Abena Agyemang
Michelle Bagnall
Ashawn Rose

User Stories

User Stories Table

As a user i want to organize my monthly expenses based on category	10 Days	Low Priority
As a user i want to be able to edit my bills monthly	5 Days	Medium Priority
As a user i want to be able to track my bills on the calendar so i don't miss a due date	7 Days	Medium Priority
As a user i want to be able to assign a priority to each category	7 Days	High Priority
As a user I want to be able to register a new account so that my expenses are associated with my account.	10 Days	High Priority
As a user I want to be able to create a password for my account so that only I can access my expense tracking account.	10 Days	High Priority
As a user I want to be able to add my bank loans to my expense tracker account so that I can track how much I have left to pay.	5 Days	Low Priority
As a user I want to be able to add my bank account to my expense tracker account so that I can organize my expenses in the tracker.	5 Days	Medium Priority

As a user I want to be able to add my income amount to the tracker so that I can set spending limits on my account.	7 Days	Medium Priority
As a user I want to be able to create a message to the user to alarm them that they have spent close to the budget.	10 Days	High Priority
As a user I want to login to my expense tracker account so that I can access the expense tracker features.	10 Days	High Priority

Story Tasks

Organizing monthly expenses by category

- Create a tab specifically for monthly expenses
- Design that tab
- Create code that allow user to add and name their monthly expenses
- Allow user to delete their monthly expenses

Alarm for user to know when they are close to their budget

- Create a loop that will update the number every time user spends
- Create a code that checks if the amount is close or not
- Then shows a message if it is

Logging in & verifying

- Create code to check to if user is already logged in
- Create code to redirect to login page
- Design the login page
- Create function that calls all of the above, verifies redirected on not logged in

Adding bank account information to the user’s account

- Create “Add Bank Account Info” button
- Design “Add Bank Account Info” page
- Ask user for their banking institution and allow user input
- Ask user for bank account number and allow user input
- Create “confirm” button that saves information to user’s account
- Display confirmation message

Adding income to expense tracker site

- Create “Add Income” button
- Design “Add Income” page
- Allow user to pick “annual salary” or “monthly income” options
- Ask user for income amount and allow user input
- Create “confirm” button that saves information to user’s account
- Display confirmation message

Contribution Summary

This is a summary of what each team member contributed to this deliverable. Abena wrote the user stories for organizing monthly expenses based on category, editing monthly bills, tracking bills on a calendar, assigning a priority to each category, creating an alarm when the user gets close to the spending limit, and logging into the expense tracker. Michelle Bagnall wrote the user stories for registering a new account, creating a password, adding income amount to the account, adding bank information to the account, and adding bank loan amounts to the account. Ashawn organized the user stories into a table and edited the document where needed.

Ashawn Rose

Use Cases

Use Case 1

Use Case Name: Login

Actors: User

Preconditions: User must be registered with Expense Tracker

Flow of Events:

1. User types in their username and password.
2. User clicks the login button.
3. Expense Tracker system verifies username and password
4. Account verification succeeds.
5. System displays the expense tracker’s home page.

Post-conditions: The user logs onto their account.

Exceptions:

1. Account verification declines user
2. System send error message to user for invalid username and password
3. System redirects user to the login/sign-up page

Use Case 2

Use Case Name: Organize monthly bills based on several categories

Actors: User

Preconditions: User is registered and logged into their account.

Flow of Events:

1. User clicks on the “bills” link from their account page.
2. System displays a list of the user’s bills with newest bills first.
3. User hovers over the list filter button.
4. System displays drop-down menu with these options:
    a. Priority
    b. Type
    c. Highest Amount
    d. Lowest Amount
    e. Alphabetical
    f. Due date
5. User clicks on any of the options.
6. System displays the bills based on the selected filter.

Post-conditions: User can organize their monthly bills through the filters.

Exceptions:

1. System fails to display the bills page.
2. System displays an error page saying the website is down and to tell the user to try again later.
3. Filter drop-down menu fails to pop up when the user hovers over the filter button.
4. User clicks on the “customer support” link on the bottom of the bills page.
5. System displays the customer support page.

Use Case 3
Use Case Name: Add bank account information to expense tracker account
Actors: User
Preconditions: User is registered and logged into their account.
Flow of Events:

1. User clicks on the “Add Bank Account Info” button from the account page.
2. System displays the Add Bank Account Info web page.
3. User types their bank account number into the bank account number box.
4. User types their banking institution’s routing number in the routing number box.
5. User clicks on the “confirm” button.
6. System displays a confirmation page with a link back to the user’s account page.

