El Monte Union High School District

Course Outline

High School DISTRICT

Title: Computer Science & Software Engineering (PLTW)_ This course meets

Transitional*_____(Eng. Dept. Only)

Sheltered (SDAIE)*____Bilingual*___

AP**_____Honors**_____

Department: Industrial Technology

Grade Level (s):_____9-12_____

Semester_____Year__X____

Year of State Framework Adoption_____

3. Describe how this course integrates the schools ESLRs (Expected School-wide Learning Results):

This course integrates the ESLRs by combining elements of hands on career technical education in various areas while using the skills of reading, writing and mathematics. Computer-based programming technology will be used. Each unit in the curriculum focuses on one or more computationally intensive career paths.

4. Describe the additional efforts/teaching techniques/methodology to be used to meet the needs of English Language Learners:

Additional efforts, techniques and methodologies used to meet the needs of ELL students include the pairing or partnering of ELL students with bilingual advanced students open to assisting struggling or challenged students. Additionally, when available, ELL students will be provided with learning materials in their native language. Furthermore, if available, instructional aides will be provided to translate and assist ELL students with coursework. Lastly, since the course has day to day objectives in place, SIOP model practices will be implemented, such as posting daily learning objectives in written form for the entire class to see.

5. Describe the interdepartmental articulation process for this course:

Skills developed in IED and POE will continue to be used in CSE. Students will develop computational thinking and problem skills that require them to develop planning, documentation, communication, and other professional skills. These target skills and involved assignments will be shared and discussed with other departments, including math and science courses, with the goal of increasing student performance in the involved departments.

6. Describe how this course will integrate academic and vocational concepts, possibly through connecting activities. Describe how this course will address work-based learning/school to career concepts:

The course aims to build students' awareness of the tremendous demand for computer specialists and for professionals in all fields who have computational skills. Each unit focuses on one or more computationally intensive career paths. The course aims to engage students to consider issues raised by the present and future societal impact of computing. Students practice problem solving with structured activities and progress to open-ended projects and problems that require them to develop planning, documentation, communication, and other professional skills.

7. Materials of Instruction (Note: Materials of instruction for English Language Learners are required and should be listed below.)

A. Textbook(s) and Core Reading(s):

As required by the PLTW materials guide (updated yearly).

B. Supplemental Materials and Resources:

As required by the PLTW materials guide (updated yearly). Activities, projects, and problems will be provided to the teacher in the form of student-ready handouts, teacher notes, and supplementary materials, including code, instructional videos, and online practice questions as appropriate.

C. Tools, Equipment, Technology, Manipulatives, Audio-Visual:

As required by the PLTW materials guide (updated yearly).

8.

- **Objectives of Course**

Building enthusiasm for rigorous computer science among students is a primary goal of the course. The course aims to develop computational thinking, to generate excitement about the field of computing, and to introduce computational tools that foster creativity, build students' awareness of the tremendous demand for computer specialists and for professionals in all fields who have computational skills, and to engage students to consider issues raised by the present and future societal impact of computing.

- **Unit detail including projects and activities including duration of units (pacing plan)**

Unit 1: Algorithms, Graphics, and Graphical User Interfaces

The goal of Unit 1 is to excite students about programming and to build their algorithmic thinking and ability to use abstraction. Student creativity is emphasized as they work with Scratch™, App Inventor, and Python® programming languages to tell graphical stories, publish games and Android™ applications, and explore various development environments and programming techniques. Students will create original code and read and modify code provided from other sources. An Agile software development process is emphasized and personal, professional, and collaborative skills take center stage. Students debate policy questions about the ownership and control of digital data and examine the implications for creative industries and consumers. In this unit students begin their exploration of career paths tied to computing.

1.1 Algorithms and Agile Development Lesson (10 days)
1.2 Mobile App Design Lesson (12 days)
1.3 Algorithms in Python Lesson (14 days)
1.4 Images and Object-Oriented Libraries Lesson (15 days)
1.5 GUIs in Python (14 days)

Lesson 1.1 Algorithms and Agile Development

The goal of this lesson is to introduce students to programming at a level appropriate to novice programme

footprint. To provide a hook, students compare the designs, strengths, and weaknesses of their favorite web pages. In this context students use an unplugged activity to understand (in broad brushstrokes) the content and flow of data when browsing the Web. They compare results from different search engines and learn to refine their search techniques. They review how to assess the trustworthiness of web-based media and consider the data flow that permits targeted advertisements. Students employ appropriate tools to explore the hierarchical nature of DNS and IP. Students identify ways that a web developer's decisions affect the user and ways that the user's decisions impact society. The tree structure of web documents is introduced alongside HTML and CSS. Paired key encryption and authentication are introduced with an unplugged activity.

Lesson 2.2 Shopping and Social on the Web

The goal for this lesson is for students to understand the role of client-side code, server-side code, and databases in delivering interactive web content. The hook is a problem in which CS students collaborate with art students to publish content on the Web. Students are provided with JavaScript and PHP code and can access an SQL database from a secure shell command line as well as through PHP. Students compare languages encountered so far to generalize the concepts of sequencing instructions, selection of instructions by conditionals, iteration, and the common roles of variables. Students explore and compare career paths within computing.

Lesson 2.3 Security and Cryptography

The goal of this lesson is for students to personally invest in maintaining online security and to improve their personal cyber security hygiene. Students focus on cyber security from the perspectives of the user, the software developer, the business, the nation, and the citizen. In the team competition at the end of the lesson, students explore parallel strands in encryption and security. Encryption is used as a route to explore the efficiency of algorithms and how the time for an algorithm to execute can be dependent on its input.

Unit 3: Raining Reigning Data

The goal of Unit 3 is for students to see the availability of large-scale data collection and analysis in every area they can imagine. Students examine very large data sets tied to themselves as well as to areas of work and society. They learn a variety of data visualization techniques and work to recognize opportunities to apply algorithmic thinking and automation when considering questions that have answers embedded in data. The complexity of the data sets, visualizations, and analysis increases in the second lesson of the unit, challenging students to generalize concepts developed in the first lesson.

3.1 Visualizing Data (20 days)
3.2 Discovering Knowledge from Data (14 days)

Lesson 3.1 Visualizing Data

The goal of this lesson is for students to be able to create visualizations to analyze sets of large data and to meaningfully interpret the patterns they uncover. They draw conclusions about themselves from relevant data, including local weather, the economics of their community, and naming trends with their name. At the beginning of the lesson, students weigh societal concerns around the collection and persistence of Big Data. The students learn how to use to make useful graphic representations of data, developing from familiar visualizations to more modern visual analyses like scaled-dot or colorized scatter plots of multidimensional data sets. Students are introduced to basic Excel® spreadsheet programming and cell manipulation. A Monte Carlo simulation is used to help students appreciate the meaning of evidence for association between two variables.

Lesson 3.2 Discovering Knowledge from Data

As in the previous lesson, the goal of this lesson is for students to be able to create a range of

then national standard should be used)

The course is designed to cover all learning objectives in the College Board's 2013 draft CS Principles framework. In specific CSE projects and problems, students create artifacts and associated writing for CS Principles performance assessment tasks. Alignment with CS Principles Learning Objectives and with CSTA Level 3B Objectives is indicated in the PLTW CSE Curriculum Framework at the activity level. Alignment with NGSS, Common Core, and other standards will be available through the PLTW Alignment web-based tool.

- **Student performance standards**