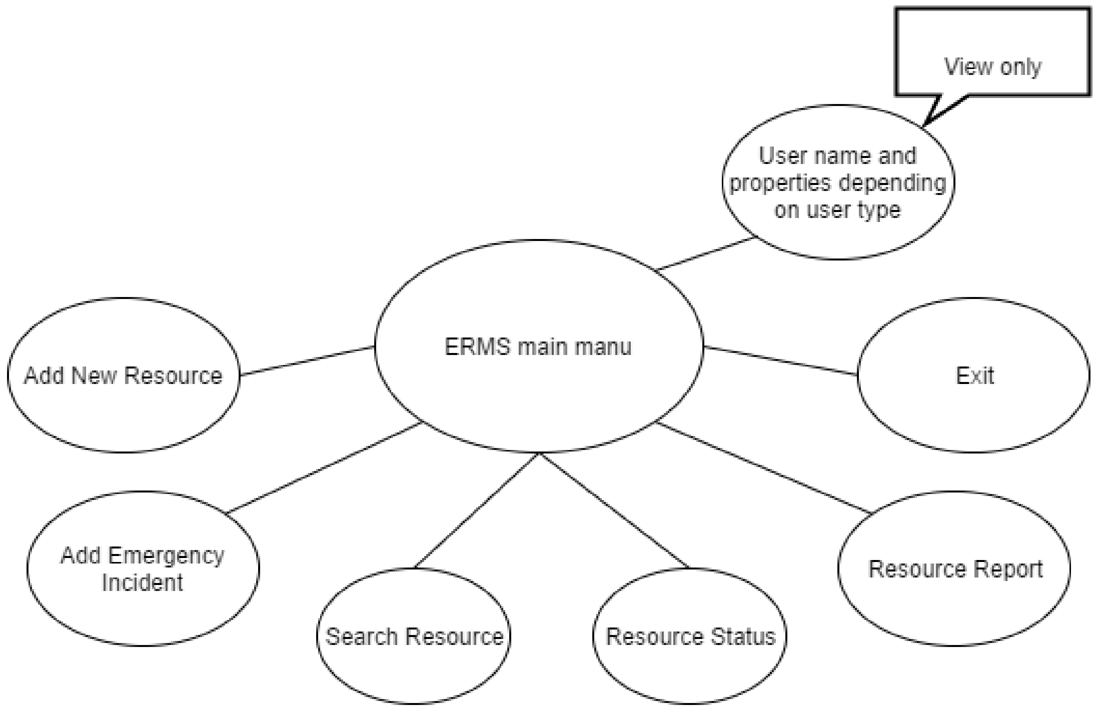


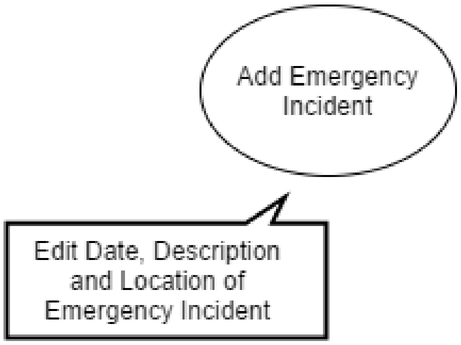
Abstract Code - ERMS main menu

```
// Find the current User using the username;
// Display user’s name, population size, jurisdiction or headquarters depending on user type;
// Display menu options;
    SELECT Individual.FirstName,  Individual.LastName, Municipality.populatoin_size,
    Gov.jurisdiction, Company.headquarters
    From   USER
    WHERE uername = $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 Location, cost);
                Go back to the ERMS main menu of user;}
                for each $Status
                UPDATE RESOURCES
                SET name = $name, ESF=$ESF, model = $model, capabilities = $capabilities,
                homeLocation = $homeLocation, cost=$cost
                WHERE username = $ username AND Res_ID = Res_ID
        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;}}
    for each $status
    DELETE FROM RESOURCE
    WHERE INCIDENT.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;}
              for each $status
              Update Incident
              SET Date = $Date, Description = $Description, Longitude = $Latitude,
                Inc_ID = Inc_ID
              WHERE username = $ username AND
       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;}
    for each $status
    DELETE FROM INCIDENT
    WHERE INCIDENT.Inc_ID = $ Inc_ID
```

Abstract Code - Search Resource

```
While no buttons are pushed, do nothing;
//populate dropdowns
        SELECT ID_description FROM ESF;
        SELECT Inc_ID, Description FROM INCIDENT;

If data validation is successful for all Keyword($Keyword), ESF($ID_description),
Location_distance($Distance) and Incident($Inc_ID, $Description):

        If Search button is pushed, then do the following:
                Display Search Result table according to search conditions;

                SELECT R.Res_ID, R.Res_name, R.Username, R.Price, R.Costunit, R.Status,
                R.Available_date, Distance
                FROM RESOURCES AS R INNER JOIN RES_CAP ON RESOURCES. Res_ID=
                RES_CAP. Res_ID
                WHERE (Res_name LIKE $Keyword OR  Model LIKE $Keyword OR Res_capability
                LIKE $Keyword) AND (ID_desc LIKE $ID_description) AND (Distance<=$Distance)
                ORDER BY Distance ASC;
                /* Distance is calculated by application through haversine formula */


        else if Cancel button is pushed: go to ERMS main menu of user;

else if data input fields are invalid, go to Search Resource form, with error message;
```

Search Result

```
Find the Incident according to search conditions;
Display the name($Description) and ID($Inc_ID) of incident;
Find the resources according to search conditions;
Display the resources($Res_ID) and their attributes, sorted by distance, then availability($Status) and
then name($Res_name);
While no buttons are pushed, do nothing;
While a button is pushed, then do the following:
        If Deploy button is pushed, then do the following: Update resource; Input expected return time;
        //update status
                for each $Status
                UPDATE RESOURCES
                SET Status='NOT AVAILABLE'
                WHERE Res_name=$Res_name AND Res_ID=$ Res_ID
                 end for


        else if Repair button is pushed: Update resource; Display Repair form;
```

input repair start date($Rep_start) and available date ($Available_date)

```
        for each $Status, $Rep_start, $Available_date
        UPDATE RESOURCES
        SET Status='IN REPAIR', Rep_start=$Rep_start, Available_date=$ Available_date
        WHERE Res_name=$Res_name AND Res_ID=$Res_ID
         end for
```

else if ***Request*** button is pushed: Update resource; update request; Display **<u>Rquest</u>** form;
input request start date($Start_date) and return date ($Return_by)

```
        for each $Start_date, $Return_by
        UPDATE REQUEST
        SET Start_date=$Start_date, Return_by=$Return_by
        WHERE Res_ID=$Res_ID
         end for
```

else if ***Close*** button is pushed: Go to **<u>ERMS main menu</u>** of user;