

ID	Name	Requirements	Priority (1 - 3)	Rationale
Upon Account Creation				
1.1	Creating Account	The user shall be able to create an account on the system		1 This will allow the user to access their account and save their progress on any device.
1.2	Account Information	The system shall obtain the following information when an account is created:		1 This will allow the application to store necessary information for future email messages or certificate creation.
1.2.1	Full Name	- The user's full name		1
1.2.2	Username	- A unique username		1 No two users should have the same username because users should be uniquely identifiable.
1.2.3	Email	- The user's email		2
1.3	Password	The user shall create a password to protect their account		1 User information should be kept secure.
1.4	Password Requirements	The password shall have the following safety requirements:		2 This will allow the password to be secure and protect the account.
1.4.1	Password Length	- Must be at least 8 characters long		2
1.4.2	Special Character	- Must include at least one special character		2
1.4.3	Numeric Character	- Must include at least one numeric character		2
1.5	Optional Knowledge Assessment	The system shall offer an optional knowledge assessment when the user initially creates an account which will provide course suggestions		3 New users will be able to join the proper courses and learn where their strengths and weaknesses lie.
1.6	Intro Tutorial	The system shall offer a tutorial to navigate through the site when the user first creates their account		2 It's important for the user to know how to navigate the site so they can begin learning with ease.
Course Registration				
2.1	Course Registration	The user shall register for a course to begin taking it		1 The user should join a course before taking it so that the system can keep track of the user's enrolled courses
2.2	Multiple Course Enrollment	The user shall have the ability to be enrolled in multiple courses at once		1 The user will be able to access any course they want
2.3	Course List	The system shall maintain a list of courses the user is enrolled in		1 This will allow the user to access the courses he or she is currently taking.
2.4	Unregistering	The user shall be able to unregister from a course		1 Users should be able to unregister from courses that they want to stop taking.
2.5	Remove Unregistered Courses	The system shall remove the course from the user's list of enrolled courses once they unregister from it		1 This will prevent the list of registered courses from becoming too crowded.
Course Content				
3.1	Course Lessons	Courses shall contain written lessons of computer science concepts		1 These lessons describe the content the course creators want to teach like loops, functions, classes, etc.
3.2	Course Examples	Courses shall provide examples of how computer science concepts are implemented in the relevant programming language		1 The courses should contain implementation examples so the user understands how the programming language works.
3.3	Section Assessments	The courses shall assess the user's knowledge with an assessment after each course section		2 It is important to assess the user's knowledge after each section of the course to keep them on track.
3.4	Final Assessments	The system shall assess the user's knowledge after each course with a final assessment		1 The final assessment is the main way the user can tell if they have a good grasp on the course as a whole.
3.5	All Multiple Choice	Assessments shall be comprised of only multiple choice questions		1 Multiple choice questions allow for automatic grading.
3.6	Question Limit	Section assessments shall be no more than 10 questions		1 There is no need for section assessments to be more than 10 questions since they only evaluate a small portion of the course content.
3.7	Final Question Limit	Final assessments for courses shall be no more than 20 questions		1 The final shall be longer than section assessments since it covers the entire course.

3.8	Automatic Grading	The system shall automatically grade assessments	1	This will allow automatic feedback for the user to see what they need to improve on.
3.9	Content Access	Users shall have access to all of a course's content once they are registered in the course	1	All of the course content will be available to the user at all times so courses can be completely self paced.
3.10	Video Graphic	The system shall have a video graphic of a rocket ship taking off when the user finishes each course section	3	This "gamification" will hopefully motivate the user to continue their courses.
3.11	Retaking Assessments	Users shall be able to retake assessments until they are satisfied with the results	2	Since lessons are the main priority of the system, not assessments, users will be able to retake all assessments
Upon Course Completion				
4.1	Removing Completed Courses	Completed courses shall be removed from the user's list of registered courses	1	Removing completed courses will prevent the registered courses list from becoming crowded.
4.2	Completed Courses List	The user shall be able to access previous courses from a list of their completed courses	2	Users should be able to revisit and view their completed courses in order to refresh on learned concepts.
4.3	Star Rating	The user shall be able to rate courses from 1 to 5 stars after completion	1	A rating system will allow users to voice their opinions of the course.
4.4	Course Certificate	The user shall receive a certificate after they complete a course with the following information:	1	Certificates will provide proof that the user has completed the course.
4.4.1	Certificate Name	- Their first and last name separated by a space	1	
4.4.2	Certificate Course	- The course title	1	
Course Listings				
5.1	Offered Languages	The system shall offer the following courses: Python, JavaScript, Java, C#, C, C++, Go, R, Swift, PHP	2	While it is important to offer these courses, we will just design the ability to build courses. We will not offer a course for each language.
5.2	List Languages by Popularity	The system shall list the courses it offers by popularity. From most popular to least, this order is Python, JavaScript, Java, C#, C, C++, Go, R, Swift, PHP	1	Showing the users which programming languages are the most popular will make it easier for them to decide where they should start.
5.3	Course Difficulty	The system shall offer courses at three levels of difficulty: easy, medium, or hard	1	This will allow the user to know which courses are best for starting out so they do not become overwhelmed.
5.4	Course Difficulty Indication	Course difficulty shall be indicated next to the name of the courses	1	The course difficulty should be clearly displayed so that the user instantly knows if the course will be appropriate for their proficiency.
5.5	Course Descriptions	The system shall display course descriptions which provide users with a basic summary of the course	1	Having a description for each course will help the user know what the course offers.
5.6	Course Ratings	Course descriptions shall indicate the rating of the course with 1 to 5 stars	1	A course rating will help users distinguish between well made and poorly made courses.
5.7	Language Uses in Description	Course descriptions shall list common applications of the programming language that the course teaches	2	Showing the common applications of the programming language will help the user decide if they want to learn that language or not.
5.8	Course Description Length	Course descriptions shall be shorter or equal to 300 characters in length	3	Users can gain a basic understanding about the courses through a clear and concise description
5.9	Sort by Rating	The system shall be able to order courses by star rating	2	Sorting by rating will allow users to find the best courses.
5.10	Key Word Search	The user shall be able to search for courses by key words in course titles	1	A search function will be helpful when users are looking for courses on a specific language or concept.
Course Creation				
6.1	Course Creation	The user shall be able to create public courses	1	The user will have the capability of creating courses for other users.
6.2	Course Creation Progress	The system shall save the user's progress when making a course	2	This will prevent the user from having to start over on their course creation progress if the system shuts down or if the user decides to take a break.

