Main Tasks

Task	Technology	Description	Details
Parsers	• Python • Clowder Geostreams	• Create python script that reads the data for idph csv files and posts it to the geostreaming API ◦ The geostream data was originally used for geodashboard, but later the geostreaming data was also used by the web application.	• Parser for both preprocessed files (by Bill) and later for unprocessed files.

Geospatial Web Application	• Flask
	• Openlayers
	• Plot.ly
	• Bootstrap
	• Javascript
		• Show MIR data in geospatial app
		  ◦ select features in map by region or trap
		  ▪ popup shows overview
		  • 'show data' opens graph of data
		  ◦ run model through interface
		  ▪ run fortran model
		  ▪ run python model (converted and generalized from fortran model)
			• Get region geojson for regions, edit for use in application

arcus.sws.illinois.edu	• Ubuntu Linux • nginx	• Install application on State Water Survey VM	
Run model (Fortran)	• Fortran • Called by Python	• Run model by copying input files from idea.sws.uiuc.edu (where fortran model is running)	
Run model (Python)	• Python	• Emulate Fortran model ◦ Parses weather data in a general way so it can run without inputs of epi weeks, uses only one weather data file	
Get weather data endpoint	• Python	• get weather data from http://www.rcc-acis.org/ api • parse into files	
Get forecast data	• Python	• get weather data from http://sats.nws.noaa.gov api • parse into files	
Get cdc week and year by date	• Node.js • Called by python	• Function to get epi week and year by calendar date ◦ https://www.npmjs.com/package/epidemiological-week	
Get daylight hours	• Python	• Get latitude center of region and get daylight hours for that location though https://api.sunrise-sunset.org	
Run biggerstaff model	• R	• Run R library to get MIR	• Not needed, the actual calculation is simple