

**ANALIZA POLA POLA NUMERU OU CODE UTILIZA
AUTOMATA**



HUSI:

Nu	NRE	Naran Kompletu	Sexo
1	112301025	Zelia Barbosa	Feto
2	71901006	João Yopi Quintas	Mane
3	112301028	Abilio da Costa	Mane

**DEPARTAMENTO TÉCNICAS DE COMPUTAÇÃO E INFORMÁTICAS
INSTITUTO PROFISSIONAL DE CANOSSA (IPDC)**

DILI

2025

LIA MAKLOKE

Ho laran luak hasa'e agradese ba nai Santissima Trindade no Inan Doben Maria Virgem basá ho Nia tulun no kbi'it tomak ne'ebé haraik ona mai hakerek na'in, bele finaliza duni *Traballu ba Materia Automata* ida ne'e, maske hasoru dezafius oin-oin. Ho tulun husi nain feto Maria Virgem nia intersetasun no harohan daet nian, hakerek na'in bele finaliza duni *Traballu ba Materia Automata* ida ne'e ho di'ak.

Agradese mos ba fundadora **Sta. Madalena de Canossa** no **Sta.Josefina Baquita** ba Nia karisma no Espiritu tomak ne'ebé maka haforsa ona hakerek na'in durante hala'o prosesu aprandizajem iha IPDC. Iha biban ida ne'e mos, hakerek na'in la haluha hato'o obrigadu no agradese wain ba:

1. Professor da materia **Sr. Constantino F. S. Amaral, M.KOM** ne'ebe fo traballu ida ne'e hanesan matadalan ba aban bainrua hakerek proposta.
2. Magnifica Reitora Instituto Profissional de Canossa (IPDC): Me. Terezinha do Menino Jesus Gosmão Guterres.
3. Familia tomak liliu ba inan aman no maun alin sira hotu, ne'ebe suporta no motiva hakerek nain iha prosesu hakerek.
4. Belun estudante tomak nia suporta ho liafuan ne'ebe maka enkoraja ona hakerek na'in hodi finaliza exame final ida ne'e. Ikus liu hakerek na'in hato'o obrigado/a barak ba ita boot sira nia apoio tomak ne'ebe mak liu husi meios oin-oin hodi finaliza exame final ida ne'e ho diak.

ABSTRATU
ANALIZA POLA POLA NUMERU OU CODE UTILIZA
AUTOMATA

Autómatu komposta husi máquina abstraitu ho estadu limitadu, oferece metodo ne'ebé efikante no automatizadu hodi analiza pola ka padraun iha lina numeriku ka kódigu. Iha ámbitu sira hanesan verifikasiestrutura, validasaun dadus, no deteksaun padräun kompleksu, autómatu bele identifika sekuénsia ne'ebé tuir regras spesífiku liu, husi transisaun entre estudu bazeia ba karakter ka valor ne'ebé prense rekejitus, sistému bele rekoniése padraun numeriku hanesan formatu CPF/NRP, kódigu pesoál, seküénsia repetitivu, ka padräun sira ne'ebé iha signifikadu lojiku iha kontextu partikulár.

Aplikasaun prátku inklui análise ba kódigu-fonte iha programasaun, ne'ebé autómatu uza atu identifika karakteristiku (símbulu) hanesan variável, konstante, ka operadór, hanesan parte husi prosesu prosesing data iha seguransa no fornese digital, autómatu intensivu uza atu sukat no deteta padraun numeriku iha arquivu boot, hanesen kódigu inisial ka padräun tranlasaun. Permite automasaun ba tarefa sira ne'ebé repetitivu no susar ba ema halo ho analiza.

Utilizasaun autómatu hodi analiza pola numeriku ka kódigu simplifika interpretasaun dadus ne'ebé komplikadu, aumenta ho velocidade prosesamentu, no minimiza erro humano. Iha mundu ne'ebé analisa ho dadus numeriku, esbosu ne'e kontribui signifikativamente ba área hanesan siénsia komputasaun, matemática, no análise tekniku, fornese solusaun abstratu ba problema real iha prosesamentu dadus.

Liafuan xave: Pola Pola Numeru, estdado e limitadu *Automata*.

INDISSE

KAPA	Error! Bookmark not defined.
LIA MAKLOKE	ii
ABSTRATU	iii
INDISSE	iv
LISTA FIGURA	vii
LISTA TABELA	viii
KAPÍTULU I	1
INTRODUSAUN	Error! Bookmark not defined.
1.1 Introdusaun	1
1.2 Limitasaun Problema	2
1.3 Formulasaun Problema	2
1.4 Objetivo Pesquiza	2
1.5 Vantazen Pesquiza	3
1.5.1 Vantazen ba Estudante	3
1.5.2 Vantazen ba Instituisaun	3
1.5.3 Vantazen ba Publik	3
1.6 Sistema Hakerek	3
KAPÍTULU II	5
REVIZAUN LITERATURA NO FUNDAMENTU TEORIA	5
2.1 Revizaun literatura	5
2.2 Fundamentu Teoria	8
2.2.1 Automata	8
2.2.2 Numeru	8
2.3 Unified Modelling Language (Uml)	8
2.3.1 <i>Use Case Diagrama</i>	8
2.3.2 <i>Activity Diagrama</i>	9
2.4 Linguagen Programasaun	10
2.4.1 Python	10

2.5	Sofware nebe Suporta	11
2.5.1	Microsoft Office.....	11
2.5.2	Astah community	11
2.5.3	Visual Studio Code	11
2.5.4	Balsamiq.....	11
	KAPITULU III.....	12
	METODOLOJIA PESKIZA	12
3.1	Metologia Rekoilamentus dadus	12
3.2	Metodología Dezenvolvimentu Sistema	12
3.3	Analiza Nesesidade Sistema	13
3.4	Dezenu Diagrama.....	14
3.4.1	<i>Use Case Diagram</i>	14
3.4.2	<i>Activity Diagram</i>	15
3.5	Dezeňu Interface	15
3.5.1	Dezeňu Interface Numeru Binario	15
3.5.2	Dezeňu Interface Numeru Par.....	16
3.5.3	Dezeňu Interface Numeru Konaba.....	16
	KAPITULU IV	17
	REZULTADU PESKIZA	17
4.1	Interface Validador Kode Binario.....	17
4.1.1	Interface Validador Erado	18
4.1.2	Interface Validador Korektu	19
4.2	Interface Validador Numeru Par	20
4.1.3	Interface Validador Erado	21
4.1.4	Interface Validador Korektu	22
4.3	Interface Validador Konaba	23
	KAPITULU V.....	26
	MAKTAKA	26
5.1	Konkluzaun	26
5.2	Sugestaun	26

BIBLIOGRAFIA	27
--------------------	----

LISTA FIGURA

Figura 3.1: Modelo Wetarfall.....	13
Figura 3.2: Aktivity Digrama.....	15
Figura 3.3: Interface Numeru Binario	15
Figura 3.4: Interface Numeru Par.....	16
Figura 5.1: Interface Aplikasaun Validador Kode Binario	17
Figura 5.2: Interface Validador kode Binario Erado.....	19
Figura 5.3: Interface Validador kode Binario Erado.....	19
Figura 5.4: Interface Aplikasaun Validador Kode Binario	20
Figura 5.5: Interface Aplikasaun Validador Laos Numaru Par.....	22
Figura 5.6: Interface Aplikasaun Validador Numaru Par	22

LISTA TABELA

Tabela 2.1; Revizaun Literatura	6
Tabela 2.2; Tabela <i>activity diagrama</i>	10
Tabela 3.1 Ekipamentu Peskizador Nian	14

KAPÍTULU I

INTRODUSAUN

1.1 Introdusaun

Iha era industriál 4.0 dezenvolvimentu teknolojia informasaun mudansa tebes, Iha prosesu desenvolvimentu Industria 4.0 hatur *software* no *hardware* hodi transforma teknolojia artifisiál intelijénsia ba ema nia nessesidade servisu ho efisente, efetivu no lais, husi ne'e ita presija dezenvolve *software* iha area hotu-hotu hodi bele hafasil servisu sira ne'ebe komplexu no bele tulun ema sira ne'ebe presija.

