

Building & distributing Linux desktop apps

Using Electron framework



Pradeep Sharma



Mindful Developer



@pradeep_io



gitcommitshow

pradeep-io.medium.com

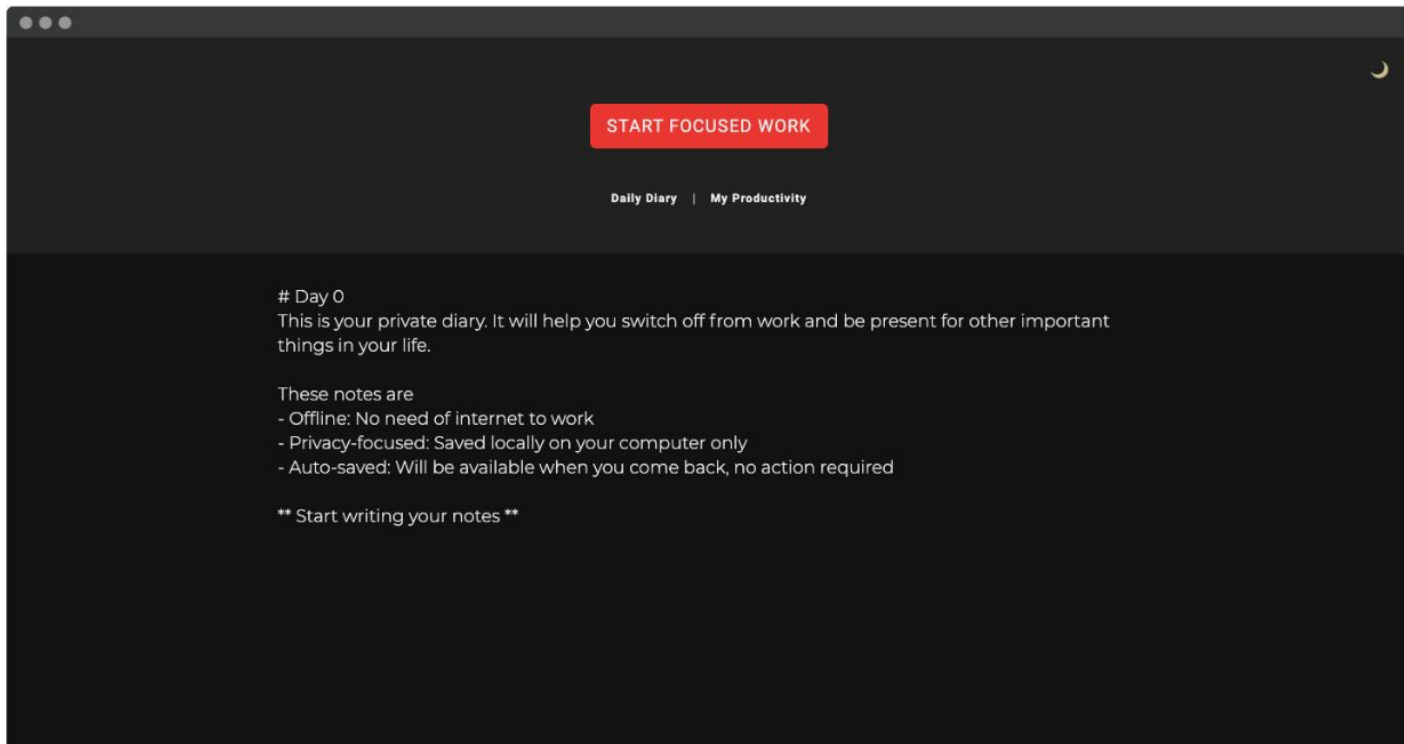
About me

- **Creator of Developer Diary**
- **Founded Invide**
Invite-only community of remote developers
- **Produced Git Commit Show**
- Open-Source Contributor and Trainer
- Chess Lover



Distraction-Free Diary

Get things out of your head to focus on the present moment



Have you used Electron before?

Yes/No



Which Linux distro have you used in last 2 years?

