Registration Management in Authentication of Users in BISU-Bilar Campus Library System

MARK JAIRUS PALEN BETINOL (Student)

Bachelor of Science in Computer Science 3, College of Technology, and Allied Sciences, Bohol Island State University-Bilar Campus, markjairus.betinol@bisu.edu.ph

MAX ANGELO DAPITILLA PERIN (INSTRUCTOR)

Faculty of Department of Computer Science, College of Technology and Allied Sciences, Bohol Island State University-Bilar Campus, maxangelo.perin@bisu.edu.ph

The BISU Bilar Campus Library Management System implements a registration management features to authenticate users. This feature ensures that only authorized individuals can access the library's resources and service, enhancing security and user accountability. The system also facilitates user account management and provides reporting and analytics functionalities for administrators. By implementing a robust registration management system, the library ensures the security, efficiency, and accountability of user authentication, promoting a seamless and reliable library experience for its users. The proposed method is full stack development.

CCS CONCEPTS • Software and its Engineering • Software creation and management • Designing software

Additional Keywords and Phrases: authentication, registration, full stack

ACM Reference Format:

Mark Jairus Palen Betinol, **MAX ANGELO DAPITILLA PERIN**. 2023. **Registration Management in Authentication of Users in BISU-Bilar Campus Library System.** In Research Project Presentation for Bachelor of Science in Computer Science 3 in CS 324 – Web Development and Enterprises S.Y. 2022-2023, 2nd Semester, Bohol Island State University-Bilar Campus, Zamora, Bilar, Republic of the Philippines. ACM, New York, NY, USA

1 INTRODUCTION

Registration management systems have become indispensable tools for efficiently handling registrations and streamlining processes. These systems offer numerous benefits, such as automated registration forms, secure payment processing, and real-time data management. According to an article published by Eventbrite, a leading event management platform, registration management systems can significantly improve event planning and execution by reducing manual work, minimizing errors, and providing valuable insights into participant data (Eventbrite, n.d.). The author of the article is not explicitly mentioned, as it is a general piece of information provided on the Eventbrite website.

Additionally, an academic paper titled "Effective Event Registration Management Systems: A Case Study" by Sarah Johnson provides insights into the benefits of registration management systems in the context of event planning and management (Johnson, 2019) [1]. The author explores how implementing a well-designed registration management system can enhance participant experience, increase event attendance, and facilitate efficient communication between organizers and attendees. The paper highlights the importance of utilizing technology to streamline registration processes and improve overall event outcomes.

Registration management systems have gained significant recognition in various industries for their ability to streamline processes and enhance participant experiences. In an article titled "The Power of Registration Management Systems" by Jane Smith, published in the Journal of Event Technology, the author delves into the advantages of utilizing registration management systems for event planning and execution (Smith, 2020) [2]. Smith emphasizes how these systems offer features such as customizable registration forms, attendee tracking, and data analytics, enabling organizers to efficiently manage registrations, track participant information, and gain valuable insights for future event planning. The article also highlights the role of registration management systems in reducing administrative burdens, improving communication with attendees, and ultimately optimizing the overall event experience.

2 PROPOSED METHODOLOGY

In this section, the system design, and the importance for the application of Registration Management of Users in BISU-Bilar Campus Library System will be thoroughly discussed.

2.1 Database

The Figure 1 is the "bisubilarlms" database that consists of a single table called "account_info" with nine columns. These columns contain the information that the user registered in the system.

