

Mobile Voting System

Project Description

The Mobile Voting Software reliably tallies votes for candidates at a show. Each of the candidates show their exhibit with a poster, and voters cast their vote to decide who they thought had the best poster. An administrator can start a tally with a command message to the waiting server to begin the voting. Once the server is waiting, voters cast their vote by sending an email to the server with the candidate ID of the poster they thought was best. Periodically, the administrator can send command messages to the server to get a report back that shows the current rankings. Once the voting is over, the administrator for the system sends a command message to end the tally and return the results. The system sends back a report showing the candidate list in ranked order with the winner listed first, the first place runner up second, and so on. The administrator can also use a new component of the system that performs a trend analysis of this year's current voting results to obtain a prediction of next year's results. This Trend Analyzer can receive messages from the System Administrator just the same as the Voting Server, or XML files can be loaded to perform the same command structure.

Features of the System

The users of the system fall into two groups, voters and administrators. Voters have essentially one interaction with the system which is to cast a vote. The administrators of the system have 6 basic interactions with the system. Three of the interactions are with the Voting components of the system which are starting a tally, getting the current results of a tally, and ending a tally. The interactions with the Trend Analysis component of the system comprise the following commands: starting a trend analysis, stopping a trend analysis, and requesting a report that predicts next year's results. These interactions are illustrated below.

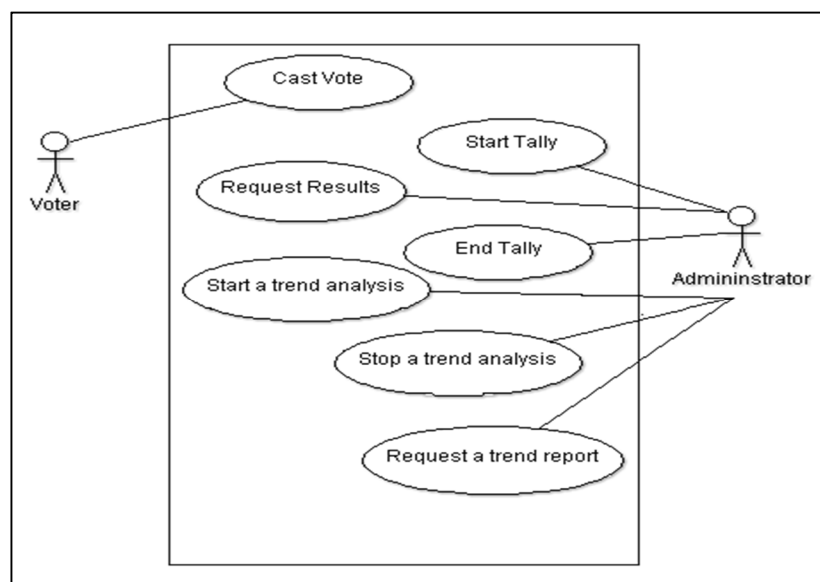


Figure 1. Use Case Scenario

Classes of the System

There are three base classes for the Voting component and three base classes for the Trend Analyzer in the system. In the Voting component, there are two objects which represent the users of the system, and another that represents the tally. The users of the system both send messages to Tally, and Tally can respond with either messages or reports. In the Trend Analysis component, the classes are mostly helper classes to the main class which is the Trend Estimator. This is illustrated below.

