

Project Plan

🕒 Created	@October 21, 2021 6:33 PM
🕒 Last Edited Time	@November 5, 2021 8:43 PM
▼ Type	Technical Spec
▼ Status	
👤 Created By	
👤 Last Edited By	
👥 Stakeholders	

BLG 411E - Software Engineering

05.11.2021

PROJECT PLAN

ILC

Furkan Hayta -

Ahmet Polat -

Utku Sabri Kaya -

Buğra Aydın -

GROUP - 8

Table of Contents

BLG 411E - Software Engineering

PROJECT PLAN

GROUP - 8

ILC

Table of Contents

1. Introduction

1.1 Scope of the Project and Document

1.2 Deliverables

1.3 Functional Requirements \ Tasks

1.4 Non-functional Issues

2. Project Plan

3. ESTIMATES

4. Resources

5. Schedule

6. Risk

1. Introduction

Inconsistency License Checker (ILC) is a web app that checks the license compatibility of the given project that is hosted on GitHub.


1.1 Scope of the Project and Document

ILC consists of three main layers: Parser, license checker and web app. Parser gets the license information about the given repository in GitHub webpage. License checker constructs the license graph of the given project and checks incompatibilities. And web app gets the repository address from the user then shows the license graph and reports the incompatibilities if any.

This document contains the general plan of the ILC project in the next 5 sections. Work breakdown structure (WBS) of the project is drawn and explained in the Project Plan section, efforts that are estimated for each task are listed in the Estimates section, team structure is defined in Resources section, timetable of the project is shown as Gantt and Pert charts in Schedule section, and in the risks section, possible risks, their possibilities and impacts are listed as a table.

1.2 Deliverables

Deliverables

<u>Aa</u> #	 Deliverable
<u>1</u>	License Consistency Table
<u>2</u>	UI Mockup

<u>Aa</u> #	☰ Deliverable
<u>3</u>	Design Document
<u>4</u>	System Reliability Report
<u>5</u>	Online Client

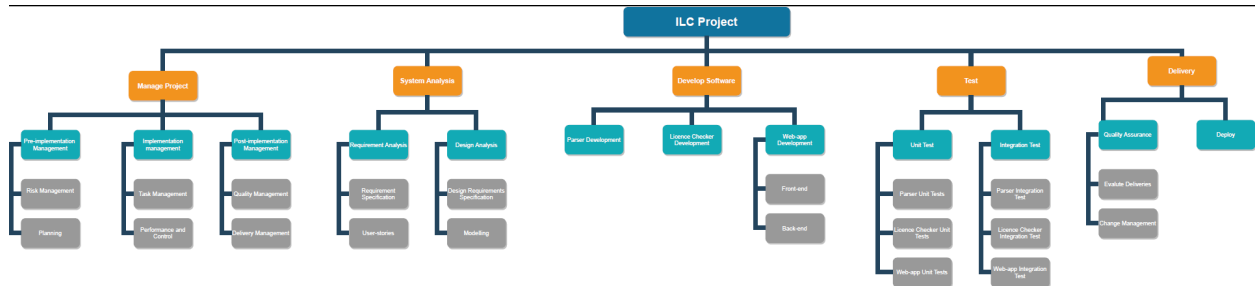
1.3 Functional Requirements \ Tasks

<u>Aa</u> #	☰ Tasks	# Deliverable #
<u>1</u>	License Compatibility List	1
<u>2</u>	Login Page Design	2
<u>3</u>	User Dashboard Design	2
<u>4</u>	Database Diagrams	3
<u>5</u>	Defining Milestones	3
<u>6</u>	Unit Tests	4
<u>7</u>	Integration Tests	4
<u>8</u>	Checker Implementation	5
<u>9</u>	Parser Implementation	5
<u>10</u>	Database Implementation	5
<u>11</u>	Web-app Frontend Implementation	5

