Project Plan Presentation
Live Platform CAD Ingestion

The Capstone Experience

Team Herman Miller
Connor Lang
Xingzhi Mei
Meigan Starr
Greg Szczerba

Department of Computer Science and Engineering
Michigan State University
Fall 2021
Functional Specifications

- Improve Live Platform service
- Implement ability to ingest CAD files into floorplan mapping system instead of images
- Use additional data from CAD files to do sensor validation
- Create a standalone web application to allow administrators to view and modify floorplans
Design Specifications

• Navigator page
  ▪ Search for floorplans
  ▪ Create, edit, and delete floorplans
  ▪ Launch floorplans in the viewer

• Viewer page
  ▪ Display selected floorplan
  ▪ Selectively display different metadata
  ▪ Visually modify sensor information
Screen Mockup: Navigator Page
Screen Mockup: Edit/Create Layer
Screen Mockup: Viewer Page
Screen Mockup:
Viewer Page with Layer selected
Technical Specifications

• Client-Side Web Application
• AWS Serverless Back End
  ▪ CAD to GeoJSON Converter
  ▪ Floorplan DynamoDB Database
  ▪ RESTful API for Database Access
  ▪ S3 Storage for Source CAD Files
• Mapbox Static Floorplan Storage
System Architecture

[Diagram showing system architecture with various components and connections such as React, Redux, S3 Static Site, Lambda API Handler, Source CAD File Storage, AWS Lambda, Conversion Queue, and Mapbox.]

Client-Side Web Application

React

Redux

API Gateway

DB Lookup

Update Floorplan

Download CAD File

Administrator

Lambda

Conversion Queue

Static Floorplan Storage

AWS

Source CAD File Storage

CAD to GeoJSON Conversion

Mapbox
System Components

• Hardware Platforms
  ▪ None. Web application with serverless back end.

• Software Platforms / Technologies
  ▪ AWS API Gateway, Lambda, DynamoDB, SQS, S3
  ▪ Docker
  ▪ Alpine Linux
  ▪ Mapbox
  ▪ React / Redux / JSX
  ▪ The Geospatial Data Abstraction Library (GDAL)
Risks

• Implementation uses shell execution.
  ▪ Current design uses shell execution in the code requires it be on a server with the command line tools installed. Might not be able to be a serverless AWS lambda function.
  ▪ To mitigate, we will have to use different method for that portion of the implementation or set up a deployment that allows for command line execution.

• Reading layers in DWG/DXF files
  ▪ CAD files with furniture data may be too large to convert into GeoJSON with current solution. Team will likely need to extract layers individually instead of all at once, but these layer names are not standardized.
  ▪ To mitigate, we will have to talk with CAD designers about layer name standardization and determine a way to determine which layers to use.

• DWG to DXF Conversion
  ▪ In order to convert the DWG CAD file to GeoJSON format, the file must first be converted to DXF first. The plan is to include this conversion in the DWG to GeoJSON pipeline, however this requires an open-source DWG to DXF converter. A cost-free conversion option must be found in order to complete the conversion pipeline.
  ▪ The conversion can also be done manually by AutoCAD, which could be performed by the client or Herman Miller before sending the file into the conversion pipeline.
Questions?