

Coupons Nao!

Software Requirements Specification

COP4331_001, Spring 2011

Modification history:

Version	Date	Who	Comment
v0.0	2/7/2011	Stephen Bryant	Filling out basic information
v0.1	2/10/2011	Stephen Bryant and Daniel Kaplan	Finishing form
V0.1	02/14/11	Stephen Bryant	
V0.1	02/15/11	Daniel Kaplan	Made minor updates to document

Team Name: The A-Team

Team Members:

- Stephen Bryant - sbryant31@gmail.com
- Taylor Kourim - tkourim@knights.ucf.edu
- Daniel Kaplan - kaplan@knights.ucf.edu
- Isaiah Walker - senthose@gmail.com
- Chris Nergard - cnergard@gmail.com

Contents of this Document

- Introduction
 - Software to be Produced
 - Reference Documents
 - Applicable Standards
 - Definition, Acronyms, and Abbreviations
- Product Overview
 - Assumptions
 - Stakeholders
 - Event Table
 - Use Case Diagram
 - Use Case Descriptions
- Specific Requirements
 - Functional Requirements
 - Interface Requirements
 - Physical Environment Requirements
 - Users and Human Factors Requirements
 - Documentation Requirements
 - Data Requirements
 - Resource Requirements
 - Security Requirements

- Quality Assurance Requirements
- Supporting Material

SECTION 1: Introduction

Software to be Produced:

We intend to build an Android application which locates a user via GPS, projects their location onto a Google map, and then queries a SQL database to find coupons which are at locations close to them.

Reference Documents:

- [Concept of Operations](#)
- [Project Management Plan](#)
- [Test Plan](#)

Applicable Standards:

See Project Management Plan

Definitions, Acronyms, and Abbreviations:

None

SECTION 2: Product Overview

Assumptions:

- The android phone will be able to run our application in real time
- Drawing the map will present no performance problems
- The phones will always be connected to the internet.
- Google maps can easily be interfaced with android's GPS system's location features.
- Android can send http request to the server and retrieve query results from the database.

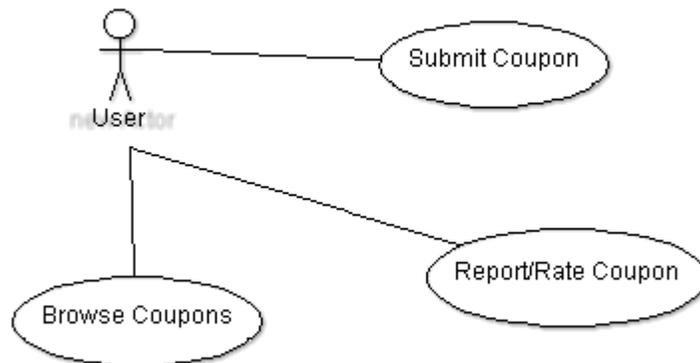
Stakeholders:

- The A-Team. This project, if properly executed, will provide us with great career experience, thus improving our employability.
- Android phone users. This project could help people easily find good deals in their local area. It could really help people who are receptive to it.

Event Table:

EVENT NAME	EXTERNAL STIMULI	EXTERNAL RESPONSES	INTERNAL DATA AND STATE
Load & Browse Coupons	Open app or click browse	Get location and display coupons for that area	User's location list of coupons
Rate/Report coupon	Thumbs up button or Thumbs down button	Disable rating for that coupon if thumbs down	Coupon is now rated by user
Submit coupon	Click submit coupon	Loads coupon storing submission form	Submit coupon data to server

Use Case Diagram:



Use Case Descriptions:

- See event log

SECTION 3: Specific Requirements

No 1: User interface
Statement: When a user opens the app for the first time, the app will read their location and present them with coupons and deals based on their location
Source: TA
Dependency: NA
Conflicts: None
Supporting Materials: None
Evaluation Method: Program loads location and presents user with coupons
Revision History: Stephen Bryant and Daniel Kaplan (2/10/11 added basics)

No 2: Backend
Statement: . A database will keep track of major businesses and their locations, and another table will keep track of coupons and deals for each business
Source: TA
Dependency: None
Conflicts: None
Supporting Materials: None
Evaluation Method: Database responds to database queries corectly
Revision History: Stephen Bryant and Daniel Kaplan (2/10/11 added basics)

No 3: Database Upkeep
Statement: . Users are able to add and rate coupons
Source: TA
Dependency: Number 2
Conflicts: None
Supporting Materials: None
Evaluation Method: Users are able to add and rate coupons
Revision History: Stephen Bryant and Daniel Kaplan (2/10/11 added basics)

3.1 Functional Requirements:

- Check inputs for special characters before issuing SQL queries.
- Notify user if communication with GPS or Internet fails and give the option to retry.
- Get user location from GPS-> submit to Database for retrieval of nearby businesses -> display list/map of nearby businesses
- Get user rating from client -> submit rating to server for coupon update -> update display based on rating
- Get user button clicks -> display resulting page
- Get user sign up info ->Verify that new users don't already exist in the system -> submit to Database
- Get user login info -> verify in database
- Get coupon image, location, title -> Submit to database

3.2 Interface Requirements:

- Database Interfacing
 - coupons, locations, and users are inputs
 - output consists of XML files or JPEG files
 - coupons will be in JPEG images and/or XML files.
 - locations will be in GPS coordinates.
 - Users will be XML files with information such as UserID, etc..
 - Locations should be accurate enough to order by distance.
 - Locations should be sent every time the reload button is clicked.
- Google Maps Interfacing
 - GPS locations are inputs
 - Output consists of a map with selected input locations highlighted
 - Map output should be fairly accurate.
 - Map output will be sent every time the reload button is clicked.

3.3 Physical Environment Requirements:

- Android phone
- Internet connection
- GPS capability

3.4 Users and Human Factors Requirements:

- Users need little to no prior knowledge to run application
- By clicking on coupon it will display the coupon in full screen in a larger format
- System will detect invalid inputs in any form that submits to the server to prevent possible hacking

3.5 Documentation Requirements:

- Documentation will be provided in the form of artifacts, a user manual, and Javadoc.
- The Javadoc will only be useful to java programmers familiar with the android API.
- The high level artifacts can be read by any business person, but the more technical artifacts will contain increasing levels of technical jargon.
- The manual will be able to be read by any lay user.

3.6 Data Requirements:

- One data calculation we will use is mapping coordinates from long/lat to Google maps API format.
- Coupon objects will be in XML format.
- User Ids will be obtained from the user's phone and validated for security

3.7 Resource Requirements:

- Skilled personnel required to build, use, and maintain the system: A-team
- Physical space: Server
- Schedule: See Concept of Operations
- Funding: none
- Hardware/software tools: Eclipse, ADT, Google API, MySQL, and PHP

3.8 Security Requirements:

- All queried data must be verified to check for possible hijacking scripts
- All users will have their activities logged by their unique phone ID. This will allow us to ban users who are submitting vast amounts of fake coupons or abusing our software in any other way.
- Server will be backed up based on administrators request

3.9 Quality Assurance Requirements

- When the user receives a list of coupons and companies from the server it will keep that data stored on the phone temporarily for that session so if the server goes down for some reason the user can still view coupons they already have queried.
- If the user loses one of the requirements (for example internet or GPS connection) the program will prompt the user to reconnect.
- We will have a script on the server to show the server status. The phone can access this script to get the status so the phone will know if it is a server problem or a phone problem.
- Users can submit bug reports and comments through the Android marketplace.

SECTION 4: Supporting Material

None.