Utiliza Teoria automata iha area edukasaun, finansa, seguransa, no teknologia informasaun, valida numeru dadus ne'ebe ho volume bo'ot no kontinua presiza métodu espesífiku atu analiza no interpreta numeru ho diak. Tanba ne'e analiza padraun (pola-pola) numeru ka kódigu sai elementu importante atu suporta prosesu desizaun no valida padraun numeru binariu, par no inpar.

Automata mak konseitu iha área matematika no siénsia komputasaun ne'ebé estuda sistema automátku ne'ebé fó resposta tuir input ne'ebé simu. Automata bele uza atu prosesu lojiku ho estruturadu iha sistema komputasionál. Iha kontekstu matematika, automata bele implementa uza linguajen programasaun *Python* atu ajuda analiza padraun numeru ka kódigu efisiénte no konsistente.

Kontextu ohin loron ita atu halo validasaun ba numeru sira ho manual, entaun halo validasaun ba numeru sira ne'ebe komplexu presija tempu ne'ebe barak, ida ne'e hanesan problema ne'ebe hakerek nain identifika, iha iniciativa dezenvolve sistema ida bele resolve problema validasaun numeru komplexu utiliza teoria Automata.

Tanba ne'e, peskiza ida ne'e foka liu ba **analiza pola-pola numeru ou kódigu uza automata** utiliza linguagen Programasaun *Python*, kria sistema ida halo validasaun ba numeru ne'ebe konplexu, ho kondisaun ida ne'ebe ita ema valida manual presija tempu ne'ebe barak, sitema ne'e bele valida ho lais tuir padraun dadus numeru komplexu, bele identifika, analiza ho metodu científiku no sistemátiku.

1.2 Limitasaun Problema

Bazeia ba introdusaum iha leten ne'e maka hakerek nain limite problema prinsipál sira ne'ebé hasoru iha peskiza mak tuir mai:

- a. Analiza ne'e limitadu de'it ba numeru Binario no Par.
- b. Métodu ne'ebé uza mak *Deterministic Finite Automata (DFA)*.
- c. Sistema ne'e foka deit ba Validasaun.
- d. Utiliza Linguagen Progremasaun *Python*.

1.3 Formulasaun Problema

Formulasaun problema iha peskiza ne'e mak kria no implementa modelu automata uza *Python* atu suporta prosesu validasaun numeru iha formulasaun ne'e, peskiza sei halo buat sira tuir mai ne'e:

- a. Oinsá dezenvolve lójika *Deterministic Finite Automata (DFA)*, halo valida ba númeru par no bináriu?
- b. Oinsá efisiénsia automata nian hodi detecta *input* numeru ne'ebé la válido?
- c. Oinsá implementa diagrama tranzisaun automata nian ba iha sistema validador ne'ebé kria ona?

1.4 Objetivo Pesquiza

Objetivu husi peskiza ida ne'e mak :

- a. Kria diagrama *Deterministic Finite Automata (DFA)* ne'ebé sai hanesan baze lójika ba validador número par no bináriu.
- b. Transforma lojika automata nian ba kodigu programasaun hodi bele *input* husi utilizador no fo rezultadu validasaun.
- c. Kapasidade sistema nian hodi distinge dadus ne'ebe valido no la valido tuir regra kodigu binariu no número par.
- d. Identifika no analiza *state* automata nian bainhira *input* ne'ebe la tuir padraun.
- e. Hanesan parte ida bele completa traballu grupu ba materia Automata Peskiza departamentu Técnicas Computação e Informática iha Instituto Profissinal de Canossa.

1.5 Vantazen Pesquiza

1.5.1 Vantazen ba Estudante

- a. Ajuda estudante **kompriende konseitu automata** iha analiza padraun numeru ou kódigu
- b. Aumenta abilidade estudante iha **implementasaun automata uza Python**
- c. Fó esperiénsia prática atu aplika automata iha konteks Data Science
- d. Desenvolve hanoin logiku, analítiku, no sistemátiku
- e. Bele uza rezultadu analize hanesan referénsia ba estudu ou peskiza seluk

1.5.2 Vantazen ba Instituisaun

- a. Fó kontribuisaun ba dezenvolvimentu akademiku iha área Data Science no Teknologia Informasaun
- b. Ajuda instituisaun aumenta kualidade peskiza estudante.
- c. Bele uza hanesan materiál aprendizajen ou referénsia kurikulum
- d. Suporta aplikasaun teknologia automata iha prosesu analize dados
- e. Aumenta reputasaun instituisaun iha área siénsia no inovasaun

1.5.3 Vantazen ba Publik

- a. Fó kompriensaun basiku kona ba analiza padraun dados uza automata.
- b. Ajuda publik kompriende utilizasaun Python no Data Science iha vida loron-loron.
- c. Bele aplikadu iha área praktiku hanesan edukasaun, administrasaun, no seguransa.
- d. Fó informasaun ne’ebé presiza, eficiente, no útil ba tomada desizaun.
- e. Aumenta literasia digital no konsiensia teknolojiku iha sosiedade.

1.6 Sistema Hakerek

Tuir mai ne’e sistema hakerek ne’ebé hakerek nain utiliza hodi hakerek trabalho iha materia *Automata* ida ne’e:

1.7.1 KAPITULU I INTRODUSAUN

Kapítulu ida ne’e hahu husi mahamosuk problema, formulasaun problema, limitasaun problema, objetivu, benefisiu peskiza no Vantagen.

1.7.2 KAPITULU II REVIZAUN LITERATURA NO FUNDAMENTU TEORIA

Kapitulu ida ne'e hahu husi revisaun literatura ne'ebe iha no teoria ne'ebe utiliza iha peskiza.

1.7.3 KAPITULU III METODOLOJIA PESKIZA

Kapitulu ne'e kompostu husi deskrisaun jeral kona ba fatin peskiza, sistema *hardware* no *software* ne'ebe utiliza hodi halao peskiza no fo analiza detallu kona ba nesesidade ne'ebe utiliza iha peskiza, metolojia peskiza ne'ebe uza halo dezenū kona ba sistema ne'ebe atu kria.

1.7.4 KAPITULU IV REZULTADU PESKIZA

Kapitulu ida ne'e sei fo esplikasaun kona ba implementasaun no esplikasaun ba sistema.

1.7.5 KAPITULU V MAKTAKA

Kapitulu ida ne'e mak kapítulu ikus nian no koalia kona ba konklusaun husi peskiza no sujestauun ba sistema atu dezenvolve.

KAPÍTULU II

REVIZAUN LITERATURA NO FUNDAMENTU TEORIA

2.1 Revizaun literatura

Revizaun litetaura hanesa bazefundamentu ida atu fortifika referensia sira ne'ebe peskiza tia ona, ho nia tópiko “Analiza Pola Pola Numeru Ou Code Utiliza Automata”, iha ne'ebe antes ne'e hakerek ona husi peskizador sira seluk hanesan tuir mai:

Tuir matenek nain (Akmal, 2025) kria Sistema ho topiku **“Analisis Data Biologis dalam Mengidentifikasi Gen atau Protein yang Memiliki Pola Ekspresi Serupa”** Sistema data analisis ida hodi halo detensaun ba protein husi hahan sira, hodi fornese hahan nebe kualidade e klinte sira.