Post-conditions: User has saved and connected their bank account information to their expense tracker account.
Exceptions:

1. User types in an bank account number containing alphabetical characters or symbols
2. System displays error message telling the user to type a bank account number with only numerical values
3. User types in a routing number with alphabetical characters or symbols
4. System displays error message telling the user to type a routing number with only numerical values

Non-functional Requirements

Portability- The system should be available to use for all versions for windows

Maintainability- easy to modify, so we can add features, remove unnecessary features and since we are using scrum we would need to continuously make changes to the code.

Reliability- since the expense tracker is calculating bills it needs to be reliable.In order for it to be successful.

Security - The expense tracker needs to be secure because it is handling very sensitive information like bank accounts, user’s personal information and payment information.

Contribution Summary

Abena completed the non-functional requirements section and worked on the use case for organizing bills based on categories. Michelle worked on the login use case and completed the use case for adding bank information to the user’s expense tracker account. Ashawn worked on the login use case and worked on the non-functional requirements with Abena. All three of us worked together on the second use case.

Michelle Bagnall
Abena Agyemang
Ashawn Rose

Architecture Design

Description

The Expense Tracker will be using the Three-Tier architecture for its design so that we can store our data on a database server, then have the clients access the download file from the application server. It will allow us to add security to our database server and protect our data from being stolen.

Task 2

Operational Requirements

Since this is a global web-based e-learning course It will be available worldwide we as a team would have to make sure it is fully operational for all customers around the world. Technology is always changing and improving so our system would need to be able to adapt to new updates and system changes so we can stay up to date and keep the app current.

System integration is an important component too because we need to be able to bring together if not one but multiple sub-systems to improve the customers experience. Our system should be able to do that without a problem. If they want to connect their email calendar or reminder to the course they should be able to.

Portability is another very important component to this e-learning course because our system needs to cater to all types of machines phones and tablets. It's going to be global and we don't use the same technology as people in other countries we we need to be able to display our course on all systems whether it is ios or android .

Maintainability there is always room for improvement so we need our system to be able to adapt to change and growth. If we can fine a way to make one or more features better we should be able to change it without messing up everything else . maybe we could add more languages or create and audio.

Performance Requirements

Efficiency is important in this e-learning course our system needs to have speed . make sure that we doing as much as we can in the shortest time possible because no one want to wait for buffering.

Capacity since this is a global course we cannot estimate how many people will be on at a certain time se we need to make sure our system can keep up with all the users whether there 10 or 1,000 users doing the same thing and estimate how much data they need.

Availability our system should be available at all times 24/7 and reliable it should always be running no matter what time.

Security Requirements
System Estimated Value, we should know how much is at stake if worst come to worst. How much value does the business hold?how much are we protecting in terms of money?
Access Control , of course we need to have some control of who is able to see what, who is allowed to modify and change things on our system. If something is changed accidentally or intentionally we need to minimize the risk by monitoring who has access to what.

Virus Control on the web we are going to run in to a series of viruses . we need to make sure our system and users are protected. By checking files for viruses before it is downloaded, by continuously running checks to make sure we haven't been infected.

Cultural Requirements
Multilingual, We need to make sure that our course is available in all languages since this is a global course.
Customization we need to be able to allow our users to customize their experience
We also want to make sure that currency and metrics are according to their specific country.

Selecting Architecture Design
For the web-based e-learning course, an architecture involving a 3-tier system would be beneficial. It would supply our service with a layer of security between our system and our clients. We would have a database for our e-courses to be drawn from.

Task 3

What do you recommend that the application include to ensure that it supports this global requirement?

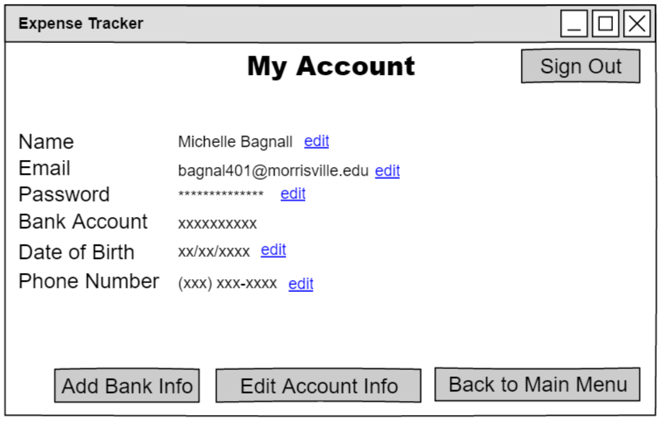
For Web-based application we should make sure it is available in different languages we need to also make sure that the performance is fast and efficient . We also need to make sure it can be displayed on other countries computers, tablets or phones and available to apply at anytime. Customize applications so that if applications are due 6pm in america it should be due 6pm in china too. Make unstated norms explicit such as the date format.

