Team 7
Altug Gunes
Arif Akkas
Fercan Sen
Mehmet Savran
Osman Altug Ak
Tugcan Akkoylu

Design Document

SE 302

Principles of Software Engineering
Izmir University of Economics

1 Introduction

1.1 Information

CyberShelf is a program that helps user to store different type of items inside it with different attributes. User can also search for a specific item , can delete and edit it. Within the Software Design Document including use case diagrams, collaboration models, object behaviour models, sequence diagrams and other supporting requirement information.

1.2 Purpose

The purpose of the Software Design Document is to provide a description of the design of a system fully enough to allow for software development to proceed with an understanding of what is to be built and how is expected to built. The Software Design Document provides information necessary to provide description of the details for the software and system to be built.

1.3 Scope

This Software Design Document is a model of a system which wanted to build , giving only base level system and critical parts of it. For this particular Software Design Document, the focus is placed on generation of the documents and modification of the documents. Via using this model, aim is the create a specific and unique project.

1.4 Overview

The Software Design Document is divided into 6 sections. The sections of the Software Design Document are :

- Design Information
- Context
- State Dynamics
- UML Diagram
- Architecture
- User Interface

2 Design Information

CyberShelf is a software for categorize any items that users specify. All segments have photographs that will help user to understand the structure and design of the CyberShelf.

CyberShelf has a user friendly visual menu system for user interface that supports easy using.All visual attributes are explained in the user interface segment. The context shows and explains the relationship between the user and the CyberShelf's functions.All functions also have been explained. The UML diagram is a visual index for classes that will built CyberShelf's software.Type, Item and Attributes classes are located in UML diagram. Architecture shows how the software is going to be built and will be a future reference for how CyberShelf's software structure will be. All the user and system dynamics are shown at state dynamics segment.

3 Context:

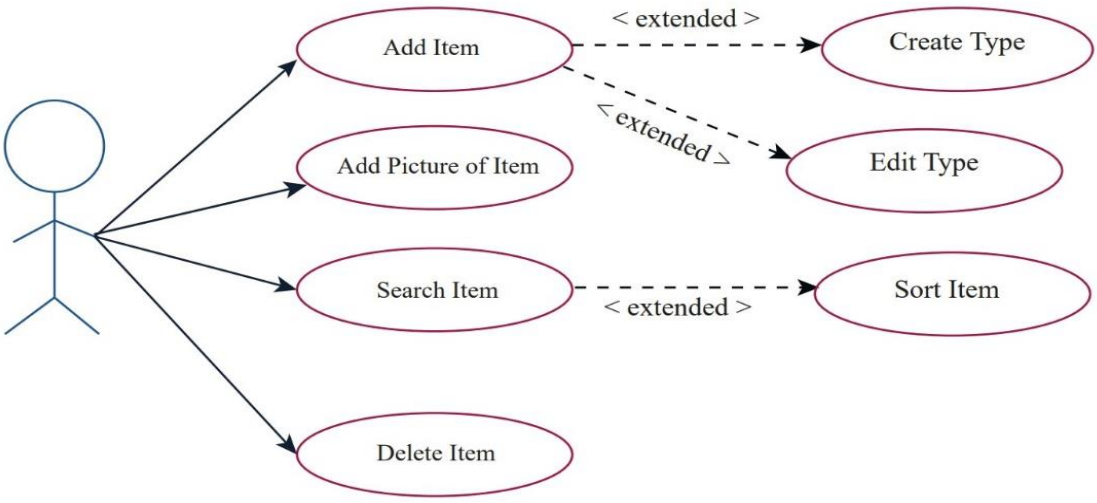

Figure 1: Context Diagram

CyberShelf provides a comfortable use for the user. User can easily add, delete and sort items according to your needs. User can also search for the added parts easily. No extra work is required. User can add images for the types to add. Also user can directly use the system and access its content any without any user authentication.

