SEN3006
Software Architecture
Project Report

Group No: 29
Project Title: POKEMON

Lab Section No:	**Student ID:**	**Student Full Name:**
903	1728978	Ceyda SEVİLMİŞ
902	1730162	Ece Deniz KÖKSAL

1. Introduction
   1.1Purpose/Project Proposal

   Pokéwiki is a website that mainly works as an information source for learning about Pokemons. It contains several databases, filter systems, a table that holds more than 800 Pokemons' information, a couple of Pokemon pages that has more information about well-known Pokemons, login and register pages, a mini quiz page, a card shop, and a card collection for every user. The website has register, login, and logout options. The user can access to quiz, shop, and collection pages once they login to the system. The system warns the user if they want to access to the quiz and collection pages without login, and it also warns when the user enters wrong or blank input to various input fields such as username, password, quiz answer etc. According to their answers to the quiz, the system gives Pokécoins to the user and the user may use them to buy Pokemon cards from the Shop. Pokéwiki also warns the user if they want to buy a card without sufficient coins.

   With this website, we aim to become a safe and fun environment for the fandom. We decided to develop this project as a website because of the increased possibility of connecting easier with the user. We know that more people prefer/can access a website/internet rather than downloading an application to their digital devices. Therefore, we eliminate the option to develop the project as a desktop application using C# or Java.

   1.2 Software Language/ Project Environment

   While developing Pokéwiki, we used HTML (as the structure layer), CSS (as the styles layer), and JavaScript (as the behavior layer) for client-side programming. For server-side programming, we used PHP and Mysql with the help of Apache Server and PhpMyAdmin as our development environment.

1.3 Work Partitioning

We divided the tasks equally and helped each other throughout the project. Before starting the implementation, we decided briefly on the main structure and functions together. We added functionalities, always consulting each other beforehand, as the implementation went on.

A list showing all business events to which the work.

Name	*Role*	*Date*	*Description*
Ceyda Sevilmiş	*Front-end development* *User-interface design*		*-Design of the pages using CSS* *-Design of the navigation bar* *-Inserting Kaggle data to database and displaying on website using PHP* *-Filter systems for the main table on the homepage using JavaScript* *-The extra Pokemon pages and adding their links to their names on the table using JavaScript* *-Inserting quiz questions to the database* *-Finding and adding icons and images to the website*
Ece Deniz Köksal	*Back-end development* *Database development*		*-Management of all HTML Forms(login, register,…) and their actions(starting/ending sessions,…)* *-Linking the pages and the navigation bar to them* *-Inserting Card Images to database and displaying them on both Collection page and Shop page* *-Error Handling/Messages*

1.4 References

General help with HTML, CSS, JS, PHP, Mysql: https://www.w3schools.com/

For our data: https://www.kaggle.com/abcsds/pokemon

General help with information for extra pokemon pages: https://www.pokemon.com/us/

Quiz questions: https://www.thequiz.com/50-questions-every-pokemon-trainer-should-be-able-to-answer/

