



OTP SUM: OTP Integration of Transit with Shared-Use Mobility Real-Time and Data Enhancements

Mobility on Demand Sandbox Program Quarterly Report Q1 2017

May 3, 2017

Challenges Addressed by Project

- OpenTripPlanner (OTP) does not currently incorporate shared-use modes.
- The trip planning process involves specifying the origin and destination of the trip, starting with a geocoding process, but the geocoding solutions available are often inadequate, leading to poor results and user frustration.
- Accessible trips are difficult to plan accurately due to the lack of data available on the accessibility of pedestrian infrastructure and the absence of these features in a trip planner.

Anticipated Outcomes, Benefits, Impacts

- Extend the OpenTripPlanner code base to support the integration of transit trip planning with shared-use mobility modes, such as bike share and transportation network companies (TNCs), as well as updated real-time transit information.
- Implement a fully functional and comprehensive open geocoder built off the existing Mapzen Pelias geocoder. A non-proprietary and non-restrictive option for address locating would substantially lower the barrier to entry for many transit systems to offer trip planning and can achieve significant cost savings for transit agencies, government agencies, and the public.
- TriMet, in collaboration with the OpenStreetMap community, established best practices for representing accessibility information and will build out this accessibility information in the OSM network and provide a model for replicating this work in other regions.

The project scope and budget are broken down into the six tasks listed below. Each task section includes the deliverables and progress for this quarter.

| | |
|--|-----------|
| Challenges Addressed by Project | 1 |
| Anticipated Outcomes, Benefits, Impacts | 1 |
| Project Financial Status | 3 |
| Task 1: Project Management | 5 |
| Quarterly Deliverables. | 5 |
| Quarterly Progress. | 5 |
| Task 2: Evaluations and Reports | 6 |
| Quarterly Deliverables. | 6 |
| Quarterly Progress. | 6 |
| Task 3: Application Development Status | 8 |
| Quarterly Deliverables. | 8 |
| Quarterly Progress. | 8 |
| Task 4: Geocoder Development | 9 |
| Quarterly Deliverables. | 9 |
| Quarterly Progress. | 9 |
| Task 5: Data Improvements | 10 |
| Quarterly Deliverables. | 10 |
| Quarterly Progress. | 10 |
| Task 6: Integrated Payment Plan | 11 |
| Quarterly Deliverables. | 11 |
| Quarterly Progress. | 11 |
| Conferences, Meetings, Workshops | 12 |
| Upcoming Highlights | 12 |
| Lessons Learned | 13 |

Project Financial Status

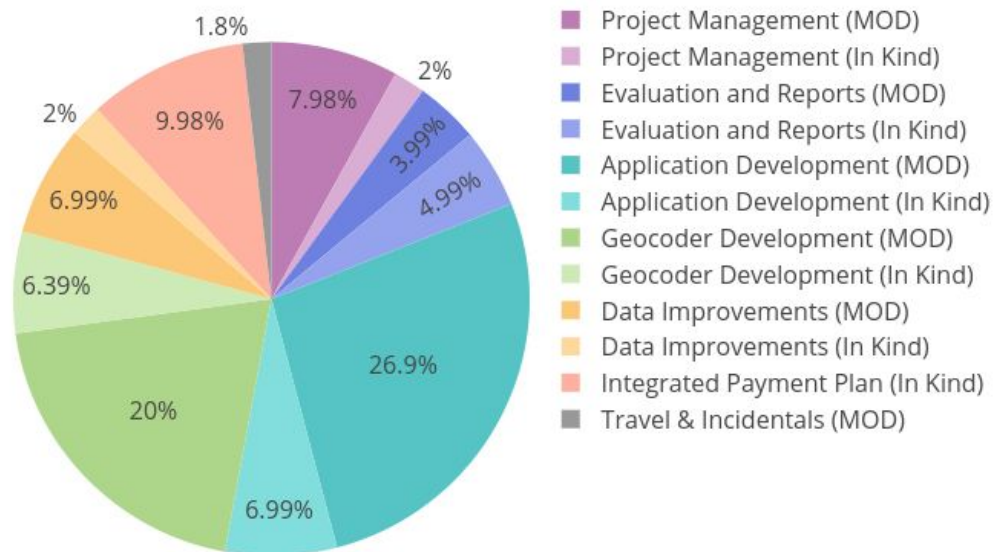
TriMet's funding allocation from the FTA of \$678,000 is matched with 32% of in-kind contributions, totaling over \$1 million.

Of the \$678,000 that TriMet received, \$58,620 (8.6% of allocated grant funds) has been spent thus far.

The expenditures through Q1 2017 are as follows:

- \$1,122 (2% of allocated in-kind funds) of in-kind contribution spent toward Evaluation and Reports;
- \$54,000 (20% of allocated grant funds) spent toward Application Development;
- \$4,620 (26% of allocated grant funds) spent toward Travel & Incidentals.

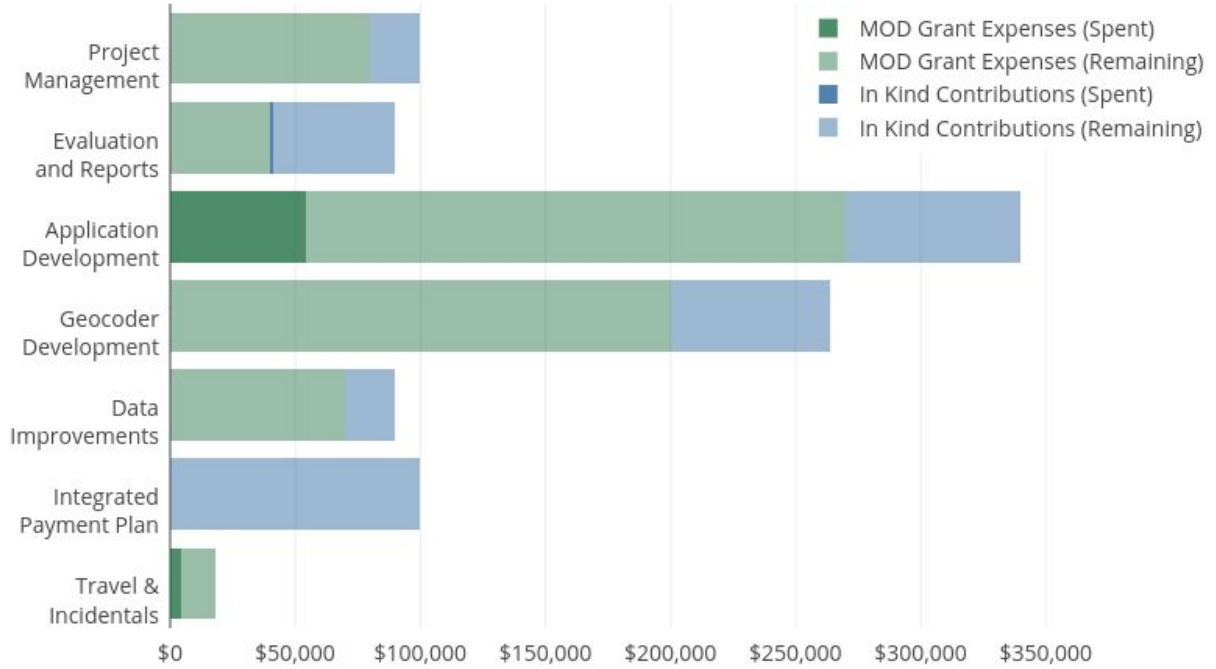
MOD Grant Budget



The above pie chart illustrates the amount and percentage of the budget allocated to each of the main tasks, divided into MOD grant expenses and in-kind contributions.

Project Financial Status (continued)

MOD Grant Spent and Remaining Funds



The above bar chart shows the current amount spent for each of the tasks.

Task 1: Project Management

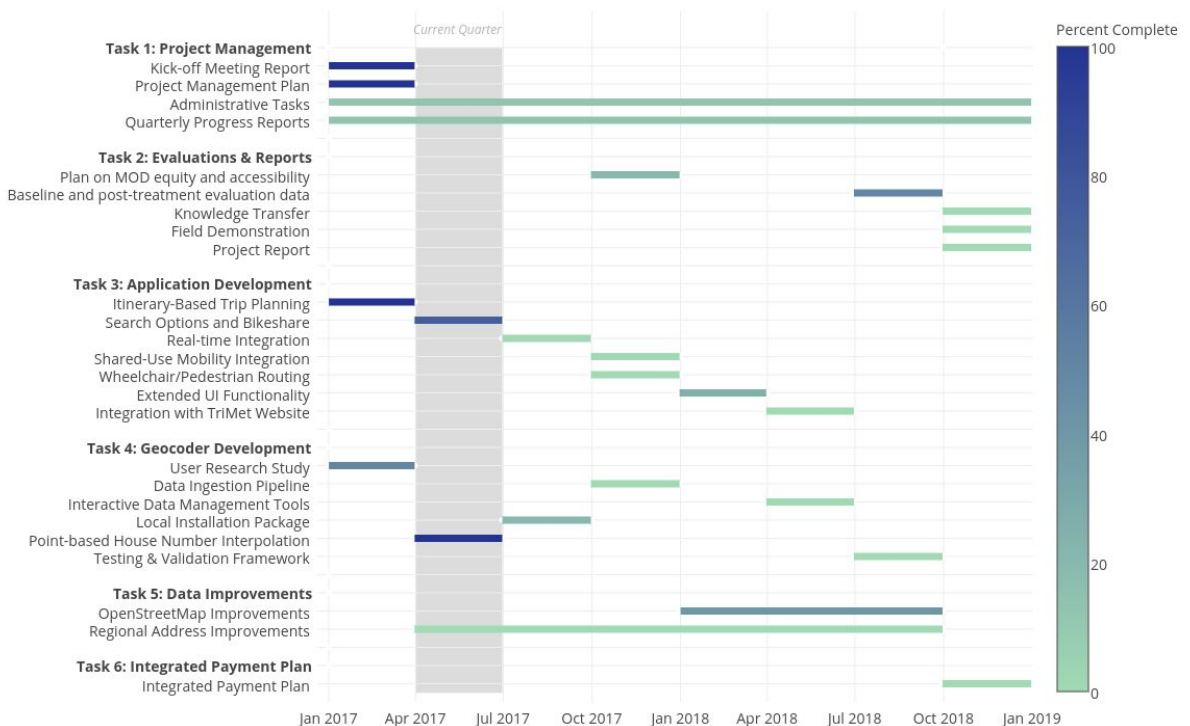
TriMet's OTP Integration of Transit with Shared-Use Mobility Real-Time and Data Enhancements have been underway since January. All milestones and deliverables have been met and we are on schedule.

Quarterly Deliverables

- Administrative Tasks
- Final PMP (**Appendix A - Project Management Plan**)
- Kickoff Meeting Report (**Appendix B - Kickoff Workshop Report**)

Quarterly Progress

- Organized a highly attended and successful kickoff workshop January 18-19, 2017 at TriMet.
- Strengthened partnerships between the public and private partners.
- Ensured continued communications through weekly scheduled meetings (Slack and webinars).
- Managing the project using Trello and a dedicated and open Google Drive.
- Using InVision for application interface development and review.
- Developed an online project dashboard, available to the public at <https://trimet.org/mod>, to ensure transparency.



Task 2: Evaluations and Reports

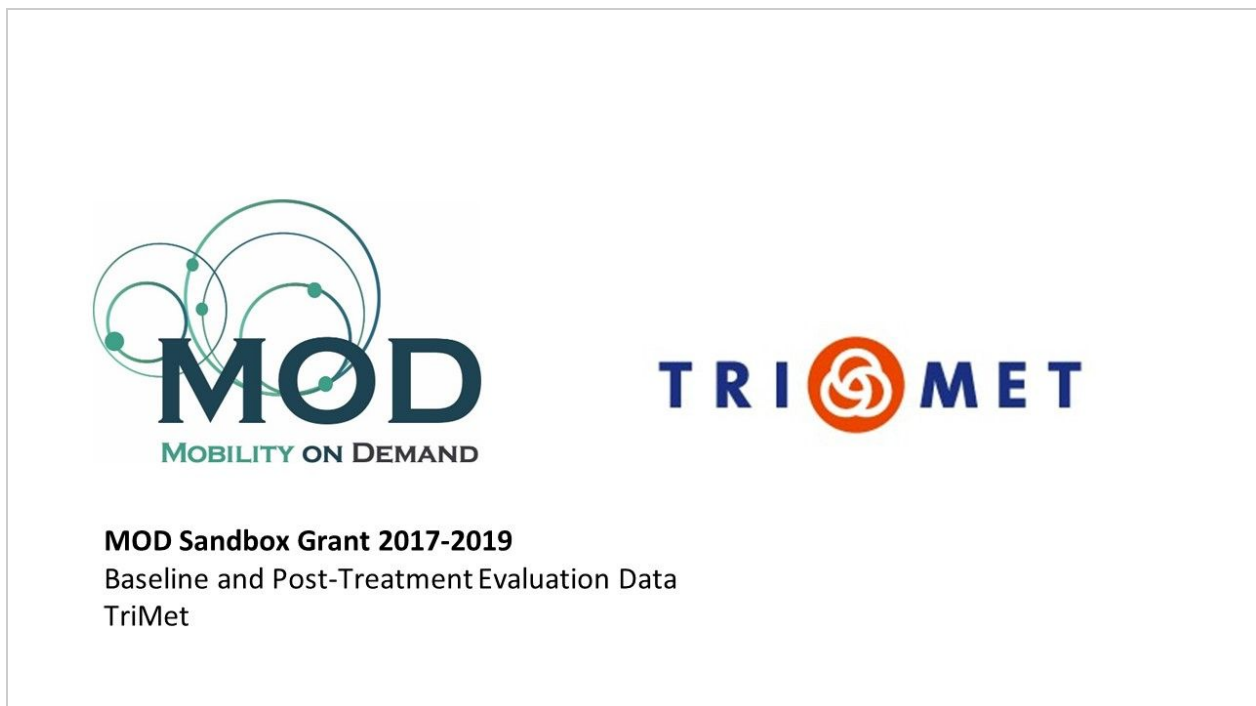
The FTA requires the following project evaluations and reports: Evaluation Plan and Report, Equity and Accessibility Plan, Knowledge Transfer, Field Demonstration, Final Project Report.

Quarterly Deliverables

- There were no scheduled deliverables for this task during this quarter.

Quarterly Progress

- TriMet has developed draft versions of both the Evaluation Plan and Report and the Equity and Accessibility Plan.



The draft Evaluation Plan and Report can be found on the TriMet Google Drive here:

https://drive.google.com/open?id=17Ok54d4-IqYNdY0dw96Soy1Lc05u_jjpi0G-yOvhukQ

TriMet's Evaluation Plan will evaluate the project on the following:

- Pelias Geocoder - match rate and accuracy improvements
- Trip Planner - time & cost comparisons, increased feasibility of routes
- User Satisfaction - application interface and travel options

Task 2: Evaluations and Reports (continued)

Equity & Accessibility

TriMet will develop a plan for how the project will address accessible and equitable mobility service for all travelers, including communities such as low income, the aging population, and persons with disabilities, including wheelchair users. In particular, the plan will discuss how the project will provide equivalent service for all travelers.

The draft Equity & Accessibility Report can be found here:

https://drive.google.com/open?id=17Ok54d4-IqYNdY0dw96Soy1Lc05u_jjpi0G-yOvhukQ

TriMet's Equity and Accessibility Plan will address the following:

- Accessible and equitable mobility service for all travelers
- Equivalent service for all communities, e.g., low income communities, aging populations, and persons with disabilities, including wheelchair users

Task 3: Application Development Status

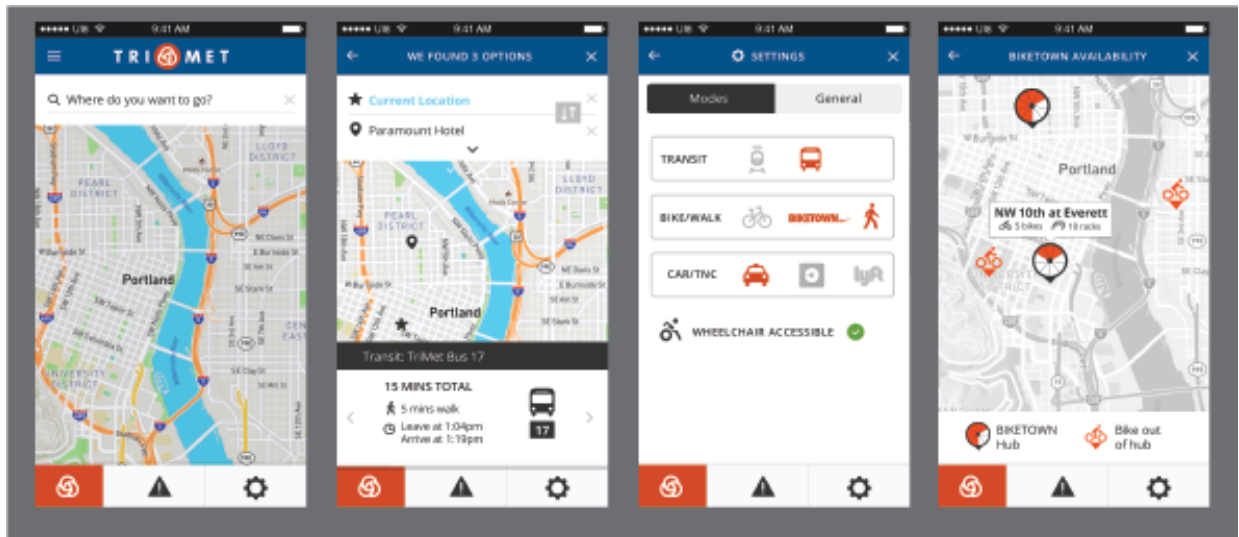
Significant progress has been made toward integrating shared-use mobility modes into the existing OpenTripPlanner application. Conveyal has designed prototypes of the new mobile-first app in InVision, with iterative improvements based on feedback from TriMet design staff.

Quarterly Deliverables

- Itinerary-Based Trip Planning feature for the OTP application (**Appendix C - Task 3 Milestone 1 Documentation**). The code for this deliverable is available on a private GitHub site until production.

Quarterly Progress

- Development and application design is ahead of schedule.
- Software tools for review and approval, as well as weekly webinars/conference calls facilitate progress.



Application Development Screenshots, Conveyal, Q1 2017

Task 4: Geocoder Development

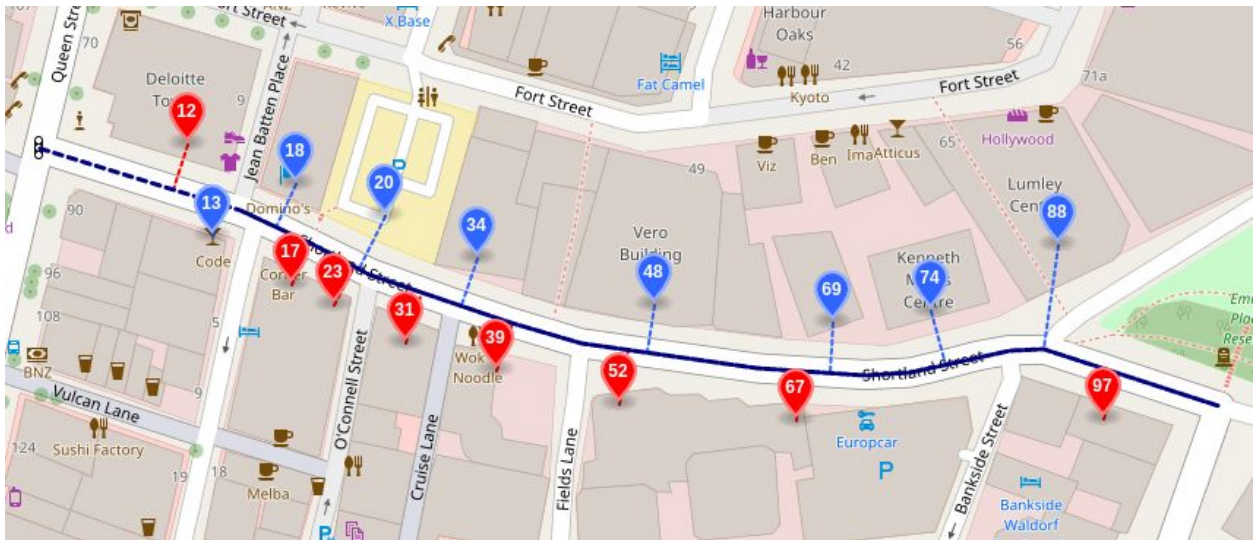
Pelias is a non-proprietary and non-restrictive option for address locating that is an important requirement for trip planning. This task includes the implementation of a reference framework for government agencies to auto-feed their authoritative address data into a publicly accessible geocoding service.

Quarterly Deliverables

- Point-based House Number Interpolation (Mapzen Milestone 5), an in-kind contribution (**Appendix D - Task 4 Milestone 5 Documentation**).
- Please note that due to contract delays, the in-kind work was performed first and the first quarter deliverable User Research Study (Mapzen Milestone 1) is scheduled for delivery next quarter.

Quarterly Progress

- Point-based House Number Interpolation has been completed. More information can be found here: <https://mapzen.com/blog/interpolation/>
- The code for the interpolation is available here: <https://github.com/pelias/interpolation>



Street Address Interpolation - Mapzen's Pelias Geocoder, February 2017

Task 5: Data Improvements

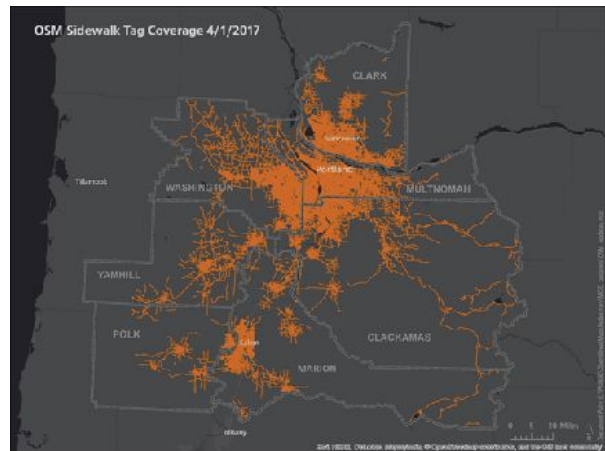
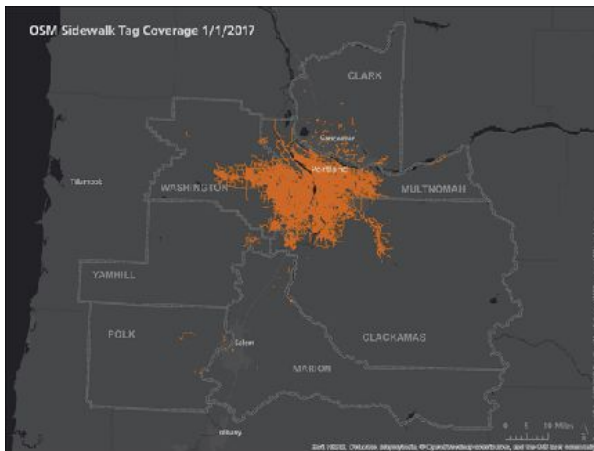
Improve OpenAddresses and OpenStreetMap (OSM) in support of comprehensive trip planning and geocoding (address matching).

Quarterly Deliverables

- There were no scheduled deliverables for this task during this quarter.

Quarterly Progress

- A substantial amount of work was performed on OpenStreetMap (OSM) data improvement.
- Work will continue throughout the length of the project.
- Improvements to the OpenStreetMap sidewalk data have been made. 35,000 segments have been tagged in this quarter.



The left image is from January 2017 and the right from April 2017. The percentage of appropriate streets tagged with sidewalks has increased from 35.7% to 72.2% during this quarter.

Task 6: Integrated Payment Plan

As a partner on this project, moovel will facilitate compatibility with their planned booking and payment features so customers can plan and pay for their trips in one app.

Quarterly Deliverables

- There were no scheduled deliverables for this task during this quarter.

Quarterly Progress

- moovel attended the project kickoff workshop and have participated in conversations.



