SOFTWARE REQUIREMENTS SPECIFICATION

For

Weather App Using DB

Prepared by:-

Team 19 :-

Harii Sankar S

Ajaykumar

Gokul G

1. Introduction

1.1 Purpose

The main objective of this document is to illustrate the requirements of the project Weather App using DB. The document gives the detailed description of the both functional and non-functional requirements proposed by the client. The Weather App using DB document for a project that involves retrieving historical weather information from a database based on user input in a mobile app. This project describes the hardware and software interface requirements using ER diagrams and UML diagrams. The primary goal of this document is to delineate the requirements for the development of a Historical Weather Information Retrieval System, designed for integration into a mobile application. This project aims to offer users a seamless experience in accessing detailed historical weather data based on their specified location and date preferences. By leveraging computer technology, the system facilitates easy retrieval and presentation of weather information, enhancing user understanding of past weather conditions.

1.2 Document Conventions

- Entire document should be justified.
- Convention for Main title
  Font Face: Times New Roman
  Font Style: Bold
  Font Size: 14
  Sub-Title Convention:
  Font Face: Times New Roman
  Font Style: Bold
  Font Size: 12
  Body Text Convention:
  Font Face: Times New Roman
  Font Size: 12

1.3 Scope of Development Project

The Historical Weather Information Retrieval System is an app-based initiative, transforming the conventional approach of obtaining historical weather data into an intuitive and user-friendly interface. The application targets users seeking comprehensive insights into past weather conditions based on specific location and date parameters.

The product will work as a complete user interface for the Weather app and app usage from ordinary users.It is especially useful for any common people where modifications in the content can be done easily according to requirements.

The project can be easily implemented under various situations. We can add new features as and when we require, making reusability possible as there is flexibility in all the modules.
The language used for developing the project is Java as it is quite advantageous than other languages in terms of performance, tools available, cross platform compatibility, libraries, cost (freely available), and development process.

1.4 Definitions, Acronyms and Abbreviations

JAVA -> platform independence
SQL-> Structured query Language
ER-> Entity Relationship
UML -> Unified Modeling Language
IDE-> Integrated Development Environment
SRS-> Software Requirement Specification

1.5 References

- Books
  - "The AMS Weather Book: The Ultimate Guide to America's Weather" by Jack Williams
  - "Weather: A Guide to Phenomena and Forecasts" by Paul E. Lehr and R. Will Burnett
  - Software Engineering: A Practitioner’s Approach Fifth Edition By Roger S. Pressman
- Websites
  - **http://www.slideshare.net/**
  - **https://www.ncei.noaa.gov/**

