Part I:

1. MessageBox.show(“some string value”) -> has LOTS of overloads
2. A method that is called when an event is raised. This method “handles” the event. For example, we want the colour of a box to change between red and blue between mouseclicks. We write an event handler that “listens” for mouse clicks and provides some functionality (change colour from red to blue) when that event is raised.
3. The steps are:
   1. Open visual studio
   2. File -> New -> Project
   3. Make sure the target framework is >= 2
   4. Select Windows Form Application
   5. Choose name of project and name of solution
   6. Choose location of solution
   7. Press Ok
4. With: this.TopMost = true. This can be useful for validation. For example, if an application allows the user to delete certain information a popup window should appear making sure the user actually wants to delete the information. This popup should stay on top of the main window until the user validates their answer.
5. **Menu:** a list of options or [commands](http://en.wikipedia.org/wiki/Command_(computing)) presented to an [operator](http://en.wikipedia.org/wiki/Operator_(profession)) by a computer or [communications system](http://en.wikipedia.org/wiki/Communications_system)\*

**Shortcut Menu:** a [menu](http://en.wikipedia.org/wiki/Menu_(computing)) in a [graphical user interface](http://en.wikipedia.org/wiki/Graphical_user_interface) (GUI) that appears upon user interaction, such as a right-click [mouse operation](http://en.wikipedia.org/wiki/Mouse_(computing)#Operation)\*

**Status Bar:** A horizontal bar, typically at the bottom of the screen or window, showing information about a document being edited or a program running.\*

**Toolbar:** A strip of icons used to perform certain functions.\*

\*Taken from Wikipedia

Part III:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Sex | Age | City | Nationality | Occupation |
| Bin Zhou | M | 20-25 | Winnipeg | China | student |
| Lihan Li | M | 20-25 | Winnipeg | China | student |
| Zifeng Qian | M | 20-25 | Winnipeg | China | student |

1. Usability Criteria

Effectiveness:

1. Users can easily query information on any type of movies from application within 5 minutes.
2. User can effectively provide feedback on a specific movie?
3. User can easily retrieve the movie information he found before.

Efficiency:

1. Time to query for information of movies.
2. Number of steps to complete a search, a rating and a store of a specific movie.
3. The mental demand of user using this system.

Safety:

1. Avoid losing user’s personal private information from personal account
2. Provide confirmation and undo for user deleting movie from favorite list.
3. Can send email to user to reset account password when user forget.

Utility:

1. Searching: It allows users to search movie by some of its information
2. Storage: It provides people with Favorite list that can record people’s favorite movie
3. Allow user to setup personal account.
4. Allow user to upload movie information to application.
5. Recommendation: It allows users to contribute personalized information such as add ratings or add recommendations.

Learnability:

1. The application should require little time for first time user to learn how to use the software to find specific movie.
2. The application should require little time for user to learn how to share movie information.

Memorability:

1. User should be able to easily remember the interaction of finding movies.
2. User should be able to easily remember the interaction of movie sharing.

User Experience Goal:

Satisfying:

1. Users can get information about those movies that they are interested through some simple interaction.
2. User can simply share their experience about a movie on application.

Motivating:

1. The searching time should be small so user will not feel frustrated
2. The interaction should not be complicated, so user will not feel tired

Helpful:

1. Users can use a short time to get information.
2. Also they can record a list of movies’ information in favorite list.
3. Identify Needs

Functional

1. Allow user to search for a specific movie, by using genre, actor, director.
2. Allow user to suggest movie to friends and get suggestion from friends.
3. Allow user to provide feedback to a movie.

Data

Movie name

Introduction

Actor

Rating

Movie type

Friend’s recommendation

Environmental (physical, social, organizational, technical)

Physical: Home, on bus

Technical: Online

User

People that loves watching movie

Movie Company (upload information)

People that don’t like movie, but can get recommend movie from friends.

Usability

The application should let user easily to find a movie using some relative information.

1. Scenarios
2. XXX is a 22 year old male student. He really enjoys watching action movies. He likes download movie online. Yesterday he looked for some action movies at home in the evening. He browsed some movie website and found that people are recommending a new movie acted by Jackie Chan. Since Jackie Chan is his favorite action movie star. He found and downloaded the resource. He watched the movie and found that it is meeting his expectation. He rated the movie online and gave some feedback.
3. YYY is a 23 year old male student. He watched movie once a week. He likes watching movie both at cinema and at home. 3 days ago, he was checking a movie website at home at 9:00pm. As he likes director Quentin Taramtino, he started searching for his movies. He found 10 results. He watched each one’s trailer and found a very interesting one. He phoned his friend and recommended him the movie. He then checked the closest cinema to his home. After reading the playing schedule for this movie. He decided to watch it the next day on 8:30 pm. After watching the movie, he really liked it. He gave a rating on this movie and commented it. As he like collect those good movies and may watch it again. He recorded the movie information on his computer.
4. Task Analysis
5. Open the movie website.
6. Search for director Quentin Taramtino
7. Browsing each movie directed by Quentin
   1. Read the movie introduction
   2. Check the rating
   3. Watch the trailer
   4. Decide whether to watch this movie. if not do step e. if yes go step 4.
   5. repeat a to d, until all Quentin’s movies are checked.
8. Suggest it to friends and invite friends
9. Find the closest cinema that plays the movie.
10. Check the playing schedule. Decide a time to watch the movie.
11. After watching the movie, rate the movie, comment it
12. Record it’s information to follow its series or further information

Invite friends 1

Find resource 2

Find a movie 0

Recommend to a friend 4.3

Give a mark 4.1

Comment a movie 4 3

Find closest cinema

2.1

Watch movie 3

Follow the movie series 5

Give a comment 4.2

Pick a time

2.3

Find schedule

2.2

Watch trailer

0.3.3

Check actor

0.3.2

Check introduction

0.3.1

Pick a movie

0.3

Find a director

0.2

Open website 0.1