

GBC Navigation App

Software Requirements Specification

1

2019-11-06

Mike Denton
Lead Software Engineer

1.0 Introduction

1.1 Purpose

A mobile application intended for new students and/or visitors to navigate to the nearest destination around and within the Casa Loma Campus at George Brown College. Visitors or newcomers are directed to one of the parking lots available within the campus.

1.2 Scope

The name chosen for this product will be called “GBC Navigation App”.

GBC Navigation App will help users navigate to a nearest parking lot and/or guide users towards and within the building. It will only be explicitly within the College campus, which will not work if navigating else where.

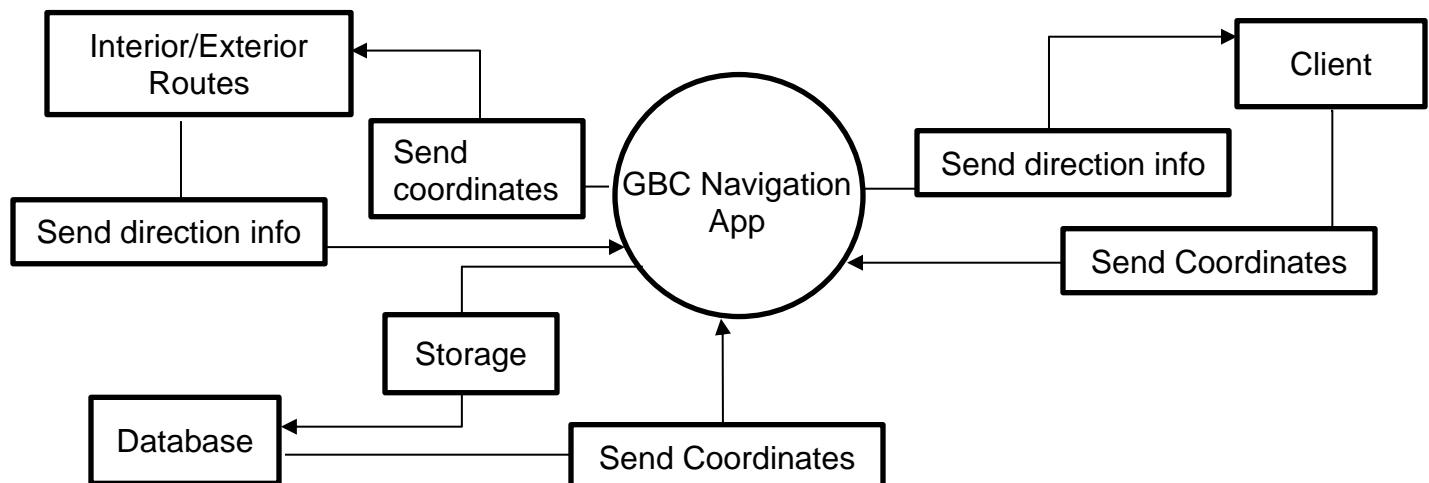
2.0 System Overview

2.1 Project Perspective

This project is self-contained system. Using existing technologies to support the developers.

2.2 System Context

The following diagram illustrates the Client navigating to their destination, while the backend of the application (Interior/Exterior Routes and Database) is requesting and querying stored static coordinates. Then a calculation is performed to determine the distance between the destination and Client. After the operation has been completed, the Client is then presented a route as to where him/her should navigate to.



2.3 General Constraints

- MongoDB only for data storage
- Designed only for GBC students
- Navigation intended only for the College campus

2.4 Assumptions and Dependencies

Technologies being implemented are Angular 8 for front end, while back end requires Node.js for handling server side, and MongoDB initial DataBase setup for Mongo Atlas Cloud hosting.

3.0 Functional Requirements

3.1

Feature #1 - Routes

Users are required to choose a desired route while using the application. A calculation is then performed to determine distance between the user and destination. A map pin icon is then displayed route is then displayed to guide users.

Feature #2 – Floor and Room Selection

When users get into the building, the application will render the internal buildings map and users are presented a dropdown selection control that lists the floors. When user selects the following, the rooms are corresponding to the floor selection.