TriMet's current mobile ticketing app, TriMet Tickets

Conferences, Meetings, Workshops

During this quarter, TriMet has organized and/or participated in the following conferences, workshops or meetings:

- January 18 – 19, Project Kickoff Workshop; Portland, OR
- February 1, NIST Global City Teams Challenge Super Action Cluster Summit, Presentation; Portland, OR
- April 5, TransITech Conference, Presentation; San Antonio, TX
- April 12, Shared-Use Mobility Center, Webinar Presentation
- April 20, Metro RLIS Stakeholders Meeting, Presentation; Portland, OR
- April 20, Mobility on Demand (MOD) Community of Practice Workshop; Washington, D.C.

TriMet conducts weekly project meetings on the following rotating Slack channels every Thursday at 10am PST. This quarter, they occurred on the following days:

- Geocoder Meetings (<https://trimet-mod-sandbox.slack.com/messages/geocoding/>) – February 9 and 23; March 9 and 23; April 6 and 20
- Application Development Meetings (<https://trimet-mod-sandbox.slack.com/messages/general/>) – February 16; March 2, 16 and 30; April 13 and 27

Upcoming Highlights

- TriMet is presenting *Integrating Transit with Shared-Use Mobility Options - MOD Sandbox Grant* at the Association for Commuter Transportation (ACT) conference, which will take place in New Orleans, LA July 30 - August 2, 2017, (http://www.actconf.org/full_schedule.cfm).
- TriMet has been selected to present *Solving the last mile problem with OpenTripPlanner (OTP), Mapzen Pelias, and open data* at the annual FOSS4G conference, which will take place in Boston, MA August 14 –19, 2017, (<http://2017.foss4g.org/accepted-presentations/#government>).

Lessons Learned

During this quarter, TriMet would like to share the following lessons learned:

1. Begin contract negotiations early.
 - a. Contracts always take longer than anticipated. Begin negotiations with partners as quickly as possible.
 - b. Look for opportunities up-front with other grant awardees to create a common contract, if applicable, for potential time and cost savings.
2. Develop the initial MOD Sandbox Grant application under the assumption that funding will be received.
 - a. The timeline is critical and doing as much as you can in advance will ensure you're ready to start upon award. It is also an opportunity to begin contract negotiations.
3. Schedule the kick-off workshop and involve all the partners and stakeholders as quickly as possible.
 - a. Refining the project scope and technical details collaboratively provided a better foundation and was more time and cost efficient than developing independently in advance for review.
 - b. Have the developers lead the technical aspects and provide a forum for them to collaborate as early as possible.
4. Anticipate change and prepare to respond quickly, creatively and collaboratively.
 - a. In addition to authoritative address data for geocoding purposes, transit specific data is also necessary to support transit trip planning. Rather than supplement the transit data outside of Mapzen's Pelias engine, as originally designed, it became evident that developing an additional loader for transit data would be more efficient and would facilitate replicability by other transit agencies. To accommodate the existing scope and budget, TriMet will be contributing the "Transit Data Loader" to the Pelias project and the open source code will then be maintained and managed by Mapzen.
5. Create a public website/dashboard for the project.
 - a. It increases transparency and information sharing.
 - b. It is more efficient and comprehensive to refer inquiries about the project to a website than addressing inquiries on an individual request basis (and you will get a lot of requests for more information!).

Lessons Learned (continued)

- c. Automate the updates of information and data on the website to reduce overhead and maintenance and to ensure consistency in content and information.
6. Collaboration tools are critical. For this project, TriMet is using the following tools:
 - a. Trello for project management (<https://trello.com/>)
 - b. Google Drive for document management (<https://drive.google.com>)
 - c. Slack for communication (<https://slack.com/>)
 - d. InVision for prototyping and design (<https://www.invisionapp.com/>)

For more info and up-to-date progress, please go to www.trimet.org/mod. This dashboard was created by TriMet to provide a snapshot of the MOD Sandbox project's progress. These key indicators help track progress toward project deliverables and the budget. In addition, we will be sharing project updates and lessons learned through this dashboard.

Appendices

Project Management Plan

Appendix A

Kickoff Workshop Report

Appendix B

Task 3 Milestone 1 Documentation

Appendix C

Task 4 Milestone 5 Documentation

Appendix D

APPENDIX A

TriMet MOD Grant Project Management Plan



Project Management Plan (PMP)

OTP SUM: OTP Integration of Transit with Shared-Use Mobility Real-Time and Data Enhancements Mobility on Demand Sandbox Program

**Prepared for
Federal Transit Administration (FTA)
U.S. Department of Transportation**

**Prepared by
Tri-County Metropolitan Transportation District of Oregon
(TriMet)**

March 9, 2017

Table of Contents

Contents

| | |
|---|-------------------------------------|
| DRAFT PROJECT MANAGEMENT PLAN (PMP) | 1 |
| TABLE OF CONTENTS | 3 |
| LIST OF FIGURES | 5 |
| LIST OF TABLES | 6 |
| 1 PROJECT OVERVIEW | 7 |
| 1.1 BACKGROUND..... | 7 |
| 1.2 PROJECT SUMMARY | 7 |
| 1.3 PROJECT TASKS AND DELIVERABLES | 8 |
| TASK 1. PROJECT MANAGEMENT | 8 |
| TASK 2. EVALUATIONS AND REPORTS | 9 |
| 2a. <i>Equity and Accessibility</i> | 9 |
| 2b. <i>Evaluation Data Collection and Coordination with the Independent Evaluator</i> | 9 |
| 2c. <i>Knowledge Transfer</i> | 9 |
| 2d. <i>Field Demonstration</i> | 10 |
| 2e. <i>Project Report</i> | 10 |
| TASK 3. APPLICATION DEVELOPMENT | 10 |
| TASK 4. GEOCODER DEVELOPMENT | 11 |
| TASK 5. DATA IMPROVEMENTS | 12 |
| TASK 6. INTEGRATED PAYMENT PLAN | 12 |
| 1.4 SCHEDULE OF TASKS, MILESTONES, AND DELIVERABLES..... | 13 |
| 1.5 BUDGET | 15 |
| 1.5.1 <i>Budget by Fiscal Year</i> | 15 |
| 1.5.2 <i>Budget by Category</i> | <i>Error! Bookmark not defined.</i> |
| 1.5.3 <i>Budget by Task</i> | 15 |
| 1.6 EVOLUTION OF THE PROJECT MANAGEMENT PLAN..... | 16 |
| 1.7 REFERENCE MATERIALS | 16 |
| 2 PROJECT TEAM ORGANIZATION | 17 |
| 2.1 ORGANIZATIONAL STRUCTURE | 17 |
| 2.2 TEAM ROLES AND RESPONSIBILITIES | 19 |
| • <i>TriMet</i> | 19 |
| • <i>IBI Group</i> | 19 |
| • <i>Conveyal</i> | 19 |
| • <i>Mapzen</i> | 19 |
| 2.3 STAFFING PLAN..... | 19 |
| 3 PROJECT COMMUNICATIONS, MONITORING AND CONTROL | 22 |
| 3.1 COORDINATION AND COMMUNICATIONS | 22 |
| 3.1.1 <i>Communications and Meeting Plan</i> | 22 |
| 3.2 PROJECT TRACKING | 22 |
| 3.3 SCOPE, SCHEDULE AND BUDGET MANAGEMENT | 23 |
| 3.3.1 <i>Scope Management</i> | 23 |
| 3.3.2 <i>Schedule Management</i> | 23 |
| 3.3.3 <i>Cost/Budget Management</i> | 23 |
| 3.4 CHANGE MANAGEMENT..... | 23 |

| | | |
|----------|---|-----------|
| 3.5 | QUALITY MANAGEMENT | 23 |
| 3.6 | RISK MANAGEMENT | 24 |
| 4 | PROJECT REPORTING | 25 |
| 4.1 | PROJECT REPORTING | 25 |
| 4.2 | DOCUMENT REVIEW/REVISION/ACCEPTANCE PROCESS | 25 |
| | APPENDIX A DOCUMENT VERSION CHANGES | 26 |

List of Figures

| | |
|--|----|
| Figure 2-1: Key Project Team Members Organization Chart..... | 18 |
|--|----|

List of Tables

No table of figures entries found.

1 PROJECT OVERVIEW

1.1 Background

The Open Trip Planner (OTP), initially released as an open source project by TriMet in 2009, was the first to introduce multiple modes in one trip with the original focus on incorporating biking and walking networks with transit. Adoption of OTP has been strong, with implementation in dozens of cities and countries worldwide. TriMet now proposes to build upon the core of OTP to incorporate shared-use mobility (SUM) options.

TriMet's OTP SUM project will create a complete open platform for the integration of transit and SUM options. The open data, software and user interfaces, responsive on both web and mobile, will help customers understand the multi-modal options to meet their mobility needs, including for the critical first and last miles of transit trips where a bus or train alone doesn't directly serve their origin or destination

1.2 Project Summary

TriMet's proposed project includes the development and expansion of two core data frameworks that current and future collaborative OTP initiatives can be built upon, producing replicable software and results for communities across the country. These two core project elements are to:

- Extend the OTP code base to integrate into transit trip planning shared-use mobility modes, such as bike share and TNCs, as well as updated real-time transit information.
- Implement a fully-functional and comprehensive open source geocoder built off the existing open source Mapzen Pelias geocoder. Geocoding, or address locating, is a primary requirement for trip planning. A non-proprietary and non-restrictive option for address locating would substantially lower the barrier to entry for many transit systems to offer a trip planning tool and can achieve significant cost savings for transit agencies, government agencies, and the public.

In addition to core elements on the foundation frameworks, the project will also include:

- Development of a comprehensive new web-based user interface that will allow users to make intermodal trip plans including shared-use modes. The new web-based user interface will also display real-time information and report impacted itineraries to users.
- Improvements to basemap data so the trip planner can support enhanced pedestrian/ wheelchair accessibility information for customers; and improvements to regional address data that will make location search and geocoding more effective and user-friendly.

- Design and implementation of compatibility for future booking and payment options in moovel’s RideTap product so customers can plan and pay for their trips in one app.

The resulting systems, all open source software and open data, will support the rapid deployment of the intermodal transit trip planner throughout the transit industry. The open source trip planner is sustainable beyond the Mobility on Demand (MOD) demonstration, and will be able to leverage new enhancements as they get rolled out to the OTP development community and transit industry. Future enhancements could include full integration with a mobile ticketing platform, meeting a common request of transit customers. Collaboration in the transit and open source software development community is growing in strength; it is important to leverage these resources as new software is constantly under development. This collaborative approach will continue to empower public transit agencies to provide low cost, sustainable, scalable solutions to customers at a national level.

1.3 Project Tasks and Deliverables

Below are the major tasks of the project and associated deliverables for each of the tasks.

Task 1. Project Management

IBI Group will be responsible for managing the project with assistance from the FTA project manager. This task provides for the overall project’s management and coordination. Included in this effort are:

- An initial kickoff meeting
- Development of a project management plan (PMP)
- Maintenance of a project scope, schedule, and budget
- Project progress reporting to FTA via periodic meetings and quarterly reports

The recipient will conduct the project in accordance with the FTA Master Agreement (<https://www.transit.dot.gov/about/regional-offices/region-10/fy-2016-master-agreement>) and Circular 6100.1E (<https://www.transit.dot.gov/regulations-and-guidance/fta-circulars/research-technical-assistance-and-training-program>).

Deliverables:

- Kickoff meeting, including meeting materials and notes
- Draft PMP
- Final PMP
- Periodic meetings (e.g., conference calls, site visits)
- Quarterly progress reports

Task 2. Evaluations and Reports

2a. Equity and Accessibility

TriMet will develop a plan for how the project will address accessible and equitable mobility service for all travelers, including communities such as low income, the aging population, and persons with disabilities, including wheelchair users. In particular, the plan will discuss how the project will provide *equivalent* service for all travelers.

Deliverables:

- Draft plan on MOD equity and accessibility
- Final plan on MOD equity and accessibility

2b. Evaluation Data Collection and Coordination with the Independent Evaluator

The MOD Sandbox Demonstration local team will support the independent evaluation by providing input and review during evaluation planning and execution. The local team will coordinate with the independent evaluator to assist it in developing an evaluation plan and will provide the independent evaluator baseline and post-treatment qualitative data or qualitative and quantitative data as specified in the evaluation plan. The MOD Sandbox Demonstration local team will support the independent evaluator by providing access to the local team staff for surveys, interviews, and/or focus groups as identified in the evaluation plan. The local team will assist the independent evaluator with organizing and conducting surveys, interviews, and/or focus groups of team staff and MOD users as identified in the evaluation plan.

Examples of data that may be provided to the independent evaluator include, but are not necessarily limited to, the following:

- Please list as appropriate.
- Number of API calls, trip planner sessions, and percentage of correct geo-coder inquiries per typical weekday.

Deliverables:

- Baseline and post-treatment evaluation data

2c. Knowledge Transfer

The MOD Sandbox Demonstration local team will assist the FTA with MOD knowledge transfer. Such knowledge transfer activities may include, but may not necessarily be limited to, coordinating and communicating with other USDOT MOD activities, participating in USDOT-sponsored MOD Sandbox workshops and meetings, sharing information with other MOD Sandbox Demonstration sites, and communicating with industry organizations to provide awareness and knowledge transfer of the project and its scope, status, and results. Examples of industry organizations are the American Public

Transportation Association (APTA), Community Transportation Association of America (CTAA), ITS America, and Transportation Research Board (TRB).

The local team will assist FTA in producing briefing and presentation materials concerning their MOD Sandbox Demonstration project as needed and may be requested to produce supporting multimedia materials (pictures, video clips).

Deliverables:

- Presentation and briefing materials, including pictures and video clips
- OTP roadmap developed as part of the initial workshop
- Overview summary of the OTP system at the end of the project

2d. Field Demonstration

The MOD Sandbox Demonstration site will operate and maintain the MOD system for 6 months, as well as support the data requirements of the evaluation as discussed in Task 2c.

Deliverables:

- Demonstration

2e. Project Report

Towards the end of the project, TriMet will produce a project report describing the MOD system and documenting the project process, results, lessons learned, recommendations for future research, etc. from the local perspective. Evaluation results do not need to be included in the project report as the independent evaluator will be responsible for producing an evaluation report for the site.

The project report will be required to meet FTA reporting guidelines. The FTA reporting guidelines may be found at:

- http://www.fta.dot.gov/about/12351_8850.html and
- <https://www.transit.dot.gov/research-innovation/preparationinstructionsforftafinalreportsjune2013>

Deliverables:

- Draft project report
- Final project report

Task 3. Application Development

Application development includes enhancements to TriMet's existing OTP-based multimodal trip planning application, including both the underlying multimodal routing engine and the user-facing web interface.

Routing capabilities will be extended to reflect ongoing trends in traveler behavior and open data availability. Enhancements include the ability to incorporate shared-mobility services into multimodal trip planning (for example, use of a TNC service such as Uber or Lyft to access transit). Advances in the quality and availability of real time transit data will also be incorporated, with enhancements to the routing engine's ability to consume real-time data and modify trip plans accordingly. Other enhancements include support for the General Bikeshare Feed Specification (GBFS), and improved support for planning and describing wheelchair-accessible trips.

In addition to the enhanced routing capabilities, a comprehensive new web-based user interface (UI) will be developed. The new UI will incorporate aspects from existing OTP front-end projects, including TriMet's existing interactive trip planner, the otp.js library, and Conveyal's Modeify project. The new UI will be written using modern web development practices and frameworks, including the React framework and Redux architecture. This architecture emphasizes modularity and reusability of components in a variety of contexts; the intention is to build a library that not only serves as the foundation for a comprehensive new OTP UI but also serves as a resource for developers working on complementary project

Deliverables:

- Itinerary-based trip planning
- Search options and profiles
- Real-time integration
- Shared-use mobility integration
- Wheelchair/pedestrian routing
- Extended UI functionality
- Integration with TriMet website

Task 4. Geocoder Development

Mapzen is well positioned to implement a reference platform that will allow government transit agencies to feed their authoritative address data into a fully featured publicly accessible geocoding service. This can be done by leveraging the existing [OpenAddresses](#) framework and the public [Mapzen Search API](#).

Although Mapzen Search already ingests OpenAddresses data on a regular basis there is a significant amount of work to be done to make it easier for agencies to feed their address data into this system. Mapzen will work with TriMet and Metro to forward a sustainable, intuitive, and resilient solution that will allow any authoritative address data to be added to the open transit ecosystem and ensure that sufficient user tutorials and documentation exist throughout the system. It is important to note that all the work will be open-sourced and based entirely on open data.

Mapzen will focus on validating this system across other agencies, not only those involved directly in the MOD project to ensure that it is in fact repeatable.

Deliverables:

- User research study
- Interactive data management tools
- Data ingestion pipeline
- Local installation package
- Point-based house number interpolation
- Testing and validation framework

Task 5. Data Improvements

Improvements will be made to the OpenStreetMap (OSM) basemap data so the trip planner can support enhanced pedestrian/ wheelchair accessibility information for customers. Regional address data will also be significantly improved to support location search and geocoding.

Deliverables:

- OSM improvements
- Regional address improvements

Task 6. Integrated Payment Plan

moovel will design and implement compatibility for future booking and payment options in moovel's RideTap product so customers can plan and pay for their trips in one app.

Deliverables:

- Report of findings to be included in final project report

1.4 Schedule of Tasks, Milestones, and Deliverables

Figure 1-1 Project Schedule

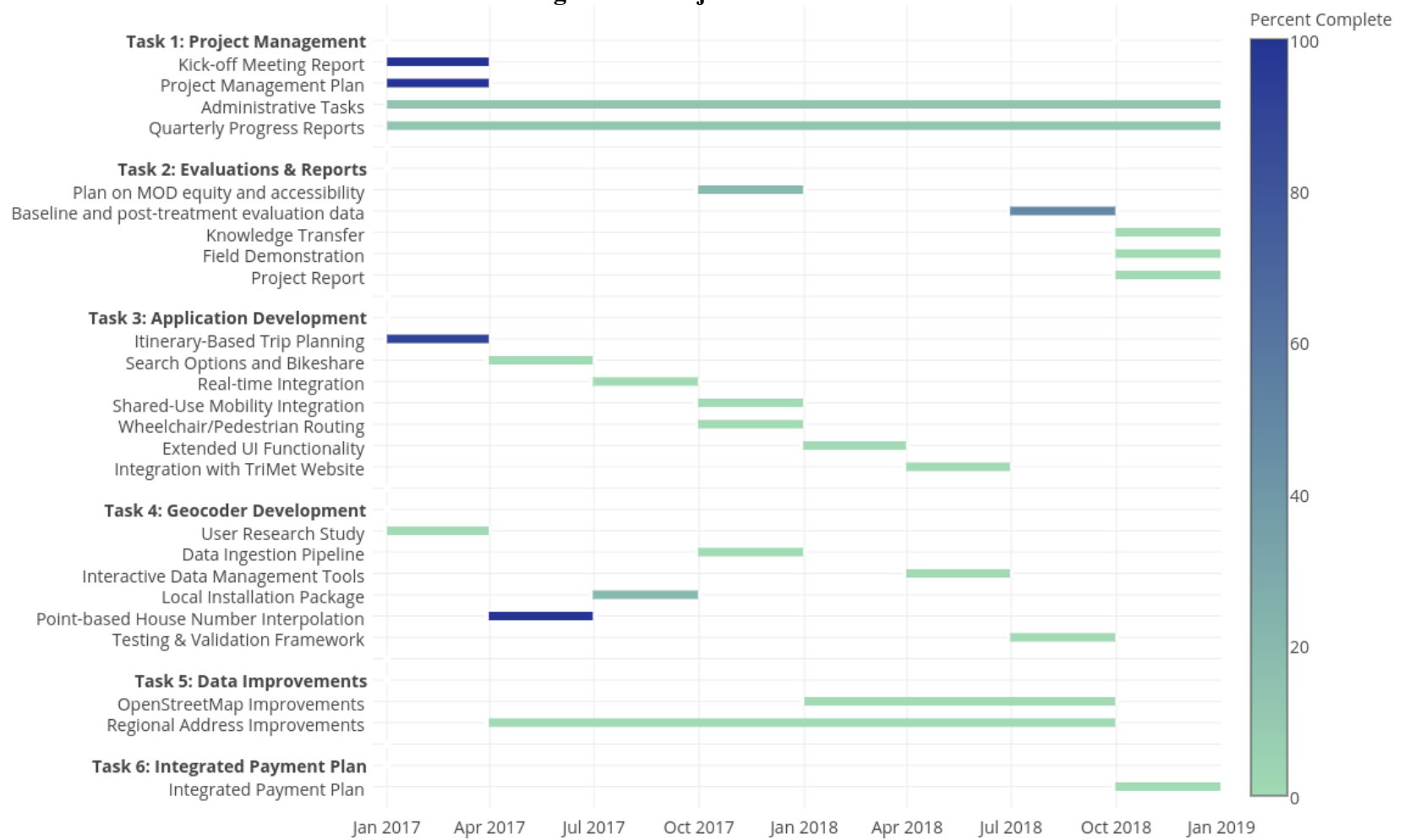


Table 1-1 Deliverables Schedule

| Task No. | Task | Deliverable(s) | Delivery Date (months from project execution) |
|-----------------|-------------------------|---|--|
| 1 | Project Management | Kickoff meeting materials and notes | One month |
| | | Draft PMP | One month |
| | | Final PMP | Two months |
| | | Administrative Tasks (periodic meetings, conference calls, site visits) | Ongoing |
| | | Progress reports | One month after end of Federal fiscal quarter |
| 2 | Evaluations and Reports | Draft MOD equity and accessibility plan | Six months |
| | | Final MOD equity and accessibility plan | Six months |
| | | Evaluation Data Collection and Coordination with the Independent Evaluator | To be specified in the evaluation plan |
| | | Knowledge Transfer: Presentation and briefing materials, including pictures and video clips | As requested |
| | | Field demonstration start | Twenty four months |
| | | Field demonstration completion | Twenty four months |
| | | Draft project report | Twenty three months |
| | | Final project report | Twenty four months |
| 3 | Application Development | Itinerary-Based Trip Planning (Conveyal Milestone 1) | Three months |
| | | Search Options and Bikeshare (Conveyal Milestone 2) | Six months |
| | | Real-time Integration (Conveyal Milestone 3) | Nine months |
| | | Shared-Use Mobility Integration (Conveyal Milestone 4) | Twelve months |
| | | Wheelchair/Pedestrian Routing (Conveyal Milestone 4) | Twelve months |
| | | Extended UI Functionality (Conveyal Milestone 5) | Fifteen months |
| | | Integration with TriMet Website (TriMet Milestone) | Eighteen months |
| 4 | Geocoder Development | User Research Study (Mapzen Milestone 1) | Three months |

| Task No. | Task | Deliverable(s) | Delivery Date (months from project execution) |
|-----------------|-------------------------|---|--|
| | | Data Ingestion Pipeline (Mapzen Milestone 2) | Twelve months |
| | | Interactive Data Management Tools (Mapzen Milestone 3) | Eighteen months |
| | | Local Installation Package (Mapzen Milestone 4) | Nine months |
| | | Point-based House Number Interpolation (Mapzen Milestone 5) | Six months |
| | | Testing & Validation Framework (Mapzen Milestone 6) | Twenty one months |
| 5 | Data Improvements | OpenStreetMap Improvements (TriMet Milestone) | Twenty one months |
| | | Regional Address Improvements (Metro Milestone) | Twenty one months |
| 6 | Integrated Payment Plan | Integrated payment plan and report (moovel Milestone) | Twenty four months |

1.5 Budget

1.5.1 Budget by Fiscal Year

The total budget of the project is \$1,002,000, of which \$678,000 is the MOD Sandbox Federal amount, and \$324,000 is in-kind contributions by project team partners. Anticipated budget breakdown by Federal fiscal year is as follows:

| | |
|---------|-----------|
| FY 2017 | \$379,000 |
| FY 2018 | \$448,750 |
| FY 2019 | \$174,250 |

1.5.2 Budget by Task

Budget breakdown by task is contained in Table 1-3 below.

