To do:

* Reverse engineer interview transcript and add to appendix
* Create hardware requirements and justifications. Then strengthen existing justifications. Use a table
* Keep record of old contents page then new one. This will let you tell her where the page numbers have changed.
* Update contents page
* Draft new pdf
* Email miss with changes and their page numbers, the new pdf and any changes to page numbers.

Old contents

Contents

[My Final Product: 2](#_Toc130511253)

[Glossary of Key Terms: 10](#_Toc130511254)

[Analysis Section 12](#_Toc130511255)

[Problem definition: 12](#_Toc130511256)

[What are the benefits to solving this problem with a computational approach? 12](#_Toc130511257)

[My computational approach: 12](#_Toc130511258)

[Thinking ahead: 13](#_Toc130511259)

[Inputs: 13](#_Toc130511260)

[Outputs: 14](#_Toc130511261)

[Preconditions: 15](#_Toc130511262)

[Thinking abstractly: 15](#_Toc130511263)

[Thinking procedurally / decomposition 16](#_Toc130511264)

[Thinking concurrently: 18](#_Toc130511265)

[Stake Holders: 18](#_Toc130511266)

[Research: 20](#_Toc130511267)

[Essential Features: 27](#_Toc130511268)

[Limitations: 28](#_Toc130511269)

[Solution Requirements: 29](#_Toc130511270)

[Success Criteria: 30](#_Toc130511271)

[Design: 34](#_Toc130511272)

[Initial Design for the system’s features and user interface: 34](#_Toc130511273)

[Breaking down the problem into smaller components as well as explaining and justifying the structure of the solution. 35](#_Toc130511274)

[The title / menu page: 45](#_Toc130511275)

[The pregame config form page: 45](#_Toc130511276)

[The chess game play page: 45](#_Toc130511277)

[The game over pop-up: 45](#_Toc130511278)

[Usability features to be used in the solution: 46](#_Toc130511279)

[Effective 46](#_Toc130511280)

[Efficient 47](#_Toc130511281)

[Error tolerant: 48](#_Toc130511282)

[Easy to learn 48](#_Toc130511283)

[Engaging 49](#_Toc130511284)

[Further Problem Analysis At a Technical Level: 50](#_Toc130511285)

[Design Objectives: 52](#_Toc130511286)

[Some changes to the design: 54](#_Toc130511287)

[Usability Features to be Used in the Solution: 57](#_Toc130511288)

[Effective 57](#_Toc130511289)

[Efficient: 57](#_Toc130511290)

[Error Tolerant: 57](#_Toc130511291)

[Easy to Learn and Engaging: 57](#_Toc130511292)

[Proposed Solution Description: 58](#_Toc130511293)

[Front End: 58](#_Toc130511294)

[Server Side API Connection and Webserver: 59](#_Toc130511295)

[GUI to communicate. 59](#_Toc130511296)

[Chess Computer: 60](#_Toc130511297)

[Benefit of Decomposition: 60](#_Toc130511298)

[Diagrams of propose solution: 61](#_Toc130511299)

[Decomposition of Chess Computer (not algorithms just functions, classes and data structures) 64](#_Toc130511300)

[Decomposition of GUI logic and API client server connection: 65](#_Toc130511301)

[Chess Computer: All major algorithms 67](#_Toc130511302)

[The Chess Engine / Chess Functions: 67](#_Toc130511303)

[The Move Engine algorithm: 76](#_Toc130511304)

[Game Manager: 81](#_Toc130511305)

[GUI logic and API client server connection: 83](#_Toc130511306)

[Describe database use (save games and minimax cache) and other persistent storage: 91](#_Toc130511307)

[Variables and key data structures used in these key algorithms: 95](#_Toc130511308)

[Complete trace tables and dry runs of key algorithms: 96](#_Toc130511309)

[Validation: 105](#_Toc130511310)

[Client side validation needed: 105](#_Toc130511311)