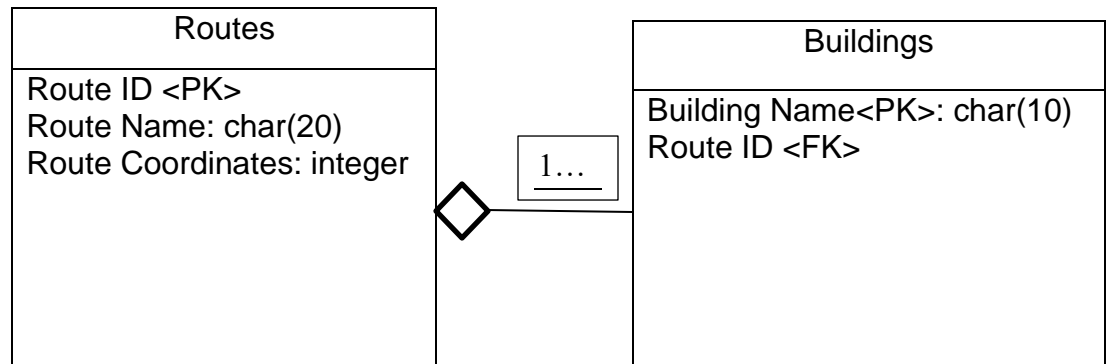
3.2 Use Cases

3.2.1 Finding a route

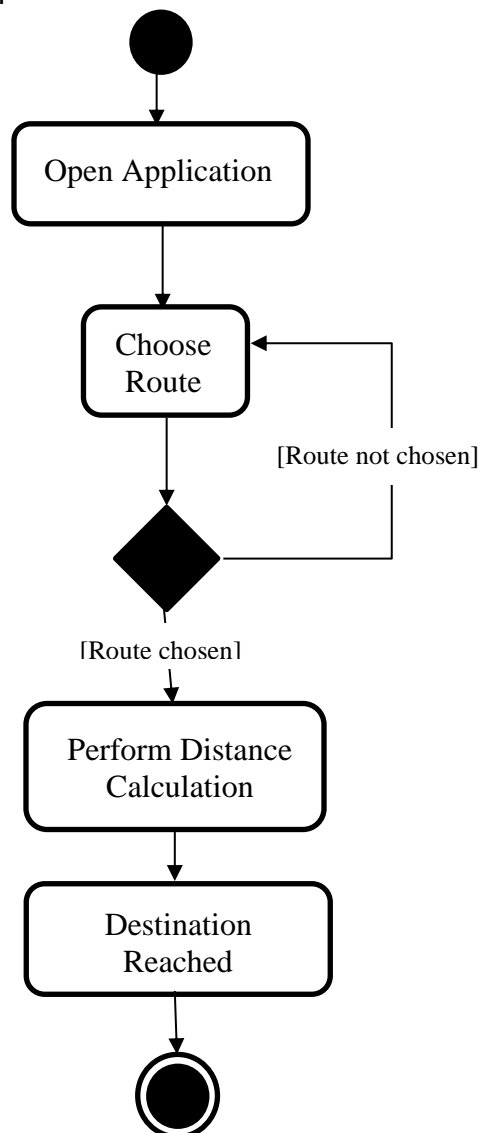
Description	User wants to be able to use the application as a guide around or within the buildings
Actors	<ul style="list-style-type: none">• Vistor• New student
Pre-conditions	The actor is a student or a visitor
Post-conditions	The route is presented to the actor
Flow of events	<ul style="list-style-type: none">• Actor chooses a route specified on map• The backend starts retrieving and data and calculating• Actor is then guided to their destination• Destination has been reached• Actor no longer requires the system

