

**Domain Analysis,
Project Size Estimation, and Plan of Work**

Eric Kreider

Department of Computer Science and Informatics, Indiana University of South Bend

System Implementation

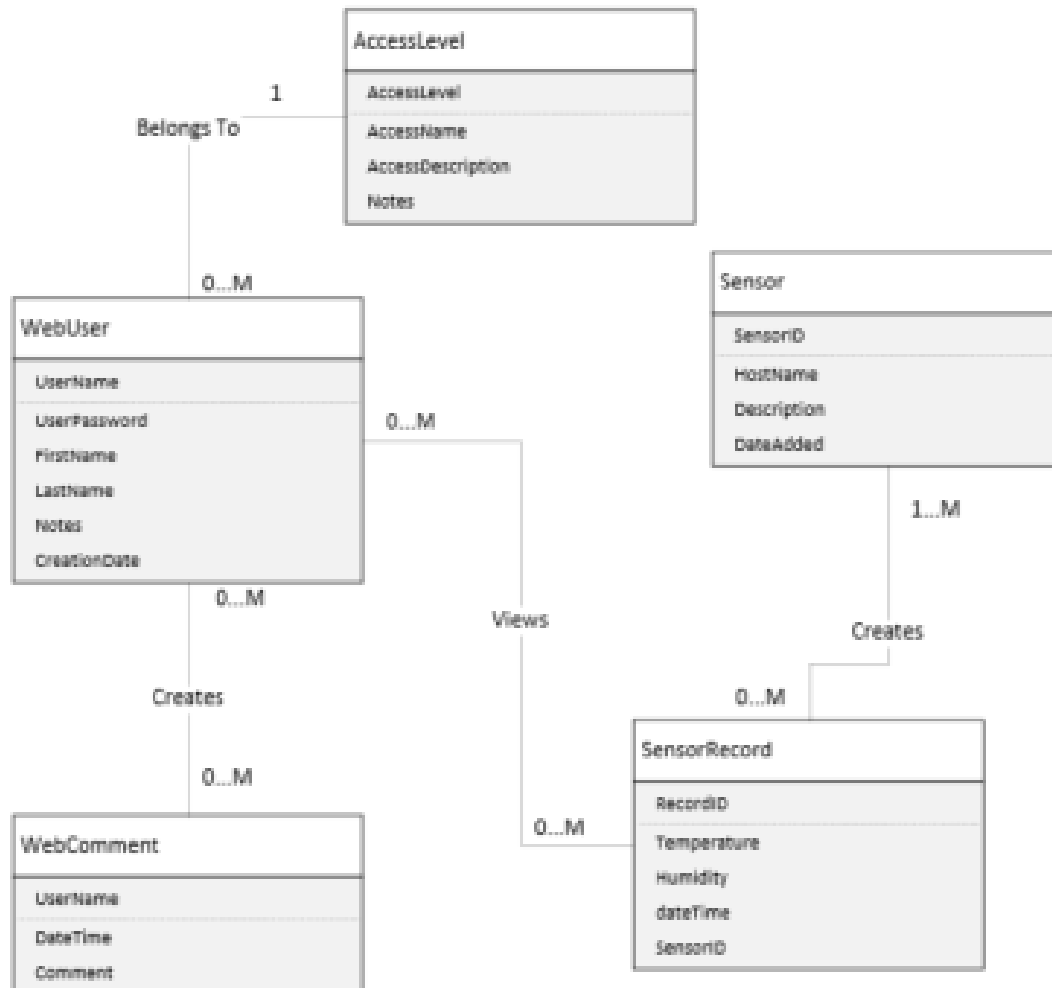
Professor Shawn Dai

February 25th, 2022

Domain Analysis

There are many useful ways of describing the relationships within a program or system. One useful method is the development of domain modeling and diagramming. A domain model works by showing both relationships and attributes of a system in order to adequately communicate how things are expected to interact. By developing this, a team can have an excellent planning backbone to build their infrastructure off of. Domain models are described through the use of class diagrams (Tilley & Rosenblatt, 2017). Each class should represent a single object as a square box with attributes. The attributes are the important bits of information associated with the class. A class then forms a relationship with other classes. These relationships have a cardinality, referring to how it interacts with other classes. Additionally, classes can be superclasses and subclasses of one another, meaning they can inherit the attributes of a parent class or pass them along to a child class. To begin to build a class diagram, a systems analyst needs to look at the use cases of a project and think about the interacting objects. It can be helpful to use UML notation to describe how the relationships between classes function (Tilley & Rosenblatt, 2017).

