

CB23332–SOFTWARE ENGINEERING–LAB MANUAL

S.No.	Date	INDEX	Page No.	Signature
1		Writing the complete problem statement		
2		Writing the software requirements specification document		
3		Entity relationship diagram		
4		Data flow diagrams at level 0 and level 1		
5		Use case diagram		
6		Activity diagram of all use cases.		
7		State chart diagram of all use cases		
8		Sequence diagram of all use cases		
9		Collaboration diagram of all use cases		
10		Assign objects in sequence diagram to classes and make class diagram		
11		Sample Code and screenshots		

EXERCISE NO. 1

AIM: To prepare a **Problem Statement** for a *startup navigator app*.

ALGORITHM:

1. Define the problem clearly, focusing on the challenges faced by young entrepreneurs in launching startups.
2. Provide a high-level, non-technical description of the problem.
3. Emphasize the need for improvement in the current process of launching startups.
4. Highlight the impact of inefficient startup planning on various stakeholders (Entrepreneurs, Investors, Advisors).
5. Ensure the problem statement addresses what needs to be done, not how it will be done.

INPUT:

- Issues with the current process of startup planning and execution.
- Broad expectations for the new system from stakeholders (Entrepreneurs, Investors, Advisors, Mentors).
- Initial requirements gathered from key users.
- Overview of existing challenges in launching startups.

Problem:

Young entrepreneurs face challenges in refining ideas, conducting market research, securing funding, and managing legal requirements. The decentralized nature of current resources leads to inefficiencies, making it harder to access the necessary support at each stage of the startup journey. This fragmentation increases the risk of failure, as entrepreneurs lack a unified platform for guidance, leading to lost time and missed opportunities.

Background:

The entrepreneurial ecosystem has seen exponential growth in recent years, with many aspiring founders eager to bring their ideas to life. However, the startup landscape is complex, requiring founders to navigate through idea validation, business model creation, funding strategies, and market entry plans. Existing support systems for entrepreneurs, such as incubators and mentorship programs, are often fragmented, leading to confusion and missteps during critical stages of a startup's development.

Without a structured guide, many entrepreneurs face significant challenges such as:

- Lack of actionable steps to move from idea to execution.
- Limited access to funding, legal advice, and market research.
- Inconsistent mentorship and fragmented resources across various platforms.

A **Startup Navigator App** can address these challenges by offering a centralized platform that provides entrepreneurs with personalized roadmaps, access to resources, and real-time guidance to launch their startups more effectively.

Relevance:

Effective planning and execution are critical for startup success. A centralized system will:

- Provide entrepreneurs with a structured roadmap tailored to their startup's specific stage and industry.
- Enable entrepreneurs to access market research, funding resources, and legal advice in one place.
- Facilitate collaboration between entrepreneurs, mentors, and investors, improving the flow of information and support.
- Reduce the risk of failure by offering clear, actionable steps and tracking progress in real-time.

This solution will streamline the startup journey, improving the chances of success for young entrepreneurs while enhancing communication between stakeholders.

Objectives:

The primary objective of this project is to develop a **Startup Navigator App** that streamlines the planning, development, and execution phases for startups. The app will provide a comprehensive roadmap and access to critical resources tailored to the entrepreneur's unique needs. Specific objectives include:

1. **For Entrepreneurs:**
 - Allow entrepreneurs to input their startup idea and stage to generate a customized roadmap.
 - Provide tools for market research, funding options, legal guidance, and business development strategies.
 - Implement a user-friendly interface that enables entrepreneurs to modify their plans and track progress.
 - Offer automatic reminders for critical milestones and deadlines.
2. **For Investors and Mentors:**
 - Enable mentors and investors to offer guidance and support through the app, providing consistent advice at each stage.
 - Create a communication system that allows entrepreneurs to connect with mentors and investors.
 - Provide tools for tracking the progress of startups, offering real-time insights and feedback.
3. **For Administrators (Incubators or Business Support Teams):**
 - Allow administrators to manage multiple startups simultaneously, providing insights into their development stages.
 - Create dashboards that offer a clear view of startup progress, enabling better resource allocation and support services.
 - Enable admins to provide real-time feedback, resolve challenges, and guide startups toward success.
4. **General:**

- Develop a centralized database to store startup details, user information, and resources for quick retrieval.
- Implement a robust notification system for all stakeholders to stay updated on progress and important deadlines.
- Ensure the system is scalable, allowing for multiple startups and users without performance issues.
- Offer a reporting system that generates summaries for post-launch evaluation and long-term success tracking.

Here's an altered version of the problem statement for a **Startup Navigator App**:

Problem Statement:

Title:

Development of a Startup Navigator App to Assist Entrepreneurs in Launching Their Startups Successfully

Aim:

To prepare a problem statement for a mobile application called the **Startup Navigator App**, which provides tailored guidance, resources, and a structured roadmap to help young entrepreneurs effectively launch and grow their startups.

Algorithm:

1. Define the problem clearly, focusing on the challenges faced by young entrepreneurs in launching startups.
 2. Provide a high-level, non-technical description of the problem.
 3. Emphasize the need for improvement in the current process of launching startups.
 4. Highlight the impact of inefficient startup planning on various stakeholders (Entrepreneurs, Investors, Advisors).
 5. Ensure the problem statement addresses what needs to be done, not how it will be done.
-

Input:

- Issues with the current process of startup planning and execution.
- Broad expectations for the new system from stakeholders (Entrepreneurs, Investors, Advisors, Mentors).
- Initial requirements gathered from key users.
- Overview of existing challenges in launching startups.

Problem:

Young entrepreneurs often struggle with various aspects of launching a startup, such as refining their business idea, performing market research, securing funding, and navigating legal challenges. The current process is decentralized, requiring entrepreneurs to seek support from different platforms or advisors at each stage of their startup journey. This fragmentation leads to inefficiencies in planning, difficulty accessing resources, and challenges in managing progress. Entrepreneurs lack a unified platform to guide them through the stages of launching a startup, resulting in lost time, missed opportunities, and increased chances of failure.

