Date Approved by DJUSD Board of Education: _____

Secondary Course Description

Course descriptions are updated and reviewed with all new text adoptions.

COVER PAGE

1. Course Title: Project Invent

2. Transcript Title / Abbreviation: Software and Systems Development 3/SSD3

3. Transcript Course Code / Number (Office Use Only)**:**

4. School: Da Vinci Charter Academy High School

5. **District**: Davis Joint Unified School District

6. Department: CTE, Computer Science

7. Graduation Requirement it meets: CTE, Elective

8. Length of Course: 1 year

9. Graduation Credits: 10

10. School / District Web Site:
www.davincicharteracademyhs.net and www.djusd.net

11. CBEDS Course Code:

12. School Contact:

Name: Tyler Millsap

Title/Position: Principal, Da Vinci Charter Academy

Phone: 530-757-7154 **Ext.:**

Fax: 530-759-2178

E-mail: tmillsap@djusd.net

13. Subject Area:

- [ ] History/Social Science
- [ ] English
- [ ] Mathematics
- [ ] Science
- [x] CTE
- [ ] Language other than English
- [ ] Visual & Performing Arts
- [ ] DJUSD Graduation Elective
- [x] College Prep Elective (will seek UC/CSU approval)

14. Grade Level(s): 11-12

15. UC/CSU Requirement:

16. Seeking "Honors" Distinction? [ ] Yes [x] No

17. GPA Types:

18. Credit Value:

- [ ] 0.5 (half year or semester equivalent)
- [x] 1.0 (one year equivalent)
- [ ] 2.0 (two year equivalent)
- [ ] Other: _____

19. Was this course previously approved by UC? [ ] Yes [x] No

If so, in what year? _____ **Under what course title?** _____

20. Pre-Requisites: Computer Science Principles

Co-Requisites:

21. Preliminary Approval - Secondary Site Principal Signature (Must be signed before proceeding to Step 22):

_____

22. Date Course Proposal with Preliminary Approval (Step 15) sent to Associate Superintendent, Educational Services: _____

23. Review & Approval:

Date		Signature
11/15/19	**Site Curriculum and Instruction Leadership Team**	**Signature/Title** Sean Glantz Computer Science Teacher
_____	**Secondary Department Articulation/Collaboration**	**Signature/Title** _____

Secondary Principal Signatures: _____Tyler Millsap_____ _____

Date: _____11/15/19_____ _____

BACKGROUND INFORMATION

Brief Course Description:

Project Invent empowers high school students to go into their communities and invent technologies that make a difference. Students gain the tools and mindsets of changemakers while learning the engineering, social-emotional, and problem-solving skills to be future ready. This course will culminate with a trip to the Bay Area in May to participate in Project Invent's Demo Day, an annual pitch event where student teams from around the country showcase their innovations to top investors and tech executives for a chance at funding.

Students will develop software, applications, and web resources to solve problems within their community. They will utilize a variety of tools within a makerspace to design, prototype, test, and revise innovations they create. Students will develop relationships with community partners to better understand the needs of their users and best practices within the industry. Students will pursue ideas that they do not know how to build (yet). Through a combination of in-class instruction, online learning modules, interactions with community experts, and student led exploration, our students will gain experience with a wide-variety of digital technologies.

Context for Course:

The fields of software, computing and computer science are plagued by tremendous underrepresentation of women, African Americans, and Hispanics. In high school, the Advanced Placement exam in Computer Science has the worst gender diversity across all courses, with 78 percent participation by men and 22 percent by women. Participation by students of color is 13 percent. These underrepresented groups represent 65 percent of the entire US population! This problem extends into the software workplace, which suffers a similar lack of diversity.

