

Term Project 3

Miguel Contreras
Brayan Rivera

ABSTRACT

In the first term project report and second report, activity diagrams, class diagrams and swimlane diagrams were provided. With those diagrams we were able to develop and provide use cases, sequence diagrams, and updated class diagrams. These provide a lower level of detail; low enough so the next step can be to create State Diagram, user interfaces and pseudocode.

This report will give us an overview of how the user will interact with the system and show a graphical design of communication between the tour guide and the visitor.

INTRODUCTION

The purpose of this term project is to design a handheld device for FAU tours using the top-down design methodology. In order to ensure a correct top-down design we make use of UML 2.0. In the previous term project reports we presented the activity diagrams, class diagrams, swimlane diagrams, use cases and sequence diagrams. In this report we present updated and more detailed versions of the sequence diagrams, state diagrams which will be used for the creation of Graphical user interfaces. Pseudocode will be developed using the updated versions of the sequence diagrams for this report.

METHOD

After discussing the sequence diagrams created for our tour guide handheld device, it was necessary to update them, covering more possible scenarios between the visitor, the tour guide and the tour manager. After the update of the sequence diagrams, the next step was to create the state diagram for the system. The development of the state diagram is an iteration process which will ultimately lead us to the Graphical user interfaces, where we will see how the system will display through the eyes of the user.

Simultaneously after updating the sequence diagrams we design some pseudocode in order to design a base for later software development for our system.

RESULTS

The updated version of the sequence diagrams gave us a more thorough view of the system interaction and showed us how processes operate with one another and in what order.

The state diagrams provided us with a description of the behavior of the system. The state diagrams helped in visualizing what the elements of the complete system should be.

By referring to the user cases given in previous report we were able as designers to identify the requirements of the system and study its high level functionality.

DISCUSSION

The iteration process of sequence diagrams to pseudocode and the design of graphical user interfaces gave us an overview of the high functionality of the system. This process gave us a better look on the interaction within the stakeholders involved in our system.

The GUI diagrams were developed by using the sequence diagrams and state diagrams and gave us a better understating of the requirements and functionality of the system, but as the top-down process design continues, there will come the need to revise and update the GUI diagrams given in this report, in order to give a more detailed view of the system.

CONCLUSION

The use of the sequence diagrams and state diagrams gave us a leap forward to a better understanding of the requirements and functionality of the system from the inside and outside. Pseudocode developed by the use of sequence diagrams can act as a tool later on for software implementation and development.

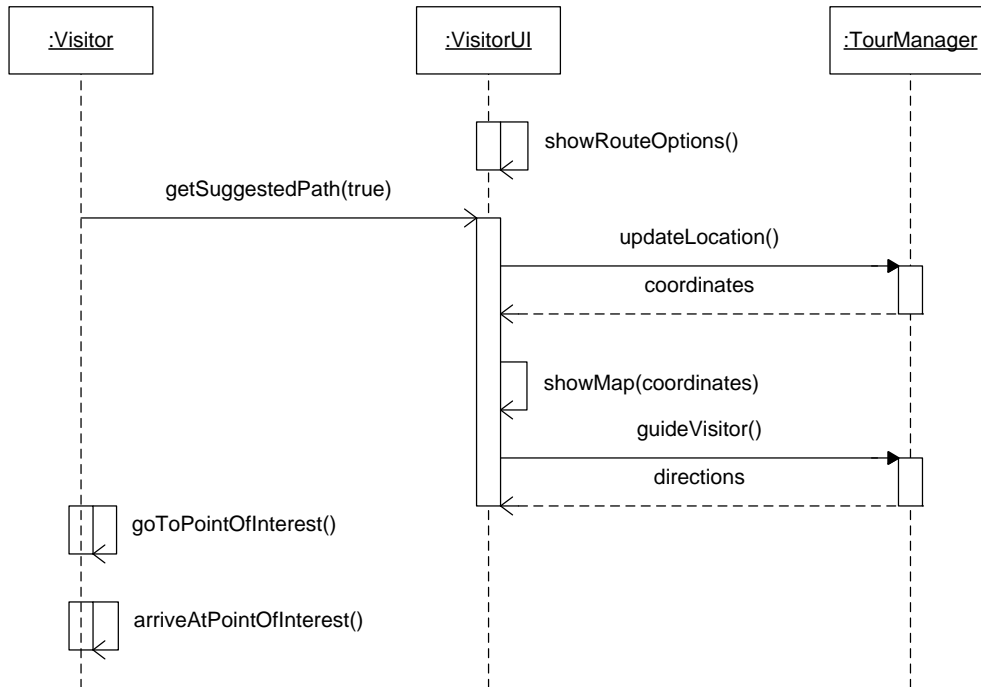
The graphical user interface design offered us a way to fully represent the information and actions available to the visitor as he moves from one point of interest to another.