Tuir matenek nain (Mansyur & Ridho, 2020) kria Sistema ho topiku **“Analisis Pola Permintaan Publikasi Data Badan Pusat Statistik Menggunakan Association Rule Apriori”** Sistema website sira halo analisis entre dadus uluk no agora hodi fornese wifi, fornese ba ema sira nebe presija.

Tuir matenek nain (Valencia & Atmojo, 2024) kria Sistema ho topiku **“Analisis Pola Pembelian pada Data Penjualan CanNgopi menggunakan Algoritma FP-Growth”** Sistema naliza sistema markado no nesesiadade nebe vendedores sira presija barak liu.

Tuir matenek nain (Purnomo, Kosala Dwidja; Anggraeni, Irma Dwi ; Riski, Abduh;, 2023) kria Sistema ho topiku **“Analisis Pola Harga Saham dengan Modifikasi Metode Eksponen Hurst dan Box counting”** Sistema naliza sistema analiza rendimentu finansail kada fulan.

Tuir matenek nain (Suci Intania Indah & Voutama, 2025) kria Sistema ho topiku **“Analisis Pendapatan dan Pola Pembatalan Reservasi Hotel Menggunakan Exploratory Data Analysis (EDA)”** Sistema atu analiza hahalok cliente nian entre Resort Hotels no City Hotels.

Tuir matenek nain (Fissiovany, Misdiyanto, & Hikmah, 2025) kria Sistema ho topiku **“Rancang Bangun Aplikasi Vape Shop Dengan Penerapan Data Mining Menggunakan Algoritma Apriori Untuk Pola Pembelian Konsumen Berbasis Web”** Sistema atu analiza redimentu finansial tama no sai.

Tabela 2.1; Revizaun Literatura

Nu	Autor	Tópiku	Explikasaun	Metodu	Rezultadu	Rekomendasau
1	M. Haikal Akmal, et al. (2025)	Analiza Data Biolójiku ba Gen/Protein	Analiza padraun espresaun protein ne'ebé hanesan hosi kazu estudu ba moras iha animal (mouse).	Spectral Clustering	Konsege identifika grupu protein ne'ebé iha relasaun ba malu hosi dataset kompleksu.	Uza métodu clustering seluk hodi kompara presizaun rezultadu nian.
2	Fachruddin Mansyur & Farid Ridho (2020)	Padraun Peditu Publikasaun BPS	Analiza oinsá públiku asesu ba dadus estatística iha website BPS nian.	Association Rule (Apriori)	Públiku baibain asesu dadus ho título hanesan maibé tinan ne'ebé la hanesan iha tempu ida de'it.	Hadia kualidade website hodi fó fasilidade download ne'ebé integradu ba tinan oioin.
3	Shanty Valencia & Wahyu T. Atmojo (2024)	Analiza Padraun Sosa iha CanNgopi	Analiza hahalok konsumidór hodi hatene produtu saida mak sosa hamutuk barak liu.	FP-Growth Algorithm	Hetan kombinasaun produtu (snack, signature, non-coffee) ho valor confidence 97%.	Kria programa "bundling" ka promo ba produtu ne'ebé ema sosa hamutuk.

4	Kosala D. Purnomo, et al. (2023)	Analiza Padraun Folin Saun (Saham)	Analiza dinamika no karakterística fraktal hosi folin asoens nian iha merkadu finanseiru.	Hurst Exponent & Box Counting	Konsege klasifika folin saham ba kategoria random, persistent, no anti-persistent.	Bele uza hodi halo prediksaun ba folin saham iha futuru ho baze fraktal.
5	Suci I. Indah & Apriade Voutama (2025)	Padraun Kanselamentu Rezerva Otél	Analiza diferença hahalok cliente entre Resort Hotel no City Hotel.	Exploratory Data Analysis (EDA)	City Hotel iha kanselamentu barak liu; Resort Hotel iha flutuasaun osan boot iha tempu feriadu.	Jere kestaun kanselamentu iha City Hotel ho sistema "non-refundable" ka sosa uluk.
6	Devangga Fisdiovany, et al. (2025)	Aplikasun Vape Shop (Data Mining)	Dezenvolvu sistema web hodi analiza padraun sosa produtu vape nian.	Association Rule (Apriori)	Sistema konsege prodús regra asosiasaun validu ba kombinasau produtu (liquid, coil, etc).	Implementa rezultadu asosiasaun ne'e hodi hili produtu ba stok foun.

2.2 Fundamentu Teoria

2.2.1 Automata

Automata mak konseitu iha matematika no siénsia komputasaun ne'ebé estuda sistema abstraktu ne'ebé bele fó resposta automátku tuir input ne'ebé simu. Automata normalmente define ho komponenti sira hanesan estado (state), input, transisaun, estado inicial, no estado final. Iha konteks analize padraun numeru ou kódigu, automata finitu (Finite Automata) uza atu rekognise padraun repetitivu no aturan logiku ne'ebé klaru. Automata ajuda simplifika prosesu analize dados tanba sistema bele halo desizaun automátku ho base iha aturan ne'ebé defini ona. Bazeia ba definisaun iha leten tamba ne ami foti matenek nain (Sttawan, 2018) nia hanoin katak, Automata maka mákina abstratu sira ne'ebé bele rekoñese, simu, ka hamosu fraze ida iha lian partikulár ida

2.2.2 Numeru

Númeru mak unidade báziku husi matemática, Iha lian número mak idea ne'ebé mai husi ita-nia kbiit hodi identifika objetu ida husi objetu seluk, uza hodi hatene quatidade husi objetu ida, Número mak Objetu lójika hanesan karakterística husi klase ka grupu ida, exemplu número 2 mak buat ida ne'ebé hanesan entre liman rua, matan rua no sira seluk tan.

2.3 Unified Modelling Language (Uml)

2.3.1 Use Case Diagrama

Use Case Diagram mak diagrama ne'ebe dezen aktor, use case no relasaun hanesan espasaun asasaun ne'ebe fo valor kalkulasaun ba aktor.(Wijana et al., 2024) *Use Case Diagram* hanesan diagrama ida iha *UML* ne'ebe halo dezen interksaun entre sistema no aktor, *Use Case Diagram* mos bele hakrekek tipu interksaun entre utilizador no sistema.

Tabela 2. 1 Use Case Diagram

Nu	Simbolu	Deskrisaun
1.		<i>Autor:</i> signifika Ema ne'ebe halo intekrasaun ho sistema, simbolu ida ne'ebe mak representa ema, hanesan Admin, estudante nsst.
2.		<i>Use case:</i> Funksionalidade ne'ebé oferese husi sistema atu halo interaksaun entre autor no sistema. Simpolo ida ne'e hodi tau tabela sira.
3.		<i>Association:</i> Atu halo koneksaun entre autor no <i>use case</i> .
4.		<i>Extend:</i> sibolu ida ne'ebe signifa hodi aumenta tan Use Case no hambrik mesak.
5.		<i>Include:</i> Normalmente <i>use case</i> ida ne'e utiliza atu halo relasaun ba usecase ne'ebé propozitu tenke liu husi prosesu <i>use case</i> balun.

2.2.2 Activity Diagrama

Activity Diagram mak dezena aktividade sira, objetu, *State* Trasizaun *state* no *event*. ho liafuan seluk maka aktividade diagrama husi servisu no dezena karakteristika sistame ba aktividade. (Wijana et al., 2024) *Activity Diagram* hanesan diagrama iha *UML* ne'ebe iha hafoun prosesu sira ne'ebe akontese iha Sistema.

