# Table of Contents

[**1. Introduction……………………………………….…….………………………………….…1**](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772091)

[1.1 Purpose 1](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772092)

[1.2 Scope 1](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772093)

[1.3 Overview of Document 1](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772093)

### [2. System…………...…………………………………..……………………..…………………..1](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772096)

2.[1 Target Environment 1](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)

[2.2 Target Market 1](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)

[2.3 Major Classes 2](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)

[2.4 Functional Requirements 2](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)

[2.5 User Interface 2](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)

[2.5.1 Restaurant Selection………………………………………………………………](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)3

[2.5.2 Game……………………...………………………………………………………](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)3

[2.5.3 Results….…………………………………………………………………………](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)4

[2.5.4 Help Window………………………………..……………………………………](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)4

[2.5.5 Settings…………….…….……………..…………………………………………](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)5

[2.6 Non-Functional Requirements](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772093)

[2.6.1 Scheduling..………………………………………………………………](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)5

[2.6.2 Cost……….…………...…….…..………………………………………](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)6

[2.6.3 Technical…….……………….……………………………………..……](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)6

[2.6.4 Performance………………….……………………………………..……](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)6

[2.6.5 Security……………………….………...…………………………..……](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772095)7

### [3. Other Deliverables……..……………………………………………………………………..7](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772099)

### [4. Risk Analysis………………..……………...............................................................................8](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772102)

[4.1 Technical Risk Assessment 8](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772103)

[4.2 Management Risk Assessment 8](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772104)

### [5. Glossary…………..….………………………………………………………………………..8](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772105)

### 6[. References………...………….……………………………………………...………………..9](file:///C:\Users\Manohar\Desktop\Proj%20plan\Project_Plan_Draft%5b1%5d.docx#_Toc213772105)

**1. Introduction**

* 1. **Purpose**

The purpose of this document is to describe software that will provide Google Android smart phone users an entertaining method of selecting a restaurant for dining that meets specific user input criteria.

* 1. **Scope**

The software application will feature an Android touch screen graphical user interface. The theme of the interface will be that of a television game show. The application will utilize user input data to determine which restaurants satisfy the user’s criteria. The user input criteria will include: type of food desired (Mexican, Italian, Pizza, Burgers, etc.), distance to restaurant, style of restaurant (buffet, fast food, casual, etc.), and expense. The restaurants which meet the requirements will randomly populate a GUI control which, when operated, will provide an answer in the form of a restaurant name and location.

* 1. **Overview of the Document**

This document outlines the major functionalities of the system and requirements specified by the customer. The documentation includes a description of the target environment that this system is planned to be deployed, the users that this system is intended for, the major functional requirements of the system, non-functional requirements that specify issues such as cost of the project , technical environment and performance of the system. The documentation also includes description about maintenance plan and risks that may arise in the development process of the system.

1. **System**
   1. **Target Environment**

The application is to be for use on the Samsung Captivate Google Android smart phone running one of the following operating systems: Google Android 2.1, 2.2, and 2..3. Operation of the application will require that the phone have activated cellular service with data plan and sufficient signal strength.

* 1. **Target Market**

All Android smart phone users age16 and up living in or traveling to Wichita Falls, TX that enjoy dining out.

* 1. **Major Classes**

The Lunch Decider software will possess two major classes. These two classes are Restaurant and User.

A **restaurant** will consist of the data variables listed below. Restaurants will be stored in a restaurant database.

* Restaurant Name
* Location Coordinates (latitude and longitude)
* Type of Food
* Style of Restaurant (only if data is available)
* Price (only if data is available)

A **user** will consist of the data variables listed below. These variables represent the user's desired criteria for restaurant selection.

* Maximum Driving Distance
* Type of Food
* Style of Restaurant
* Maximum Price
  1. **Functional Requirements**

The following major functions will be performed by the Lunch Decider application.

**Build/Update Database** This operation will construct a database containing information regarding all restaurants within the city limits of Wichita Falls, TX. The database will be built/updated by accessing web based services to obtain the required restaurant data.

**Obtain User Location**

Utilize the built in GPS capabilities of the smart phone device to obtain the user's current location coordinates and pass this data to the application.

