



# Course Catalog

**for Inventive Academy, LLC**

7701 North Lamar Blvd. - Suite 500  
Austin, TX 78752

866.500.2706

# Table of Contents

<b>History</b>	<b>4</b>
<b>Description of available space, facilities, and equipment</b>	<b>4</b>
<b>Officers, Directors, or Trustees</b>	<b>4</b>
<b>Faculty</b>	<b>5</b>
Andrew Siemer	5
Miguel Gonzalez	5
James Shaw	5
<b>Tuition</b>	<b>6</b>
Veterans Scholarship Program	6
Women in Tech Scholarship Program	6
Rainbow Scholarship	7
Underserved Scholarship Program	8
First Responders Scholarship Program	8
<b>2019 Calendar</b>	<b>9</b>
<b>Daily Class Schedule</b>	<b>10</b>
<b>Hours of Operation</b>	<b>10</b>
<b>Admission Enrollment Policy</b>	<b>11</b>
<b>Cancellation Policy</b>	<b>11</b>
<b>Refund Policy</b>	<b>12</b>
<b>Recommended Bandwidth for Meetings, Attendees and Panelists</b>	<b>13</b>
<b>Course</b>	<b>14</b>
Full Stack Immersive with Node.js	14
Objective / Purpose of Course	14
Complete Listing of Subjects	14
Course Completion Time	15
Type of Certificate, Diploma, or Degree Awarded	15
Synopsis of Each Subject Offered	16
FSI-JS-W1-1 - Introduction to the program	16
FSI-JS-W1-2 - Introduction to hypertext markup language (HTML)	16
FSI-JS-W1-3 - (1) PROJECT: Profile site	16

FSI-JS-W2-1 - Introduction to the terminal and the file system	17
FSI-JS-W2-2 - Introduction to source control.	17
FSI-JS-W3 - Introduction to cascading style sheets (CSS)	17
FSI-JS-W4-1 - Introduction to agile project management	18
FSI-JS-W4-2 - Github Hosting	18
FSI-JS-W4-3 - Introduction to the Internet	19
FSI-JS-W4-4 - Building responsive web sites	19
FSI-JS-W4-4 - (2) PROJECT: Responsive profile site	19
FSI-JS-W5 - HTML & CSS exercises	20
FSI-JS-W6-1 - Introduction to Bootstrap	20
FSI-JS-W6-2 - (3) PROJECT: Building an enterprise responsive site	20
FSI-JS-W7 - Introduction to HTML5	21
FSI-JS-W8 - Introduction to JavaScript fundamentals	21
FSI-JS-W9 - JavaScript exercises	22
FSI-JS-W10 - (4) PROJECT: Parking lot simulator & restaurant simulator	22
FSI-JS-W11-1 - JavaScript and the document object model (DOM)	22
FSI-JS-W11-2 - (5) PROJECT: Building a project management board (Trello)	23
FSI-JS-W12-1 - Animations in the DOM	23
FSI-JS-W12-2 - (6) PROJECT: Driving a car around the block	24
FSI-JS-W13 - (7) PROJECT: Parking lot simulator (with UI)	24
FSI-JS-W14-1 - Testing JavaScript with Mocha and Chai	24
FSI-JS-W14-2 - Introduction to ES6 & 7	25
FSI-JS-W14-3 - Charting with D3	25
FSI-JS-W14-4 - (8) PROJECT: Building an administration dashboard	25
FSI-JS-W15-1 - Introduction to jquery	26
FSI-JS-W15-2 - Local build scripts with Gulp	26
FSI-JS-W15-3 - Introduction to SASS	26
FSI-JS-W16-1 - Introduction to node.js	27
FSI-JS-W16-2 - Infrastructure Management with Docker	27
FSI-JS-W17 - (9) PROJECT: Refactoring Trello application to Node	28
FSI-JS-W18-1 - Introduction to data storage	28
FSI-JS-W18-2 - (10) PROJECT: Add data persistence to the Trello application	28
FSI-JS-W19 - Introduction to soft skills and team dynamics	29
FSI-JS-W20 - Building desktop applications with Electron	29
FSI-JS-W21 - Introduction to react.js	29
FSI-JS-W22 - Building mobile applications with React Native	30
FSI-JS-W23 - (11) PROJECT: Convert Trello board to a react application	30
FSI-JS-W24-26 - (12) FINAL PROJECT	31
<b>Requirements for “Certificate of Completion”</b>	<b>31</b>

<b>Requirements for “Inventive Certification”</b>	<b>31</b>
<b>Grading System</b>	<b>31</b>
<b>Project Rubrics</b>	<b>33</b>
Profile Site	33
Responsive Profile Site	34
Building an Enterprise Responsive Site	35
TODO List	37
Simulators - Parking Lot & Restaurant	38
Building a Project Management Board	39
Driving a Car Around the Block	40
Visual Parking Lot Simulator	41
Building an Administration Dashboard	42
Refactoring PM Board to Node.js	43
Project Management Board Data Persistence	44
Project Management Board with React	46
Final Project	47
<b>Presentation Rubric</b>	<b>49</b>
<b>Class Work Rubric</b>	<b>50</b>
<b>Attendance Policy and Make-Up Work</b>	<b>51</b>
<b>Student Code of Conduct</b>	<b>52</b>
Academic Probation	53
Termination	53
<b>Readmission Policy</b>	<b>54</b>
<b>Student Complaint Policy</b>	<b>54</b>
<b>True and Correct Statement</b>	<b>55</b>

## History

As hiring managers, the founders of Inventive Academy, LLC saw a gap in the education and preparation from a majority of job applicants and those coming out of university computer science and related programs. Most had some training on specific computer languages, algorithms, and academic level information but not on how to build products, work together as a team or with other business professionals and lacked soft skill development critical in gaining employment and advancing in one's career. As a result, an early incarnation of the Academy resulted in an effort to deliver video tutorials online through VeteranCoder.com. This helped lay the foundation for delivery of cutting edge curriculum but missed the in-person dialog that was truly needed in order to prepare students for entry into the workforce. For this reason, the founders launched Inventive Academy, LLC to provide immersive in-person training and education that would produce professionals capable of entry and advancement into their respective job fields. Our mission is to change people's lives with technology and that extends to people of all walks of life. However, we are particularly passionate about veterans, women in tech and underrepresented groups and provide scholarships and other company led initiatives to help further those efforts in representation as a whole. The joy of changing people's lives and providing them with a skill set that will last a lifetime is what drives us to continue growing the Inventive Academy, LLC offering.

Approved and Regulated by the Texas Workforce Commission, Career Schools and Colleges, Austin, Texas.

## Description of available space, facilities, and equipment

Inventive Academy, LLC is located on the 5th floor of 7701 North Lamar Dr, Austin, Texas. It is 5500 square feet, accessible for persons with disabilities and includes a classroom and a breakroom. All students have access to building restrooms, elevators and stairs.

## Officers, Directors, or Trustees

Andrew Siemer, Chief Executive Officer  
Miguel Gonzalez, Chief Technical Officer  
James Shaw, Chief Operating Officer

# Faculty

## Andrew Siemer

Mr. Siemer has worked in the field of software engineering as a consultant at many fortune 500 and fortune 50 companies for the past 20 years. He has published several publications on the topic of software engineering, has written many paid articles, and has spoken at several large national conferences. Prior to starting the Inventive Academy, LLC, Mr. Siemer was Chief Architect at Dell in charge of rebuilding the dell.com commerce web site.

### Certifications

Google Cloud Platform Training, GCP (2017)  
Advanced Distributed Systems Design (2011)  
Scrum Master Training (2011)

## Miguel Gonzalez

Miguel has worked as a Software Engineer/Architect/Manager for the past 20 years. In that time he has worked for a variety of companies in many different industries including software consulting, emergency management, media, insurance, real estate, logistics, marketing, education, retail, and e-commerce. He has published a book, written articles about technology, and spoken at several user groups and conferences.

### Certifications

Bachelors of Art in Computer Science, University of Texas at Austin (1998)  
Certified ScrumMaster® (CSM)  
Advanced Distributed Systems Design (2011)  
Google Cloud Platform Training. GCP (2017)

## James Shaw

James has been professionally programming for 30 years, starting in C, C++ and moving to C# in 2000. He received ASP.NET MVP award 3 times and was a founding member of ASP Insiders. He's very entrepreneurial and has built and sold 3 software companies, and been responsible for leading over 100 software engineers and \$30MM revenue.

### Certifications

Bachelors of Science in Engineering, University of London (1984)

Certified ScrumMaster® (2012)

## Tuition

The Full Stack Immersive with Node.js program is offered online and at the physical location of Inventive Academy. The price of the class depends on the type of class you would register for. See below for pricing.

- \$13,000 for the onsite class
- \$11,000 for the Distance Education Program (online)

## Veterans Scholarship Program

At Inventive, we believe in doing our part to help get our returning soldiers back to work and reintegrated into society. We do this best by providing our education capabilities at a reduced cost, or even better, for free. It is our goal to grow this program and be a major part of retraining and employing our veterans!

As part of our commitment to veterans, 50% of our scholarship fund is available for educating and placing veterans into the software engineering field. For every 10th student enrolled in a class we set aside a scholarship slot for a qualified applicant. These available scholarships provide 100% tuition coverage and job placement assistance post graduation.

### Selection Criteria

- **Clear Communication** - The software engineering profession requires that one be able to communicate well with technical and non-technical people at a high level. We look for students who can clearly articulate who they are and why they are interested in applying for a scholarship to an Inventive Academy, LLC bootcamp.
- **Passionate Learners** - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.
- **Ambitious & Driven** - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.

