**Adriana Arzola**

**Erik Whiting**

SWEN5232 - Software Construction

Assignment Number: **Final**

Professor: Dr. Findler

findler@uhcl.edu

Teaching Assistant:

NA

**Table of Contents**

[Revision History 6](#_Toc4234166)

[Github Project 6](#_Toc4234167)

[Overview 9](#_Toc4234168)

[Constraints 9](#_Toc4234169)

[Conceptual framework 9](#_Toc4234170)

[Brainstorm 10](#_Toc4234171)

[Figure 1-1 - Acolyte Brainstorming Diagram 10](#_Toc4234172)

[Use Cases 11](#_Toc4234173)

[Figure 1-2 - Acolyte Main Use-Case Diagram 11](#_Toc4234174)

[Use Case Diagrams 12](#_Toc4234175)

[Figure 1-3 - Application Assistance 12](#_Toc4234176)

[Figure 1-4 - Audio Settings 13](#_Toc4234177)

[Figure 1-5 - Display Settings 14](#_Toc4234178)

[Figure 1-6 - Manage Bluetooth 15](#_Toc4234179)

[Figure 1-7 - Manage Computer Settings 16](#_Toc4234180)

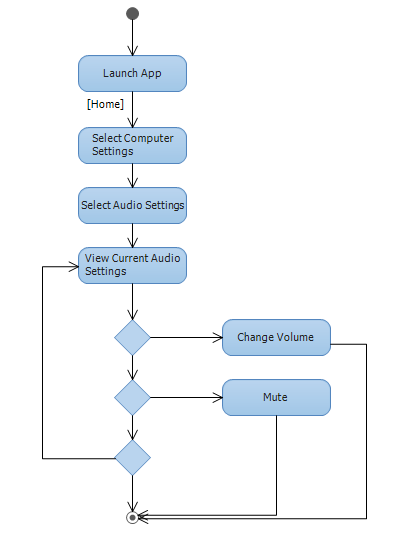
[Figure 1-8 - Manage WiFi Settings 17](#_Toc4234181)

[Figure 1-9 - View Computer Statistics 18](#_Toc4234182)

[Figure 1-10 - Web Browser Assistance 20](#_Toc4234183)

[Activity Diagrams 22](#_Toc4234184)

[Figure 2-1 Application Assistance Activity Diagram 22](#_Toc4234185)

[ 23](#_Toc4234186)

[Figure 2-2 Audio Settings Activity Diagram 23](#_Toc4234187)

[Figure 2-3 Manage Bluetooth Activity Diagram 24](#_Toc4234188)

[Figure 2-4 Manage Display Settings Activity Diagram 25](#_Toc4234189)

[Figure 2-5 Manage WiFi Activity Diagram 26](#_Toc4234190)

[Figure 2-6 View Computer Statistics Activity Diagram 27](#_Toc4234191)

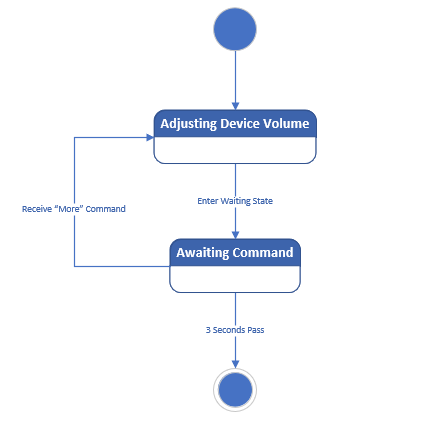
[Figure 2-7 Web Browser Assistance Activity Diagram 28](#_Toc4234192)

[State Diagrams 29](#_Toc4234193)

[Figure 3-1 High Level State Diagram 29](#_Toc4234194)

[Figure 3-2 Connect to WiFi 30](#_Toc4234195)

[Figure 3-3 Get Computer Statistics 31](#_Toc4234196)

[ 32](#_Toc4234197)

[Figure 3-4 Increase or Decrease Volume 32](#_Toc4234198)

[Figure 3-5 Browser Use-Cases State Diagram 33](#_Toc4234199)

[Class Diagrams 34](#_Toc4234200)

[Figure 4-1 Application Class 34](#_Toc4234201)

[Figure 4-2 Browser Classes 35](#_Toc4234202)

[Figure 4-3 System Statistic classes 36](#_Toc4234203)

[Figure 4-4 Devices Classes 37](#_Toc4234204)

[Figure 4-5 Monitor Classes 38](#_Toc4234205)

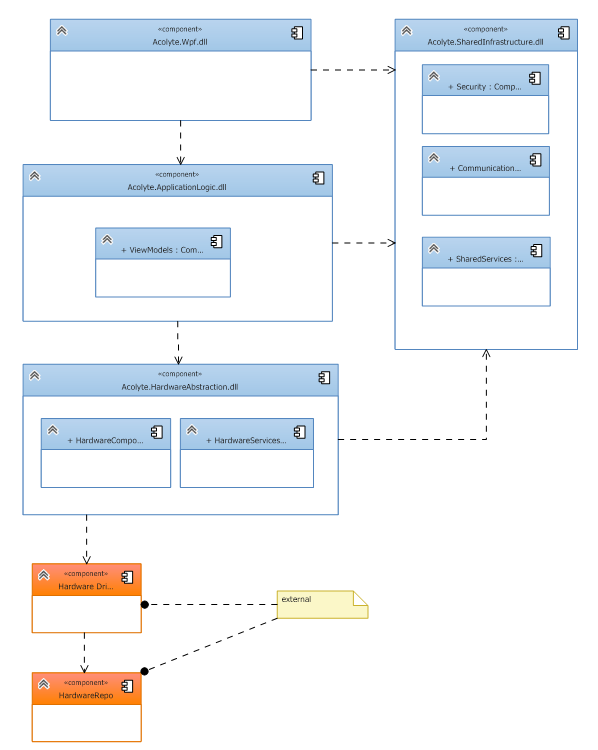
[Layer Diagrams 39](#_Toc4234206)

[Figure 5-1 Acolyte Layer Diagram 39](#_Toc4234207)

[Domain Model 40](#_Toc4234208)

[Figure 6-1 Domain Model 40](#_Toc4234209)

[Component Diagram 41](#_Toc4234210)

[ 41](#_Toc4234211)

[Figure 7-1 Component Diagram 41](#_Toc4234212)

[Components 41](#_Toc4234213)

[External Components 42](#_Toc4234214)

[Figure 7-2 External Components 42](#_Toc4234215)

[Acolyte.SharedInfrastructure.dll 43](#_Toc4234216)

[Figure 7-3 Shared Infrastructure Component 43](#_Toc4234217)

[Acolyte.HardwareAbstraction.dll 43](#_Toc4234218)

[Figure 7-4 Hardware Abstraction Component 44](#_Toc4234219)

[Acolyte.ApplicationLogic.dll 44](#_Toc4234220)

[Figure 7-5 Application Logic Component 44](#_Toc4234221)

[Acolyte.Wpf.dll 44](#_Toc4234222)

[Figure 7-6 Acolyte WPF Component 45](#_Toc4234223)

[Sequence Diagram 46](#_Toc4234224)

[Figure 8-1 Sequence Diagram 46](#_Toc4234225)

[How to Use the Acolyte Prototype 47](#_Toc4234226)

[Git History: 48](#_Toc4234227)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Date | Changes | Version |
| Erik Whiting | 3/18/2019 | Initial Draft and Framework | 1.0 |
| Erik Whiting | 3/20/2019 | Use cases | 2.0 |
| Erik Whiting | 3/21/2019 | State Diagrams | 2.1 |
| Adriana Arzola | 3/22/2019 | Updated diagrams | 2.2 |
| Adriana Arzola | 3/23/2019 | Add Activity Log section | 2.3 |

# Github Project

The github project can be found at <https://github.com/swen5232/Final>

# Overview

Acolyte is designed to be a personal virtual assistant to assist less tech savvy people use their personal computers via a simple interface. The application will run as a Windows Service in the background, but will also have a simple UI. The service runs so that users can use voice commands or keyboard shortcuts to utilize Acolyte’s features. The UI will expose all the features as well, in case the user does not feel comfortable making verbal commands to their computer.