Computer science is now foundational knowledge for all 21st-centruy careers, making access to this field a critical equity issue. For students who do not pursue a career in computer science, the skills learned in this class will set them apart in today's competitive workforce by giving them the tools to work with the technology they will inevitably encounter in any career path. This course and pathway are aimed at recruiting students who are traditionally underrepresented in computer science fields. The curriculum is designed to reframe the misconception that computer science classes masquerade as math classes with computers. By learning in demand skills and solving real problems, this course will help students build self-confidence, empathy, and resilience.

List the State/District Standards addressed in this course.

In addition to the CTE Anchor Standards, this course will address the standards from the Information and Communication Technologies Software and Systems Development Pathway listed below. It will also cover several standards from the Product Innovation and Design Pathway of the Manufacturing and Product Development industry sector.

Software and Systems Development Pathway (Information and Communication Industry Sector)

C1.0 Identify and apply the systems development process.

- C1.1 Identify the phases of the systems development life cycle, including analysis, design, programming, testing, implementation, maintenance, and improvement.
- C1.2 Identify and describe models of systems development, systems development life cycle (SDLC), and agile computing.
- C1.3 Identify and describe how specifications and requirements are developed for new and existing software applications.
- C1.4 Work as a member of, and within the scope and boundaries of, a development project team.
- C1.5 Track development project milestones using the concept of versions.
- C1.6 Diagram processes using flowcharts and the Unified Modeling Language.

C2.0 Define and analyze systems and software requirements.

- C2.1 Describe the major purposes and benefits of development, including automation, improving productivity, modeling and analysis, and entertainment.
- C2.2 Recognize and prevent unintended consequences of development work: programming errors, security issues, health and environmental risks, and privacy concerns.
- C2.3 Develop strategies that target the specific needs and desires of the customer.
- C2.4 Analyze customers' needs for development.
- C2.5 Determine and document the requirements and alternative solutions to fulfill the customers' needs.

C3.0 Create effective interfaces between humans and technology.

- C3.1 Describe and apply the basic process of input, processing, and output.
- 3.2 Design effective and intuitive interfaces using knowledge of cognitive, physical, and social interactions.
- C3.3 Support methods of accessibility for all potential users, including users with disabilities and non-English-speaking users.

C4.0 Develop software using programming languages.

- C4.1 Identify and describe the abstraction level of programming languages from low-level, hardware-based languages to high-level, interpreted, Web-based languages.
- C4.2 Describe the interaction and integration of programming languages and protocols such as how client-side programming can work with server-side programming to use a query language to access a database.
- C4.3 Identify and use different authoring tools and integrated development environments (IDEs).
- C4.4 Identify and apply data types and encoding.
- C4.5 Demonstrate awareness of various programming paradigms, including procedural, object oriented, event-driven, and multithreaded programing.
- C4.6 Use proper programming language syntax.
- C4.7 Use various data structures, arrays, objects, files, and databases.
- C4.8 Use object oriented programming concepts, properties, methods, and inheritance.
- C4.9 Create programs using control structures, procedures, functions, parameters, variables, error recovery, and recursion.
- C4.10 Create and know the comparative advantages of various queue, sorting, and searching algorithms.
- C4.11 Document development work for various audiences, such as comments for other programmers, and manuals for users.

C5.0 Test, debug, and improve software development work.

- C5.1 Identify the characteristics of reliable, effective, and efficient products.
- C5.2 Describe the ways in which specification changes and technological advances can require the modification of programs.
- C5.3 Use strategies to optimize code for improved performance.

- C5.4 Test software and projects.
- C5.5 Evaluate results against initial requirements.
- C5.6 Debug software as part of the quality assurance process.

C6.0 Integrate a variety of media into development projects.

- C6.1 Identify the basic design elements necessary to produce effective print, video, audio, and interactive media.
- C6.2 Describe the various encoding methods of media and trade-offs: vector graphics vs. bitmaps, and bit depth.
- C6.3 Use media design and editing software: keyframe animation, drawing software, image editors, and three-dimensional design.
- C6.4 Develop a presentation or other multimedia project: video, game, or interactive Web sites, from storyboard to production.
- C6.5 Analyze the use of media to determine the appropriate file format and level of compression.
- C6.6 Integrate media into a full project using appropriate tools.
- C6.7 Create and/or capture professional-quality media, images, documents, audio, and video clips.