Stakeholders such as investors, advisors, and mentors also face difficulties in providing consistent guidance and support to entrepreneurs due to the lack of a centralized system. These inefficiencies hinder collaboration, making it harder for entrepreneurs to receive the right advice at the right time.

Background:

The entrepreneurial ecosystem has seen exponential growth in recent years, with many aspiring founders eager to bring their ideas to life. However, the startup landscape is complex, requiring founders to navigate through idea validation, business model creation, funding strategies, and market entry plans. Existing support systems for entrepreneurs, such as incubators and mentorship programs, are often fragmented, leading to confusion and missteps during critical stages of a startup's development.

Without a structured guide, many entrepreneurs face significant challenges such as:

- Lack of actionable steps to move from idea to execution.
- Limited access to funding, legal advice, and market research.
- Inconsistent mentorship and fragmented resources across various platforms.

A **Startup Navigator App** can address these challenges by offering a centralized platform that provides entrepreneurs with personalized roadmaps, access to resources, and real-time guidance to launch their startups more effectively.

Relevance:

Effective planning and execution are critical for startup success. A centralized system will:

- Provide entrepreneurs with a structured roadmap tailored to their startup's specific stage and industry.
- Enable entrepreneurs to access market research, funding resources, and legal advice in one place.
- Facilitate collaboration between entrepreneurs, mentors, and investors, improving the flow of information and support.

- Reduce the risk of failure by offering clear, actionable steps and tracking progress in real-time.

This solution will streamline the startup journey, improving the chances of success for young entrepreneurs while enhancing communication between stakeholders.

Objectives:

The primary objective of this project is to develop a **Startup Navigator App** that streamlines the planning, development, and execution phases for startups. The app will provide a comprehensive roadmap and access to critical resources tailored to the entrepreneur's unique needs. Specific objectives include:

1. For Entrepreneurs:

- Allow entrepreneurs to input their startup idea and stage to generate a customized roadmap.
- Provide tools for market research, funding options, legal guidance, and business development strategies.
- Implement a user-friendly interface that enables entrepreneurs to modify their plans and track progress.
- Offer automatic reminders for critical milestones and deadlines.

2. For Investors and Mentors:

- Enable mentors and investors to offer guidance and support through the app, providing consistent advice at each stage.
- Create a communication system that allows entrepreneurs to connect with mentors and investors.
- Provide tools for tracking the progress of startups, offering real-time insights and feedback.

3. For Administrators (Incubators or Business Support Teams):

- Allow administrators to manage multiple startups simultaneously, providing insights into their development stages.
- Create dashboards that offer a clear view of startup progress, enabling better resource allocation and support services.
- Enable admins to provide real-time feedback, resolve challenges, and guide startups toward success.

4. General:

- Develop a centralized database to store startup details, user information, and resources for quick retrieval.
 - Implement a robust notification system for all stakeholders to stay updated on progress and important deadlines.
 - Ensure the system is scalable, allowing for multiple startups and users without performance issues.
 - Offer a reporting system that generates summaries for post-launch evaluation and long-term success tracking.
-

Sample Output:

Example 1: Tech Startup Founder in Idea Stage

- **Roadmap:**
 1. Market Research (2 weeks)
 2. Develop MVP (1 month)
 3. Validate Product with Early Users (1 month)
 4. Seek Seed Funding (2 months)
- **Resources:**
 - Articles on market research techniques.
 - Tools for building an MVP (e.g., no-code platforms).
 - Connections to angel investors.

Result:

The problem statement has been successfully created, outlining the challenges faced by entrepreneurs in launching startups and proposing a centralized solution to streamline the startup journey. The **Startup Navigator App** will provide clear roadmaps, tailored resources, and a collaborative platform for all stakeholders involved in the entrepreneurial ecosystem.

EXERCISE NO. 2

AIM: To do requirement analysis and develop Software Requirement Specification Sheet (SRS) for startup navigator app

1. INTRODUCTION

1.1 Purpose

The purpose of this document is to define the **Startup Navigator App**, aimed at providing young entrepreneurs with structured guidance and resources to launch their startups successfully. The app will streamline the process of idea evaluation, roadmap creation, business development, funding assistance, and legal compliance. The main stakeholders include entrepreneurs, investors, mentors, and administrators.

1.2 Document Conventions

This document uses the following abbreviations:

- DB: Database
- GUI: Graphical User Interface
- API: Application Programming Interface

1.3 Intended Audience and Reading Suggestions

This SRS is intended for:

- Entrepreneurs who need guidance for developing and launching their startups.
- Mentors and Investors who provide feedback, mentorship, and funding.
- Administrators who manage platform operations, user support, and content moderation.

1.4 Project Scope

The **Startup Navigator App** will help entrepreneurs by providing a centralized platform for creating customized startup roadmaps, accessing resources, and connecting with investors and mentors. It will offer features like idea evaluation, business development tools, legal guidance, and funding options, aiming to streamline the startup process.

1.5 References

- "Lean Startup" by Eric Ries
- Various startup resources and guidelines
- Existing platforms (e.g., Startup Genome, AngelList)

2. OVERALL DESCRIPTION

2.1 Product Perspective

The app will operate as a cross-platform mobile and web application. It will integrate with databases to store entrepreneur information, business ideas, and mentor/investor profiles.

- **Idea Evaluation:** Users can input their business idea for analysis and feedback.
- **Roadmap Creation:** Entrepreneurs will receive a personalized startup roadmap based on their progress.
- **Mentor and Investor Connections:** The app will match users with relevant investors and mentors.
- **Learning Modules:** Provide access to educational content tailored to the startup phase.

