

# Заглавна страница

Спецификация на изискванията

## Мобилно чат приложение

Версия 1.0

Версия	Автор	Дата	Състояние	Бележка
1.0	А. Златанов	07.05.2019	одобрен	Release

Ангел Стефанов Златанов

Ф№: 1601261021

Информатика, 3 курс

2018/2019

# 1. Въведение

## 1.1. Цел

Изграждане на чат приложение за андроид, което представлява „single page application“. Предоставя възможност на потребителите да изпращат и получават текстови съобщения и изображения на своите приятели в него.

Критерии:

<sup>35</sup><sub>17</sub> задължителни:

- създаване на профил от приложението
- търсене на потребители по „username“
- добавяне, премахване, блокиране на потребители
- изпращане, получаване на текстови съобщения и изобр.
- разглеждане и обновяване на собствения си профил

<sup>35</sup><sub>17</sub> незадължителни:

- разглеждане на чужди профили

# 1. Въведение

## 1.2. Обхват

- а. област на приложение на софтуера – всеки „smartphone“, който е с операционна система „Android 5.1“ и по-нова и достъп до интернет.
- б. групи от потребители – потребители

## 1.3 Дефиниции

- \* Системата е защитена.
- \* Потребителите могат да:
  - \* създават и редактират профила си
  - \* търсят потребители по „username“
  - \* преглеждат своите приятели и чатове
  - \* изпращат и получават текстови съобщения и изображения на своите приятели
  - \* изпращат и получават покани за приятелство, да премахват приятели, както и да блокират потребители

# 1. Въведение

## 1.4. Референции

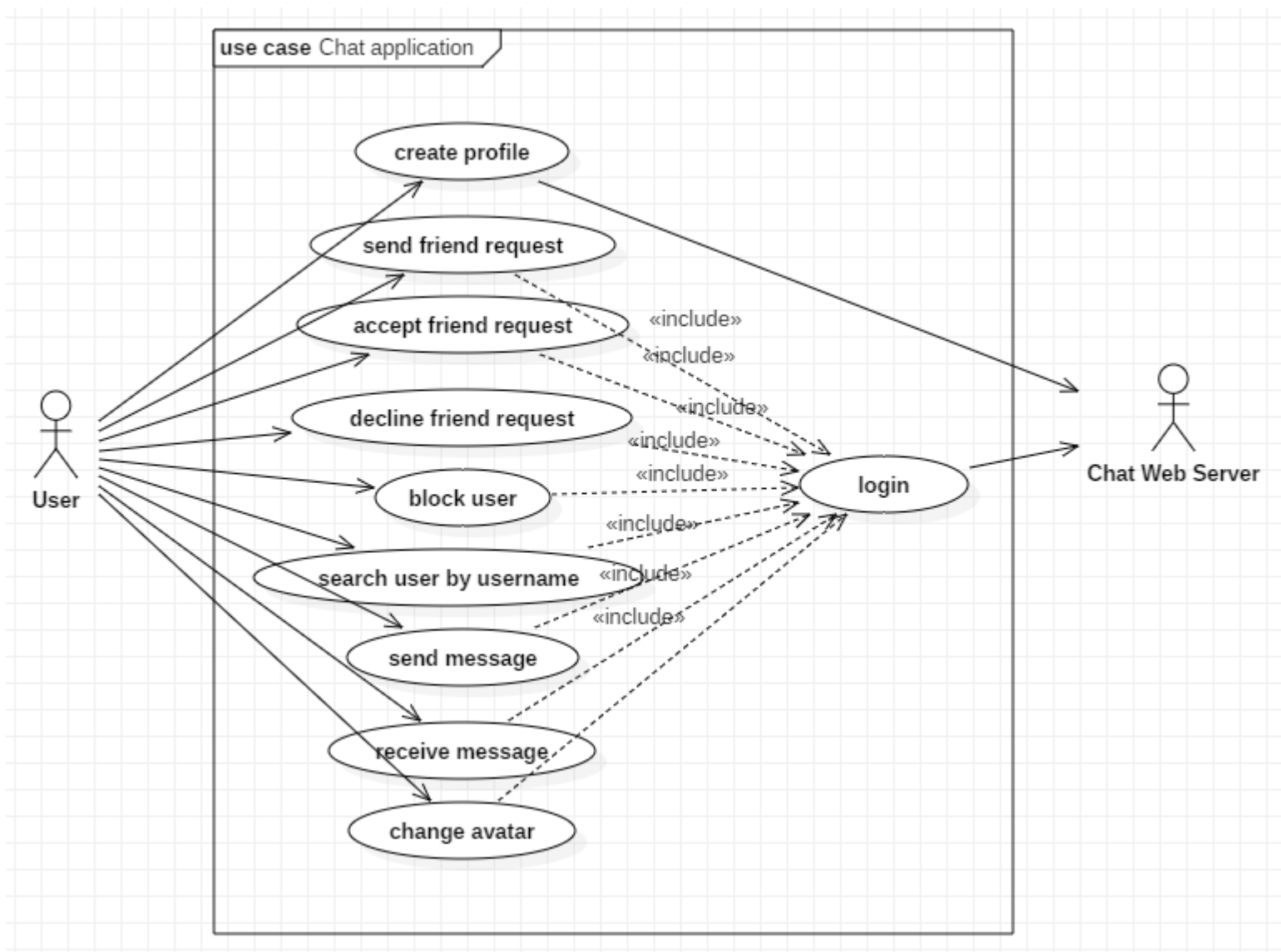
Софтуерът има връзка към:

<sup>35</sup><sub>17</sub> уеб сървър (http & web socket)

<sup>35</sup><sub>17</sub> локална датабаза (SQLite), в която да запазва вече изтеглените съобщения от сървъра, информация за потребители и изображения, както и да тегли от нея с цел показване на графична информация преди успешно изтегляне на актуална информация от сървъра и обновяване

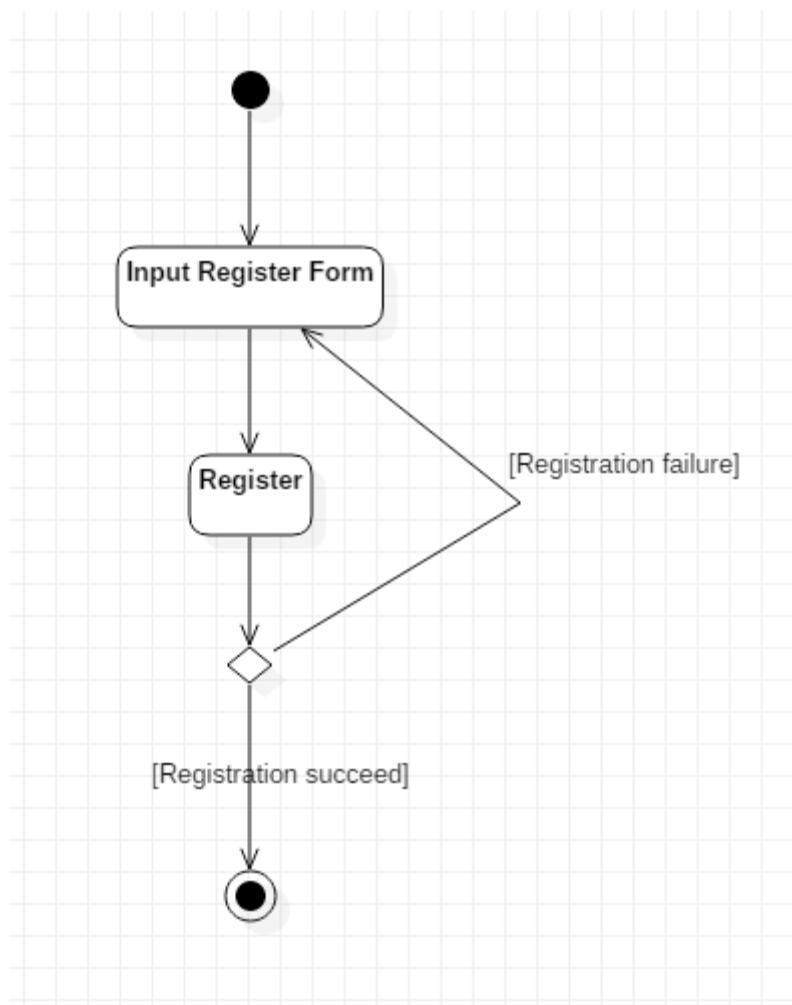
# 1. Въведение

## 1.5. Преглед (use-case diagram)

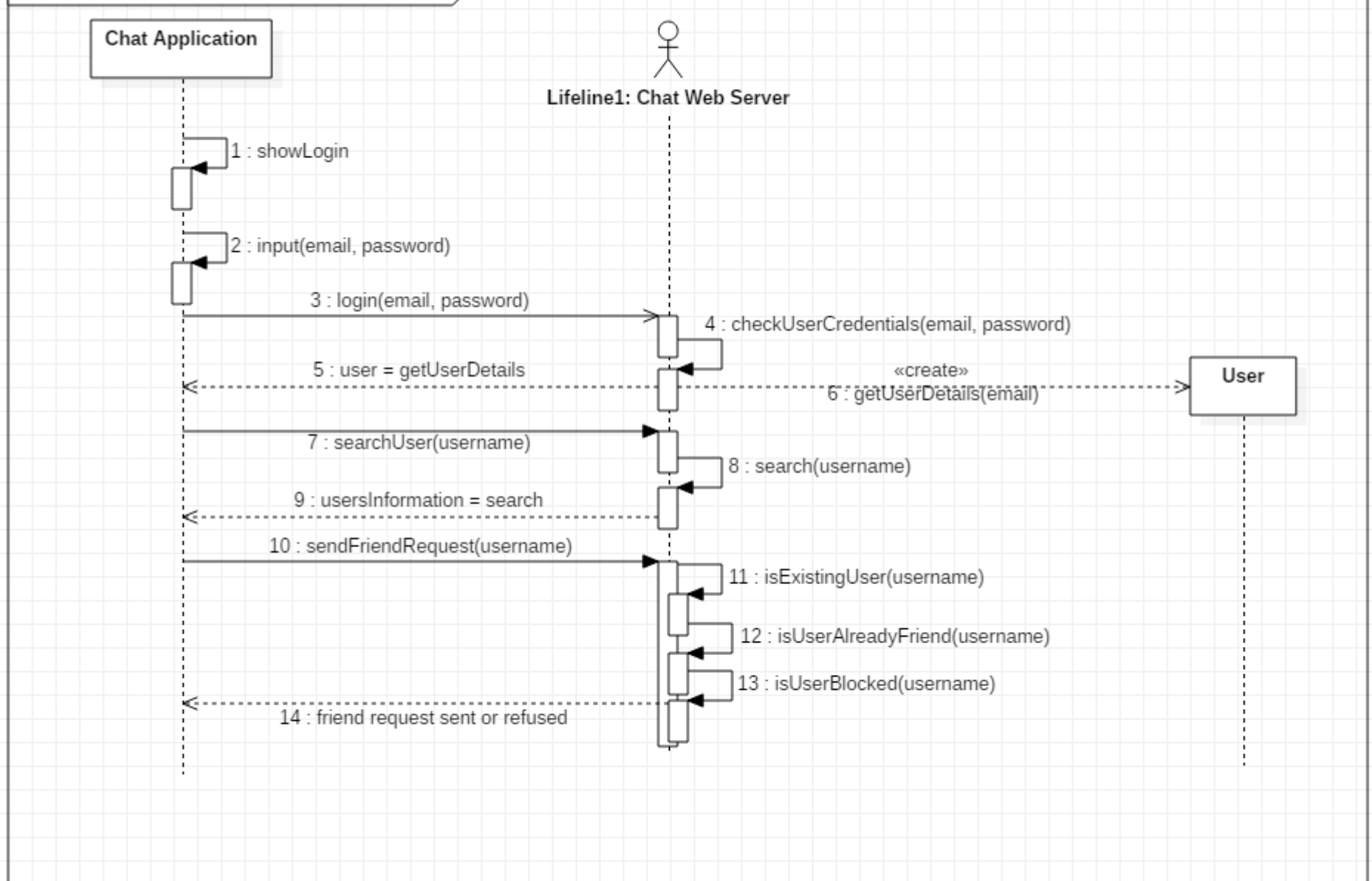


