**Task I**

1. Requirements
   1. Complete the attached "Software Development Capstone Topic Approval Form." Include a signature and date from your course instructor to indicate that this form has been approved.
      1. Sample: <https://drive.google.com/file/d/1iWQOs9M05N1CU1dSk169B9g9VJstXNN_/view>
   2. Complete the attached "Capstone Waiver" after choosing the option that applies to your capstone project and digitally signing it where applicable.
2. Rubric
   1. The completed “Software Development Capstone Topic Approval Form” is complete, including the signature and date from the course instructor. The proposed solution is original.
   2. The completed “Capstone Waiver Form” is submitted and includes *all* required signatures.

**Task II**

1. Requirements
   1. Write a comprehensive summary that addresses the following requirements:
      1. Sample: <https://drive.google.com/file/d/1FbhNlMu0kO5HNBQLNW3rpF_Al4KHKAUx/view>
      2. the business problem or opportunity you are solving for, including a description of the customers and why this application will fulfill their needs
      3. existing gaps in the software application you are replacing or modifying (if applicable)
      4. the software development life cycle methodology you use to guide and support software development activities
      5. deliverables associated with the applied software development life cycle methodology
      6. the plan for implementation of your software solution, including the anticipated outcomes from this development
      7. the methods for validating and verifying that the developed software application meets the requirements and subsequently the needs of the customers
      8. the programming environments and any related costs, as well as the human resources that are necessary to execute each task in the development of the software application
      9. a projected timeline including milestones, start and end dates, duration for each milestone, dependencies, and resources assigned to each task
   2. Design and develop a fully functional software application that addresses your identified business problem or organizational need. Include each of the following attributes as they are the *minimum required elements* for the application:
      1. **one** of the following application types: mobile, web, or stand-alone application
      2. code including inheritance, polymorphism, and encapsulation
      3. search functionality with multiple row results and displays
      4. a database component with the functionality to securely add, modify, and delete the data
      5. ability to generate reports with multiple columns, multiple rows, date-time stamp, and title
      6. exception controls
      7. validation functionality
      8. industry appropriate security features
      9. design elements that make the application scalable
      10. a user-friendly, functional GUI
   3. Create each of the following forms of documentation for the application you have developed:
      1. Sample: <https://drive.google.com/file/d/16KmBK4qc1ekEpR16p0o6Ci_4I3JrBfkH/view>
      2. May combine the Design Document and Testing Document into a single PDF or Word file, but must keep separate from User Guides file.
      3. a design document including a class diagram and design diagram
         1. <https://app.creately.com/manage/recent>
         2. <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/>
      4. a test plan for a unit test, including screenshots
      5. the results of the unit test based on the provided test plan, including screenshots
      6. source code and executable file
      7. link to where web app is hosted with HTML code (if applicable)
      8. user guide for setting up and running the application for maintenance purposes
      9. user guide for running the application from a user perspective
2. Rubric
   1. The submission demonstrates the ability to apply core information technology skills and support organizational functions in each of the following areas: IT systems, operating systems, networking, security, scripting and programming, data management, project management, and web development.
   2. The submission demonstrates the ability to develop functioning, secure, and scalable object-oriented software applications that address an organizational need.
   3. The submission demonstrates the ability to architect data-centric, multitiered software solutions, including mobile and web applications.
   4. The submission demonstrates the ability to architect software solutions using software engineering best practices and process modeling.
   5. The submission demonstrates the ability to produce documentation and reports to support all phases of the software development cycle.

**Application**

1. Organizational structure
   1. Users
      1. To differentiate between the tracked objects of individuals
      2. Login with username and password
      3. User can have multiple profiles
      4. Features
         1. Username
         2. Password
         3. List of profiles
         4. Stats
            1. Stats for each subsequent level

# of Profiles

# of Buildings

# of Rooms

# of Units

# of Boxes

# of Items

Graph for changes over time?

* + - * 1. Creation datetime
      1. Edit button
  1. Profiles
     1. Profiles can be used for different purposes of a user
        1. Examples: work, home, D&D, storage rental, borrowed items, etc.
        2. Can also just have one Profile
     2. Top level organizational tier
     3. Features
        1. Name
        2. Description
        3. List of Buildings
        4. Stats
           1. Stats for each subsequent level

# of Buildings

# of Rooms

# of Units

# of Boxes

# of Items

Graph for changes over time?

* + - * 1. Creation datetime
        2. Last change made datetime
      1. Edit button
  1. Buildings
     1. Tracks the individual places/buildings/locations that a user may store items
        1. Examples: work office, work truck, home, shed, storage unit, etc.
        2. Can also just have one Building
     2. Second level organization tier
     3. Features
        1. Name
        2. Description
        3. List of Rooms
        4. Stats
           1. Stats for each subsequent level

# of Rooms

# of Units

# of Boxes

# of Items

Graph for changes over time?