**Collect User Input Criteria**

Display a graphical user interface (GUI) to enable the user to input his/her desired restaurant criteria values into the application. The restaurant criteria will include the following.

**Return Restaurant**

Using a random selection technique, select a restaurant from the subset of restaurants in the database which meet the user input criteria. Return the name and location (address) of selected restaurant.

* 1. **User Interface**

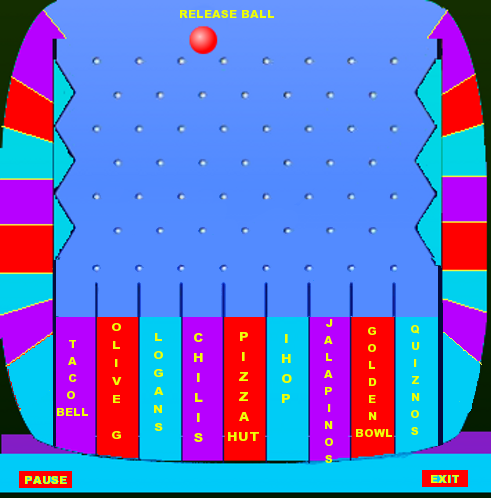
The user interface is designed for Android Smartphone. The primary user interface design involves five screens.

* + 1. **Restaurant Selection**

The **Restaurant Selection** interface allows the user to input filter criteria for the restaurant selection. The user has the option to select the style of the restaurant, type of food, cost and maximum driving direction on this screen. After making the selection, the user can launch the game by touching the launch Game button.

The help button allows the user to open a help window that would allow the user to get information on how to use the application. More detail about the help screen is provided on section 2.5.4.

The settings button placed on the top of the screen opens the settings window. More detail on the settings window is provided on section 2.5.5.



* + 1. **Game**

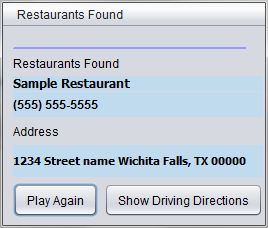
The **Game** interface selects a restaurant randomly from the available restaurants list. Restaurants meeting the criteria are put along the bottom of the game screen. User clicks on the ball which is placed at the top of the screen to start the game.

User has the option of suspending the game by touching the pause button. The button can also be used to resume the game once the game is in pause mode.

The user can also use the exit button to exit from the current game and will be returned back to the restaurant selection window.

* + 1. **Results**

The **Results** interface displays the selected by the Game. User has the option of playing the game again or can touch the show driving directions button to have a look at the driving directions of the selected restaurant.



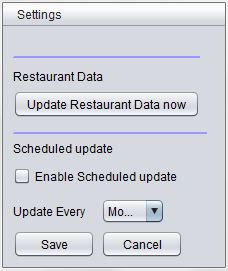
* + 1. **Help Window**

The help window allows the user to get instructions on how to use the system and includes the rules on how to play the lunch selector game.



* + 1. **Settings**

The **Setting** interface allows the user to change the way the restaurant data is updated. The user will have the options to update restaurant Data using the Update the Restaurant Data button, schedule a future update and turning on and off the sound.



* 1. **Non-Functional Requirements**

The non-functional requirements for the Lunch Decider project include scheduling, cost, software quality, and reliability.

* + 1. **Scheduling**

Listed below are the six major project milestones along with their projected completion dates.

* Requirements Specification Document - March 3, 2011
* Project Plan - March 10, 2011
* System Design Document - March 28, 2011
* Test Plan and Test Report – April 29, 2011
* Deployment of Application - May 6, 2011
* Project Presentation and Evaluation Report - May 10, 2011
  + 1. **Cost**

As this project is developed by students as a partial fulfillment of a master's degree, there are to be no costs associated with this software application. Therefore all resources required for development must be open source and/or available at no cost.

The application is to be available to users for no cost. However, the application will require an Android smart phone with an activated service plan. Therefore the users will be subject to the terms and agreements of the service provider.

* + 1. **Technical**
       1. **Target Environment**

The application is to be for use on Google Android smart phones running the following operating systems: Google Android 2.1, 2.2, and 2.3. Operation of the application will require that the phone have activated cellular service with data plan and sufficient signal strength. Application will require that the device have 4MB of available memory.

