Summer Training Report On

“FAULTY ITEMS SEND FOR REPAIR”



Recruitment & Assessment Centre

Defence Research & Development Organization Timarpur,

New Delhi,110054

in Partial Fulfillment of the Requirements for the Degree of

Bachelor of Technology

***in***

Computer Science

**Submitted by: Hitesh Manral Submitted to: Mr. Rishi Kalia**

ACKNOWLEDGMENT

I would like to express my gratitude for the project report on 'Faulty Items send for repair’. I am thankful to, Defence Research and Development Organization for allowing me to intern with their company. I would like to thank my guide Mr. Rishi Kalia for her valuable guidance, which helped me a lot.

## Hitesh Manral

**Table of Content**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TOPICS** | **PAGE.NO** |
| 1 | Introduction To DRDO | 4 |
| 2 | Tech Stack | 6 |
| 3 | Software And System | 8 |
| 4 | Project overview and code | 27 |
| 5 | Output | 38 |

# CHAPTER 1 :- INTRODUCTION TO DRDO

The Defence Research and Development Organization (DRDO) (IAST: Raksā Anūsandhān Evam Vikās Sangaṭhan) is an agency under the Department of Defence Research and Development in Ministry of Defence of the Government of India, charged with the military's research and development, headquartered in Delhi, India. It was formed in 1958 by the merger of the Technical Development Establishment and the Directorate of Technical Development and Production of the Indian Ordnance Factories with the Defence Science Organisation under the administration of Jawaharlal Nehru. Subsequently, Defence Research C Development Service (DRDS) was constituted in 1979 as a service of Group 'A' Officers / Scientists directly under the administrative control of the Ministry of Defence.

With a network of 52 laboratories that are engaged in developing defence technologies covering various fields like aeronautics, armaments, electronics, land combat engineering, life sciences, materials, missiles, and naval systems, DRDO is India's largest and most diverse research organisation. The organisation includes around 5,000 scientists belonging to the DRDS and about 25,000 other subordinate scientific, technical, and supporting personnel.

#### History

The DRDO was established in 1958 by combining the Defence Science Organisation and some of the technical development establishments. A separate Department of Defence Research and Development was formed in 1980, which later administered DRDO and its almost 30 laboratories and establishments (there were almost 52 labs before merging). Most of the time, the Defence Research and Development Organisation was treated as if it were a vendor and the Army Headquarters or the Air Force Headquarters were the customers. Because the Army and the Air Force themselves did not have any design or construction responsibility, they tended to treat the designer or Indian industry at par with their corresponding designer in the world market. If they could get a MiG-21 from the world market, they wanted a MiG-21 from DRDO.[citation needed]

DRDO started its first major project in surface-to-air missiles (SAM) known as Project Indigo in the 1960s. Indigo was discontinued in later years without achieving full success. Project Indigo led Project Devil, along with Project Valiant, to develop short-range SAM and ICBM in the 1970s. Project Devil itself led to the later development of the Prithvi missile under the Integrated Guided Missile Development Programme (IGMDP) in the 1980s. IGMDP was an Indian Ministry of Defence programme between the early 1980s and 2007 for the development of a comprehensive range of missiles, including the Agni missile, Prithvi ballistic missile, Akash missile, Trishul missile and Nag Missile. In 2010, the defence minister A. K. Antony ordered the restructuring of the DRDO to give a major boost to defence research in the country and to ensure effective participation of the private sector in defence technology. The key measures to make DRDO effective in its functioning include the establishment of a Defence TechnologyCommission with the defence minister as its chairman. The programmes which were largely managed by DRDO have seen considerable success with many of the systems seeing rapid deployment as well as yielding significant technological benefits.

Since its establishment, DRDO has created other major systems and critical technologies such as aircraft avionics, UAVs, small arms, artillery systems, EW Systems, tanks and armoured vehicles, sonar systems, command and control systems and missile systems.

#### Plans for reforms

The Government of India has plans to overhaul and reform DRDO, and a committee was formed to suggest reforms and overhauls.The 9-member committee, which was under the Prime Minister’s Office (PMO), was led by former Principal Scientific Adviser to the Government of India K. Vijayraghavan. The committee submitted its report on the reforms for DRDO in July 2024.Reportedly, the recommendations from the committee have been described as contentious and it was also reported that the DRDO has agreed to implement approximately 60% of the major reforms.As of October 2024, the implementation of structural reforms has been delayed due to opposition from senior DRDO officials.

# CHAPTER 2:- TECHSTACK

### WEB APPLICATION

A web application is a software application that can be run without being installed on the client and that has several parts: A part of it will run on the remote web server, another part will run on the client, usually inside a web browser. Both parts communicate over a computer network, for example the internet. The protocol they often use is called HTTP. This kind of application design is called client-server model. A Web application can be advantageous because the use of browsers allows the application to be compatible with most standard computers and operating systems whilst it does not take up memory on a computer’s hard drive and is accessible from nearly any computer or device a person might use. The parts are usually called "tiers". The most common kind is a three-tier application: One tier is called Presentation tier, another is called Application tier, and the third is called Storage tier. Common examples of web applications are those used for web mail, or for online banking. Social media sites, such as Wikipedia are also web applications.



#### Benefits and Drawbacks

##### Benefits:

* Web applications run "inside" a browser; no complex installation is needed.
* Web applications require very little disk space (or computing power) on the client. All the client does is display the data.

. • Web applications solve some of the "compatibility issues" (Windows, Mac, Linux); all that is needed is a browser.

* In many cases, the data is stored remotely too. As with other cloud computing, this can allow easy communication and cooperation.
* Help for communication and mail.

#### Drawbacks:

* Because they run inside a web browser, most web applications "look" very different to regular programs. The user experience or ease of use is different and some may dislike it.
* Web applications need to be coded so they follow standards. Any browser that also follows the standard can be used. Small changes in a given browser's implementation of a standard may prevent that browser from using the web application.
* Web applications need a connection to the server where the application runs, all the time. The connection may need a certain bandwidth. Without an adequate connection, the application may not be usable; in the worst case, data may be lost.
* Many applications are dependent on the server that hosts them. When the server is switched off, or the company goes bust, the application is no longer usable. Traditional applications continue to work.
* The company offering the web application has complete control over it. This also means that they may launch a new version when they want to; the option to "skip" a less popular version does not exist. • In many cases, the data is stored remotely too. It may not be possible to export the data so that it can be used with another application.
* The company can theoretically track anything the users do. This can cause privacy.

**CHAPTER 3:- SOFTWARE AND SYSTEM**

### SOFTWARE USES

* VISUAL STUDIO
* XAMPP
* My SQL
* React JavaScript

### SYSTEM REQUIRMENT

* 8GB RAM
* MS OFFICE
* 2.4GHZ PROCESSOR
* 1GB MINIMUM STORAGE

## Visual Studio

Visual Studio is an integrated development environment (IDE) developed by Microsoft. It is used to develop computer programs including websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms including Windows API, Windows Forms, Windows Presentation Foundation (WPF), Microsoft Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works as both a source- level debugger and as a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that expand the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Azure DevOps client: Team Explorer).

Visual Studio supports 36 different programming languages [citation needed] and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C,C++, C++/CLI, Visual Basic .NET, C#, F#, JavaScript, TypeScript, XML, XSLT, HTML, and CSS. Support for other languages such as Python,[8] Ruby, Node.js, and M among others is available via plug- ins. Java (and J#) were supported in the past.

The most basic edition of Visual Studio, the Community edition, is available free of charge. The slogan for Visual Studio Community edition is "Free, fully-featured IDE for students, open- source and individual developers". As of February 19, 2024, Visual Studio 2022 is a current production-ready version. Visual Studio 2013, 2015 and 2017 are on Extended Support, while 2019 is on Mainstream Support.

## Features

##### Code editor

Visual Studio includes a code editor that supports syntax highlighting and code completion using IntelliSense for variables, functions, methods, loops, and LINQ queries.

IntelliSense is supported for the included languages, as well as for XML, Cascading Style Sheets, and JavaScript when developing web sites and web applications.

Autocomplete suggestions appear in a modeless list box over the code editor window, in proximity of the editing cursor. In Visual Studio 2008 onwards, it can be made temporarily semi-transparent to see the code obstructed by it.The code editor is used for all supported languages.

The code editor in Visual Studio also supports setting bookmarks in code for quick navigation. Other navigational aids include collapsing code blocks and incremental search, in addition to normal text search and regex search. The code editor also includes a multi- item clipboard and a task list. The code editor supports code snippets, which are saved templates for repetitive code and can be inserted into code and customized for the project being worked on. A management tool for code snippets is built in as well. These tools are surfaced as floating windows which can be set to automatically hide when unused or docked to the side of the screen. The code editor in Visual Studio also supports code refactoring including parameter reordering, variable and method renaming, interface extraction, and encapsulation of class members inside properties, among others.

**XAMPP**

XAMPP (/ˈzæmp/ or /ˈɛks.æmp/) is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source) [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [web](https://en.wikipedia.org/wiki/Web_server) [server](https://en.wikipedia.org/wiki/Web_server) [solution stack](https://en.wikipedia.org/wiki/Solution_stack) package developed by Apache Friends,[[2]](https://en.wikipedia.org/wiki/XAMPP#cite_note-kaiseidlerinterview-2) consisting mainly of the [Apache HTTP Server](https://en.wikipedia.org/wiki/Apache_HTTP_Server), [MariaDB](https://en.wikipedia.org/wiki/MariaDB) [database](https://en.wikipedia.org/wiki/Database), and [interpreters](https://en.wikipedia.org/wiki/Interpreter_(computing)) for scripts written in the [PHP](https://en.wikipedia.org/wiki/PHP) and [Perl](https://en.wikipedia.org/wiki/Perl) [programming languages](https://en.wikipedia.org/wiki/Programming_language). Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible.

XAMPP's ease of deployment means a [WAMP](https://en.wikipedia.org/wiki/WAMP) or [LAMP](https://en.wikipedia.org/wiki/LAMP_(software_bundle)) stack can be installed quickly and simply on an operating system by a developer, with the advantage that common add-in applications such as [WordPress](https://en.wikipedia.org/wiki/WordPress) and [Joomla!](https://en.wikipedia.org/wiki/Joomla!) can also be installed with similar ease using [Bitnami](https://en.wikipedia.org/wiki/Bitnami).

##### Etymology

The Apache Friends website indicates that XAMPP stands for "XAMPP [Apache](https://en.wikipedia.org/wiki/Apache_HTTP_Server) + [MariaDB](https://en.wikipedia.org/wiki/MariaDB) + [PHP](https://en.wikipedia.org/wiki/PHP) + [Perl](https://en.wikipedia.org/wiki/Perl)", making it a [recursive acronym](https://en.wikipedia.org/wiki/Recursive_acronym). XAMPP formerly used MySQL, but this was replaced with MariaDB on 19 October 2015 and beginning with XAMPP versions 5.5.30 and 5.6.14, altering the meaning of the acronym. It originally stood for Cross-Platform + Apache + MySQL + PHP + Perl.