Table 1-3 Project Budget by Task

| Tasks and Other Activities | MOD Sandbox Federal Amount (\$) | MOD Sandbox Cost Share (\$) | Total Cost (\$) |
|-----------------------------------|--|------------------------------------|------------------------|
| | | | |

| | | | |
|---------------------------------|------------------|------------------|--------------------|
| Task 1: Project Management | \$80,000 | \$20,000 | \$100,000 |
| Task 2: Evaluations and Reports | \$40,000 | \$50,000 | \$90,000 |
| Task 3: Application Development | \$270,000 | \$70,000 | \$340,000 |
| Task 4: Geocoder Development | \$200,000 | \$64,000 | \$264,000 |
| Task 5: Data Improvements | \$70,000 | \$20,000 | \$90,000 |
| Task 6: Integrated Payment Plan | | \$100,000 | \$100,000 |
| Travel and Incidentals | \$18,000 | | \$18,000 |
| Total Cost | \$678,000 | \$324,000 | \$1,002,000 |
| Cost Share Breakdown | 68% | 32% | |

1.6 Evolution of the Project Management Plan

To be an effective management and communication tool, the plan must be a living document that is updated as conditions change. At a minimum, the project management team will review the PMP quarterly, and as major milestones are achieved. The version changes for the PMP are recorded in Appendix A.

1.7 Reference Materials

Additional information on schedule and budget can be found at the project website (<https://trimet.org/mod/>).

2 PROJECT TEAM ORGANIZATION

2.1 Organizational Structure

The organizational chart on the following page shows the key members of the OTP SUM team. The team is structured primarily around the main software development tasks (for OTP and the Pelias geocoder). In addition to being the client agency and lead on the grant, TriMet is responsible for integration of OTP SUM and the Pelias geocoder with existing systems at the Agency, including the TriMet.org website. IBI Group provides project management support, stakeholder outreach, and overall technical coordination between the development teams.

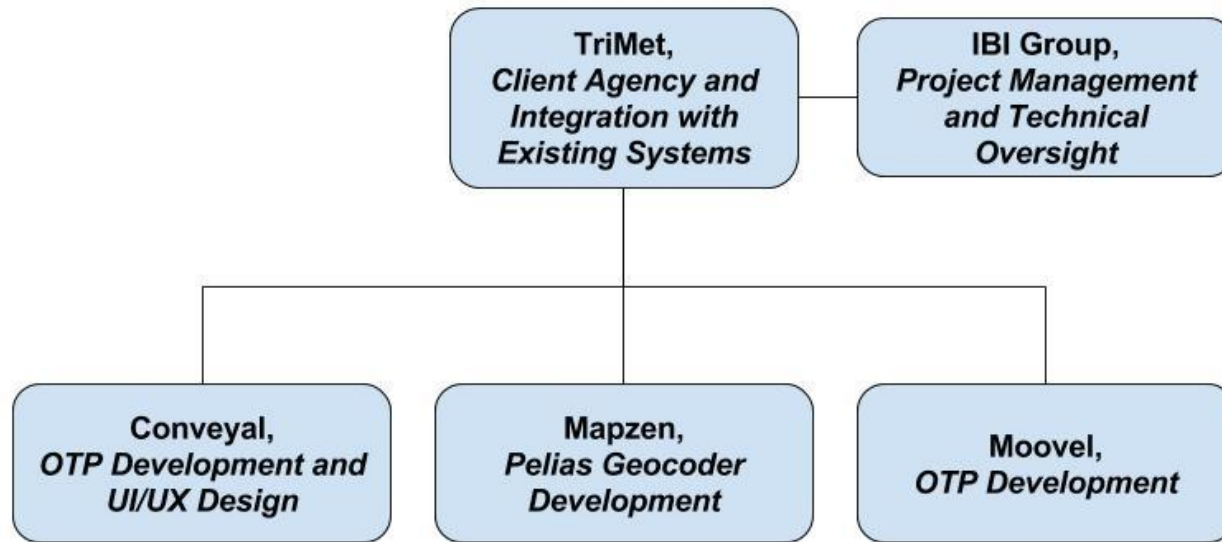


Figure 2-1: Key Project Team Members Organization Chart

2.2 Team Roles and Responsibilities

The roles and responsibilities of the team are highlighted below.

- TriMet
 - Lead Agency
 - Project Management
 - Local implementation of OTP and Pelias geocoder
- IBI Group
 - Project Management
 - Stakeholder Coordination
 - Technical Oversight
- Conveyal
 - Development team for OTP routing engine and front-end JS library
 - UI/UX Design
- Mapzen
 - Development team for geocoding engine, and OpenAddress data ingestion tool
- Moovel
 - Integrated payment plan

2.3 Staffing Plan

The staffing plan is outlined below in Tables 2-1, which identifies the tasks in which the key staff will be involved and the key staffs' general functions.

Table 2-1 Project Key Staff and Functions

| Organization | Name | Contact Information | Project Task Number(s) | Role/Function |
|--------------|-----------------|---------------------|------------------------|--|
| TriMet | Bibiana McHugh | mchughb@trimet.org | All | Principal Investigator |
| | Madeline Steele | steelem@trimet.org | 1, 2c, 5 | Project coordination, OSM data improvements lead |

| | | | | |
|-----------|------------------|----------------------------|---------|--|
| | Tom Lin | lint@trimet.org | 3, 4, 5 | General tech support, Geocoder testing |
| | Frank Purcell | purcellf@trimet.org | 3 | General OTP support |
| | Grant Humphries | humphrig@trimet.org | 3, 4 | Implementation of Pelias geocoder and OTP |
| | John Zimmerman | zimmermj@trimet.org | 3, 4 | Implementation of Pelias geocoder and OTP |
| | Ginger Shank | shankv@trimet.org | 2b, 2d | Evaluation project lead, Beta testing project lead |
| | Dave Whipple | whipped@trimet.org | 3 | UI/UX design and TriMet branding |
| | Hannah Quinsey | quinseyh@trimet.org | 2a | Accessibility advisor |
| | Jake Warr | warrj@trimet.org | 2a | Equity advisor |
| IBI Group | Ritesh Warade | ritesh.warade@ibigroup.com | 1, 2c | Project management, Stakeholder outreach |
| | Jon Campbell | jon.campbell@ibigroup.com | 1, 2c | Project management, Stakeholder outreach |
| Conveyal | Dave Emory | demory@conveyal.com | 3 | OTP development lead |
| | Kate Chanba | kchanba@conveyal.com | 3 | UI/UX design lead |
| | Landon Reed | lreed@conveyal.com | 3 | OTP development |
| Mapzen | Diana Shkolnikov | diana@mapzen.com | 4 | Pelias development lead |
| Moovel | Ali Waters | ali.waters@moovel.com | 6 | Integrated payment plan |

| | | | | |
|--|-------------------|---------------------------|---|----------------------------|
| | Regina Clewlow | regina.clewlow@moovel.com | 6 | Integrated payment plan |
|--|-------------------|---------------------------|---|----------------------------|

3 PROJECT COMMUNICATIONS, MONITORING AND CONTROL

3.1 Coordination and Communications

Communications between the project team and the USDOT and its contractors will be primarily between the project team lead and FTA project manager. However, direct communications may be made between various MOD program entities (e.g., between the project team evaluation lead and independent evaluator) while informing the FTA project manager of the correspondence.

The following sections describe the proposed mechanisms for communicating and coordinating among the various program and project participants, and at the various management levels of the project. Meetings may be conducted in person or via conference call or webinar.

3.1.1 Communications and Meeting Plan

- Kick-off meeting to discuss the project goals and expectations, specifically the project approach, tasks and deliverables, partner roles and responsibilities, staffing plan, schedule, budget, and travel requirements.
- Bi-weekly (every other week) project status meetings and conference calls to review project activities and status. The meetings can be with all project team members or only among project management team members. An agenda and standard check-list will be developed to assure all aspects are covered at the meeting.
- Monthly project status meetings between OTP SUM project management team and FTA Project Manager.
- The USDOT may conduct site visits periodically throughout the project as a part of its project management and oversight responsibilities.
- The USDOT may organize workshops or communities of practice among the various program participants to discuss and address certain issues or topics of interest among MOD Sandbox Demonstration sites and/or the transit industry.

3.2 Project Tracking

The project will be tracked via the following methods and measures:

- A project website (<https://trimet.org/mod/>) has been developed for tracking progress. Most of the website is public facing, incorporating the following project tracking elements:
 - Interactive Gantt chart tracking task progress
 - Quarterly reports
 - Trello dashboard for real time task status, to-do lists and schedule milestones
 - Event and meeting summaries
- An internal version of the website also includes budget tracking

3.3 Scope, Schedule and Budget Management

The following sections outline the approach for managing the project scope, schedule and budget.

3.3.1 Scope Management

The request for scope changes will be first discussed within the project team and then elevated to FTA project manager for approval. A cooperative agreement revision/amendment is required by FTA for material changes in the work scope, in accordance with FTA Circular 6100.1E (IV.6)

3.3.2 Schedule Management

A Gantt chart and deliverables table will be used to monitor the schedule. During project meetings, team members will review the schedule status, and discuss actions/directions required to resolve schedule issues, if any. Minor schedule adjustments – those that do not affect the overall project schedule/timeframe – may be approved by the project team lead in consultation with the FTA project manager. Significant schedule changes will be decided by the project team and then elevated to FTA project manager for review and approval. A significant schedule change may be accomplished via an administrative amendment as outlined in FTA Circular 6100.1E (IV.6)

3.3.3 Cost/Budget Management

Cost/budget will be managed by the project team lead following Federal rules, regulations, and laws and local (e.g., state) rules, regulations and laws. Invoices will be submitted after the project team lead (or designated official) reviews the project progress, schedule, and expenditures. Payments will be based on progress (costs incurred for a period of time) or milestone completion. Requests for payments will be conducted in accordance with FTA Circular 6100.1E (V.8).

It is not anticipated that the budget for this project will change. Budget revisions, if any, will be conducted in accordance with FTA Circular 6100.1E (IV.6).

3.4 Change Management

As this is primarily a software development and implementation project, software changes are managed by GitHub repositories. Public repositories are available for all major project components. Software changes for TriMet specific implementations of OTP and Pelias are maintained in a separate, public repository. Other agencies can follow similar procedures.

3.5 Quality Management

Partners such as Conveyal and Mapzen will provide the main branch of stable code release as well as branches for development and staging. All deployments will be fully tested before going into production.

3.6 Risk Management

Potential risks and possible mitigation measures are identified in Table 3-1. These risks will be closely monitored and evaluated using the project progress review process. The table will be updated when status changes or new risks are identified.

Table 3-1 Potential Project Risks

| Potential Risks | Mitigation Measures | Status |
|---|---|---|
| Data providers (such as Uber, Lyft, Biketown) will not provide required data in required format | TriMet working with other MOD Sandbox grant recipient agencies and TransitCenter to harmonize data requests from providers and work together to reach agreements with data providers. | OTP SUM team developing initial data request and requirements |
| OTP improvements will be delayed | OTP SUM team conducting progress review meetings every two weeks with development team | Development teams are on schedule |
| Geocoder improvements will be delayed | OTP SUM team conducting progress review meetings every two weeks with development team | Development teams are on schedule |

4 PROJECT REPORTING

4.1 Project Reporting

The following reports will be produced to provide information on the project progress:

- Quarterly Progress Report – The project team lead shall submit a quarterly progress report (in MS Word format) to the FTA project manager via e-mail by the end of the month following the federal fiscal quarter, namely April 30, July 31, October 31, and January 31. The report shall include the significant accomplishments for the quarter; anticipated work for the following quarter; issues, if any, and recommended solutions; expenditures of the quarter and to date (cumulative), and submittal status of deliverables (see also FTA Circular 6100.1E [IV.4.d]). These reports will be posted in TRAMS by the FTA project manager.
- Milestone Progress Report – The project team lead (or the designated official) will provide a brief summary of the project progress, including milestone status, for the quarter in TRAMS as outlined in FTA Circular 6100.1E (IV.4.d). The report will be completed by the end of the month following the federal fiscal quarter.
- Federal Financial Report – The project team lead (or designated official) will submit a financial status report for the quarter in TrAMS as outlined in FTA Circular C 6100.1E (IV.4.c). The report will be completed by the end of the month following the federal fiscal quarter.

4.2 Document Review/Revision/Acceptance Process

The deliverable review flow is as follows: draft for internal team review and comment → revision (if required) → draft for FTA review and comment → revision/final draft → team review (and revision if required) → submit to FTA for final review (and comment if required) → approval or final revision if required. The document review schedule will be closely monitored and tracked.

Appendix A Document Version Changes

| Version No. | Date | Description of Changes | Status |
|--------------------|-------------|---|---------------|
| 1 | 03/16/2017 | Initial draft | |
| 2 | 3/29/2017 | Removed section 1.5.2 – Budget by Category at direction of Nazy Sodhi, FTA. | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

APPENDIX B

TriMet MOD Grant Kickoff Workshop Report

TriMet
OTP SUM: OTP Integration of Transit with Shared-Use Mobility,
Real-Time, and Data Enhancements

**Mobility on Demand Sandbox Program
Workshop Kickoff Report
January 18-19, 2017**

PROJECT SUMMARY

Overview

TriMet's project includes the development and expansion of two core data frameworks that current and future collaborative OpenTripPlanner (OTP) initiatives can be built upon, producing replicable software and results for communities across the country. These two core project elements are to:

- Extend the OTP code base to integrate shared-use mobility modes into transit trip planning, such as bike share and Transportation Network Companies (TNCs), and updated real-time transit information.
- Implement a fully-functional and comprehensive open source geocoder built off the existing open source Mapzen Pelias geocoder. Geocoding, or address locating, is a primary requirement for trip planning. A non-proprietary and non-restrictive option for address locating would substantially lower the barrier to entry for many transit systems to offer a trip planning tool and can achieve significant cost savings for transit agencies, government agencies, and the public.

In addition to developing and expanding core elements on the foundation frameworks, the project will also include:

- Development of a new, comprehensive web-based user interface that will allow users to make intermodal trip plans including shared-use modes. The new interface will also display real-time information and report impacted itineraries to users.

- Improvements to basemap data, enabling the trip planner to support enhanced pedestrian/wheelchair accessibility information.
- Improvements to regional address data that will make location search and geocoding more effective and user-friendly.
- Design and implementation of compatibility for future booking and payment options in moovel’s RideTap product, allowing customers to plan and pay for their trips in one app.

The resulting systems, all utilizing open source software and open data, will support the rapid deployment of the intermodal transit trip planner throughout the transit industry. The open source trip planner is sustainable beyond the Mobility on Demand (MOD) demonstration, and will be able to leverage new enhancements as it gets rolled out to the OTP development community and transit industry. Future enhancements could include full integration with a mobile ticketing platform, meeting a common request of transit customers. Collaboration in the transit and open source software development community is growing in strength; it is important to leverage these resources as new software is constantly under development. This collaborative approach will continue to empower public transit agencies to provide low cost, sustainable, scalable solutions to customers at a national level.

Project Goals and Objectives

Goal: Improve the open source, non-proprietary OTP system and make it easier to deploy for other transit agencies.

Specific objectives:

- Allow users to get information about and compare Shared Use Mobility (SUM) options in addition to transit, bike, and walking options in OTP
- Provide users with a more accurate matching of addresses when using OTP
- Improve the usability and design of the web-based OTP interface
- Provide users with real-time information regarding their trip plans, and any impacts thereon

Project Key Partners

| Organization | Organization Type | Role | Contact |
|---------------------|--------------------------|-------------------------|------------------|
| IBI | For-profit entity | Project Management | Ritesh Warade |
| Conveyal | For-profit entity | Application Development | David Emory |
| Mapzen | For-profit entity | Geocoder Development | Diana Shkolnikov |

| | | | |
|-----------------------------------|-------------------|---|----------------|
| Oregon Metro Data Resource Center | MPO | Regional address data maintainer/provider | Robert Kirkman |
| moovel | For-profit entity | Integrated payment plan | Regina Clewlow |

WORKSHOP GOALS AND OBJECTIVES

On January 18th and 19th, 2017, TriMet held a two-day workshop to kickoff the OTP SUM project. With attendees representing not only the OTP SUM project team, but also other OTP stakeholders, the workshop had the following goals and objectives.

Goals:

- Establish a vision for OpenTripPlanner incorporating the various OTP initiatives underway
- Kickoff and coordinate technical development for the TriMet OTP SUM project

Objectives/Expected Outcomes:

Everyone leaves workshop with common understanding of:

- What their development objectives are
- What they need to do to meet TriMet technical requirements
- How their tasks interface with the other components of the project and broader OTP ecosystem

WORKSHOP ATTENDEES



| Participating Agencies and Companies | Out-of-Town | Local | Remote | Total |
|--------------------------------------|-------------|-------|--------|-------|
|--------------------------------------|-------------|-------|--------|-------|

| | | | | |
|--|--------------------|--------------|---------------|--------------|
| AC Transit Piush Dahal Manjit Sooch | | | 2 | 2 |
| Cambridge Systematics Paul Sorenson David Benoff Simon Jacobs | 3 | | | 3 |
| City of Portland & PBOT Steve Hoyt-McBeth Anne Hill | | 1 | 1 | 2 |
| Conveyal David Emory Kate Chanba Landon Reed | 3 | | | 3 |
| CUTR Sean Barbeau | | | 1 | 1 |
| IBI Ritesh Warade Jon Campbell | 2 | | | 2 |
| LA Metro Doug Anderson Mike Gibbs | 2 | | | 2 |
| Lyft Debs Schrimmer | | 1 | | 1 |
| Participating Agencies and Companies | Out-of-Town | Local | Remote | Total |
| Metro Robert Kirkman Tom Rippetoe Caleb Winter | | 3 | | 3 |
| moovel Regina Clewlow Courtney Longfellow Ali Waters Michal Nakashimada Scott Wilson Erin Collins | 1 | 5 | | 6 |
| ODOT Mathew Barnes | 1 | | | 1 |
| Ride Connection | | 1 | | 1 |

| | | | | |
|---|-----------|-----------|----------|-----------|
| Kevin Chambers | | | | |
| RTD Jonathan Wade Brett McDavid Judy Wang | 3 | | | 3 |
| Trillium Transit Aaron Antrim Thomas Craig | | 2 | | 2 |
| Uber Jon Isaacs Andrew Freeman | | 2 | | 2 |
| VTA Marshall Ballard | | | 1 | 1 |
| Totals: 20 | 17 | 15 | 8 | 38 |
| TriMet | | 13 | 2 | 15 |

Note:

Out-of-Town = person(s) traveled from outside Portland Metro Area to attend in person

Local = person(s) from Portland Metro Area/vicinity and attended in person

Remote = person(s) attended remotely

WORKSHOP AGENDA

| | |
|---|--|
| <p>AGENDA TriMet Mobility On Demand Sandbox Workshop OpenTripPlanner (OTP) and Pelias Enhancements January 18-19, 2017</p> | <p>LOCATION TriMet Administrative Office Harrison Square Building Saltzman Conference Room (Located on P2, Parking Level) 1800 SW 1st Avenue Portland, OR 97201</p> |
|---|--|

| DAY 1 WEDNESDAY JANUARY 18th | | |
|------------------------------|---|--|
| 8:30am | COFFEE/BAGELS | |
| 9:00am | Nazy Sobhi, FTA | Opening Remarks MOD Sandbox Grant Award |
| 9:05am | Gustave Cordahi, Booz Allen Hamilton | Independent Evaluation of MOD Sandbox Projects |
| 9:10am | Participant Introductions | Participant Introductions Briefly state: name, organization, position, role/interest in project |
| 9:30am | Ritesh Warade, IBI Group | OpenTripPlanner (OTP) Initiatives Overview <ul style="list-style-type: none"> - The OTP initiatives underway - How do these initiatives fit together? - What do we want to accomplish over the next two days? |
| 9:45am | Bibiana McHugh, TriMet | OTP Shared-Use Mobility (SUM) Project Overview <ul style="list-style-type: none"> - Workshop Overview - OTP SUM Project Scope and Objectives |
| 10:00am | Group Discussion | Vision for OTP (SUM, FLEX, and other initiatives) in the MOD SANDBOX Context - Part 1 |
| 11:00am | BREAK (breaks will be 20 minutes to allow people to check messages, etc.) | |
| 11:20am | Group Discussion | Vision for OTP (SUM, FLEX, and other initiatives) in the MOD SANDBOX Context - Part 2 |
| 12:20pm | Neil McFarlane, TriMet General Manager Metro Councilor Bob Stacey | Welcoming Remarks |
| 12:30pm | LUNCH (provided) | |

| DAY 1 WEDNESDAY JANUARY 18th | | |
|------------------------------|--|--|
| 12:45pm | Ritesh Warade, IBI Group | Recap and Role of Next Sessions: - Brief recap and additional observations on the prior sessions - Overview of afternoon sessions and their goals |
| 1:00pm | David Emory and Kate Chanda, Conveyal | Technical Presentation - Conveyal |
| 1:30pm | Thomas Craig, Trillium Solutions | Technical Presentation - Trillium and Cambridge Systematics |
| 2:00pm | Diana Shkolnikov, Mapzen | Technical Presentation - Mapzen |
| 2:30pm | Regina Clewlow, moovel Group | Technical Presentation - moovel |
| 2:40pm | BREAK | |
| 3:00pm | Bibiana McHugh, TriMet | Technical Presentation and Discussion - TriMet |
| 4:00pm | Group Discussion | Planning for Technical Breakout Sessions (unconference sign-up style) |
| 4:45pm | Bibiana McHugh | Wrap-Up - What's happening tomorrow? - Other next steps |
| 6:00pm | DINNER Rock Bottom Brewery 206 SW Morrison St Portland, OR 97204 | |

Note: Afternoon technical presentations will include main project components and preliminary technical issues.

| DAY 2 THURSDAY JANUARY 19th | | | |
|-----------------------------|---|----------------------|--|
| 8:30am | COFFEE/BAGELS | | |
| 9:00am | Technical Breakout Sessions <ul style="list-style-type: none"> • Routing Engine: Flex+SUM+RT+Accessibility • UI/UX: Journeys and Scenarios; Testing Process • Geocoding: Integration with OTP/Trimet Website | | |
| 10:30am | BREAK | | |
| 10:45am | Technical Breakout Sessions <ul style="list-style-type: none"> • OTP Infrastructure & Testing: Build Management Decisions and Test Procedures/Change Management/Component Integration • UI/UX: Flex+SUM+RT+Accessibility | | |
| 12:30pm | LUNCH (provided) | | |
| 1:00pm | Technical Breakout Sessions <ul style="list-style-type: none"> • OTP Web Services/Middle Tier • UI/UX: Integration with Trimet • Fares/Incentives in OTP | | |
| 2:30pm | <table border="1"> <tr> <td>Breakout Group Leads</td> <td> Breakout Groups Report-Back <ul style="list-style-type: none"> - Summary of what was covered - Roadmap for next steps </td> </tr> </table> | Breakout Group Leads | Breakout Groups Report-Back <ul style="list-style-type: none"> - Summary of what was covered - Roadmap for next steps |
| Breakout Group Leads | Breakout Groups Report-Back <ul style="list-style-type: none"> - Summary of what was covered - Roadmap for next steps | | |
| 3:45pm | <table border="1"> <tr> <td>Group Discussions</td> <td> Business Strategies <ul style="list-style-type: none"> - Identify Key Barriers for Agency Adoption - Brainstorm, Discuss, and Prioritize Options and Solutions - Business Strategy (marketing, procurement processes, etc.) - Open Source Community Building Strategy </td> </tr> </table> | Group Discussions | Business Strategies <ul style="list-style-type: none"> - Identify Key Barriers for Agency Adoption - Brainstorm, Discuss, and Prioritize Options and Solutions - Business Strategy (marketing, procurement processes, etc.) - Open Source Community Building Strategy |
| Group Discussions | Business Strategies <ul style="list-style-type: none"> - Identify Key Barriers for Agency Adoption - Brainstorm, Discuss, and Prioritize Options and Solutions - Business Strategy (marketing, procurement processes, etc.) - Open Source Community Building Strategy | | |
| 4:45pm | Wrap-Up <ul style="list-style-type: none"> - Key Observations - Summary and Conclusions - Identify Next Steps - Meeting and Conference Call Schedules - Draft Project's Scope, Plan and Schedule | | |
| 6:00pm | FINISH | | |

SUMMARY DAY 1



The first day of the workshop was focused on laying the groundwork for the OTP SUM project. This began with project introductions, with FTA describing the MOD Sandbox grant program, Booz Allen Hamilton overviewing the program evaluation process, and Trimet and IBI outlining the goals and objectives for the workshop. The remainder of the day was spent laying the groundwork for the OTP SUM project.

