



**BANNARI AMMAN INSTITUTE OF TECHNOLOGY**  
An Autonomous Institution Affiliated to Anna University - Chennai, Accredited by NAAC with A+ Grade  
Sathyamangalam - 638401 Erode District, Tamil Nadu, India

**Name:** ADESH S S  
**Seat No:** 143  
**Project ID:** 23

## Approval Of On-duty

### Technical Components:

Components	Technology Stack
Frontend	Vue JS
Backend	Node JS, Express JS
Database	MongoDB
API	RESTful

**Timeline:**

Stage No.	Stage	Deadline	Status
1	Planning and Requirement Gathering	02/05/2024, 03/05/2024	Under review ▾
2	Design and UI / UX Prototyping		Not started ▾
3	Database Design and Implementation		Not started ▾
4	Backend Development		Not started ▾
5	Integration and Testing		Not started ▾
6	Deployment		Not started ▾

## Table Of Contents:

Sl.No.	Content	Page
1.	Introduction <ul style="list-style-type: none"><li>● Problem Statement</li><li>● Overview</li><li>● Objectives</li><li>● Scope</li><li>● Deliverables</li></ul>	4
2.	Requirements <ul style="list-style-type: none"><li>● Functional Requirements</li><li>● Non-Functional Requirements</li><li>● Constraints</li><li>● Assumptions</li></ul>	6
3.	Workflow <ul style="list-style-type: none"><li>● Fig: 1</li><li>● Detailed Step-by-step workflow</li><li>● Decision Points and Conditions</li></ul>	9
4.	Role Definitions <ul style="list-style-type: none"><li>● Student</li><li>● Special Lab In-charge</li><li>● Mentor</li><li>● Admin</li></ul>	12
5.	Process Flow <ul style="list-style-type: none"><li>● Fig: 2</li></ul>	15
6.	Test Cases and Scenarios	15
7.	Appendices <ul style="list-style-type: none"><li>● Glossary of Terms</li><li>● Abbreviations and Acronyms</li><li>● Image Legends</li></ul>	18

# **1. Introduction**

## **1.1 Problem Statement:**

Build a portal/app for approving the On-duty for our students. The condition to approve on-duty

1. Both academic and placement FA should be above 50%
2. Mentor has to provide permission after receiving acknowledgement from the parent.
3. The special lab incharge should approve the permission after verifying event details.

## **1.2 Overview:**

This project is a web application that automates the process of obtaining approvals for student on-duty requests. When a student applies for an on-duty, it triggers a workflow that involves notifying the lab in-charge, seeking parental approval via an IVR call, obtaining mentor approval, and finally storing the approved on-duty details. The application also provides an admin interface to view the number of students on-duty categorised by academic year.

## **1.3 Objectives:**

1. To develop a centralised web application that allows students to submit on-duty requests and automates the approval process involving lab in-charges, parents (through IVR calls), and mentors.
2. To implement a secure and organised database for storing approved on-duty requests along with relevant details, ensuring proper documentation and record-keeping.
3. To provide an administrative interface that enables monitoring and reporting of on-duty statistics categorised by academic year, allowing for better oversight and decision-making.

## **1.4 Scope:**

1. Develop a web application for students to submit on-duty requests.
2. Implement an automated workflow for obtaining approvals from lab in-charges, parents (via IVR calls), and mentors.
3. Integrate an Interactive Voice Response (IVR) system for obtaining parental approval over phone calls.
4. Develop a secure database to store approved on-duty requests and related details.
5. Create an administrative interface for monitoring and reporting on-duty statistics.
6. Implement user authentication and access control mechanisms.
7. Ensure data privacy and security measures are in place.

## **1.5 Deliverables:**

1. Interactive Voice Response (IVR) system integrated with the web application for obtaining parental approval over phone calls.
2. Administrative dashboard with the following features:
  - View the number of students on on-duty categorised by academic year
  - Generate reports and analytics related to on-duty requests
3. User documentation for all stakeholders (students, lab in-charges, mentors, administrators).
4. Technical documentation, including system architecture, database design, and API documentation.
5. Deployment and maintenance plan for the web application and associated systems.
6. Testing documentation, including test reports.