##### Prerequisites

XAMPP requires only one [zip](https://en.wikipedia.org/wiki/ZIP_(file_format)), [tar](https://en.wikipedia.org/wiki/Tar_(file_format)), [7z](https://en.wikipedia.org/wiki/7z_(file_format)), or [exe](https://en.wikipedia.org/wiki/EXE) file to be downloaded and run, and little or no configuration of the various components that make up the web server is required. The Windows version of XAMPP requires Microsoft Visual C++ 2017 Redistributable.

##### Features

XAMPP is regularly updated to the latest releases of [Apache](https://en.wikipedia.org/wiki/Apache_HTTP_Server), [MariaDB](https://en.wikipedia.org/wiki/MariaDB), [PHP](https://en.wikipedia.org/wiki/PHP) and [Perl](https://en.wikipedia.org/wiki/Perl). It also comes with a number of other modules, including [OpenSSL](https://en.wikipedia.org/wiki/OpenSSL), [phpMyAdmin](https://en.wikipedia.org/wiki/PhpMyAdmin), [MediaWiki](https://en.wikipedia.org/wiki/MediaWiki), [Joomla](https://en.wikipedia.org/wiki/Joomla), [WordPress](https://en.wikipedia.org/wiki/WordPress) and more. Self- contained, multiple instances of XAMPP can exist on a single computer, and any given instance can be copied from one computer to another. XAMPP is offered in both a full and a standard version (Smaller version).

##### Usage

The most obvious characteristic of XAMPP is the ease at which a [WAMP](https://en.wikipedia.org/wiki/WAMP) webserver stack can be deployed and instantiated Later, some common packaged applications that could be easily installed were provided by [Bitnami](https://en.wikipedia.org/wiki/Bitnami).

Officially, XAMPP's designers intended it for use only as a development tool, to allow website designers and programmers to test their work on their own computers without any access to the Internet. To make this as easy as possible, many important security features are disabled by default. XAMPP has the ability to serve web pages on the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web).[[11]](https://en.wikipedia.org/wiki/XAMPP#cite_note-useage-11) A special tool is provided to [password-protect](https://en.wikipedia.org/wiki/Password) the most important parts of the package.[[12]](https://en.wikipedia.org/wiki/XAMPP#cite_note-12)

XAMPP also provides support for creating and manipulating databases in [MariaDB](https://en.wikipedia.org/wiki/MariaDB) and [SQLite](https://en.wikipedia.org/wiki/SQLite), among others.

Once XAMPP is installed, it is possible to treat a [local host](https://en.wikipedia.org/wiki/Localhost) like a remote host by connecting using an [FTP](https://en.wikipedia.org/wiki/File_Transfer_Protocol) client. Using a program like [FileZilla](https://en.wikipedia.org/wiki/FileZilla) has many advantages when installing a [content management system](https://en.wikipedia.org/wiki/Content_management_system) (CMS) like [Joomla](https://en.wikipedia.org/wiki/Joomla) or [WordPress](https://en.wikipedia.org/wiki/WordPress) It is also possible to connect to localhost via FTP with an [HTML editor](https://en.wikipedia.org/wiki/HTML_editor).

## MySQL

MySQL is an [open-source](https://en.wikipedia.org/wiki/Open-source_software) [relational database management system](https://en.wikipedia.org/wiki/Relational_database_management_system) (RDBMS). Its name is a combination of "My", the name of co-founder [Michael Widenius](https://en.wikipedia.org/wiki/Michael_Widenius)'s daughter My, and "SQL", the acronym for [Structured Query Language](https://en.wikipedia.org/wiki/Structured_Query_Language). A [relational database](https://en.wikipedia.org/wiki/Relational_database) organizes data into one or more data tables in which data may be related to each other; these relations help structure the data. SQL is a language that programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an [operating system](https://en.wikipedia.org/wiki/Operating_system) to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

MySQL is [free and open-source software](https://en.wikipedia.org/wiki/Free_and_open-source_software) under the terms of the [GNU General Public License](https://en.wikipedia.org/wiki/GNU_General_Public_License), and is also available under a variety of [proprietary](https://en.wikipedia.org/wiki/Proprietary_software) licenses. MySQL was owned and sponsored by the [Swedish](https://en.wikipedia.org/wiki/Sweden) company [MySQL AB](https://en.wikipedia.org/wiki/MySQL_AB), which was bought by [Sun](https://en.wikipedia.org/wiki/Sun_Microsystems) [Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems) (now [Oracle Corporation](https://en.wikipedia.org/wiki/Oracle_Corporation)). In 2010, when [Oracle acquired Sun](https://en.wikipedia.org/wiki/Acquisition_of_Sun_Microsystems_by_Oracle_Corporation), Widenius [forked](https://en.wikipedia.org/wiki/Fork_(software_development)) the [open-source](https://en.wikipedia.org/wiki/Open-source) MySQL project to create [MariaDB](https://en.wikipedia.org/wiki/MariaDB).

MySQL has stand-alone clients that allow users to interact directly with a MySQL database using SQL, but more often, MySQL is used with other programs to implement applications that need relational database capability. MySQL is a component of the [LAMP](https://en.wikipedia.org/wiki/LAMP_(software_bundle)) [web](https://en.wikipedia.org/wiki/Web_application) [application](https://en.wikipedia.org/wiki/Web_application) [software stack](https://en.wikipedia.org/wiki/Software_stack) (and [others](https://en.wikipedia.org/wiki/List_of_AMP_packages)), which is an acronym for [*Linux*](https://en.wikipedia.org/wiki/Linux)*,* [*Apache*](https://en.wikipedia.org/wiki/Apache_HTTP_Server)*, MySǪL,* [*Perl*](https://en.wikipedia.org/wiki/Perl)*/*[*PHP*](https://en.wikipedia.org/wiki/PHP)*/*[*Python*](https://en.wikipedia.org/wiki/Python_(programming_language)). MySQL is used by many database-driven web applications, including [Drupal](https://en.wikipedia.org/wiki/Drupal), [Joomla](https://en.wikipedia.org/wiki/Joomla), [phpBB](https://en.wikipedia.org/wiki/PhpBB), and [WordPress](https://en.wikipedia.org/wiki/WordPress). MySQL is also used by many popular [websites](https://en.wikipedia.org/wiki/Website), including [Facebook](https://en.wikipedia.org/wiki/Facebook), [Flickr](https://en.wikipedia.org/wiki/Flickr), [MediaWiki](https://en.wikipedia.org/wiki/MediaWiki), [Twitter](https://en.wikipedia.org/wiki/Twitter), and [YouTube](https://en.wikipedia.org/wiki/YouTube).

##### Overview

MySQL is written in [C](https://en.wikipedia.org/wiki/C_(programming_language)) and [C++](https://en.wikipedia.org/wiki/C%2B%2B). Its SQL parser is written in [yacc](https://en.wikipedia.org/wiki/Yacc), but it uses a home- brewed [lexical analyzer](https://en.wikipedia.org/wiki/Lexical_analysis). MySQL works on many [system platforms](https://en.wikipedia.org/wiki/System_platform), including [AIX](https://en.wikipedia.org/wiki/AIX_operating_system), [BSDi](https://en.wikipedia.org/wiki/BSD/OS), [FreeBSD](https://en.wikipedia.org/wiki/FreeBSD), [HP-UX](https://en.wikipedia.org/wiki/HP-UX), [ArcaOS](https://en.wikipedia.org/wiki/ArcaOS), [eComStation](https://en.wikipedia.org/wiki/EComStation), [IBM](https://en.wikipedia.org/wiki/IBM_i)

[i](https://en.wikipedia.org/wiki/IBM_i), [IRIX](https://en.wikipedia.org/wiki/IRIX), [Linux](https://en.wikipedia.org/wiki/Linux), [macOS](https://en.wikipedia.org/wiki/MacOS), [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows), [NetBSD](https://en.wikipedia.org/wiki/NetBSD), [Novell](https://en.wikipedia.org/wiki/Novell_NetWare) [NetWare](https://en.wikipedia.org/wiki/Novell_NetWare), [OpenBSD](https://en.wikipedia.org/wiki/OpenBSD), [OpenSolaris](https://en.wikipedia.org/wiki/OpenSolaris), [OS/2](https://en.wikipedia.org/wiki/OS/2) Warp, [QNX](https://en.wikipedia.org/wiki/QNX), [Oracle Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system)), [Symbian](https://en.wikipedia.org/wiki/Symbian), [SunOS](https://en.wikipedia.org/wiki/SunOS), [SCO](https://en.wikipedia.org/wiki/SCO_OpenServer) [OpenServer](https://en.wikipedia.org/wiki/SCO_OpenServer), SCO [UnixWare](https://en.wikipedia.org/wiki/UnixWare), Sanos and [Tru64](https://en.wikipedia.org/wiki/Tru64_UNIX). A port of MySQL to [OpenVMS](https://en.wikipedia.org/wiki/OpenVMS) also exists.[[18]](https://en.wikipedia.org/wiki/MySQL#cite_note-18)

The MySQL server software itself and the client libraries use [dual-licensing](https://en.wikipedia.org/wiki/Dual_license) distribution. They are offered under [GPL](https://en.wikipedia.org/wiki/GNU_General_Public_License) version 2, or a proprietary license.

Support can be obtained from the official manual. Free support additionally is available in different IRC channels and forums. Oracle offers paid support via its

MySQL Enterprise products. They differ in the scope of services and in price. Additionally, a number of third party organisations exist to provide support and services.

MySQL has received positive reviews, and reviewers noticed it "performs extremely well in the average case" and that the "developer interfaces are there, and the documentation (not to mention feedback in the real world via Web sites and the like) is very, very good".It has also been tested to be a "fast, stable and true multi-user, multi-threaded SQL database server".

#### History

****

David Axmark (left) and Michael "Monty" Widenius, founders of MySQL AB, in 2003

MySQL was created by a Swedish company, [MySQL AB](https://en.wikipedia.org/wiki/MySQL_AB), founded by [Swedes](https://en.wikipedia.org/wiki/Swedes) [David Axmark](https://en.wikipedia.org/wiki/David_Axmark), Allan Larsson and [Finnish](https://en.wikipedia.org/wiki/Finns) [Michael "Monty" Widenius](https://en.wikipedia.org/wiki/Michael_Widenius). Original development of MySQL by Widenius and Axmark began in 1994. The first version of MySQL appeared on 23 May 1995. It was initially created for personal usage from [mSQL](https://en.wikipedia.org/wiki/MSQL) based on the low-level language [ISAM](https://en.wikipedia.org/wiki/ISAM), which the creators considered too slow and inflexible. They created a new [SQL](https://en.wikipedia.org/wiki/SQL) interface, while keeping the same [API](https://en.wikipedia.org/wiki/API) as mSQL. By keeping the API consistent with the mSQL system, many developers were able to use MySQL instead of the (proprietarily licensed) mSQL antecedent.

