Metronome

Contents

[Feasibility Study 1](#_Toc123728023)

[Analysis 1](#_Toc123728024)

[Design 1](#_Toc123728025)

[Implementation 2](#_Toc123728026)

[Testing 2](#_Toc123728027)

[Installation 2](#_Toc123728028)

[Evaluation 2](#_Toc123728029)

[Maintenance 3](#_Toc123728030)

# Feasibility Study

I would like to make a metronome. A metronome is a constant sound that is used by musicians to stay in time.

I know that this problem is solvable because the problem can be solved in a finite number of steps. The main calculation is converting a BPM to the delay (in milliseconds) between each beep. This is computationally simple using a theoretical approach.

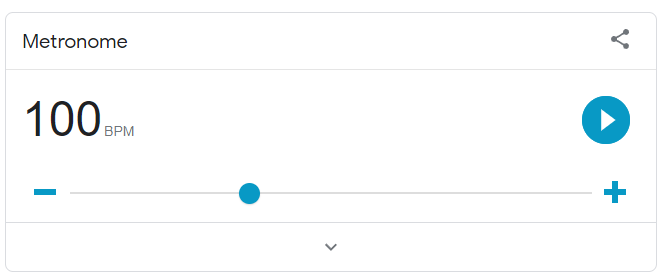
The budget is limited (£0) and we have 2 weeks to complete the project.

# Analysis

## Stakeholders

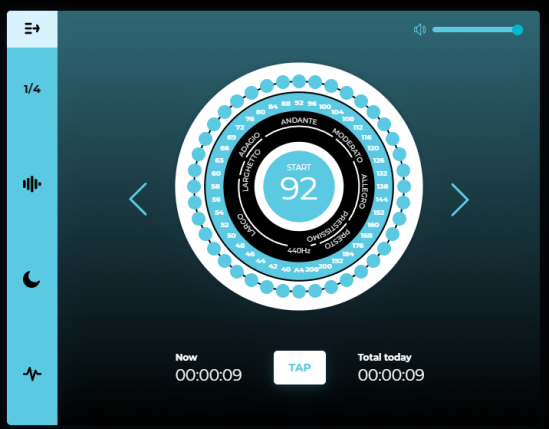
Matt is a 17-year-old boy who’s just started playing the piano and needs a metronome for staying in time.

## Research



<https://www.google.com/search?q=google+metronome&rlz=1C1GCEA_enGB813GB813&oq=google+metronome&aqs=chrome..69i57j0i512l9.2992j0j1&sourceid=chrome&ie=UTF-8&safe=active&ssui=on>

Google has a built-in online metronome. It’s free and easy to use. It has nice visual effects such as a pulsing button meaning that you don’t need to have audio enabled to use it. Although, it doesn’t indicate how far through each tick you are like a normal metronome.



<https://www.metronomeonline.com/>

Metronomeonline is a free online metronome that can be accessed on a device with Wi-Fi connectability. The design looks good but from afar the white on the blue can be quite hard to see. Also, there is no visual effect to represent the metronome meaning that people with a lack of sight won’t be able to use this metronome. One thing that is good is that it’s easy to choose what BPM you want since there a ring of buttons with different numbers are indicating the BPM that you want to choose.



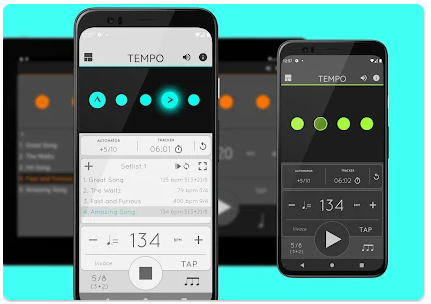
<https://metronome-beats.en.softonic.com/android>

Metronome Beats is an android app that can be accessed with an android device. Once downloaded, it doesn’t require internet access to use which makes it useful in situations where internet isn’t available which the case for the previous examples wasn’t. The design is simple and bold making it easier to see at a slight distance. It is also easy to choose the BPM you want due to the increase and decrease buttons where you can change how much you increase and decrease by. Also, it is suitable for colour blind people since the colour scheme is mostly from black to white (including scales of grey).



<https://kitgrenonguitartutor.com/3-reasons-to-use-a-metronome/>

A physical metronome is useful since it is clear, easy to see and has both a visual and audible representation to show the metronome. Although, it is not free since you have to buy the physical metronome but since it’s physical, it doesn’t require a device to access it and doesn’t require internet connection. It also has a slider which is used to change the BPM of the metronome which is easy to use.



<https://play.google.com/store/apps/details?id=com.frozenape.tempo&hl=en&gl=US>

Metronome: Tempo is a paid metronome for android devices which costs $1.49. The design is simple but effective with black and white but uses some bright colours to make certain aspects of the app stand out more. There is a visual and audible metronome meaning that people with hearing or sight loss can use this app. It doesn’t need internet access once downloaded and you can choose specific BPMs of different songs meaning that users don’t have to use their time searching it up themselves. Although, it requires an android device to be able to use the app. The BPM is also easy to change with increase and decrease buttons.

