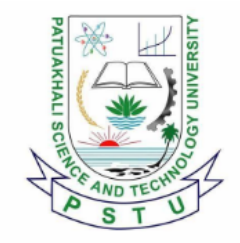

Patuakhali Science and Technology University
Faculty of Computer Science and Engineering

CCE 310 :: Software Development Project-I

Sessional Project Proposal

Project Title : Linkerine
Submission Date : Fri 18, July 2025

Submitted from,	**Submitted to,**
Md. Sharafat Karim **ID** : 2102024, **Reg** : 10151, **Semester** : 5 (Level-3, Semester-1)	1. **Prof. Dr. Md Samsuzzaman** Professor, Department of Computer and Communication Engineering, Patuakhali Science and Technology University. 2. **Sarna Majumder** Associate Professor, Department of Computer and Communication Engineering, Patuakhali Science and Technology University.

Contents

Linkerine

1. Abstract

Linkerine is a cross-platform bookmark management application designed to help users seamlessly store, organize, and sync **bookmarks**, **articles**, **notes**, **passwords**, and **RSS feeds** across multiple devices. In an age of increasing digital information overload, Linkerine aims to provide a simplified and powerful interface to manage and retrieve valuable resources from anywhere.

2. Objectives

- To build a unified, cross-platform system to organize bookmarks, articles, and RSS feeds.
- To provide secure sync and storage for user data including notes and passwords.
- To implement intuitive features like nested folders, duplicate URL detection, powerful search, and anonymous usage.
- To offer a seamless experience across Android, iOS, and web platforms.

3. Problem Statement

Existing solutions for bookmark and content management are either limited to specific platforms, paid-only, lack Android support, or provide limited functionality. Furthermore, most do not support features like anonymous mode, password storage, nested folders, and offline-first architecture. There is a need for a comprehensive, user-friendly tool that works everywhere and adapts to user privacy and organization needs.

4. Related Work

- [1] **Mozilla Pocket** - Popular read-it-later app, but now has been **deprecated** (as of July 8, 2025).
- [2] **Goodlinks** - Paid and available only on macOS.
- [3] **Futurwise** - Chromium-only extension. Also mobile apps are not available.
- [4] **Instapaper** - Focused on reading but lacks strong organization.
- [5] **Matter** - No Android support, limited availability.
- [6] **Raindrop.io** - Feature-rich but lacks anonymous mode and some sync customizations.
- [7] **Readwise** - Premium-focused and still in beta.

5. Methodology

The development of Linkerine will follow an agile methodology, allowing for iterative improvements and user feedback. Our technology stack will includes,

Frontend	Flutter[8] (Dart)
Backend	Supabase
UI Design	Material Design (android), Cupertino (IOS)
Database	PostgreSQL
Authentication	JWT, OAuth2
Hosting (web app)	Vercel or Netlify
CI/CD	GitHub Actions, Docker

