

LameDucks Coffee Cart Reward System

Risk Assessment

TEAM 007
Version: Final

Overview

This document describes the risks involved and their mitigation throughout the development process of the LameDucks Coffee Cart Reward System(CCRS). The risks will be ordered by greatest threat and will explain the description, likelihood, impact and mitigation strategy of each risk.

| | |
|----------------------------|---|
| Risk - Business | <i>VIP card adoption</i> |
| Threat Level | High |
| Description | This is a new system and a new business process. The adoption of a new business process ultimately can define the success of the software system as well. |
| Likelihood | The likelihood of the reward system will be reliant upon the clients ability to market and grow the business process. |
| Impact | If the VIP rewards card is not needed or adopted by the clients customers than the system could be deemed disposable. |
| Mitigation Strategy | None |

| | |
|----------------------------|---|
| Risk - Technical | <i>Network connectivity for Android devices can be unreliable</i> |
| Threat Level | High |
| Description | The application will need to be in-sync with the other coffee cart locations. Unreliable network connectivity could make syncing problematic. |
| Likelihood | The likelihood of network connectivity being unreliable is dependent upon the location of the coffee cart. |
| Impact | If network connectivity is unavailable for a significant amount of time there could be rewards that go unaccounted for. This could also lead to pre-ordered items being over ordered or not accounted for. |
| Mitigation Strategy | A set of instructions on how to verify network connectivity by location could be created. There are also ways to enable Android devices to use a Locally connected internet connection which is more reliable than depending on cellular or other wireless technologies. We can also implement a hybrid approach to store data locally until a network connection is made and then the data will be synced. |

| | |
|----------------------------|---|
| Risk - Technical | <i>Owners deleted a product from Parse Database</i> |
| Threat Level | High |
| Description | The application ordering and pre-order system uses a relationship table to product. If the the owners delete a product instead of making it inactive, the relationship will not be connected to something that exist, causing the app to to work or display data correctly. |
| Likelihood | The likelihood of this happening is lower assuming proper training of the owners |
| Impact | Improper data is displayed in the app for report and purchase histories. The app will still function. |
| Mitigation Strategy | Train owners to properly use and update the Parse database. Eventually build a backend system for the owners. |

| | |
|----------------------------|---|
| Risk - Technical | <i>Information deleted from the system is removed permanently and not recoverable.</i> |
| Threat Level | High |
| Description | The application has the ability to remove VIP customers from the database. While these actions require a confirmation, if deleted accidentally all information associated with the VIP Customer will be lost. |
| Likelihood | The likelihood of this happening is dependent on the Owners and Managers training on the system. |
| Impact | If a VIP Customer is removed by mistake all order and reward history will be lost. |
| Mitigation Strategy | Communicate that once a VIP customer is removed from the application it is not recoverable and to proceed with caution. |

| | |
|----------------------------|--|
| Risk - Technical | <i>Android feature fragmentation.</i> |
| Threat Level | High |
| Description | Android is a widely used platform that has thousands of compatible devices. Having consistent hardware and API level can be a challenge when designing this system. |
| Likelihood | The likelihood of our hardware and API level being different is fair high as the Android platform is so widely used. |
| Impact | Having inconsistent hardware and API level can cause a disarray of expected outcomes to features within the system. In some cases it could render the system unusable. |
| Mitigation Strategy | Having the client purchase a specific device, or subset, will allow our team to design to test against a specific API level and hardware. |

| | |
|----------------------------|---|
| Risk - Technical | <i>Third Party Platform Dependency</i> |
| Threat Level | Moderate |
| Description | Using a third party platform to manage our backend syncing will save time on the development time needed for the project, but will increase the risk in the event the third party company ends support in the future. |
| Likelihood | The likelihood of our third party ending support is fairly low as it's their main product and they have many users who depend on their product. |
| Impact | The backend syncing will have to be migrated to a new platform or rewritten by the team. This could also result in some refactoring depending on the new platforms requirements. |
| Mitigation Strategy | Select a third party platform who is reputable and active. Selecting a platform that is open sourced would also allow the team to fork the platform in the event the third party decides to end support for the platform. |

| | |
|----------------------------|---|
| Risk - Business | <i>Inadequate Training for Managers to use the system</i> |
| Threat Level | Moderate |
| Description | Manager may be unfamiliar with use of smart phones, the Android system, or how to use and navigate the app. They may also not know what to do in the event of technical problems or bugs. |
| Likelihood | The likelihood of inadequate training is common in implementations of new systems and will be the responsibility of LameDucks. |
| Impact | If the app is not correctly used or not used because it is seen as too much trouble then the system could be deemed disposable. |
| Mitigation Strategy | Recommend training for Coffee Cart Managers. |