Debian

Fedora

ArchLinux

OpenSuse

Other

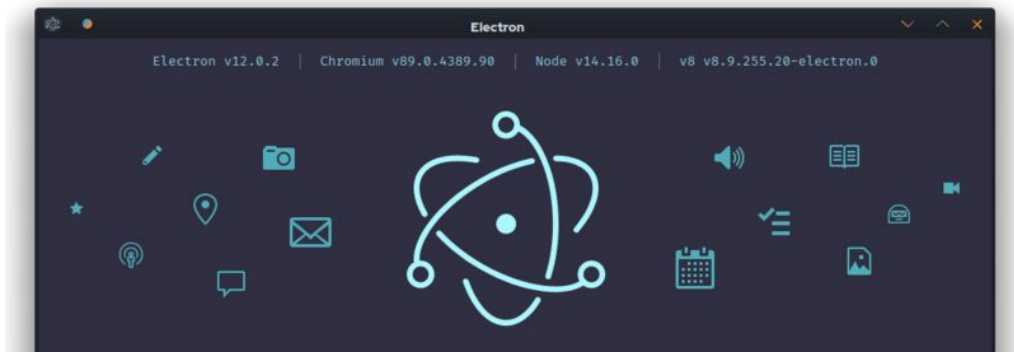


Introduction to Electron



Electron

A framework to build cross-platform desktop apps

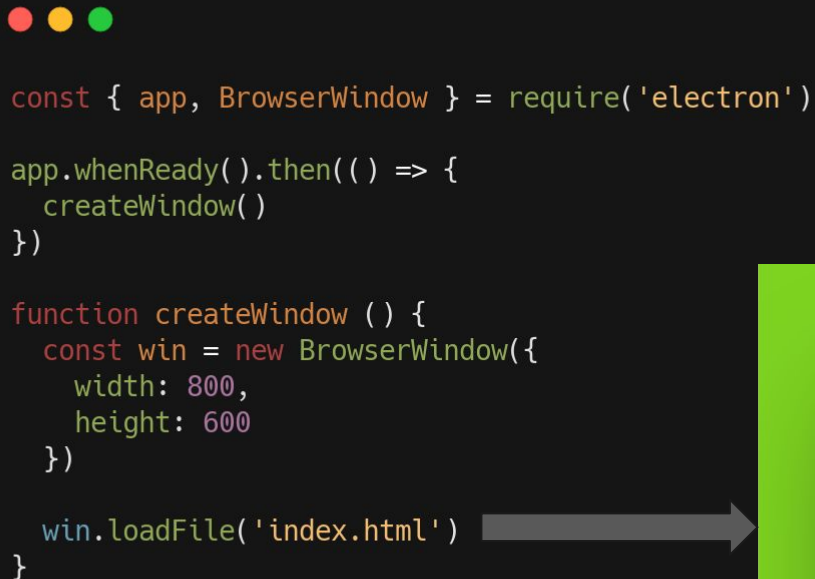


Chromium rendering engine + Node.js runtime



Code using web technologies

Hello world



```
const { app, BrowserWindow } = require('electron')

app.whenReady().then(() => {
  createWindow()
})

function createWindow () {
  const win = new BrowserWindow({
    width: 800,
    height: 600
  })

  win.loadFile('index.html')
}
```



```
<html>
<head>
  <title>Hello World!</title>
</head>
<body style="background: white;">
  <h1>Hello World!</h1>
</body>
</html>
```

Packaging your desktop app



Packaging the app



electron-builder

- Supported package targets
AppImage, snap, deb, rpm, freebsd, pacman, p5p, apk, 7z, zip, tar.xz, tar.lz, tar.gz, tar.bz2, dir
- Build package for all distributions on one platform

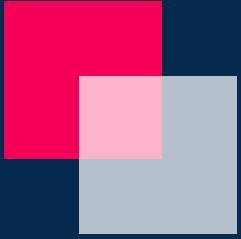
```
// package.json
{
  "devDependencies": {
    "electron-builder": "^22.9.1",
    ...
  }
  "build": {
    "asar": true,
    "appId": "com.invidi Labs.diary",
    "linux": {
      "target": 'deb'
      "icon": "build/icons",
      "category": "Productivity",
      ...
    }
  }
}
```



electron-builder

Distributing your app



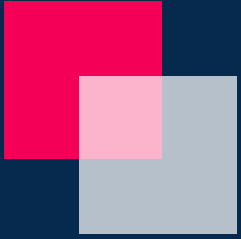


Direct distribution

Auto updates

```
autoUpdater.on('update-downloaded', (event, releaseNotes, releaseName) => {  
  const downloadAlertOptions = {  
    type: 'info',  
    buttons: ['Restart', 'Later'],  
    title: 'Application Update',  
    detail: 'New version downloaded. Apply updates now?'  
  }  
  
  dialog.showMessageBox(downloadAlertOptions).then((returnValue) => {  
    if (returnValue.response === 0) autoUpdater.quitAndInstall()  
  })  
})
```

- Only for AppImage
- Needs a release server
 - [Electron-release-server](#)
 - Hazel
 - Nuts
 - Public Github releases



