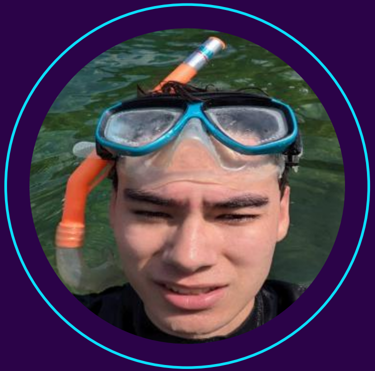
STUDENT RESIDENCE MANAGEMENT SYSTEM

By: Brock Young, Jose Contreras, Alikhan Madeni, and Ryan Cropley

OUR TEAM



BROCK

Grunt

RYAN'S DOG

Ate his photo

JOSE

Gacha Player

ALIKHAN

Wanted to be a woman

INTRODUCTION

- Developed a Student Residence Management System (SRMS) and related documentation
- Address need for modern management tool
- Increase key factors such as efficiency, reliability, modifiability, accessibility, scalability, reusability

DESIGN PROBLEM

- University requires new software solution to replace existing SRMS
- Current paper-based system has issues relating to
  - Limited scalability
  - Loss of documents
  - Large effort for basic tasks
  - Minimal automation
  - Lack of easy access to records
- Designed solution aims to address above issues with minimal cost and satisfying all design requirements

OBJECTIVES & REQUIREMENTS

Objectives

- Availability & Reliability
- Modifiability
- Scalability
- Reusability

Requirements

- User Management
- Payment Services and Billing
- Residence Management
- Security & Privacy

CONSIDERED SOLUTIONS

Solution 1

Initial Logical Model:

- Simple design with basic relationships
- Managers oversee multiple students, each student only has 1 manager
- No controller class

Solution 2

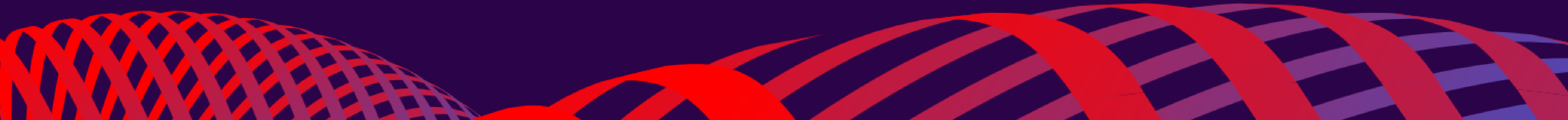
Improved Structure with Controller:

- Introduced the SRMS controller class
- Removed Bed class to instead make an Bed array within the Residence class
- Controller class is missing some required features

Solution 3

Prototype for Final System:

- Controller class updated with all required features
- Introduced Exception handling
- Defines core class structures and key functionalities
- Prototype created to view basic user interaction with the system



FINAL SOLUTION

- Solution 3 was the most complete
- Exception handling
- Descriptive class diagram
- Highest evaluation against decision matrix

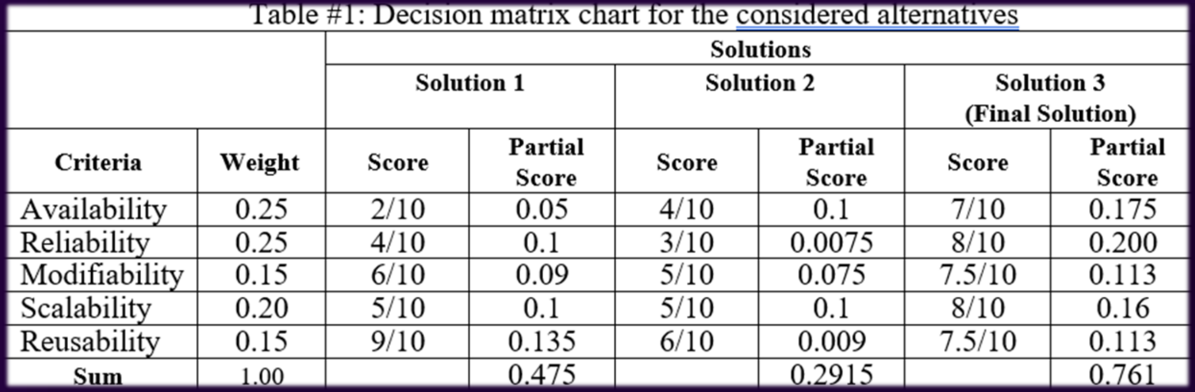


Table #1: Decision matrix chart for the considered alternatives

		Solutions					
		Solution 1		Solution 2		Solution 3 (Final Solution)	
Criteria	**Weight**	**Score**	**Partial Score**	**Score**	**Partial Score**	**Score**	**Partial Score**
Availability	0.25	2/10	0.05	4/10	0.1	7/10	0.175
Reliability	0.25	4/10	0.1	3/10	0.0075	8/10	0.200
Modifiability	0.15	6/10	0.09	5/10	0.075	7.5/10	0.113
Scalability	0.20	5/10	0.1	5/10	0.1	8/10	0.16
Reusability	0.15	9/10	0.135	6/10	0.009	7.5/10	0.113
Sum	1.00		0.475		0.2915		0.761