C7.0 Develop Web and online projects.

- C7.1 Identify the hardware (server) and software required for Web hosting and other services.
- C7.2 Describe the full process of online content delivery, registering domain names, setting up hosting, and setting up e-mail addresses.
- C7.3 Attract Web-site visitors through search engine optimization using various strategies like keywords and meta-tags.
- C7.4 Enable e-commerce capabilities to sell products, create a shopping cart, and handle credit card transactions.
- C7.5 Create an online project, Web-based business, and e-portfolio.
- C7.6 Optimize fast delivery and retrieval of online content such as Web pages.

C8.0 Develop databases.

- C8.1 Describe the critical function of databases in modern organizations.
- C8.2 Identify and use the basic structures of databases, fields, records, tables, and views.
- C8.3 Identify and explain the types of relationships between tables (one-to-one, one-to-many, many-to-many) and use methods to establish these relationships, including primary keys, foreign keys, and indexes.
- C8.4 Use data modeling techniques to create databases based upon business needs.
- C8.5 Use queries to extract and manipulate data (select queries, action queries).
- C8.6 Develop databases that are properly normalized using appropriate schemas.
- C8.7 Export and import data to and from other applications and a database recognizing the limitations and challenges inherent in the process.
- C8.8 Analyze and display data to assist with decision making using methods like cross tabulations, graphs, and charts.

C9.0 Develop software for a variety of devices, including robotics.

- C9.1 Demonstrate awareness of the applications of device development work, including personalized computing, robotics, and smart appliances.
- 9.2 Install equipment, assemble hardware, and perform tests using appropriate tools and technology.
- C9.3 Use hardware to gain input, process information, and take action.
- C9.4 Apply the concepts of embedded programming, including digital logic, machine-level representation of data, and memory-system organization.
- C9.5 Program a micro-controller for a device or robot.

C10.0 Develop intelligent computing.

- C10.1 Describe models of intelligent behavior and what distinguishes humans from machines.
- C10.2 Describe the major areas of intelligent computing, including perception, proximity, processing, and control.
- C10.3 Know artificial intelligence methods such as neural networks, Bayesian inferences, fuzzy logic, and finite state machines.
- C10.4 Implement artificial intelligent behavior through various methods: mathematical modeling, reinforcement learning, and probabilistic analysis.

<u>Product Innovation and Design Pathway Standards (Manufacturing and Product Development Industry Sector)</u>

D6.0 Produce a prototype of a product.

- D6.1 Build a looks-like, works-like prototype of the model using the appropriate fabrication, manufacturing, or reproduction techniques or technologies.
- D6.2 Assess the outcome of the prototype product and analyze any issues that need redesigning or refining related to function, construction, or other factors.
- D6.3 Resolve and/or redesign issues with a prototype.

D7.0 Evaluate the prototype to determine if it meets the requirements and objectives.

- D7.1 Create a performance criteria and a quality standard to measure and evaluate a prototype.
- D7.2 Test the functionality and other features of the prototype against the performance criteria and quality standard and evaluate the results.
- D7.3 Identify any redesigning or additional corrections required to improve the overall quality, look, and performance of the prototype model

D10.0 Produce a presentation of the product, product line, system design, or service.

