Assignment 2 – Project Management

Department *of Computing, Letterkenny Institute of Technology*

Table of Contents

Introduction

The assignment given by the lecturer required the student to compose a project and implement the project within Microsoft Project (MSP). The implementation within MSP includes a Gantt Chart, project calendar, assigned resources, leveled resources, critical path and tracking progress. The chosen project is based on a software implementation for a cinema ticket booking app that is negotiated with an external client. The project is set to start at the 27th of April 2020.

Work Breakdown Structure (WBS)

The WBS is split into 6 categories: (1) Project Planning, (2) Requirement Analysis, (3) UI/UX Design, (4) Implementation, (5) Testing and Legal and (6) Deployment and Integration. These phases are based on a more modern SDLC that keep the customer highly engaged for the duration of the project. An overview of the project can be seen in Figure 1.

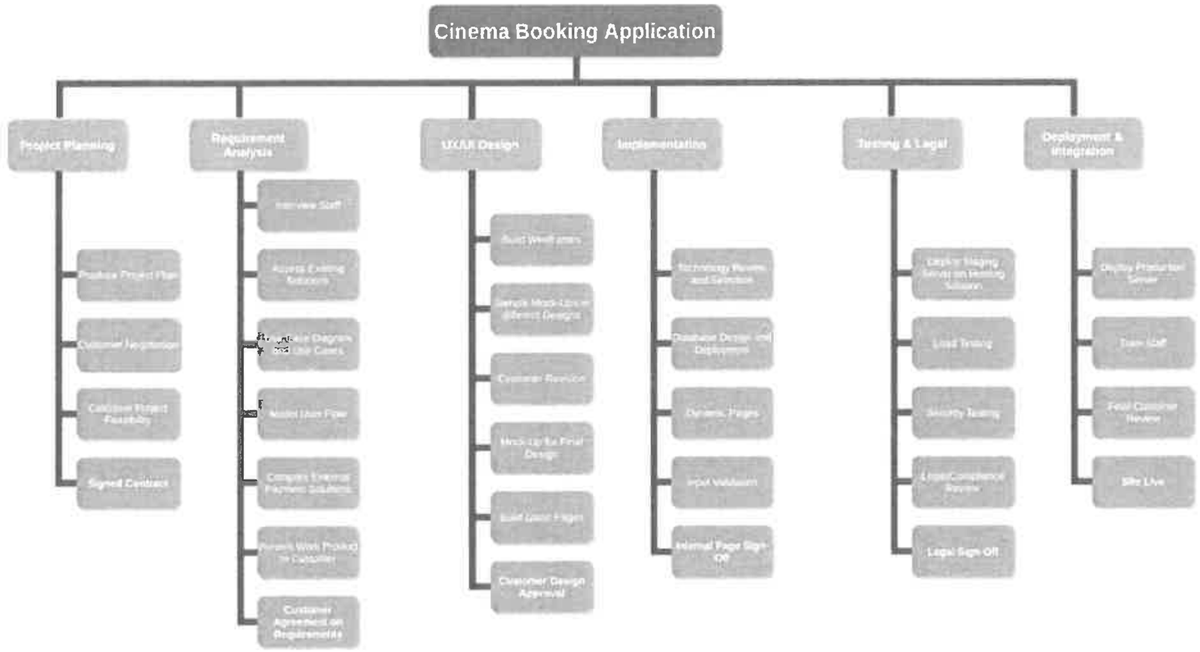


Figure 1: The work breakdown structure for the cinema ticket booking application.

Gantt Chart

The Gantt Chart shown in Figure 2 is an implementation of the WBS shown in Figure 1. The chart includes ***lag time***, for example 3.6 and 3.7. Another objective included is ***lead time*** between 4.3 and 4.4. ***Parallel tasks*** are also included as seen between 2.1 and 2.2 or 6.2 and 6.3. The ***summary tasks*** are clearly visible, relating to the WBS categories describes earlier. Every phase ends with a ***milestones*** that has a date next to it, as seen below. To further explain parallel tasks, lead and lag time, a more in-depth description will be given below:

(1) Parallel Tasks

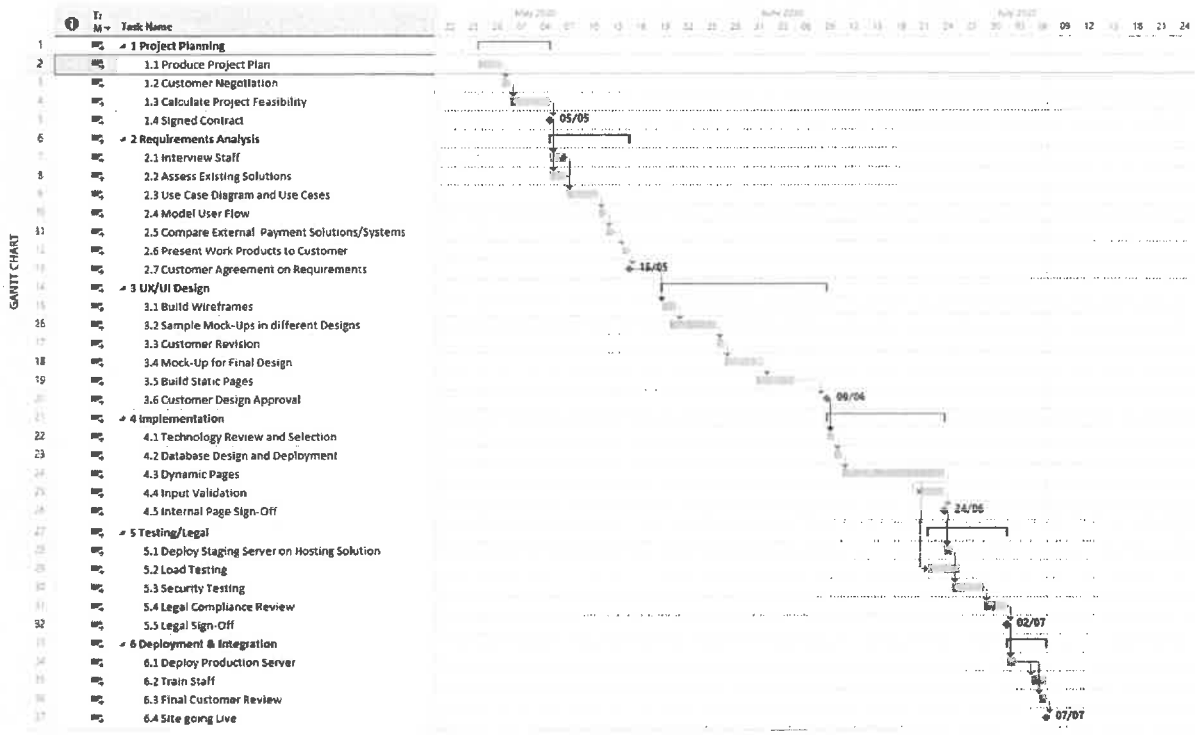
Tasks that can run simultaneously are called parallel tasks. During the requirement analysis phase, for example, the tasks *'Interviewing Staff'* and *'Assess Existing Solutions'* can run parallel as they do not require each other's completion.