2. Overall Descriptions

2.1 Product Perspective

Use Case Diagram of Weather App using DB

*

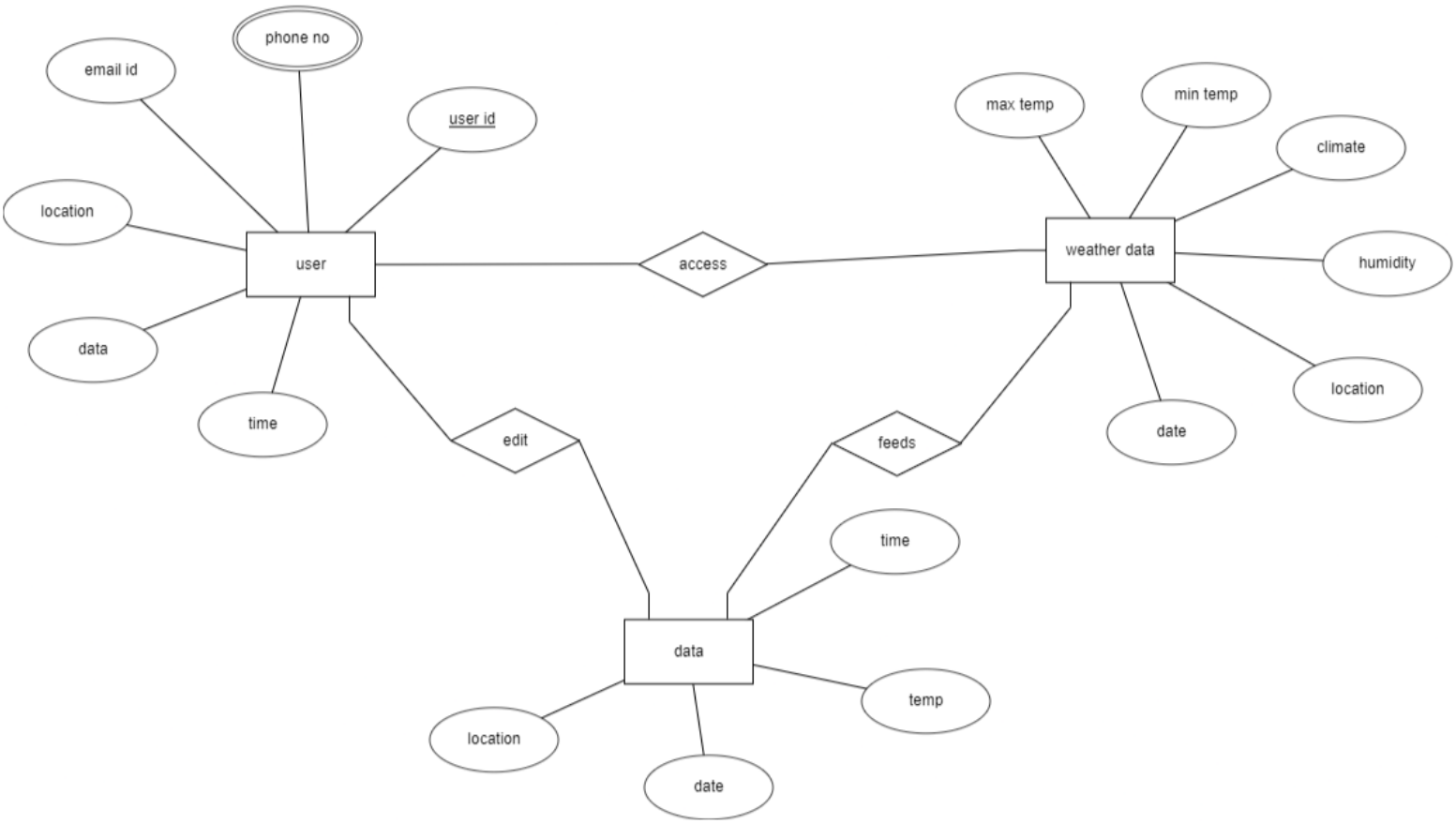
This is a broad level diagram of the project showing a basic overview. The users can be either

weather enthusiasts or general users intrested in historical weather data. This System will provide a search functionality to facilitate the past weather conditions based on date and location. This search will be based on location and date . Further

the user can add/update the resources from the system.

2.2 Product Function

Entity Relationship Diagram of Library Management System



The Online Library System provides online real time information about the books available in the Library and the user information. The main purpose of this project is to reduce the manual work. This software is capable of managing Book Issues, Returns, Calculating/Managing Fine, Generating various Reports for Record-Keeping according to end user requirements. The Librarian will act as the administrator to control members and manage books. The member's status of issue/return is maintained in the library database. The member's details can be fetched by the librarian from the database as and when required. The valid members are also allowed to view their account information.

2.3 User Classes and Characteristics

User Roles in Weather App:

The Weather App provides different services based on user types: Explorer and Curator

Features for Curators

Issue historical weather data to Explorers.
View different categories of historical weather information.
View the list of historical weather data available in each category.
Manage returned historical weather information.
Add new weather data and update existing records in the database.
Edit information of existing historical weather data.
Generate reports for available historical weather data.
Generate reports for issued historical weather data.
Access all accounts of Explorers.
Features for Explorers :

View different categories of historical weather information.
View the list of historical weather data available in each category.
Create and own an account in the weather app.
View the historical weather data issued to them.
Request historical weather data for a specific location and date.
View the history of historical weather data issued to them previously.
Search for specific historical weather data

2.4 Operating Environment

The product will be operating in windows environment. The Weather App shall operate in all famous browsers, for a model we are taking Microsoft Internet Explorer,Google Chrome,and Mozilla Firefox.Also it will be compatible with the IE 6.0. Most of the features will be compatible with the Mozilla Firefox & Opera 7.0 or higher version. The only requirement to use this online product would be the internet connection.
The hardware configuration include Hard Disk: 40 GB, Monitor: 15" Color monitor, Keyboard: 122 keys. The basic input devices required are keyboard, mouse and output devices are monitor, printer etc.

