**Table of Contents**

1. Introduction
2. Purpose
3. Scope
4. System Architecture
5. Functional Requirements
6. User Dialogs & Control Flow
7. Background Tasks
8. Database Models
9. Interfaces to Other Systems
10. Nonfunctional Requirements
11. Terms

**Introduction**

The purpose of this document is to collect, analyze, and define high-level needs and features of Foodex. It focuses on the capabilities needed by the stakeholders, and the target users, and why these needs exist. The details of how Foodex fulfills needs are detailed in the user manual.

**Purpose**

Grocery indexing and suggestive application based on what the users have bought, including packaged and fresh food. This application will suggest easy everyday recipes to help to create an efficient and healthy lifestyle.

**Scope**

Foodex — an application developed and intended for mobile devices — is capable of sending notification and social media sharing. The early development process involves using APIs, algorithms, and databases, which will work together to generate a list of food and recipes based on user input.

**System Architecture**

The system will contain APIs, algorithms, and databases. They will all intertwine and communicate with one another. The APIs and databases will be accessed first from the user input, and then the algorithms will take place next. The algorithms will convert and make sense of the data into recipes for the user.

**Functional Requirements**

The functional requirements are to decipher text from a picture, take user input, read and pull up relations from a database, recommend relative recipes, send push notifications, and share to social media.

**User Dialogs & Control Flow**

* 1. Input food or take picture of recipe
  2. Food will be sorted
  3. Algorithm take place to find recipes

**Background Tasks**

The background tasks are the databases and APIs being utilized, pictures will be converted and deciphered, recipes will be determined by the recipe selection algorithm, push notifications will be sent more aggressively when application is unfrequently used, and the transition from the picture taken to social media.

**Database Model**

* + Food
    - Keys: Name, nutrition, expiration date
  + Recipe
    - Keys: Ingredients, nutrition
  + Social Media
    - Keys: User account, picture

**Interfaces to Other Systems**

* + Facebook, Twitter, Instagram, Snapchat
    1. Interfaces to connect to social media to share used recipe

**Nonfunctional Requirements**

* + User interface must look presentable
    1. Straightforward, intuitive, and easily understood
  + Application must run fast, reliable, efficient, and provide a seamless experience
  + Can run on both mobile platforms: Android and iOS
  + Login security should be encrypted for safety
  + Requires internet access

**Terms**

1. Database: Structured set of data
2. API (Application Programming Interface): Set of tools for building software applications, how software applications should interact
3. Social Media: An Internet-required medium for users to create and share content