2.2 Product Features

Key features include:

- **Idea Evaluation:** Automated and mentor-based feedback on submitted business ideas.
- **Roadmap Creation:** Generate personalized startup roadmaps with actionable steps.
- **Mentor Matching:** Entrepreneurs can connect with mentors or advisors.
- **Funding Assistance:** Offer resources and links to funding opportunities.
- **Notifications:** Send reminders, updates, and advice for each startup phase.
- **Post-Launch Reporting:** Provide tools for startup progress tracking and reporting.

2.3 User Classes and Characteristics

- **Entrepreneurs:** Users seeking guidance to launch their startups. They will have access to roadmap creation, mentorship, and educational tools.
- **Mentors & Investors:** Professionals providing feedback, support, and funding.
- **Administrators:** Users who manage app operations, content moderation, and support.

2.4 Operating Environment

The system will run on:

- **Server:** Cloud-based hosting.
- **Client:** Modern web browsers and mobile devices (iOS and Android).
- **Database:** SQL or NoSQL database to store user and startup information.

2.5 Design and Implementation Constraints

- **Compliance:** Adhere to data privacy regulations (e.g., GDPR).
- **Tech Stack:** Front-end using HTML, CSS, JavaScript; back-end using Python or Node.js.
- **Data Security:** Encryption and authentication protocols to ensure data privacy.

2.6 Assumptions and Dependencies

- **Internet Access:** The app assumes stable internet access for user operations.
- **User Accounts:** Entrepreneurs, mentors, and investors must create verified accounts to access the platform.

3. SYSTEM FEATURES

3.1 Functionality

- **Idea Submission:** Entrepreneurs submit their business ideas for evaluation.
- **Roadmap Creation:** Automatically generate personalized startup roadmaps based on user input.
- **Mentorship:** Entrepreneurs can seek mentorship and feedback through the app.
- **Funding Resources:** Provide access to investors and funding opportunities.
- **Notifications:** Send alerts and reminders for important startup milestones.
- **Post-Launch Reporting:** Track the progress and impact of launched startups.

3.2 External Interface Requirements

3.2.1 User Interfaces

- **Web Interface:** A responsive web app accessible on browsers.
- **Mobile Interface:** A mobile-optimized version for iOS and Android devices.

3.2.2 Hardware Interfaces

- **Client Devices:** Smartphones, tablets, and desktops/laptops.

3.2.3 Software Interfaces

- **Operating Systems:** Android, iOS, Windows, and macOS.
- **Database:** SQL or NoSQL for data storage (e.g., MongoDB, MySQL).
- **APIs:** Integration with third-party funding sources and mentor platforms.

3.2.4 Communication Interfaces

- **Internet Protocols:** HTTPS for secure data transmission.

4. NON-FUNCTIONAL REQUIREMENTS

4.1 Performance Requirements

- **Response Time:** System response time must be less than 2 seconds for any user operation.
- **Availability:** 99.9% uptime to ensure users can access the app at all times.

4.2 Safety Requirements

- **Data Backup:** Regular automated backups to ensure no loss of user data.
- **Error Handling:** System errors should not affect ongoing user operations or progress.

4.3 Security Requirements

- **User Authentication:** Secure login via OAuth or other methods.
- **Data Encryption:** Encrypt sensitive user data, including startup plans and financial data.

4.4 Software Quality Attributes

- **Usability:** Easy-to-navigate interface for all user classes.
- **Maintainability:** Easily updatable for future features or bug fixes.
- **Scalability:** System must handle increased numbers of users and startups as the platform grows.

Result:

The **SRS for Startup Navigator App** has been successfully developed, outlining core features, functionalities, and requirements for improving startup success rates through a streamlined, centralized platform.

3- Exercise: Drawing the Entity Relationship Diagram (ERD)

AIM:

To draw the Entity Relationship Diagram (ERD) for the **Startup Navigator App**.

ALGORITHM:

1. Mapping of Regular Entity Types

- Identify the main entities involved in the system:
 - **Entrepreneurs**
 - **Mentors**
 - **Investors**
 - **Startups**
 - **Administrators**

2. Mapping of Weak Entity Types

- Determine if there are any weak entities (e.g., specific startup roadmap or funding information that is dependent on the startup entity). In this case, no weak entities are identified.

3. Mapping of Binary 1:1 Relationship Types

- Identify any relationships where each entity is related to only one instance of another entity. No 1:1 relationships are defined in this system.

4. Mapping of Binary 1: N Relationship Types

- Define relationships where one entity can relate to multiple instances of another entity:
 - **Entrepreneurs** create and manage multiple **Startups** (1
).
 - **Administrators** manage multiple **Startups** (1
).

5. Mapping of Binary M: N Relationship Types

- Define relationships where many instances of one entity can relate to many instances of another entity:
 - **Mentors** can guide multiple **Startups**, and each **Startup** can have multiple **Mentors** (M
).
 - **Investors** can invest in multiple **Startups**, and each **Startup** can receive investment from multiple **Investors** (M
).

INPUT:

- **Entities:**
 - **Entrepreneurs**
 - **Mentors**
 - **Investors**
 - **Startups**
 - **Administrators**
- **Entity Relationship Matrix:**
 - **Entrepreneurs ↔ Startups: 1**

(One entrepreneur can create multiple startups)

- **Startups ↔ Mentors: M**

(Many mentors can guide many startups)

- **Startups ↔ Investors: M**

(Many investors can invest in many startups)

