⚠️ Errors, Issues, Blockers

Date	Issue	Description	Status / Fix
Week 1 – Onboarding & Setup (July 22)	No onboarding / solo setup	Inherited a project with no team, limited context, and no clear starting point.	Began mapping the project in Notion and created a Week 1 action plan.
Week 2 – Access & Restoration (July 29)	Delayed access to correct FGS folder and SQL database	No access to actual project environment until late Week 2. Studied WordPress and PHP basics while waiting	Restoration tasks began immediately after but slightly behind schedule.
Week 2 – Access & Restoration (July 29)	SQL file too large to import	Database .sql file exceeded phpMyAdmin's size limit on Hostinger.	Planned to use BigDump or contact support for import options.
Week 2 – Access & Restoration (July 29)	Unclear restore order	Unsure if FGS files or SQL DB should go first.	Clarified correct order: import DB → upload FGS folder → config check.
Week 2 – Access & Restoration (July 30)	Live site showing default WordPress	fgs.alankan.net loaded basic install instead of full project.	Confirmed missing SQL and theme files were the cause.
Week 2 – Access & Restoration (July 30)	Couldn't find import DB option	Struggled to locate SQL import in Hostinger dashboard.	Found it under **phpMyAdmin** in Databases.
Week 2 – Access & Restoration (July 30)	Unclear which files to delete in FGS	Unsure if it was safe to delete default WordPress files.	Confirmed they can be deleted if replaced with correct project files.
Week 2 – Access &	Project tracking setup	Needed to get Notion up and	Weekly checklist, blockers table, and Word

Date	Issue	Description	Status / Fix
Restoration (July 30)		running with a structured workflow.	export done.
Week 3 – Debugging & Recovery (Aug 7)	Rogue MySQL string on homepage	Strange command appeared on homepage and dashboard.	Traced to wp-config.php, deleted rogue string above <?php, issue resolved.
Week 5 – Custom Features (Aug 21)	No event system	The legacy site had no structured way to display temple events or activities. I needed a scalable solution that allowed admins to easily add, edit, and archive events.	I built a custom post type using CPT UI and ACF. Registered fields for event title, date, location, and description. Created a custom archive template and rendered events using shortcode logic. Styled the grid manually for full control over layout. This system is modular, scalable, and future-proof.
Week 6 – CSS Architecture (Aug 28)	Styles tangled across pages	CSS was hard to maintain and scale.	Refactored into scoped CSS files, created /css folder.
Week 6 – CSS Architecture (Aug 30)	Layout issues on mobile/tablet	Responsive design broke on smaller screens.	Created QA tracker, documented issues, applied fixes.
Week 7 – GitHub & QA (Sept 4)	Repo not structured for handover	Needed clean folder layout and documentation.	Created /css, /php, /js, /docs, wrote README.
Week 8 – Presentation Prep (Sept 11)	Emotional story not aligned with tech	Needed to connect technical wins with personal journey.	Drafted slides, speaker notes, and visual timeline.
Week 9 – Final Polish (Sept 18)	Final QA and documentation	Last round of testing and cleanup.	Tagged release, exported Notion journal, finalized submission.
Week 10 – Reflection & Submission (Sept 25)	Final reflection and growth	Synthesized technical wins and emotional journey.	Submitted repo, slides, written report, and Notion journal.

Database Connection Error

- What happened: WordPress could not connect to the database
- Why: MySQL password was changed in Hostinger but not updated in wp config
- Fix: Updated wp config with new password and connection worked

WordPress Install Screen

- What happened: WordPress showed install screen instead of site
- Why: Database was connected but core tables like wp options were missing
- Fix: Imported backup database using SSH with force flag to skip errors

SQL Import Errors

- What happened: Errors like table already exists and incorrect table definition
- Why: SQL file had create and alter commands for tables that were already present
- Fix: Used force flag to skip errors without dropping tables as advised by mentor

Site Looked Basic After Import

- What happened: Site showed default Hello world post
- Why: Theme was not set to show correct content and homepage settings were default
- Fixes:
  - Homepage set to static page ID 8068 then changed to ID 8034 Latest News
  - Updated wp options directly using SQL
  - Fixed site URLs to use https
  - Reset admin password and logged into dashboard
  - Updated admin email and waiting for confirmation
  - Purged LiteSpeed cache and confirmed QUIC cloud was off

Current Status

- Logged into WP admin successfully

- Homepage set to Latest News

- Layout still looks default

Next Steps

- Check theme files like front page home and index php

- Confirm homepage layout matches Figma design

Insights and Learnings

- Install screen can appear if core tables are missing even with database connected

- Homepage behavior is controlled by wp options and theme files

- SSH imports with force flag help preserve data without dropping tables

missing plugins and assets

some plugins expected from figma design are missing like custom post types and seo tools
theme images and icons not showing in assets folder
template parts referenced in figma not found in theme files
checked assets folder here fgs theme images folder

console and php errors

rogue mysql string showed up on homepage and dashboard
looked like this
mysql -u u774246273_prfCC -p u774246273_39hrI < database.sql

found it hardcoded above php tag in wp config
deleted everything above <?php and it fixed the issue
no console errors now but php logs show memory limit warnings sometimes

layout and image bugs

homepage layout not matching figma
missing padding and sections not aligned
hero banner image not loading on mobile
temple map and tea house sections need responsive fixes

screenshots to add

rogue string in dashboard
broken layout on mobile
missing image folders