2.5 Assumptions and Dependencies

The assumptions are:-

2.5.1 The coding should be error free
2.5.2 The system should be user-friendly so that it is easy to use for the users
2.5.3 The information of all users, books and libraries must be stored in a database that isaccessible by the website
2.5.4 The system should have more storage capacity and provide fast access to the database
2.5.5 The system should provide search facility and support quick transactions
2.5.6 The Library System is running 24 hours a day
2.5.7 Users may access from any computer that has Internet browsing capabilities and an

Internet connection

2.5.8 Users must have their correct usernames and passwords to enter into their online accountsand do actions

The dependencies are:-

2.5.9 The specific hardware and software due to which the product will be run

2.5.10 On the basis of listing requirements and specification the project will be developed andrun

2.5.11 The end users (admin) should have proper understanding of the product

2.5.12 The system should have the general report stored

2.5.13 The information of all the users must be stored in a database that is accessible by theLibrary System

2.5.14 Any update regarding the book from the library is to be recorded to the database and thedata entered should be correct

2.6 Requirement

Software Configuration:-

This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL Server as the back end to store the database.

Operating System: Windows NT, windows 98, Windows XP

Language: Java Runtime Environment, Net beans 7.0.1 (front end)

Database: MS SQL Server (back end)

Hardware Configuration:-

Processor: Pentium(R)Dual-core CPU

Hard Disk: 40GB

RAM: 256 MB or more

2.7 Data Requirement

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be the queries as fired by the users like create an account, selecting cities . Now the output will be visible when the user requests the server to get details of their account in the form of time, date.

3 External Interface Requirement

3.4 GUI

The software provides good graphical interface for the user and the administrator can operate on the system, performing the required task such as create, update, viewing the details of the book.

3.4.1 It allows user to view quick reports like Temp in between particular date.

3.4.2 It provides historical data and search facility based on different criteria.

3.4.3 The user interface must be customizable by the administrator

3.4.4 All the modules provided with the software must fit into this graphical user interface andaccomplish to the standard defined

3.4.5 The design should be simple and all the different interfaces should follow a standard

template

3.4.6 The user interface should be able to interact with the user management module and a partof the interface must be dedicated to the login/logout module

Login Interface:-
In case the user is not yet registered, he can enter the details and register to create his account. Once his account is created he can 'Login' which asks the user to type his username and password. If the user entered either his username or password incorrectly then an error message appears.

Search:-
The user can enter the name of the city for which he is looking for and the date range he is interestedin to retrieve the data from the database , then he can search for the weather data by entering the above mentioned details.

4 System Features

The users of the system should be provided the surety that their account is secure. This is possible by providing:-

- User authentication and validation of members using their unique member ID
- Proper monitoring by the administrator which includes updating account status, showing a popup if the member attempts to issue weather data requests of cities not provided by the app .
- Proper accountability which includes not allowing a member to see other member's account. Only administrator will see and manage all member accounts

5 Other Non-functional Requirements

5.4 Performance Requirement

The proposed system that we are going to develop will be used as the Chief performance system within the different campuses of the university which interacts with the university staff and students. Therefore, it is expected that the database would perform functionally all the requirements that are specified by the university.

5.4.1 The performance of the system should be fast and accurate

5.4.2 The Weather App shall handle expected and non-expected errors in ways that prevent loss in information and long downtime period. Thus it should have inbuilt error testing to identify invalid username/password

5.4.3 The system should be able to handle large amount of data. Thus it should accommodate high number of historial for a reasonable period of time and users without any fault

5.5 Safety Requirement

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup so that the database is not lost. Proper UPS/inverter facility should be there in case of power supply failure.

5.6 Security Requirement

5.6.1 System will use secured database

5.6.2 Normal users can just read information but they cannot edit or modify anything except their personal and some other information.

