IJIRIS: International Journal of Innovative Research in Information Security
Volume 12, Issue 08, November 2026
E-ISSN: 2349-7017
P-ISSN: 2349-7009
https://www.ijiris.com/archives
https://doi.org/10.26562/ijiris.2026.v1208.03

ORBIXA: A Web-Based Smart Travel Companion for Personalized Trip Discovery and Collaborative Planning

Dr.Mythili M
Associate Professor, Department of CSE (Cyber Security)
Vemana Institute of Technology, Bengaluru, India
mythili.m@vemanait.edu.in
Garadi Nandini, Manasa CA, G Charitha, Madhuri P
Department of ISE, Vemana Institute of Technology, Bengaluru, India
garadinandini01@gmail.com, manasaca15@gmail.com
gcharitha17@gmail.com, madhuripr15@gmail.com

Publication History
Manuscript Reference No: IJIRIS/RS/Vol.12/Issue08/NVIS10082
Research Article | Open Access | Double-Blind Peer-Reviewed| Article ID: IJIRAE/RS/Vol.12/Issue08/NVIS10082
Received: 02,October 2026, Revised: 14,October 2026, Accepted: 29,October 2026, Published Online: 25,November 2026
https://www.ijiris.com/volumes/Vol12/iss-08/03.NVIS10082.pdf
Article Citation:Dr.Mythili,Garadi,Manasa,Charitha,Madhuri(2026),ORBIXA: A Web-Based Smart Travel Companion for Personalized Trip Discovery and Collaborative Planning. IJIRIS: International Journal of Innovative Research in Information Security, Volume 12, Issue 08 of 2026 pages 390-394 https://doi.org/10.26562/ijiris.2026.v1208.03
BibTeX Key: Dr.Mythili@2026ORBIXA
IJIRIS papers should be cited as IJIRIS(International Journal of Innovative Research in Information Security, AM Publications, India 2026, ISSN 2349-7017, https://doi.org/10.26562/ijiris.2026.v1208.03 The journal's official abbreviation is IJIRIS. **ORCiD:** https://orcid.org/0009-0004-9398-7488 About the License: Copyright©2026 copyright by the authors. This article is an open access and license under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/)

Abstract: The process of planning a trip together requires much more than just selecting the destination. Travelers have to find common interest, schedule their trip, find fellow travelers, manage the finances and organize their conversations. ORBIXA is a travel companion website that integrates all of these processes in one easy-to-use place. The application integrates such functionality as destination recommendation, travel profile, preference-based trip recommendations, group's creation, invitation handling, instant messaging and money tracking. The application was implemented using Flask, SQL Alchemy, SQLite and Flask-Socket IO. Trip recommendations were made using an interpretable scoring algorithm, not a machine learning model trained by this data. In this paper, we review the system requirements, architecture, main algorithms, interface workflow and provide proof of implementation using source code of the project and screenshots from the application.

Index Terms: smart travel buddy, travel destination search, cooperative planning, travel recommendations, Flask, team communication, travel portal.

I. INTRODUCTION

A. Background and Motivation

Traveling can be organized using disconnected systems of online technologies. One can use one site to look at the destinations while using another app to arrange for company during the trip and yet another tool to record costs for shared expenses. Such a fragmented organization can work well for an individual but not for multiple people trying to organize a trip together. In ORBIXA, the problem is tackled through the linking of travel discovery and group coordination. The user will be able to define his interest, get destination information, formulate a travel plan, ask for joining the existing group and talk once the request has been approved. Instead of viewing the recommendation process as a black box, the system compares the user's preferences against the attributes of the trips.

B. Problem Statement

The problem here is the lack of a unified, simple environment that would enable people to find appropriate tours and to handle all the social and technical details that need to be sorted out for organizing the tours. An appropriate solution should provide for identity management, destination content, matching, membership control, and group messaging without the constant switching of services by users.

II. OBJECTIVES, SCOPE AND RELATED WORK

A. Objectives

These objectives include reducing fragmentation in the planning process, giving easy trip recommendations, offering a framework for organizing the group, and saving critical data about the trip in a relational database. The project will also try to ensure that the interface remains user-friendly even for those planning their first group trip.

B. Scope and Boundaries

Destination and places data are currently curate rather than sourced from a live places provider. Availability, payment, optimization, and training do not form part of the implementation at this time.
These constraints will be relevant in understanding the capabilities that the screenshots showcase.