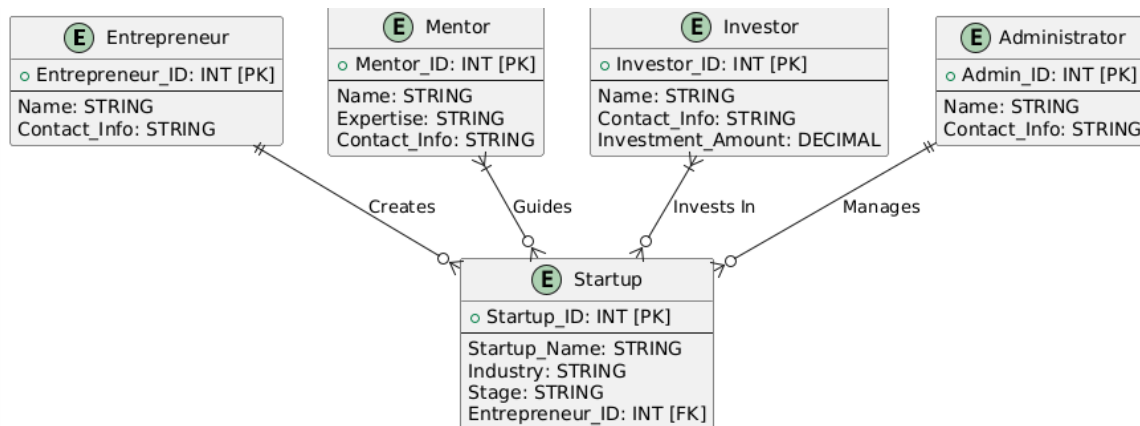
- **Primary Keys:**
 - **Entrepreneur_ID** (for Entrepreneurs)
 - **Mentor_ID** (for Mentors)
 - **Investor_ID** (for Investors)
 - **Admin_ID** (for Administrators)
 - **Startup_ID** (for Startups)
- **Attributes:**
 - **Entrepreneurs:**
 - Entrepreneur_ID
 - Name
 - Contact_Info
 - **Mentors:**
 - Mentor_ID
 - Name
 - Expertise
 - Contact_Info
 - **Investors:**
 - Investor_ID
 - Name
 - Contact_Info
 - Investment_Amount
 - **Startups:**
 - Startup_ID
 - Startup_Name
 - Industry
 - Stage

- Entrepreneur_ID (FK)
- **Administrators:**
 - Admin_ID
 - Name
 - Contact_Info

Mapping of Attributes with Entities:

- **Entrepreneurs** → [Entrepreneur_ID, Name, Contact_Info]
- **Mentors** → [Mentor_ID, Name, Expertise, Contact_Info]
- **Investors** → [Investor_ID, Name, Contact_Info, Investment_Amount]
- **Startups** → [Startup_ID, Startup_Name, Industry, Stage, Entrepreneur_ID]
- **Administrators** → [Admin_ID, Name, Contact_Info]

OUTPUT:



RESULT:

The entity relationship diagram was made successfully by following the steps described above, effectively illustrating the relationships and attributes relevant to the Startup Navigator.

4- Drawing the Data Flow Diagram (DFD)

AIM:

To draw the **Data Flow Diagram (DFD)** for the **Startup Navigator** app and list the application modules.

ALGORITHM:

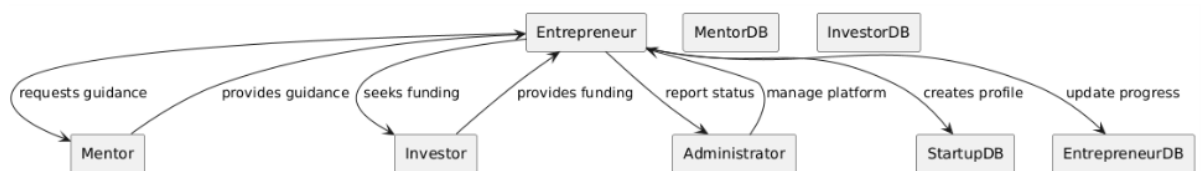
1. **Open Tool:** Use diagramming software like **Visual Paradigm** or **Lucidchart**.
2. **Select Template:** Choose a **DFD template** to begin designing.
3. **Name the DFD:** Title it appropriately (e.g., "Startup Navigator DFD").
4. **Add External Entities:** Include:
 - **Entrepreneur:** The user who will interact with the platform.
 - **Mentor:** Provides guidance and support.
 - **Investor:** Provides funding.
 - **Administrator:** Oversees the system.
5. **Add Processes:** Include:
 - **Create Startup Profile**
 - **Receive Mentorship**
 - **Seek Investment**
 - **Manage Startup Progress**
6. **Add Data Stores:** Include:
 - **Startup Database** (stores startup details)
 - **Entrepreneur Database** (stores user data)
 - **Mentor Database** (stores mentor profiles)
 - **Investor Database** (stores investor profiles)
7. **Add Data Flow:** Connect external entities and processes with labeled arrows indicating the data exchanged.
8. **Customize DFD:** Use colors and fonts to clarify processes and connections.
9. **Add Title and Share:** Include a title (e.g., "Startup Navigator DFD") and share your DFD.

INPUT:

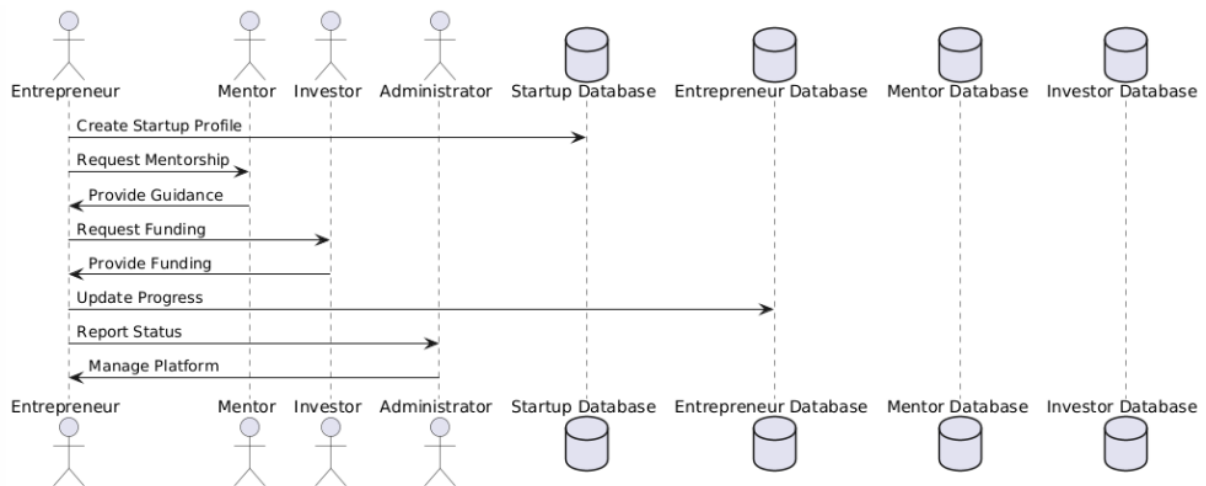
- **Processes:**
 - Create Startup Profile
 - Receive Mentorship
 - Seek Investment
 - Manage Startup Progress
- **Data Stores:**
 - Startup Database
 - Entrepreneur Database
 - Mentor Database
 - Investor Database
- **External Entities:**
 - Entrepreneur
 - Mentor
 - Investor
 - Administrator