(2) Lag Time

Some tasks require a certain wait period before the next task can start. One example here is *'Build Wireframes'* which starts after *'Customer Agreement on Requirements'*. Let's imagine the following scenario: The product manager meets with the customer but the customer doesn't agree with the presented work, but the UI/UX team is supposed to start the next day with the Wireframing – then these resources would potentially be wasted for 1-2 days as the work can't start without customer agreement. It might make sense to build in a natural buffer that allows the project some room from the beginning on, hence the included lag time.

(3) Lead Time

The opposite of lag time is lead time. Lead time means that a task starts before another tasks is fully completed. One example is *'Input Validation'* starting a few days before *'Dynamic Pages'* are fully done. The reason is that some of the dynamic pages need to be finished before input validation can start, while the goal is to let both finish at the same time to achieve an efficient project plan.



Task Name

1 Project Planning

1.1 Produce Project Plan

1.2 Customer Negotiation

1.3 Calculate Project Feasibility

1.4 Signed Contract

2 Requirements Analysis

2.1 Interview Staff

2.2 Assess Existing Solutions

2.3 Use Case Diagram and Use Ceses

2.4 Model User Flow

2.5 Compare External Payment Solutions/Systems

2.6 Present Work Products to Customer

2.7 Customer Agreement on Requirements

3 UX/UI Design

3.1 Build Wireframes

3.2 Sample Mock-Ups in different Designs

3.3 Customer Revision

3.4 Mock-Up for Final Design

3.5 Build Static Pages

3.6 Customer Design Approval

4 Implementation

4.1 Technology Review and Selection

4.2 Database Design and Deployment

4.3 Dynamic Pages

4.4 Input Validation

4.5 Internal Page Sign-Off

5 Testing/Legal

5.1 Deploy Staging Server on Hosting Solution

5.2 Load Testing

5.3 Security Testing

5.4 Legal Compliance Review

5.5 Legal Sign-Off

6 Deployment & Integration

6.1 Deploy Production Server

6.2 Train Staff

6.3 Final Customer Review

6.4 Site going Live

Figure 2: Basic Gantt Chart for the application.

Over-allocated Resources

The next step required the student to assign resources to the project and over-allocate the resources. Over-allocation means that one resource is, as per plan, planned to work on more than 1 tasks at the same time. It is essentially a scheduling conflict for resources. Figure 3 shows multiple of these scheduling conflicts with red-figures next to the individual tasks.

The following resources were used for the project:

1. ***Product Owner***

   Is the link between the customer and the developers. The PO takes care of all issues that the team might encounter, administrative and technical. In our example the PO is simply used to schedule communication with the customer.

2. ***Sales Consultant***

   The SC is used exclusively for the customer negotiation phase and is responsible for selling the product and helping the PO to calculate project profitability.

3. ***Senior Developer***

   The Sr. Developer is a more experienced that can assist the Junior Developer as a point of contact. Since Senior Developers are more expensive, their time is mainly used for critical tasks or tasks that require to be finished on time to allow the project to finish on time.

4. ***Junior Developer***

   A Jr. Developer is less experienced than the Sr. counterpart. The Jr. Developer needs more time to finish tasks but is significantly cheaper than a Sr. Developer.

5. ***Senior UI/UX Designer***

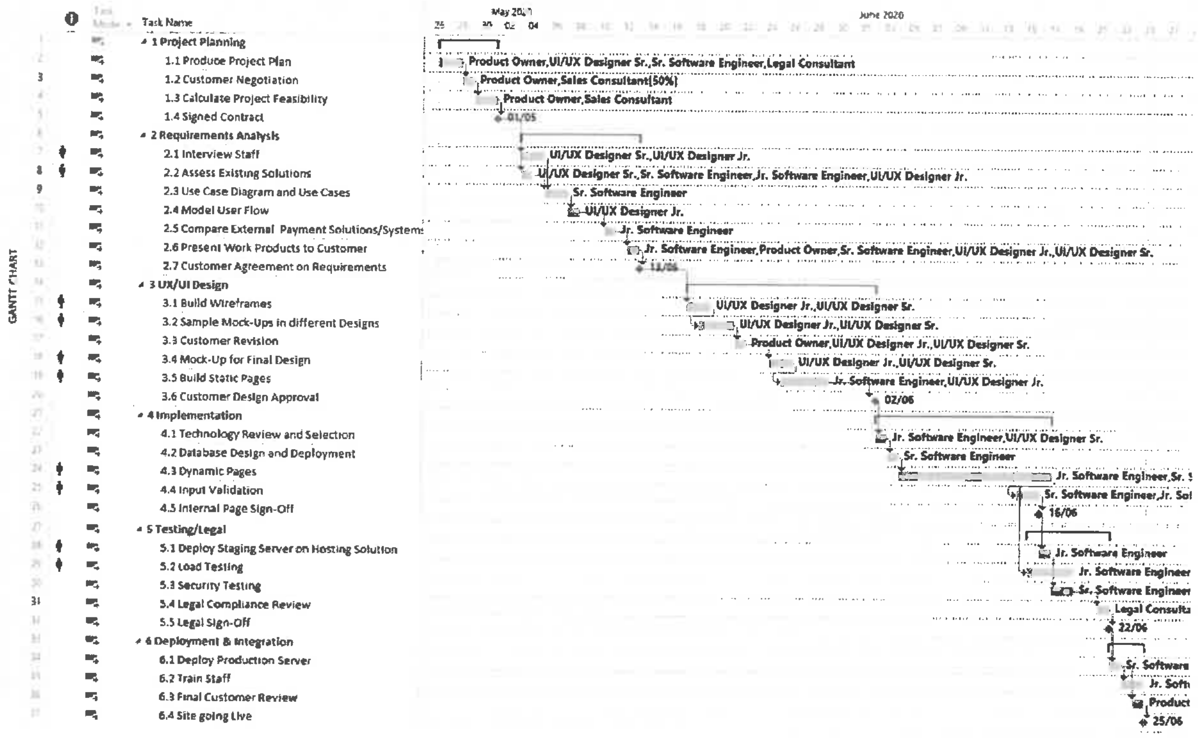
   For a cinema ticket booking application the user experience is of the utmost importance. The use of an experienced UI/UX Designer ensures that the products looks good and is usable for the customer. The same pros and cons apply to the Sr. Designer that apply to the Sr. Developer.