3.3 Data Modelling and Analysis

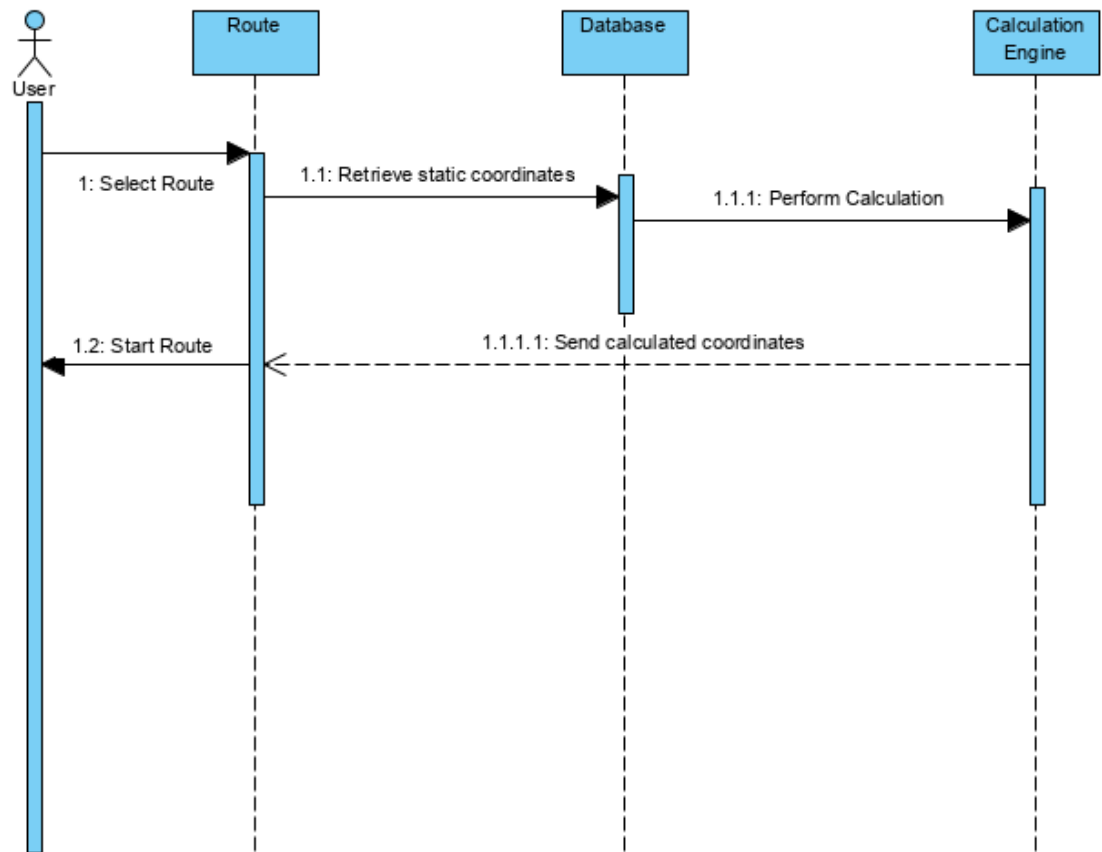
- Normalized Data Diagram



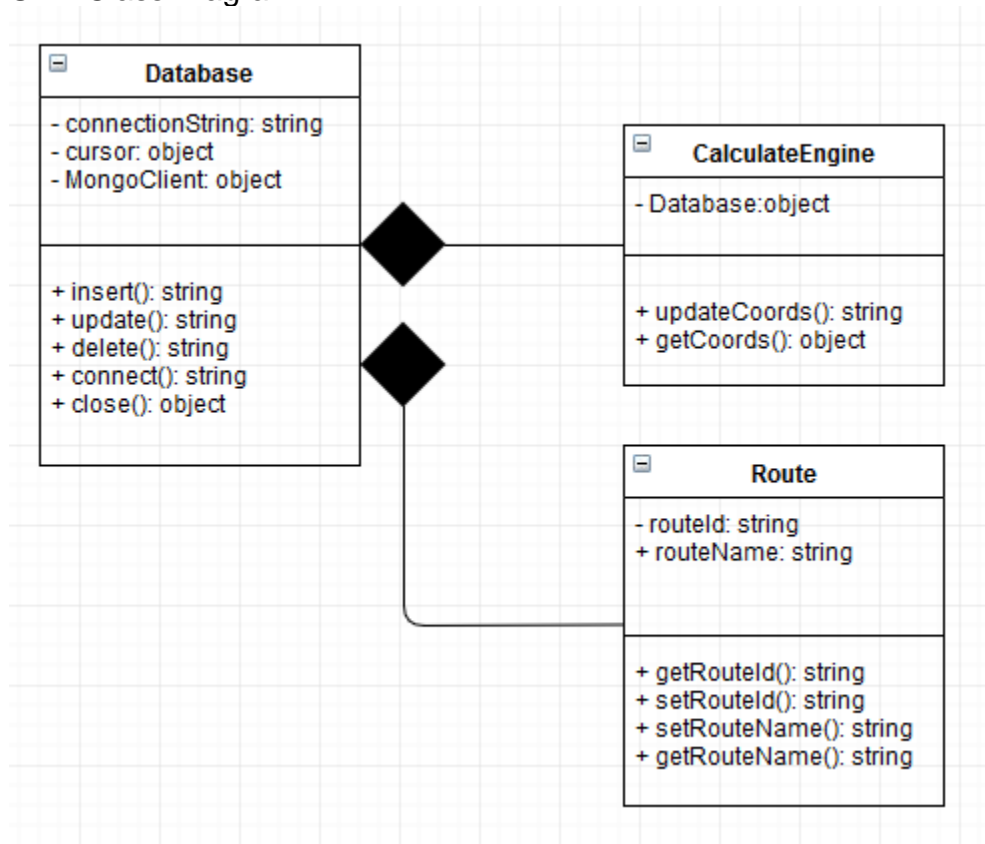
- Activity Diagram



- Sequence Diagram

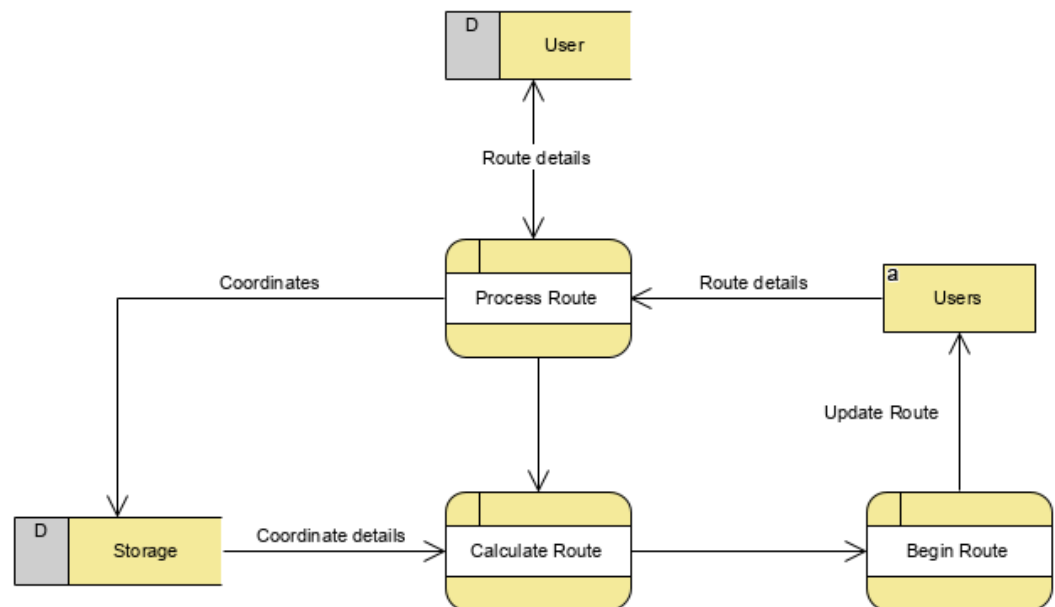


- UML Class Diagram



3.4 Process Modelling

- Data Flow Diagram



4.0 Non-Functional Requirements

90% of the routing will be processed through the calculation engine and the database synchronously within 1000 milliseconds. Log files are saved onto an alternative workspace to save storage.

The application can prevent SQL injections and redundant data. The applications database must also be able to be available overtime between planned maintenance.

The application database will be monitored for routing details exceeding their non-functional requirements at the 90th percentile for longer than 1 minute.

5.0 Logical Database Requirements

MongoDb (Mongo Atlas Cloud hosting) will handle data storage from incoming request from the Client and Node.js. MongoDb data will be stored in JSON (JavaScript Object Notation) format document.

Storage in MongoDb are referred to as Clusters. TTL indexes available to MongoDb will be used to handle data retention.

6.0 Other Requirements

7.0 Approval

The signatures below indicate their approval of the contents of this document.

Project Role	Name	Signature	Date
Lead Developer	Mike Denton	M.D	2019-11-11
Stakeholder	Jawad Butt	J.B	2019-11-11
Programmer	Talha Sultan	T.S	2019-11-09
Programmer	Jason Thai	J.T	2019-11-09