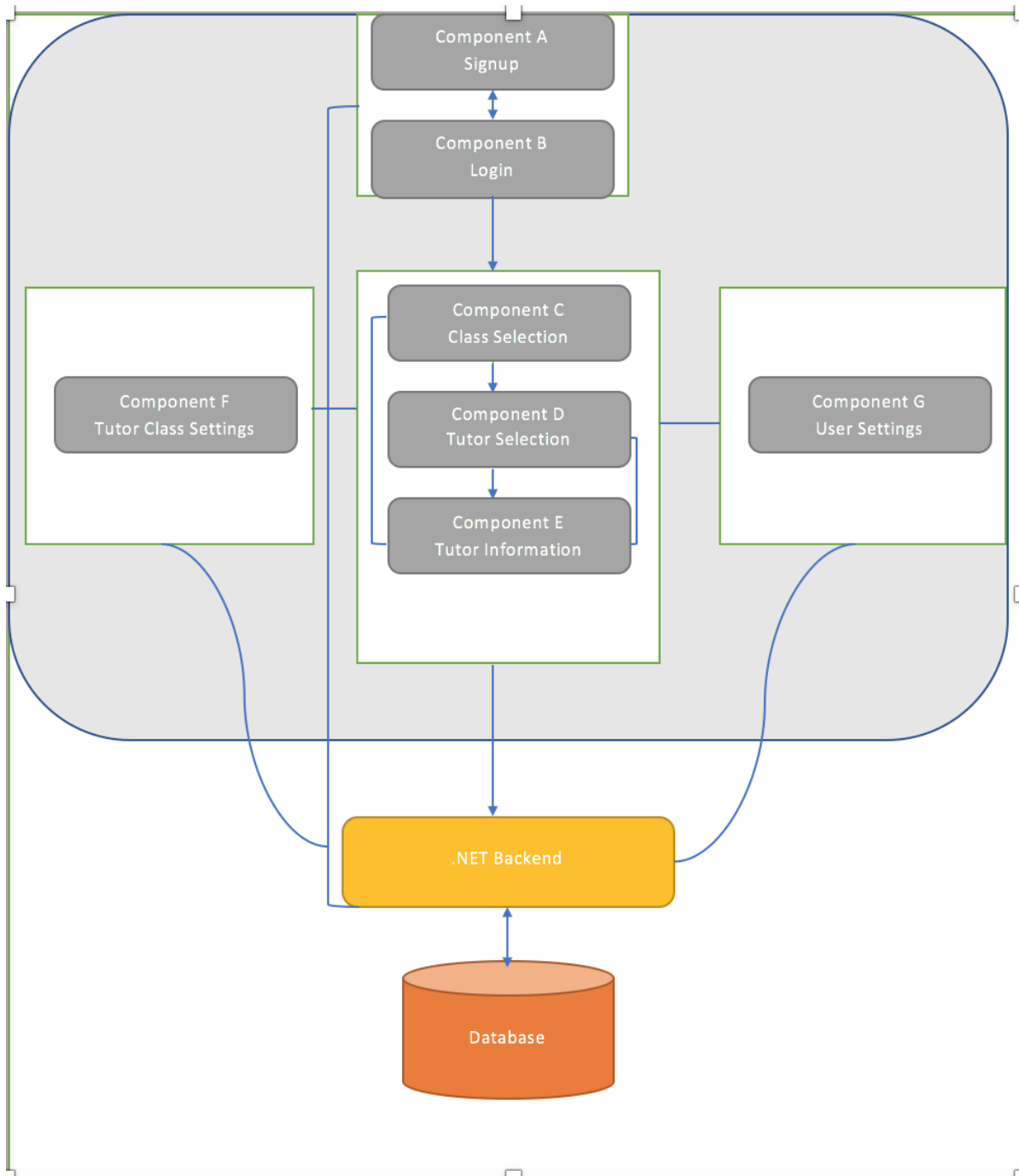


CS408: Incremental Testing and Regression Testing

Sprint 2

Component Diagram

The component diagram below shows the different components of the “Find my tutor”.



The frontend container encapsulates all components which are dependent on the .NET Backend and which has been connected with the database.

Overview of Components

- Component A: Signup Page
 - ❑ Inputs - First Name, Last Name, Password, Confirm Password, E-Mail
 - ❑ Outputs - Redirect to TutorSearch page or Error Message
 - ❑ Input - Click Login button
 - ❑ Output - Redirect to Login page
 - ❑ Dependent Components: login, tutorSearch page
- Component B: Login Page
 - Inputs - UserName, UserPass.
 - Outputs - Redirect to TutorSearch page or Failure Message.
 - Input - Click Signup button
 - Output - Redirect to Signup Page
 - Dependent Components: tutorSearch page
- TutorSearch Page
 - ❖ Component C: Class Selection
 - Input - Class
 - Output - Tutors for class
 - Dependent Components: page navigation, tutor selection, tutor profile, user profile
 - ❖ Component D: Tutor Selection
 - Input - Tutor
 - Output - Tutor name, email, rating, available times, biography
 - Dependent Components: tutor information, page navigation, user profile, tutor profile
 - ❖ Component E: Tutor Information
 - Input - Tutor rating
 - Output - Changed tutor rating
 - Dependent Components: page navigation, tutor selection, class selection, user profile, tutor profile
- Tutor Profile Page
 - ❖ Component F: Tutor Class Settings
 - Inputs - Course Name, Course Number, Price per hour
 - Outputs - Success Message (Listing Updated Successfully)
 - Dependent Components: Add Tutor Class, tutor search page, user profile

