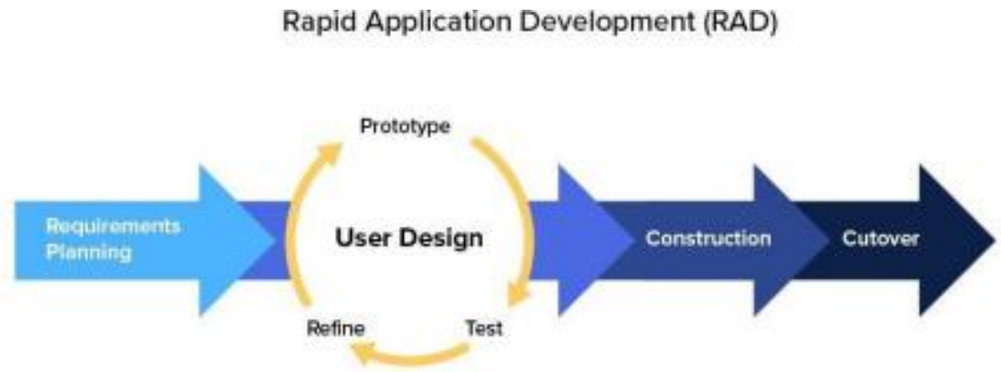
Software Development Life Cycle



Rapid Application Development (RAD) is a software development approach that focuses on fast delivery, prototyping, and continuous user feedback, allowing quick adjustments, and shorter development time compared to traditional methods. RAD is suitable for Labalan Funeral Homes' Customer Service Management System because it enables faster development, allows staff feedback through prototyping, and provides flexibility to adjust features, ensuring the system meets real customer service needs.

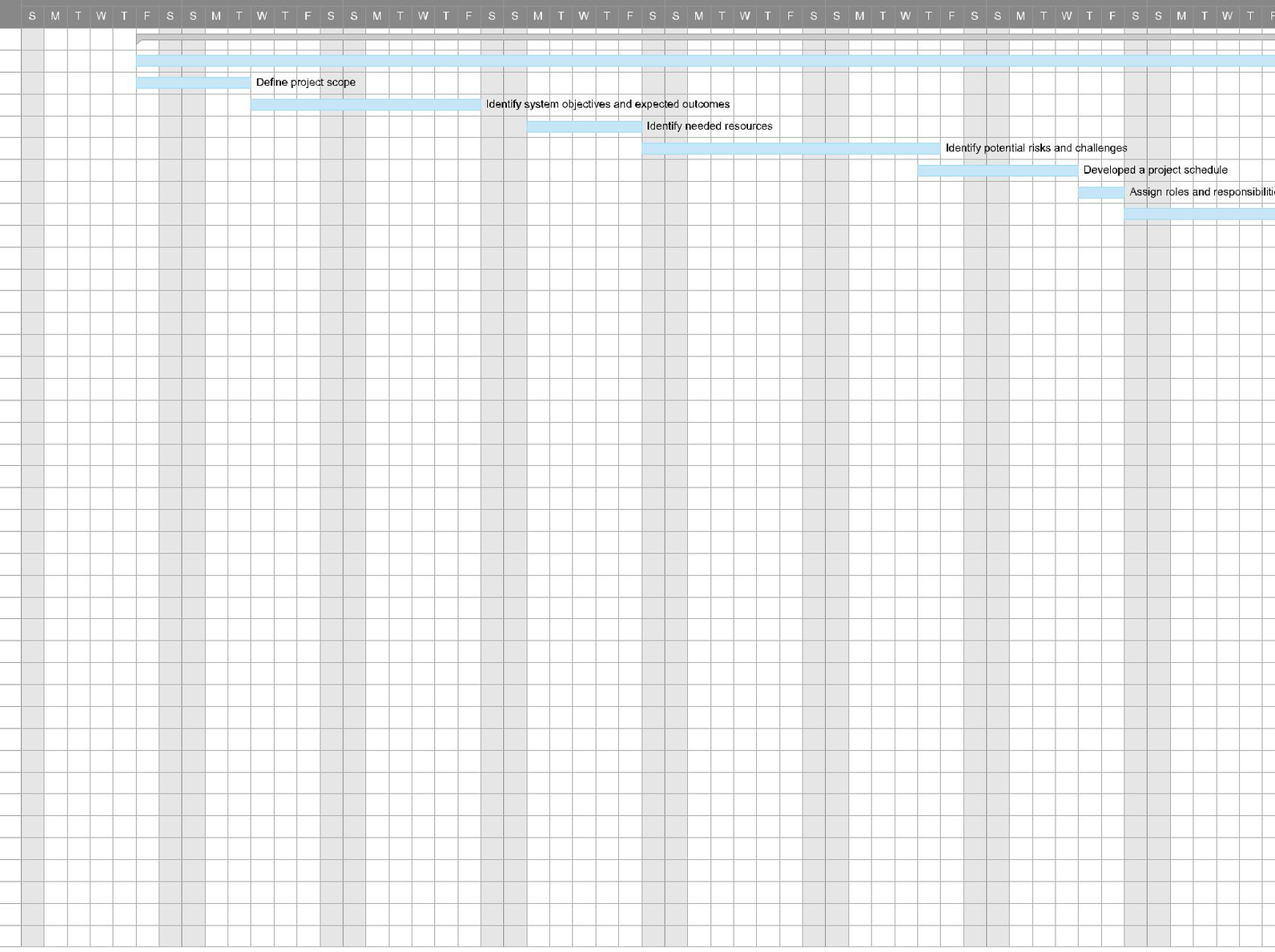
Requirements Planning: The Requirements Planning phase began with stakeholder consultations to identify key challenges in Labalan Funeral Homes' operations, such as unorganized data, passive tracking, and inefficient scheduling. Through iterative prototyping, core functionalities including client management, real-time scheduling, inventory tracking, and financial reporting were defined. User feedback was continuously gathered to refine requirements, ensuring the system addressed pain points like manual record-keeping and miscommunication. Early prototypes were tested for feasibility, aligning the project scope with ISO/IEC 25010 standards for functionality, usability, and efficiency.

