Bilkent University Computer Science Department

Object Oriented Software Engineering

CS-319

Summer Term Project Analysis Report

Dear Diary

Group Members:

- Abdurrezak Efe   21301883
- Ayşegül Sümeyye Kütük   20900538
- Enes Kavak   21302618

1. Introduction

Dear Diary is a Java GUI application that aims to provide users with a fancy interface and a password protected diary experience. Dear Diary has lots of features such as Notes, Special Days, Albums, Entries etc. User can take notes, create photo albums with descriptions of the photos, store special days such as birthdays, anniversaries etc. and most importantly can make diary entries and later display them. Finally, Dear Diary provides its users with a chatbot named Nyx that can participate in conversations.

The purpose of this project is fill the void in the industry as there are almost no desktop diary applications that is capable of mentioned features. The inputs are taken from keyboard and mouse.

This report aims to enlighten the readers about the technical and functional background of the project mentioned.

2. Application Overwiev

Dear Diary is a multiuser application that has many features that allow users to interact with the system. A user, who signed up to the system, can create a new entry which is the structure that stores his/her input information. He can also create notes to be reminded in future in a different frame. Storing special days, displaying old entries and creating albums is also provided for the user.

The application stores all the data of all of the users in .ser files withing the project file and it is not possible for them to be displayed outside of the application. Thus, the data of a user is only displayable for him as long as he does not forget his password. There is no password recovery system as the application does not work on a platform supported with internet. In this part, all of the mentioned features will be elaborated with details.

a. User

The user is the actor in this system. It is possible to create more than one account for multiple users. A user object has his entries, notes, albums, days and "Nyx talks". A Nyx talk is a conversation with the user and the chatbot. They are also optionally stored.

All of the other components that will be mentioned are simple supplementary objects for the user.

b. Entry

Dear Diary is designed to give its users a daily diary experience so the application has an entry interface that a user can write and add photos on a specific day and time. We did not specify any limitations concerning the number of entry that a user can make on a specific day so any number of entry can be made by the user and they will all be recorded through the system on that day.

When the logged in user attempt to make an entry, he can assign a title he desire or choose not to do so. Then the application will automatically assign the exact date of the system when the user records the entry. Users are also offered the chance to add images through their records. Old entries are saved and can be reached by the logged in users. They can view with all the content but any change cannot be done through the system including deletion. Once they are recorded in the system, they will remain in our database forever.

c. Note

The application cares about its users' life so it allows them to store the information that they would like to be reminded of. The Note object allows its user to add pictures about the note, an explanation and a title. If the user do not enter any of

them, the system will not save anything. If the title is the only missing thing, the system will make its title the date it is created as default. A note does not have to include a picture so it is not obligatory for the user to add an image to every note he creates.

The system will remind the logged in user about the upcoming ones before one day and the note's day. The notes that are passed by three or more days are deleted by the system. All of the existing notes can be displayed by the logged in user and can be deleted by himself.

d. Day

The application has a Day object that stores special dates for the logged in user. The days are classified into three subobjects:

- **Happy day (Marriage etc.)**
  - A Happy day object aims to store information about a Religious, National holiday or a celebration date.
- **Birthday**
  - A birthday, as the name suggests, stores data about the birthdays of the people the users care to store.
- **Sad day (Deaths etc.)**
  - This is, also self explanatory, for users to save about death anniversaries, seperations, farewells etc.

The user can add as many days as he wishes with their explanations. However, it is not possible to add images to days as these objects are representing a date not an event.

e. Album

Our application offers its logged in users to make an album with the images they desire to record. When a user does not want to write but wishes to keep the record of events with a number of images, s/he is able to build it with the application we built. For each album, the logged in user can add an explanation. In addition to the explanation, the application holds the certain date and time of the album had been made. The application system also keeps the information of the date of each and every photo created in the system.

f. Nyx

The name "Nyx" refers to the night goddess from ancient Greek. Nyx seemed to be a good name for our Dear Diary chatbot as people tend to write into their diaries during night.

It is going to be implemented in Java programming language as well as other parts. It is diplayed in a special Frame with a fancy GUI to make it a bit more attractive. The chatbot is just to make the program more fun for the users.

g. Statistics

The statistics is to help the logged in user to display analysis of his past inputs. It will show him the most used words, the most common hours that he logs in, the most common image types he saves(according to titles of the images) etc.

This part is going to be displayed in Home screen whenever the user logs in and will be refreshed in runtime whenever he uploads an entry, creates an album, deletes a note etc.

h. Methods

Methods are the series of events that determines the movements/actions that the logged in user make through the usage of the application.

1. Opening screen

    Sign Up: When a new user attempts to use the application, the sign up process has to be completed so that s/he can use all the features the application offers.

    Log In: When a new user complete the "Sign Up" process successfully, s/he can log in with the username and password they chose during the registration via the "Log In" operation.

2. Entry

    New Entry: The logged in user can make new entries through the new entry operation.

    Save Entry: When the logged in user use the new entry property and write a diary item, s/he has to save the content to record and display it later via the save entry operation.

    Display Entry: A logged in user is able to view the old entries s/he saved.

    Add Media: A logged in user can add as many number of images as s/he wants to the ongoing entry via the add media operation.

3. Notes

Add Note: A logged in user can keep the information of the appointments or reminders concerning a specific day in the future via the add note operation. The application will notify the user one day earlier and on the assigned date. The notification will remain on the notes part until it is completed.

Save Note: When the logged in user use the add note operation and write a note item, s/he has to save the content to record and display it later via the save note operation.