## Women in Tech Scholarship Program

At Inventive, we believe in doing our part to help level the playing field in tech for women by offering our education capabilities at a reduced cost, or even better, for free. It is our goal to grow this program and find ways to promote more equal representation for women in the engineering fields.

As part of our commitment to women, 50% of our scholarship fund is available for educating and placing women into the software engineering field. For every 10th student enrolled in a class we set aside a scholarship slot for a qualified applicant. These available scholarships provide 100% tuition coverage and job placement assistance post graduation.

### **Selection Criteria**

- **Clear Communication** - The software engineering profession requires that one be able to communicate well with technical and non-technical people at a high level. We look for students who can clearly articulate who they are and why they are interested in applying for a scholarship to an Inventive Academy, LLC bootcamp.
- **Passionate Learners** - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.
- **Ambitious & Driven** - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.

## **Rainbow Scholarship**

We are invested in helping bring more diversity to the tech workplace. Our Rainbow Scholarship program is awarded to deserving lesbian, gay, bisexual, transgender, or queer (LGBTQ+) students who want to make a difference in tech.

As part of our commitment to the LGBTQ+ community, 25% of our scholarship fund is available for educating and placing eligible students into the software engineering field. For every 10th student enrolled in a class we set aside a scholarship slot for a qualified applicant. These available scholarships provide 100% tuition coverage and job placement assistance post-graduation.

### **Selection Criteria**

- **Clear Communication** - The software engineering profession requires that one be able to communicate well with technical and non-technical people at a high level. We look for students who can clearly articulate who they are and why they are interested in applying for a scholarship to an Inventive Academy, LLC bootcamp.
- **Passionate Learners** - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.



- **Ambitious & Driven** - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.

## Underserved Scholarship Program

We believe everyone should have the same access to opportunities in tech. Whether you live in a rural community or are poverty-stricken, we are committed to providing you with access to training resources that will impact your life in a positive way.

As part of our commitment to underserved communities, 25% of our scholarship fund is available for educating and placing eligible students into the software engineering field. For every 10th student enrolled in a class we set aside a scholarship slot for a qualified applicant. These available scholarships provide 100% tuition coverage and job placement assistance post-graduation.

### Selection Criteria

- **Clear Communication** - The software engineering profession requires that one be able to communicate well with technical and non-technical people at a high level. We look for students who can clearly articulate who they are and why they are interested in applying for a scholarship to an Inventive Academy, LLC bootcamp.
- **Passionate Learners** - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.
- **Ambitious & Driven** - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.

## First Responders Scholarship Program

At Inventive, we believe in helping people that help our community. First responders definitely fit in that category! We want to help first responders by contributing to their technology education so that they can turn around and help their departments. Or, for folks that have contributed as a first responder while figuring out what their next career steps are, we would like to educate those people into the next steps of their career.

As part of our commitment to first responders, 20% of our scholarship fund is available for educating and placing first responders into the software engineering field. For every 10th student enrolled in a class we set aside a scholarship slot for a qualified applicant. These available scholarships provide 100% tuition coverage and job placement assistance post-graduation.

For eligible scholarship applicants that aren't awarded a full scholarship, you are also eligible to receive a 10% discount off of the program of your choice.

**Selection Criteria**

- Clear Communication - The software engineering profession requires that one be able to communicate well with technical and non-technical people at a high level. We look for students who can clearly articulate who they are and why they are interested in applying for a scholarship to an Inventive Academy, LLC bootcamp.
- Passionate Learners - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.
- Ambitious & Driven - With an ever changing technical landscape, the software engineering career path requires that one be constantly in a state of learning and education. We look for students who have a voracious appetite to learn and willingness to push themselves into the unknown.

## 2019 Calendar

January 28	Full Stack Immersive with Node.js Cohort 1 Start Date
February 18, 2018	Student/Employee holiday
February 25, 2019	Full Stack Immersive with Node.js Cohort 2 Start Date
March 18-21, 2019	Student Spring Break holiday
March 25, 2019	Full Stack Immersive with Node.js Cohort 3 Start Date
April 19, 2019	Student/Employee holiday
April 29, 2019	Full Stack Immersive with Node.js 4 Start Date
May 27, 2019	Student Holiday
June 3, 2019	Full Stack Immersive with Node.js Cohort 5 Start Date
June 24, 2019	Full Stack Immersive with Node.js Cohort 6 Start Date
July 4, 2019	Student/Employee holiday

July 22, 2019	Full Stack Immersive with Node.js Cohort 7 Start Date
August 26, 2019	Full Stack Immersive with Node.js Cohort 8 Start Date
September 2, 2019	Student/Employee holiday
September 23, 2019	Full Stack Immersive with Node.js Cohort 9 Start Date
October 14, 2019	Student/Employee holiday
October 28, 2019	Full Stack Immersive with Node.js Cohort 10 Start Date
October 14, 2019	Student/Employee holiday
Nov 11, 2019	Student/Employee holiday
November 28-29, 2019	Student/Employee holiday
December 21, 2019-January 5, 2020	Student/Employee holiday

An onsite and virtual cohort will be offered for each start date. Each class is 26 weeks of instruction time. See below for start and end dates.

January 28-August 2, 2019	Full Stack Immersive with Node.js Cohort 1
February 25-August 29, 2019	Full Stack Immersive with Node.js Cohort 2
March 25-September 19, 2019	Full Stack Immersive with Node.js Cohort 3
April 29-October 24, 2019	Full Stack Immersive with Node.js Cohort 4
June 3-November 27, 2019	Full Stack Immersive with Node.js Cohort 5
June 24-December 19, 2019	Full Stack Immersive with Node.js Cohort 6
July 22- January 30, 2020	Full Stack Immersive with Node.js Cohort 7
August 26-March 5, 2020	Full Stack Immersive with Node.js Cohort 8
September 23-April 9, 2020	Full Stack Immersive with Node.js Cohort 9
October 28-May 14, 2020	Full Stack Immersive with Node.js Cohort 10

## Daily Class Schedule

Classes at the physical location will be held Monday-Thursday, 6pm-10pm CST for approximately 6 months or 520 contact hours.

A ten minute break will be taken for the last ten minutes of each hour. Tutoring is offered daily, for both classes Monday-Thursday, at 5pm CST.

## Hours of Operation

School Hours of Operation  
5pm - 10pm CST

Office Hours of Operation  
8am - 5pm CST

Distance Education Classes are held online in the evenings 6pm - 10pm CST.

On premise classes are held at 7701 N. Lamar Dr., Suite 500, Austin, TX 78752 in the evenings 6pm - 10pm CST.

## Admission Enrollment Policy

1. MINIMUM AGE: 18 years old
2. Specific Entrance Requirements
  - a. Student should have a Windows or Mac laptop for the first day of class
  - b. Student should be able to touch type at least 40 words per minute
  - c. Student will need to complete the three step online Inventive Academy, LLC application found online here: <https://inventive.io/academy/onsite/full-stack-immersive/apply>
    - i. Student information application
    - ii. Javascript coding exercise
    - iii. Schedule a student interview
  - d. Student should have access to a working internet connection that will support standard video conferencing using the video conferencing package <http://www.zoom.us>.
    - i. The student should be able to participate in video conferencing with a camera and microphone.

## Cancellation Policy

A full refund will be made to any student who cancels the enrollment contract within 72 hours (until midnight of the third day excluding Saturdays, Sundays and legal holidays) after the enrollment contract is signed. A full refund will also be made to any student who cancels enrollment within the student's first three scheduled class days, except that the school may retain not more than \$100 in any administrative fees charged, as well as items of extra expense that are necessary for the portion of the program attended and stated separately on the enrollment agreement.

## Refund Policy

1. Refund computations will be based on scheduled course time of class attendance through the last date of attendance. Leaves of absence, suspensions and school holidays will not be counted as part of the scheduled class attendance.
2. The effective date of termination for refund purposes will be the earliest of the following: a. The last day of attendance, if the student is terminated by the school; b. The date of receipt of written notice from the student; or c. Ten school days following the last date of attendance.
3. If tuition and fees are collected in advance of entrance, and if after expiration of the 72 hour cancellation privilege the student does not enter school, not more than \$100 in any administrative fees charged shall be retained by the school for the entire residence program or synchronous distance education course.
4. If a student enters a residence or synchronous distance education program and withdraws or is otherwise terminated after the cancellation period, the school or college may retain not more than \$100 in any administrative fees charged for the entire program. The minimum refund of the remaining tuition and fees will be the pro rata portion of tuition, fees, and other charges that the number of hours remaining in the portion of the course or program for which the student has been charged after the effective date of termination bears to the total number of hours in the portion of the course or program for which the student has been charged, except that a student may not collect a refund if the student has completed 75 percent or more of the total number of hours in the portion of the program for which the student has been charged on the effective date of termination.
5. Refunds for items of extra expense to the student, such as books, tools, or other supplies are to be handled separately from refund of tuition and other academic fees. The student will not be required to purchase instructional supplies, books and tools until such time as these materials are required. Once these materials are purchased, no refund will be made. For full refunds, the school can withhold costs for these types of items from the refund as long as they were necessary for the portion of the program attended and separately stated in the enrollment agreement. Any such items not required for the portion of the program attended must be included in the refund.
6. A student who withdraws for a reason unrelated to the student's academic status after the 75 percent completion mark and requests a grade at the time of withdrawal shall be given a grade of "incomplete" and permitted to re-enroll in the course or program during the 12-month period following the date the student withdrew without payment of additional tuition for that portion of the course or program.
7. A full refund of all tuition and fees is due and refundable in each of the following cases:
  - a. An enrollee is not accepted by the school;
  - b. If the course of instruction is discontinued by the school and this prevents the student from completing the course; or
  - c. If the student's enrollment was procured as a result of any misrepresentation in advertising, promotional materials of the school, or representations by the owner or representatives of the school.

