

---

**IT314 Software Engineering**  
**Lab 4 - Specifying Tools and Technology**  
**Project Name: AthleteTalk**  
**Group 18**

---

**Project members :**

Name	ID
PATEL KHUSHKUMAR GIRISHBHAI	202001127
ACHARYA HARSH MAHAVIRBHAI	202001131
UPADHYAY DHVANI VIPULKUMAR	202001136
PORIYA NEEL DHARMESHBHAI	202001139
PATEL PARTHKUMAR KANTIBHAI	202001147
VEGAD RAJ MAHESHBHAI	202001160
CHIRAG CHAVDA	202001164
SUVAGIYA ABHAY JAGDISHBHAI	202001171
VIRADIYA ABHAY	202001174

# 1. You must finalize/identify the tools, technologies, and frameworks you will use to develop/implement your project.

Framework for developing web-application:

Next.js is a flexible **full stack framework** that gives you building blocks to create fast **web applications**.

Next.js, on the inside, uses **React** as a front-end framework.

React.js:

ReactJS, is an open-source **JavaScript library** used for building user interfaces (UIs) and UI components. It was developed by **Facebook** and is now maintained by Facebook and a community of individual developers and companies.

## Reasons:

- **React components:** React allows code reusability by using components which can be used multiple times by referencing the same component.
- **Fast Rendering:** React uses a virtual DOM, which allows for faster rendering and better performance.
- **Large Community Support:** Being developed and maintained by Facebook, React has a strong backing and is constantly being improved and updated.
- **JSX:** JSX is a syntax extension for JavaScript that allows developers to write HTML-like code directly in their JavaScript files. This makes it easy to create complex UIs without having to switch between different languages or file types.
- **Hooks:** React introduced Hooks in newer versions, which allows developers to use state and other React features in functional components, making them easier to write and more reusable.

Next.js uses express.js inside its backend stack.

Express.js:

ExpressJS, or simply Express, is a popular open-source web application framework for Node.js. It provides a set of features and tools for building web and mobile applications, including robust routing, middleware support, and a flexible plugin architecture.

**Reasons:**

- **Minimalist and easy of use:** Express is a straightforward and adaptable framework that offers programmers a selection of fundamental features and resources. This enables developers to create unique solutions that are suited to their unique demands.
- **Robust routing:** Express provides a robust routing system that allows developers to handle different HTTP requests and responses easily.
- **Scalability:** Express is designed to be scalable and can handle a large number of requests simultaneously.
- **Compatible with other Node.js modules:** Express is compatible with other Node.js modules, making it easy to integrate with other tools and technologies.
- **Middleware functions:** Express provides middleware functions, which are functions that can access the request and response objects and can modify them or perform additional operations before passing the control to the next middleware function.
- **Error handling:** Express provides a robust error-handling system that allows developers to catch and handle errors that occur during the execution of middleware functions or routes.

2. For your project, you have to use the NoSQL databases of your choice strictly, and you can also explore and use ElasticSearch (DB) Database for the same.

Database: **Mongodb**

MongoDB is a popular open-source document-oriented NoSQL database system. Unlike traditional relational databases, MongoDB stores data in a flexible, JSON-like document format, making it easy to store and retrieve complex, hierarchical data structures.

**Reasons:**

- **Cloud support:** MongoDB offers cloud-based hosting services through its MongoDB Atlas platform, making it easy for developers to deploy and manage their MongoDB databases in the cloud.
- **High performance:** MongoDB provides high performance due to its document-oriented storage, which allows for fast reads and writes. It also supports indexing, sharding, and load balancing
- **Scalability and flexibility:** MongoDB is designed to scale horizontally, which means it can easily handle large amounts of data and traffic. It also provides flexible data modeling, allowing developers to change the schema of their data on-the-fly without having to modify the database.
- **Open-source:** MongoDB is open-source software, meaning that it is free to use, modify, and distribute, making it an accessible choice for many developers.

## Unadjusted Use Case Weight (UUCW):

Use Case Complexity	Number of Transactions	Use-Case Weight
Simple	≤3	5
Average	4 to 7	10
Complex	> 7	15

Use case name	Number of Transactions	Complexity
User Sign Up	2	Simple
User Login	1	Simple
Personalized content recommendation	4	Average
Nutrition tracking	3	Simple
User Profile	1	Simple
Community discussion	4	Average
Upload a video	5	Average
Health reminder	3	Simple
Exercise tracking and Health report	3	Simple
Share achievements	4	Average

Use Case Complexity	Use - Case weight	Number of Transactions	Product
Simple	5	6	30
Average	10	4	40
Complex	15	0	0
<b>Unadjusted Use - Case Weight (UUCW)</b>			<b>70</b>

## Unadjusted Actor Weight (UAW):

Actor Complexity	Example	Actor weight
Simple	External system that must interact with the system using a well-defined API.	1
Average	External system that must interact with the system using standard communication protocols (e.g. TCP/IP, FTP, HTTP, database)	2
Complex	Human actor using a GUI application interface	3

Actor Name	Category	Weight
Athlete	Complex	3
Coach	Complex	3
System	Simple	1
Database	Average	2
<b>Unadjusted Actor Weight (UAW)</b>		<b>9</b>

$$UUCP = UUCW + UAW$$

$$UUCP = 70 + 9$$

$$UUCP = 79$$

### Technical Complexity Factor (TCF):

Factor	Description	Weight (W)	Assigned Value	Weight x Assigned Value
T1	Distributed System	2	3	6
T2	Response Time/Performance Objectives	1	5	5
T3	End-user Efficiency	1	4	4
T4	Internal Processing Complexity	1	3	3
T5	Code Reusability	1	5	5
T6	Easy to Install	0.5	0	0
T7	Easy to Use	0.5	5	2.5
T8	Portability to other platforms	2	4	8
T9	System maintenance	1	3	3
T10	Concurrent/parallel processing	1	2	2
T11	Security features	1	4	4
T12	Access for third parties	1	1	1
T13	End user training	1	2	2
<b>Total (TF):</b>				<b>45.5</b>

$$\text{TCF} = 0.6 + (\text{TF}/100) = 0.6 + (45.5/100) = 0.6 + 0.455 = 1.055$$

## Environmental Complexity Factor (ECF):

Factor	Description	Weight (W)	Assigned Value	Weight x Assigned Value
E1	Familiarity with development process used	1.5	3	4.5
E2	Application experience	0.5	2	1
E3	Object-oriented experience of team	1.0	4	4
E4	Lead analyst capability	0.5	4	2
E5	Motivation of the team	1.0	5	5
E6	Stability of requirements	2.0	3	6
E7	Part-time staff	-1.0	0	0
E8	Difficult programming language	-1.0	3	-3
<b>Total (EF):</b>				<b>19.5</b>

$$ECF = 1.4 + (-0.03 \times EF) = 1.4 + (-0.03 \times 19.5)$$

$$ECF = 0.815$$



Factor	Description	Weight
UUCP	Unadjusted Use - Case Weight	79
TCF	Technical Complexity Factor	1.055
ECF	Environmental Complexity Factor	0.815

$$UCP = UUCP \times TCF \times ECF$$

$$UCP = 79 \times 1.055 \times 0.815$$

$$\mathbf{UCP = 67.926175}$$

$$\text{Total Working Hours} = UCP \times (\text{Hours}/UCP)$$

Here , Working Hours per UCP

$$\mathbf{\text{Hours}/UCP = 16}$$

$$\text{Total Working Hours} = 67.926175 \times 16$$

$$\mathbf{\text{Total Working Hours} = 1086.82 \text{ hours}}$$