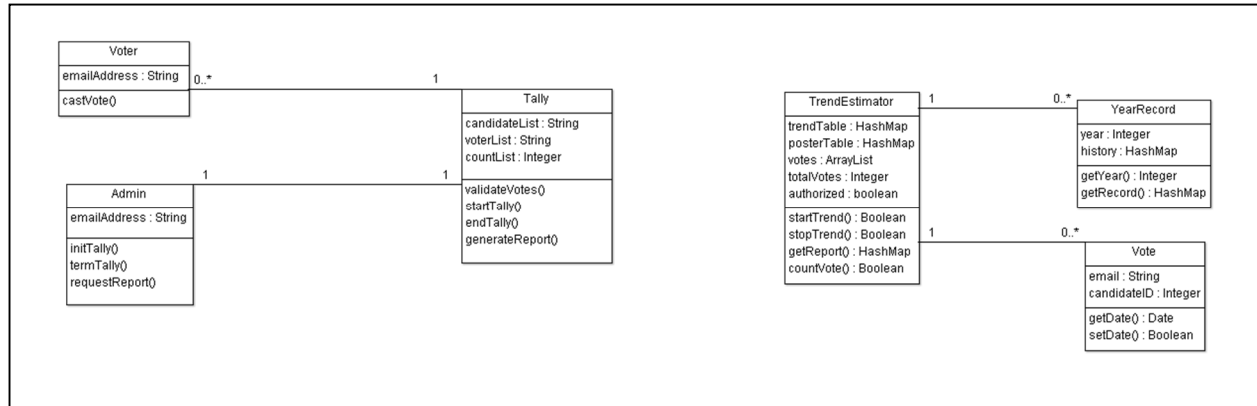


Figure 2. Class Diagram

States of the System

The system is planned to handle asynchronous messages from both voters and administrators. The first message however, must be from the administrator to start a tally. This will initialize the tables and start listening for votes. The administrator can also optionally start a trend analysis of the current years votes. This can also be done in real-time processing. The system is by no means bound by the Trend Analyzer receiving messages, but it does listen for them and respond accordingly. Please see the diagram illustrating this below.

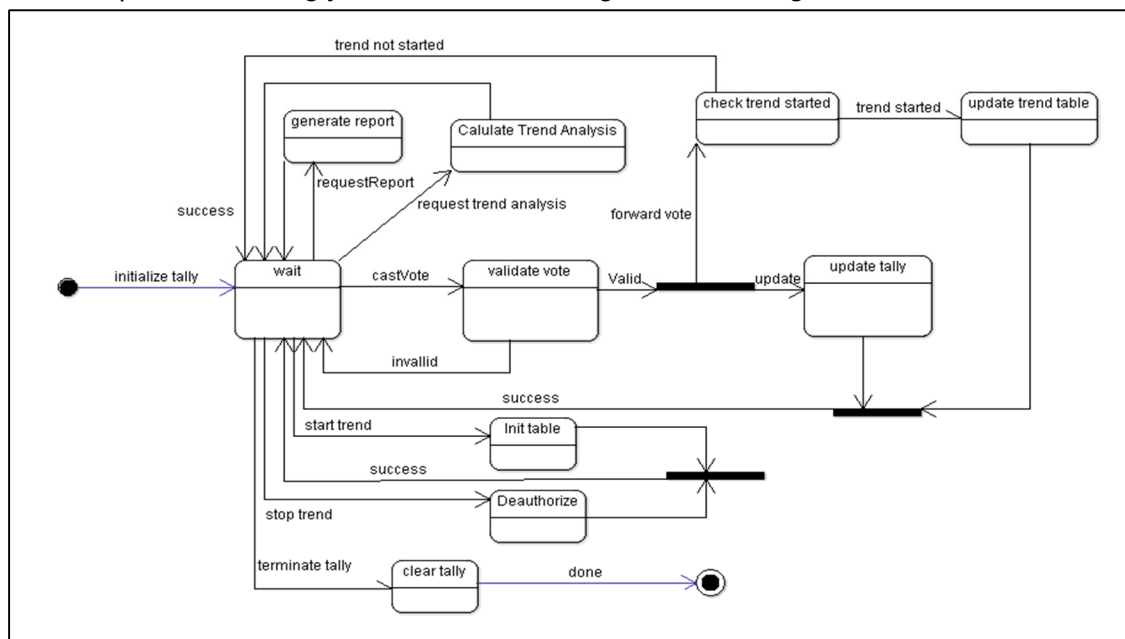


Figure 3. Statechart Diagram

Collaboration Diagram

The Collaboration diagram show how the objects in the system interact with each other. Each of the objects act asynchronously from other objects in the system. The messages in each interaction show how data is passed for each of the functions of the system. A user (a voter or administrator) send messages asynchronously to the system, and the interface handles the messages as an intermediary between the two components of the system, the Voting component and the Trend Analysis component.

