

MICHIGAN STATE
U N I V E R S I T Y

Alpha Presentation

Catastrophic Claims Unit Mobilization

The Capstone Experience

Team Auto-Owners

Nicole Lawrence

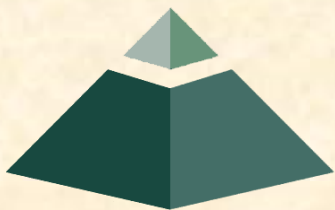
Nick Reuter

Jason Steele

Matt Wiechec

Department of Computer Science and Engineering
Michigan State University

Spring 2016

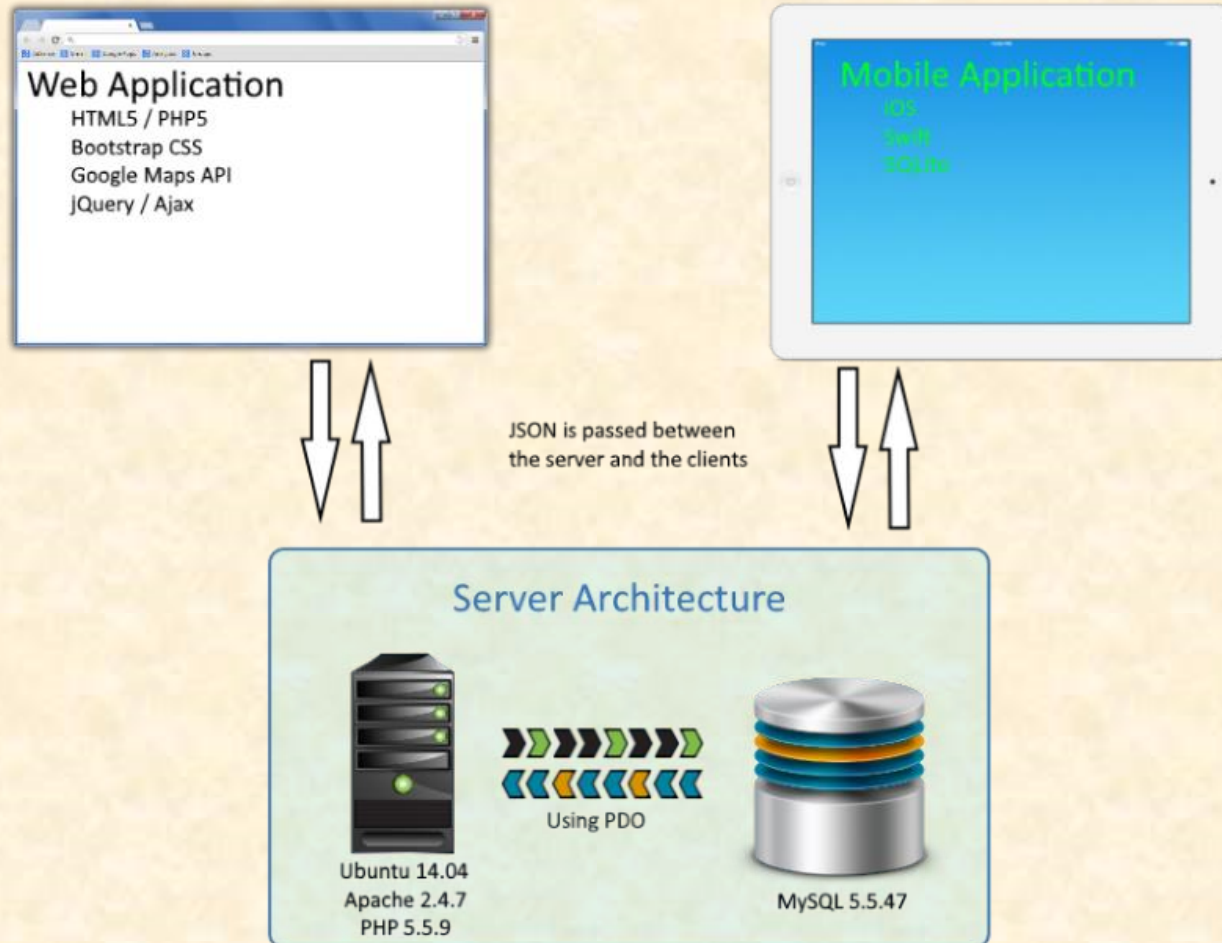


From Students...