Tabela 2.3 Tabela *activity diagrama*

Nu	Figura	Naran	Funsaun
1		<i>Generalization</i>	<i>Generalization</i> : funau atu liga husi aktividade ida ba aktividade seluk, ou ka elementu ida ba elementu seluk
2		Aktividade	Aktividade ne'e hatudu oinsa sistema ne'e bele interaksaun ho ema.
3		<i>Initial Node</i>	<i>Initial Node</i> signifika katak sistema ida sei hahu husi simbolu ida ne'e
4		<i>Activity Final Node</i>	<i>Activity Final Node</i> ne'e katak remata husi sistema ida.
5		<i>Nary Association</i>	<i>Nary Association</i> katak desizaun ne'ebé desidi husi opsaun rua ne'ebé akontese iha sistema los no salah.
6		Símbolu <i>Fork</i>	<i>Fork</i> ne'e utiliza atu hatudu Aktividades ne'ebé paralelu.

2.4 Linguagen Programasaun

2.4.1 Python

Tuir matenek nain (Sidyk, p. 2025) hateten, Python maka linguajen programasaun nivel aas ida ne'ebé maka popular tebes entre cientista dadus sira, analista sira, no dezenolvedor software sira. Susesu Python nian iha siénsia dadus nian bele atribui ba nia sintaxe ne'ebé moos no fasil atu komprende no biblioteka oioin ne'ebé suporta tarefa analítica oioin. Python nia utilizasaun sira

ihá siénsia dadus inklui análize dadus, vizualizasaun, no dezenvolvimentu modelu aprendizajen mákina.

2.5 Sofware nebe Suporta

2.5.1 Microsoft Office

Microsoft Office mak software ida hanesan apartamentu aplikasaun eskritóriu ne'ebé kria husi *Microsoft*. Software ne'e hala'o iha sistema operasaun *Windows*, *MacOS* no *Linux* nia okos. Iha aplikasaun oin-oin ne'ebé koñesidu iha *Microsoft* nia eskritóriu no uza barak hanesan *Microsoft Word*, *Powerpoint* (ramadan, 2025).

2.5.2 Astah community

Astah community mak *software* intuitivu ida-ne'ebé fasil atu kria *uml* no tipu diagramus sira seluk iha arranju ida-ne'ebé simples. Iha janela prinsipál, bele haree diagram, nia estrutura no propriedade objetu, ida-idak iha seksaun ketak ida. Kona-ba tipu diagrama ne'ebé suporta, programa ne'e permite ita atu kria Atividade, Composite Structure, Class, Communication, UseCase, Deployment, Sequence and Component diagrams. (In, 2024).

2.5.3 Visual Studio Code

Tuir matenek nain (Huda, 2022) hateten, Visual Studio Code maka editór kódigu gratuito ne'ebé dezenvolve hosi Microsoft ne'ebé bele hala'o iha kualkér dispozitivu desktop. Ninja karakterística no estensaun sira ne'ebé komprensivu halo editór kódigu ida-ne'e sai opsaun di'ak liu ba dezenvolvedor sira. Visual Studio Code suporta kuaze sistema operasaun hotu-hotu, inklui Windows, Mac OS no Linux.

2.5.4 Balsamiq

Mak *software* ida ne'ebe uju hodi deseñu interfase Sistema ninia, maibe nia interasaun ho lingunjen programasaun sira. (Remy Sanjaya et al., 2022) *Balsamiq* kria deseñu nia oin *Interface* ne'ebe husi aplikasaun *Balsamiq mockup* ho utiliza metodu *user centres desing* no rezultadu husi kria deseñu hodi hamosu *interface* tuir nesesidade Sistema.

KAPITULU III

METODOLOJIA PESKIZA

3.1 Metologia Rekoilamentus dadus

Iha metodologia peskiza dadus ida ne'e rekolla dadus liu husi pontus mak hanesan tuir mai ne'e :

a. Metodologia Literatura Dizital

Peskizadór hala'o rekolla dadus informasaun bazeia komputadór, *artikel*, livru no jurnal sai hanesan dadus referénsia ne'ebé iha ligasaun ho sistema teste ba Numeru **Binario** no **Numeru Par** no **Inpar**, referensia hirak ne'e peskizadór bele forma dadus sira tuir teoria ne'ebé iha hodi dezenvolve peskizadór nia título refere.

3.2 Metodologia Dezenvolvimentu Sistema

Metodologia ne'ebe hakerek nain sita hodi hodi desenvelope Sistema ne'e mak modelo waterfall no hanesan:

a. Requezitos Sistema

Define objetivu no requisitu sira husi projetu, determina saida mak hakarak atinji ho projetu no problema sira ne'ebé projetu tenke resolve, Identifika mos rekursu sira ne'ebé presiza, tempu implementasaun (prazu) no orzamentu. Fase ida ne'e importante tebes ba susesu projetu, tanba se requisitu sira la klaru, projetu bele falla.

b. Dezeñu Sistema

Dezenhu detalhadu ba projetu, inklui diagrama fluxo no diagramas prosesu, define oinsá produtu sei harii, komponente saida mak uza no oinsá komponente sira ne'e sei integra malu, determina mos requisitu *hardware* no *software* ne'ebé presiza ba projetu.

c. Implementasaun

Implementasaun hanesan fase ida ne'ebe halo teste unitáriu, teste integrasaun no teste aseitasaun husi utilizadór, verifika se *software* ka *hardware* ne'ebé dezenvolve ona tuir parametro sira ne'ebé define iha fase requisitu, ka korrije erru sira se presiza.

d. Teste

Teste hanesan fase kontrolu kualidade (*Quality Assurance – QA testers*) ne’ebé buka no identifika erru sira hotu antes tama ba fase Dezenvolvimentu. Prosesu hotu-hotu tenke halo dokumentasaun.

e. Dezenvolvimentu

Dezenvolvimentu hanesan fase ne’ebé mak momentu ida iha metodolojia *Waterfall* ne’ebé halo teste ba produtu ho cliente sira ne’ebé mak utilizadór final.

f. Manutensuan

Hahu husi entrega *software* ba funzionariu hodi hahu halo operasaun, iha tempu operasionál, *software* sei bele mosu mós *erru*, tanba ne’e *software* refere sei presiza prosesu (Manutensaun) husi tempu ba tempu.

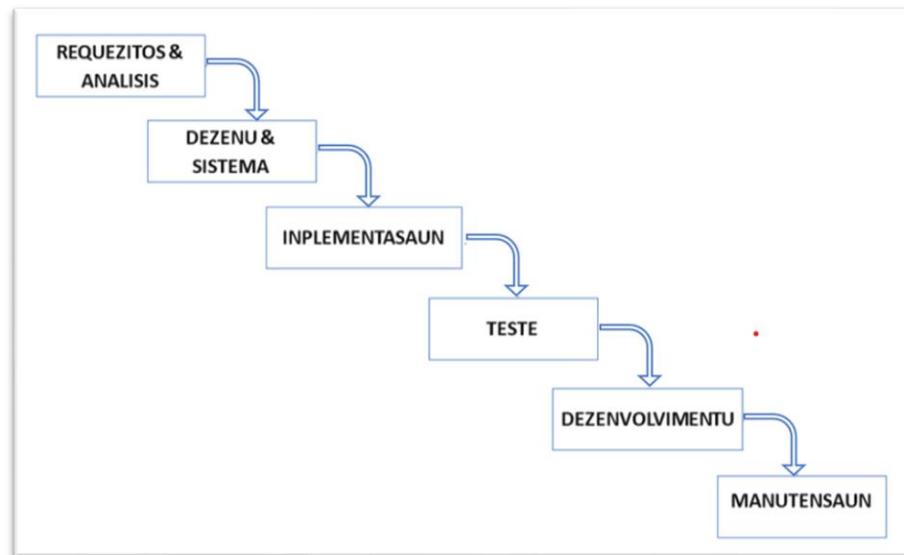


Figura 3.1: Modelo Wetarfall

3.3 Analiza Nesesidade Sistema

Tuir sistemátiku ba sistema validasaun husi numeru matematika tuir etapa hodi la’o to’o nia objetivu maka hanesan *Input, Proses* no *output*. Iha etapa tolu ne’e mak sai faze fundamental ba sistema validador, mak peskizadór utiliza iha sistema “**analiza pola pola numeru ou code utiliza automata**”.

a. *Input*

Input mak hanesan etapa dahuluk rekolla dadus hodi *input* dadus liu husi *keyboard* mak hanesan dadus transfere informasaun ba sistema Apoio Desizaun.

b. *Proses*

Proses mak etapa daruak hodi halo operasaun ka kontrola dadus sira atu nune'e bele fó sai nia rezultadu.

c. *Output*

Output mak etapa datoluk atu fó sai rezultadu final husi dadus mak prosesu ona sistus online.