##### Milestones

Additional milestones in MySQL development included:

* First internal release on 23 May 1995
* Version 3.19: End of 1996, from [www.tcx.se](http://www.tcx.se/)
* Version 3.20: January 1997
* Windows version was released on 8 January 1998 for Windows 95 and NT
* Version 3.21: production release 1998, from [www.mysql.com](http://www.mysql.com/)
* Version 3.22: alpha, beta from 1998
* Version 3.23: beta from June 2000, production release 22 January 2001
* Version 4.0: beta from August 2002, production release March 2003 ([unions](https://en.wikipedia.org/wiki/Set_operations_(SQL))).
* Version 4.1: beta from June 2004, production release October 2004 ([R-trees](https://en.wikipedia.org/wiki/R-tree) and [B-](https://en.wikipedia.org/wiki/B-tree) [trees](https://en.wikipedia.org/wiki/B-tree), subqueries, prepared statements).
* Version 5.0: beta from March 2005, production release October 2005 (cursors, stored procedures, triggers, views, [XA transactions](https://en.wikipedia.org/wiki/Database_transaction)).
  + The developer of the Federated Storage Engine states that "The Federated Storage Engine is a [proof-of-concept](https://en.wikipedia.org/wiki/Proof_of_concept) storage engine", but the main distributions of MySQL version 5.0 included it and turned it on by default. Documentation of some of the short-comings appears in "MySQL Federated Tables: The Missing Manual".
* [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems) acquired MySQL AB in 2008.[[28]](https://en.wikipedia.org/wiki/MySQL#cite_note-28)
* Version 5.1: production release 27 November 2008 (event scheduler, [partitioning](https://en.wikipedia.org/wiki/Partition_(database)), plugin API, row-based replication, [server log](https://en.wikipedia.org/wiki/Server_log) tables)
  + Version 5.1 contained 20 known crashing and wrong result bugs in addition to the 35 present in version 5.0 *(almost all fixed as of release 5.1.51)*.
  + MySQL 5.1 and 6.0-alpha showed poor performance when used for [data](https://en.wikipedia.org/wiki/Data_warehousing) [warehousing](https://en.wikipedia.org/wiki/Data_warehousing) – partly due to its inability to utilize multiple CPU cores for processing a single query.
* Oracle acquired Sun Microsystems on 27 January 2010.
* The day Oracle announced the purchase of Sun, Michael "Monty" Widenius forked MySQL, launching [MariaDB](https://en.wikipedia.org/wiki/MariaDB), and took a swath of MySQL developers with him.



Geir Høydalsvik, current Senior Software Development Director for MySQL at Oracle in 2018

* MySQL Server 5.5 was generally available (as of December 2010). Enhancements and features include:
  + The default storage engine is [InnoDB](https://en.wikipedia.org/wiki/InnoDB), which supports transactions and referential integrity constraints.
  + Improved InnoDB I/O subsystem
  + Improved [SMP](https://en.wikipedia.org/wiki/Symmetric_multiprocessing) support
  + Semisynchronous replication.
  + SIGNAL and RESIGNAL statement in compliance with the SQL standard.
  + Support for supplementary Unicode character sets utf16, utf32, and utf8mb4.
  + New options for user-defined partitioning.
* MySQL Server 6.0.11-alpha was announcedon 22 May 2009 as the last release of the

6.0 line. Future MySQL Server development uses a New Release Model. Features developed for 6.0 are being incorporated into future releases.

* The general availability of MySQL 5.6 was announced in February 2013. New features included performance improvements to the [query optimizer](https://en.wikipedia.org/wiki/Query_optimizer), higher transactional throughput in InnoDB, new [NoSQL](https://en.wikipedia.org/wiki/NoSQL)-style memcached APIs, improvements to partitioning for querying and managing very large tables, TIMESTAMP column type that correctly stores milliseconds, improvements to replication, and better performance monitoring by expanding the data available through the PERFORMANCE\_SCHEMA. The InnoDB .
* storage engine also included support for full-text search and improved group commit performance.
* The general availability of MySQL 5.7 was announced in October 2015.[]](https://en.wikipedia.org/wiki/MySQL#cite_note-41)As of MySQL 5.7.8, August 2015, MySQL supports a native [JSON](https://en.wikipedia.org/wiki/JSON) data type defined by RFC 7159.
* MySQL Server 8.0 was announced in April 2018,[[43]](https://en.wikipedia.org/wiki/MySQL#cite_note-44) including NoSQL Document Store, atomic and crash safe DDL sentences and [JSON](https://en.wikipedia.org/wiki/JSON) Extended syntax, new functions, such as JSON table functions, improved sorting, and partial updates. Previous MySQL Server 8.0.0-dmr (Milestone Release) was announced 12 September 2016.
* MySQL was declared DBMS of the year 2019 from the [DB-Engines ranking](https://en.wikipedia.org/wiki/DB-Engines_ranking)

##### Features

MySQL is offered under two different editions: the [open source](https://en.wikipedia.org/wiki/Open-source_software) MySQL Community Server and the proprietary [Enterprise Server](https://en.wikipedia.org/wiki/MySQL_Enterprise). MySQL Enterprise Server is differentiated by a series of proprietary extensions which install as server plugins, but otherwise shares the version numbering system and is built from the same code base.

Major features as available in MySQL 5.6:

* A broad subset of ANSI SQL 99, as well as extensions
* Cross-platform support
* [Stored procedures](https://en.wikipedia.org/wiki/Stored_procedure), using a procedural language that closely adheres to [SQL/PSM](https://en.wikipedia.org/wiki/SQL/PSM)
* [Triggers](https://en.wikipedia.org/wiki/Database_trigger)
* [Cursors](https://en.wikipedia.org/wiki/Cursor_(databases))
* Updatable [views](https://en.wikipedia.org/wiki/View_(SQL))
* Online [Data Definition Language](https://en.wikipedia.org/wiki/Data_Definition_Language) (DDL) when using the InnoDB Storage Engine.
* [Information schema](https://en.wikipedia.org/wiki/Information_schema)
* Performance Schema that collects and aggregates statistics about server execution and query performance for monitoring purposes.
* A set of SQL Mode options to control [runtime](https://en.wikipedia.org/wiki/Run_time_(program_lifecycle_phase)) behavior, including a strict mode to better adhere to SQL standards.
* [X/Open XA](https://en.wikipedia.org/wiki/X/Open_XA) [distributed transaction processing](https://en.wikipedia.org/wiki/Distributed_transaction_processing) (DTP) support; [two phase commit](https://en.wikipedia.org/wiki/Two-phase_commit_protocol) as part of this, using the default [InnoDB](https://en.wikipedia.org/wiki/InnoDB) storage engine
* Transactions with [savepoints](https://en.wikipedia.org/wiki/Savepoint) when using the default InnoDB Storage Engine. The NDB Cluster Storage Engine also supports transactions.
* [ACID](https://en.wikipedia.org/wiki/ACID) compliance when using InnoDB and NDB Cluster Storage Engines
* [SSL](https://en.wikipedia.org/wiki/Secure_Sockets_Layer) support
* Query [caching](https://en.wikipedia.org/wiki/Cache_(computing))
* Sub-[SELECTs](https://en.wikipedia.org/wiki/Select_(SQL)) (i.e. nested SELECTs)
* Built-in [replication](https://en.wikipedia.org/wiki/Database_replication) support
  + Asynchronous replication: [master-slave](https://en.wikipedia.org/wiki/Master/slave_(technology)) from one master to many slaves or many masters to one slave
  + Semi synchronous replication: Master to slave replication where the master waits on replication
  + Synchronous replication: [Multi-master replication](https://en.wikipedia.org/wiki/Multi-master_replication) is provided in [MySQL](https://en.wikipedia.org/wiki/MySQL_Cluster) [Cluster](https://en.wikipedia.org/wiki/MySQL_Cluster).
  + [Virtual Synchronous](https://en.wikipedia.org/wiki/Virtual_synchrony): Self managed groups of MySQL servers with multi master support can be done using: Galera Cluster or the built in Group Replication plugin
* Full text [indexing](https://en.wikipedia.org/wiki/Index_(database)) and searching
* Embedded database library
* [Unicode](https://en.wikipedia.org/wiki/Unicode) support
* Partitioned tables with pruning of partitions in optimizer
* [Shared-nothing](https://en.wikipedia.org/wiki/Shared-nothing) clustering through [MySQL Cluster](https://en.wikipedia.org/wiki/MySQL_Cluster)
* Multiple storage engines, allowing one to choose the one that is most effective for each table in the application.
* Native storage engines [InnoDB](https://en.wikipedia.org/wiki/InnoDB), [MyISAM](https://en.wikipedia.org/wiki/MyISAM), Merge, Memory (heap), [Federated](https://en.wikipedia.org/wiki/MySQL_Federated), Archive, [CSV](https://en.wikipedia.org/wiki/Comma-separated_values), Blackhole, NDB Cluster.
* Commit grouping, gathering multiple transactions from multiple connections together to increase the number of commits per second.

The developers release minor updates of the MySQL Server approximately every two months. The sources can be obtained from MySQL's website or from MySQL's [GitHub](https://en.wikipedia.org/wiki/GitHub) repository, both under the GPL license.

## JavaScript

**JavaScript** (/ˈdʒɑːvəskrɪpt/), often abbreviated as **JS**, is a [programming language](https://en.wikipedia.org/wiki/Programming_language) and core technology of [the Web](https://en.wikipedia.org/wiki/World_Wide_Web), alongside [HTML](https://en.wikipedia.org/wiki/HTML) and [CSS](https://en.wikipedia.org/wiki/CSS). 99% of [websites](https://en.wikipedia.org/wiki/Website) use JavaScript on the [client](https://en.wikipedia.org/wiki/Client_(computing)) side for [webpage](https://en.wikipedia.org/wiki/Web_page) behavior.

[Web browsers](https://en.wikipedia.org/wiki/Web_browser) have a dedicated [JavaScript engine](https://en.wikipedia.org/wiki/JavaScript_engine) that executes the client [code](https://en.wikipedia.org/wiki/Source_code). These engines are also utilized in some [servers](https://en.wikipedia.org/wiki/Server_(computing)) and a variety of [apps](https://en.wikipedia.org/wiki/Application_software). The most popular [runtime](https://en.wikipedia.org/wiki/Runtime_system) [system](https://en.wikipedia.org/wiki/Runtime_system) for non-browser usage is [Node.js](https://en.wikipedia.org/wiki/Node.js).