OTP Vision

During the morning session, discussion focused on developing a long range vision for OTP as an open source initiative. Through this discussion, we were able to identify what OTP currently does really well for agencies, where additional focus and development is necessary, as well as determine technical priorities for tackling those issues. Key priorities that emerged from this discussion include:

- The integration of SUM and FLEX services into the OTP trip planning engine
- Future integration (in some form) of transit and SUM fare and payment information
- Improved open source alternative for OTP dependencies such as map tiles, geocoders

Technical Presentations

The afternoon sessions featured members of the OTP SUM development team introducing their proposed technical approach and functional requirements for the

project. Conveyal introduced their proposed React/Redux architecture to update OTP to a modern, responsive design, as well as mock-ups of the UI elements that will help OTP users integrate SUM modes into their transit trips. Mapzen provided an overview of Pelias, their open source geocoding engine, describing opportunities for customization when implementing a local instance for a project such as OTP SUM. In addition to the two primary OTP SUM development teams, moovel introduced their fare payment applications, describing how they have integrated SUM modes into their mobile applications. Also, Trillium Transit and Cambridge Systematics introduced their development approach for the VTrans MOD project to integrate GTFS-Flex into OTP.

For additional detail on the technical presentations, please consult the following appendices.

Appendix 3: Conveyal_TriMet_MOD_Kickoff

Appendix 4: Trillium - TriMet workshop flex presentation

Appendix 5: MapZen - MOD Kick-off 2017

Appendix 6: moovel TriMet OTP Kickoff

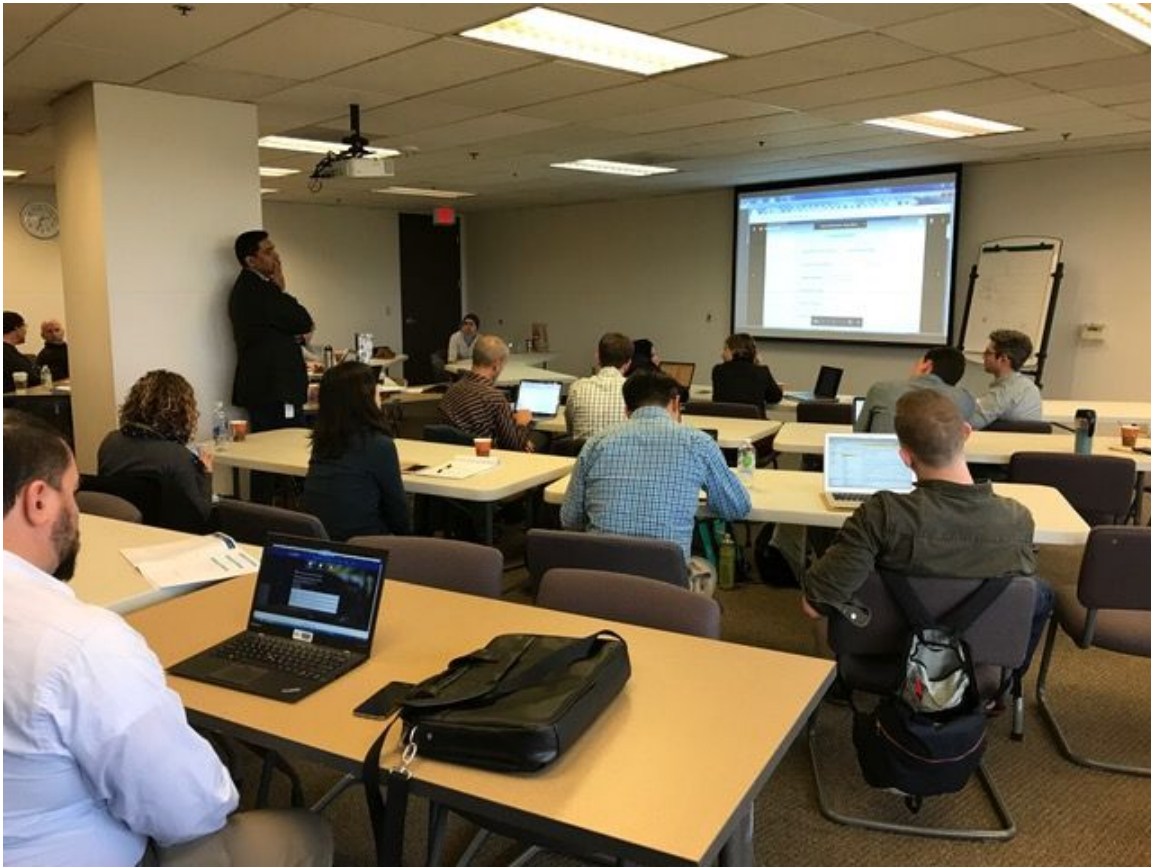
TriMet Technical Overview and Requirements

Bibiana McHugh provided a technical overview of Trimet's objectives and functional requirements for their implementation of the new OTP SUM front and back end. This included discussion of the functionality of existing geocoding solution used by TriMet (SOLR), which will need to be matched/surpassed by Pelias as part of OTP SUM. In addition, topics for future discussion emerged including the desired development process/approach for OTP/SUM and Pelias, defining a roadmap for integration of OTP SUM with the existing website, and ADA and other accessibility requirements on system design. This session closed with an overview of the management and collaboration tools and communication channels that will be used by the OTP SUM team.

For additional information, please see Appendix 7: TriMet Trip Planning Roadmap and Requirements

The day concluded with collaborative planning of the next day's technical work sessions.

SUMMARY DAY 2



TECHNICAL BREAKOUT SESSIONS

The workshop's second day consisted primarily of working sessions to tackle the technical questions and issues identified during the previous day's discussions, and leverage having so many OTP stakeholders and developers together to jump start design and development of the components of OTP SUM. Over three one and a half hour sessions, eight different working groups met. Each of these is described below.

1. Fares/Incentives in OTP

The Fares and Incentives working session focused on how to best incorporate fares into OpenTripPlanner, as well as the challenges of fitting fares into a data model such as GTFS. The discussion drew from the experiences of agencies such as LA Metro, who built a separate fare database and logic to handle fares during trip planning. In addition, they discussed the issue of how to handle fares in the OTP Graph. Solving for minimum fares would often yield cheap, but impractical trips. Instead, sorting by fare after identifying trips that best fit user preferences would be a way to better reflect how users incorporate consideration of fare into their trip planning process. Finally, the group

touched on integration of fare payment and potential for incentives within OTP, looking at moovel's deep link integration of ride-hail trips with Lyft as an example.

2. Geocoding: Integration with OTP/TriMet Website

The geocoding session began with participants describing their agencies' experience with current geocoding solutions, emphasizing challenges such as complexity (incorporating transit-specific locations and price [high API license fees]). This pivoted to a discussion of requirements for Pelias, including address interpolation, location bias, and accounting for spelling errors in user input. The session concluded with a discussion of what needs to happen to update Pelias from its current state to meet TriMet's requirements, with a focus of incorporating custom data into the Pelias ElasticSearch data store.

3. OTP Infrastructure & Testing

a. Build Management Decisions

During this session, representatives from TriMet and Conveyal discussed the development process and build management tools for OTP SUM. TriMet and Conveyal both outlined how they each currently approach development and builds. In addition, Mapzen gave an overview of their development processes for the Pelias geocoder to inform TriMet's local install. This session ensured that the development teams and TriMet will be working in the same environments and frameworks during the OTP development process.

b. Test Procedures, Change Management, and Component Integration

This session established a testing and change management strategy for the OTP SUM project. TriMet will establish build, stage, and production environments, which will be consistent across project components. Integration testing and acceptance tests were discussed. A variety of acceptance testing approaches were described, and future discussion among the project team will be necessary to finalize the OTP testing plan. Security considerations were also discussed including sensitive information (not an issue during current phase, but becomes a challenge with payment integration) and system stability concerns (rate limiting, load balancing, etc.).

4. OTP Web Services/Middle Tier

This session focused on whether UI's should interface directly with the server-side OTP routing engine or rely on a middle tier service to intercept API calls. Considerations for a middle tier included the ability to plan multi-leg (i.e. trip chaining) trips as a "stitched itinerary" as well as flexibility for what trip planning engine powers a particular UI instance. Several agencies shared their experiences using trip planner middle layers including RTD, whose middle tier filtered landmarks and other transit specific locations, and LA, whose middle tier ran fare calculations for planned trips. Key to this discussion was what aspects of a trip should be handled in the back end (common to all OTP

implementations) and what should be handled by a middle tier (configured for each implementation).

5. Routing Engine: Flex+SUM+RT+Accessibility

This session focused on how to best integrate SUM, flex-service, real-time, and accessibility data into the OTP back-end routing engine. For accessibility, OTP can already leverage OpenStreetMap tags, which are being updated as part of this project to better map pedestrian and wheelchair accessibility. For SUM modes, the focus was on reaching consensus on what information is necessary to help OTP users plan a trip that includes a SUM mode. While additional discussion is needed to finalize SUM data requirements, key considerations were identified, including availability and time/space granularity of data. Whatever format the SUM data is provided in, it needs to be able to translate lat/long and time into availability and price information for that mode. Further discussion focused on user preferences for SUM trips, as well as the different types of trip planning scenarios (ie- real time data for planning a trip right now, or aggregate data for exploratory, narrative trip planning).

6. UI/UX: Flex+SUM+RT+Accessibility

This session focused on incorporating SUM and flex services into the OTP user experience. One particular challenge for both SUM and flex trips is communicating the complexity/details of these services (i.e.- the need to book a trip, or be on a certain side of the street to flag a stop), without overwhelming the user with text, particularly if the user is accessing OTP on a mobile device. This included discussion of what information about SUM trips (availability, wait times, estimated price) should be communicated to users, as well as graphic design considerations for how to display SUM or flex portions of trips. Next steps include: further developing icon sets and message sets for SUM and flex trips, and figuring out narrative directions vs. map visualization for SUM and flex trips.

7. UI/UX: Integration with TriMet

This session focused on how to integrate the new React/Redux OTP components into the existing TriMet website. Currently the homepage functions as a three task “app” with the trip planner, transit tracker, and service alerts. With the implementation of the new OTP, keeping the tool as integrated as possible is a priority. In addition, the interaction between OTP and Pelias was discussed, with OTP components passing user search inputs to TriMet’s Pelias instance for geocoding.

8. UI/UX: Journeys and Scenarios; Testing Process

The final UI/UX work session focused on identifying user “journeys and scenarios.” By mapping out how users will interact with OTP SUM, from first awareness to exiting the trip planner, the design team can tailor the application’s UI to best meet user needs and expectations. A major consideration during this discussion was how to get people to use something they’re less familiar with. The UI/UX must emphasize the “added value” of the updated OTP SUM over other trip planning tools. In addition, different users will have different preferences about taking multimodal trips. Getting the default setting for these preferences correct will be critical, as well as finding a balance between making it

easy for users to change their preference settings and not overwhelming users with choices/options/complexity. This session also touched on an initial discussion of the user testing that will occur during this project's second year.

9. Technical Work Session Debrief

Following the technical work sessions, representatives from each session gave a brief (5-10 minute) report back to the full group of workshop attendees. This gave everyone a level of familiarity with what was accomplished over the course of the day. In addition, teams were able to recognize areas of overlap or interaction between project elements that would benefit from additional discussion or meetings.

BUSINESS STRATEGIES DISCUSSION

To close out the kickoff workshop, all attendees reconvened for a discussion and brainstorming session about business strategies for OTP. This included strategies for both the broad OTP initiative, as well as individual agency implementations of OTP. Agencies in attendance explained their primary reasons for implementing OTP, including:

- The desire to not rely on third-parties for core agency services,
- Critical functionality is missing from 'out of the box' trip planning tools
- High cost of other (proprietary) trip planning options.

In addition, the group identified key barriers to further adoption of OTP by more agencies, which include:

- Procurement processes not designed to handle open source projects
- Lack of resources to do OTP build and integration
- Perception of IT as secondary service

Considering these two perspectives, the group discussed what OTP stakeholders could do to encourage adoption and improve the experience for agencies, with the end goal of making OTP the go-to option for agencies looking to implement a first trip planner or upgrade their trip planning capabilities. Peer exchanges and skill shares between agencies could help increase familiarity with and reduce uncertainty/anxiety over open source solutions such as OTP. In addition, the possibility of an OTP consortium of agencies, consultants, and developers, along with a clear visual branding of OTP could help spread familiarity of OTP amongst agency executives. An OTP consortium could also provide a structure for oversight and coordination across various OTP initiatives.

The experience of other transit tech consortia was discussed, raising the questions:

- Are we headed towards consortium overload? With the proliferation of transit technology/data consortia, will stakeholders have enough bandwidth/capacity to meet the goals of these consortia?
- Are resources stretched to thin/diluted because of this?
- A lot of these groups are asking:
 - How do we fund this?
 - What value are we bringing to members?
 - How do we make it a durable initiative?

The current collaboration among the MOD Sandbox OTP projects, as well as the other active OTP initiatives is a critical first step to build an active network of stakeholders contributing to OTP. This will lay the groundwork for any future more formal OTP organization.

CONCLUSIONS

Key Observations

- There is momentum behind making significant changes and improvements to OTP in the coming few years, and especially as a result of the MOD grants
- The various parties - agencies, consultants, and vendors - involved in the various OTP improvement efforts need to, and are coordinating their effort
- TriMet, as the convenor of this workshop, can and is willing to help coordinate the various OTP improvement efforts

Next Steps

- Set up regular coordination calls/meetings for the various streams of activity for the TriMet OTP SUM project
- Set up coordination calls/meeting between the various other OTP improvement projects

APPENDIX C

TriMet MOD Grant Task 3 Milestone 1 Documentation

This repository | Search

Pull requests Issues Gist



opentripplanner / otp-react-redux

Watch 9 Star 5 Fork 2

Code Issues 2 Pull requests 1 Projects 0 Wiki Pulse Graphs

A library for writing modern OpenTripPlanner-compatible multimodal journey planning applications using React and Redux.

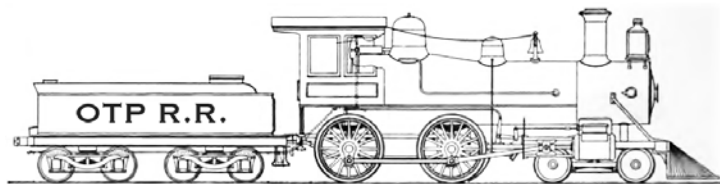
93 commits 8 branches 9 releases 3 contributors MIT

Branch: dev New pull request Create new file Upload files Find file Clone or download

| | | |
|---|--|----------------------------------|
| evansiroky committed on GitHub Merge pull request #24 from opentripplanner/form-control ... | | Latest commit 7049707 5 days ago |
| _tests_ | feat(form): add config options for form control | 6 days ago |
| configurations | fix(build): add dist folder to npm | a month ago |
| lib | docs(form): add code comments to add clarity | 5 days ago |
| .gitignore | fix(npm): include built css file in npm package | 18 days ago |
| .npmignore | refactor(npm): include dist folder when publishing | 18 days ago |
| .travis.yml | fix(npm): include built css file in npm package | 18 days ago |
| LICENSE | Initial commit | 2 months ago |
| README.md | fix(build): Update docs for new 0.1.x release | a month ago |
| example-config.yml | major component build out | 2 months ago |
| example.css | refactor(css): move some css around | 13 days ago |
| example.js | chore(example): re-comment out initialQuery | 14 days ago |
| index.html | refactor(css): move some css around | 13 days ago |
| otpr.png | Initial commit | 2 months ago |
| package.json | feat(form): add config options for form control | 6 days ago |
| yarn.lock | feat(form): add config options for form control | 6 days ago |

README.md

otp-react-redux



A library for writing modern [OpenTripPlanner](#)-compatible multimodal journey planning applications using [React](#) and [Redux](#).

Running the Example

A simple example of an OTP-RR application is included in the repository.

To run, first clone the repo and install [yarn](#) if needed.

Copy `example-config.yml` to `config.yml`. Update `config.yml` with your Mapzen API key, and optionally, the OTP endpoint and initial map origin. (The default values are for a Conveyal test server for Portland, OR.)

Install the dependencies and start a test instance using yarn:

```
yarn install  
yarn start
```

Library Documentation

Coming Soon



This repository | Search

Pull requests Issues Gist



conveyal / trimet-mod-otp

Watch 6

Star 0

Fork 0

Code

Issues 0

Pull requests 0

Projects 0

Wiki

Pulse

Graphs

No description, website, or topics provided.

4 commits

1 branch

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

| | | |
|--|---|-----------------------------------|
| | demory Add webpack-dev-server to package.json | Latest commit 95410f4 25 days ago |
| | lib TriMet branding | 26 days ago |
| | .gitignore Initial commit | a month ago |
| | README.md Update README.md | a month ago |
| | package.json Add webpack-dev-server to package.json | 25 days ago |
| | webpack.config.js Initial commit | a month ago |
| | yarn.lock Add webpack-dev-server to package.json | 25 days ago |

README.md

trimet-mod-otp

TriMet-specific implementation of the otp-react-redux library, managed using yarn and webpack 2.

Getting Started

Install [yarn](#), [webpack](#) (version 2.x), and [webpack-dev-server](#) (for testing).

Copy `lib/config-template.yml` to `lib/config.yml` and add any needed API keys, etc.

To start a demo server using webpack-dev-server:

```
yarn install
webpack-dev-server
```





TriMet Trip Planner

7 Screens
29 Conversations



Landon
R.



Kate C.



Dave D.



David E.



Evan E.



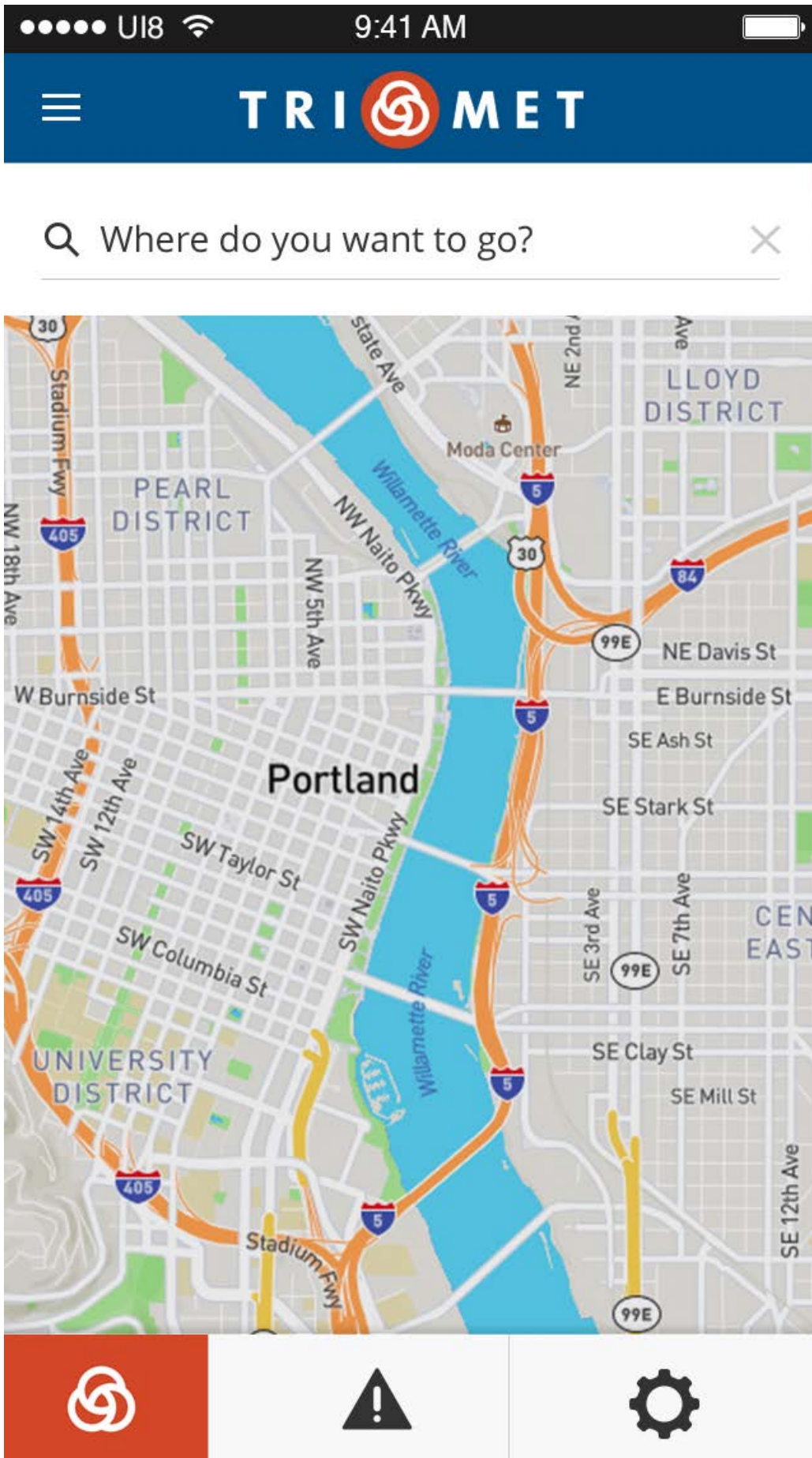
Madeline S.

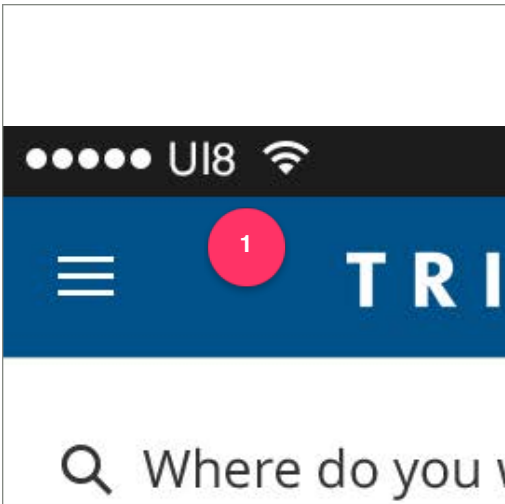


bibiana
m.



Jonathan H.





Evan on Mar 22 at 5:21pm

What would happen upon pressing this button?



Kate Chanba on Mar 22 at 5:25pm

@Evan depending on what TriMet wants, we may retrieve menu options from their mobile site. If not, we will remove this menu.



Dave on Apr 3 at 9:27pm

@Evan @KateChanba Hi Evan and Kate! A couple thoughts on this: 1. Our research with riders from a couple years back showed pretty bad understanding of the hamburger menu across rider types. We'd suggest using the word MENU to be 100% clear for all users. 2. We are considering having this app become the new trimet.org homepage, if we can either squeeze in all the features our riders need or add them ourselves later -- versus linking off to this app from the homepage. (This is, in fact, the stuff people are coming to our site for, so why not put it front and center? thinking other agencies may want to go this route as well.) IF we do integrate the OTP app onto the trimet.org homepage this menu would serve two purposes -- it could contain universal features related to the map/trip planner (not trip-specific settings), as well as links to the "other" site content and top user tasks/keywords, such as what is on the trimet.org homepage/nav right now: <http://trimet.org>. For example, it might include:

- Plan a trip
- Get arrival times for your bus or train
- Check service alerts
- Get updates by email

-
- Routes & Schedules
 - Fares
 - Rider's Guide
 - Support
 - About



Kate Chanba on Apr 5 at 9:06pm

@Dave thanks for the comments, we'll be able to accommodate these top level needs for the home page through functionality.
- Plan a trip

- Get arrival times for your bus or train
- Check service alerts

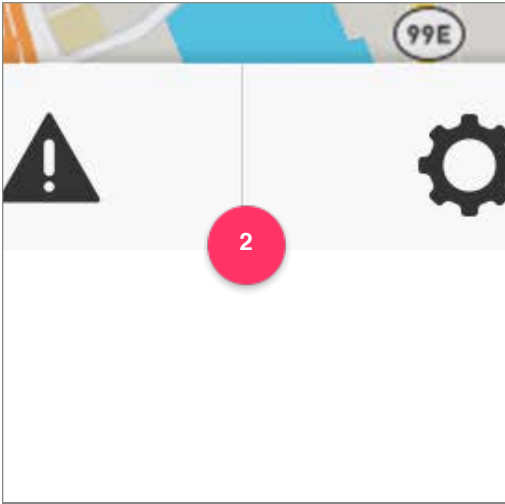
As far as these below, we can link out to separate pages that you all design and have content for:

- Get updates by email
- Routes & Schedules
- Fares
- Rider's Guide
- Support
- About

In this version of the design, the TriMet logo serves as access to these menu options. Working on some redesigns re: comments on the bottom drawer.