Tabela 3. 1 Ekipamentu Peskizador Nian

<i>Hardware</i>	<i>Software</i>
<ul style="list-style-type: none">• <i>Lenovo</i>• <i>Memory Ram 8.00 GB</i>• <i>Hard Disk 1 TB</i>• <i>Keyboard</i>• <i>Mouse</i>	<ul style="list-style-type: none">• Sistema Operasaun <i>Windows 11 pro</i>• <i>Xampp V7.4.4</i>• <i>Visio Stude Code</i>• <i>Astah Community</i>• <i>Balsamiq farware</i>

3.4 Dezenu Diagrama

3.4.1 *Use Case Diagram*

Diagrama ida ne'e atu halo deseña aplikasaun kona ba lalaok utilizadòr nian. Iha sistema ida ne'e utilizadòr ba sistema ida ne'e nia lalaok ne'ebé utilizadòr sira halo'o klaru liu bele hare'e iha fugura tuir mai ne'e.

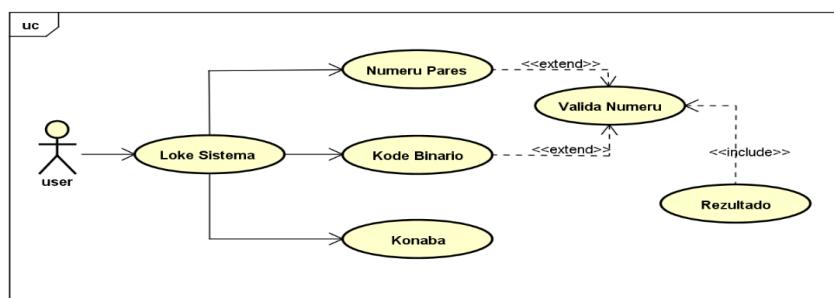


Figura 3.2. *Use Case Diagram*

3.4.2 Activity Diagram

Dezeňa lalaok aktividades iha sistema ne'ebé atu dezenvolve ne'e nia laran no oinsa atu hahu aktividade no oinsa atu remata aktividade iha sistema mak iha figura tuir mai.

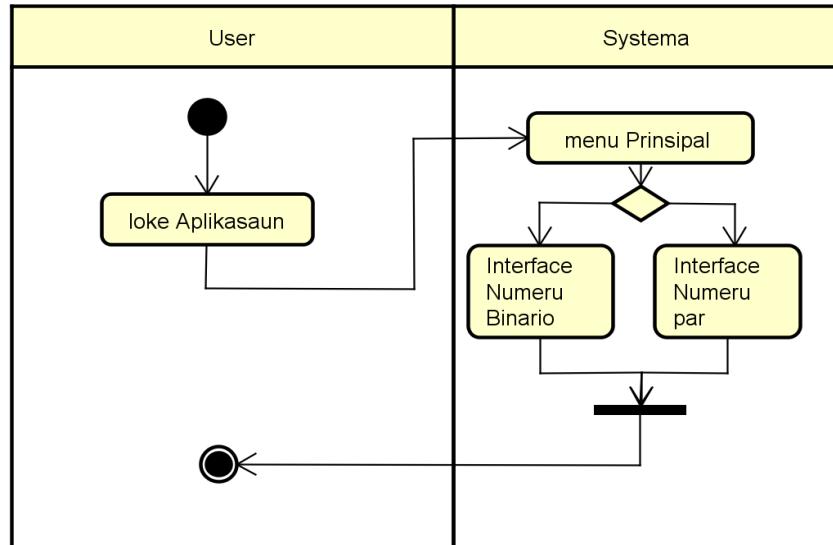


Figura 3.3 Aktivity Digrama

3.5 Dezeňu Interface

3.5.1 Dezeňu Interface Numeru Binario

Interface Validador Numeru Binario hanesa *interface fron-end* husi sistema ne'ebe user hodi asesu ba valida dadus, ho modelu dezeňu interface hanesan tuir mai ne'e:

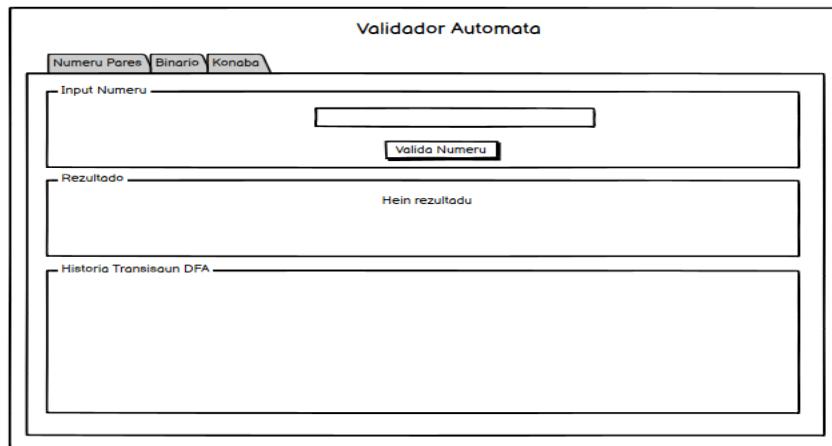


Figura 3.4, Interface Numeru Binario

3.5.2 Dezeñu Interface Numeru Par

Interface Validador Numeru Par hanesa *interface fron-end* husi sistema ne'ebe user hodi asesu ba valida dadus, ho modelu deseñu interface hanesan tuir mai ne'e:

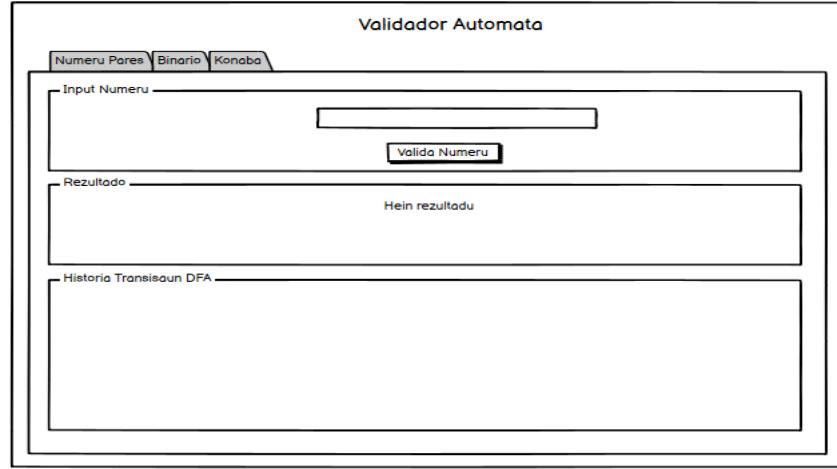


Figura 3.5, Interface Numeru Par

3.5.3 Dezeñu Interface Numeru Konaba

Interface konaba husi sistema ne'ebe user hodi asesu hodi hatene Informasaun konaba sistema, ho modelu deseñu interface hanesan tuir mai ne'e:

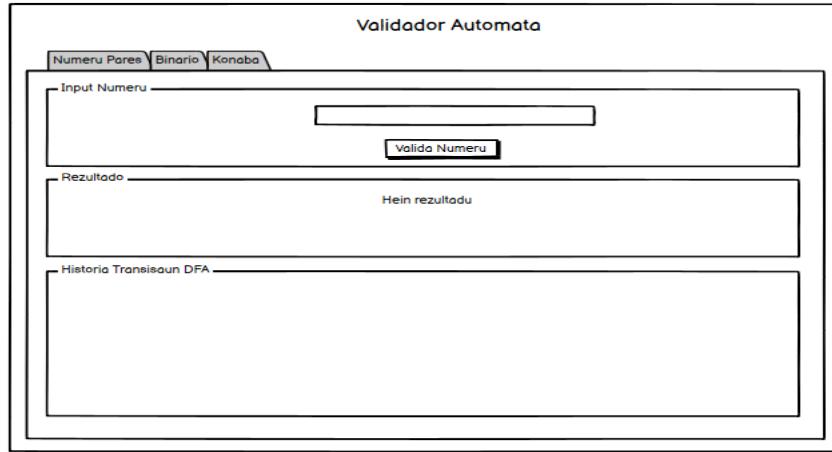


Figura 3.6, Interface Konaba

KAPITULU IV

REZULTADU PESKIZA

4.1 Interface Validator Kode Binario

Pagina **interface prinsipál** husi aplikasaun desktop *Validator Automata*, ne’ebé dezenvolve uza konseitu **Finite State Automata (DFA)** atu valida númeru kode bináriu.

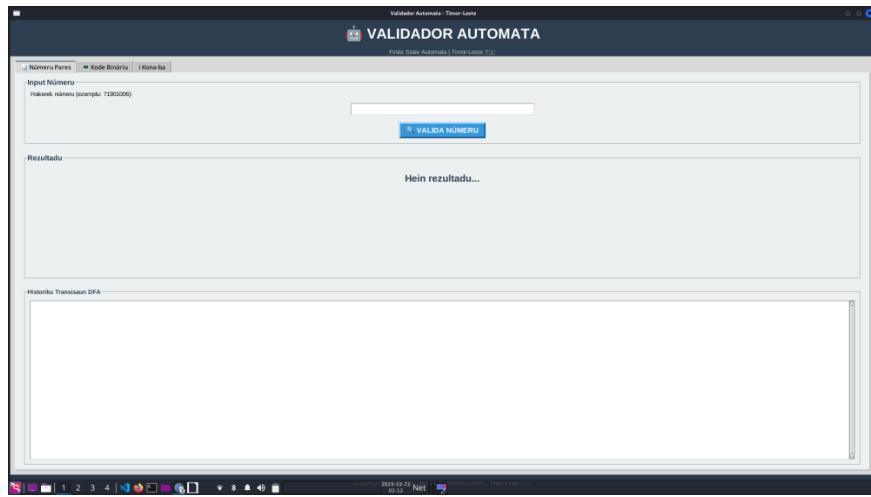


Figura 5.1: Interface Aplikasaun Validator Kode Binario

```
class DFAKodeBinariu:  
    """Deterministic Finite Automaton ba valida kode bináriu"""  
  
    def __init__(self):  
        self.estado_atual = 'start'  
        self.historiku = []  
  
    def reset(self):  
        """Reset ba estadu inisiál"""  
        self.estado_atual = 'start'  
        self.historiku.clear()  
  
    def transisaun(self, karakter):  
        """Aplika funsaun transisaun"""
```

```

estado_anterior = self.estado_atual

if self.estado_atual == 'rejeita':
    pass # Trap state
elif karakter in '01':
    self.estado_atual = 'válidu'
else:
    self.estado_atual = 'rejeita'
    self.historiku.append({
        'karakter': karakter,
        'estado_anterior': estado_anterior,
        'estado_foun': self.estado_atual
    })
return self.estado_atual

def prosesa(self, kode):
    """Prosesa string kode kompletu"""
    self.reset()
    if not kode:
        return False, "String mamuk la válidu"
    for karakter in kode:
        self.transisaun(karakter)
    if self.estado_atual == 'rejeita':
        return False, f"Karakter inválidu: '{karakter}'"
    resultado = self.estado_atual == 'válidu'
    return resultado, f"Total bit: {len(kode)}"
```

4.1.1 Interface Validador Erado

Pagina **resultadu prosesu validasaun** bainhira utilizadór input númeru laos kode binario iha aplikasaun *Validador Automata*, sei erro hanesan figura kraik.

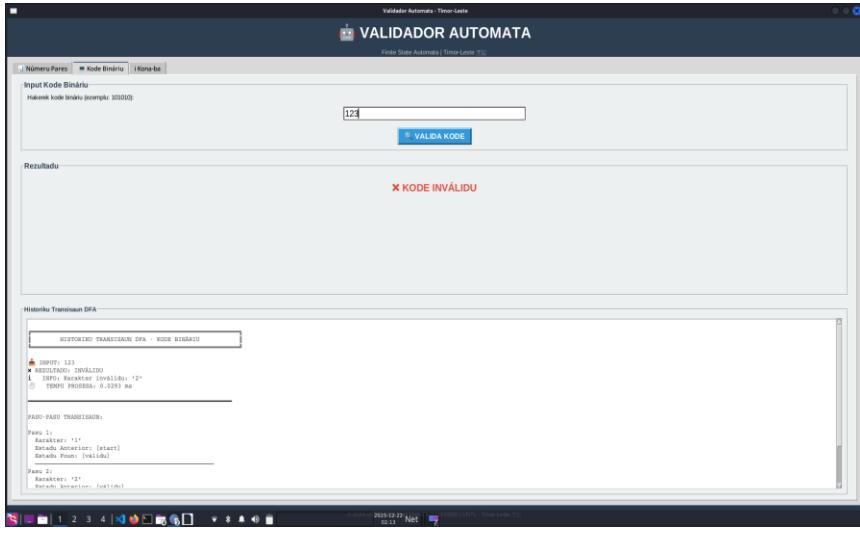


Figura 5.2: Interface Validador kode Binario Erado

4.1.2 Interface Validador Korektu

Pagina **resultadu prosesu validasaun** bainhira utilizadór input kode binario los iha aplikasaun *Validador Automata*, korektu hanesan figura kraik.

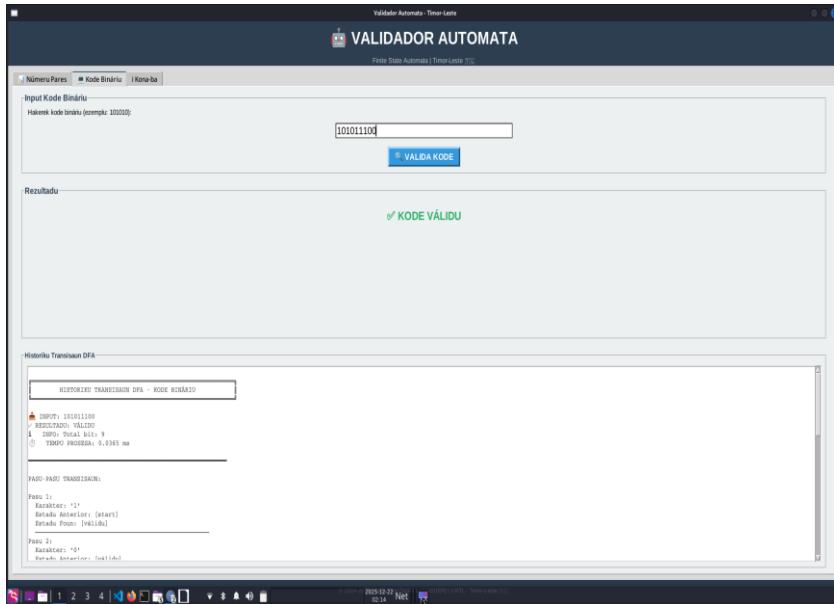


Figura 5.3: Interface Validador kode Binario Erado

4.2 Interface Validator Numeru Par

Pagina **interface prinsipál** husi aplikasaun desktop *Validator Automata*, ne’ebé dezenvolve uza konseitu **Finite State Automata (DFA)** atu valida número **Par**.

