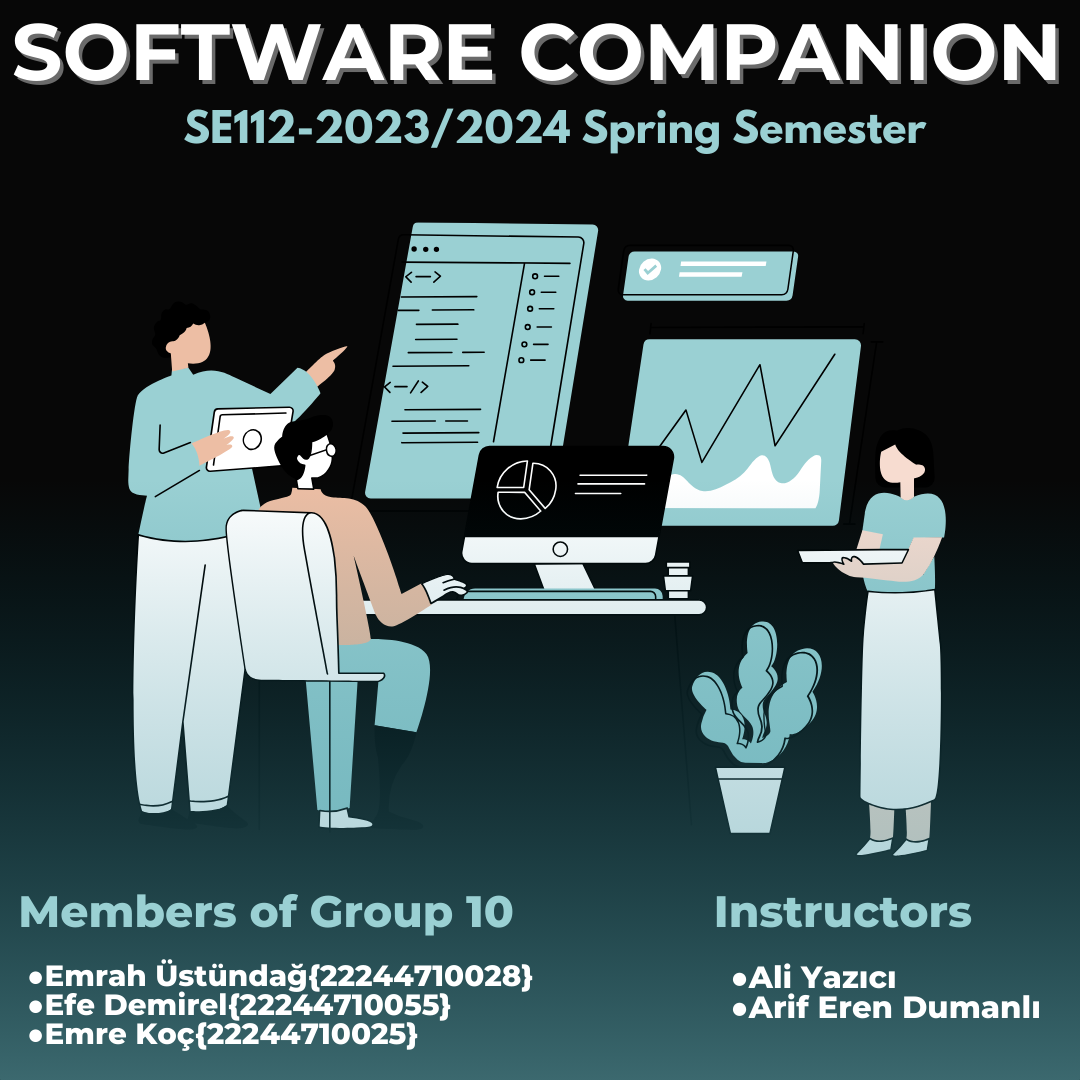
******

**CONTENTS**

[**SOFTWARE COMPANION** 3](#_Toc165591683)

[**Project Management** 4](#_Toc165591684)

[**Project Plan** 4](#_Toc165591685)

[**Gantt Diagram** 6](#_Toc165591686)

[**Risk Plan** 6](#_Toc165591687)

[**Requirements** 7](#_Toc165591688)

[**User Story** 8](#_Toc165591689)

[**Use-Case Diagram** 8](#_Toc165591690)

[**Plant UML Script** 9](#_Toc165591691)

[**Class Diagram** 10](#_Toc165591692)

[**Plant UML Script** 10](#_Toc165591693)

[**Sequence Diagram** 11](#_Toc165591694)

