

Software Requirements Specification (SRS) Document

Water Management System in IIIT

Team 31

Anshul Padhi
Saransh Rajput
Jayati Narang
Fiza Husain

Brief problem statement

The broader objective is to analyse and improve the management of water in the IIIT campus. For this, we plan to build a web-app to show analysis of water consumed and meter readings for each bore well and manjeera. Based on the data we plan to detect leakages, find consumption patterns etc. and raise alarm in case a leakage is detected.

System requirements

- 1) MySQL
- 2) Python3
- 3) Django
- 4) Javascript
- 5) Onem2m(for TDS and pH of water-coolers)

Users profile

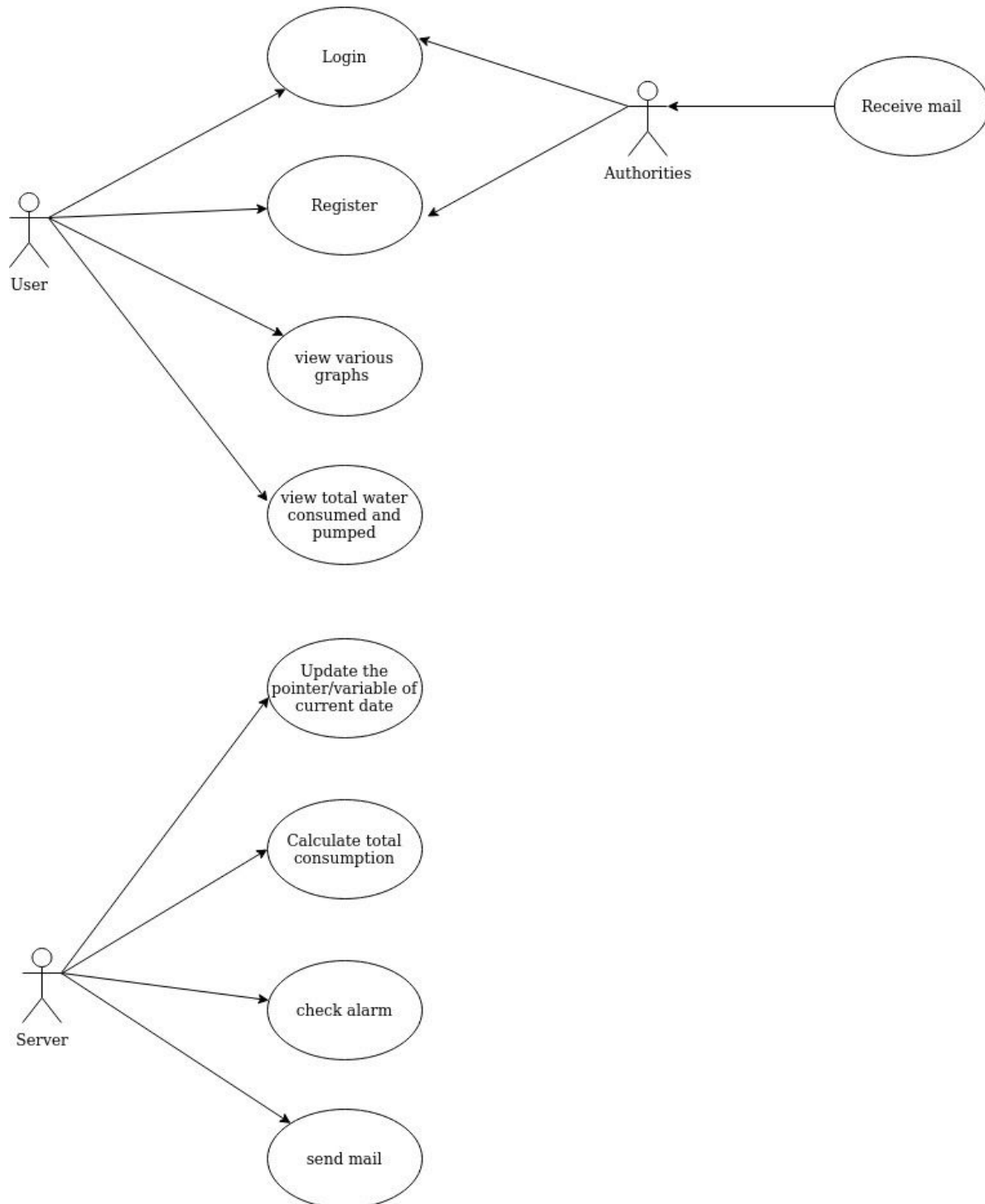
The web-app will mostly be used by IIIT engineering department and IoT lab of IIIT-H. No technical knowledge is required to use the dashboard. It will also be very helpful in detecting leakages in the IIIT water pipelines. Students and researchers can also use the app. Mails will be sent to the authorities through the app.