JavaScript is a [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), often [just-in-time compiled](https://en.wikipedia.org/wiki/Just-in-time_compilation) language that conforms to the [ECMAScript](https://en.wikipedia.org/wiki/ECMAScript) standard It has [dynamic typing](https://en.wikipedia.org/wiki/Dynamic_typing), [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming) [object-orientation](https://en.wikipedia.org/wiki/Object-oriented_programming), and [first-class functions](https://en.wikipedia.org/wiki/First-class_function). It is [multi-paradigm](https://en.wikipedia.org/wiki/Programming_paradigm), supporting [event-driven](https://en.wikipedia.org/wiki/Event-driven_programming), [functional](https://en.wikipedia.org/wiki/Functional_programming), and [imperative](https://en.wikipedia.org/wiki/Imperative_programming) [programming styles](https://en.wikipedia.org/wiki/Programming_paradigm). It has [application programming interfaces](https://en.wikipedia.org/wiki/Application_programming_interface) (APIs) for working with text, dates, [regular expressions](https://en.wikipedia.org/wiki/Regular_expression), standard [data structures](https://en.wikipedia.org/wiki/Data_structure), and the [Document](https://en.wikipedia.org/wiki/Document_Object_Model) [Object Model](https://en.wikipedia.org/wiki/Document_Object_Model) (DOM).

The ECMAScript standard does not include any [input/output](https://en.wikipedia.org/wiki/Input/output) (I/O), such as [networking](https://en.wikipedia.org/wiki/Computer_network), [storage](https://en.wikipedia.org/wiki/Data_storage), or [graphics](https://en.wikipedia.org/wiki/Computer_graphics) facilities. In practice, the web browser or other runtime system provides JavaScript APIs for I/O.

Although [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) and JavaScript are similar in name and [syntax](https://en.wikipedia.org/wiki/Syntax_(programming_languages)), the two languages are distinct and differ greatly in design.

##### History

**Creation at Netscape**

The first popular [web browser](https://en.wikipedia.org/wiki/Web_browser) with a [graphical user interface](https://en.wikipedia.org/wiki/Graphical_user_interface), [Mosaic](https://en.wikipedia.org/wiki/Mosaic_(web_browser)), was released in 1993. Accessible to non-technical people, it played a prominent role in the rapid growth of the early [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web). The lead developers of Mosaic then founded the [Netscape](https://en.wikipedia.org/wiki/Netscape) corporation, which released a more polished browser, [Netscape Navigator](https://en.wikipedia.org/wiki/Netscape_Navigator), in 1994. This quickly became the most-used.

During these formative years of the Web, [web pages](https://en.wikipedia.org/wiki/Web_page) could only be static, lacking the capability for dynamic behavior after the page was loaded in the browser. There was a desire in the flourishing web development scene to remove this limitation, so in 1995, Netscape decided to add a [programming language](https://en.wikipedia.org/wiki/Programming_language) to Navigator. They pursued two routes to achieve this: collaborating with [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems) to embed the [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) language, while also hiring [Brendan Eich](https://en.wikipedia.org/wiki/Brendan_Eich) to embed the [Scheme](https://en.wikipedia.org/wiki/Scheme_(programming_language)) language.

The goal was a "language for the masses","to help nonprogrammers create dynamic, interactive [Web sites](https://en.wikipedia.org/wiki/Website)".Netscape management soon decided that the best option was for Eich to devise a new language, with syntax similar to Java and less like Scheme or other

extant [scripting languages](https://en.wikipedia.org/wiki/Scripting_language). Although the new language and its [interpreter](https://en.wikipedia.org/wiki/Interpreter_(computing)) implementation were called LiveScript when first shipped as part of a Navigator [beta](https://en.wikipedia.org/wiki/Beta_software) in September 1995, the name was changed to JavaScript for the official release in December.

The choice of the JavaScript name has caused confusion, implying that it is directly related to Java. At the time, the [dot-com boom](https://en.wikipedia.org/wiki/Dot-com_bubble) had begun and Java was a popular new language, so Eich considered the JavaScript name a marketing ploy by Netscape.

##### Adoption by Microsoft

[Microsoft](https://en.wikipedia.org/wiki/Microsoft) debuted [Internet Explorer](https://en.wikipedia.org/wiki/Internet_Explorer) in 1995, leading to a [browser war](https://en.wikipedia.org/wiki/Browser_war) with Netscape. On the JavaScript front, Microsoft created its own [interpreter](https://en.wikipedia.org/wiki/Interpreter_(computing)) called [JScript](https://en.wikipedia.org/wiki/JScript).

Microsoft first released JScript in 1996, alongside initial support for [CSS](https://en.wikipedia.org/wiki/CSS) and extensions to [HTML](https://en.wikipedia.org/wiki/HTML). Each of these [implementations](https://en.wikipedia.org/wiki/Implementation) was noticeably different from their counterparts in [Netscape Navigator](https://en.wikipedia.org/wiki/Netscape_Navigator). These differences made it difficult for developers to make their websites work well in both browsers, leading to widespread use of "best viewed in Netscape" and "best viewed in Internet Explorer" logos for several years.

##### The rise of JScript

[Brendan Eich](https://en.wikipedia.org/wiki/Brendan_Eich) later said of this period: "It's still kind of a [sidekick](https://en.wikipedia.org/wiki/Sidekick) language. It's considered slow or annoying. People do [pop-ups](https://en.wikipedia.org/wiki/Pop-up_ad) or those scrolling messages in the old [status bar](https://en.wikipedia.org/wiki/Status_bar) at the bottom of your old [browser](https://en.wikipedia.org/wiki/Web_browser)."

In November 1996, [Netscape](https://en.wikipedia.org/wiki/Netscape) submitted JavaScript to [Ecma International](https://en.wikipedia.org/wiki/Ecma_International), as the starting point for a standard specification that all browser vendors could conform to. This led to the official release of the first [ECMAScript](https://en.wikipedia.org/wiki/ECMAScript) language specification in June 1997.

The standards process continued for a few years, with the release of ECMAScript 2 in June 1998 and ECMAScript 3 in December 1999. Work on ECMAScript 4 began in 2000.

However, the effort to fully standardize the language was undermined by [Microsoft](https://en.wikipedia.org/wiki/Microsoft) gaining an increasingly dominant position in the browser market. By the early 2000s, [Internet Explorer'](https://en.wikipedia.org/wiki/Internet_Explorer)s market share reached 95%.This meant that [JScript](https://en.wikipedia.org/wiki/JScript) became the de facto standard for [client-](https://en.wikipedia.org/wiki/Client-side_scripting) [side scripting](https://en.wikipedia.org/wiki/Client-side_scripting) on the Web.

Microsoft initially participated in the standards process and implemented some proposals in its JScript language, but eventually it stopped collaborating on ECMA work. Thus ECMAScript 4 was mothballed.

## React JavaScript

**React** (also known as **React.js** or **ReactJS**) is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source_software) [front-end](https://en.wikipedia.org/wiki/Frontend_and_backend) [JavaScript](https://en.wikipedia.org/wiki/JavaScript_library) [library](https://en.wikipedia.org/wiki/JavaScript_library) that aims to make building [user interfaces](https://en.wikipedia.org/wiki/User_interface) based on [components](https://en.wikipedia.org/wiki/Component-based_software_engineering) more "seamless".[[5]](https://en.wikipedia.org/wiki/React_(software)#cite_note-reactjs.org-3-5) It is maintained by [Meta](https://en.wikipedia.org/wiki/Meta_Platforms) (formerly Facebook) and a community of individual developers and companies.

React can be used to develop [single-page](https://en.wikipedia.org/wiki/Single-page_application), mobile, or [server-rendered](https://en.wikipedia.org/wiki/Server-side_rendering) applications with frameworks like [Next.js](https://en.wikipedia.org/wiki/Next.js). Because React is only concerned with the user interface and rendering components to the [DOM](https://en.wikipedia.org/wiki/Document_Object_Model), React applications often rely on [libraries](https://en.wikipedia.org/wiki/JavaScript_libraries) for routing and other client-side functionality.[[10](https://en.wikipedia.org/wiki/React_(software)#cite_note-FOOTNOTEDere2017-10)[][11]](https://en.wikipedia.org/wiki/React_(software)#cite_note-FOOTNOTEPanchal2022-11) A key advantage of React is that it only re-renders those parts of the page that have changed, avoiding unnecessary re-rendering of unchanged DOM elements.

##### Notable features Declarative

React adheres to the [declarative programming](https://en.wikipedia.org/wiki/Declarative_programming) [paradigm](https://en.wikipedia.org/wiki/Programming_paradigm). Developers design views for each state of an application, and React updates and renders components when data changes. This is in contrast with [imperative programming](https://en.wikipedia.org/wiki/Imperative_programming).

##### Components

React code is made of entities called [components](https://en.wikipedia.org/wiki/Component-based_software_engineering). These components are modular and reusable React applications typically consist of many layers of components. The components are rendered to a root element in the [DOM](https://en.wikipedia.org/wiki/Document_Object_Model) using the React DOM library. When rendering a component, values are passed between components through *props* (short for "properties")*.* Values internal to a component are called its *state.*

The two primary ways of declaring components in React are through function components and class components.

##### Function components

Function components are declared with a function (using JavaScript function syntax or an [arrow function expression](https://en.wikipedia.org/wiki/Anonymous_function)) that accepts a single "props" argument and returns JSX. From React v16.8 onwards, function components can use state with the useState Hook.

##### React Hooks

On February 16, 2019, React 16.8 was released to the public, introducing React Hooks.[[17]](https://en.wikipedia.org/wiki/React_(software)#cite_note-17) Hooks are functions that let developers "hook into" React state and lifecycle features from function components.[[18]](https://en.wikipedia.org/wiki/React_(software)#cite_note-18) Notably, Hooks do not work inside classes — they let developers use more features of React without classes.[[19]](https://en.wikipedia.org/wiki/React_(software)#cite_note-19)

React provides several built-in hooks such as useState, useContext, useReducer useMemo and useEffect. Others are documented in the Hooks API Reference. useState and useEffect, which are the most commonly used, are for controlling stateand side effects, respectively.

##### Rules of hooks

There are two rules of hooks which describe the characteristic code patterns that hooks rely on:

1. "Only call hooks at the top level" — do not call hooks from inside loops, conditions, or nested statements so that the hooks are called in the same order each render.
2. "Only call hooks from React functions" — do not call hooks from plain JavaScript functions so that stateful logic stays with the component.

Although these rules cannot be enforced at runtime, code analysis tools such as [linters](https://en.wikipedia.org/wiki/Lint_(software)) can be configured to detect many mistakes during development. The rules apply to both usage of Hooks and the implementation of custom Hooks, which may call other Hooks.

##### Server components

React server components (RSC) are function components that run exclusively on the server. The concept was first introduced in the talk "Data Fetching with Server Components".[[27]](https://en.wikipedia.org/wiki/React_(software)#cite_note-27) Though a similar concept to Server Side Rendering, RSCs do not send corresponding JavaScript to the client as no hydration occurs. As a result, they have no access to hooks. However, they may be [asynchronous function](https://en.wikipedia.org/wiki/Async/await), allowing them to directly perform asynchronous operations:

**CODE EX: -**

**async function** MyComponent() {

**const** message = **await** fetchMessageFromDb();

##### return (

<**div**>Message: {message}</**div**>

);

}

Currently, server components are most readily usable with [Next.js](https://en.wikipedia.org/wiki/Next.js).

##### Class components

Class components are declared using [ES6](https://en.wikipedia.org/wiki/ECMAScript) classes. They behave the same way that function components do, but instead of using Hooks to manage state and lifecycle events, they use the lifecycle methods on the React.Component [base class](https://en.wikipedia.org/wiki/Inheritance_(object-oriented_programming)).