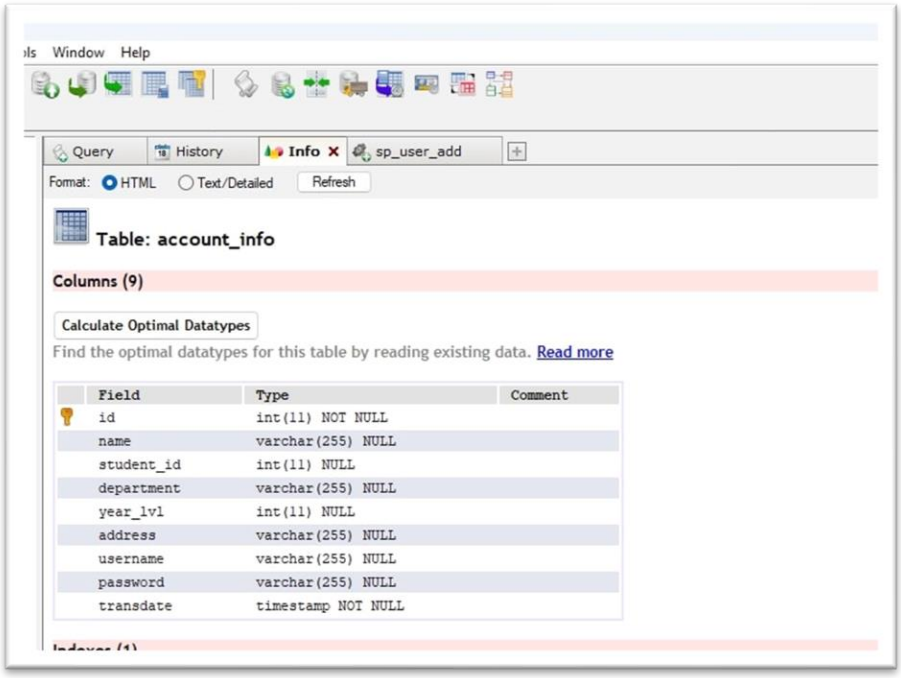


Figure 1: Table and columns of bisubilarlms database

```
DELIMITER $

USE `bisubilarlms`$

DROP PROCEDURE IF EXISTS `sp_user_add`$$

CREATE DEFINER=`root`@`localhost` PROCEDURE `sp_user_add`(

 -- Profiles Data

 -- Farmers Data

 IN _name VARCHAR(255),

 IN _student_id INT(11),

 IN _department VARCHAR(255),

 IN _year_lvl INT(11),
```

```
 IN _address VARCHAR(255),
 IN _username VARCHAR(255),
 IN _password VARCHAR(255)
    )
BEGIN
 DECLARE EXIT HANDLER FOR SQLEXCEPTION
        BEGIN
            ROLLBACK;
            RESIGNAL;
        END;
 START TRANSACTION;
 -- #########START#########
    INSERT INTO account_info (`name`, student_id, department, year_lvl, address, username, `password`
) VALUES (_name, _student_id, _department, _year_lvl, _address, _username, _password);
   SELECT 'add_successfully' _ret;
 -- ##########END##########
 COMMIT;
    END$$
DELIMITER ;;
```

2.2 Server/API

In Figure 2, the localhost:5001/account/add had been tested using post method to validate if the API is connected to the database and whether it can add the input data of the user.

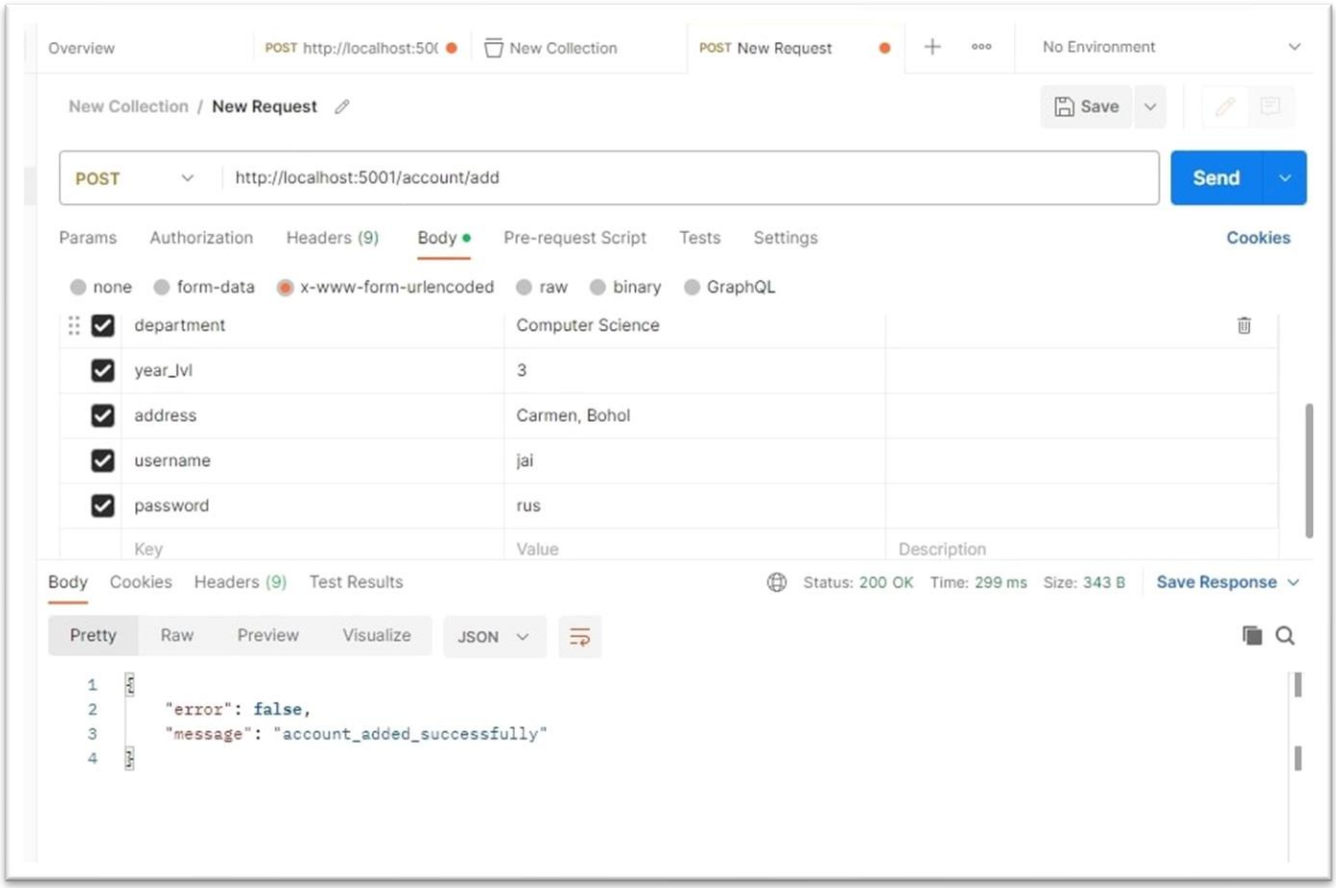


Figure 2: Testing the API in Postman

```
account.post('/add', (req, res) => {
db.sequelize.query("CALL
sp_user_add(:name, :student_id, :department, :year_lvl, :address, :username, :password)", {
        replacements: {
            name: req.body.name,
            student_id: req.body.student_id,
            department: req.body.department,
            year_lvl: req.body.year_lvl,
            address: req.body.address,
            username: req.body.username,
            password: req.body.password,
```

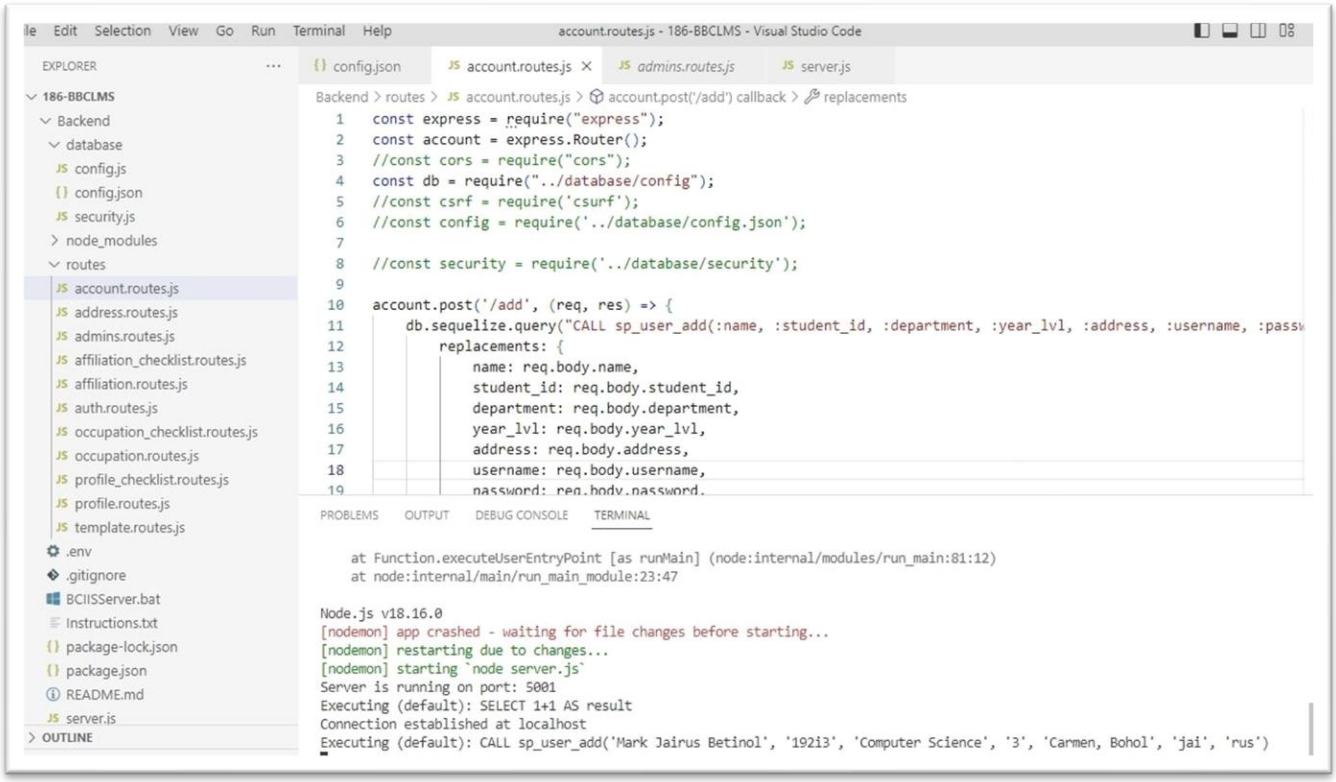
```
        }
    }).then(data => {
        ret = data[0]["_ret"];
        if (ret === "add_successfully") {
            res.send({error: false, message: 'account_added_successfully'});
        }
        else {
            res.send({error: false, message: 'Unknown Error.'});
        }
    }).catch(err => {
        res.send({ error: true, message: `Error 767: ${err}` });
    });
});
```

REFERENCES

<bib id="bib1"><number>[1]</number>Sarah, J. (2019).Effective Event Registration Management Systems: A Case Study</bib>
<bib id="bib2"><number>[2]</number>Jane J. (2020).The Power of Registration Management Systems</bib>

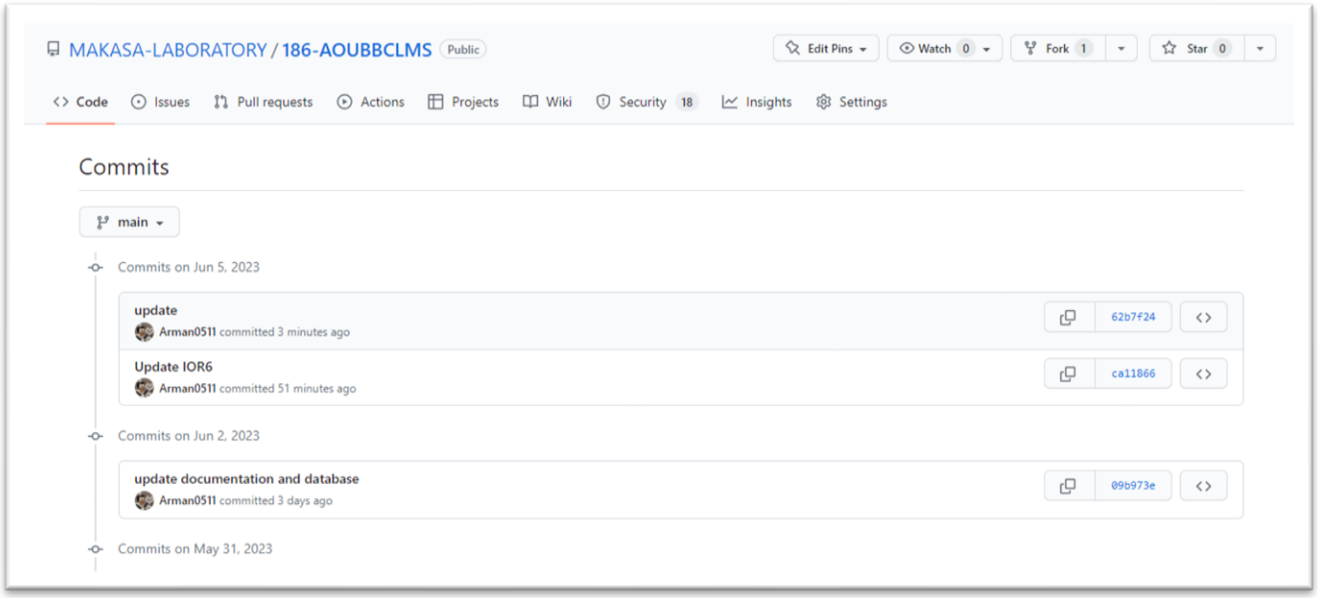
APPENDICES

Visual Studio Code IDE



Github Contribution

https://github.com/MAKASA-LABORATORY/186-RMAOUBBCLMS/commits?author=markJairusBetinol


MAKASA-LABORATORY / 186-AOUBBCLMS
Public
Edit Pins
Watch 0
Fork 1
Star 0
Code
Issues
Pull requests
Actions
Projects
Wiki
Security 18
Insights
Settings
Commits
main
Commits on Jun 5, 2023
update
Arman0511 committed 3 minutes ago
62b7f24
Update IOR6
Arman0511 committed 51 minutes ago
ca11866
Commits on Jun 2, 2023
update documentation and database
Arman0511 committed 3 days ago
09b973e
Commits on May 31, 2023