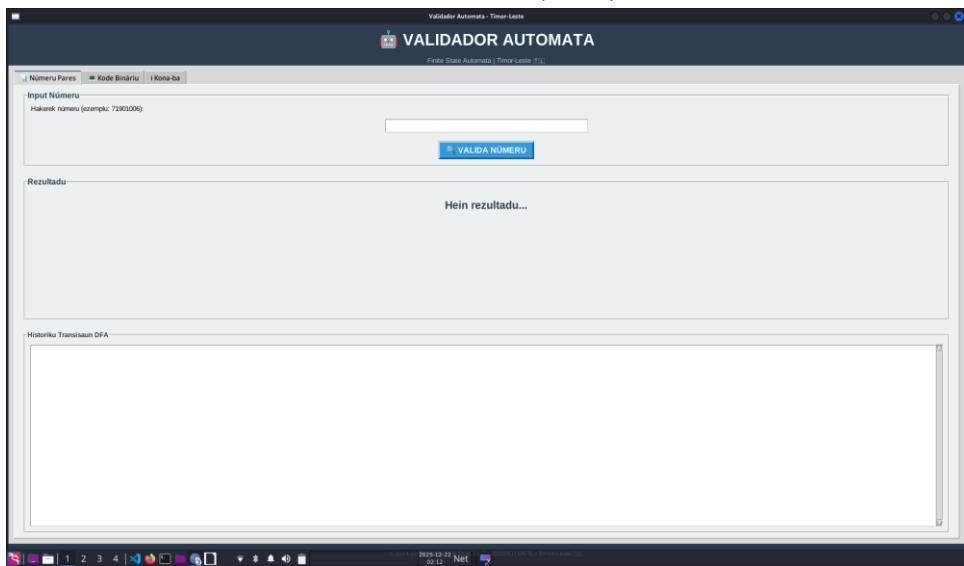


Figura 5.4: Interface Aplikasaun Validator Kode Binario

```
class DFANumeruPares:  
    """Deterministic Finite Automaton ba deteta número pares"""  
  
    def __init__(self):  
        self.estado_atual = 'par'  
        self.historiku = []  
  
    def reset(self):  
        """Reset ba estadu inisiál"""  
        self.estado_atual = 'par'  
        self.historiku.clear()  
  
    def transisaun(self, digitu):  
        """Aplika funsaun transisaun"""  
        estado_anterior = self.estado_atual  
        if digitu in '02468':
```

```

self.estado_atual = 'par'
elif digitu in '13579':
    self.estado_atual = 'ímpar'
else:
    raise ValueError(f'Dígitu inválidu: {digitu}')
self.historiku.append({
    'digitu': digitu,
    'estado_anterior': estado_anterior,
    'estado_foun': self.estado_atual
})
return self.estado_atual

def prosesa(self, numero):
    """Prosesa string número kompletu"""
    self.reset()
    if not numero:
        return True, "String mamuk konsidera par"
    for digitu in numero:
        if not digitu.isdigit():
            raise ValueError(f'Input la'ós número: {numero}')
        self.transisaun(digitu)
    resultado = self.estado_atual == 'par'
    return resultado, f'Dígitu ikus: {numero[-1]}'

```

4.1.3 Interface Validador Erado

Pagina **resultadu prosesu validasaun** bainhira utilizadór input laos número par iha aplikasaun *Validador Automata*, sei erro hanesan figura kraik.

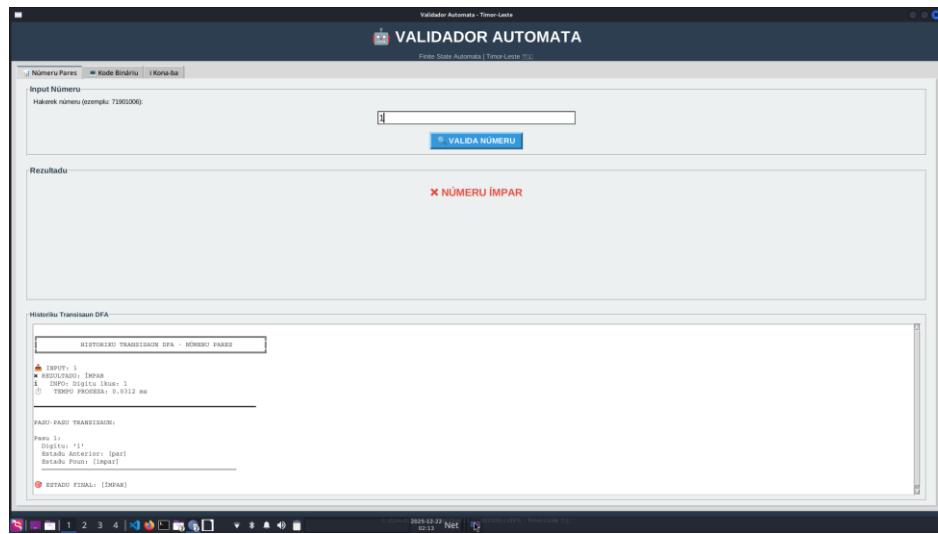


Figura 5.5: Interface Aplikasaun Validador Laos Numaru Par

4.1.4 Interface Validador Korektu

Pagina **resultadu prosesu validasaun** bainhira utilizadór input número par iha aplikasaun *Validador Automata*, sei lamosu error hanesan figura kraik.

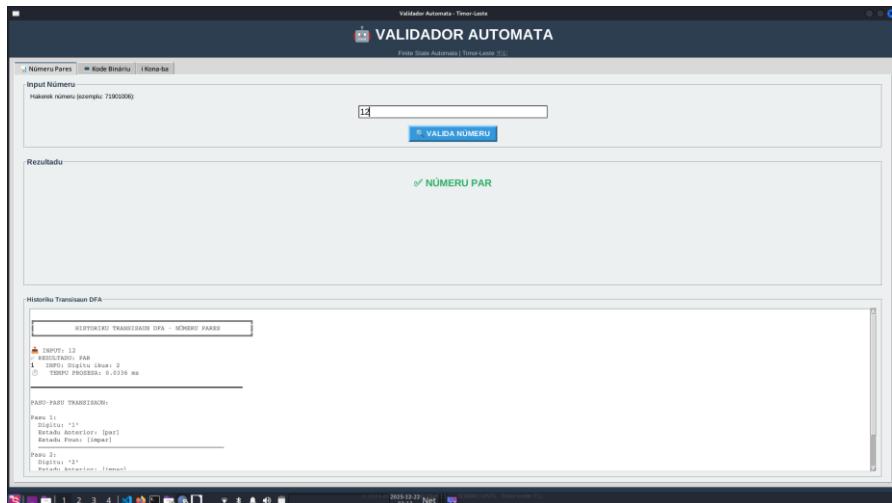


Figura 5.6: Interface Aplikasaun Validador Numaru Par

4.3 Interface Validador Konaba

Pagina **Konaba** bainhira utilizadór atu atiliza presija le'e uluk informasaun sira konba funzionamentu hui Sistema automata hanesan figura kraik.