**CODE EX: -**

**class** ParentComponent **extends** React.Component { state = { color: 'green' };

render() {

##### return (

<**ChildComponent** color={**this**.state.color} />

);

}

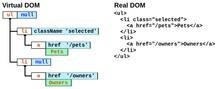
}

The introduction of React Hooks with React 16.8 in February 2019 allowed developers to manage state and lifecycle behaviors within functional components, reducing the reliance on class components.

This trend aligns with the broader industry movement towards functional programming and modular design. As React continues to evolve, it is essential for developers to consider the benefits of functional components and React Hooks when building new applications or refactoring existing ones

##### Routing

React itself does not come with built-in support for [routing](https://en.wikipedia.org/wiki/Routing). React is primarily a library for building user interfaces, and it does not include a full-fledged routing solution out of the box. Third-party libraries can be used to handle routing in React applications.[[29]](https://en.wikipedia.org/wiki/React_(software)#cite_note-29) It allows the developer to define routes, manage navigation, and handle URL changes in a React-friendly way.



There is a Virtual DOM that is used to implement the real DOM

##### Virtual DOM

Another notable feature is the use of a virtual [Document Object Model](https://en.wikipedia.org/wiki/Document_Object_Model), or [Virtual DOM](https://en.wikipedia.org/wiki/Virtual_DOM). React creates an [in-memory](https://en.wikipedia.org/wiki/In-memory_processing) data-structure cache, computes the resulting differences, and then updates the browser's displayed DOM efficiently.[[30]](https://en.wikipedia.org/wiki/React_(software)#cite_note-React_Blog-30) This process is called **reconciliation**. This allows the programmer to write code as if the entire page is rendered on each change, while React only renders the components that actually change. This selective rendering provides a major

##### JSX

[JSX](https://en.wikipedia.org/wiki/JSX_(JavaScript)), or JavaScript XML, is an extension to the JavaScript language syntax. Similar in appearance to HTML JSX provides a way to structure component rendering using syntax familiar to many developers. React components are typically written using JSX, although they do not have to be (components may also be written in pure JavaScript). JSX is similar to another extension syntax created by Facebook for [PHP](https://en.wikipedia.org/wiki/PHP) called [XHP](https://en.wikipedia.org/wiki/XHP).

##### An example of JSX code:

**class** App **extends** React.Component { render() {

##### return (

<**div**>

<**p**>Header</**p**>

<**p**>Content</**p**>

<**p**>Footer</**p**>

</**div**>

);

}

}

##### Architecture beyond HTML

The basic [architecture](https://en.wikipedia.org/wiki/Software_architecture) of React applies beyond rendering HTML in the browser. For example, Facebook has dynamic charts that render to <canvas> tags, and Netflix and [PayPal](https://en.wikipedia.org/wiki/PayPal) use universal loading to render identical HTML on both the server and client.

##### Server-side rendering

[Server-side rendering](https://en.wikipedia.org/wiki/Server-side_scripting) (SSR) refers to the process of rendering a client-side JavaScript application on the server, rather than in the browser. This can improve the performance of the application, especially for users on slower connections or devices.

With SSR, the initial HTML that is sent to the client includes the fully rendered UI of the application. This allows the client's browser to display the UI immediately, rather than having to wait for the JavaScript to download and execute before rendering the UI.

React supports SSR, which allows developers to render React components on the server and send the resulting HTML to the client. This can be useful for improving the performance of the application, as well as for [search engine optimization](https://en.wikipedia.org/wiki/Search_engine_optimization) purposes.

##### Common idioms

React does not attempt to provide a complete application library. It is designed specifically for building user interfaces and therefore does not include many of the tools some developers might consider necessary to build an application. This allows the choice of whichever libraries the developer prefers to accomplish tasks such as performing network access or local data storage. Common patterns of usage have emerged as the library matures

##### Unidirectional data flow

To support React's concept of unidirectional data flow (which might be contrasted with [AngularJS](https://en.wikipedia.org/wiki/AngularJS)'s bidirectional flow), the Flux architecture was developed as an alternative to the popular [model–view–controller](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller) architecture. Flux features *actions* which are sent through a central *dispatcher* to a *store*, and changes to the store are propagated back to the view.[]](https://en.wikipedia.org/wiki/React_(software)#cite_note-Flux-39) When used with React, this propagation is accomplished through component properties. Since its conception, Flux has been superseded by libraries such as [Redux](https://en.wikipedia.org/wiki/Redux_(JavaScript_library)) and MobX.

Flux can be considered a variant of the [observer pattern](https://en.wikipedia.org/wiki/Observer_pattern).

A React component under the Flux architecture should not directly modify any props passed to it, but should be passed [callback functions](https://en.wikipedia.org/wiki/Callback_function) that create *actions* which are sent by the dispatcher to modify the store. The action is an object whose responsibility is to describe what has taken place: for example, an action describing one user "following" another might contain a user id, a target user id, and the type USER\_FOLLOWED\_ANOTHER\_USER.[[42]](https://en.wikipedia.org/wiki/React_(software)#cite_note-42) The stores, which can be thought of as models, can alter themselves in response to actions received from the dispatcher.

This pattern is sometimes expressed as "properties flow down, actions flow up". Many implementations of Flux have been created since its inception, perhaps the most well-known being [Redux](https://en.wikipedia.org/wiki/Redux_(JavaScript_library)), which features a single store, often called a [single source of truth](https://en.wikipedia.org/wiki/Single_source_of_truth).[[43]](https://en.wikipedia.org/wiki/React_(software)#cite_note-43)

In February 2019, useReducer was introduced as a [React hook](https://en.wikipedia.org/wiki/React_(web_framework)#React_Hooks) in the 16.8 release. It provides an API that is consistent with Redux, enabling developers to create Redux-like stores that are local to component states.

##### Future development

Project status can be tracked via the core team discussion forum. However, major changes to React go through the Future of React repository issues and [pull requests](https://en.wikipedia.org/wiki/Pull_request). This enables the React community to provide feedback on new potential features, experimental APIs and JavaScript syntax improvements.

## CHAPTER 4 :- Project overview and code

##### Overview

This project aims to develop a web application using React and MySQL to efficiently track the repair process of faulty items. The application will provide a user-friendly interface for both customers and administrators to monitor the status of repair orders, generate reports, and manage inventory. By leveraging the power of React for a dynamic user interface and MySQL for efficient data storage and retrieval, this application will streamline the repair process and enhance customer satisfaction.

##### Objectives

* Streamline the tracking of faulty items (Laptop Speaker) during the repair process.
* Provide an interface for users to input and update key repair details.
* Offer insights into repair locations, inspection results, and approval statuses.
* Ensure a structured and organized process to avoid data loss or errors.

##### Technology Stack

* **Frontend**: React.js (with Material-UI or Bootstrap for design)
* **Backend**: Express.js (Node.js framework)
* **Database**: MySQL

##### Features

1. **User Role Management**:
   * **Admin**: Manage users, oversee all repairs, and generate reports.
   * **Customer/User**: Log and track the repair status of faulty items.

##### JWT login:

* + - * JWT stands for JSON Web Token, and it is a commonly used stateless user authentication standard used to securely transmit information between client and server in a JSON format. A JWT is encoded and not encrypted by default. It is digitally signed using a secret key known only to the server.

##### Data Integrity:

* + Store all data securely in a relational database with appropriate constraints.

##### Key Functionalities

1. **Form for Faulty Item Logging**:
   * Fields for item details such as:
     + Item Name (Pre-filled as *Motherboard*)
     + User name
     + User address
     + Item pickup location
     + Enter repair location (Ex: Delhi , Gurugram , Mumbai)
     + Enter phone no
     + Tell us about your Faulty iteam details

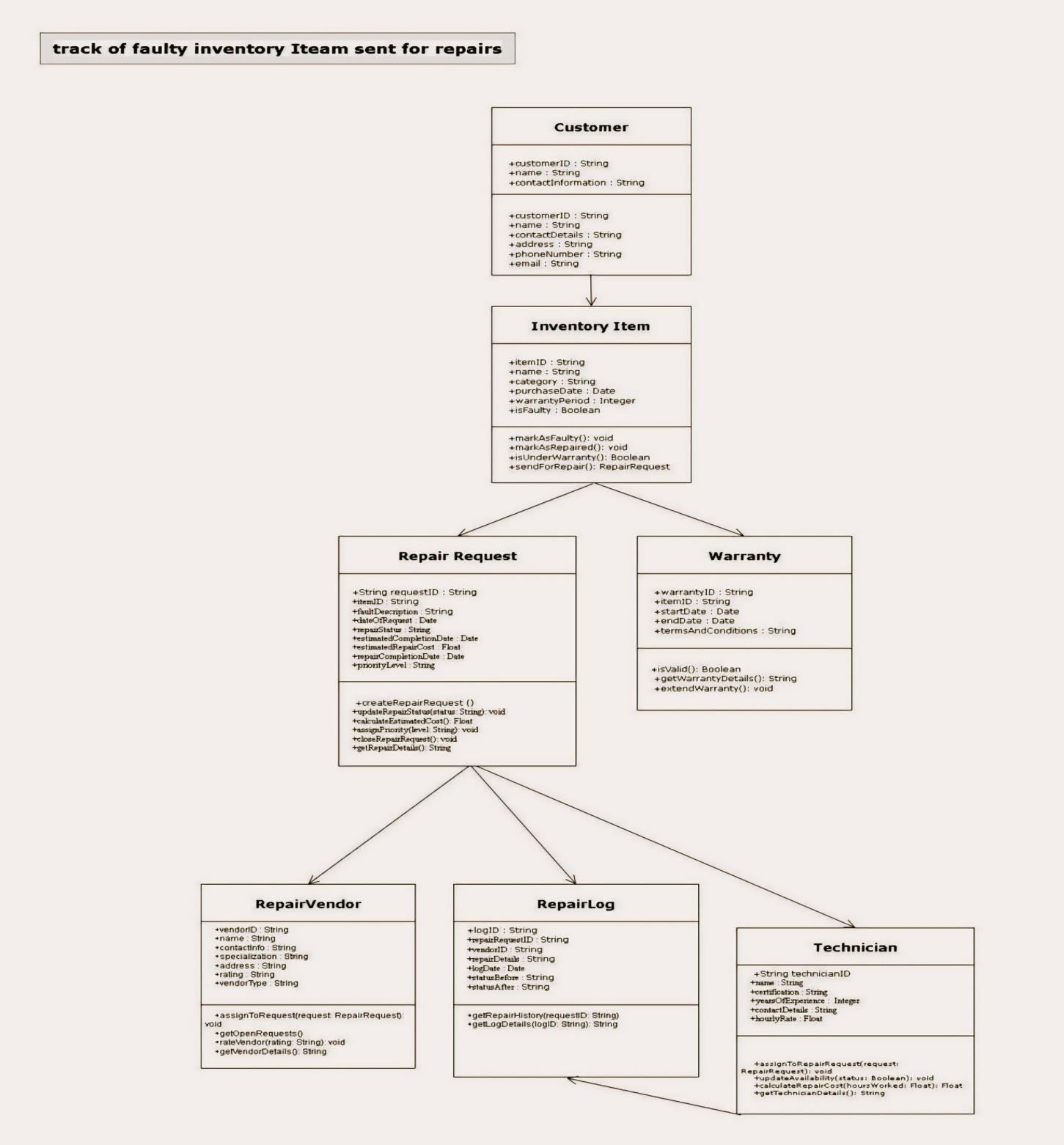
**Styling with Tailwind CSS**

The application's UI is styled using Tailwind CSS.

