

## Customer Journey Map:

# TrafficTelligence: User Journey Map

Scenario: Submitting traffic context data, receiving a prediction, and re-engaging based on useful insights

Entice	Enter	Engage	Exit	Extend
<p>"Help me discover a smarter way to estimate traffic before I start my trip."</p> <ul style="list-style-type: none"> <li>Steps: User sees a promise or link to Traffic telligence (via blog-travel or navigation app)</li> <li>Interactions: Website landing page / search engine result snippet integrated tool in another site</li> <li>Positive: Clean UI...hold claim "ML-powered"</li> </ul>	<p>"Help me give the app the right context with minimal effort."</p> <ul style="list-style-type: none"> <li>Steps: Reaching Input page and selects. weather temperature data, time, holiday, etc</li> <li>Interactions: HTML form with-dropdowns, toggles, calendar and sliders</li> <li>Positive/Places: Typically done solely on a computer or mobile browser</li> <li>Positive</li> </ul>	<p>"Help me get an accurate prediction and feel confident about what I see"</p> <ul style="list-style-type: none"> <li>Steps: Form-data sent to Flask backend</li> <li>Inputs: preprocessed (type conversion, encoding)</li> <li>ML model: predicts traffic volume</li> <li>interactions, Backend AFI (Flask). Randomforest Regressor model</li> <li>Styled result</li> </ul>	<p>"Help me wrap up with clarity and usefulness."</p> <ul style="list-style-type: none"> <li>Steps: User results and closes app</li> <li>Optional rating prompt or feedback request</li> <li>Option to download) or share-result</li> <li>Positive Moments: Feeling of control over traffic plans</li> <li>Sense of Insight</li> <li>Pain Points: No export/share option</li> </ul>	<p>"Help me keep getting smarter insights the more I use this."</p> <ul style="list-style-type: none"> <li>Steps: Personalized suggestions: "Try again at a different hour"</li> <li>Past predictions saved in profile (if logged in)</li> <li>Follow-up email "Did this estimate match your real experience?"</li> <li>Opportunities: ML model learns from user feedback</li> <li>Users return</li> </ul>