## **2. Requirements:**

### **2.1 Functional Requirements:**

1. User Authentication and Authorization
  - a. Integration with Google authentication for student login.
  - b. Role-based access control to ensure mentors and special lab in-charges can only access and perform respective authorised actions.
2. On-Duty Request Submission
  - a. Students should be able to submit on-duty requests through a web form.
  - b. The form should capture relevant details such as event name, date, time, and any additional information.
3. Notification and Approval Workflow
  - a. Upon submitting an on-duty request, the system should send a notification email to the student's lab in-charge.
  - b. If the lab in-charge approves the request, the system should initiate an IVR call to the student's parent's mobile number for approval.
  - c. If the parent approves, the system should send a notification email to the student's mentor for approval.
  - d. Each stakeholder (lab in-charge, parent, mentor) should be able to approve or reject the request.
  - e. Appropriate notifications should be sent to the relevant parties at each stage of the approval process.
4. IVR Integration
  - a. The system should integrate with an IVR (Interactive Voice Response) system to obtain parental approval over phone calls.
  - b. The IVR should prompt the parent to approve or reject the on-duty request (e.g., press 1 for YES, 2 for NO).
5. On-Duty Request Storage and Retrieval

- a. Approved on-duty requests should be stored in a secure database along with relevant details (event, date, time, approvals, etc.).
  - b. The system should provide functionality to retrieve and display on-duty requests for students, lab in-charges, mentors, and administrators based on their respective access levels.
6. Administrative Reporting
  - a. Administrators should be able to view the number of students on-duty categorised by academic year (first year, second year, etc.).
  - b. Administrators should be able to generate reports and analytics related to on-duty requests.

## **2.2 Non-Functional Requirements:**

1. Performance and Scalability
  - The system should be able to handle a large number of concurrent users and on-duty requests without compromising performance.
  - Appropriate load testing and performance tuning should be performed to ensure optimal performance.
2. Usability and User Experience
  - The system should provide clear instructions, error messages, and help documentation to guide users through various tasks.
  - The system should be responsive and accessible across different devices and screen sizes.
3. Backup and Disaster Recovery
  - The system should have a robust backup and disaster recovery plan to ensure data integrity and business continuity in the event of failures or disasters.
  - Regular backups should be scheduled, and disaster recovery procedures should be tested periodically.

## **2.3 Constraints:**

1. Integration with Existing Systems:
  - The on-duty approval system may need to integrate with existing systems or services within the institution(Camps). This integration

may impose constraints on the system's architecture, data formats, or communication protocols.

2. Regulatory and Policy Compliance:

- The system must comply with relevant regulations and institutional policies related to data privacy, security, and accessibility standards. This may constrain the way data is handled, stored, and transmitted within the system.

3. Legacy System Migration:

- The institution's existing on-duty approval process or system in place, there may be constraints related to migrating historical data and ensuring compatibility with legacy systems during the transition phase.

4. Resource Constraints:

- The development and implementation of the on-duty approval system may be constrained by available resources, such as budget, staffing, infrastructure, or time constraints, which could impact the system's scope, features, or timeline.

5. Third-Party Dependencies:

- The system may rely on third-party services or components, such as the IVR system or email service providers. Any limitations or constraints imposed by these third-party services could affect the system's functionality or performance.

6. Mobile Compatibility:

- If the system needs to support mobile devices or responsive design, there may be constraints related to device compatibility, screen sizes, and platform-specific considerations.

## **2.4 Assumptions:**

1. Email and Mobile Communication:

- It is assumed that all stakeholders (lab in-charges, parents, mentors) have valid and accessible email addresses and mobile phone numbers for receiving notifications and IVR calls.

2. IVR Integration:



- It is assumed that a reliable and scalable IVR system or service will be available for integration, capable of handling the expected volume of parental approval calls.
3. Data Integrity and Accuracy:
- It is assumed that the data entered into the system by users (students, lab in-charges, mentors) is accurate and truthful, and appropriate measures will be taken to ensure data quality and integrity.