6. ***Junior UI/UX Designer***

   Equivalent to the Jr. Developer but for UI/UX.

Calendars

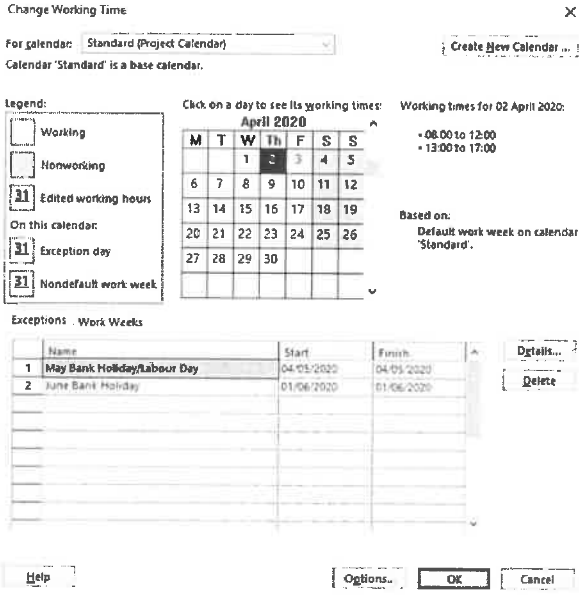
Every country, company and department has different work hours and off days. The calendar allows to account for those differences. Due to the project schedule, 2 Irish public holidays were included in the calendar, as seen in Figure 4. Those days will be left out when the software calculates the project – hence the project will be extended for 2 days.



	Task Name
1	1 Project Planning
2	1.1 Produce Project Plan
3	1.2 Customer Negotiation
4	1.3 Calculate Project Feasibility
5	1.4 Signed Contract
6	2 Requirements Analysis
7	2.1 Interview Staff
8	2.2 Assess Existing Solutions
9	2.3 Use Case Diagram and Use Cases
10	2.4 Model User Flow
11	2.5 Compare External Payment Solutions/Systems
12	2.6 Present Work Products to Customer
13	2.7 Customer Agreement on Requirements
14	3 UX/UI Design
15	3.1 Build Wireframes
16	3.2 Sample Mock-Ups in different Designs
17	3.3 Customer Revision
18	3.4 Mock-Up for Final Design
19	3.5 Build Static Pages
20	3.6 Customer Design Approval
21	4 Implementation
22	4.1 Technology Review and Selection
23	4.2 Database Design and Deployment
24	4.3 Dynamic Pages
25	4.4 Input Validation
26	4.5 Internal Page Sign-Off
27	5 Testing/Legal
28	5.1 Deploy Staging Server on Hosting Solution
29	5.2 Load Testing
30	5.3 Security Testing
31	5.4 Legal Compliance Review
32	5.5 Legal Sign-Off
33	6 Deployment & Integration
34	6.1 Deploy Production Server
35	6.2 Train Staff
36	6.3 Final Customer Review
37	6.4 Site going Live

GANTT CHART
Task Mode
May 2020
June 2020
Product Owner,UI/UX Designer Sr.,Sr. Software Engineer,Legal Consultant
Product Owner,Sales Consultant[50%]
Product Owner,Sales Consultant
01/05
UI/UX Designer Sr.,UI/UX Designer Jr.
UI/UX Designer Sr.,Sr. Software Engineer,Jr. Software Engineer,UI/UX Designer Jr.
Sr. Software Engineer
UI/UX Designer Jr.
Jr. Software Engineer
Jr. Software Engineer,Product Owner,Sr. Software Engineer,UI/UX Designer Jr.,UI/UX Designer Sr.
11/05
UI/UX Designer Jr.,UI/UX Designer Sr.
UI/UX Designer Jr.,UI/UX Designer Sr.
Product Owner,UI/UX Designer Jr.,UI/UX Designer Sr.
UI/UX Designer Jr.,UI/UX Designer Sr.
Jr. Software Engineer,UI/UX Designer Jr.
02/06
Jr. Software Engineer,UI/UX Designer Sr.
Sr. Software Engineer
Jr. Software Engineer,Sr. S
Sr. Software Engineer,Jr. Sof
16/06
Jr. Software Engineer
Jr. Software Engineer
Sr. Software Engineer
Legal Consulta
22/06
Sr. Software
Jr. Softw
Product
25/06

Figure 3: Gantt Chant with added Conflicting Resources



Change Working Time

For calendar: Standard (Project Calendar)

Calendar 'Standard' is a base calendar.

Legend:

- Working
- Nonworking
- 31 Edited working hours

On this calendar:

- 31 Exception day
- 31 Nondefault work week

Click on a day to see its working times:

April 2020

M	T	W	Th	F	S	S
		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			

Working times for 02 April 2020:

- 08:00 to 12:00
- 13:00 to 17:00

Based on:

Default work week on calendar 'Standard'.

Exceptions Work Weeks

	Name	Start	Finish
1	May Bank Holiday/Labour Day	04/05/2020	04/05/2020
2	June Bank Holiday	01/06/2020	01/06/2020

Figure 4: Calendar entries for the Project

Levelled Resources

The resources section assigned designated resources to the project; these resources included scheduling conflicts though. To fix this, a variety of techniques were used which will be looked at in detail below.