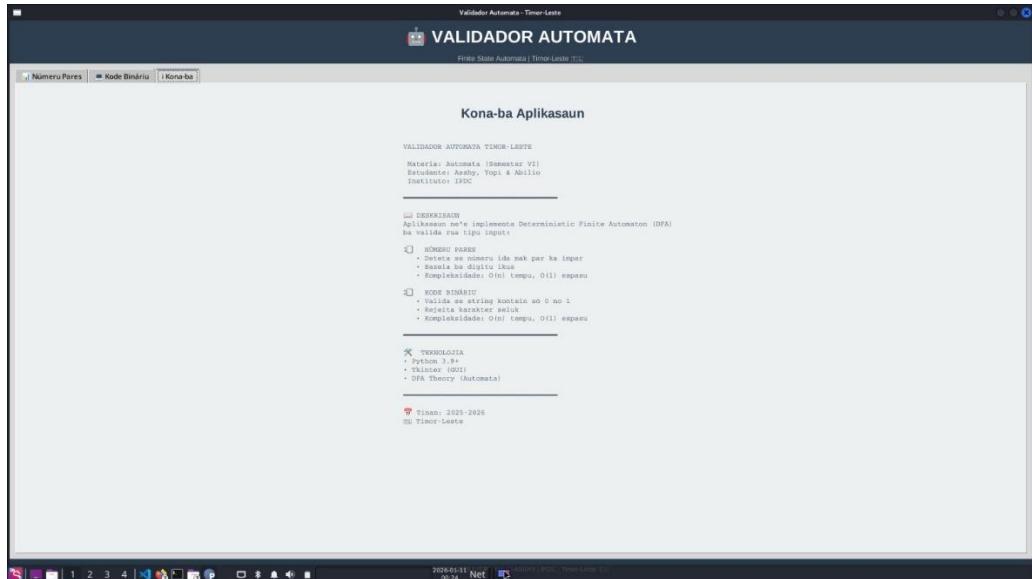


Figura: 5.7 Interface Konaba iha aplikasaun validador

```
def kria_tab_kona(self):  
    """Kria tab informasaun kona-ba aplikasaun""""  
  
    # Frame Konteúdu  
  
    content_frame = tk.Frame(self.tab_kona, bg=self.cor_background)  
    content_frame.pack(fill=tk.BOTH, expand=True, padx=30, pady=30)  
  
    # Título  
  
    tk.Label(  
        content_frame,  
        text=" Kona-ba Aplikasaun",  
        font=("Arial", 18, "bold"),  
        bg=self.cor_background,  
        fg=self.cor_primaria  
    ).pack(pady=10)  
  
    # Informasaun  
  
    info_text = """"
```

VALIDADOR AUTOMATA TIMOR-LESTE

Materia: Automata (Semester VI)

Estudante: Asshy, Yopi & Abilio

Instituto: IPDC

DESKRISAUN

Aplikasaun ne'e implementa Deterministic Finite Automaton (DFA)

ba valida rua tipu input:

1 NÚMERU PARES

- Deteta se número ida mak par ka ímpar
- Bazeia ba dígitu ikus
- Kompleksidade: $O(n)$ tempu, $O(1)$ espasu

2 KODE BINÁRIU

- Valida se string kontein só 0 no 1
 - Rejeita karakter seluk
 - Kompleksidade: $O(n)$ tempu, $O(1)$ espasu
-
-

TEKNOLOJIA

- Python 3.9+
 - Tkinter (GUI)
 - DFA Theory (Automata)
-
-

Tinan: 2025-2026

TL Timor-Leste

.....

```
tk.Label(  
    content_frame,  
    text=info_text,  
    font=("Courier", 10),  
    bg=self.cor_background,  
    fg=self.cor_primaria,  
    justify=tk.LEFT  
).pack(pady=10)  
  
def kria_footer(self):  
    """Kria footer aplikasaun"""  
    footer_frame = tk.Frame(self.root, bg=self.cor_primaria, height=40)  
    footer_frame.pack(fill=tk.X, side=tk.BOTTOM)  
    footer_frame.pack_propagate(False)  
  
    footer_text = "© 2025-2026 | YOPI ABILIO ASSHY | IPDC - Timor-Leste TL"
```

KAPITULU V

MAKTAKA

5.1 Konklusaun

Estudu ida ne'e hakerek nain foti konklusaun uza **Automata** hodi analiza pola numeru facilita tebes prosesu validasaun ba kódigu kompleksu, sistema validador ne'ebé dezenvolve ona hatudu katak hodi identifika numeru **Par no Bináriu**, ita la presiza kalkulasaun naruk, maibé presiza de'it define estadu (*states*), tranzisaun ne'ebé lójiku, ne'ebé prova katak automata mak fundasaun ne'ebé forte ba dezenvolvimentu *compiler* no seguransa dadus iha mundu teknolojia informasaun.

Bazeia ba rezultadu analiza, dezenvolvimentu sistema ne'ebé halo ona, bele konklui hanesan tuir mai ne'e:

1. Implementasaun *Finite State Automata* (FSA) konsege deteta loos ba pola numeru nian, automata sai hanesan métodu ne'ebé eficiente tebes hodi halo validasaun ba numeru sira ne'ebé komplesu e específica.
2. Sistema ne'e susesu *input* numeru binario 0 no 1, se *input* ne'e iha número seluk, automata sei hatudu informasaun *error state*, ne'ebé hatudu katak kódigu ne'e la válidu.
3. Sistema ne'ebé kria ona konsege identifika numeru par liu husi analiza numeru par nian hanesan (2, 4, 6, 8, nst..) uza tranzisaun estadu (*state*) ne'ebé simples.

Rezultadu sitema hatudu teste lójika matemática husi automata bele transforma ba kódigu programasaun hodi rezolve problema komplesu matematika nian.

5.2 Sugestaun

Bazeia ba sistema ida ne'e hakerek nian sujere ba hirak ne'ebé foti Sistema Validasaun liliu relasiona ho titlu ida ne'e maka hanesan tuir mai ne'e:

- a) Kontinua desezenvolve foti Sistema Validasaun ba Fale.
- b) Utiliza metodu sira seluk hodi reforsa Sistema Validasaun.

BIBLIOGRAFIA

- Agung Yulyianto Nugroho, S. M. (2024). PENGANTAR DATA SCIENCE TERMASUK APLIKASI PRAKTIS DAN STUDI KASUS. *CV BRAVO PRESS INDONESIA*.
- Akmal, M. H. (2025). Analisis Data Biologis dalam Mengidentifikasi Gen atau Protein yang . *Pustakan data*.
- Fisdiovany, D., Misdiyanto, & Hikmah, N. (2025). Rancang Bangun Aplikasi Vape Shop Dengan. *Jurnal ProTekInfo*.
- Huda, N. (2022). Pengertian, fitur, keunggulan dan Jenisnya. *dewaweb*.
- In, t. (2024). <https://www.bytesin.com/software/Astah-Community/>.
- Mansyur, F., & Ridho, F. (2020). ANALISIS POLA PERMINTAAN PUBLIKASI DATA. *Kumpulan jurnal Ilmu Komputer (KLIK)*.
- Purnomo, Kosala Dwidja; Anggraeni, Irma Dwi ; Riski, Abduh;. (2023). Analisis Pola Harga Saham dengan Modifikasi Metode Eksponen Hurst. *Emerging Statistics and Data Science Journal*.
- ramadan, w. c. (2025). MICROSOFT OFFICE 2010 . *wiki loves*.
- S. M. (2026). Analisis data, visualisasi dan pembelajaran mesin. *Phyton untuk data scince*.
- Sttawan, H. (2018). Pengertian automata dan fungsinya. *Garuda indonesia*.
- Suci Intania Indah, S. I., & Voutama, A. (2025). Analisis Pendapatan dan Pola Pembatalan Reservasi Hotel. *Jurnal Informasi, Sains dan Teknologi*.
- Valencia, S., & Atmojo, W. T. (2024). Analisis Pola Pembelian pada Data Penjualan CanNgopi. *Jurnal Komtika*, Vol. 8 No. 2.