





















3/28/2025

# Weekly Report 2



---

## Progress Summary

### Scope Progress

Scope Table		
Table1 		
Title 	Status 	Date Completed 
get static website set up and hosted	Complete 	3/19/2025
design lightweight database schema	Complete 	3/24/2025
database is accessible through docker-compose (for testing and development)	Complete 	3/24/2025
set up core views & viewmodels with navigation	Complete 	3/19/2025
API is set up in Azure	Complete 	3/25/2025
automated testing for API deployment	Complete 	3/24/2025
set up pipeline for linting and treating build warnings as errors	Complete 	3/25/2025
Set up app-wide consistent styling	Complete 	3/25/2025
set up database in Azure	In Progress 	
API endpoints for courses, assignments, and to-do items	In Progress 	
basic auth setup	In Progress 	
integration tests for API, database	In Progress 	
unit tests	In Progress 	
services set up for interacting with the db	In Progress 	
API for managing assignments, authentication, and To-Do items	In Progress 	
To-Do list is functional (adding tasks/subtasks, updating status)	In Progress 	
Assignments - can add assignments and associate them with courses	In Progress 	
testing for in-memory services	Not Started 	
Get data for stats page	Not Started 	
View and viewmodel for displaying stats	Not Started 	

## Learning Outcomes Progress

Learning Outcomes			
Table2 			
Title	 Status	Date Completed	
Become familiar with a mobile application development environment.	Complete	3/18/2025	
Utilize advanced techniques such as data binding to minimize the amount of UI code required in a mobile application.	Complete	3/21/2025	
Be able to recognize and use common GUI framework elements, patterns and concepts.	Complete	3/26/2025	
Be able to deploy an existing application to the public cloud.	Complete	3/20/2025	
Design and lay out mobile application user interfaces.	In Progress		
Be comfortable testing mobile application code.	In Progress		
Use security controls and technologies available to properly isolate and secure cloud resources.	In Progress		
Practice automating the compiling, testing and deploying of your code directly into a production environment.	In Progress		
Understand the basics of mobile application lifecycle.	Not Started		
Understand the differences between virtualization and containerization.	Not Started		
Develop skills necessary to deploy and manage code in a public cloud environment.	Not Started		
Be able to compare and contrast the differences and tradeoffs between IaaS, PaaS, SaaS, and other cloud models.	Not Started		

Work Logs

These time logs outline our meetings spent mob programming, debugging, and planning as a group.

Group Sessions								
Date	Attendance	Time Started	Anticipated Duration	Goal/Focus	How You'll Know You're Done	Time Stopped	Elapsed	Additional Notes
3/24/2025	Audrey, Cody, Mateo, Taft	1:18:00 PM	60 min	- review DB schema - basic testing setup for API services - make sure everyone's editor config is working	have a setup for testing the API services with a database	2:24:45 PM	1:06:45	
3/24/2025	Audrey, Cody, Mateo, Taft	3:30:00 PM	2.5 hrs	- API and testing for database services - integration tests running in a workflow	When we have api endpoints that can be tested in the workflow	6:03:00 PM	2:33:00	We failed at getting the endpoints accessible
3/25/2025	Audrey, Cody, Mateo, Taft	1:18:00 PM	30 min	- Review Taft's update on To-Do list - Do we need to upload these to the DB? - Review tests (unit) - Discuss API Services	Fix on To-Do items Decide on unit tests for Friday	1:53:00 PM	0:35:00	Linting fails in pipeline (or just enforce with dotnet format)
3/25/2025	Audrey, Cody, Mateo, Taft	4:55:45 PM	1 hr	- Linting for Alex's class - Mock Services - Unit tests for mocked services - Static site docs?	Tests are running in the pipeline In-memory service is testable and ready Discussed docs	5:55:00 PM	0:59:15	
3/26/2025	Audrey, Cody, Taft	12:27:00 PM	2 hrs	- Draw API paths - Discuss MAUI styling - Discuss plans for future styling? - In-memory service testing	When we have a good idea of our API endpoints	2:20:40 PM	1:53:40	
3/26/2025	Audrey, Cody, Mateo, Taft	3:30 PM	2.5 hrs	- Discuss Mateo's workaround - Finish MAUI ToDo Integration with API	have API endpoints for To-Do	5:04:30 PM	1:34:30	
3/27/2025	Audrey, Cody, Mateo	1:00:16 PM	1 hr	- Discuss Auth? - Discuss DB stuff - Start Kubernetes stuff	When class starts	1:58:00 PM	0:57:44	
3/27/2025	Audrey, Cody, Mateo, Taft	4:50 PM	1.25 hrs	- Pair programming, layouts - Work on Kubernetes - Work on Auth	Have a rough view for assignment management, and get API and DB in kubernetes	6:17:14 PM	1:27:14	
3/28/2025	Audrey, Cody, Mateo, Taft	1:21:22 PM	1 hr	- quickly run through Alex's assignments - discuss linting pipeline - outline for Monday's demo - revisit & build out scope table - compile weekly report	- have a current scope table for tomorrow's submission - fixed pipeline	2:19:20 PM	0:57:58	