[Server side validation: 107](#_Toc130511312)

[Functionality that each prototype will have: 107](#_Toc130511313)

[Prototype 1: 107](#_Toc130511314)

[Prototype 2: 108](#_Toc130511315)

[Prototype 3: 109](#_Toc130511316)

[Test Plans 109](#_Toc130511317)

[Test Plan for Prototype 1: 109](#_Toc130511318)

[Test Plan for Prototype 2: 109](#_Toc130511319)

[Test Plan for prototype 3: 113](#_Toc130511320)

[Post development testing against development criteria (black box alpha testing): 115](#_Toc130511321)

[Acceptance testing for success criteria (beta testing): 118](#_Toc130511322)

[Iterative Development – Prototype 1 119](#_Toc130511323)

[Aims for this iteration 119](#_Toc130511324)

[The main aim is to create a working prototype for a system to solve a simpler problem that can be tackled in a similar way to chess. 119](#_Toc130511325)

[Functionality that the prototype will have 120](#_Toc130511326)

[Annotated code screenshots with description 121](#_Toc130511327)

[Libraries, Languages and other assets: 123](#_Toc130511328)

[The code: 123](#_Toc130511329)

[Here are some screenshots of the final product: 137](#_Toc130511330)

[Problems throughout and how I tackled them: 143](#_Toc130511331)

[Test Plan for this version 152](#_Toc130511332)

[Changes to code made after testing: 159](#_Toc130511333)

[Validation: 162](#_Toc130511334)

[Feedback from Stakeholder 162](#_Toc130511335)

[Planned changes after stakeholder feedback 163](#_Toc130511336)

[Evaluation 164](#_Toc130511337)

[Iterative Development – Prototype 2 166](#_Toc130511338)

[Aims for this iteration 166](#_Toc130511339)

[Functionality that the prototype will have 166](#_Toc130511340)

[Annotated code screenshots with description 167](#_Toc130511341)

[Testing 248](#_Toc130511342)

[Testing of Chess Engine and Move Engine: 248](#_Toc130511343)

[Testing of Console Chess: 272](#_Toc130511344)

[Validation: 275](#_Toc130511345)

[Feedback from Stakeholders 276](#_Toc130511346)

[Changes/Fixes that I now plan to make to the design or code as a result of testing and feedback 277](#_Toc130511347)

[Evaluation 277](#_Toc130511348)

[Iterative Development – Prototype 3: 279](#_Toc130511349)

[Functionality that the prototype should have: 279](#_Toc130511350)

[Annotated code screenshots with description: 279](#_Toc130511351)

[Assorted Module: 284](#_Toc130511352)

[General.py 284](#_Toc130511353)

[chess\_exceptions.py 285](#_Toc130511354)

[Chess Functions Module (Chess Engine): 286](#_Toc130511355)

[Move Engine: 287](#_Toc130511356)

[Minimax\_parallel.py 301](#_Toc130511357)

[Game Manager 321](#_Toc130511358)

[Database module: 336](#_Toc130511359)

[Create\_database.py 336](#_Toc130511360)

[Models.py 338](#_Toc130511361)

[handle\_games.py 340](#_Toc130511362)

[Schema Module: 341](#_Toc130511363)

[Socket\_shemas.py: 345](#_Toc130511364)

[Website Module: 350](#_Toc130511365)

[app.py 350](#_Toc130511366)

[Chess\_game.html: 364](#_Toc130511367)

[Main.js 368](#_Toc130511368)

[Demonstrating the end result 393](#_Toc130511369)

[Example Play 400](#_Toc130511370)

[Testing: 420](#_Toc130511371)

[Automated unit tests: 420](#_Toc130511372)

[Here are the main issues that were detected by the automated unit tests: 441](#_Toc130511373)

[Timed Minimax not working: 441](#_Toc130511374)

[Tests completed on final product: 444](#_Toc130511375)

[Knights cannot take pieces 450](#_Toc130511376)