6. Visual Models

6.1. Flow Chart Diagram

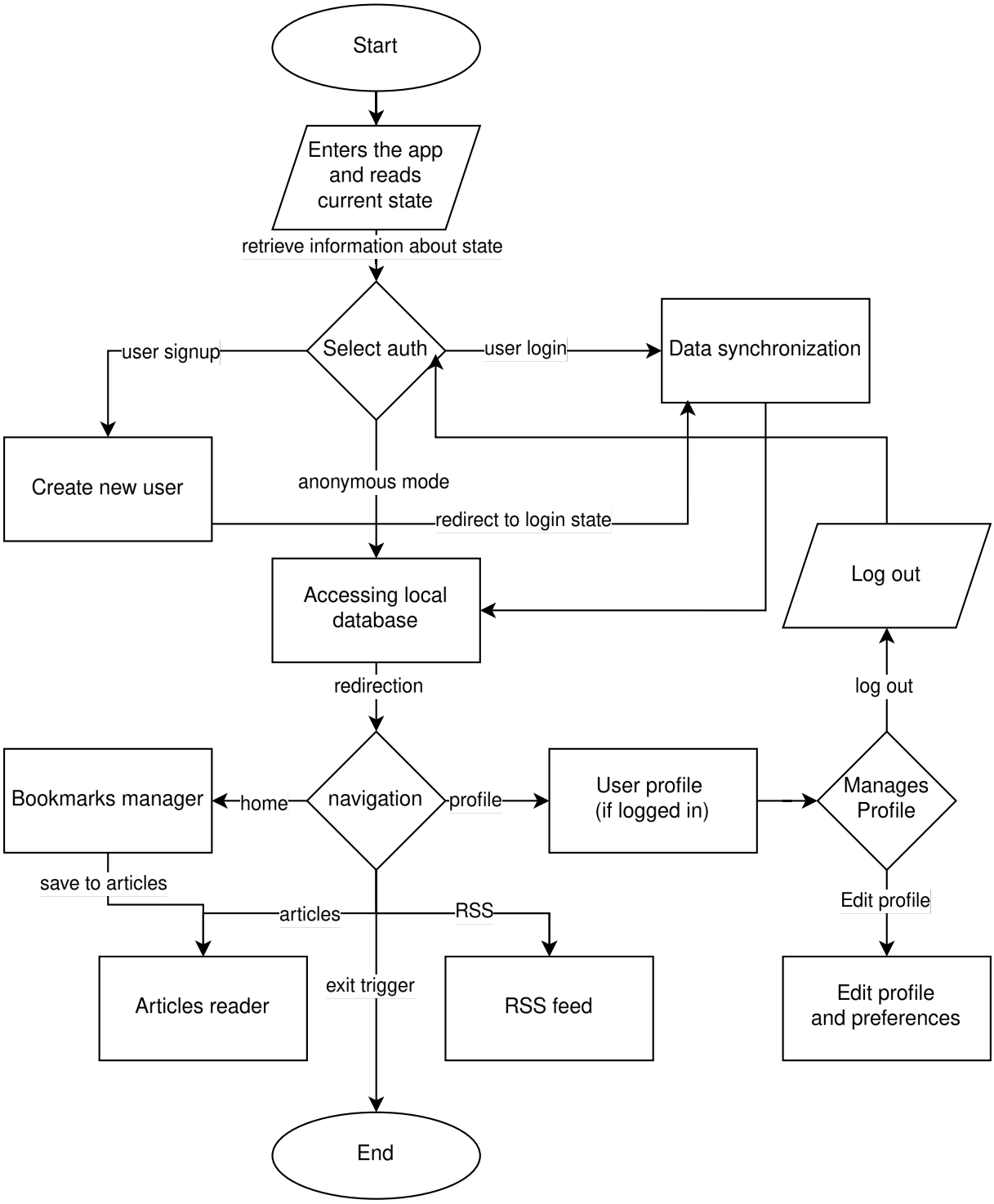

Figure 1: Flow Chart of Linkerine Architecture

Here the above Figure 1 illustrates the high-level architecture of Linkerine, showing how different components interact with each other. Mainly the frontend communication is portrayed here.

In the frontend part, the user interacts with the application, which communicates with the backend server. The backend server handles requests, processes data, and interacts with the database for storage and retrieval of bookmarks, articles, notes, passwords, and RSS feeds.

