[CS361] HW 5 Requirements Assignment

Group 17 Members
Arthur Liou
Eric Edwards
Jason Adams
Gurbir Behniwal

8/3/18

Outline

- Describe your 10-20 user stories (approx 2 pages) - **All**
- For each user story, indicate when the story is due (if at all) and list the corresponding tasks. For each task, indicate if you think that it logically should be completed before or after another task. In addition, how long you think that it would take a pair of programmers, working together, to complete the task. (approx 2 pages) - **All**
- For each user story due next week, either give a UML sequence diagram showing what your implementation will look like, or describe a spike that you did in order to learn about how to implement the user story (approx 3 pages)
- Outline your plan for implementing the stories due next week -- who is doing what, when? (approx 1 page)
- In one sentence, briefly summarize whether your customer was willing and able to meet with you. Also, indicate if you think that the customer was reasonable about what will be due on Tuesday. If your customer cannot meet with you on those days, then assign one of your teammates to play the role of the customer.
- Briefly summarize the contribution of each of your team members.

User Stories

Template: As a [xyz user], I want to do [action].

Goal: 12; 3-4 each

Use Case 1:

Story: As a potential tutor, I want to register as a tutor on the website.

- Enters information into HTML form on website
- Website will receive the information and send it to the database
- The database will save the user's information if it is not already present.
- Once the database confirms the user is successfully entered into the system, the database will let the server know the user was successfully added.
- The server will send a response to the front-end informing the potential tutor that their application and registration form went through successfully & that their application is pending, waiting on a background check.
- The server will inform the respective authority (or call a third-party API to initiate a background agency like Checkr) to inform them they need to order a background check for the potential tutor.

Due Date: August 12

Estimated Time: 10 hours

Corresponding Tasks:

- Create search page and form in HTML
- Create route in our server's js for search form submission
- Create MySQL queries for the database lookup for matching tutors
- Create express handlebars view for displaying search results

Story: As a web administrator or tutor manager, I want to log whether a tutor successfully passed a background check.

- After tutor completes their registration, a background check will be initiated to a background check agency.
- Agency will report back whether the tutor applicant passed or not.
- Add those results to the database
- Add passed tutors to a list of active and available tutors
- Notify tutors they have passed the background check and are ready to tutor students!

Date Due: We will not get to this user story in the final two weeks

Estimated Time: 20 hours

Corresponding Task:

- Create “passed background check” form in HTML
- Create route in our server’s js for receiving a completed tutor application
- Create MySQL queries for the database edit for whether tutor passed or failed background checks
- Create express handlebars view for displaying search results

Story: As a tutor, I want to be able to select one or multiple subjects to tutor.

- Logs on with username and password into HTML form on website
- The JS running on our page send the entered credentials to our server using a POST command
- Our server runs a mysql query using the entered username as its search field and checks if the returned user row has a password field that matches what was received from the POST
- If yes the user is authenticated and a new login session is created

Date Due: Second week

Estimated Time: 15 hours

Corresponding Task:

- Set up tutor homepage after login with a section for a tutor to add more subjects.
- Create route in our server’s js for grabbing available subjects & a tutor’s current subjects
- Create MySQL queries for the database to view subjects & edit a tutor’s current subjects
- Create express handlebars view for displaying updated current subjects

<u>Use Case 2: Register Student for tutoring program</u>

User Story

As a teacher, I want to sign into my account on the website

- Enters username and password into HTML form on website
- The JS running on our page send the entered credentials to our server using a POST command
- Server runs a query with the username and password and checks the database if the username exists and if it does it checks whether the password given in the POST matches the password field
- If yes then the user is authenticated and a new login session is created
- If no then an error message will occur asking the user to try again

Date Due: We will not get to this user story in the final two weeks

Corresponding Task:

- Set up server to communicate both with directing the user to the right website and with sending/receiving information from the database. This task will be done using node.js to host the website. Will need to be familiar with express and handlebars as well as node.js. MySQL will be used to communicate with the database
- Will need to research how to encrypt information and have the time to implement this. A signin and password is not something that we will want to send without an encryption.
- The SQL database must be set up with the correct tables and relationships for this user story.
- HTML/CSS will be used to create the login webpage
- Logically, we would want this done before the teacher is able to sign up a student but due to the time constraints this will not be possible for us