...to Professionals

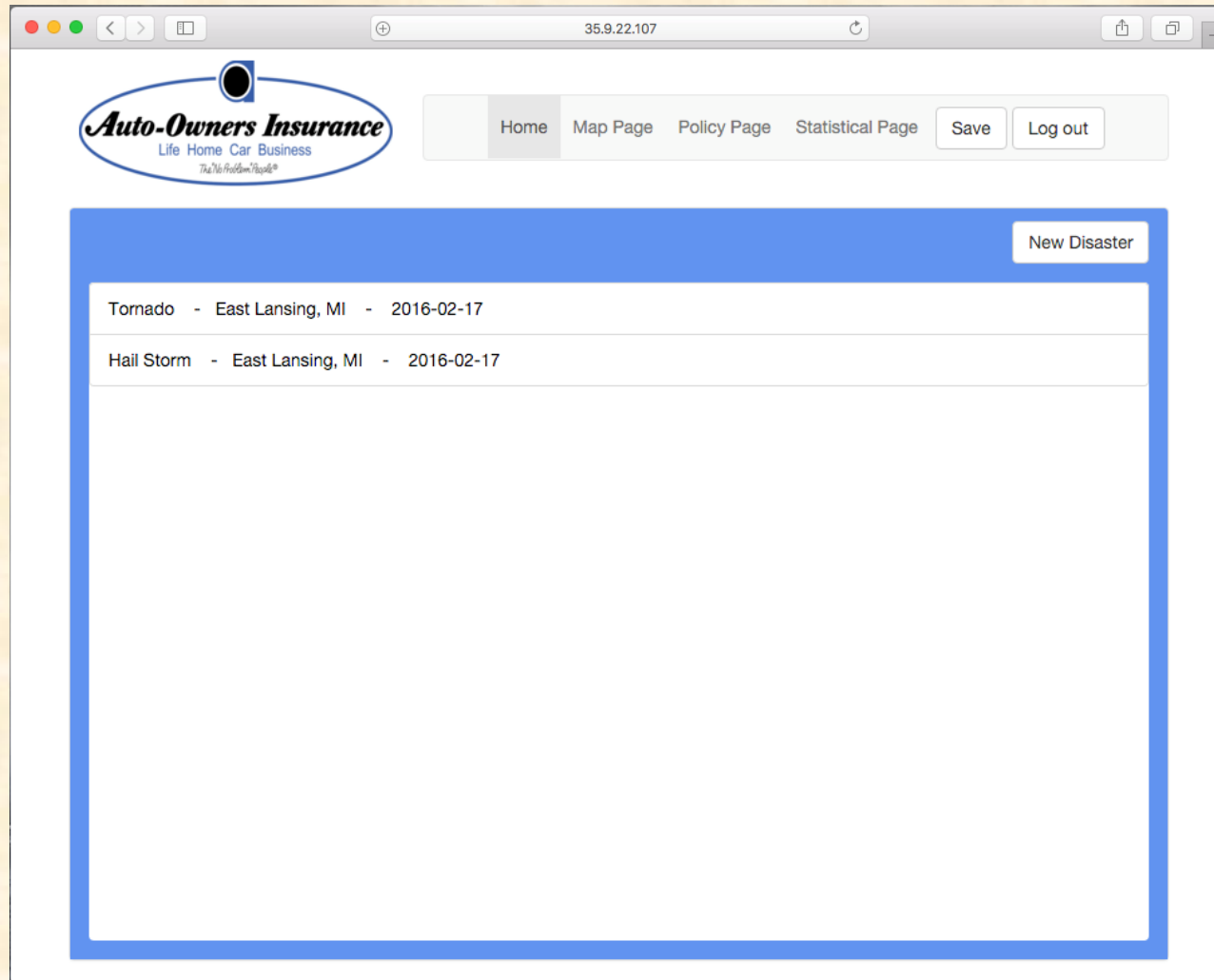
Project Overview

- Functionalities
 - Manage insurance claims after weather incident
 - Provide a way for policy holders to submit claims
 - Check on weather event and see claim progress
- Features
 - Map tool to prepare mobile tablets
 - Offline access to relevant policies and claims forms
 - Statistics view for claims status and trends

