Model Development Phase Template

Date	18 June 2025
Team ID	SWTID1749709635
Project Title	Mental Health Prediction
Maximum Marks	4 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

Initial Model Training Code:

```python
88    # Prepare data
89    X, y, label_encoders, scaler, le_target = preprocess_data(df)
90    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
91
92    # Define models
93    all_models = {
94        'Logistic Regression': LogisticRegression(max_iter=1000, random_state=42),
95        'KNN': KNeighborsClassifier(),
96        'Decision Tree': DecisionTreeClassifier(random_state=42),
97        'Random Forest': RandomForestClassifier(random_state=42),
98        'Naive Bayes': GaussianNB(),
99        'SVM': SVC(probability=True, random_state=42),
100       'XGBoost': XGBClassifier(use_label_encoder=False, eval_metric='logloss', random_state=42),
101       'AdaBoost': AdaBoostClassifier(random_state=42),
102       'Gradient Boosting': GradientBoostingClassifier(random_state=42)
103   }
104
105
106   best_model = None
107   best_score = 0
108   best_model_name = ''
```

```
107    best_score = 0
108    best_model_name = ''
109
110    print("\n=== Training and Evaluation of All Models ===")
111    for name, model in all_models.items():
112        print(f"\n--- {name} ---")
113        model.fit(X_train, y_train)
114        y_pred = model.predict(X_test)
115
116        acc = accuracy_score(y_test, y_pred)
117        print(f"Accuracy: {acc:.4f}")
118        print("Classification Report:\n", classification_report(y_test, y_pred, zero_division=0))
119        print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
120
121        if acc > best_score:
122            best_score = acc
123            best_model = model
124            best_model_name = name
125
126    print(f"\nBest Model: {best_model_name} with Accuracy: {best_score:.4f}")
127    print("\n=== Final Model Evaluation ===")
128    y_pred = best_model.predict(X_test)
129    print("Accuracy:", accuracy_score(y_test, y_pred))
130    print("Classification Report:\n", classification_report(y_test, y_pred))
131    print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
132
```

Model Validation and Evaluation Report:

Model	Classification Report	Accuracy	Confusion Matrix

Logistic Regression	--- Logistic Regression ---
	Accuracy: 0.7559
	Classification Report:
	        precision recall f1-score support
	
	    0 0.76 0.69 0.73 118
	    1 0.75 0.81 0.78 136
	
	  accuracy 0.76 254
	 macro avg 0.76 0.75 0.75 254
	weighted avg 0.76 0.76 0.75 254
		0.7559	Confusion Matrix:
			[[82 36]
			 [26 110]]

K-Nearest Neighbors	Classification Report:
	        precision recall f1-score support
	
	    0 0.66 0.81 0.73 118
	    1 0.79 0.64 0.71 136
	
	  accuracy 0.72 254
	 macro avg 0.73 0.72 0.72 254
	weighted avg 0.73 0.72 0.72 254
		0.7165	Confusion Matrix:
			[[95 23]
			 [49 87]]

Model	Classification Report	Accuracy	Confusion Matrix

Decision Tree	Classification Report:
	
	       precision recall f1-score support
	
	   0 0.79 0.92 0.85 118
	   1 0.92 0.79 0.85 136
	
	  accuracy 0.85 254
	 macro avg 0.86 0.86 0.85 254
	weighted avg 0.86 0.85 0.85 254
		0.8504	Confusion Matrix:
			[[109 9]
			[29 107]]

Random Forest	Classification Report:
	
	       precision recall f1-score support
	
	   0 0.91 0.92 0.92 118
	   1 0.93 0.92 0.93 136
	
	  accuracy 0.92 254
	 macro avg 0.92 0.92 0.92 254
	weighted avg 0.92 0.92 0.92 254
		0.9213	Confusion Matrix:
			[[109 9]
			[11 125]]

Naïve Bayes	Classification Report:
	
	       precision recall f1-score support
	
	   0 0.71 0.68 0.69 118
	   1 0.73 0.76 0.74 136
	
	  accuracy 0.72 254
	 macro avg 0.72 0.72 0.72 254
	weighted avg 0.72 0.72 0.72 254
		0.7205	Confusion Matrix:
			[[80 38]
			[33 103]]

Support Vector Machines	Classification Report:
	
	       precision recall f1-score support
	
	   0 0.78 0.73 0.75 118
	   1 0.78 0.82 0.80 136
	
	  accuracy 0.78 254
	 macro avg 0.78 0.78 0.78 254
	weighted avg 0.78 0.78 0.78 254
		0.7795	Confusion Matrix:
			[[86 32]
			[24 112]]

Extreme Gradient Boosting	Classification Report:
	
	       precision recall f1-score support
	
	   0 0.81 0.93 0.87 118
	   1 0.93 0.82 0.87 136
	
	  accuracy 0.87 254
	 macro avg 0.87 0.87 0.87 254
	weighted avg 0.88 0.87 0.87 254
		0.8701	Confusion Matrix:
			[[110 8]
			[25 111]]

Adaptive Boosting	Classification Report:
	
	``
	Classification Report:
	              precision recall f1-score support
	
	           0 0.76 0.67 0.71 118
	           1 0.74 0.82 0.78 136
	
	    accuracy 0.75 254
	   macro avg 0.75 0.74 0.74 254
	weighted avg 0.75 0.75 0.75 254
	``
		0.7480	```
			Confusion Matrix:
			[[79 39]
			 [25 111]]
			``

Gradient Boosting	```
	Classification Report:
	              precision recall f1-score support
	
	           0 0.83 0.82 0.83 118
	           1 0.85 0.85 0.85 136
	
	    accuracy 0.84 254
	   macro avg 0.84 0.84 0.84 254
	weighted avg 0.84 0.84 0.84 254
	``
		0.8386	```
			Confusion Matrix:
			[[97 21]
			 [20 116]]
			``