5.6.3 System will have different types of users and every user has access constraints

5.6.4 Proper user authentication should be provided

5.6.5 No one should be able to hack users' password

5.6.6 There should be separate accounts for admin and members such that no member canaccess the database and only admin has the rights to update the database.

5.7 Requirement attributes

5.7.1 There may be multiple admins creating the project, all of them will have the right to create changes to the system. But the members or other users cannot do changes

5.7.2 The project should be open source

5.7.3 The Quality of the database is maintained in such a way so that it can be very userfriendly to all the users of the database

5.7.4 The user be able to easily download and install the system

5.8 Business Rules

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, make a decision, or infer new data from existing data.This includes the rules and regulations that the System users should abide by. This includes the cost of the project and the discount offers provided. The users should avoid illegal rules and protocols. Neither admin nor member should cross the rules and regulations.

5.9 User Requirement

The users of the system are members and Librarian of the university who act as administrator to maintain the system. The members are assumed to have basic knowledge of the computers and internet browsing. The administrators of the system should have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, user manual, online help and the guide to install and maintain the system must be sufficient to educate the users on how to use the system without any problems.

The admin provides certain facilities to the users in the form of:-

5.9.1 Backup and Recovery

5.9.2 Forgot Password

5.9.3 Data migration i.e. whenever user registers for the first time then the data is stored in theserver

5.9.4 Data replication i.e. if the data is lost in one branch, it is still stored with the server

5.9.5 Auto Recovery i.e. frequently auto saving the information

5.9.6 Maintaining files i.e. File Organization

5.9.7 The server must be maintained regularly and it has to be updated from time to time

6 Other Requirements

6.4 Data and Category Requirement

There are different categories of users namely teaching staff, Librarian, Admin, students etc. Depending upon the category of user the access rights are decided.It means if the user is an administrator then he can be able to modify the data,delete, append etc. All other users except the Librarian only have the rights to retrieve the information about database. Similarly there will be different categories of books available. According to the categories of books their relevant data should be displayed. The categories and the data related to each category should be coded in the particular format.

6.5 Appendix

A: Admin, Abbreviation, Acronym, Assumptions; B: Business rules; C: Class, Client, Conventions; D: Data requirement, Dependencies; G: GUI; K: Key; M: Member; N: Non-functional Requirement; O: Operating environment; P: Performance,Perspective,Purpose; R: Requirement, Requirement attributes; S: Safety, Scope, Security, System features; U: User, User class and characteristics, User requirement;

6.6 Glossary

The following are the list of conventions and acronyms used in this document and the project as well:

6.6.1 Administrator: A login id representing a user with user administration privileges to thesoftware

6.6.2 User: A general login id assigned to most users

6.6.3 Client: Intended users for the software

6.6.4 SQL: Structured Query Language; used to retrieve information from a database

6.6.5 SQL Server: A server used to store data in an organized format

6.6.6 Layer: Represents a section of the project

6.6.7 User Interface Layer: The section of the assignment referring to what the user interactswith directly

6.6.8 Application Logic Layer: The section of the assignment referring to the Web Server. Thisis where all computations are completed

6.6.9 Data Storage Layer: The section of the assignment referring to where all data is recorded

6.6.10 Use Case: A broad level diagram of the project showing a basic overview

6.6.11 Class diagram: It is a type of static structure diagram that describes the structure of a system by showing the system's cases, their attributes, and the relationships between the classes

6.6.12 Interface: Something used to communicate across different mediums

6.6.13 Unique Key: Used to differentiate entries in a database

6.7 Class Diagram

A class is an abstract, user-defined description of a type of data. It identifies the attributes of the data and the operations that can be performed on instances (i.e. objects) of the data. A class of data has a name, a set of attributes that describes its characteristics, and a set of operations that can be performed on the objects of that class. The classes' structure and their relationships to each other frozen in time represent the static model. In this project there are certain main classes

which are related to other classes required for their working. There are different kinds of relationships between the classes as shown in the diagram like normal association, aggregation, and generalization. The relationships are depicted using a role name and multiplicities. Here ‘User’, ‘Administrater’ are the most important classes which are related to other classes.