

---

# Software Requirements Specification

من أجل

<جمعية خيرية افتراضية>

إعداد: <رنيم الحرفي محمد علاوي راما سويداني عمار الحسن خالد صوان >

# Table of Contents

خطأ! الإشارة المرجعية غير معروفة.	Table of Contents	
خطأ! الإشارة المرجعية غير معروفة.	Revision History	
خطأ! الإشارة المرجعية غير معروفة.	Introduction	1.
خطأ! الإشارة المرجعية غير معروفة.	Purpose	1.1
خطأ! الإشارة المرجعية غير معروفة.	Document Conventions	1.2
خطأ! الإشارة المرجعية غير معروفة.	Intended Audience and Reading Suggestions	1.3
خطأ! الإشارة المرجعية غير معروفة.	Product Scope	1.4
خطأ! الإشارة المرجعية غير معروفة.	References	1.5
خطأ! الإشارة المرجعية غير معروفة.	Overall Description	2.
خطأ! الإشارة المرجعية غير معروفة.	Product Perspective	2.1
خطأ! الإشارة المرجعية غير معروفة.	Product Functions	2.2
خطأ! الإشارة المرجعية غير معروفة.	User Classes and Characteristics	2.3
خطأ! الإشارة المرجعية غير معروفة.	Operating Environment	2.4
خطأ! الإشارة المرجعية غير معروفة.	Design and Implementation Constraints	2.5
خطأ! الإشارة المرجعية غير معروفة.	User Documentation	2.6
خطأ! الإشارة المرجعية غير معروفة.	Assumptions and Dependencies	2.7
خطأ! الإشارة المرجعية غير معروفة.	External Interface Requirements	3.
خطأ! الإشارة المرجعية غير معروفة.	User Interfaces	3.1
خطأ! الإشارة المرجعية غير معروفة.	Hardware Interfaces	3.2
خطأ! الإشارة المرجعية غير معروفة.	Software Interfaces	3.3
خطأ! الإشارة المرجعية غير معروفة.	Communications Interfaces	3.4
خطأ! الإشارة المرجعية غير معروفة.	System Features	4.
خطأ! الإشارة المرجعية غير معروفة.	System Feature 1	4.1
خطأ! الإشارة المرجعية غير معروفة.	Other Nonfunctional Requirements	5.
خطأ! الإشارة المرجعية غير معروفة.	Performance Requirements	5.1
خطأ! الإشارة المرجعية غير معروفة.	Safety Requirements	5.2
خطأ! الإشارة المرجعية غير معروفة.	Security Requirements	5.3
خطأ! الإشارة المرجعية غير معروفة.	Software Quality Attributes	5.4
خطأ! الإشارة المرجعية غير معروفة.	Business Rules	5.5

## 1. مقدمة

يوفر القسم التالي نظرة عامة على مواصفات متطلبات البرنامج (SRS) لجمعية خيرية افتراضية

### 1.1 الأهداف:

الغرض من SRS هو تحديد كل من المتطلبات الوظيفية وغير الوظيفية لنظام جمعية خيرية. يوفر المستند أيضًا وصفًا شاملاً لنظام إدارة جمعية مع نماذج تحليل UML.

### 1.2 غرض المنتج:

إن نظام الجمعية الخيرية هو برمجية تقدم خدمة جمع تبرعات من المتبرعين وتقديمها إلى مجموعات سكنية لتخفيض أجارات المنازل وأيضاً خدمة اختيار الطلاب منازل في مجموعات سكنية بإيجار ملائم للظروف المادية.

## 2. الوصف العام

### 2.1 تكوين المنتج:

إن هذا النظام الموصوف في هذا ال SRS هو التطبيق البرمجي لنظام جمعية خيرية افتراضية موجهة بهدف التخفيف من مشاكل الإيجار وتوفير سكن للطلاب.

### 2.2 وظيفة النظام:

إن الوظيفة الأساسية للنظام هي تجسيد نظام جمعية خيرية افتراضية تقوم بجمع تبرعات وإعطائه لأصحاب المنازل المعروضة للإيجار لتخفيض أسعار الإيجار على الطلاب الجامعيين وذلك يتم عن طريق مكتب عقاري يعمل كوسيط بين أصحاب المنازل والجمعية.

### 2.3 المستخدمين والمهام الخاصة بهم:

2.3.1 **الطالب:** وهو المستخدم الذي يبحث عن منزل لاستئجاره بمبلغ مناسب يتوافق مع وضعه.

#### المهام الرئيسية:

- تسجيل الدخول على النظام
- التقدم على المنازل لاستئجارها

2.3.2 **المتبرع:** وهو المستخدم الذي يتبرع بالأموال للجمعية الخيرية.