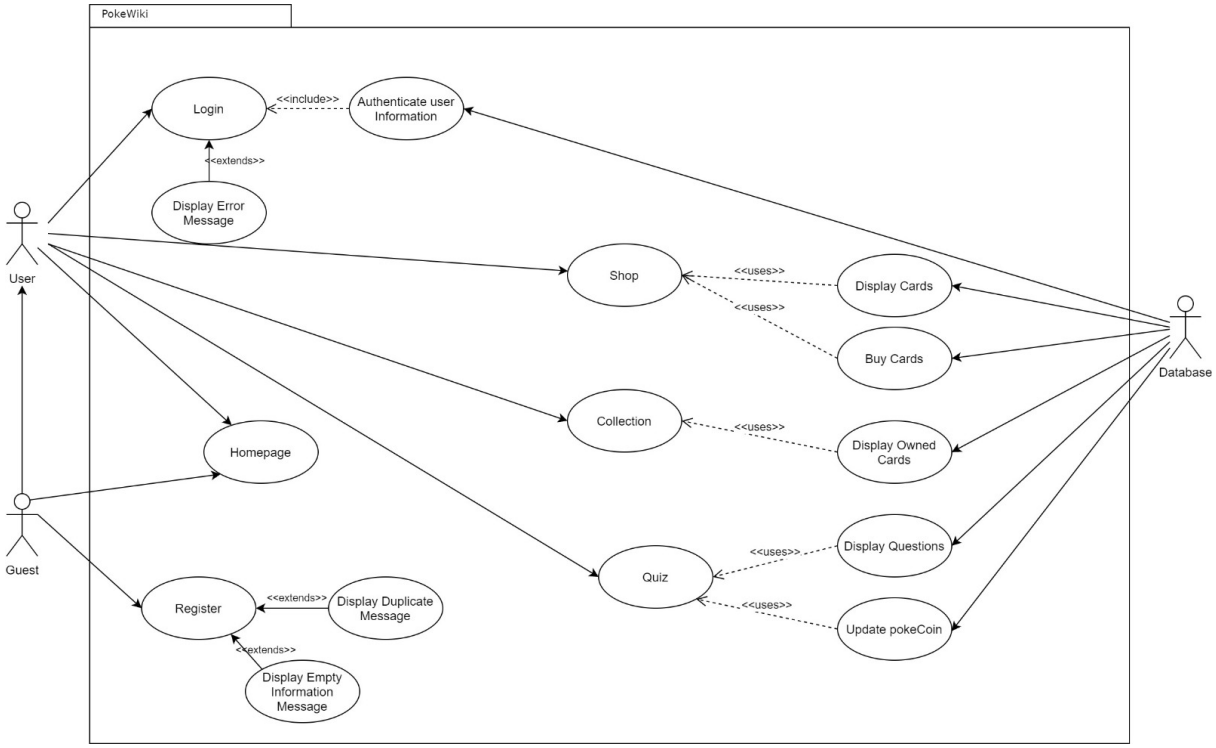
2. Architectural Goals and Constraints

Risk	Likelihood	Severity	Risk Score	Management of Risk
Hardware fail	Possible	Major	High Risk(12)	Replacement of the failed piece(s) ASAP
Inconsistency between our codes	Likely	Minor	Moderate Risk(8)	Communication, research and fixing
Lack of knowledge	Likely	Major	Extreme Risk(16)	Research in lacking fields
Unanticipated changes in requirements or functions	Possible	Moderate	Moderate Risk(9)	Requirement analysis
Database error	Possible	Major	High Risk(12)	Check the code for mistakes

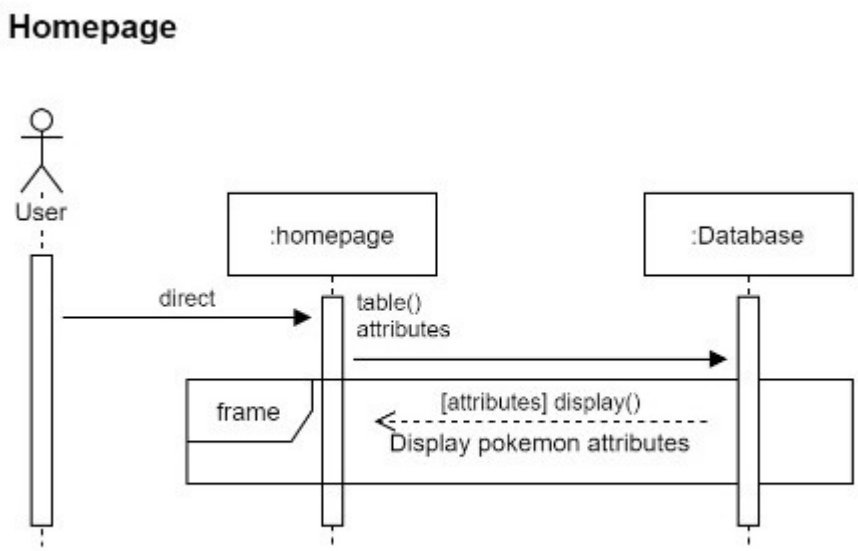
LIKELIHOOD	SEVERITY				
	Insignificant (1)	Minor (2)	Moderate (3)	Major (4)	Catastrophic (5)
Almost Certain (5)	Moderate Risk (5)	High Risk (10)	Extreme Risk (15)	Extreme Risk (20)	Extreme Risk (25)
Likely (4)	Moderate Risk (4)	Moderate Risk (8)	High Risk (12)	Extreme Risk (16)	Extreme Risk (20)
Possible (3)	Low Risk (3)	Moderate Risk (6)	Moderate Risk (9)	High Risk (12)	Extreme Risk (15)

3. Architectural Representation

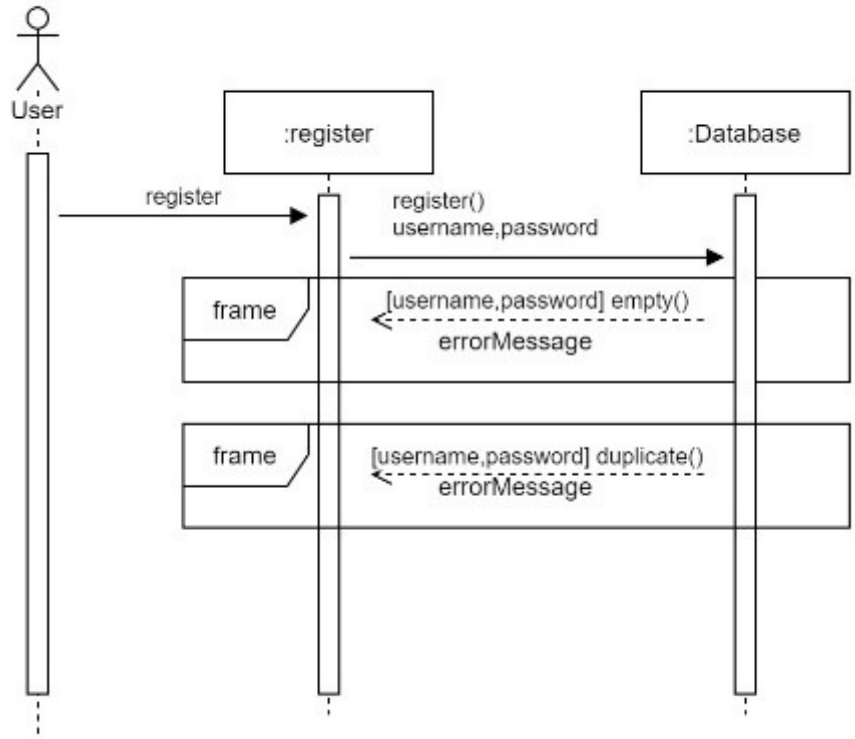
3.1 Use Case Diagram



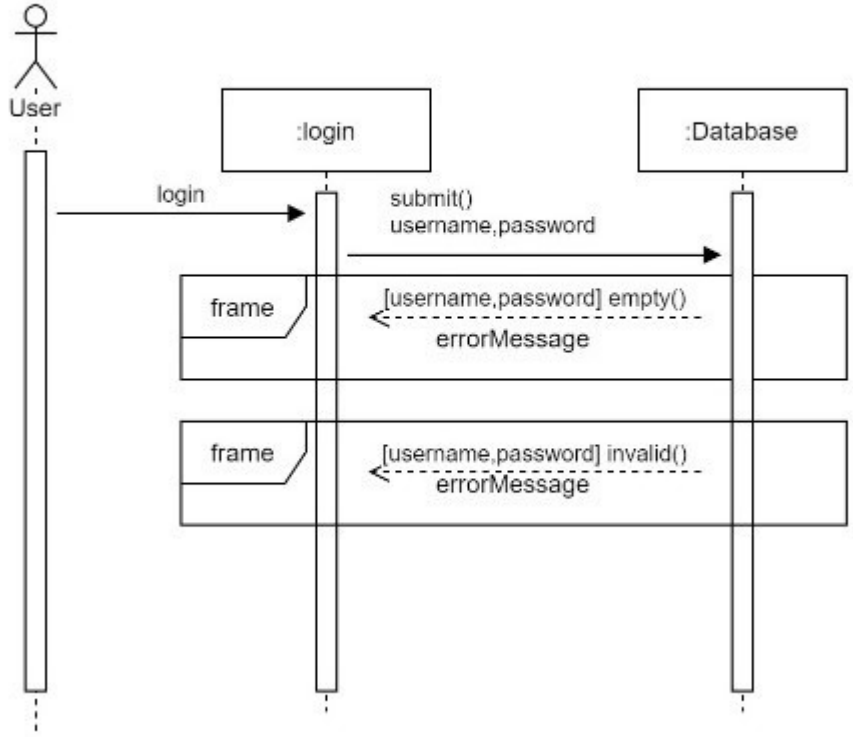
3.2 Sequence Diagram



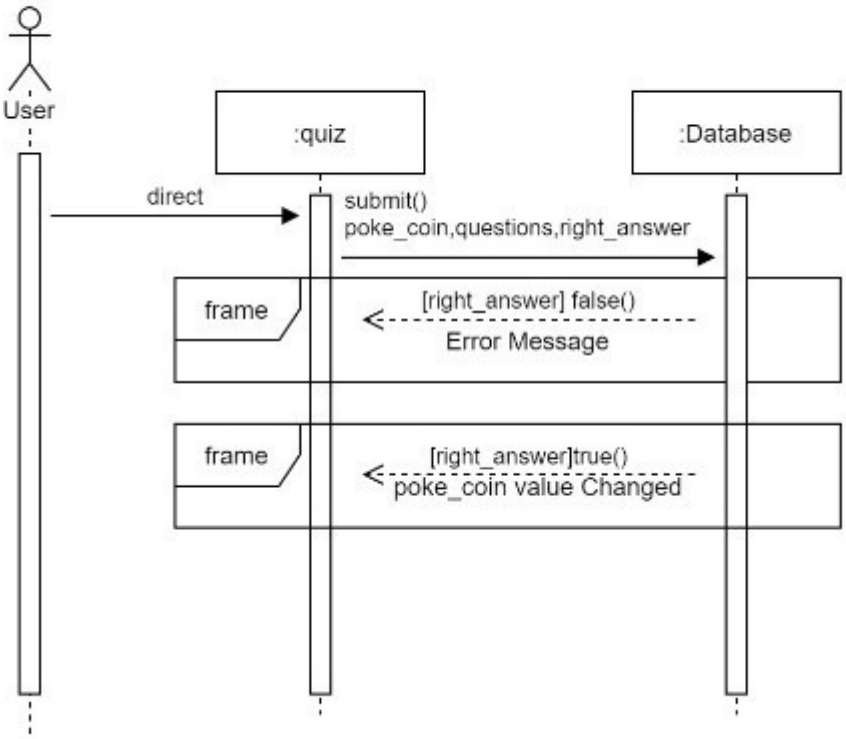
Register



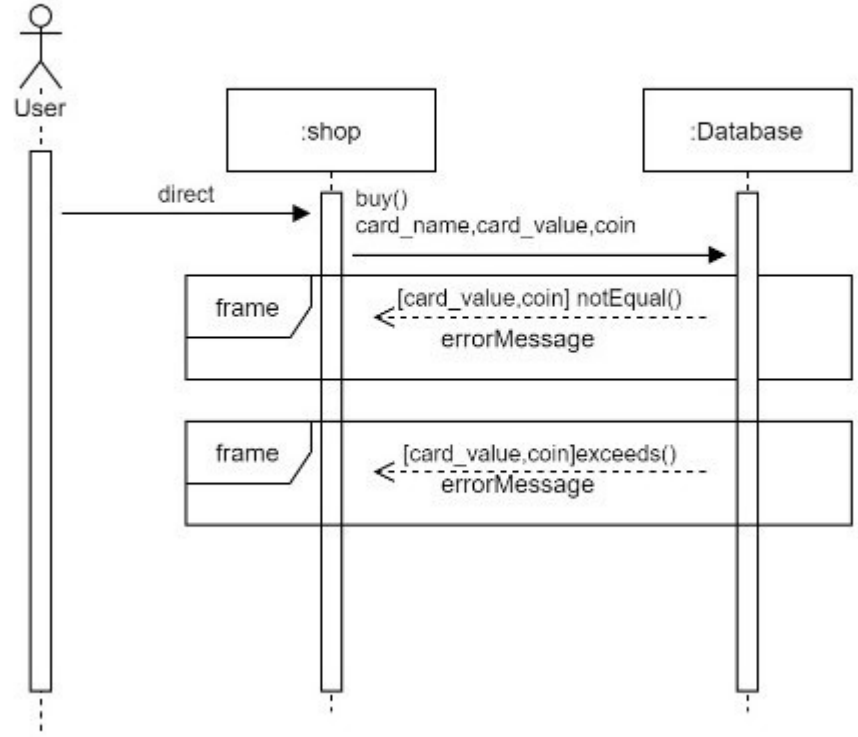
Login



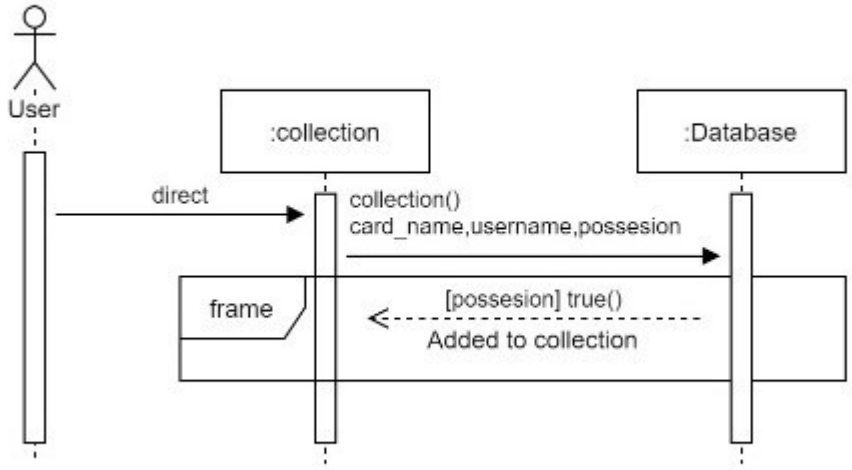
Quiz


User
:quiz
:Database
direct
submit()
poke_coin,questions,right_answer
frame
[right_answer] false()
Error Message
frame
[right_answer]true()
poke_coin value Changed

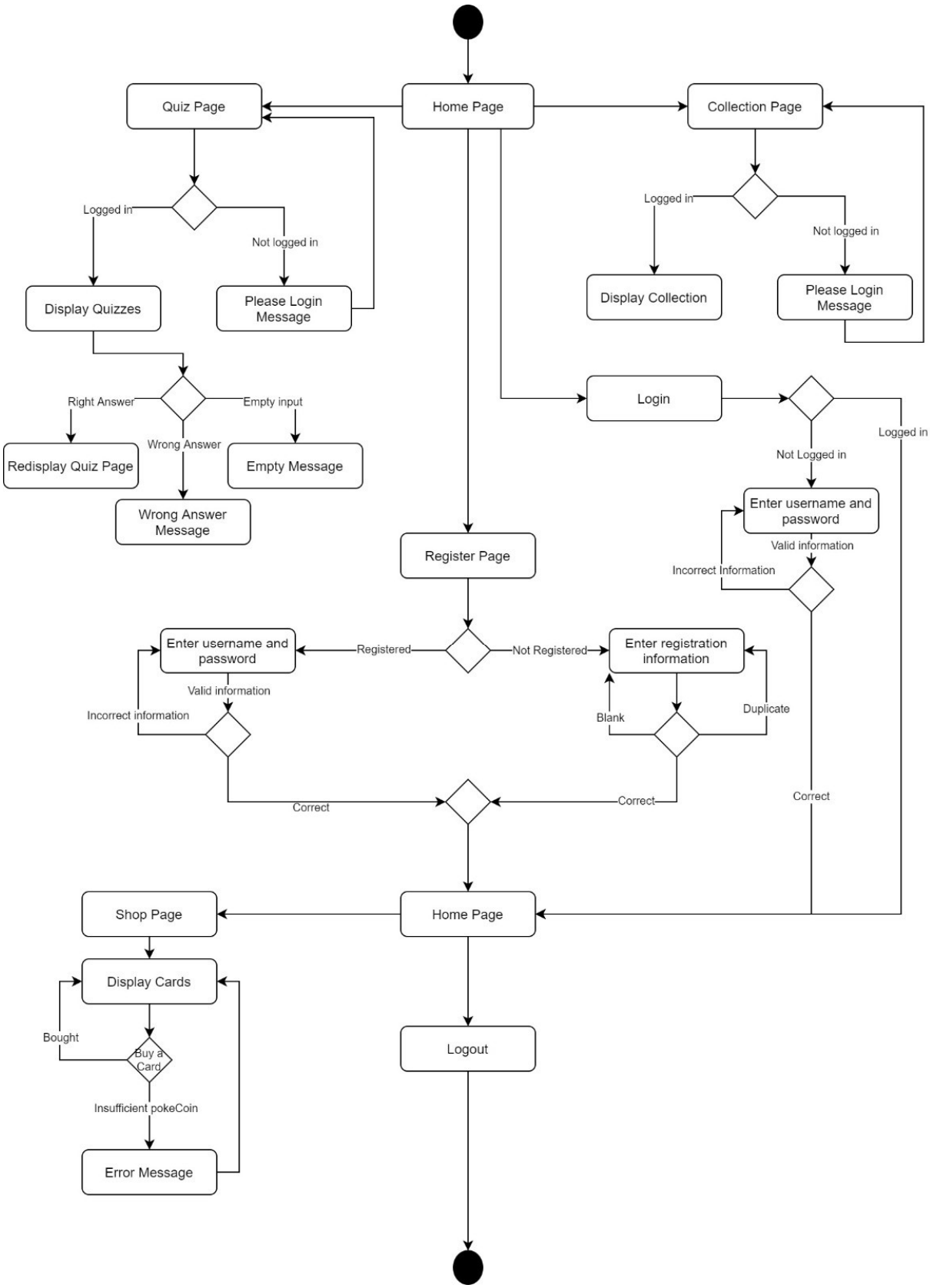
Shop


User
:shop
:Database
direct
buy()
card_name,card_value,coin
frame
[card_value,coin] notEqual()
errorMessage
frame
[card_value,coin]exceeds()
errorMessage

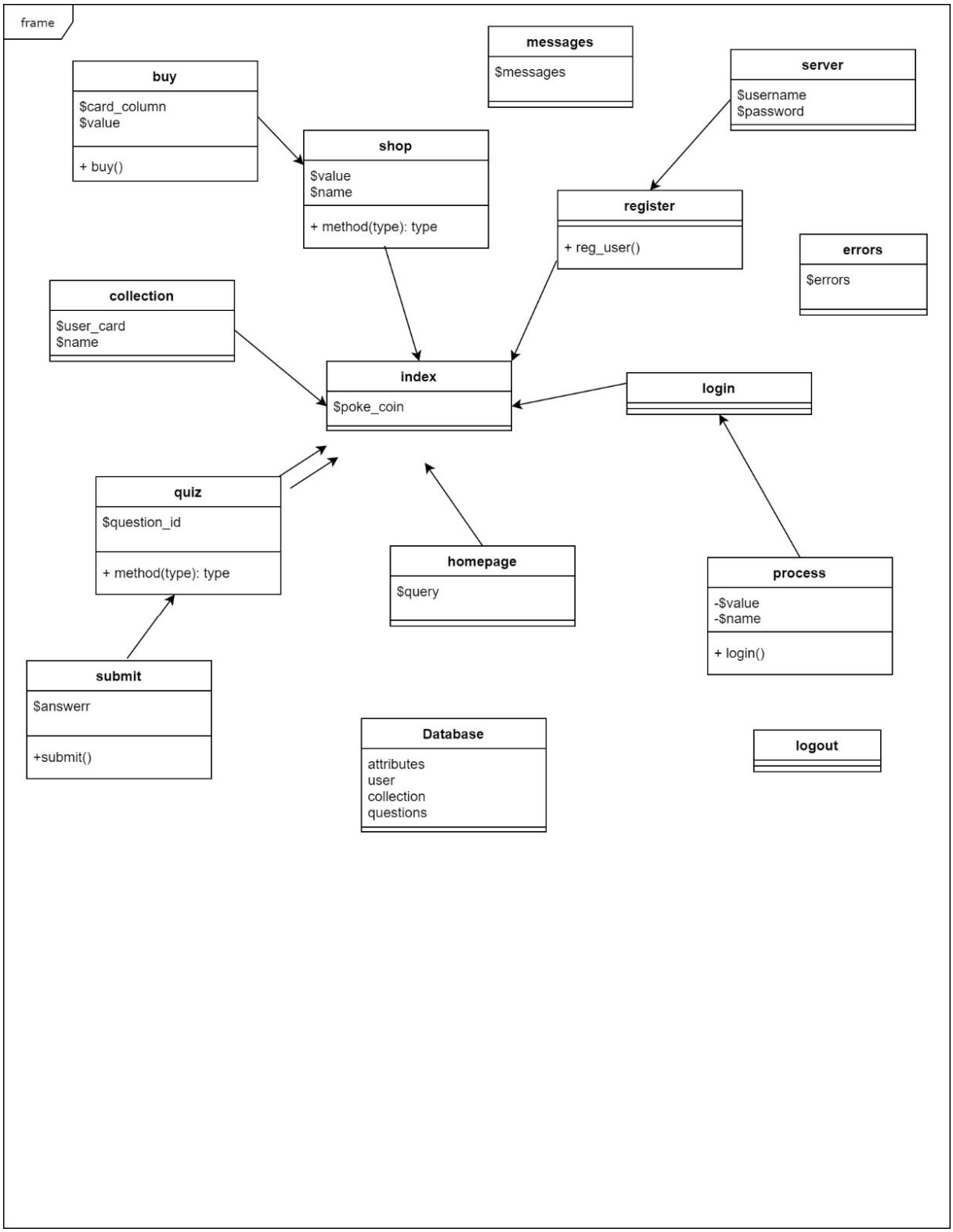
Collection


User
:collection
:Database
direct
collection()
card_name,username,possesion
frame
[possesion] true()
Added to collection

3.3 Activity Diagram

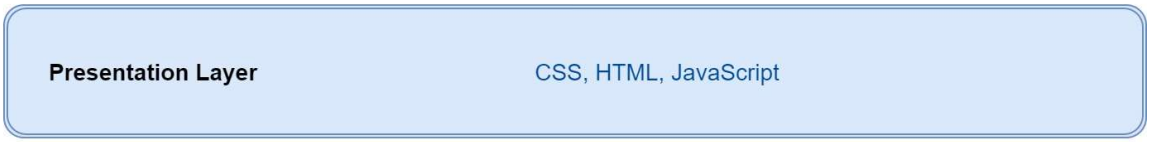


3.4 Class Diagram

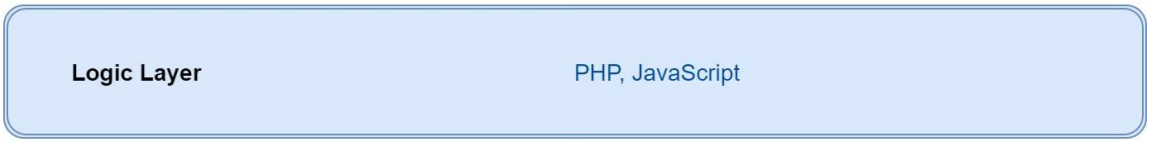


4. Software Architecture Patterns/Layers

4.1 Presentation / View / Interface Layer



4.2 Controlling / Processing / Logic Layer



4.3 Data / Model Layer

5. Conclusion / Summary

   Briefly, we talked about what is our project, what functions does it have, which elements did we used while implementation, and how did we manage the process. Pokéwiki is a website that you can access without registration. That way you can view information about Pokemons. By being a user, you can enjoy quizzes and earn coins. Then you can buy cards with those coins and gather those cards in your collection. Just like how Pokemon Trainers gather Pokemons and add them to their collection.

   Shop and Collection uses the same database and this provides us with a way of validation for purchases. Another validation is for users, we also store them in the database as well as Pokemon attributes and Quiz questions.

   We would like to note that before this project we both only knew some basic HTML and CSS. This was a real challenge for us but at the end we both learned so much. Thank you for this opportunity.