4 State Dynamics

4.1 System State Dynamics

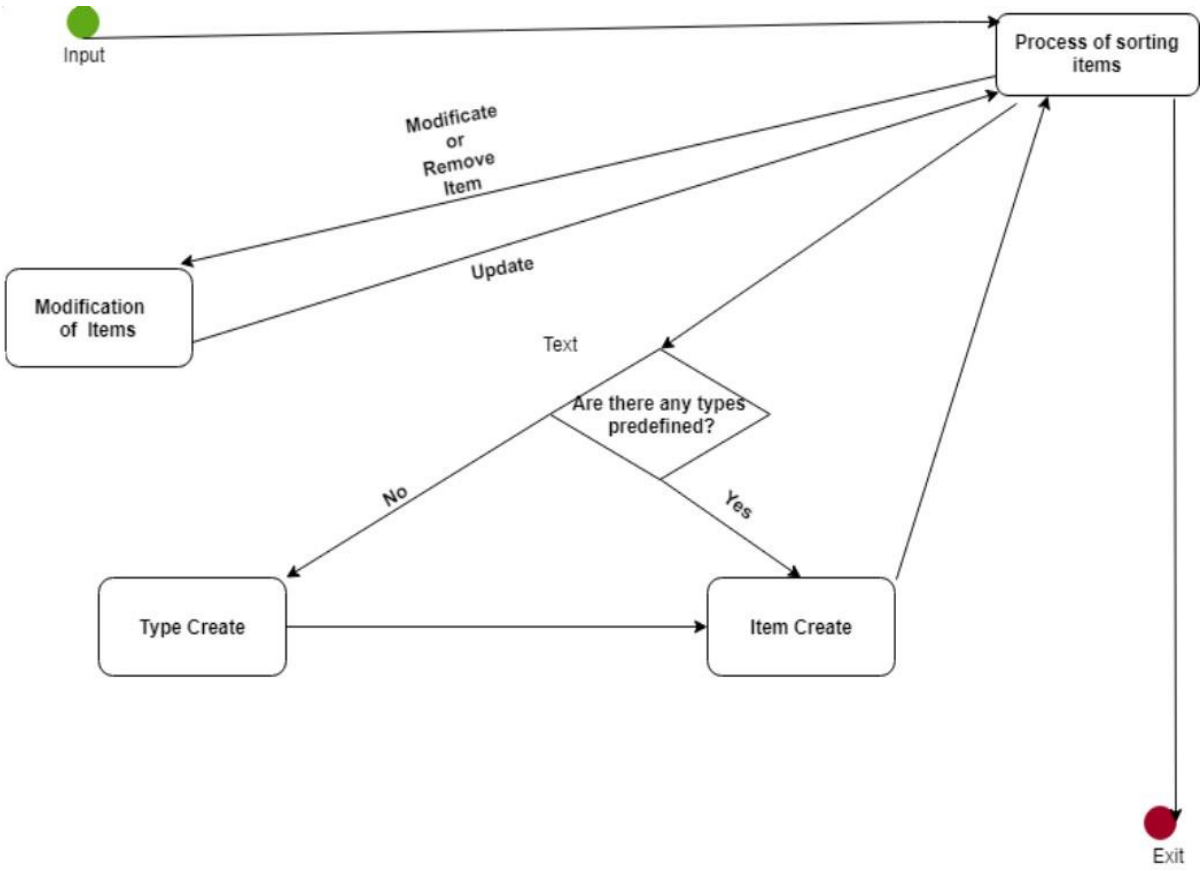

Figure 2: System State Diagram

This application will categorize the information of objects, items or work of arts etc... The application should work like a virtual shelf for collectors. User should be able to add any type of item and category he/she wishes.

Application bonds database and Java with connection using java.sql* package. For instance, we can search items using search class property of Java. As a result, in the nal product, user should be able to make a search and/or lter the results within the list of added items and sort the listed items. Using the same connection between Java and the database, user should be able to edit or delete the items stored in the database.