*A full or partial refund may also be due in other circumstances of program deficiencies or violations of requirements for career schools and colleges.*

## REFUND POLICY FOR STUDENTS CALLED TO ACTIVE MILITARY SERVICE

6. A student of the school or college who withdraws from the school or college as a result of the student being called to active duty in a military service of the United States or the Texas National Guard may elect one of the following options for each program in which the student is enrolled:
- (a) if tuition and fees are collected in advance of the withdrawal, a pro rata refund of any tuition, fees, or other charges paid by the student for the program and a cancellation of any unpaid tuition, fees, or other charges owed by the student for the portion of the program the student does not complete following withdrawal;
  - (b) a grade of incomplete with the designation "withdrawn-military" for the courses in the program, other than courses for which the student has previously received a grade on the student's transcript, and the right to re-enroll in the program, or a substantially equivalent program if that program is no longer available, not later than the first anniversary of the date the student is discharged from active military duty without payment of additional tuition, fees, or other charges for the program other than any previously unpaid balance of the original tuition, fees, and charges for books for the program; or
  - (c) the assignment of an appropriate final grade or credit for the courses in the program, but only if the instructor or instructors of the program determine that the student has:
    - (A) satisfactorily completed at least 90 percent of the required coursework for the program; and
    - (B) demonstrated sufficient mastery of the program material to receive credit for completing the program.
7. Refunds will be totally consummated within 60 days after the effective date of termination.

## Recommended Bandwidth for Meetings, Attendees and Panelists

### Recommended Bandwidth for Meetings and Webinar Panelists:

- For 1:1 video calling: 600kbps (up/down) for HQ video and 1.2 Mbps (up/down) for HD video
- For group video calling: 600kbps/1.2Mbps (up/down) for HQ video. For gallery view: 1.5Mbps/1.5Mbps (up/down).
- For screen sharing only (no video thumbnail): 50-75kbps
- For screen sharing with video thumbnail: 50-150kbps
- For audio VoIP: 60-80kbps

### Recommended Bandwidth for Webinar Attendees:

- For 1:1 video calling: 600kbps (down) for HQ video and 1.2 Mbps (down) for HD video
- For screen sharing only (no video thumbnail): 50-75kbps (down)
- For screen sharing with video thumbnail: 50-150kbps (down)
- For audio VoIP: 60-80kbps (down)

# Course

## Full Stack Immersive with Node.js

### Objective / Purpose of Course

The “Full Stack Immersive with Node.js” program aims to teach students all the skills required to start a job as a web developer. After completing this course, students can apply for jobs such as the following: web developer, front end web developer, software engineer, marketing developer, full stack web developer, full stack software engineer. Students can apply the skills they learn in this course in many different industries: banking, oil and gas, ecommerce, chemical weapons, healthcare, real estate, sports, finance, and gaming.

### Complete Listing of Subjects

Subject Id	Title	Hours
FSI-JS-W1-1	Introduction to the program	4 / 0 / 0 / 4
FSI-JS-W1-2	Introduction to hypertext markup language (HTML)	6 / 6 / 0 / 12
FSI-JS-W1-3	(1) PROJECT: Profile site	0 / 4 / 0 / 4
FSI-JS-W2-1	Introduction to the terminal and the file system	4 / 6 / 0 / 10
FSI-JS-W2-2	Introduction to source control.	4 / 6 / 0 / 10
FSI-JS-W3	Introduction to cascading style sheets (CSS)	8 / 12 / 0 / 20
FSI-JS-W4-1	Introduction to agile project management	2 / 2 / 0 / 4
FSI-JS-W4-2	Github Hosting	2 / 2 / 0 / 4
FSI-JS-W4-3	Introduction to the Internet	2 / 2 / 0 / 4
FSI-JS-W4-4	Building responsive web sites	2 / 2 / 0 / 4
FSI-JS-W4-4	(2) PROJECT: Responsive profile site	0 / 4 / 0 / 4
FSI-JS-W5	HTML & CSS exercises	5 / 15 / 0 / 20
FSI-JS-W6-1	Introduction to Bootstrap	4 / 3 / 0 / 7
FSI-JS-W6-2	(3) PROJECT: Building an enterprise responsive site	4 / 9 / 0 / 13
FSI-JS-W7	Introduction to HTML5	8 / 12 / 0 / 20
FSI-JS-W8	Introduction to JavaScript fundamentals	8 / 12 / 0 / 20
FSI-JS-W9	JavaScript exercises	8 / 12 / 0 / 20
FSI-JS-W10	(4) PROJECT: Parking lot simulator & restaurant simulator	8 / 12 / 0 / 20

FSI-JS-W11-1	JavaScript and the document object model (DOM)	2 / 3 / 0 / 5
FSI-JS-W11-2	(5) PROJECT: Building a project management board (Trello)	6 / 9 / 0 / 15
FSI-JS-W12-1	Animations in the DOM	2 / 4 / 0 / 6
FSI-JS-W12-2	(6) PROJECT: Driving a car around the block	6 / 8 / 0 / 14
FSI-JS-W13	(7) PROJECT: Parking lot simulator (with UI)	8 / 12 / 0 / 20
FSI-JS-W14-1	Testing JavaScript with Mocha and Chai	2 / 3 / 0 / 5
FSI-JS-W14-2	Introduction to ES6 & 7	2 / 3 / 0 / 5
FSI-JS-W14-3	Charting with D3	2 / 3 / 0 / 5
FSI-JS-W14-4	(8) PROJECT: Building an administration dashboard	2 / 3 / 0 / 5
FSI-JS-W15-1	Introduction to jquery	2 / 3 / 0 / 5
FSI-JS-W15-2	Local build scripts with Gulp	4 / 6 / 0 / 10
FSI-JS-W15-3	Introduction to SASS	2 / 3 / 0 / 5
FSI-JS-W16-1	Introduction to node.js	4 / 6 / 0 / 10
FSI-JS-W16-2	Infrastructure Management with Docker	4 / 6 / 0 / 10
FSI-JS-W17	(9) PROJECT: Refactoring Trello application to Node	4 / 16 / 0 / 20
FSI-JS-W18-1	Introduction to data storage	6 / 8 / 0 / 14
FSI-JS-W18-2	(10) PROJECT: Add data persistence to the Trello application	2 / 4 / 0 / 6
FSI-JS-W19	Introduction to soft skills and team dynamics	8 / 12 / 0 / 20
FSI-JS-W20	Building desktop applications with Electron	8 / 12 / 0 / 20
FSI-JS-W21	Introduction to react.js	8 / 12 / 0 / 20
FSI-JS-W22	Building mobile applications with React Native	8 / 12 / 0 / 20
FSI-JS-W23	(11) PROJECT: Convert Trello board to a react application	8 / 12 / 0 / 20
FSI-JS-W24-26	(12) FINAL PROJECT	0 / 60 / 0 / 60
	<b>Total Course Hours:</b>	179 / 341 / 0 / 520

## Course Completion Time

26 weeks

## Type of Certificate, Diploma, or Degree Awarded

Certificate



## Synopsis of Each Subject Offered

### FSI-JS-W1-1 - Introduction to the program

**SUBJECT HOURS:** 4 course time hours (4 hours lecture, 0 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:** Introduction to the program, what is a full stack developer

Introduction to the program will introduce students to the way the class is structured and what it means to be a full stack developer. Students will learn: 1) what is the thayer method or flipped classroom, 2) what is a full stack developer, 3) what are all the technologies that are used as a full stack developer, 4) what types of applications can a full stack developer create

### FSI-JS-W1-2 - Introduction to hypertext markup language (HTML)

**SUBJECT HOURS:** 12 course time hours (6 hours lecture, 6 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:** Introduction to HTML, building your brand, your first website

Introduction to hypertext markup language (HTML) will teach the students all about the basics of the HTML markup language. Students will learn: 1) HTML structure, 2) document hierarchy, 3) layout tags, 4) forms, 5) buttons, 6) working in the browsers debugger, 7) how to test your work

### FSI-JS-W1-3 - (1) PROJECT: Profile site

**SUBJECT HOURS:** 4 course time hours (0 hours lecture, 4 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:**

Build a single page profile site.

## FSI-JS-W2-1 - Introduction to the terminal and the file system

**SUBJECT HOURS:** 10 course time hours (4 hours lecture, 6 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:** Introduction to the terminal and the file system

Introduction to the terminal and the file system will teach students how to navigate and manage their file system using a GUI and the command console. Students will learn: 1) file and folder management in the GUI, 2) the basic command console, 3) installing iterm for a better command line experience, 4) how to navigate your file system from the command console, 5) how to manage files from the command console

## FSI-JS-W2-2 - Introduction to source control.

**SUBJECT HOURS:** 10 course time hours (4 hours lecture, 6 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to the terminal and the file system.