6.2. ERD (Entry Relationship Diagram)

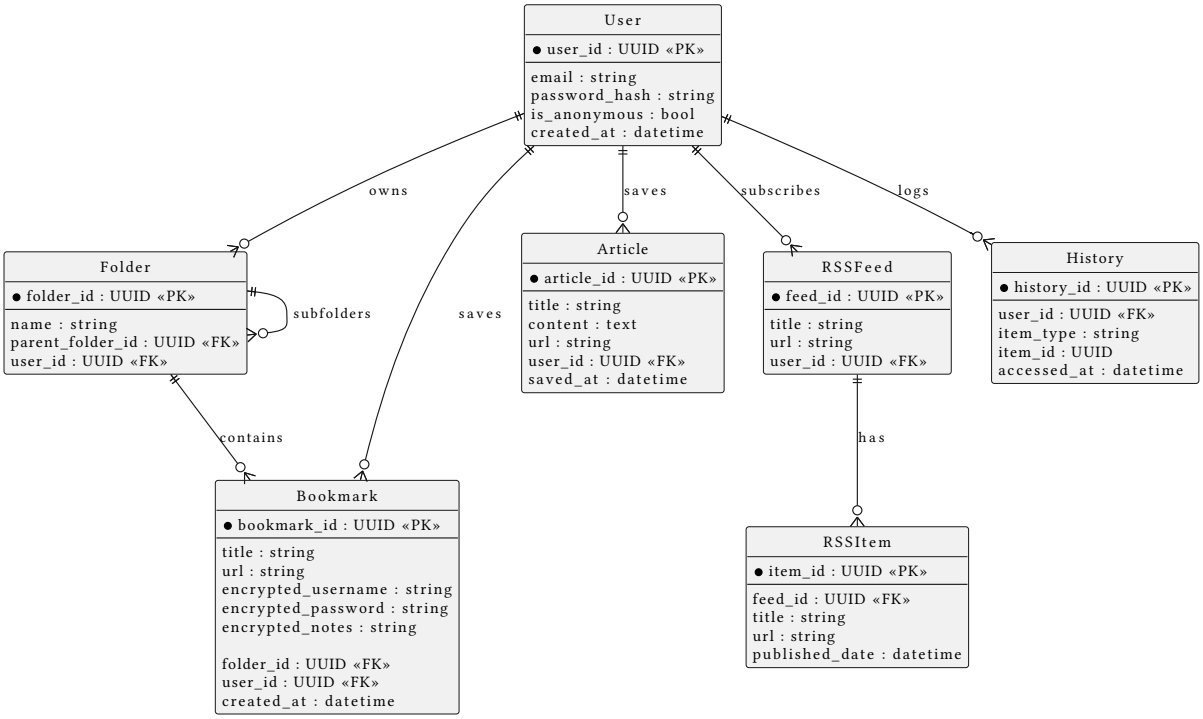

Figure 2: Entity Relationship Diagram of Linkerine

Here the above Figure 2 illustrates the backend database structure of Linkerine, showing the relationships between different entities such as users, bookmarks, articles, notes, passwords, and RSS feeds.

6.3. Timeline

The base timeline for the development of Linkerine is as follows,

Task	**Week 1-2**	**Week 3-4**	**Week 5-6**	**Week 7-8**	**Week 9**	**Week 10**	**Week 11**	**Week 12**
Requirements & UI Mockup	✓		✓					
Authentication + Supabase DB		✓	✓					
Bookmark CRUD + Nested Folders			✓	✓				
Article Reader + RSS Parsing				✓				
History + Search Module			✓	✓	✓			
Duplicate Removal & Testing						✓		
UI Polish & Documentation						✓	✓	
Final Testing & Deployment								✓