## Essential features

My solution must allow you to set the BPM and to start and stop the metronome. On each beat it should make a sound and should give a visual indication to assist for people who can’t hear the noise.

The solution must work on web enabled device with a clear, touch screen friendly interface.

There should be limited text so that it’ suitable for any age.

There should be buttons that allow you to change the BPM by 5 without stopping the beat.

It should be available online for free.

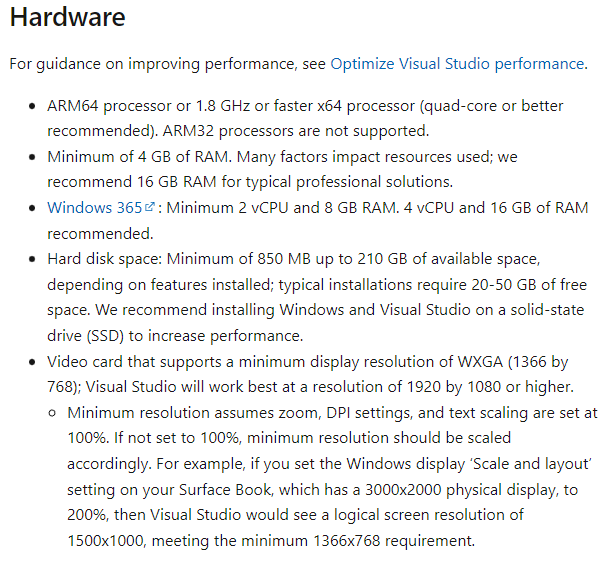
## Limitations

There is insufficient time to add additional features to the metronome but they could be added later if given extra time. My solution will not allow you to compare your playing to the beat. There will be no feature to log in and track your progress. Adding a feature to change the BPM automatically over time would be good but since it’s not essential, I won’t be able to add it within the time.

There is no requirement to make the metronome a native app since the metronome will be purely web based and freely available to as many people as possible.

## Hardware and Software requirements

For development, I need to be able to us Visual Studio 2017 which requires:



<https://learn.microsoft.com/en-us/visualstudio/releases/2022/system-requirements>

For the stakeholders to run the metronome they need a web available device running a modern browser (e.g. chrome)

This is so that all features of css / javascript / HTML5 work as expected.

My stakeholder has a Samsung phone

## Success Criteria

|  |  |  |
| --- | --- | --- |
| Number | Criteria | Justification |
| 1 | Must be accessible on an Android 12 device in Chrome connected to the internet | The stakeholder has an android 12 device with a reliable Wi-Fi connection. Any updates to the app can be automatically rolled out to the users |
| 2 | User can set the BPM | The stakeholder wants to practice between 50BPM and 120BPM |
| 3 | Any invalid BPM will be automatically corrected to the closest sensible value between 50-120BPM | The stakeholder wants to focus on playing the piano and might accidentally type invalid data |
| 4 | The user should be able to increase the BPM by 5 using a touch button | The stakeholder wants to be able to slightly change the BMP without having to start and stop the metronome |
| 5 | The user should be able to decrease the BPM by 5 using a touch button |
| 6 | There should be a start and stop button that toggles when you press it | The user should be able to control the metronome with their finger with a simple UI |
| 7 | When the metronome starts there should be an audible beep | The stakeholder wants to play along to a beep sound in headphones |
| 8 | When the metronome starts there should be a visual indication showing whenever it would ‘tick’ | Sometimes the metronome would be used with audio off |
| 9 | There should be a high contrast mode button which toggles a black and white colour scheme with larger text | Sometimes the metronome will be on a small screen far away from the musician so it will be hard to see |

# Design

## Algorithms

The user should be able to enter the BPM score (beats per minute). The browser needs to know how long to pause between each tick in milliseconds. This can be done using the following algorithm

BPM = user input

Interval = 1 \* 60 \* 1,000/BPM

## Usability features

50

+5

-5

The high contrast option should be all black and white with larger text so that I can be visible from a distance on small screens with a high DPI. It will also be usable by people who are partially sighted or colour blind.

The slider allows for the user to easily change the BPM without having to press a button multiple times although there is still the option to + or - 5 so it’s easy to change the BPM by a small amount.

# Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test number** | **Description** | **Success Criteria** | **Test data** | **Expected result** |
| 1 | Basic web page | 1 | Page load | A web page with a title and text box to enter the BPM score ad two buttons to increase / decrease the BPM score displays in chrome |
| 2 | BPM | 2 | 120 | Valid data is accepted |
| 3 | BPM | 3 | “” | BPM to be set to 50 |
| 4 | BPM | 3 | 30 | BPM to be set to 50 |
| 5 | BPM | 3 | 130 | BPM set to 120 |
| 6 | BPM | 3 | Potato | BPM set to 50 |
| 7 | Increasing BPM | 4 | BPM set to 60 and press increase once | BPM increases to 65 |
| 8 | Increasing BPM | 4 | BPM set to 120 and press increase once | BPM stays at 120 |
| 9 | Increase BPM | 4 | BPM set to 119 and press increase once | BPM set to 120 |
| 10 | Decrease BPM | 5 | BPM set to 100 and press decrease once | BPM set to 95 |
| 11 | Decrease BPM | 5 | BPM set to 50 and press decrease once | BPM stays at 50 |
| 12 | Decrease BPM | 5 | BPM set to 51 and press decrease once | BPM set to 50 |
| 13 | Start button | 6 | Press play button | Metronome should begin to play |
| 14 | Start button change | 6 | Press the play button | Play button should change to a stop button when pressed |
| 14 | Stop button | 6 | Press stop button | Metronome should stop playing |
| 15 | Audible sound | 7 | Press the play button when BPM set to 60 | Sound should be played every second |
| 16 | Audible sound | 7 | Press play button when BPM set to 120 | Sound should be player every two seconds |
| 17 | Visual indication | 8 | Press the play button when BPM set to 60 | Visual indication should be displayed every second |
| 18 | Visual indication | 8 | Press play button when BPM set to 120 | Visual indication should be displayed every two seconds |

