

Krav- och analysdokumentation för Fia med knuff

Hanna Boquist, Amanda Cyrén, Johan Selin, Emma
Stålberg, Philip Winsnes

HEJAP

TDA367 Objektorienterat programmeringsprojekt
Chalmers tekniska högskola
24 oktober 2021

Innehåll

1	Introduktion	1
1.1	Generella egenskaper för applikationen	1
1.2	Definitioner, akronymer och förkortningar	1
2	Krav	2
2.1	Användarberättelser	2
2.2	Definition of Done	10
3	Användargränssnitt	11
3.1	Startsidan	11
3.2	Skapelsen av ett spel	12
3.3	Spelvyn	12
3.4	Regelboken	14
3.5	Topplista	14
4	Domänmodell	15
4.1	Klassansvar	15
5	Referenser	16
5.1	Verktyg	16
5.2	Bibliotek	16

1 Introduktion

Projektet ämnar att skapa en Android-applikation av brädspellet Fia med knuff, som tillåter en eller flera spelare att spela mot en CPU eller mot varandra. Applikationen ska tillhandahålla spelaren ett konventionellt Fia med knuff i digitalt format men också erbjuda extra funktionalitet såsom lokalt flerspelarläge och anpassningar av spelet efter antalet spelare.

1.1 Generella egenskaper för applikationen

- Programmet är huvudsakligen utvecklat för Android-läsplattor.
- Programmet är skrivet i Java.
- Spelet är en digital tolkning av ett konventionellt Fia med knuff. Spelet öppnar dessutom upp för eventuella speciallägen som skiljer sig från ett traditionellt Fia.
- Antalet spelare är valbart.
- Spelet hjälper visuellt till med tillåtna drag och turordning.
- En regelbok hjälper spelare att förstå hur spelet ska spelas.
- Spelare kan välja att spela mot en eller flera CPU:er.
- Under spelets gång är det valbart att lyssna på stämningsfull musik.
- Spelet har stöd för både svenska och engelska.

1.2 Definitioner, akronymer och förkortningar

- GUI, grafiskt användargränssnitt.
- Java, ett programmeringsspråk som är plattformsoberoende.
- JUnit, ett ramverk för testning i programmeringsspråket Java.
- JavaDoc, ett standardiserat sätt att skapa dokumentation till javaprojekt.
- UML-diagram, olika diagram som visar en översikt på kodstrukturen.
- Aktivitet, en skärm där användargränssnittet ritas upp och som användaren interagerar med.
- Fragment, tillhör en aktivitet som en modulär del av gränssnittet.
- CPU, en spelare som är kontrollerad av datorn och utför beräknade drag.
- Vy, en visuell representation av gränssnittet.
- Android, ett mobilt operativsystem.

2 Krav

2.1 Användarberättelser

Prioritet 1	Prioritet 2	Prioritet 3	Prioritet 4
US01: Rita upp spelplan	US10: Spela mot en CPU	US16: Feedback när en pjäs går ut	US19: Spelresultat
US02: Slå tärning	US11: Avsluta ett pågående spel	US17: Spelet går att anpassa efter olika enheter	US20: Sätta på/stänga av musik
US03: Välja mellan drag	US12: Starta om ett pågående spel	US18: Feedback vid vinst	US21: Topplista
US04: Endast göra tillåtna drag	US13: Läsa regler för spelet		US22: Spela fler än 4 spelare
US05: Urskilja pjäser	US14: Pjäs hoppar stegvis framåt		US23: Timer
US06: Välja antal spelare	US15: Huvudmeny		US24: Spelläge med specialregler
US07: Se vilken spelares tur			US25: Anpassa svårighetsgrad
US08: Turen går vidare till nästa spelare			US26: Välja tema
US09: Knuffa ut pjäs			

Figur 1: Färdiga User Stories är markerade i grönt

Prioritet 1

US01: Rita upp spelplan

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna se spelplanen för att kunna spela spelet.

Acceptanskriterier

- Spelbrädet ska synas.
- Varje spelare ska ha varsin färg.
- Varje spelare ska ha fyra pjäser i sin färg.
- Det ska synas hur många spelare som är med.
- Man ska se pjäserna för varje spelare.

US02: Slå tärning

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna slå en tärning så att jag kan göra ett drag.