### 3. Workflow:

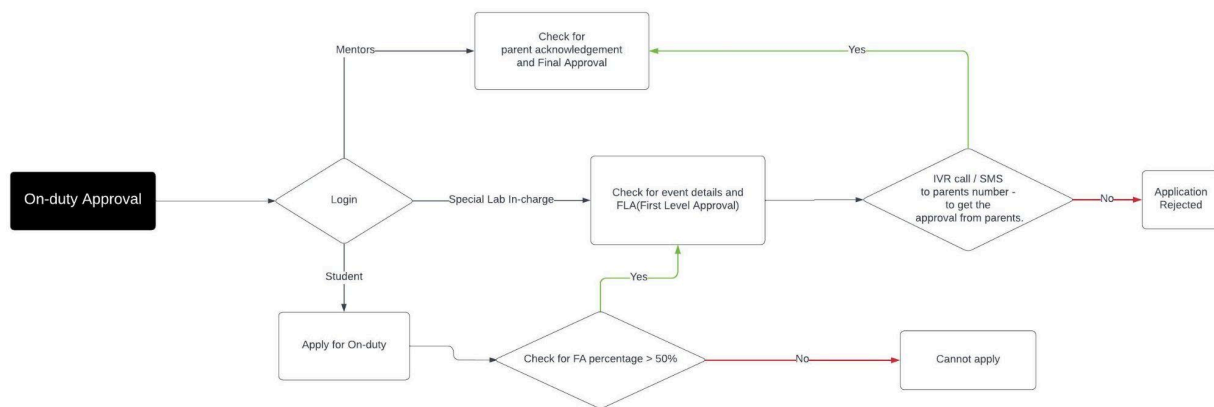


Fig: 1

#### 3.1 Detailed Step-by-step workflow

1. Student Submits On-Duty Request
  - a. Student logs into the web application
  - b. Students fill out the on-duty request form with details (event, date, time, etc.)
  - c. Student submits the on-duty request
2. Notify Lab In-Charge
  - a. The system sends an email notification to the student's lab in-charge informing them about the on-duty request
3. Lab In-Charge Approval
  - a. Lab in-charge reviews the on-duty request
  - b. If approved, the process moves to the next step

- c. If rejected, the student is notified along with the reason, and the process ends
- 4. Parental Approval via IVR
  - a. The system initiates an IVR (Interactive Voice Response) call to the student's parent's mobile number
  - b. The IVR system prompts the parent to approve or reject the on-duty request (e.g., press 1 for YES, 2 for NO)
  - c. If approved, the process moves to the next step
  - d. If rejected, the student is notified stating the reason and the application is cancelled, and the process ends
- 5. Mentor Approval
  - a. The system sends an email notification to the student's mentor, along with the details of the on-duty request and the parental approval
  - b. The mentor reviews the request
  - c. If approved, the on-duty is granted and moves to next step
  - d. If rejected, the student is notified stating the reason, application is cancelled, and the process ends
- 6. Store Approved On-Duty Request
  - a. If all approvals are obtained (lab in-charge, parent, mentor), the on-duty request details are stored in the database.
- 7. Administrative Reporting
  - a. The admin can access the administrative interface
  - b. The admin can view the number of students on-duty categorised by academic year (first year, second year, etc.).

### **3.2 Decision Points and Conditions:**

- 1. Student Submits On-Duty Request
  - a. Decision Point: Is the on-duty request form filled out completely and correctly?
    - Condition: If the form is incomplete or contains invalid data, the student should be prompted to correct the errors before submission.
    - Condition: If the form is complete and valid, the request should be accepted and proceed to the next step.

- b. Decision Point: If the student has FA(Formative Assessment) percentage for both academics and placement above 50%
    - Condition: If above 50% eligible for FLA(First Level Approval).
    - Condition: If below 50% cannot proceed to have on-duty.
- 2. Lab In-Charge Approval
  - a. Decision Point: Does the lab in-charge approve or reject the on-duty request?
    - Condition: If the lab in-charge approves the request, the process should proceed to the parental approval step.
    - Condition: If the lab in-charge rejects the request, the student should be notified, and the process should end, update the status as rejected and store in the database, and notify the student .
- 3. Parental Approval via IVR
  - a. Decision Point: Does the parent approve or reject the on-duty request through the IVR system?
    - Condition: If the parent approves the request (e.g., presses 1 for YES), the process should proceed to the mentor approval step.
    - Condition: If the parent rejects the request (e.g., presses 2 for NO), the process should end, update the status to rejected and stored in the database, and notify students, special lab in-charge.
- 4. Mentor Approval
  - a. Decision Point: Does the mentor approve or reject the on-duty request?
    - Condition: If the mentor approves the request, the process should proceed to store the approved on-duty request.
    - Condition: If the mentor rejects the request, the student, lab in-charge, and parent should be notified, and the process should end.
- 5. Store Approved On-Duty Request
  - a. Decision Point: Have all required approvals (lab in-charge, parent, mentor) been obtained?

- Condition: If all approvals have been obtained, the on-duty request details should be stored in the database, and the relevant stakeholders should be notified of the approval.
- Condition: If any of the required approvals are missing or rejected, the process should end storing the request.

