

***Blue Way ©ROCK'N MOBILE***

***Design Specification***

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **VERSION** | **AUTHOR** | **CHANGE** | **DATE** |
| 1.0 | Burak Demirtaş | Template Release | 01.11.2008 |
| 1.1 | Doğan and Gökhan | Draft Relase 1 | 18.02.2009 |
| 01/02/09 | Doğan and Gökhan | Relase 2 | 27/02/09 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Table of Contents

Insert Table of contents here

## Description

\*Explain briefly the Design in generally like tools, general aim of the design etc..\*

There is a top down design approach. It is planned to find out the way by starting from the big picture and exploring the details. This document will explain the functionalities that are mentioned in product specification with more details.

We will use MsProject for design documentations and planing. Eclipse and Netbeans IDE’s will be used for implementing easily (coding and interface design).

## Functional Requirements

\*Insert all design macro blocks here and explain functionalities in design perspective with UML tools and case scenarios in 4 sections\*

Work Breakdown Structure

1. Capturing audio and video.
   1. Capturing audio and video on Linux PC side J2SE.

A multimedia capturing device can act as a source for multimedia data

delivery. For example, a microphone can capture raw audio input or a dig-

ital video capture board might deliver digital video from a camera. Such

capture devices are abstracted as DataSources. For example, a device that

provides timely delivery of data can be represented as a PushDataSource.

Any type of DataSource can be used as a capture DataSource: PushData-

Source, PushBufferDataSource, PullDataSource, or PullBufferDataSource.

Some devices deliver multiple data streams—for example, an audio/

video conferencing board might deliver both an audio and a video stream.

The corresponding DataSource can contain multiple SourceStreams that

map to the data streams provided by the device.

* + 1. Web cam usage

JMF(Java Media Framework) API is used to control media source such as web cam,microphone all other data sources. We need to grap the frame from cam source in a specified timely manner and prepare for packetizing to send it to remote device. This operation must be optimized according to bandwidth and time limitations of transmission.

* + 1. Audio device usage

This part of design is responsible for taking multiple

audio samples and distributing them into packets of a particular size that can be streamed over the underlying network. Audio samples are typically grouped together

* 1. Capturing audio and video on mobile symbian OS side J2ME.

We need to use the MMAPI (mobile media API) in J2ME for controlling media devices and processing audio and video.

There are two parts to multimedia processing:

* *Protocol Handling*: reading data from a source such as a file or a streaming server into a media-processing system.
* *Content Handling*: parsing or decoding the media data and rendering it to an output device such as an audio speaker or video display.

To facilitate these operations, the API provides two high-level object types:

* DataSource encapsulates protocol handling by hiding the details of how the data is read from its source. This object's utility methods enable the Player object to handle the content.
* Player reads the data from DataSource, processes it, and renders it to an output device. This object provides methods to control media playback, including methods for type-specific controls to access features for specific media types.

MMAPI specifies a third object, a factory mechanism known as the Manager, to enable your application to create Players from DataSources, and also from InputStreams.

* + 1. Cam usage

MMAPI includes support for a camera, with a special locator capture://video used to create its Player. An application can use the VideoControl to display a viewfinder on the screen, then take a picture using VideoControl.getSnapshot(String imageType). The default image format is PNG. You can use the imageType parameter to select any other supported format, and query the system property video.snapshot.encodings to find out what formats are supported.

* + 1. Audio device usage

The MMAPI can capture voice from microphone and has the same approach like video capture. "capture://audio": Allows capturing audio from the device microphone. We can define the audio format from the provided formats.

1. Compressing and decompressing with codecs.
   1. Compressing
   2. Decompressing
   3. Merging Audio and Video
2. Transmitting the data via Bluetooth.

3.1. Buffering

3.1.1 Video Buffering

3.1.2 Audio Buffering

3.2. Querying and queuing

3.3. Transmitting

1. Presenting the data.

4.1 Use the JMF to play the streamed media.

4.1.1.

4.2 Use the MMAPI to play the streamed media.

1. Graphical user interface.

5.1.