1.4 Non-functional Issues

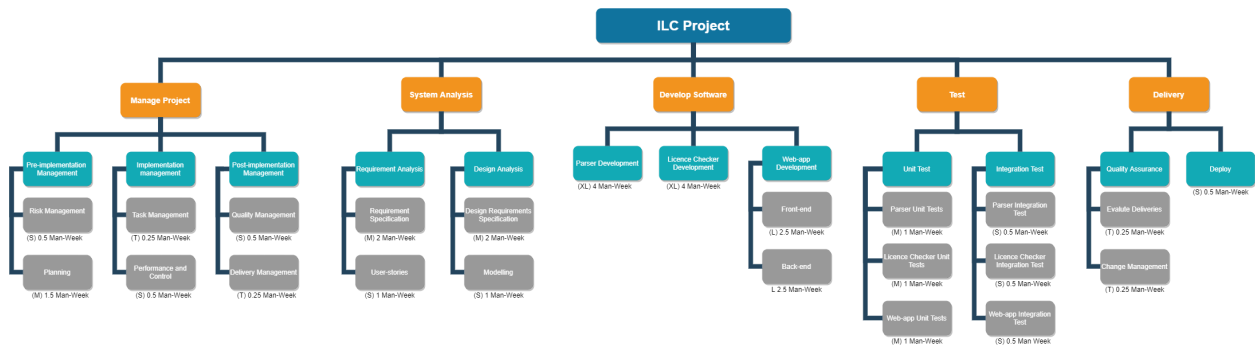
<u>Aa</u> #	☰ Issues
<u>1</u>	Checking should be done at most 5 sec.
<u>2</u>	For each level of the tree, different colors should be used.
<u>3</u>	Given Report should include the source of the license information of the incompatible libraries.

