

Milestone 3: Test Plan & Prototype

for Music School

Prepared by

Team's Name: AI

Ang Wee Yi, Alex	2102852
Iphigene Peh Yureng	2102741
Iphigene Peh Yuxi	2102744

Lab Group: P2-9

Github handle: ICT2201-P2-9

Date: 26/11/2022

Table of Contents

1 Introduction	3
1.1 Product Scope	3
1.2 Intended Audience and Document Overview	3
1.3 References and Acknowledgments	4
2 Software Design	5
2.1 Updated Sequence Diagram	5
3 Black-box Testing	6
4 Whitebox Testing	28
5 Project Prototype & Wireframes	33
5.1 Login and Forgot Password Page	33
5.2 Manager Pages	35
5.3 Staff Pages	37
5.4 IT Administrator Pages	39
6 Project Management	43
6.1 Updated Gantt Chart	43
6.2 Updated work breakdown structure	43
7 Individual Members Task Reflections	44
8 Appendix A – Use Case Descriptions	45
9 Appendix B - Overall Class Diagram	52
10 Appendix C - Updated Sequence Diagram	53
11 Appendix D – Data Dictionary	54
12 Appendix E – Test Cases Result	56

1 Introduction

In the music school, work allocations are manually assigned monthly to teachers by the managers. Hence, to facilitate better workload allocation, the team has built a web-based workload management system to show an overview of the company's manpower strength at any time and informative availability of employees and engagement. The system will provide an interactive and informative way for both employees and managers to visualise and allocate their workload optimally which will in turn improve work-life balance and make the workplace environment more conducive and attractive. The following subsections are organised as such: Subsection 1.1 describes the scope of the product. Subsection 1.2 shows the related background literature. Subsection 1.3 discusses the intended audience and the overview of the document. Finally, subsection 1.4 included references and acknowledgements.

1.1 Product Scope

The scope of the product is to create a web-based application that will be utilised by managers, staff and IT administrators for workload allocation in the Music School.

The employees of Music School should be able to see their job assignments and working hours engaged/assigned. They will also be able to indicate their availability up to one month earlier and indicate if they cannot fulfil any assigned jobs. This product would greatly benefit the employees as they will now have a clear overview of their work schedules, allowing them to plan both days off and working days conveniently.

In addition, the managers should be able to visualise the manpower availability up to one month earlier, as well as job assignments and allocate jobs. This product would benefit managers as they are able to view regular updates on staff availability and engagement more effectively, allowing them to follow up and plan an optimised workload schedule.

Furthermore, the IT administrators will oversee staff and managers accounts by creating, editing or removing them. This product would be useful for the IT administrators as they could manage accounts more efficiently as all accounts are in a centralised system.

1.2 Intended Audience and Document Overview

The intended audience of this document are the client and the professor. In this project, the client refers to an entity that has engaged the team to design and build a custom software according to their requirements and specifications. The professor would provide relevant and essential feedback throughout the project and documentation process for the team and ensure this document meets the stated requirements and specification.

The recommended sequence for reading the document is:

1. Section 1 offers a good overview and understanding of the entire project.
2. Section 8, 9, 10, and 11. These sections provide useful information regarding individual use cases descriptions and a data dictionary that consists of the different terminology used in this document. Apart from that, section 9 and 10 also include important updates from M2.

3. Section 2 illustrates the specific requirements of the product which comprises the requirement elicitation methods, product overview, product functionality and requirements, and updated use case model from milestone 1 and 2.
4. Section 3 details the black box testing for the project. The black box testing phase consists of test scenarios, test cases, test data, expected results, steps taken to achieve the testing as well as decision tables and system state diagrams.
5. Section 4 illustrates the white box testing phase for this project. White box testing includes pseudocode, control flow graphs, cyclomatic complexity and paths, as well as the test cases which address each path identified.
6. Section 5 would be most pertinent to the client as it encapsulates the prototype for the entire project.
7. Section 6 would illustrate the project plan which includes the project management plan. These sections would give the client a rough estimation of the project cost, resources and time required before the release of the product.
8. Section 12 consists of the appendix for test case results, in which the readers can cross-reference when reading the decision tables.
9. Section 7 consists of individual team reflection regarding this project.

In general, all of the sections are pertinent to the professor for grading and reviewing the team effort and work done.

1.3 References and Acknowledgments

[1] D. E. Guest, "Perspectives on the study of work-life balance - david E. guest, 2002," Sage Journals, Jun-2002. [Online]. Available:
<https://journals.sagepub.com/doi/10.1177/0539018402041002005>.

[2] I. Lupu and M. Ruiz-Castro, "Work-life balance is a cycle, not an achievement," Harvard Business Review, 29-Jan-2021. [Online]. Available:
<https://hbr.org/2021/01/work-life-balance-is-a-cycle-not-an-achievement>.

[3] R. Knight, "Make sure your team's workload is divided fairly," Harvard Business Review, 03-May-2017. [Online]. Available:
<https://hbr.org/2016/11/make-sure-your-teams-workload-is-divided-fairly>.

[4] H. Inegbedion, E. Inegbedion, A. Peter, and L. Harry, "Perception of workload balance and employee job satisfaction in work organisations," Heliyon, 09-Jan-2020. [Online]. Available:
<https://www.sciencedirect.com/science/article/pii/S2405844020300050>.

[5] B. Roseke and A. B. Roseke, "Learn," ProjectEngineer, 19-Feb-2016. [Online]. Available:
<https://www.projectengineer.net/estimating-task-durations/>.

2 Software Design

Updated class diagram to include missing methods for the system state diagram, refer to [Appendix B](#) for the updated class diagram. The remaining diagrams remain the same as it does not affect the testing involved in M3.

2.1 Updated Sequence Diagram

Reject Job Assigned

[Appendix C](#) shows the sequence diagram for the Reject Job Assigned use case where the staff chooses to reject jobs assigned to them. The system would first display a warning message reminding staff to talk to their manager first before rejecting. After the staff have accepted the message, the system will update the database by changing the job status to 'Rejected' and displays a confirmation message on the StaffDashboardUI. The alternate scenario includes the possibility of staff not being able to reject assigned jobs. Hence, the system will not update job status and return an error message with reasons on the StaffDashboardUI. This updated sequence diagram resolves the issue where the boundary class is not supposed to call a function to the user and display messages directly.

3 Black-box Testing

The following section would illustrate a system state diagram that is clearly informed by the team's use cases in Appendix A as well as the decision tables for the mentioned use cases that comprises the test scenario, test cases, values, pre-conditions, test steps, test data and expected results.

System State Diagram

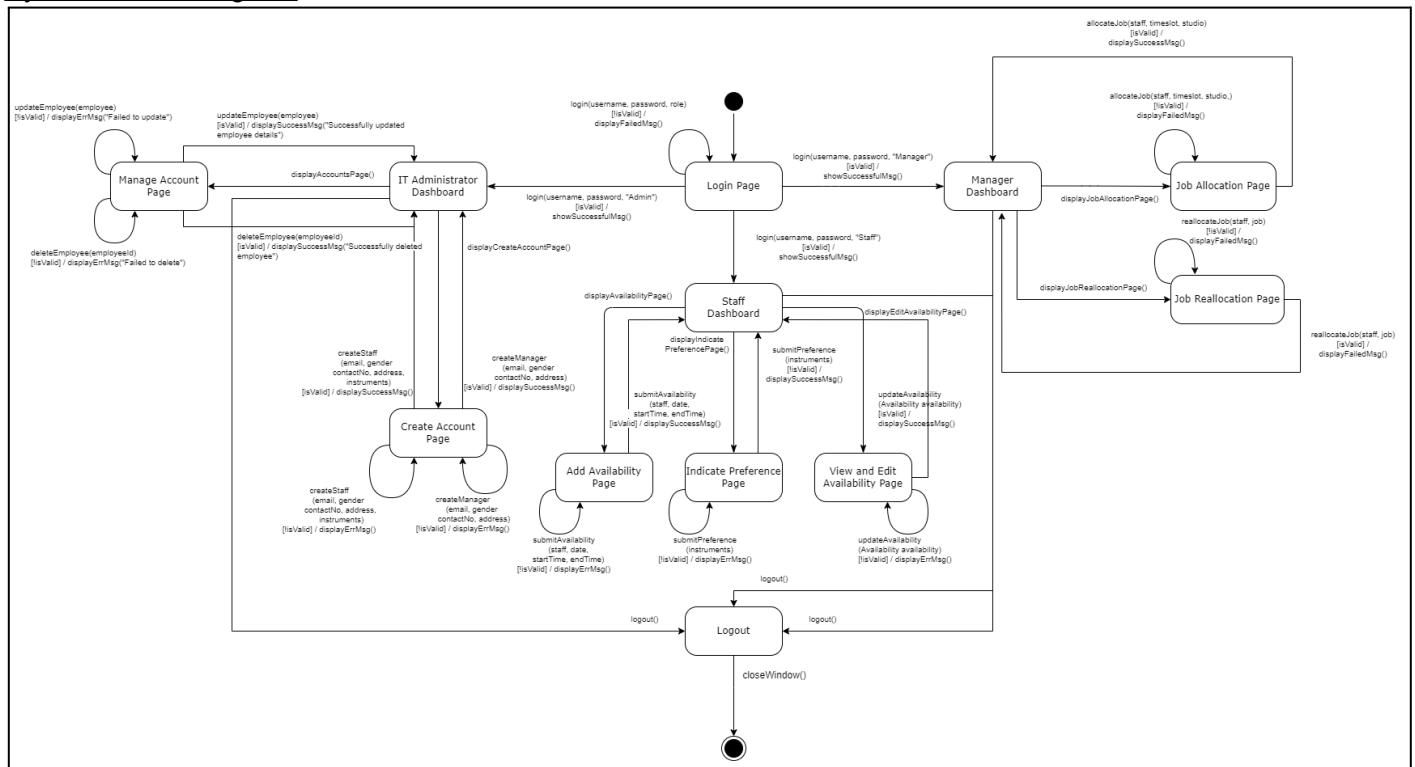


Figure 1: System State Diagram

For the creation of the decision tables, we have selected the use cases that are more complex in nature. These use cases includes:

1. Login Page (Check Login Functionality)
2. Create Account Page (Check Create Account Functionality) for both the manager and staff role
3. Job Allocation Page (Check Job Allocation Functionality)
4. Job Reallocation Page (Check Job Reallocation Functionality)
5. Add Availability Page (Check Add Availability Functionality)

Test Plan & Prototype

The following guidelines were adhered to while constructing the decision table for the mentioned use cases.

1. List all Causes in the decision table
2. Calculate the number of possible combinations
3. Fill columns with all possible combinations (inputs)
4. Add effects to the table (expected outputs)
5. Reduce Test Combinations
6. Check Covered Combination

Login Page (Check Login Functionality)

Login Decision table

Causes		Values	1	2	3	4
C1	Valid Email	Y/N	N	N	Y	Y
C2	Valid Password	Y/N	N	Y	N	Y
Effects						
E1	Accept					X
E2	Reject		X	X	X	

Table 1: Login Decision Table

Reduced Test Combinations

Causes		Values	1-2	3	4
C1	Valid Email	Y/N	N	Y	Y
C2	Valid Password	Y/N	-	N	Y
Effects					
E1	Login Successfully				X
E2	Login Unsuccessful		X	X	

Table 2: Login Reduced Tests Decision Table

<u>Equivalence Partitioning</u>				
	Invalid	Invalid	Invalid	Valid
Email	No Domain Name	-	contains !#\$%^&*()= \?<>,~';{}[]/	Registered domain and matches database
Password	length < 8	not alphanumeric with one uppercase and one lower case	length > 15	length >= 8 and alphanumeric with one uppercase and one lowercase

Table 3: Login Equivalent Partitioning Table

<u>Boundary Value Analysis</u>			
	Invalid (min -1)	Valid (min, +min, -max, max)	Invalid (max + 1)
Password	7	8, 9, 14, 15	16

Boundary Value Analysis for password field:	Test Cases
<ol style="list-style-type: none"> 1. Minimum boundary value is 8 2. Maximum boundary value is 15 3. Valid text length is 8, 9, 14, 15 4. Invalid text length is 7, 16 	<ul style="list-style-type: none"> • Test case 1: Text length of 7 (min-1) = Invalid • Test case 2: Text length of exactly 8 (min) = Valid • Test case 3: Text length of 9 (min+1) = Valid • Test case 4: Text length of 14 (max-1) = Valid • Test case 5: Text length of exactly 15 (max) = Valid • Test case 6: Text length of 16 (max+1) = Invalid

Table 4: Login Boundary Value Analysis Table

Test Plan & Prototype

Test Scenario	Test Case No.	Test Case	Pre-conditions	Test Steps	Test Data	Expected Results	Actual Result	Pass (P) / Fail (F)
Check Login Functionality	1-2	Check response on entering invalid email	Navigate to web application to login Pre-existing user Database: Email: bob@gmail.com Password: admin123	1. Enter invalid email 2. Enter Invalid/ Valid Password 3. Click Submit button	Email: test Password: admin123	Login must be unsuccessful	Login Failed	P
	3	Check response on entering invalid password		1. Enter valid Username 2. Enter invalid Password 3. Click Submit button	Email: bob@gmail.com Password: wrong@123			
	4	Check response on entering valid email and valid password		1. Enter valid Username 2. Enter valid Password 3. Click Submit button	Email: manager@gmail.com Password: admin123	Login must be successful	See Appendix E , Figure 20, 21, and 22	P

Table 5: Login Test Cases Table

Create Account Page (Check Create Account Functionality)

Create Staff Account Decision Table

Causes		Values	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	
C1	Valid Staff ID	Y/N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
C2	Valid Staff Name	Y/N	N	N	N	N	N	N	N	N	Y	Y	Y	Y	Y	Y	Y	Y	N	N	N	N	N	N	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y
C3	Valid Staff Email	Y/N	N	N	N	Y	Y	Y	Y	N	N	N	N	Y	Y	Y	Y	N	N	N	N	Y	Y	Y	Y	N	N	N	N	Y	Y	Y	Y		
C4	Valid Staff Contact No	Y/N	N	N	Y	N	Y	Y	N	Y	N	N	Y	N	Y	Y	N	Y	N	N	Y	N	Y	Y	N	Y	Y	N	N	Y	Y	N	Y		
C5	Instrument Qualification	Y/N	N	Y	N	N	Y	N	Y	Y	N	Y	N	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y	Y	N	Y	N	Y	N	Y			
Effects																																			
E1	Account Created Successfully																																x		
E2	Account Creation Failed		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x		

Table 6: Create Staff Account Decision Table

Reduced Test Combinations

Causes		Values	1-16	17-24	25-28	29-30	31	32
C1	Valid Staff ID	Y/N	N	Y	Y	Y	Y	Y
C2	Valid Staff Name	Y/N	-	N	Y	Y	Y	Y
C2	Valid Staff Email	Y/N	-	-	N	Y	Y	Y
C3	Valid Staff Contact No	Y/N	-	-	-	N	Y	Y
C4	Instrument Qualification	Y/N	-	-	-	N	N	Y
Effects								
E1	Account Created Successfully							x
E2	Account Creation Failed		x	x	x	x	x	

Table 7: Create Staff Account Reduced Tests Decision Table

Create Manager Account Decision Table

Causes		Values	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
C1	Valid Manager ID	Y/N	N	N	N	N	N	N	N	N	Y	Y	Y	Y	Y	Y	Y	Y
C2	Valid Manager Name	Y/N	N	N	N	Y	Y	Y	Y	Y	N	N	N	N	Y	Y	Y	Y
C3	Valid Manager Email	Y/N	N	N	Y	N	N	N	Y	Y	N	N	Y	N	N	Y	Y	Y
C4	Valid Manager contact No	Y/N	N	Y	N	N	N	Y	N	Y	N	Y	N	N	Y	N	Y	Y
Effects																		
E1	Account Created Successfully																	x
E2	Account Creation Failed		x	x	x	x	x	x	x	x	x	x	x	x	x	x		

Table 8: Create Manager Decision Table

Reduced Test Combinations

Causes		Values	33-40	41-44	45-46	47	48
C1	Valid Manager ID	Y/N	N	Y	Y	Y	Y
C2	Valid Manager Name	Y/N	-	N	Y	Y	Y
C3	Valid Manager Email	Y/N	-	-	N	Y	Y
C4	Valid Manager contact no	Y/N	-	-	-	N	Y
Effects							
E1	Account Created						x

Test Plan & Prototype

	Successfully					
E2	Account Creation Failed		x	x	x	x

Table 9: Create Manager Reduced Tests Decision Table

<u>Equivalence Partitioning</u>				
	Invalid	Invalid	Valid	Invalid
Name	Digits	length < 3	a-z	!@#\$%^&*()_+=\ ?<>.`~-';":{}[]/
ID	a-z	-	Digits	!@#\$%^&*()_+=\ ?<>,.`~-';":{}[]/
Email	No domain name	-	Registered domain	!#\$\$%^&*()= \ ?<>.`~-';":{}[]/
Contact number	Digits > 8	a-z	Digits = 8	Digits < 8

Table 10: Create Manager Equivalence Partitioning Table

<u>Boundary Value Analysis</u>			
	Invalid (min -1)	Valid	Invalid (max + 1)
Contact number	7	8	9

Boundary Value Analysis for password field:	Test Cases
<ol style="list-style-type: none"> 1. Minimum boundary value is 8 2. Maximum boundary value is 8 3. Valid text length is 8 4. Invalid text length is <8 and >8 	<ul style="list-style-type: none"> • Test case 1: Digit length of 7 (min-1) = Invalid • Test case 2: Digit length of exactly 8 (min) = Valid • Test case 3: Digit length of 9 (min+1) = Invalid

Table 11: Create Manager Boundary Value Analysis Table

Test Plan & Prototype

Test scenario	Test Case No.	Test Case	Pre-conditions	Test Steps	Test Data	Expected Results	Actual Result	Pass (P) / Fail (F)
Check Create Account Functionality	1-16	Check response on invalid staff ID for creating staff account	The user must be in the Admin Role. The user must be logged in.	1. Selects "Staff" for account type 2. Enter an invalid staff ID 3. Enter valid staff name 4. Enter valid staff email 5. Enter valid staff contact number 6. Select instrument 7. Submit Information	Role: Staff Staff ID: -1 Staff Name: Test Staff Email: test@gmail.com Staff Contact: 92345678 Instrument: Piano	System displays message "Invalid Staff ID"	System displayed "Invalid Staff ID"	P
	17-24	Check response on invalid name for creating staff account		1. Selects "Staff" for account type 2. Enter an valid staff ID 3. Enter invalid staff name 4. Enter valid staff email 5. Enter valid staff contact number 6. Select instrument 7. Submit Information	Role: Staff Staff ID: 1 Staff Name: a Staff Email: test@gmail.com Staff Contact: 92345678 Instrument: Piano	System displays message "Invalid Staff Name"	System displayed "Invalid Staff Name"	P

Test Plan & Prototype

	25-28	Check response on invalid email for creating staff account		1. Selects "Staff" for account type 2. Enter an valid staff ID 3. Enter valid staff name 4. Enter invalid staff email 5. Enter valid staff contact number 6. Select instrument 7. Submit Information	Role: Staff Staff ID: 1 Staff Name: Tester Test Staff Email: test Staff Contact: 92345678 Instrument: Piano	System display message "Invalid email" System displayed "Invalid Staff Email"	P
	29-30	Check response on invalid contact number for creating staff account		1. Selects "Staff" for account type 2. Enter an valid staff ID 3. Enter valid staff name 4. Enter invalid staff email 5. Enter invalid staff contact number 6. Select instrument 7. Submit Information	Role: Staff Staff ID: 1 Staff Name: Tester Test Staff Email: test@gmail.com Staff Contact No: 923424 Instrument: Piano	System display message "Invalid Contact Number" System displayed "Invalid Contact Number"	P
	31	Check response on missing instrument for creating staff		1. Selects "Staff" for account type	Role: Staff Staff ID: 1	System display message "Select an instrument" System displayed "Select an instrument"	P

Test Plan & Prototype

		account		2. Enter an valid staff ID 3. Enter valid staff name 4. Enter invalid staff email 5. Enter invalid staff contact number 6. Select instrument 7. Submit Information	Staff Name: Tester Test Staff Email: test@gmail.com Staff Contact No: 92342324 Instrument:		
	32	Check response on entering all valid fields for creating staff account		1. Selects "Staff" for account type 2. Enter an valid staff ID 3. Enter valid staff name 4. Enter valid staff email 5. Enter valid staff contact number 6. Select instrument 7. Submit Information	Role: Staff Staff ID: 1 Staff Name: Tester Test Staff Email: test@gmail.com Staff Contact No: 92342324 Instrument: Piano	System display message "Account Created Successfully" See Appendix E , Figure 23	P
	33-40	Check response on entering invalid manager ID for creating manager account		1. Selects "Manager" for account type 2. Enter an invalid manager ID 3. Enter valid manager name	Role: Manager Manager ID: -1 Manager Name: Tester Test	System displays message "Invalid Manager ID" System displayed "Invalid Manager ID"	P

Test Plan & Prototype

				4. Enter valid manager email 5. Enter valid manager contact number 6. Submit information	Manager Email: test@gmail.com Manager Contact No: 92342324			
41-44	Check response on invalid name for creating manager account			1. Selects “Manager” for account type 2. Enter a valid manager ID 3. Enter invalid manager name 4. Enter valid manager email 5. Enter valid manager contact number 6. Submit information	Role: Manager Manager ID: 1 Manager Name: c Manager Email: test@gmail.com Manager Contact No: 92342324	System displays message “Invalid Manager Name”	System displayed “Invalid Manager Name”	P
45-46	Check response on invalid email for creating manager account			1. Selects “Manager” for account type 2. Enter a valid manager ID 3. Enter valid manager name 4. Enter invalid manager email 5. Enter valid manager contact number	Role: Manager Manager ID: 1 Manager Name: Test Manager Manager Email: testmanager Manager Contact No: 92342324	System display message “Invalid email”	System displayed “Invalid Email”	P

Test Plan & Prototype

			6. Submit information				
47	Check response on invalid contact number for creating manager account		1. Selects “Manager” for account type 2. Enter a valid manager ID 3. Enter valid manager name 4. Enter valid manager email 5. Enter invalid manager contact number 6. Submit information	Role: Manager Manager ID: 1 Manager Name: Test Manager Manager Email: testmanager@g mail.com Manager Contact No: 9234232	System display message “Invalid Contact Number”	System displayed “Invalid Contact Number”	P
48	Check response on entering all valid fields for creating manager account		1. Selects “Manager” for account type 2. Enter a valid manager ID 3. Enter valid manager name 4. Enter valid manager email 5. Enter valid manager contact number 6. Submit information	Role: Manager Manager ID: 1 Manager Name: Test Manager Manager Email: testmanager@g mail.com Manager Contact No: 92342321	System display message “Account created Successfully”	See Appendix E , Figure 23	P

Table 12: Create Manager Test Cases Table

Test Plan & Prototype

Job Allocation Page (Check Job Allocation Functionality)

Allocate Job Decision table

Causes		Values	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	
C1	Valid Date and Time Slot	Y/N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	
C2	Available Staff	Y/N	N	N	N	N	N	N	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	N	N	N	N	N	N	N	N	Y	Y	Y	Y	Y	Y	
C3	Available Studio	Y/N	N	N	N	Y	Y	Y	Y	N	N	N	Y	Y	Y	Y	Y	Y	N	N	N	N	Y	Y	Y	Y	N	N	N	Y	Y	Y	Y		
C4	Valid Studio	Y/N	N	N	Y	N	Y	Y	N	Y	N	N	Y	N	Y	Y	N	Y	N	N	N	Y	Y	Y	N	Y	N	N	Y	Y	N	N	Y	Y	
C5	Valid Staff	Y/N	N	Y	N	N	Y	N	Y	Y	N	Y	N	N	Y	N	Y	Y	N	Y	N	N	Y	N	Y	Y	N	Y	N	Y	N	Y	N	Y	
Effects																																			
E1	Job Allocated Successfully																																x		
E2	Job Allocation Failed		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Table 13: Allocate Job Decision Table

Reduced Test Combinations

Causes		Values	1-16	17-24	25-28	29-30	31	32
C1	Valid Date and Time Slot	Y/N	N	Y	Y	Y	Y	Y
C2	Available Staff	Y/N	-	N	Y	Y	Y	Y
C3	Available Studio	Y/N	-	-	N	Y	Y	Y

Test Plan & Prototype

C4	Valid Studio	Y/N	-	-	-	N	Y	Y
C5	Valid Staff	Y/N	-	-	-	-	N	Y
Effects								
E1	Job Allocated Successfully							X
E2	Job Allocation Failed		X	X	X	X	X	

Table 14: Allocate Job Reduced Tests Decision Table

<u>Boundary Value Analysis</u>		
	Valid (min, +min, -max, max)	Invalid (max + 1)
Valid Date & Time slot	<= 1 week from current date at a time	> 1 weeks from current date at a time
Available Staff	<= 72 hours a week	> 72 hours a week

Boundary Value Analysis for available staff:	Test cases
<ol style="list-style-type: none"> 1. Minimum boundary value is 72 hours including weekends. 2. Maximum boundary value is 72 hours including weekends. 3. Valid hours are between 1 to 72 hours. 4. Invalid hours are more than 72 hours a week. 	<p>[Numbers in weeks]</p> <ul style="list-style-type: none"> • Test case 1: Available Staff with 0 hours (min-1) = Invalid • Test case 2: Available Staff with exactly 1 hour (min) = Valid • Test case 3: Available Staff with 2 hours (min+1) = Valid • Test case 4: Available Staff with 71 hours (max-1) = Valid • Test case 5: Available Staff with exactly 72 hour (max) = Valid • Test case 6: Available Staff with 73 hours (max+1) = Invalid

Table 15: Allocate Job Boundary Value Analysis Table

Test Plan & Prototype

Test Scenario	Test Case No.	Test Case	Pre-conditions	Test Steps	Test Data	Expected Results	Actual Result	Pass (P) / Fail (F)
Check Job Allocation Functionality	1-16	Check response on submitting invalid date and invalid timeslot	Users must be in the Manager Dashboard. Users must be logged in.	1. Select type of lesson 2. Submit Information	Lesson: Piano	System display message "Please select a date and time slot"	System displayed message "Please select a date and time slot"	P
	17-24	Check response when there is no available staff		1. Select type of lesson 2. Select date and time slot	Staff Available: 0 Studio Available: 1 Lesson: Piano Date: 25/11/2022 Time: 0900-0930	System display message "No available staff"	System displayed message "No available staff"	P
	25-28	Check response when there is no available studio		1. Select type of lesson 2. Select date and time slot	Staff Available: 1 Studio Available: 0 Lesson: Piano Date: 25/11/2022 Time: 0900-0930	System display message "No available studio"	System displayed message "No available studio"	P

Test Plan & Prototype

	29-30	Check response on submitting invalid staff for allocate job		<ol style="list-style-type: none"> 1. Select type of lesson 2. Select date 3. Select time slot 4. Submit Information 	Staff Available: 1 Studio Available: 1 Lesson: Piano Date: 25/11/2022 Time: 0900-0930	System display message "Please select a staff"	System displayed message "Please select a staff"	P
	31	Check response on submitting invalid studio for allocate job		<ol style="list-style-type: none"> 1. Select type of lesson 2. Select date 3. Select time slot 4. Select staff 5. Submit Information 	Staff Available: 1 Studio Available: 1 Lesson: Piano Date: 25/11/2022 Time: 0900-0930 Staff: Bob	System display message "Please select a studio"	System displayed message "Please select a studio"	P
	32	Check response for entering all valid fields for job allocation		<ol style="list-style-type: none"> 1. Select type of lesson 2. Select date 3. Select time slot 4. Select staff 5. Select studio 6. Submit Information 	Staff Available: 1 Studio Available: 1 Lesson: Piano Date: 25/11/2022 Time: 0900-0930 Staff: Bob Studio: 1	System display message "Job allocated Successfully"	See Appendix E , Figure 24	P

Table 16: Allocate Job Test Cases Table

Job Reallocation Page (Check Job Reallocation Functionality)

Reallocate Job Decision table

Causes	Values	1	2	3	4
C1	Available Staff	Y/N	N	N	Y
C2	Valid Staff	Y/N	N	Y	N
Effects					
E1	Job Allocated Successfully				x
E2	Job Allocation Failed	x	x	x	

Table 17: Reallocate Job Decision Table

Reduced Test Combinations

Causes	Values	1-2	3	4
C1	Y/N	N	Y	Y
C2	Y/N	-	N	Y
Effects				
E1	Job reallocated Successfully			x
E2	Job reallocation Failed	x	x	

Table 18: Reallocate Job Reduce Tests Decision Table

Test Scenario	Test Case No.	Test Case	Pre-conditions	Test Steps	Test Data	Expected Results	Actual Result	Pass (P) / Fail (F)
Check Job Reallocation Functionality	1-2	Check response when there is no available staff		1. Select job to reallocate	Available Staff: 0	System display message "No available staff"	System displayed message "No available staff"	P
	3	Check response on submitting invalid staff		1. Select job to reallocate 2. Submit information	Staff Available: 1 Staff:	System display message "Please select a staff to reallocate the job"	System displayed message "Please select a staff to reallocate the job"	P
	4	Check response for entering all valid fields for job reallocation		1. Select job to reallocate 2. Select valid staff 3. Submit information	Staff Available: 1 Staff: Bob	System display message "Job reallocated successfully"	See Appendix E , Figure 25	P

Table 19: Reallocate Job Test Cases Table

Add Availability Page (Check Add Availability Functionality)

Add Availability Decision table

Causes		Values	1	2	3	4
C1	Valid Date	Y/N	N	N	Y	Y
C2	Valid Time Slots	Y/N	N	Y	N	Y
Effects						
E1	Added Availability Successfully					X
E2	Failed to add availability		X	X	X	

Table 20: Add Availability Decision Table

Reduced Test Combinations

Causes		Values	1-2	3	4
C1	Valid Date	Y/N	N	Y	Y
C2	Valid Time slots	Y/N	-	N	Y
Effects					
E1	Added Availability Successfully				X
E2	Failed to add availability		X	X	

Table 21: Add Availability Reduce Tests Decision Table

<u>Boundary Value Analysis</u>			
	Invalid (min -1)	Valid (min, +min, -max, max)	Invalid (max + 1)
Valid Date	<1 weeks from current date	<= 5 weeks from current date	> 5 weeks from current date
Available Time Slot	Weekdays (before 9am and after 9pm) Weekends (before 8am and after 9pm)	Weekdays (between 9am to 9pm) Weekends (between 8am to 9pm)	0

Boundary Value Analysis for date field:	Test cases
<ol style="list-style-type: none"> 1. Minimum boundary value is 1 week of availability for date. 2. Maximum boundary value is 5 weeks of availability for date. 3. Valid date is 1-5 weeks of availability for date. 4. Invalid date is less than 1 week availability, more than 5 weeks of availability for date. 	<p>[Numbers in weeks]</p> <ul style="list-style-type: none"> • Test case 1: Valid date of 0 (min-1) = Invalid • Test case 2: Valid date of exactly 1(min) = Valid • Test case 3: Valid date of 2 (min+1) = Valid • Test case 4: Valid date of 4 (max-1) = Valid • Test case 5: Valid date of exactly 5 (max) = Valid • Test case 6: Valid date of 6 (max+1) = Invalid

Table 22: Add Availability Boundary Value Analysis Table

Test Plan & Prototype

Test scenario	Test Case No.	Test Case	Pre-conditions	Test Steps	Test Data	Expected Results	Actual Result	Pass (P) / Fail (F)
Check Add Availability Functionality	1-2	Check response on submitting without date		1. Submit Information	Date: Timeslots:	System display message "Please select a date"	System displayed message "Please select a date"	P
	3	Check response on submitting without timeslots		1. Select Date 2. Submit Information	Date: 25/11/2022 Timeslots:	System display message "Please select a timeslot"	System displayed message "Please select a timeslot"	P
	4	Check response for entering all valid fields for job reallocation		1. Select Date 2. Select time slots 3. Submit Information	Date: 25/11/2022 Timeslots: 0900-0930 0930-1000	System display message "Availability Added successfully"	See Appendix E , Figure 26	P

Table 23: Add Availability Test Cases Table

4 Whitebox Testing

For white-box testing, the allocateJob function was selected as it is the most complex use case of the whole system. Ensuring that this function is extensively tested is paramount to achieving the system's intended outcome of allowing managers to do job management.

```
1  function allocateJob(staff: Staff, studio: Studio, instrument: Instrument, date: Date, startTime: Time, endTime: Time) {
2      if (staff.instrument != instrument)
3          return {success: false, msg: "Staff not qualified to teach this instrument!"};
4      if (!studio.getInstruments().includes(instrument))
5          return {success: false, msg: "Studio does not support this instrument type!"};
6      var currentAvailabilities = staff.getAvailabilityByDate(date);
7      var available = false;
8      for (var i = 0; i < currentAvailabilities.length; i++) {
9          if (currentAvailabilities[i].getStartTime() <= startTime && currentAvailabilities[i].getEndTime() >= endTime) {
10              available = true;
11              break;
12          }
13      }
14      if (!available)
15          return {success: false, msg: "Staff not available to be allocated this job!"};
16      var currentJobs = staff.getJobsByDate(date);
17      var concurrent_counter = 1;
18      var first_job_startTime;
19      var previous_job_endTime;
20      for (var i = 0; i < currentJobs.length; i++) {
21          if (currentJobs[i].getStartTime() == startTime)
22              return {success: false, msg: "Staff already has an allocated job at this time!"};
23          if (previous_job_endTime == currentJobs[i].getStartTime()) {
24              concurrent_counter++;
25          } else {
26              concurrent_counter = 1;
27              first_job_startTime = currentJobs[i].getStartTime;
28          }
29          previous_job_endTime = currentJobs[i].getEndTime();
30          if (concurrent_counter == 8 && currentJobs[i].endTime.increaseMins(60) > startTime || concurrent_counter == 8 && first_job_startTime.decreaseMins(60) < endTime) {
31              return {success: false, msg: "Staff is already assigned 4 consecutive hours without break!"};
32          }
33      }
34      const newJob = new Job(staff, instrument, date, studio, startTime, endTime);
35      JobModel.insertOne(newJob);
36      return {success: true, msg: "Job successfully allocated"};
37  }
```

Figure 2: allocateJob code

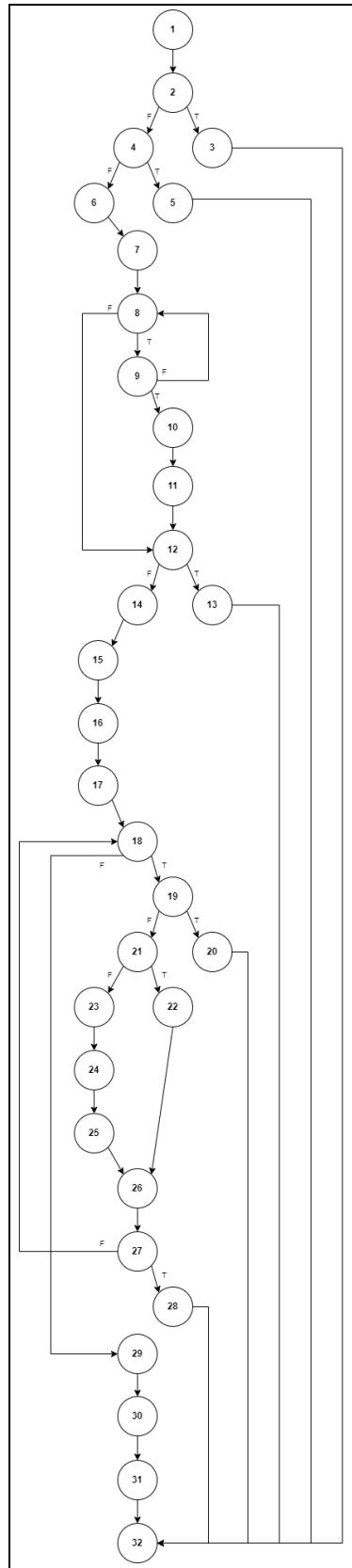


Figure 3: Control Flow Graph of `allocateJob`

Test Plan & Prototype

Cyclomatic Complexity

$$= E - N + 2P$$

$$= 40 - 32 + 2(1)$$

$$= 10$$

Determine a set of basis paths

Steps

1. Pick a “baseline” path that corresponds to normal execution (aim to have as many decisions as possible)
2. To get succeeding basis paths, back track the baseline until you reach a decision node. “Flip” the decision (take another alternative) and continue as much of the baseline as possible
3. Repeat this until all decisions have been flipped. When you reach $V(G)$ basis paths, you’re done!

Path 1: 1-2-3-32

Path 2: 1-2-4-5-32

Path 3: 1-2-4-6-7-8-9-10-11-12-14-15-16-17-18-19-20-32

Path 4: 1-2-4-6-7-8-9-10-11-12-14-15-16-17-18-19-21-23-24-25-26-27-18-29-30-31-32

Path 5:

1-2-4-6-7-8-9-10-11-12-14-15-16-17-18-19-21-23-24-25-26-27-18-19-21-22-26-27-18-29-30-31-32

Path 6:

1-2-4-6-7-8-9-10-11-12-14-15-16-17-18-19-21-23-24-25-26-27-18-19-21-22-26-27-18-19-21-22-26-27-18-19-21-22-26-27-18-19-21-22-26-27-18-19-21-22-26-27-18-19-21-22-26-27-28-32

Path 7:

1-2-4-6-7-8-9-10-11-12-14-15-16-17-18-19-21-23-24-25-26-27-18-19-21-22-26-27-18-19-21-22-26-27-18-19-21-22-26-27-18-19-21-22-26-27-18-19-21-22-26-27-18-19-21-22-26-27-18-29-30-31-32

Path 8: 1-2-4-6-7-8-9-10-11-12-14-15-16-17-18-29-30-31-32

Path 9: 1-2-4-6-7-8-9-8-12-13-32

Path 10: 1-2-4-6-7-8-12-13-32

Test Cases

Path	Instrument qualified	Instrument for job	Instrument in studio	Availability	New job time period	Allocated jobs time periods	Result
1	Drums	Piano	Drums, Piano, Violin, Trumpet	0900-1300	1100-1130	1200-1230	F, "Staff not qualified to teach this instrument"
2	Drums	Drums	Piano, Violin, Trumpet	0900-1300	1100-1130	1200-1230	F, "Studio does not support this instrument type!"
3	Drums	Drums	Drums, Piano, Violin, Trumpet	0900-1300	1100-1130	1100-1130	F, "Staff already has an allocated job at this time!"
4	Piano	Piano	Drums, Piano, Violin, Trumpet	0900-1500	1400-1430	1330-1400	T, "Job successfully allocated"
5	Drums	Drums	Drums, Piano, Violin, Trumpet	0900-1500	1300-1330	0900-0930 0930-1000	T, "Job successfully allocated"
6	Drums	Drums	Drums, Piano, Violin, Trumpet	0900-1500	1300-1330	0900-0930 0930-1000 1000-1030 1030-1100 1100-1130 1130-1200 1200-1230 1230-1300	F, "Staff is already assigned 4 consecutive hours without break"
7	Drums	Drums	Drums, Piano, Violin, Trumpet	0900-1500	1400-1500	0900-0930 0930-1000 1000-1030 1030-1100	T, "Job successfully allocated"

Test Plan & Prototype

						1100-1130 1130-1200 1200-1230 1230-1300	
8	Piano	Piano	Drums, Piano, Violin, Trumpet	0900-1500	1000-1030		T, "Job successfully allocated"
9	Piano	Piano	Drums, Piano, Violin, Trumpet	0900-1500	1500-1530		F, "Staff not available to be allocated this job!"
10	Piano	Piano	Drums, Piano, Violin, Trumpet		1500-1530		F, "Staff not available to be allocated this job!"

Table 24: allocateJob Test Cases Table

5 Project Prototype & Wireframes

The prototype has a readme file located at the Github repository, which clearly explains how to access and navigate it. The video link to the short demonstration of the prototype is as follows: https://youtu.be/3muN_WrOLic

The following prototype is built based on the requirements of the Music School Project. The concepts, flow, and user interaction of the application will be detailed below accompanied with screenshots of the prototype.

5.1 Login and Forgot Password Page

Login Page

Upon visiting the Music School website, the users (Manager, Staff and IT Admin will see a login page, as shown in Figure 4. The different users will see different features and functionality of the website according to their role in the company. For this prototype, we will be able to access the respective user accounts, shown in the Table below, based on the input we entered in the Email field and the tabs that the user has selected.

Email	Account
manager@musicschool.com	Manager
staff@musicschool.com	Staff
admin@musicschool.com	IT Administrator

Table 25: Email and Account

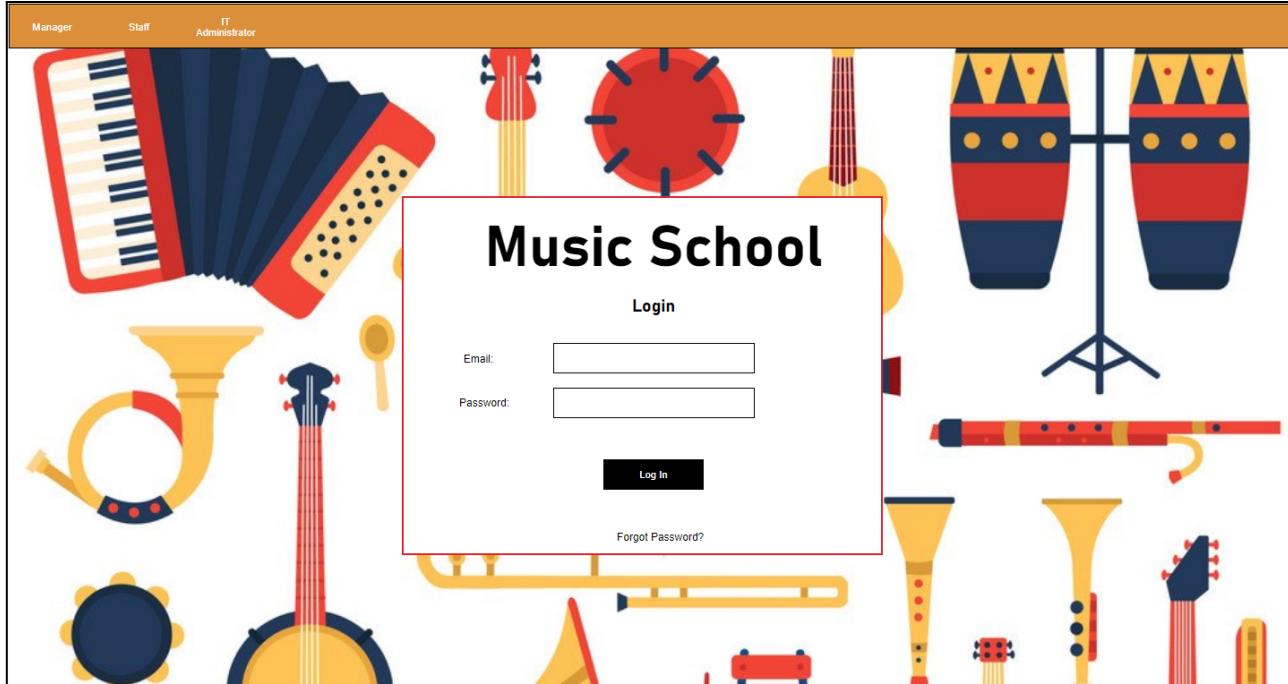


Figure 4: Login Page

Forget Password Page

The Forget Password Page, as shown in Figure 5, is displayed if the user clicked on the “Forget Password?” button shown in the above figure. The users will be prompted to enter their email and mobile phone number to recover their password. Following which, they will be redirected back to the Login Page once they are clicked on the “Done” button.

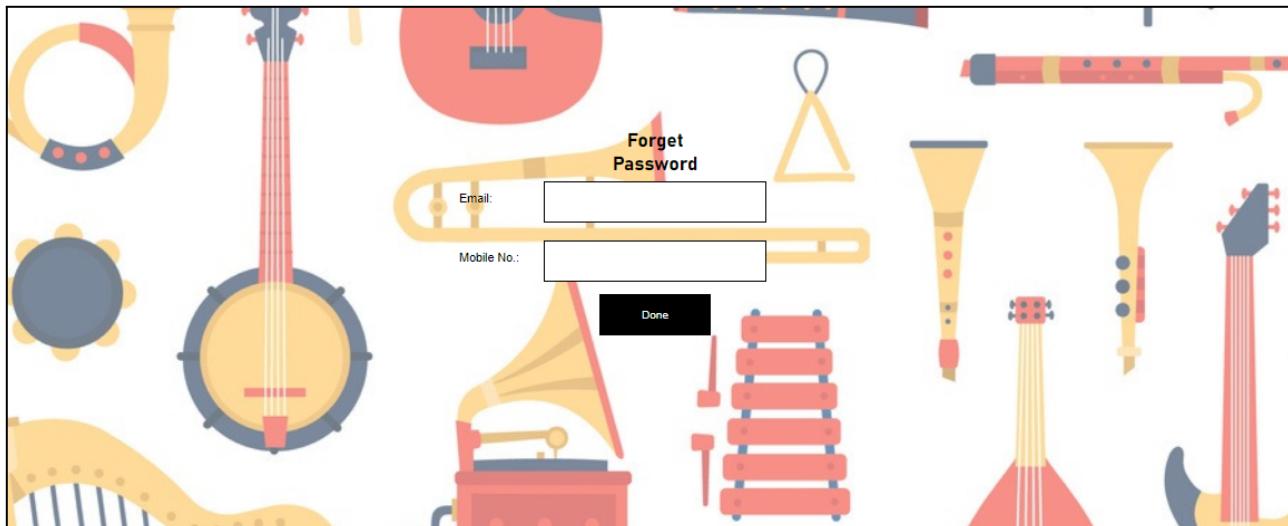


Figure 5: Forget Password Page

5.2 Manager Pages

The following wireframes will demonstrate the use cases UC1-4: View Staff Workload, Staff Availability, Allocate Job and Reallocate Job.

Dashboard Page

The Manager Dashboard Page, as shown in Figure 6. is displayed after the manager has logged in. The manager will be able to visualise the staff workload, top three staff with the lowest workload and all staff over 40 hours of jobs allocated are highlighted in red. Additionally, the overall instrument workload is known to the manager so that better arrangement could be made to allocate staff accordingly.

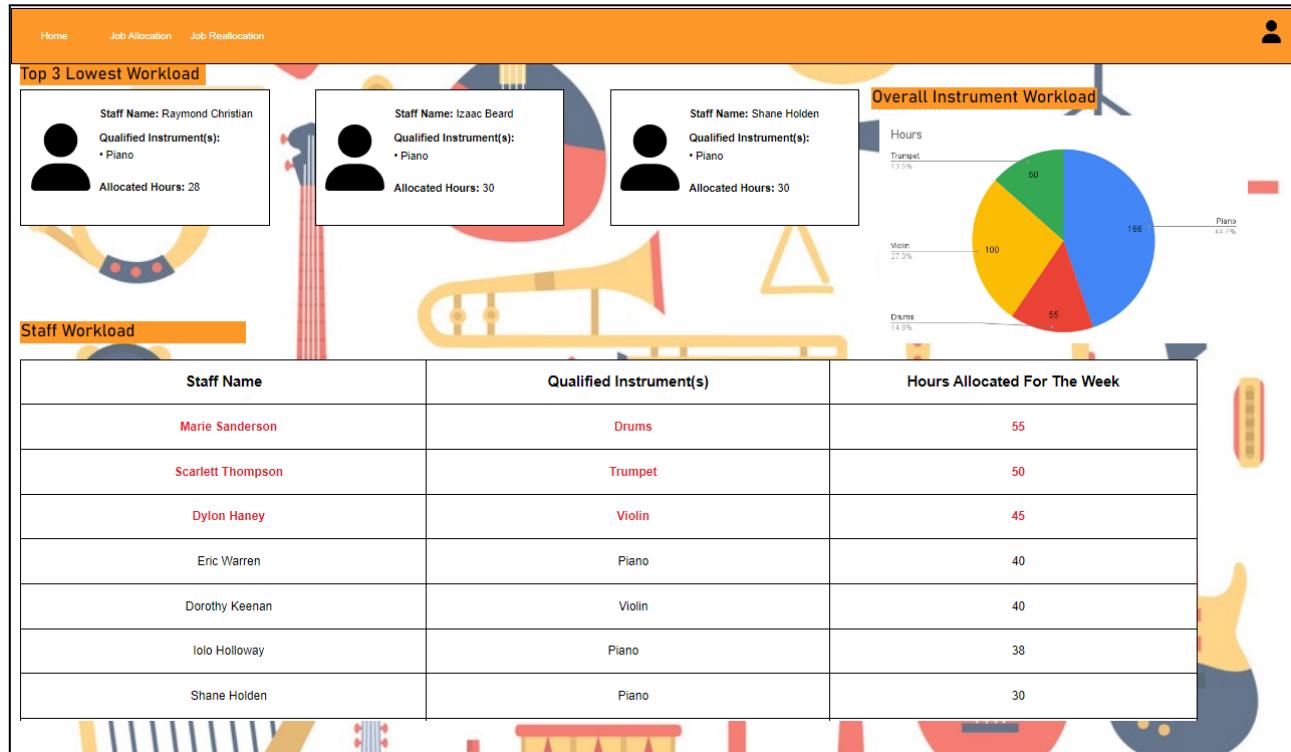


Figure 6: Manager Dashboard Page

Job Allocation Page

The Manager Job Allocation Page, as shown in Figure 7, is displayed after the manager chooses to allocate jobs. The manager will be able to allocate jobs to staff for one week at a time, view up to three staff availability and any relevant information. When displaying the staff availability, the workload assigned, staff's job preference, staff's location at a particular date, and availability for the week are shown to the manager.

After viewing the staff availability, the manager can select the type of lesson, date and time slots. The system will then fetch available studio and staff and a dropdown list of studio and staff will be shown to the manager. After the manager selects the studio and the staff, they will be able to submit the information. A green confirmation message "Job Allocated Successfully" will be displayed after the system added the job into the database.

Figure 7: Manager Job Allocation Page

Job Reallocation Page

The Manager Job Reallocation Page, as shown in Figure 8, is displayed after the manager chooses to reallocate the rejected job(s). The manager will be able to select which job to reallocate. The manager will be able to view staff availability and any relevant information. When displaying the staff availability, the workload assigned, staff's job preference, staff's location at a particular date, and availability for the week are shown to the manager.

After viewing the staff availability, the manager can allocate the job based on staff availability. After the manager submits the information, a green confirmation message “Job Reallocated Successfully” will be displayed when the system updates the job in the database.

Figure 8: Manager Job Reallocation Page

5.3 Staff Pages

The following wireframes will demonstrate the use cases UC5-10: View Overall Workload, Reject Job Assigned, Add Availability, Indicate Job Preference, Edit Availability and View Availability.

Dashboard Page

The Staff Dashboard Page, as shown in Figure 9, is displayed after the staff has logged in. The staff will be able to view their weekly job assignments and overall workload for the month. The staff can reject jobs assigned to them, but they will be warned to discuss the jobs with their manager before proceeding with the rejection, as shown in Figure 10. In the dashboard, the staff is able to view overall workload, reject job assigned and view availability,

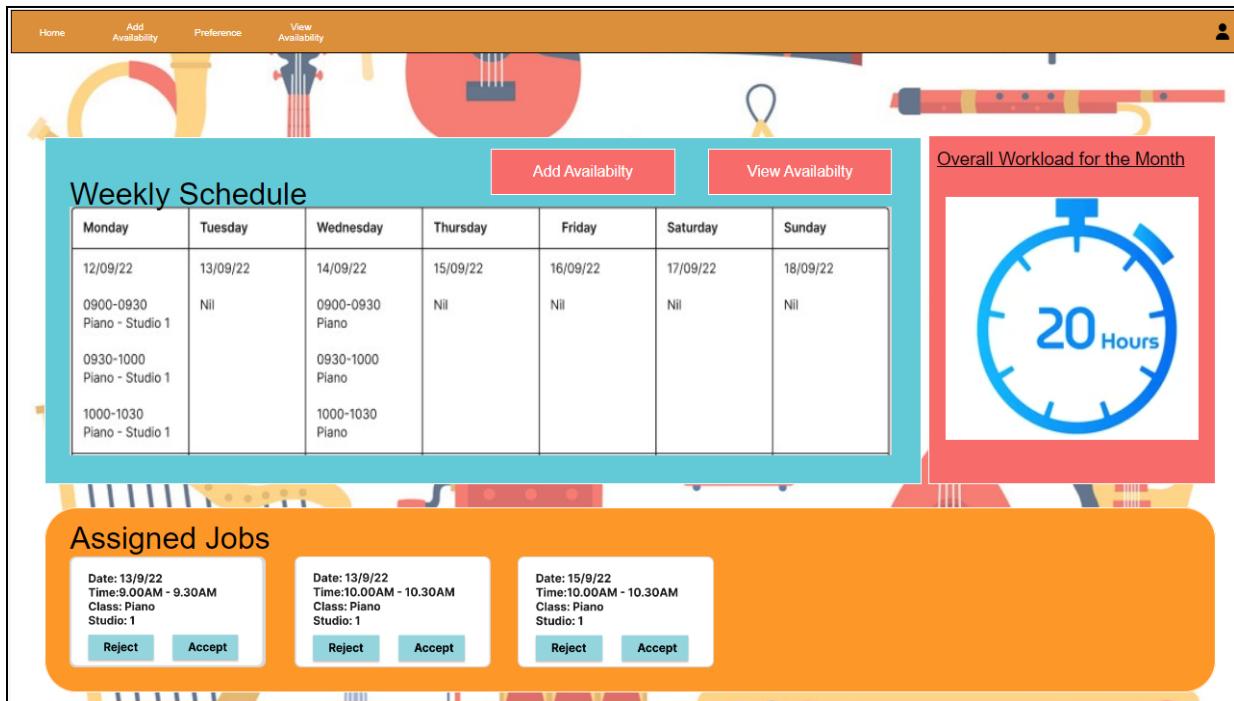


Figure 9: Staff Dashboard Page

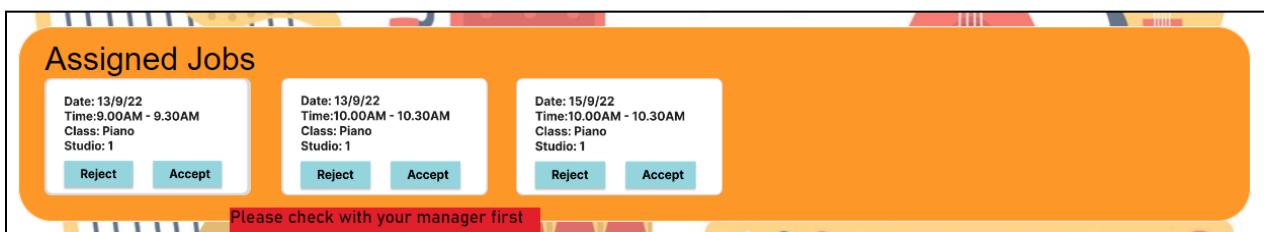


Figure 10: Staff Dashboard Page Warning Message

Add Availability Page

The Staff Add Availability Page, as shown in Figure 11, is displayed after the staff chooses to add their availability. The staff will be able to select the date and time slots in 30 mins intervals up to 5 weeks ahead of time. After submitting their availability, a green confirmation message “Availability Added Successfully” will be displayed when the system added their availability into the database. In this page, the staff is able to add availability into the system.

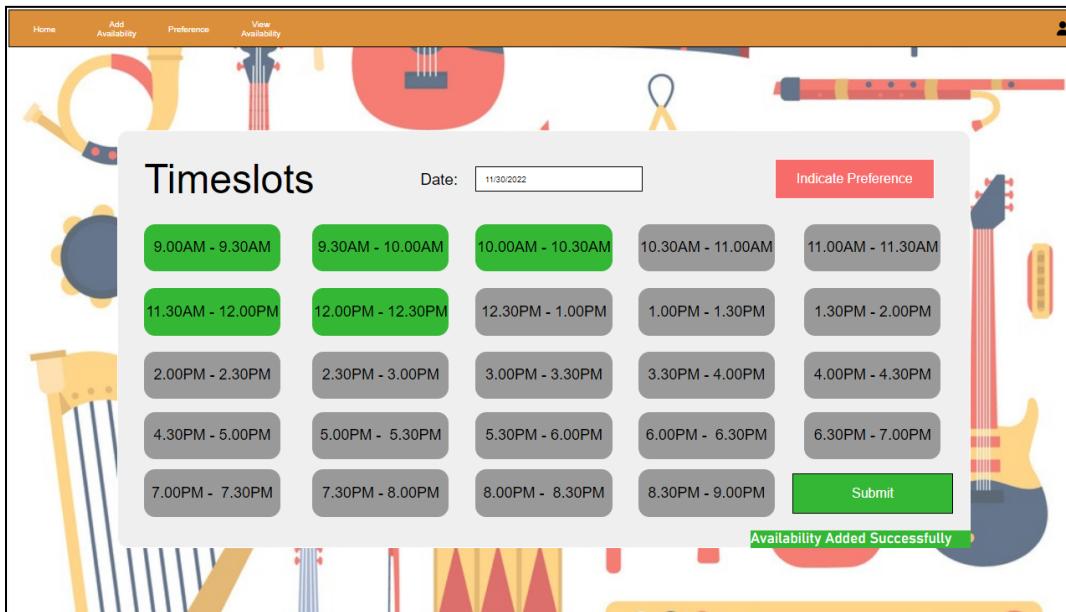


Figure 11: Staff Add Availability Page

Indicate Job Preference Page

The Staff Indicate Job Preference Page as shown in Figure 12, is displayed after the staff chooses to indicate preference. The staff will be able to view the qualifications that they have, indicated as green boxes. Thereafter, they can select one or more instruments that they are qualified for as their job preference for the week. After submitting their job preference, a green confirmation message “Preference Indicated Successfully” will be displayed after the system added their preferences into the database.

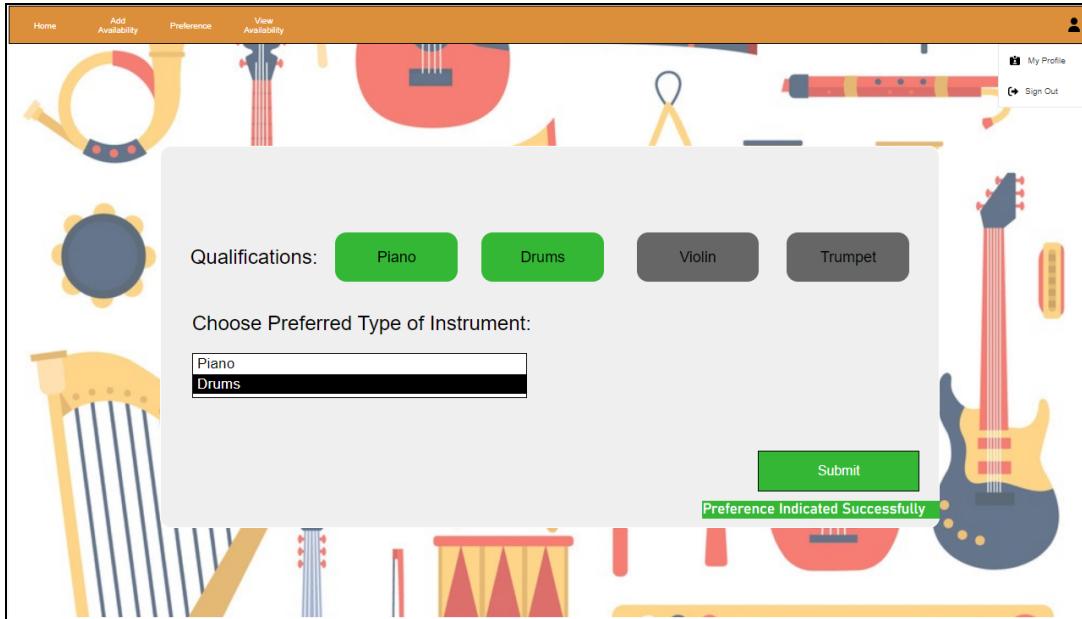


Figure 12: Staff Indicate Job Preference Page

View and Edit Availability Page

The Staff View and Edit Availability Page as shown in Figure 13, is displayed after the staff chooses to view availability. The staff is able to view and edit availability up to 5 weeks ahead of time that were previously submitted by them. After viewing their availability, the staff can choose to edit them by selecting the modify option next to the specific date. Thereafter, time slots will be available for the staff to reselect their timings. After submitting their new availability, a green confirmation message “Edit Availability Successfully” will be displayed after the system modified the availability of the staff in the database.

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
12/09/22 Edit	13/09/22	14/09/22 Edit	15/09/22	16/09/22	17/09/22	18/09/22 Edit
<input checked="" type="checkbox"/> 9:00 AM - 9:30 AM <input checked="" type="checkbox"/> 9:30 AM - 10:00 AM <input type="checkbox"/> 10:00 AM - 10:30 AM	Nil	9.00AM - 9 PM	Nil	Nil	Nil	<input type="checkbox"/> 9:00 AM - 9:30 AM <input checked="" type="checkbox"/> 9:30 AM - 10:00 AM <input type="checkbox"/> 10:00 AM - 10:30 AM
19/09/22	20/09/22	21/09/22	22/09/22	23/09/22	24/09/22	25/09/22
Nil	Nil	Nil	Nil	Nil	Nil	Nil
26/09/22	27/09/22	28/09/22	29/09/22	30/09/22	01/10/22	02/10/22
Nil	Nil	Nil	Nil	Nil	Nil	Nil
03/10/22	04/10/22	05/10/22	06/10/22	07/10/22	08/10/22	09/10/22
Nil	Nil	Nil	Nil	Nil	Nil	Nil
10/10/22	11/10/22	12/10/22	13/10/22	14/10/22	15/10/22	16/10/22
Nil	Nil	Nil	Nil	Nil	Nil	Nil

Figure 13: Staff View and Edit Availability Page

5.4 IT Administrator Pages

The following wireframes will demonstrate the use cases UC11-15: View Accounts, Add Accounts, Edit Accounts and Delete Accounts.

Dashboard Page

The IT Administrator Dashboard Page, as shown in Figure 14, is displayed after the staff has logged in. The IT Administrator will oversee adding new staff and managers to the system as well as managing accounts.



Figure 14: IT Admin Dashboard Page

Create Account Page

The IT Administrator Create Account Page as shown in Figure 15, is displayed after the IT administrator chooses to create an account. For the input fields, the IT administrator can select either “Manager” or “Staff” for the Staff role, the staff/manager ID for the Staff ID, the qualification for the Staff qualification if creating a staff account, the staff/manager name for the Staff name, the staff/manager email for the Staff email, the staff/manager contact information for the Staff contact, and lastly the password for the Staff password.

After submitting the new account information, a green confirmation message “Account Created Successfully” will be displayed after the system adds the account into the database. Thereafter, the system will send an email to the account’s email with the login details of the account.

Figure 15: IT Admin Create Account Page

Manage Account Page

The IT Administrator Manage Account Page as shown in Figure 16, is displayed after the IT administrator chooses to manage an account by viewing, editing or deleting accounts.

Name	Instrument(s) Qualified	Email	Mobile No.	Password			
Eric Warren	Piano	ericwarren@gmail.com	91231234	*****	P	E	S
Iolo Holloway	Piano	ioloholloway@gmail.com	81232412	*****	P	E	S
Shane Holden	Piano	shaneholden@gmail.com	85123451	*****	P	E	S
Izaac Beard	Piano	izaacbeard@gmail.com	85352321	*****	P	E	S
Raymond Christian	Piano	raymondchristian@gmail.com	88341234	*****	P	E	S
Marie Sanderson	Drums	mariesanderson@gmail.com	87534212	*****	P	E	S
Dylon Haney	Violin	dylonhaney@gmail.com	98421221	*****	P	E	S
Dorothy Keenan	Violin	dorothykeenan@gmail.com	874231451	*****	P	E	S
Scarlett Thompson	Trumpet	scarlettthompson@gmail.com	98231221	*****	P	E	S

Figure 16: IT Admin Manage Staff Account Page

Test Plan & Prototype

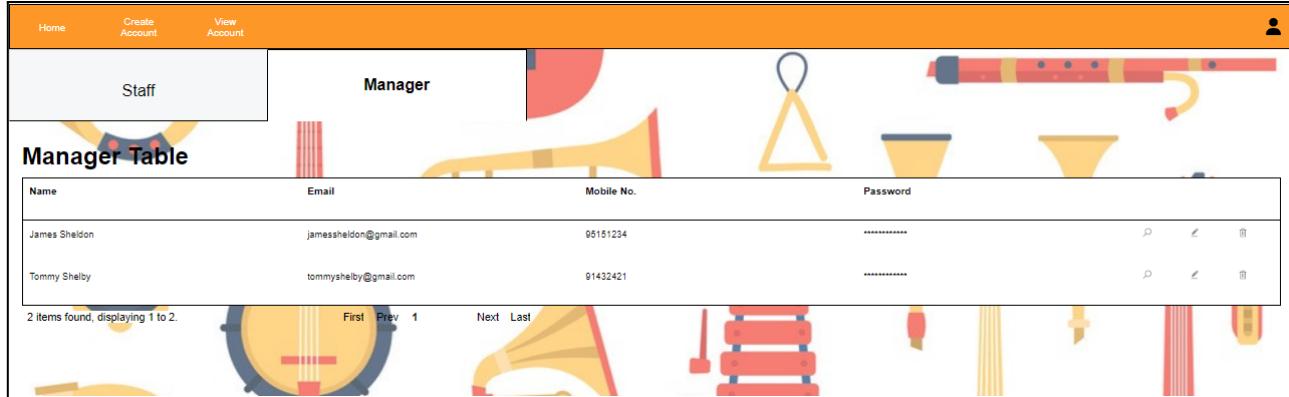


Figure 17: IT Admin Manage Manager Account Page

6 Project Management

The project management section would include the current status of the project with a Gantt chart and the work allocation for this project via the work breakdown structure (WBS) diagram. In addition, an explanation of any overruns in the project will also be included.

The main resource identified for this project would be people power (i.e., time). Hence, the Gantt chart in Figure 18, and WBS diagram in Figure 19 would be the plans to assign the main resource.

6.1 Updated Gantt Chart

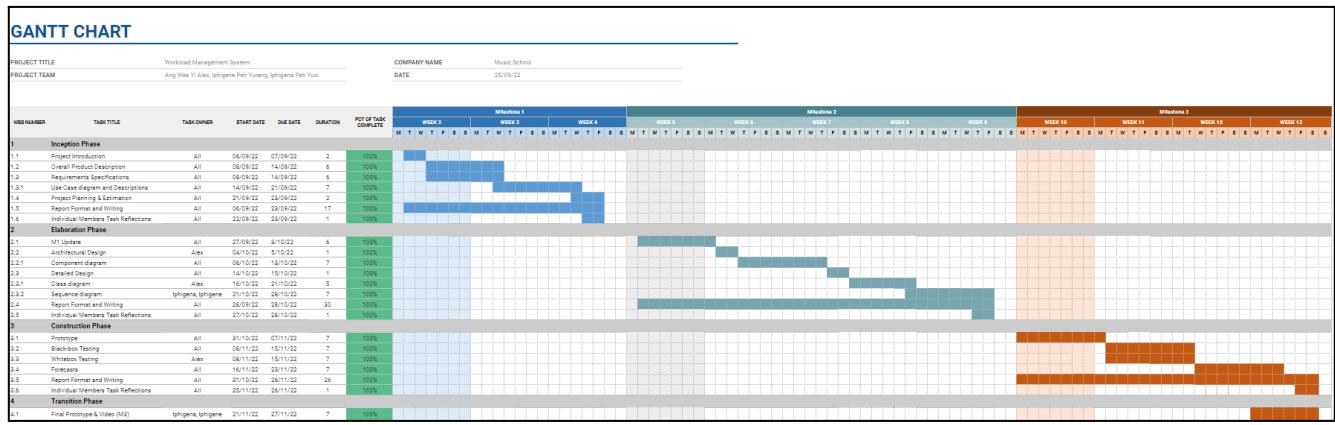


Figure 18: Project Gantt Chart

6.2 Updated work breakdown structure

WORK BREAKDOWN STRUCTURE					
Task ID	Task Description	Dependent Upon	Task Owner	Task Resources	Task Status
1	Inception Phase	(Task ID)			100%
1.1	Project Introduction	-	All	Project Description Documentation	100%
1.2	Overall Product Description	-	All	Project Description Documentation	100%
1.3	Requirements Specifications	1.1, 1.2	All		100%
1.3.1	Use Case diagram and Descriptions	1.2	All	Project Description Documentation	100%
1.4	Project Planning & Estimation	1.3	All	Client Meeting Interview	100%
1.5	Report Format and Writing	All	All	Prototyping	100%
1.6	Individual Members Task Reflections	All	All		100%
2	Elaboration Phase				100%
2.1	Milestone 1 Update	All Task 1	All	M1 Report Feedback	100%
2.2	Architectural Design	1.3	Alex		100%
2.2.1	Component diagram	1.3	All		100%
2.3	Detailed Design	1.3	All	Project Description Documentation	100%
2.3.1	Class diagram	1.3	Alex	Client Meeting Interview	100%
2.3.2	Sequence diagram	1.3	Iphigenia, Iphigene	Prototyping	100%
2.4	Report Format and Writing	All	All		100%
2.5	Individual Members Task Reflections	All	All		100%
3	Construction Phase				100%
3.1	Prototype	All	All	Project Description Documentation	100%
3.2	Black-box Testing	All	All	Client Meeting Interview	100%
3.3	Whitebox Testing	All	All	Prototyping	100%
3.4	Forecasts	All	All	M1 and M2 Report	100%
3.5	Report Format and Writing	All	All	Project Description Documentation	100%
3.6	Individual Members Task Reflections	All	All	Client Meeting Interview	100%
4	Transition Phase			Prototyping	100%
4.1	Final Prototype Video (M3)	All	All	M1 and M2 Report	100%

Figure 19: WBS Breakdown

7 Individual Members Task Reflections

Alex Ang

For milestone 3, I was in charge of identifying tasks that can be concurrently done to allocate to everyone. My main allocated tasks are white-box testing which includes the code and control flow graph and test cases. In addition, I also contributed to the black-box testing which includes the system state diagram and test cases. I also made changes to the class diagram for methods that are missing for the system state diagram. Throughout this milestone, I understood how tedious and extensive it is to ensure that quality test cases are developed for each use case and that it might not be possible to develop test cases for every use case without spending extended periods of time.

If given a chance to do it again, I would choose a less complex method to do white-box testing as this method involves if else statements in a loop which makes it difficult to come up with test cases as well as the control flow graph. As a team of only 3 team members, this project over the three milestones was difficult to manage as there are many requirements that need to be fulfilled but not enough manpower to spread the workload.

Iphigene Peh Yureng

For milestone 3, I was in charge of black box testing and IT admin prototype. I also made changes to the sequence diagram and project management requirements from M2. Throughout this milestone, I learned how intensive each testing procedure can be as it requires careful consideration and attention to detail before deriving the specific test cases for the decision table, boundary value analysis and equivalent partitioning.

If given a chance to do it again, I would choose to understand and clarify any doubts I have with various black box testing techniques such as the boundary value analysis and equivalent partitioning in advance. These techniques can be rather confusing if applied otherwise and without a complete understanding of its purpose. The project would be smoother and more manageable if the required team size of 4 was met, nonetheless the team tried our best to complete all milestones with limited resources/manpower.

Iphigene Peh Yuxi

For milestone 3, I was in charge of all the users' prototypes which included Manager, Staff and IT Administrator, and recording and editing of the 3-min prototype video. I also made changes to the system state diagram and report. Throughout this milestone, I understood how tedious and extensive it is to ensure that the prototype is following all the mentioned use cases and requirements. This helped me to understand how a real web application could be like in real life as creating a prototype greatly simulated the process.

If given a chance to do it again, I would double check the requirements from the very start so as to avoid rechecking the prototype multiple times at the end . As a team of only 3 team members, the workload was quite intensive for the team as everyone had to work on multiple tasks at one go.

8 Appendix A – Use Case Descriptions

Use Case ID:	UC-1
Use Case Name:	View Staff Workload
Description:	This View Staff Workload use case allows the Manager to view the staff workload on the landing page.
Primary Actor:	Manager
Preconditions:	Manager is logged into the system.
Postconditions:	Staff Workload was displayed successfully.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. System displays the workload in hours allocated to each type of instrument 2. System displays the top three staff with the lowest workload. 3. System displays staff with over 40 hours of jobs allocated.
Alternative Scenarios:	

Use Case ID:	UC-2
Use Case Name:	Allocate Job
Description:	The Allocate Job use case allows the Manager to allocate jobs to the staff.
Primary Actor:	Manager
Preconditions:	Manager is logged into the system.
Postconditions:	System allows managers to view staff assigned jobs after they have allocated the job.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. System displays the job allocation page. 2. Manager chooses to view staff availability. 3. System displays staff availability. 4. Manager selects the type of lesson. 5. Manager selects the date and time slots. 6. System fetch available studio and staff. 7. System display dropdown list of studio and staff. 8. Manager selects the studio. 9. Manager selects the staff. 10. Manager submits the information. 11. System adds the job into the database, and displays a confirmation message.
Alternative Scenarios:	<p>7a. No studios available for allocation.</p> <p>7a1. System displays an error message that no studio is available for the selected time slot</p> <p>7b. No staff available for allocation</p>

Test Plan & Prototype

	<p>7b1. System displays an error message that no staff is available for the selected time slot</p> <p>8a. Staff exceeds the four hour consecutive work without 1 hour of rest</p> <p> 8a1. System displays an error message that the staff has exceeded the workload limit without rest.</p>
--	---

Use Case ID:	UC-3
Use Case Name:	Reallocate Job
Description:	The Reallocate Job use case allows the Manager to only reallocate the jobs that were rejected by the staff. Hence, the <>extend>> relationship with the Allocate Job use case.
Primary Actor:	Manager
Preconditions:	Manager is logged into the system. Staff has rejected the job that was previously assigned by the manager.
Postconditions:	The rejected job is reallocated successfully.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. System reflects the rejected jobs in the database. 2. Manager chooses to reallocate the rejected job(s). 3. Manager selects the job to allocate. 4. System displays staff availability 5. Manager allocates the job based on staff availability. 6. System updates the job in the database, and displays a confirmation message.
Alternative Scenarios:	<p>4a. No staff is available for the manager to assign the job.</p> <p> 4a1. System displays an error message saying there is no staff available</p> <p>6a. Staff exceeds the four hour consecutive work without 1 hour of rest</p> <p> 6a1. System displays an error message that the staff has exceeded the workload limit without rest.</p>

Use Case ID:	UC-4
Use Case Name:	View Staff Availability
Description:	The View Staff Availability use case allows the Manager to view the staff availability for job allocation.
Primary Actor:	Manager
Preconditions:	Manager is logged into the system.
Postconditions:	<p>Staff availability was displayed successfully.</p> <p>System allows managers to view staff preference after they have viewed the staff availability.</p>

Test Plan & Prototype

Main Success Scenarios:	<ol style="list-style-type: none"> 1. System extracts the list of staff availability, assigned jobs and job preference from the database. 2. System displays the list of staff availability, assigned jobs and job preference.
Alternative Scenarios:	1a. No staff availability is submitted into the database. 1a1. System displays an error message saying no staff available and provides the reason.

Use Case ID:	UC-5
Use Case Name:	View Assigned Jobs
Description:	The View Assigned Jobs use case allows the Staff to view the jobs assigned by the manager.
Primary Actor:	Staff
Preconditions:	Staff is logged into the system. Manager has already assigned the job(s) to the staff.
Postconditions:	Assigned jobs are displayed successfully.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. System extracts the list of jobs assigned to the staff. 2. System displays the assigned jobs
Alternative Scenarios:	

Use Case ID:	UC-6
Use Case Name:	View Overall Workload
Description:	The View Overall Workload use case allows the staff to view their workload for the current month.
Primary Actor:	Staff
Preconditions:	Staff is logged into the system. System has reflected jobs assigned by managers.
Postconditions:	Overall workload for the staff is displayed successfully.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. System extracts the list of jobs assigned to the staff. 2. System displays staff workload for the current month.
Alternative Scenarios:	

Use Case ID:	UC-7
Use Case Name:	Reject Job Assigned
Description:	The Reject Job Assigned use case allows the Staff to reject the job assigned by the manager.

Test Plan & Prototype

Primary Actor:	Staff
Preconditions:	Staff is logged into the system. Job was previously assigned by managers and recorded into the database.
Postconditions:	Job assigned to staff is rejected successfully.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. System extracts the jobs assigned to the staff from the database. 2. System displays jobs assigned to the staff. 3. Staff selects the job to reject. 4. System display warning message to staff 5. System record rejected jobs into the database.
Alternative Scenarios:	3a. Unable to reject job assigned by manager 3a1. System displays an error message saying unable to reject the job and provides the reason.

Use Case ID:	UC-8
Use Case Name:	Add Availability
Description:	The Add Availability use case allows the staff to add their availability up to 5 weeks ahead of time.
Primary Actor:	Staff
Preconditions:	Staff is logged into the system.
Postconditions:	Staff availability was indicated in the system successfully. System allows staff to indicate their job preference.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. System displays the available time slots in 30 mins intervals. 2. Staff selects the time slots. 3. Staff submit information. 4. System displays a confirmation message.
Alternative Scenarios:	

Use Case ID:	UC-9
Use Case Name:	Indicate Job Preference
Description:	The Indicate Job Preference use case allows the staff to indicate the instruments they want to teach for the week if they have more than one instrument qualification. If the staff only has one qualification, they will not be allowed to indicate their job preference. Staff indicates their preference while adding their availability. Hence, the <<extend>> relationship with the Add Availability use case.
Primary Actor:	Staff
Preconditions:	Staff is logged into the system. Staff have more than one instrument qualification. Staff is adding their availability.

Test Plan & Prototype

Postconditions:	Job preference is indicated in the system successfully.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. System displays the instruments according to the staff qualification. 2. Staff selects the instrument. 3. Staff submits information. 4. System displays a confirmation message.
Alternative Scenarios:	

Use Case ID:	UC-10
Use Case Name:	Edit Availability
Description:	The Edit Availability use case allows the staff to edit their availability up to 5 weeks ahead of time.
Primary Actor:	Staff
Preconditions:	Staff is logged into the system. Staff has viewed their availability.
Postconditions:	Staff availability was updated in the system successfully. System allows staff to indicate their job preference.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. Staff selects the modify option next to the specific date. 2. System displays a popup with time slots selected previously. 3. Staff modifies the information. 4. Staff submit information. 5. System modifies the availability in the database, and displays a confirmation message.
Alternative Scenarios:	

Use Case ID:	UC-11
Use Case Name:	View Availability
Description:	The View Availability use case allows the staff to view their availability up to 5 weeks ahead of time.
Primary Actor:	Staff
Preconditions:	Staff is logged into the system.
Postconditions:	Staff availability was displayed successfully.
Main Success Scenarios:	<ol style="list-style-type: none"> 1. System extracts their availability from the database. 2. System displays their availability.
Alternative Scenarios:	2a. Staff did not submit availability. 2a1. System displays a message saying no availability was added.

Test Plan & Prototype

Use Case ID:	UC-12
Use Case Name:	View Accounts
Description:	The View Accounts use case allows the IT administrators to manage staff and manager accounts by viewing the accounts.
Primary Actor:	IT administrator
Preconditions:	IT administrator is logged into the system.
Postconditions:	Accounts are displayed successfully.
Main Success Scenarios:	<ol style="list-style-type: none">1. System extracts a list of accounts from the database.2. System displays the list of accounts available.
Alternative Scenarios:	<p>1a. No accounts are created for the IT administrator to view. 1a1. System displays an error message saying no accounts created.</p>

Use Case ID:	UC-13
Use Case Name:	Edit Accounts
Description:	The Edit Accounts use case allows the IT administrators to manage staff and manager accounts by editing the account information.
Primary Actor:	IT administrator
Preconditions:	IT administrator is logged into the system. IT administrator has viewed the list of accounts created.
Postconditions:	Account was modified successfully.
Main Success Scenarios:	<ol style="list-style-type: none">1. IT administrator selects the edit option next to the account.2. System displays a popup with details about the selected account.3. IT administrator edits the information.4. IT administrator submits the information.5. System modifies the account information in the database, and displays a confirmation message.
Alternative Scenarios:	

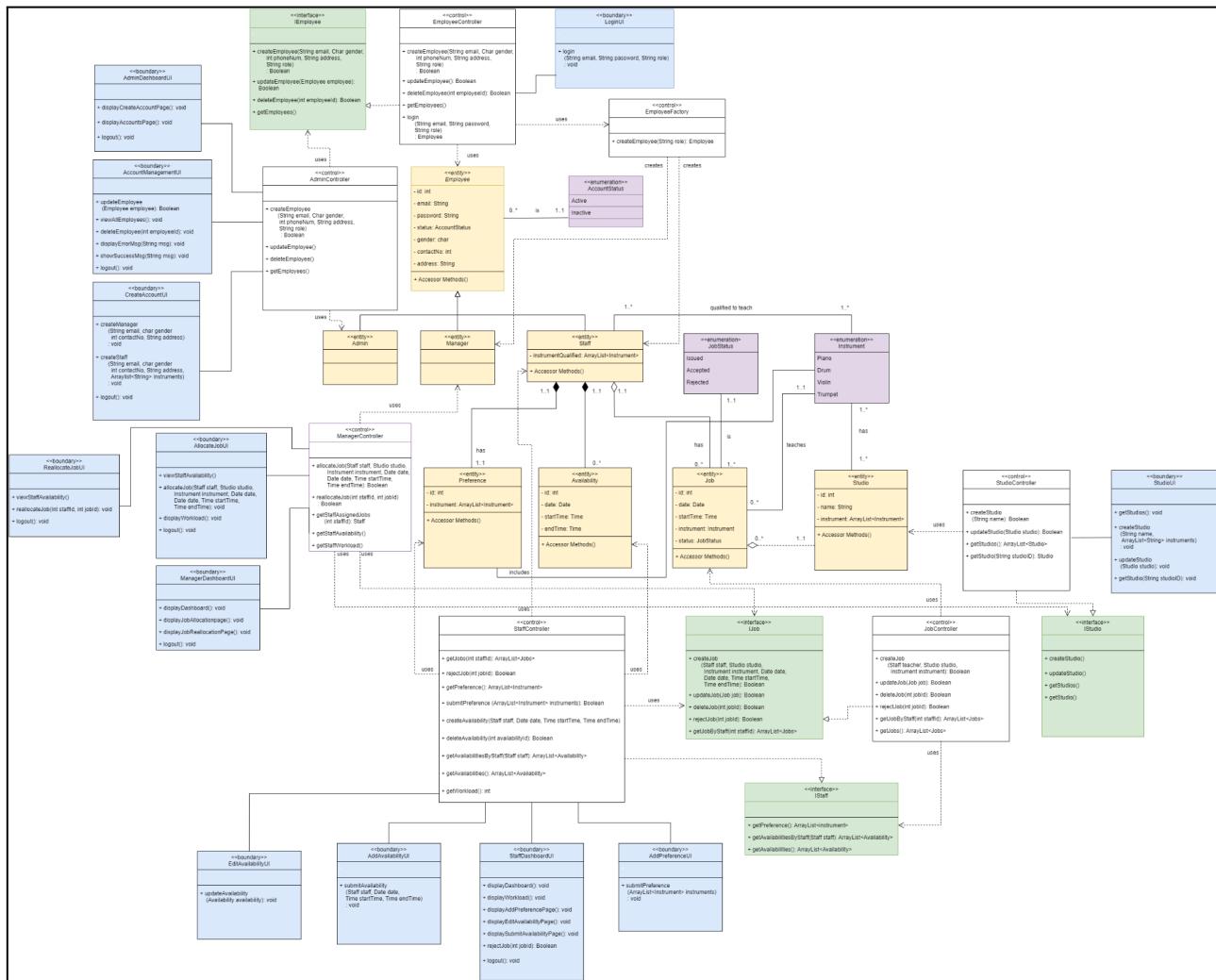
Use Case ID:	UC-14
Use Case Name:	Delete Accounts
Description:	The Delete Accounts use case allows the IT administrator to manage the staff and managers account by deleting the accounts.
Primary Actor:	IT administrator

Test Plan & Prototype

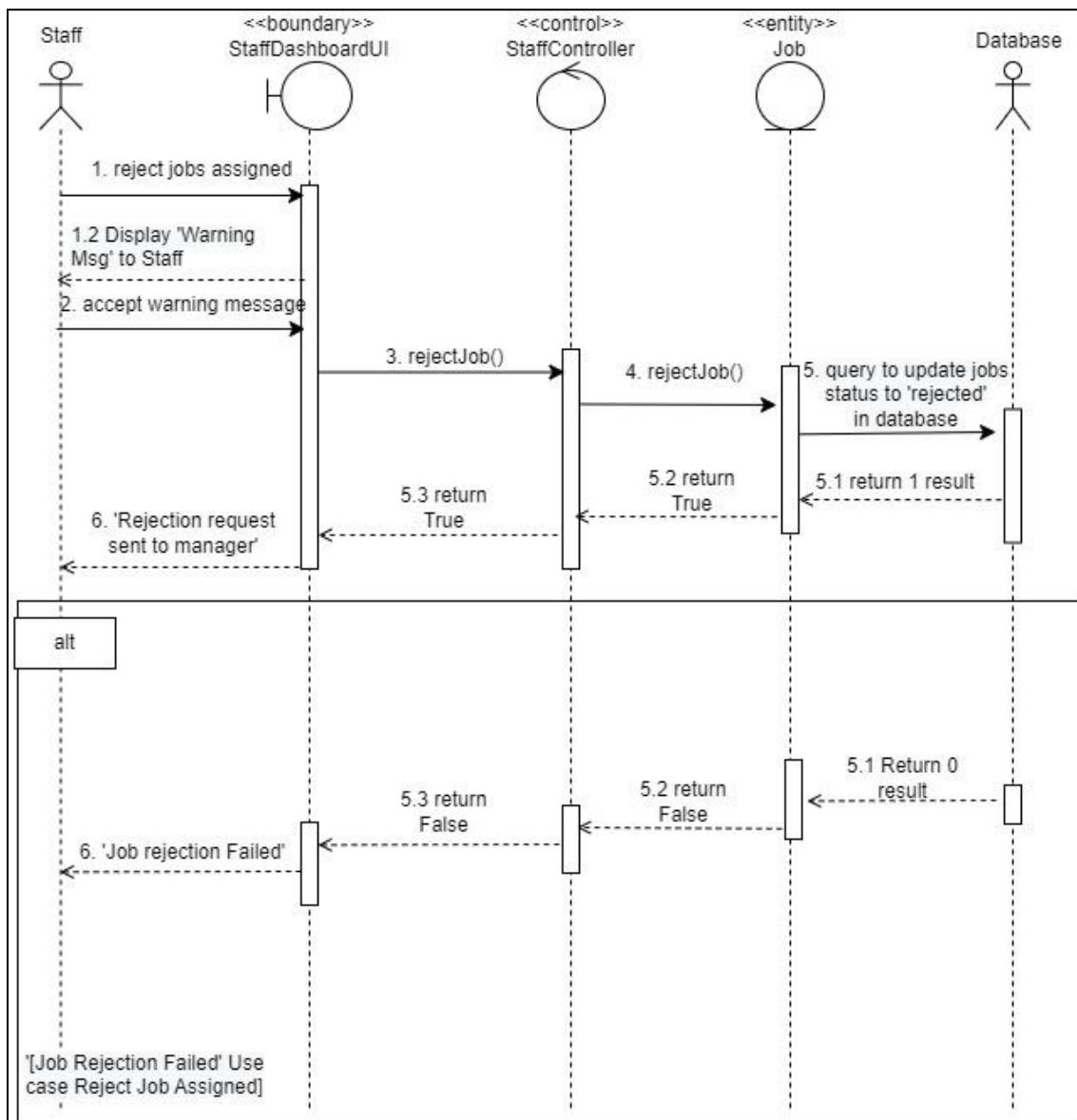
Preconditions:	IT administrator is logged into the system. IT administrator has viewed the list of accounts created.
Postconditions:	Account was deleted successfully.
Main Success Scenarios:	<ol style="list-style-type: none">1. IT administrator selects the delete option next to the account.2. System prompts for delete account confirmation.3. IT administrator confirms by clicking delete.4. System deletes the account in the database, and displays a confirmation message.
Alternative Scenarios:	

Use Case ID:	UC-15
Use Case Name:	Create Accounts
Description:	The Create Accounts use case allows the IT administrator to manage the staff and managers account by creating the accounts.
Primary Actor:	IT administrator
Preconditions:	IT administrator is logged into the system.
Postconditions:	Account was created successfully.
Main Success Scenarios:	<ol style="list-style-type: none">1. IT administrator selects either “Manager” or “Staff”.2. IT administrator enters the staff/manager ID.3. IT administrator enters the staff/manager name.4. IT administrator enters the staff/manager email.5. IT administrator enters the staff/manager contact information.6. IT administrator enters the staff/manager password.7. If creating a staff account, IT administrator enters the qualification.8. IT administrator submit information.9. System adds the account into the database, and displays a confirmation message.10. System sends an email to the account’s email with the login details of the account
Alternative Scenarios:	4a. Email already exists. 4a1. System displays an error message saying the email already exists.

9 Appendix B - Overall Class Diagram



10 Appendix C - Updated Sequence Diagram



11 Appendix D – Data Dictionary

MongoDB

MongoDB is a source-available cross-platform document-oriented database program which uses JSON-like documents with optional schemas.

ExpressJS

Express.js is a back end web application framework, designed for building web applications and RESTful APIs with Node.js. It is released as free and open-source software under the MIT License.

NodeJS

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on a JavaScript Engine and executes JavaScript code outside a web browser.

AngularJS

AngularJS is a discontinued free and open-source JavaScript-based web framework for developing single-page applications.

Application Programming Interface

Application Programming Interface (API) is a set of definitions and protocols for building and integrating application software.

AES256

256-bit AES encryption is a technique that uses a key length of 256 bits for this process. The AES-256 key has the mathematical equivalent of 2²⁵⁶ potential combinations because key combinations increase exponentially with key size. Using 256-bit AES encryption assures the security of your data at rest.

2FA

Two-factor authentication (2FA) is a security mechanism for identity and access management that requires two forms of identification to access services and data.

MFA

Multi-factor authentication is a type of electronic authentication in which a user is only permitted access to a website or application after successfully providing two or more pieces of evidence to an authentication mechanism.

SSL

Secure Sockets Layer (SSL), is an encryption-based Internet security protocol. It was first developed by Netscape in 1995 to ensure privacy, authentication, and data integrity in Internet communications. SSL is the predecessor to the modern TLS encryption used today.

Denial of Service

Denial of service (DoS) is a type of cyber attack that aims to disable, shut down or disrupt a network, website or service.

UR

Test Plan & Prototype

User Interface Requirement (UR) includes content presentation, application navigation, and user assistance of the product.

PF

Product Functionality (PR) is the functions that the product must perform or must let the user perform.

FR

Functional Requirements (FR) captures the intended behaviour of the system. This behavior may be expressed as services, tasks or functions the system is required to perform.

NFR

Non-functional Requirements (FR) includes generic properties of the system that focuses on system usability.

12 Appendix E – Test Cases Result

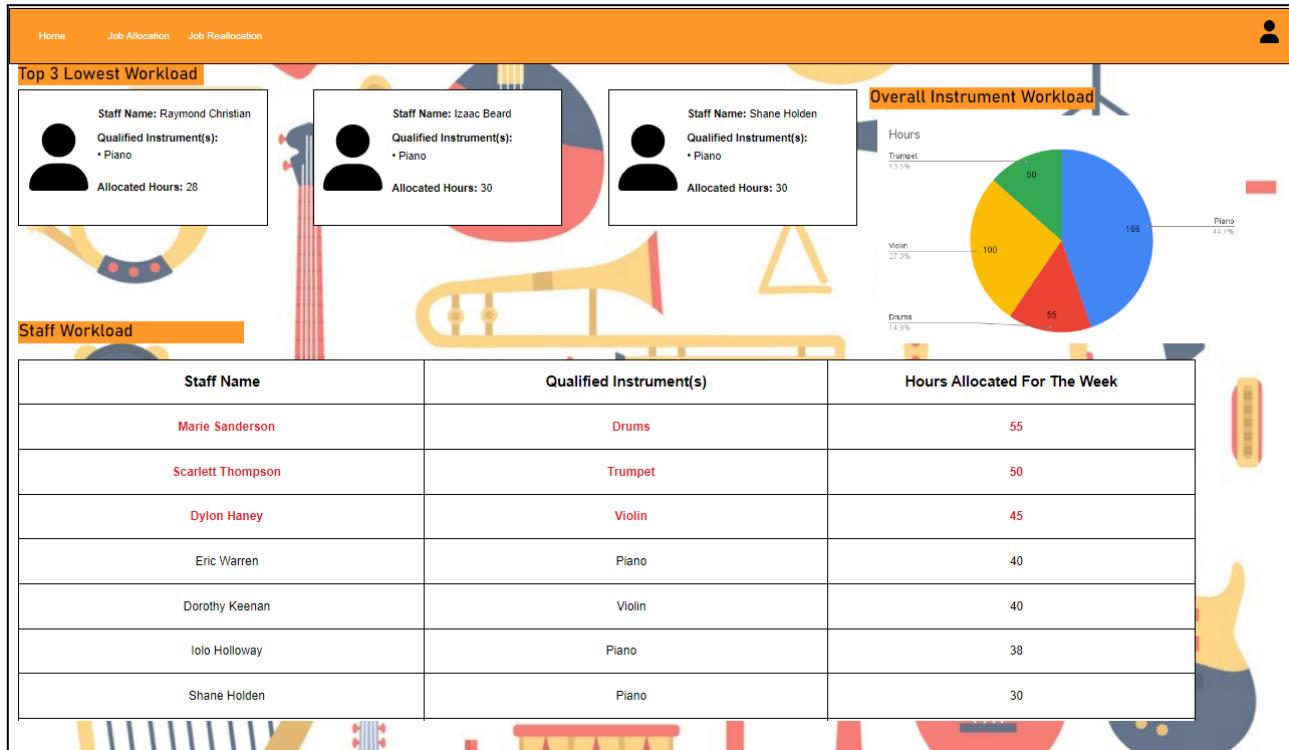


Figure 20: Successfully transit to staff dashboard state after Successful Login

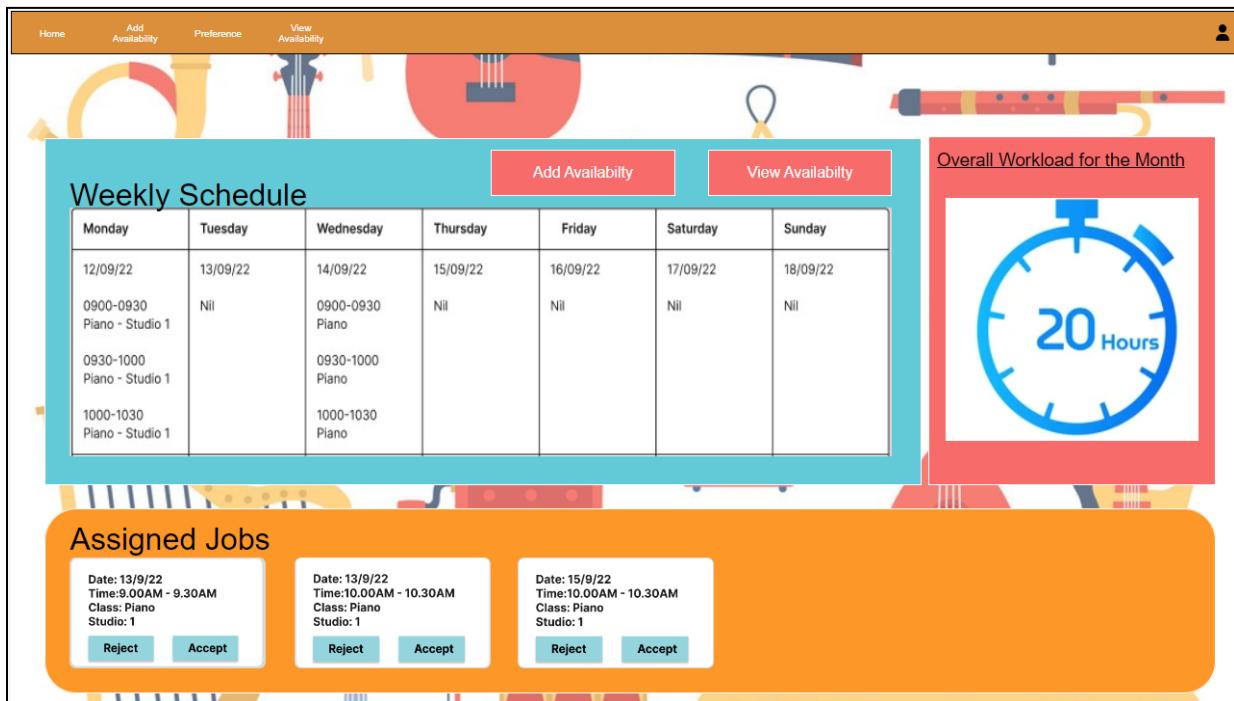


Figure 21: Successfully transit to manager dashboard state after Successful Login



Figure 22: Successfully transit to IT admin dashboard state after Successful Login

A screenshot of a "Create Account" form. The form fields include: Staff role (Manager selected), Staff ID (006), Staff qualification (Drums, Piano), Staff name (Laury), Staff email (Laury@musicchool.com), Staff contact (98761234), and Staff password (PassW0rd123). A green "Submit" button is at the bottom. A red-bordered box highlights the message "Account Created Successfully" which appears below the submit button. The background features various musical instruments.

Figure 23: Account Created Successfully Message

Test Plan & Prototype

The screenshot shows a user interface for job allocation. At the top, there are tabs for Home, Job Allocation, and Job Reallocation. Below the tabs, a search bar displays "Staff ID: 001" and a button labeled "Show Staff Availability". A section titled "Staff Availability" contains a table for "Staff 1" with fields for Name (Raymond Christian), Workload Assigned (Allocated Hours: 28), Job Preference (Piano), Staff Location (Location A), and Availabilities (a grid of availability status for each day of the week). To the right, there are filters for Lesson Type (Piano, Drums, Violin, Trumpet), Date (11/30/2022), Time (09:00 AM to 12:00 PM), Studio (2), and Staff (Jane). A "Submit" button is present, and a green success message at the bottom right reads "Job Allocated Successfully".

Figure 24: Job Allocated Successfully Message

The screenshot shows a user interface for job reallocation. At the top, there are tabs for Home, Job Allocation, and Job Reallocation. A red banner at the top left says "Rejected Jobs" and lists two rejected staff entries: "Staff Name: Jane, Staff ID: 001, Date: 5/12/22, Class: Piano, Studio: 1" and "Staff Name: Raymond, Staff ID: 002, Date: 6/12/22, Time: 10:00 AM -10:30 AM, Class: Piano, Studio: 1", each with a "Reallocate" button. Below this, a search bar shows "Staff ID: 002" and a "Show Staff Availability" button. A section titled "Staff 2" contains fields for Name (Raymond Christian), Workload Assigned (Allocated Hours: 28), Job Preference (Piano), Staff Location (Location A), and Availabilities (a grid of availability status for each day of the week). To the right, a "Reallocation" section shows a job entry for "Staff Name: Jane, Staff ID: 001, Date: 5/12/22, Class: Piano, Studio: 1" and a "Reallocated To:" field containing "Staff ID: 002". A "Submit" button is present, and a green success message at the bottom right reads "Job Reallocated Successfully".

Figure 25: Job Reallocated Successfully Message

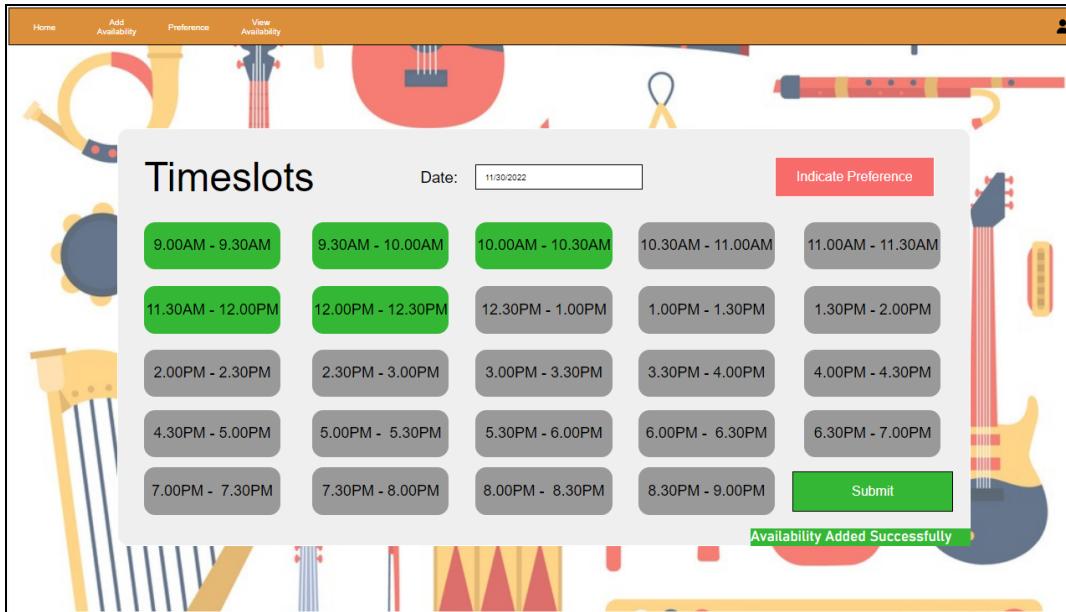


Figure 26: Availability Added Successfully Message