CSC 431

Lightweight offline map feature annotation tool

Software Requirements Specification (SRS)

Nerd Union

Alejandro Newsom	Tracing
Yuanzhi Liu	UI
Hal Milenkovic	UI/ Communication
Haoxuan Zhang	UI
Jacob Harris	Tracing
Jae Shin	Server
Sisi Zhong	Tracing
Jeremiah Hassett	Server/ Communication
Haosheng Liu	Server/Tracing
Francisco Delger	Tracing

Version History

Version	Date	Author(s)	Change Comments
1	02/26/17	Full team	

Table of Contents

Table of Figures

1. System Requirements

1.1 Functional Requirements

1.1.1 Server Connectivity

Title	Server Connectivity
Description	The app needs to be able to download and upload the annotated maps to and from the device (tablet)
Source Scenario	<Code for associated scenario in SCD>
Priority	0
Precondition(s)	We need to receive the server information so that we can program the app to pull the maps from that server. Also we need to ensure that the server is organized in a similar fashion as our app is. With the way that projects are broken down and selected.
Postconditions(s)	The app has a local version of the image map ready for annotation.
Use Case Diagram	<Link or number, if present>

1.1.2 Essential functionality

Title	Tracing, Annotating, Zoom
Description	Users should be able to trace polygons, lines, or place markers wherever it is desired on the map. They should also be able to add small annotations in the form of a popup and adjust the zoom level of the map.
Priority	5
Precondition(s)	<What needs to happen before>

Postconditions(s)	<What happens as a result>
Use Case Diagram	<Link or number, if present>

1.1. 3 Offline Capability

Title	Offline Capatbility
Description	The app needs to be fully functional offline minus the pulling and pushing the annotated maps to and from the server.
Source Scenario	<Code for associated scenario in SCD>
Priority	0
Precondition(s)	N/A
Postconditions(s)	All the applications features can be used offline.
Use Case Diagram	<Link or number, if present>

1.1. 4 User Interface

Title	User Interface
Description	A user interface so that only people with accounts can access the annotated maps. In this interface they can choose what type of annotation they wish to make.
Source Scenario	<Code for associated scenario in SCD>
Priority	0
Precondition(s)	Determine where account information is stored and who gets an account.
Postconditions(s)	The selected users will have access to the annotated maps.
Use Case Diagram	<Link or number, if present>

1.1. 5 Mobile

Title	Mobile
Description	The maps need to be on a mobile platform so that they can be annotated by the user walking through the street.
Source Scenario	<Code for associated scenario in SCD>
Priority	0
Precondition(s)	Needs to be coded in a language that is compatible with Android, Apple or both.
Postconditions(s)	Users will be able to walk around and complete their annotation tasks.
Use Case Diagram	<Link or number, if present>

1.2.1 Color Coding and Locking Shapes and Annotations

Title	Color Coding and Tracing
Description	When user finished annotating/tracing, the shape should automatically color itself. The tracing line should not lag.
Source Scenario	<Code for associated scenario in SCD>
Priority	3
Applicable FR(s)	Annotation

1.2 Non-Functional Requirements

1.2.1 Requirement Title

Title	Tracing, Annotations, Zoom
Description	To finish tracing, for maximum accuracy, the figure should close itself whenever you click on the first point again. Annotations should be editable and there should be various zoom levels.
Priority	5
Applicable FR(s)	<Which functional requirement(s) is this applicable to?>

1.2.3 GeoJSON compatible

Title	GeoJSON compatible
Description	Once the user finishes tracing, the trace should be saved as a GeoJson object.The app needs to be able to utilize the GeoJSON image format. This is the format the image objects will be saved on the server.
Source Scenario	<Code for associated scenario in SCD>
Priority	0
Precondition(s)	The app needs to be coded in a language that is GeoJSON compatible.
Postconditions(s)	The app will be able to access the images for Annotation.
Use Case Diagram	<Link or number, if present>

2. System Constraints

2.1 Tool Constraints

2.1.1 AWS

Title	Amazon Web Services hosting
Description	The application will be hosted in AWS.
Priority	<Priority from 0 (highest) – 5 (lowest)>

2.2 Language Constraints

2.2.1 Javascript

Title	Javascript
Description	Must be coded in javascript for browser based usage.
Priority	0

2.3 Platform Constraints

2.3.1 Browser Based App

Title	Browser Based Applicaiton
Description	Needs to be browser based so it can run on cross platform, requires an API.
Priority	1

2.4 Hardware Constraints

2.4.1 Tablet

Title	Tablet
Description	This apps ideal platform is a tablet that should have internet access for the application. From there, an internet connection is not needed until the user chooses to upload his work.
Priority	0

2.5 Network Constraints

2.5.1 Access to the Provided Server

Title	Access to the Provided Server
Description	The app needs to be able to access the provided server to pull and push the images directly from there
Priority	0

2.6 Deployment Constraints

2.6.1 Requirement Title

Title	Browser Based Application
Description	The system will be a Browser Based Application so that it is easily used and accessed.
Priority	0

2.7 Transition & Support Constraints

2.7.1 Requirement Title

Title	Support
Description	Leaflet - javascript library, GeoJSON
Priority	2

2.8 Budget & Schedule Constraints

2.8.1 Requirement Title

Title	AWS cost
Description	There is a cost for hosting the application on AWS which means support for this hosting will only be available for the semester.
Priority	5

2.9 Miscellaneous Constraints

2.9.1 Requirement Title

Title	unable to find
Description	NA
Priority	50

3. Requirements Modeling