**SUBJECT TOPICS:** Introduction to source control, introduction to Github, introduction to git, introduction to git clients

Introduction to source control will teach students the importance of centrally backing up their files to a source control system. Students will learn: 1) what is source control, 2) the different types of source control, 3) how source control helps when working in teams, 4) branching strategies for supporting larger code bases, 5) how to create a repository, 6) how to get a repository to your local computer, 7) how to commit files to your repository, 8) how to push your work to the source control, 9) how to set up Github, 10) how to merge between branches

## FSI-JS-W3 - Introduction to cascading style sheets (CSS)

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML

**SUBJECT TOPICS:** Introduction to CSS, CSS selectors, CSS basics, style locations, box model, floats, positioning, z-index, CSS variables, CSS debugging, CSS measurements, fonts, colors, buttons, icons, flexbox.

Introduction to cascading style sheets (CSS) will teach students how to make their HTML pages beautiful with CSS. Students will learn: 1) what is CSS, 2) how to apply CSS to their HTML, 3) the basics of CSS, 4) positioning your HTML, 5) flexbox for better layouts, 6) the box model

## FSI-JS-W4-1 - Introduction to agile project management

**SUBJECT HOURS:** 4 course time hours (2 hours lecture, 2 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:** Introduction to agile process, scrum, user stories

Introduction to agile project management will show students several ways to manage software projects. Students will learn: 1) types of software project management, 2) how the Agile project management system works, 3) their role in agile projects, 4) how to manage an agile project using tools such as Trello, 5) what is a user story and how to write a good one, 6) a day in the life of an agile team

## FSI-JS-W4-2 - Github Hosting

**SUBJECT HOURS:** 4 course time hours (2 hours lecture, 2 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:** Hosting your website with Github

Github hosting will show students how to deploy their static websites on the internet for all to see. Students will learn: 1) how to configure a repository for site hosting, 2) how to structure their web project for hosting on Github, 3) how to publish their work to the hosted website

## FSI-JS-W4-3 - Introduction to the Internet

**SUBJECT HOURS:** 4 course time hours (2 hours lecture, 2 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:** Internet basics, the terminal and the internet

Introduction to the Internet will teach students the basic interactions when requesting a web page from the internet. Students will learn: 1) the request lifecycle for web page requests, 2) the breakdown of elements in a web page request, 3) how DNS works, 4) how a web server receives a web request, 5) how domain name registration works

## FSI-JS-W4-4 - Building responsive web sites

**SUBJECT HOURS:** 4 course time hours (2 hours lecture, 2 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, Introduction to CSS

**SUBJECT TOPICS:** Introduction to responsive

Building responsive websites will show students how to build a website that works on mobile, tablets, and desktop environments, without using any fancy responsive frameworks. Students will learn: 1) what is responsive, 2) how to use media queries, 3) setting up breakpoints to target different screen sizes, 4) what is mobile first design, 5) design patterns for mobile first web applications

## FSI-JS-W4-4 - (2) PROJECT: Responsive profile site

**SUBJECT HOURS:** 4 course time hours (0 hours lecture, 4 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:**

Update the one page profile site to be responsive.

## FSI-JS-W5 - HTML & CSS exercises

**SUBJECT HOURS:** 20 course time hours (5 hours lecture, 15 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, Introduction to CSS

**SUBJECT TOPICS:** HTML & CSS exercises

HTML & CSS exercises will provide students with practical problems to be solved using basic HTML and CSS skills. Students will learn: 1) how to set up breakpoints to target a mobile first experience using a real world web application, 2) how to take HTML markup that can't be modified and apply CSS to create custom designs (using the ZenGarden challenge)

## FSI-JS-W6-1 - Introduction to Bootstrap

**SUBJECT HOURS:** 7 course time hours (4 hours lecture, 3 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, Introduction to CSS

**SUBJECT TOPICS:** Introduction to Bootstrap, PROJECT: build jeep.com responsive site

Introduction to Bootstrap will teach students how to quickly build websites that look great and provide a wonderful user experience using the Bootstrap framework. Students will learn: 1) how to install Bootstrap, 2) components provided in Bootstrap, 3) how to use the Bootstrap documentation to quickly learn different component configurations, 4) how to use icons in Bootstrap, 5) how to use different fonts with Bootstrap, 6) browser plugins to make building web pages easier

## FSI-JS-W6-2 - (3) PROJECT: Building an enterprise responsive site

**SUBJECT HOURS:** 13 course time hours (4 hours lecture, 9 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, Introduction to CSS, Introduction to Bootstrap

**SUBJECT TOPICS:**

The "Building an enterprise responsive site" project will show students how to build a real enterprise web page layout using HTML, CSS, and Bootstrap. Students will learn: 1) how to debug existing sites for ideas, 2) how to leverage Bootstrap to build complex functionality, 3) how to build the web page with a mobile first mindset, 4) how to achieve pixel perfect layouts when given an exact mockup to follow

## FSI-JS-W7 - Introduction to HTML5

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML

**SUBJECT TOPICS:** Introduction to HTML5, audio, video, SVG, storage options, putting it all together

Introduction to HTML5 will teach students the new features in the HTML standard that make building media rich, mobile enabled, and SEO friendly web pages much easier. Students will learn: 1) HTML5 features, 2) semantic page layout concepts, 3) data attributes, 4) audio, 5) video, 6) persistent local storage options, 7) forms 2.0 elements

## FSI-JS-W8 - Introduction to JavaScript fundamentals

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, Introduction to CSS

**SUBJECT TOPICS:** Introduction to JavaScript, data types, operators, functions, objects and this, build a calculator, program flow, events and timers

Introduction to JavaScript fundamentals will teach students how to build applications with JavaScript. Students will learn: 1) what is JavaScript, 2) how to declare variables, 3) data types, operators, and functions, 4) program flow, 5) events and timers, 6) program structure

## FSI-JS-W9 - JavaScript exercises

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript

**SUBJECT TOPICS:** JavaScript exercises

JavaScript exercises will reinforce what has been taught in the fundamentals class. Students will learn: 1) how to declare different data types, 2) find an element in an array, 3) perform math using JavaScript, 4) how to manipulate strings, 5) how to alphabetize a string, 6) splitting a string, 7) create an algorithm to find the longest word, 8) convert money to coins

## FSI-JS-W10 - (4) PROJECT: Parking lot simulator & restaurant simulator

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript

**SUBJECT TOPICS:**

The "Parking lot simulator & restaurant simulator" projects will teach students how to structure a larger application that needs to maintain local application data. Students will learn: 1) application structure, 2) how to work with arrays of data, 3) how to work with timers, 4) breaking down large problems into small problems, 5) writing functions that do one thing

## FSI-JS-W11-1 - JavaScript and the document object model (DOM)

**SUBJECT HOURS:** 5 course time hours (2 hours lecture, 3 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript

**SUBJECT TOPICS:** Introduction to the Document Object Model (DOM), Window, Modernizr, Document, DOM Selections

JavaScript and the document object model (DOM) will teach students how to merge HTML and JavaScript to create applications with rich user experiences. Students will learn: 1) document object model hierarchy, 2) how to manipulate the DOM, 3) how to build dynamic user interfaces, 4) how to store data in element attributes, 5) how to insert elements into the DOM

## FSI-JS-W11-2 - (5) PROJECT: Building a project management board (Trello)

**SUBJECT HOURS:** 15 course time hours (6 hours lecture, 9 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, introduction to CSS, introduction to JavaScript, JavaScript and the DOM

**SUBJECT TOPICS:**

The "Building a project management board (Trello)" project will teach students how to build an entire web application with a dynamically created user interface. Students will learn: 1) how to build an entire UI programmatically with JavaScript, 2) how to manipulate the DOM using UI elements, 3) how to manage data in your application

## FSI-JS-W12-1 - Animations in the DOM

**SUBJECT HOURS:** 6 course time hours (2 hours lecture, 4 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript, introduction to CSS

**SUBJECT TOPICS:** Animation with JavaScript, CSS transitions, CSS animation

Animations in the DOM will teach students a few different approaches to animating elements in their HTML pages. Students will learn: 1) how to animate HTML elements using JavaScript, 2) how to animate HTML elements with CSS transforms, 3) how to animate HTML elements with CSS animations



## FSI-JS-W12-2 - (6) PROJECT: Driving a car around the block

**SUBJECT HOURS:** 14 course time hours (6 hours lecture, 8 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript, introduction to CSS

**SUBJECT TOPICS:**

Driving a car around the block will teach students how to write a time based application that continuously updates the state of their application. Students will learn: 1) collision detection, 2) how to work with a game loop, 3) maintaining complex application state, 4) controlling logic and application flow, 5) simplifying application structure

## FSI-JS-W13 - (7) PROJECT: Parking lot simulator (with UI)

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript, introduction to CSS

**SUBJECT TOPICS:**

The "Parking lot simulator (with UI)" project will teach students how to manage complex simulation applications. Students will learn: 1) collision detection, 2) how to work with a game loop, 3) maintaining state for many objects in a single world, 4) combining animation types on a single object

## FSI-JS-W14-1 - Testing JavaScript with Mocha and Chai

**SUBJECT HOURS:** 5 course time hours (2 hours lecture, 3 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript

**SUBJECT TOPICS:** Testing JavaScript with Mocha and Chai

Testing JavaScript with Mocha and Chai will teach students how to write unit and functional tests for their application. Students will learn: 1) what is a software test, 2) different types of tests, 3) what makes a good

test, 4) how to mock and stub, 5) running a test suite on your application, 6) appropriate structure for a good test, 7) software testing patterns

## FSI-JS-W14-2 - Introduction to ES6 & 7

**SUBJECT HOURS:** 5 course time hours (2 hours lecture, 3 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript

**SUBJECT TOPICS:** Introduction to ES6 & 7

Introduction to ES6 & 7 will teach students about the latest and greatest JavaScript. Students will learn: 1) declaring variables, 2) constructors, 3) new array functions, 4) managing scope, 5) default parameters, 6) template literals, 7) multi-line strings, 8) enhanced object literals, 9) arrow functions, 10) promises

## FSI-JS-W14-3 - Charting with D3

**SUBJECT HOURS:** 5 course time hours (2 hours lecture, 3 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, introduction to JavaScript

**SUBJECT TOPICS:** Charting with D3

Charting with D3 will teach students how to build reports using JavaScript and D3.js. Students will learn: 1) what is D3.js, 2) how to build SVG reports, 3) using dimple.js to simplify D3

## FSI-JS-W14-4 - (8) PROJECT: Building an administration dashboard

**SUBJECT HOURS:** 5 course time hours (2 hours lecture, 3 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, introduction to CSS, introduction to JavaScript, JavaScript and the DOM

**SUBJECT TOPICS:**

The "Building an administration dashboard" project will teach students how to build an administration dashboard with the D3 charting library.