[Pawns cannot move forward 2 places: 453](#_Toc130511377)

[Concurrent workers not caching 455](#_Toc130511378)

[Erroneous Results: 463](#_Toc130511379)

[Demonstration video of the program working: 468](#_Toc130511380)

[Stakeholder Feedback: 469](#_Toc130511381)

[GUI: 469](#_Toc130511382)

[Special Features: 469](#_Toc130511383)

[Reliability (after the fixes from my testing): 469](#_Toc130511384)

[Suggestions for improvement: 469](#_Toc130511385)

[Evaluation (just of prototype 3): 469](#_Toc130511386)

[Evaluation: Evaluative Testing: 471](#_Toc130511387)

[Post-Development Testing 471](#_Toc130511388)

[Usability Survey: 475](#_Toc130511389)

[Usability Survey Results 477](#_Toc130511390)

[Breakdown of success criteria (what has been achieved): 484](#_Toc130511391)

[Limitations and Maintenance: 488](#_Toc130511392)

[Limitations with the program itself include: 488](#_Toc130511393)

[As part of maintenance, certain issues may arise: 489](#_Toc130511394)

[Some of the key ways in which this chess program could be improved: 489](#_Toc130511395)

[Appendix: 490](#_Toc130511396)

[Sources: 490](#_Toc130511397)

[Code Listing: 491](#_Toc130511398)

[Prototype 1: 491](#_Toc130511399)

[directory structure: 491](#_Toc130511400)

[app.py 492](#_Toc130511401)

[logger.py 493](#_Toc130511402)

[game\_engine.py 495](#_Toc130511403)

[index.html 498](#_Toc130511404)

[main.js 500](#_Toc130511405)

[Style.css 507](#_Toc130511406)

[Prototype 2: Vue page 508](#_Toc130511407)

[ChessGame.vue 508](#_Toc130511408)

[external.js 513](#_Toc130511409)

[Prototype 2: Final Code 513](#_Toc130511410)

[Directory Structure: 513](#_Toc130511411)

[assorted.py 514](#_Toc130511412)

[vector.py 515](#_Toc130511413)

[pieces.py 516](#_Toc130511414)

[board\_state.py 527](#_Toc130511415)

[game.py 532](#_Toc130511416)

[console\_chess.py 536](#_Toc130511417)

[test\_vector.py 538](#_Toc130511418)

[test\_pieces.py 539](#_Toc130511419)

[test\_board\_state.py 541](#_Toc130511420)

[test\_minimax.py 546](#_Toc130511421)

[test\_data\vector\from\_square.yaml 556](#_Toc130511422)

[test\_data\vector\vector\_add.yaml 557](#_Toc130511423)

[test\_data\vector\vector\_in\_board.yaml 558](#_Toc130511424)

[test\_data\vector\vector\_multiply.yaml 559](#_Toc130511425)

[test\_data\vector\vector\_to\_square.yaml 559](#_Toc130511426)

[test\_data\pieces\test\_board\_populated.yaml 560](#_Toc130511427)

[test\_data\pieces\test\_empty\_board.yaml 562](#_Toc130511428)

[test\_data\board\_state\color\_in\_check.yaml 563](#_Toc130511429)

[test\_data\board\_state\game\_over.yaml 566](#_Toc130511430)

[test\_data\board\_state\generate\_all\_pieces.yaml 568](#_Toc130511431)

[test\_data\board\_state\generate\_legal\_moves.yaml 569](#_Toc130511432)

[test\_data\board\_state\generate\_pieces\_of\_color.yaml 571](#_Toc130511433)

[test\_data\board\_state\piece\_at\_vector.yaml 572](#_Toc130511434)

[Prototype 3: Final code 573](#_Toc130511435)

[Directory Structure 573](#_Toc130511436)

[\_\_init\_\_.py 576](#_Toc130511437)

[app.py 576](#_Toc130511438)