MODULES IN THE APPLICATION:

1. **Startup Management Module**
2. **Mentorship Module**
3. **Investment Module**
4. **Admin Module**
5. **Notification Module**
6. **Reporting Module**



OUTPUT :



Result: The Data Flow diagram was made successfully by following the steps described above.

EXERCISE NO. 5

AIM:

To Draw the Use Case Diagram for the Startup Navigator App.

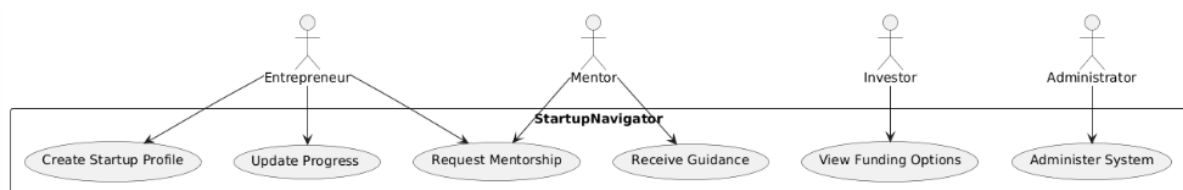
ALGORITHM:

1. Identify Actors: Identify the primary users interacting with the system (e.g., Entrepreneur, Mentor, Investor, Administrator).
2. Identify Use Cases: List the main tasks the actors perform within the system (e.g., Create Startup Profile, Request Mentorship).
3. Connect Actors and Use Cases: Draw lines connecting each actor to the use cases they interact with.
4. Add System Boundary: Encapsulate the use cases within a boundary that represents the Startup Navigator system.
5. Define Relationships: Specify include or extend relationships between use cases if applicable.
6. Review and Refine: Ensure that all important tasks and interactions are captured accurately.
7. Validate: Validate the diagram by checking it against the system's requirements and functionality.

INPUTS:

- Actors: Entrepreneur, Mentor, Investor, Administrator
- Use Cases: Create Startup Profile, Request Mentorship, View Funding Options, Update Progress, Receive Guidance, Administer System
- Relations: Direct connections between actors and use cases

OUTPUT:



Result: The use case diagram has been created successfully by following the steps given.

EXERCISE NO. 6

AIM:

To Draw the Activity Diagram for the Startup Navigator App.

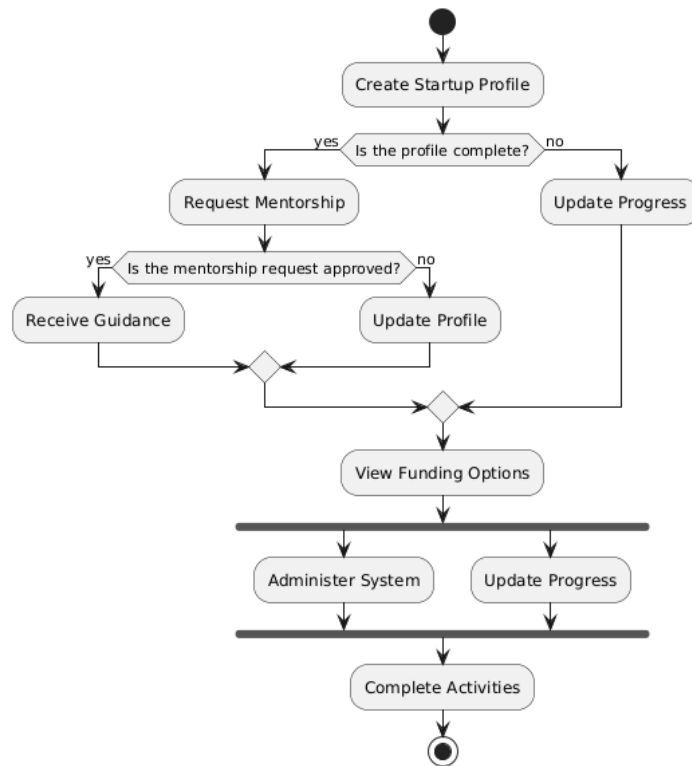
ALGORITHM:

1. Identify Initial State and Final States:
 - Define the starting point (initial state) of the activity and the end point (final state) where the process concludes.
2. Identify Intermediate Activities Needed:
 - List the main activities that occur between the initial and final states (e.g., creating a startup profile, requesting mentorship, etc.).
3. Identify Conditions or Constraints:
 - Determine any conditions or decision points that affect the flow of activities (e.g., has the entrepreneur completed the profile?).
4. Draw the Diagram with Appropriate Notations:
 - Use standard notations like ovals for initial and final states, rectangles for activities, diamonds for decision points, and arrows to show the flow of control.

INPUTS:

- Activities:
 - Create Startup Profile
 - Request Mentorship
 - View Funding Options
 - Update Progress
 - Receive Guidance
 - Administer System
- Decision Points:
 - Is the profile complete?
 - Is the mentorship request approved?
- Guards:
 - If the profile is complete, allow submission.
 - If the request is approved, allow access to mentorship resources.
- Parallel Activities:
 - Entrepreneurs can update progress while receiving guidance from mentors.
- Conditions:
 - Ensure all mandatory fields are filled before submitting the startup profile.
 - Check for any conflicts in mentorship scheduling.