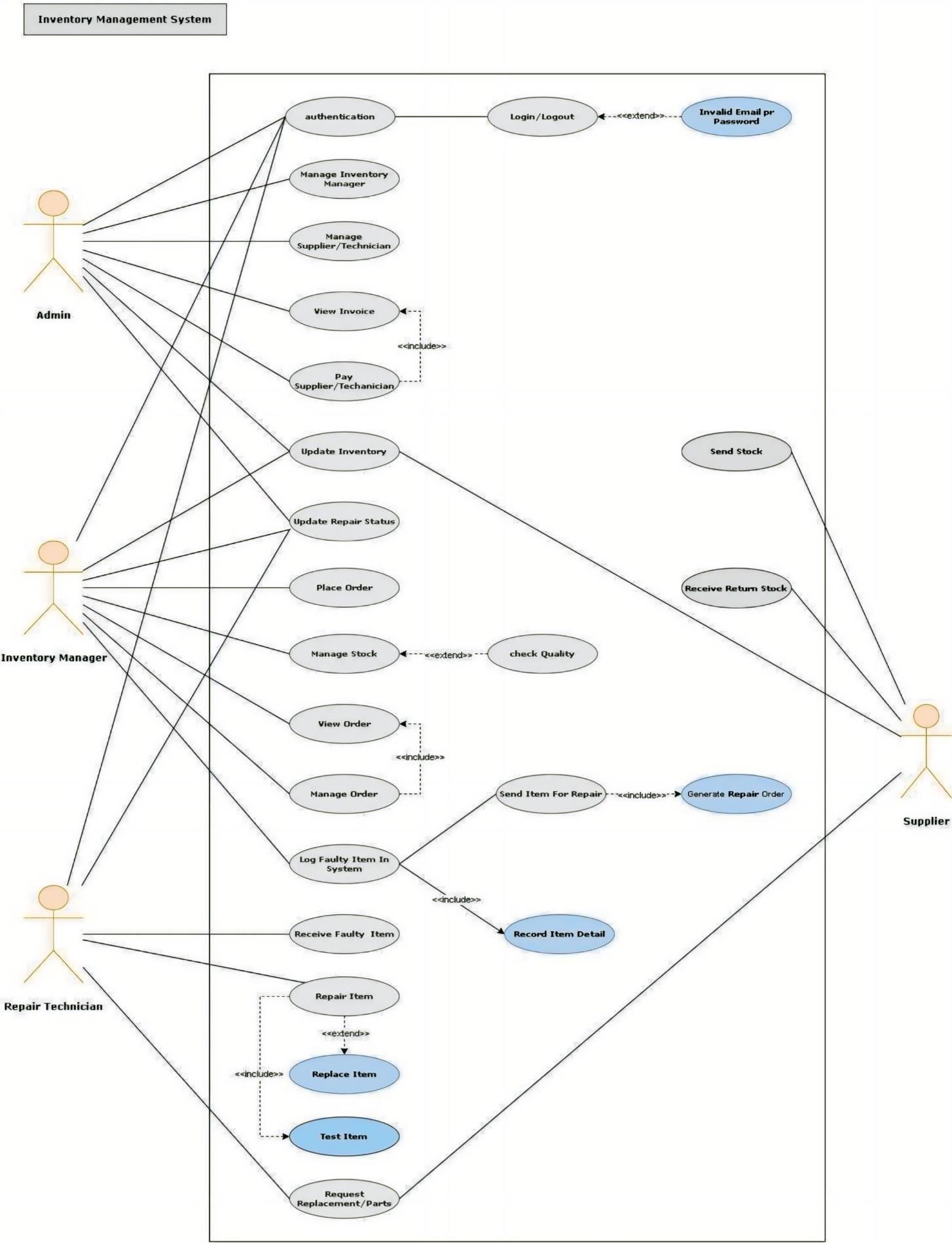
**Component Library**

This project utilizes Component Library Preline, ComponentLand for cards, pages, and other components.

#### Class Diagram

****

**Use Case Diagram**

****

##### Technical Implementation Frontend (React):

* User Interface: Design and develop an intuitive and visually appealing user interface using React components.
* Form Handling: Implement forms for item registration, status updates, and report generation.
* State Management: Utilize React's state management solutions (e.g., Redux or Context API) to manage application state efficiently.
* Data Fetching and Display: Fetch data from the backend API to populate the UI with real- time information.
* User Authentication: Implement secure authentication and authorization mechanisms to protect sensitive data.

##### Error Handling: Implement robust error handling to provide informative feedback to users.

**Backend (Node.js and Express.js):**

* API Development: Create RESTful APIs to handle requests from the frontend and interact with the database.
* Data Validation: Validate user input to ensure data integrity.
* Database Interactions: Use MySQL to store and retrieve data related to items, repairs, customers, and inventory.
* Email and SMS Notifications: Integrate with email and SMS services to send automated notifications.
* Security: Implement security measures to protect user data and prevent unauthorized access.

##### Database (MySQL):

* Database Design: Design a well-structured database schema to store information about items, repairs, customers, and inventory.
* Data Storage: Store data efficiently and securely in the MySQL database.
* Query Optimization: Optimize database queries to improve performance and scalability.

## Code snippets

## Server.js

const express = require("express");

const cors = require("cors");

const mysql = require("mysql2");

const app = express();

// Create database if it doesn't exist

const connection = mysql.createConnection({

  host: "localhost",

  user: "root",

  password: "",

});

connection.connect((err) => {

  if (err) throw err;

  console.log("Connected to MySQL server");

  connection.query("CREATE DATABASE IF NOT EXISTS fixit\_db", (err) => {

    if (err) throw err;

    console.log("Database 'fixit\_db' is ready");

  });

});

const complaintRoutes = require("./routes/complaintRoutes");

const customerRoutes = require("./routes/customerRoutes");

const adminRoutes = require("./routes/adminRoutes");

const userRoutes = require("./routes/userRoutes");

app.use(cors());

app.use(express.json());

app.use("/", complaintRoutes);

app.use("/", customerRoutes);

app.use("/", adminRoutes);

app.use("/", userRoutes);

app.listen(3000, () => {

  console.log("Application is running on port 3000");

});

**Complaint.jsx (**complaint form**)**

import React, { useState, useEffect } from "react";

import { useAuth } from "../utils/Auth";

const formatTimestamp = (timestamp) => {

  const date = new Date(timestamp);

  const options = {

    year: "numeric",

    month: "short",

    day: "numeric",

    hour: "numeric",

    minute: "numeric",

    second: "numeric",

  };

  return new Intl.DateTimeFormat("en-US", options).format(date);

};

const formatTimestamp1 = (timestamp) => {

  const date = new Date(timestamp);

  const options = {

    year: "numeric",

    month: "short",

    day: "numeric",

  };

  return new Intl.DateTimeFormat("en-US", options).format(date);

};

const ComplaintForm = () => {

  const { authToken, headers } = useAuth();

  const [name, setName] = useState("");

  const [description, setDescription] = useState("");

  const [fault, setFault] = useState("");

  const [address, setAddress] = useState(null); // Updated from blockId

  const [pickupLocation, setPickupLocation] = useState(null); // Updated from customerId

  const [phoneNo, setPhoneNo] = useState(""); // New state for phone\_no

  const [repairLocation, setRepairLocation] = useState(""); // New state for repair\_location

  const onSubmitForm = async (e) => {

    e.preventDefault();

    if (!name || name.trim() === "") {

      alert("Please enter a valid name.");

      return;

    }

    if (!fault || fault.trim() === "") {

      alert("Please enter Room No.");

      return;

    }

    if (!description || description.trim() === "") {

      alert("Please enter a valid complaint.");

      return;

    }

    try {

      const body = {

        name,

        description,

        fault,

        address, // Updated

        pickup\_location: pickupLocation, // Updated

        phone\_no: phoneNo, // Include phone\_no

        repair\_location: repairLocation, // Include repair\_location

      };

      const response = await fetch("http://localhost:3000/complaints", {

        method: "POST",

        headers: headers,

        body: JSON.stringify(body),

      });

      window.location = "/";

    } catch (err) {

      console.error(err.message);

    }

  };

  return (

    <>

      <section class="bg-gray-100 py-12 text-gray-800 sm:py-24 h-full">

        <div class="bg-gray-100 mx-auto flex max-w-md flex-col rounded-lg lg:max-w-screen-xl lg:flex-row">

          <div class="max-w-2xl px-4 lg:pr-24">

            <p class="mb-2 text-blue-600">Welcome to FixIT Hub</p>

            <h3 class="mb-5 text-3xl font-semibold">Submit Your Grievance</h3>

            <p class="mb-16 text-md text-gray-600">

              At FixIT Hub, we're more than just a repair platform. With a

              passion for problem-solving and a commitment to customer

              satisfaction, we've been serving our community with top-notch

              computer and gadget repairs since 2020.

            </p>

            <div class="mb-5 flex font-medium">

              <div class="mr-4">

                <svg

                  xmlns="http://www.w3.org/2000/svg"

                  fill="none"

                  viewBox="0 0 24 24"

                  stroke-width="1.5"

                  stroke="currentColor"

                  class="h-7 w-7 text-blue-500"

                >

                  <path

                    stroke-linecap="round"

                    stroke-linejoin="round"

                    d="M7.5 7.5h-.75A2.25 2.25 0 004.5 9.75v7.5a2.25 2.25 0 002.25 2.25h7.5a2.25 2.25 0 002.25-2.25v-7.5a2.25 2.25 0 00-2.25-2.25h-.75m-6 3.75l3 3m0 0l3-3m-3 3V1.5m6 9h.75a2.25 2.25 0 012.25 2.25v7.5a2.25 2.25 0 01-2.25 2.25h-7.5a2.25 2.25 0 01-2.25-2.25v-.75"

                  />

                </svg>

              </div>

              <div class="">

                <p class="mb-2">Our Mission</p>

                <span class="font-normal text-gray-600">

                  Our mission is simple: to breathe new life into your devices

                  and keep you connected to what matters most. Whether it's a

                  cracked screen, a sluggish system, or a mysterious

                  malfunction, our team of experienced technicians is here to

                  diagnose, repair, and rejuvenate your technology quickly and

                  affordably.

