Glendale Unified School District

High School

February 5, 2019

Department: Career Technical Education

Course Title: Applied Math and Mobile App Development
                     (Formerly Introduction to Mobile Apps 1-2)

Course Code: 2257V/2258V

Grade Level(s):   9-12

Course Credits: 10

Recommended   none
Prerequisite:

Recommended
Textbook: Ralph Morelli and Ralph Walde, Java, Java, Java: Object-Oriented
                     Problem Solving, December 30, 2016
                     http://www.cs.trincoll.edu/~ram/jjj/jjj-os.pdf

Course Overview: Introduction to Mobile Apps is the introductory course for the Software
                     and Systems Development and Networking Pathway under the
                     Information and Communication Technologies (ICT) industry sector. The
                     course is designed to be an introductory course for students who want to
                     explore the process of designing and building apps for mobile devices. It
                     is designed to introduce students to the tools, methods and opportunities
                     in the app-development field. There is high interest among our student
                     population to have an experience exploring this field of technology. This
                     course will fill that need. In addition, it will provide the base for a
                     sequence of programming courses, the next being Application to
                     Programming using Java, mobile application development with Android
                     and iOS.

                     Career Readiness Anchor Standards for Information and Communication
                     Technologies (ICT): 2.1, 2.2, 2.3, 2.4, 3.1, 3.2, 3.3, 4.3, 4.4, 5.6, 5.7, 6.8, 6.10,
                     6.11, 8.8, 10.1, 10.2 10.3, 10.8, 10.9 10.14

First Semester-Course Content

Unit 1: **Introduction to Mobile Computing** *(4 weeks)*

STANDARDS
California CTE Software and Systems Development Pathway Standards: C1.1, C1.2, C1.3
Common Core Standards: CCSS LS 11-12.1, 11-12.2, 11-12.5, CCSS RSIT 11-12.3, 11-12.5, CCSS
writing 11-12.1, 11-12.211-12.3

A. The introductory unit for this course utilizes a series of digital mazes to introduce the concept of algorithms and programs. Students learn that programmers use the control structures of sequence, selection (if/then), and repetition when they design an algorithm to solve a problem or perform a computation. Students create their own websites using Google sites. The websites will be used to share their reflections on learning and their portfolio of app designs with their instructor and classmates. Students are introduced to App Inventor, the programming language they will be using throughout the course to design their apps. They will create their own App Inventor account and test it out on a mobile device (tablet or cell phone). As a basis for their reflections, students will use readings from the book Blown to Bits throughout the course to focus on the central issues of computing with a primary focus on how society is impacted by the digital world. The final element of unit one is an introduction to the Mobile CSP forum, a place where students from all over the country can pose questions and share thoughts relating to the digital world in general and App Inventor in particular.

B. With the introductory nature of unit one, setting up the Google components correctly will be critical to student success. Glendale Unified School District sets up a Google Gmail account for each student in the district. Students will use this account as a springboard to create a website using Google sites. The individual student websites are where students post their reflections as well as their completed mobile applications. The primary focuses of the reflections for this unit are on analyzing the historical components of computing; understanding where we come from to better understand where we are now. Privacy issues around computing will also are a key area of focus for reflections.

Unit 2: **Mobile Computers and Mobile Apps**   *(6 weeks)*

STANDARDS
California CTE Software and Systems Development Pathway Standards: C.1.4, C1.5, C1.6
Common Core Standards: CCSS LS 11-12.5, 11-12.6, CCSS RSIT 11-12.3, 11-12.4, CCSS Writing 11-12.2

A. This unit focuses on getting to know the App Inventor development environment and mobile computing as well as related topics from the CS Principles curriculum. It sets the stage for the rest of the course, emphasizing the collaborative and creative nature of computing. The use of pair programming as a pedagogical tool is introduced. The CSP principles emphasized in this unit are: creativity, abstraction, programming, and global impact. Students use the creative development process when using computing tools and techniques to create computational artifacts for creative expression of ideas or to solve a problem. Students use a variety of abstractions built upon binary sequences to represent all digital data. Multiple levels of abstraction are used to write programs or to create other computational artifacts. The programs students develop are used to; solve problems, for creative expression, to satisfy personal curiosity, or to create new knowledge. Students make a global impact through computing by enhancing communication, interaction, and cognition.