Acceptanskriterier

- Spelaren kan slå en tärning.
- Bara den aktiva spelaren kan använda tärningen.
- Efter slaget kan spelaren göra ett drag enligt resultatet på tärningen.
- Spelaren kan bara slå tärningen en gång.

US03: Välja mellan drag

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna välja mellan olika drag för att flytta mina pjäser och ta mig fram i spelet.

Acceptanskriterier

- Spelaren ska bara kunna välja att flytta sina egna pjäser.
- Spelaren ska inte kunna flytta en pjäs som inte går att flytta.
- Efter slaget kan spelaren göra ett drag enligt resultatet på tärningen.
- Spelaren kan välja vilken av sina pjäser som den vill flytta.

US04: Endast göra tillåtna drag

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag endast kunna utföra tillåtna drag så att jag spelar enligt spelets regler.

Acceptanskriterier

- Ett otillåtet drag kan ej genomföras.
- En spelare som står i boet ska inte kunna gå ut på en 3:a.

US05: Urskilja pjäser

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna urskilja mina pjäser från de andra för att kunna förstå spelet lättare.

Acceptanskriterier

- Alla spelares pjäser har en och samma färg.
- Flera spelares pjäser kan inte ha samma färg.
- Pjäserna syns på brädet.

US06: Välja antal spelare

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna välja antalet spelare (2-4st) så att spelet kan anpassas utefter det.

Acceptanskriterier

- Det ska gå att välja antalet spelare.
- Det ska inte gå att välja att det är fler spelare än 4.
- spelet ska anpassas utefter antalet spelare som är valda, dvs antalet pjäser = $4 \cdot$ antalet spelare och antalet bon är lika många som antalet spelare.
- Varje spelare har tilldelats en färg med tillhörande pjäser.

US07: Se vilken spelares tur

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna se vilken spelares tur det är så att jag kan följa spelet.

Acceptanskriterier

- Tärningen placeras vid aktiv spelare.
- Endast en spelare ska vara aktiv i taget.

US08: Turen går vidare till nästa spelare

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag att turen att spela går över till nästa person när jag gjort mitt drag så att spelet kan fortsätta.

Acceptanskriterier

- Den aktiva spelaren ska ha en tärning vid sitt namn.
- Turen ska gå vidare till nästa spelare när en spelare gjort ett drag.
- Nästa person att spela kan aldrig bli någon som redan gått ut med alla sina pjäser.
- Om inget drag är möjligt ska turen gå över direkt till nästa spelare.
- nästa person att spela kan aldrig bli någon som redan gått ut med alla sina pjäser.
- Om inget drag är möjligt ska turen gå över direkt till nästa spelare.

US09: Knuffa ut pjäs

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna knuffa ut en annan spelares pjäs.

Acceptanskriterier

- Pjäsen som blir knuffad flyttas tillbaka till sitt bo.
- Man inte knuffa ut en egen pjäs.
- Om en pjäs hamnar på samma position som en annan pjäs så ska den pjäsen knuffas ut.
- Det ska synas visuellt när en pjäs knuffas ut.

Prioritet 2

US10: Spela mot en CPU

Implementerat?

- Delvis

Beskrivning

- Som spelare vill jag kunna spela mot en CPU så att jag kan spela när jag är ensam.

Acceptanskriterier

- Spelaren ska kunna välja att spela mot datorn.
- CPU-spelaren sköter sig själv under spelets gång.
- CPU-spelaren har logik som gör att den väljer smarta drag.
- CPU-spelarens drag visas upp visuellt i vyn.

US11: Avsluta ett pågående spel

Implementerat?

- Delvis

Beskrivning

- Som spelare vill jag enkelt kunna avsluta spelet så att jag inte är fast i spelet för länge.

Acceptanskriterier

- Tillbakaknapp i sidopanelen för att komma till huvudmenyn från spelvyn.
- Dialogruta som ber om att bekräfta valet.
- Det ska finnas ett val att spara det pågående spelet för att senare återuppta spelet (feature).
- Det ska gå att avsluta ett pågående spel för att sedan starta ett nytt utan oönskade konsekvenser.
- Efter ett avslutat spel ska spelarna få chansen att återvända till huvudmenyn.