Thanks!
Kate

Comments:



Kate Chanba on Mar 24 at 5:16pm

@Evan @LandonReed @DavidEmory based on a conversation with @Dave Whipple we talked about the ability to have a bottom drawer on mobile responsive safari – is it possible to keep it fixed as in this design.



Dave on Apr 3 at 9:06pm

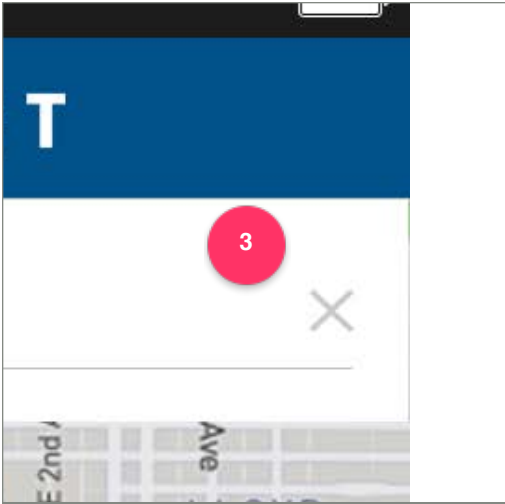
@Evan @LandonReed @DavidEmory @KateChanba I'm also wondering how this would affect things like scrolling results panes for trip itineraries. The bottom drawer is taking up valuable real estate from the map, and in generally I'd like to make sure we are confident that these three buttons are important enough to take up that space. I'd really like to make this map as usable as possible for people, and the more space the better! @KateChanba: question: where does the rosette button take you?



Kate Chanba on Apr 5 at 9:13pm

@Evan @LandonReed @DavidEmory @Dave Thanks Dave! We are going to propose a few more navigation options to compare against the bottom drawer. Real estate is definitely a concern. The rosette in this design would take you to a list of TriMet navigate-away menu options (things like Fares, Route Guide, About etc.)

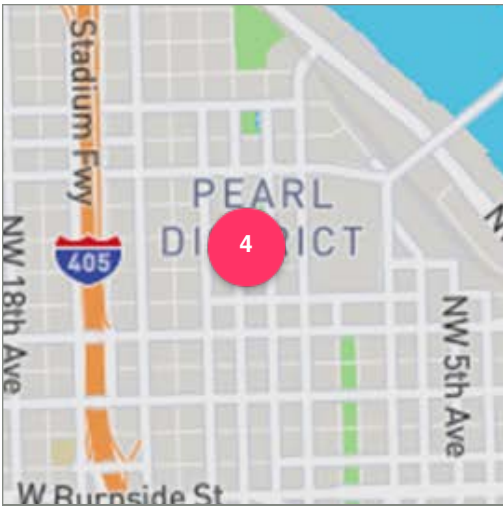
Comments:



Kate Chanba on Mar 24 at 5:19pm

Favorite and clear form. Doesn't necessarily have to be an ellipses.

Comments:



Kate Chanba on Mar 24 at 5:24pm

@DavidEmory @LandonReed @Dave how could we show real time on the map? Dave is interested in showing vehicles actual locations, this may be outside of the grant. Perhaps there are other options.



Kate Chanba on Mar 27 at 6:07pm

@DavidEmory @LandonReed @Dave we will overlay stops on the map and we could have RT info appear in interactive components.



Dave on Apr 3 at 9:48pm

@DavidEmory @LandonReed @KateChanba Hi team! For some background on this, one of the new features we (TriMet marketing) wishes to integrate on the trimet.org map is real-time vehicle locations.

Check out this rough prototype we created a couple years back:

<http://beta.trimet.org/map/#tracker/stop/3635> (login admin/TriMet@1969).

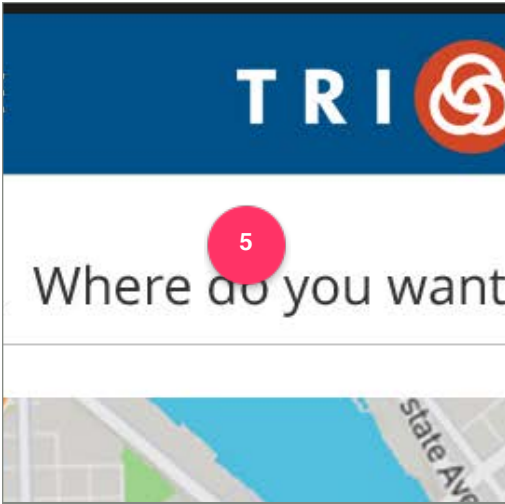
This of course may be out of scope for the OTP project, but the idea came about because we know our customers don't think of "trip planning" as simply getting an itinerary from point A to B. The real-time arrival info (what we call TransitTracker) is another critical part of taking a transit trip, and the map provides a great way to reassure riders that their vehicle is en route (or not!) -- and especially useful for visual learners. Plus, during disruptions, we have found that providing "distance away" information has proven very useful for riders when an accurate arrival countdown is not available. So, showing real-time vehicle locations on the map builds on this concept to help riders make a better informed decision when an arrival countdown is not available, helps confirm whether or not a vehicle is en route, and completes the trip-planning process. (And a minor but important benefit: the "cool factor" will help us compete better with Uber, Lyft and others who prominently show real-time locations on a map!)



Kate Chanba on Apr 5 at 9:10pm

@DavidEmory @LandonReed @Dave Thanks Dave! I just sent an email to you all to discuss the potential of including RT vehicle locations in the Route Viewer.

Comments:



Kate Chanba on Mar 24 at 5:27pm

Coordinate with geocoder



Dave on Apr 3 at 9:31pm

@KateChanba Looks like the new geocoder will be amazing... finally will let riders type in a business or landmark name ala Google. One of our riders' top requests! Like today, I assume it will allow riders to type in a stop ID or stop name as well? I also am wondering -- since this is now an omnibox approach for our UI and we're implying that you can do anything with this one form field -- if this form field shouldn't also accept the names of routes themselves. That is, you could type in 14 and one of the autocomplete results would be bus line 14-Hawthorne. When you tap that result it shows you Line 14 (and ideally the real-time vehicle locations) on the map. (One more easy way to get the info you want, without having to navigate through menus.)

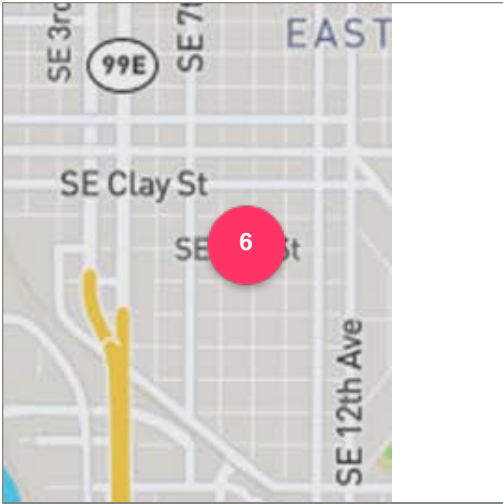


Kate Chanba on Apr 5 at 9:09pm

@Dave regarding viewing entire routes, it seems that on the TriMet site, the Transit Tracker shows a line viewer associated with a stop. We may be able to reach another 'Transit Tracker' modal viewer to explore a line, and will connect with the geocoding team to see if a stop would be an input they would be able to work with.

The route viewer comes later in the project and we can talk about access to that re: my last email to the Conveyal team, you and Bibiana.

Comments:



Dave on Apr 3 at 10:45pm

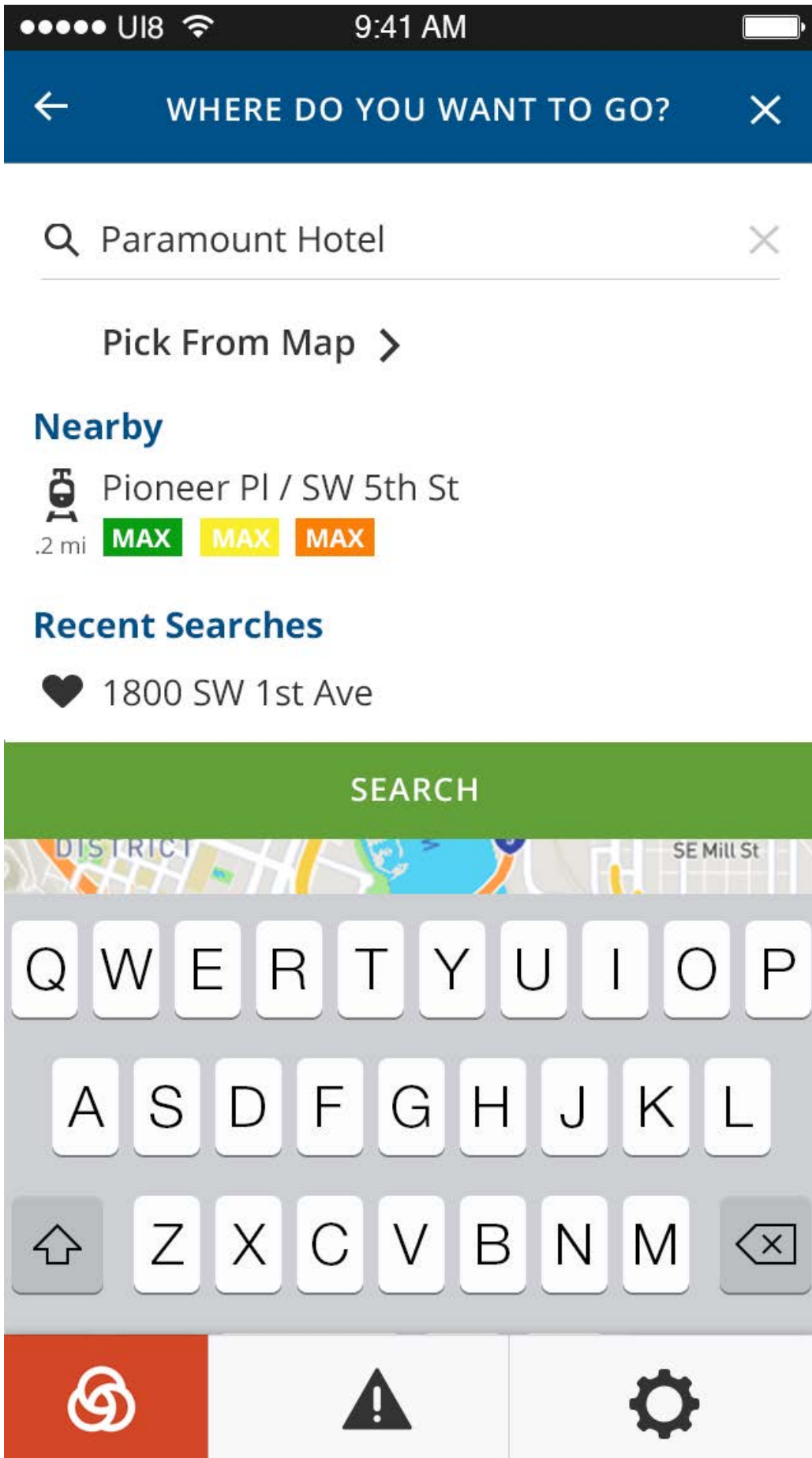
@KateChanba @Evan Where do you propose we put the map layers control? Also, have you considered a button on the map to show your current location on the map? (Trying to accommodate using this app in multiple ways when riders are out and about -- whether via exploring the map or via the form fields.)

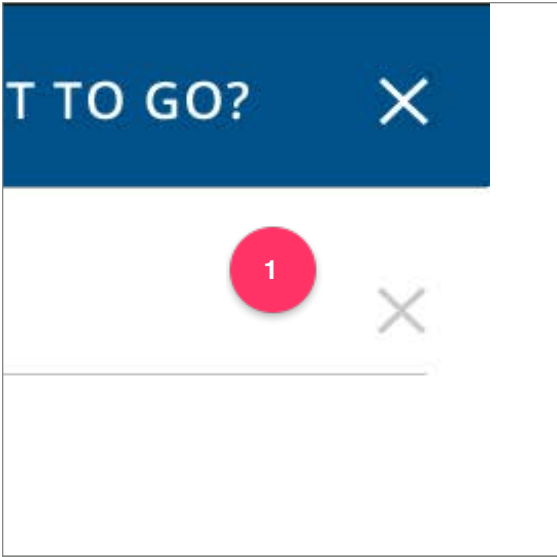


Kate Chanba on Apr 5 at 9:12pm

@Evan @Dave Yes we can definitely show current locations and have zoom in/zoom out functions and layers control (ability to view Transit or Bikeshare layer). I will include these screens in the next revision.

Comments:





Evan on Mar 22 at 5:13pm

What does this button by the side of the search input do? Is it even needed?



Kate Chanba on Mar 22 at 5:14pm

@Evan This will have a dropdown of menu options, including favoriting a location.



Evan on Mar 22 at 5:20pm

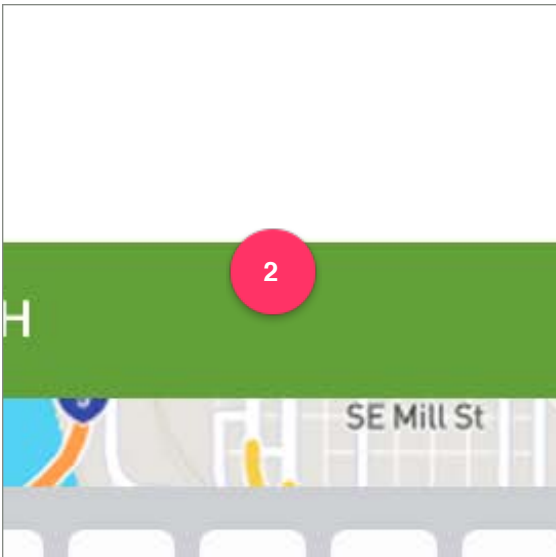
What other options would there be in the dropdown?



Kate Chanba on Mar 22 at 5:30pm

@Evan perhaps similar to the ones we're needing to implement for CFAZ minus CarFreeNearMe

Comments:



Evan on Mar 22 at 5:16pm

Is this search button needed? Right now we simply have requests happen automatically once a user has entered in an address and then another request happens when the form changes.



Evan on Mar 22 at 5:16pm

Is this search button needed? Right now we simply have requests happen automatically once a user has entered in an address and then another request happens when the form changes.



Kate Chanba on Mar 22 at 5:25pm

@Evan That's something we've debated with CFAZ. I'm open to the idea of removing it or keeping it, maybe pin for user testing.



Dave on Apr 3 at 9:54pm

@Evan @KateChanba Does this relate to Frank's recent geocoder questions about interpolated addresses not showing up in autocomplete? Love the streamlining opportunity, although there might be some implications for removing this button.



Dave on Apr 3 at 9:55pm

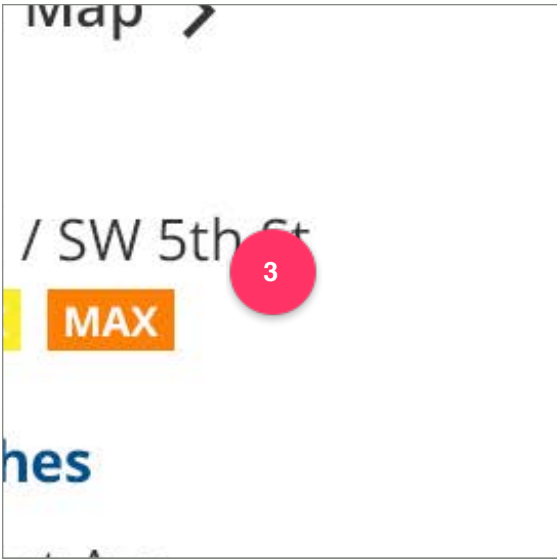
@Evan @KateChanba When we get into design I'd want to look at other button styles that aren't necessarily full width. At first I wasn't sure if this was a heading or a button. :)



Kate Chanba on Apr 5 at 8:10pm

Is 'Search/Select' necessary here?

Comments:



Madeline Steele on Mar 27 at 5:38pm
Love these color-coded MAX labels!

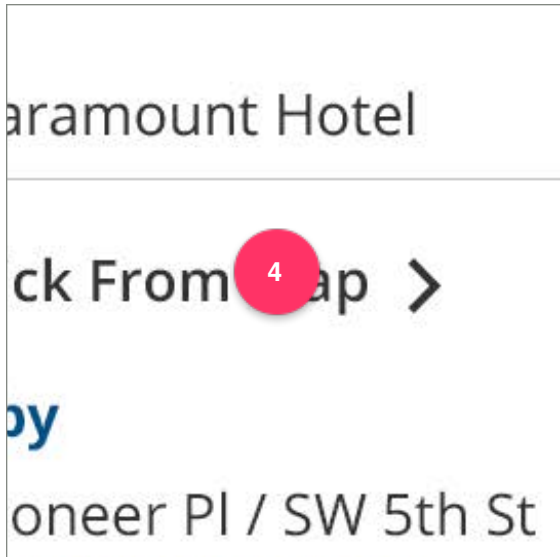


Dave on Apr 3 at 9:52pm
@MadelineSteele I love these too but we'll need to be mindful of colorblind and low-vision users per ADA.



Kate Chanba on Apr 5 at 9:15pm
@MadelineSteele @Dave Good point and will make sure to comply with ADA

Comments:



Dave on Apr 3 at 9:57pm

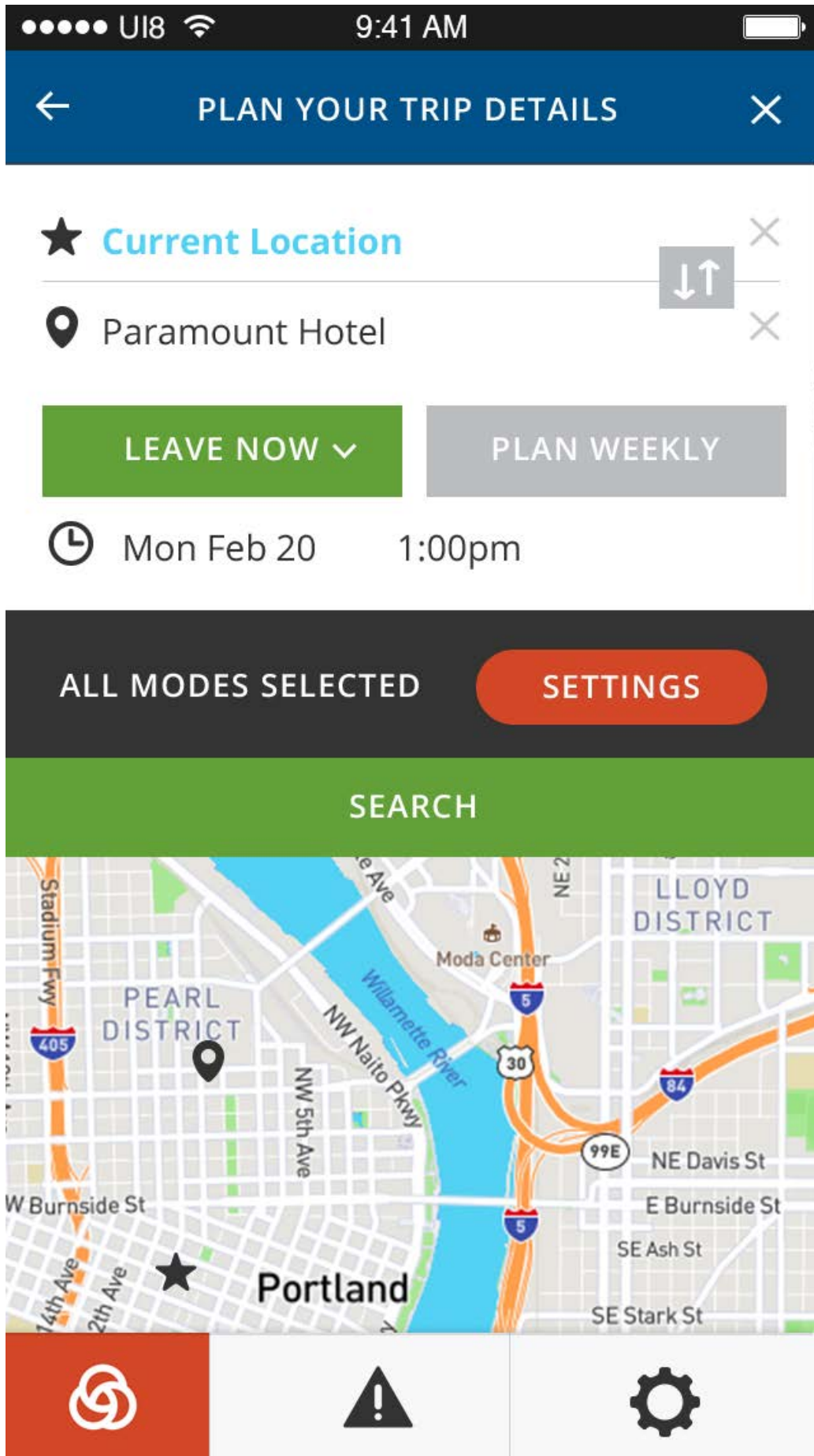
@KateChanba @Evan Hi guys, can you tell me more about what scenario is represented here? If the user just typed in Paramount Hotel, wouldn't this dropdown be showing a list of matching locations, starting with Paramount Hotel at the top?

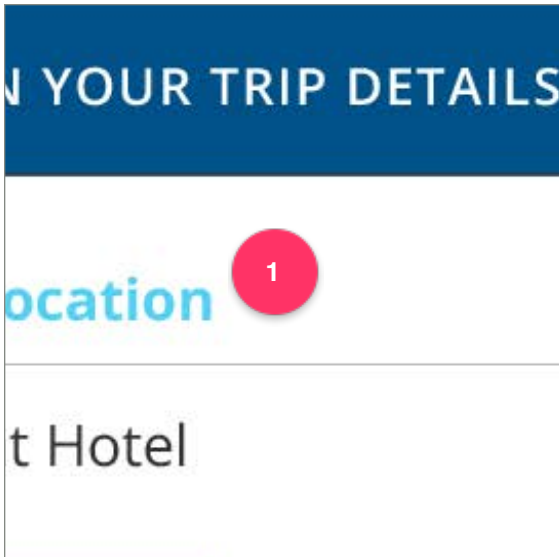


Kate Chanba on Apr 5 at 9:14pm

@Evan @Dave You're right Dave, we are planning to redesign this screen as a 'Search Modal' where it would be showing different screen/navigation items than the Results View.

Comments:





Evan on Mar 22 at 5:15pm

What were your thoughts for when the app would show the from location? In the Modeify App I'm revealing it immediately after the to location is focused. Also, the icons have changed from the magnifying glass to the star and placemaker. Should the app have only the placemaker from the beginning?



Evan on Mar 22 at 5:15pm

What were your thoughts for when the app would show the from location? In the Modeify App I'm revealing it immediately after the to location is focused. Also, the icons have changed from the magnifying glass to the star and placemaker. Should the app have only the placemaker from the beginning?



Kate Chanba on Mar 22 at 5:32pm

@Evan From location appears after to location is designated and on mobile automatically designates from as 'current location.'

Search icon changes to star and placemaker to indicate that locations have been designated.



Dave on Apr 3 at 10:28pm

@Evan @KateChanba Howdy! We've been working through designs for this screen on our end as well. Some ideas to consider:

1. We know lots of TriMet riders just want to explore the map on their own, find a stop, see service nearby,

see where route lines go nearby, etc. And given our new "omnibox" approach -- and Google Maps behavior, which many riders are familiar with -- maybe we should treat the "Where do you want to go?" input as more of a search field. That is, we'd show the selected location on the map (with a Plan a trip link visible in a popup, perhaps), instead of automatically assuming they want to plan a trip to that location. The user would have to take another step to initiate a trip plan. Does that make sense? What are the pros/cons of this?

2. To help inform our decisions about "Current Location" in the UI: When are we proposing asking the user for permission to use their location? My preference (in the case that we replace the entirety of the trimet.org homepage with this app) is immediately upon coming to trimet.org. My team is split on this but I think given that the vast majority of users are coming for service info and the majority of those users have smartphones, it's not out of the question!

