

## **sdmay19-18: Real time Route Optimization**

### **Report 3**

**February 22 - March 1**

#### **Team Members**

**Junjie Wen** — *Backend Developer; Data Analytics Lead*

**Zhanghao Wen** — *Web Developer*

**Yuhang Xie** — *Web Developer; UI Lead*

**Xinhe Yang** — *Web Developer*

**Tianhao Zhao** — *Communication Leader; Technical Writing*

---

#### **Summary of Progress this Report**

- After a face-to-face discussion with the client, we have a better understanding of the project. Client provides us with real history data.
  - We performed a detailed analysis of the data and tried to replace our simulation data.
  - We import most of the data into our database, and the backend provides an interface to access this data.
  - Finally we displayed the real data on Google Maps.
- 

#### **Past Week Accomplishments**

We got the snow plow history data from other cities in the past months. We are downloading data from the website provided to me by the client. The website url is <https://hed-rcm.canect.io>. The follow image is how the data look like.

Timestamp	Value	Location (alt, lat, lng)
2019-02-20T15:01:56.372416Z		{"alt":245.978,"lat":41.98936079,"lng":-88.15705461}
2019-02-20T15:01:56.372331Z	0.141	
2019-02-20T15:01:51.369566Z		{"alt":246.455,"lat":41.9893597,"lng":-88.15706033}
2019-02-20T15:01:51.369481Z	0.056	
2019-02-20T15:01:46.374024Z		{"alt":247.292,"lat":41.98935498,"lng":-88.15707447}
2019-02-20T15:01:46.373940Z	0.026	
2019-02-20T15:01:41.372268Z		{"alt":247.809,"lat":41.98933457,"lng":-88.15710829}
2019-02-20T15:01:41.372185Z	1.549	
2019-02-20T15:01:36.369969Z		{"alt":247.502,"lat":41.98927274,"lng":-88.15729049}
2019-02-20T15:01:36.369886Z	2.807	
2019-02-20T15:01:31.368420Z		{"alt":251.001,"lat":41.98935909,"lng":-88.15750223}
2019-02-20T15:01:31.368336Z	5.039	
2019-02-20T15:01:26.364189Z		{"alt":256.312,"lat":41.98942991,"lng":-88.15787154}
2019-02-20T15:01:26.364101Z	4.853	
2019-02-20T15:01:25.624870Z		
2019-02-20T15:01:22.623914Z		
2019-02-20T15:01:21.357759Z		{"alt":254.462,"lat":41.98957753,"lng":-88.15818658}
2019-02-20T15:01:21.357675Z	4.752	
2019-02-20T15:01:16.355347Z		{"alt":256.466,"lat":41.98978755,"lng":-88.15820139}
2019-02-20T15:01:16.355265Z	5.597	
2019-02-20T15:01:11.368825Z		{"alt":257.285,"lat":41.99004897,"lng":-88.15815919}
2019-02-20T15:01:11.368738Z	6.903	

Because we don't have permission to get this data through the website interface provided by the client, we import all the historical data into our own database and provide interface access for ourselves. We found that the data was not organized, and each data only provided data for the current sensor, and the others were empty. For example, in the data with GPS, other data such as pressure is empty.

```

▼ [0 ... 99]
  ▶ 0: {_id: "5c802320b39e173aefc8fff6", timestamp: "2019-02-01T05:59:57.847886Z", Hanover Park IL TGS - ESN (): "", Hano...
  ▶ 1: {_id: "5c802320b39e173aefc8fff7", timestamp: "2019-02-01T05:59:56.847091Z", Hanover Park IL TGS - ESN (): "", Hano...
  ▼ 2:
    Hanover Park IL TGS - Conveyor_RPM (): ""
    Hanover Park IL TGS - ESN (): ""
    Hanover Park IL TGS - GPS_gs (meter_per_sec): ""
    Hanover Park IL TGS - GPS_gs - transformed (meter_per_sec): ""
    Hanover Park IL TGS - Liq_GPM (gal/min): ""
    Hanover Park IL TGS - Liq_GPM - transformed (gal/min): ""
    Hanover Park IL TGS - Location (): "{"alt":243.262,"lat":42.01441379,"lng":-88.12891733}"
    Hanover Park IL TGS - Outside_Temp (°F): ""
    Hanover Park IL TGS - Pressure (psi): ""
    Hanover Park IL TGS - Reflector_status (): ""
    Hanover Park IL TGS - Reflector_url (): ""
    timestamp: "2019-02-01T05:59:55.994176Z"
    _id: "5c802320b39e173aefc8fff8"
    ▶ __proto__: Object
  ▶ 3: {_id: "5c802320b39e173aefc8fff9", timestamp: "2019-02-01T05:59:55.994092Z", Hanover Park IL TGS - ESN (): "", Hano...
  ▶ 4: {_id: "5c802320b39e173aefc8fffa", timestamp: "2019-02-01T05:59:52.847754Z", Hanover Park IL TGS - ESN (): "", Hano...
  ▶ 5: {_id: "5c802320b39e173aefc8fffb", timestamp: "2019-02-01T05:59:51.846981Z", Hanover Park IL TGS - ESN (): "", Hano...

```

We will sort these data in the front-end js code, then we get the following image





---

## Pending Issues

- We got the real data, but we still unable to get live data through the website provided by the client
- Find an algorithm to solve the situation of trucks in different weather conditions
- Waiting for the interface to provide time range for real data

---

## Plans for Upcoming Reporting Period

- Try to use real data to simulator live data for out project
- Added functionality, by selecting the time interface to get different time periods is the data and displayed on the google map.
- Solve algorithms that provide advice to dispatchers in different weather or temperature conditions. For example, when the temperature reaches a certain value, the dispatcher should be reminded to use the corresponding tool.

---

### Individual Contributions

<b>Team Member</b>	<b>Contribution</b>	<b>Weekly Hours</b>	<b>Total Hours</b>
Junjie Wen	Build apis in backend server to show data	11	30
Zhanghao Wen	Analyzed data and data type from storage server and how we can get and use real data	6	21
Yuhang Xie	Improve the record page, get real data from the back end and organize the data. Display real data in a visual form on a google map.	12	34
Xinhe Yang	Design a historical record page and show history data on the google map	11	31
Tianhao Zhao	Learn frontend coding styles, discuss with client and adviser about project implementation.	8	28