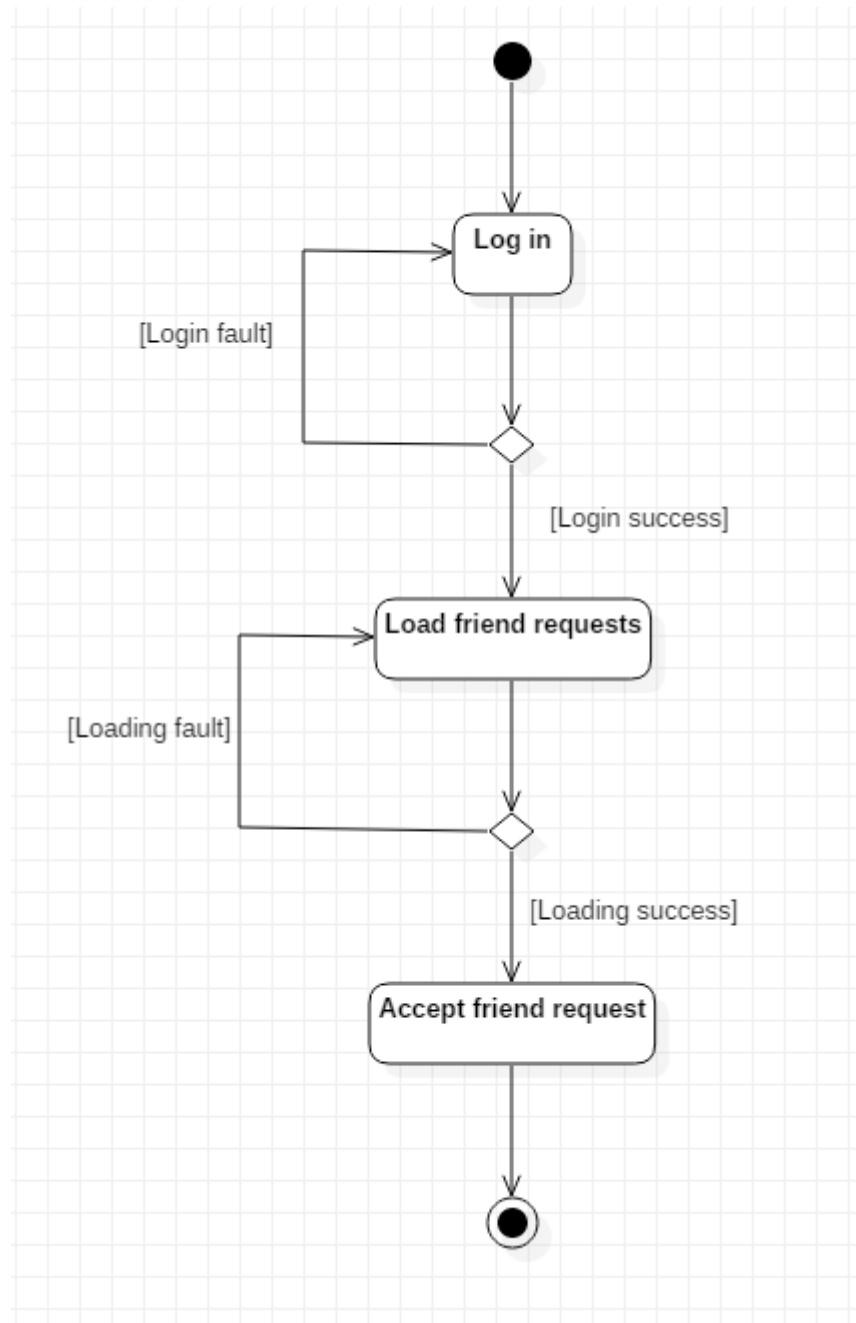
## 2. Описание на системата

### 2.1. Функции на продукта

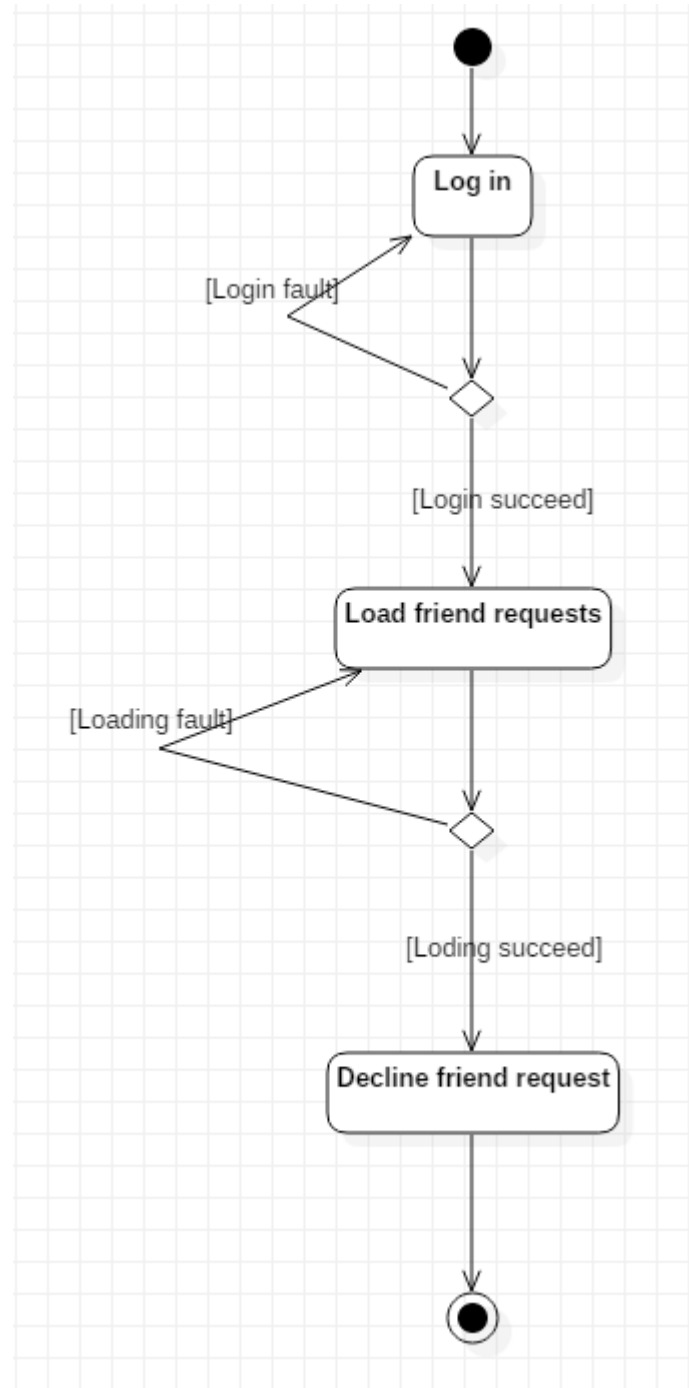


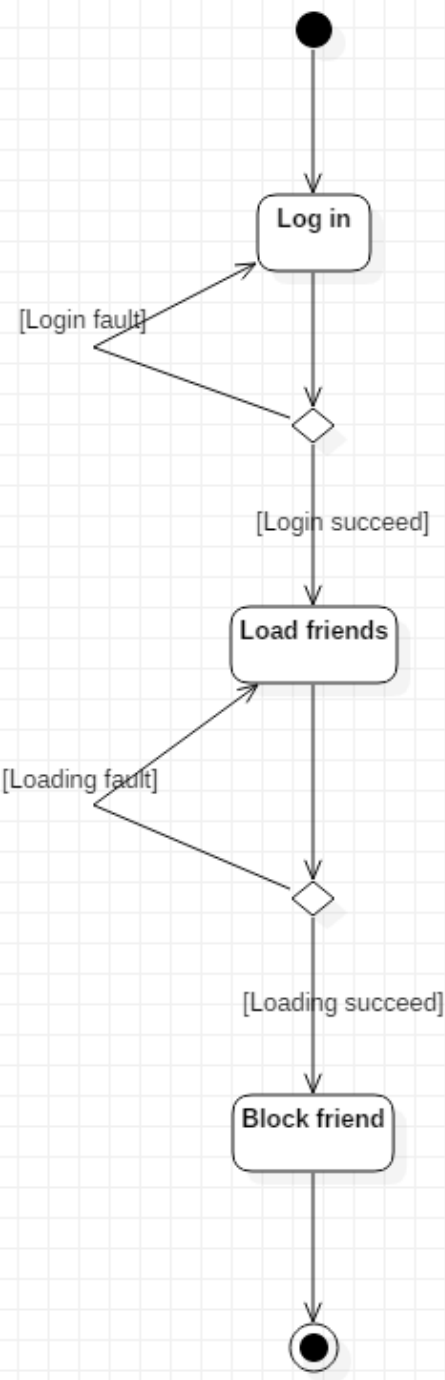
interaction SequenceDiagram sendFriendRequest