System Architecture



Index Page – Disaster List

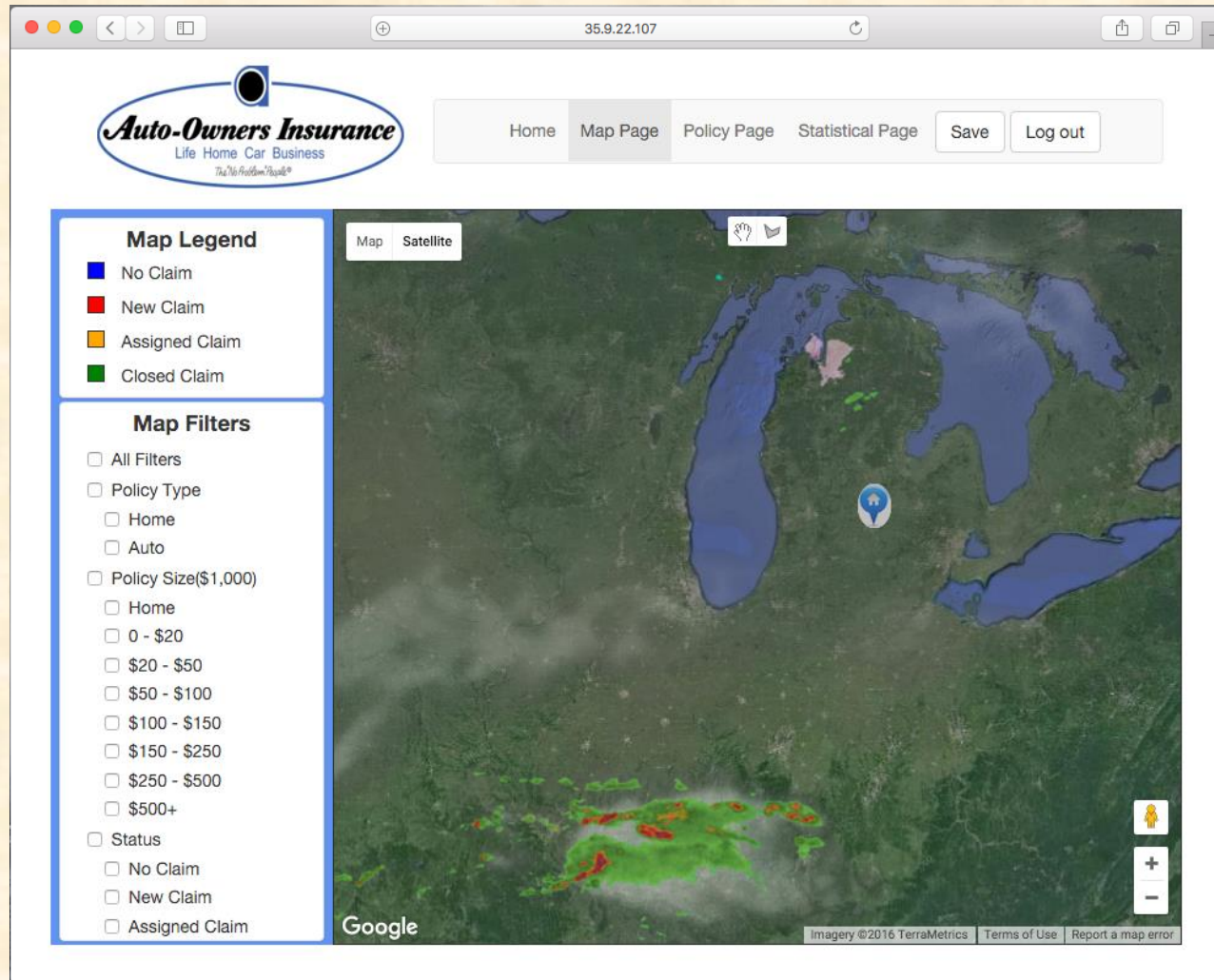


Map Page – All Policies

The screenshot displays the 'Map Page' of the Auto-Owners Insurance website. The browser window shows the URL '35.9.22.107'. The website header includes the Auto-Owners Insurance logo with the tagline 'Life Home Car Business' and 'The 7th Freedom People®'. Navigation links for 'Home', 'Map Page', 'Policy Page', and 'Statistical Page' are present, along with 'Save' and 'Log out' buttons. The main content area features a satellite map with several blue location pins. On the left, a 'Map Legend' indicates four claim statuses: 'No Claim' (blue), 'New Claim' (red), 'Assigned Claim' (orange), and 'Closed Claim' (green). Below the legend, 'Map Filters' allow users to select 'All Filters', 'Policy Type' (Home, Auto), 'Policy Size (\$1,000)' (ranging from 0 to \$500+), and 'Status' (No Claim, New Claim, Assigned Claim). The map interface includes 'Map' and 'Satellite' toggle buttons, a hand icon for panning, and zoom controls. At the bottom, Google imagery copyright information is visible.