## **4. Role Definitions:**

### **4.1 Students:**

1. Submit On-Duty Requests:
  - a. Students should be able to submit proposals for on-duty requests, providing details such as the reason for the request, the event or activity they plan to attend, date and time, and any other relevant information.
2. View Request Status:
  - a. Students should be able to see the status of their submitted on-duty requests, whether they are pending approval, approved, rejected, or cancelled.
3. Edit or Cancel Requests:
  - a. Students may need the ability to edit details or cancel their on-duty requests before they are approved or rejected, especially if there are changes in their plans or circumstances.
4. View History:
  - a. It can be helpful for students to have access to their on-duty request history, including past approvals, rejections, and cancellations, for reference purposes.
5. Feedback and Support:
  - a. Students should have a way to provide feedback about the on-duty approval process or seek support if they encounter any issues or have questions.

### **4.2 Special Lab In-charge:**

1. Review On-Duty Requests:

- a. Special Lab In-charge should have access to view on-duty requests that have been sent by students and are pending FLA(First Level Approval) from them.
  - b. They can review the event details provided by students and ensure that they align with the requirements for on-duty approval.
2. Verify Event Details:
  - a. Special Lab In-charge should thoroughly verify the event details mentioned in the on-duty requests, such as the purpose of the event, location, date, time, and any other relevant information.
  - b. They may also verify the authenticity of supporting documents or references related to the event.
3. Approve or Reject Requests:
  - a. Based on their verification of event details and compliance with institutional policies, Special Lab In-charge can provide final approval or rejection for on-duty requests.
  - b. If the event details are incomplete or do not meet the required criteria, they can reject the request and provide feedback or guidance to the student.
4. View Request History:
  - a. Special Lab In-charge should have access to view the history of on-duty requests that have been processed through their verification stage, including approved, rejected, or pending requests.
5. Feedback and Support:
  - a. Special Lab In-charge should have a way to provide feedback about the on-duty approval process, report any issues or discrepancies, and seek support if needed.

### **4.3 Mentors:**

1. Review On-Duty Requests:
  - a. Mentors should have access to view on-duty requests approved by parents.
  - b. They can review the details of each request, including the reason, event details, and any attached documentation.
2. Acknowledge Parental Consent:

- a. If parental acknowledgment is required for on-duty requests, mentors should verify that students have obtained this consent before proceeding with approvals.
- 3. Provide Approval or Rejection:
  - a. Mentors can provide FnLA(Final Level Approval) on-duty requests based on the specified criteria, such as the student's formative assessment scores and compliance with academic requirements.
- 4. View Request History:
  - a. Mentors should have access to view the history of on-duty requests submitted by students under their mentorship, including past approvals, rejections, and cancellations.
- 5. Feedback and Support:
  - a. Mentors should have a way to provide feedback about the on-duty approval process, report any issues or discrepancies, and seek support if needed.

#### **4.4 Admin:**

- 1. User Management:
  - a. Create, modify, and delete user accounts for students, mentors, Special Lab In-charge, and other administrative users as needed.
  - b. Assign roles and permissions to users based on their responsibilities within the system.
- 2. Data Management:
  - a. Manage and maintain the database of on-duty requests, approvals, rejections, and related data.
- 3. Monitoring and Reporting:
  - a. Monitor the overall performance and functionality of the on-duty approval system.
  - b. Generate reports and analytics on on-duty requests, approval rates, compliance metrics, and other key performance indicators.
- 4. Feedback and Improvement:
  - a. Gather feedback from users, stakeholders, and system usage metrics to identify areas for improvement.

- b. Implement enhancements, new features, or workflow optimizations to enhance user experience and system efficiency.

