Mail-Direct

Assignment 3- OSS Individual Project by Randell Carrido

Outline

About

- Option 3- Daily life problem that can be fixed with a new app.
- Mobile Application- Developing an app for iPhone. Languages: Swift for iOS and Java for Raspberry Pi.
- Project idea: Mail-Direct is a mobile application used to notify students when mail is currently in their mailboxes.

The Big Picture

We already have a system in TCNJ that sends emails to students to notify them when a package has arrived, but we don't have the same for mail such as letters that are put in our mail boxes. We must constantly check our mailbox to see if our mail has arrived. Mail-Direct eliminates this need for always having to check the mailbox for mail. The proposed system will use a sensor (Raspberry Pi) to detect when mail is present in a mailbox. This information will be updated

hourly and sent to a server in which the app can pull the information from when opened. Besides being a somewhat hi-tech project, Mail-Direct will also be one that is very useful for many students.

Back End

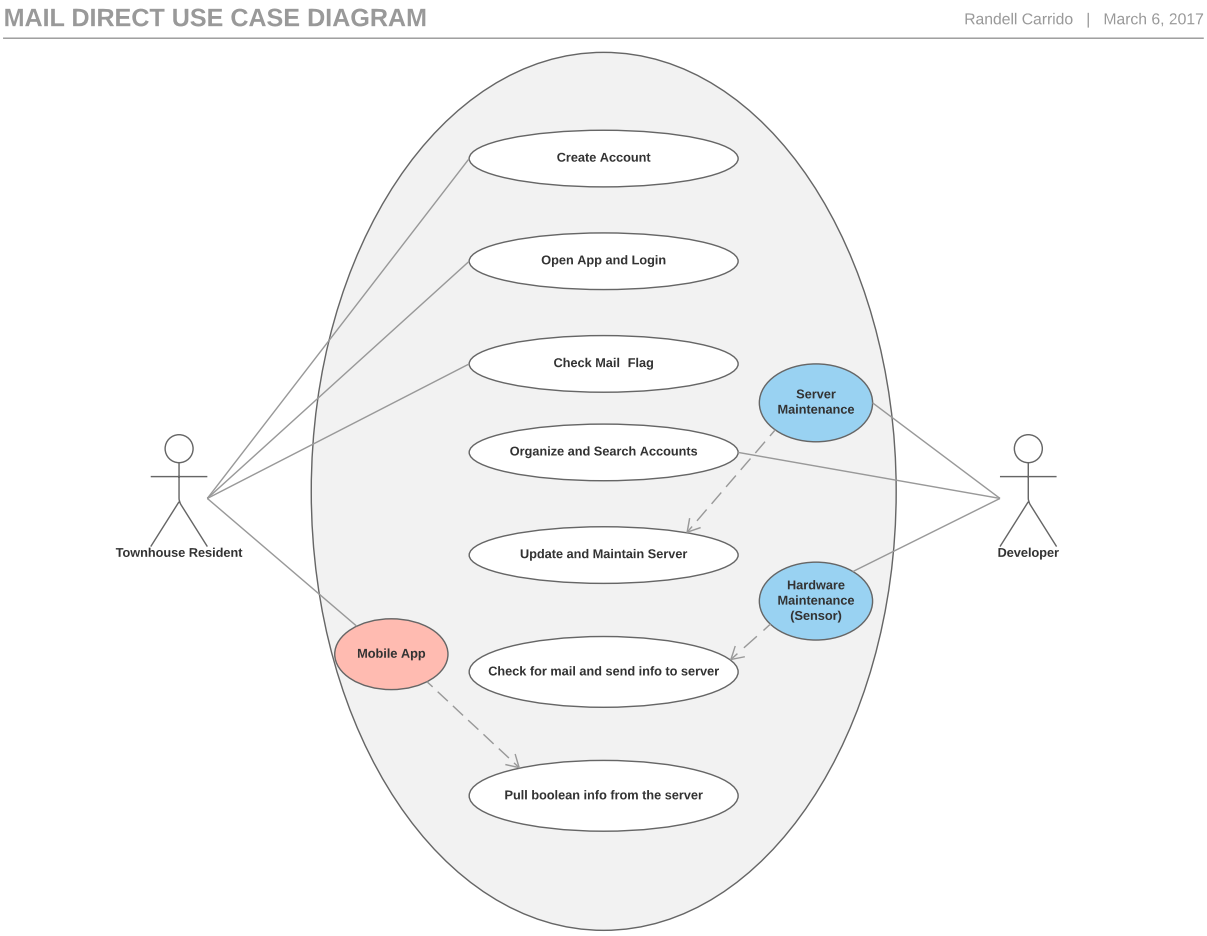
Algorithm
The sensor will be programmed to give Boolean values, returning true if it detects the presence of mail within the mailbox or returning false if does not detect any mail. Every hour, this information will be sent to a server (my personal server). I plan to have this app run in the background within mobile devices so that it can receive information for push notifications every hour or so when there is mail. If one does not want the app running in the background, an alternative is having the app pull in the information only when it is opened. Search and sort algorithms may be used as explained in the "Data Structure" section.