Estimated Time: 3 weeks

User Story

As a teacher, I want to fill out a form to register a new student

- User enters information into a HTML form on website
- Website will receive this information and after the user hits submit will send a POST request to the server
- If required fields are not completed in the form then the website will send an error message to the user to try again.
- The server sends a mysql query that will add the new student info with the information into the appropriate tables in the database.
- If this query comes back successful then the server will create a student account with his/her own username and password

Date Due: August 12, 2018

Corresponding Task:

- Set up server to communicate both with directing the user to the right website and with sending/receiving information from the database. This task will be done using node.js to host the website. Will need to be familiar with express and handlebars as well as node.js. MySQL will be used to communicate with the database
- The SQL database must be set up with the correct tables and relationships for this user story.
- HTML/CSS will be used to create the webpage with the POST form

Estimated Time: 1 week

User Story

As a teacher, I want to be able to change which subjects the student is allowed to sign up for.

- HTML page that displays a list of student account that the user registered
- User is able to click on one of the student accounts to edit and the js redirects to a HTML page that displays the students' current list of subjects
- After it is submitted, the server runs an update mysql query to update the students' account with the new data
- If the query is successful then the teacher is redirected back to his/her main account page.

Date Due: We will not get to this user story in the final two weeks

Corresponding Task:

- Set up server to communicate both with directing the user to the right website and with sending/receiving information from the database. This task will be done using node.js to host the website. Will need to be familiar with express and handlebars as well as node.js. MySQL will be used to communicate with the database
- The SQL database must be set up with the correct tables and relationships for this user story.
- HTML/CSS will be used to create the webpage
- Logically, this will be done after teacher is able to register new students

Estimated Time: 1 week

Use Case 3: Student signs up for a tutoring session

Story: Student signs into website

- Enters username and password into HTML form on website
- The JS running on our page send the entered credentials to our server using a POST command
- Our server runs a mysql query using the entered username as its search field and checks if the returned user row has a password field that matches what was received from the POST
- If yes the user is authenticated and a new login session is created

Date due: We will not get to this within the next two weeks
Estimated Time: 7 hours

Corresponding Tasks:

- Create HTML form for login page
- Create server js file for route handling and write MySQL query for user authentication

Story: Student searches for nearby tutors

- Student must be signed in
- Student navigates to search page in our web application
- Student enters session search criteria such as subject, time, date, etc, into an HTML form
- Our route handler from Node.js submits the form to our server via a POST request
- Our server queries our database's Tutors table to find matches for the subject, available times, available dates, sorting the returned table by distance from the student's location
- The user is taken to a results page where data from our query is displayed utilizing an express view for our HTML

Estimated Time: 6 hours
Date Due: Second week of work

Corresponding Tasks:

- Create search page and form in HTML
- Create route in our server's js for search form submission
- Create MySQL queries for the database lookup for matching tutors

- Create express handlebars view for displaying search results

Story: Student selects and confirms a tutoring session from a search

- A session search must have been performed
- From the search results page, the user checks a box by their desired tutor and clicks submit
- Form is submitted via POST to our server
- Server makes a new row in our Sessions table with the appropriate entries for the date, time, tutor_id, student_id fields
- - Server sends a confirmation email to both the student and tutor involved using the email properties of their entries in the Students and Tutors tables respectively.

Date due: We will not get to this in the next two weeks
Estimated Time: 10 hours

Corresponding tasks:

- Create route handler for session confirmation submission
- Create MySQL query to create new row in our Sessions table with the student_id, and tutor_id of the session participants as well as corresponding date/time/location information
- Create automated confirmation email script

Extra User Stories

Story: Tutor cancels tutoring session

- A session search is performed
- User clicks on session that they want to cancel
- Logic checks that there is more than 48 hours between session start time and current time
    - If less than 48 hours notice, user is notified that they cannot cancel session
- Server updates database and deletes session from database
- Email sent to teacher/student account informing of cancelled session
- Tutors session cancel total is increased by 1
- Logic checks that tutor has not canceled more than 8 tutoring sessions

Due Date: None

Corresponding Tasks:

- Create search page and form in HTML
- Create route in our server's js for search form submission
- Create MySQL queries for the database lookup for matching tutors
- Create express handlebars view for displaying search results
- Create form to delete data using MySQL
- Create path to delete database

Story: School is created

- Administrator creates a school entity
- Adds address, school name

Date due: We will not get to this in the next two weeks
Estimated Time: 15 hours

Corresponding Tasks:

- Create HTML page that allows user to click "add school option"
- Create HTML "add school page"
- Create route in our server's js for search form submission
- Create MySQL queries for the database insertion of form data