6.3	Resuming Course Creation	The system shall allow the user to resume their course-making progress after taking a break	2	This will allow the user to take breaks instead of making the entire course in one sitting.
6.4	Submitting Created Course	The user shall be able to submit their course for review once they have finished creating it	2	The user needs to send their created course to the system admin for review.
6.5	Course Approval	User created courses shall be approved by the system administrator before being published	2	All courses will have to be approved to maintain high quality content.
6.6	Course Template	The user shall follow a template provided by the system when creating a course	1	A required template will streamline course creation and make all courses similar.
6.7	Course Edits	The user shall be able to make edits to the courses that they created	3	The user should be able to update their own courses in the case where a mistake slipped through or if they want to update information.
6.8	Submitting Course Edits	The user shall submit their course edits for review	3	Any edits need to be submitted so that the system admin has the opportunity to review them.
6.9	Approving Course Edits	Course edits shall be approved by the system administrator before updated	3	If the edits are deemed good, the system admin will update the course with the provided edits.

Questions and Answers

7.1	Public Comment Forum	Courses shall have public forums for users to converse	2	Users can chat with other users about the courses they are enrolled in to offer help and ask for advice
7.2	Hiding Public Forums	The system shall hide public forums from users under 13	2	Young users shall not be able to see forums in case they are exposed to inappropriate content.
7.3	FAQ Page	The system shall have an FAQ page with the following:	3	An FAQ page will help users quickly find solutions to problems they have with the system.
7.3.1	Questions	- The FAQ page will have commonly asked questions about using the system	3	
7.3.2	Answers	- The FAQ page will have answers to these questions provided by the system admin	3	

System Capabilities/Miscellaneous

8.1	Login Information	The user shall login to their account with their username and password	1	This will allow the system to have unique login information for all users.
8.2	Save Progress	The system shall save the user's current progress in all enrolled courses	1	Users should be able to exit the application and resume their work.
8.3	Weekly Time Goal	The user shall be able to set a weekly time goal which is completed when they spend the chosen amount of time in course lessons	3	This function will assist in time management and help the user keep track of course work.
8.4	Completion of Time Goal	The user shall be notified with a congratulations on the completion of a weekly time goal	3	Congratulating the user will help them build confidence in their education and continue using our site for learning purposes.
8.5	Completion Percentage	The system shall display a completion percentage for each course that the user is taking	1	This will allow users to know how close they are to finishing the course.

"Classroom" Functionality

9.1	Creating Classrooms	The user shall be able to create a "classroom" to monitor other users	3	This will help businesses with training or schools with online classes
9.2	Classroom Security	Classrooms shall be protected by a randomly generated key	3	Classrooms should be protected to prevent random users from joining
9.3	Classroom Keys	Classroom keys shall be 8 digits in length	3	To properly protect classrooms, keys should be a sufficient length
9.4	Classroom Invitations	Classroom creators shall be able to invite other users into their classroom	3	Classroom creators should be able to invite relevant users to their classroom
9.5	Invitation Email	Classroom invitations shall be sent to the user's email		This will provide an easy way for users to access an invitation
9.6	Invitation Acceptance	Users shall be able to accept classroom invitations	3	Users can accept invitations to automatically join classrooms
9.7	Invitation Declining	Users shall be able to decline classroom invitations	3	Users can decline invitations if they do not want to join the classroom

9.8	Joining Classrooms	Users shall be able to join classrooms with the key	3	Users with the classroom key should have the ability to join classrooms
9.9	Viewing Progress	Classroom creators shall be able to view the course progress of other users in their classroom	3	This will allow classroom creators to monitor the users and know which users are working hard
9.10	Course Assignment	Classroom creators shall be able to assign courses to the users in their classroom	3	Classroom creators will assign required courses for training or for online classes to the users in their classroom
9.11	Course Deadlines	Classroom creators shall be able to set a deadline for courses assigned to the users in their classroom	3	Deadlines will allow classroom creators to pace their lessons and hold the classroom members accountable