## FSI-JS-W15-1 - Introduction to jquery

**SUBJECT HOURS:** 5 course time hours (2 hours lecture, 3 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript

**SUBJECT TOPICS:** Introduction to jquery

Introduction to jquery will show students how to work with the DOM in a more streamlined manner. Students will learn: 1) how to install jquery, 2) how to use jquery to simplify DOM manipulations and queries, 3) how to animate DOM elements with jquery

## FSI-JS-W15-2 - Local build scripts with Gulp

**SUBJECT HOURS:** 10 course time hours (4 hours lecture, 6 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript

**SUBJECT TOPICS:** Local build scripts with Gulp

Local build scripts with Gulp will show students how to chain together local build steps such as transpilation, css compilation, unit test automation, and more. Students will learn: 1) ES6 transpilation, 2) SASS compilation, 3) unit test automation, 4) static build asset creation, 5) local web server configuration, 6) image optimization

## FSI-JS-W15-3 - Introduction to SASS

**SUBJECT HOURS:** 5 course time hours (2 hours lecture, 3 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to CSS

**SUBJECT TOPICS:** CSS compilation with SASS

Introduction to SASS will show students how to better manage their styling assets on large sites. Students will learn: 1) how to structure your CSS assets with partials, 2) variables in your CSS, 3) nesting styles, 4) operators

## FSI-JS-W16-1 - Introduction to node.js

**SUBJECT HOURS:** 10 course time hours (4 hours lecture, 6 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript

**SUBJECT TOPICS:** Introduction to node.js, building APIs with restify.js, command line applications with commander.js, building server side web applications with express.js

Introduction to node.js will show students how they can use JavaScript outside of the browser environment for web server interactions and more. Students will learn: 1) an introduction to node.js and how to set up a simple node server, 2) how to use express.js to create a full web application, 3) how to use restify.js to create a web api, 4) how to use commander.js to create command line applications

## FSI-JS-W16-2 - Infrastructure Management with Docker

**SUBJECT HOURS:** 10 course time hours (4 hours lecture, 6 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to the terminal and the file system

**SUBJECT TOPICS:** Introduction to docker, docker compose, creating images, docker registry

Infrastructure management with Docker will teach students how to containerize their web applications. Students will learn: 1) how to work with Docker images, 2) how to start a docker image, 3) how to ssh into a container, 4) how to deploy your application into a container, 5) how to work with a container registry

## FSI-JS-W17 - (9) PROJECT: Refactoring Trello application to Node

**SUBJECT HOURS:** 20 course time hours (4 hours lecture, 16 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:**

The "Refactoring Trello application to Node" project will teach students how to host their web site into an express and restify application. Students will learn: 1) how to migrate a static site to a dynamic web application, 2) how to drive their application with an API, 3) how to host their application in a Docker container

## FSI-JS-W18-1 - Introduction to data storage

**SUBJECT HOURS:** 14 course time hours (6 hours lecture, 8 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript, introduction to node.js

**SUBJECT TOPICS:** Mongoose and MongoDB, RoboMongo, sqlizer and MySQL, MySQL Workbench

Introduction to data storage will show students how to persist their application data to a durable storage tool. Students will learn: 1) how to use a document data store (MongoDB), 2) how to use a relational data store (MySQL), 3) how to use RoboMongo to manage your MongoDB data, 4) how to use MySQL Workbench to manage your MySQL data

## FSI-JS-W18-2 - (10) PROJECT: Add data persistence to the Trello application

**SUBJECT HOURS:** 6 course time hours (2 hours lecture, 4 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to JavaScript, introduction to node.js

**SUBJECT TOPICS:**

The "Add data persistence to the Trello application" project will show students how to refactor an existing app to manage persisted data. Students will learn: 1) how to update an application to store its data, 2) how to

update an application to read its initial data from a data store, 3) how to perform standard CRUD operations (create, read, update, delete)

## FSI-JS-W19 - Introduction to soft skills and team dynamics

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** None

**SUBJECT TOPICS:** Working in teams, personality profiling, what is the software development life cycle, modeling on the whiteboard, software estimation, branching strategies, advanced git, pull requests in Github, resume writing, online profile readiness

Introduction to soft skills and team dynamics will teach students soft skills required to work in teams and get a job as a software developer. Students will learn: 1) how to work in teams, 2) personality profiling, 3) what is the software development life cycle, 4) modeling on the whiteboard, 5) software estimation, 6) branching strategies, 7) advanced git, 8) pull requests in Github, 9) resume writing, 10) online profile readiness

## FSI-JS-W20 - Building desktop applications with Electron

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, Introduction to CSS, Introduction to JavaScript

**SUBJECT TOPICS:** Introduction to Electron

Building desktop applications with Electron will teach students how to apply their HTML, CSS, and JavaScript skills to building desktop applications. Students will learn: 1) how to build a simple desktop app, 2) how to open dialogs, 3) how to manage menus, 4) how to manage pages in your application, 5) how to manage data

## FSI-JS-W21 - Introduction to react.js

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, Introduction to CSS, Introduction to JavaScript

**SUBJECT TOPICS:** Introduction to react.js

Introduction to react.js will teach students how to build a component based user interface. Students will learn: 1) how to install react, 2) how to build a simple react application, 3) how to manage state, 4) how to create components

## FSI-JS-W22 - Building mobile applications with React Native

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to HTML, Introduction to CSS, Introduction to JavaScript, Introduction to react.js

**SUBJECT TOPICS:** Introduction to React Native

Building mobile applications with React Native will teach students how to leverage their react skills for mobile application development. Students will learn: 1) how to install react native, 2) how to set up a project, 3) how to build the application, 4) how to test the application, 5) how to deploy the application

## FSI-JS-W23 - (11) PROJECT: Convert Trello board to a react application

**SUBJECT HOURS:** 20 course time hours (8 hours lecture, 12 hours lab, 0 hours externship)

**PREREQUISITES:** Introduction to react.js

**SUBJECT TOPICS:**

The "Convert Trello board to a react application" project will teach students how to apply React to an existing application. Students will learn: 1) how to install and configure react in an existing project, 2) how to componentize existing application interfaces, 3) how to manage state and data

## FSI-JS-W24-26 - (12) FINAL PROJECT

**SUBJECT HOURS:** 60 course time hours (0 hours lecture, 60 hours lab, 0 hours externship)

**PREREQUISITES:** FINAL PROJECT

**SUBJECT TOPICS:**

The Final Project is a team based project that is built from scratch. This greenfield project is chosen and prepped a month ahead of time per each team to ensure the project can be completed in time. This project is teacher guided, but student led.

### Requirements for “Certificate of Completion”

To complete an Inventive course you must have performed the following items successfully between the start and end dates of the course:

- Not miss more than 10 days of class
- Complete all classroom assignments
- Complete all projects
- A total score of 70% or greater

### Requirements for “Inventive Certification”

To be certified by Inventive you must have performed the following items successfully between the start and end dates of the course:

- Not miss more than 5 days of class
- Complete all classroom assignments
- Complete all projects
- A total score of 85% or greater

### Grading System

The following scoring system will be used to determine the grade of each project. The sum of all project scores, in class work, and attendance shall determine the class grade.