US12: Starta om ett pågående spel

Implementerat?

- Delvis

Beskrivning

- Som spelare vill jag kunna starta om spelet för att smidigt fortsätta spela.

Acceptanskriterier

- Knapp i sidopanelen som startar om samma pågående spel.
- Dialogruta som bekräftar valet.
- Spelplanen återställs med samma spelare. Om en spelare gått ut med sina pjäser ska också den spelaren återställas. Tärningen hamnar i sin startposition och turen hamnar hos den första spelaren.
- Efter ett avslutat spel ska spelarna få chansen att starta om spelet.

US13: Läsa regler för spelet

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna läsa reglerna så att jag vet hur jag ska spela.

Acceptanskriterier

- Regelboken kan nås från menyn.
- Regelboken kan nås från spelet.
- Man kommer tillbaka till huvudmenyn / spelet när regelboken stängs.

US14: Pjäs hoppar stegvis framåt

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna se min pjäs hoppa dit jag vill så att det blir mer verklighetstroget.

Acceptanskriterier

- Någon form av animation för flyttning av pjäser ska vara verklighetstroget och snyggt.
- Pjäsen kommer inte direkt fram till sin slutposition.
- Pjäsen försvinner från rutan den tidigare stått på när den flyttas/dyker upp på en ny.

US15: Huvudmeny

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag ha en huvudmeny med knappar för nytt spel, regler och topplista för att hålla spelet organiserat.

Acceptanskriterier

- En huvudmeny med knappar för nytt spel, regler och topplista ska finnas.

- Knapparna i huvudmenyn ska vara klickbara.

Prioritet 3

US16: Feedback när en pjäs går ut

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag ha någon sorts feedback när jag går ut med en pjäs så att jag känner mig nöjd.

Acceptanskriterier

- Spelare ska kunna gå ut med en pjäs.
- Kolla så att draget är tillåtet.
- Pjäsen försvinner från spelplanen.
- Grafisk feedback visas upp.

US17: Spelet går att anpassa efter olika enheter

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag att det ska finnas stöd för att kunna anpassa spelet efter flera enheter.

Acceptanskriterier

- Innehållet ska skalas om proportionerligt på två valda storlekar (320dp och 480dp).
- Alla värden ska vara satta med hjälp av dimensionsfiler, så att det är enkelt att lägga till värden för en ny enhetsstorlek.

US18: Feedback vid vinst

Implementerat?

- Delvis

Beskrivning

- Som spelare vill jag ha någon feedback när jag vinner så att jag känner mig nöjd och vill vinna igen (spela igen).

Acceptanskriterier

- Spelare ska kunna gå ut med sista pjäsen.
- Kolla så att draget är tillåtet.
- Pjäsen försvinner.
- Grafisk feedback om vinst visas.
- Dialogruta dyker upp med meddelande.
- Val i dialogruta leder till passande utgång.

Prioritet 4

US19: Spelresultat

Implementerat?

- Delvis

Beskrivning

- Som spelare vill jag att det ska finnas en lista som visar resultatet av spelet så att jag kan se vilken ordning vi kom på.

Acceptanskriterier

- En lista ska finnas med slutordningen på spelet.
- Listan med slutordningen ska visas i slutet av spelet.

US20: Sätta på/av musik

Implementerat?

- Ja

Beskrivning

- Som spelare vill jag kunna sätta på/stänga av spelmusik så att jag kan sätta stämningen.

Acceptanskriterier

- Knapp för volym finns tillgänglig.
- Knappen för volym är klickbar.
- Vid klick på ljudknapp ska ljudet stängas av och på.
- Symbolen för ljud visar tydligt om ljudet är på eller av.

US21: Topplista

Implementerat?

- Delvis

Beskrivning

- Som spelare vill jag kunna se en topplista med tidigare vinnare och hur många gånger de vunnit, så att jag kan komma ihåg hur det har gått i tidigare matcher.

Acceptanskriterier

- Det ska finnas en topplista.
- Topplistan ska uppdateras efter avslutat spel.
- Topplistan ska gå att nå från huvudmenyn.

US22: Spela fler än 4 spelare

Implementerat?

- Delvis

Beskrivning

- Som spelare vill vi kunna spela fler än fyra spelare så att vi kan vara fler tillsammans.