#### المهام الرئيسية:

- تسجيل الدخول على النظام.
- إدخال المبلغ المالي المراد التبرع به للجمعية.
- مراجعة سجلات التبرعات الخاصة به.

2.3.3 **المكتب العقاري:** وهو المستخدم الذي يقوم بإضافة المنازل المتاحة في منطقته للإيجار.

#### المهام الرئيسية:

- تسجيل الدخول على النظام.
- إضافة منزل.
- حذف منزل من النظام.

### 2.3.4 المدير: هو المسؤول الأساسي عن النظام

#### المهام الرئيسية:

- إدارة عملية التسجيل على النظام.
- إدارة عملية التبرعات.
- إدارة عملية التقدم على المنازل.

### 2.4 بيئة التشغيل

في هذا القسم سيتم توصيف البيئة التي سيتم إنشاء النظام بها من بيئة برمجية وبيئة عتاديه والبيئة التي يعمل عليها النظام

#### 2.4.1 المتطلبات البرمجية لعمل النظام:

- حاسب شخصي أو محمول (Over than Windows 7)
- هاتف محمول

#### 2.4.2 متطلبات العتاد الصلب للنظام:

- سيرفر اونلاين
- رام 2 على الهاتف او الحاسب
- مساحة تخزين 100 ميغا على الأقل

### 2.5 شروط التنفيذ والتصميم:

- على المستخدم إدخال جميع معلوماته الشخصية دون إنقاص أي معلومة.
- ألا يقل عمر المستخدم عن 18 عاما.
- التأكد من إدخال الطالب لصورة البطاقة الجامعية الخاصة به.
- أن يتم استخدام بيئة flutter من قبل المطور البرمجي.
- أن يتم استخدام larval من قبل المطور البرمجي.
- أن يتم استخدام الألوان الرمادي E8E7E7 والأخضر 1D7365 في تصميم واجهات المستخدم.
- أن يتم استخدام الخط segoe-ui في تصميم الواجهات.
- أن تكون اللغة عربية

### 2.6 دليل المستخدم

يجب على مستخدم النظام أن يقوم بتسجيل الدخول أو إنشاء حساب ومن ثم إدخال جميع الوثائق والمعلومات المطلوبة دون إنقاص أي معلومة مطلوبة وإلا لن يتم تسجيل المدخلات ثم يجب الانتظار حتي يتم الموافقة على الحساب من قبل المدير وبعدها يمكن للمستخدم القيام بالوظيفة الرئيسية المتاحة له من تبرع أو تسجيل على منزل.

### 2.7 الافتراضات والتبعيات

يعتمد النظام على العديد من الأمور وهي:

- موافقة من وزارة الشؤون الاجتماعية والعمل للحصول على تصريح للجمعية.
- معلومات من شؤون الطلبة للتأكد من صحة المعلومات المقدمة من الطالب.
- اتباع نظام الجمعيات الخيرية في طريقة التعامل مع المتبرعين والمحتاجين وذلك عن طريق أخذ العديد من الوثائق المؤكدة لمصداقية الشخص الذي تتعامل معه الجمعية سواء كان محتاج أو متبرع فإذا كان محتاج فمن الضروري معرفة جميع المعلومات الخاصة بظروفه المعيشية والأسرية والمادية وإذا كان متبرع فيجب عليه تقديم وصف عن عمله

### 3. متطلبات الواجهات الخارجية

#### 3.1 واجهات المستخدم

- عندما يفتح المستخدم التطبيق تظهر له واجهة الترحيب.
- ثم تظهر له واجهة تعريفية تحتوي على معلومات الجمعية ويمكن للمستخدم الوصول لها دون تسجيل الدخول، أيضا يحتوي على زر تسجيل الدخول للانتقال الى واجهة تسجيل الدخول
- عند الانتقال لواجهة تسجيل الدخول يمكن للمستخدم إدخال كلمة مرور وإيميل أو يمكنه إنشاء حساب في حال كان مستخدم جديد
- ثم يتم الانتقال إلى واجهة رئيسية خاصة بكل مستخدم تحتوي على نافذة جانبية تمكن المستخدم من الانتقال إلى الملف الشخصي الخاص به أو تسجيل الخروج أو تغيير كلمة المرور الخاصة به
- يمكن للمستخدم إذا كان متطوع الوصول من واجهة الرئيسية إلى سجلات التبرع الخاصة به
- يمكن للمستخدم إذا كان طالب الوصول

#### 3.2 واجهات العتاد الصلب

تتطلب خدمة النظام جهاز كمبيوتر قادرًا على تشغيل أي متصفح. يوصى بمساحة خالية تبلغ 100 ميغا بايت في ذاكرة الوصول العشوائي. تتطلب خدمة خادم النظام القدرة على النمو ديناميكيًا بناءً على عدد العملاء وتكرار الاتصالات ولكنها لا تتطلب في البداية قدرًا كبيرًا من المساحة أو الموارد للتشغيل. يلزم وجود ماوس ولوحة مفاتيح وظيفيين من أجل الوظائف الكاملة للمسؤول.