## Constraints

The system assumes that the user has a minimal English knowledge as of now.  
The system can only run on Windows operating systems at this time.

## Conceptual framework

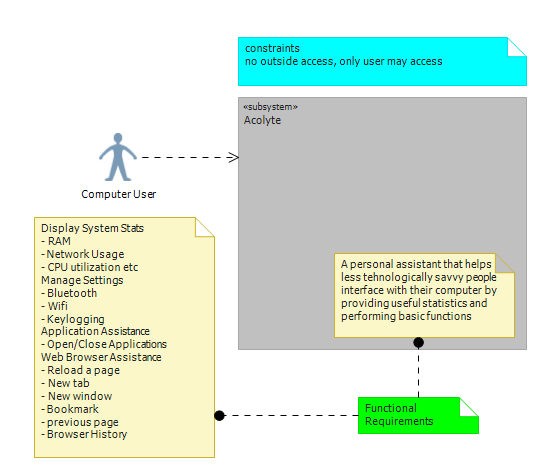
There is always scope for improvement. So the application built using M.V.C architecture, incremental process design. These design structures follow the modular approach which makes it easy to add in new features to the system.

The system also implements the singleton pattern and the single responsibility principle which ensure the individual functioning of the modules.

Functions: The system provides all the below stated functions.

* displaying system information such as network usage, ram etc.
* opening and closing an application
* altering system brightness, volume
* accessing Bluetooth, WIFI -bookmarking a webpage
* creating a new window or a tab in browser
* going to the previous webpage on the web browser
* reloading a webpage on the web browser
* keylogging with keystrokes segregated according to the application and time stamps.
* browser history tracking

# Brainstorm

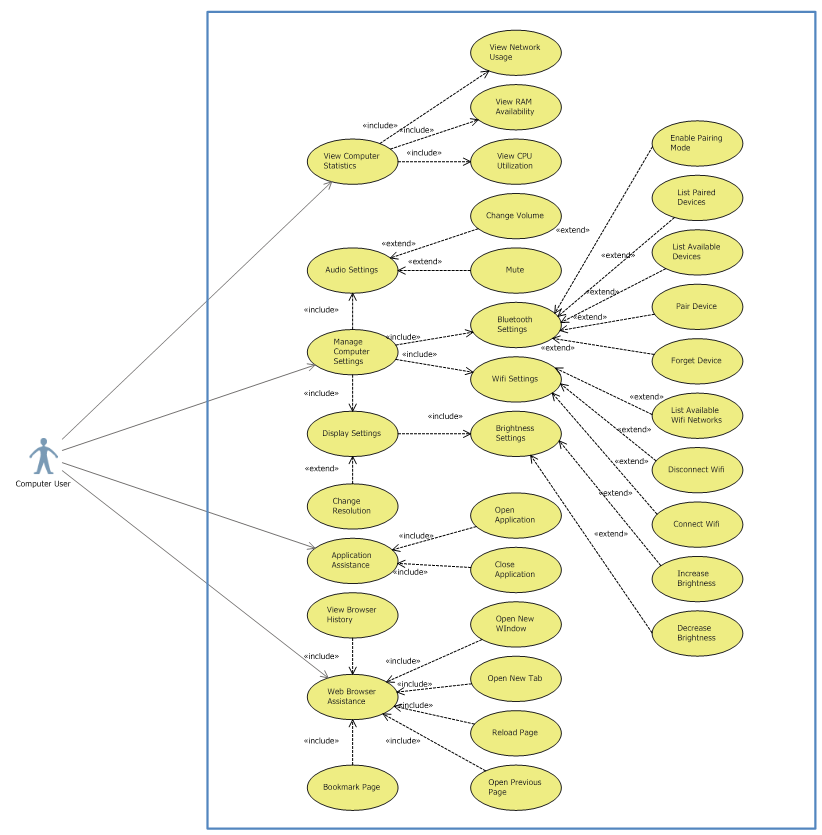


### Figure 1-1 - Acolyte Brainstorming Diagram

# 

# 

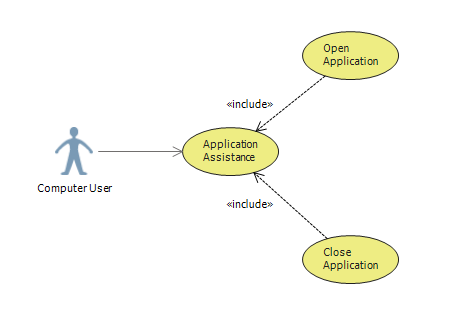
# Use Cases



### Figure 1-2 - Acolyte Main Use-Case Diagram

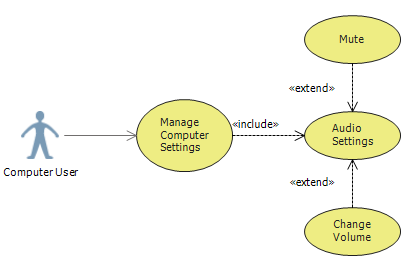
### 

## Use Case Diagrams



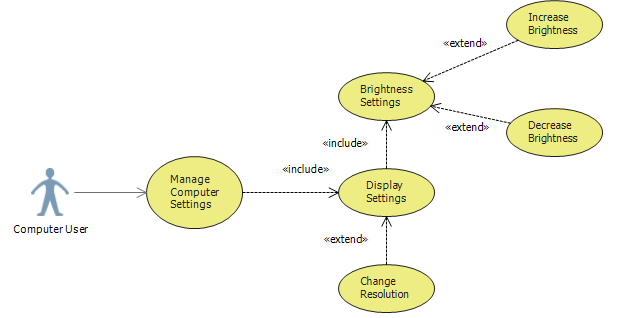
### Figure 1-3 - Application Assistance

|  |  |
| --- | --- |
| **Use Case #** |  |
| **Name** | Application Assistance |
| **Actors** | Computer User |
| **Description** | User can utilize Acolyte for application assistance. The application assistance feature will be able to do anything a Task-Manager type application can do. User can bring an application into view, minimize, maximize, close, and get statistics about any application. |
| **Workflow** | 1. User activates Acolyte (voice, keyboard shortcut, or clicking application icon) 2. User says/types/clicks on operation    1. Open    2. Close    3. Show    4. Minimize    5. Maximize 3. User says/types/clicks application name    1. If user is not using voice activation, a list of applications will be displayed |
| **Preconditions** | * Acolyte is running on the system * Application is capable of performing operation specified by user * Application invoked by user exists on system |
| **Postconditions** | Application is opened, closed, minimized, or maximized. If user said “show,” application statistics like CPU usage are displayed |



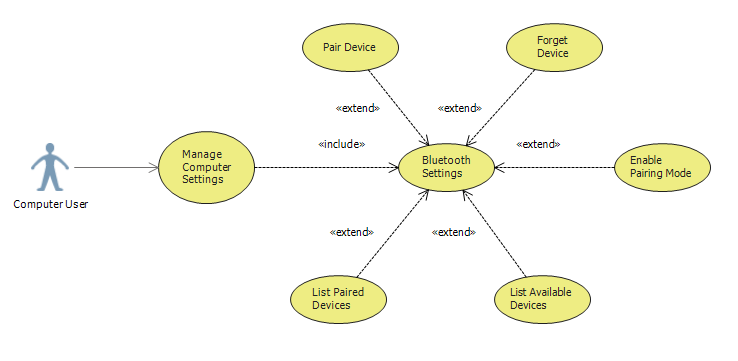
### Figure 1-4 - Audio Settings

|  |  |
| --- | --- |
| **Use Case #** |  |
| **Name** | Audio Settings |
| **Actors** | Computer User |
| **Description** | User can utilize Acolyte to control computer volume or peripheral audio device audio. This includes internal speakers, WiFi, Bluetooth, or plugged in speakers, and microphone volume (mic sensitivity). User can also mute speakers or deactivate microphone. If user says “turn up/down” there is a brief moment in which the user can say “more” to repeat the action on the same device. |
| **Workflow** | 1. User activates Acolyte (voice, keyboard shortcut, or clicking application icon) 2. User says/types/clicks on operation    1. Turn up       1. More    2. Turn down       1. More    3. Mute 3. User says/types/clicks device on which action is to be performed |
| **Preconditions** | * Acolyte is running on the system * Device specified by user is connected to the PC on which Acolyte is running * Device is capable of performing specified operation |
| **Postconditions** | Device volume is increased, decreased, or muted as requested by the user |