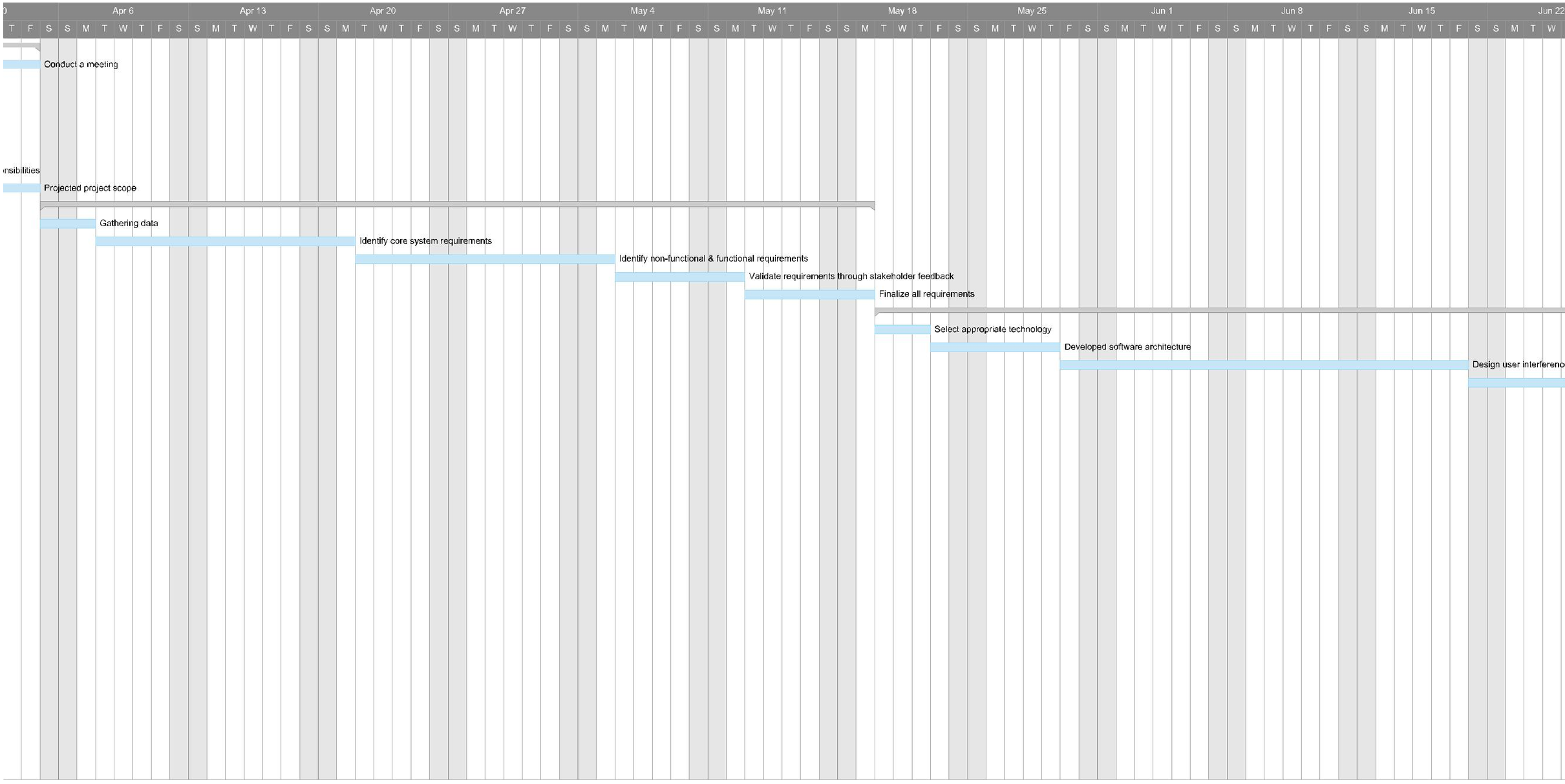
User Design: In the User Design phase, the development team created interactive wireframes and mock-ups for the web-based platform, prioritizing an intuitive interface for funeral home staff. Key modules such as client profiles, service scheduling dashboards, and inventory management were designed with input from end-users to ensure ease of navigation. The design incorporated real- time data visualization for scheduling and reporting, while MySQL was selected for centralized data storage. Continuous feedback loops allowed for rapid adjustments, ensuring the final design met usability goals and streamlined daily operations.