D

Dave on Apr 3 at 10:30pm

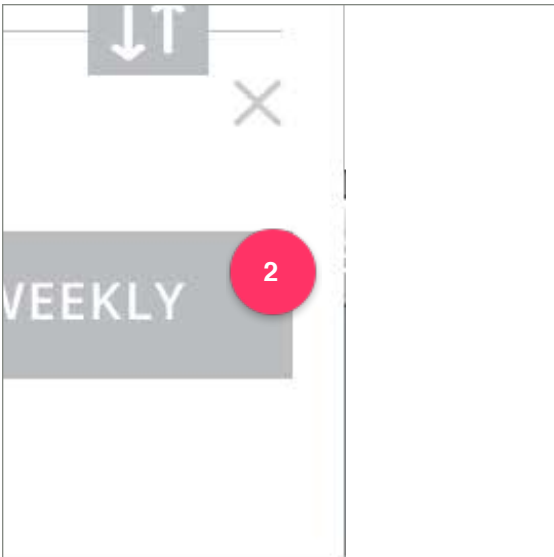
@Evan @KateChanba What are your thoughts on using star as the icon for current location? Is this a convention now? I concur that the placemaker icon is good for the selected location.

D

Dave on Apr 3 at 10:32pm

@Evan @KateChanba If

Comments:



MS

Madeline Steele on Mar 27 at 5:34pm
I take it that "Plan Weekly" takes you to profile based planning? Clear messaging around this really is tricky. I know I'm not really a representative user, but I feel like something like "Flexible time range" might be more intuitive to me.



Kate Chanba on Mar 27 at 6:09pm
@MadelineSteele thanks! Good thinking on this. I like that wording too.

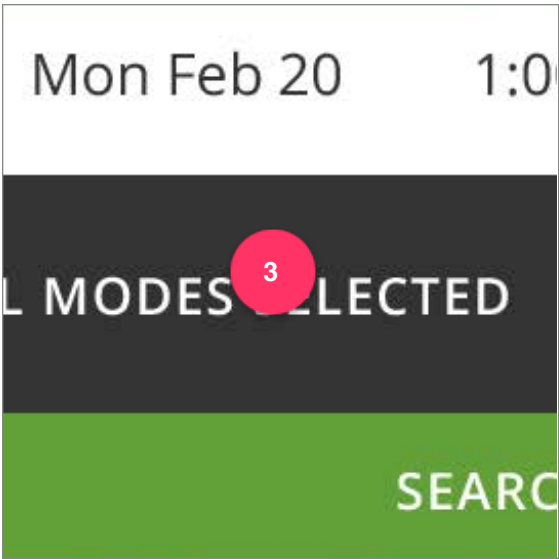
D

Dave on Apr 3 at 10:32pm
@MadelineSteele @KateChanba I agree... We should drum up some more ideas for this piece. Also: since this is a time-related setting, could it be included in the dropdown with Leave Now and the other time-related settings? (This could be a logical grouping, while freeing up UI space and not giving it equal weight with the arguably more important "Leave Now")



Kate Chanba on Apr 5 at 9:18pm
@MadelineSteele @Dave Very good point you guys, the user should not be required to make the differentiation between Itinerary and Profile routing. We should make 'time type' selection all appear in one dropdown and should work on getting the copy right so that users know how it works. 'Set Range' maybe?

Comments:



Dave on Apr 3 at 10:39pm

@KateChanba @Evan We generally steer away from using the word "mode" in our rider tools and communications. It's just not a word people use in conversation, and it sounds very technical and clinical. I would suggest "Services" or something else (we can help with this if needed.)

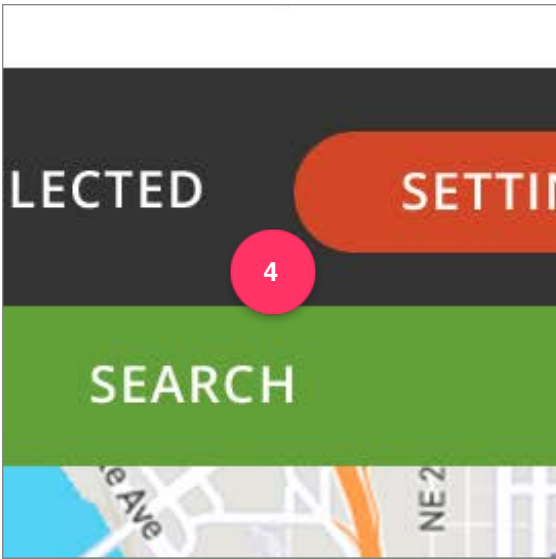
A couple questions: 1. would this space show a list of specific modes, in the case that the user selected some but not all modes? (curious how that would look...) and 2. when we say "all modes" what are we implying exactly? Are we saying we're going to plan a transit+bike+uber trip for you? if so, is it going to be clear to riders?



Kate Chanba on Apr 5 at 9:17pm

@Evan @Dave Thanks for the thoughts and concerns Dave. We are going to do a large redesign screen based on your comments?

Comments:

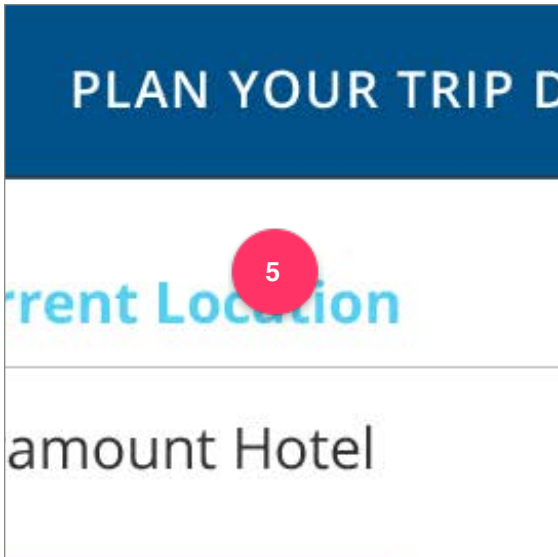


Dave on Apr 3 at 10:40pm
@KateChanba @Evan Similar to the other screen, I suggest we style buttons to look clickable/tappable and distinct from headings.



Kate Chanba on Apr 5 at 9:17pm
@Evan @Dave Good point!

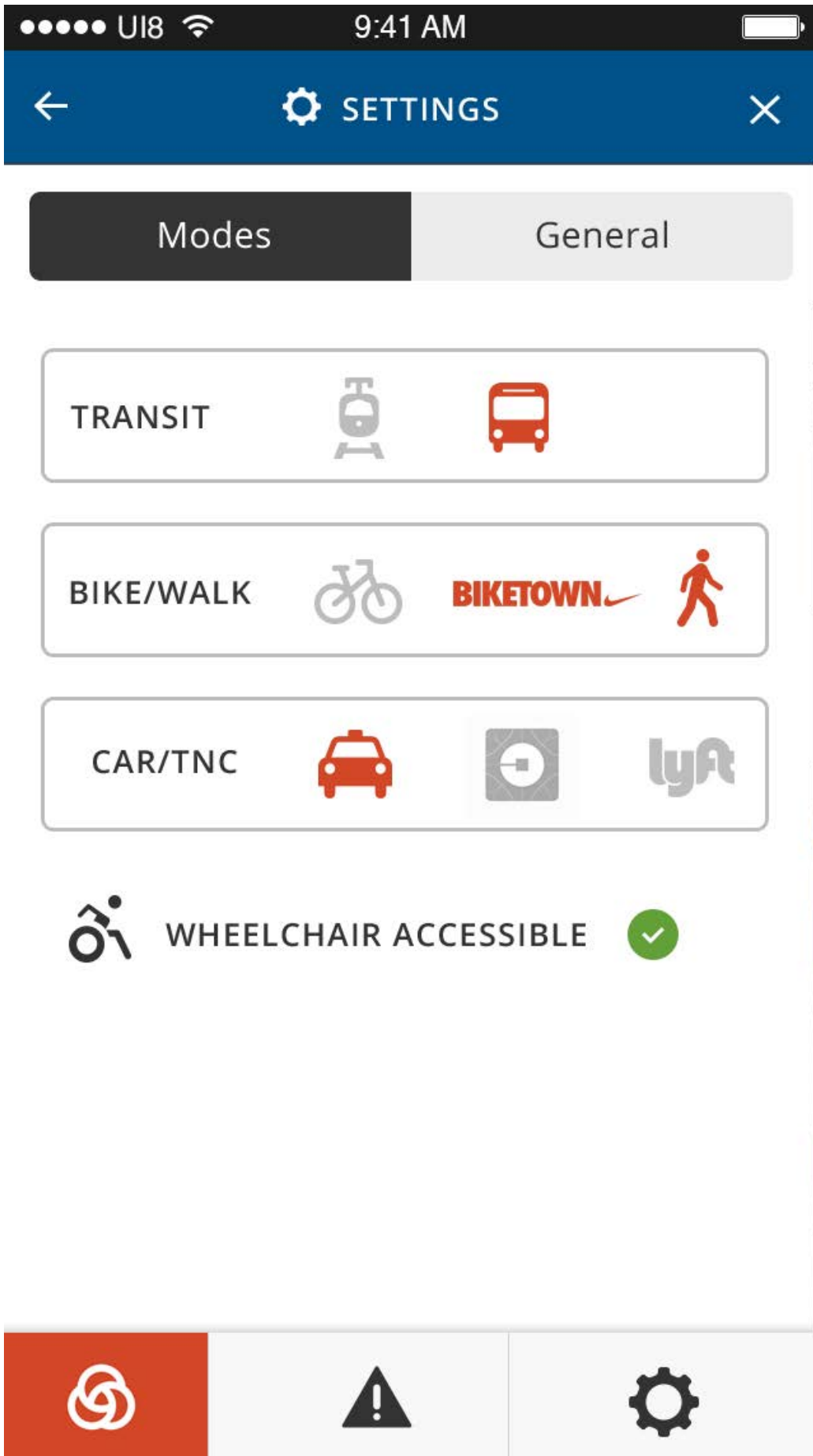
Comments:

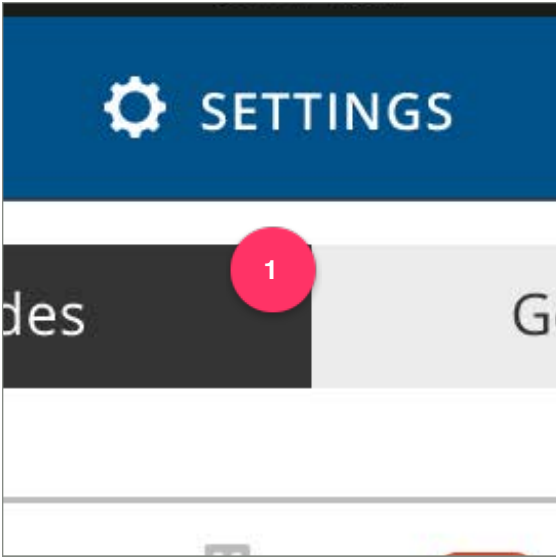


Dave on Apr 3 at 10:51pm

A UI detail request: Let's not let people edit the words "Current Location" as if it were a line of text. It shows up as a line of text but it should disappear when you X out that field or tap to start typing there.

Comments:





Evan on Mar 22 at 5:18pm

This tabular selection is confusing to me. Since the General Tab has no border it appears to me that it is currently being focused, but the content appears to be for the mode selection.



Evan on Mar 22 at 5:18pm

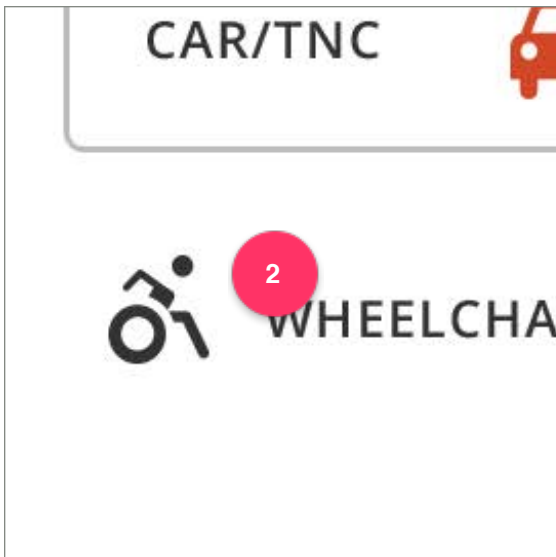
This tabular selection is confusing to me. Since the General Tab has no border it appears to me that it is currently being focused, but the content appears to be for the mode selection.



Kate Chanba on Mar 22 at 5:34pm

@Evan haven't designed the 'General' tab yet, the bar is meant to indicate that we are on the 'Modes' screen. We can try for a different indicator. This screen is reached by selecting 'Settings.'

Comments:



MS

Madeline Steele on Mar 27 at 5:32pm

Hi Kate! This looks fantastic! I'm really excited to see these screens. One comment: for TriMet, I don't think it makes sense have this wheelchair accessibility option exposed. All of our transit stops and vehicles are accessible anyhow (unlike some agencies where certain subway platforms are inaccessible), and although we will be able to favor routes with sidewalks, we won't know whether those sidewalks have curb cuts. I don't want to over promise on this point. On the other hand, wheelchair accessibility might be relevant for SUM vehicles? We can discuss further at the next meeting.

D

Dave on Apr 3 at 10:52pm

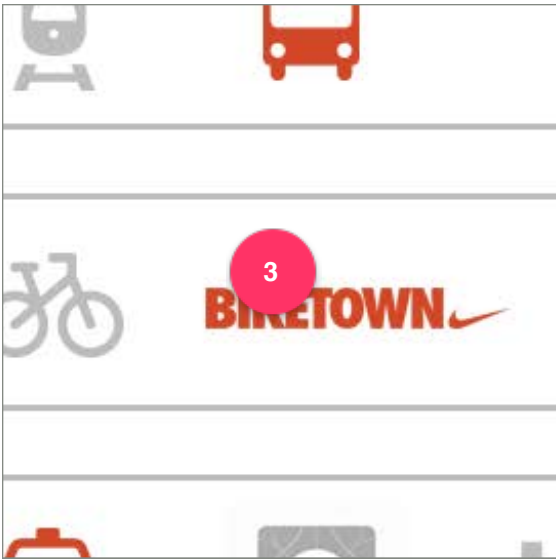
@MadelineSteele Good point, Madeline. I was wondering about that too. Seems like in our case we'd want to reveal accessibility details at the stop level, as an option for those who are interested. Is that what you had in mind?



Kate Chanba on Apr 5 at 9:21pm

@MadelineSteele @Dave we do have a way to include accessibility in profile routing, though perhaps this option should be unchecked by default as it influences results. To a Tri-Met user is accessibility implied? And what about stairs and elevators in stations?

Comments:



Madeline Steele on Mar 27 at 5:37pm
 I really like this highlighted icons mode selection UI! I wonder what the best way to express the difference between personal bikes and bike share might be? In Portland, if it works from a design perspective, the clearest thing might be to just use the BIKETOWN logo here... Wish it was more square shaped!



Kate Chanba on Mar 27 at 6:10pm
 @MadelineSteele Good idea, we will change this to have the BIKETOWN logo



Dave on Apr 3 at 10:54pm
 @MadelineSteele @KateChanba Love it. Good call!! Now to figure out how to differentiate between MAX, WES and Streetcar via icons (they all pretty much look the same... we may need to add text labels.) :)



Kate Chanba on Apr 5 at 9:19pm
 @MadelineSteele @Dave Text labels for sure! Working that out in the next iteration.

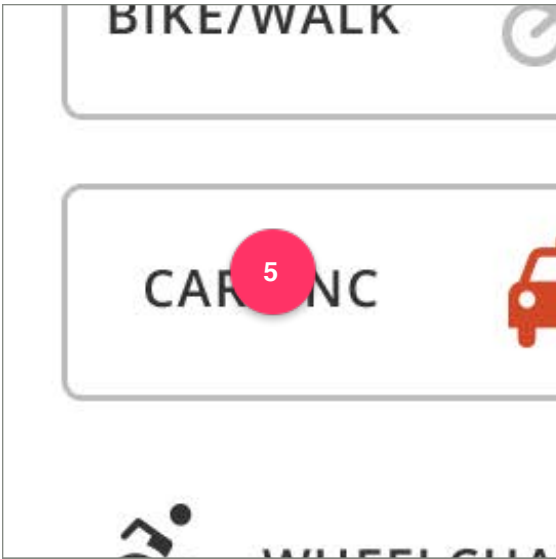
Comments:



Dave on Apr 3 at 10:55pm

Does this represent taxi? Without a text label it could be interpreted to mean you can plan a car trip.

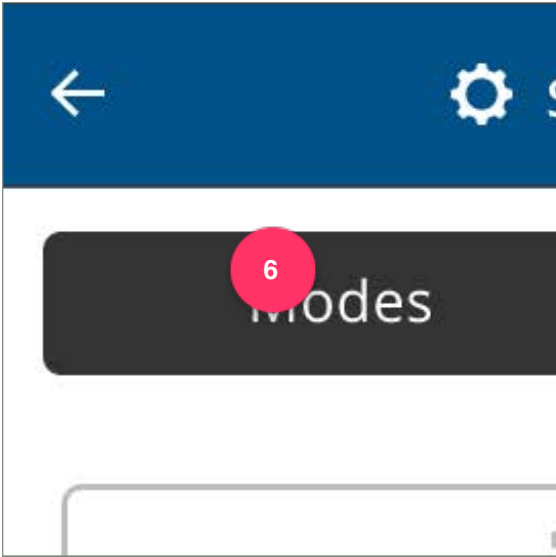
Comments:



Dave on Apr 3 at 10:57pm

Let's try to avoid using "TNC" anywhere in the app. Nobody will know what that means. How about "SHARE" (covers ridesharing and carsharing)?

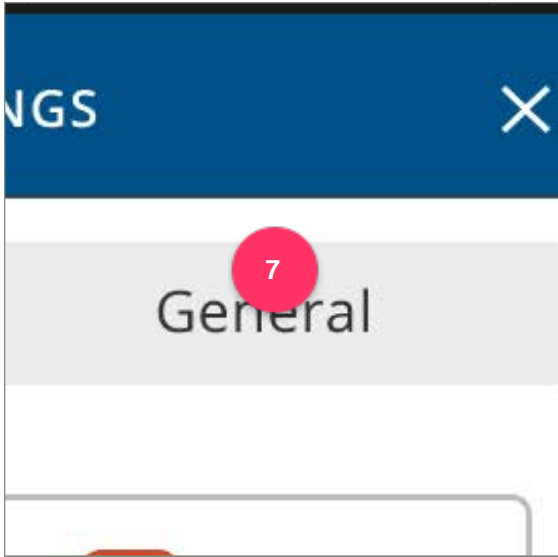
Comments:



Dave on Apr 3 at 10:57pm
Services?

Comments:

A large, empty white rectangular box with a thin grey border, intended for user comments.



Dave on Apr 3 at 10:58pm

"General" may be a bit too general... perhaps "Preferences"?

Comments:

UI8 9:41 AM

WE FOUND 3 OPTIONS

★ **Current Location**

📍 Paramount Hotel

Transit: TriMet Bus 17

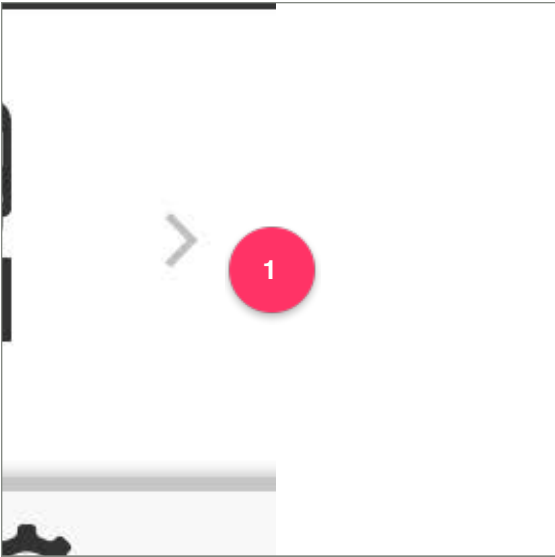
15 MINS TOTAL

🚶 5 mins walk

🕒 Leave at 1:04pm
Arrive at 1:19pm

🚌
17

🔗 ⚠️ ⚙️



Kate Chanba on Mar 24 at 5:30pm

@Dave what do you think about adding specific mode/service branding here?



Dave on Apr 3 at 11:19pm

@KateChanba Ugh... Tough one because it could end up being a big mess of icons, if there are multiple agencies involved, no? For TriMet/Streetcar purposes, I'd say no we should not add agency branding. But if we incorporate C-TRAN service that could change. :)

Curious: this mock shows a one-seat ride. How would you represent a trip with multiple lines and/or modes?

I assume we'd also show the selected itinerary (defaulting to option 1) on the map as well, ala Google Maps? And curious what the corresponding experience would be on desktop. Seems like we'd want to be showing the itinerary on the map at this point for sure. For me it brings up the question: Maybe we should skip this step of selecting an itinerary option (given that the majority of people will want option 1 anyway) and just go ahead and show option 1, THEN give people the option to select 2 and 3 ("See other trip options" or similar). This would line up with "auto plan" concept too.

Although we won't, other agencies may want to show the fare here as part of the comparison.

As for the slider/carousel UI, which I like, I think we need to make it more evident that it is a list of options you

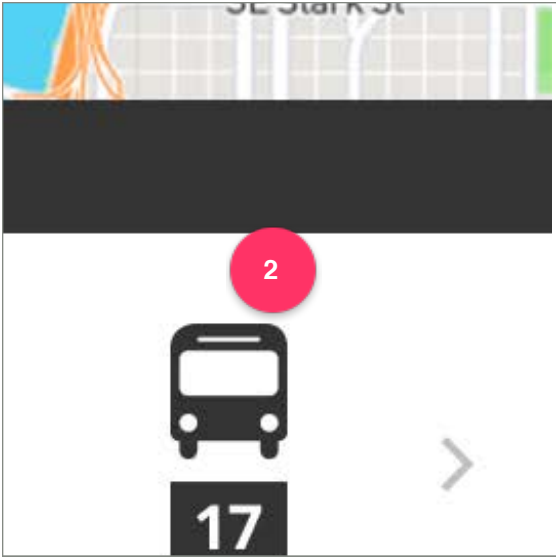
can cycle through somehow. Given our experience testing websites with riders in focus groups, I would anticipate several in the group not knowing that was a slider at the bottom. Also, I'm not sure it's clear enough how to select the itinerary you want. What is tappable/clickable (I know the answer, but I'm not sure the majority of riders will).



Kate Chanba on Apr 5 at 9:22pm

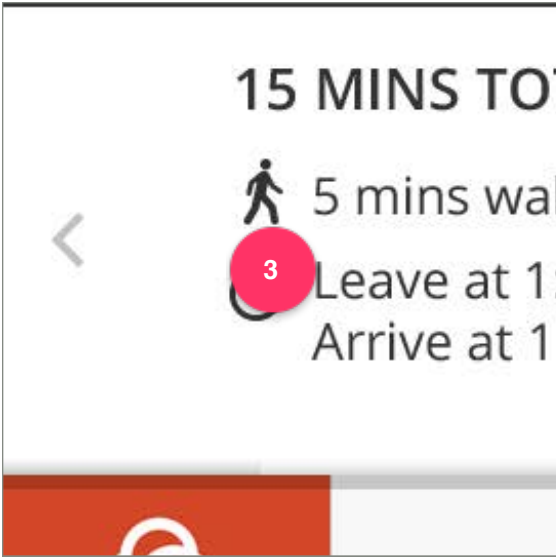
@Dave Thanks! The redesign will definitely include the route geometry in this view and I think we should stick with separate screens between Summary Options and Detailed Itinerary view. We will address making the carousel functionality clearer to the user.