## 5. Process Flow:

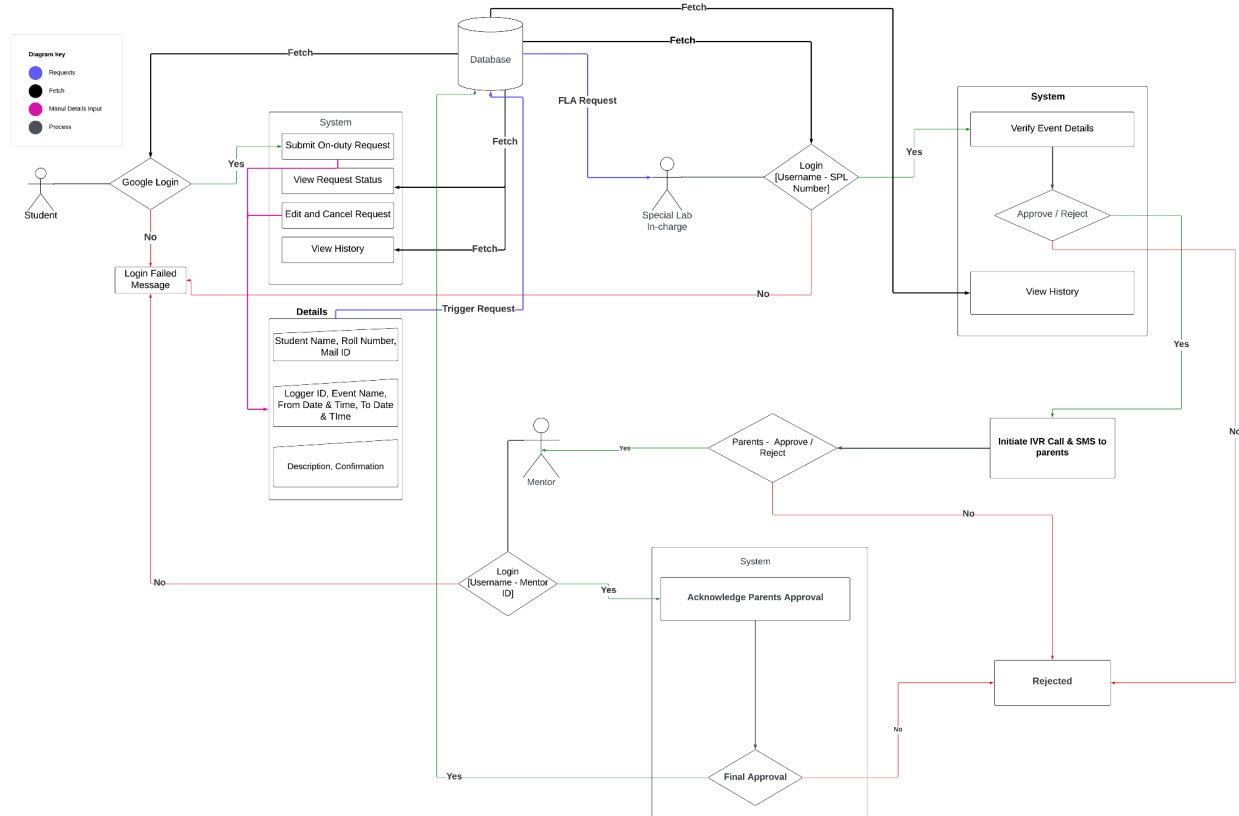


Fig: 2

## 6. Test Cases and Scenarios:

1. User Authentication and Authorization
  - a. Test cases for valid and invalid login credentials (students, lab in-charges, mentors, admins)
  - b. Test cases for role-based access control and authorization checks
  - c. Test cases for Google authentication integration
2. On-Duty Request Submission
  - a. Test cases for submitting a valid on-duty request with all required fields

- b. Test cases for input validation (e.g., invalid dates, missing fields, special characters)
  - c. Test cases for handling duplicate or conflicting on-duty requests
- 3. Notification and Approval Workflow
  - a. Test cases for verifying notifications sent to lab in-charge, parent, and mentor at each stage
  - b. Test cases for approving and rejecting on-duty requests by different stakeholders
  - c. Test cases for handling edge cases (e.g., approvals in different order, multiple rejections)
- 4. IVR Integration
  - a. Test cases for successful IVR call initiation and response recording
  - b. Test cases for handling IVR failures or timeouts
  - c. Test cases for verifying correct interpretation of IVR responses (e.g., 1 for YES, 2 for NO)
- 5. On-Duty Request Storage and Retrieval
  - a. Test cases for storing and retrieving approved on-duty requests in the database
  - b. Test cases for handling large volumes of on-duty request data
  - c. Test cases for data integrity and consistency checks
- 6. Administrative Reporting
  - a. Test cases for generating reports and analytics for on-duty requests
  - b. Test cases for filtering and sorting on-duty requests by different criteria (e.g., academic year)
  - c. Test cases for handling edge cases (e.g., no data, large data sets)
- 7. User Dashboard
  - a. Test cases for displaying on-duty request history, status tracking, and notifications
  - b. Test cases for submitting new on-duty requests from the dashboard
  - c. Test cases for handling various user interactions and edge cases
- 8. Email and Notification Management
  - a. Test cases for verifying email notifications sent at different stages
  - b. Test cases for configuring and updating notification templates and settings
  - c. Test cases for handling email delivery failures or bounces