Feature requirements (described using use cases)

No.	User Case Name	Description	Release
1.	Leakage display on the dashboard	Leakage detected(amount) displayed on IIIT water map	R1
2.	Login Authorisation	Technique to only allow registered users from accessing the web-app	R2
3.	Analysis of water consumed	Compute consumption patterns based on the day, month, etc. and display the results in the form of graphs and charts	R1
4.	Alarm Management	Raise alarm on the detection of water leakages in the form of a notification	R2
5.	Leakage Detection	Device a method for detection of leakages in the water pipelines	R1
6.	IIIT water network map creation	Display the IIIT water network map with the water flow paths and leakage points	R1
7.	User Registration	Method to register new users. Should be part of IIIT	R2
8.	Database with water consumed and meter reading	Keep a database to store the data of the meter readings(of both borewell and manjeera) and water consumed.	R1

9.	Contact Info Support	A page to display the email ids of the users along with their name and number	R2
10.	Reports	Monthly reports mailed to the users	R2
11.	NRW Display	Percentage of Non-revenue water will be displayed on web app	R1
12	Ranking in water consumption and water pumped	Comparisons will be done between building/hostels/mess/ and between various borewells and manjeera water will be done and ranks will be displayed.	R2
13	Total consumption display bar	Bar with a total amount of water consumed last day and last month for each building.	R2
14	Anomaly detection	Any anomaly detected in the meter readings mailed to the concerned authorities.	R2

Use case diagram



Use case description

Use Case Number:	UC-1
Use Case Name:	Feeding Data
Overview:	An online form which will input the daily meter readings in the various pumps in IIIT. It will take the data and store it in a database.
Actors:	Special users
Pre condition:	User should be registered as a special user
Flow:	Main (success) Flow: 1.User logs in/registers 2.User clicks on 'Readings' 3.On the page he clicks on the 'Add' button 4.Enters the readings 5.Click the 'Confirm' button
	Alternate Flows: User is not special user: 1.User is redirected to 'Readings' User does not enter data: 1.Tell user to enter data
Post Condition:	1.Data should be stored in database 2.Success message should be displayed on adding readings

Use Case Number:	UC-2
Use Case Name:	Login Authorisation
Overview	An online form which takes the user's credentials and verifies them
Actors:	All users
Pre condition:	User should be registered
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1.User opens the dashboard 2.User enters his id and password 3.Credentials are verified 4.On verification user is allowed access
	<p>Alternate Flows:</p> <p>Credentials are incorrect:</p> <ol style="list-style-type: none"> 1.Tell user to re-enter credentials
Post Condition:	1.Main page of the dashboard is displayed

Use Case Number:	UC-3
Use Case Name:	Data Analysis

Overview:	Based on the daily meter readings, calculates the daily consumption and displays consumption patterns based on factors like the day, season, use, etc. in the form of different graphs
Actors:	All Users
Pre condition:	User should be registered
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1.User logs in/registers 2.User clicks on 'Consumption' 3.On the page to get the desired graph he clicks on the correct option(options will be self explanatory)
	<p>Alternate Flows:</p> <p>User is not authorised:</p> <ol style="list-style-type: none"> 1.User will not be redirected to main page
Post Condition:	1.Desired graph should be displayed on the page

Use Case Number:	UC-4
Use Case Name:	Alarm Management
Overview:	On the detection of a leakage an alarm should be raised in the form of a notification. Also the leakage point should be colored in the map displayed on the main page
Actors:	All Users

Pre condition:	User should be registered
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1.On the detection of a leak a notification is sent. 2.Notification will contain location
	<p>Alternate Flows:</p> <p>The user could choose to open the dashboard on notification:</p> <ol style="list-style-type: none"> 1.User logs in 2.The location of the leak will be colored on the IIIT water network map displayed on the main page
Post Condition:	<ol style="list-style-type: none"> 1.Notification is sent to the user (if logged in, displayed instantly), can also be sent by mail 2.Location of the leak will be colored red on the network map

Use Case Number:	UC-5
Use Case Name:	Leakage Detection
Overview:	Using load balancing the app will detect leakages in the water pipelines based on the data fed by the plumbers.
Actors:	
Pre condition:	Load balance is not satisfied beyond a certain margin
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1.Program checks all the readings for the day

	2.Uses the load balancing algorithm to check for leakages in all the pathways in the network 3.Raises alarm on detection of leakage
Post Condition:	1.Notification is sent to the user (if logged in, displayed instantly), can also be sent by mail 2.The location of the leak should be colored red on the map

Use Case Number:	UC-6
Use Case Name:	IIIT Water Network Map Creation
Overview:	A graphical map displayed on the main page of the dashboard which shows the water network layout and the network pipelines. Also show the locations of leakages(colored in red)
Actors:	Users
Pre condition:	User should be registered.
Flow:	Main (success) Flow: 1.User logs in/registers 2.Map is displayed on the main page itself
	Alternate Flows: User is not registered: 1.Main page won't load and error message will be displayed

