# DESIGN DOCUMENT


ASTRAEA: *Guided by the stars*

**Prepared By: Damen Tomassi, Ryan Hudson, Nick Hand, Matt Paule, Mike Marchina, Jacob Kershaw**

**Organization: Senior Project - Team ASTRAEA**

**Date: Feb 4, 2017**

**Version: 1.0**

**Table of Contents**

## 1. Purpose

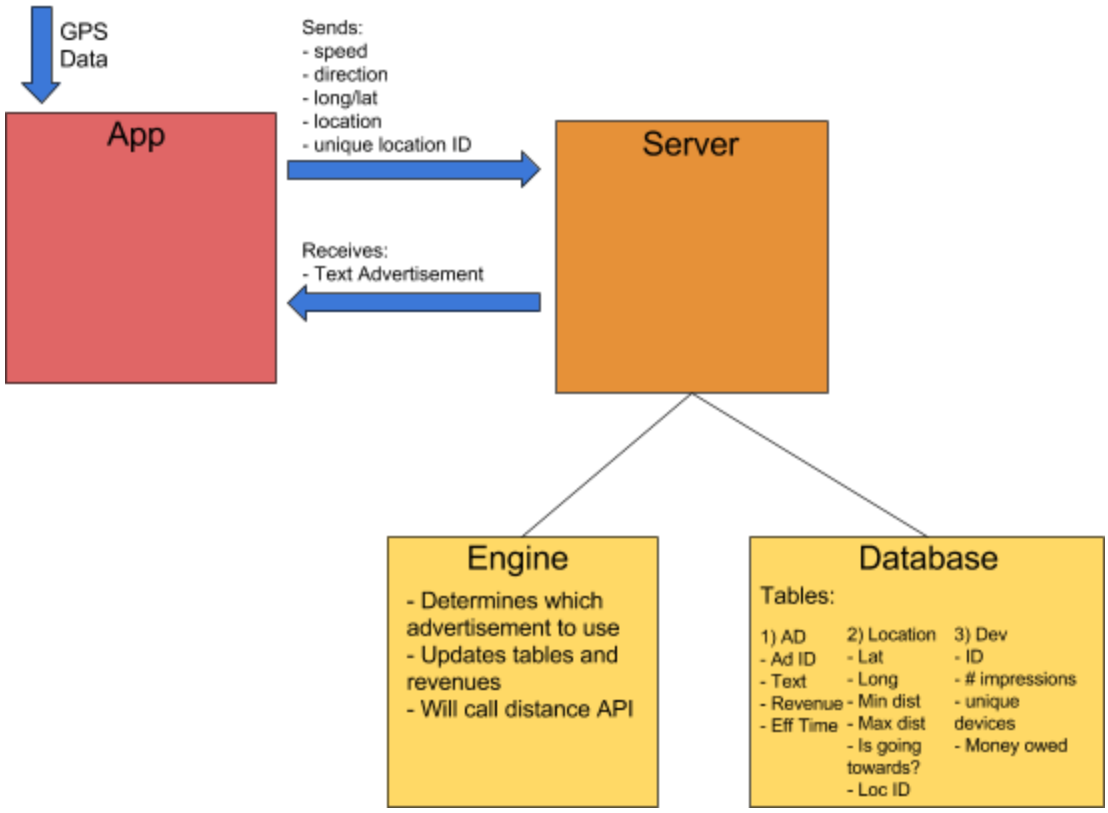
This document presents the Software Development Lifecycle (SDLC) Design / Installation Information for the development and implementation of a system capable of delivering hyper localized targeted advertisements to various users based upon their current positioning vector. The document also outlines the technical design of the ASTRAEA Advertising System and provides an overview for the ASTRAEA Advertising Systems Implementation.

## 2. Scope

Position, distance, direction, and range all are important components used to calculate which advertisements should be delivered to a user. The scope of the Hyper Localized Vector Based Targeted Advertisement System is to create an API that utilizes these components to deliver relevant, proprietary advertisements to the user. The advertisement is targeted to the user based on the user's current vector location. The implementation for demonstration purposes will be an application that showcases how the API can be consumed. Advertisements will be manually entered into the database.
*Location will be test constrained to Glassboro, NJ.

## 3. Solution Design Overview

Describes the approach, architectural goals and constraints, guiding principles used in design and development.




# 4. Technical Architecture

## 4.1 Hardware Inventory, Specifications and Locations

### 4.1.1 Server

A single cloud based computer running the application will be acting as a server to host all the data for the project.

