Architectural Design Requirements

Functional Requirement	What	Why	How
Authentication	The system must verify the authenticity of the user	Only VIT support staff should be able to access the system	register / login system with encryption user database with password encryption
Confidentiality	The system must prevent third parties from accessing data sent between client and server	Prevent leakage of private information of involved staff/student members	Https and authentication
Data Persistence	The system must keep timetable data between uses	Timetable can be accesses/changed at any time	Database (MongoDB) save and autosaving functions
Data Processing	The system must be able to upload and derive information from excel spreadsheets	End users should be able to import information without having to manually add it	file upload and reading CSV into database
Timetable Algorithm	The system must be able to generate a functional timetable data it currently has	Core client need	develop timetabling algorithm based on client demands
Data Readability	The system must display timetable data in a human readable way	End users need to be able to read and change data	react webapp with spreadsheet display and timetable display
Data modification	The system must allow users to modify any and all data	End users need to be able to read and change data	Quick access to spreadsheets via timetable view, editing available throughout webapp

Non-functional Requirements

Scalability: In order to advocate for potential growing number of users the system should maintain its performance without degradation. To achieve this, some advanced techniques that could be employed if needed include sharding in MongoDB and the use of a load balancer to distribute the traffic across mulitple servers.

Maintainability: In order to the system with ease, updates and testing will be conducted regularly. Furthermore, using clean and well-documented code which follow the best practices (in our case PEP8) will ensure smooth maintainability. Using our automated testing with continuous integration/continuous deploymen pipeling (CICD) this advance feature will enhance maintainability to the next level.

Extendability: To meet the evolving needs of Rahul and the VIT staff team, the system needs to be adaptable. By using a modular design system allowing new features to be added without disruption of previous versions will ensure the underlyign infrastructure can be upgraded and expanded as needed.

Architechural Design