IJIRIS: International Journal of Innovative Research in Information Security
Volume 12, Issue 08, November 2026
E-ISSN: 2349-7017
P-ISSN: 2349-7009
https://www.ijiris.com/archives
https://doi.org/10.26562/ijiris.2026.v1208.03

C. Literature Context

In digital tourism systems, the focus usually lies on destination information, accommodation search, routes, or bookings. Collaborative travel software has another task: meeting the requirements of several users and keeping decisions common throughout the course of a trip. ORBIXA is designed to be used as an accompanying system in planning and coordination and not as a booking solution. Preference-based recommendations are applicable in situations where there is no history of interaction with the system. Content-based methods are based on a comparison of user preferences with item features, whereas collaborative filtering needs history of many users' actions. ORBIXA utilizes several explicit features such as interests, preferred destinations, budget range, preferred month of travelling, and number of persons traveling together. Coordination in groups is also dependent on permissions. The group organizer must be allowed to check membership applications, and only authorized users can discuss trips privately. This requirement is implemented in the application by trip membership statuses and permission checks related to messages and expenses.

III. SYSTEM SPECIFICATION

A. User and Platform Requirements

Roles of users include Visitor, Registered Traveler, Trip Organizer, and Administrator. Visitors will be able to browse available destinations and trips. Registered Travelers will be able to manage their profiles and trips, as well as apply for membership. Organizers will be able to accept or deny requests and manage the statuses of their trips. The server-side requests and routes are handled by Python and Flask. The application objects are mapped to the database using Flask-SQLAlchemy with SQLite as the local default data storage engine. Flask-SocketIO enables the sending of messages to specified rooms. The passwords of the accounts are hashed using Werkzeug and session cookies hold the authentication status of the browsers. The non-functional aspects considered are readability, navigability, data integrity, authentication, maintenance, and invalid input handling.

B. Functional Modules

Authentication and Profile: User registration and sign-in, preferences storage, Destinations and destinations: Search for destination, administration of destination, Travel tips and destinations: Publishing trips, rating recommendations Membership: Request handling for joining, approving/rejecting by organizers, Group Coordination: Chat for approved members, tracking expenses

IV. PROPOSED SYSTEM AND METHODOLOGY

A. Overall Architecture

ORBIXA is designed using a multi-layered architecture of a web application. The front-end side of ORBIXA has an interface that allows users to register, create profiles, destinations pages, plan their trips, and organize group activities. Flask routes handle the requests sent to the server and authenticate or validate roles before executing any database queries. SQL Alchemy has been used to create models for all the entities and relationships used in the system. Socket IO works along with the regular HTTP request and response to broadcast the newly posted chat message to the relevant trip room.

B. Recommendation and Scoring Method

ORBIXA employs an easily understandable score to sort all eligible trips. ORBIXA will first filter out those trips which don't have availability, those trips which are added by the current user and trips which start in the past. ORBIXA will compare the stored profile with the destination and trip description for each of the trips. Two points are awarded per interest match, while a perfect destination match will receive five points. For instance, the budget aspect acknowledges budget trips up to 10,000, middle range trips over 10,000 but not more than 30,000, and high-end trips over 30,000 based on the total cost of the trip. The matching month gets two points while the matching group size gets one point. The algorithm used for scoring is deterministic in nature and can be easily explained to users. The scoring method does not prove that the recommendations generated by the system are the best possible and outperform other recommendation systems. Future research can determine if the chosen weight values correspond to traveler satisfaction.

C. Data Model

The User table consists of identity, password hash, biography, city, interests related to traveling, budget category, language, favorite places, month of traveling, and preferred group size. The Destination table includes information about the places which were published to be traveled. The Place table includes a relationship with the destination and is an attraction, restaurant, hotel, or activity. The Trip table consists of the name, destination, organizer, time period, maximum group size, budget, activities, description, and status. The Membership table connects the user with the trip and indicates whether the membership request is pending or accepted. The Message table includes information about trip-related messages, and the Expense table contains the amount of money, name, and submitter of expense.

D. Authentication and Authorization

All new accounts have their credentials verified based on the minimum standard of names, emails, and passwords. Passwords are kept encrypted rather than as clear text. The session login verifies the current user, and decorators secure any route. Trip organizers are verified before processing any request or modifying trip status. Approved membership is needed to post messages or add expenses.

V. IMPLEMENTATION AND APPLICATION WORKFLOW

A. Workflow Overview

Typical trip starts from traveler signing up and creating his profile. Traveler searches for places to go, looks at recommended or open trips and makes a choice between creating his own journey or applying for joining some other one. Organizer considers the request.