[assorted\\_\_init\_\_.py 577](#_Toc130511439)

[assorted\chess\_exceptions.py 577](#_Toc130511440)

[assorted\general.py 578](#_Toc130511441)

[assorted\safe\_hash.py 578](#_Toc130511442)

[chess\_functions\\_\_init\_\_.py 579](#_Toc130511443)

[chess\_functions\vector.py 579](#_Toc130511444)

[chess\_functions\pieces.py 581](#_Toc130511445)

[chess\_functions\board\_state.py 596](#_Toc130511446)

[chess\_functions\test\_fast\_vector.py 605](#_Toc130511447)

[chess\_functions\test\_fast\_pieces.py 606](#_Toc130511448)

[chess\_functions\test\_fast\_board\_state.py 608](#_Toc130511449)

[chess\_functions\test\_data\vector\from\_square.yaml 618](#_Toc130511450)

[chess\_functions\test\_data\vector\vector\_add.yaml 619](#_Toc130511451)

[chess\_functions\test\_data\vector\vector\_in\_board.yaml 619](#_Toc130511452)

[chess\_functions\test\_data\vector\vector\_multiply.yaml 620](#_Toc130511453)

[chess\_functions\test\_data\vector\vector\_to\_square.yaml 621](#_Toc130511454)

[chess\_functions\test\_data\pieces\test\_board\_populated.yaml 621](#_Toc130511455)

[chess\_functions\test\_data\pieces\test\_empty\_board.yaml 624](#_Toc130511456)

[chess\_functions\test\_data\board\_state\color\_in\_check.yaml 625](#_Toc130511457)

[chess\_functions\test\_data\board\_state\game\_over.yaml 628](#_Toc130511458)

[chess\_functions\test\_data\board\_state\generate\_all\_pieces.yaml 630](#_Toc130511459)

[chess\_functions\test\_data\board\_state\generate\_legal\_moves.yaml 630](#_Toc130511460)

[chess\_functions\test\_data\board\_state\generate\_pieces\_of\_color.yaml 633](#_Toc130511461)

[chess\_functions\test\_data\board\_state\piece\_at\_vector.yaml 634](#_Toc130511462)

[chess\_game\\_\_init\_\_.py 635](#_Toc130511463)

[chess\_game\console\_chess.py 635](#_Toc130511464)

[chess\_game\game\_web.py 638](#_Toc130511465)

[chess\_game\game.py 645](#_Toc130511466)

[chess\_game\test\_fast\_game.py 652](#_Toc130511467)

[database\\_\_init\_\_.py 656](#_Toc130511468)

[database\create\_database.py 656](#_Toc130511469)

[database\handle\_games.py 657](#_Toc130511470)

[database\models.py 659](#_Toc130511471)

[move\_engine\\_\_init\_\_.py 660](#_Toc130511472)

[move\_engine\cache\_managers.py 661](#_Toc130511473)

[move\_engine\minimax\_parallel.py 670](#_Toc130511474)

[move\_engine\minimax.py 689](#_Toc130511475)

[move\_engine\parallel\_minimax\_testing.py 700](#_Toc130511476)

[move\_engine\test\_slow\_timed\_minimax\_engine.py 706](#_Toc130511477)

[schemas\\_\_init\_\_.py 723](#_Toc130511478)

[schemas\cache\_item\_schema.py 723](#_Toc130511479)

[schemas\socket\_schemas.py 726](#_Toc130511480)

[website\\_\_init\_\_.py 729](#_Toc130511481)

[website\flask\_server.py 729](#_Toc130511482)

[website\templates\chess\_game.html 740](#_Toc130511483)

[website\static\main.js 744](#_Toc130511484)

[website\static\chess\_class.js 752](#_Toc130511485)

[website\static\vector.js 762](#_Toc130511486)

[website\static\initial\_game\_data.js 763](#_Toc130511487)

[website\static\style.css 763](#_Toc130511488)

[database\database.db 768](#_Toc130511489)

