Proposed Solution Template:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Detect the probability of occurrence of Parkinson's disease in patients when no observable symptoms are present.
2.	Idea / Solution description	Make the patient use pencil and paper to draw spirals and waves. Take a photo of each spiral and wave, and upload it to the server via an app, by connecting to the internet. The app will display the probability of Parkinson's disease by processing the images using Machine Learning.
3.	Novelty / Uniqueness	Compared to traditional methods of diagnosis of Parkinson's, this method can be applied easily at home as a pre-diagnosis; since ML model is tuned to minimize false positives, unnecessary visits to the hospital can be avoided. By performing this simple test regularly (every three months or so), onset of Parkinson's can be detected in an early stage, so that physiotherapy can be organised to help patients cope with the disease.
4.	Social Impact / Customer Satisfaction	Will be particularly useful in old age homes and medical centres for the elderly, wherein a community of elderly people can be monitored without regular hospital visits. This app can be integrated with other health monitoring systems to provide a holistic regular check-up without frequent visits to the hospital.
5.	Business Model (Revenue Model)	Since the product consists mainly of an app making use of IBM cloud services, a one-time fee can be charged for lifetime access to the app. Advertisement-based and subscription-based revenue models were also considered, but were deemed unsuitable due to the fact that the app will not be used often, on a monthly basis at most. Integration with a more holistic health-monitoring system can prove to be more profiting in terms of revenue generated.
6.	Scalability of the Solution	The solution is scalable since the cloud resource requirement will not increase drastically with increasing users. Previously uploaded images by users can be deleted, since the ML model is not dynamically trained and works on a single image at a time. Computational resources required will also not increase too much, since app will be used very infrequently, and computation can be scheduled easily.