Weekly Report 2 Summary			
	Individual Hours	Group Hours	Total Hours
Audrey	6:19:47	12:05:06	18:24:52
Cody	3:08:00	12:05:06	15:13:06
Mateo	2:20:34	10:05:06	12:25:40
Taft	1:30:00	11:08:06	12:38:06
Group Hours	12:05:06		

<b>Total Time Spent</b>	<b>46:03:38</b>		
-------------------------	-----------------	--	--

## Blocking Issues

Share 3+ different blocking issues that you were stuck on for a while, and how you resolved them.

### 1. Deploying API to Azure

- **Description:**
  - Deployment failed after modifying the default workflow file and removing artifacts.
- **Attempts To Fix:**
  - Tried printing the working directory in the pipeline. That and reading the errors revealed that it wasn't finding the right dll file.
- **Solution:**
  - Added the path to "myapp" to the **package** in the deployment step.

### 2. Integration tests don't run in workflow

- **Description:**
  - Integration tests fail every time, even in the correct directory.
- **Attempts To Fix:**
  - Tried running 'ls' in the Dockerfile, and saw that we were in the correct directory.
- **Solution:**
  - Read the error message again and realized that the T-Unit test project was using net8.0, and the rest of our project was using net9.0. Changing it to 9.0 solved the problem.

### 3. Connection refused in integration tests

- **Description:**
  - In our first integration tests, we tried to create an HTTP Client to access the API on localhost port 8080. When we tried to get a result from the endpoint, it says "connection refused."
- **Attempts To Fix:**
  - Added a logging step to the docker-compose to see the startup logs for the API container. The logs didn't actually print in our workflow logs.
- **Solution:**
  - Realized our target URI was https instead of http. Also, the API's appsettings.json needed to have 0.0.0.0 instead of localhost. Then, we could access the API being run in the docker compose.

### 4. Testing MAUI in Deployment Pipeline

- **Description:**
  - The MAUI project gets recognized as a class library when **dotnet test** gets run in the pipeline. It always results in a failing test, failing the pipeline.
- **Attempts To Fix:**

- Looked for a docker image that could be used to run the tests correctly. Most of them were for the Windows OS, but we found one for Linux.
- **Solution:**
  - Escalated to Alex, who confirmed that we should separate our viewmodel logic into a shared project and test that instead of the MAUI project.

## 5. Controller Routing

- **Description:**
  - Can't assign additional routes on top of the default controller route.
- **Attempts To Fix:**
  - Tried using [Route("update")], but that caused problems in Swagger.
- **Solution:**
  - Just a syntax problem with slashes. Also realized we didn't need a separate attribute for the path.

## 6. Calling Async Methods in Constructor (Without Warnings)

- **Description:**
  - We can't have any warnings (for our deployment pipeline), but we don't know of a clean way to call the relevant async method to get the To-Dos when the viewmodel is created.
- **Attempts To Fix:**
  - Looked through some potential solutions, like [this one](#).
- **Solution:**
  - Call InitializeItems in the code-behind OnAppearing method, treating it like the OnInitializedAsync in Blazor.