Story: Teacher is created

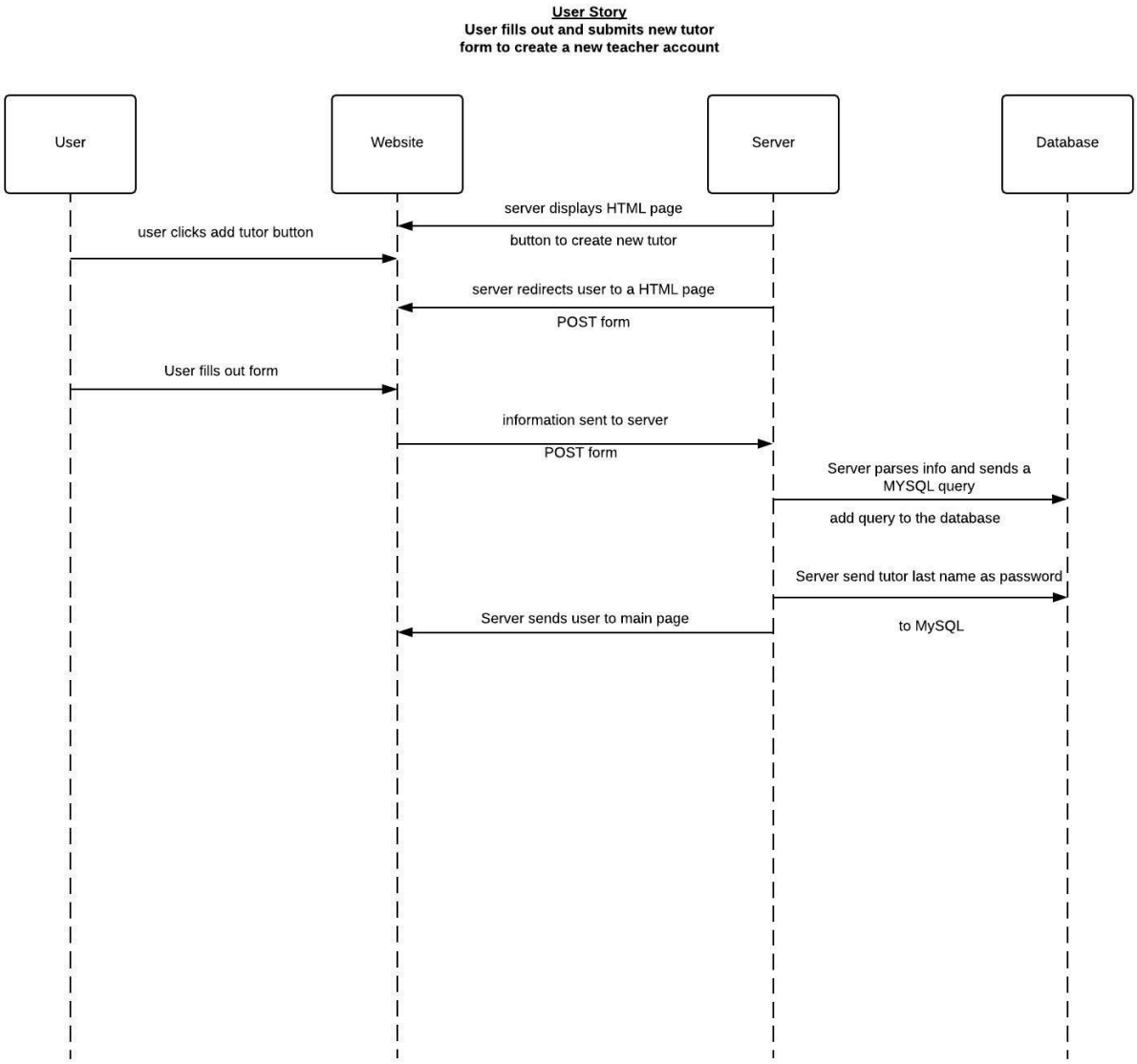
- List of active teachers sent to administrator
- Post form g
- Teacher information sent to server
- Database updated with the following:
    - Teacher first name, last name and id entered
    - Teacher assigned a school
    - Unique teacher code generated
    - Teacher_id set as password

