Research Paper 5

HAYDEN VASS | DVP 2
E: HAVASS@STUDENT.FULLSAIL.EDU

MOBILE DEVELOPMENT
FULL SAIL UNIVERSITY

Application Deconstruction

TACC APPLICATION

SECTION 1

The fictitious application company called TACC has been able to find success in their children friendly learning applications by sticking to a single lay out. This research paper will discuss how TACC accomplishes these applications by looking into the possible ways their developers could have coded it.

The home page to all TACC applications have the same two buttons. These buttons are “read” and “buy books”. This home screen most likely operates off an initial switch statement. The second case which is “buy books”, takes the user to a different page that shows what book options are available. If this application is web based, then this switch case could take the user to the applications website. The website would have all the book information available to the user, and pulls this information from a databased owned by TACC. The user could then purchase and download new books as they wish.

The users books would be a custom data type. This book class would contain properties such as title, author, and publishing date. Depending on the content of the book. The text could stored as simple blocks of strings, or in a text document that gets pulled into the application. The books the user have available would be stored in a list called usersBooks. This would make it simple for the application to loop through each book and choose the option the user wanted to read.

The other option from the home page is called “Read”. This option would loop through the list of usersBooks and print their available options to the screen. Once the user picks a book, that book most likely gets assigned to a currentBook variable and another switch statement gets shown to the user. This switch statement contains methods and classes for the user’s ability to read a book, do book related activities, and to buy books.

The “Read Book Button” takes in the currentBook parameter. And presents the user with a list of chapters available in that book. This is most likely done through another switch statement and a loop of chapter titles that are contained in a list from the book class. Each chapter would an instance of a chapter class. The chapter class would inherit from the book class and contain properties such as chapter number and chapter title, and page number. There could be a “Read” method that takes in a chapter as a parameter and show just the information from that chapter.

The activities section of the application would hold another switch statement that would switch between the coloring aspect of the app, the quiz section and anything else

TACC would want to add. In this example, each book would have files associated with each book These files could be stored locally or from a database. A method would be used to pull in these files and put them on the device. This image would sit in the back ground as the user colored over it.

The quiz section could have another switch statement and a menu to choose from a list of quizzes to take. This is another opportunity to utilize a custom data type. TACC developers could have a quiz class with questions, answer keys and a method to calculate score. A currentQuiz variable could be created to remain null until the user chooses a quiz. The selected quiz files would then be pulled in from the device and assigned to the place holder variable. These quizzes could be stored in a list from which the user could choose from. There could also be a random quiz function. In this case, a random method would be instantiated to pick a number between one and the count of the list containing all the quizzes.

The game button would contain another menu function that held a switch statement to hold all the possible options. Much like the rest of the application, a currentGame variable would be populated from the parameters pulled in from either the device or a database.

The final button in the main feature is the buy book button. This would simply redirect the user through the same path from the initial home page to the website and purchase more books.

POSSIBLE CODE USED BY TACC

SECTION 2

MENU THAT RUNS THE MAIN FUNCTION OF THE APPLICATION

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Thesis5
{
    class Menu
    {
        private Users currentUser;
        private Book currentBook;
        private Game currentGame;
        private Activites curretnActivity;
        private string userInput;

        public Menu()
        {
            SwitchStatement();
        }


        private void SwitchStatement()
        {
            switch (userInput)
            {
                case "Read":
                    {
                        currentBook = new Book();
                        Console.WriteLine("Please choose a book.");
                       foreach(Book book in currentUser.usersBook)
                        {
                            Console.WriteLine(book.Title);
                        }
                        string input = Console.ReadLine();
                        foreach(Book book in currentUser.usersBook)
```

MENU THAT RUNS THE MAIN FUNCTION OF THE APPLICATION

SECTION 2

```
                    {
                        if (book.Title.Contains(input))
                        {
                            currentBook = book;
                        }
                        else
                        {
                            Console.WriteLine("book does not exist.");
                        }
                    }
                }
                break;
            case "Activites":
                {
                    curretnActivity = new Activites();
                    Console.WriteLine("Please choose an activity.");
                    foreach (Activites activity in currentUser.userActivites)
                    {
                        Console.WriteLine(activity.Name);
                    }
                    string input = Console.ReadLine();
                    foreach (Activites activity in currentUser.userActivites)
                    {
                        if (activity.Name.Contains(input))
                        {
                            curretnActivity = activity;
                        }
                        else
                        {
                            Console.WriteLine("Activity does not exist.");
                        }
                    }
                    curretnActivity.DoActivity();

                }
                break;
            case "Games":
                {
                    currentGame = new Game();
                    foreach(Game game in currentUser.userGames)
                    {
                        Console.WriteLine(game.Name);
                    }
                    Console.WriteLine("Please choose a game.");
                    string input = Console.ReadLine();
                    foreach (Game game in currentUser.userGames)
                    {
                        if (game.Name.Contains(input))
                        {
                            currentGame = game;
                        }
                        else
                        {
                            Console.WriteLine("game does not exist");
                        }
                    }
                    currentGame.PlayGame();
                }
                break;
            default:
                Console.WriteLine("please choose a valid response");
                break;
        }
    }
}
```

BOOK CLASS

SECTION 2