FEATURES

Add Student

addStudent()

Add Manager

addManager()

Assign Manager

assignManagerToStudent()

Unassign Manager

dropAssociation()

Display Empty Beds

displayEmptyBeds()

Assign Beds

assignBed()

Remove Student

releaseStudent()

Display Info

systemState()

LIMITATIONS

Lack of database

No Remote Access

No login portal

Text based user interface not GUI

Lack of data encryption

Lack of Payment services / Billing

This is an early prototype model that only contains the essential features.

Therefore we are missing some of the non-essential features

Search for users or managers

No tutorial for navigation

CONSIDERATIONS

ENVIRONMENTAL

- Reduces paper waste
- Energy efficient code
- Proprietary software
- Efficiently scalable

SOCIETAL

- Improve access to housing
- Modernize university image
- Reduce tedious tasks for staff
- Enhance student experience

SAFETY & SECURITY

- Reduce risk of data loss
- List of managers & students
- Handle exceptional cases
- Version control for code safety

ECONOMIC

- Save future costs
- Satisfy budget constraints
- Dynamically scalable
- Code reuse in future systems

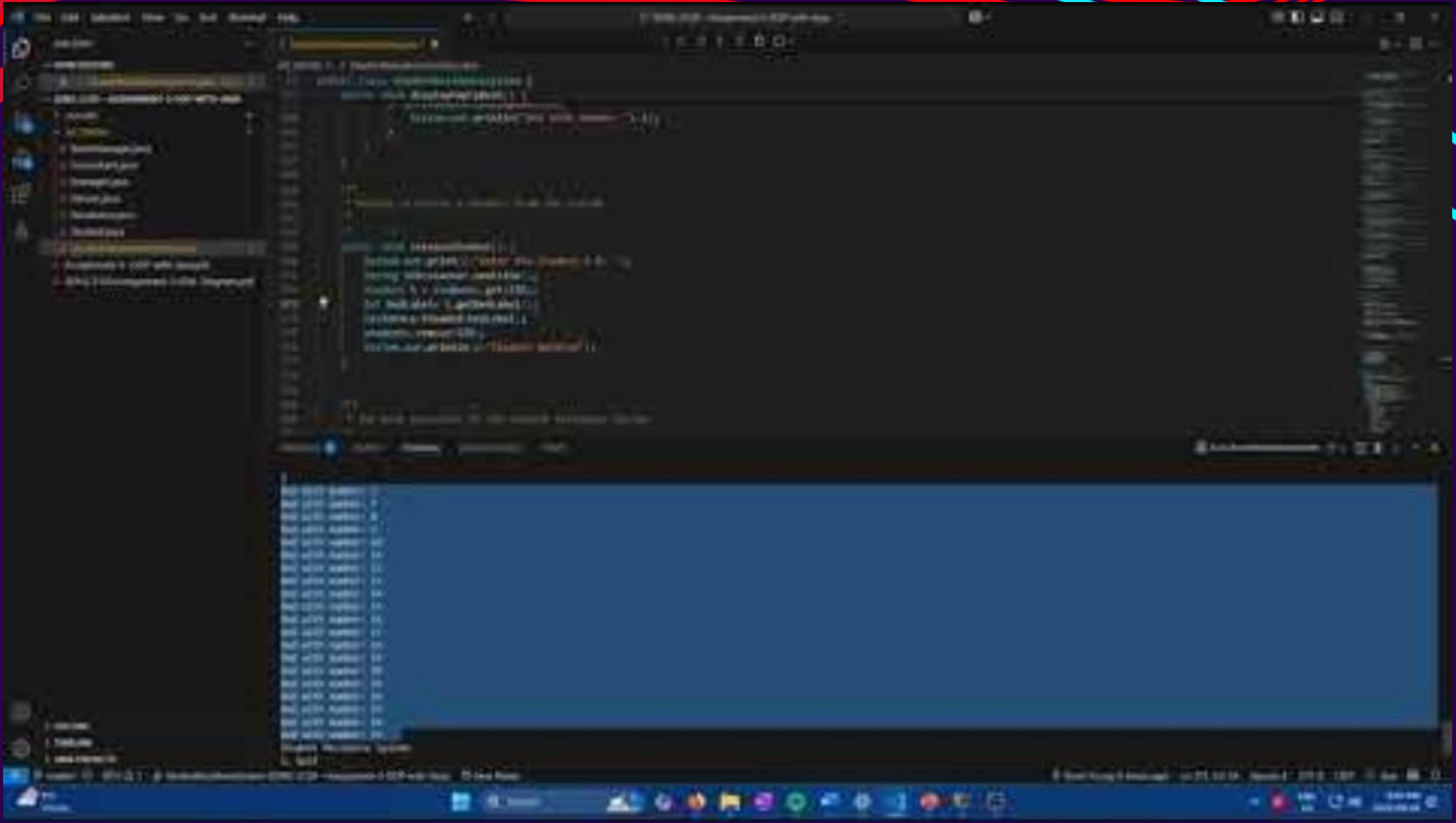
CONCLUSION & FUTURE WORK

CONCLUSION

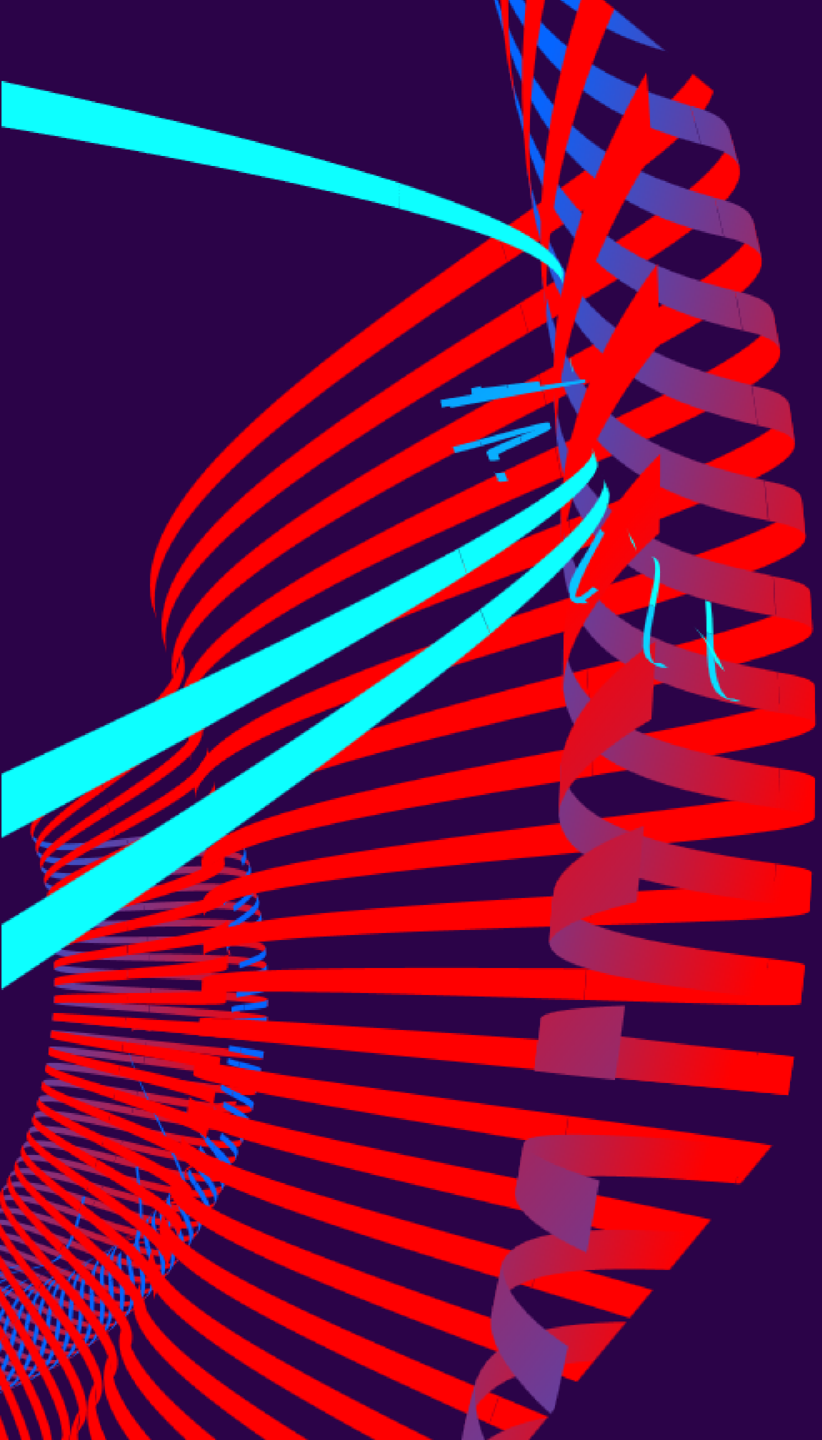
- Solution served as critical step in advancing universities technological infrastructure & management capabilities
- Best solution was evaluated from comparison matrix based on key objectives
- Key functionality was established showing successful feasibility study for future iterations

FUTURE WORK

- Data Persistence
- User Interface
- Security Enhancements
- Payment Services & Billing
- Advanced Search and Filtering
- Input Validation and Error Handling
- System Remote Access



PROTOTYPE VIDEO



THANK YOU