Numeric Grade	Letter Grade	Grade Points
90 - 100	A	4.00



80 -89	B	3.00
70 - 79	C	2.00
60 - 69	D	1.00
Below 60	F	0.00
Incomplete	I	0.00
Withdrawal	W	0.00

# Project Rubrics

<b>Rubric Table</b>	Use this rubric when grading:	
<b>1</b>	<b>Profile Site</b>	
Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Layout matches the provided wireframe</li> <li>● All required aspects are present in the page</li> <li>● Fonts are appropriately sized and spaced</li> <li>● All links work as expected</li> </ul>
<b>Grammar, Spelling</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● There are no grammatical errors</li> <li>● There are no spelling errors</li> </ul>
<b>Design</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Elements are appropriately spaced</li> <li>● Appropriate color pallet is chosen</li> <li>● Profile image is appropriate for page</li> </ul>
<b>Code</b>	<b>8</b>	<ul style="list-style-type: none"> <li>● Spacing of code aspects are correct</li> <li>● Tabbing of code elements are correct</li> <li>● All HTML elements are appropriately closed</li> <li>● Appropriate HTML elements used for page layout</li> <li>● Application structure is well organized</li> </ul>
<b>Images</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Appropriately sized image is used</li> <li>● Image is appropriate for purpose</li> </ul>
<b>Purpose</b>	<b>26</b>	<ul style="list-style-type: none"> <li>● Accurately reflects the purpose of a profile site</li> </ul>

**Total Possible Points**                      **50**

<b>Rubric Table</b> <span style="font-size: 2em; font-weight: bold; margin-left: 20px;">2</span>		Use this rubric when grading:  <b>Responsive Profile Site</b>
Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	<b>4</b>	<ul style="list-style-type: none"> <li>Layout matches the provided wireframe</li> <li>All required aspects are present in the page</li> <li>Fonts are appropriately sized and spaced</li> <li>All links work as expected</li> </ul>
<b>Grammar, Spelling</b>	<b>4</b>	<ul style="list-style-type: none"> <li>There are no grammatical errors</li> <li>There are no spelling errors</li> </ul>
<b>Design</b>	<b>4</b>	<ul style="list-style-type: none"> <li>Elements are appropriately spaced</li> <li>Appropriate color pallet is chosen</li> <li>Profile image is appropriate for page</li> </ul>
<b>Code</b>	<b>8</b>	<ul style="list-style-type: none"> <li>Spacing of code aspects are correct</li> <li>Tabbing of code elements are correct</li> <li>All HTML elements are appropriately closed</li> <li>Appropriate HTML elements used for page layout</li> <li>Application structure is well organized</li> </ul>
<b>Images</b>	<b>4</b>	<ul style="list-style-type: none"> <li>Appropriately sized image is used</li> <li>Image is appropriate for purpose</li> </ul>
<b>Purpose</b>	<b>26</b>	<ul style="list-style-type: none"> <li>Accurately reflects the purpose of a profile site</li> <li>Renders appropriately in a desktop viewport</li> <li>Renders appropriately in a tablet viewport</li> <li>Renders appropriately in a mobile viewport</li> </ul>

**Total Possible Points** **50**

<b>Rubric Table</b>		Use this rubric when grading:
<b>3</b>		<b>Building an Enterprise Responsive Site</b>
Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Layout matches the provided wireframe</li> <li>● All required aspects are present in the page</li> <li>● Fonts are appropriately sized and spaced</li> <li>● All links work as expected</li> </ul>
<b>Grammar, Spelling</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● There are no grammatical errors</li> <li>● There are no spelling errors</li> </ul>
<b>Design</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Elements are appropriately spaced</li> <li>● Appropriate color pallet is chosen</li> <li>● Profile image is appropriate for page</li> </ul>
<b>Code</b>	<b>8</b>	<ul style="list-style-type: none"> <li>● Spacing of code aspects are correct</li> <li>● Tabbing of code elements are correct</li> <li>● All HTML elements are appropriately closed</li> <li>● Appropriate HTML elements used for page layout</li> <li>● Application structure is well organized</li> </ul>
<b>Images</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Appropriately sized image is used</li> <li>● Image is appropriate for purpose</li> </ul>
<b>Purpose</b>	<b>26</b>	<ul style="list-style-type: none"> <li>● Matches jeep.com homepage in all viewports</li> <li>● Matches Wrangler homepage in all viewports</li> </ul>
<b>Total Possible Points</b>		<b>50</b>



<b>Rubric Table</b>  <b>4</b>	Use this rubric when grading:  <b>TODO List</b>	
Rubric Category	Possible Points	Criteria for Maximum Points
<b>Presentation</b>	<b>5</b>	<ul style="list-style-type: none"> <li>● UI looks and works as defined</li> </ul>
<b>Grammar, Spelling</b>	<b>5</b>	<ul style="list-style-type: none"> <li>● There are no grammatical errors</li> <li>● There are no spelling errors</li> </ul>
<b>Code</b>	<b>10</b>	<ul style="list-style-type: none"> <li>● Spacing of code elements are correct</li> <li>● Tabbing of code elements are correct</li> <li>● Application structure is well organized</li> <li>● All HTML elements are appropriately closed</li> </ul>
<b>Purpose</b>	<b>30</b>	<ul style="list-style-type: none"> <li>● Text field to capture new todo</li> <li>● Button click to persist the new todo</li> <li>● Todo is persisted to a local storage mechanism</li> <li>● Todo is displayed in a list of todo items</li> <li>● Can delete a todo item</li> </ul>

**Total Possible Points**                      **50**

<p style="margin: 0;"><b>Rubric Table</b></p> <p style="font-size: 2em; margin: 0;"><b>5</b></p>	<p>Use this rubric when grading:</p> <h2 style="margin: 0;">Simulators - Parking Lot &amp; Restaurant</h2>
--	--

Rubric Category	Possible Points	Criteria for Maximum Points
<b>Presentation</b>	<b>5</b>	<ul style="list-style-type: none"> <li>All required data metrics are present in the console</li> </ul>
<b>Grammar, Spelling</b>	<b>5</b>	<ul style="list-style-type: none"> <li>There are no grammatical errors</li> <li>There are no spelling errors</li> </ul>
<b>Code</b>	<b>10</b>	<ul style="list-style-type: none"> <li>Spacing of code aspects are correct</li> <li>Tabbing of code elements are correct</li> <li>Application structure is well organized</li> </ul>
<b>Purpose</b>	<b>30</b>	<ul style="list-style-type: none"> <li>All cars parked in the parking lot</li> <li>All customers seated at the restaurant</li> <li>Events used to manage workflow</li> <li>Appropriate data structures utilized</li> </ul>

**Total Possible Points**                      **50**

<b>Rubric Table</b>  <span style="font-size: 2em; font-weight: bold;">6</span>		Use this rubric when grading:  <b>Building a Project Management Board</b>
Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Layout matches the provided wireframe</li> <li>● All required aspects are present in the page</li> </ul>
<b>Grammar, Spelling</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● There are no grammatical errors</li> <li>● There are no spelling errors</li> </ul>
<b>Design</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Elements are appropriately spaced</li> <li>● Appropriate color pallet is chosen</li> </ul>
<b>Code</b>	<b>9</b>	<ul style="list-style-type: none"> <li>● Spacing of code aspects are correct</li> <li>● Tabbing of code elements are correct</li> <li>● All HTML elements are appropriately closed</li> <li>● Appropriate HTML elements used for page layout</li> <li>● Application structure is well organized</li> </ul>
<b>Images</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Appropriately sized images are used</li> <li>● Images are appropriate for purpose</li> </ul>
<b>Purpose</b>	<b>25</b>	<ul style="list-style-type: none"> <li>● Able to name the board</li> <li>● Able to add a swimlane</li> <li>● Able to add a card</li> <li>● Able to move a card to a different swimlane</li> <li>● Able to change the order of a card</li> <li>● Able to reorder the swimlanes</li> <li>● Able to edit a card</li> <li>● Able to edit a swimlane name</li> <li>● Able to delete a card</li> <li>● Able to delete a swimlane (and all cards)</li> <li>● Able to edit your board</li> </ul>



**Total Possible Points**

**50**

**Rubric Table**

**7**

Use this rubric when grading:

**Driving a Car Around the Block**

Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	4	<ul style="list-style-type: none"> <li>Layout matches the provided wireframe</li> <li>All required aspects are present in the page</li> <li>Fonts are appropriately sized and spaced</li> <li>All links work as expected</li> </ul>
<b>Grammar, Spelling</b>	4	<ul style="list-style-type: none"> <li>There are no grammatical errors</li> <li>There are no spelling errors</li> </ul>
<b>Design</b>	4	<ul style="list-style-type: none"> <li>Elements are appropriately spaced</li> <li>Appropriate color pallet is chosen</li> </ul>
<b>Code</b>	9	<ul style="list-style-type: none"> <li>Spacing of code aspects are correct</li> <li>Tabbing of code elements are correct</li> <li>All HTML elements are appropriately closed</li> <li>Appropriate HTML elements used for page layout</li> <li>Application structure is well organized</li> </ul>
<b>Images</b>	4	<ul style="list-style-type: none"> <li>Appropriately sized images are used</li> <li>Images are appropriate for purpose</li> </ul>
<b>Purpose</b>	25	<ul style="list-style-type: none"> <li>Ability to drive a car from one intersection to another</li> <li>Ability to make a turn at the appropriate time to the next intersection</li> <li>Ability to drive a car around the block continuously</li> <li>Ability for one car to pass a slower car</li> <li>Ability for passing car to drive faster</li> </ul>

		<ul style="list-style-type: none"> <li>Ability for passing car to not pass while in an intersection</li> </ul>
--	--	--

**Total Possible Points**                      **50**

<b>Rubric Table</b>  <span style="font-size: 2em; font-weight: bold;">8</span>	<p>Use this rubric when grading:</p> <h2 style="text-align: center;">Visual Parking Lot Simulator</h2>
--	--

Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	<b>4</b>	<ul style="list-style-type: none"> <li>Layout matches the provided wireframe</li> <li>All required aspects are present in the page</li> <li>Fonts are appropriately sized and spaced</li> <li>All links work as expected</li> </ul>
<b>Grammar, Spelling</b>	<b>4</b>	<ul style="list-style-type: none"> <li>There are no grammatical errors</li> <li>There are no spelling errors</li> </ul>
<b>Design</b>	<b>4</b>	<ul style="list-style-type: none"> <li>Elements are appropriately spaced</li> <li>Appropriate color pallet is chosen</li> </ul>
<b>Code</b>	<b>9</b>	<ul style="list-style-type: none"> <li>Spacing of code aspects are correct</li> <li>Tabbing of code elements are correct</li> <li>All HTML elements are appropriately closed</li> <li>Appropriate HTML elements used for page layout</li> <li>Application structure is well organized</li> </ul>
<b>Images</b>	<b>4</b>	<ul style="list-style-type: none"> <li>Appropriately sized images are used</li> <li>Images are appropriate for purpose</li> </ul>
<b>Purpose</b>	<b>25</b>	<ul style="list-style-type: none"> <li>Ability to park a car</li> <li>Ability to unpark a car</li> <li>Ability for a car to leave the parking lot</li> </ul>