(1) Increased maximum of resource units

The assumption for the project was that the made-up software company is a small independent contractor. Increasing resources for a project is difficult for a small business like this, as the initial project plan should account and allocate all necessary resources. I used a second Jr. Developer for the implementation of the dynamic pages. In the resource sheet a new developer was created, alternatively it would have been possible to adjust the Jr. Developers availability to 200%. This would assume though, that the second developer would be available for the entire project, I think it made more sense to create a separate resource in this specific example.

(2) Assignment of alternative resource

This is the preferred way, as it doesn't lead to an extension of the project or higher costs – as long as a cheaper resource is assigned or an equally expensive. This was utilized for multiple tasks. Since Jr. Developers are significantly cheaper than their Sr. counterparts, it was ensured to allocate as many hours as possible to such.

(3) Adjustment of task relationship

This was used at one occasion, for the 'Input Validation'. A decision was made to reduce the lead time for the task so that the projects length is not affected. Due to the nature of the project it was not possible to further adjust task relationships.

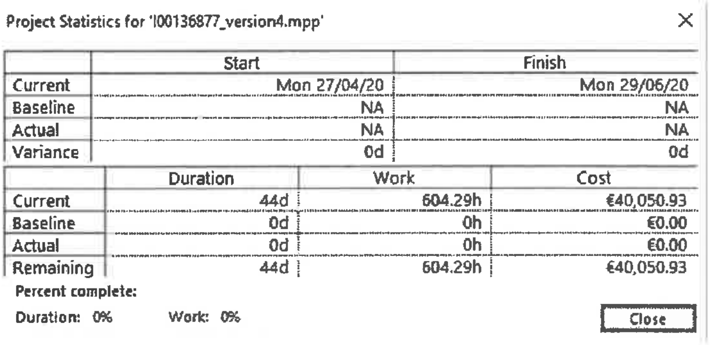
(4) Allowing Overtime

This was used for the 'peak' only. The peak occurred during the implementation of the dynamic pages, here the Jr. Developers were allowed to work 10% overtime. This was achieved by allowing the resource overtime in the task interface.

(5) Extending working hours

The regular work hours were adjusted from 08:00 – 17:00 to 08:00 – 17:30, making the work week 2.5h longer, seen in Figure 7. This is the standard in many German companies and should be reasonable due to the high compensation of employees. It is worth mentioning, that employees are not compensated by worked hours. The calculated cost is only used to calculate project profitability and internal accounting.

The project time was cut by 4.5 days by making these adjustments, as seen in Figure 5 and 6. Another benefit is the reduced project cost of around 3.000 Euro. The reduced project cost is a result of the maximized hours for Jr. Developers and Jr. Designers.



Project Statistics for '100136877_version4.mpp'

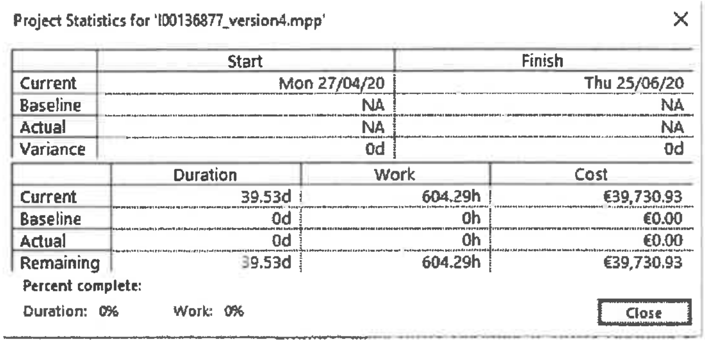
	Start	Finish
Current	Mon 27/04/20	Mon 29/06/20
Baseline	NA	NA
Actual	NA	NA
Variance	0d	0d

	Duration	Work	Cost
Current	44d	604.29h	€40,050.93
Baseline	0d	0h	€0.00
Actual	0d	0h	€0.00
Remaining	44d	604.29h	€40,050.93

Percent complete:

Duration: 0% Work: 0%

Figure 5: Project Statistics Overview.



Project Statistics for '100136877_version4.mpp'

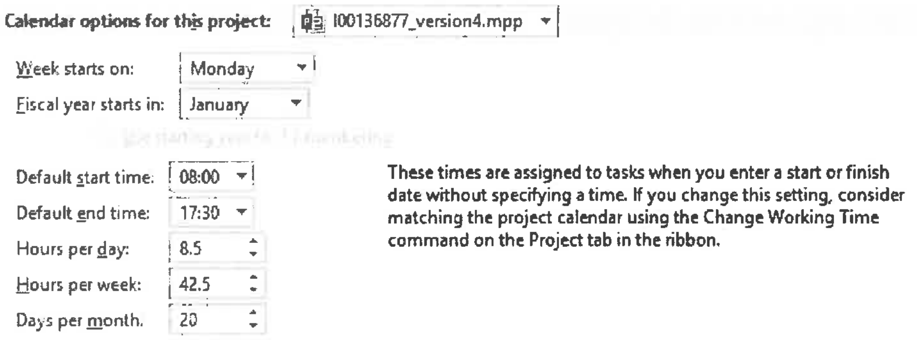
	Start	Finish
Current	Mon 27/04/20	Thu 25/06/20
Baseline	NA	NA
Actual	NA	NA
Variance	0d	0d

	Duration	Work	Cost
Current	39.53d	604.29h	€39,730.93
Baseline	0d	0h	€0.00
Actual	0d	0h	€0.00
Remaining	39.53d	604.29h	€39,730.93

Percent complete:

Duration: 0% Work: 0%

Figure 6: Project Cost after adjustments being made.



Calendar options for this project: I00136877_version4.mpp

Week starts on: Monday

Fiscal year starts in: January

Default start time: 08:00

Default end time: 17:30

Hours per day: 8.5

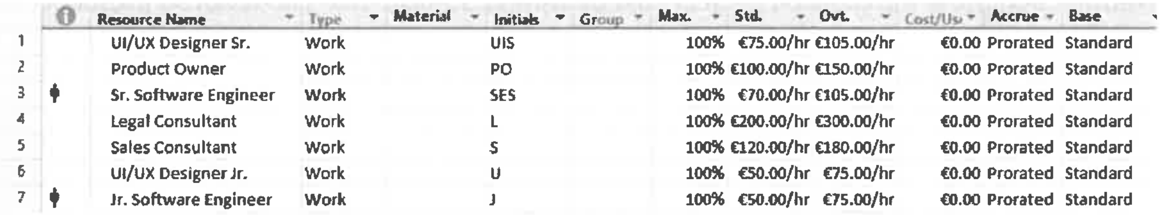
Hours per week: 42.5

Days per month: 20

These times are assigned to tasks when you enter a start or finish date without specifying a time. If you change this setting, consider matching the project calendar using the Change Working Time command on the Project tab in the ribbon.