REFERENCE

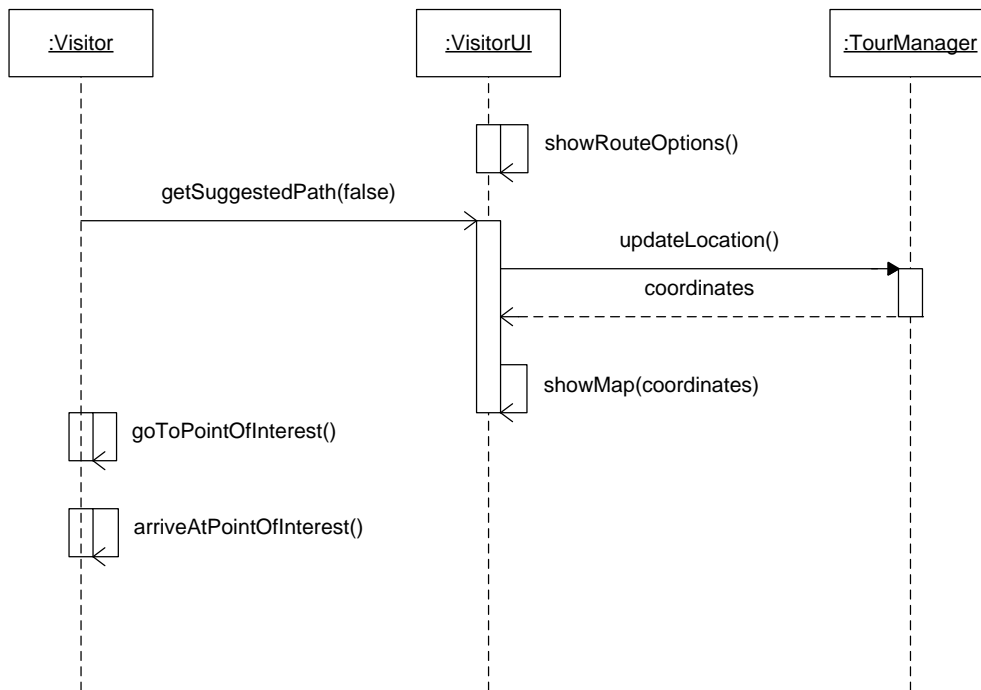
Contreras-Rivera_TermProject2_041609
Sams Teach Yourself UML in 24 Hours 2nd Edition.

