

```

connect 169.150.247.194:26279
{
  try {
    const filepath: string = './uploads/converted.pdf'
    const wordFile: any = req.file;
    //const wordFile: any = req.file?.buffer.toString();
    console.log(wordFile)
    const pdfStream: any = fs.createWriteStream(filepath);
    const result = await mammoth.extractRawText({ arrayBuffer: wordFile });
    const html = result.value;
    const doc: any = new pdf();
    doc.pipe(pdfStream);
    doc.text(html);
    doc.end();
    pdfStream.on('finish', () => {
      res.download(filepath, 'converted.pdf', (err: Error) => {
        if (err) {
          console.error(err);
          res.status(500).send('Server Error');
        } else {
          console.log("success");
          // fs.unlinkSync(filepath);
        }
      });
    });
  } catch (err) {
    res.status(500).send(err);
  }
}
app.post("/word2pdf", upload.single('wordFile'), async (req: Request, res: Response) => {
  const filepath: string = './uploads/converted.pdf'
  const wordData: any = req.file?.buffer;
  console.log(wordData);
  const pdfStream: any = fs.createWriteStream(filepath);

```

```

    }
    try {
    }
    {
    }
    {
    const doc: any = new pdf();
    doc.pipe(pdfStream);
    doc.text();
    doc.end();
    pdfStream.on('finish', () => {
        res.download(filepath, 'converted.pdf', (err) => {
            if (err) {
                console.error(err);
                res.status(500).send('Server Error');
            } else {
                console.log("success");
            }
        });
    });
} catch (err) {
    console.error(err);
    res.status(500).send('Conversion Error');
}
});
<GoogleLogin text='signin_with' shape='pill' logo_alignment='left'
    onSuccess={credentialResponse => {
    }
    if (credentialResponse.credential) {
        console.log(credentialResponse)
        var decodedCredential = jwtDecode(credentialResponse.credential);
        console.log(decodedCredential)
        navigate(`${decodedCredential.sub}`, {state: decodedCredential})
    } else {
        console.error("Credential is undefined");
    }
    }
    onError={() => {
        console.log('Login Failed');
    }}

```

/>