Post Condition:	1.IIIT Water Network Map should be displayed on the main page
------------------------	---

Use Case Number:	UC-7
Use Case Name:	User Registration
Overview:	Online form which takes in user's details and on verification adds him to the list of registered users
Actors:	Everyone
Pre condition:	User must be on the registration/login page
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1.User clicks on register 2.User fills relevant details
	<p>Alternate Flows:</p> <p>Right to edit data:</p> <ol style="list-style-type: none"> 1.If user is a plumber or someone in authority who handles the data he has the right to add/delete data. He can select the option to get access to edit data during registration. He will be registered as a special user <p>Prospective user does not work at IIIT:</p> <ol style="list-style-type: none"> 1.Error message will be displayed and the details will not be added to the list of registered users
Post Condition:	<ol style="list-style-type: none"> 1.User should be added to the list of registered users 2.Success message should be displayed on successful registration

Use Case Number:	UC-8
Use Case Name:	Database Management
Overview:	A database should be implemented which will store the meter readings. Special users should be able to add/delete data to/from the database
Actors:	-
Pre condition:	-
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1.Special User logs in 2.User adds/deletes reading 3.Reading is passed to a database 4.Change is stored in the database
	<p>Alternate Flows:</p> <p>User is not authorised:</p> <ol style="list-style-type: none"> 1.Error message will be displayed 2.App does not display main page
Post Condition:	1.Change should be stored in the database

Use Case Number:	UC-9
-------------------------	------

Use Case Name:	Contact Information
Overview:	A page which displays the email ids of users and special users along with their names and numbers. Useful for communication within the department
Actors:	All Users
Pre condition:	User should be registered.
Flow:	Main (success) Flow: 1.User logs in/registers 2.User clicks on 'Contacts' button
	Alternate Flows: User is not registered: 1.Error message will be displayed 2.App will not display main page
Post Condition:	1.All the users' contact info should be displayed on the 'Contacts' page

Use case number:	UC-10
Use case name:	Reports
Overview:	At the end of a stipulated time period (monthly,weekly), reports are sent to the users with the consumption data, leakages and other analysis
Actors:	Users
Pre condition:	The person should be a registered user to receive a report
Flow:	1. Generic Report is generated 2. The report is personalised as per the user

	3. The Report is mailed to the user
Post condition:	Mail is received by the user which contains the report

Use case number:	UC-11
Use case name:	NRW Data Display
Overview:	NRW refers to Non Revenue Water. NRW percentage calculations and it's monetary effect in terms of graphs and charts must be displayed.
Actors:	Users
Pre condition:	The person must be a registered user to view this data and must be logged in
Flow:	<p>Done after a stipulated amount of time:</p> <ol style="list-style-type: none"> 1. NRW calculation from the consumption data 2. Generation of graphs and charts from the data <p>Once done:</p> <ol style="list-style-type: none"> 1. Logged in User clicks on Display NRW data
	<p>Alternate Flow:</p> <ol style="list-style-type: none"> 1. If a user does not have authority to view this data, an error message is displayed.
Post condition:	The graph/chart is loaded.

Use case number:	UC-12
Use case name:	Ranking as per water consumption and water pumped(for the sources)

