information on a book and EasyBib.com provides users with a platform for creating bibliographies for a wide range of sources. Ultimately, CamLib will hav an edge on the competition by automatically gathering information for the user rather than having the user find and enter information for bibliographies, relevant searches, etc.

User and Sales Statistics

The following article from The Next Web provides information o the number of bibliographies and users EasyBib.com received in 2012:

https://thenextweb.com/apps/2012/04/03/easybib-boasts-half-a-billion-citations-with-34-million-students-on-its-platform/

The article states that around 34 million students use EasyBib.com services. In addition to books, EasyBib.com also has various features for individual articles, movies, and other forms of media. With a low estimate, we can assume that only 1% of the 34 million students, or 340,000 students, use EasyBib.com for bibliography generation of books. A target number of users for CamLib has been set at 1% of the estimated 340,000 students using EasyBib.com for book citations at a total of 3,400 students.

Budget

CamLib will be a free to use app with no subscriptions. As a result, there is currently $0 of planned income. For planned expenses, I am assuming CamLib will take around 50 hours to develop with labor cost priced at $25/hour with an additional $250 to cover the deprecation of the development machine. Additionally, there will be a $99 developer license fee to add CamLib to the App Store. Furthermore, to give CamLib a jump start, a current budget of $1000 could be used for advertisement purposes. This gives a total of $2599 of planned expenses.

If the app explodes in popularity, then there are possibilities for revenue generation to offset the planned expenses. If, for example, the app were to be downloaded by 34,000 people rather than 3,400 people, or 10% of the estimated users rather than 1%, then options would open up for potential revenue generation. If the option for in-app advertisements is added to the app, revenue could be generated to begin offsetting the $2599 of expenses. Companies would be interested in putting ads in CamLib due to the exceptionally high number of users.

User Scenarios

Choosing Options for Finding Book Info

WHAT DOES THE USER SEE?

The user sees a menu with two options for either taking a picture of a book or selecting a photo of a book saved on the phone.

WHAT DOES THE USER CONTROL?

The user controls which option to select.

WHAT STEPS ARE FOLLOWED AND WHAT OPTIONS DO THEY HAVE?

The user must select which menu option they want. This brings the user to the appropriate next page to give the app a picture or photo.

Taking Picture or Selecting Photo

WHAT DOES THE USER SEE?

The user either sees a camera with a box outlining where the book should be in the picture, or a menu asking the user to select a photo saved on the phone. One an image is selected, the user will be able to edit the photo to define the area of the cover of the book.

WHAT DOES THE USER CONTROL?

The user controls when to take a picture, what saved photo to use, and how to orient the picture/photo to allow the app to find the book in the image.

WHAT STEPS ARE FOLLOWED AND WHAT OPTIONS DO THEY HAVE?

For taking a picture, the user first takes a picture of the book, then orients the picture to fit the outlined box, then confirms the picture. For selecting a photo, the user finds a saved photo from the phone, orients the photo to fit the outlined box, then confirms the picture. The user also has the option to reelect or retake the picture of the book.

Choosing Information to View

WHAT DOES THE USER SEE?

The user sees a cover image of the book followed by a series of tabs containing information about the book selected. If there is no cover image for the book, the image the user took is displayed in the information section.

WHAT DOES THE USER CONTROL?

The user controls which tab is open allowing the user to control what information is displayed.

WHAT STEPS ARE FOLLOWED AND WHAT OPTIONS DO THEY HAVE?

The user has to select which tab is to be displayed in the view from a tab bar.

App Data Management

CamLib will use data from the saved photos on the device as well as the camera to capture any live images of books. The app may possibly use preferences to allow for sharing of information about books via email, iMessage, etc. The image processing will use Apple's Vision framework and information about a book will come from the Open Library API.

Milestone List and Tasks

Milestones

BASIC UI SETUP

Estimated Time: ~1 Week

Dependent Tasks: 1, 9

Deliverables: A menu, option screen for selecting photos or saved images, and views for information on a book.

CAMERA & PHOTO FUNCTIONALITY

Estimated Time: ~1-2 Weeks

Dependent Tasks: 2, 3, 9

Deliverables: The ability to take pictures or select photos of books.

IMAGE RECOGNITION OF SELECTED PHOTOS

Estimated Time: ~2-3 weeks

Dependent Tasks: 4, 9

Deliverables: The ability to pick out words on the cover of a book to be used in the information collection of pictures or select photos of books.

GENERATION OF BOOK INFORMATION

Estimated Time: ~1-2 weeks

Dependent Tasks: 5, 6, 7, 8, 9

Deliverables: The ability to search for the information of a book via captured picture or saved photo.

Tasks

1 - CREATING UI STRUCTURE FOR A MULTI-VIEW XCODE PROJECT
2 - IMPLEMENT CAMERA CAPTURE FUNCTIONALITY
3 - IMPLEMENT PHOTO LIBRARY FUNCTIONALITY
4 - ENABLE COMPUTER VISION FRAMEWORK TO DETECT TEXT IN IMAGE
5 - SEND AND RECEIVE MESSAGES THROUGH OPEN LIBRARY API
6 - GENERATE BIBLIOGRAPHIES FROM OPEN LIBRARY INFORMATION
7 - GENERATE BOOK INFORMATION FROM OPEN LIBRARY INFORMATION
8 - DISPLAY GENERATED INFORMATION
9 - TEST FUNCTIONALITIES

Future Work

Additional Features

There are three additional features that would enhance the capabilities of CamLib. First, an option to view a history of captured books would allow users to not have to re-locate a book to generate information. Second, a way to autogenerate the ratings of books would benefit users who want to use CamLib as a tool for shopping. Third, an integration with Amazon would allow users to directly open the Amazon app to the book they are currently generating information for.

Plans for Future Releases

During the school year, there is little free time to develop additional features for CamLib. As a result, no future releases will occur during the school year. During winter and summer breaks, features can be developed and additional releases can be rolled out.

Multi-platform Support

There is currently no plan to incorporate multi-platform support. In the future, it might be possible to develop a complementary app for the iPad and one day develop similar apps for Android devices.

Fallback Strategy

The following are features that could be removed from CamLib in order for the app to be released on time.

LIMITED INFORMATION GENERATION

The goal of the information section is to supply users with bibliographies, summary about the book, author info, and more. It is possible to remove the many planed features and just supply users with generated bibliographies to simplify this feature for release.

NO SAVED PHOTOGRAPH SUPPORT

The app is planned to allow users to both take a live picture of a book or upload a saved photo of a book to the app. It will be possible to simplify the capability of the app by removing the ability to upload images from saved photos while maintaining the overall functionality of the app.

NO ISBN SUPPORT

As a last resort for if image detection fails is to allow users to enter the ISBN code of the book to the app. This feature was not originally in the scope of the project but was a recommendation given to me to allow a fallback strategy for users. Eliminating this feature from the initial release will save time during development.