Gaming Clip Capture Software

Nolan Waldschmidt
Samford University
800 Lakeshore Dr
Birmingham, AL 35229
+1 678 817 8106
nwaldsch@samford.edu

Brian Toone
Samford University
800 Lakeshore Dr
Birmingham, AL 35229
+1 205 726 2960
brtoone@samford.edu

ABSTRACT

The goal of this project is to apply the accumulation of what I have learned as a computer science student. My project is to create a version of screen capture software that captures smaller video segments. This software is meant for use in capturing gameplay and will implement features that other similar software lacks. One of the things that I have learned from my classes is the Agile method called "Scrum", which is a project management methodology. Scrum involves creating a backlog of tasks and splitting those tasks up into "sprints" which define a short period of time in which to complete those tasks. For the actual project I am using two different programming languages: Python and C++. Python powers the front end, and C++ powers the back end connection to SQLite data storage. One final goal of this research is to develop my expertise in these technologies that will be quite helpful in my future career as a computer scientist.

Categories and Subject Descriptors

User interface design, Input/Output, Object oriented development, Database design and models, Relational database model, Cloud based storage, Graphical user interfaces

General Terms

Algorithms, Management, Documentation, Design, Human Factors.

Keywords

User Interface, Audio Capture, Relational Database Design, Python, C++, SQLite, Cloud Storage, Database Access.

1. INTRODUCTION

Existing game play software currently captures the video portion of the user experience. Capture of player audible information is desirable. This project captures audible information and associates it with video game play. This is stored in a cloud relational database for future reference.

2. BACKGROUND

OpenCV is an AI machine learning library that I will be using to capture screenshots and play them back in real time.

SQLite is a database system that I will be using for the project. It has bindings to python so I will be able to connect the database with the front end and access it through the UI.

3. METHODOLOGY

Samford University Department of Math and Computer Science, May 2011, Birmingham, AL. Copyright 2011 Samford University…$10.00.

3.1 Window Capture

Window (screen) capture involves saving real time image data from an active game window. OpenCV will be used to implement screen capture. This implementation will take the individual captured images of a selected window and bind them together to create a video.

3.2 Audio Capture

Real time audio information generated by the game and the microphone audio information will be captured and merged into a single audio stream.

3.3 Video and Audio Merge

Captured audio and window files will be merged into a single media object containing this video and audio information The source object information will be synchronized to create a coherent target media stream.

3.4 User Interface

An overlay activated by a hotkey will activate customization options of window capture settings. to be able to customize the settings of the window capture. Some options address the ability to activate capture, modify capture duration, or storage location.

3.5 Video Storage

The videos will be uploaded to cloud based storage places such as dropbox or google drive and the hyperlink to them will be stored in the database. This cuts down on the database taking up too much storage.

3.6 Cloud Interface

I will have an account set up on dropbox or google drive for use parallel to the database. For hosting the database AWS or Oracle Cloud.

4. Expected Outcome

The expected outcome of this project is software that captures a game's image and audio information. This includes both the audio generated by the game and the audio from the player microphone input. This information is synchronized, merged, and saved to a cloud storage database. and saves them to a cloud storage database. It will also have a basic UI so that some of the parameters can be changed without having to directly modify the code.

5. CONCLUSION

Using core knowledge obtained in computer science, a student or early career computer scientist can use existing processes and software tools to efficiently and inexpensively model and implement desirable end user changes in off-the-shelf software.

6. REFERENCES.

[1] Owens, Michael, and D. Richard Hipp. *The Definitive Guide to SQLite*. Apress, 2006.

[2] Bhosale, S.T., et al. "Research Article Sqlite: Light Database System." *Https://Www.researchgate.net/*, Apr. 2015,

[3] Phillips, Dusty. *Python 3 Object Oriented Programming*. Packt Publishing, 2015.

CCS

```
\begin{CCSXML}
<ccs2012>
   <concept>
       <concept_id>10003120.10003123.10010860.10010858</concept_id>
       <concept_desc>Human-centered computing~User interface design</concept_desc>
       <concept_significance>300</concept_significance>
               </concept>
   <concept>
       <concept_id>10011007.10010940.10010941.10010949.10010965.10010967</concept_id>
       <concept_desc>Software and its engineering~Input / output</concept_desc>
       <concept_significance>100</concept_significance>
              </concept>
   <concept>
       <concept_id>10011007.10011074.10011092.10011093</concept_id>
       <concept_desc>Software and its engineering~Object oriented development</concept_desc>
       <concept_significance>300</concept_significance>
              </concept>
   <concept>
        <concept_id>10002951.10002952.10002953</concept_id>
        <concept_desc>Information systems~Database design and models</concept_desc>
        <concept_significance>500</concept_significance>
              </concept>
   <concept>
         <concept_id>10002951.10002952.10002953.10002955</concept_id>
         <concept_desc>Information systems~Relational database model</concept_desc>
         <concept_significance>500</concept_significance>
              </concept>
<concept>
         <concept_id>10002951.10003152.10003517.10003176</concept_id>
         <concept_desc>Information systems~Cloud based storage</concept_desc>
         <concept_significance>500</concept_significance>
              </concept>
   <concept>
          <concept_id>10003120.10003121.10003124.10010865</concept_id>
          <concept_desc>Human-centered computing~Graphical user interfaces</concept_desc>
          <concept_significance>500</concept_significance>
              </concept>
 </ccs2012>
\end{CCSXML}

\ccsdesc[300]{Human-centered computing~User interface design}
\ccsdesc[100]{Software and its engineering~Input / output}
\ccsdesc[300]{Software and its engineering~Object oriented development}
\ccsdesc[500]{Information systems~Database design and models}
\ccsdesc[500]{Information systems~Relational database model}
\ccsdesc[500]{Information systems~Cloud based storage}
\ccsdesc[500]{Human-centered computing~Graphical user interfaces}
```