New contents

Contents

[My Final Product: 2](#_Toc134652882)

[Glossary of Key Terms: 10](#_Toc134652883)

[Analysis Section 12](#_Toc134652884)

[Problem definition: 12](#_Toc134652885)

[What are the benefits to solving this problem with a computational approach? 12](#_Toc134652886)

[My computational approach: 12](#_Toc134652887)

[Thinking ahead: 13](#_Toc134652888)

[Inputs: 13](#_Toc134652889)

[Outputs: 14](#_Toc134652890)

[Preconditions: 15](#_Toc134652891)

[Thinking abstractly: 15](#_Toc134652892)

[Thinking procedurally / decomposition 16](#_Toc134652893)

[Thinking concurrently: 18](#_Toc134652894)

[Stake Holders: 18](#_Toc134652895)

[Research: 20](#_Toc134652896)

[I then conducted an interview with Roger: 25](#_Toc134652897)

[Essential Features: 27](#_Toc134652898)

[Limitations: 28](#_Toc134652899)

[Solution Requirements: 29](#_Toc134652900)

[Software Requirements: 29](#_Toc134652901)

[Hardware Requirements: 30](#_Toc134652902)

[Other External Services and Dependencies: 31](#_Toc134652903)

[Success Criteria: 32](#_Toc134652904)

[Design: 36](#_Toc134652905)

[Initial Design for the system’s features and user interface: 36](#_Toc134652906)

[Breaking down the problem into smaller components as well as explaining and justifying the structure of the solution. 37](#_Toc134652907)

[The title / menu page: 48](#_Toc134652908)

[The pregame config form page: 48](#_Toc134652909)

[The chess game play page: 48](#_Toc134652910)

[The game over pop-up: 48](#_Toc134652911)

[Usability features to be used in the solution: 49](#_Toc134652912)

[Effective 49](#_Toc134652913)

[Efficient 50](#_Toc134652914)

[Error tolerant: 51](#_Toc134652915)

[Easy to learn 51](#_Toc134652916)

[Engaging 52](#_Toc134652917)

[Further Problem Analysis At a Technical Level: 53](#_Toc134652918)

[Design Objectives: 55](#_Toc134652919)

[Some changes to the design: 57](#_Toc134652920)

[Usability Features to be Used in the Solution: 60](#_Toc134652921)

[Effective 60](#_Toc134652922)

[Efficient: 60](#_Toc134652923)

[Error Tolerant: 60](#_Toc134652924)

[Easy to Learn and Engaging: 60](#_Toc134652925)

[Proposed Solution Description: 61](#_Toc134652926)

[Front End: 61](#_Toc134652927)

[Server Side API Connection and Webserver: 62](#_Toc134652928)

[GUI to communicate. 62](#_Toc134652929)

[Chess Computer: 63](#_Toc134652930)

[Benefit of Decomposition: 63](#_Toc134652931)

[Diagrams of propose solution: 64](#_Toc134652932)

[Decomposition of Chess Computer (not algorithms just functions, classes and data structures) 67](#_Toc134652933)

[Decomposition of GUI logic and API client server connection: 68](#_Toc134652934)

[Chess Computer: All major algorithms 70](#_Toc134652935)

[The Chess Engine / Chess Functions: 70](#_Toc134652936)

[The Move Engine algorithm: 79](#_Toc134652937)

[Game Manager: 84](#_Toc134652938)

[GUI logic and API client server connection: 86](#_Toc134652939)

[Describe database use (save games and minimax cache) and other persistent storage: 94](#_Toc134652940)

[Variables and key data structures used in these key algorithms: 98](#_Toc134652941)

[Complete trace tables and dry runs of key algorithms: 99](#_Toc134652942)

[Validation: 108](#_Toc134652943)

[Client side validation needed: 108](#_Toc134652944)

[Server side validation: 110](#_Toc134652945)

[Functionality that each prototype will have: 110](#_Toc134652946)