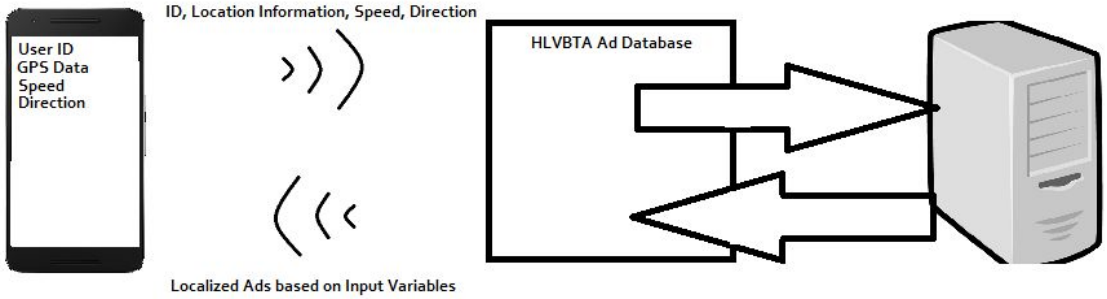
### 4.1.2 Input / Output Devices

- Android Phone
- Personal Computer

### 4.1.3 Middleware Hosting

- MySQL Workbench (version #6.3): MySQL is the most popular open source database software. It is easy to use, fast and reliable.
- Map distance Matrix API
- Google Places API
- SparkJava
- Hibernate - Pipeline between Java and MySQL

## 4.2 Physical Layout




# 5. Configuration Specification

The configuration specification defines the required software configurations for the software in the system, this consists of configuration settings and/or parameters. Any operational description is limited to clarification or refinement of information presented within the design specification.

This includes the following:

- Android OS Marshmallow or greater running on a modern android device, with an active wireless connection.
- Required software module setup parameters.

- Configuration parameters for any associated utility programs, for example, reporting tools, data mapping/transfer tools.
- Logical security parameters
- Configuration of database queries and parameters.
- Hardware components configuration, for example, servers, mobile devices, etc.
- Network and port configuration parameters, for example, server names, domains, DNS server
- Client configuration parameters
- Utility configuration

## 6. Solution Design Specification

### 6.1 Software Description

The system is going to be used with a large number of mobile devices, so it will need to be designed to coordinate with multiple users at what appears to the user to be simultaneously. We will utilize MySQL workbench as a database, as it has open source advantages. We will also utilize Java to write the mobile application. We will be using the following softwares for this project:

- NetBeans, Intellij and Eclipse for java developing the source code.
- Github for decoding between all members of teams.
- Google Docs for document writing.
- Android Studios for developing the mobile application.

### 6.2 Coding Standards

No specific coding standards will be practiced during the development of this application.

### 6.3 Solution Data, Information View, and Data Requirements

**Database**

- Default name of the database is “database.db” and is located in a folder called “data”, which will be created in the directory the .JAR executable is held in.
- The database will utilize MYSQL and will connect to the application using hibernate as a pipeline.
- The advertisements will be stored in the AD table in the database and will be pulled from the database using the corresponding call in Java.

**Advertisements**

- The advertisements will be stored as plaintext in the database and will be pulled from the database as needed.
- Location will be tracked and that will be the deciding factor in which ad gets selected.

**Database Tables**

| AD | Location | Dev |
|---|---|---|
| ID | ID | ID |
| Display Text | Latitude | Number of Impressions |
| Revenue | Longitude | Unique devices |
| Efficient Time | Min Distance | Money owned |
| | Max Distance | |
| | Is going towards? | |

The most important component of the system is the database design. It can be easily realized, if we consider the amount and the scale of the inventories. To have a strong and flexible system, the design must be well studied.

### 6.4 Module Description

The system's code has not been developed yet. In the future, the module description will be provided.

### 6.5 RESTful API Endpoints

- getAd()
  Purpose: requests an advertisement be sent to the calling mobile device.
  Input: Speed, Direction, Longitude/Latitude, Location, Unique ID
  Output: String representation of an Ad
  Implementation: This endpoint will send the information to the API and matches it with the best possible Advertisement to be displayed on the screen from the database.
- setAdFeedback()
  Purpose: Indicates to the server that the user has interacted with the advertisement, and what interaction took place.
  Input: None
  Output: String feedback
  Implementation: Clicking on the Ad will send a message handled by the server

## 7. Roles and Responsibilities

Our team will be responsible for implementing a fully functional API, as well as developing an application capable of consuming and utilizing the API, and thus displaying advertisements. If a developer of an application wishes to utilize the ASTRAEA advertising network, they must implement the appropriate code in their application to call our API, and designate an appropriate amount of space on their applications screen to display the advertisement returned by the Astraea advertising network.

### 7.1 Personal Roles

- Mobile Application:

    - Acquire location data. - DT
    - Add text area to display ad text on screen. -DT
    - “Package” Location data in preparation to be sent over the network. - DT
- DataBase (mySQL)
    - Create three tables: AD, Location, DEV
        - AD Table :
        - Location Table:
        - DEV Table:
- API Development
    - Create both end points. - Mike / Ryan
    - Set up server on Amazon. Mike/ Ryan
    - “Unpack” Location data package from App over network. Mike / Ryan
- Ad Server (Jacob)
    - The main method that connects all the components together: API, Database, and Mobile Application.
    - Updates all tables in the Database.
    - Determines which Advertisements will be displayed.

## 8. Terms and Definitions

**Vector** - a quantity having direction as well as magnitude, especially as determining the position of one point in space relative to another.

**Application program interface (API)** - a set of routines, protocols, and tools for building software applications.

**REST API** - An API that uses RESTful programming principles. It relies on a stateless client-server, cacheable communications protocol (typically HTTP).

## 9. Supporting References

There are no supporting references specific to this document.

## 10. Revision History

| Version | Version Date | Revisions |
|---|---|---|
| 1.0 | | Initial Release |