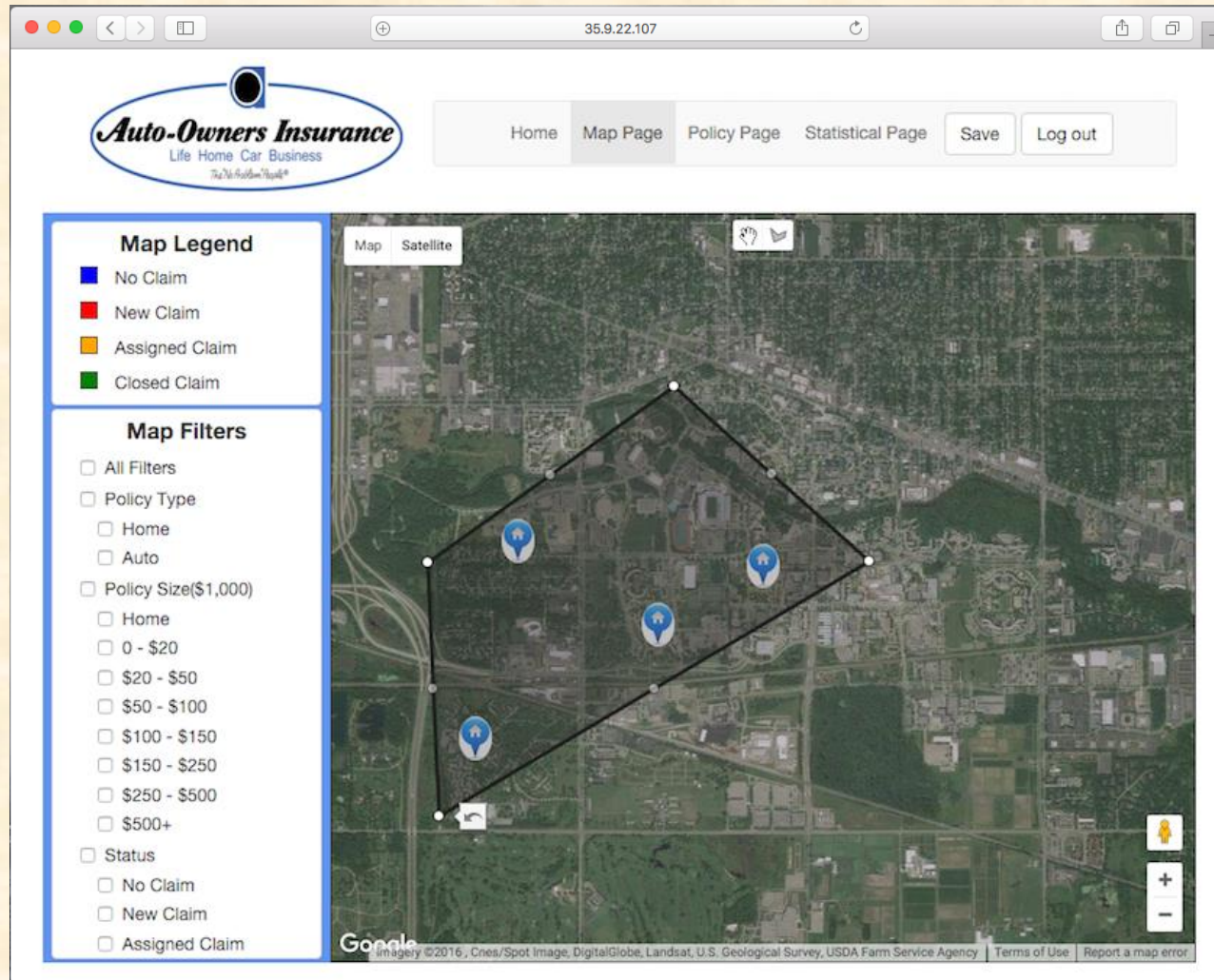
Map Page – Weather Overlay



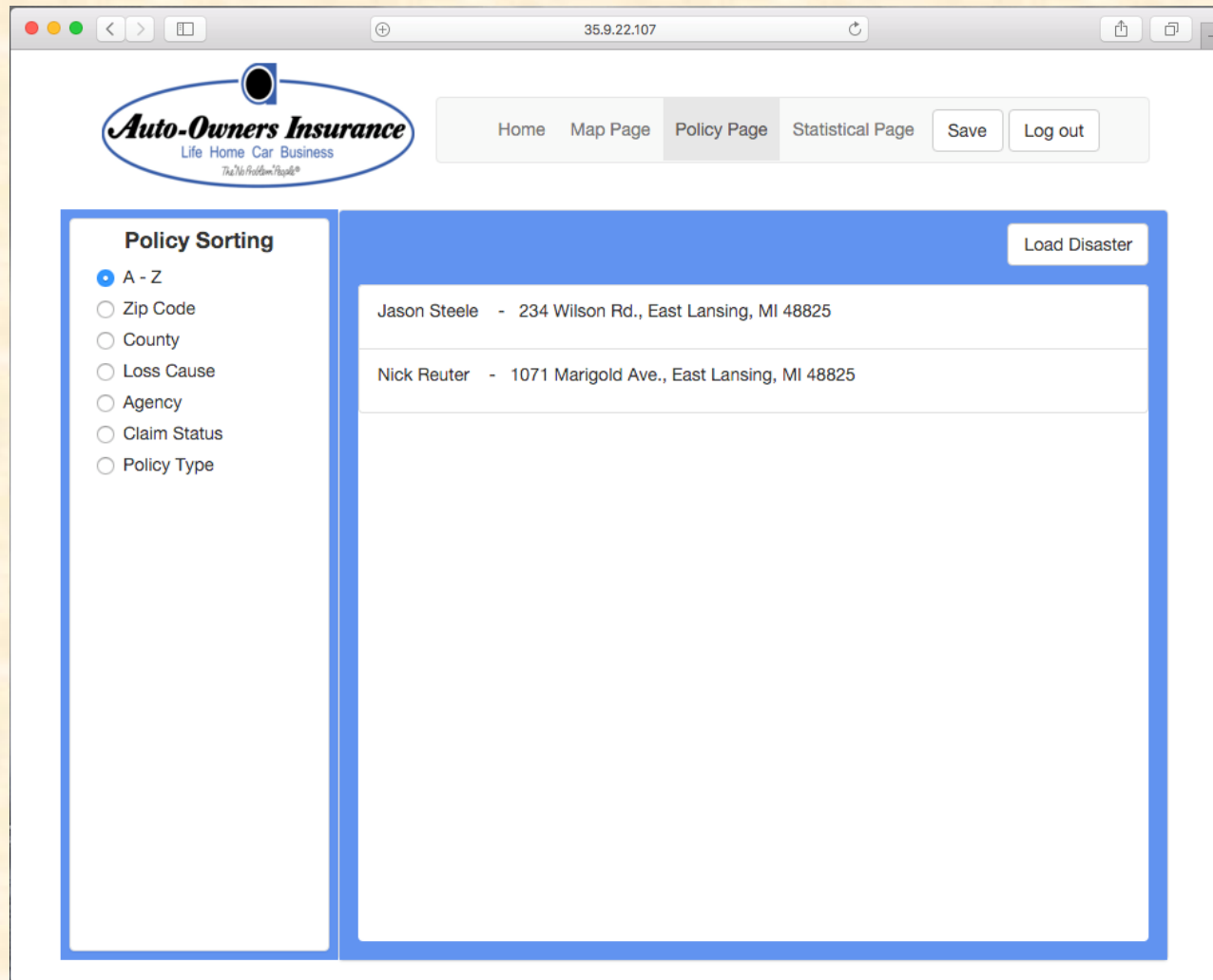
Map Page – Drawing a Polygon

The screenshot displays the 'Auto-Owners Insurance' web application interface. At the top, the company logo is on the left, and navigation links for 'Home', 'Map Page', 'Policy Page', and 'Statistical Page' are in the center. 'Save' and 'Log out' buttons are on the right. The main content area is divided into a left sidebar and a right map panel. The sidebar contains a 'Map Legend' with four categories: 'No Claim' (blue square), 'New Claim' (red square), 'Assigned Claim' (orange square), and 'Closed Claim' (green square). Below this is the 'Map Filters' section, which includes checkboxes for 'All Filters', 'Policy Type' (Home, Auto), 'Policy Size(\$1,000)' (ranging from 0 to \$500+), and 'Status' (No Claim, New Claim, Assigned Claim). The right panel shows a satellite map with a black polygon drawn over a residential area. Inside the polygon, there are four blue location pins. The map includes standard Google Maps controls like zoom in/out buttons and a street view pegman. The bottom of the map shows copyright information for 2016 and links to 'Terms of Use' and 'Report a map error'.