#### 3.3 واجهات البرمجيات

- يتطلب النظام لتفعيله برمجيا برنامج مثل android studio ببيئة flutte .
- يتطلب إنشاء واجهة المسؤول التعامل مع بيئة مثل dashboard.
- يتطلب النظام إنشاء قاعدة بيانات التعامل مع MySQL.

### 4. متطلبات النظام

يحتوي هذا القسم على شرح وتوصيف كامل للنظام بمتطلباته الوظيفية وغير الوظيفية.

#### 4.1 ميزات النظام 1

##### 4.1.3 المتطلبات الوظيفية

يحتوي هذا القسم على مواصفات المتطلبات الوظيفية للنظام

1. واجهة dashboard للمسؤول.
2. تسجيل الدخول وإنشاء حساب لكل مستخدم للنظام .
3. إذا كان المستخدم طالب:

- الوصول المباشر لمعلومات الجمعية .
- الوصول المباشر لمعلومات المنازل المعروضة للإيجار.
- اختيار منزل من المنازل والتقدم بطلب للمسؤول .
- تتبع طلب الإيجار الذي قدمه (سجل طلبات)

4. إذا كان المستخدم متبرع:

- الوصول المباشر لمعلومات الجمعية الخيرية.
- التقدم بمبلغ للتبرع به للجمعية الخيرية.
- الوصول إلى سجل تبرعاته.

5. إذا كان المستخدم مكتب عقاري:

- الوصول المباشر إلى معلومات الجمعية الخيرية.
- إضافة وحذف منازل.

6. المسؤول:

- يدير حسابات المستخدمين.
- إدارة عملية إضافة المنازل من قبل المكاتب العقارية.
- إدارة (قبول ورفض) عمليات التقدم على المنازل.
- إدارة عملية التبرع

## 5. المتطلبات الغير وظيفية

### 5.1 متطلبات الأداء:

- إن النظام يستجيب للمستخدم في وقت بسيط حتى يأتي القبول من المسؤول ، يظهر القبول للمستخدم على شكل رسالة او إشعار أو أي صيغة مشابهة .

### 5.2 متطلبات الحماية:

- إن النظام يقوم بالتحقق من أن الحساب موجود في قاعدة البيانات
- يقوم النظام بمصادقة تسجيل الدخول لحفظ الأنشطة الخاصة بكل مستخدم
- يتم نقل المعلومات بشكل آمن إلى قاعدة البيانات بشكل دائم دون أي تغيير في المعلومات
- ستكون المعلومات الخاصة بالمستخدم مشفرة ومحمية وغير قابلة للوصول من أي مستخدم آخر
- لا يمكن لأي شخص الوصول إلى المعلومات ما لم يكن المسؤول (Admin) ولديه صلاحيات الوصول للنظام

### 5.3 قابلية التوسع:

يمتلك النظام القدرة على التعامل بكفاءة مع الكم المتزايد من العمل، و القدرة على التلاؤم مع النمو السريع في البيانات.

### 5.4 قابلية الاستخدام:

يحقق النظام خاصية قابلية الاستخدام حيث أنه يمكن استخدامه من قبل مستخدمين ذوي خبرات عالية أو خبرات متوسطة أو ضعيفة لتحقيق الأهداف المرجوة من النظام وذلك عن طريق استخدام تصميم مبسط خالي من التعقيدات يمكن فهمه والتعامل معه بسهولة.

### 5.5 عناصر جودة النظام

#### 5.5.1 التوافر

النظام يعمل معظم الوقت والخادم لا يتعطل لأكثر من بضع دقائق لتجنب إزعاج العملاء.

#### 5.5.2 المرونة

إذا دعت الحاجة في المستقبل، يمكن تعديل البرنامج لتغيير المتطلبات.

#### 5.5.3 قابلية الصيانة

يمكن إصلاح البرنامج بسهولة في حالة حدوث خطأ.

#### 5.5.4 قابلية التنقل

يمكن تثبيت البرنامج بسهولة على الأجهزة وسيعمل بسلاسة وفقاً للمتطلبات.