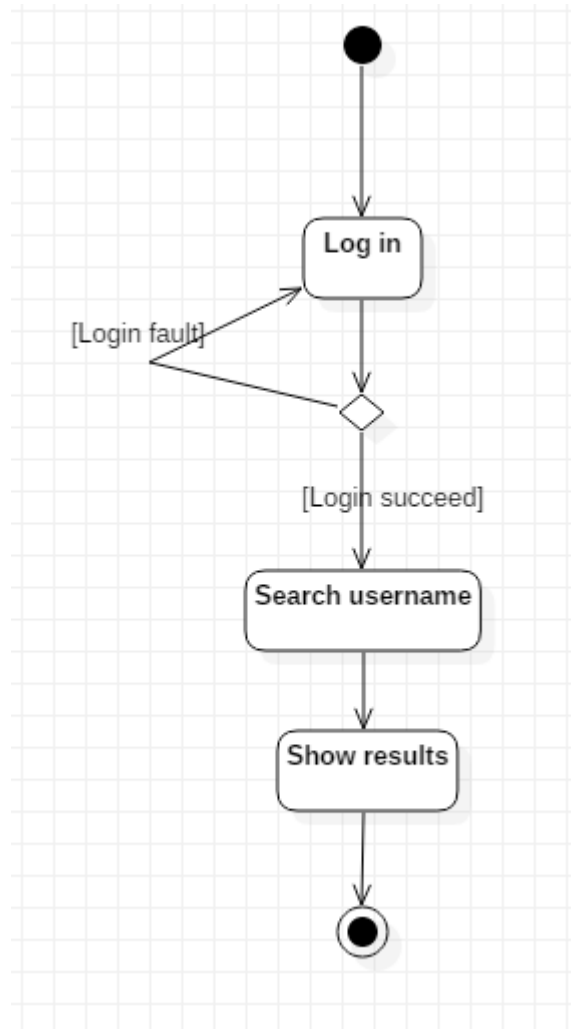


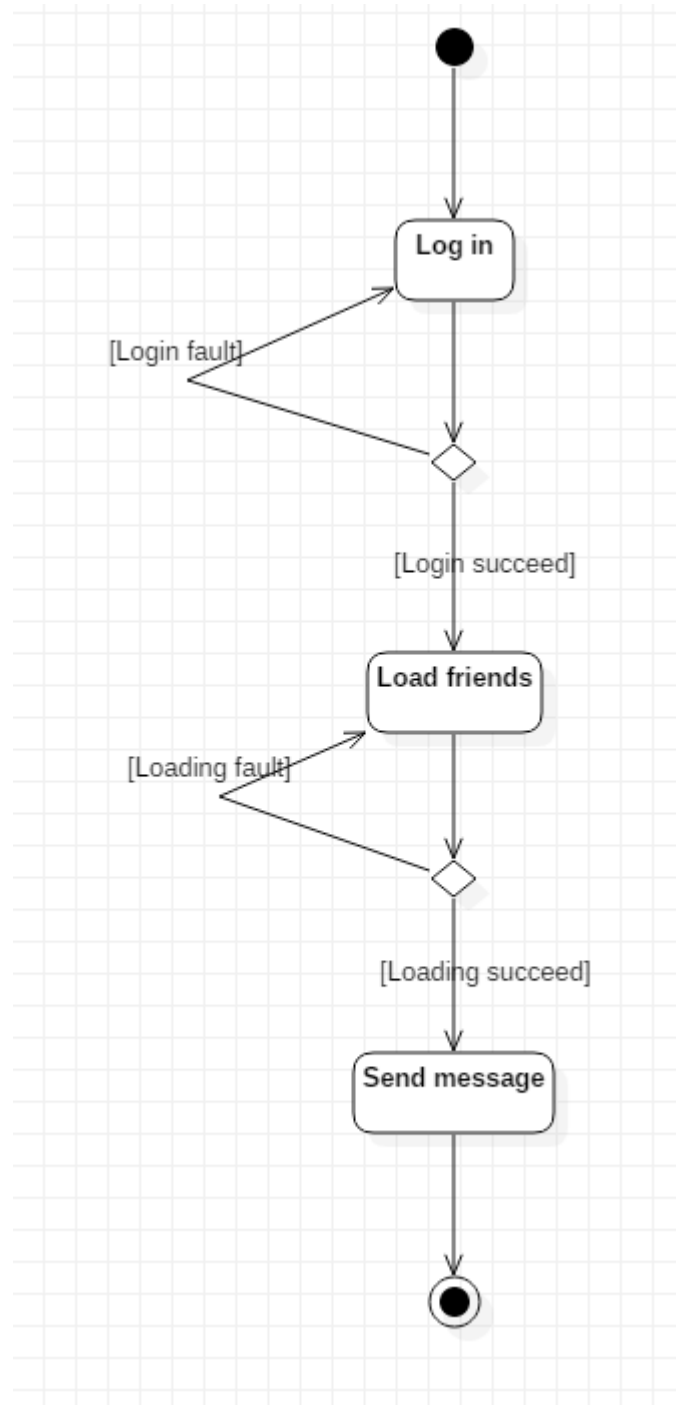


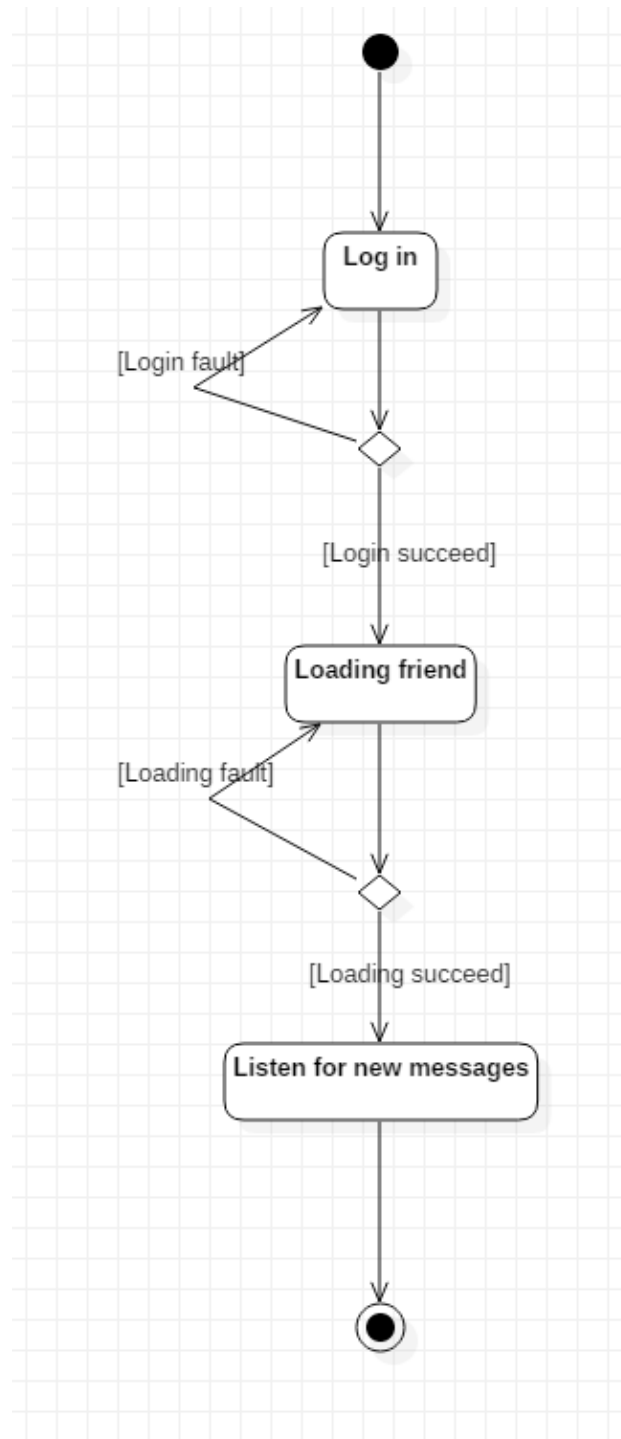


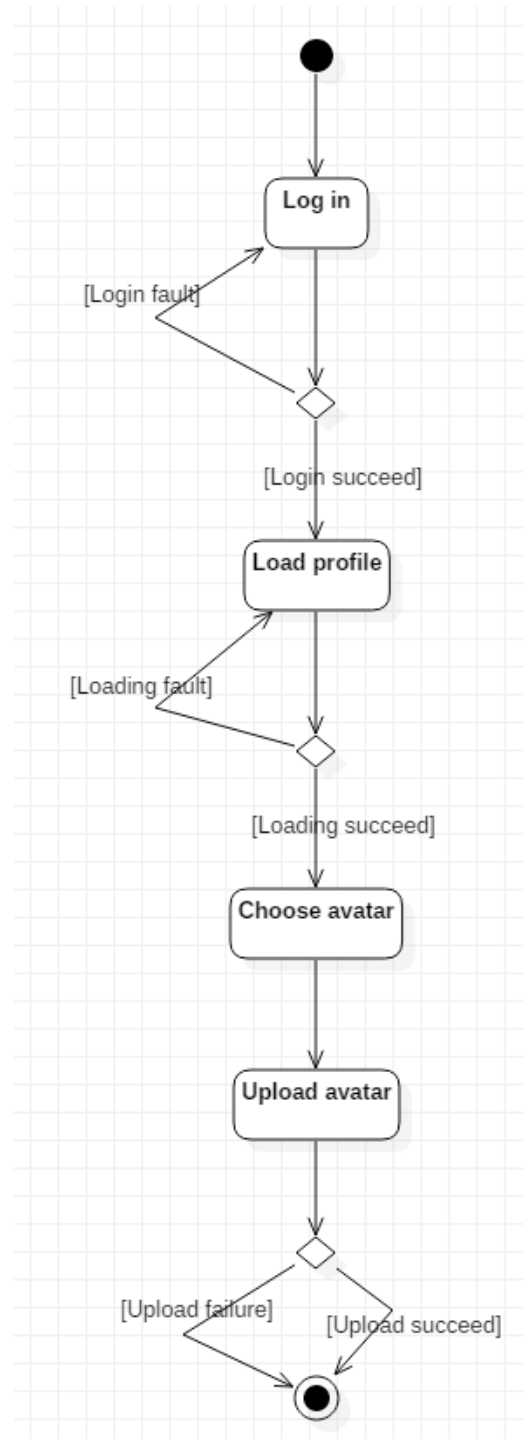












## 2.2. Характеристики на потребителите

### а) потребител

- профил – защитен с email и password
- роля – user
- права – търси потребители, изпраща покани, приема покани, получава и изпраща съобщения, следи онлайн статуса на приятелите си, блокира потребители, обновява профила си (аватара си)

## 2.3. Ограничения

### а) общи:

- няма

### б) от областта на приложение:

- няма

### в) времеви ограничения:

- разработка – 1 седмица

## 2.4. Перспективи на продукта

- реализация като iOS и стандартно уеб приложение за браузър

# 3. Специфични изисквания

## 3.1. Изисквания за потребителския интерфейс

- тъмни, но не прекалено тъмни нюанси на графичния интерфейс
- да има прилични и съобразни ефекти при преминаване към различни части от интерфейса

## 3.2. Изисквания за данни

- в приложението ще се пазят стари съобщения, данни за вече търсени потребители и приятели

## 3.3. Изисквания за производителност

### а) статични:

- много много много на брой едновременно работещи потребители



б) динамични:

- времето за отговор на заявка трябва да бъде много бързо (части от секундата при добра свързаност)
- брой транзакции, които се обработват за минута – от 0 до няколко хиляди

### 3.4. Ограничения при проектирането

- хардуерни ограничения – колкото сабя покаже
- системата има само 2 отворени порта за приемане на http заявки при логин и регистрация и един за websockets. Всичко друго е затворено. Заявките през веб сокета стават посредством правила за писане и четене и е много добре защитено. XD

# Критерии за качество (ISO 9126)

	мн. добро	добро	нормално	неважно
<b>Функционалност</b>				
* Удобство	+			
* Прецизност	+			
* Възможност за интеграция	+			
* Съгласуваност	+			
* Сигурност	+			
<b>Надеждност</b>				
* Допустими грешки		+		
* Възстановимост след грешка	+			
<b>Ефективност</b>				
* Поведение към време	+			

	мн. добро	добро	нормално	неважно
*Поведение към ресурси			+	
<b>Използваемост</b>				
* Разбираемост	+			
* Разучаемост	+			
* Оперирруемост	+			
<b>Гъвкавост</b>				
* Възможност за промяна		+		
* Възможност за поддръжка		+		
* Възможност за анализ			+	
* Възможност за тестване		+		
<b>Преносимост</b>	+			

### 3.6. Технически изисквания

- езици за разработка – Java
- технологии (android, sqlite, spring boot, websockets, mongodb)
- архитектура на средата – 3-слойна
- хардуер – консерва

### 4. Цена на софтуер {Метод на функционалните точки}

тип	прост	междинен	сложен	общо
Външен входен (IT)	4x3	2x4	1x6	26
Външен изходен (OT)	4x4	2x5	1x7	33
Вътрешен логически файлове (FT)	5x7	1x10	1x15	51
Външен интерфейсен файлове (EI)	1x5	1x7	1x10	22
Външен справочен (QT)	4x3	2x4	1x6	26
Общо FC:				158

1. Данните и управляващата информация се изпращат или получават по комуникационни линии - 2
  2. Има разпределена обработка на данни - 1
  3. Важно е достигането на висока ефективност - 3
  4. Експлоатация върху силно натоварена операционна конфигурация – хардуер, софтуер - 4
  5. Интензивността на транзакциите е висока - 5
  6. Наличен е интерактивен режим на въвеждане на данните - 2
  7. Цели се ефективност от гледище на потребителя - 1
  8. Наличен е интерактивен режим на актуализирането на данните - 3
  9. Логиката на обработките е сложна - 0
  10. Програмният код трябва да е reusable - 1
  11. Цели се лесно инсталиране - 2
  12. Цели се лесна експлоатация - 1
  13. Може да се използва от разнообразни потребители - 4
  14. Приложението е гъвкаво и лесно се модифицира - 2
- 0 – не влияе  
1 – незначително влияе  
2 – умерено влияе  
3 – средно влияе  
4 – значително влияе  
5 – силно влияе

$$PC = 31$$

$$PCA = 0.65 + (0.01 \times 31) = 0.96$$

$$FP = FC \times PCA = 158 \times 0.96 = 151.68$$

$$\text{Effort} = 0.6 FP + 0.001 FP^2 = 91.008 + 23.0068224 = 114.0148224$$

## 2.5 Документ „Поток на събитията“

Документ „Поток на събитията“, описващ изпращането на заявка за приятелство:

Даденият use-case изпълнява поредица от операции, чрез които потребител може да направи заявка за приятелство на друг потребител.

Преди да има тази възможност обаче, трябва да бъде логнат в профила си.

Ако е регистриран и се впише успешно, ще има достъп да изпраща покани за приятелство на потребители, които не са го блокирали.

След като е вписан, той трябва да отвори търсачката и да потърси за специфичен username или поне част от него. След това ще му бъде върната информация (username,аватар,userID) на всички потребители, които започват с търсения username или частта от него.

Потребителят избира иконка за изпращане на покана и изскача прозорец за въвеждане на поздравително съобщение, което по подразбиране е „Hey \*username\*! Let's be friends!“. След потвърждаването на това съобщение, потребителят получава информация за това дали заявката е изпратена успешно или пък не, ако правилата за писане не са изпълнени (например е блокиран от потребителя).

Самото приложение прави проверка в правилата за писане на заявки, когато потребител се опита да запише нещо в колекцията за заявки. Правилата са следните:

1. дали съществува потребител с ID, към което се изпраща поканата
2. дали потребителят, към който се изпраща поканата и изпращащият вече не са приятели
3. дали потребителят, към когото се изпраща поканата за приятелство, не е блокирал изпращащият и обратното

След като тези правила са проверени с данните от базата, следва успешен запис - изпълнена заявка за приятелство и към изпращащият потребител се връща отговор за успешен/неуспешен запис.