Data Structure
With a system that allows for sensors for each mailbox, I would use an array of size 20 which would hold townhouse/mailbox objects (the index number is the same as the house/mailbox number), each being able to hold 10 user accounts (because there are 10 people per townhouse). If working with only my mailbox, I would still use an array, but of size 10 to keep track of the users within the house. The array would hold user objects that hold information such as room letter, full name, password, cellular number, username, etc. It can then be sorted by a certain parameter using an efficient algorithm such as merge sort or quicksort and searched through using binary search (mainly for development and back end purposes).

New Software Engineering Concepts

- Programming with hardware tools (Raspberry Pi)
- Working with a server
- Using/building Raspberry Pi
- Account creation and organization of the information
- Account Security
- Allowing a mobile app and a sensor to connect online
- Coordinating all systems to work together (sensor, mobile app, and server)

Use Case Diagram



Plan for Learning New Material

For many of the questions I may have for implementing the whole system, I would usually resort to Google for the answers. For more in depth explanations of unfamiliar concepts, I would utilize online tutorials such as tutorialspoint.com, lynda.com, udemy.com, etc. In addition, I may also consult with a professor for any new confusing subjects related to my project.

GitHub

Repository Name: MailDirect
Full Pathname: Randellcar/MailDirect
Includes wiki, proposal, and license.

OSS License Comparison

The three most commonly used OSS licenses are MIT License, GNU General Public License Version 2.0, and Apache License Version 2.0. The MIT License is a short and simple permissive license that puts very limited restriction on reuse and modification of code. Under this license, people can do anything with the code if they give acknowledgement to the original developer and don't hold him/her liable. The GNU GPL Ver. 2.0 license is a copyleft license which allows end users the ability to run, share, and modify the software under the said license. Those who distribute or modify the code must make the source available under the same terms and give an express grant of patent rights from contributors to users. The Apache License Version 2.0 is a permissive license that requires preservation of copyright and license notices. Contributors must give an express grant of patent rights. Unlike the GNU GPL Ver. 2.0 license, Apache License 2.0 allows works and modifications to be distributed under different terms.

License for Project

Apache License Version 2.0 will be used for Mail-Direct. The license puts a main concern on patents, meaning that it provides an express grant of patent rights from contributors to users. Mail-Direct, in my opinion, is a very useful tool that possibly numerous people will want to improve and derive from. This makes Apache License Version 2.0 great for the project compared to MIT which is short, simple, and permissive and GNU GPL Ver. 2.0 which centers around sharing improvements.

References

https://blog.codinghorror.com/pick-a-license-any-license/
http://www.apache.org/licenses/LICENSE-2.0
http://www.gnu.org/licenses/gpl-2.0.html
https://opensource.org/licenses/mit-license.php
https://www.blackducksoftware.com/top-open-source-licenses