APPENDIX A – Sequence Diagrams

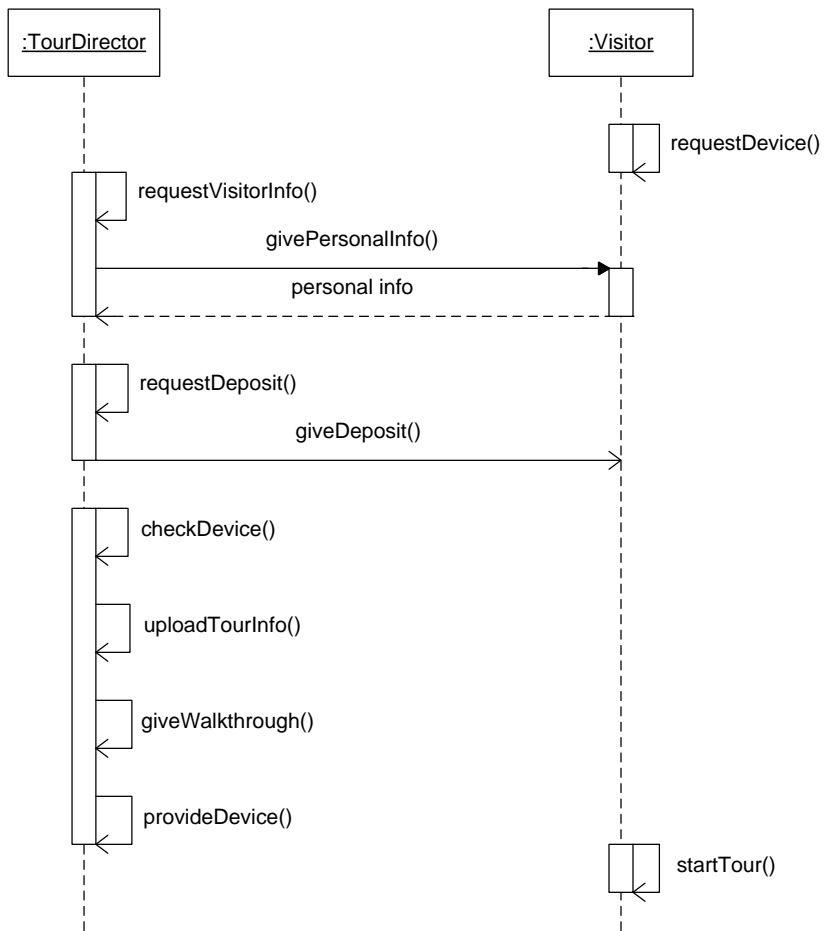
Visitor takes suggested path



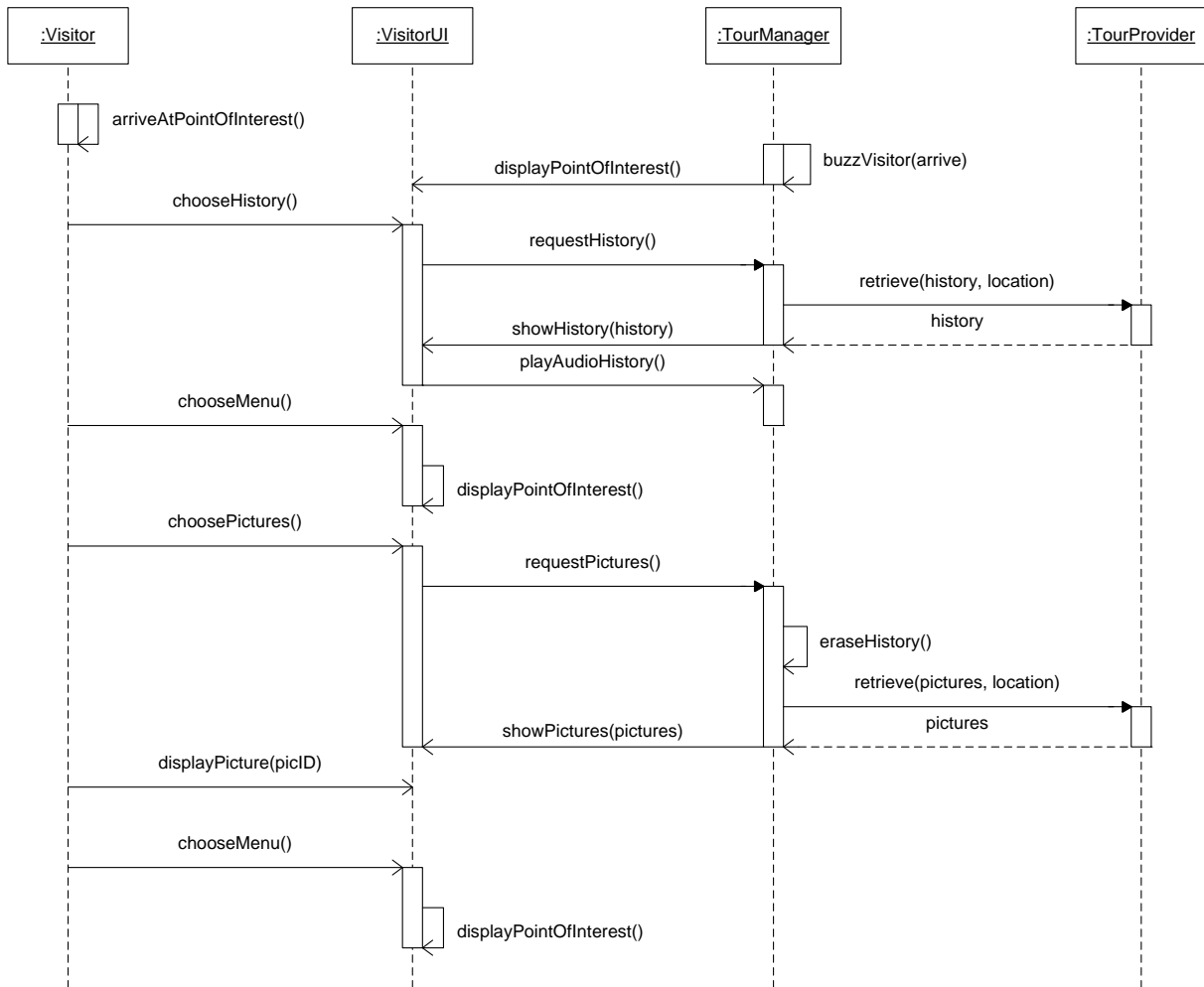