B. Students create an app that plays Martin Luther King Jr's "I Have a Dream" speech. Over the course of this unit the app is modified to add additional levels of complexity, including the addition of a button that allows you to listen to a speech of Malcolm X's, and the ability to toggle back and forth between the two speeches. Students add a feature that pauses a speech as it is being played. Students will also add a talk to text feature that allows the mobile device to speak words instead of play a speech. An accelerometer featured is added to the app that makes the app play one of the speeches when it is shaken. Each additional element in the app introduces a new feature of the programming language to students and offers them the ability to test it out for themselves. Students will then apply and expand on what they have learned by designing and developing their own soundboard app that incorporates the skills and features presented in the I Have a Dream project. This extension will also introduce students to copyright and the protection of intellectual property.

Unit 3: **Graphics and Drawing** *(6 weeks)*

STANDARDS
California CTE Software and Systems Development Pathway Standards: C2.1, C2.2, C2.3
Common Core Standards: CCSS LS 11-12.1, 11-12.2, 11-12.3, 11-12.4, CCSS RSIT 11-12.1, 11-12.2, 11-12.7, CCSS WS 11-12.6, 11-12.7

A. This unit builds on and extends the concepts of digital data and abstraction by introducing digital graphics. Understanding of the binary system and how digital images are represented in binary is a key component of this unit. Students learn how the application of binary numbers can be used to help with error detection during data transmission. These lessons are complemented nicely by a Blown to Bits reading that focuses on digital documents, including how they can be used to hide secrets. Students will build simple two dimensional graphics to be used in an app. Students learn to practice incorporating graphics and using drawing tools in applications. Students will begin developing the skill of debugging, or error detection in their projects. In addition to using the drawing elements, students will learn to begin defining variables, use event handlers, and begin to use abstraction.

B. Paint Pot is a project that involves creating simple two-dimensional graphics. Students will build an app that allows users to select colors, and draw lines or dots with their finger on the device screen, clear the screen, and use the device camera to take another image to use as the background. This project teaches using the canvas and arrangement components, handling touch and drag events on the device surface, and using event handlers that have arguments. Paint Pot 2 builds on the Paint Pot tutorial project but adds in additional complexity. The app incorporates abstraction and introduction of defining variables by adjusting the dot-size that can be used in the app.

Second Semester-Course Content

Unit 4: **Abstraction (Animation, Modeling and Simulation)** *(4 weeks)*

STANDARDS
California CTE Software and Systems Development Pathway Standards: C2.4, C2.5, C3.1, C3.2, C3.3
Common Core Standards: CCSS LS 11-12.1, 11-12.3, 11-12.3, 11-12.4, CCSS RIT 11-12.1, 11-12.2, CCSS WS 1-12.2

A.   Abstraction is a central problem-solving technique in computer programming that is a process, a strategy, and the result of reducing detail to focus on concepts relevant to understanding and solving problems. This course includes examples of abstractions used in modeling the world, managing complexity, and communicating with people as well as with machines. Students in this course learn to work with multiple levels of abstraction while engaging with computational problems and systems; use models and simulations that simplify complex topics in graphical, textual, and tabular formats; and use snapshots of models and simulation outputs to understand how data is changing, identify patterns, and recognize abstractions.

B.   Mole Mash is a project that mimics whac-a-mole. This project involves using sprites, the clock, the sound and, button components, generating random numbers, and using addition and subtraction elements. This project has students build and name procedures into a customized block and use tactile feedback by having the phone vibrate when the mole sprite is hit. Coin Flip is an app that simulates the flipping of a two-sided coin. This app uses a random number generator and two images to simulate the coin flip. Students will define a global variable and assign it an initial value, and use a conditional statement in the building of this app.