OUTPUT:



RESULT :The Activity diagram has been created successfully by following the steps given.

EXERCISE NO. 7

Objects to be Analyzed:

1. Startup
2. Entrepreneur
3. Mentor

States:

1. Startup:
 - Planned: The startup is in the planning phase.
 - Active: The startup is actively being developed.
 - Completed: The startup has been completed or launched.
 - Paused: The startup is temporarily halted due to some issues or awaiting decision.
 - Canceled: The startup is canceled before completion.
2. Entrepreneur:
 - Inactive: The entrepreneur has not registered or is not using the system.
 - Registered: The entrepreneur has created an account and registered the startup.
 - Active: The entrepreneur is actively working on the startup and requesting feedback or funding.
 - Feedback Received: The entrepreneur has received feedback from a mentor or investor.
3. Mentor:
 - Inactive: The mentor has not logged in or registered.
 - Available: The mentor is available to provide feedback or guidance.
 - Providing Feedback: The mentor is actively giving feedback to an entrepreneur.
 - Unavailable: The mentor is currently unavailable.

Events (Triggers for State Changes):

1. Startup Events:
 - Event Created: The entrepreneur creates the startup profile, initiating the "Planned" state.
 - Event Started: The startup moves to "Active" once it begins development.
 - Event Completed: The startup reaches the "Completed" state after its successful launch.
 - Event Canceled: The startup is canceled, and its state changes to "Canceled."

- Event Paused: The startup moves to the "Paused" state if the entrepreneur halts its progress.

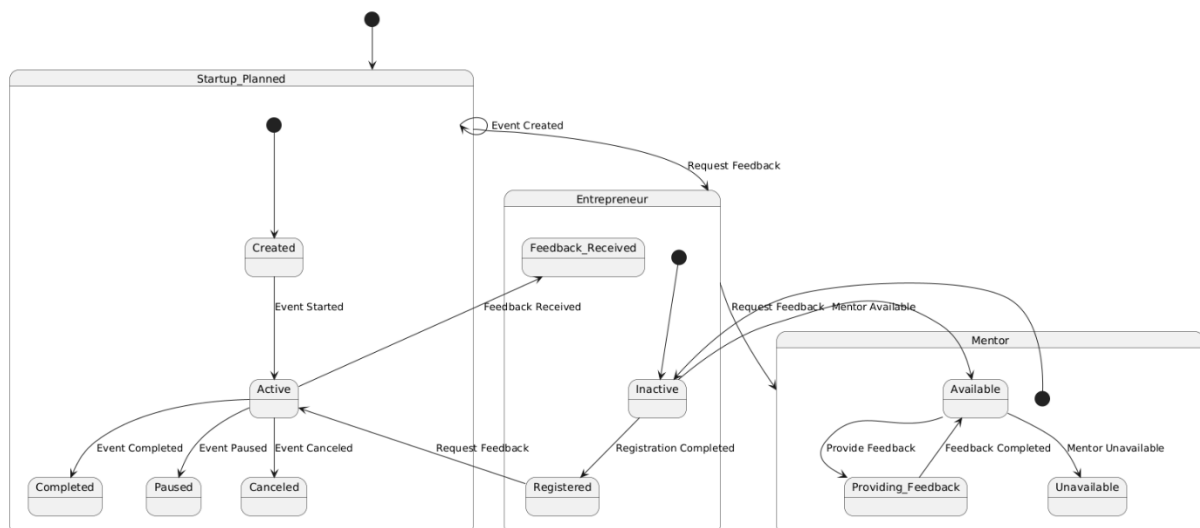
2. Entrepreneur Events:

- Registration Completed: The entrepreneur completes registration, transitioning to "Registered" state.
- Request Feedback: The entrepreneur requests feedback, changing the state to "Active."
- Feedback Received: Once feedback is received, the entrepreneur's state changes to "Feedback Received."

3. Mentor Events:

- Mentor Available: The mentor becomes available to provide feedback.
- Provide Feedback: The mentor moves to the "Providing Feedback" state when providing guidance.

OUTPUT :



Result: The State Chart diagram has been created successfully by following the steps given.

EXERCISE NO. 8

AIM:

To Draw the Sequence Diagram for the Startup Navigator app.

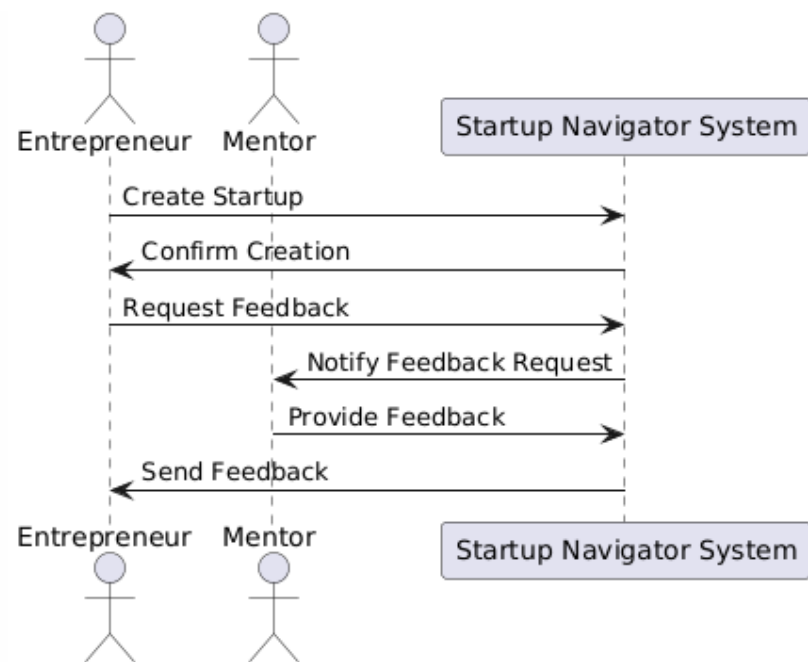
ALGORITHM:

