1.Introduction

Project Tittle:Cookbook your Virtual kitchen Assistant

Team Members**:**

**Team Leader Name: SALINI G**

**Team Member: SANMAKI D**

**Team Member: SANTHIYA K**

**Team Member: SARAVANAPRIYA S**

2.Project Overview

Purpose:

The CookBook applications is designed to simply meal preparation by offering a virtual kitchen assistant

User can browse, save and manage their favorite recipes with detailed instructions and ingredients lists

Features:

-Extensive Recipe Library with Search&Filter option

-step-by-step Cooking Instructions

-Bookmark features for favorite recipes

-Light/Dark Mode for enhanced virtual Comfort

-responsive Design for optimal viewing on all Devices

3.Architectures

Component structure:

App.js- Root component managing layout and routes

Header. js-Navigation bar with search functionality

RecipeList.js-Display categorized recipes cards

RecipesCard.js-Detailed view of selected recipes

Footer.js-Detailed view of selected recipes

State Management:Context API

Routing:React Router

4.Setup Instructions

Prerequisites:

-Node.js (v18 Or higher)

-npm Or yarn

Installation:

1.Install Dependencies:

‘npm install'

2.create.env file:

‘REACT\_APP\_RECIPE\_API\_KEY=your\_api\_key\_here'

3.Run the Application:

‘npm start'

5.Folder Structure

/src

|--/components

| |--Header.js

| |--RecipeList.js

| |--Recipecard.js

| |--RecipeDeatail.js

| |--Folder.js

|-----/pages

| |---home.js

| |---Categories.js

| |---Bookmark.js

|-----/asset

| |---image

| |---iconic

| |---styles

|-----/utils

| |---api.js

| | ---helpers.js

| |---constant.js

|-----App.js

|-----index.js

6.Running the application

Fronted server Command:’npm start’

7.Component Documentation

Key Component:

Header:provide navigation and search functionality

RecipeList:Display available recipes categorized by type

RecipeCard:Display individual recipe details

RecipeDetails: Offers step-by-step instructions

Reusable Component:

Button component:used for action like ‘View Recipes’ or ‘Bookmark’

Model component: Display detailed recipe information

8.State Management

Global state:Managed via Context API

Local state:Managed using React ‘useState()’

9.User Interface

- Clean and intuitive layout optimizes for smooth navigation

-Mobile- first for design for seamless browsing

-Elegant Light/Dark mode toggle for improved accessibility

10.styling

CSS Framework/Libraries:Tailwind CSS

Theming:Integrated light/dark mode

11.Testing

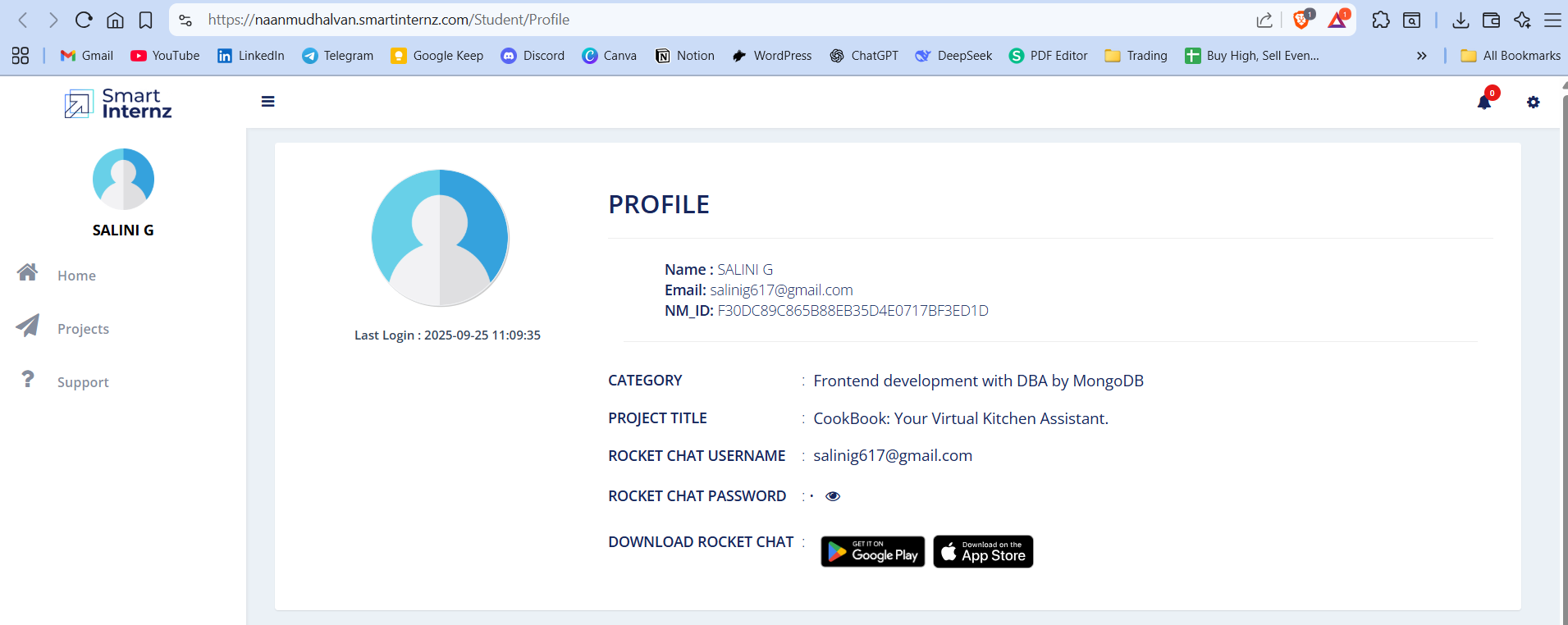
Testing strategy:

Jest for unit testing

React Testing Library for component testing

Code coverage:Ensured high test coverage with jest reports

12.Screenshort Or Demo video



13.Futures Enhancements

-AI-Based Recipes suggestions

-user Authentication for personalized recipes collection

- Integration with Grocery shopping platforms

-offline mode for accessing saved recipes