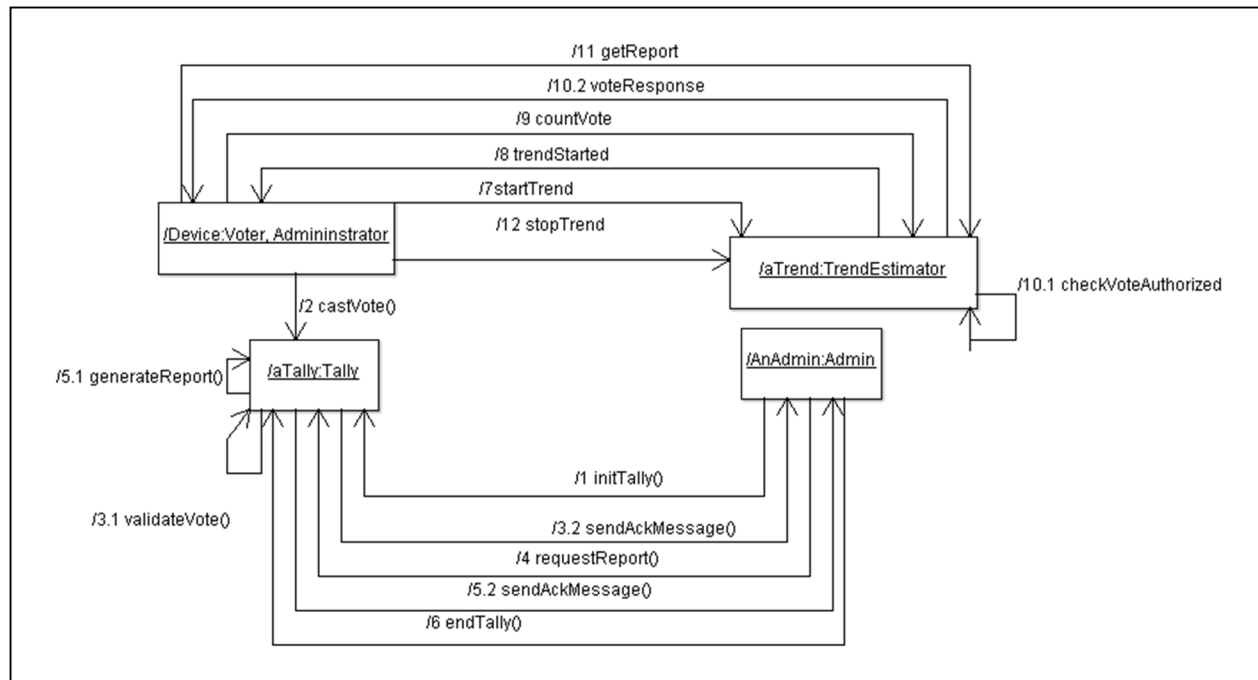


Figure 4. Collaboration Diagram

Overview of the System

The system operates in a simple client/server model as illustrated in the diagram below. The Mobile Voting System resides on a Dell Laptop running a LAMP Stack (Apache Webserver, PHP, and MySQL) and listens for any requests that come in from either administrators or voters. Voters or administrators can use any email enabled device to send messages to the system, but smartphones were chosen as the user end device since the command messages and voting will take place at a convention. The Mobile Voting System displays two components residing within the same server. The two component interact via an interface within the system, and exchange messages with each other and the user's of the system.