6.4. UI Mockups

Figure 3: UI Mockups of Linkerine

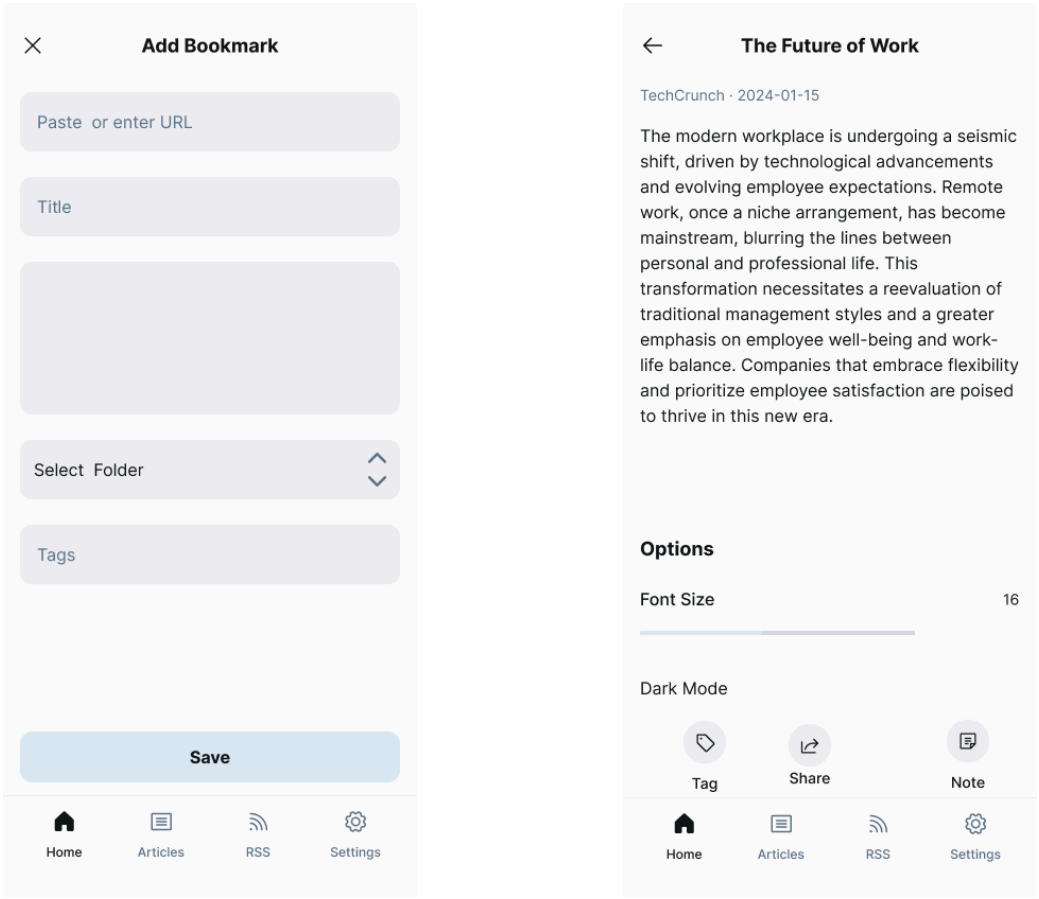

Figure 4: Add Bookmark page and reader page's UI concept

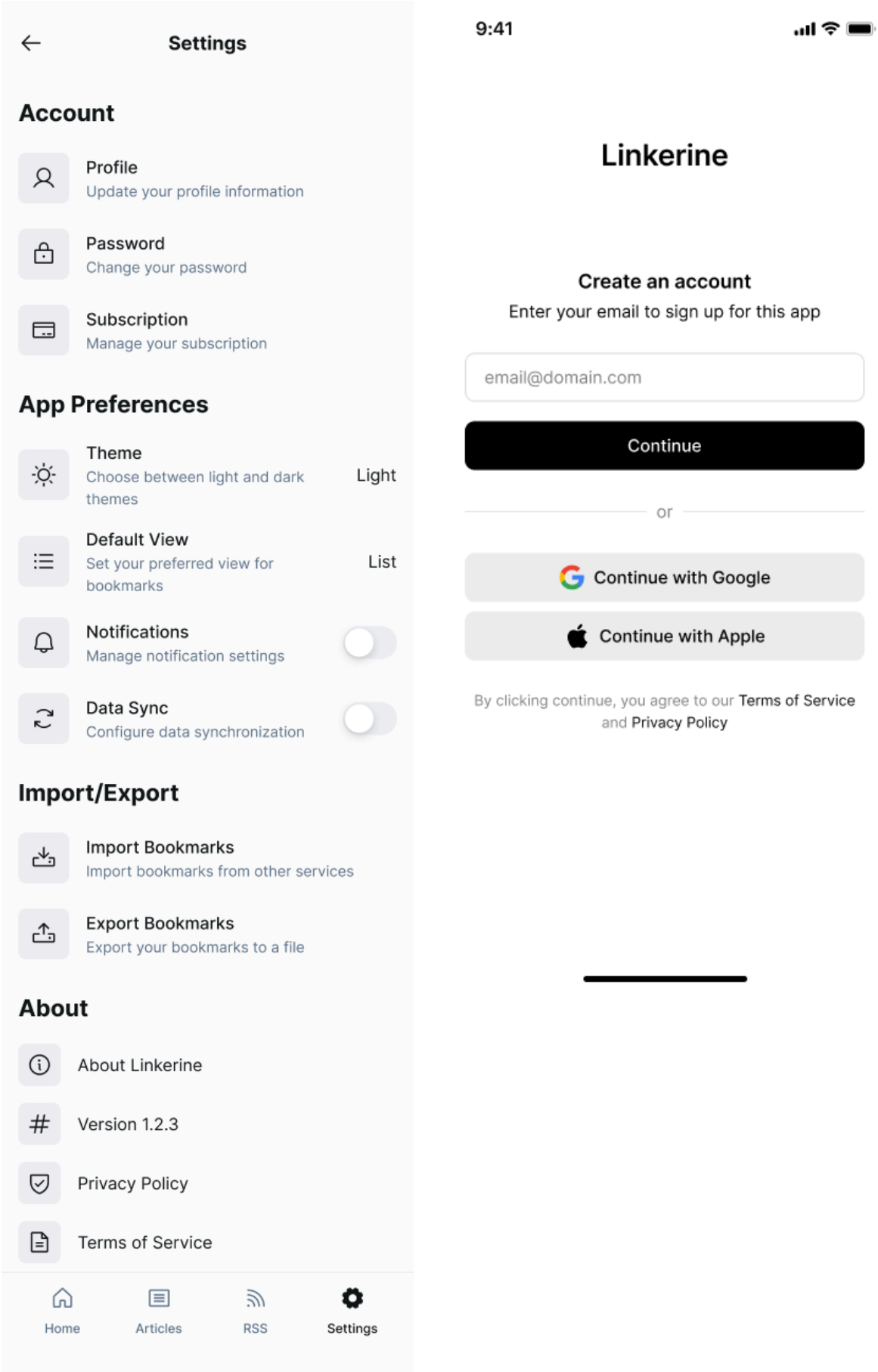

Figure 5: Settings and sign in page's UI concept

7. Scope

Linkerine will be developed as a cross-platform application using Flutter[8], allowing it to run on Android, iOS, and web browsers. The application will focus on providing a rich set of features for managing bookmarks, articles, notes, passwords, and RSS feeds. It will also include advanced functionalities like nested folders, duplicate URL detection, and a powerful search engine.

Later on, it can be extended to include URL shortening, generating QR codes, importing or exporting bookmarks with different browsers compatibiolity, sharing bookmarks globally, tagging and so on...

8. Limitations

1. Currently browser extensions are not planned, but the application will be designed to allow easy integration with popular browsers in the future.
2. The initial version will focus on core functionalities and may not include all advanced features like URL shortening or QR code generation.

9. Result

The expected outcome of Linkerine is a fully functional, cross-platform application that allows users to manage their digital resources efficiently.

Bibliography

[1] "Mozilla Pocket." Accessed: Jul. 18, 2025. [Online]. Available: https://getpocket.com/

[2] "GoodLinks." Accessed: Jul. 18, 2025. [Online]. Available: https://goodlinks.app/

[3] "Futurwise." Accessed: Jul. 18, 2025. [Online]. Available: https://www.futurwise.com/

[4] "Instapaper." Accessed: Jul. 18, 2025. [Online]. Available: https://instapaper.com/

[5] "Matter." Accessed: Jul. 18, 2025. [Online]. Available: https://hq.getmatter.com/

[6] "Raindrop.io." Accessed: Jul. 18, 2025. [Online]. Available: https://raindrop.io/

[7] "Readwise." Accessed: Jul. 18, 2025. [Online]. Available: https://readwise.io/

[8] "Flutter." [Online]. Available: https://flutter.dev/

THE END