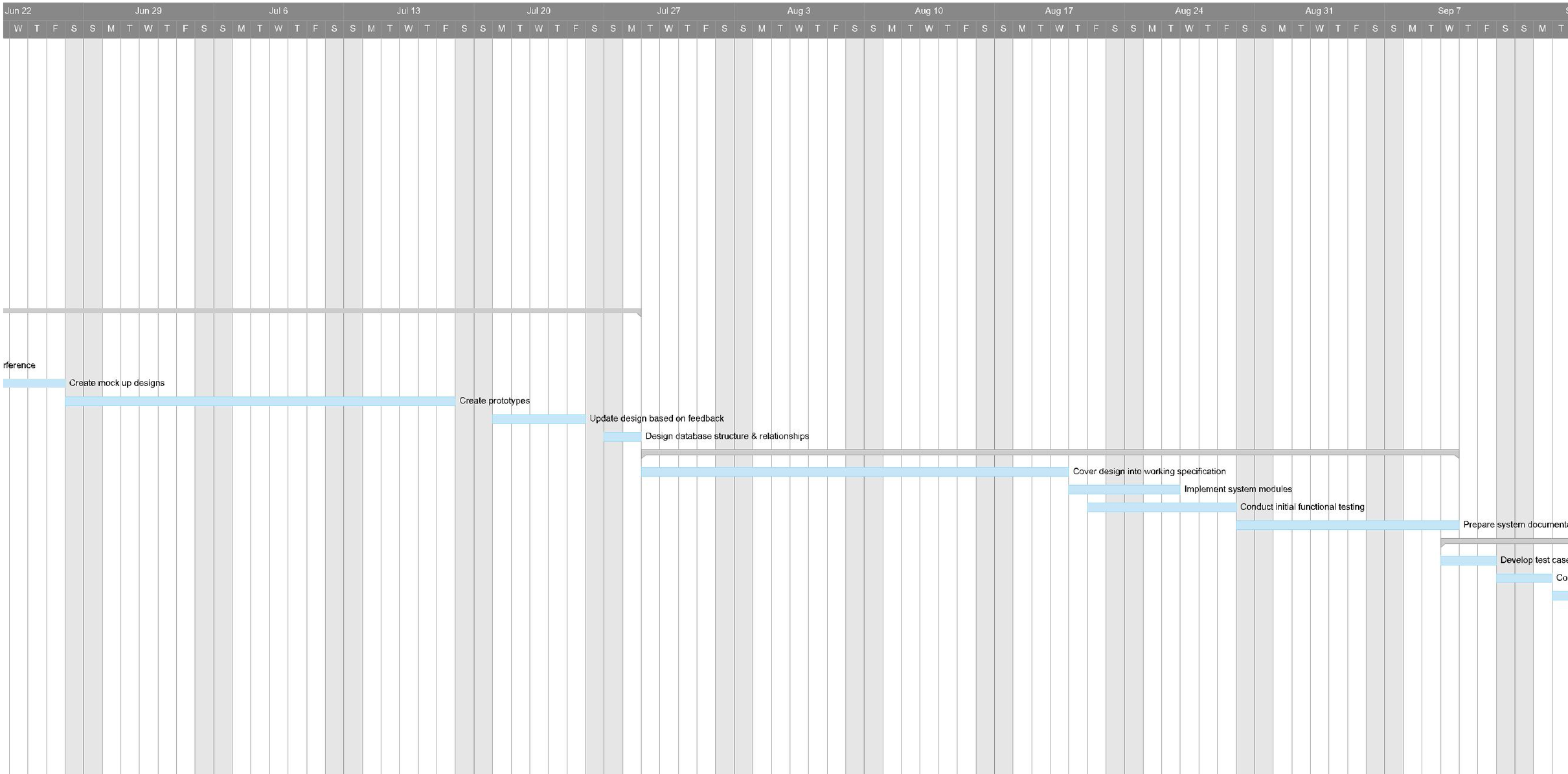
Construction: During the Construction phase, the system was built iteratively using WordPress for the front-end and MySQL for back-end data management. Developers focused on modular components, such as automated booking, inventory alerts, and financial tracking, integrating each feature incrementally. Rigorous testing was conducted at each stage to validate performance against ISO/IEC 25010 criteria, with fixes applied in real time. By the end of this phase, a fully functional prototype was ready, featuring seamless coordination between client records, scheduling, and inventory modules.