Due Date: August 10th
Estimated Time: 15 hours

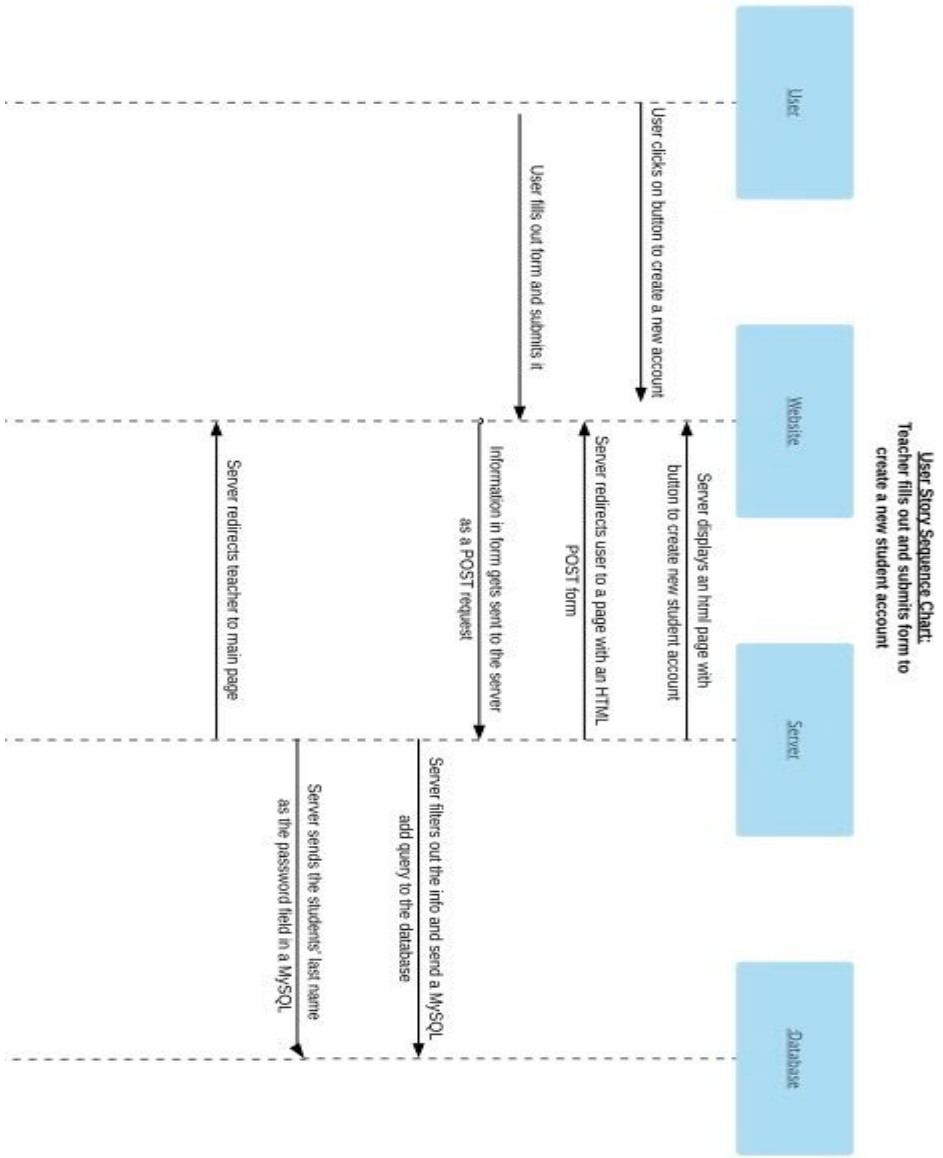
Corresponding Tasks:

- Create HTML "add teacher page"
- Create route in our server's js for search form submission
- Create MySQL queries to insert form data into database