## Variables and Validation

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable** | Data type | Validation | Justification |
| BPM | Integer | Range check (between 50 - 120) | The best per minute should b anything from just less than one to two beats per second |
| BPMUserInput | String | Presence Check and TypeCheck | The BPM should be able to b converted to an integer between 50 and 120 BPM |
| IntervalMS | Integer | Make sure it’s rounded to 0 dp | The browser needs an interval between ‘ticks’ in milliseconds as a whole number |
| HighContrastMode | Boolean | Checkbox can be ticked or not ticked | Checkbox can be ticked or not ticked |
| IsPlaying | Boolen | Either true or false | The metronome can either be playing or paused. |

## Post development testing

Here are some questions to give my stakeholders after I’ve finished developing the solution:

Functionality

What did you want the program to do? What did you click on and in what order? What happened? Did the program do everything you wanted it to do? Were there any features you believed it was missing?

Robustness

Did the program crash? If so, when and what were you doing?

Usability

Was the high contrast mode good?

# Implementation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test number** | **Description** | **Success Criteria** | **Test data** | **Expected result** | **Actual result** |
| 1 | Basic web page | 1 | Page load | A web page with a title and text box to enter the BPM score ad two buttons to increase / decrease the BPM score displays in chrome |  |
| 2 | BPM | 2 | 120 | Valid data is accepted |  |
| 3 | BPM | 3 | “” | BPM to be set to 50 |  |
| 4 | BPM | 3 | 30 | BPM to be set to 50 |  |
| 5 | BPM | 3 | 130 | BPM set to 120 |  |
| 6 | BPM | 3 | Potato | BPM set to 50 |  |
| 7 | Increasing BPM | 4 | BPM set to 60 and press increase once | BPM increases to 65 |  |
| 8 | Increasing BPM | 4 | BPM set to 120 and press increase once | BPM stays at 120 |  |
| 9 | Increase BPM | 4 | BPM set to 119 and press increase once | BPM set to 120 |  |
| 10 | Decrease BPM | 5 | BPM set to 100 and press decrease once | BPM set to 95 |  |
| 11 | Decrease BPM | 5 | BPM set to 50 and press decrease once | BPM stays at 50 |  |
| 12 | Decrease BPM | 5 | BPM set to 51 and press decrease once | BPM set to 50 |  |
| 13 | Start button | 6 | Press play button | Metronome should begin to play |  |
| 14 | Start button change | 6 | Press the play button | Play button should change to a stop button when pressed |  |
| 14 | Stop button | 6 | Press stop button | Metronome should stop playing |  |
| 15 | Audible sound | 7 | Press the play button when BPM set to 60 | Sound should be played every second |  |
| 16 | Audible sound | 7 | Press play button when BPM set to 120 | Sound should be player every two seconds |  |
| 17 | Visual indication | 8 | Press the play button when BPM set to 60 | Visual indication should be displayed every second |  |
| 18 | Visual indication | 8 | Press play button when BPM set to 120 | Visual indication should be displayed every two seconds |  |

# Installation

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.

Nunc viverra imperdiet enim. Fusce est. Vivamus a tellus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin pharetra nonummy pede. Mauris et orci.

Aenean nec lorem. In porttitor. Donec laoreet nonummy augue.

Suspendisse dui purus, scelerisque at, vulputate vitae, pretium mattis, nunc. Mauris eget neque at sem venenatis eleifend. Ut nonummy.

# Evaluation

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.

Nunc viverra imperdiet enim. Fusce est. Vivamus a tellus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin pharetra nonummy pede. Mauris et orci.

Aenean nec lorem. In porttitor. Donec laoreet nonummy augue.

Suspendisse dui purus, scelerisque at, vulputate vitae, pretium mattis, nunc. Mauris eget neque at sem venenatis eleifend. Ut nonummy.

# Maintenance

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.

Nunc viverra imperdiet enim. Fusce est. Vivamus a tellus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin pharetra nonummy pede. Mauris et orci.

Aenean nec lorem. In porttitor. Donec laoreet nonummy augue.

Suspendisse dui purus, scelerisque at, vulputate vitae, pretium mattis, nunc. Mauris eget neque at sem venenatis eleifend. Ut nonummy.