- UserProfile Page
 - ❖ Component G: User Settings
 - Inputs - First Name, Last Name, Email, Password
 - Outputs - Success Message
 - Dependent Components: user settings, tutor search page, tutor profile

- .NET Backend
 - ❖ Input - Contains all functions which are necessary for creating accounts, updating and managing accounts, authenticating login, adding a tutor, class search, tutor search, tutor information and user settings. For most of the functions as an input userID initiating the app along with login information, or class added or tutor added or time added passed.
 - ❖ Output - Contains functions returned to the frontend that contains the required information from the backend that fills up the data required to be filled like classes, tutor names, tutor information etc.
 - ❖ Dependent Components - database, Components: A, B, C, D, E, F, G.

- Database
 - ❖ MySQL database that stores all the information for every user, including all user account information, tutor class information, class selection, tutor selection, tutor information and user setting information.
 - ❖ Input - data passed from .NET Backend by way of SQL query
 - ❖ Output - resulting success relation for selection queries and error messages on query relations.
 - ❖ Dependent Components - .NET Backend

Incremental Testing Method

For incremental testing, we decided to go with bottom-up approach testing. When coding our application, we began writing the subcomponents first and then working our way to the main component, which was essentially the container for most of our sub modules (Tutor Search page). Utilizing bottom up approach enabled us to do more effortlessly characterize our test conditions and see the quick outcomes, which we observed to be more efficient as we had an expansive number of sub-segments whose usefulness was endless supply of numerous other sub-segments. We observed this to be a significantly more powerful method for testing to ensure that the various collaborations between the segments all worked rationally together by our normal prerequisites.

For eg. In order to check whether Tutor search page works or need, we needed to implement tutor profile page.

INCREMENTAL AND REGRESSION TESTING

Module	Component A: Sign up
---------------	----------------------

Incremental Testing

Defect No.	Description	Severity	How to Correct
1.	The application should not crash when creating new user accounts.	1	Modify the signup controller to validate input across all of the different input boxes before attempting to update the database.
2.	Users with invalid emails are successfully registered	2	Within the signup controller, add a check to ensure that the email is of the correct form.
3.	Multiple registration of same account.	1	Check that the email address is unique before adding to the database.
4.	Allows registration even if the password boxes do not match.	3	Modify the signup controller, add view to display an alert to the user and highlight the field.

Regression Testing

Defect No.	Description	Severity	How to Correct
1.	Fixing the controller to generate valid tokens results in the crash of server, if there are way too many connections.	1	Have an expiry for each token.
2.	Fixing the account duplication of the same user did not resolve.	2	Manually delete the accounts to avoid account duplication.

Module	Component B: Login
---------------	--------------------

Incremental Testing

Defect No.	Description	Severity	How to Correct
1.	After a user logs in, the token is stored on the frontend. Login is no longer required to access TutorSearch unless the user hits 'Log Out'.	2	Leaving the main TutorSearch page should reset the token.

Regression Testing

Defect No.	Description	Severity	How o Correct
1.	Fixing the error in Component B caused no noticeable errors in the other components.	3	No need for correction.

Module	Component C: Class Selection
---------------	------------------------------

Incremental Testing

Defect No.	Description	Severity	How to Correct
	Transition to new design format broke class tab design creation	1	Changed JS to dynamically create tabs with <u> elements, rather than the previous <u> format
3	Transition to new design format broke ability to close tutor info	2	Redo 'close' button on tutor info to account for new layout
	Loading classes gave an overflow error, as there was no limit checking in javascript	1	Make first item in array the length of the array
	Hovering over a class name would immediately load the tutors, slowing performance	3	Changed hover function to onclick

Regression Testing

Defect No.	Description	Severity	How to Correct
	Adding tutor ID to each tutor tab interfered with CSS styling	3	Changed tutor string for loading to JSON object with ID and name as properties
	Changing from hover to onclick forced tutor lists to no longer hide when selecting another class' tutors	2	In onclick function, forced added 'close' to all other tutor lists

Module	Component D: Tutor Selection
---------------	------------------------------

Incremental Testing

Defect No.	Description	Severity	How To Correct
2	No tutors available for selected class	1	Add warning message for lack of tutors
	Loading class tutors requires the class name to be passed, but name wasn't stored correctly, so it was loading all tutors, no matter the class	1	Class name added to onclick function for each class

Regression Testing

Defect No.	Description	Severity	How To Correct
3	After adding 'close' button, tutor list also closes	3	Prevent tutor list from retracting upon info close
	Using JSON broke JS parser	1	Changed parser to a for loop for each subject in the JSON object

Module	Component E: Tutor Information
---------------	--------------------------------

Incremental Testing

Defect No.	Description	Severity	How To Correct
	Calendar allows user to modify a tutor's schedule	1	Disable different handlers in Calendar script.
	Calendar displays out of positions in the html container	3	Force padding-left to be 0
	Calendar displays wrong dates, which are not current	2	Forces calendar to display only days of week (Ex. Monday, Tuesday, etc.) instead of date.

