# Front Matter

## Table of contents

# Introduction

Have you ever thought about finding a good or new burger around town? Well, our client Ammar Shallal, thought the same thing and decided to do something about it. He and a group of friends that were on a mission to find the best burgers in New York City founded the **Burgerator** app for iOS. The Burgerator app allows users to rate, find, and keep track of the burgers they’ve had. It’s a wonderfully simple concept, and works well. However, not everyone has an iPhone, and Ammar wanted his brainchild to be accessible by a larger audience. That is where the Hamburgerler’s come in. Our team will be developing an Android port of the app and will perform some new feature requests from the client. The new features will include more social interaction options for users, such as the ability to follow other users to see what great (or gross) burgers they’re eating. Additionally, we will be looking at performing a database restructure that moves the focus from the burger joints to the burgers themselves, and a few other requirements that will heighten the user experience of the Burgerator.

# Website

Our website is <https://trello.com/hamburgerlers>

# Project Overview

# Project Management Plan

# Requirements

**Development, Operation, and Maintenance Environments**

Android Burgerator will run as a native Android application on Android devices. Development, operation, and maintenance will utilize physical Android phones as well as virtual Android emulators.

TODO: Min Android API?

TODO: Target Android API?

**System Model**

* + High-level view showing the major components of the existing and proposed system:
  + Existing System:

IOS Burgerator (fig 1)



* IOS Burgerator is what currently exists. It is a mobile application that allows users to rate burgers within a geographic location (predominantly used in New York). Android Burgerator Base is intended to be an exact replication of IOS Burgerator.
  + Proposed Systems:

Android Burgerator Goal

Android Burgerator Base

* The proposed system Android Burgerator Base is ideally identical to IOS Burgerator. However, the Android Burgerator Goal is intended to be a more flexible extension of Burgerator that incorporates more social media aspects into Burgerator. For example, allowing you to share burgers with friends.
* TODO: Include a system level diagram for Android Burgerator.

**User Interaction**.

* + The user of the mobile application Andorid Burgerator Base, which is currently the focus for the project, is to allow individuals to visit a restaurant, take photos of a hamburger that they ordered and leave a rating for the burger in question.
  + Use-case diagrams and scenarios are an effective way to describe the interaction.
  + Refer to the use case diagram and corresponding scenarios(figs x-x)

**Functional Requirements**

* + The application must:
  + Allow the user to sign in(email, facebook, twitter)
  + Allow the user to review five main sections of the application(Find a burger,Burger Feed, burger rating, top burgers, user profile)
  + Allow the user to logout
  + Allow the user to control setting such as location from within the application

**Nonfunctional Requirements**. Detail the constraints under which your system must operate.

* + Given that Burgerator is location based, there must be access to location or a manual way to enter the location.
  + Constraints that the hardware imposes on the application are the same that other applications have. Memory, data, and battery constraints should be minimal.
  + The portability of the project is apparent given the underlying android Platform. This advantage opens up to application to the majority of the mobile maketshare.
  + The reliability of the application will rely on the servers that support it.

**Feasibility**.

* + Android base base,
    1. Base System Diagram(fix x)



* + android base,
    1. System diagram(fig x)
  + android goal base,
    1. TBD
  + android goal
    1. TBD

# Conclusion