Acceptanskriterier

- Man kan välja antal spelare.
- Negativa eller för stora spelarantal ska inte vara möjligt.
- Spelplanen ska anpassas för olika antal (ex. 5 spelare ger en 5-stjärna).

US23: Timer

Implementerat?

- Nej

Beskrivning

- Som spelare vill jag kunna se en timer för att se hur lång tid spelet har pågått.

Acceptanskriterier

- Synlig timer.
- När en spelomgång startar så startar timern.
- När en spelomgång är slut så stannar timern.
- Timern pausas om man går ur spelet på något vis.
- Timern pausas om man går ur spelet på något vis.
- Timern pausas om man läser reglerna eller öppnar hjälp-guiden.

US24: Spelläge med specialregler

Implementerat?

- Nej

Beskrivning

- Som spelare vill jag kunna ändra spelläget till ett spel med mer specialregler.

Acceptanskriterier

- Specialregler ska kunna väljas från startkonfigurationen av nytt spel samt under spelets gång.
- Specialreglerna ska inte vara på som standard och fördefinierade värden ska vara ifyllda vid start av ett nytt spel för att underlätta och förenkla.

US25: Anpassa svårighetsgrad

Implementerat?

- Nej

Beskrivning

- Som spelare vill jag kunna välja funktionaliteter så att jag kan anpassa svårighetsgraden och välja vilken utmaning jag vill ha.

Acceptanskriterier

- Fördefinierade inställningar och värden ska redan vara ifyllda vid skapandet av ett nytt spel för att underlätta för spelarna.
- Inställningarna ska kunna påverka svårighetsgraden för spelet.

- Det finns en meny för att välja extra inställningar.
- Det ska gå att välja att se vilka följder ett drag får (visa var pjäsen landar).

US26: Välja tema

Implementerat?

- Nej

Beskrivning

- Som spelare vill jag kunna sätta ett tema så att spelet passar min stil.

Acceptanskriterier

- Knapp för val av tema finns tillgänglig.
- Knapp för val av tema är klickbar.
- Möjliga val av tema dyker upp i t.ex. en rullgardinsmeny.
- Vid val av tema ändras speldesignen (färg/typsnitt med mera).

2.2 Definition of Done

För alla användarberättelser gäller följande acceptanskriterier:

- Applikationen bygger och kompilerar.
- All implementerad funktionalitet har klarat JUnit-testerna.
- Hela kodutvecklingen är dokumenterad via versionskontroll.
- Alla specifika acceptanskriterier för de olika användarberättelserna ska vara uppnådda.
- Alla publika metoder och klasser i kodbasen är dokumenterade via JavaDoc.
- Kodstrukturen ska överensstämma med samtliga UML-diagram.
- All kod är kontrollerad av minst två gruppmedlemmar.
- Koden är referentgranskad.

3 Användargränssnitt

Syftet med den här sektionen är att beskriva och visualisera applikationens användargränssnitt med fokus på navigationen mellan programmets olika vyer. Målet med applikationens användargränssnitt är att det ska vara användarvänligt samt föra tankarna till traditionella brädspel.

Generella användarvänliga designdrag är att knappar visar sin klickbarhet genom dimning vid klick. Texter följer i stora drag samma typsnitt vilket bidrar till en enhetlig design. Beroende på enhetens språkinställningar ändras översättning mellan svenska och engelska. Eftersom applikationen är designad för Android, tar navigationen hänsyn till de konventionella knapparna som finns på en Android-enhet. Därför visas inga ytterligare visuella tillbakaknappar.

3.1 Startsidan

Vid start av applikationen nås startsidan (se figur 2) där användaren har möjlighet att antingen starta ett nytt spel, kolla topplistan eller läsa på regler för spelet.



Figur 2: Huvudmenyn

3.2 Skapelsen av ett spel

När användaren trycker på knappen "nytt spel" på startsidan, öppnas menyn i figur 3. Där väljs antal spelare, namn på spelarna samt om någon av spelarna ska fungera som en CPU.

För att ge spelarna en förståelse över hur spelet kommer sättas upp, ändras listan med spelare dynamiskt utifrån det angivna spelarantalet. För varje spelare finns en fördefinierad färg som visar vilken färg som kommer tilldelas spelaren. För att en spelare ska kontrolleras av programmet, bockas CPU-rutan i för den spelaren. CPU-spelaren kan då dessutom tilldelas ett önskat namn. Spelet går inte att starta om några av spelarna har samma namn.