[**Steps:** 11](#_Toc165591696)

[**Activity Diagram** 12](#_Toc165591697)

[**Mock Ups** 13](#_Toc165591698)

# **SOFTWARE COMPANION**

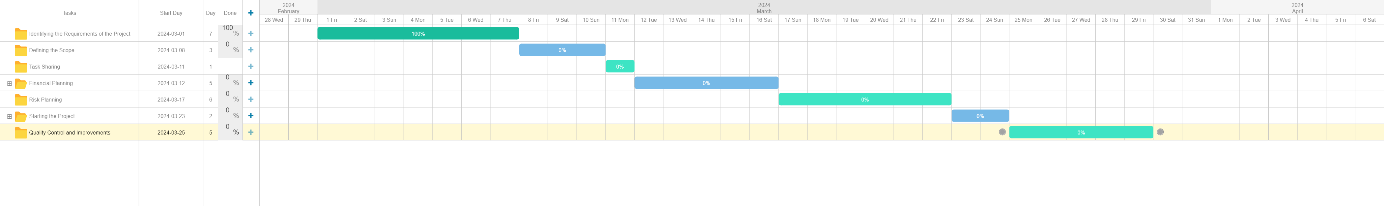
The aim of the project is to enable junior developers to benefit from the experience of senior developers and to integrate more easily into business life. Vision/What: Software Companion is a web-based application designed for the junior engineers to improve themselves. Users who want to learn the information/experiences that they cannot get from their environment can easily interact with people outside their own geography by using translation effectively thanks to the artificial intelligence integrated into the platform and reach the most appropriate information and guidance on the subject they want to get help on. This help is not just about code and algorithms, but it is more about experience (project management, human relations, finding the right resources, etc.). The same artificial intelligence also offers the user the chance to examine the solution methods and documents of people who have experienced similar problems and achieved solutions. The platform suggests the best method and people with experience to solve the user's problem. Depending on the user's preference, it offers further steps to the user. Artificial intelligence technology and payment methods will be integrated and used in this platform. This application will enable less experienced software developers to integrate into the sector more easily and quickly. Also, agile software development methodology will be used to develop this project.

# **Project Management**

## **Project Plan**

|  |  |  |
| --- | --- | --- |
| ***TASK*** | **TIME{DAYS}** | ***PREDECESSORS*** |
| A. Identifying the requirements of the project | 7 | - |
| B. Defining the scope | 4 | - |
| C. Task sharing | 1 | A |
| D. Financial planning | 5 | A,B |
| E. Risk planning | 6 | D |
| F. Starting the project | 2 | E |
| G. Quality control and improvements | 5 | F |

## **Gantt Diagram**



## **Risk Plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk Item** | **Impact** | **Action** | **Type** | **Risk Ratio** |
| **Some of the employees may want to quit during the project process** | **Increase in project completion time** | **If an employee quits, one of those waiting for a job in the reserve quota will be hired** | **Medium Risk** | **%7** |
| **Insufficient budget** | **Decline in motivation** | **Costs will be reduced by restricting some features** | **Low Risk** | **%3** |
| **Not enough time allocated** | **Too many complaints from customers** | **Since it is late, some campaigns can be made to appease customers** | **High Risk** | **%11** |

# **Requirements**

|  |  |  |
| --- | --- | --- |
| Functional Reqs ID | Explanation | MOSCOW Priority (M, S, C, W) |
| FR1 | The app will have a home page. | M |
| FR2 | There will be a registration and login screen. | M |
| FR3 | There will be a database where users' information will be stored. | M |
| FR4 | The app will be available 24/7. | S |
| FR5 | There will be an authentication system. | M |
| FR6 | There will be a panel where the user can ask their question. | M |
| FR7 | The application will include an algorithm that will find developers suitable for the question. | S |
| FR8 | There will be a chat panel where customers and developers can talk | M |
| FR9 | An AI-powered translation will be integrated into the chat panel. | C |
| FR10 | A panel will be created for the customer to pay. | S |
| Non-Functional Reqs ID | Explanation | MOSCOW Priority  (M, S, C, W) |
| NFR1 | Home page should open in maximum 3 seconds. | S |
| NFR2 | Minimum capacity should be 50.000 people. | S |
| NFR3 | User information should be stored with level 3 encryption standards. | C |

# **User Story**

**Software Companion** is used to help junior software engineers to solve their problems in business life. For example **Customers(user)** should be able to get information about their questions from **senior developers(user). Identify\_Provider** may **check** the authorization of the users and whether processes are right or not. All users (User/Customer,Developers) need to **login** to Software Companion before using it. Also, **Identify\_Provider** can control users’ information.