- D10.1 Create a presentation of the design solution (e.g., product, product line, system design, or service) that effectively communicates its features and benefits.
- D10.2 Integrate into the presentation a marketing plan that may include an advertisement, promotion, and packaging/retail strategy using one or more visual communication tools (e.g., graphics, multimedia)

History of Course Development:

The Project Invent curriculum is the product of 5 years of iteration and working directly with 200+ students to invent products for a better world. After years of development, our process has been fine-tuned for high school students to successfully invent technologies that make a difference. At Project Invent, we want every student to be equipped with the skills and mindsets to build a better world. That's why we offer this curriculum for free so students anywhere can go from passion to product. This curriculum is inspired by the work of places like the Stanford d.school, IDEO, and MIT, and curated specifically for what works with young high school students. Using the Project Invent process, students have created everything from Stria, a smart waistband that prevents blind people from veering into dangerous traffic, to Unwind, a stress ball that pulses to a calm heartbeat to prevent anxiety. You can see more project examples on projectinvent.org. Students come into the program passionate about making a positive impact on the world and come out with a real invention to show.

COURSE GOALS AND/OR MAJOR STUDENT OUTCOMES

- Increase diversity in computer science
- Provide students with the tools and support to transform an innovation from concept to reality
- Empower students to view themselves as creators rather than consumers of technology
- Transform maker spaces into impactful spaces driving social change
- Teams will showcase their innovations to top investors and tech executives for a chance at funding by participating in Project Invent Demo Day.
- Prepare students for careers related to computer science that involve the design, development, implementation, maintenance, and management of systems that rely on software programs to satisfy the needs of modern business organizations

COURSE OBJECTIVES

- Students will become proficient in an in demand programming language such as Python
- Students will learn the foundations of web design including HTML, CSS, and Bootstrap
- Students will engage in an online community of software developers through GiTHub and develop a professional portfolio
- Students will develop skills in computer aided design and rapid prototyping using industry standard equipment within the makerspace
- Students will develop empathy through user-centered design processes and community interactions
- Student teams will create a physical technology and travel to Palo Alto to participate in Project Invent Demo Day
- Students will communicate clearly and effectively with teammates, community members, and potential investors
- Students will work productively in teams while integrating cultural and global competence.
- Students will model integrity, ethical leadership, and effective management
- Students will demonstrate creativity and innovation

COURSE OUTLINE

1. Orientation
    a. Why design & why invent? Dive into how technology can create a social impact and how students can play a role in building a better world
2. Needfinding
    a. Understanding the problem. Focus on how to best understand what problems exist around us. Through interviewing, observing, experiencing, and researching, students build layers of empathy and understanding for their target user.
3. Synthesizing & Ideation
    a. Extracting ideas and insights. Focus on taking all of the information students gathered and making observations into actionable insights. Identifying key problems and brainstorm creative solutions.
4. Maker Orientation
    a. Creating WITH technology. Students learn to use industry standard tools for rapid prototyping through short, student led projects in the makerspace.
5. Idea Selection & Prototyping
    a. Choosing a great idea. Students explore the art and science behind selecting the best idea for both their community partner and their team. Introduce multiple tools for selecting their most impactful, most exciting idea.
6. Programming Deep Dive
    a. Creating THE technology. Students will build on their understanding of computer programming from CSD (intro) and AP CSP (concentrator). Students will learn the programming language Python. Students will also use a microcomputer (Raspberry Pi) to interface their software with physical objects. Students may also explore additional programming languages that are relevant to their project needs.
7. Project Planning
    a. Making an idea a reality. Building an idea will be challenging, but that means it's worth doing. Figure out how to build their idea, learn engineering skills, and understand the building process. This segment will emphasize the need for continued prototyping, testing, soliciting user input, and revising.
8. Pitching
    a. Telling a great story. In order to maximize impact, a great product is not enough. Students must also convince others that their idea can change the world. Students craft a compelling story and share what they built to a wider audience. Students will learn HTML, CSS, Javascript, and Bootstrap to make interactive websites for their innovation. They will also create professional quality multimedia to share at the Demo Day event.