Display Note: A logged in user is able to view the old notes s/he saved. (Reminder: Old notes will be deleted by the system call three days later than the assigned date of the note item so they will not be displayed.)

Add Media: A logged in user can add as many number of images as s/he wants to the ongoing note item via the add media operation.

Delete Note: The logged in users can delete the old notes they saved through the delete note operation.

4. Days

Add Day: A logged in user can add a special day information grouped by happy day, sad day and birthday via the add day operation.

Delete Day: The logged in users can delete the old notes they saved through the delete note operation.

5. Albums

    Add Album: A logged in user can upload a series of images via the album operation.

    Save Album: When the logged in user use the add album operation and upload images, s/he has to save the content to record and display it later via the save album operation.

    Display Album: A logged in user is able to view the albums s/he saved via the display album operation.

    Delete Album: The logged in users can delete the old albums they saved through the delete album operation.

6. Nyx

    Talk to Nyx: Via the talk to Nyx operation, the logged in user can talk to the chatbot of the our application.

    Save Nyx Talk: The logged in users can save content of the talks they have made with the chatbot Nyx depending on their wish via the save Nyx talk operation.

3. Requirements Specification

   a. Functional Requirements

      i. Use Dear Diary

         Dear Diary is a Java GUI application that aims for its users to create entries, albums, notes, special days and update them however they wish.

         A candidate user first signs up to the system, logs in, creates an entry, determines its title, writes the input, adds media, saves it and can display it later. However, a user can never delete his entries if saved.

         Logged in user can create new notes about future appointments, due dates etc. and display them later. For effectiveness the notes can include photos chosen by the user. The user may also delete a note whenever he wishes. Moreover, a note will be deleted by the system if the date of the note is past by three days by default.

      ii. Change Settings

         Dear Diary provides its users with a flexible set of settings. The settings can be reached by the logged user from any frame inside the application.

         The user is able to change following properties:
         - His username
         - His password

- His profile picture
- The number of days before a note is deleted
- Background color of entry input text area and title

The changes applied in runtime so whenever a change is made and a new frame is created, the changes will be applied by the system.

iii. Show Statistics

Dear Diary collects as much data as possible from the user's inputs and calculates statistics based on that data. The statistics calculates will be displayed on Home screen. Some of them are:

- Most used words
- Most used hours of the application of the logged in user
- Most used hours of the application of the all users as a whole
- Most used word in titles

It is hoped that this information will provide us, the developers, with new ideas to add properties to the application's future versions if any.

iv. Terms of usage

The users, when signing up, have to accept the terms of usage, an aggreement between the users and developers, to be able to get signed up.

Terms of usage consist of some basic rules such as;

- If you do not save something before logging off, it is not going to be saved.
- If you forget your password you will never be able to recover it.

The detailed terms of usage will be provided with the application.

v. How to use?

Dear Diary provides its users with a detailed "How to use?" article with images. The article will include;

- What they can actually do
- What they will not be able to do
- Which button leads to what
- How to create, save, update and delete things
- How to change settings

Not only does this document help the users about how to use the program but also it guarantees them to make the most out of it.

vi. About us

Any user can make contact with the developers by using the contact informations given by the developers in About us frame. Via communicating, it is expected that the programs future versions will be better and more bug free.

b. Non-functional Requirements

i. Application Performance

The intended performance of Dear Diary is very high as people would not like to wait while creating new entries or albums etc. Since there is a chance there will be numerous users using the application on the same computer and as the instantaneous changes will be processed in runtime, the data to stream from storage should be as small as possible. Thus, it is considered best for Dear Diary to only stream the data of the current user and not all of the users. To achieve this, Dear Diary will store Java objects, using Serialization methods, into *".ser"* files which are a secure and common way to store objects. As a result of this user based storage and streaming, the data will be processed very fast.

Additionally, whenever a new frame is created, e.g. user pushes Notes button to see his notes when he is on Old Entries screen, the frame he will not be set invisible but rather it will be disposed. Hence, the memory performance will be better than its rivals.

ii. Graphical Decorativeness

As a diary application, Dear Diary will be considered one of the fanciest diary applications ever built. It is no secret that people are attracted by vivid colors, objects, and images. Thus, as developers of Dear Diary we intend to make the GUI as fancy as possible but not fancier. That is, whenever the fanciness of the system overhadows the useability, it means the trade-off is not thought enough.

   iii. Extendibility

     Starting to a project, a team needs to consider the possible future versions and additions for the project. We believe that the flexible design of Dear Diary will be a good example of the projects that can be extended, advanced and enhanced easily.

4. System Model

 a. Use Case Model

  This section provides information about the main use case models of Dear Diary application, detailed use case explanations are included below.

   i. **Use Case Name:** Add new entry
   ii. **Primary Actor**: Logged In User
   iii. **Stakeholders and Interests:**
    -The logged in user wants to add a new entry to his/her diary archive.
    -System helps him/her save the content s/he writes in the entry.
   iv. **Pre-conditions:** The user must be registered and logged in.
   v. **Post-condition: -**
   vi. **Entry Condition:** The logged in user selects "Add Entry" from his/her home page.
   vii. **Exit Condition:** The logged in user selects "Keep it secret" or do not choose to save and return to his/her home page.
   viii. **Success Scenario Event Flow:**
    ● System provides an entry interface for the logged in user to create content.
    ● Logged in user creates content via the provided interface.
   ix. **Alternative Flows:**

- If the logged in user desires to return to home page at any time:
- Logged in user selects "Return to Home Page" button to return main menu without saving the content created.
- Logged in user hits the "Keep it secret!" button and system displays the home page.