[Prototype 1: 110](#_Toc134652947)

[Prototype 2: 111](#_Toc134652948)

[Prototype 3: 112](#_Toc134652949)

[Test Plans 112](#_Toc134652950)

[Test Plan for Prototype 1: 112](#_Toc134652951)

[Test Plan for Prototype 2: 112](#_Toc134652952)

[Test Plan for prototype 3: 116](#_Toc134652953)

[Post development testing against development criteria (black box alpha testing): 118](#_Toc134652954)

[Acceptance testing for success criteria (beta testing): 121](#_Toc134652955)

[Iterative Development – Prototype 1 122](#_Toc134652956)

[Aims for this iteration 122](#_Toc134652957)

[The main aim is to create a working prototype for a system to solve a simpler problem that can be tackled in a similar way to chess. 122](#_Toc134652958)

[Functionality that the prototype will have 123](#_Toc134652959)

[Annotated code screenshots with description 124](#_Toc134652960)

[Libraries, Languages and other assets: 126](#_Toc134652961)

[The code: 126](#_Toc134652962)

[Here are some screenshots of the final product: 140](#_Toc134652963)

[Problems throughout and how I tackled them: 146](#_Toc134652964)

[Test Plan for this version 155](#_Toc134652965)

[Changes to code made after testing: 162](#_Toc134652966)

[Validation: 165](#_Toc134652967)

[Feedback from Stakeholder 165](#_Toc134652968)

[Planned changes after stakeholder feedback 166](#_Toc134652969)

[Evaluation 167](#_Toc134652970)

[Iterative Development – Prototype 2 169](#_Toc134652971)

[Aims for this iteration 169](#_Toc134652972)

[Functionality that the prototype will have 169](#_Toc134652973)

[Annotated code screenshots with description 170](#_Toc134652974)

[Testing 251](#_Toc134652975)

[Testing of Chess Engine and Move Engine: 251](#_Toc134652976)

[Testing of Console Chess: 275](#_Toc134652977)

[Validation: 278](#_Toc134652978)

[Feedback from Stakeholders 279](#_Toc134652979)

[Changes/Fixes that I now plan to make to the design or code as a result of testing and feedback 280](#_Toc134652980)

[Evaluation 280](#_Toc134652981)

[Iterative Development – Prototype 3: 282](#_Toc134652982)

[Functionality that the prototype should have: 282](#_Toc134652983)

[Annotated code screenshots with description: 282](#_Toc134652984)

[Assorted Module: 287](#_Toc134652985)

[General.py 287](#_Toc134652986)

[chess\_exceptions.py 288](#_Toc134652987)

[Chess Functions Module (Chess Engine): 289](#_Toc134652988)

[Move Engine: 290](#_Toc134652989)

[Minimax\_parallel.py 304](#_Toc134652990)

[Game Manager 324](#_Toc134652991)

[Database module: 339](#_Toc134652992)

[Create\_database.py 339](#_Toc134652993)

[Models.py 341](#_Toc134652994)

[handle\_games.py 343](#_Toc134652995)

[Schema Module: 344](#_Toc134652996)

[Socket\_shemas.py: 348](#_Toc134652997)

[Website Module: 353](#_Toc134652998)

[app.py 353](#_Toc134652999)

[Chess\_game.html: 367](#_Toc134653000)

[Main.js 371](#_Toc134653001)

[Demonstrating the end result 396](#_Toc134653002)

[Example Play 403](#_Toc134653003)

[Testing: 423](#_Toc134653004)

[Automated unit tests: 423](#_Toc134653005)

[Here are the main issues that were detected by the automated unit tests: 444](#_Toc134653006)

[Timed Minimax not working: 444](#_Toc134653007)

[Tests completed on final product: 447](#_Toc134653008)

[Knights cannot take pieces 453](#_Toc134653009)

[Pawns cannot move forward 2 places: 456](#_Toc134653010)

