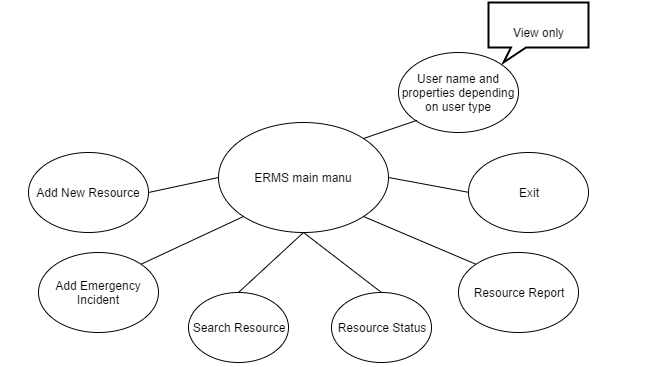
Phase 2 Abstract Code w/SQL

CS 6400 – Fall 2016

Team 028



# ERMS main menu

Abstract Code

* User enters *username* ($Username) and *password* ($Password);
* If data validation is successful for both *username* and *password*, then:
* When Login button is clicked:

SELECT \* FROM User

* + If User
* Display user’s name, population size, jurisdiction or headquarters depending on user type;

Display menu options;

// current user provide $username

// if user type is COMPANY

SELECT USER.name, COMPANY.headquater

From USER INNER JOIN COMPANY ON USER.username = COMPANY.username

WHERE username = $username;

// if user type is GOVERNMENT\_AGENCY

SELECT USER.name, GOVERNMENT\_AGENCY. jurisdiction

From USER INNER JOIN GOVERNMENT\_AGENCY

ON USER.username = GOVERNMENT\_AGENCY. username

WHERE username = $username;

// if user type is municipality

SELECT USER.name, MUNICIPALITY.Population\_size

From USER INNER JOIN MUNICIPALITY ON USER.username = MUNICIPALITY.username

WHERE username = $username;

# **Abstract Code - Add New Resource**

View auto assigned Resource ID, Owner Name;

While no buttons are pushed, do nothing;

While a button is pushed, then do the following:

{If Save:

Validate all fields

if all fields are valid:

{Update attributes of Resource (name, ESF, model, capabilities, $Home\_lat, $Home\_long, $cost\_unit, $price);

Go back to the ERMS main menu of user;}

// assume new information is $Res\_name, $ID\_desc, $Model, $Home\_lat, $Home\_long, $Price, $Cost\_unit

// add new resource

for each $Res\_name, $ID\_desc, $model, $Home\_lat, $Home\_long, $Price, Status, $Available\_date, $Rep\_start, $Rep\_by, $Cost\_unit

INSERT INTO RESOURCE (Username, Res\_ID, Res\_name, ID\_desc, Model, Home\_lat, Home\_long, Price, Status, Available\_date, Rep\_start, Rep\_by, Cost\_unit)

VALUES ($username, $Res\_ID, $Capabilities, $Res\_name, $ID\_desc, $Model, $Home\_lat, $Home\_long, $Price, ‘Available’, NULL, NULL, NULL, $Cost\_unit)

end for

for each $Capabilities

INSERT INTO RES\_CAP (Res\_ID, Res\_capability) VALUES ($Res\_ID, $Capabilities)

end for;

else:

{Show invalid fields and ask for new valid inputs;

Wait for another button push;}

If Cancel:

{Void the auto assigned ID;

Go to ERMS main menu of user;}}

// if click cancel

DELETE FROM RESOURCE

WHERE username = $username AND Res\_ID = $Res\_ID;

# 

# **Abstract Code - Add Emergency Incident**

Display auto assigned Incident ID;

While no buttons are pushed, do nothing;

While a button is pushed, then do the following:

{If Save:

Validate all fields

if all field are valid:

{Update attributes of Incident (Date, Description, Location);

Go back to the ERMS main menu of user;}

// assume the user’s username is $username

// assume new information is $Inc\_ID, $Date, $Description, $Longitude, $ Latitude

// add new incident

for each $Date, $Description, $Longitude, $Latitude

INSERT INTO Incident (Username, Inc\_ID, Date, Description, Longitude, Latitude)

VALUES ($Username, $Inc\_ID, $Date, $Description, $Longitude, $Latitude)

end for;

else:

{Show invalid fields and ask for new inputs;

Wait for another button push;}

If Cancel:

Go back to the ERMS main menu of user;}

DELETE FROM INCIDENT

WHERE username = $username AND Inc\_ID = $ Inc\_ID;