Visitor takes own path



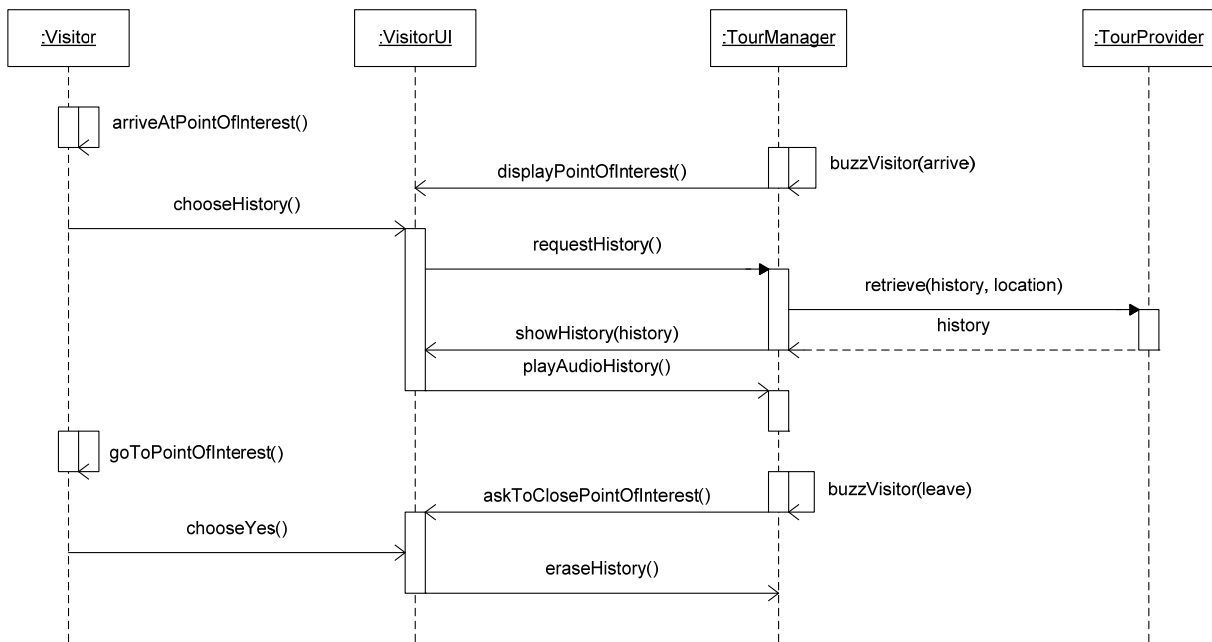
Provision device