* + - * 1. Creation datetime
        2. Last change made datetime
      1. Edit button
  1. Rooms
     1. Tracks the rooms/areas of a Building
        1. Examples: kitchen, cubicle, garage, backyard/outdoors, attic, etc.
        2. Can also just have one Room
     2. Third level organizational tier
     3. Features
        1. Name
        2. Description
        3. List of Units (optional)
           1. May list both Boxes and Units
           2. May list Boxes directly
        4. List of Boxes (optional)
           1. May list both Boxes and Units
        5. Stats
           1. Stats for each subsequent level

# of Units

# of Boxes

# of Items

Graph for changes over time?

* + - * 1. Creation datetime
        2. Last change made datetime
      1. Edit button
  1. Units (optional)
     1. Tracks the units of storage infrastructure of a Room
        1. Units can also be parts of the Room, such as floor, closet, etc.
        2. Examples: refrigerator, floor, shelf, car, lawn, desk, etc.
     2. Optional subdivision of a Room
     3. Features
        1. Name
        2. Description
        3. List of Boxes (optional)
           1. May list both Boxes and Items
           2. May list Items directly
        4. List of Items (optional)
           1. May list both Boxes and Items
        5. “Move to other [Room|Building|Profile]” button
        6. Stats
           1. Stats for each subsequent level

# of Boxes

# of Items

Graph for changes over time?

* + - * 1. Creation datetime
        2. Last change made datetime
      1. Edit button
  1. “Boxes” (optional)
     1. Tracks the Items in a Unit that can hold other Items and/or Boxes
        1. Boxes can hold other boxes
        2. Examples: box, specific shelf of a shelf unit, specific drawer of a nightstand, folder, wallet, etc.
     2. Optional subdivision of a Unit or other Box
     3. Features
        1. Name
        2. Description
        3. Tags
        4. List of Items
           1. May list both Boxes and Items
        5. List of Boxes (optional)
           1. May list both Boxes and Items
        6. “Move to other [Box|Unit|Room|Building|Profile]” button
        7. Stats
           1. Stats for each subsequent level

# of Boxes

# of Items

Graph for changes over time?

* + - * 1. Creation datetime
        2. Last change made datetime
      1. Edit button
  1. Items
     1. Tracks individual items
        1. Examples: literally anything!
     2. The finest organizational tier
     3. Features
        1. Name
        2. Description
        3. Tags
           1. Keyword/phrase tags
           2. Color tags
           3. Will need third table because of many-to-many relationship
        4. “Move to other [Box|Unit|Room|Building|Profile]” button
        5. “In use” toggle
           1. Set reminder to put item back?

Prompt for purpose and location

Prompt for timer until reminder

* + - 1. Stats
         1. Creation datetime
         2. Last change made datetime
      2. Expiration date/time
         1. A datetime variable to track time sensitive items
         2. Notification sent upon reaching date/time
         3. Examples: food “best by” date, warranty expiration, borrowed item return deadline, etc.
      3. Edit button

1. Search
   1. Located at top of every page
   2. Search within currently selected object
      1. Toggle to search globally within user
   3. Returns list of objects
      1. Grouped by tier
      2. Sorted by relevance
         1. Lower tier objects have higher priority
         2. Priority is applied within tiers by source of matching keyword
            1. Name > Tags > Description > Child Object (Name, Description, etc.), Parent Object (Name, Description, etc.)
   4. Options to search specifically by Name and Description, Tags, Location (parent object Name), or Date Added or Last Changed
2. Global/user settings
   1. About application
      1. How to guides
      2. App information
   2. Application settings
      1. Theme
   3. User settings
      1. Username
      2. Password

**UI**

1. Login/Signup
   1. Input for username and password
   2. Login/Sign up button
   3. Redirects to Main Page
2. Standard page layout
   1. Hamburger button menu in top-left corner
      1. Global search without options
      2. Shows current object’s breadtrail
      3. Stats button
      4. Settings button
      5. Logout
   2. Settings and Stats button in top-right corner
   3. Search bar between hamburger button and stats/settings buttons
   4. Top third has info about currently selected object
      1. Name
      2. Description
      3. Tags (if applicable)
      4. “Move” button
      5. “In Use” toggle (if applicable)
      6. “Stats” button
      7. Schedule expiration date button
      8. Edit button
   5. Next are buttons to add more items
      1. Redirects to add object page
   6. Bottom half has list of child objects
3. Search
   1. Creates a dropdown panel with more options
      1. Toggle for global vs current object search
      2. Sort by options
      3. Narrow search by type (name and description, tags, location, date added, date last changed)
   2. Results are shown on new page with all matching objects listed
      1. Grouped by tier
         1. Lower tiers have higher priority
      2. Sorted by “sort by” option (default by relevance)
4. Add object
5. Edit object