Unit 5: **Algorithms and Procedural Abstraction**   *(6 weeks)*

STANDARDS
California CTE Software and Systems Development Pathway Standards: C4.1, C4.2, C4.3 C4.4
Common Core Standards: CCSS LS 11-12.1, 11-12.2, 11-12.3, 11-12.4, CCSS RSIT 11-12.1, 11-12.2, 11-12.2, 11-12.4, 11.12.7, CCSS WS 11-12.1, 11-12.2, 11-12.4, 11-12.6

A.   This unit will teach students how structure their apps to make decisions through the use of algorithms and procedures. An algorithm is a method or formula that we use to solve a problem. A good algorithm breaks a large problem down into discrete steps that solve recurrent problems. In this unit, students will learn to map the flow of a program by breaking it down into specific procedures that help the student think about and implement the target application. The development of these procedures is called Procedural Abstraction, where procedures are used to help manage the complexity of a larger problem. Students will learn additional advantages of using procedures: first, it is easy to test them as a distinct unit of code. Second, if a procedure contains an error, we must only fix it one time instead of looking for occurrences across a program. Third, if a change needs to be made to the algorithm, it must only be made once.

B. Xylophone is a project that allows the user to play different notes by touching 8 different colored buttons on the screen. The program keeps track of the notes that have been played, and will play them back when a play button is pressed. If the user chooses to being a new song, a reset button will erase previously stored notes. This project requires students to come up with the flow-chart algorithm that maps the order of tasks that must be accomplished. They will then practice procedural abstraction by assigning specific tasks to procedures and selecting appropriate parameters.

Unit 6: **Lists, Databases, Data and Information**   *(5 weeks)*

STANDARDS
California CTE Software and Systems Development Pathway Standards: C4.5, C4.6, C4.7, C4.8, C4.9, C4.10, C4.11
Common Core Standards: CCSS LS 11-12.3, 11-12.4, CCSS RSIT 11-12.1, 11-12.2, 11-12.2, 11-12.4, 11.12.7, CCSS WS 11-12.2, 11-12.4, 11-12.6

A. This unit teaches students how to manage data. They must be able to instantiate their own data, ask for user input as data, access and manipulate that data, and store the data in a database. Lists are a way of storing data while a program is running. An indexed list is essential for the storage and retrieval of related data types. But all data is lost when the program ends. To store the information persistently it must be transferred to a database. Databases are critical for storing any data that must exist before and after a program ends, so that it can be used again when the app reopens. In this unit, students will learn how to make their apps communicate with databases.

B. The Make Quiz/Take Quiz project will have students design two apps that create and administer quizzes. The Make Quiz app will allow a teacher to create a quiz using an input form. The questions and answers will be stored in a web database. Those questions and answers will be retrieved by the Take Quiz app which will ask students those same questions. Students will learn how to use input forms to collect user input, and how to use an indexed list to store data, and a for-each loop to access it. They will also learn how to make a database that can be created by one app and used in another, and how to implement data sharing in a web database.

Unit 7: **The Internet**   *(5 weeks)*

STANDARDS
California CTE Software and Systems Development Pathway Standards: C4.5, C4.6, C4.7, C4.8, C4.9, C4.10, C4.11
Common Core Standards: CCSS LS 11-12.1, 11-12.2, 11-12.3, 11-12.4, CCSS RSIT 11-12.1, 11-12.2, 11-12.2, 11-12.4, 11.12.7, CCSS WS 11-12.4, 11-12.6, 11-12.8, 11-12.9

A. This unit introduces students to the Internet; what it is and how it works. Students distinguish between the Internet and the World Wide Web. Students make use of online tools to measure bandwidth and latency. The fundamentals of how the Internet works will introduce the students to packet switching, Internet protocols especially TCP/IP, IP addressing and DNS. The concepts of cloud computing will be examined and issues related to cloud computing such as privacy will be examined. Students will analyze the benefits and pitfalls of cloud computing for various users. Students will reflect on computer ethics standards and their importance to the tech field.

B. Students develop three apps that show different ways to harness the power of the Internet. A No Texting While Busy app that automatically responds to text messages while the user is busy. This app uses Google Voice to text over WiFi. A Broadcast Hub app is created where users text a join message to the hub and then they are included in the members list. Anytime a member texts the list their message is broadcast to all members. Students create a My Directions app, which uses the device's GPS and the Google Maps API to find directions from a user's current location to a list of set destinations.

Additional Recommended Materials –

Allen B. Downey and Chris Mayfield, Think Java How to Think Like a Computer Scientist, 2016
http://greenteapress.com/thinkjava6/thinkjava.pdf

David J. Eck, Introduction to Programming Using Java, Seventh Edition, Version 7.0, August 2014, http://math.hws.edu/javanotes/