# PROJECT AND TEAM INFORMATION

## Project Title

|  |
| --- |
| **Optifly** : Optimizing Airline Route for Efficiency and cost Saving. |

## Student / Team Information

|  |  |
| --- | --- |
| Team Name: Tech Resolutions  Team Lead: Abhay Kanojia | **Tech Resolutions** |
| **ABHAY KANOJIA (Team Lead)**  student ID: 230115137  email: abhaykanojia51@gmail.com |  |
| **ANVESHA**  student ID: 230122228  email: dipirawat830@gmail.com |  |
| **TANMAY CHAUHAN**  student ID: 23011566  email: chauhantanmay253@gmail.com |  |

# PROPOSAL DESCRIPTION

## Motivation

|  |
| --- |
| The problem we want to solve is the inefficiency in finding the optimal route between two cities based on various factors like distance, cost, and layovers. This problem is important because it affects millions of travelers worldwide, causing frustration, wasted time, and increased costs. Our project aims to develop an intelligent route optimization system that provides personalized travel recommendations, taking into account individual preferences and real-time data. |

## State of the Art / Current solution

|  |
| --- |
| Currently, the problem of finding the optimal route between two cities based on various factors like distance, cost, and layovers is addressed by creating a graph representation of nodes (airports) and edges (flights). This graph provides a visual understanding of the distribution of airports and flights, particularly for major airports.  To solve this problem, the graph is first inserted into computer memory. Then, pathfinding algorithms are applied to determine the optimal route based on the user's input. This approach enables the identification of the most efficient route, taking into account various constraints such as layovers, flight availability, and time efficiency.  However, this current solution has limitations, and our project aims to improve upon it by developing a more efficient and scalable solution that can handle large datasets and provide personalized routes to users. |

## 

## Project Goals and Milestones

|  |
| --- |
| **Project Goals:**  **1. Develop an efficient route optimization algorithm:** Design and implement a route optimization algorithm that can find the optimal route between two cities based on various factors like distance, cost, and layovers.  **2. Create a user-friendly interface**: Develop a user-friendly interface that allows users to input their travel preferences and receive personalized route recommendations.  **3. Integrate real-time data feeds:** Integrate real-time data feeds from airlines and airports to provide accurate and up-to-date route information.  **4. Ensure scalability and performance**: Ensure that the system can handle a large volume of users and data while maintaining optimal performance.  **Initial Milestones:**  **1. Literature review and algorithm selection**: Conduct a literature review of existing route optimization algorithms and select the most suitable algorithm for the project.  **2. Data collection**: Collect and preprocess the necessary data for the project, including airport and flight information. |

## Project Approach

|  |
| --- |
| Our approach to solving the **OptiFly** problem involves a combination of data structures, algorithms, and software engineering techniques.   * **Data Collection and Preprocessing**   + Collect real-world dataset of airports and flight connections.   + Preprocess data to extract relevant information (e.g., airport codes, flight numbers, departure/arrival times) * **Graph Construction and Representation**   + Implement graph data structure using adjacency list or adjacency matrix   + Represent airports as nodes and flights as edges   + Assign weights to edges based on distance, cost, or time * **Shortest Path Algorithm Selection and Implementation**   + Select suitable shortest path algorithm (e.g., Dijkstra's, Bellman-Ford, Floyd-Warshall)   + Implement selected algorithm using priority queue (min-heap) for efficient computation   + Optimize algorithm for performance and scalability * **Route Reconstruction and Visualization**   + Implement route reconstruction function to extract optimal path from source to destination   + Visualize optimal route using suitable visualization tools (e.g., Matplotlib, Plotly) * **User Interface Development**   + Implement Command-Line Interface (CLI) or Graphical User Interface (GUI) for user interaction   + Allow users to input source and destination cities and display optimal route * Programming language: c++, python and its libraries. |

## 

## System Architecture (High Level Diagram)

|  |
| --- |
| Here's a high-level system architecture diagram for the OptiFly project:   1. +--------------------+ 2) +-------------------------------------+ 3) +---------------------------------------------+   | User Interface | 🡪 | Route Optimization Module | 🡪 | Graph Database (Airports, Flights) | 🡪  +--------------------+ +-------------------------------------+ +---------------------------------------------+  4) +----------------------------------------+ 5) +-----------------------------------------------------+  | Route Reconstruction Module | 🡪 | Output Module (Display Optimal Route) |  +----------------------------------------+ +-----------------------------------------------------+    This diagram shows the main components and interfaces of the OptiFly system:  **1. User Interface:** Allows users to input source and destination cities and display the optimal route.  **2. Route Optimization Module:** Implements the route optimization algorithm (e.g., Dijkstra's, Bellman-Ford, Floyd-Warshall) to find the optimal route.  **3. Graph Database:** Stores the graph representation of airports and flights.  **4. Route Reconstruction Module**: Reconstructs the optimal route from the source to the destination.  **5. Output Module:** Displays the optimal route to the user. |

## Project Outcome / Deliverables

|  |
| --- |
| **1. Optimal Route Finder**: A software system that can find the optimal route between two cities based on various factors like distance, cost, and layovers.  **2. Graph-Based Route Optimization:** An implementation of graph-based route optimization algorithms (e.g., Dijkstra's, Bellman-Ford, Floyd-Warshall) to find the shortest or cheapest route.  **3. User-Friendly Interface:** A user-friendly interface that allows users to input source and destination cities and display the optimal route. |

# Assumptions

|  |
| --- |
| 1. Travelers will provide accurate and reliable preference information.  2. The system will be designed and implemented using scalable and secure technologies. |

## References

|  |
| --- |
| 1."Route Optimization Algorithms: A Review" by A. K. Singh et al. (2020)  https://link.springer.com/content/pdf/10.1007/978-981-19-4606-6\_78.pdf?pdf=inline%20link  2. "Real-time Route Optimization using Machine Learning" by J. Lee et al. (2019)  https://yellow.systems/blog/real-time-route-optimization-with-ai  3. "Airline Route Optimization: A Survey" by S. K. Goyal et al. (2018)  https://academic.oup.com/iti/article/doi/10.1093/iti/liad026/7459776 |