IJIRIS: International Journal of Innovative Research in Information Security
Volume 12, Issue 08, November 2026
E-ISSN: 2349-7017
P-ISSN: 2349-7009
https://www.ijiris.com/archives
https://doi.org/10.26562/ijiris.2026.v1208.03

After the confirmation process, members can communicate with each other and log their expenses in a group.

B. Group Creation and Membership

A registered user can create a trip that includes all of the following details: destination, title, date range, capacity, activities, and budget (optional). The system ensures that the dates are valid, the number of people in the group is between 2 and 30, and the budget is non-negative. The person who creates the trip is automatically considered a validated user. Other users apply to be part of the trip.

C. Messaging and Shared Expenses

The approved members have the ability to see the trip conversation and leave posts. The approved posts get saved in the database and broadcast to the trip SocketIO room. An individual must first become a member before accessing the room. Similarly, the expense posts need an approved member and must contain a positive number.

D. Destination Curation

Destinations can be published and the places can be stored categorically. The initial database would contain places such as Goa, Manali, Jaipur, and Munnar.

VI. RESULTS, OBSERVATIONS AND EVALUATION PLAN

A. Observations from the Available Prototype

Screenshots illustrate a consistent flow of processes starting with creating an account up until the creation of trips. Join request checking, group messages, and expense tracking are additional features included in the code but not depicted in the selected screenshots. Taken together, the provided interface examples and source files demonstrate that ORBIXA has the basic components of a travel planning collaboration prototype.

The screenshots do not show any improvement in terms of numbers in planning times, recommendation accuracy, user satisfaction, or server processing rate. This information would have needed to be collected by conducting a controlled experiment with documented participants. It is for this reason that the evaluation is provided as implementation evidence.

B. Evaluation

The attached screens show account creation, profile editing, destination discovery, and trip creation.

C. Limitations

The recommendation score uses explicit preferences and fixed weights, so its relevance depends on profile completeness. Destination and place listings are curated rather than live verified. Booking, payments, full expense settlement, and learned recommendation models are not implemented. Production use would require security review, stronger deployment controls, privacy safeguards, and broader device testing.

VII. APPLICATIONS, FUTURE ENHANCEMENTS AND CONCLUSION

A. Future Enhancements

Further work could connect a live places API, map-based itineraries, notifications, verified booking links, improved localization, and expense settlement. A consent-based user study could measure recommendation relevance, task completion time, and satisfaction without assuming improvements in advance.

B. Conclusion

ORBIXA brings together destination discovery, preference-aware trip suggestions, group formation, real-time communication, and expense records. Its transparent matching method and integrated workflow provide a useful foundation for continued development and empirical evaluation.

C. ORBIXA Application Output Screenshots

The registration screen collects a traveler's name, email, password, and confirmation. It establishes the starting point for a personal travel account. The corresponding server logic validates the submitted information and stores a password hash.

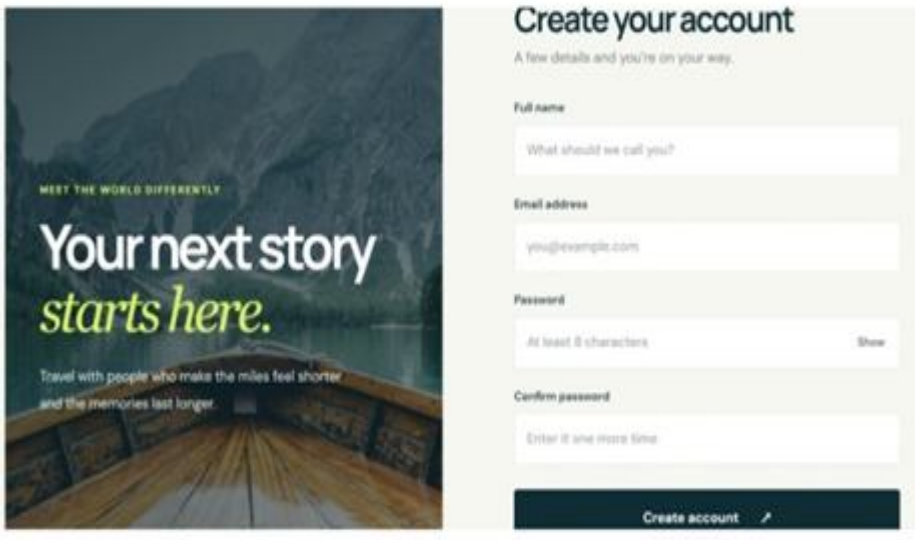


Fig.1. Account registration.

