**תכנון ארכיטקטורה מבוססת micro services עבור מערכת לניהול בתי-דין**

**User Management Service**

*מנהל רישום משתמשים, אימות ותפקידים.*

* -- רישום מתשתמש
* -- אימות משתמש
* -- אחזור פרטי משתמש

**Case Management Service**

*מטפל ביצירה, עדכון ואחזור של תיקים, כמו גם ניהול שלבי מחזור החיים של תיק.*

* -- יצירת תיק
* -- אחזור תיק
* -- עדכון תיק
* -- סגירת תיק
* -- עדכון שלב התיק

**Document Management Service**

* *מנהל העלאה, אחסון ושליפה של מסמכים הקשורים לתיקים*/דיונים/משתמשים
* -- העלאת מסמך
* -- אחזור מסמך
* -- מחיקת מסמך
* -- רשימת מסמכים

**Email Service**

*מנהל את שליחת המיילים של האפליקציה*

* -- שליחת מייל

**Other Services**

בדומה לשירותים שפורטו, שירותים נוספים הקשורים לתיקים לפי הצורך.

(לדוג' הפצות יוצאות לגורמים הרלוונטיים, בקשות ביניים שהוגשו על ידי הלקוח ועוד)

**Decision Management Service**

*מטפל בהחלטות הקשורות בתיקים.*

* -- הוספת החלטה
* -- רשימת החלטות
* -- אחזור החלטה
* -- עדכון החלטה
* -- ביטול החלטה

**Discussion Management Service**

*מנהל דיונים ושימועים הקשורים לתיקים.*

* -- זימון דיון
* -- רשימת דיונים
* -- אחזור דיון
* -- עדכון דיון
* --ביטול דיון
* --סגירת דיון

**פירוט חתימות פונקציות עיקריות:**

public class **UserManagementService** {

public User RegisterUser(UserData userData); // POST /users: Register a new user

public AuthToken AuthenticateUser(LoginData loginData); // POST /users/login: Authenticate a user

public User GetUserById(Guid userId); // GET /users/{id}: Retrieve user details

public User UpdateUser(Guid userId, UserData userData); // PUT /users/{id}: Update user details

public void DeleteUser(Guid userId); // DELETE /users/{id}: Delete a user

}

public class **CaseManagementService** {

public Case CreateCase(CaseData caseData); // POST /cases: Create a new case

public Case GetCaseById(Guid caseId); // GET /cases/{id}: Retrieve case details

public List<Case> GetAllCases(); // GET /cases: List all cases

public Case UpdateCase(Guid caseId, CaseData caseData); // PUT /cases/{id}: Update case details

public void CloseCase(Guid caseId); // DELETE /cases/{id}: Delete a case

public Case UpdateCaseStage(Guid caseId, CaseStage newStage); // POST /cases/{id}/stage: Update case stage (e.g., to secretary examination, discussion, decision)

}

public class **DocumentManagementService** {

public Document UploadDocument(Guid caseId, DocumentData documentData,

Guid userId); // POST /cases/{caseId}/documents: Upload a document

public Document GetDocumentById(Guid caseId, Guid documentId); // GET /cases/{caseId}/documents/{documentId}: Retrieve a document

public void DeleteDocument(Guid caseId, Guid documentId); // DELETE /cases/{caseId}/documents/{documentId}: Delete a document

public List<Document> GetDocumentsByCaseId(Guid caseId); // GET /cases/{caseId}/documents: List all documents for a case

}

public class **DiscussionManagementService** {

public Discussion ScheduleDiscussion(Guid caseId, DiscussionData discussionData, Guid userId);

public Discussion GetDiscussionById(Guid caseId, Guid discussionId);

public List<Discussion> GetDiscussionsByCaseId(Guid caseId);

public Discussion UpdateDiscussion(Guid caseId, Guid discussionId, DiscussionData discussionData);

public void CloseDiscussion(Guid caseId, Guid discussionId);

public void CancelDiscussion(Guid caseId, Guid discussionId);

}

public class **DecisionManagementService** {

public Decision AddDecision(Guid caseId, DecisionData decisionData, Guid userId);

public Decision GetDecisionById(Guid caseId, Guid decisionId);

public List<Decision> GetDecisionsByCaseId(Guid caseId);

public Decision UpdateDecision(Guid caseId, Guid decisionId, DecisionData decisionData);

public void CloseDecision(Guid caseId, Guid decisionId);

}