External distribution

Classic Distribution

- deb
- rpm
- pacman
- freebsd

Universal distribution

- AppImage
- Snap
- Flatpak

Classic distribution

deb



```
dpkg -i DeveloperDiary.deb
```

Distribute via your own repository

```
sudo add-apt-repository your.repo  
apt install developerdiary
```

Distribute via Debian Package Repository

License restriction: Free, allows to modify and distribute

```
apt install developerdiary
```


rpm



```
rpm -i DeveloperDiary.rpm
```

[Distribute via yum](#)

```
Yum install developerdiary
```

pacman



```
pacman -S DeveloperDiary
```

freebsd



```
pkg install developerdiary
```

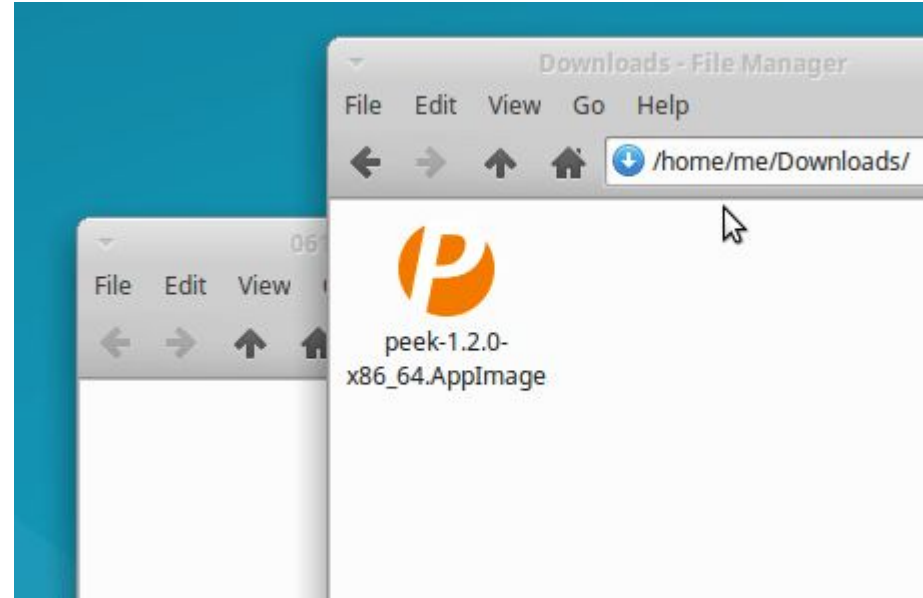
Universal distribution

ApplImage



Just click to run

- No desktop integration (gnome)
- No auto-start(gnome)
- Security
- No extra step to publish
- Supports auto-update out of the box



Snapcraft



```
snap install developerdiary
```

- Sandboxing
- Delta updates
- Client is OSS, server is not
- [Effort to publish is low](#)

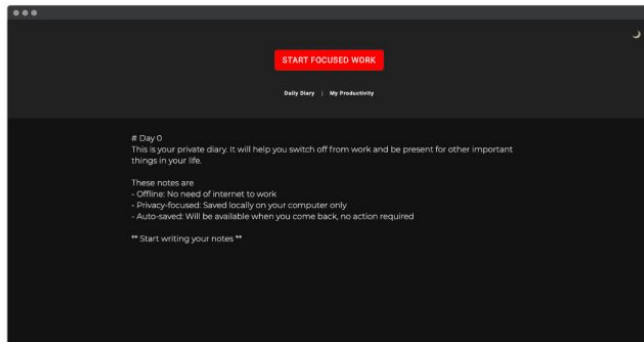


Developer Diary

Invide Labs (invide) • [Productivity](#) • [Utilities](#)

Distraction-Free Diary

Get things out of your head to focus on the present moment



Developer Diary

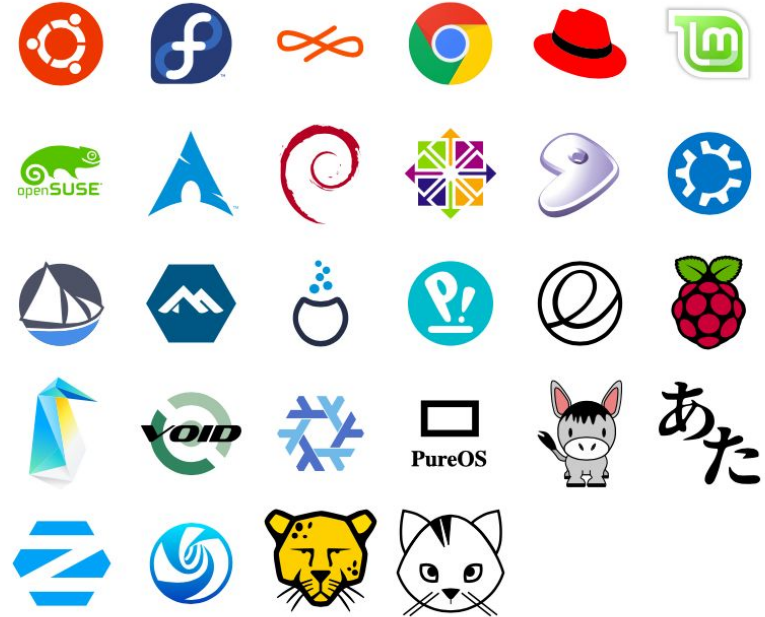
A minimal diary for developers available right in time with shortcut `Ctrl+Shift+I`