1. **Identify the Scenario:**
(e.g., Entrepreneur creating a startup and seeking feedback).
2. **List the Participants:**
 - Entrepreneur
 - Mentor
 - Startup Navigator System
3. **Define Lifelines for Each Participant.**
4. **Arrange Lifelines Vertically.**
5. **Add Activation Bars to Represent Active Periods.**
6. **Draw Messages Between Participants:**
(e.g., feedback requests).
7. **Include Return Messages After Processing.**
8. **Indicate Timing and Order of Events.**
9. **Include Conditions and Loops (if applicable).**
10. **Consider Parallel Execution (if applicable).**
11. **Review and Refine the Diagram for Clarity.**
12. **Add Annotations and Comments for Better Understanding.**
13. **Document Assumptions and Constraints.**
14. **Use a Tool (like PlantUML or Lucidchart) to Create a Neat Sequence Diagram.**

INPUTS:

- **Objects Participating in the Interaction:**
 - Entrepreneur
 - Mentor
 - Startup Navigator System
- **Message Flows Among the Objects:**
 - Startup creation request
 - Feedback request
 - Confirmation response
- **The Sequence of Messages:**
 - Entrepreneur -> Startup Navigator System: Create Startup
 - Startup Navigator System -> Entrepreneur: Confirm Creation
 - Entrepreneur -> Startup Navigator System: Request Feedback
 - Startup Navigator System -> Mentor: Notify Feedback Request
 - Mentor -> Startup Navigator System: Provide Feedback
 - Startup Navigator System -> Entrepreneur: Send Feedback

Output :



Result: The Sequence diagram has been created successfully by following the steps given.

EXERCISE NO. 9

AIM:

To Draw the Collaboration Diagram for the Startup Navigator app.

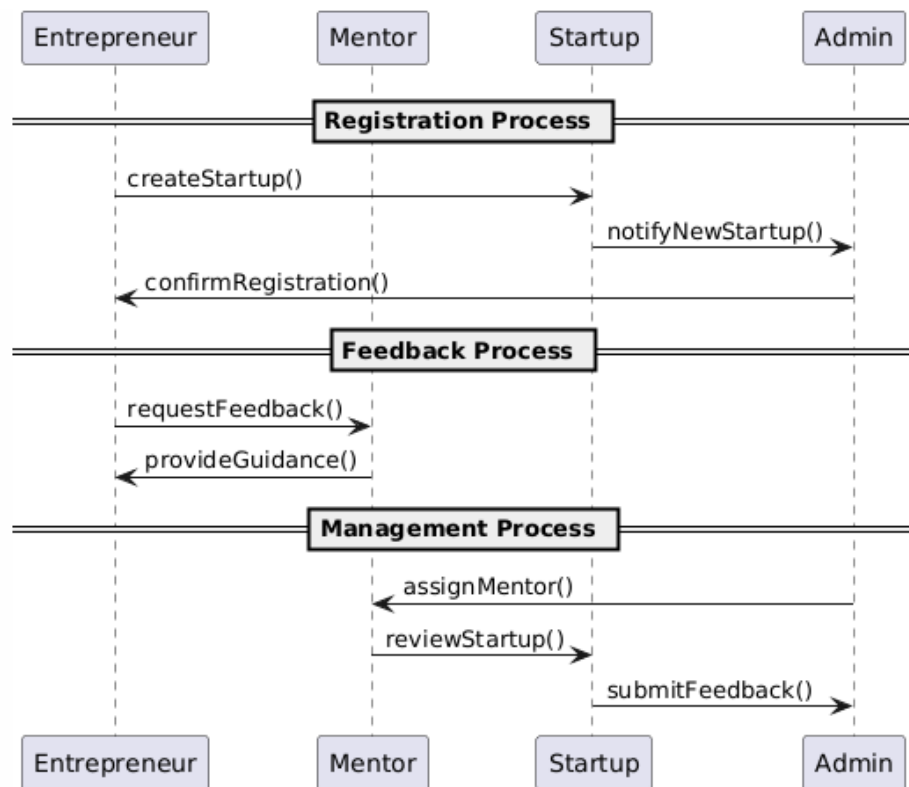
ALGORITHM:

1. **Identify Objects/Participants:**
 - List the key participants (e.g., Entrepreneur, Mentor, Startup Navigator System).
2. **Define Interactions:**
 - Determine how these participants interact with each other.
3. **Add Messages:**
 - Document the messages exchanged between participants, including the order of messages.
4. **Consider Relationships:**
 - Define the relationships (associations) between the objects in the diagram.
5. **Document the Collaboration Diagram:**
 - Create a neat and organized diagram.
 - Include any relevant explanations or annotations to clarify interactions.

INPUTS:

- **Objects:**
 - Entrepreneur
 - Mentor
 - Startup Navigator System
 - Admin
- **Message Flows:**
 - The specific messages sent between participants (e.g., "Create Startup," "Request Feedback," "Provide Guidance").
- **Sequence of Messages:**
 - The order in which messages are exchanged:
 - Entrepreneur requests to create a startup.
 - Startup Navigator System confirms the creation.
 - Entrepreneur requests feedback on the startup idea.
 - Mentor receives the feedback request.
 - Mentor provides feedback to the Startup Navigator System.
 - Startup Navigator System sends the feedback to the Entrepreneur.
- **Object Organization:**
 - How objects relate to one another, indicating associations between the Entrepreneur, Mentor, and the Startup Navigator System.

OUTPUT:



Result: The Collaboration diagram has been created successfully by following the steps given.

EXERCISE NO. 10

AIM:

To Draw the Class Diagram for the Startup Navigator app.