## 9. Data Security and Privacy

- a. Test cases for data encryption and secure communication protocols
- b. Test cases for access control and authorization checks
- c. Test cases for handling sensitive data and compliance with regulations (e.g., GDPR)

## 10. System Logging and Auditing

- a. Test cases for verifying log entries for user actions, approvals, rejections, and system events
- b. Test cases for log management, rotation, and archiving
- c. Test cases for log analysis and troubleshooting scenarios

## 11. Performance and Load Testing

- a. Test cases for simulating concurrent users and on-duty requests
- b. Test cases for measuring response times and system performance under load
- c. Test cases for verifying system scalability and resource utilization

## 12. Usability and Accessibility Testing

- a. Test cases for user interface navigation, usability, and ease of use
- b. Test cases for accessibility compliance (e.g., WCAG guidelines)
- c. Test cases for responsive design and cross-device compatibility

## 13. Integration Testing

- a. Test cases for integrating with external systems (e.g., IVR, email servers, Google authentication)
- b. Test cases for verifying API and interface functionality
- c. Test cases for end-to-end scenarios involving multiple integrated components

## 14. Security Testing

- a. Test cases for vulnerability scanning and penetration testing
- b. Test cases for testing authentication and authorization mechanisms
- c. Test cases for verifying data encryption and secure communication protocols

## 15. Backup and Disaster Recovery Testing

- a. Test cases for verifying backup and restore procedures
- b. Test cases for simulating disaster scenarios and testing recovery processes
- c. Test cases for data integrity checks after recovery

## **7. Appendices:**

### **7.1 Glossary of terms:**

1. On-Duty Request:
  - A formal request submitted by a student to obtain approval for attending an event or activity during regular class/lab hours.
2. Lab In-Charge:
  - The faculty member or instructor responsible for a particular lab or course, who needs to approve a student's on-duty request.
3. Mentor:
  - A designated faculty member or advisor assigned to a student, who needs to approve the student's on-duty request.
4. IVR:
  - An automated telephony system that interacts with callers through pre-recorded voice prompts and touch-tone keypad input, used in this system to obtain parental approval for on-duty requests.
5. Approval Workflow:
  - The sequence of steps and stakeholders involved in the process of approving an on-duty request, including lab in-charge approval, parental approval, and mentor approval.
6. Academic Year:
  - The annual period of study divided into semesters, quarters, or trimesters, used to categorise and report on-duty request statistics.
7. User Dashboard:
  - A personalised interface for students, providing access to submit on-duty requests, view request history, and track the approval status.
8. Administrative Dashboard:
  - An interface for authorised administrators to monitor and generate reports on on-duty requests, categorised by academic year or other criteria.
9. Notification:
  - An electronic communication (e.g., email, SMS) sent to stakeholders (lab in-charge, parent, mentor) at various stages of the approval process.

#### 10. Authentication:

- The process of verifying the identity of a user, typically through a username and password or other credentials (e.g., Google authentication).

#### 11. Authorization:

- The process of granting or denying access to specific resources or actions within the system based on the user's role and permissions.

#### 12. RBAC:

- A security model that restricts system access and permissions based on the user's assigned role (e.g., student, lab in-charge, mentor, admin).

#### 13. Data Integrity:

- The accuracy, completeness, and consistency of data stored and processed within the system.

#### 14. Backup and Disaster Recovery:

- The processes and procedures for creating redundant copies of data and systems to ensure business continuity and data restoration in the event of failures or disasters.

## 7.2 Abbreviations and Acronyms:

1. MEVN → MongoDB, Express JS, Vue JS, Node JS
2. API → Application Programming Interface
3. REST → Representational State Transfer
4. UI → User Interface
5. UX → User Experience
6. FA → Formative Assessment
7. IVR → Interactive Voice Response
8. FLA → First Level Approval
9. FnLA → Final Level Approval
10. GDPR → General Data Protection Regulation
11. RBAC → Role Based Access Control

## 7.3 Image Legends:

1. Fig: 1 → Workflow

2. Fig: 2 → Process Flow between Students, Special Lab In-charges, Mentors, Parents.