The profile page allows users to describe themselves and record travel interests and home city. These details provide context for trip discovery and personalization. A linked area displays trips in which the traveler participates.

IJIRIS: International Journal of Innovative Research in Information Security
Volume 12, Issue 08, November 2026
E-ISSN: 2349-7017
P-ISSN: 2349-7009
https://www.ijiris.com/archives
https://doi.org/10.26562/ijiris.2026.v1208.03

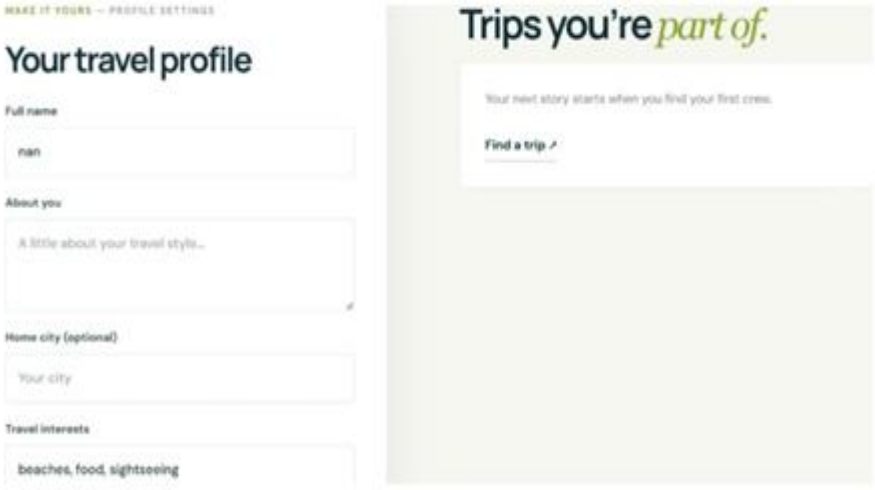


Fig.2. Traveler profile.

Destination cards present visually distinct places, including Goa, Manali, and Jaipur. The interface makes exploration more approachable by pairing location names with imagery and broad travel themes. The records come from curated application data.

Fig.3. Destination browsing

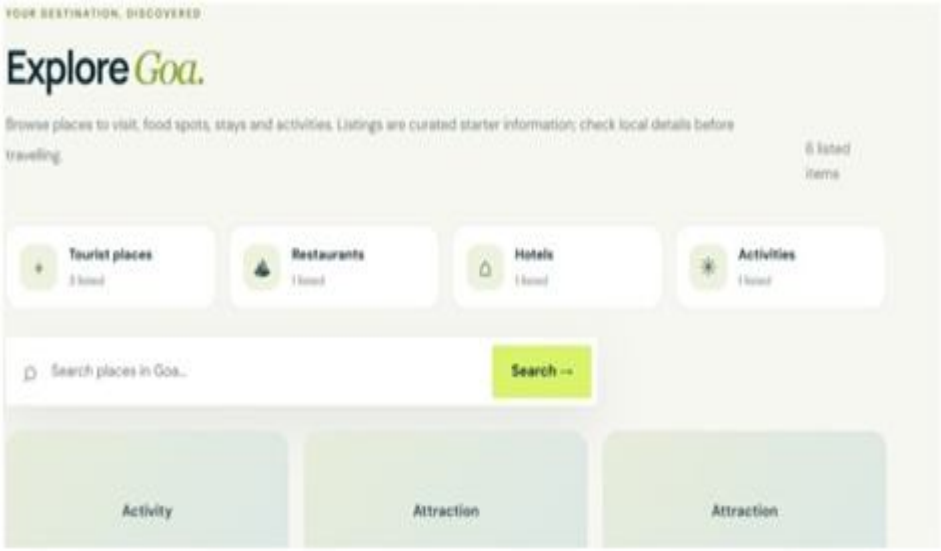


Fig.4. Destination Explorer

The Goa explorer organizes local content into tourist places, restaurants, hotels, and activities. Category controls and a search field help narrow the displayed listings. These entries are curated examples rather than live provider results.