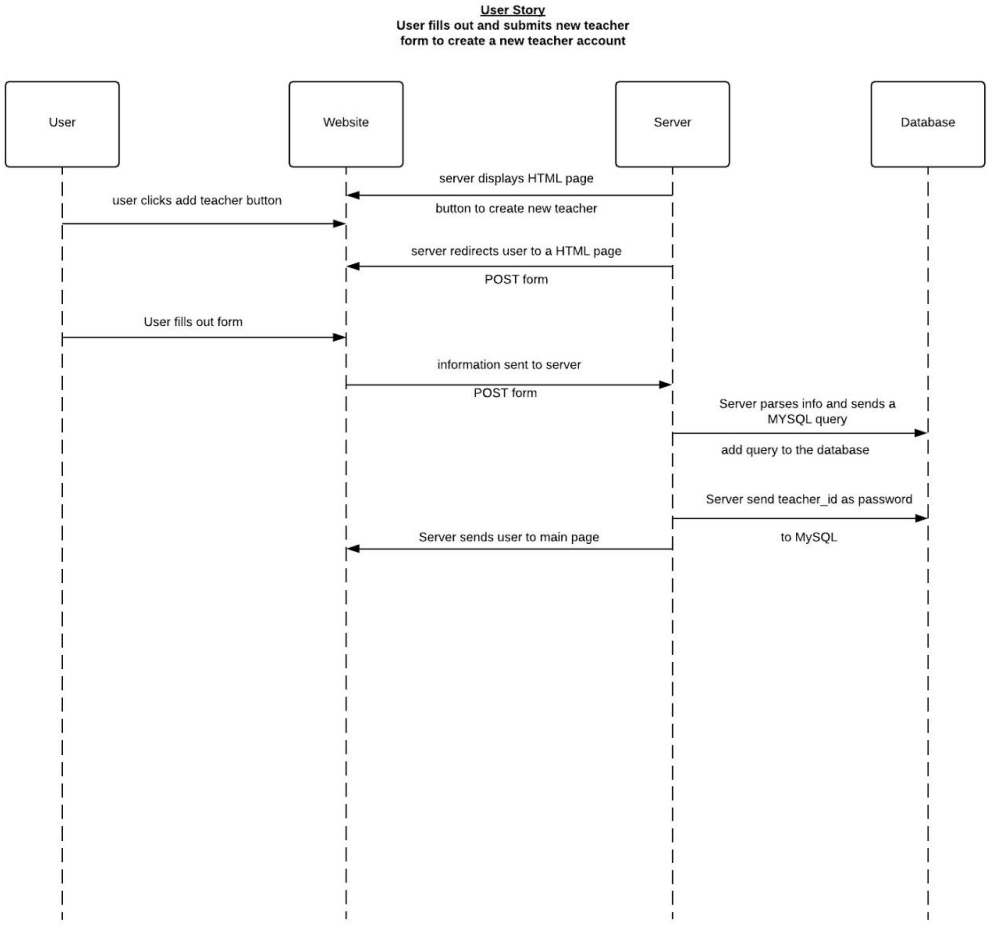
UML Diagram: Use case 1 - Add Tutor



UML Diagram: Use case 2 - Teacher adds student



UML Diagram: Use case 2 - Add Teacher



Outline of implementing next week's stories:

Who: Arthur
What: User Story: As a tutor, I want to fill out a form to register myself in the system.

- Create the HTML page for the tutor registration form
- Create js functions to register the new user's submitted information into the Tutors table in database.
- Create HTML page to confirm to the tutor that registration has completed successfully

Who: Gurbir
What: Database creation

- Will work on creating the database definition queries to set-up the MySQL tables.
- Tables to create this week: Students, Tutors, Teachers
- Ensure proper creation of foreign keys in appropriate tables to ensure we can make use of the relationships we've outlined in our earlier plans
- To be finished ny Wednesday so other group members can start writing their queries.

Who: Jason
What: User Story: As a teacher, I want to fill out a form to register a new student
Tasks to do this week:

- Create a HTML Form through handlebars
- Create another HTML page that will direct the teacher to the the HTML form
- Setup the server to receive this information
- Set up the SQL database to handle this
- Create the MySQL queries to add info the database

Who: Eric
What: User Story: Create teacher users:

- Create a HTML Form through handlebars
- Create another HTML page that will direct the teacher to the the HTML form
- Setup the server to receive this information
- Set up the SQL database to handle this
- Create the MySQL queries to add info the database

Briefly summarize the contribution of each of your team members.
Arthur - Wrote up 3 user stories and their corresponding tasks, time estimation. Contributed to the outline by adding what he plans on implementing this week

Jason- Wrote up 3 user stories and their corresponding tasks, time estimation. Created a UML sequence chart for one of the user stories. Contributed to the outline by adding what he plans on implementing this week

Gurbir - Wrote 3 user stories regarding tutor sessions and their corresponding tasks and time estimates. Created UML sequence chart for tutor search user story. Described his tasks for the upcoming week in the outline section.

Eric- Wrote up 3 user stories and their corresponding tasks, time estimation. Created a UML sequence chart for one of the user stories. Contributed to the outline by adding what the plans on implementing this week

Customer Contribution:
Paige met with the group and helped coming up with the user stories.