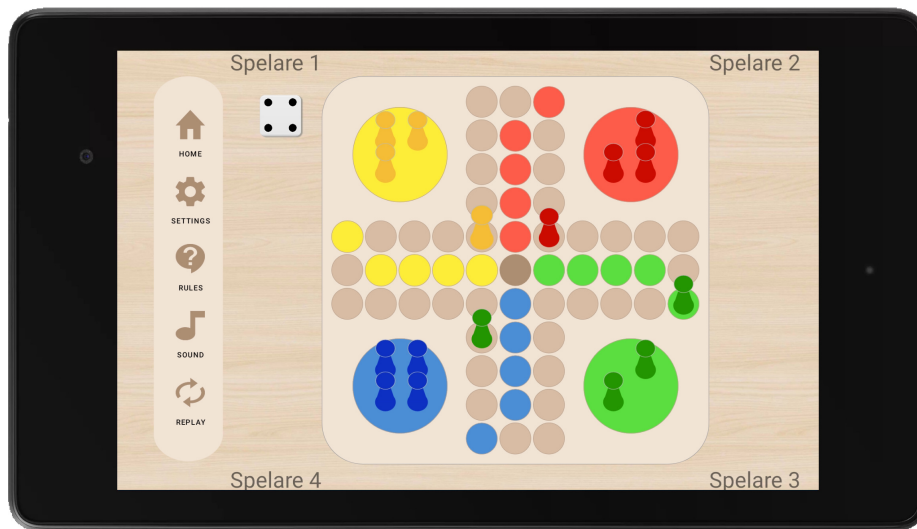


Figur 3: Skapelsen av ett spel

3.3 Spelvyn

När spelet sätts igång kommer användaren till spelvyn (se figur 4) där spelplanen ritas upp med pjäser utifrån hur många spelare som valts. Till vänster finns en sidomeny med olika navigeringsmöjligheter. Tärningen placeras vid start hos spelaren som ska börja slå och flyttas sedan runt till de olika spelarna.

Beroende på vem av spelarnas tur det är hamnar tärningen hos spelarens namn. Det signalerar för spelaren att det är spelarens tur. Då tärningen slås visas en animering vilket ger en realistisk känsla av tärningen. Därefter oscillerar de pjäser som är flyttbara för spelaren. När väl en pjäs väljs, vandrar pjäsen till den ruta den hamnar på, vilket ger ett realistiskt intryck på spelet. Dessa designval bidrar till en enklare förståelse och ett flyt i spelkänslan.



Figur 4: Spelvyn

En sidopanel låter användaren gå till hemskärmen, inställningar, regelboken, tysta bakgrundsmusiken och spela spelet från början. Vid val att gå till hemskärmen visas en dialogruta (se vänster i figur 5). Detta gäller även för valet att spela om spelet (se höger i figur 5).



Figur 5: Dialogrutor

3.4 Regelboken

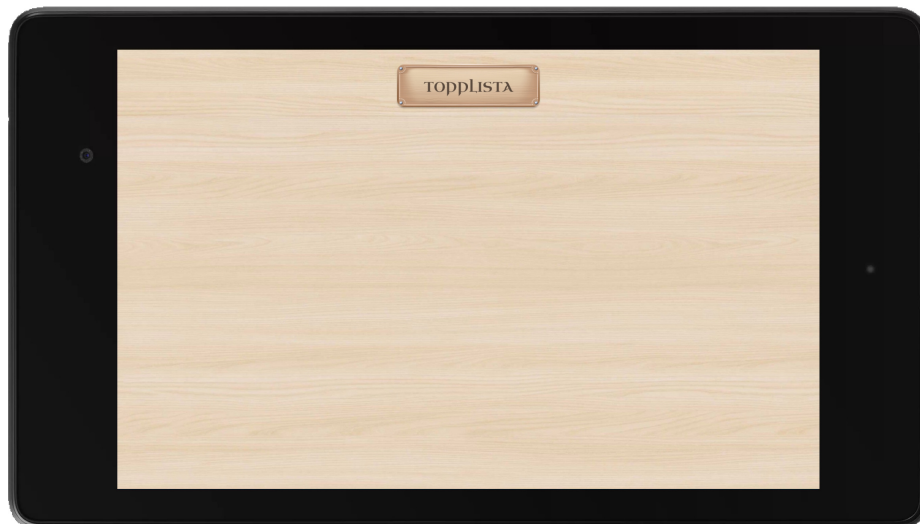
Regelboken hjälper användaren med att få en förståelse över reglerna i spelet, och kan ses i figur 6. Vyn går att nå från både sidopanelen i spelvyn och i huvudmenyn. Regelboken går att bläddra i på liknande sätt man bläddrar i en fysisk bok.