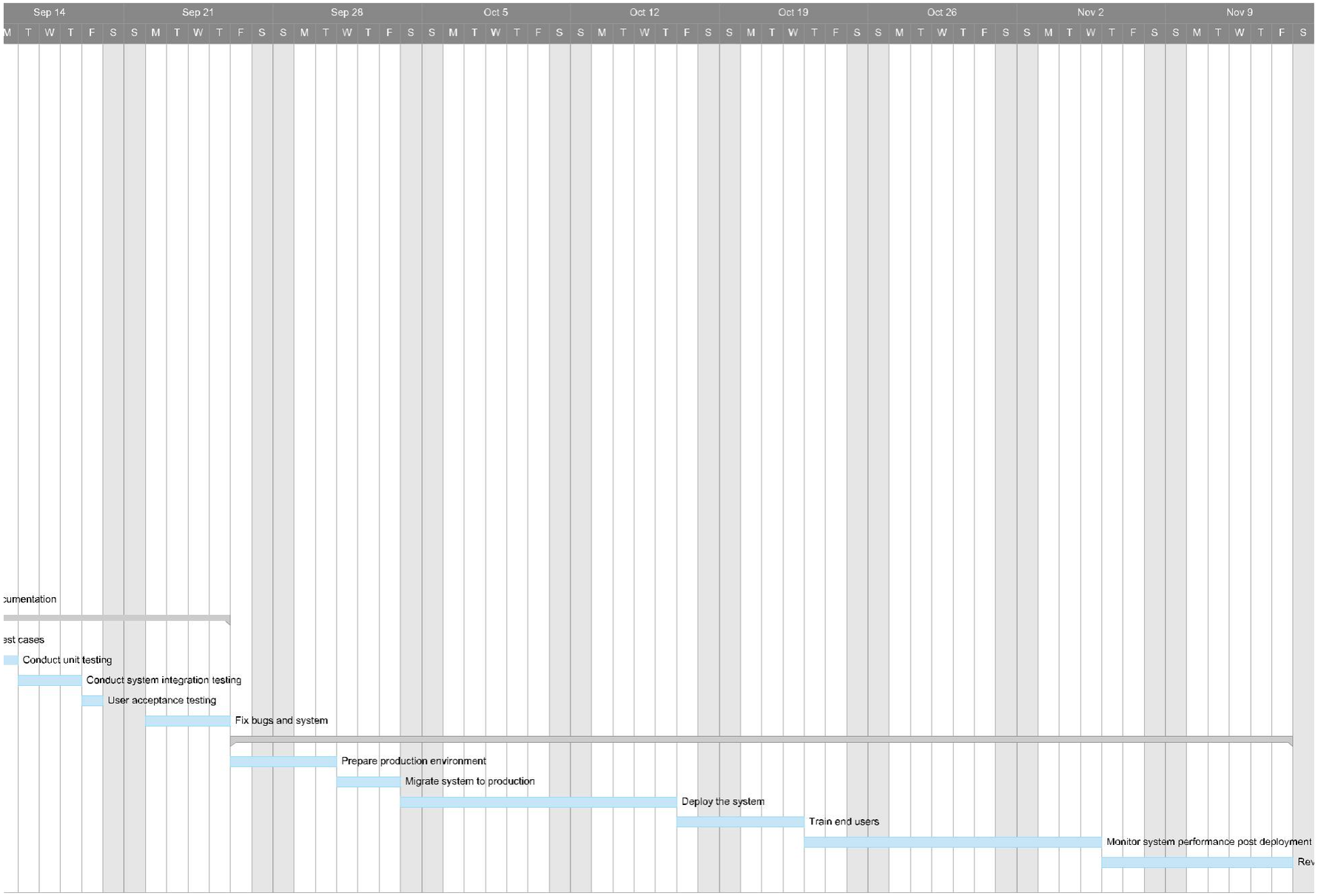
Cutover: The Cutover phase involved deploying the system to Labalan Funeral Homes' operations, including data migration from legacy records and staff training. A phased roll-out minimized disruption, with the team monitoring system stability and user adoption. Post-launch support addressed immediate feedback, ensuring smooth transitions in scheduling, inventory updates, and reporting. The final system delivered a centralized, efficient platform that eliminated manual errors and improved service coordination, fully aligning with the project's objectives.