Flatpak



```
flatpak install flathub  
org.invidielabs.DEVELOPERDIARY
```

- Sandboxing
- Delta updates
- Effort to publish is non-trivial





Comparison of universal distribution channels

Application Distribution

Feature	ApplImage	Snap	Flatpak
Central Repo / Directory	ApplImageHub	Snap Store	FlatHub
Fully decentralized without central gatekeepers	✓ Yes	✗ No (one dominant app store) [1]	✓ Yes
Individual App Repositories	✗ No (not stored in repositories)	✗ No (you can only have one repo per device)	✓ Yes
Can have multiple versions in parallel (including historical ones)	✓ Yes (unlimited number of arbitrary versions)	✓ Yes (one per channel)	✓ Yes (any version available in OSTree can be installed in parallel)
Once the application is installed, it still can be easily copied to another machine (e.g., share with a friend locally)	✓ Yes (one app=one file; there is no "installation" so the app is in the same form at all times)	✓ Yes (but also need to copy snaps it depends on)	✓ Yes (you can use <code>flatpak create-usb</code> to copy to USB drive)

Application Updates

Feature	ApplImage	Snap	Flatpak
Update mechanism	ApplImageUpdate	From Repo	From Repo
Binary delta updates	✓ Yes (using zsync with no need to generate deltas in advance)	✓ Yes (Only if using a private server-side service that needs to generate the deltas)	✓ Yes (using OSTree to provide atomic updates)
Applications can be self-updating	✓ Yes (using embedded information)	✓ Yes	✓ Yes

Linux Distribution Support

Feature	ApplImage	Snap	Flatpak
Earliest Ubuntu Supported	Ubuntu 10.04	Ubuntu 14.04	Ubuntu 16.04
Earliest OpenSUSE Supported	OpenSUSE 11.3	Leap 42.2	Leap 42.1
Earliest Fedora Supported	Fedora 12	Fedora 24	Fedora 23
Earliest Debian Supported	Debian 6	Debian 9	Debian 9
Earliest CentOS Supported	CentOS 6	CentOS 7.6	CentOS 7
Runs on Ubuntu out-of-the-box	✓ Yes	✓ Yes	✗ No
Runs on OpenSUSE out-of-the-box	✓ Yes	✗ No	tbc
Runs on Fedora out-of-the-box	✓ Yes	✗ No	✓ Yes
Runs on Debian out-of-the-box	✓ Yes	✗ No	tbc
Runs on CentOS out-of-the-box	✓ Yes	✗ No	✓ Yes
Live systems (e.g., Live ISO, Live USB, Live CD, Live network boot)	✓ Full	⚠ Partial (starting with 18.04, but it is limited by a kernel limitation and "a pain to work with, we spend almost zero time with that" according to a Canonical developer)	⚠ Partial (Session must be restarted for exports to be picked up)
Can run on Chrome OS (Crostini)	✓ Yes (Chrome OS 73)	✓ Yes (Chrome OS 73)	✓ Yes

[Source](#)

Maintain your app



Maintenance

- Upgrade electron and node versions every quarter
- Tests before packaging
- Use typescript
- Managing different versions
 - How will you test?
 - Can you test all the channels on the same system?
 - Different product names
 - Test server for test channels
 - Flag for client version

Analyze and improve



Analytics

- Linux users expect privacy at the core
- Do not use external analytics services such as Google Analytics
- Anonymous analytics using update request logs
 - # of users
 - # of returning users
 - Avg. time spent



```
calcDownloads() {  
    matchPeriodL=$1  
    cat ${sourceFile} | awk '$0~matchPeriod && /GET \/download\/*(\.exe)|(\.deb)|(\.dmg)/ {print $1}'  
    matchPeriod="${matchPeriodL}" | sort -u | wc -l  
}
```



Pradeep Sharma



Mindful Developer



@pradeep_io



gitcommitshow

pradeep-io.medium.com