**Use-Case Diagram**

diyagram, çizgi, daire, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

## **Plant UML Script**

@startuml

class "Software Companion" {

+Name

+User\_Type

+User\_Name

+Password

}

class User {

+Name

+ID

+Question()

}

class "Chat Box" {

+Content

+Software\_Branch

+Language

}

class "Identity Provider" {

+Check\_Status()

}

class Customer {

+Department

}

class Developers {

+Experience

}

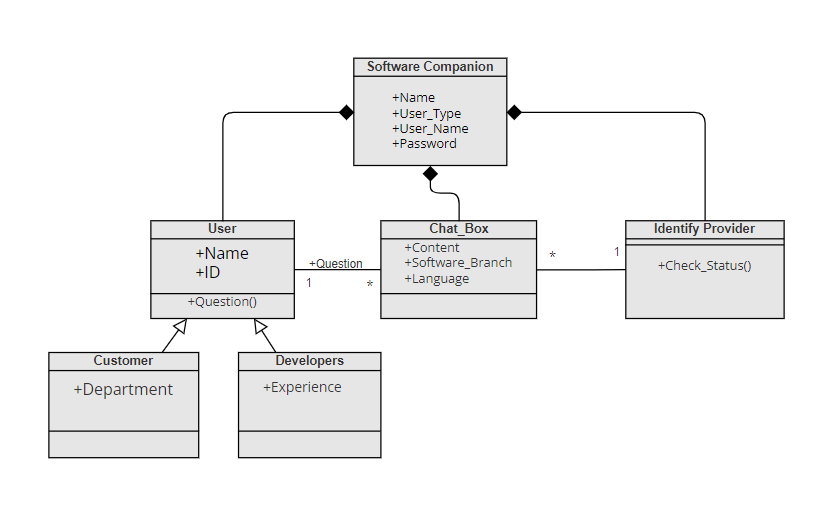
"Software Companion" – “User

“User” -- "Chat Box"

"Chat Box" -- "Identity Provider"

@enduml

# **Class Diagram**



## **Plant UML Script**

@startuml

class "Software Companion" {

+Name

+User\_Type

+User\_Name

+Password

}

class User {

+Name

+ID

+Question()

}

class "Chat Box" {

+Content

+Software\_Branch

+Language

}

class "Identity Provider" {

+Check\_Status()

}

class Customer {

+Department

}

class Developers {

+Experience

}

"Software Companion" – “User

“User” -- "Chat Box"

"Chat Box" -- "Identity Provider"

@enduml

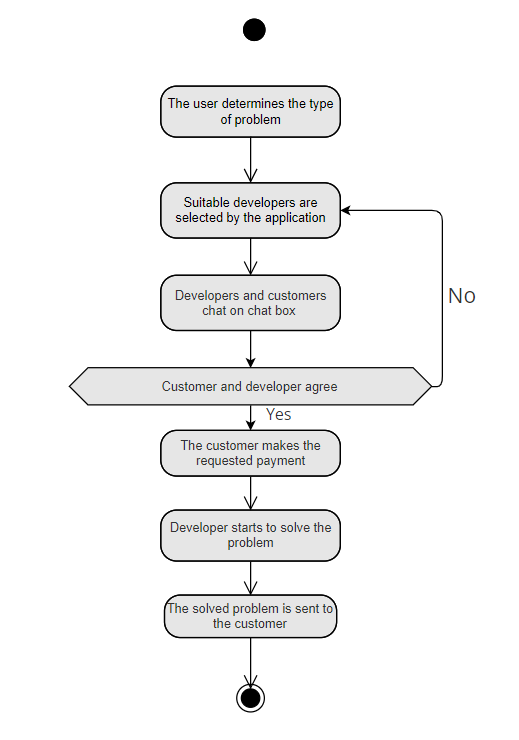
# **Sequence Diagram**

# metin, diyagram, plan, paralel içeren bir resim Açıklama otomatik olarak oluşturuldu

## **Steps:**

1. The application is opened by the user.
2. Login to the app and authentication is applied.
3. The application asks the user to specify the type of problem.
4. Detect suitiable developers.
5. Developers that can solve the problem is find from the database.
6. Developers that can solve the problem is displayed to the user.
7. Customer makes the payment.
8. Software developer solves the problem and sends to the user.

# **Activity Diagram**



# **Mock Ups{Figma }**

metin, yazı tipi, ekran görüntüsü, diyagram içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazı tipi, grafik tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazı tipi, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazı tipi, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu