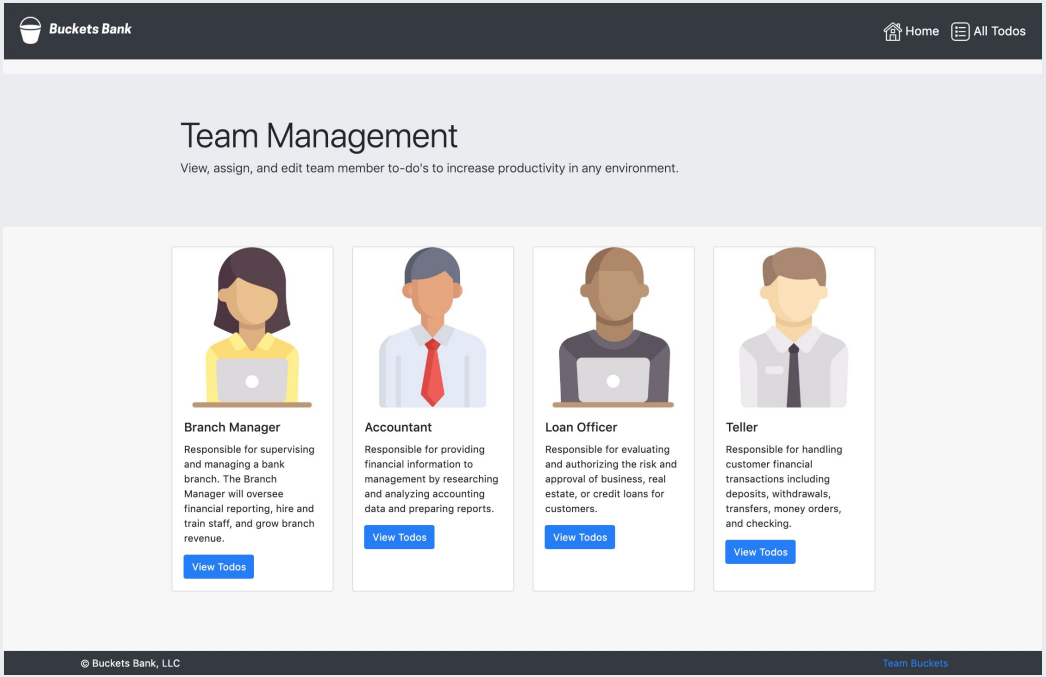
Buckets Bank

Team Buckets:
- John Truong (Product Owner)
- Solon Burleson (Scrum Master)
- Bennett Black
- Orlando Chavez
- Jordan Mao

Business Owner:
- Chethan Shriyan

Stakeholders:
- Samuel Dijana
- Courtney Heinrich
- Matthew Truelove



Presentation Overview

1.	Introduction	Bennett
2.	Swim Lane Diagrams	Jordan
3.	Gantt Chart	Solon
4.	Challenges	Solon
5.	Demo	All
6.	Testing	John
7.	Conclusion	John
8.	Future Implementations	Orlando

Introduction

Who is it for?

- This particular app is for employees of common banking positions, Accountants, Branch Managers, Tellers, Loan officers

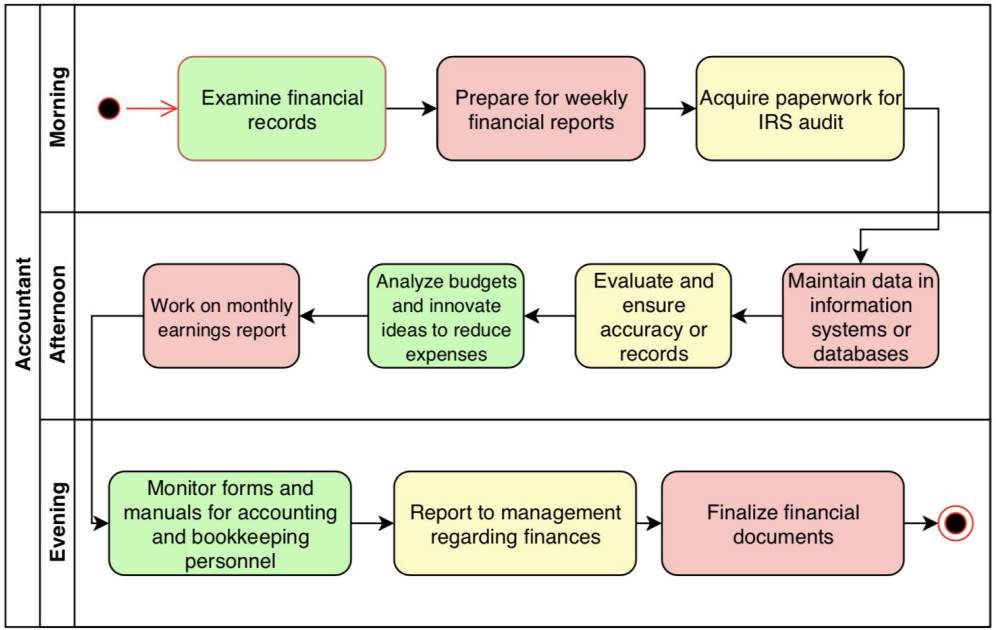
Why did we make it?

- We made this app to help multiple users of one organization stay organized and increase productivity

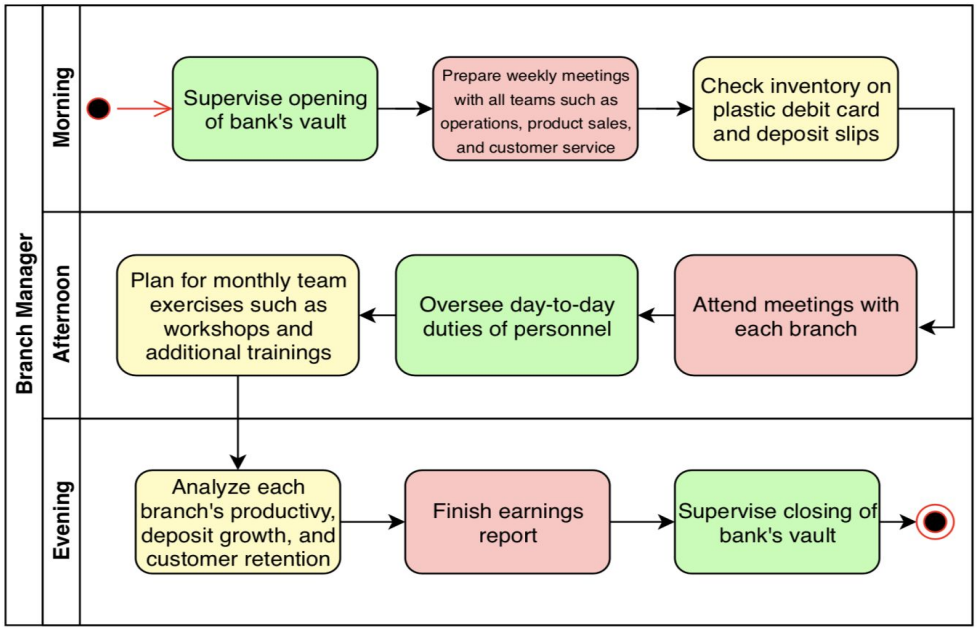
What does it do?

- Allows you to easily add to-do tasks, set a due dates & priorities, and easily mark them as complete.

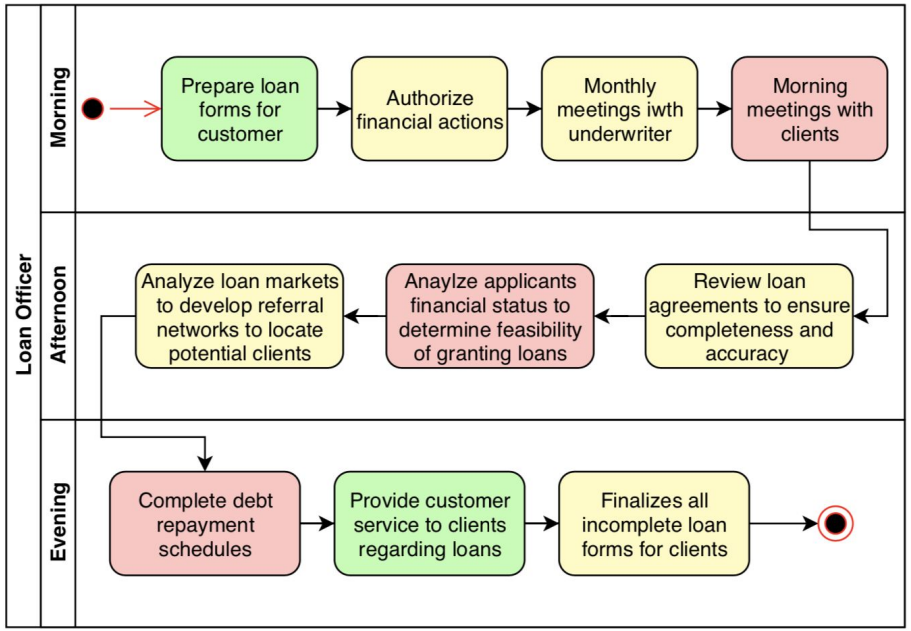
Swim Lane Diagrams (Accountant)



Swim Lane Diagrams (Branch Manager)



Swim Lane Diagrams (Loan Officer)

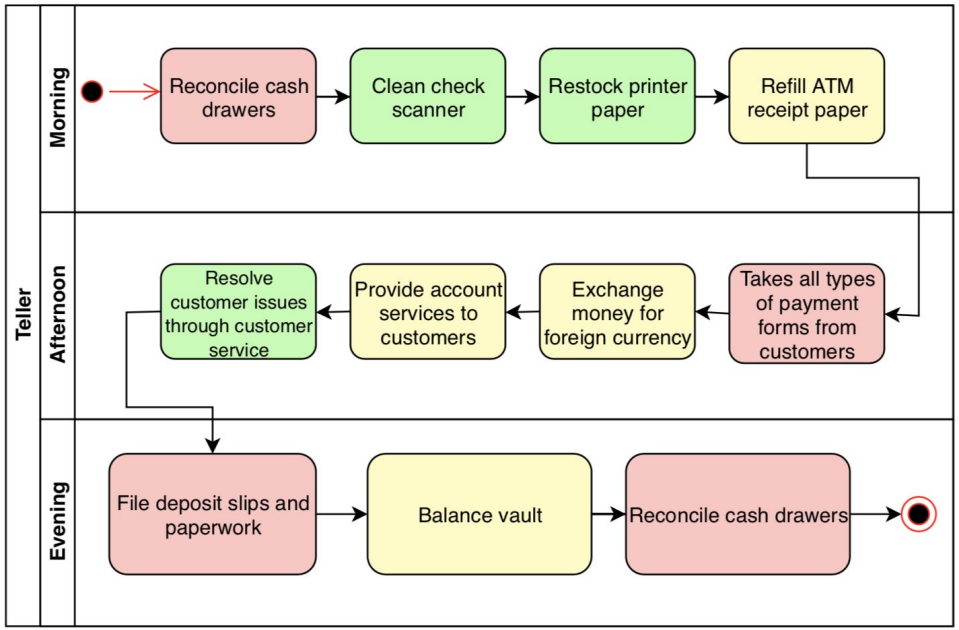


High

Medium

Low

Swim Lane Diagrams (Teller)



High

Medium

Low

Gantt Chart

TASK NAME	START DATE	DAY OF MONTH*	END DATE	DURATION* (WORK DAYS)	DAYS COMPLETE*	DAYS REMAINING*	TEAM MEMBER	PERCENT COMPLETE
REST API/UI Layout								
GitHub Setup	1/17	17	1/17	1	1	0	Solon	100%
Study Git Branching	1/18	18	1/18	1	1	0	All	100%
User Model	1/18	18	1/18	1	1	0	John	100%
Todo Model	1/18	18	1/18	1	1	0	Bennett	100%
User Service	1/18	18	1/18	1	1	0	Orlando	100%
Todo Service	1/18	18	1/18	1	1	0	Jordan	100%
Basic UI Layout	1/18	18	1/18	1	1	0	Solon	100%
Study Angular Features	1/18	18	1/18	1	1	0	All	100%
Second Sample Project								
Service Placeholder Replace/Controller	1/19	19	1/19	1	1	0	Bennett	100%
Study&Implementation Angular	1/19	19	1/20	2	0.5	2	John	25%
Study&Implementation CORS	1/19	19	1/20	2	2	0	All	100%
Study&Implementation Angular Subscribe	1/19	19	1/20	2	2	0	Orlando,	100%
Third Sample Project								
Complete Swim Lane Diagram	1/21	21	1/21	1	1	0	John	100%
HATEOAS/Unit Testing	1/21	21	1/22	2	2	0	John	100%
Landing page	1/21	21	1/22	2	2	0	Bennett	100%
Post/Edit/Delete Todo	1/21	21	1/22	2	2	0	Solon,	100%
User/Todo Services Angular	1/21	21	1/22	2	2	0	Solon	100%
Prepare for Final Presentation	1/21	21	1/22	2	2	0	All	100%

Challenges

- First Sprint Cycle
  - Establishing one-to-many & many-to-one mappings
  - Passing date formats from Angular to Java to MySQL
- Second Sprint Cycle
  - Trouble connecting front end components with backend endpoints
  - Git conflicts with multiple users
- Third Sprint Cycle
  - HATEOAS implementation issues
  - CRUD form submission buttons
  - Junit testing with mock mvc

DEMO TIME!

Frontend Testing

Test Case ID	FE1	Test Case Description	Test view of each component		
Created By	John	Reviewed By	Bennett	Version	1.0
QA Tester's Log	Tested with HATEOUS incomplete				
Tester's Name	John	Date Tested	January 19, 2020	Test Case (Pass/Fail/Not	Pass

S #	Prerequisites:	S #	Test Data
1	Server hosting Angular app on port 4200	1	
2		2	
3		3	
4		4	

Test Scenario	Verify that the view of each component is available by clicking through website

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Open localhost:4200	Home, header, and footer components are rendered	As Expected	Pass
2	Navigate to tab "All Todos"	List of all todos is rendered	As Expected	Pass
3	Navigate through navbar items	Header/footer component is rendered	As Expected	Pass

Frontend Testing

Test Case ID	FE2	**Test Case Description**	Test routing of navigation buttons and navbar items		
Created By	John	**Reviewed By**	Bennett	**Version**	1.0
QA Tester's Log	Tested before second scrum cycle				
Tester's Name	John	**Date Tested**	January 21, 2020	**Test Case (Pass/Fail/Not**	Pass

S #	Prerequisites:	S #	Test Data
1	Server hosting Angular app on port 4200	1	
2		2	
3		3	
4		4	

Test Scenario	Click on buttons and navbar items ensuring that its routed their respective uri

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Click buttons attached to home component checking routing	Redirects the user to specific uri mapping to specific role in the bank	As Expected	Pass
2	Click through all navbar items	Redirects to respective uri page	As Expected	Pass
3				

Frontend Testing

Test Case ID	BE1	**Test Case Description**	Verify UserController REST api endpoints		
Created By	John	**Reviewed By**	Solon	**Version**	1.0
QA Tester's Log	Tested before second scrum cycle				
Tester's Name	John	**Date Tested**	January 22, 2020	**Test Case (Pass/Fail/Not**	Pass

S #	Prerequisites:
1	Tomcat hosting Spring boot app on port 8080
2	Postman
3	
4	

S #	Test Data
1	{ "todo": "General todo", "dueDate": "2020-01-20", "status": false, "priority": 2 }
2	{ "todo": "Edited todo", "dueDate": "2020-01-20", "status": false, "priority": 2 }
3	
4	

Test Scenario Utilize postman to check functionalities of GET, POST, PUT, and DELETE http request corresponding to endpoints

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Use postman request body to POST new todos	Todo is added to remote database	As Expected	Pass
2	Use postman request body to PUT new todos	Updates the todo using the path variables to associate which todo is updating	As Expected	Pass
3	Provide proper uri for postman to GET todos and check output	Get JSON formatted output of list of todos and todo specified by todo_id	As Expected	Pass
4	Provide proper uri for postman to DELETE todo	Todo is deleted from remote database	As Expected	Pass

Cloud Database Testing

Test Case ID	DB1	**Test Case Description**	Test the Login Functionality in remote database in AWS		
Created By	John	**Reviewed By**	Bennett	**Version**	1.0
QA Tester's Log	Tested before starting on rest api endpoints				
Tester's Name	John	**Date Tested**	January 22, 2020	**Test Case (Pass/Fail/Not**	Pass

S #	Prerequisites:
1	
2	
3	
4	

S #	Test Data
1	Username = Bennett
2	Password = [redacted]
3	
4	

Test Scenario Verify that we can connect to remote database using correct credentials

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Create remote connection in MySql workbench	New localhost connection is created in MySql workbench	As Expected	Pass
2	Enter user credentials to database	Connection is established	As Expected	Pass
3	Verify successful connection	Database is loaded in MySql workbench	As Expected	Pass

Conclusion

- Developed fully functional restful web application designated for different users in bank
- Allows users to add, edit, and delete todos making it easy planning out their daily schedule
- Cloud database
- Met business owner requirements

Future Implementations

- User registration
- User authentication
- User profiles
- Complete cloud deployment
- Pagination and search features
- Junit test with mock mvc

Questions?