Map Page – Drawing a Polygon



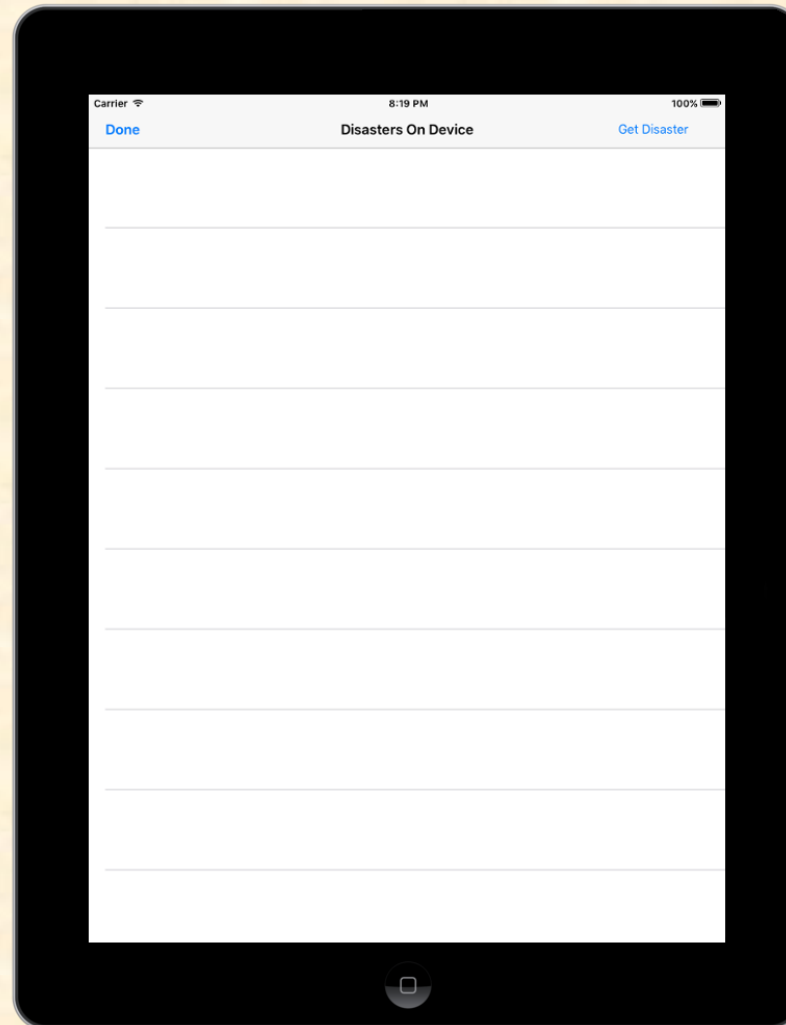
Policy Page – Policy List



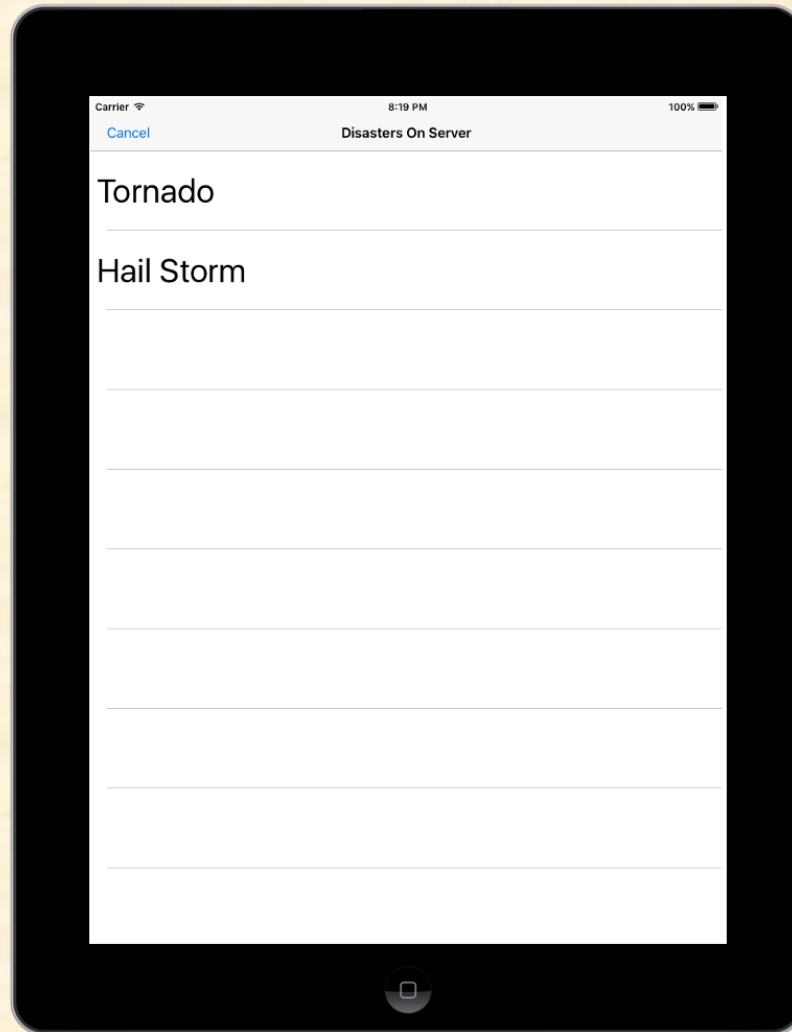
Statistics Page



Disaster List Page – Initially Empty



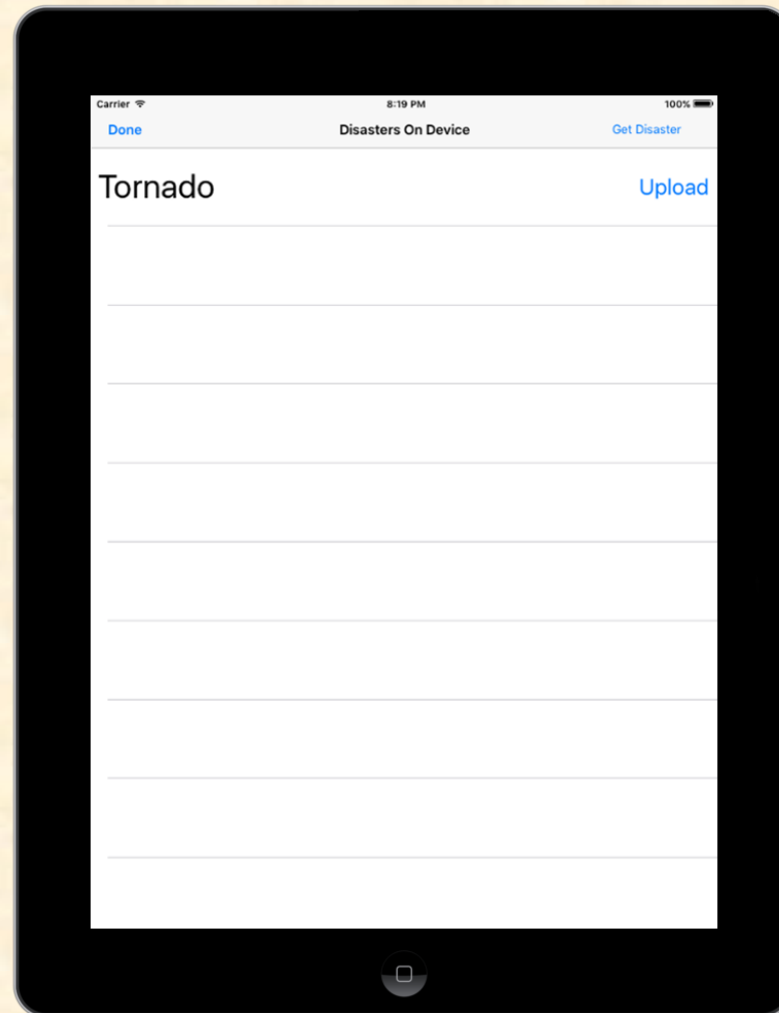
Disasters on Server



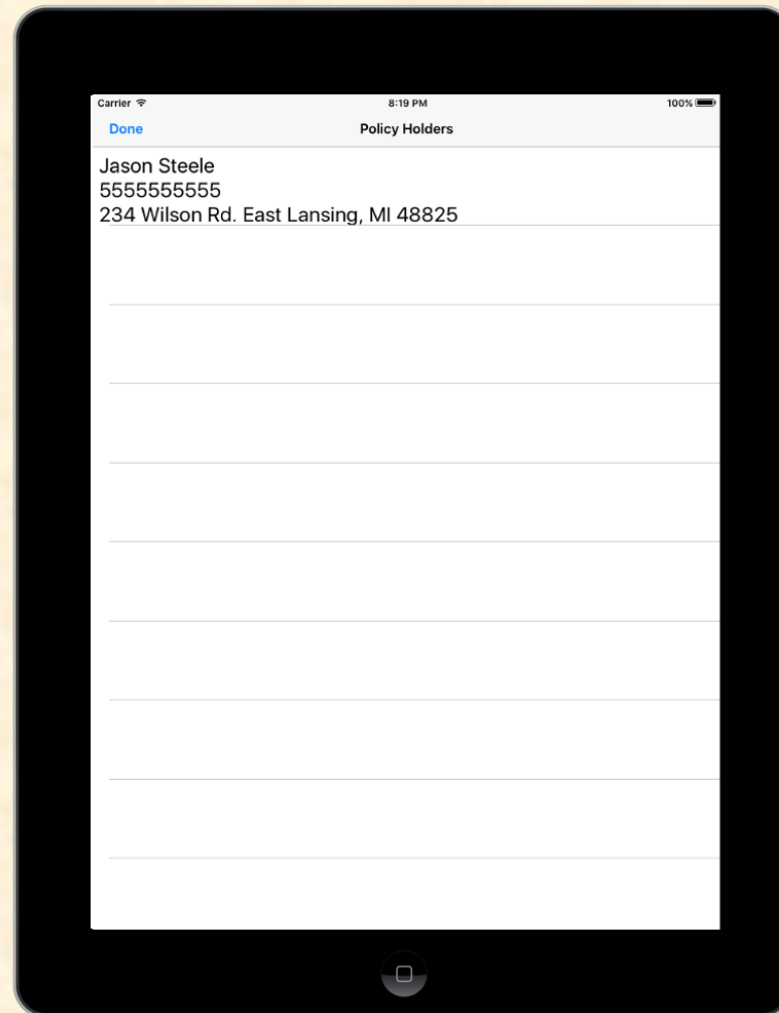
The image shows a tablet screen with a form titled "Disasters On Server". The form has a "Cancel" button in the top left corner. Below the title, there are two input fields. The first input field contains the text "Tornado". The second input field contains the text "Hail Storm". There are several empty input fields below the second one. The tablet has a black bezel and a home button at the bottom.