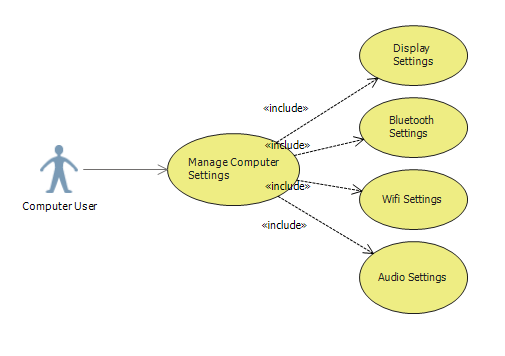
### Figure 1-5 - Display Settings

|  |  |
| --- | --- |
| **Use Case #** |  |
| **Name** | Display Settings |
| **Actors** | Computer User |
| **Description** | User may invoke acolyte to configure display settings such as brightness, contrast, resolution, etc. If user says “Increase/decrease” there will be a brief moment in which the user can say “more” to repeat the operation on the same setting. |
| **Workflow** | 1. User activates Acolyte (voice, keyboard shortcut, or clicking application icon) 2. User says/types/clicks operation    1. Increase       1. More    2. Decrease       1. More 3. User says/types/clicks setting on which operation is to be performed |
| **Preconditions** | * Acolyte is running on the system * Display settings can be configured via the computer |
| **Postconditions** | Display settings are changed according to user’s request |



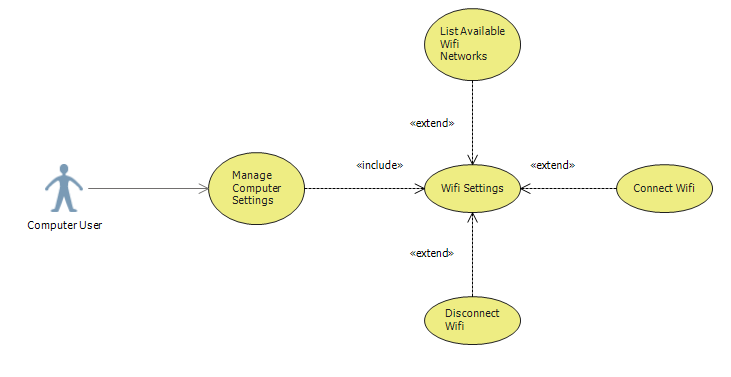
### Figure 1-6 - Manage Bluetooth

|  |  |
| --- | --- |
| **Use Case #** |  |
| **Name** | Manage Bluetooth |
| **Actors** | Computer User |
| **Description** | User can utilize Acolyte to perform Bluetooth management operations. The program will be able to tell the PC to pair with and forget a device, turn pairing mode on and off, and list devices available and paired |
| **Workflow** | 1. User activates Acolyte (voice, keyboard shortcut, or clicking application icon) 2. User says/types/clicks operation    1. Enable Bluetooth    2. Disable Bluetooth    3. List available devices    4. List paired devices    5. Pair with    6. Forget 3. If the user said “Pair with” or “forget,” the user will then specify the device in which the operation is to be performed |
| **Preconditions** | * Acolyte is running on the system * Computer is able to perform Bluetooth operations |
| **Postconditions** | The operation completes and alerts the user either by showing the requested list, or confirming the specified device has been paired with or forgotten |



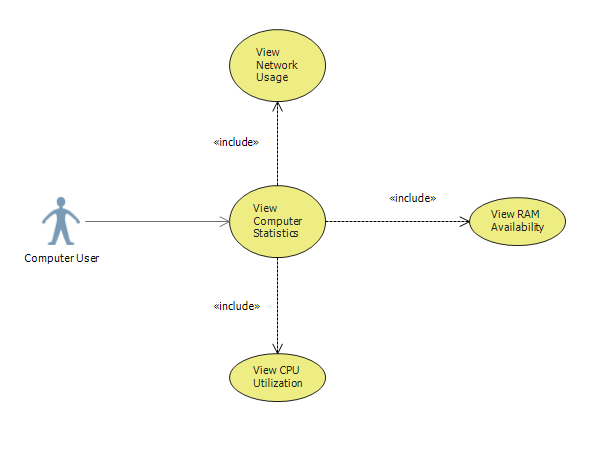
### Figure 1-7 - Manage Computer Settings

|  |  |
| --- | --- |
| **Use Case #** |  |
| **Name** | Manage Computer Settings |
| **Actors** | Computer User |
| **Description** | User can invoke Acolyte to change several settings in the system including WiFi, Audio, Bluetooth, and Display. Each of these have more specific use cases, but this use case is for listing the status of these settings and turning them on or off |
| **Workflow** | 1. User activates Acolyte (voice, keyboard shortcut, or clicking application icon) 2. User says/types/clicks operation    1. Activate    2. Deactivate    3. List 3. User specifies which settings to perform the operation on    1. Audio    2. Bluetooth    3. Display    4. WiFi |
| **Preconditions** | * Acolyte is running on the system * Settings specified by user are present on PC * The specified operation can be performed on the settings listed (for example, you cannot “deactivate display”) |
| **Postconditions** | Acolyte either turns on, turns off, or lists the current status of the specified settings |



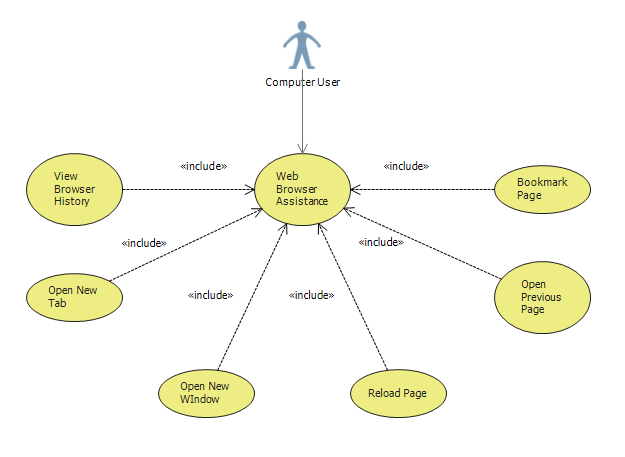
### Figure 1-8 - Manage WiFi Settings

|  |  |
| --- | --- |
| **Use Case #** |  |
| **Name** | Manage WiFi Settings |
| **Actors** | Computer User |
| **Description** | Acolyte can be used to handle WiFi and network connectivity operations, similar to what happens when the network symbol in the bottom right of a Windows computer is clicked |
| **Workflow** | 1. User activates Acolyte (voice, keyboard shortcut, or clicking application icon) 2. User says/types/clicks operation    1. List Networks    2. Connect to    3. Disconnect 3. If a network is being connected to and requires log in, the system will display that network’s login dialogue |
| **Preconditions** | * Acolyte is running on the system * The system has networking capabilities |
| **Postconditions** | The PC is connected to the specified network, or Acolyte lists the available devices  Acolyte displays login prompt if necessary |