```
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.IO;

namespace Thesis5
{
    class Book
    {
        public string Title { get; set; }
        public int Pages { get; set; }
        private string _file = @"bookTitle.txt";
        private string _folder = @"..\..\output\";
        public List<Chapter> ChaptersList;

        public Book(Book currentBook)
        {
            ChaptersList = new List<Chapter>();
            load();
            TableOfContents();
            BookSwitch();
        }
        public Book()
        {

        }
```

BOOK CLASS

SECTION 2

```
29
30          protected void TableOfContents()
31          {
32              foreach(Chapter chapter in ChaptersList)
33              {
34                  Console.WriteLine($"{chapter.ChapterNumber} {chapter.ChapterNumber} ..... {chapter.StartingPage}");
35              }
36          }
37          protected void BookSwitch()
38          {
39              Console.WriteLine("What chapter would you like to read?");
40              int userInput = int.Parse(Console.ReadLine());
41              foreach(Chapter chapter in ChaptersList)
42              {
43                  Console.WriteLine(chapter);
44              }
45              switch (userInput)
46              {
47                  case 1:
48                      Read(0);
49                      break;
50                  case 2:
51                      Read(1);
52                      break;
53                  case 3:
54                      Read(2);
55                      break;
56                  default:
57                      Console.WriteLine("Please choose a valid choice."); ;
58                      break;
59              }
60              Read(userInput);
61          }
62          protected void load ()
63          {
64
65              using (StreamReader inStream = new StreamReader(_folder + _file))
66              {
67                  while (inStream.Peek() > -1)
68                  {
69                      string line = inStream.ReadLine();
70
71
72                      ChaptersList.Add(new Chapter(line));
73                  }
74              }
75          }
76          public void Read(int chapter)
77          {
78              Console.WriteLine(ChaptersList[chapter]);
79          }
80      }
81
82  }
```

BOOK CHAPTER CLASS

SECTION 2

```
6
7   namespace Thesis5
8   {
9       class Chapter: Book
10      {
11          private static Book currentBook;
12
13          public int ChapterNumber { get; set; }
14          public string ChapterName { get; set; }
15          public int StartingPage { get; set; }
16          public string Content { get; set; }
17
18          public Chapter(string line) : base(currentBook)
19          {
20              string[] book = line.Split(',');
21              ChapterName = book[1];
22              ChapterNumber = int.Parse(book[0]);
23              StartingPage = int.Parse(book[2]);
24              Content = book[3];
25
26          }
27
28      }
29  }
30
```

SECTION 2

GAME CLASS

```
7   namespace Thesis5
8   {
9       class Game
10      {
11          public string Name { get; set; }
12          public string Rating { get; set; }
13          public Game()
14          {
15
16          }
17
18          public void PlayGame()
19          {
20              //
21          }
22
23      }
24  }
25
```

ACTIVITIES CLASS

```
namespace Thesis5
{
    class Activites
    {
        public string Name { get; set; }

        public Activites()
        {

        }

        public void DoActivity()
        {
            // do activity
        }

    }
```

USER CLASS

SECTION 2

```
using System.Threading.Tasks;

namespace Thesis5
{
    class Users
    {
        public string Name { get; set; }
        public List<Book> usersBook = new List<Book>();
         public List<Game> userGames = new List<Game>();
         public List<Activites> userActivites = new List<Activites>();

         public Users()
         {

         }
    }
}
```

REFERENCES

Cao, J. (2015). How to design with developers. The Next Web. Retrieved 16 September 2018, from https://thenextweb.com/dd/2015/10/20/how-to-design-with-developers/

The Pros and Cons of Mobile Apps in Modern Business - Mobile Messaging. (2018). Mobile Messaging. Retrieved 16 September 2018, from https://uplandsoftware.com/mobile-messaging/resources/blog/pros-cons-mobile-apps-modern-business/

Gerber, S. (2013). 11 Tips for Developing a Mobile App that Users will Love. The Next Web. Retrieved 16 September 2018, from https://thenextweb.com/dd/2013/05/25/11-tips-for-developing-a-startup-mobile-app-users-love/

How To Make An App - Create An App In 12 Easy Steps. (2016). Bizness Apps. Retrieved 16 September 2018, from https://www.biznessapps.com/blog/how-to-build-a-mobile-app-in-12-easy-steps/

Pros and Cons of Mobile Websites and Mobile Apps. (2018). Rubygarage.org. Retrieved 16 September 2018, from https://rubygarage.org/blog/mobile-app-vs-mobile-website

Pros and Cons of Mobile Websites and Mobile Apps. (2018). Rubygarage.org. Retrieved 16 September 2018, from https://rubygarage.org/blog/mobile-app-vs-mobile-website

5 things to consider when designing an app - We Are Hinge. (2018). We Are Hinge. Retrieved 16 September 2018, from https://www.wearehinge.com/blog/designing-an-app/

HAYDEN VASS | DVP 2
E: HAVASS@STUDENT.FULLSAIL.EDU

MOBILE DEVELOPMENT
FULL SAIL UNIVERSITY