                </span>

              </div>

            </div>

            <div class="mb-5 flex font-medium">

              <div class="mr-4">

                <svg

                  xmlns="http://www.w3.org/2000/svg"

                  fill="none"

                  viewBox="0 0 24 24"

                  stroke-width="1.5"

                  stroke="currentColor"

                  class="h-7 w-7 text-blue-500"

                >

                  <path

                    stroke-linecap="round"

                    stroke-linejoin="round"

                    d="M16.023 9.348h4.992v-.001M2.985 19.644v-4.992m0 0h4.992m-4.993 0l3.181 3.183a8.25 8.25 0 0013.803-3.7M4.031 9.865a8.25 8.25 0 0113.803-3.7l3.181 3.182m0-4.991v4.99"

                  />

                </svg>

              </div>

              <div class="">

                <p class="mb-2">Why Choose FixIT Hub?</p>

                <span class="font-normal text-gray-600">

                  <p style={{ fontWeight: "300" }}>

                    <p style={{ fontSize: "30px" }}>

                      <li>Expertise</li>

                    </p>

                    Our technicians are highly skilled and certified in a wide

                    range of repairs, ensuring your devices are in capable

                    hands.

                  </p>

                  <p style={{ fontWeight: "300" }}>

                    <p style={{ fontSize: "30px" }}>

                      <li>Quality Service</li>

                    </p>{" "}

                    We believe in doing the job right the first time. With

                    meticulous attention to detail and thorough testing, we

                    ensure that your device leaves our shop in optimal

                    condition.

                  </p>

                  <p style={{ fontWeight: "300" }}>

                    <p style={{ fontSize: "30px" }}>

                      <li>Fast Turnaround</li>

                    </p>{" "}

                    We understand the importance of your devices in your daily

                    life. That's why we strive to provide prompt service without

                    compromising on quality.

                  </p>

                  <p style={{ fontWeight: "300" }}>

                    <p style={{ fontSize: "30px" }}>

                      <li>Affordability</li>

                    </p>{" "}

                    We offer competitive pricing and transparent quotes, so you

                    know exactly what to expect without any surprises.

                  </p>

                  <p style={{ fontWeight: "300" }}>

                    <p style={{ fontSize: "30px" }}>

                      <li>Customer Satisfaction</li>

                    </p>{" "}

                    Your satisfaction is our priority. From the moment you walk

                    through our doors to the completion of your repair, we're

                    dedicated to providing friendly, professional service that

                    exceeds your expectations.

                  </p>

                </span>

              </div>

            </div>

            <div class="mb-5 flex font-medium">

              <div class="mr-4">

                <svg

                  xmlns="http://www.w3.org/2000/svg"

                  fill="none"

                  viewBox="0 0 24 24"

                  stroke-width="1.5"

                  stroke="currentColor"

                  class="h-7 w-7 text-blue-500"

                >

                  <path

                    stroke-linecap="round"

                    stroke-linejoin="round"

                    d="M3.75 13.5l10.5-11.25L12 10.5h8.25L9.75 21.75 12 13.5H3.75z"

                  />

                </svg>

              </div>

              <div class="">

                <p class="mb-2">Our Services</p>

                <span class="font-normal text-gray-600">

                  <p>

                    At FixIT Hub, we specialize in a wide range of repair

                    services, including

                  </p>

                  <p>

                    <li>Laptop Repair</li>

                  </p>

                  <p>

                    <li>Desktop Repair</li>

                  </p>

                  <p>

                    <li>Smartphone Repair</li>

                  </p>

                  <p>

                    <li>Tablet Repair</li>

                  </p>

                  <p style={{ fontSize: "40px" }}>And much more!</p>

                </span>

              </div>

            </div>

          </div>

          <div class="border border-gray-100 shadow-gray-500/20 mt-8 mb-8 max-w-md bg-white shadow-sm sm:rounded-lg sm:shadow-lg lg:mt-0">

            <div class="relative border-b border-gray-300 p-4 py-8 sm:px-8">

              <h3 class="mb-1 inline-block text-3xl font-medium">

                <span class="mr-4">Submit Complaint</span>

                <span class="inline-block rounded-md bg-blue-100 px-2 py-1 text-sm text-blue-700 sm:inline">

                  Quick Response

                </span>

              </h3>

              <p class="text-gray-600">Contact us for faulty items repair</p>

            </div>

            <div class="p-4 sm:p-8">

              <input

                id="name"

                type="text"

                class="mt-1 w-full resize-y overflow-auto rounded-lg border border-gray-300 px-4 py-2 shadow-sm focus:border-blue-500 focus:outline-none hover:border-blue-500"

                placeholder="Enter  name"

                onChange={(e) => setName(e.target.value)}

              />

              <input

                id="fault"

                type="text"

                class="peer mt-8 w-full resize-y overflow-auto rounded-lg border border-gray-300 px-4 py-2 shadow-sm focus:border-blue-500 focus:outline-none hover:border-blue-500"

                placeholder="Enter Faulty item name"

                onChange={(e) => setFault(e.target.value)}

              />

              <input

                id="address"

                type="text"

                class="peer mt-8 w-full resize-y overflow-auto rounded-lg border border-gray-300 px-4 py-2 shadow-sm focus:border-blue-500 focus:outline-none hover:border-blue-500"

                placeholder="User address"

                onChange={(e) => setAddress(e.target.value)}

              />

              <input

                id="location"

                type="text"

                class="peer mt-8 w-full resize-y overflow-auto rounded-lg border border-gray-300 px-4 py-2 shadow-sm focus:border-blue-500 focus:outline-none hover:border-blue-500"

                placeholder="Item pickup location"

                onChange={(e) => setPickupLocation(e.target.value)}

              />

              <input

                id="repairLocation"

                type="text"

                list="repair-locations"

                class="peer mt-8 w-full resize-y overflow-auto rounded-lg border border-gray-300 px-4 py-2 shadow-sm focus:border-blue-500 focus:outline-none hover:border-blue-500"

                placeholder="Enter repair location"

                onChange={(e) => setRepairLocation(e.target.value)}

              />

              <datalist id="repair-locations">

                <option value="Mumbai Repair Center" />

                <option value="Delhi Service Hub" />

                <option value="Bangalore Tech Repair" />

                <option value="Chennai Electronics Fix" />

                <option value="Hyderabad Service Zone" />

                <option value="Kolkata Repair Hub" />

                <option value="Pune Gadget Care" />

                <option value="Ahmedabad Service Center" />

                <option value="Jaipur Repair Workshop" />

                <option value="Lucknow Device Fix" />

              </datalist>

              <input

                id="phoneNo"

                type="text"

                class="peer mt-8 w-full resize-y overflow-auto rounded-lg border border-gray-300 px-4 py-2 shadow-sm focus:border-blue-500 focus:outline-none hover:border-blue-500"

                placeholder="Enter phone no"

                onChange={(e) => setPhoneNo(e.target.value)}

              />

              <label class="mt-5 mb-2 inline-block max-w-full">

                Tell us about your Faulty iteam details

              </label>

              <textarea

                id="about"

                class="mb-8 w-full resize-y overflow-auto rounded-lg border border-gray-300 px-4 py-2 shadow-sm focus:border-blue-500 focus:outline-none hover:border-blue-500"

                onChange={(e) => setDescription(e.target.value)}

              ></textarea>

              <button

                class="w-full rounded-lg border border-blue-700 bg-blue-700 p-3 text-center font-medium text-white outline-none transition focus:ring hover:border-blue-700 hover:bg-blue-600 hover:text-white"

                onClick={onSubmitForm}

              >

                Submit

              </button>

            </div>

          </div>

        </div>

      </section>

    </>

  );

};

const Complaint = () => {

  const { headers } = useAuth();

  const [complaints, setComplaints] = useState([]);

  const getComplaints = async (e) => {

    try {

      const response = await fetch("http://localhost:3000/complaints", {

        method: "GET",

        headers: headers,

      });

      if (!response.ok) {

        throw new Error("Failed to fetch complaints");

      }

      const jsonData = await response.json();

      // Ensure data is an array before setting state

      if (Array.isArray(jsonData)) {

        setComplaints(jsonData);

      } else {

        console.error("Received invalid complaints data:", jsonData);

        setComplaints([]);

      }

    } catch (err) {

      console.error("Error fetching complaints:", err.message);

      setComplaints([]);

    }

  };

  useEffect(() => {

    getComplaints();

  }, []);

  console.log(complaints);

  return (

    <>

      <div className="bg-gray-100 p-4 sm:p-8 md:p-10 h-screen">

        <h1 className="text-2xl font-bold mt-20 mb-8">

          Faulty items Complaints

        </h1>

        {complaints.length === 0 ? (

          <p className="ml-4 mt-2 text-gray-600 text-xl">

            No complaints registered yet.

          </p>

        ) : (

          <div className="container mx-auto grid gap-8 md:grid-cols-3 sm:grid-cols-1">

            {complaints.map((complaint) => (

              <div

                key={complaint.complaint\_id}

                className="relative flex h-full flex-col rounded-md border border-gray-200 bg-white p-2.5 hover:border-gray-400 sm:rounded-lg sm:p-5"

              >

                <div className="text-lg mb-2 font-semibold text-gray-900 hover:text-black sm:mb-1.5 sm:text-2xl">

                  {complaint.name}

                </div>

                <p className="text-sm">

                  Created on {formatTimestamp1(complaint.created\_at)}

                </p>

                <p className="mb-4 text-sm">

                  {complaint.assigned\_at

                    ? `Completed on ${formatTimestamp(complaint.assigned\_at)}`

                    : null}

                </p>

                <div

                  className="text-md leading-normal text-gray-400 sm:block overflow-hidden"

                  style={{ maxHeight: "100px" }}

                >

                  {complaint.description}

                </div>

                <button

                  className={`group flex w-1/3 mt-3 cursor-pointer items-center justify-center rounded-md px-4 py-2 text-white transition text-sm ${

                    complaint.is\_completed ? "bg-green-500" : "bg-red-600"

                  }`}

                >

                  <span className="group flex w-full items-center justify-center rounded py-1 text-center font-bold">

