

C868 - Software Development Capstone

Project Summary

Task 2 – Section A

David Bozin

Contents

Business Opportunity	1
Existing Gaps	3
Software Development Life Cycle Methodology.....	3
Deliverables	4
Implementation	5
Requirements Validation	5
Environments and Costs	5
Timeline	6

Business Opportunity

Customer

The customer for this project is The Sofa Shop by Berkowitz, a formerly locally owned Australian furniture retailer location now owned and operated by Berkowitz Furniture following the recent retirement of the previous owner and manager. The Sofa Shop serves the greater Adelaide area, but also receives international customers that make the pilgrimage to shop there. Although The Sofa Shop has never developed an application for their customers before, they have been very successful with some of their viral marketing attempts, especially with their 80s sitcom theme song style advertisement, which has gained a cult following in recent years. The Sofa Shop's first priority is providing an excellent customer experience with whatever means, or medium necessary. Their hope is that this application will come to assist not only their own customers, but also furniture shoppers and users around the world.

Business Case

Currently, The Sofa Shop gets many customers that are frustrated with their organizational system, or lack thereof. They are frustrated that their current systems, albeit often well organized, do not lend themselves to quick and easy searching for a specific item. Customers will often purchase large and expensive shelves, cabinets, and other organizational and storage furniture to solve the problem, but end up returning them when they fail to provide a solution.

This application will provide the organizational solution that their customers so desperately need. The organizational and storage furniture will be presented as an aide for the problem, rather than the solution itself. Additionally, The Sofa Shop hopes that noncustomers who use the application may become customers upon realizing that proper, quality organizational and storage furniture enhances the system quite well, and come to shop at The Sofa Shop.

Fulfillment

The success of this application is fully dependent on how well it meets the needs of the client's customers. Therefore, fulfilling the needs of the customers fulfills the needs of the client.

The application will be an Android application located completely on the user's device, and not dependent on any external systems or connections. It will be designed to be run on the user's device without the need for an Internet connection beyond the initial installation and set-up.

The structure of the organizational system is as follows:

- The structure is a hierarchal tree structure.
- The top tier of the system is User, which is used to store and track all of the lower tiers, keep and present statistics, and save personal settings for the application.
- The second tier of the system is Profile. This is used to separate the different roles of a user, such as keeping work items separate from home items, etc.
- The third tier is Building. This tier is similar to the Profile tier, but more granular.
- Fourth tier is Room. This tier is more granular than the Building tier, but is different in that it allows the user to skip a tier and have both tier types of items coexist within it.
- Fifth tier is Unit. This tier is optional, but provides another level of granularity as a way to subdivide a room (e.g. cabinet, floor, desk, etc.). This tier is the highest one that can hold the final tier directly. Starting with this tier and all lower tiers, there will be an option to move this object to a different parent tier.
- Sixth tier is Box. This is the final level of granularity; however, Boxes may contain other nested Boxes, allowing for infinite granularity. This tier is also optional; however, either this tier or Units must be used between the Room and Item tiers. This tier and the Items tier both have a Tags feature, which associates keywords or phrases with the specific object to assist with searching.
- The final tier is Item, which is the individual item that needs to be recorded and tracked. The only additional feature is an optional "In use" toggle that can be used to record the reason for temporarily moving the item and the likely location it can be found until it is

returned. The “In use” feature also allows the user to set a timer reminder to put the item back in its proper location or change the location in the application. This tier can also store an expiration date or time variable for items that may be time sensitive, such as food items, items with warranties, or borrowed items that have a return deadline.

- Every tier will allow for individual objects to store a Name (required), Description, Settings, and Statistics/Reports of all of the child objects, creation time, and time of last change.
- Each object is presented on its own page, as well as its child objects and a button to return to its parent object (if applicable).

The user is encouraged to find a spare weekend to input all of the items needed to be tracked. Alternatively, the user may input items as they are acquired or moved.

The hierarchal structure is useful when inputting a new item, or finding an item in a generally known location, but may be very unhelpful when the general location of a searched item is unknown. A search bar will appear at the top of each page for this purpose.

- The search bar will by default only search within the children of the currently selected object. A toggle will allow for the ability to search globally within the selected user.
- The search will return a list of objects grouped by tier, sorted by relevance, giving priority to lower tier objects.
 - The order of the results will also depend on the source of the matched keyword or phrase. In order of higher priority to lower priority: Name, Tags, Description, Child Object, Parent Object/Location.
- The search feature will also provide options to search by Name and Description, Tags, Location, or Date Added/Last Changed.

Existing Gaps

Not applicable since the client does not currently, nor has in the past developed, owned, or managed such a software solution for this problem.

Software Development Life Cycle Methodology

The SDLC most suitable for this project is the Waterfall method for the following reasons:

- Very little guidance will be needed from the client. They will provide us with all the necessary requirements and expectations we need to complete this project satisfactorily, and allow us to work on our own terms, without feedback during the development process. The only feedback given, following the initial providing of the requirements, will be after the Design stage, and during the Verification and Maintenance stages.
- The requirements are clear and unlikely to change.

