*Florida International University*

*School of Computing and Information Sciences*

CIS 4911 - Senior Capstone Project

Software Engineering Focus

Final Deliverable

BOLO Flier Creator Version 4

**Team Members:**

Burke, John

Gonzalez, Emmanuel

Medina, Jorge

Munoz, Nilton

**Product Owner(s)**:

Samuel Ceballos

Jason Cohen

**Mentor(s)**:

Juan Caraballo

Robert Loredo

**Instructor**: Masoud Sadjadi

The MIT License (MIT)

Copyright (c) *2016 <copyright holders>*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

***Abstract***

*The Pinecrest Police Department is making an effort to bring policing into the future. They seek to replace the long paper-based process that takes place when a new BOLO (Be On the Lookout flyer) is posted. This is why they have enlisted IBM and Florida International University to help in this endeavor.*

*This document provides a solution in the* ***BOLO 5.0*** *system, a user-driven approach that allows law enforcement agents to instantly publish BOLOs. This document contains an extensive list of implemented user stories. It also provides a detailed plan of execution that includes hardware and software specifications. System design is explained through specific diagrams. System validation is also discussed. A glossary with essential domain terminology can also be found. Finally, the last chapter contains a series of appendices with UML samples, screenshots, and progress reports.*

Table of contents goes on this page

# Introduction

BOLO 5.0 is a private web application that offers police officers the chance to distribute electronic BOLOs (Be on the Lookout flyers) so that their colleagues in the area can have instant access to them. This is meant to agile the current system, which is considered slow by certain high ranking officers in the force. BOLO 5.0 has a user interface that lets officers upload media and crime details through an electronic form which is contained in a persistent storage provided by IBM.

This introductory chapter is divided into four segments. The first section, Current System, describes how the Pinecrest Police Department handles BOLO distribution currently and how this may not be the most effective way to handle recent crime. The second section, Purpose of New System, explains the main objective of BOLO 5.0 and how it will impact law enforcement in a positive way.

## Current System

The process of BOLO distribution that the Pinecrest Police Department uses is not very competent in terms of time constrictions. In the event that a crime takes place, an officer in the field is asked to write a report on paper which contains as many details of what happened as possible. This document is passed on to a supervisor for approval. If the document is approved, it will be sent to a physical storage that allows access to detectives or other law enforcement agents that may be interested.

This process is believed to be ineffective due to high costs, crowded involvement, and slow pace. This is detrimental to the main objective of such reports, to fight crime as effectively as possible. This is why it is currently being revised.

## 

## Purpose of New System

BOLO 5.0 is meant to apply constantly evolving technologies to law enforcement in the Miami-Dade area. The system’s main objective is to take advantage of the in-patrol computers and mobile devices that police officers have access to in order to make the distribution of BOLOs an instant task.

The system offers officers an interactive view that will allow them to upload electronic media and detailed information of crime scenes. This information can be published as a BOLO flyer on a page that other registered users will have access to. This makes BOLO distribution as fast as the current technology.

# User Stories

This is a comprehensive list of the user stories that were implemented for the fourth version of BOLO. No planned user stories are contained in this chapter given that all of the ones that were requested by the product owners were delivered on time.

## Implemented User Stories

# Project Plan

The overall goal coming into BOLO version 4.0 was to implement all remaining features that were necessary to bring the web application to the functionality that existed within the first two iterations of the application. This was to be accomplished by building on the work done by the prior team on version 4.0, in continuing their transition to a nodejs and express environment. The application will still be hosted on the bluemix platform, and will make use of several of its services including the cloudant noSQL database and the sendgrid service for emails.

The first segment of this chapter delves into the hardware and software that is necessary to integrate the system in terms of development and deployment. The second segment goes into detail on the individual sprints and which user stories were produced.

## Hardware and Software Resources

In order to develop and deploy the application some hardware and software is required. The following sections will describe the environments needed.

### *Development Environment*

In order to develop the application, the follow hardware specs are recommended:

· Windows/Apple Computer

· Min of 8Gb of RAM

· At least 128 GB of disk space

· At least 2.0 Ghz processor

For software, developers will need to have working knowledge of Javascript as well as the following:

· Experience with Jade Templating for HTML

· NodeJS 4.2+

o NodeJS is a Javascript runtime environment which is used in the project as the main development language

· NPM

o NPM is a package management tool required to install project dependencies required for executing the application in the NodeJS environment

· Express Framework

o Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.

· Preferred Browser

· Preferred Development Environment such as Webstorm IDE or Atom

### *Deployment Requirements*

The project is targeted for deployment on the IBM Bluemix Platform-as-a-Service Cloudfoundry environment. Besides an IBM Bluemix account the following is also required.

· Cloudant NOSQL Database

o Access tokens are required to be set as environment variables for the application to work in both deployment and development.

· SendGrid

o This is the service the application uses for sending notifications. Again, access keys are required in order for the application to work in both deployment and development environments.

## Sprints Plan

These are the user stories that were released per sprint.

### Sprint 1

(05/##/2016 - 05/##/2016)

### Sprint 2

(05/##/2016 - 06/##/2016)

### Sprint 3

(06/##/2016 – 06/##/2016)

### Sprint 4

(06/##/2016 – 07/##/2016)

### Sprint 5

(07/##/2016 – 07/##/2016)

### Sprint 6

(07/##/2016 – 07/##/2016)

# System Design

BOLO 5.0 uses a hexagonal architecture pattern, which means that the domain is what shapes the entire system. In this case, the domain is comprised of Bolos, Agencies, and Users. These domain components also serve as subsystems. BOLO 5.0 is deployed on an IBM Bluemix server which access to relevant services such as Cloudant and Sendgrid. Some of the most influential design patterns are Repository, so that the database can be replaced; Observer, which allows for asynchronous interactions; and Adapter, which acts as a way for services to be attached easily.

This chapter is divided into four main sections. First, Architectural Patterns identifies the system architecture and the reasons behind its selection. Second, System and Subsystem Decomposition illustrates how the system has been broken into individual parts that are meant to coordinate operations equally and cohesively. Third, a Deployment Diagram offers a graphic depiction of how the system is made functional. Finally, the Design Patterns section explains class division and why they were distributed in this manner.

## 

## Architectural Patterns

BOLO 5.0 uses a Hexagonal architecture. The main reason is that there are three main elements that shape the system’s functionality; users, agencies, and bolos. These comprise the domain, which means that they all depend on each other to guarantee the proper functioning of the system. Another reason for this architecture is the use of a multi-layered approach to the subsystems. For example, the Bolo views are controlled by the routes, which send requests to the service, which controls the repository that interacts with Cloudant. This helps to decouple the system for security, testing, and scalability purposes.

The hexagonal architecture also gives a great deal of flexibility on the services that are used. They can be replaced with ease without affecting vital segments of the system’s functionality. For example, if one were to replace Cloudant as the database, it could be easily done without affecting the services, routes, or the domain components.

## System and Subsystem Decomposition

BOLO 4.0 is divided into three major subsystems; agencies, users, and bolos. All three are comprised of views, a router, a service, and a repository. They all depend on each other for proper functionality.

**Agency Subsystem**: It handles agency creation, edition, and deactivation. Without an agency, no user or bolo can exist in the system.

**User Subsystem**: It handles account and user management. It allows users to be created, edited, and deleted. It also oversees account functionality such as passwords, login, and logout.

**Bolo Subsystem**: Being the most extensive one, it manages all bolo-related functionality such as creation, previewing, PDF, Email, and others.