                    {complaint.is\_completed ? "Completed" : "Not Completed"}

                  </span>

                </button>

              </div>

            ))}

          </div>

        )}

        <ComplaintForm />

      </div>

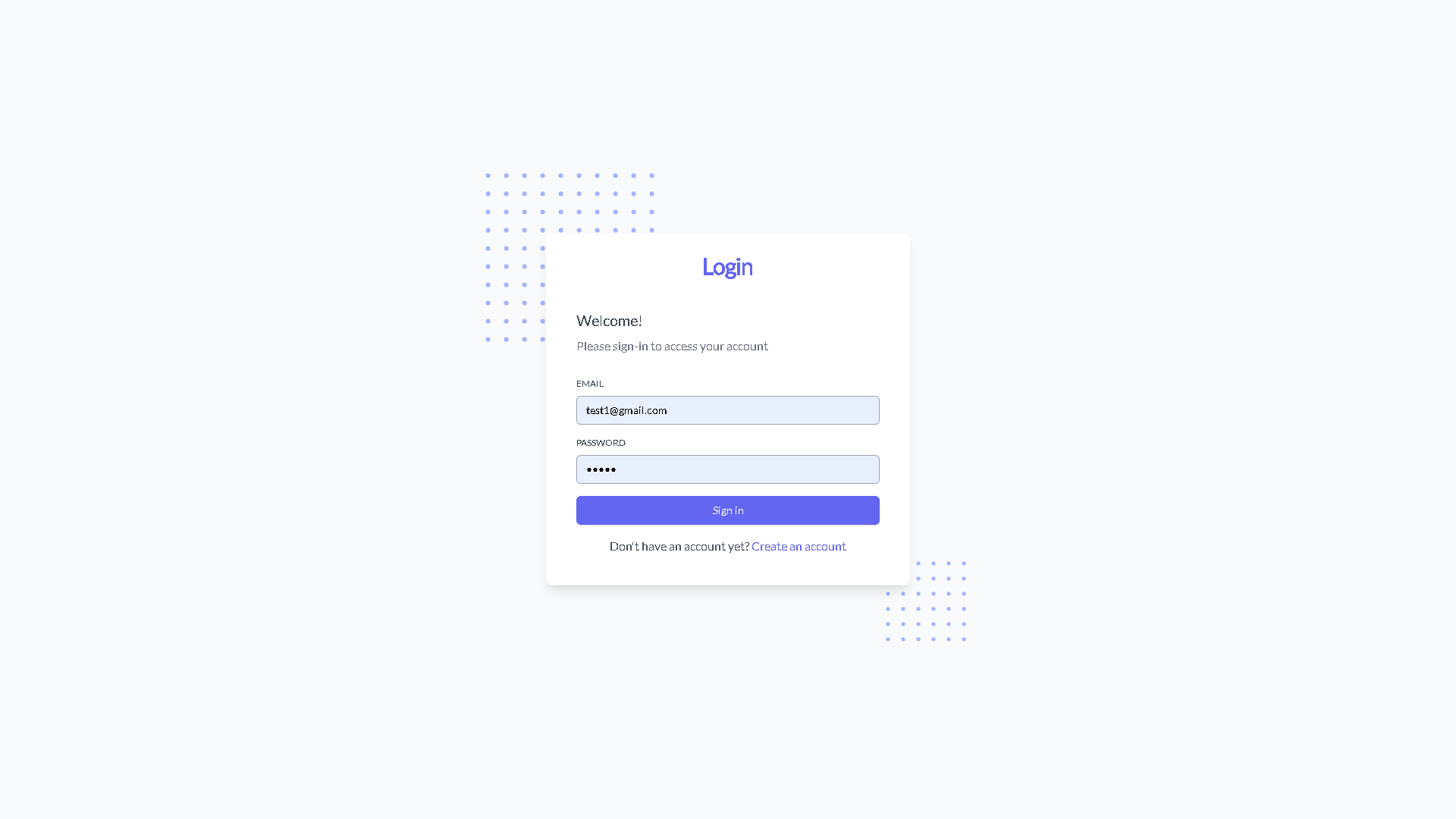
    </>

  );

};

export default Complaint;

## CHAPTER 5 :- Output

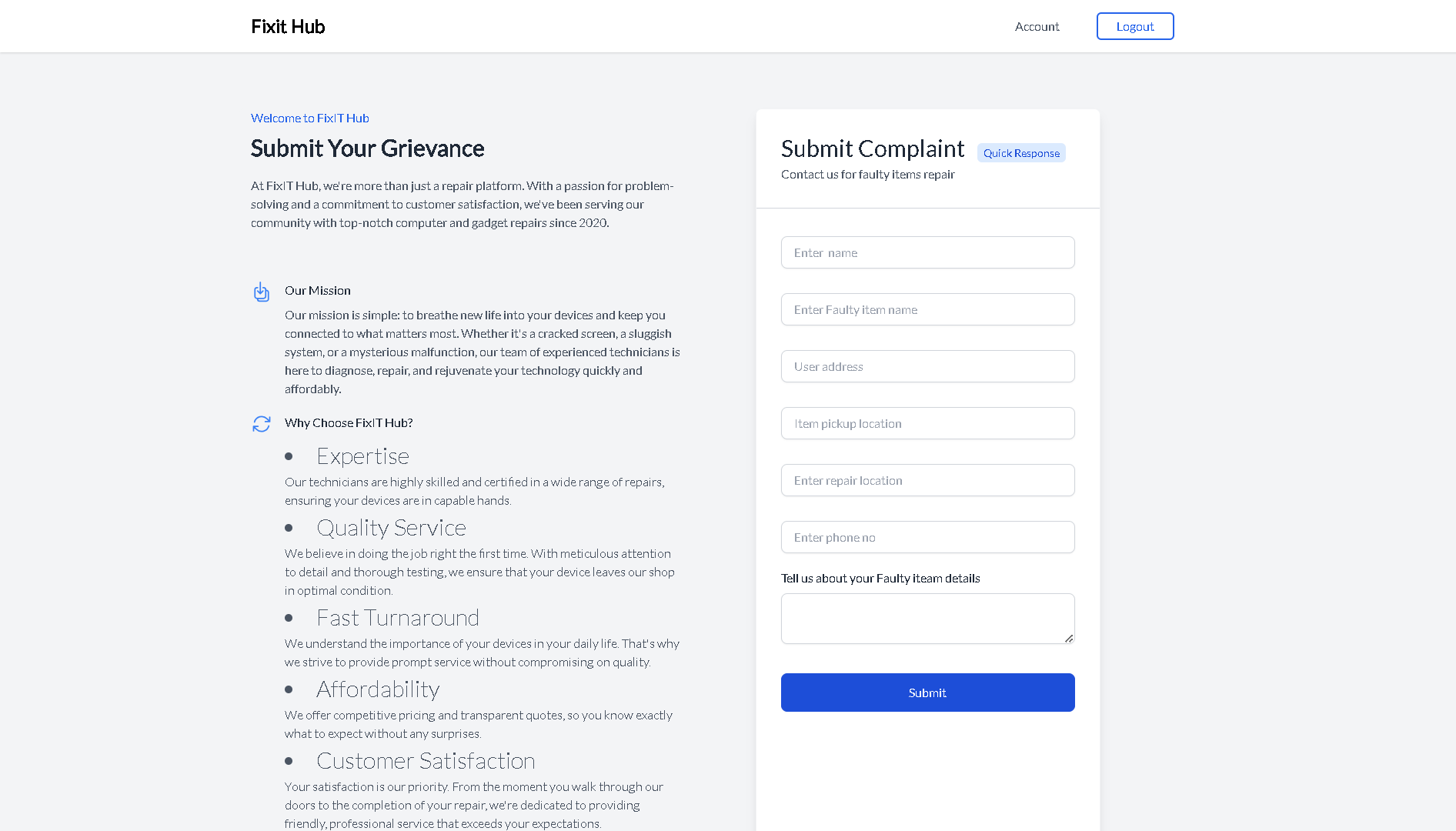
****

**Fig no 1: -login form**

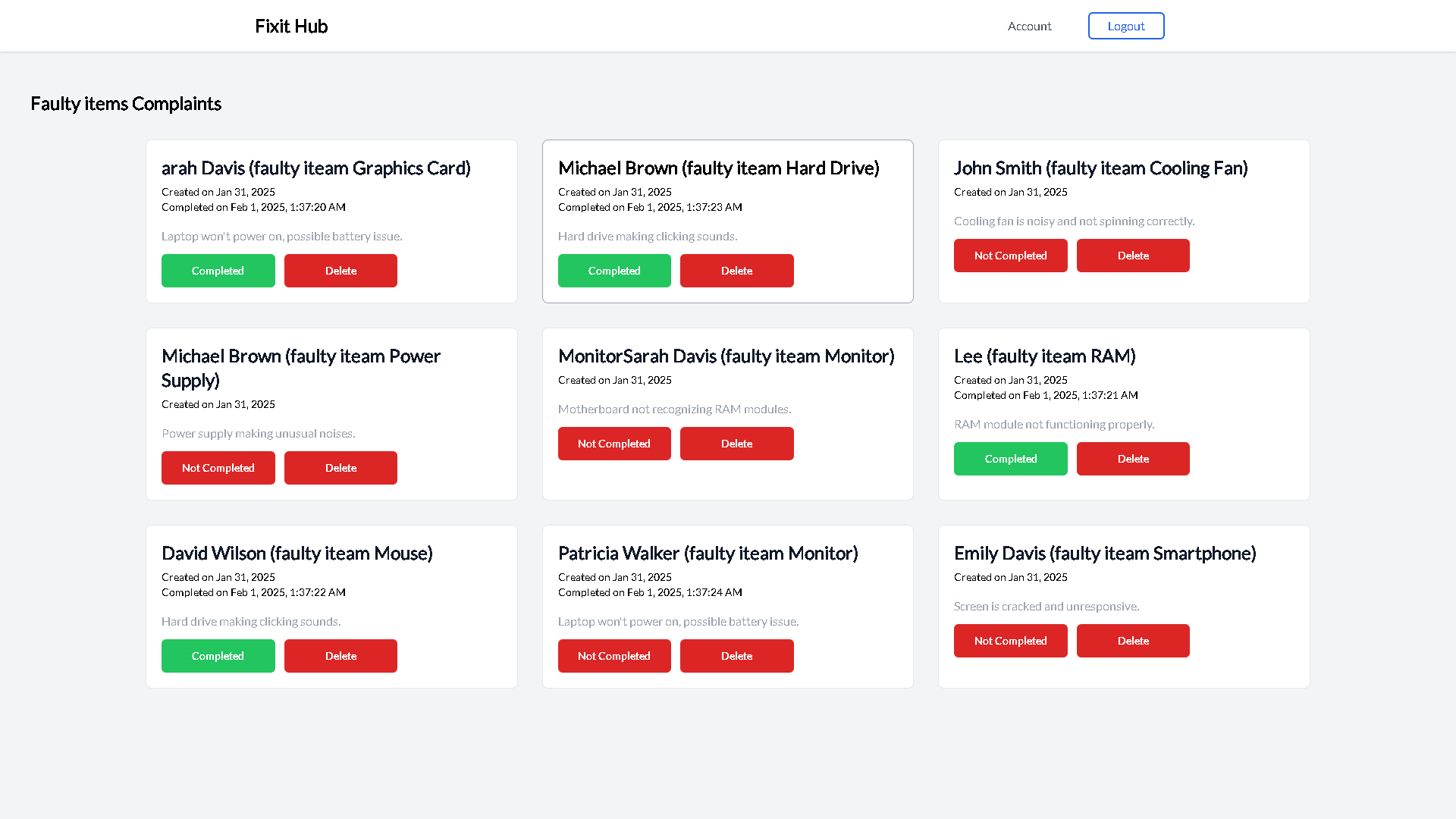
**A screenshot of a computer screen

Description automatically generated**

**Fig no 2: -signup form**

****

**Fig no 3: -Customer form\**

****

**Fig no 4: -Admin Form**