2. Project Plan

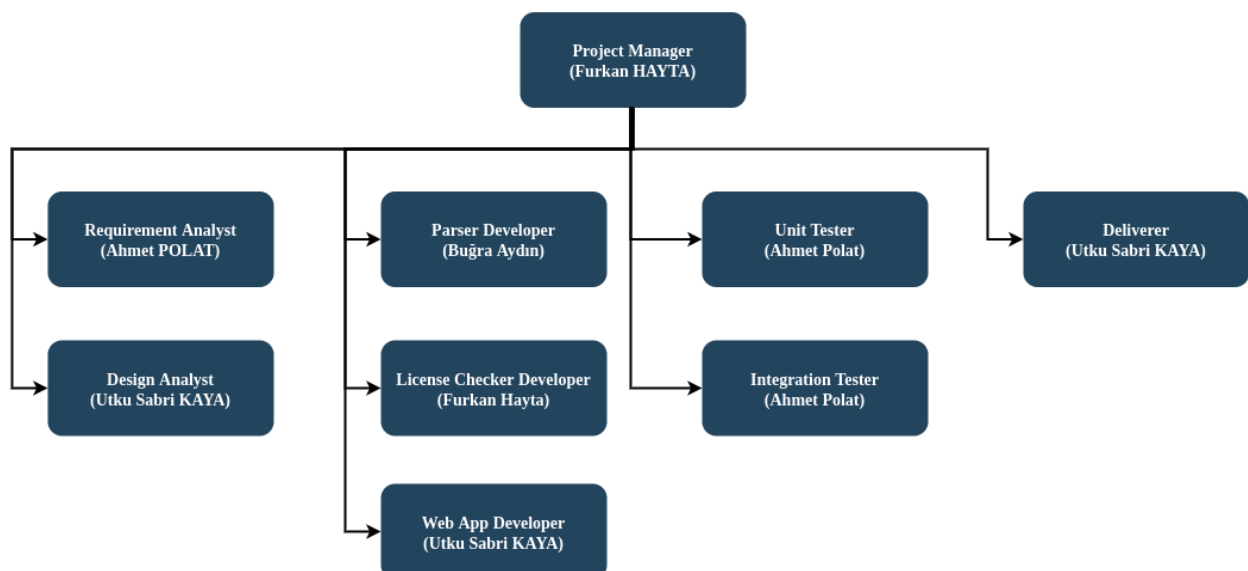


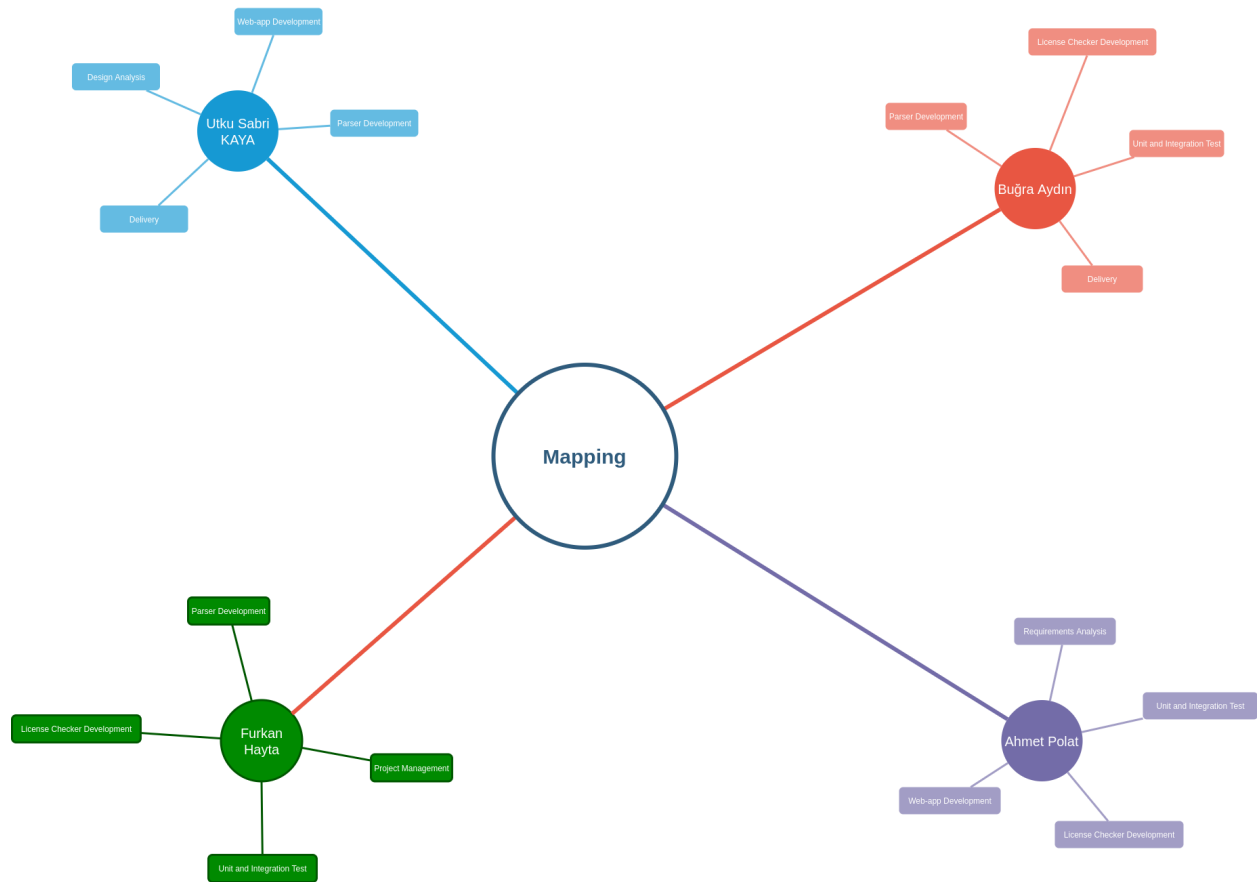
3. ESTIMATES

We used the Expert judgement method to estimate what needs to be done in the project plan. We distributed the tasks according to the competencies of the group employees and measured how long and with what quality they could complete the assigned task.