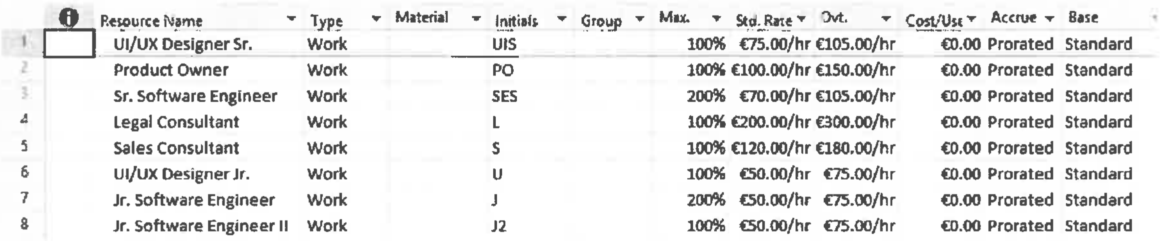
Figure 7: Adjusted Work Hours for the Project.

Figure 8 and 9 show the added resource and adjusted maximum availability. The difference of cost between Junior and Senior employees is noteworthy, as this difference was leveraged to achieve a significantly lower project cost.



	ⓘ	Resource Name	Type	Material	Initials	Group	Max.	Std.	Ovt.	Cost/Use	Accrue	Base
1		UI/UX Designer Sr.	Work		UIS		100%	€75.00/hr	€105.00/hr	€0.00	Prorated	Standard
2		Product Owner	Work		PO		100%	€100.00/hr	€150.00/hr	€0.00	Prorated	Standard
3	♦	Sr. Software Engineer	Work		SES		100%	€70.00/hr	€105.00/hr	€0.00	Prorated	Standard
4		Legal Consultant	Work		L		100%	€200.00/hr	€300.00/hr	€0.00	Prorated	Standard
5		Sales Consultant	Work		S		100%	€120.00/hr	€180.00/hr	€0.00	Prorated	Standard
6		UI/UX Designer Jr.	Work		U		100%	€50.00/hr	€75.00/hr	€0.00	Prorated	Standard
7	♦	Jr. Software Engineer	Work		J		100%	€50.00/hr	€75.00/hr	€0.00	Prorated	Standard

Figure 8: Project Resources and their associated cost before leveling.



	ⓘ	Resource Name	Type	Material	Initials	Group	Max.	Std. Rate	Ovt.	Cost/Use	Accrue	Base
1		UI/UX Designer Sr.	Work		UIS		100%	€75.00/hr	€105.00/hr	€0.00	Prorated	Standard
2		Product Owner	Work		PO		100%	€100.00/hr	€150.00/hr	€0.00	Prorated	Standard
3		Sr. Software Engineer	Work		SES		200%	€70.00/hr	€105.00/hr	€0.00	Prorated	Standard
4		Legal Consultant	Work		L		100%	€200.00/hr	€300.00/hr	€0.00	Prorated	Standard
5		Sales Consultant	Work		S		100%	€120.00/hr	€180.00/hr	€0.00	Prorated	Standard
6		UI/UX Designer Jr.	Work		U		100%	€50.00/hr	€75.00/hr	€0.00	Prorated	Standard
7		Jr. Software Engineer	Work		J		200%	€50.00/hr	€75.00/hr	€0.00	Prorated	Standard
8		Jr. Software Engineer II	Work		J2		100%	€50.00/hr	€75.00/hr	€0.00	Prorated	Standard

Figure 9: Project Resources and their associated cost after leveling.

The final Gantt Chart after leveling the resources is shown in Figure 10. When compared to Figure 2 it becomes clear, that how much more efficient the project was made by leveling the resources.

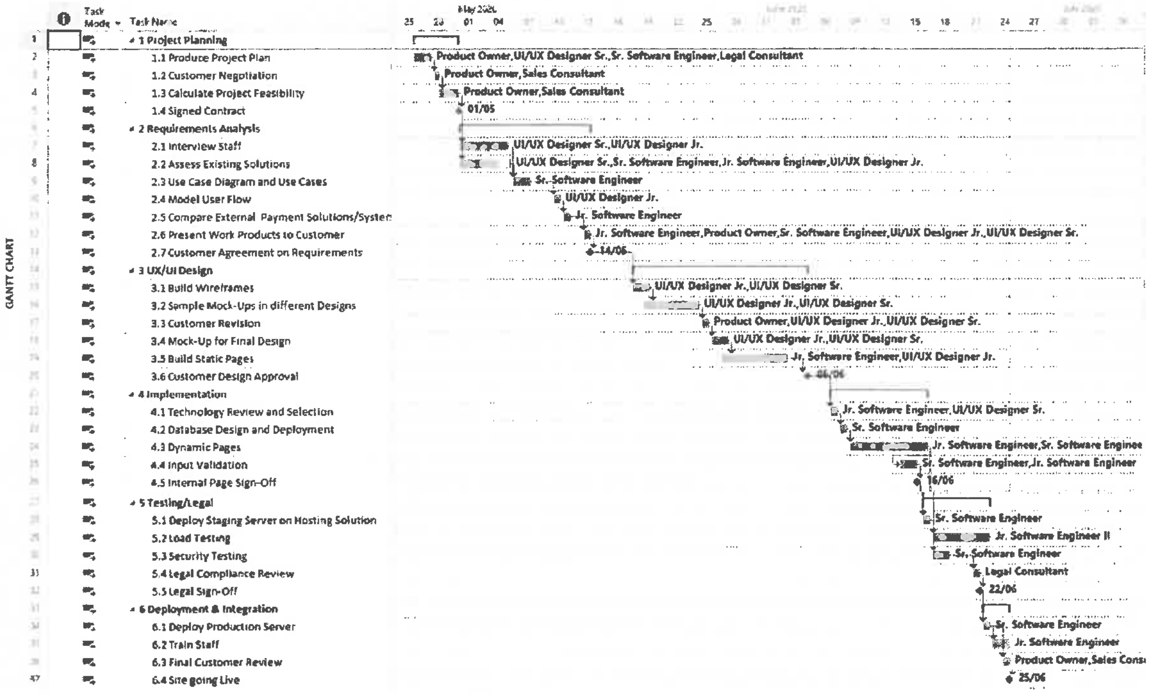


Figure 10: Gantt Chart after leveling the resources.