[Concurrent workers not caching 458](#_Toc134653011)

[Erroneous Results: 466](#_Toc134653012)

[Demonstration video of the program working: 471](#_Toc134653013)

[Stakeholder Feedback: 472](#_Toc134653014)

[GUI: 472](#_Toc134653015)

[Special Features: 472](#_Toc134653016)

[Reliability (after the fixes from my testing): 472](#_Toc134653017)

[Suggestions for improvement: 472](#_Toc134653018)

[Evaluation (just of prototype 3): 472](#_Toc134653019)

[Evaluation: Evaluative Testing: 474](#_Toc134653020)

[Post-Development Testing 474](#_Toc134653021)

[Usability Survey: 478](#_Toc134653022)

[Usability Survey Results 480](#_Toc134653023)

[Breakdown of success criteria (what has been achieved): 487](#_Toc134653024)

[Limitations and Maintenance: 491](#_Toc134653025)

[Limitations with the program itself include: 491](#_Toc134653026)

[As part of maintenance, certain issues may arise: 492](#_Toc134653027)

[Some of the key ways in which this chess program could be improved: 492](#_Toc134653028)

[Appendix: 493](#_Toc134653029)

[Sources: 493](#_Toc134653030)

[Code Listing: 494](#_Toc134653031)

[Prototype 1: 494](#_Toc134653032)

[directory structure: 494](#_Toc134653033)

[app.py 495](#_Toc134653034)

[logger.py 496](#_Toc134653035)

[game\_engine.py 498](#_Toc134653036)

[index.html 501](#_Toc134653037)

[main.js 503](#_Toc134653038)

[Style.css 510](#_Toc134653039)

[Prototype 2: Vue page 511](#_Toc134653040)

[ChessGame.vue 511](#_Toc134653041)

[external.js 516](#_Toc134653042)

[Prototype 2: Final Code 516](#_Toc134653043)

[Directory Structure: 516](#_Toc134653044)

[assorted.py 517](#_Toc134653045)

[vector.py 518](#_Toc134653046)

[pieces.py 519](#_Toc134653047)

[board\_state.py 530](#_Toc134653048)

[game.py 535](#_Toc134653049)

[console\_chess.py 539](#_Toc134653050)

[test\_vector.py 541](#_Toc134653051)

[test\_pieces.py 542](#_Toc134653052)

[test\_board\_state.py 544](#_Toc134653053)

[test\_minimax.py 549](#_Toc134653054)

[test\_data\vector\from\_square.yaml 559](#_Toc134653055)

[test\_data\vector\vector\_add.yaml 560](#_Toc134653056)

[test\_data\vector\vector\_in\_board.yaml 561](#_Toc134653057)

[test\_data\vector\vector\_multiply.yaml 562](#_Toc134653058)

[test\_data\vector\vector\_to\_square.yaml 562](#_Toc134653059)

[test\_data\pieces\test\_board\_populated.yaml 563](#_Toc134653060)

[test\_data\pieces\test\_empty\_board.yaml 565](#_Toc134653061)

[test\_data\board\_state\color\_in\_check.yaml 566](#_Toc134653062)

[test\_data\board\_state\game\_over.yaml 569](#_Toc134653063)

[test\_data\board\_state\generate\_all\_pieces.yaml 571](#_Toc134653064)

[test\_data\board\_state\generate\_legal\_moves.yaml 572](#_Toc134653065)

[test\_data\board\_state\generate\_pieces\_of\_color.yaml 574](#_Toc134653066)

[test\_data\board\_state\piece\_at\_vector.yaml 575](#_Toc134653067)

[Prototype 3: Final code 576](#_Toc134653068)

[Directory Structure 576](#_Toc134653069)

[\_\_init\_\_.py 579](#_Toc134653070)

[app.py 579](#_Toc134653071)

[assorted\\_\_init\_\_.py 580](#_Toc134653072)

[assorted\chess\_exceptions.py 580](#_Toc134653073)