Contribution Summary
Abena worked on the majority of the Architecture Design section. Michelle added the section for which architecture design the team would use for the e-learning website. Ashawn added the section for Task 3.

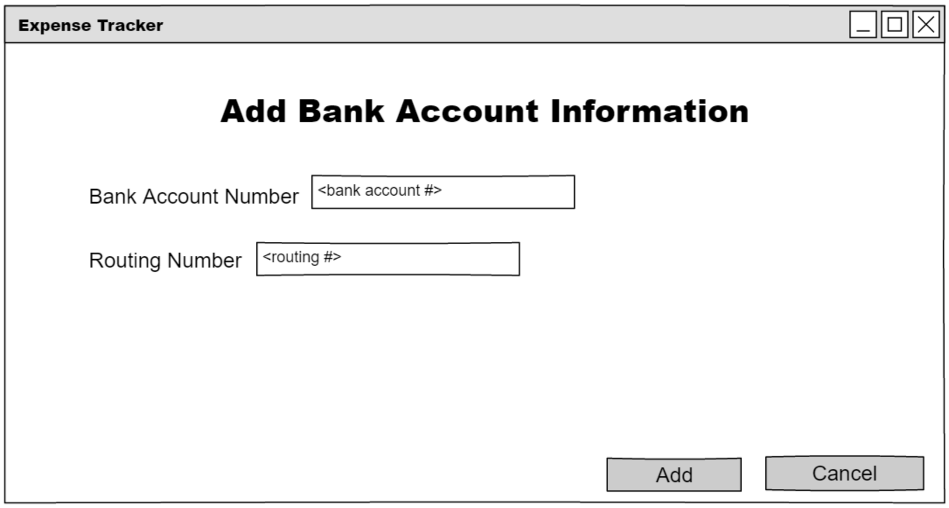
Abena Agyemang
Michelle Bagnall
Ashawn Rose

User Interface

Task 3. Adding Bank Account Information

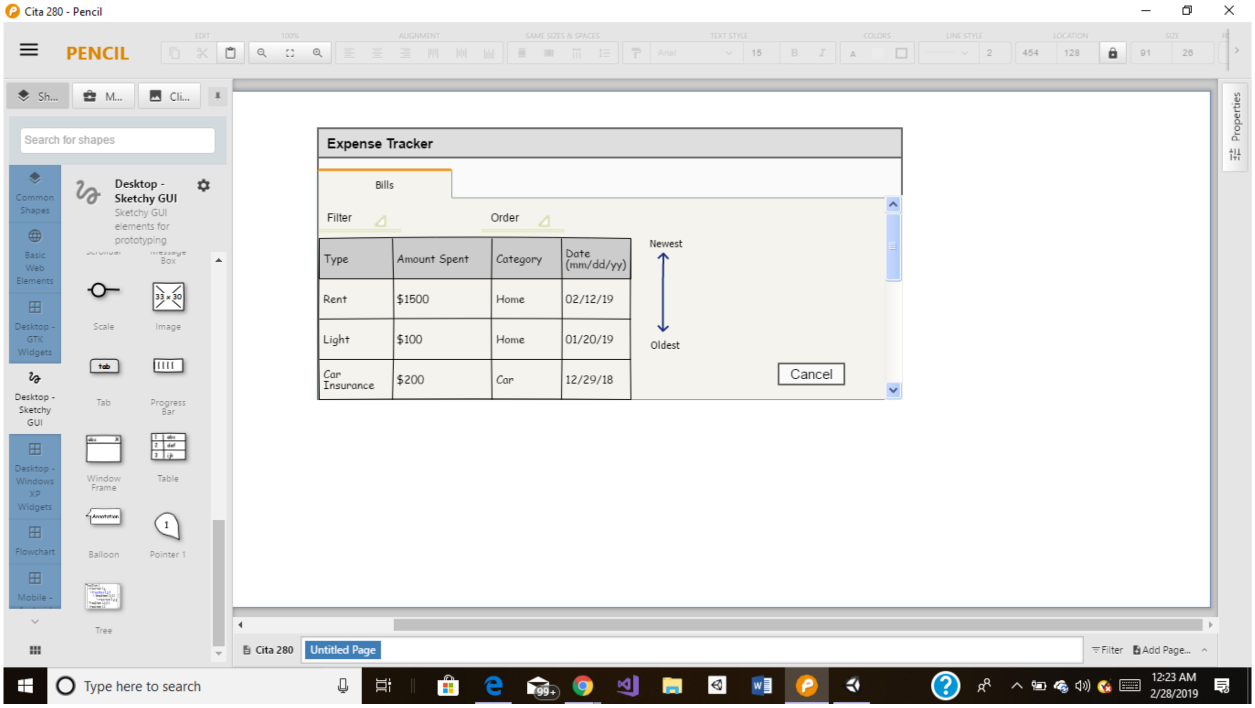


Name	Michelle Bagnall	edit
Email	bagnal401@morrisville.edu	edit
Password	**************	edit
Bank Account	xxxxxxxxxx	
Date of Birth	xx/xx/xxxx	edit
Phone Number	(xxx) xxx-xxxx	edit