### Figure 1-9 - View Computer Statistics

|  |  |
| --- | --- |
| **Use Case #** |  |
| **Name** | View Computer Statistics |
| **Actors** | Computer User |
| **Description** | User may ask Acolyte to retrieve system information about the system, equivalent to the “Performance” tab on the task manager in Windows. |
| **Workflow** | 1. User activates Acolyte (voice, keyboard shortcut, or clicking application icon) 2. User says/types/clicks statistic to be queried    1. Show CPU    2. Show Memory    3. Show Disk    4. Show Ethernet    5. Show WiFi    6. Show Bluetooth    7. Show GPU 3. User may optionally specify a specific statistic to be queried    1. In Use    2. Available    3. Speed    4. Committed    5. Utilization    6. Speed    7. Threads    8. Processes    9. Up Time    10. etc |
| **Preconditions** | * Acolyte is running on the system * The statistic queried has the data point asked for (for example, CPU does not have a “Paged Pool” attribute) |
| **Postconditions** | Acolyte informs user of data queried via voice or text |

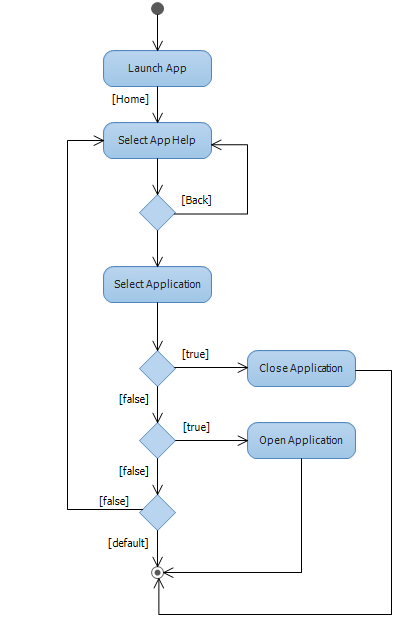


### Figure 1-10 - Web Browser Assistance

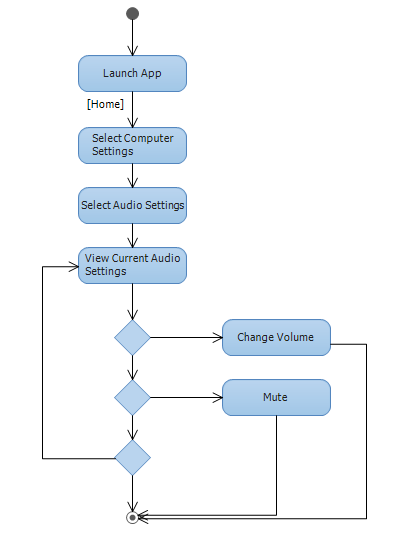
|  |  |
| --- | --- |
| **Use Case #** |  |
| **Name** | Web Browser Assistance |
| **Actors** | Computer User |
| **Description** | Acolyte may be used to perform several actions against a web browser. By default, Acolyte should use the PC’s default browser and be able to do all operations regardless of which browser is being used. User can open a new window or new tab, reload a page, go to the previous page, bookmark a page, or view history |
| **Workflow** | 1. User activates Acolyte (voice, keyboard shortcut, or clicking application icon) 2. User says/types/clicks operation    1. Open       1. Tab       2. Window    2. Go back    3. Refresh    4. Bookmark    5. Show browser history |
| **Preconditions** | * Acolyte is running on the system * A web browser is installed on the system * Web browser to be used can integrate with Acolyte * Operations invoked can be performed (for example, user cannot open previous page if they have just started a session) |
| **Postconditions** | Browser operation is performed. There will be no audible feedback for any of these operations, just the completion of the requested task |

# 

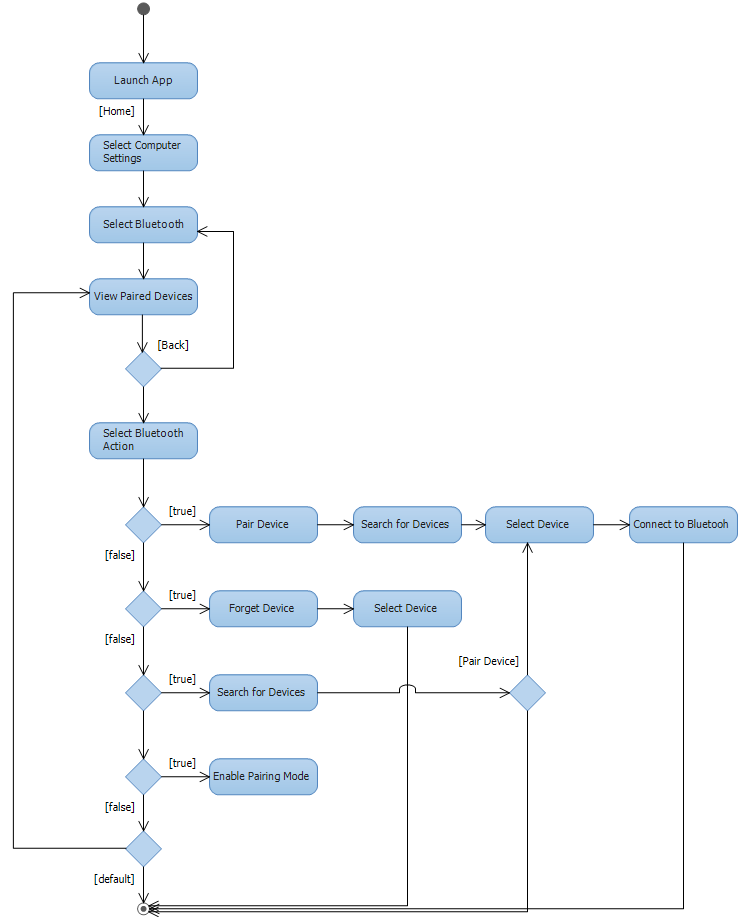
# Activity Diagrams



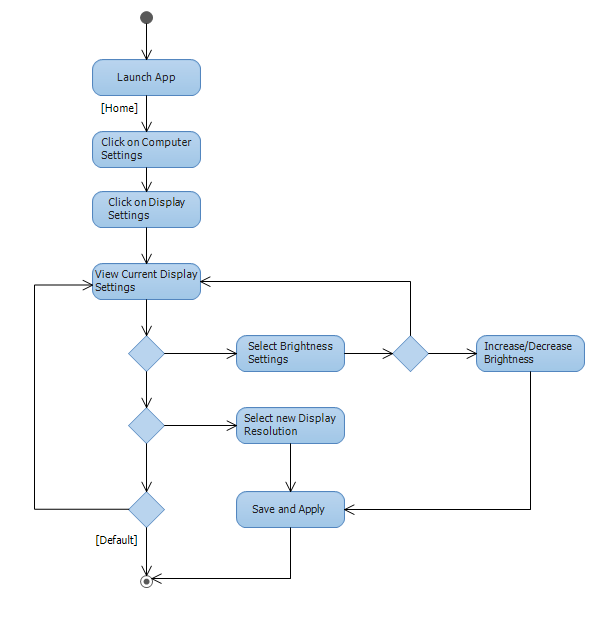
### Figure 2-1 Application Assistance Activity Diagram



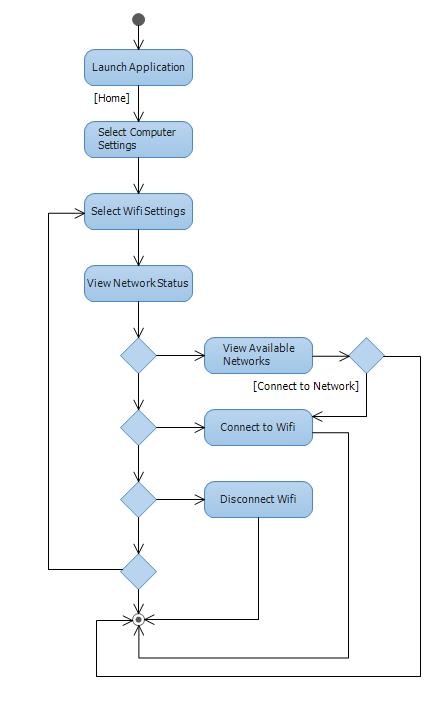
### Figure 2-2 Audio Settings Activity Diagram