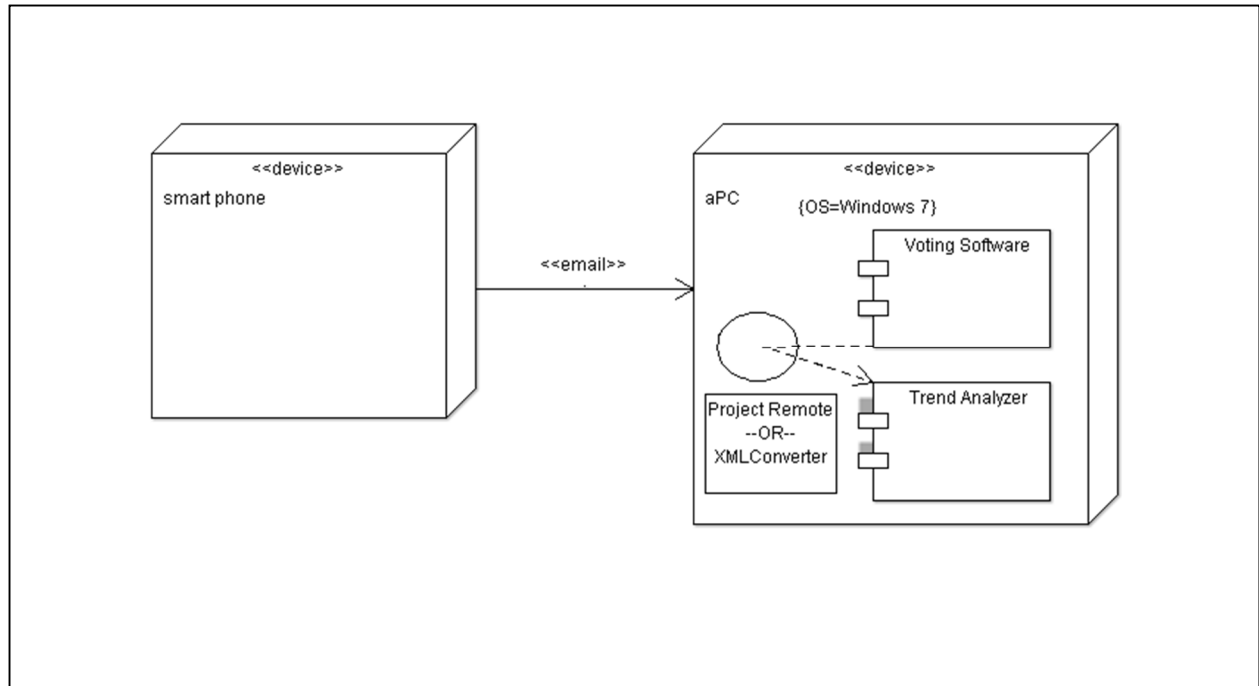


Figure 5. Deployment Diagram

Scenario Walkthrough

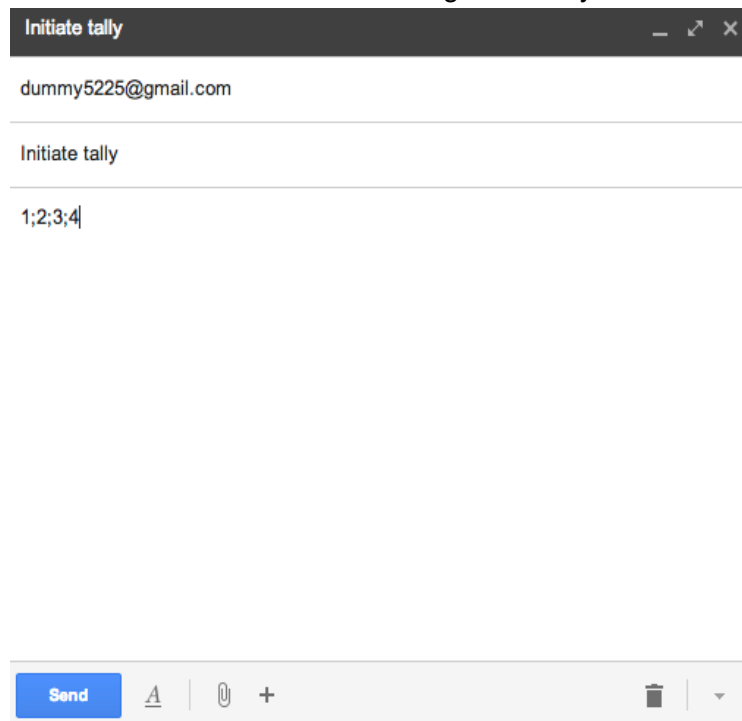
Scenario Description

In these scenarios, we will walk through the administrator initializing the system, starting a trend analysis for the current voting session, request the results of the current voting, stop the trend analysis, and request a trend report for next year. We will also display users casting votes for posters, and we will show the system, including the voting component and the trend analysis component, responding to user votes.

Scenario I

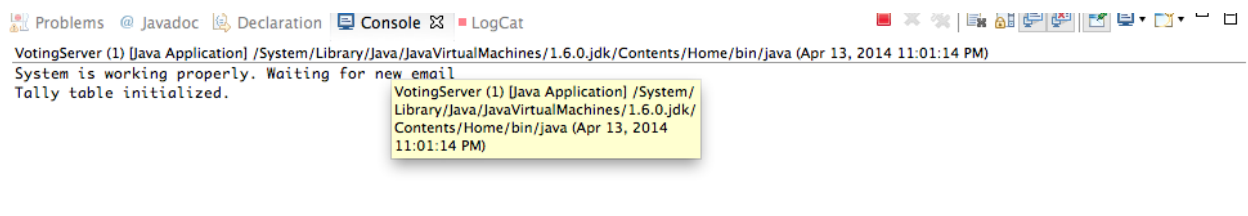
The administrator will initialize the voting system, begin a voting and trend analysis session.

1. The System administrator sends a message to the system to initialize the tally table.



The screenshot shows a dialog box titled "Initiate tally". It contains a text input field with the email address "dummy5225@gmail.com". Below this is a label "Initiate tally" and a text input field containing the command "1;2;3;4". At the bottom of the dialog is a blue "Send" button and a toolbar with icons for text formatting, attachments, and a trash can.

2. System response to command message.



3. The response from PrjRemote

Server's IP: 127.0.0.1 Port Number: 7999 Connect

Sending Message

Path: r/xml/InitXML/CreateVSXMLConverter.XML Load

Key	Value
MsgID	20
Description	Create VSXMLConverter
Name	VSXMLConverter
OutputMsgID1	800
OutputMsgID2	801
OutputMsgID3	901
OutputMsgID4	802

Create... Save Send

Message Received

Key	Value
MsgID	800
Description	startTrend
admin	1234

Refresh Rate (s/Msg): 2.0 Apply Rate

Save Message

4. The system forwards the command message to the Trend Analysis component. The response is below.

```
CreateTAXMLConverter [Java Application] /System/Library/Java/JavaVirtu
Connect to SISServer successful.
Message MsgName:startTrend MsgID:800 received, authorizi
Authorized!
```

Scenario II

Users will cast their votes using pc's and smartphones.

1. Voter uses smartphones to cast a vote to the system.

Cancel	Cast vote	Send
To: dummy5225@gmail.com		
Cc/Bcc, From: mao73@pitt.edu		
Subject: Cast vote		
3		

Cancel	Cast vote	Send
To: dummy5225@gmail.com		
Cc/Bcc, From: mattohanlon3@hotmail....		
Subject: Cast vote		
2		

To: ▾	<input type="text" value="dummy5225@gmail.com"/>	<input type="button" value="remove"/>
CC: ▾	<input type="text"/>	<input type="button" value="remove"/>
BCC: ▾	<input type="text"/>	<input type="button" value="remove"/>
Subject:	<input type="text" value="Cast vote"/>	
Priority	Normal ▾ Receipt: <input type="checkbox"/> On Read <input type="checkbox"/> On Delivery	
<input type="button" value="Signature"/> <input type="button" value="Addresses"/> <input type="button" value="Save Draft"/> <input type="button" value="Send"/> <input type="button" value="Check Spelling"/> <input type="button" value="Cancel"/>		
<div>2</div>		

2. The system responds to votes.

```
Problems @ Javadoc Declaration Console LogCat
VotingServer (1) [Java Application] /System/Library/Java/JavaVirtualMachines/1.6.0.jdk/Contents/Home/bin/java (Apr 13, 2014 11:01:14 PM)
System is working properly. Waiting for new email
Tally table initialized.
Message MsgName:startTrend MsgID:800 sent.
Vote successfully cast!
Message MsgName:startTrend MsgID:901 sent.
```

```
Console
CreateTAXMLConverter [Java Application] /System/Library/Java/JavaVirtu
Connect to SISServer successful.
Message MsgName:startTrend MsgID:800 received, authorizi
Authorized!

Message MsgID:901 received, casting vote.
Succeed
```

3. The message is reflected in PrjRemote.

Server's IP127.0.0.1Port Number7999Connect

Sending Message

:r/xml/InitXML/CreateVXMLConverter.XMLLoad

Key	Value
MsgID	20
Description	Create VXMLConverter
Name	VXMLConverter
OutputMsgID1	800
OutputMsgID2	801
OutputMsgID3	901
OutputMsgID4	802

Message Received

Key	Value
MsgID	901
Description	countVote
VoterEmail	KEL58@pitt.edu
PosterID	2
date	2014

Refresh Rate (s/Msg)2.0Apply Rate

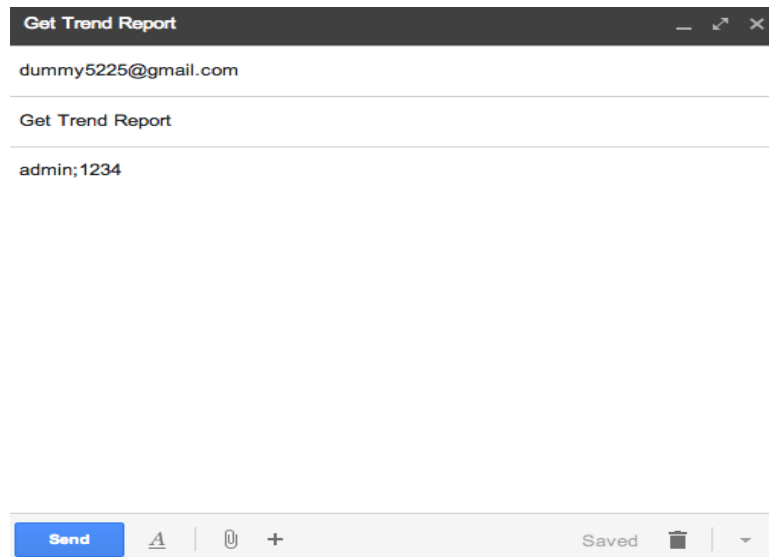
Save Message

Create...SaveSend

Scenario III

The administrator will request the results of the trend report.

1. The system administrator prepares an email message to request a Trend Report



The screenshot shows an email composition window with the title "Get Trend Report". The "To" field contains "dummy5225@gmail.com". The "Subject" field contains "Get Trend Report". The "Body" field contains "admin;1234". At the bottom, there is a "Send" button and a status bar that says "Saved".

2. The system responds to the administrator request for the Trend Report.

```
CreateTAXMLConverter [Java Application] /System/Library/Java/JavaVirtu
Connect to SISServer successful.
Message MsgName:startTrend MsgID:800 received, authorizi
Authorized!

Message MsgID:901 received, casting vote.
Succeed

Message MsgID:901 received, casting vote.
Succeed

Message MsgID:901 received, casting vote.
Succeed

Message MsgName:getTrendReport MsgID:802 received, retri
Trend Report Generated!
```

3. The message returned by the Trend Analyzer component is reflected in PrjRemote.

Server's IP127.0.0.1Port Number7999Connect

Sending Message

:/xml/InitXML/CreateVXMLConverter.XMLLoad

Key	Value
MsgID	20
Description	Create VXMLConverter
Name	VXMLConverter
OutputMsgID1	800
OutputMsgID2	801
OutputMsgID3	901
OutputMsgID4	802