5.2.

### Input Requirements

Explain, all macro block and system input requirements for functionality #X \*

2.1.1 Requirements for mobile side

* + 1. Requirements for Pc side
       - The PC must have Suse Linux 10 or higher version.
       - There must be JRE(Java Runtime Environment higher than v1.6 ) installed.
       - A Camera device must be installed on system to capture video from real world.
       - A microphone must be exist on PC system to capture audio from environment.
       - A bluetooth device and a bluetooth stack on OS must be installed for the bluetooth transfer.
       - Another application which uses the bluetooth connection may reduce efficiency of the transsmission.
       - The position of PC should ideally be closer than 20m to get more efficency from bluetooth.
    2. Requirements for Mobile side
* The phone is operating by Symbian OS which support S60 development platform and feature pack 2.
* There must be exist the API's that providing capturing,streaming and transmitting.
* The phone must have Bluetooth 2.0 property.
* Midlet Applications can be run.

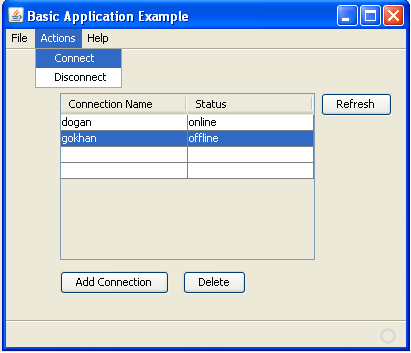
### User Interface Requirements

\* Explain, all macro block and system User Interface requirements for functionality #X \*

PC Interface :

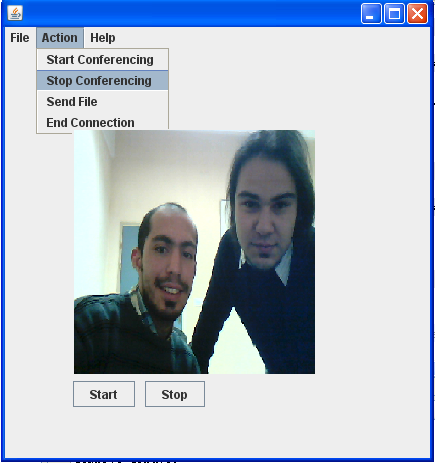
A ) Main window

1. The first window in which there is a list of connections which were available and connected before.
2. The user can choose one of them and set up a connection if the status of remote device is online. This is done either by double clicking the item in of the list of devices or by choosing the “Connect” item in the Action menu after selecting the specified connection from the list.
3. The refresh button is used to refresh the status of the devices.
4. If the device which the user wans to connect is not in the list the user clicks the “Add Connection” button and then a new window appears which allows the user to add a new connection.
5. The “Delete” button is used to delete a connection in the list.



B) Action Window

* + 1. This window is used for managing and controlling all actions. These are video conferencing, file transfer and capturing media.
    2. To start a video conference user has to choose the “Start conferencing ” item in the actions menu.
    3. To stop a running conference user has to choose “Stop conferencing” item in the actions menu.
    4. To send a file user has to choose the “Send file ” item in the actions menu and browse the file path in the appearing window and click send.
    5. To end connection user has to choose the “End connection” item in the action menu.



### Functionality

* + 1. Activity Diagram



* + 1. Explanations of Activity

During the capture stage, data is read from a video and audio source and passed in buffers to the processing stage.  The input stage may consist of reading data from a local capture device (such as a webcam) or a file on disk.

Processing stage consist of compressing with codecs and multiplexing with the merge function of the JMF. And the output of these operations is sent to buffer to transmit remote device .

Explain, all macro block and system functionality requirements for functionality #X with UML diagrams and so on

### Output Requirements

Explain, all macro block and system output requirements for functionality #X

## Abbreviations – References – Glossary

Explain All abbreviations in the document, list all references such as books web sites, Explain all the technical terms in the document that are not clear to the reader

J2SE: Java 2 Standard Edition

J2ME: Java 2 Micro Edition

WTK: Wireless Toolkit

**- 1 –**