Critical Path

The critical path is the time it takes to finish a project, ergo the longest path. The project started out by finishing at 25.06.2020 and was optimized by shortening the critical path to finish at 19.06.2020. This is nearly a full week that was saved by simply optimizing the critical path.

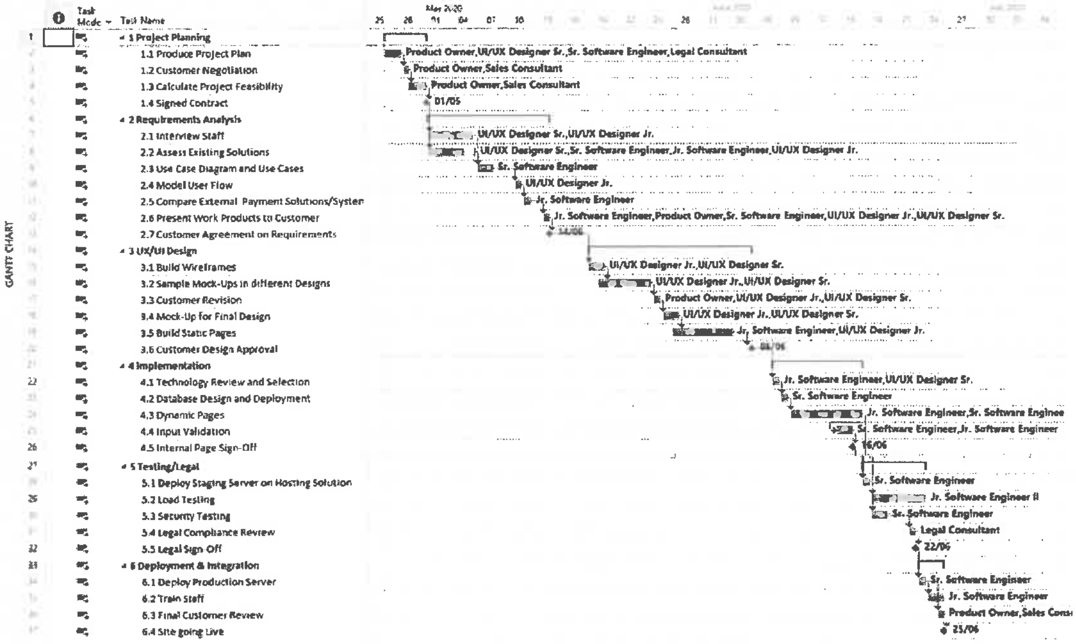


Figure 11: The Critical Path drawn for the Project before optimization.

The same optimization methods as for leveling resources can be utilized to shorten the critical path.

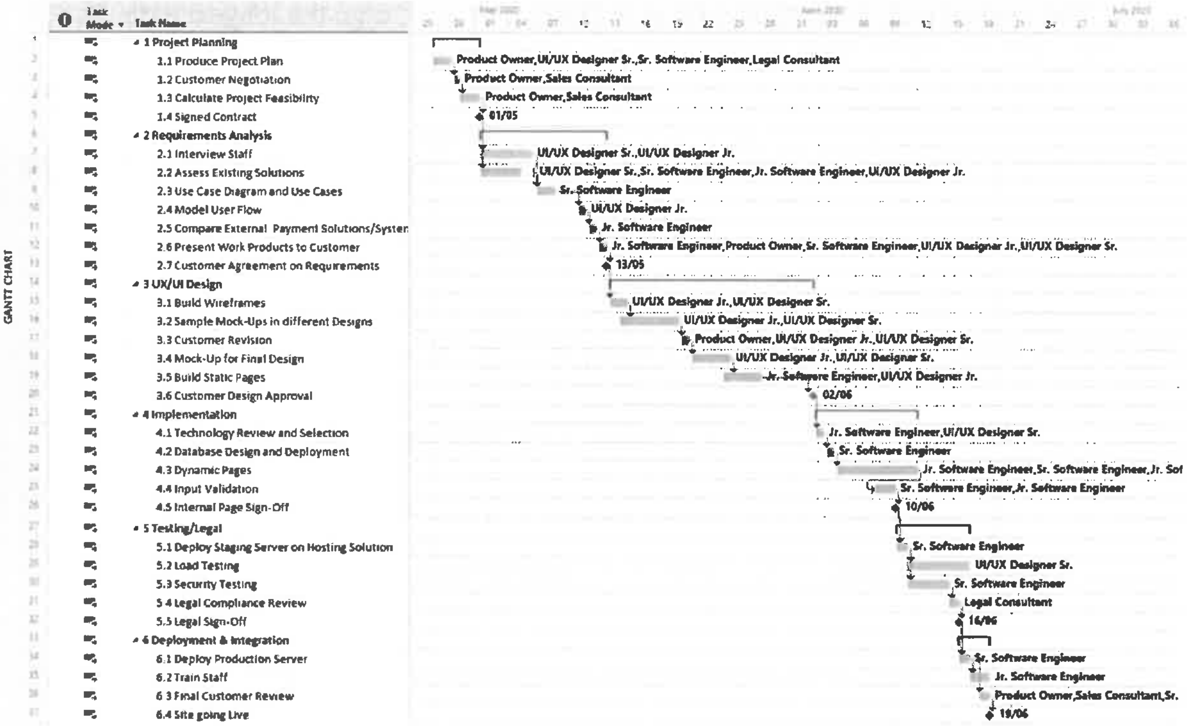
(1) Allocating more resources,

(2) Allowing overtime for critical tasks,

(3) Restructuring task relationships and

(4) Assigning Alternative Resources.