Comments:



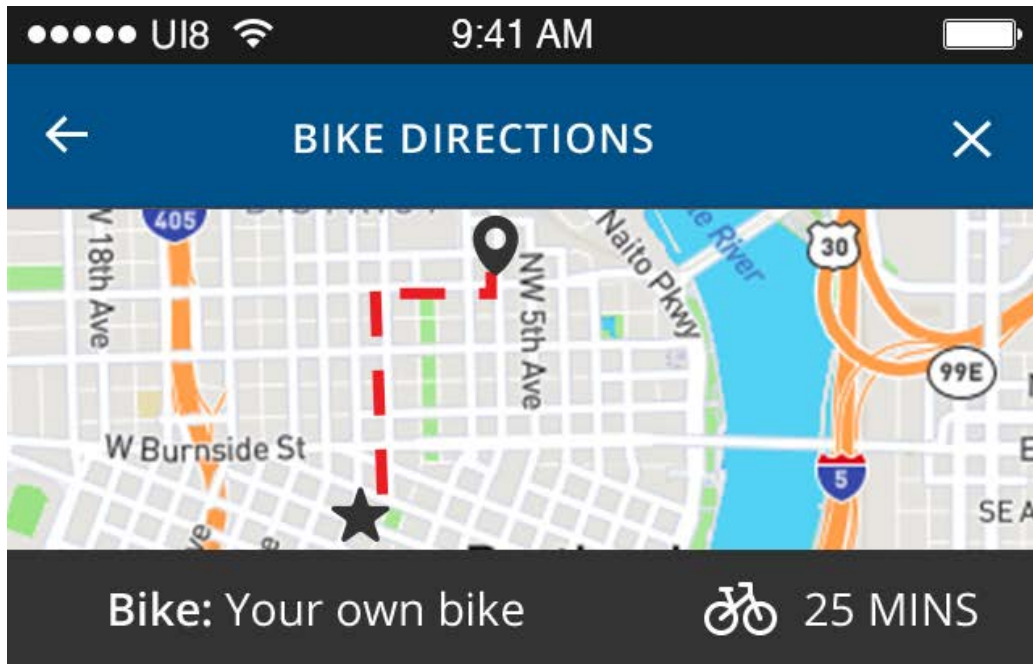
Kate Chanba on Apr 5 at 8:17pm
RT arrivals in summary view

Comments:



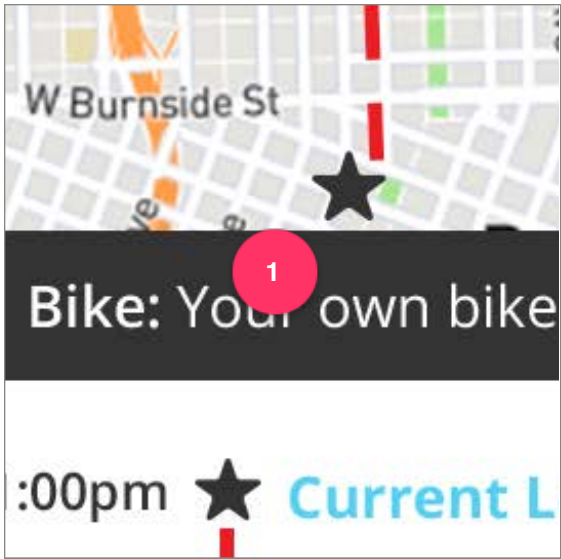
Kate Chanba on Apr 5 at 8:17pm
RT icon

Comments:



- 1:00pm ★ **Current Location**
- ↑ Head west on Main St
.2 mi
- Turn **right** on X St
400 ft
- ← Turn **left** on NW 4th St
100 ft
- 1:25pm 📍 **Paramount Hotel**



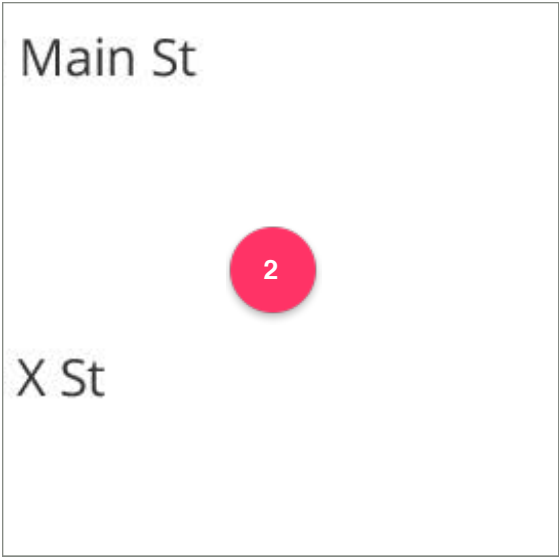


D

Dave on Apr 3 at 11:21pm

Curious how this would look if trip involved multiple modes. Could make this chunk much longer/bigger, with description and icons of the various modes.

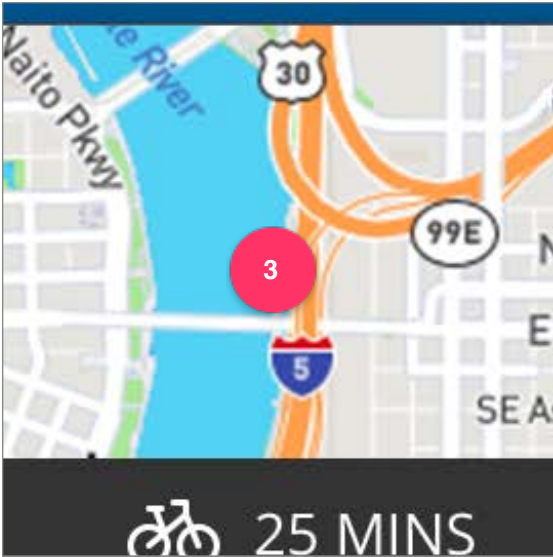
Comments:



Dave on Apr 3 at 11:23pm

Looks great to start! Beeeyootiful. We should mock up a transit trip with multiple modes and transfers, so we can cover all the UI scenarios.

Comments:



Dave on Apr 3 at 11:27pm

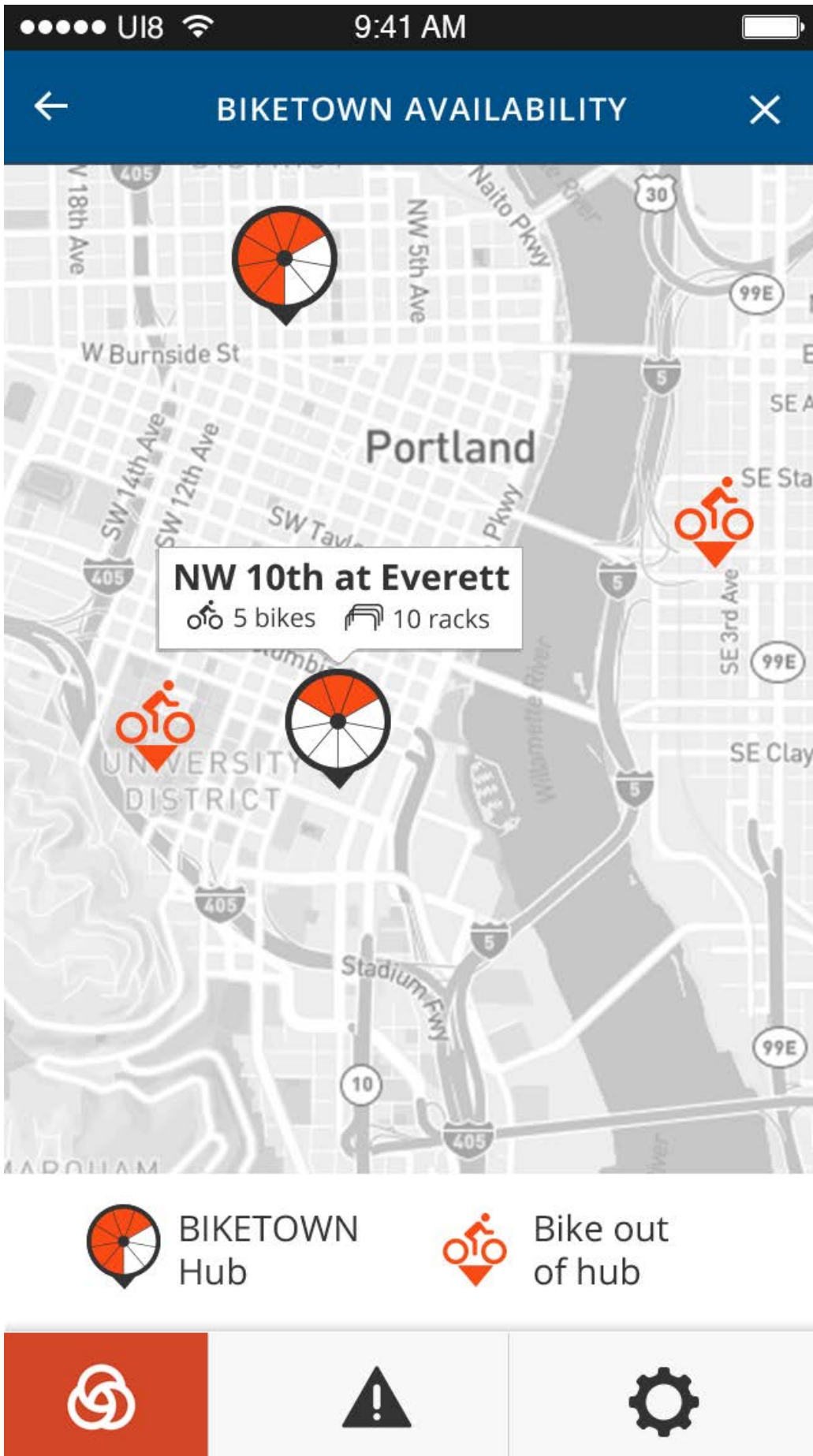
We're assuming at this point you could choose to tap on the map to minimize the itinerary and explore the map. Is that your assumption? You could see the transit route lines for your trip, where they go, tap on origin/destination and stops for service/transfer details, etc. Also, when it comes to a transit itinerary, this is the ideal place to present real-time arrival info.

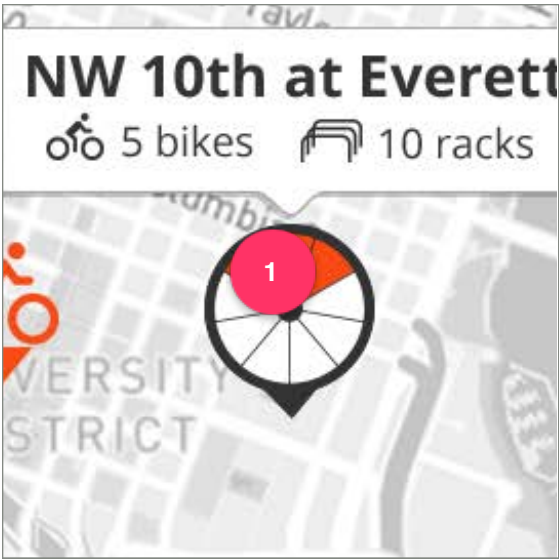


Kate Chanba on Apr 5 at 9:23pm

@Dave Yes, definitely both to the exploration in the map view and the opportunity to show RT info in the Itinerary and previous screen Summary.

Comments:





Dave on Apr 3 at 11:30pm

Sorry if I missed this detail in the call -- is this piechart only visible after you tap on the bike icon?



Kate Chanba on Apr 5 at 9:24pm

@Dave This 'pie chart' is a view of a bikeshare hub vs. an individual bike. Might need a bit more context on how this system works to design better for it. Will map access to this screen in the next iteration.

Comments:

APPENDIX D

TriMet MOD Grant Task 4 Milestone 5 Documentation

Street address interpolation

[search \(/tag/search\)](#) [geocoding \(/tag/geocoding\)](#) [data \(/tag/data\)](#) [engineering \(/tag/engineering\)](#)

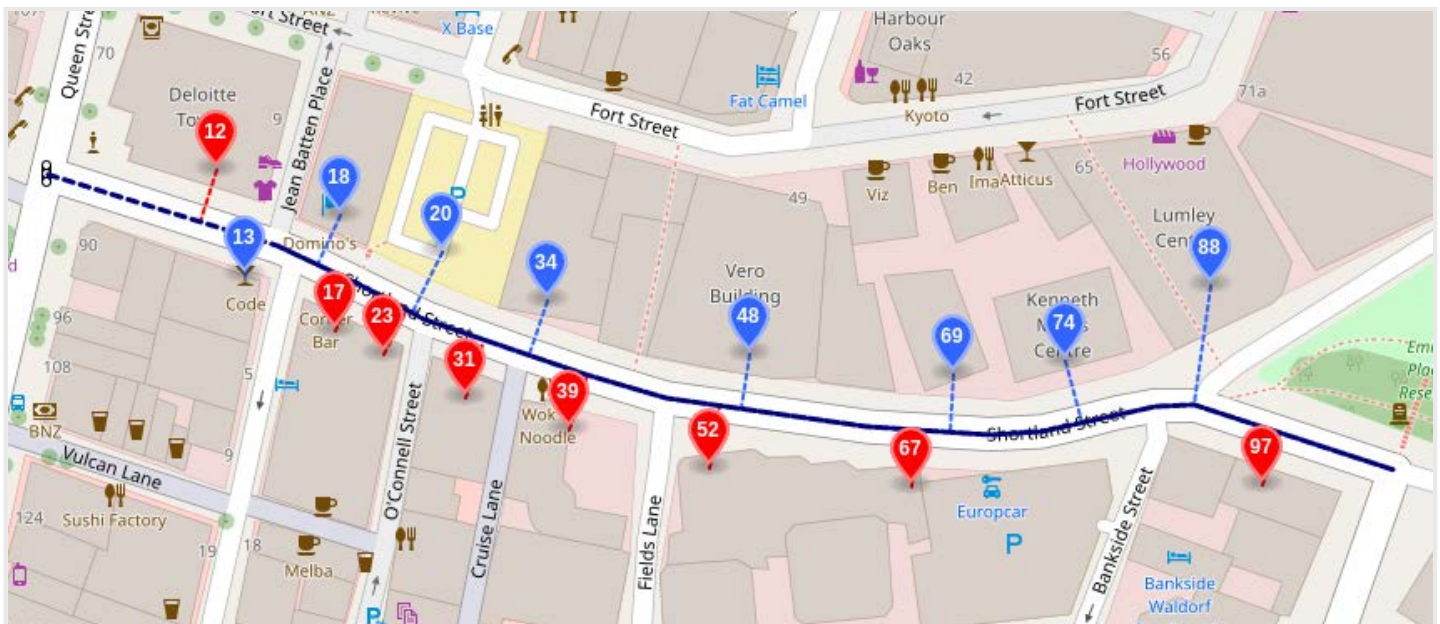
Today we're announcing a new address interpolation system. It was designed for the **Pelias geocoder** (<https://github.com/pelias/pelias>) and is now available in **Mapzen Search** (<https://mapzen.com/products/search/>), but can also be used as a stand-alone application or be included with any other geographic software or search engine.

The term 'interpolation' refers to a method of creating new data points from a set of known data points. For geographic search, this is important for street address data where some house numbers may be missing from our data providers.

The **OpenStreetMap** (<http://www.openstreetmap.com>) and **OpenAddresses** (<http://www.openaddresses.com>) projects provide a huge cache of street address information; between them over 500 million geographic address points are freely available to download and use for routing people and cars from A to B.

The issue is that these datasets are rarely complete, they contain 'holes' in the data where house numbers or geographic coordinates are not available.

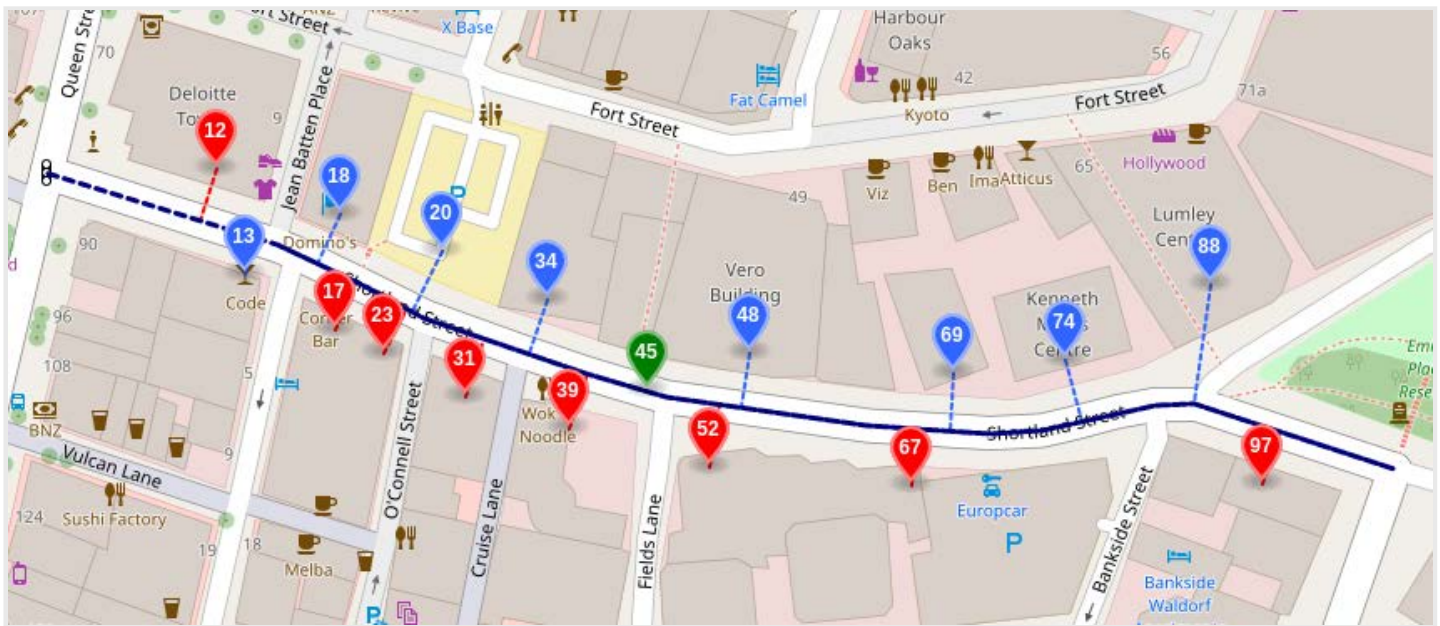
Consider a driver asking for directions to 45 Shortland Street, Auckland, NZ :



Red and blue dots are known address points from OpenStreetMap and OpenAddresses.

The 'holes' in the data result in our search engine returning a less granular result for queries which request the location of one of these missing addresses.

By filling in these gaps in the data, we can intelligently estimate where the missing house numbers would lie on each road:



The system is also aware of the geometric shape of each street and uses this information, along with the point data, to ensure that the location it returns lies on the road network and not somewhere which could be a potential driving hazard.

Interpolated data points are never as precise as the rooftop accuracy data we get from our providers. However, we aim to provide a location which is not only safe to drive to but also close enough for a driver to visually identify the letter box or driveway of their destination.

Improving address discovery

Over the last few months we have been rolling out improvements to Mapzen Search on indexing and retrieving postal addresses.

Back in September we finished a project which imports all of the **OpenStreetMap road network in to Pelias** (<https://github.com/pelias/polylines>). This first step allowed us to return the mid-point of each road geometry in lieu of having the coordinates for an exact house number.

The next step was to produce an interpolation engine which could go one step further and return an interpolated address location based off the other data points we have for that street.

Design goals

We set out with the following strategic goals in mind:

- Ensuring every street in OpenStreetMap is indexed and retrievable.
- Supporting address ranges as provided by OpenStreetMap, TIGER, et al.
- Combining and de-duplicating distinct address point data sets.
- Designing the system to scale beyond 1 billion address points.
- Allow room for future extension / improvements.

These changes will improve the user experience by:

- Providing house number interpolation where address range data exists.
- Falling back to a street centroid in lieu of a satisfactory house number.
- Reducing noise by only showing a maximum of one result per street.

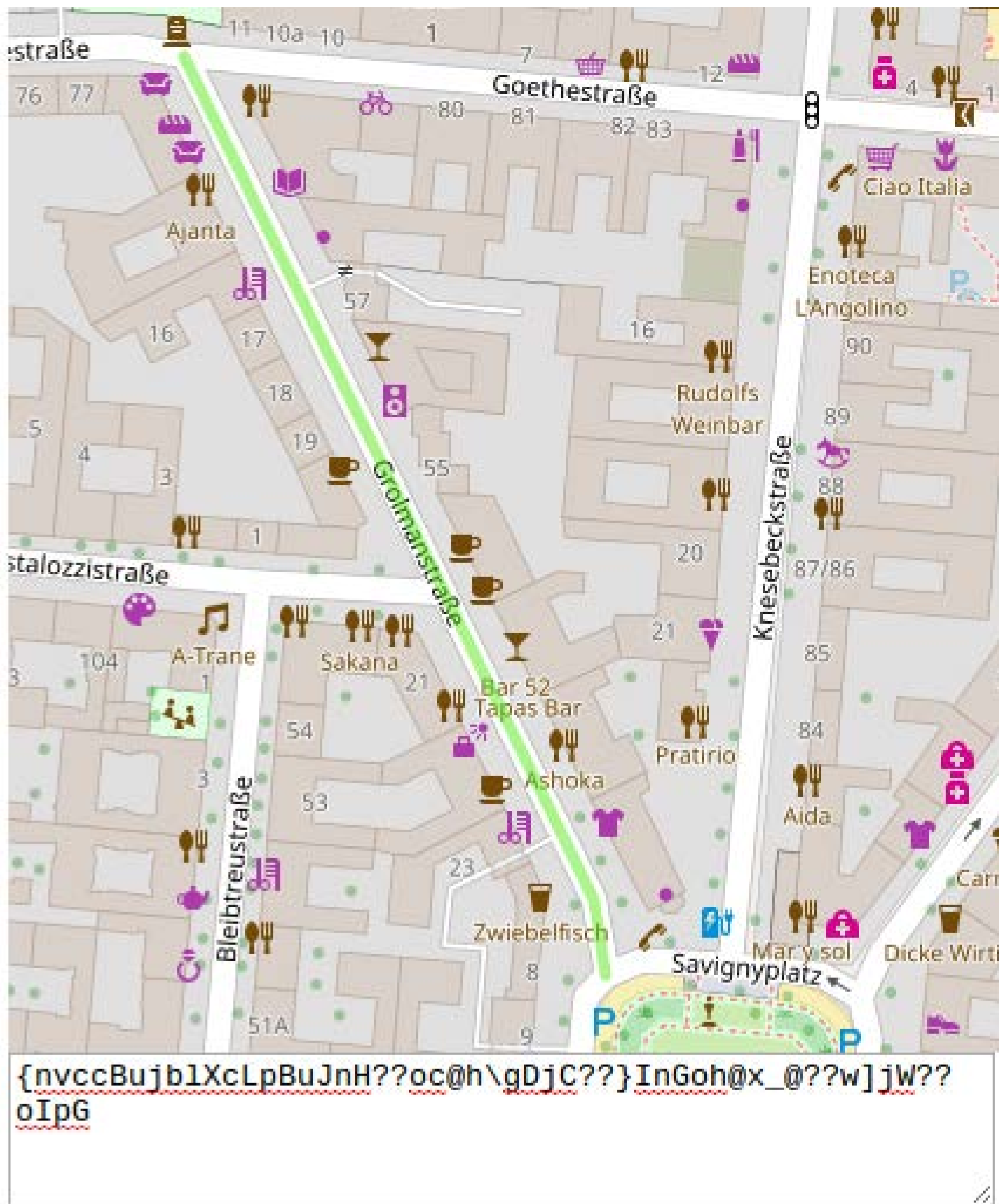
How it works

Road network

First we need a good source of open road geometry data. We can get this from OpenStreetMap.

The data is provided as 'ways' which first need to be joined together in order to form the longest contiguous line string. This will allow us to not only interpolate over 'ranges' of data (such as blocks) but also along the whole length of the street, even if it runs between different countries!

To accomplish this we use the **Valhalla** (<https://mapzen.com/blog/valhalla-intro/>) routing engine to export the routing graph into an encoded line geometry format called **Polyline** (<https://developers.google.com/maps/documentation/utilities/polylinealgorithm>).



The polyline for Berlin's Grolmanstraße

Once we have these poly-lines for each street in the world, we can go ahead and compute a point halfway along the path to use as a mid-point value, then those streets can be imported directly in to **Pelias** (<https://github.com/pelias/polylines>) for use in a 'fallback' scenario when we cannot find the exact address the user is searching for.

For the purposes of interpolation, we use the same polyline dump to import those geometries in to `sqlite3`. We use the **rtree** (<https://www.sqlite.org/rtree.html>) module to create a fast spatial index of the street bounds, then we store the geometry along with the bounds and the street name(s) in a database called `street.db`.



Putting Grolmanstraße in a bounding box.