		<ul style="list-style-type: none"> <li>• Ability to continuously park all cars</li> <li>• Ability to wait until a parking spot is free</li> <li>• Ability for a car to navigate the parking lot</li> <li>• All cars have been parked</li> <li>• All cars have been unparked</li> </ul>
--	--	--

**Total Possible Points**

**50**

<p><b>Rubric Table</b></p> <p style="font-size: 2em; font-weight: bold;">9</p>	<p>Use this rubric when grading:</p> <h2 style="text-align: center;">Building an Administration Dashboard</h2>
--	--

Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	<b>4</b>	<ul style="list-style-type: none"> <li>• Layout matches the provided wireframe</li> <li>• All required aspects are present in the page</li> <li>• Fonts are appropriately sized and spaced</li> <li>• All links work as expected</li> </ul>
<b>Grammar, Spelling</b>	<b>4</b>	<ul style="list-style-type: none"> <li>• There are no grammatical errors</li> <li>• There are no spelling errors</li> </ul>
<b>Design</b>	<b>4</b>	<ul style="list-style-type: none"> <li>• Elements are appropriately spaced</li> <li>• Appropriate color pallet is chosen</li> </ul>
<b>Code</b>	<b>9</b>	<ul style="list-style-type: none"> <li>• Spacing of code aspects are correct</li> <li>• Tabbing of code elements are correct</li> <li>• All HTML elements are appropriately closed</li> <li>• Appropriate HTML elements used for page layout</li> <li>• Application structure is well organized</li> </ul>
<b>Images</b>	<b>4</b>	<ul style="list-style-type: none"> <li>• Appropriately sized images are used</li> <li>• Images are appropriate for purpose</li> </ul>



<b>Purpose</b>	<b>25</b>	<ul style="list-style-type: none"> <li>• Node app created with Restify</li> <li>• In memory data store created</li> <li>• Endpoints created for board management</li> <li>• Endpoints created for swimlane management</li> <li>• Endpoints created for card management</li> <li>• Can reload the board from scratch</li> </ul>
----------------	-----------	--

**Total Possible Points** **50**

<b>Rubric Table</b>  <span style="font-size: 2em; font-weight: bold;">11</span>		Use this rubric when grading:  <h2 style="text-align: center;">Project Management Board Data Persistence</h2>
Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	<b>4</b>	<ul style="list-style-type: none"> <li>• Layout matches the provided wireframe</li> <li>• All required aspects are present in the page</li> </ul>
<b>Grammar, Spelling</b>	<b>4</b>	<ul style="list-style-type: none"> <li>• There are no grammatical errors</li> <li>• There are no spelling errors</li> </ul>
<b>Design</b>	<b>4</b>	<ul style="list-style-type: none"> <li>• Elements are appropriately spaced</li> <li>• Appropriate color pallet is chosen</li> </ul>
<b>Code</b>	<b>9</b>	<ul style="list-style-type: none"> <li>• Spacing of code aspects are correct</li> <li>• Tabbing of code elements are correct</li> <li>• All HTML elements are appropriately closed</li> <li>• Appropriate HTML elements used for page layout</li> <li>• Application structure is well organized</li> </ul>
<b>Images</b>	<b>4</b>	<ul style="list-style-type: none"> <li>• Appropriately sized images are used</li> <li>• Images are appropriate for purpose</li> </ul>

<b>Purpose</b>	<b>25</b>	<ul style="list-style-type: none"> <li>● Refactored all node endpoints to use sequelize data persistence over MySQL</li> <li>● Able to save board information</li> <li>● Able to save swimlane information</li> <li>● Able to save card information</li> </ul>
<b>Total Possible Points</b>	<b>50</b>	

<b>Rubric Table</b>  <span style="font-size: 2em; font-weight: bold;">12</span>		Use this rubric when grading:  <b>Project Management Board with React</b>
Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Layout matches the provided wireframe</li> <li>● All required aspects are present in the page</li> </ul>
<b>Grammar, Spelling</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● There are no grammatical errors</li> <li>● There are no spelling errors</li> </ul>
<b>Design</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Elements are appropriately spaced</li> <li>● Appropriate color pallet is chosen</li> </ul>
<b>Code</b>	<b>9</b>	<ul style="list-style-type: none"> <li>● Spacing of code aspects are correct</li> <li>● Tabbing of code elements are correct</li> <li>● All HTML elements are appropriately closed</li> <li>● Appropriate HTML elements used for page layout</li> <li>● Application structure is well organized</li> </ul>
<b>Images</b>	<b>4</b>	<ul style="list-style-type: none"> <li>● Appropriately sized images are used</li> <li>● Images are appropriate for purpose</li> </ul>
<b>Purpose</b>	<b>25</b>	<ul style="list-style-type: none"> <li>● Created a React app</li> <li>● Created components for board</li> <li>● Created components for swimlane</li> <li>● Created components for card</li> </ul>

**Total Possible Points** **50**

<b>Rubric Table</b>  <span style="font-size: 2em; font-weight: bold;">13</span>		Use this rubric when grading:  <b>Final Project</b>
Rubric Category	Possible Points	Criteria for Maximum Points
<b>Display Format and Presentation</b>	<b>5</b>	<ul style="list-style-type: none"> <li>● Performed ideation exercise on whiteboard</li> <li>● Created wireframes</li> <li>● Followed wireframes during project creation</li> </ul>
<b>Grammar, Spelling</b>	<b>5</b>	<ul style="list-style-type: none"> <li>● There are no grammatical errors</li> <li>● There are no spelling errors</li> </ul>
<b>Design</b>	<b>5</b>	<ul style="list-style-type: none"> <li>● Elements are appropriately spaced</li> <li>● Appropriate color pallet is chosen</li> </ul>
<b>Code</b>	<b>20</b>	<ul style="list-style-type: none"> <li>● Spacing of code aspects are correct</li> <li>● Tabbing of code elements are correct</li> <li>● All HTML elements are appropriately closed</li> <li>● Appropriate HTML elements used for page layout</li> <li>● Application structure is well organized</li> </ul>
<b>Images</b>	<b>5</b>	<ul style="list-style-type: none"> <li>● Appropriately sized images are used</li> <li>● Images are appropriate for purpose</li> </ul>
<b>Planning</b>	<b>10</b>	<ul style="list-style-type: none"> <li>● Project broken down into appropriately sized user stories</li> </ul>
<b>Purpose</b>	<b>50</b>	<ul style="list-style-type: none"> <li>● Used appropriate technologies for project</li> <li>● Project scope fits timeline for completion</li> <li>● Application works as planned</li> </ul>

**Total Possible Points** **100**





# Presentation Rubric

4 - Excellent	3 - Good	2 - Fair	1 - Needs Improvement	
<ul style="list-style-type: none"> <li>• Holds attention of entire audience with the use of direct eye contact, seldom looking at notes</li> <li>• Speaks with fluctuation in volume and inflection to maintain audience interest and emphasize key points</li> </ul>	<ul style="list-style-type: none"> <li>• Consistent use of direct eye contact with audience, but still returns to notes</li> <li>• Speaks with satisfactory variation of volume and inflection</li> </ul>	<ul style="list-style-type: none"> <li>• Displays minimal eye contact with audience, while reading mostly from the notes</li> <li>• Speaks in uneven volume with little or no inflection</li> </ul>	<ul style="list-style-type: none"> <li>• Holds no eye contact with audience, as entire report is read from notes</li> <li>• Speaks in low volume and/ or monotonous tone, which causes audience to disengage</li> </ul>	<p><b>Delivery</b></p>
<ul style="list-style-type: none"> <li>• Demonstrates full knowledge by answering all class questions with explanations and elaboration</li> <li>• Provides clear purpose and subject; pertinent examples, facts, and/or statistics; supports conclusions/ideas with evidence</li> </ul>	<ul style="list-style-type: none"> <li>• Is at ease with expected answers to all questions, without elaboration</li> <li>• Has somewhat clear purpose and subject; some examples, facts, and/or statistics that support the subject; includes some data or evidence that supports conclusions</li> </ul>	<ul style="list-style-type: none"> <li>• Is uncomfortable with information and is able to answer only rudimentary questions</li> <li>• Attempts to define purpose and subject; provides weak examples, facts, and/ or statistics, which do not adequately support the subject; includes very thin data or evidence</li> </ul>	<ul style="list-style-type: none"> <li>• Does not have grasp of information and cannot answer questions about subject</li> <li>• Does not clearly define subject and purpose; provides weak or no support of subject; gives insufficient support for ideas or conclusions</li> </ul>	<p><b>Content / Organization</b></p>
<ul style="list-style-type: none"> <li>• Demonstrates strong enthusiasm about topic during entire presentation</li> <li>• Significantly increases audience understanding and knowledge of topic; convinces an audience to recognize the</li> </ul>	<ul style="list-style-type: none"> <li>• Shows some enthusiastic feelings about topic</li> <li>• Raises audience understanding and awareness of most points</li> </ul>	<ul style="list-style-type: none"> <li>• Shows little or mixed feelings about the topic being presented</li> <li>• Raises audience understanding and knowledge of some points</li> </ul>	<ul style="list-style-type: none"> <li>• Shows no interest in topic presented</li> <li>• Fails to increase audience understanding of knowledge of topic</li> </ul>	<p><b>Enthusiasm / Audience Awareness</b></p>

validity and importance of the subject				
--	--	--	--	--

## Class Work Rubric

5	4	3	2	1	Points
<b>Completion of programming task</b>					
					Demonstrates synthesis of solution and creates alternatives by combining knowledge and information.
					Demonstrates a clear understanding of how various pieces of the problem relate to each other and the whole.
					Uses software tools and computing resources correctly and effectively.
<b>Correctness</b>					
					Program executes without errors.
					Program contains error checking code and handles all special cases.
<b>Use of Comments / Documentation</b>					
					Clear and effective use of comments to indicate why the code was written.
					Answers/Solutions are properly labeled/commented in great details.
					Supports design procedure with more than sufficient documentation and references.
					Thorough and organized testing has been completed and output from test cases is included in documentation.
<b>Presentation and Organization</b>					
					Name, date, assignment specification and description are included.
					Excellent use of white space (indentation, blank lines)
					Creatively organized work