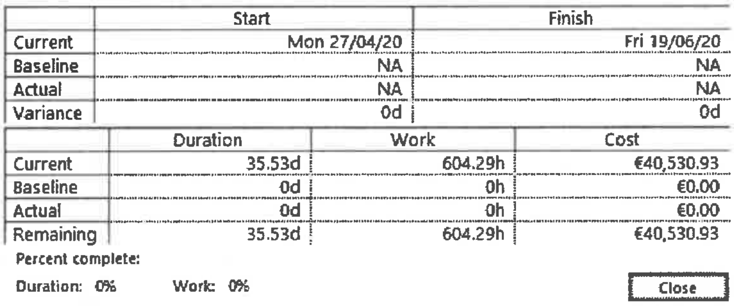
A decision was made that shortening the critical path but increasing cost was not an option. Instead adjustments regarding lag and lead time were heavily utilized. Especially during the design and implementation phase, several tasks were adjusted by adding a day of lead time as it should be possible to run most tasks earlier with enough resources available on short notice. The load testing was also reassigned to the Senior UI/UX Designer. The final Gantt Chart for the critical path is shown in Figure 12.



Task Name	
1 Project Planning	
1.1 Produce Project Plan	Product Owner,UI/UX Designer Sr.,Sr. Software Engineer,Legal Consultant
1.2 Customer Negotiation	Product Owner,Sales Consultant
1.3 Calculate Project Feasibility	Product Owner,Sales Consultant
1.4 Signed Contract	01/05
2 Requirements Analysis	
2.1 Interview Staff	UI/UX Designer Sr.,UI/UX Designer Jr.
2.2 Assess Existing Solutions	UI/UX Designer Sr.,Sr. Software Engineer,Jr. Software Engineer,UI/UX Designer Jr.
2.3 Use Case Diagram and Use Cases	Sr. Software Engineer
2.4 Model User Flow	UI/UX Designer Jr.
2.5 Compare External Payment Solutions/Systen	Jr. Software Engineer
2.6 Present Work Products to Customer	Jr. Software Engineer,Product Owner,Sr. Software Engineer,UI/UX Designer Jr.,UI/UX Designer Sr.
2.7 Customer Agreement on Requirements	13/05
3 UX/UI Design	
3.1 Build Wireframes	UI/UX Designer Jr.,UI/UX Designer Sr.
3.2 Sample Mock-Ups in different Designs	UI/UX Designer Jr.,UI/UX Designer Sr.
3.3 Customer Revision	Product Owner,UI/UX Designer Jr.,UI/UX Designer Sr.
3.4 Mock-Up for Final Design	UI/UX Designer Jr.,UI/UX Designer Sr.
3.5 Build Static Pages	Jr. Software Engineer,UI/UX Designer Jr.
3.6 Customer Design Approval	02/06
4 Implementation	
4.1 Technology Review and Selection	Jr. Software Engineer,UI/UX Designer Sr.
4.2 Database Design and Deployment	Sr. Software Engineer
4.3 Dynamic Pages	Jr. Software Engineer,Sr. Software Engineer,Jr. Sof
4.4 Input Validation	Sr. Software Engineer,Jr. Software Engineer
4.5 Internal Page Sign-Off	10/06
5 Testing/Legal	
5.1 Deploy Staging Server on Hosting Solution	Sr. Software Engineer
5.2 Load Testing	UI/UX Designer Sr.
5.3 Security Testing	Sr. Software Engineer
5.4 Legal Compliance Review	Legal Consultant
5.5 Legal Sign-Off	16/06
6 Deployment & Integration	
6.1 Deploy Production Server	Sr. Software Engineer
6.2 Train Staff	Jr. Software Engineer
6.3 Final Customer Review	Product Owner,Sales Consultant,Sr.
6.4 Site going Live	19/06

Figure 12: The Critical Path for the Project after optimization.

The final project overview is shown in Figure 13. It is important to note that the cost did go up slightly due to the reassignment of the *'Load Testing'* task. All the other adjustments were made by adjusting lag and lead time as well as allocating resources that are priced the same, for example 2 Jr. Developers instead of 1.



	Start	Finish
Current	Mon 27/04/20	Fri 19/06/20
Baseline	NA	NA
Actual	NA	NA
Variance	0d	0d

	Duration	Work	Cost
Current	35.53d	604.29h	€40,530.93
Baseline	0d	0h	€0.00
Actual	0d	0h	€0.00
Remaining	35.53d	604.29h	€40,530.93

Percent complete:

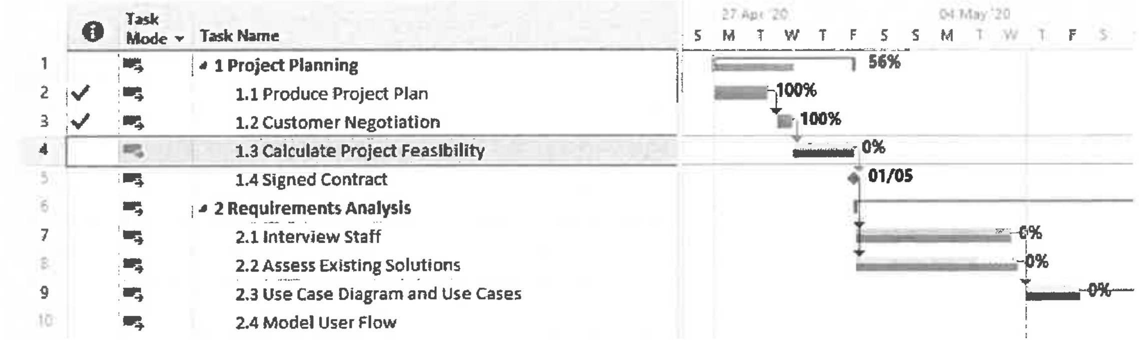
Duration: 0% Work: 0%

Figure 13: The Project Overview after applying optimization.

Tracking Progress

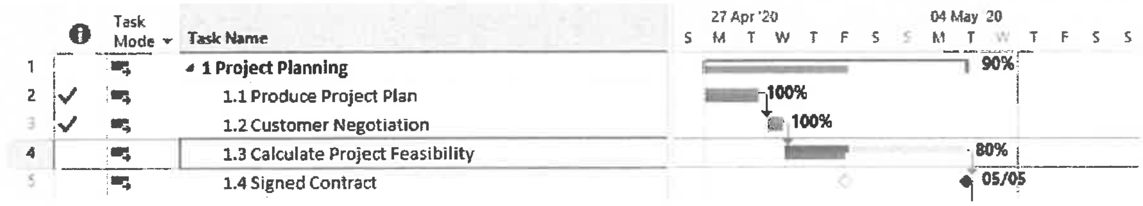
A project plan is of course not a static product but a plan that will change as the project progresses. It is possible to track the project by setting a baseline (planned timeline) and then adjusting the actual project flow accordingly. Figure 14 shows the use of a baseline and tasks completed according to plan.