Bank Account Number <bank account #>

Routing Number <routing #>



Expense Tracker

Bills

Filter

Order

Type	Amount Spent	Category	Date (mm/dd/yy)
Rent	$1500	Home	02/12/19
Light	$100	Home	01/20/19
Car Insurance	$200	Car	12/29/18

Newest

Oldest

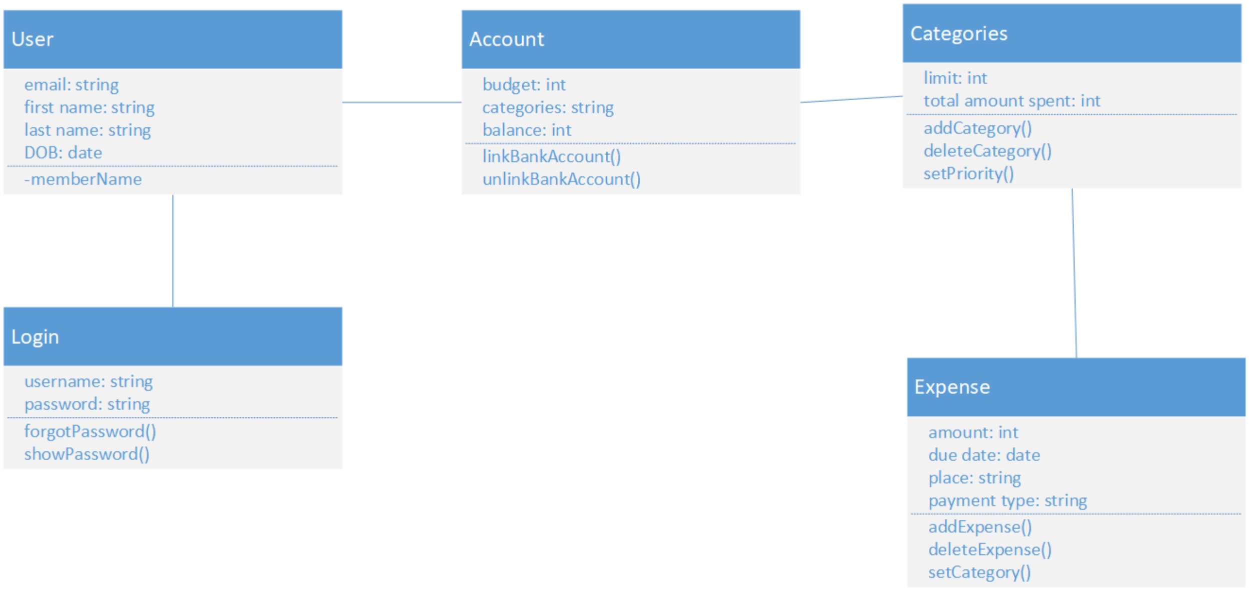
Cancel

Take pictures of Abena’s UI design and put them here

Contribution Summary

Abena Agyemang
Michelle Bagnall
Ashawn Rose

UML Class Diagram



Contribution Summary

Michelle Designed the UML Class Diagram. Abena came up with the contents of the diagram and Ashawn also came up with contents for the diagram.

Assignment 8

Our team's repository: https://github.com/bagnal401/cita280-team3

User Story: Login Page

Now break the user story into tasks

As a user I want to be able to register a new account so that my expenses are associated with my account.	10 Days	High Priority

As a user I want to be able to create a password for my account so that only I can access my expense tracking account.	10 Days	High Priority

As a user I want to login to my expense tracker account so that I can access the expense tracker features.	10 Days	High Priority

Story Tasks

Logging in & verifying

- Create code to check to if user is already logged in
- Create code to redirect to login page
- Design the login page
- Create function that calls all of the above, verifies redirected on not logged in

Use Case 1

Use Case Name: Login

Actors: User

Preconditions: User must be registered with Expense Tracker

Flow of Events:

6. User types in their username and password.
7. User clicks the login button.
8. Expense Tracker system verifies username and password
9. Account verification succeeds.

10. System displays the expense tracker’s home page.

Post-conditions: The user logs onto their account.

Exceptions:

4. Account verification declines user
5. System send error message to user for invalid username and password
6. System redirects user to the login/sign-up page