4.2 User State Dynamics

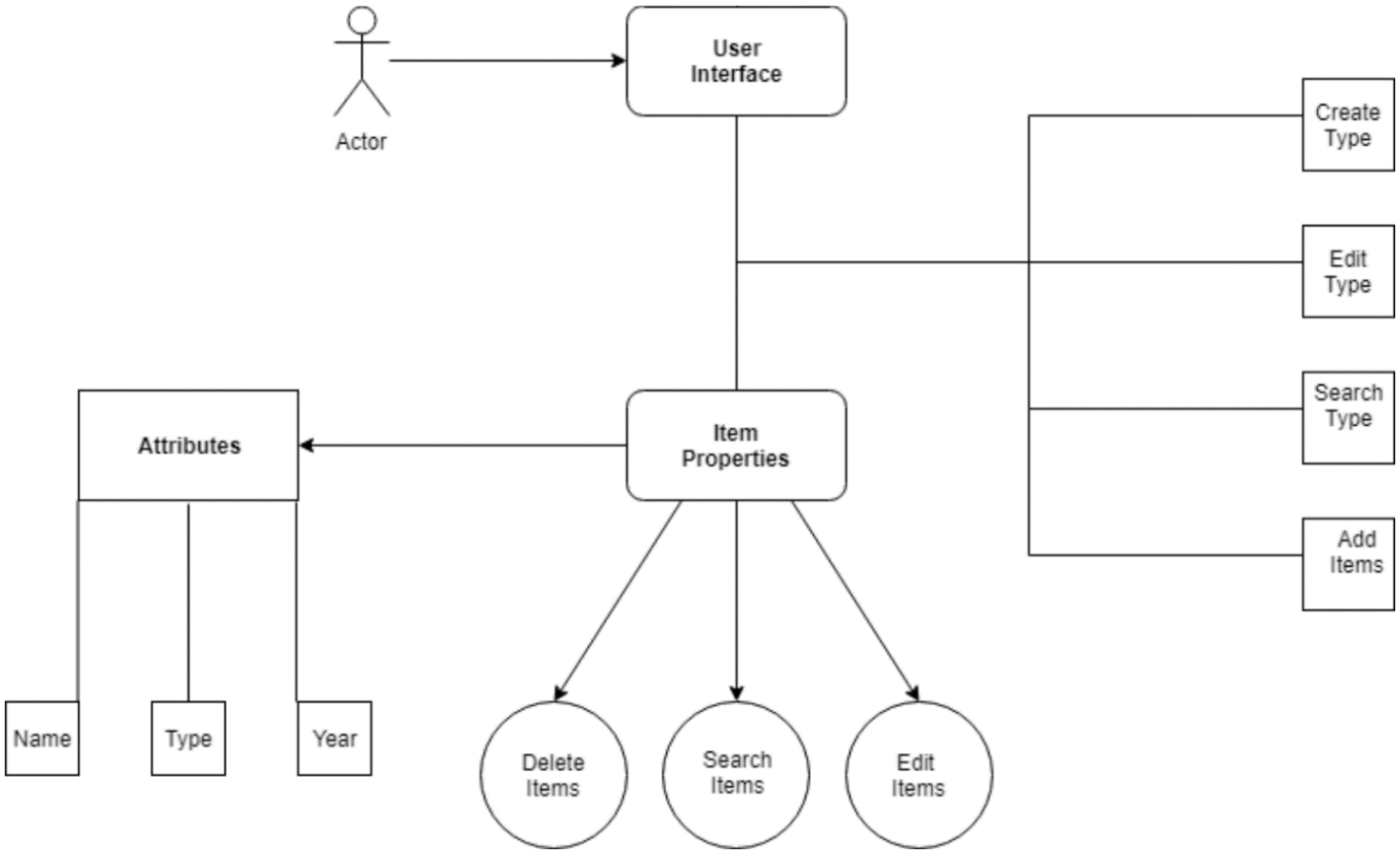

Figure 3: User State Diagram

5 UML Diagram

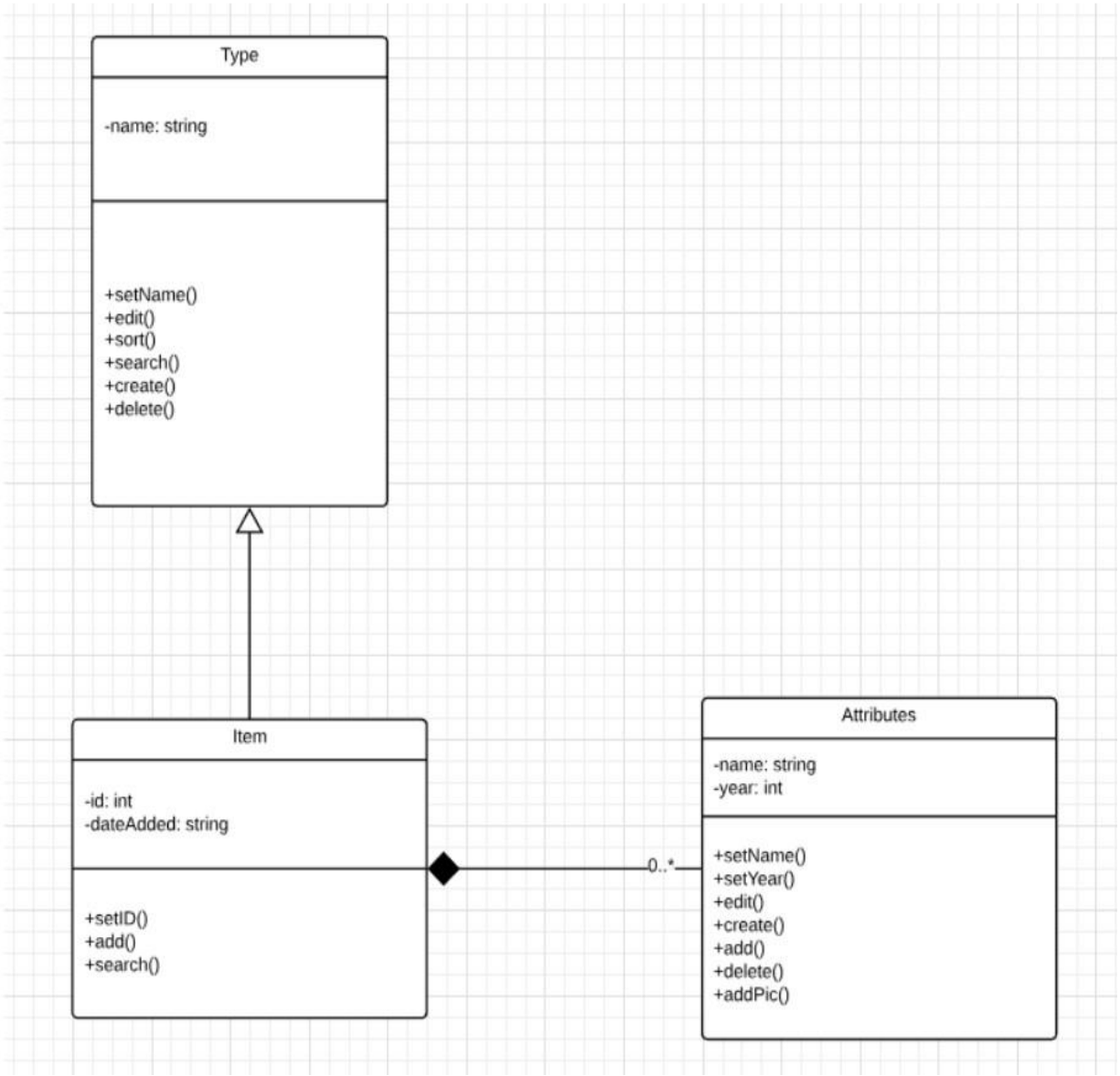

Figure 4: UML Diagram

6 Architecture

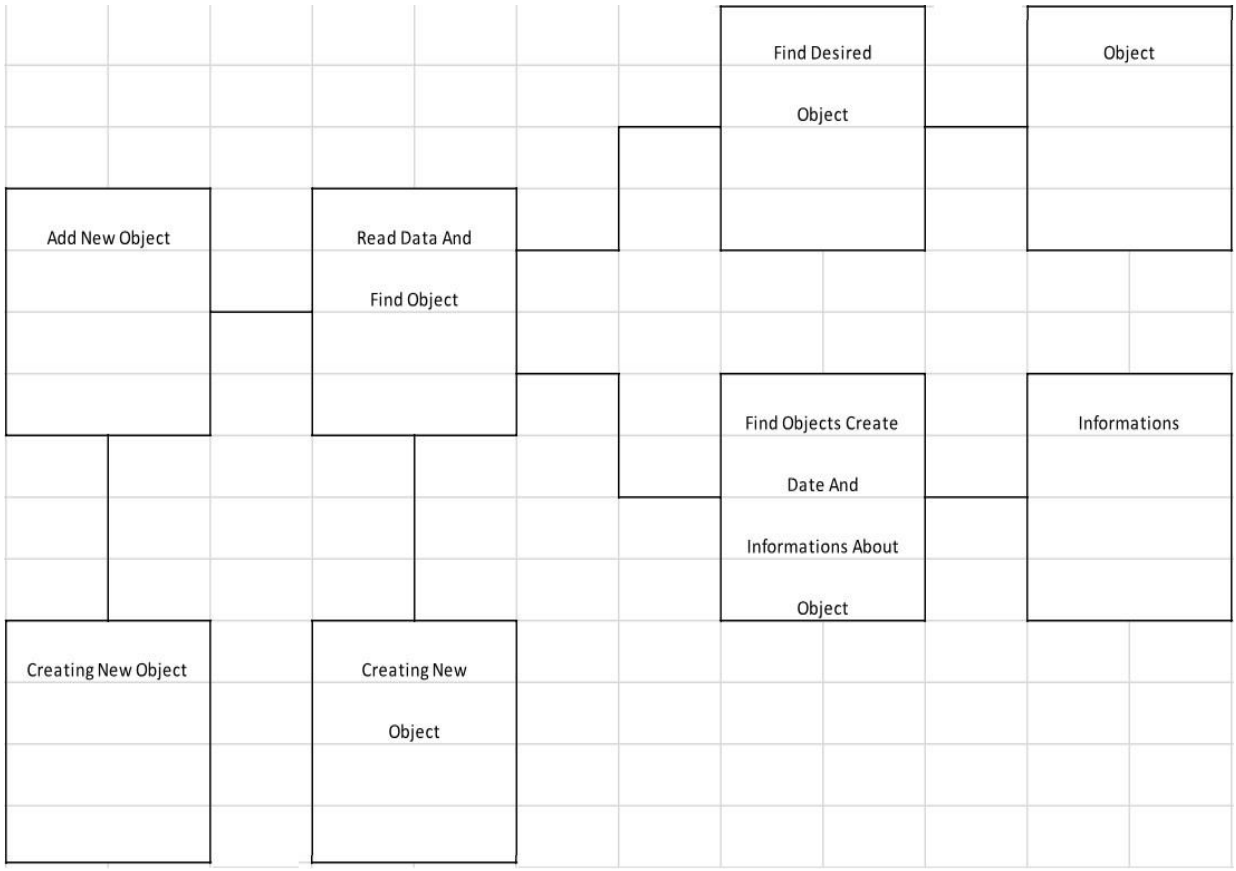

Figure 5: Context Diagram

7 User Interface