Regression Testing

Defect No.	Description	Severity	How To Correct
	Loading the tutor info for each tutor wasn't loading id-specific tutors, no tutors with same name weren't showing correctly	2	Set ID of each tutor property to the tutor's ID

Module	Component F: Tutor Class Settings
---------------	-----------------------------------

Incremental Testing

Defect No.	Description	Severity	How to Correct
2	Choosing 'add class' loads 'class detail' module instead of the 'add class' module	1	Change method call in JavaScript file

Regression Testing

Defect No.	Description	Severity	How to Correct
1	Adding non-breaking space between first and last name pushed tutor description box to the right	3	Fix positioning of description box
3	Making class list vertically scrollable makes "class list" header also scroll	3	Create separate 'navbar' specifically for header

Module	Component G: User Settings
---------------	----------------------------

Incremental Testing

Defect No.	Description	Severity	How To Correct
1	Email address of wrong format gets accepted when Submit button is pressed	2	Use Regex to make sure email is of correct format
2	Empty Current Password field shows an error for password being empty	2	Empty Current Password field means user does not want to change password. Check and make sure New and Repeat Password field are empty too. If yes, do not implement changing password
3	New Password field was accepted when empty	2	Do not let the user proceed with empty New Password. Give an error.
4	Repeat New Password field was accepted when empty	2	Do not let the user proceed with empty Repeat New Password. Give an error.

Regression Testing

Defect No.	Description	Severity	How To Correct
1	After fixing Defect 1, blank email address was not being accepted by Regex	2	Make sure if email address is blank, don't check Regex. This means that user does not wish to change the email address
2	After fixing Defect 2, empty New Password field was still being tagged as error when Current Password was empty	2	If current password was empty, do not tag empty New Password as error. Infact tag the inverse.
3	User was allowed to type in the New Password and Repeat New Password text box, with Current Password field empty	2	New Password and Repeat New Password fields should be disabled, unless the user types the Current Password.

Updated Backlog

ID	Functional Requirement	Hours	Status
1	As a user, I would like a create an account.	6	Completed in: Sprint 1
2	As a student, I would like to see available tutors, for selected courses	4	Completed in: Sprint 2
3	As a student, I would like to view available times for the tutor I've selected	4	Completed in: Sprint 2
4	As a student, I would like to be able to rate tutors	3	Completed in: Sprint 2
5	As a student, I would like to be able to see ratings of a tutor	3	Completed in: Sprint 2
6	As a student, I would like to view tutor's contact information	8	Completed in: Sprint 2
7	As a student, I would like to search for a specific course	6	Completed in: Sprint 2
8	As a student, I would like to search for a specific tutor	6	Completed in: Sprint 2
9	As a tutor, I would like to edit a tutor profile	8	Completed in: Sprint 2
10	As a student, I would like to edit a student profile	8	Completed in: Sprint 2
11	As a tutor, I would like to input the classes I can tutor.	6	Completed in: Sprint 2
12	As a tutor, I would like to input the days I can tutor.	4	Completed in: Sprint 2
13	As a tutor, I would like to input the times I can tutor.	4	Completed in: Sprint 2
14	As a tutor, I would like to input the price I tutor for.	4	Completed in: Sprint 2
15	As a tutor, I would like to be able to create a bio.	4	Completed in: Sprint 2
16	As a tutor, I would like to be able to see my rating.	2	Completed in: Sprint 2
17	As a tutor, I would like to remove my tutoring availability.	4	Completed in: Sprint 2

18	As a tutor, I would like to modify my tutoring availability.	4	Sprint 2
19	As a student, I would like to input the classes I am taking	6	Completed in: Sprint 2
20	As a student, I would like to input the time I am available weekly.	4	Completed in: Sprint 2
21	As a tutor, I would like to decide if I am willing to tutor group sessions or not.	5	Sprint 2
22	As a student, when I click on a class, I want to see a master calendar displayed showing the timings of all the tutors.	7	Sprint 2
23	As a student, I would like to send a reservation notice to the tutor.	6	Sprint 2
24	As a tutor, approve or deny a student's reservation.	10	Sprint 2
25	As a tutor, I can check the maximum and minimum prices.	4	Sprint 2
26	As a student, I would like to see which time slots are for group/individual	5	Completed in: Sprint 2
27	As a student, I would like to see the time of tutors that matches my free time.	6	Completed in: Sprint 2
28	As a student, I would like to be notified when a new tutor signs for a class that I am taking.	6	Sprint 2
29	As a user, I would like to login back to my account created.	2	Completed in: Sprint 1
30	As a user, I would like to logout of my account.	2	Completed in: Sprint 1
31	As a student, I would like to view the tutor's profile {rating, price, bio etc.}.	1	Completed in: Sprint 2
	Total:	152	