

# FBI Virtual Case System

Jason Boyd, Liz Howell, Ian Macfarlane, Josh Kropf

## The Quick Project Requirements

- provide the capability to find information in FBI databases without having prior knowledge of its location, and to search all FBI databases with a single query through the use of search engines;
- Web-enable the existing investigative applications;
- improve capabilities to share information inside and outside the FBI;
- provide access to authorized information from both internal and external databases
- allow the evaluation of cases and crime patterns through the use of commercial and FBI-enhanced analytical and case management tools.

## Solution High Level Design

### 1. Security

- a. The application will house and serve highly sensitive information regarding the national security of the United States. Security should be the top priority in the application. Authentication systems need to be robust and reliable to avoid any false identity logins and break-ins.
- b. Files and information passed over the network also need to be handled in a very secure way. The entire system of data transmission needs to be protected and encrypted from the outside to avoid leaking sensitive information.

### 2. Hardware

- a. The goal for the VCS should be to move entirely online. This means having a reliable and very fast network to share information to and from network points simultaneously. This will require state of the art network components including cables, modems, gateways, and routers.
- b. In addition to a fast network, the data being saved also needs to be quick in saving and retrieving. This means highly functional servers will also be necessary in delivering information to and from sharers instantaneously. The project will require enough server space to house current FBI information and information for years and years to come.

### 3. User Interface

- a. First and foremost, the user interface must be simple. The program will be used by a variety of people who are not in the technical business of developing the program, so hiding most of the complicated operations from the user will be necessary.
- b. The application should feature a navigational model that uses a homepage that will branch off into other useful pages depending on what the user needs to accomplish. One central point where everything can be searched and found will be this main starting point, the home screen.

### 4. Database

- a. The application will probably need to store the data in a fully operational relational database that will make use of tables to store data efficiently.
- b. Audit trails will be necessary as tracking whoever made changes to sensitive data will be necessary when something goes wrong.
- c. User access should probably be limited to individuals who have the security clearance and necessary motives to access information within the database to avoid information being spread unnecessarily.

**Were the requirements precisely known?**

Because the FBI was under such pressure to complete the VCF, they skipped the entirely planning process of the software development cycle to get onto development as soon as possible. As a result of their mistakes the entire product failed.

**What aspects of the project are high risk?**

Transitioning previous data into a new database system is very high risk. If something goes wrong with the development part of this stage, all data previously known in the old database could be lost, which would be catastrophic for everybody involved.

**Did your design address all the high risk items?**

It answers most of them in terms of database and protecting sensitive data, but it does not address what would happen if actual information were to leak.

**Did the requirements change after the project started?**

Most certainly they did. Since the requirements planning phase of the software development life cycle was never touched on, only the most basic and needed requirements were of importance. After development had begun, the FBI realized many more requirements were necessary for the VCF. Further along when software had hit a demonstrable phase, the amount of change requests coming to the developers also became a problem since those requirements were not thought of in the first place.

**Did the team have enough experience on similar projects?**

It appears that the Science Applications International Corporation (SAIC) had developed these specialized types of programs before, the problem was just in the execution and poor setup of the project that caused it to fail.

**Was the team given enough time?**

The team was initially given three years in their contract with the FBI to develop the applications. That was probably a reasonable amount of time for completion, however their execution in the product is what resulted in its failure.

**Would the waterfall method have been successful for the VCS project?**

Yes, the waterfall method of software development would have been better than no software development life cycle method at all. According to the OIG, the VCS project failed because it did not follow those waterfall steps. For example, they skipped gathering all the proper

requirements, estimating the cost and time of development, scheduling unrealistic tasks, and others.

**Identify the product owner.**

The product owner would most likely be the FBI, with the scrum master being whoever is head of the development team at SAIC, and of course the development team who actually create the working program.

**What would your objectives be in the first 30-day sprint?**

The objectives for the first 30 day sprint would probably get a working simple program that allows users to share documents in a timely manner that is simple. The security part of the application should be worried about later in development, the first goal should be getting a working application. Another objective would probably be to begin working on building the servers and databases to house the data and slowly transition the old FBI information into the new system.

**Are the requirements from #1 too broad to be incorporated into scrum sprint?**

Absolutely they are, the requirements from #1 should at the least be broken up into smaller parts in order to be even considered for a scrum sprint. A scrum sprint should aim to implement a small and reasonable feature, and those requirements are too vague and broad that many implementation question cannot be answered.

**Write a few requirements that could be completed in a scrum sprint.**

Scrum sprint features could include developing a web interface for the application, developing a robust authentication system using existing or newly created security methods, create a method of sharing documents and information inside the system, etc.