### Score Description

1. Falls short of expectations
2. Almost meeting expectation
3. Meets expectations
4. More than meets expectations

## 5. Exceeds expectations

# Attendance Policy and Make-Up Work

Students are expected to attend all lectures and labs, and to be punctual in attending classes. Instructors will maintain a positive record of attendance for the online evening class as well as the onsite evening classes. Attendance will be taken 30 minutes after class has begun.

**Onsite Class:** A tardy is defined as arriving in the classroom after the designated time for the beginning of the class or for the continuation of class after breaks. Five tardies to class will be counted as one absence. All tests missed due to the absence of a student must be taken on the first day of attendance after the student's absence.

**Online Class:** A tardy is defined as signing into the video conference platform after the designated time for the beginning of the class or for the continuation of class after breaks. Five tardies to class will be counted as one absence. All tests missed due to the absence of a student must be taken on the first day of attendance after the student's absence.

## MAKE-UP WORK

No more than 5% of the total course time hours for a program may be made up. Makeup work shall:

- (1) be supervised by an instructor approved for the class being made up;
- (2) require the student to demonstrate substantially the same level of knowledge or competence expected of a student who attended the scheduled class session;
- (3) be completed within two weeks of the end of the grading period during which the absence occurred;
- (4) be documented by the school as being completed, recording the date, time, duration of the make-up session, and the name of the supervising instructor; and
- (5) be signed and dated by the student to acknowledge the make-up session.

**Note:** Make-up of missed hours of the externship must be arranged with the externship site supervisor. Make-up time for absences during the externship is limited to 10 hours. This time may be further limited due to supervised make-up work earlier in the term.

A student who misses more than 10% of class hours will be placed on attendance probation with conditions stipulated by the school director. Enrollment in the program will be \*terminated for a student who is absent for more than 20% (41 hours) of the class hours of the program. A student whose enrollment was terminated due to unsatisfactory attendance may re-enroll on attendance probation in a subsequent term in the next 12 calendar months. Such reenrollment does not circumvent the approved refund policy. If the re-enrolled student on attendance probation is absent for more than 20% of the

class hours, the student's enrollment in the program will be \*terminated, and the student will not be readmitted to the program.

## Student Code of Conduct

Students on the RCT campus are expected to behave in a manner that will create a safe and orderly academic environment for themselves and others. Students found in violation of these conduct expectations will be subject to disciplinary action which may include written warning, suspension, dismissal, and/or referral to law enforcement officials. Below is a partial list of inappropriate behaviors that will be subject to disciplinary action. This list is not all-inclusive.

1. Academic dishonesty, including any form of plagiarism, cheating, falsification of records, or collaboration with others to defraud
2. Actions that disrupt teaching, learning, administration, or interfere with the rights of others
3. Non-compliance with the directives of school faculty and staff
4. Violation of written policies, rules, or procedures
5. Theft of any kind, and related behaviors such as possessing stolen property or using the property of others without their permission
6. Damage to property or destruction of property
7. Creation of unsafe conditions
8. Carrying out a false alarm or creating an emergency situation such as a fire or a bomb threat
9. Hurting others, threatening others, or engaging in behavior that may result in harm to others
10. Selling, consuming, and/or possessing alcoholic beverages
11. Possessing or using drugs not prescribed for the student by a physician; selling any drugs; possessing or using illegal drugs or narcotics
12. Possessing a firearm or other deadly or dangerous weapons such as knives, knuckles, clubs, baseball bats, and hammers while on the property of the school or in any part of the school building
13. Sexual harassment in any form by students or any member of the administration, faculty, or staff is prohibited. The school is committed to creating and maintaining an environment for all school personnel

and students that is free of harassment, forced sexual activity, or any other sexual communication or conduct that interferes with performance in the classroom or the workplace

### **Sexual Harassment Defined**

Unwelcome sexual advances, requests for sexual favors, and other verbal or physical conduct of a sexual nature constitute sexual harassment when

- a. submission to such conduct is made either explicitly or implicitly a term or condition of an individual's employment,
- b. submission to or rejection of such conduct by an individual is used as the basis for employment decisions affecting such individuals, or
- c. such conduct has the purpose or effect of unreasonably interfering with an individual's work performance or creating an intimidating, hostile, or offensive working environment.

### **Re-enrollment after Dismissal for Violation of Student Conduct Expectations**

After one six-week term has elapsed, a student whose enrollment was terminated for violation of student conduct expectations that did not result in the involvement of law enforcement officials will have one opportunity for re-enrollment in a subsequent term in the next 12 calendar months.

## Academic Probation

A student who is not making satisfactory progress as defined above at the completion of a subject class will be placed on academic probation for the next subject class. If a student on academic probation achieves satisfactory progress for the subsequent subject class, but does not achieve the required grades to meet overall satisfactory progress for the course, the student may be continued on academic probation for one more subject class. If a student on academic probation fails to achieve satisfactory progress for the first probationary subject class, the student's enrollment will be terminated. The enrollment of a student who fails to achieve overall satisfactory progress for the program at the end of two successive subject classes will be terminated.

\*Incomplete: An "I" for Incomplete is assigned when all the work of a subject class cannot be completed due to circumstances beyond the control of the student. The student may complete the work by the end of the term, or the student can notify the school registrar for readmission for one opportunity to complete the work in a subsequent term beginning no later than 12 calendar months after the end of the term in which the student was assigned the "I". There will be no additional administrative or tuition fees charged for students who exercise this option.

## Termination

Failure to meet the usual examination and grade requirements entitles the academy to place the student on probation for a duration of time. Any student who is deemed to be making insufficient academic progress or whose work is not of adequate quality as determined by the academy may be dismissed. The director or academic program chair must inform the Academy as soon as the student is notified and the probationary

period begins. Probation and dismissal may occur at any time during a student's work toward a completion of the course..

When a department/program determines that a student is not making satisfactory progress, the director in the program and/or instructor will recommend a face-to-face meeting with the student. If, after this meeting, the director/instructor and the student can agree on a plan to remedy the situation, the director and the student will jointly sign a document enumerating steps to take. If, on the other hand, the department/program and the student disagree on issues of progress, the DGS or chair may send the student a letter placing the student on probation.

The letter placing a student on probation must include an explicit statement of what must be accomplished and by what date in order for the student to be removed from probation and returned to good standing in the program. If the student does not comply with the conditions of probation, a letter will be sent to the student with notification of dismissal from the program. Termination letters must inform the student of the right to appeal, first, to the program, and second, to the Director of the Academy.

Students have the right to appeal dismissal from their program. As long as a student is in an appeal process, the student should maintain enrollment and continue working on program completion requirements. A student's first appeal of dismissal must be made to the program. If the student does not appeal, the Academy will send the student an official notice of dismissal from the program.

## Readmission Policy

Students who leave the Academy in good standing may resume their studies by sending an email to [hello@inventive.io](mailto:hello@inventive.io) requesting the readmission form. A readmission fee of \$50 will be added to the student's bill upon re-enrollment in class. The student will be notified in writing of your readmission status via email.

Students who have been dismissed from the Academy and wish to resume their education should send an email of appeal to [hello@inventive.io](mailto:hello@inventive.io) to request an appointment for an interview. Subject line should addressed to the Academy Director.

## Student Complaint Policy

**This school has a Certificate of Approval from the Texas Workforce Commission (TWC)  
The TWC-assigned school number is: S5326**

Complaints are defined as any student concern regarding the school programs, services, or staff. A student who has a concern about a school-related issue is encouraged to schedule a conference with the school director to find resolution. If an issue is not resolved to a student's satisfaction through the conference, the

student can file a formal complaint in writing with the school director who will formally investigate the complaint, take appropriate action, and provide a written response to the student by the 10th business day after the day the formal written complaint is received by member of the school faculty or staff. Note: a conference with the director is not required before a student files a formal written complaint.

### **Notification of Complaint to the Texas Workforce Commission**

A student who is dissatisfied with the school director's response can file a complaint with the Texas Workforce Commission:

Texas Workforce Commission  
Career Schools and Colleges, Room 226T  
101 East 15th Street Austin, Texas 78778-0001  
Phone: (512) 936-3100

Information on filing a complaint with TWC can be found on TWC's Career Schools and Colleges Website at [www.texasworkforce.org/careerschools](http://www.texasworkforce.org/careerschools).

## True and Correct Statement

I hereby certify that the statements and information in this catalog are true and correct to the best of my knowledge and belief.

Andrew Siemer, CEO



---