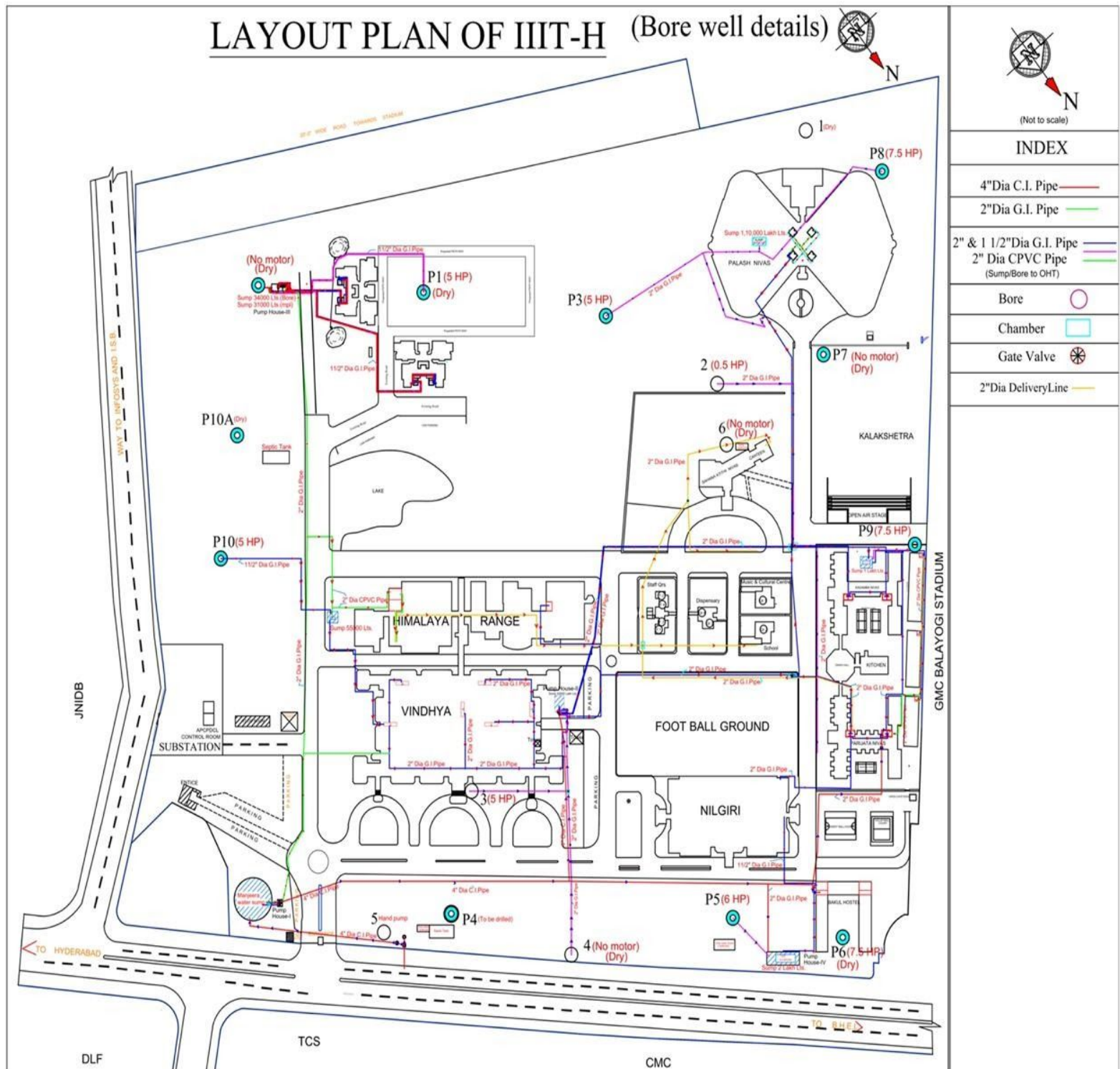
Overview:	Assigning a sort of efficiency score to each building/block as per consumption data while taking into account the population of the building/block and using this score to create a ranking amongst the buildings/blocks. The water sources will be ranked according to the amount of water pumped from them. (This will also represent how much water is pumped from the manjeera vs borewell?)
Actors:	Users
Pre condition:	The person must be a registered user to view this data and must be logged in
Flow:	Done after a stipulated amount of time: <ol style="list-style-type: none"> 1. Assigning a score using the consumption data population of the building 2. Use this score to create ranking/hierarchy in terms of efficiency of water usage 3. The rankings can be displayed in the form of graphs and charts Once done: <ol style="list-style-type: none"> 1. Logged in User clicks on Display block-wise Rankings
	Alternate Flow: <ol style="list-style-type: none"> 1. If user does not have authority to view this data, an error message is displayed.
Post condition:	The ranking data is displayed

Use case number:	UC-13
Use case name:	Total consumption Data
Overview:	Total water consumption data is displayed (per day/week/month) per block and each block's share in the overall consumption
Actors:	Users
Pre condition:	The person must be a registered user to view this data and must be logged in
Flow:	Done after a stipulated amount of time: <ol style="list-style-type: none"> 1. Total consumption per block is calculated per week/day/month 2. Total consumption per block and total overall consumption is used to

	<p>find share of blocks in water consumption</p> <p>3. These shares can be represented pictorially using graphs/charts</p> <p>Once done:</p> <p>1. Logged in User clicks on Display Block-wise Consumption</p>
Post condition:	The consumption data per Block is displayed

Use case number:	UC-14
Use case name:	Anomaly Detection
Overview:	The meter readings are collected manually and it's quite possible that the person doesn't actually go and collect the readings but rather just fills it in as he wishes. The objective here is to detect such cases.
Actors:	Only specific users (Engineer)
Pre condition:	The user be of a specific kind must have authority to view this data and must be logged in
Flow:	<p>1. Using Huge amounts of consumption data from around the year expected readings are generated.</p> <p>2. Using these expected readings, the readings provided are graded as correct or anomalous.</p> <p>3. Once done, the user(engineer) can view this data</p>
	<p>Alternate Flow:</p> <p>1. If the data available is insufficient, the above calculations cannot be done and an error is displayed.</p>
Post condition:	The data is displayed

IIIT-H Water Distribution Map



Current Water Distribution:

