

---

# **Change Control Process for eVote**

**Version 3.0**

**Prepared by Javier Torres**

**Sylvia Rodriguez**

**Laura Cruz**

**Angel Vega**



## Contents

<b>Change Control Process for eVote Version 3.0 .....</b>	<b>i</b>
<b>Introduction.....</b>	<b>1</b>
<b>Version 1.0. Write-In .....</b>	<b>2</b>
Impact .....	2
<b>Database modifications .....</b>	<b>3</b>
<b>Scalability Simulation .....</b>	<b>4</b>
<b>Screens Modifications .....</b>	<b>5</b>
<b>Version 2.0. ....</b>	<b>6</b>
<b>Database History Table .....</b>	<b>6</b>
<b>Firmware Screens .....</b>	<b>6</b>
<b>Version 2.1. ....</b>	<b>8</b>
<b>Code Composer Migration .....</b>	<b>8</b>
<b>Final Physical Prototype .....</b>	<b>8</b>
<b>Version 3.0 .....</b>	<b>9</b>
<b>Microprocessor Change .....</b>	<b>9</b>
<b>Contact Information .....</b>	<b>10</b>

## Revision History

Name	Date	Reason For Changes	Version
eVote Write-In Feature, Database modifications and Scalability simulation	Oct 26, 2008	Additional Client features desired and database modifications made. Also adding to simulate additional devices connected to an eVote system.	1.0
Screen, Admin application and Database designs modified.	Nov 6, 2008	The code space allowed has exceeded the 4kb amount. Screens were redesigned to allow for optimization of code. The database was modified since it was found that there is no need for the history table.	2.0
Migration to Code Composer and final physical prototype	Nov 10, 2008	Migration from the IAR workbench to be able to have more firmware code space, but there is a hindrance in performance. The final prototype was transferred to a wirewrapped circuit but due to complications was reverted to a breadboard circuit.	2.1.
Microprocessor Change	Dec 4, 2008	Due to a damaged internal crystal, the microprocessor needed to be changed in order to complete the eVote	3.0

*Change Control Process*

		UART functionality	
--	--	--------------------	--

## Introduction

### Purpose

---

#### Version 1.0

This document represents changes made to the proposal document given to the client on September 10, 2008. The changes include a write-in feature to allow the eVote system to type in a name using an on screen keyboard with an external keypad. This additional requirement causes an impact to the budget as well as resources for eVote and is specified in this document.

#### Version 2.0

Additional eVote specifications modifications have been made to ensure functionality of the firmware due to space limitations

- **Version 2.1. :** The performance of the code composer firmware.

#### Version 3.0

The MSP430FG4619 suffered static electricity damage, requiring a change in microprocessor development kits.

---

### Scope

- Version 1.0.
    - Write-In feature specifications
    - Database design modifications
    - Scalability simulation
      - Removal of UART driver
  - Version 2.0.
    - Database History Table modified
    - Firmware screens modified
  - Version 2.1.
    - Migration to Code composer software
    - Final Physical prototype
  - Version 3.0
    - Microprocessor changed
    - Migration of Code to an MSP430F149 microprocessor
-

## **Version 1.0.**

### **Write-In**

Due to recent developments, the write-in feature is now desired for implementation for the eVote project. The write-in will be implemented by providing an onscreen alphabet that is navigated through with the keypad that was already involved in the navigational design. The keypad will allow the user to cycle through the alphabet and press the “ok” button once they have the desired letter selected. The write-in will only support a maximum of 28 characters so that the name can be viewed on one row on screen. The alphabet will be displayed in a single row as well with a backspace key and a finish key which will be represented by “fin” allowing the user to complete their write-in.

### **Impact**

Additional lines of code are required to implement this, and due to the microprocessor limitations, there is a possibility that the entire code for eVote may not fit into one microchip of the microprocessor currently being used (msp430fg4619). If this is to occur, then the demonstration will be divided into segments of code, demonstrating the individual functionality of the different functions of the eVote firmware.

More resources are required, but due to the limited size of the team, task reassignment has been done to allow for completion of this feature. Angel will now be in charge of completing the encryption and decryption modules for eVote. Laura who was initially assigned the task of decryption will now divert all her time to completing the write-in feature. Due to this more hours will be clocked in, therefore a budget increase may be necessary.

The increase in the budget would be \$1153.80 due to two additional weeks needed for designing and implementing this feature at this stage of eVote. This value is calculated by using added two weeks of Laura’s salary to the main budget and it may be necessary to push the end date of the eVote project. The total budget would then total \$64,711.67 as opposed to the initial budget request.

For the moment everything is going according to the schedule and this impact has been mitigated by distributing the tasks upon the employees of JSAL. The budget increase is a possible impact if the design and implementation of this feature affects the main schedule of eVote.

## **Database modifications**

As the project progressed some changes were necessary in the database design. Initially a table called “Position” listed all the positions candidates were running for in the current elections. Each position had a unique ID and in the Candidate table, each Candidate was related to a Position ID, therefore, relating a candidate with a position. However, Laura realized that the relationship would be better understood if a candidate simply had a field that indicated in text the position he or she was running for. For this reason, the Position table was dropped and the Position field in the Candidate table was edited to specify in text the position the candidate was running for.