ALGORITHM:

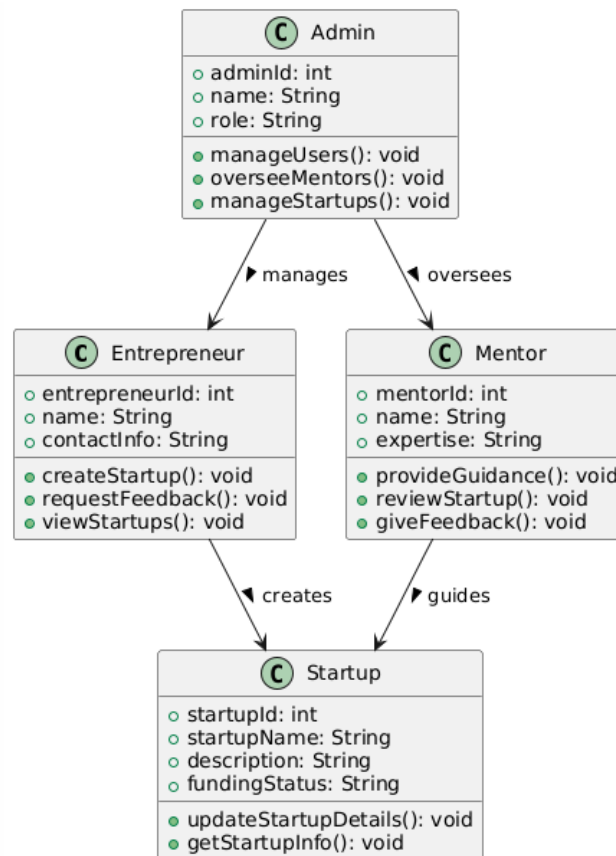
1. **Identify Classes:**
 - Determine the main classes required for the system, such as Entrepreneur, Mentor, Startup, and Admin.
2. **List Attributes and Methods:**
 - Define the attributes (data members) and methods (functions) for each class.
3. **Identify Relationships:**
 - Determine how classes interact with each other (e.g., associations, inheritance).
4. **Create Class Boxes:**
 - Design individual boxes for each class.
5. **Add Attributes and Methods:**
 - Include the attributes and methods within the respective class boxes.
6. **Draw Relationships:**
 - Connect classes using lines to represent their relationships.
7. **Label Relationships:**
 - Clearly label the type of relationships (e.g., one-to-many, inheritance).
8. **Review and Refine:**

- Check for any inconsistencies and make necessary adjustments.
9. **Use Tools for Digital Drawing**

INPUTS:

1. **Class Names:**
 - Entrepreneur
 - Mentor
 - Startup
 - Admin
2. **Attributes:**
 - **Entrepreneur:** entrepreneurId, name, contactInfo
 - **Mentor:** mentorId, name, expertise
 - **Startup:** startupId, startupName, description, fundingStatus
 - **Admin:** adminId, name, role
3. **Methods:**
 - **Entrepreneur:** createStartup(), requestFeedback(), viewStartups()
 - **Mentor:** provideGuidance(), reviewStartup(), giveFeedback()
 - **Startup:** updateStartupDetails(), getStartupInfo()
 - **Admin:** manageUsers(), overseeMentors(), manageStartups()
4. **Visibility Notation:**
 - + for public
 - - for private
 - # for protected

OUTPUT :



Result: The Class diagram has been created successfully by following the steps given.

SAMPLE CODE:

```
1  import streamlit as st
2
3  # App Title
4  st.title("Startup Navigator")
5
6  # Sections
7  st.header("Welcome to Startup Navigator")
8  st.write("This app helps young entrepreneurs to navigate the startup ecosystem.")
9
10 # User Inputs
11 user_name = st.text_input("Enter your name:")
12 if st.button("Submit"):
13     st.write(f"Hello, {user_name}! Let's get started.")
14
15 import streamlit as st
16
17 st.title("Startup Navigator")
18
19 # Add a section for Idea Evaluation
20 st.header("Idea Evaluation")
21
22 idea = st.text_input("Describe your startup idea:")
23 if idea:
24     st.write(f"Your idea: {idea}")
25     st.success("This idea can be processed further!")
26 else:
27     st.info("Please enter your startup idea.")
28
29 # Add a section for Roadmap Creation
30 st.header("Roadmap Creation")
31
32 st.write("Create your roadmap by setting key milestones:")
```

```
26 else:
27     st.info("Please enter your startup idea.")
28
29 # Add a section for Roadmap Creation
30 st.header("Roadmap Creation")
31
32 st.write("Create your roadmap by setting key milestones:")
33 milestones = st.text_area("List your milestones (one per line):")
34 if milestones:
35     milestones_list = milestones.split("\n")
36     st.write("Your roadmap:")
37     for i, milestone in enumerate(milestones_list, start=1):
38         st.write(f"{i}. {milestone}")
39 else:
40     st.info("Please list some milestones for your startup.")
41
42 # Add a section for Networking
43 st.header("Networking")
44
45 st.write("Find collaborators or partners for your startup:")
46 skills_needed = st.multiselect(
47     "What skills are you looking for?",
48     ["Marketing", "Finance", "Technical", "Legal", "Other"]
49 )
50 if skills_needed:
51     st.write("You're looking for collaborators with the following skills:")
52     for skill in skills_needed:
53         st.write(f"- {skill}")
54 else:
55     st.info("Please select the skills you need.")
```

Startup Navigator ↗

Welcome to Startup Navigator

This app helps young entrepreneurs to navigate the startup ecosystem.

Enter your name:

ELSA

Submit

Hello, ELSA! Let's get started.

Startup Navigator

Idea Evaluation

Describe your startup idea:

Please enter your startup idea.

Roadmap Creation

Create your roadmap by setting key milestones:

List your milestones (one per line):

Please list some milestones for your startup.

Networking

Find collaborators or partners for your startup:

What skills are you looking for?

Choose an option

Marketing

Finance

Technical

Legal

Other