Project Proposal – Web-Tech – II

By Reyan Tariq - 2112231

SWEEP is a digital platform designed to bridge the gap between students seeking practical work experience and employers looking to delegate tasks. By connecting students with micro-tasks from organizations, SWEEP enables skill development while providing employers with cost-effective solutions. Students earn "Sweep Tokens" as rewards, creating a ecosystem that incentivizes participation. Built on React (frontend) and .NET (backend), SWEEP aims to revolutionize entry-level work experience acquisition.

Objectives

We wish to create a secure, scalable web application for task delegation and student-employer interaction, where we implement distinct workflows for students and employers. Employers will be able to post, manage, and approve tasks with customizable requirements, employers can send a 'task-request' to students that fit their needs best. Upon successful completion of tasks, students will be awarded SWEEP tokens that they can mention in their resume as valid work experience.

Features

- Student Profile
- Employer Profile
- Token Balance
- Task Creation -> Completion Workflow
- Authentication Mechanisms
- Role Based Access
- Intuitive UI/UX

Tech Stack

- React
- .NET
- SQL Server
- Zustand
- Clerk

Outcomes

1. **Student Benefits**:

   o   Gain real-world experience and token-based credentials.

   o   Improve employability through verified task completion.

2. **Employer Benefits**:

   o   Reduce workload for full-time staff with vetted student talent.

   o   Cost-effective task outsourcing.

Conclusion

SWEEP addresses critical gaps in student employability and organizational efficiency. By gamifying work experience through a tokenized system, it creates a win-win ecosystem for students, employers, and educational institutions. With a robust technical foundation and clear roadmap, SWEEP has the potential to become the go-to platform for micro-task outsourcing and skill development.

Submitted By: Reyan Tariq – 2112231

Date: 5/7/2025

Entity Relationship Diagram

User
int	UserId	PK
string	Email	
byte[]	PasswordHash	
byte[]	PasswordSalt	
string	ClerkId	
UserRole	Role	
DateTime	CreatedAt	
DateTime	UpdatedAt	

RefreshToken
int	TokenId	PK
int	UserId	FK
string	Token	
DateTime	ExpiresAt	
DateTime	CreatedAt	
DateTime	RevokedAt	
string	ReplacedBy	

Field
int	FieldId	PK
string	Name	

EmployerProfile
int	UserId	PK,FK
string	EmployerName	
string	Organization	

StudentProfile
int	UserId	PK,FK
string	StudentName	
string	University	
decimal	CGPA	
int	GraduationYear	
int	TokenBalance	
int	PriorExperienceYears	
int	FieldId	FK

JobTask
int	JobTaskId	PK
int	UserId	FK
string	Title	
string	Description	
DateTime	Deadline	
bool	RequiresExperience	
Complexity	Complexity	
decimal	MonetaryCompensation	
int	FieldId	FK
DateTime	CreatedAt	
DateTime	UpdatedAt	

TaskAssignment
int	AssignmentId	PK
int	JobTaskId	FK
int	UserId	FK
DateTime	RequestedAt	
DateTime	AcceptedAt	
DateTime	CompletedAt	
DateTime	ApprovedAt	
int	TokensAwarded	

TaskAttachment
int	AttachmentId	PK
int	JobTaskId	FK
string	FileUrl	
string	FileName	
string	FileType	
DateTime	UploadedAt	

Relationships: User has RefreshToken, User has StudentProfile, User has EmployerProfile, Field belongs to StudentProfile, Field categorized by JobTask, EmployerProfile creates JobTask, StudentProfile requests TaskAssignment, JobTask assigned to TaskAssignment, JobTask has TaskAttachment