In addition to this, a History table was added in order to associate an Official with a Voter. This way, an official can easily be identified if a voter has any sort of complaint, or vice versa. Another change was the addition of a Bar Code field in the Candidate table. This barcode is needed at the time of printing the voting receipt. Finally, each kiosk was identified a local ID order to identify them with ease locally (in their respective units).

## **Scalability Simulation**

After realizing that C# includes driver classes for UART devices (universal asynchronous receiver/transmitter), this task is no longer required for the 3<sup>rd</sup> phase of the eVote implementation. The new task intended to replace the driver is with a simulation of additional connected devices. The simulation will basically verify if there is another device connected, but instead of an additional eVote device, a pen drive will be used. The pen drive will contain additional voting information that belongs to voters of the kiosk that the pen drive will simulate. C# code will implement additional threads to read text files with the voting summaries located within the pen drives. The threads will prove that more than one device can be connected at once to the eVote system. This task will be assigned to Angel due to the driver implementation was originally his to complete.

## **Screens Modifications**

The proposal indicated that when an item is selected a rectangle will select the highlighted item and in the ballot's screen, a circle will fill when highlighting a specific row. This has been modified as follows:

1. The confirmation screens that have yes or no choices will have parenthesis below them and the highlighted item will have a cursor flashing between the parenthesis of the yes or no choice. As seen below.

Yes	no
(     )	(   )

2. The ballots screen will also implement the parenthesis and cursor highlighting methods.



## Version 2.0.

### Database History Table

The history table was removed from the database since it was more efficient to add an extra column to the voter table. This was done in order to store the information that links an elections official with the voter they registered using the administrator application.

### Firmware Screens

The screens designed to be used with the eVote LCD use up too much code and due to space limitations they must be removed. The screens will no longer display a stick figure next to each candidate and will also not display horizontal lines as seen in Figure 1.





	CANDIDATO #1 GOVERNADOR	<input type="radio"/>
	CANDIDATO #2 GOVERNADOR	<input checked="" type="radio"/>
	CANDIDATO #3 GOVERNADOR	<input type="radio"/>
	CANDIDATO #4 GOVERNADOR	<input type="radio"/>

Figure 1 – eVote ballot design

Also to maximize optimization of code, the write-in option will be visible on the same screen as the ballot's screen. After the choices of candidates an additional option of a write-in will be available. Once selecting this item, the user will be directed to the write-in feature of eVote.

The last of the modifications to the specifications of eVote is the added back functionality to the ballot screens. This option will allow the user to return to the previous screen.

Since the maximum amount of code allowed by the IAR workbench software has been reached, JSAL is changing their development software to the Code Composer development software which allows 16 kb of firmware code. This will provide enough space to completely implement the navigation code without space problems.

## **Version 2.1.**

### **Code Composer Migration**

Due to the coding limitations of the IAR workbench, eVote has been successfully migrated to the code composer software. An unexpected tradeoff of this migration of firmware code is the performance speed has decreased significantly.

It was considered to add more graphics due the fact there is more code space available, but the software execution speed has reduced significantly and adding any additional screen elements would make the user voting experience a frustrating one.

### **Final Physical Prototype**

The eVote LCD inverter circuit was transferred to a soldered pc board circuit but suffered a loss in voltage. Analysis of the nodes of the circuit proved that the circuit was correctly connected. After removing the components and then wirewrapping them to the pc board to have a better view of the nodes, the problem was still present. The inverter circuit was then put back on the pc board with the same problem. The output voltage was not high enough to provide a darker display for the LCD therefore not allowing anything to be seen on screen. It was then noticed that a resistance was faulty and after replacing it, the output voltage satisfied the negative voltage requirements of the LCD Vo pin which ranges -15volts to -10 volts.

Since so much time and resources have been consumed to locate the error, JSAL has decided that in order to complete the functionality of the eVote system, the final prototype will remain mounted on a breadboard within a box to be turned in.

## **Version 3.0**

### **Microprocessor Change**

Upon the implementation of the UART cable with the computer software and the MSP430FG4619, the hyper-terminal was not displaying the text I was sending from the microprocessor through the UART cable. An 'a' would be sent but only random trash would be received. The same firmware code was testing (with modifications) on an MS430PF149 and no problems surfaced. The fact that the code worked perfectly on a different processor led the team to believe that the internal crystal was damaged. The use of an external crystal did not work either, after consulting with a hardware consultant, it was realized that the UART would never work as desired unless we moved on to the MSP430F149.

The change was accepted to prevent more analysis for a solution consuming anymore time. The JSAL team is on a strict deadline, the solution has already been discussed with the client (Nayda Santiago) and was accepted. This change requires no change in budget since the microprocessor was temporarily donated to our project.

## **Contact Information**

For any additional information please contact the JSAL project manager.

Javier Torres

Cel. 787-248-5262

Email: [Javier.Torres@ece.uprm.edu](mailto:Javier.Torres@ece.uprm.edu)