For Figure 15, the task *'Calculate Project Feasibility'* was delayed as a family emergency for the Product Owner forced the task to be put on hold. This means that the baseline task is now being overrun. This is visually shown by the red line running over the grey bar.



	ℹ	Task Mode	Task Name
1			1 Project Planning
2	✓		1.1 Produce Project Plan
3	✓		1.2 Customer Negotiation
4			1.3 Calculate Project Feasibility
5			1.4 Signed Contract
6			2 Requirements Analysis
7			2.1 Interview Staff
8			2.2 Assess Existing Solutions
9			2.3 Use Case Diagram and Use Cases
10			2.4 Model User Flow

Figure 14: Task completion with Baseline on Target.



	ℹ	Task Mode	Task Name
1			1 Project Planning
2	✓		1.1 Produce Project Plan
3	✓		1.2 Customer Negotiation
4			1.3 Calculate Project Feasibility
5			1.4 Signed Contract

Figure 15: Task completion with Baseline being off Target.

Figure 16 shows the final Gantt Chart with the adjustment of the delayed task taken into consideration. It is worth mentioning that the delay is reflected in the project end, which is now the 22.06.2020. The previous project end was projected to be the 19th of June. The project summary, as seen in Figure 17, reflects these changes exactly. From this we can also derive that the extended task was a critical path task, since its extension led to an extension of the entire project. The added cost is of course a reflection of task 1.3 running over time and budget.

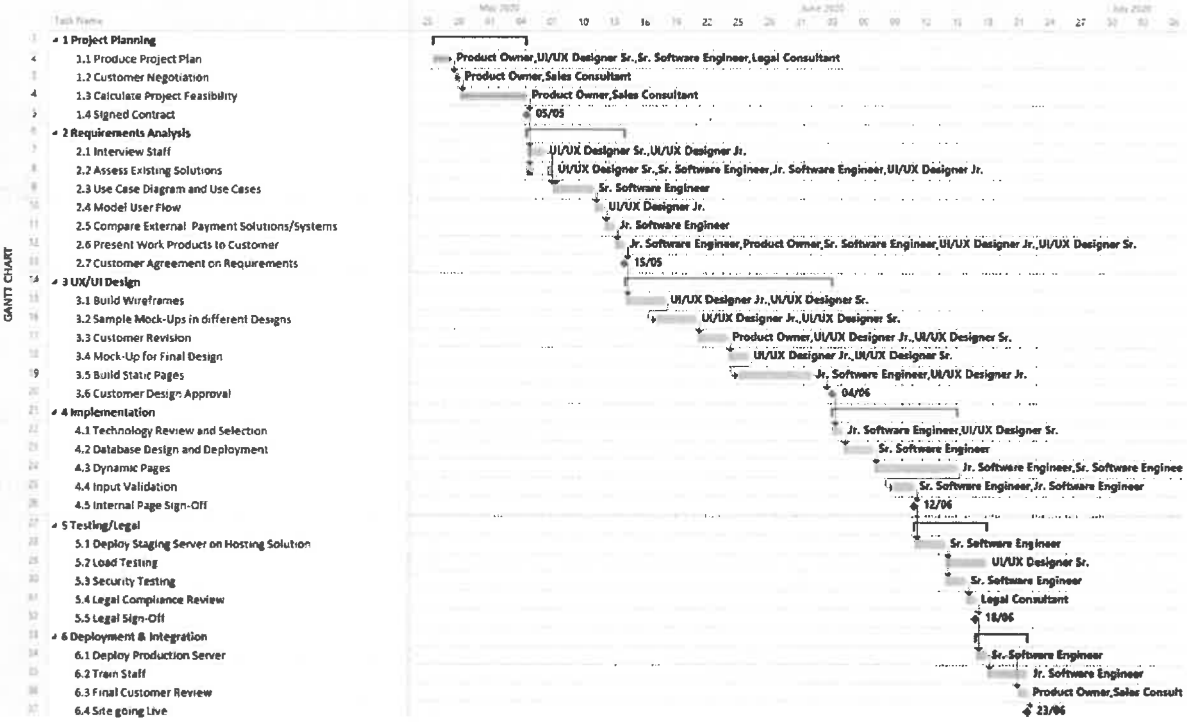
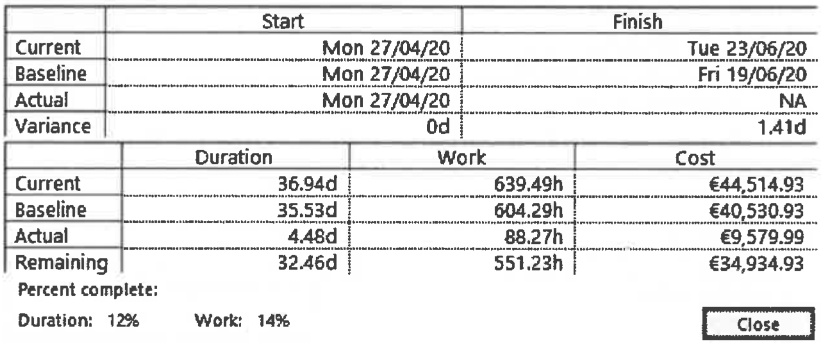


Figure 16: The Project running over because Task 2.2 is delayed.



	Start	Finish
Current	Mon 27/04/20	Tue 23/06/20
Baseline	Mon 27/04/20	Fri 19/06/20
Actual	Mon 27/04/20	NA
Variance	0d	1.41d

	Duration	Work	Cost
Current	36.94d	639.49h	€44,514.93
Baseline	35.53d	604.29h	€40,530.93
Actual	4.48d	88.27h	€9,579.99
Remaining	32.46d	551.23h	€34,934.93

Percent complete:

Duration: 12% Work: 14%

Figure 17: Project Overrunning and added Costs.