Class Diagram – Quality Sensor System



System Operation Contracts

Operation:	createUser(userName, userPassword, firstName, lastName, notes, creationDate, AccessLevel)
Cross References:	WebUser
Preconditions:	WebUser has access level to create new user
Postconditions:	A new WebUser created

Operation:	createRecord(RecordID, Temperature, Humidity, dateTime, SensorID)
Cross References:	SensorRecord, Sensor
Preconditions:	Sensor exists, is running
Postconditions:	A new SensorRecord is created

Operation:	viewAllData(UserName, AccessLevel)
Cross References:	WebUser, AccessLevel, Sensor, SensorRecord
Preconditions:	User has access to sensor data
Postconditions:	All sensors display data

Operation:	viewSingleSensor(UserName, AccessLevel, SensorID)
Cross References:	WebUser, AccessLevel, Sensor, SensorRecord
Preconditions:	User has access to sensor data
Postconditions:	Display records from single sensor

Operation:	createComment(UserName, dateTime, Comment)
Cross References:	WebUser, WebComment
Preconditions:	User has access to sensor data
Postconditions:	Display records from single sensor

Operation:	viewComment()
Cross References:	WebUser, WebComment
Preconditions:	User has authenticated, goes to appropriate page
Postconditions:	Displays all comments

Operation:	authenticate (UserName, UserPassword)
Cross References:	WebUser, AccessLevel
Preconditions:	User has account, entered credentials correctly
Postconditions:	Authenticates into the system

Project Size Estimation

The estimation of a project can be difficult. One good way of doing it based on use cases is by looking at the complexity of individual use cases. There are a number of different use cases involved within this system: View Single Sensor Data, View Sensor Dashboard, Export Data, View Trend Data, Log In, Change Your Password, Create New User, Reset Other Password, View Temperature Modules, and Create Comments. Of those use cases, one involves only one actor or class, five involve two classes, and three involve three or more classes. Through the usage of use case points, an analyst can come up with an estimation of difficulty based on actors, use case complexity, and external factors. While this is useful, this number does not have number of hours based on it. Using an estimation that a simple use case might be done in half an hour, a calculation can be done that gives an estimation. For this project, an estimation of total involvement ends up being between ninety project hours and three hundred and sixty project hours per an online calculation system (*Use Case Points Calculator* ,2022). While the upper range of this seems a bit longer than expected, the lower range feels like it might be more likely.

Plan of Work

Item	Description	Completion Date
Create Database Schema	Create database schema for storing necessary data backend	02/27/22
Create Database Server	Configure Pi for database backend. Install MariaDB. Implement schema. Create necessary user accounts for WebServer to connect. Create AccessLevels and other non-dynamically generated database values. Set static IP.	02/27/22
Create Web Server	Configure Pi for webserver. Install Apache webserver. Set static IP. Install backend language libraries. Test connection to MariaDB.	02/27/22
Create Sensors	Install sensor module component on another Pi. Install necessary Python libraries to read from sensor. Create script that writes to DB. Test.	03/06/22
Prototype Webpages	Prototype the necessary webpages to fulfill the needs. This will provide the front end scaffolding to support user needs.	03/06/22
Turn on Dataflow	Begin accumulating temperature data and troubleshoot any issues.	03/13/22
Build Webpages	Begin building webpages and get the core functionality working. Focus on the performance and functionality.	04/01/22
Implement Style	Focus on front-end development and making style and presentation attractive.	04/08/22
Usability Studies	Focus on usability to ensure project is meeting necessary needs and is intuitive.	04/15/22
Cleaning and stretch goals	Polish system and verify no bugs. Implement secondary stretch goals / add additional functionalities.	05/01/22

References:

Group, D. (n.d.). *Apache Documentation*. Documentation: Apache HTTP Server - The Apache HTTP Server Project. Retrieved February 26, 2022, from <https://httpd.apache.org/docs/>

How to use mariadb: Get started. MariaDB. (n.d.). Retrieved February 26, 2022, from <https://mariadb.com/get-started-with-mariadb/>

Tilley, S. R., & Rosenblatt, H. J. (2017). *Systems analysis and design*. Course Technology Cengage Learning.

Use case points calculator. Use Case Points Calculator. (n.d.). Retrieved February 26, 2022, from <http://ucp-calculator.appspot.com/>

W3Schools - free online web tutorials. W3Schools Online Web Tutorials. (n.d.). Retrieved February 26, 2022, from <https://www.w3schools.com/>

Note: the references listed above that are not cited within the paper are technology support resources I intend to use to assist me in building this project. They include details about the database and webserver infrastructure as well as coding guidelines useful for the web design portion of the process.