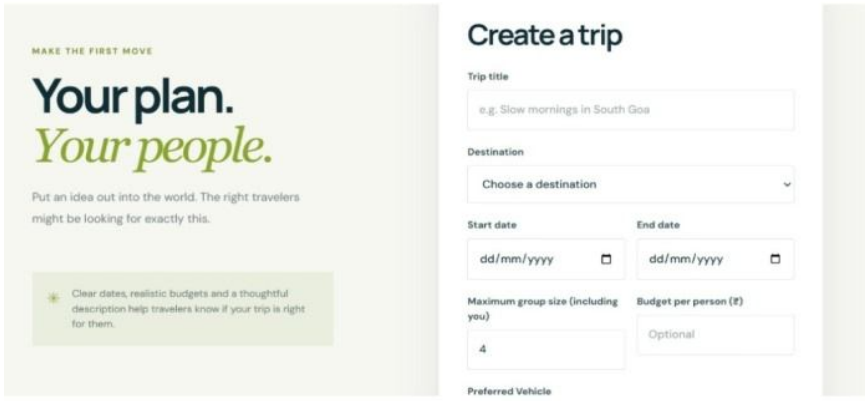


Fig.5. Trip creation form

The form captures a trip title, destination, dates, group capacity, and budget. It gives the organizer a structured way to publish an upcoming journey. Server-side checks prevent invalid dates and unsupported capacity values.

V. AUTHOR CONTRIBUTION STATEMENT

Conceptualization: Dr.Mythili M
Literature Review and Methodology design: Garadi Nandini, Manasa CA
Validation: G Charitha, Madhuri P
Investigation: Garadi Nandini, Manasa CA
Resources: Dr.Mythili M
Writing original draft preparation: Garadi Nandini, Manasa CA
Writing review and Editing: G Charitha, Madhuri P
Supervision: Dr.Mythili M
Conflict of interest: The authors declare no conflicts of interest.

Data availability statement: Data supporting these findings are available within the article, at https://doi.org/10.26562/ijiris.2026.v1208.03, or upon request.

Supplementary materials: The supplementary materials are available at https://doi.org/10.26562/ijiris.2026.v1208.03

Publisher's note: AM Publications, India. IJIRIS (International Journal of Innovative Research in Information Security) Journals stays neutral with regard to jurisdictional claims in published maps and institutional affiliations. All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher. https://ampublications-india.com/

REFERENCES

1. M. Grinberg, Flask Web Development: Developing Web Applications with Python, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2018.
2. Bayer, "SQLAlchemy," in The Architecture of Open Source Applications, vol. 2, A. Brown and G. Wilson, Eds. 2012.
3. M. Grinberg, "Flask-SocketIO documentation," Read the Docs. Available: https://flask-socketio.readthedocs.io/.
4. R. D. Hipp, "SQLite documentation," SQLite. [Online]. Available: https://www.sqlite.org/docs.html.
5. Pallets Projects, "Werkzeug documentation: Security helpers," Available: https://werkzeug.palletsprojects.com/
6. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in Recommender Systems Handbook. Boston, MA, USA: Springer, 2011, pp. 1–35.
7. P. Lops, M. de Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in Recommender Systems Handbook. Boston, MA, USA: Springer, 2011, pp. 73–105.
8. Abhinav,Archit,Nikita(2025),A Study on Arduinouno Based Women Safety. IJIRAE:: International Journal of Innovative Research in Advanced Engineering, Volume 12, Issue 03 of 2025 pages 161-167 https://doi.org/10.26562/ijirae.2025.v1204.06
9. Basith,Sridhar,Gokul,Krithikdas,Mohamed(2025),Enhancing Leaf Disease Detection in Short – Statured Crops using a Smart Agribot with Sensor Fusion and Image Analysis . IJIRAE:: International Journal of Innovative Research in Advanced Engineering, Volume 12, Issue 03 of 2025 pages 168-181 https://doi.org/10.26562/ijirae.2025.v1204.07
10. Dr.Shilpa,Chethan,Jai,Vishwas,Kushala (2025), A Machine Learning Approach to Crop Recommendation, Plant Disease Identification, and Yield Estimation. IJIRAE:: International Journal of Innovative Research in Advanced Engineering, Volume 12, Issue 03 of 2025 pages 182-194 https://doi.org/10.26562/ijirae.2025.v1204.08
11. Jagadamba,Meghana,Dharani,Hasya,Spurthi(2025), Smart Mirror using Raspberry Pi. IJIRAE: International Journal of Innovative Research in Advanced Engineering, Volume 12, Issue 03 of 2025 pages 195-200 https://doi.org/10.26562/ijirae.2025.v1204.09
12. Ashwini,Bhoomika,Leena,Kavya,Harshini(2025),AgriReach: IJIRAE: International Journal of Innovative Research in Advanced Engineering, Volume 12, Issue 03 of 2025 pages 201-205 https://doi.org/10.26562/ijirae.2025.v1204.10