TEXTS AND SUPPLEMENTAL INSTRUCTIONAL MATERIALS

Title, Author, Publisher, Edition:

Previously Adopted? ☐ **Yes** ☐ **No (If no, provide information directly below)**

Cost per book	Total Cost	Budget Source

Other:

DIFFERENTIATED INSTRUCTIONAL METHODS AND/OR STRATEGIES

This top objective of this course is to improve equity in computer science. As such, accessibility is at the center of every decision made involving this curriculum. Some of the strategies used include:

Collaborative Learning: Structured and unstructured group work allows students to support each other in problem solving and sense-making. Team projects provide opportunities for students to assign roles and adjust expectations based on student need.

Project Based Learning: We employ a variety of strategies at Da Vinci to explicitly teach collaboration and reflection.

Think-Pair-Share: This protocol is used frequently in class discussions to provide students with adequate processing time, as well as an opportunity to discuss ideas in small groups before sharing ideas with the class

Technology Integration: Agendas will be posted daily online with instructions and links to resources. Text based readings and assignments allow students to use text-to-speech, speech-to-text, text translation, large text, and high contrast accessibility features to access curriculum.

Mindfulness: Mindfulness practices, including guided meditations and “wellness walks” are integrated into daily lesson plans to give students a chance to deescalate and train their attention. Mindfulness practices are deeply engrained in many of the high performing tech industries, and our classroom aims to mirror these environments.

Online tutorials for skill building: Many of our lesson sequences take advantage of online tutorials and learning management systems, allowing students to work at their own pace and take advantage of additional supports as needed. Additional resources will be provided for accelerated learners who wish to develop skills outside the scope of this curriculum, such as learning a different programming language uniquely tailored to their project.

Low Floor-High Ceiling Projects: Our skill building projects that lead up to the culminating team project have expectations that ensure all students can be successful while also providing many avenues for extension so that accelerated learners are appropriately challenged.

Electronic communication, including e-mail and virtual meetings, will be used to facilitate collaboration between students, teachers, families, and industry mentors.

Project Rubrics can be modified to meet the needs of individual students

ASSESSMENT METHODS AND/OR TOOLS

The proposed course is primarily project-based. Even though many academic principles will be taught (and are necessary to the success of the projects), the most important criteria for success in the course will be similar to those required for success in a career. The best way to assess these criteria is to conduct periodic performance reviews of each student using a standardized rubric and provide guidance on how to improve. Performance reviews will be conducted at each grading period and the result of those reviews will form a part of the students' grades.

In addition to the performance review, students will be required to keep a design notebook that they will continually update. The notebook will be used to record information learned during workshops and lectures as well as provide a space for students to reflect on their learning. This will be assessed using a rubric to evaluate professionalism and learning mindset.

Quizzes will be used to assess learning of key conceptual ideas.

Rubrics will be used to assess team and individual projects

Peer Evaluation Assessments using standardized rubrics will be used to evaluate collaboration skills

ASSESSMENT CRITERIA

Collaboration- The ability to be a productive member of diverse teams through strong interpersonal communication, a commitment to shared success, leadership, and initiative.

Critical Thinking -This grade reflects the student's ability to perform higher order thinking skills including evaluation, synthesis and problem solving. Thinking analytically and creatively, using logical reasoning and interpreting information.

Curricular Literacy-The State of California and DVCA have identified key knowledge that students should have when they graduate from high school. This grade reflects the student's knowledge and understanding of the concepts included in Content standards (CTE Model Curriculum Standards)

Learning Mindset-Students build intelligence and skills through effort, practice, solicitation of feedback, revision, and seeking challenge. Students are given feedback and growth in their capacity to "learn how to learn" and monitor their own progress to be successful in tasks, school, and life.

Professionalism-Students show the ability to submit complete and timely work that meets the expectations of academic and professional settings. Additionally, students demonstrate the ability to meet dress and communication standards required for a professional environment.

Written Communication-The ability to effectively communicate knowledge and thinking through writing. This is demonstrated by organizing and structuring ideas, using discipline appropriate language, and utilizing proper English Language conventions.

Oral Communication–The ability to communicate knowledge and thinking through effective oral presentations

HONORS COURSES ONLY
Indicate how this honors course is different from the standard course.