Project Proposal: The Fatigue-Aware Slow Travel Agent

1. Introduction

In the post-pandemic era, a polarizing trend known as "Special Forces Tourism" has emerged, characterized by high-intensity itineraries aimed at maximizing visited attractions within minimal timeframes [1]. While time-efficient, this approach often leads to physical exhaustion and superficial cultural engagement, contradicting the principles of "Slow Travel," which emphasizes pacing, sustainability, and deep connection to local environments [2].
Traditional travel recommendation systems (TRSs) typically utilize Point-of-Interest (POI) sequencing algorithms that optimize solely for distance or popularity, neglecting the traveler's real-time physical state or fatigue accumulation [3]. However, the rapid advancement of Large Language Models (LLMs) has enabled the creation of **Autonomous Agents** capable of complex reasoning, tool use, and state management [4, 5]. Unlike static algorithms, these agents can perceive dynamic environments and adjust plans adaptively using frameworks like **ReAct** (Reasoning + Acting) [6].
This project proposes a **"Slow Travel Planner Agent"** powered by **LangGraph** [7]. This system addresses the limitations of rigid scheduling by integrating **computer vision-based web browsing (Browser-use)** [8] to assess real-time crowd levels and **Amap (Gaode) APIs** [9] for precise pathfinding. By maintaining a stateful memory of the user's "fatigue metric" (derived from step count and transit time), the Agent dynamically inserts rest nodes (e.g., scenic cafes) into the itinerary, ensuring a travel experience that balances exploration with well-being [10, 11].

2. Objectives

The primary objective is to build an Agentic AI system that optimizes travel itineraries for physical comfort and experience quality rather than just speed.

1. **Develop a Fatigue-Adaptive Logic:** To implement a state-machine workflow that tracks cumulative walking distance and triggers "rest interventions" when specific thresholds (e.g., 15,000 steps) are exceeded.
2. **Integrate Multi-Modal Tool Usage:** To synergize structured geolocation data (Amap API) with unstructured web data (real-time ticket availability and crowd forecasts via Browser-use).
3. **Implement Cyclic Planning with LangGraph:** To move beyond linear processing chains and create a cyclic graph architecture that allows for self-correction and iterative re-planning based on environmental feedback.
4. **Create a Human-Centric Interface:** To deploy a user interface (e.g., Streamlit) that visualizes the "slow travel" logic, explicitly showing users where and why rest stops were

added.

3. Research Plan

Task 1: Infrastructure and Tool Development

- **Subtask 1.1:** detailed study of the Amap (Gaode) Web Service API to implement Python wrappers for Geocoding, Walking Route Planning, and Traffic Status.
- **Subtask 1.2:** Deployment of the browser-use library combined with a Vision-Language Model (e.g., GPT-4o or Gemini) to automate the retrieval of unstructured data, such as parsing weather warnings or checking ticket status on scenic spot websites.

Task 2: Agent Architecture Design (LangGraph)

- **Subtask 2.1:** Definition of the AgentState schema, including variables for the user itinerary, current fatigue level (step count), and environmental constraints (weather).
- **Subtask 2.2:** Implementation of the **"Planner Node"** for initial route generation and the **"Fatigue Monitor Node."** This monitor will utilize conditional logic: if cumulative_steps > threshold, the workflow routes to a "Coffee/Rest Finder" node instead of the next attraction.

Task 3: System Integration and Optimization

- **Subtask 3.1:** Construction of the complete graph with Conditional Edges (e.g., *If ticket sold out -> Return to Planner -> Select Alternative*).
- **Subtask 3.2:** Optimization of the "Anti-Special Forces" prompt engineering to ensure the LLM prioritizes relaxed pacing over maximizing POI counts.

Task 4: Evaluation and Interface

- **Subtask 4.1:** Development of a web-based dashboard using Streamlit to visualize the itinerary and the "Fatigue Curve."
- **Subtask 4.2:** Simulation testing against standard "greedy" algorithms to demonstrate the reduction in estimated fatigue and improvement in schedule feasibility.

4. Timeline

Tasks	**Deliverable/s**	**Due**
1. Theoretical Framework	Literature Review & Architecture Diagram (LangGraph)	Week 2
2. Tool Implementation	Functional Python modules for Amap API & Browser-use	Week 4
3. Core Agent Development	Working Prototype of the "Fatigue Monitoring" Logic	Week 6
4. Integration & UI	Streamlit Web App v1.0	Week 8
5. Testing & Refinement	Performance Report (Success rate of re-planning)	Week 9
6. Final Submission	Final Project Report & Presentation Deck	Week 10

5. References

1. **[Tourism Trend]** Cheer, J. M., Milano, C., & Novelli, M. (2019). Tourism and community resilience in the Anthropocene: Accentuating temporal overtourism. *Journal of Sustainable Tourism*, 27(4), 554-572.
2. **[Slow Travel]** Dickinson, J. E., Lumsdon, L. M., & Robbins, D. (2011). Slow travel: Issues for tourism and climate change. *Journal of Sustainable Tourism*, 19(3), 281-302.
3. **[Itinerary Problems]** Lim, K. H., Chan, J., Karunasekera, S., & Leckie, C. (2019). Tour recommendation and itinerary planning: A survey. *Artificial Intelligence Review*, 52, 405-439.
4. **[LLM Agents]** Xi, Z., Chen, W., Guo, X., et al. (2023). The Rise and Potential of Large Language Model Based Agents: A Survey. *arXiv preprint arXiv:2309.07864*.
5. **[Tool Use]** Schick, T., Dwivedi-Yu, J., Dessì, R., et al. (2023). Toolformer: Language Models Can Teach Themselves to Use Tools. *arXiv preprint arXiv:2302.04761*.
6. **[ReAct Logic]** Yao, S., Zhao, J., Yu, D., et al. (2023). ReAct: Synergizing Reasoning and Acting in Language Models. *International Conference on Learning Representations (ICLR)*.
7. **[LangGraph Framework]** LangChain AI. (2024). *LangGraph: Building Stateful, Multi-Actor Applications with LLMs*. Retrieved from https://langchain-ai.github.io/langgraph/
8. **[Browser Automation]** Browser-use. (2025). *Browser-use: Interface for AI Agents to Interact with the Web*. GitHub Repository. Retrieved from https://github.com/browser-use/browser-use
9. **[Map Data]** AutoNavi Software Co., Ltd. (2025). *Gaode Map (Amap) Web Service API Documentation*. Retrieved from https://lbs.amap.com/api/webservice/summary/
10. **[LLM Planning]** Zhang, Y., et al. (2024). ItiNera: Integrating Spatial Optimization with Large Language Models for Open-domain Urban Itinerary Planning. *arXiv preprint arXiv:2402.07204*.
11. **[Web Agents]** Zhou, S., et al. (2023). WebArena: A Realistic Web Environment for Building Autonomous Agents. *arXiv preprint arXiv:2307.13854*.