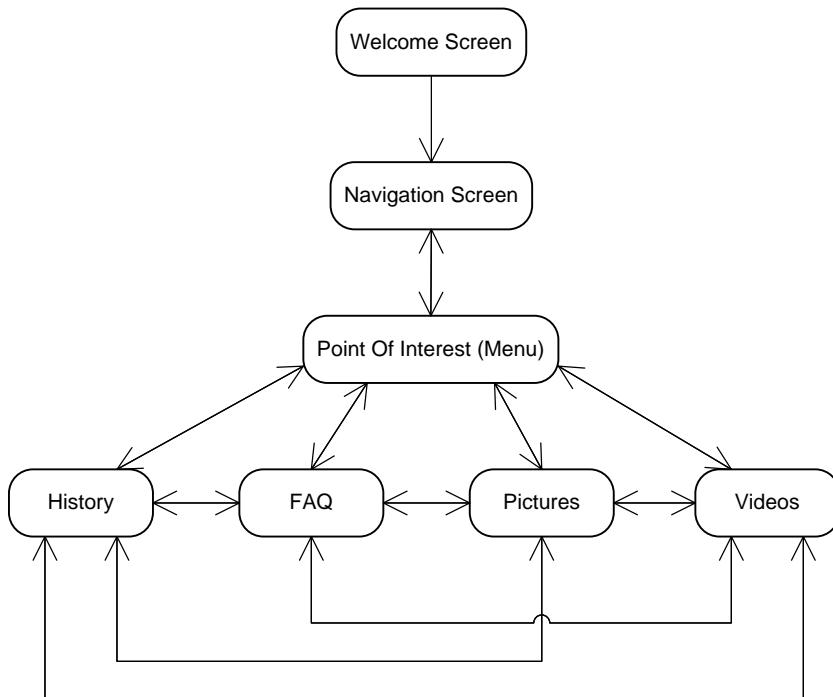
Requesting Data



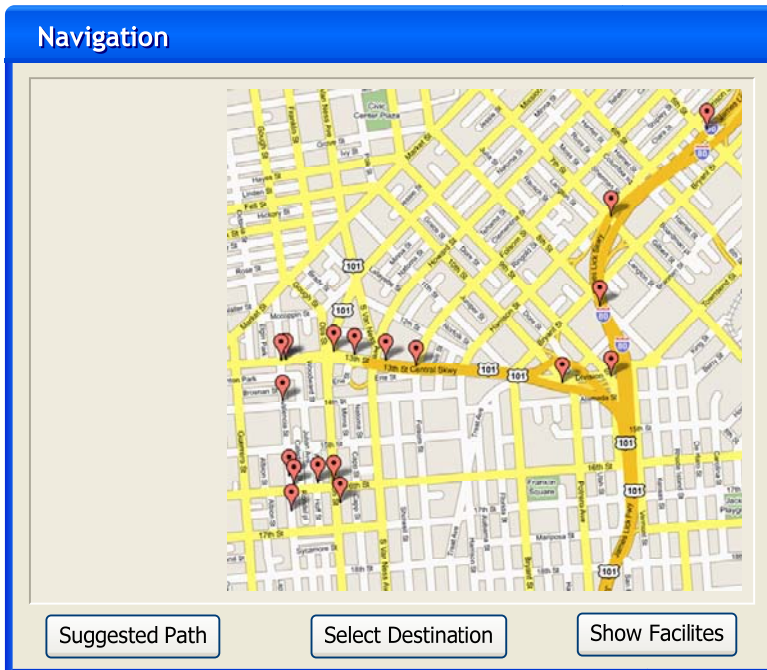
Visitor walks away while viewing history