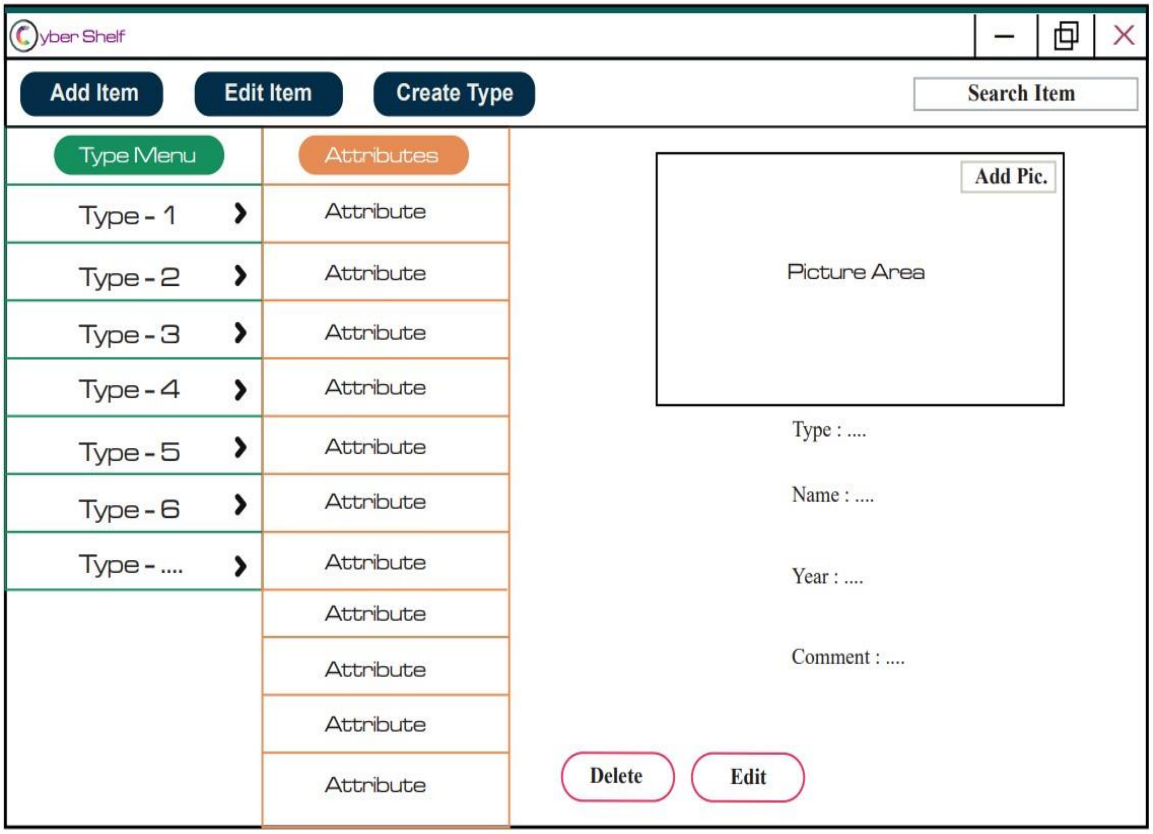

Figure 6: User Interface

The CyberShelf user interface has been designed in a modern and eyecatching way. It has been created as a user focused remote. The species to be added are grouped. User can easily add type from top menu and group these types on left side. The user can easily add types from the top menu and can group these types on the left side. Each type can divide itself into smaller segments. This provides an easy grouping option. You can also enter detailed information into these segments.