Disasters On Server
Tornado
Hail Storm

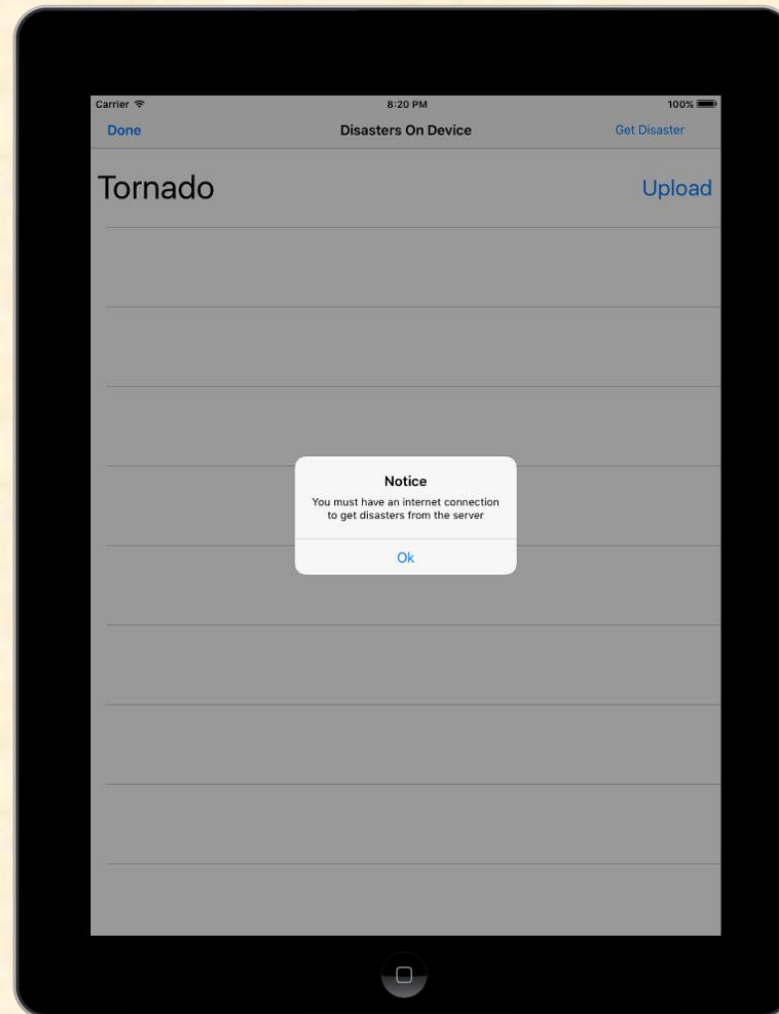
Disaster List – Disaster Downloaded



Policy List of Disaster



Connection Attempt w/o Internet



What's left to do? (Web)

- Create / edit disaster
 - Add saving functionality
 - Change database entries
- Dynamically create policy page
 - Grab appropriate information based on properties
 - Refine display styling
- Get sorting / filtering working
 - Functional filter attributes on map page
 - Functional sort attributes on policy page
- Have functional statistics page for an existing disaster
 - Generate graph charts based on live data
 - Create pie chart from claim status
- Find most central location for mobile unit

What's left to do? (iOS)

- Create / upload claims
 - Create claim and store internally on mobile device
 - Send claims table as JSON to server
- Search functionality
 - Formatting a search screen
 - Develop search functionality by policy holder name or phone number
- Policy information
 - Allow user to view policy information
 - Download and store all policy information on device
- Layouts
 - Make layouts for policy information and preprinted loss form
 - Update layouts to enhance user experience
- Concurrency
 - Display loading overlay when communicating with network
 - Prevent user from interacting with out of sync information