- Because the client is located in Australia, scheduling concurrent appointments would be difficult and often cause more problems than they would solve, due to the small overlap of business hours.
- The project is relatively small and isolated. The software does not need to integrate with any of the client's systems, and just needs to be deployed to mobile application stores upon approval from the client.

The Waterfall method used for this project will be composed of the following stages:

- **Requirements:** The client requirements and expectations are gathered, aggregated, and analyzed. A detailed specification plan is compiled based on the requirements and expectations.
- **Design:** Based on the specification plan, a design document is compiled. The design document consists of the class diagram and the application UI wireframe mockups.
- **Development:** Based on the specification plan and design document, the application is coded and tested until it meets all the requirements and is fully functional.
- **Verification:** A copy of the final application is delivered to the client for approval. If the original requirements and expectations are not met, then the development cycle returns to the previous stage. Otherwise, the application will be submitted to the mobile application store for distribution to the client's customers.
- **Maintenance:** Feedback will be collected from the users of the application via application reviews and customer support emails. Based on this feedback, errors in the application will be corrected for as long as one year, at which point the client has the option to extend our maintenance contract. The scope of this project does not cover adding new features not included in the original requirements.

Deliverables

Since the client has decided to remain less involved with the development process, there will be relatively few deliverables. These are the deliverables for this project in chronological order:

- **Project Timeline:** The estimated timeline for the project is required for the purpose of keeping the development on schedule for the planned release.
- **Wireframes:** Low fidelity wireframes are going to be the only resource for receiving approval of the design from the client. The wireframes will be delivered along with the timeline during the design phase.
- **Prototype:** After the application is fully developed and tested, a working prototype will be sent to the client for confirmation that it has met all the requirements and approval to finalize the project.
- **Copy of the finalized application:** Upon approval for finalization from the client, the finalized application will be submitted for publishing to the mobile application store. Both a copy of the application and a link to the mobile store application page will be sent to the client.

Implementation

Because this is a new application, rather than in iteration on an existing application, and because the application will be available to users through the mobile application store, the implementation for this project will be incredibly simple. Upon receiving approval for the final application, it will be properly prepared for upload to the mobile application store, along with all required supporting documents and files. A new developer account will be made for this purpose. A copy of the application and a link to the application store page will be sent to the client. The link will be used by the client for promotional purposes and placed on their website. After all the proper configurations and tests are completed on the mobile store, the application will be launched for the public. For the following year, the flaws in the application will be updated based on user reviews. At the end of the year, the client has the option to take over full control of the developer account that manages the application, or to start another contract for us to manage the application for them. If the client chooses to take over the management of the application, one of their employees will be given training and documentation on how to properly do so.

Requirements Validation

Before the prototype is sent to the client for testing, our own development and testing teams will thoroughly test the application to confirm that all requirements are met. The application prototype will then be sent to the client for testing on their end. The client will have several of their own employees use the application to confirm the requirements, as well as send it to select customers for comprehensive usability testing. The client will set up various simulated environments, input various objects of each tier, and run various scenarios to test each feature and confirm that the requirements are met. The customers selected for usability testing will be given the application to test in their own environments. They will be directed to use the application with minimal guidance for several days before being given a usability test and interview. The results of the tests and interviews will help guide decisions on future development and promotional direction. The purpose of the usability tests is to determine how individuals with varied backgrounds use the application, and will help guide future decisions for the development of the application.

Environments and Costs

Programming and Design Environments

- Windows 10 for coding, design, and testing
- Visual Studio 2019 for coding and testing
- MockFlow.com web application for designing wireframes
- Mobile device running Android 10 for testing

- A well-stocked office with several rooms for testing

Environment Costs

Because this is a simple and straightforward project, there will be no additional environment costs, as all the necessary resources are already owned and in operation.

Human Resource Costs (all values in USD)

- Planning and Design: ~10 hours x \$400/hr ≈ \$4000
- Coding and Project Management: ~60-70 hours x \$300/hr ≈ \$18000-\$21,000
- Documentation: ~20 hours x \$80/hr ≈ \$1600
- Total: ~\$23,600-\$26,600
- During the one year maintenance period following the publishing of the application, the cost of coding and project management will decrease both in hourly cost and total time spent. Hourly cost will be \$150/hr with a maximum of 40 hours per month.

Publishing Costs

- \$25 USD for mobile store developer account registration fee
- \$25,000 USD for advertising and marketing the application

Timeline

Phase	Deliverables	Description	Recourses	Dates
Requirements	NA	Collect requirements from client	Human resources costs	9/08/2020
Design	Timeline and Wireframes	Design class diagram and wireframes	Environment costs	9/08/2020 – 9/11/2020
Development	Prototype	Code and test application	Environment and human resources costs	9/14/2020 – 9/25/2020
Verification	Copy of finalized application	Send application to client for testing and approval	NA	9/28/2020 – 10/02/2020
Maintenance	NA	Maintain and update application	Publishing and one year maintenance costs	10/02/2020 – 10/02/2021