Home
Announcements
Textbooks
Calendar
Syllabus
Assignments
Site Settings
Roster
Mailtool
Gradebook
Library Reserves
Files & Resources
Piazza
Evaluation
Help

Assignments

Viewing assignment...

Settings for "2013 Individual Assignment 3: Testing"

Created by	Gail E. Kaiser
Date created	Oct 29, 2013 2:25 pm
Open	Nov 7, 2013 12:00 am
Due	Nov 14, 2013 5:00 pm
Accept Until	Nov 14, 2013 5:00 pm
Modified by instructor	Nov 7, 2013 11:17 am
Student Submissions	Single Uploaded File only
Number of resubmissions allowed	Unlimited
Accept Resubmission Until	Nov 14, 2013 5:00 pm
Grade	Points (max 20.0)
Alert:	No
Honor pledge:	Yes

Assignment Instructions

For this assignment, you will plan out testing for any two of the three systems described below. Base credit will be granted for the best two; extra credit will be granted if you complete all three. This is an *individual* assignment, you may not work together with the other members of your team (nor with anyone else).

For each system, describe the following for each of the highlighted use case(s). Additional use cases are provided only for context.

- Black Box Testing using Equivalence Partitions and Boundary Value Analysis (4 points). Describe specific equivalence classes and how you would use them to choose test cases. Describe specific boundaries and, again, how you would then devise test cases. Make sure to include the actual test cases (in prose).
- Black Box Testing using Metamorphic Testing (4 points). Describe specific metamorphic properties, and how you would use them to derive metamorphic test cases and check the results of those test cases. Make sure to include the actual test cases (in prose).
- Grey and White Box Testing (2 points). Plan out the specific approach you would take. Pretend we have given you the source code and you have access to the APIs (but not source code) for all external data sources.

A. CabTouch

The best way to get a yellow Taxicab quickly! The idea is to develop an application (mostly mobile) for tourists to New York who wish to take registered Yellow Taxicabs as against the black ones.

Drivers
1. Register for the application.
2. The available drivers (ready to take passengers) would set their status as “Available” and specify
their location.
3. The drivers who have picked up passengers or they are not in business update their status as

“Unavailable”.

Passengers
1. The users (potential passengers) of the app would allow for their location to be detected.
2. Depending on their location, the available drivers in/near the location would be returned.
3. When the user selects a particular driver, he/she can check information of that driver (Information provided by NYC Opendata plus other optimal data).
4. After check the driver, the users can choose to call the driver, or a notification (message/email) would be sent to the driver.
5. If the driver agrees to take the passenger, he notifies the passenger and updates his status to
“unavailable”.

B. Library+

Sometimes when we want to borrow books from libraries for a course project, it would be really frustrating if we had to find the availability of the book by searching from library to library, especially when the book is “horribly” unpopular and rare. Life would become much easier were we able to get such information in one place, by several simple clicks.

Library+ is a iOS app which allows users to search for nearby libraries as well as information about books they are keeping. A user can specify the title of the book they’d like to borrow and the app returns a list of libraries having this book, along with their locations shown on a map. Information on whether this book has been checked out will also be available for the user, by either displaying it on the app or showing a link of the library webpage to the user. Some basic books info such as author, publisher, abstract will be provided for helping the user find the right book.

Case 1: User searches for nearby libraries
A user who is looking for a library provides the location where she’d like to stay. The app gets the location, searches its database and retrieves coordinates of those libraries that meet the user’s requirement. And it also provides the library general information including the link of the library website, the opening hours and days, etc. Then the app returns the results to the user by showing them on a map.

Case 2: Find the optimal path to the Libraries searched by case 1
A user can follow the directions by our app to reach the library in shortest time. Users choose three methods: subway, walking, or driving.

Case 3: User searches for the availability of a book
A user, who is thinking about borrowing a book and has no idea where she could get one, provides the title (probably other information such as ISBN) of the book to the app. After searching the database, the app returns a list of libraries that are holding this book. A list of books meeting the same searching requirements would be returned as well, with their basic info, to help the user find the right one. Once both the right book and the libraries that hold it were found, the user might be able to know the status of this book in each library (e.g. checked out or not). (depends on whether the library provides a way of performing such a query.). The app will also provide the users with a list of books that are similar to the query content if we fail to get any result for the initial query.

C. The Social Circle

Today, the web holds so much information that can be used at any point of time to make a ton of decisions. This application aims to help a user figure what he would want to do with all the resources available next to him. Basically, the user would enter his zip code and the application would present him with all the possible places he could visit in terms of the following options

1) Night life
2) Restaurants
3) Café
4) Land marks
5) Movies

As an application we also further aim to provide the user with information related to the nutrition value of the food that he chooses to have. All in all, this would act as a go to for anyone who would like to spend his time in the best possible way