* + - 1. **Development Environment**

The Java API for Android Smart phone provided by the Android SDK, the Android SDK version 2.3, the Android emulator plug-in for Eclipse and Eclipse-Helios 3.6 development IDE will be used as development environment. In addition, Google Custom Search API, Google Maps API, Yelp and Urbanspoon will be used as service providers.

* + - 1. **Programming Language**

The Java programming language

* + 1. **Performance**
       1. **Efficiency**

1. The total device memory required for the complete application will be 4MB or less in size. The complete application consists of the application program and the application database. The application program is to be 2MB or less in size. Initial application database creation and subsequent database updates are each to also be 2MB or less in size. This will facilitate quick download times and minimize data plan usage.
2. Upon the user selecting to run the application, the load time of the application will not exceed 4 seconds. Once loaded, the time required to switch between GUI screens will not exceed 2 seconds. This time is measured from the time the user touches a control which causes a screen switch to the time that the newly selected screen is displayed. Note: performance stated is for use on an average performing device and may vary due to performance limitations of various Android smart phones.
   * + 1. **Reliability**

In order to make the system reliable, it will be tested during the development process. Testing techniques such as system testing, unit testing and integration testing will be conducted during the development process.

* + - 1. **Usability**

The use of the application will be very easy and intuitive so that 95% of the target market will be able to operate the application without the need for instructions or training. However, a help screen will be included and accessible within the application.

* + 1. **Security**

The system might collect private data such as current user location for its optimal operation. Such kind of data will never be collected without notifying and getting permission from the user. In addition the system will run on a sandbox environment that is provided by the Java platform. The sandbox environment will provide extra security feature in such a way that the application will have restricted access on external features of the whole system.

1. **Other Deliverables**

The deliverables of the system will include the following.

* Requirement Specification document
* Application Description Document
* Project plan document
* System Analysis Document
* System Design Document
* Test plan document
* System evaluation document
* Team evaluation document
* Different designs and diagrams
* Final Completed Application

1. **Risk Analysis**
   1. **Technical Risk Assessment**

Since the Android Application development environment is relatively new, team members might face a problem of having adequate knowledge about the development environment.

To tackle this problem, the project team members will use online resources and update their technical skill about the development environment.

* 1. **Management Risk Assessment**

The team has assessed that there might be a risk in integrating the documentation of the project during the development process. The team will use integration tools such as Dropbox to mitigate such kind of risks. In addition meeting schedules will be planned to enable team members communicate and update each other on the work progress.

1. **Glossary**

* API: a collection of application program interfaces that allow software developers to obtain services from a service provider.
* Emulator: An emulator in [computing](http://en.wikipedia.org/wiki/Computing) duplicates the functions of one [system](http://en.wikipedia.org/wiki/System) using a different system.
* Google Android: The operating system that is provided by Google for smart phones.
* GPS: Global Positioning System.
* GUI: Graphical User Interface
* IDE (Integrated Development Environment): A software development system. An IDE provides a comprehensive set of tools for the development of software.
* Java: A high level programming language.
* SDK: A software development kit is typically a set of development tools that allows for the creation of [applications](http://en.wikipedia.org/wiki/Application_software) for a certain [software](http://en.wikipedia.org/wiki/Software) package
* Smart Phone: A mobile phone that offers more advanced computing ability and connectivity than a traditional style mobile phone. A smart phone typically has features usually only found on a personal computer.
* UrbanSpoon: A web based restaurant data service Provider Company.
* Yelp: A web based data service Provider Company.

1. **References**

* Stephen R. Schach, *Object-Oriented and Classical Software Engineering*. 8th. McGraw-Hill, 2010.
* Pressman, Roger. *Software Engineering: A Practitioner's Approach*. 6th. McGraw-Hill Science/Engineering/Math, 2007
* Wikipedia - http://en.wikipedia.org/wiki/Eclipse\_IDE
* Wikipedia - http://en.wikipedia.org/wiki/Integrated\_development\_environment
* Wikipedia - http://en.wikipedia.org/wiki/Smart\_phone
* Wikipedia - http://en.wikipedia.org/wiki/Api