Create... Save Send

Message Received

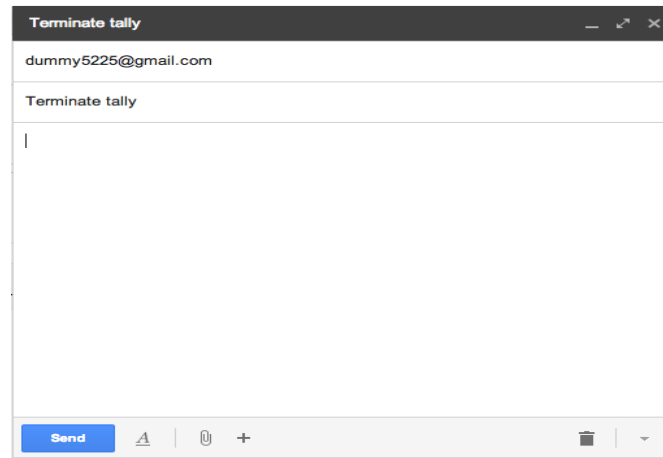
Key	Value
MsgID	803
Description	Trend Prediction
Year	2015
Multi	58.333333333333336
Game	15.0
Prod	8.333333333333334

Refresh Rate (s/Msg)2.0Apply RateSave Message

Scenario IV


The administrator will stop the tally and the trend analysis.

1. The system administrator prepares an email to send to the system to terminate the tally.



2. The system responds to the command message. Another message is automatically forwarded to the Trend Analyzer to stop the trend as well.

```
<terminated> VotingServer (1) [Java Application] /System/Library/Java/JavaVirtual
Vote successfully cast!
Message MsgName:startTrend MsgID:901 sent.
Duplicate vote attempted.
Vote successfully cast!
Message MsgName:startTrend MsgID:901 sent.
Vote successfully cast!
Message MsgName:startTrend MsgID:901 sent.
2, 2
3, 1
1, 0
4, 0
Message MsgName:startTrend MsgID:802 sent.
Message MsgName:startTrend MsgID:801 sent.
Table cleared.
```

A screenshot of a Java console window titled "Console". It displays a series of log messages from a Java application. The messages include authorization steps, successful casting of votes (MsgID:901), receipt of a trend report (MsgID:802), and successful termination of the trend analyzer (MsgID:801).

```
Console
CreateTAXMLConverter [Java Application] /System/Library/Java/JavaVirtualMachines/1.6.0.jdk/Contents/H
Message MsgName:startTrend MsgID:800 received, authorizing....
Authorized!

Message MsgID:901 received, casting vote.
Succeed

Message MsgID:901 received, casting vote.
Succeed

Message MsgID:901 received, casting vote.
Succeed

Message MsgName:getTrendReport MsgID:802 received, retrieving Trend Report.
Trend Report Generated!

Message MsgName:stopTrend MsgID:801 received, terminating Trend Analyzer.
Succeed!
```

3. Here is the response of the Trend analysis component reflected in PrjRemote.

The screenshot shows the PrjRemote application window. At the top, there are input fields for 'Server's IP' (127.0.0.1) and 'Port Number' (7999), along with a 'Connect' button. The main interface is divided into two panels: 'Sending Message' on the left and 'Message Received' on the right.

Sending Message Panel:

- A text field contains the path: `:r/xml/InitXML/CreateVSXMLConverter.XML`. A 'Load' button is next to it.
- A table with 2 columns: 'Key' and 'Value'.

Key	Value
MsgID	20
Description	Create VSXMLConverter
Name	VSXMLConverter
OutputMsgID1	800
OutputMsgID2	801
OutputMsgID3	901
OutputMsgID4	802
- At the bottom of the panel are three buttons: 'Create...' (with a dropdown arrow), 'Save', and 'Send'.

Message Received Panel:

- A table with 2 columns: 'Key' and 'Value'.

Key	Value
MsgID	801
Description	stopTrend
admin	1234
- Below the table is a large empty rectangular area.
- At the bottom of the panel are two buttons: 'Refresh Rate (s/Msg)' (with a value of 2.0) and 'Apply Rate'.
- At the very bottom of the right panel is a 'Save Message' button.