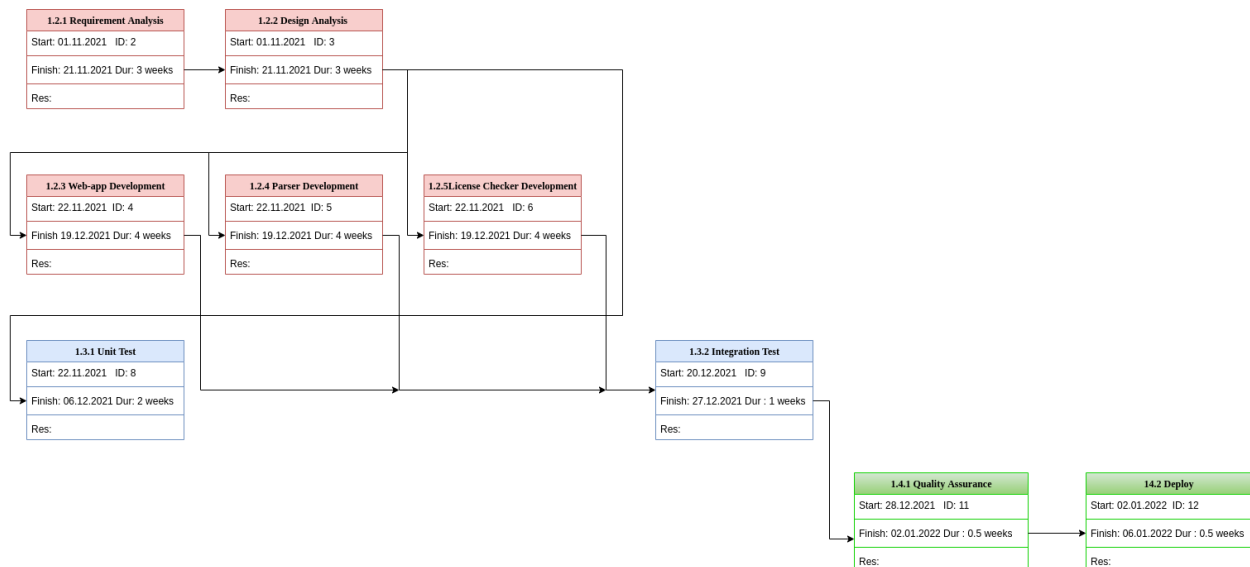


4. Resources



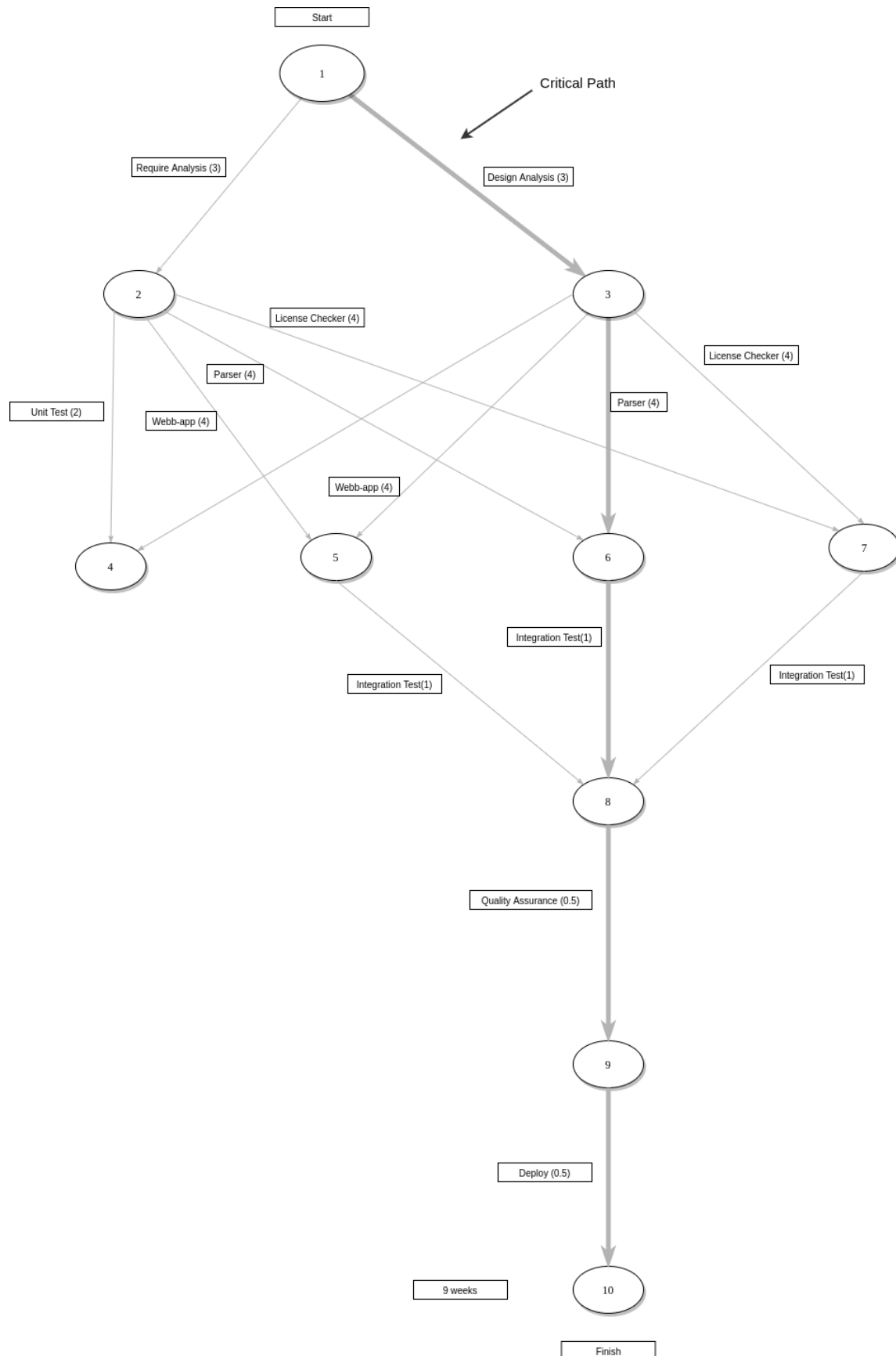


5. Schedule



Timeline

	November				December				January
	W1	W2	W3	W4	W1	W2	W3	W4	W1
Requirements Analysis									
Design Analysis									
Web-app Development									
Parser Development									
License Checker Development									
Unit Testing									
Integration Test									
Quality Assurance									
Deployment									






6. Risk

Risk impact probability chart

<u>Aa</u> Risk	<u>≡</u> Risk Type	<u>≡</u> Probability (%)	<u>≡</u> Impact
<u>High complexity in implementation</u>	Technical Risk	40	3
<u>Non-reusable components</u>	Technical Risk	50	2
<u>Insufficient resources</u>	Operational Risk	50	2
<u>Unsufficient database qualities</u>	Technical Risk	60	3
<u>Frequent project scope expansion</u>	Schedule Risk	60	2
<u>Incorrect Timing</u>	Schedule Risk	70	2

Risk Mitigation (Aversion) Table

<u>Aa</u> Risk	<u>≡</u> Affects	<u>≡</u> Description	<u>≡</u> Mitigation
<u>Incorrect Timing</u>	Project	The time required to develop the software is underestimated	Actively following the project implementation process by the manager and taking the necessary steps (Risk Reducing)
<u>Frequent project scope expansion</u>	Project	As the product is developed, the project is constantly expanded and the current development is wasted.	Establishing a well-defined (strict) scope that allows the project to be expanded (Risk Avoidance)
<u>Unsufficient database qualities</u> <u>Wrong database selection</u>	Product	The database used in the system cannot process as many transactions per second as expected.	Testing the database before the product is delivered and changing database (Risk Avoidance)
<u>Insufficient resources</u>	Product	Insufficient resources for up-to-date licenses and their compliance so that license checker can detect license incompatibility in multiple languages	Expanding the resources of the compliance status of the licenses during the project (Risk Reducing)

<u>Aa</u> Risk	 Affects	 Description	 Mitigation
<u>Non-reusable components</u>	Project	Software components intended for reuse contain defects and components that prevent reuse.	When noticed, updating the component to make reusable or making more than one component
<u>High complexity in implementation</u>	Project and Product	The project becomes difficult to update and the product runs slower than expected	Giving up some features or spending extra effort to make them more optimal (Risk Avoidance)