In order to normalize the street names we run the street names through the **libpostal** (<https://github.com/openvenues/libpostal>) address expander. This allows us to do things like remove diacritics, handle unicode and perform synonym substitutions.

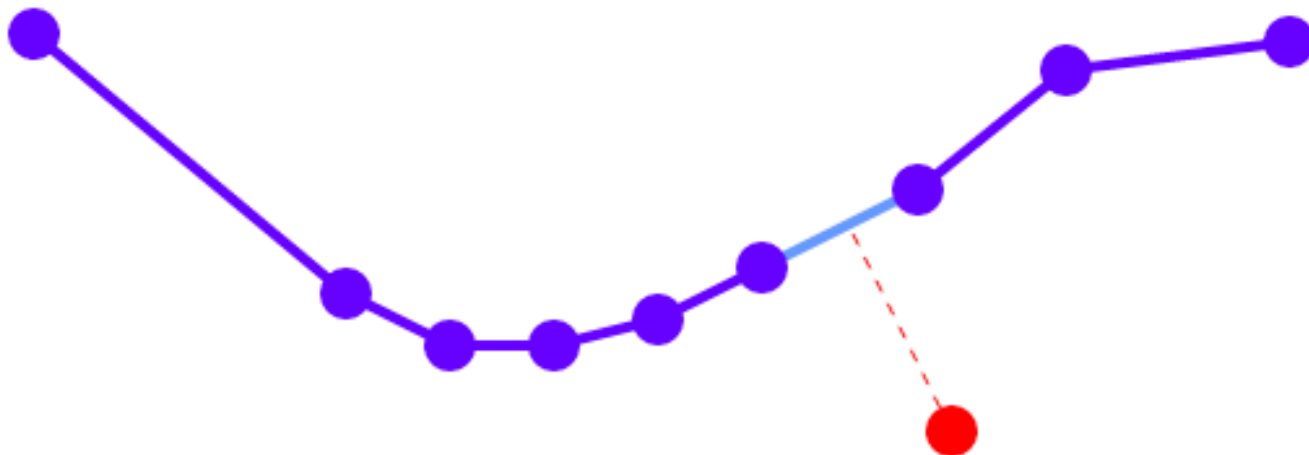
The importer takes a few hours to run and produces a single `street.db` file (~5.3G) which can be shared with any other applications which need a conflated road network data set. Since the file is `sqlite3` it can be accessed from any programming language or copied to an embedded device such as a mobile phone or even your car!

Addresses Points

Next we download >400 million **OpenAddresses** (<https://openaddresses.io/>) (OA) and perform street name normalization which allows us to use text matching techniques to match them with the road network.

For each data point in the OA files we attempt to find a corresponding street in the `street.db` using the rtree index. This results in a >80% success rate for matching one data set with the other.

Now we have the address points and the street geometry we can then 'project' each point on to the shape of the street. To do this we use a linear algebra function to find its orthogonal projection; this is basically the point on the road closest to the front door of the house.



Click for an **interactive demo** (<http://geo.wiz.co.nz/projection.html>).

We then store each address in a database called `address.db` : each row contains both the original position and the projected position, along with id of the corresponding street in `street.db` .

But it's not only OpenAddresses! We also import OpenStreetMap entities tagged with both `addr:street` and `addr:housenumber` .

We then finish off by importing all the block ranges from the **US Census Dept 2016 TIGER files** (<https://www.census.gov/geo/maps-data/data/tiger.html>), giving us a very high level of coverage in the United States.

Following the contour of the street

The final part of the build is to perform some pre-computation on the street geometry itself; we want to build an index which is highly performant at search time.

In order to avoid loading the street geometry every time we want to perform a search, we iterate over each vertex (corner) of the street geometry and compute a 'fractional house number'.

Using the data we already have about where each house number lies on the path, we can now ask the system "if this corner of the road was a house, what house number would it have?"

| rowid | id | source | housenumber | lat | lon | parity | proj_lat | proj_lon |
|-------|----|--------|-------------|-------------|-------------|--------|-------------|-------------|
| 5 | 1 | OA | 1.0 | -41.2871999 | 174.766753 | R | -41.287285 | 174.7666662 |
| 23 | 1 | VERTEX | 2.535 | | | | -41.287388 | 174.766845 |
| 22 | 1 | VERTEX | 3.376 | | | | -41.287461 | 174.766921 |
| 21 | 1 | VERTEX | 4.154 | | | | -41.287533 | 174.766983 |
| 20 | 1 | VERTEX | 4.884 | | | | -41.287606 | 174.767028 |
| 10 | 1 | OA | 7.0 | -41.2877481 | 174.7674712 | R | -41.2878291 | 174.7671188 |
| 11 | 1 | OA | 9.0 | -41.2878189 | 174.7673061 | R | -41.2878591 | 174.767131 |

This step is very important because it allows us to preserve the shape of the street, meaning we will never return a point which does not lie on the path.

We designed the system in such a way that a vehicle driver would never be given a destination which lies in a hazard or is not routable using turn-by-turn navigation systems.

Contributing & Feedback

As always, this project is open-source and **available on Github (<https://github.com/pelias/interpolation>)**. Please support the project by opening descriptive issues when you find them. (Note that when Mapzen Search returns an interpolated address, you'll see `interpolated` for the `match_type` in the results.)

```
},
"properties": {
  [...],
  "name": "207 Spear Street",
  "housenumber": "207",
  "street": "Spear Street",
  "confidence": 0.8,
  "match_type": "interpolated",
  "accuracy": "point",
  [...],
}
```

We will be soon be offering database downloads of the interpolation index and we'll continue to improve the data quality and interpolation algorithms as we work towards having over a billion address points available for search worldwide.

As always, we would love to get your feedback on Mapzen Search and our open source geocoding engine, Pelias. You can find us on **here (<http://github.com/pelias/pelias>)**, **there (<https://gitter.im/pelias/pelias>)**, and **way over there (search@mapzen.com)**

· 23 February 2017 ·



Peter Johnson

Peter works on the search team, helping people get from A to B.

Get a rundown of Mapzen happenings at the beginning of every month!

SIGN UP FOR OUR NEWSLETTER (<https://mapzen.com/newsletter-signup/>)



pelias / interpolation

Watch 7
 Star 9
 Fork 0

- [Code](#)
[Issues 10](#)
[Pull requests 9](#)
[Projects 0](#)
[Wiki](#)
[Pulse](#)
[Graphs](#)

global street address interpolation service (beta) <http://interpolation.wiz.co.nz/demo/>

349 commits

26 branches

0 releases

4 contributors

Branch: master

New pull request

Create new file

Upload files

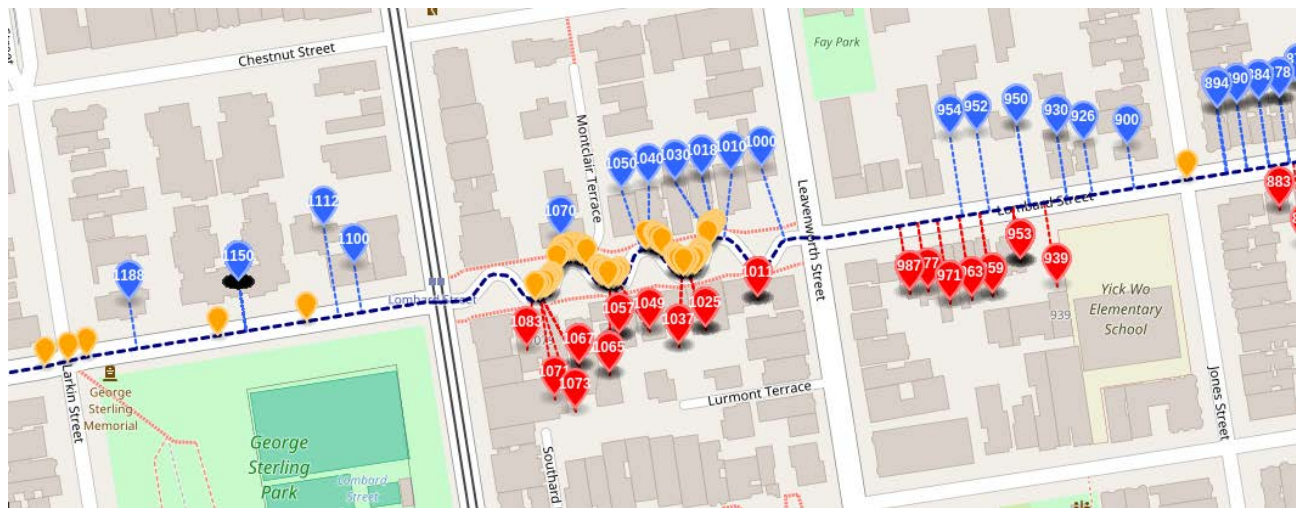
Find file

Clone or download

| | | |
|---|---|----------------------------------|
| dianashk BREAKING CHANGE: first version | | Latest commit e09c5e1 4 days ago |
| api | tiger: replace npm:shp2json with ogr2ogr | 2 months ago |
| cmd | tiger: importer | 2 months ago |
| demo | demo: examples | 4 months ago |
| dev | tiger: dev example script | 2 months ago |
| lib | store source ids in database | 3 months ago |
| query | store source ids in database | 3 months ago |
| script | typo | 2 months ago |
| stream | tiger: sort feature collection for performant batching | 2 months ago |
| test | tiger: functional tests | 2 months ago |
| .dockerignore | add Dockerfile | 6 months ago |
| .gitignore | gitignore | 6 months ago |
| .jshintignore | add linter, precommit hook. lint entire project and fix errors | 6 months ago |
| .jshintrc | .jshintrc | 5 months ago |
| .travis.yml | feat: add semantic release and travis | 4 days ago |
| Dockerfile | Dockerfile: add apt package gdal-bin which provides application ogr2ogr | 2 months ago |
| index.js | model: finish refactoring to use models | 5 months ago |
| interpolate | tiger: bin help | 2 months ago |
| package.json | BREAKING CHANGE: first version | 4 days ago |
| readme.md | readme: add documentation for tiger import | 2 months ago |

readme.md

An open source + open data project to perform global street address interpolation queries. Sponsored by [mapzen](#).



About

The [Openstreetmap](#) and [Openaddresses](#) projects provide a huge cache of street address information; between them around 500 million address points are freely available to download.

Some countries like Germany and the USA have dense address coverage while other have only sparse data available.

This project aims to 'fill in the gaps' in the data by intelligently estimating where the missing house numbers would lie on the road.

The service was designed for use with the [pelias geocoder](#), however it can also be used as a stand-alone application or included with other geographic software / search engines.

more info: [\[design doc\]](#) [\[relationship to pelias\]](#) [\[existing standards\]](#) [\[conflation\]](#)

Architecture

The software is written in javascript to run in nodejs, the storage engine is sqlite3.

Client libraries can be written in any language which can read sqlite3. If you wish to write a client in another language please open an issue and we can explain which functions you will need to port.

The software is split in to 6 distinct parts:

- the street (polyline) importer
- the openaddresses address importer
- the openstreetmap address + address range importer
- the T.I.G.E.R. block range importer
- the geometry (vertices) interpolation
- the client APIs (the webserver and CLI interface)

The data is split in to 2 different sqlite3 databases:

- street.db (holds information about streets, geometry, their names and bounding boxes)
- address.db (holds address point data, both rooftop accuracy and pre-interpolated vertex data)

Downloading pre-built data

Mapzen provides data extracts which you can download and get going immediately.

[coming soon]

Workflow

street database

Firstly you need to build the `street.db` database.

You will need a polyline data file which contains all the streets you wish to import, you can find some pre-made extracts [here](#) and there is also information on that readme about how to generate your own extracts.

See the [building the databases](#) section below for detailed information on which commands to run.

There is also a script named `./script/import.sh` in this repository which makes running this process much easier.

note: We only support the `polyline` format, you will need to format-shift data from other formats in order to import it.

address database

Next you need to build the `address.db` database.

You will need to download one or more `openaddresses` or `openstreetmap` files for the addresses you wish to import.

See the [building the databases](#) section below for detailed information on which commands to run.

There are scripts named `./script/conflate_oa.sh` and `./script/conflate_osm.sh` in this repository which make running this process much easier.

note: We only support `openaddresses` and `openstreetmap` formats, you will need to create a custom importer for other sources.

precompute geometry

Finally we will compute the fractional house numbers for each vertex (corner) of the street and add them to the `address.db` database.

See the [building the databases](#) section below for detailed information on which commands to run.

Using the command line

help

get information about which commands are available and list the arguments they accept

```
$ ./interpolate help
```

```
Usage: interpolate [command] [options]
```

```
Note: you will need to pipe data in to the import/conflate commands
```

help

```
search [address_db] [street_db] [lat] [lon] [house_number] [street_name]
```

```
polyline [street_db]
```

```
oa [address_db] [street_db]
```

```
osm [address_db] [street_db]
```

```
tiger [address_db] [street_db]
```

```
vertices [address_db] [street_db]
```

```
extract [address_db] [street_db] [lat] [lon] [street_name]
```

```
server [address_db] [street_db]
```

```
build
```

```
output usage information
```

```
search database for specified housenumber
```

```
import polyline data in to [street_db]
```

```
conflate oa csv file in to [address_db] u
```

```
conflate osm file in to [address_db] usir
```

```
conflate tiger address range geojson file
```

```
compute fractional house numbers for line
```

```
extract street address data for debugging
```

```
start a web server
```

```
run the import script
```

search

search the db for an address, return an interpolated value if an exact match does not exist

note: the lat/lon values you provide are in order to disambiguate the street, they must lie within the bounding box of the desired street.

```
./interpolate search address.db street.db "-41.288788" "174.766843" "16" "glasgow street"
```

```
type    interpolated
source  mixed
number  16
lat     -41.2886487
lon     174.7670925
```

extract

extract address data from the db for a specific street

note: the lat/lon values you provide are in order to disambiguate the street, they must lie within the bounding box of the desired street.

```
./interpolate extract address.db street.db "-41.288788" "174.766843" "glasgow street"
```

| rowid | id | source | houenumber | lat | lon | parity | proj_lat | proj_lon |
|-------|----|--------|------------|-------------|------------|--------|------------|-------------|
| 5 | 1 | OA | 1 | -41.2871999 | 174.766753 | R | -41.287285 | 174.7666662 |
| 23 | 1 | VERTEX | 2.535 | | | | -41.287388 | 174.766845 |
| 22 | 1 | VERTEX | 3.376 | | | | -41.287461 | 174.766921 |

...

Using the web server

Start the web server

run a web server which exposes the search APIs via an HTTP interface

note: you can set an environment variable named 'PORT' to change the port number.

```
./interpolate server address.db street.db
```

```
server listening on port 3000
```

GET /search/{format}

search the db for an address, return an interpolated value if an exact match does not exist

```
geojson : /search/geojson?lat=-41.288788&lon=174.766843&number=16&street=glasgow%20street
```

```
html : /search/table?lat=-41.288788&lon=174.766843&number=16&street=glasgow%20street
```

GET /extract/{format}

extract address data from the db for a specific street


```
geojson : /extract/geojson?lat=-41.288788&lon=174.766843&names=glasgow%20street
```

```
html : /extract/table?lat=-41.288788&lon=174.766843&names=glasgow%20street
```

GET /street/near/geojson

find the 100 nearest streets to a specific lat/lon pair, ordered by distance ASC

```
geojson : /street/near/geojson?lat=-41.288788&lon=174.766843
```

GET /street/{id}/geojson

return the geometry for a specific street id

```
geojson : /street/10/geojson
```

see: [source](#) for more information.

Building the databases

polyline

import road network data in the polyline format

find data here: <https://github.com/pelias/polylines>

```
./interpolate polyline street.db < /data/new_zealand.polylines
```

oa

import openaddresses data and conflate it with the street data

find data here: <https://openaddresses.io/>

```
./interpolate oa address.db street.db < /data/oa/nz/countrywide.csv
```

note: sorting the openaddresses files so that addresses on the same street are adjacent will significantly speed up imports, you can find an example of the commands required to sort the data in `./script/concat_oa.sh`.

osm

import openstreetmap data and conflate it with the street data

find data here: <https://mapzen.com/data/metro-extracts/>

the importer expects the OSM data in the JSON format exported by <https://github.com/pelias/pbf2json>, this format is not strictly equivalent to the http://overpass-api.de/output_formats.html#json standard, be aware.

for now it's best to use `pbf2json` to convert a `.osm.pbf` file in to json, then pipe that data in to `./interpolate osm`:

```
./build/pbf2json.linux-x64 -tags="addr:housenumber+addr:street" london.osm.pbf > osm_data.json
```

```
./interpolate osm address.db street.db < osm_data.json
```

tiger

import US Census Bureau TIGER data and conflate it with the street data

find data here: <https://www.census.gov/geo/maps-data/data/tiger-line.html>

a script is provided in `./script/update_tiger.sh` which will download files for the whole of the USA, this script is safe to run multiple times as it will only update the data which has changed.

```
./interpolate tiger address.db street.db
```

vertices

compute fractional house numbers for the street vertices

```
./interpolate vertices address.db street.db
```

logging

you can record a log of addresses which do not find a matching street. simply create an additional file descriptor, this will trigger the process to use it for logging. eg:

```
cat /data/oa/nz/countrywide.csv | ./interpolate oa address.db street.db 3> skip.list
```

docker

build docker image

this can take some time for the first build due to installing libpostal from source

```
docker build -t pelias/interpolation .
```

you can confirm that worked with:

```
docker images
REPOSITORY          TAG          IMAGE ID          CREATED           SIZE
pelias/interpolation latest       7ca651b86a63     16 minutes ago   3.068 GB
```

run docker image

this will run a new container based off the image created above

notes:

- `-p` controls port mapping (port `3000` in the container maps to `5000` in the host)
- `-v` controls volume mapping (`/data` in the container maps to `/data` in the host)
- `-d` tells docker to run the container in the background (daemonize)

by default this will launch the server using the databases `/data/address.db` and `/data/street.db` which must be present on the host machine

```
docker run -p 5000:3000 -v /data:/data -d pelias/interpolation
```

you can confirm that worked with:

```
$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
ac9c8f607b2e  pelias/interpolation  "./interpolate server"  14 minutes ago  Up 14 minutes  0.0.0.0:5
```

you should now be able to access the web server locally at `http://localhost:5000/demo/`

running scripts other than `server` in the docker container

you can run any command supported by `./interpolate` via the docker container, such as:

```
cat /data/new_zealand.polylines | docker run -i -v /data:/data pelias/interpolation polyline /data/nz.db
```

running a build in the docker container

the build scripts are configurable via environment variables, you will need to download your data before running the build command.

```
# prepare a build directory and a data directory to hold the newly created database files
mkdir -p /tmp/data/berlin

# download polyline street data
curl -s http://missinglink.files.s3.amazonaws.com/berlin.gz | gzip -d > /tmp/data/berlin.0sv

# download and extract openaddresses data
curl -s https://s3.amazonaws.com/data.openaddresses.io/runs/142027/de/berlin.zip > /tmp/data/berlin.zip
unzip /tmp/data/berlin.zip -d /tmp/data

# download openstreetmap data
curl -s https://s3.amazonaws.com/metro-extracts.mapzen.com/berlin_germany.osm.pbf > /tmp/data/berlin.osm.pbf
```

we will mount `/tmp/data` on the local machine as `/data` inside the container, so be careful to set paths as they appear inside the container.

```
docker run -i \ # run interactively (optionally daemonize with -d)
-v /tmp/data:/data \ # volume mapping
-e 'BUILDDIR=/data/berlin' \ # location where the db files will be created
-e 'POLYLINE_FILE=/data/berlin.0sv' \ # location of the polyline data
-e 'OAPATH=/data/de' \ # location of the openaddresses data
-e 'PBF2JSON_FILE=/data/berlin.osm.pbf' \ # location of the openstreetmap data
pelias/interpolation build
```

once completed you should find the newly created `street.db` and `address.db` files in `/tmp/data/berlin` on your local machine.

development

install dependencies

note: [libpostal](#) must be installed on your system before you continue!

The `Dockerfile` in this repo has complete instructions on how to install everything from scratch on Ubuntu.

```
npm install
```

run tests

```
npm test
```

run linter

note: if you are using the `atom` editor, we recommend the [jshint](#) plugin.

```
git commit
```

functional tests

this repo contains a bunch of functional end-to-end tests in the `./test/functional` directory.

each test contains a `reports` directory which contains human-readable visual output of each test case, including a geojson map view showing all the point data.

create a new functional test

the easiest/best way to debug an issue is to create a new functional test case and use the test suite to assert conditions and generate visual output which you can inspect.

these are the steps I took to create the `potsdamerplatz` test case:

```
# copy an existing test case
cp -r test/functional/basic test/functional/potsdamerplatz

# extract the relevant polylines from a large polyline source
grep -Pia "potsdamer\s?(platz|strasse|straße)" /data/planet.polylines > test/functional/potsdamerplatz/osm.polylines

# extract the relevant address points from a large openaddresses file (header line then body)
head -n1 /data/oa/de/berlin.csv > test/functional/potsdamerplatz/oa.csv
grep -Pia "potsdamer\s?(platz|strasse|straße)" /data/oa/de/berlin.csv >> test/functional/potsdamerplatz/oa.csv

# extract the relevant address points from an openstreetmap PBF extract
# see: https://github.com/pelias/pbf2json
./build/pbf2json.linux-x64 -tags="addr:housenumber+addr:street" /data/extract/greater-london-latest.osm.pbf | grep -i
```

next add that test case to `./test/_func.js` in order to it run every time anyone runs `npm test`.

you can now edit the contents of `test/functional/potsdamerplatz/run.js` to suit your needs, you should rename the text at the bottom of the file which says something like `"functional: basic"` to be more descriptive, in this case we will call it `"functional: potsdamerplatz"`. now the output from `npm test` will include that label next to each assertion run in the file.

great! you can skip the units tests and only run the functional tests with `npm run funcs`, go ahead and do that now and you will see your new tests failing; which is good! if you are going to be running that command a lot and you don't care to wait on the other tests, you can comment them out in `./test/_func.js`.

now your test case is running it's time to have a poke around in that new directory you made.

running the tests will produce new `street.db` & `address.db` files, you can query them directly from the command line to check what's inside them looks correct:

```
sqlite3 test/functional/potsdamerplatz/street.db "SELECT * FROM polyline JOIN names ON polyline.id = names.id WHERE r
```

or you can start an interactive shell and have a poke around in there:

```
sqlite3 test/functional/potsdamerplatz/street.db

SQLite version 3.8.11.1 2015-07-29 20:00:57
Enter ".help" for usage hints.
```

```
sqlite> select count(*) from polyline;
64

sqlite> .exit
```

you'll find a subdirectory called `./fixture` which is where all your data fixtures will live, you can query the database directly and save the response to that directory with a command such as:

```
sqlite3 test/functional/potsdamerplatz/address.db "SELECT * FROM address WHERE id = 1" > test/functional/potsdamerpla
```

if you're hunting for a specific road segment to debug, you can open up your `test/functional/potsdamerplatz/osm.polylines` file and try to find the appropriate line in there, the line numbers will correspond to the ids, so the first line in that file is `id=1` in the `street.db`.

to visually inspect the polylines, you can cut them before the name and paste them in the search box here:

<http://valhalla.github.io/demos/polyline/>

example:

```
ii|ccBsskoX_@wLaAkUi@sJ{@oM?mHn@??HtL~IUvGc@zJyA|Cg@tCkAhBkA~EyDpBaDnGgIeBqEoEnGwGvEqAz@wAr@qC|@wM1@mJ^
```

likewise if you are looking for a specific address, you can open up `test/functional/potsdamerplatz/oa.csv` and find the address in there (be aware that some very precise floating point numbers get truncated and so may not match exactly in tools like `grep`), you should then be able to find them in the `address.db`:

```
sqlite3 test/functional/potsdamerplatz/address.db "SELECT * FROM address" | grep "52.5046774"
```

```
45|1|0A|59.0|52.5046774|13.3676731|R|52.5047673|13.3674717
```

you'll also find a subdirectory called `./reports` which you can use to spot check the data, if you have [geojsonio](#) installed you can pipe the file directly to the web:

```
cat test/functional/potsdamerplatz/reports/preview.geojson | geojsonio
```

once you have completed your tests and committed the files, your preview files will be visible to everyone via github.

the `./reports` directory also contains the `stdio` files from each command that was executed and a list of records which failed to conflate. these files are ignored by `.gitignore` so they don't show up on github.