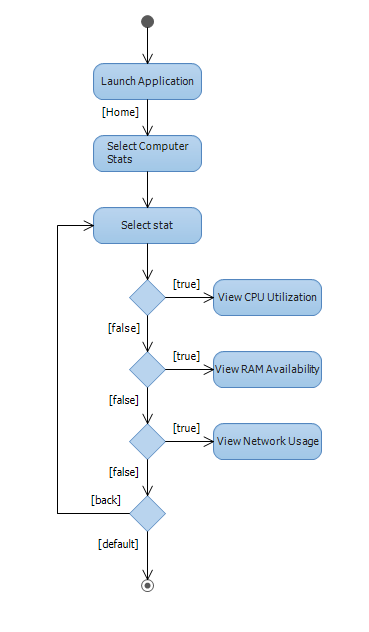
### Figure 2-3 Manage Bluetooth Activity Diagram



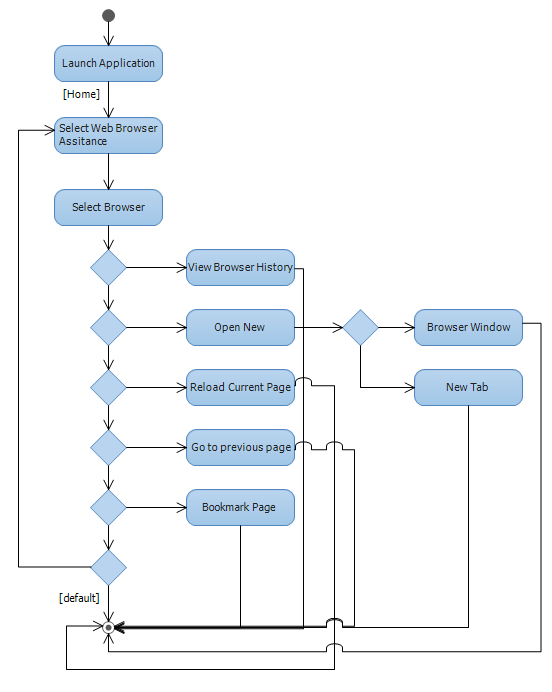
### Figure 2-4 Manage Display Settings Activity Diagram



### Figure 2-5 Manage WiFi Activity Diagram



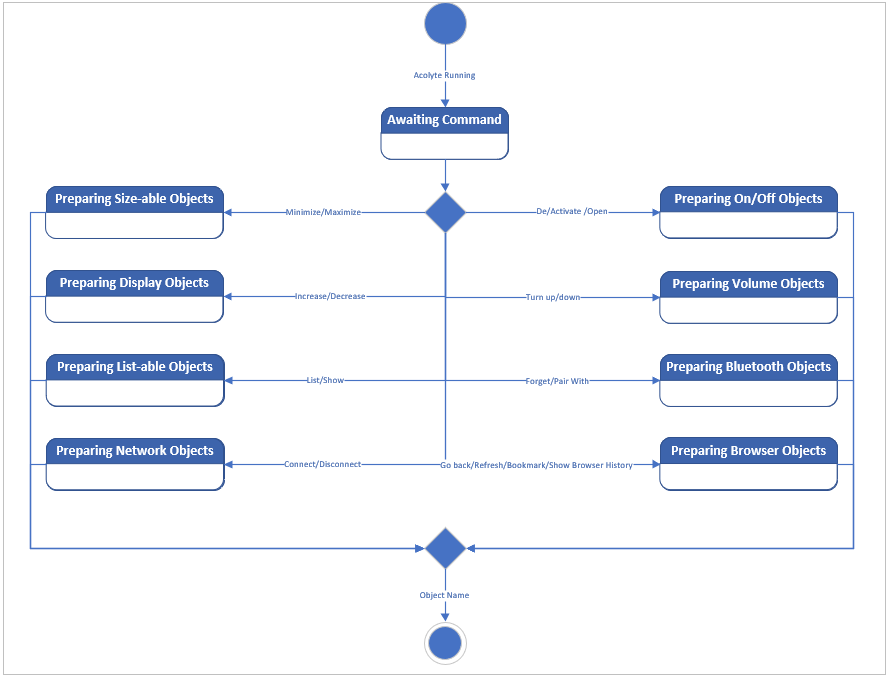
### Figure 2-6 View Computer Statistics Activity Diagram



### Figure 2-7 Web Browser Assistance Activity Diagram

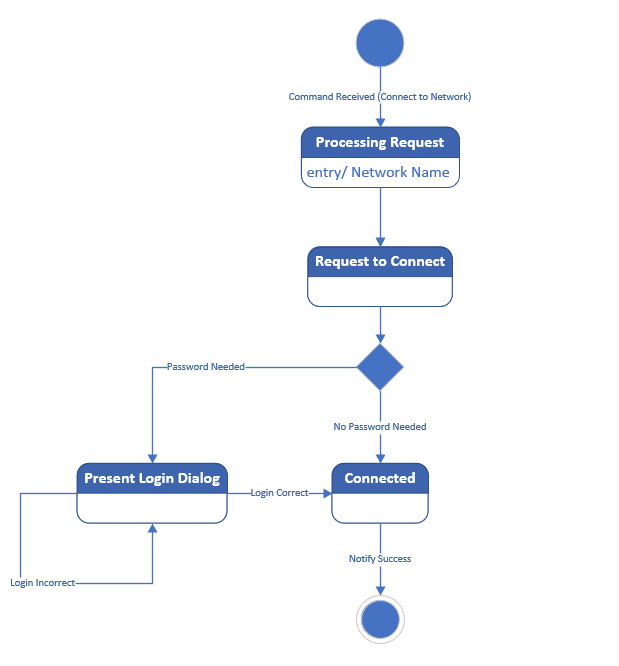
# 

# State Diagrams

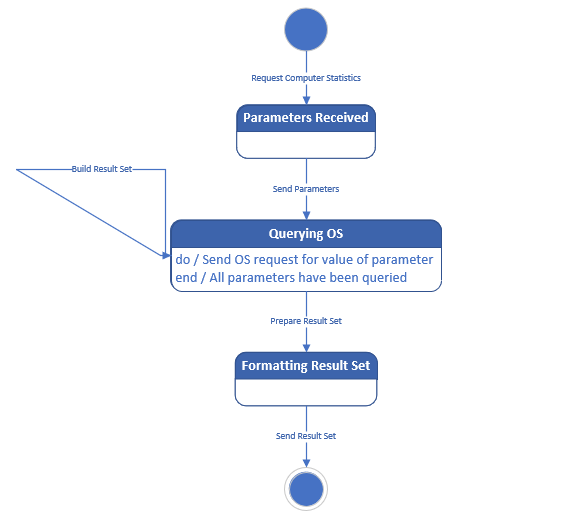


### Figure 3-1 High Level State Diagram

### 



### Figure 3-2 Connect to WiFi



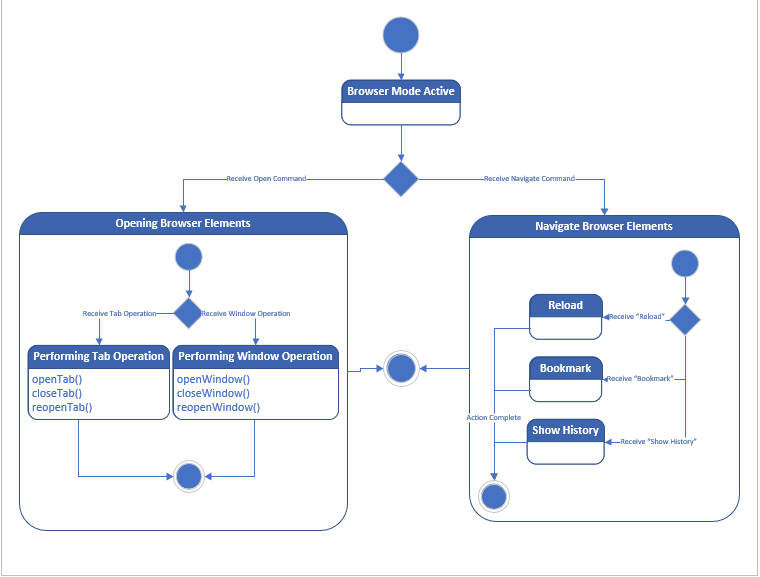
### Figure 3-3 Get Computer Statistics

### 

### 

### 

### Figure 3-4 Increase or Decrease Volume

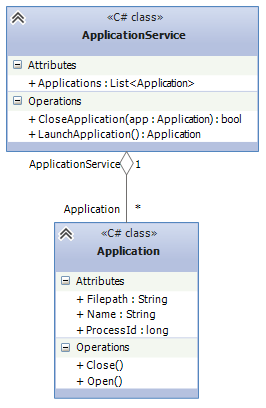


### Figure 3-5 Browser Use-Cases State Diagram

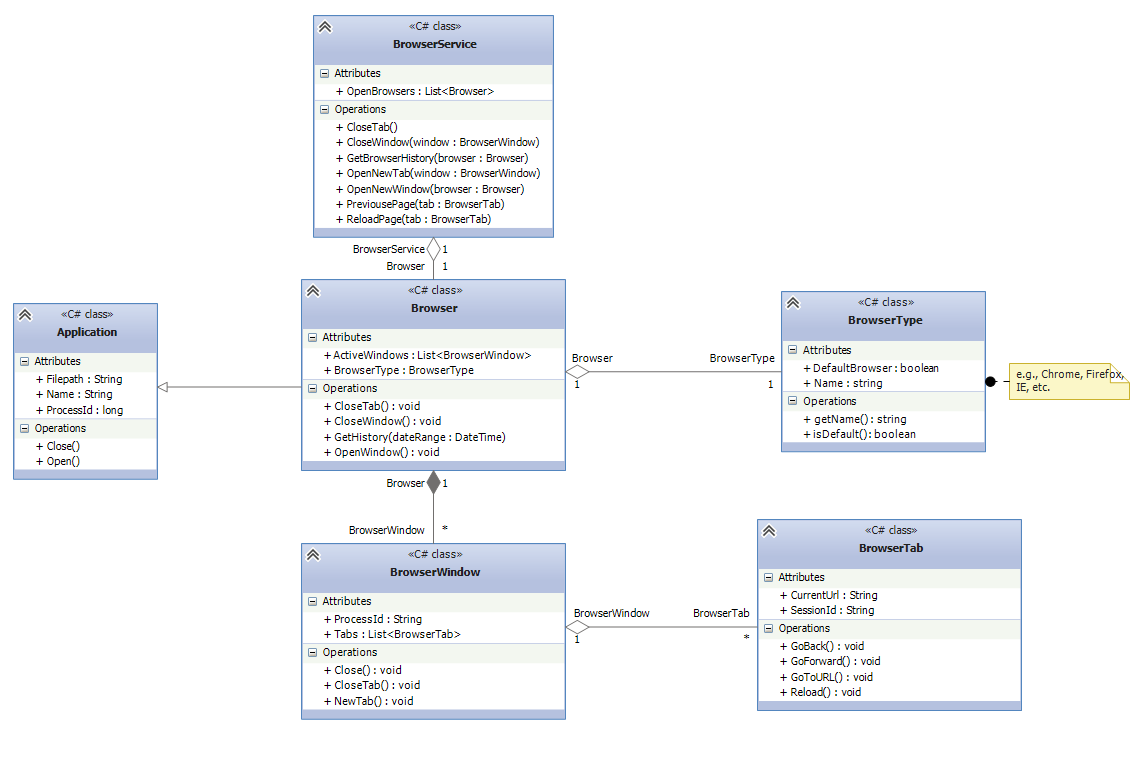
### 

### 

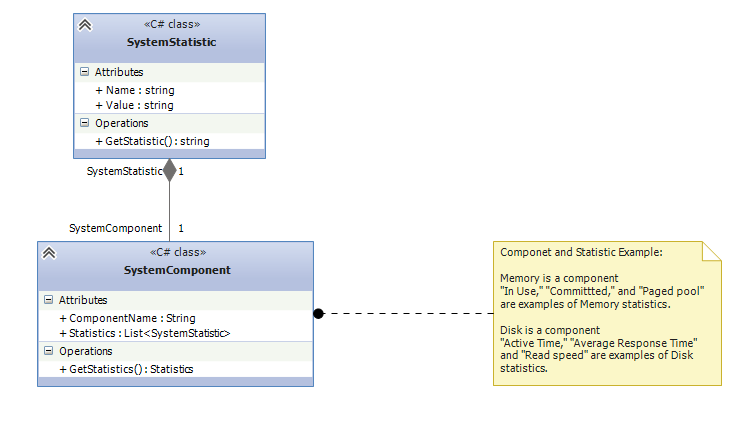
# Class Diagrams



### Figure 4-1 Application Class

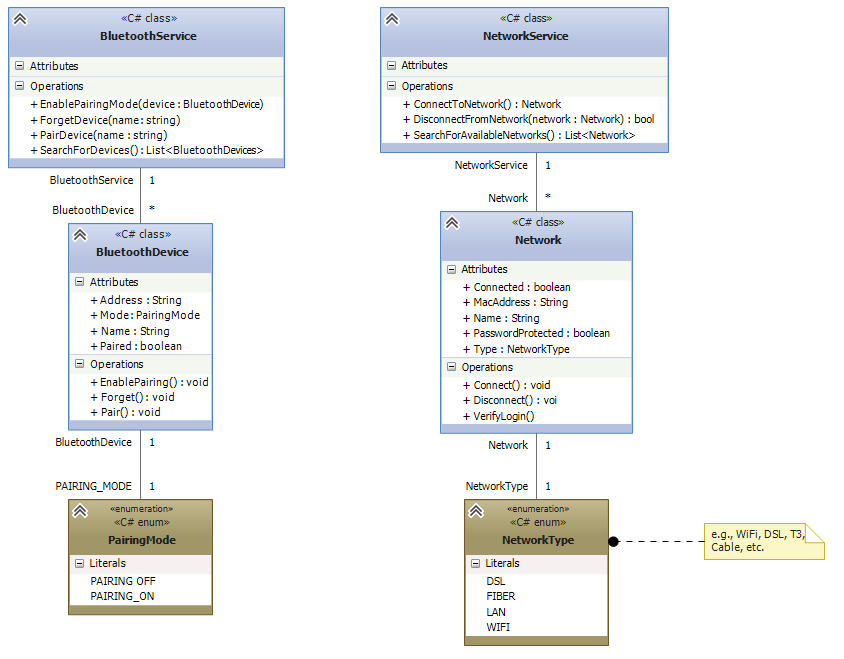


### Figure 4-2 Browser Classes

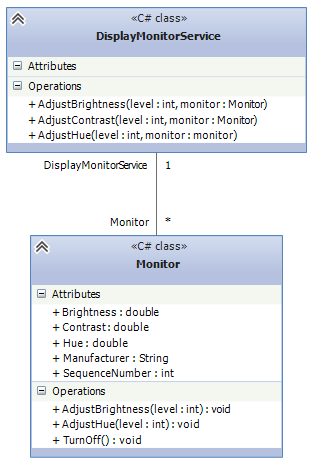


### Figure 4-3 System Statistic classes

### 

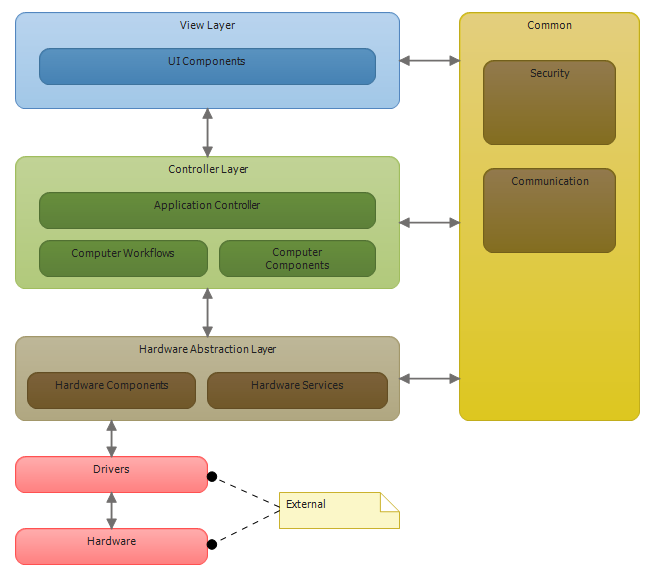


### Figure 4-4 Devices Classes



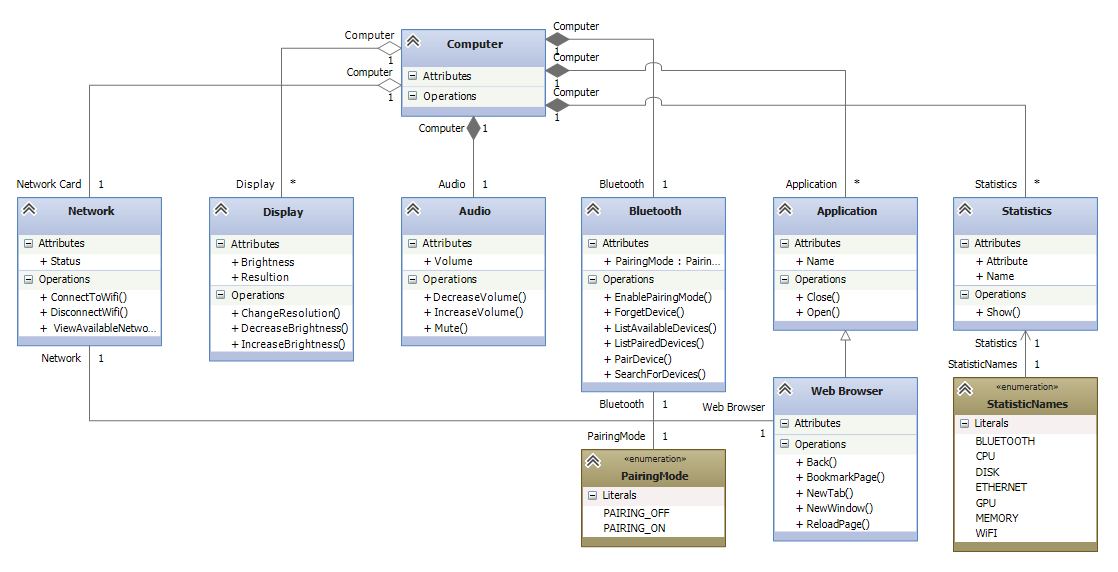
### Figure 4-5 Monitor Classes

# Layer Diagrams



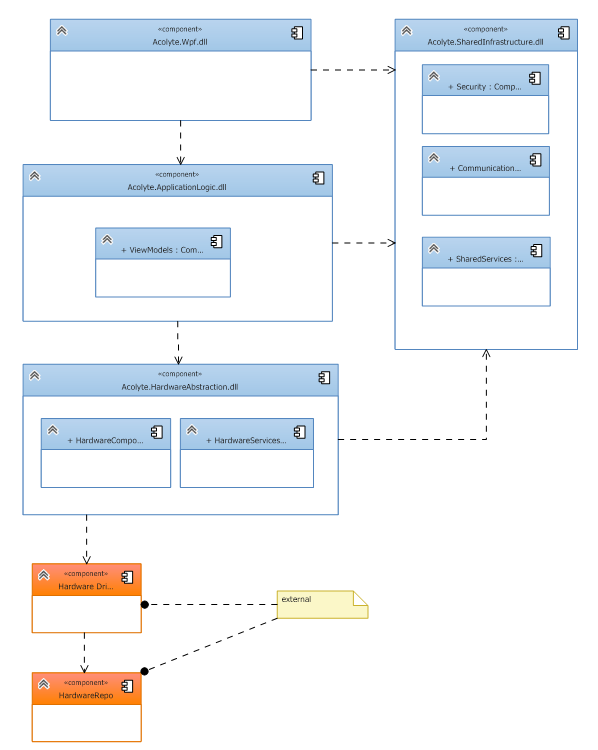
### Figure 5-1 Acolyte Layer Diagram

# Domain Model



### Figure 6-1 Domain Model

# Component Diagram



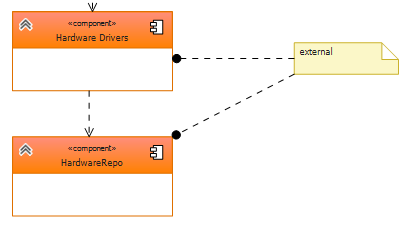
### Figure 7-1 Component Diagram

## Components

Acolyte will utilize four developed components and two external components. Each component is a layer of abstraction that is eventually utilized by the Acolyte Service component. The Acolyte WPF component is what the user will interface with exclusively and exposes the entire functionality of the system.

### External Components

These components represent the actual hardware and drivers that run on the computer in which Acolyte is running. They are used by the Hardware Abstraction component to expose the necessary functionality for the system to achieve its purpose.



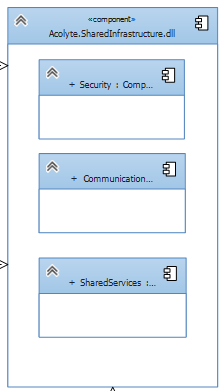
#### Figure 7-2 External Components

### 

### 

### Acolyte.SharedInfrastructure.dll

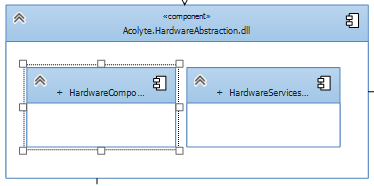
The Shared Infrastructure component is a library for abstracting the system’s Windows Services, communication functions, and security settings. This component is also utilized by the Hardware Abstraction component to allow Acolyte to perform its designed functionality.



#### Figure 7-3 Shared Infrastructure Component

### Acolyte.HardwareAbstraction.dll

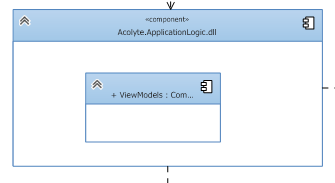
The Hardware Abstraction component contains collections of abstracted hardware entities. The entities can only be derived from the system in which Acolyte is running. The component includes interfaces for manipulating the system’s hardware components



#### Figure 7-4 Hardware Abstraction Component

### Acolyte.ApplicationLogic.dll

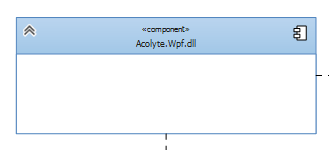
The Application Logic component receives user input objects from the WPF component (more details in next section) and routes the objects to the appropriate components. From the input objects, the Application Logic component deciphers what operation is to be executed and on what system entity. This layer will then receive the response from running those functions and, if applicable, will return the response to the WPF component. In some cases, it is not applicable to show the user a response, but the other components will always return at least a boolean indicating function success or failure. In cases where a user needs to see the response, such as listing available WiFi networks, the Application Logic component will build that data, but not format it in a user friendly way.



#### Figure 7-5 Application Logic Component

### Acolyte.Wpf.dll

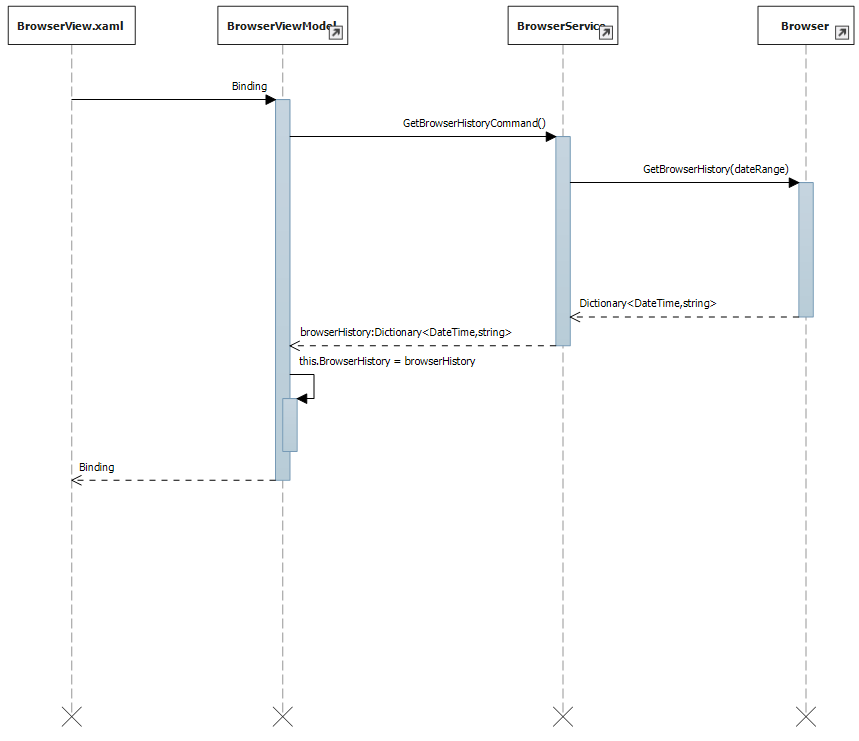
Finally, the main component of the system, the Acolyte WPF component, is what runs in the background as a Windows service and handles all user interaction. This is the main hub of the Acolyte application and is what makes all the other components useful. In this component live the functions for gathering user input and routing it to the Application Logic component. It translates user input into objects that will allow the Application Logic to decipher what the system should do and routes those objects to the logic component. In cases where the user will expect a visual response, like a list of available WiFi networks, this layer will take the data returned by the Application Logic component and format it in a user friendly way.



#### Figure 7-6 Acolyte WPF Component

# 

# Sequence Diagram



### Figure 8-1 Sequence Diagram

# How to Use the Acolyte Prototype

To use the prototype, first clone the application from the Github repository linked at the top of this document. After the project has been cloned, build it and run the resulting executable. Currently, the prototype simply shows one piece of planned functionality: listing all running applications.

The prototype is a console application. When you run the executable, you will be prompted to enter the letter “a.” Doing so will list all applications currently running on your system in the console.

The prototype accomplishes this by implementing an Application class built from information from in the .NET class “Process” from the System.Diagnostics assembly. Once an Application class has been instantiated, it adds the object to a List variable. Finally, the prototype loops through the list and displays the Application’s name attribute, pauses for half a second, and continues.

# Activity Log

Individual’s efforts can be found both on the Github commit history and in the revision history of this document. Erik also exclusively worked on the powerpoint and documents while Adriana made edits.

### Revision History

See page 1.

### Commit History

$ git log --date=local --pretty=format:"%h%x09%an%x09%ad%x09%s"

1579ebf erik-whiting Sat Mar 23 11:33:17 2019 Cleaned up code, renamed some things to make usage obvious

1f93190 erik-whiting Sat Mar 23 11:17:54 2019 Update AcolytePresentation.pptx

a56cb7b Adriana Arzola Sat Mar 23 11:04:38 2019 Add Sequence Diagram for Browser History

5a07c58 Adriana Arzola Sat Mar 23 10:37:34 2019 UpdateTypos

270aed8 Adriana Arzola Sat Mar 23 10:33:00 2019 Remove component from component diagram

d73dedf Adriana Arzola Sat Mar 23 10:30:32 2019 Overall Acolyte Class Diagram and Changes to Smaller Diagrams

6c90bd1 Adriana Arzola Sat Mar 23 00:30:07 2019 Merge branch 'master' of https://github.com/swen5232/Final.git

d5cbe83 Adriana Arzola Sat Mar 23 00:29:42 2019 Changes to Layer & Component Diagrams

50a802a erik-whiting Sat Mar 23 00:26:13 2019 Create AcolytePresentation.pptx

d66672d erik-whiting Fri Mar 22 22:01:27 2019 console protoype application working

b6f3034 erik-whiting Fri Mar 22 20:55:22 2019 Added Entities project with generated models plus some cleanup

63a4aa2 erik-whiting Fri Mar 22 20:27:16 2019 More class diagrams

34b0c7a erik-whiting Thu Mar 21 23:36:47 2019 Update Acolyte.AnalysisModel.modelproj

ff8c291 erik-whiting Thu Mar 21 23:34:20 2019 Added browser class diagram

eb7add3 erik-whiting Thu Mar 21 22:37:42 2019 Component diagram

fb10c3c erik-whiting Thu Mar 21 21:44:09 2019 Devices Class diagram

ef7d528 erik-whiting Thu Mar 21 21:25:43 2019 Initial Class diagram for computer statistics

20ab1d6 Adriana Arzola Thu Mar 21 21:06:34 2019 Add architecture to Layer Diagram

c97120d erik-whiting Thu Mar 21 20:03:03 2019 More state diagrams

70b0df4 erik-whiting Thu Mar 21 19:39:32 2019 Added state diagrams

7130c83 Adriana Arzola Thu Mar 21 16:54:28 2019 Remove Duplicate

750712f Adriana Arzola Thu Mar 21 16:51:59 2019 Update class diagram and blank layer diagram

33e7c7e Adriana Arzola Thu Mar 21 16:48:52 2019 Update State Diagram location

ad3aa09 erik-whiting Thu Mar 21 00:11:55 2019 Added high-level state diagram

69aaebd erik-whiting Wed Mar 20 23:33:26 2019 Added on table to DomainModel

1f2f7ac SOLARISOILFIELD\adriana.arzola Wed Mar 20 17:00:49 2019 First Pass DomainModel

94693b7 SOLARISOILFIELD\adriana.arzola Wed Mar 20 16:47:53 2019 Update folder Structure to split between both projects

9051bae SOLARISOILFIELD\adriana.arzola Wed Mar 20 16:36:47 2019 Restructure to add folder for domain models

f23c9d9 SOLARISOILFIELD\adriana.arzola Wed Mar 20 16:31:20 2019 Add missing activity diagrams for audio and display settings

b145c6a SOLARISOILFIELD\adriana.arzola Wed Mar 20 16:13:21 2019 Added use cases for audio and display settings

156e90e SOLARISOILFIELD\adriana.arzola Wed Mar 20 15:49:06 2019 Activity Diagrams

4ccce28 erik-whiting Mon Mar 18 20:09:43 2019 Made individual use cases

ebedfde erik-whiting Mon Mar 18 19:30:32 2019 Merge branch 'master' of https://github.com/swen5232/Final

f8871e1 erik-whiting Mon Mar 18 19:30:28 2019 Structuring

481e6c2 erik-whiting Mon Mar 18 19:04:39 2019 Update README.md

0b2c34d SOLARISOILFIELD\adriana.arzola Mon Mar 18 15:02:16 2019 Use Case Diagram

b5db374 SOLARISOILFIELD\adriana.arzola Mon Mar 18 14:31:55 2019 Add Brainstorm

05c5ec0 SOLARISOILFIELD\adriana.arzola Mon Mar 18 14:16:46 2019 Remove duplicate project

432161f SOLARISOILFIELD\adriana.arzola Mon Mar 18 14:10:17 2019 Add Analysis Model & Design Model Project & Update Read me

897c3d5 adrycontreras23 Mon Mar 18 13:45:49 2019 Initial commit