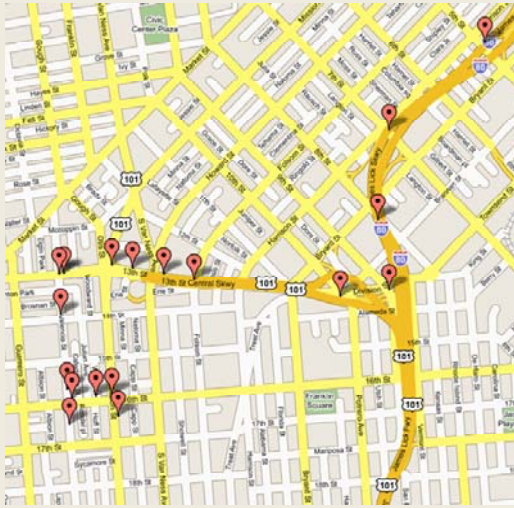
APPENDIX B – State Diagrams



APPENDIX C – GUI Diagrams



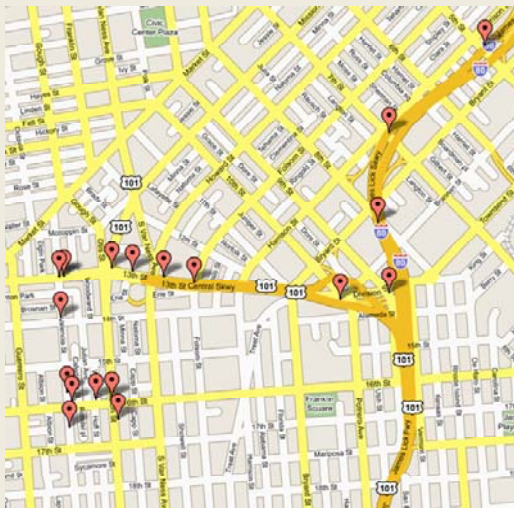
Navigation



Choose Own Path

Show Facilities

Navigation



History

FAQ

Pictures

Videos

Continue Tour

Pictures

History

FAQ

Videos




Continue Tour

Picture

History

FAQ

Videos



CL-MM03

Back to pictures

Continue Tour

Videos

History

FAQ

Pictures



xbox media center Now Playing - Music

View List	01. Sigur Rós - Untitled 1	6:38
Shuffle	Nirvana - In Bloom.mp3	4:13
Repeat: Off	05. Radiohead - Go To Sleep	3:30
Save	10. The Black Crowes - Time Will Tell	4:06
Clear	11. John Fogerty - Centerfield	3:54
	19. The Clash - Train in Vain (Stand by Me)	3:10
	03. The Black Crowes - Thorn In My Pride	6:03
	06. U2 - Red Hill Mining Town	4:54
	17. Echo & The Bunnymen - Lips Like Sugar	4:52
	08. The Sugarbabes - Deus	4:08

00:07
The Clash | Train in Vain

10 Objects

Continue Tour

Video

History

FAQ

Pictures



E.M. TOTAL VIDEO PLAYER

File View Play DVD Help

00:25

Back to Videos

Continue Tour

FAQ

History

Pictures

FAQ



Continue Tour

History

FAQ

Pictures

Videos

Causes of Exploration

Print Glossary Help

Routes, Riches, and Religion

In summary, the causes of the European exploration included routes, riches, and religion. The terms on the left represent events, items, or people associated with these causes. Match each term with its cause.

Drag each term to the cause with which it is associated.

Terms	Religion	Routes	Riches
A Monks			
B Constantinople			
C Giovanni da Montecorvino			
D Eastern spices			
E Astrolabe			
F Caravel			
G Precious stones			
H Converts			
I Porcelain			

Page 10 of 11

Menu Back Next

Continue Tour

APPENDIX D –Pseudo code

- WelcomeScreen()

```
{
    ShowRouteOptions();
    If(GetSuggestedPath = True)
    {
        UpdateLocation();
        ShowMap(coordinates);
        GuideVisitor();
    }
    Else if(GetSuggestedPath = False)
    {
        UpdateLocation();
        ShowMap(coordinates);
        GuideVisitor();
    }
}
```
- DataRequest()

```
{
    BuzzVisitor();
    DisplayPointOfInterest();
    DisplayMenu();

    Do while(visitorLeaves = False)
    {
        If(VisitorRequest = History)
        {
            DisplayPointOfInterest();
            RetrieveHistory(history, location;
            ShowHistory(history);
            PlayAudioHistory();
        }
        Elseif(VisitorRequest = Picture)
        {
            DisplayPointOfInterest();
            RequestPictures();
            EraseHistory();
            Retrieve(pictures, location)
            ShowPictures(pictures)
        }
    }
    EraseHistory();
    ErasePictures();
}
```