Figur 6: Regelboken

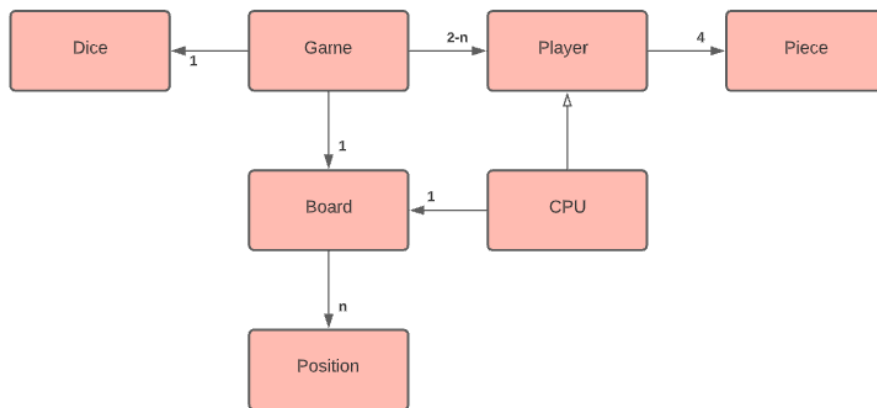
3.5 Topplista

Topplistan visar spelnamnen på de spelare som vunnit tidigare matcher. Den som har vunnit flest matcher ligger överst i listan. Topplistan ses i figur 7.



Figur 7: Topplista

4 Domänmodell



Figur 8: Domänmodell

4.1 Klassansvar

- **Dice** - Skapar en tärning som går att slå samt håller värdet av tärningsslaget.
- **Game** - Har huvudansvar över hur ett spel fungerar genom att koppla ihop de olika komponenterna som finns med.
- **Player** - Skapar och representerar spelare i modellen och ansvarar för att skapa sina pjäser.
- **Piece** - Representerar en pjäs i spelet och urskiljer sig visuellt med ett färg-attribut. Ansvarar för data om var den ska gå ut på brädet och hur många steg den tagit.
- **Board** - Skapar och representerar ett bräde av spelet. Ansvarar för att flytta alla pjäser och hålla koll på var de står.
- **CPU** - En klass som skapar CPU-objekt samt kopplar ihop spelaren med datorlogik för hur en spelare kan välja sina drag.
- **Position** - Skapar och håller ett index vilket används för att förbinda en plats i GUI:t med en position.

5 Referenser

5.1 Verkt yg

- [Android Studio](#) Projektets IDE.
- [Figma](#) En applikation f r design av grafiska anv ndargr nssnitt.
- [GitHub](#) Versionshantering med hj lp av Git.
- [Trello](#) Verkt yg f r att kunna planera och visualisera status i projektet.
- [Lucid](#) Ett webbverkt yg f r att skapa och jobba med UML-diagram online.
- [CircleCI](#) Verkt yg f r kontinuerlig integrering.
- [Google Drive](#) En delad mapp f r att ha m tesprotokoll och dokument samlade.
- [Slack](#) Plattform f r kommunikation inom projektet.
- [Zoom](#) Verkt yg f r distansm ten.

5.2 Bibliotek

- [JUnit](#) Javas ramverk f r “unit-testing”.
- [AssertJ](#) Ytterligare assertions f r JUnit och f rb tttrar l sbarheten av tester.
- [Apache Commons](#) Underl tta serialisering.
- [numAndroidPageCurlEffect](#) Anv nds f r att skapa en effekt av att v nda ett blad i regelboken.