[assorted\general.py 581](#_Toc134653074)

[assorted\safe\_hash.py 581](#_Toc134653075)

[chess\_functions\\_\_init\_\_.py 582](#_Toc134653076)

[chess\_functions\vector.py 582](#_Toc134653077)

[chess\_functions\pieces.py 584](#_Toc134653078)

[chess\_functions\board\_state.py 599](#_Toc134653079)

[chess\_functions\test\_fast\_vector.py 608](#_Toc134653080)

[chess\_functions\test\_fast\_pieces.py 609](#_Toc134653081)

[chess\_functions\test\_fast\_board\_state.py 611](#_Toc134653082)

[chess\_functions\test\_data\vector\from\_square.yaml 621](#_Toc134653083)

[chess\_functions\test\_data\vector\vector\_add.yaml 622](#_Toc134653084)

[chess\_functions\test\_data\vector\vector\_in\_board.yaml 622](#_Toc134653085)

[chess\_functions\test\_data\vector\vector\_multiply.yaml 623](#_Toc134653086)

[chess\_functions\test\_data\vector\vector\_to\_square.yaml 624](#_Toc134653087)

[chess\_functions\test\_data\pieces\test\_board\_populated.yaml 624](#_Toc134653088)

[chess\_functions\test\_data\pieces\test\_empty\_board.yaml 627](#_Toc134653089)

[chess\_functions\test\_data\board\_state\color\_in\_check.yaml 628](#_Toc134653090)

[chess\_functions\test\_data\board\_state\game\_over.yaml 631](#_Toc134653091)

[chess\_functions\test\_data\board\_state\generate\_all\_pieces.yaml 633](#_Toc134653092)

[chess\_functions\test\_data\board\_state\generate\_legal\_moves.yaml 633](#_Toc134653093)

[chess\_functions\test\_data\board\_state\generate\_pieces\_of\_color.yaml 636](#_Toc134653094)

[chess\_functions\test\_data\board\_state\piece\_at\_vector.yaml 637](#_Toc134653095)

[chess\_game\\_\_init\_\_.py 638](#_Toc134653096)

[chess\_game\console\_chess.py 638](#_Toc134653097)

[chess\_game\game\_web.py 641](#_Toc134653098)

[chess\_game\game.py 648](#_Toc134653099)

[chess\_game\test\_fast\_game.py 655](#_Toc134653100)

[database\\_\_init\_\_.py 659](#_Toc134653101)

[database\create\_database.py 659](#_Toc134653102)

[database\handle\_games.py 660](#_Toc134653103)

[database\models.py 662](#_Toc134653104)

[move\_engine\\_\_init\_\_.py 663](#_Toc134653105)

[move\_engine\cache\_managers.py 664](#_Toc134653106)

[move\_engine\minimax\_parallel.py 673](#_Toc134653107)

[move\_engine\minimax.py 692](#_Toc134653108)

[move\_engine\parallel\_minimax\_testing.py 703](#_Toc134653109)

[move\_engine\test\_slow\_timed\_minimax\_engine.py 709](#_Toc134653110)

[schemas\\_\_init\_\_.py 726](#_Toc134653111)

[schemas\cache\_item\_schema.py 726](#_Toc134653112)

[schemas\socket\_schemas.py 729](#_Toc134653113)

[website\\_\_init\_\_.py 732](#_Toc134653114)

[website\flask\_server.py 732](#_Toc134653115)

[website\templates\chess\_game.html 743](#_Toc134653116)

[website\static\main.js 747](#_Toc134653117)

[website\static\chess\_class.js 755](#_Toc134653118)

[website\static\vector.js 765](#_Toc134653119)

[website\static\initial\_game\_data.js 766](#_Toc134653120)

[website\static\style.css 766](#_Toc134653121)

[database\database.db 771](#_Toc134653122)

[Interview Transcript: Interview with Roger: 772](#_Toc134653123)