Task ID	Task Description	Start Date	Finish
Planning Phase		02/14/25	04/04/25
T1	Conduct a meeting	02/14/25	04/04/25
T2	Define project scope	02/14/25	02/18/25
T3	Identify system objectives and expected outcomes	02/19/25	02/28/25
T4	Identify needed resources	03/03/25	03/07/25
T5	Identify potential risks and challenges	03/08/25	03/20/25
T6	Developed a project schedule	03/20/25	03/26/25
T7	Assign roles and responsibilities	03/27/25	03/28/25
T8	Projected project scope	03/29/25	04/04/25
Requirements Analysis Phase		04/05/25	05/19/25
T9	Gathering data	04/05/25	04/07/25
T10	Identify core system requirements	04/08/25	04/21/25
T11	Identify non-functional & functional requirements	04/22/25	05/05/25
T12	Validate requirements through stakeholder feedback	05/06/25	05/12/25
T13	Finalize all requirements	05/13/25	05/19/25
System Design		05/20/25	07/28/25
T14	Select appropriate technology	05/20/25	05/22/25
T15	Developed software architecture	05/23/25	05/29/25
T16	Design user interference	05/30/25	06/20/25
T17	Create mock up designs	06/21/25	06/27/25
T18	Create prototypes	06/28/25	07/18/25
T19	Update design based on feedback	07/21/25	07/25/25
T20	Design database structure & relationships	07/27/25	07/28/25
Implementation Phase		07/29/25	09/10/25
T21	Cover design into working specification	07/29/25	08/20/25
T22	Implement system modules	08/21/25	08/26/25
T23	Conduct initial functional testing	08/22/25	08/29/25
T24	Prepare system documentation	08/30/25	09/10/25
Testing Phase		09/10/25	09/25/25
T25	Develop test cases	09/10/25	09/12/25
T26	Conduct unit testing	09/13/25	09/15/25
T27	Conduct system integration testing	09/16/25	09/18/25
T28	User acceptance testing	09/19/25	09/19/25
T29	Fix bugs and system	09/22/25	09/25/25
Deployment, Maintenance, & Support Phase		09/26/25	11/14/25
T30	Prepare production environment	09/26/25	09/30/25
T31	Migrate system to production	10/01/25	10/03/25
T32	Deploy the system	10/04/25	10/16/25
T33	Train end users	10/17/25	10/22/25
T34	Monitor system performance post deployment	10/23/25	11/05/25
T35	Review system performance	11/06/25	11/14/25
T36	Plan for future updates & maintenance		




Apr 6
Apr 13
Apr 20
Apr 27
May 4
May 11
May 18
May 25
Jun 1
Jun 8
Jun 15
Conduct a meeting
nsibilities
Projected project scope
Gathering data
Identify core system requirements
Identify non-functional & functional requirements
Validate requirements through stakeholder feedback
Finalize all requirements
Select appropriate technology
Developed software architecture
Design user interferenc


Jun 22
Jun 29
Jul 6
Jul 13
Jul 20
Jul 27
Aug 3
Aug 10
Aug 17
Aug 24
Aug 31
Sep 7
rference
Create mock up designs
Create prototypes
Update design based on feedback
Design database structure & relationships
Cover design into working specification
Implement system modules
Conduct initial functional testing
Prepare system documenta
Develop test case
Co


Sep 14
Sep 21
Sep 28
Oct 5
Oct 12
Oct 19
Oct 26
Nov 2
Nov 9
M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S
cumentation
est cases
Conduct unit testing
Conduct system integration testing
User acceptance testing
Fix bugs and system
Prepare production environment
Migrate system to production
Deploy the system
Train end users
Monitor system performance post deployment
Rev